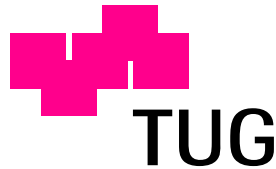


Christian Feldbauer

# Sparse Pulsed Auditory Representations For Speech and Audio Coding

Dissertation

vorgelegt an der  
Technischen Universität Graz



zur Erlangung des akademischen Grades  
*Doktor der Technischen Wissenschaften*

durchgeführt am  
Institut für Signalverarbeitung und Sprachkommunikation

September 2005



*To the memory of my mother.*



# Acknowledgments

I would like to thank my supervisors *Gernot Kubin* and *Bastiaan Kleijn* for their primary work on ‘coding in the perceptual domain,’ which builds the basis of this thesis, for their helpful ideas, support, and good advice; Bastiaan Kleijn also for his frame-theoretical contributions and the numerous suggestions to improve my writing style; *Jan Plasberg* for the fruitful discussions; *Stefan Münkner* for the useful hints concerning the implementation of the adaptation loops; *Birgit* for accompanying me all the years, for her tireless support, and the many hours she spent on reading and correcting my stuff; Matt Groening for creating ‘The Simpsons;’ ‘Strapping Young Lad’ for their great, aggressive and yet melodic music.



# Kurzfassung

Gehörmodellierung ist eine gut bekannte Methode, die Einblicke in die menschliche Wahrnehmung gewährt und für Kodierungsanwendungen die Extrahierung von jenen Signalmerkmalen ermöglicht, die für einen menschlichen Zuhörer am wichtigsten sind. Diese Dissertation beschäftigt sich mit dem Ansatz der ‘Kodierung in der perceptiven Domäne’ und hat ein invertierbares Gehörmodell als Grundlage, das eine gepulste, gehörbezogene Signaldarstellung des Sprach- oder Audiosignales liefert. Es ist üblich, nur die Signalwerte ungleich Null von gepulsten Signaldarstellungen zu kodieren und die Positionen dieser Werte als Seiteninformation anzugeben. Bei der hier untersuchten gehörbezogenen Signaldarstellung ist die Anzahl von Pulsen und somit die Menge an Seiteninformation zu groß, um eine effiziente Kodierung mit geringer Bitrate zu erreichen.

Der Schwerpunkt dieser Arbeit ist das ‘Ausdünnen’ der gepulsten Signaldarstellung, d.h. das Entfernen von perzeptueller Irrelevanz und Redundanz, um eine kompakte Signaldarstellung zu erhalten, die einerseits eine effiziente Kodierung ermöglicht und andererseits die Rekonstruktion des Signales mit perzeptuell transparenter Qualität zulässt. Zu diesem Zweck schlagen wir die ‘Transmultiplexer’-Betrachtungsweise von Wahrnehmungsdomänen-Kodierung vor, welche zu einem neuen Maskierungsmodell führt. Dieses Maskierungsmodell wird erfolgreich angewendet, um eine kompakte, gepulste Signaldarstellung zu erhalten. Die Experimente zeigen, dass diese vorgeschlagene Signaldarstellung einen bemerkenswert hohen Rekonstruktionsfehler maskieren kann. Wir diskutieren Ansätze zur effizienten Kodierung von dünnbesetzten, gepulsten Signaldarstellungen. Weiters beschäftigen wir uns mit rechenaufwandseffizienten Implementierungsmethoden für gehörbezogene Filterbänke, die Schlüsselkomponenten in praktisch allen Gehörmodellen darstellen.



# Abstract

Auditory modeling is a well-established methodology that provides insight into human perception and that facilitates the extraction of signal features most relevant to the human listener for coding applications. This thesis deals with the approach of ‘coding in the perceptual domain’ and is based on an invertible auditory model that provides a pulsed auditory representation of the input speech or audio signal. It is natural for pulsed signal representations to encode only the non-zero samples by specifying their positions as side information. For the considered auditory representation, the number of pulses and, therefore, the amount of side information is too high for an efficient encoding at a relatively low bit rate.

The focus of this work is to ‘sparsify’ the pulsed signal representation, i.e., to remove its perceptual irrelevance and its redundancy, to obtain a compact signal representation, which facilitates efficient encoding and from which the signal can nevertheless be reconstructed with perceptually transparent quality. For this purpose, the ‘transmultiplexer view’ of perceptual-domain coding is proposed, which leads to a new masking model. This masking model is successfully applied to obtain a sparse pulsed signal representation. Experiments show that the proposed sparse signal representation is able to hide a remarkable amount of reconstruction errors. We discuss approaches to efficiently encode sparse pulsed signal representations. We also deal with computationally efficient implementation methods for auditory filterbanks, which are key components of virtually all auditory models.



# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Motivation . . . . .	1
1.2	Outline of the Thesis . . . . .	5
1.3	Nomenclature . . . . .	7
1.3.1	Symbols and Mathematical Notation . . . . .	7
1.3.2	Abbreviations . . . . .	9
<b>2</b>	<b>Coding in the Perceptual Domain</b>	<b>11</b>
2.1	A Physiologically Motivated Subband Coder . . . . .	12
2.2	Auditory Analysis . . . . .	14
2.2.1	Basilar Membrane Filterbank . . . . .	14
2.2.2	Inner-Hair-Cell Model . . . . .	20
2.2.3	Neuron Model . . . . .	21
2.3	Auditory Synthesis . . . . .	23
2.3.1	Inversion of Neuron and Inner-Hair-Cell Models . . . . .	24
2.3.2	Synthesis Filterbank . . . . .	27
2.3.3	Simulation Results . . . . .	29
2.3.4	Frame-Theoretic Interpretation of Auditory Synthesis . . . . .	30
<b>3</b>	<b>Auditory-Domain Sparsification</b>	<b>35</b>
3.1	How sparse can we make the auditory representation for coding? . . . . .	35
3.2	The Transmultiplexer View of Perceptual-Domain Coding . . . . .	37
3.2.1	Matched-Filter Communication Scenario . . . . .	38
3.2.2	The Role of Dominant Pulses . . . . .	40
3.3	A New Model for Simultaneous and Temporal Masking . . . . .	43
3.3.1	Auditory Excitation Pattern Model . . . . .	43
3.3.2	Is exhaustive search feasible? . . . . .	44
3.3.3	Isolated-Pulse BM Excitation Patterns . . . . .	44
3.3.4	Decision for Pulse Deletion . . . . .	46
3.4	Pulse Amplitude Correction . . . . .	49
3.4.1	Pulse Distance Based Weighting . . . . .	50
3.4.2	BM Excitation Based Correction Factor . . . . .	51

3.4.3	The New Coder Structure . . . . .	53
3.5	Experimental Results . . . . .	53
3.5.1	Sparsification Capability . . . . .	55
3.5.2	Quality of the New Masking Model . . . . .	63
3.6	Consideration of Nonlinear Effects of Perception . . . . .	84
3.7	Conclusions . . . . .	90
<b>4</b>	<b>Auditory-Domain Quantization: First Attempts</b>	<b>93</b>
4.1	The Encoding of Sparse Signals . . . . .	93
4.2	Pulse Positions . . . . .	94
4.2.1	Lossless Encoding . . . . .	94
4.2.2	Lossy Encoding . . . . .	97
4.2.3	Discussion . . . . .	99
4.3	Pulse Amplitudes . . . . .	101
4.4	Vector Quantization . . . . .	104
4.4.1	Distance/Similarity Measure for Sparse Vectors . . . . .	105
4.4.2	Variable Frame Length Vector Quantization . . . . .	109
4.5	Conclusions . . . . .	113
<b>5</b>	<b>Alternative Filterbank Implementation Methods</b>	<b>115</b>
5.1	IIR Filterbank . . . . .	116
5.2	Frequency-Warped Transform Filterbank . . . . .	117
5.2.1	Principle . . . . .	117
5.2.2	Design Recipe for an Auditory Filterbank . . . . .	120
5.2.3	Filterbank Inversion . . . . .	128
<b>6</b>	<b>Summary and General Conclusions</b>	<b>131</b>
<b>A</b>	<b>Outer and Middle Ear (OME) Weighting</b>	<b>133</b>
<b>B</b>	<b>Modeling Temporal Adaptation</b>	<b>135</b>
<b>C</b>	<b>Critically Sampled Frequency-Warped PR Filterbank</b>	<b>139</b>
C.1	Exact Solution for the Synthesis Filters . . . . .	140
C.1.1	Existence and Uniqueness . . . . .	141
C.1.2	FIR Property . . . . .	142
C.1.3	Causality and Delay . . . . .	143
C.2	Oversampled Filterbanks . . . . .	143
C.3	Discussion and Experiments . . . . .	143
C.4	Conclusions . . . . .	147
	<b>Bibliography</b>	<b>149</b>

# Chapter 1

## Introduction

### 1.1 Motivation

The encoding of an analog signal such as speech or music for digital transmission at a finite rate requires quantization and introduces distortion [1, 2, 3]. Models of the human auditory system can be exploited to minimize the audible distortion (as quantified by the auditory model) for a given rate [2, 4]. The rate can be specified either as an average or as a fixed rate. Signal features are then specified with a precision that reflects audible distortion. The incorporation of auditory models into coding schemes leads to so-called *perceptual coders*. In Fig. 1.1 the general principle of perceptual coders is shown as a block diagram. In these schemes the task of the hearing model is to estimate the amount of quantization noise that will be masked by the audio signal. This general principle can be found in perceptual audio coders of the first generation [5, 6] as well as in the core encoder of the latest generation [7, 8]. For a detailed overview of perceptual coders refer to [3, 9].

In addition to digital transmission, efficient storage of audio data is an important application. Since the market introduction of the compact disc (CD) in the early 1980's, digital audio has become particularly popular because it offers high quality, robustness, and a high degree of convenience, however, at the expense of a high data rate. The invention of perceptual audio coders facilitated the efficient storage and transmission of digital audio. The data rate of the CD has been reduced by a factor of ten while providing perceptually transparent quality. After the standardization of first-generation perceptual coders [10], a real hype was triggered (e.g., 'mp3', internet radio, or Apple's 'iPod' and 'iTunes').

State-of-the-art perceptual audio coders use relatively simple hearing models that typically consider frequency-domain masking only [2, 6, 11, 12]. In currently available speech coders, which are based on linear prediction, the auditory model is even reduced to a simple quantization-error weighting filter [2, 13]. The simple

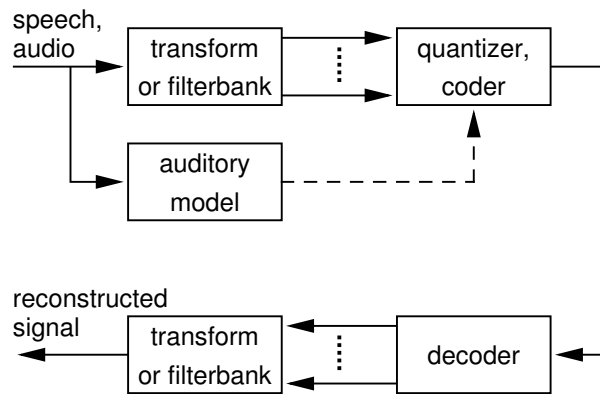


Figure 1.1: Structure of state-of-the-art perceptual coders. The auditory model controls the quantizer.

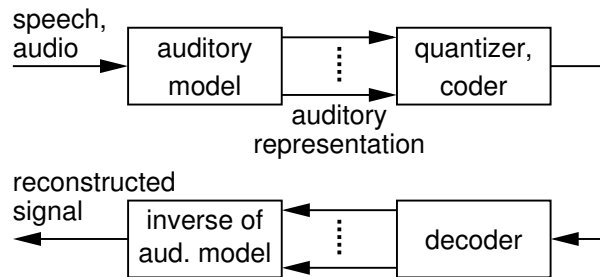


Figure 1.2: The principle of coding in the perceptual domain: quantization and encoding of an auditory representation.

distortion criteria used in practical systems result from a desire to perform efficient quantization at reasonable computational complexity (e.g., scalar quantization). Such efficient, low-complexity quantization is facilitated by three conditions:

1. The (vector) variable is of low dimension;
2. The distortion criterion is single-letter (i.e., the distortion forms a sum over sample distortions);
3. The variables are independent.

These requirements are particularly well illustrated by discrete cosine transform (DCT) based lapped transforms commonly used in audio coding [14]. These transforms allow a spectrally-weighted squared error distortion measure to be approximated as a single-letter criterion. The weighting should be in accordance with a curve for simultaneous masking, which can be recalculated for every signal block. For wide-sense stationary signals, the results of the DCT are asymptotically equivalent to the results of the Karhunen-Loève transform, thus performing an approximate decorrelation of the data. Finally, simple scalar quantization is sufficient and yet efficient and guarantees a low complexity. However, for non-stationary signals, this scheme is no longer optimal because (*i*) the data is not decorrelated anymore and (*ii*) consideration of simultaneous masking only is not sufficient any longer.

Our objective is to use distortion criteria based on sophisticated auditory models without the significant approximations commonly used. In related literature, more complex auditory models have been suggested to be incorporated into the common architecture of state-of-the-art audio coders (e.g., [15] or [16]). However, these require computationally expensive analysis-by-synthesis schemes. Most sophisticated, quantitative models of human auditory perception provide an auditory representation (or ‘internal representation’) of the acoustic signal as output, but generally do not include a quantitative measure of the perceptual distance of two realizations of the auditory representation. In [17] a correlation measure of the internal representations was proposed as an objective distortion measure. Such a measure is closely related to a single-letter weighted squared error measure. We will assume that a single-letter distortion criterion on the auditory representation can provide a high-quality distortion measure.

The introduction of knowledge of the auditory system into coding has been handicapped by delay and computational constraints. For instance, temporal masking or the temporal adaptation to a stimulus are highly nonlinear effects [18, 19]. A time-localized quantization error in the perceived signal can result in a significant change in the auditory nerve firings over a response time interval that can last on the order of hundreds of ms. Therefore, the effect of time-localized

quantization errors that are hundreds of ms apart cannot be separated into additive terms. As a result, it is difficult to include such dependencies of quantization errors during the quantization process.

The usage of sophisticated distortion criteria within the existing coding architectures leads to so-called delayed-decision coding. Delayed-decision coding methods have been used in the context of a squared-error criterion and linear-prediction based waveform coding (e.g., [20]). In the delayed-decision approach, the quantization of a signal block is decided only after consideration of the quantization of a certain number of future blocks. Even when using pruning procedures that eliminate the consideration of unlikely configurations, this method becomes computationally very expensive for distortion measures that have the long time responses associated with hearing models [21].

In [22, 23] another approach has been suggested to address the problem of long-term dependencies in distortion measures based on accurate auditory models. This approach uses a squared error weighted by the so-called ‘sensitivity matrix’ as the distortion measure. In contrast to the simple (and therefore not accurate) weighted squared error criteria with diagonal weighting matrices, the sensitivity matrix is generally not diagonal [22, 24], unless the signal is transformed into the perceptual domain. A non-diagonal weighting matrix means that we do not obtain a single-letter distortion criterion. Just as delayed-decision coding, this approach cannot solve the problem of high computational complexity. This fact motivates the consideration of less conventional coding architectures.

In addition to the fact that state-of-the-art coders (such as the MPEG coders) use only simple auditory models, another common drawback is that the output of the perceptual model is based on an entirely different signal representation than the one used for the main signal path where the quantization is performed. Therefore, the actual effect of the quantization error of a signal parameter on the resynthesized signal is not well reflected by the predicted masking threshold (see for instance [25]). Furthermore, this mismatch becomes particularly noticeable due to the fact that virtually all currently used speech and audio coding methods operate on a block-by-block basis (e.g. [6, 12, 14, 20]). For subband/transform coding for example, decimated filterbanks or lapped transforms are used that introduce block boundaries at regular time positions (generally independent of the actual audio signal). Such a signal representation allows only a suboptimal quantization (in the sense of rate versus distortion) since a signal is generally not stationary within a block and audible artefacts such as pre-echoes or musical noise can occur [2, 6]. In recent coding schemes [6, 9] so-called window switching has been incorporated, to be able to adapt to the temporal signal characteristics to some extent. Our goal is to use only a single signal representation for both the quantization signal path and the perceptual model to avoid the mentioned mismatch. Furthermore, the signal representation should be free of block boundaries. The mentioned drawbacks

motivate the investigation of new coding schemes.

This thesis deals with the coding paradigm introduced by Kubin and Kleijn in [26]. In this paradigm, the coding is performed in the perceptual domain where a simple single-letter distortion criterion forms an accurate and meaningful measure for the perceived distortion. The discussion above concerning the sensitivity matrix, which can provide an accurate distortion measure, revealed already the advantage of transforming the signal into the auditory domain. The valid usage of a single-letter distortion criterion facilitates efficient quantization at a reasonable computational complexity. The principle of this coding approach is shown in Fig. 1.2. The speech or audio signal is transformed into an auditory representation by passing it through an auditory model. This auditory representation is quantized and encoded and the signal can be reconstructed in the decoder by an inverse of the auditory model. This approach is new and different from the one used in classical perceptual audio or speech coders, where an auditory model is used only in the analysis stage in parallel to the main signal path to control the quantization and bit allocation [6] (compare Fig. 1.1 and Fig. 1.2).

The new coding approach avoids the high computational complexity of the delayed-decision approach or the sensitivity matrix approach by exploiting the single-letter nature of the criterion in the auditory representation. This auditory-domain approach towards coding allows the usage of a single-letter distortion criterion and yet accounts for the dependency of perceived distortion on errors in the signal that are far apart in time. However, the transform from the acoustic to the auditory domain can be many-to-one, making the inverse transform in general non-unique. Finally, we note that the parameters making up the auditory representation generally are not independent. That is, coding of the auditory representation removes computational complexity associated with the distortion criterion, but it does not eliminate the need for signal modeling. In this thesis we will discuss methods that deal with this redundancy in an efficient manner.

## 1.2 Outline of the Thesis

This thesis is structured as follows: Chapter 2 discusses the requirements an auditory model has to meet in order to be used for a perceptual-domain coder. Sections 2.1 and 2.2 describe the simple, invertible auditory model proposed by Kubin and Kleijn in [26]. This model provides the base system used throughout this thesis. We will discuss physiological and psychophysical facts to show whether the model is plausible or not. Section 2.3 deals with the efficient inversion procedure to reconstruct the speech or audio signal from its auditory representation obtained by this model.

Chapter 3 addresses the problem of ‘auditory-domain sparsification’. This is

the elimination of perceptually less important components of the auditory representation to facilitate efficient encoding. For this purpose, we propose a new masking model based on the so-called ‘transmultiplexer’ that is formed by the decoder and a human listener. Section 3.2 describes the transmultiplexer and discusses its similarity to existing communication systems. In section 3.3, the masking model, which accounts for both simultaneous and temporal masking, is presented. Since the elimination of signal components removes signal energy, an amplitude correction is necessary to ensure proper signal reconstruction. Section 3.4 deals with this problem and proposes a new correction method. In section 3.5 we show experimental results including data from a listening test for a subjective evaluation of the resynthesis quality. Finally, section 3.6 proposes a method to incorporate sophisticated models of temporal adaptation to exploit temporal masking more accurately for the auditory-domain sparsification process.

The auditory representation obtained by our model consists of sparse pulse trains. Chapter 4 discusses first approaches to the challenging problem of quantizing and encoding spike signals. This is the joint encoding of pulse amplitudes and pulse positions. While section 4.2 focuses on a lossless encoding of the positions, which enables an independent quantization of the amplitudes (section 4.3), section 4.4 considers a joint encoding by means of vector quantization. For this purpose, a new similarity measure for sparse vectors is suggested, and we combine run-length encoding and vector quantization to achieve a variable frame length encoding that requires only small codebooks. The considered encoding strategies in chapter 4 as well as the presented statistical analysis of our sparse pulsed signal representation should build the basis for the design of an efficient quantization scheme for further research work.

In order to achieve a low computational complexity, state-of-the-art audio coders mainly use FFT-based filterbanks for their masking models and/or to compute the subband signals. For the masking models, often a simple grouping of FFT bins is performed to approximate the frequency selectivity of the human hearing system (e.g., [5, 10]). To get auditory filterbanks that are more accurate and yet computationally efficient, chapter 5 deals with alternative filterbank implementation methods. Special attention is paid to frequency-warped transform filterbanks, which are memory efficient and enable the usage of the FFT, in section 5.2. A design recipe is proposed to accurately approximate a gammatone filterbank with a relatively short prototype filter. The chapter also describes inversion methods of the considered filterbanks, which are essential for the usage in a perceptual-domain coder. Finally, chapter 6 summarizes the thesis and discusses necessary issues for further research to obtain efficient high-quality coders based on the approach of coding in the perceptual domain.

## 1.3 Nomenclature

### 1.3.1 Symbols and Mathematical Notation

#### Elementary Identifiers

$t$	continuous time in seconds
$n$	discrete time as sample index
$f$	frequency in Hz
$\theta$	angular frequency in radians per sample
$f_s$	sampling rate in Hz
$j$	imaginary unit $j = \sqrt{-1}$
$e$	natural logarithm base $e = \exp(1)$
$W_N$	'twiddle factor' $W_N = e^{-j2\pi/N}$ , $N \in \mathbb{N}$
$x$	input signal of a system
$y$	output signal of a system
$\hat{x}$	approximation of $x$
$h$	impulse response of a linear, time-invariant (LTI) system

#### Number Sets

$\mathbb{N}$	set of cardinal numbers
$\mathbb{N}_0$	set of cardinal numbers and 0
$\mathbb{Z}$	set of integers (whole numbers)
$\mathbb{R}$	set of real numbers

#### Functions and Sequences

$x(t)$	function, usually a continuous-time signal, $t \in \mathbb{R}$
$x[n]$	sequence, usually a discrete-time signal, $n \in \mathbb{Z}$
$\delta(t)$	Dirac delta distribution
$\delta_{2\pi}(t)$	$2\pi$ -periodic Dirac delta function
$\delta[n]$	unit impulse sequence
$\gamma(t)$	gamma function of 4th order
$H(e^{j\theta})$	discrete-time Fourier transform of $h[n]$
$H(z)$	$z$ -transform of $h[n]$

**Auditory Filterbank/Model Entities**

$K$	number of channels in a filterbank
$h_k$	impulse response of the $k$ th channel of an analysis filterbank
$g_k$	impulse response of the $k$ th channel of a synthesis filterbank
$q_k$	channel weight to account for the outer and the middle ear
$\kappa_k$	correction factor due to adaptive downsampling
$1/\alpha_k, \beta_k$	correction factors due to off-peak errors
$f_c$	center frequency of an auditory filter in Hz
$N_{gt}$	length of a gammatone impulse response in samples
$c$	power-law exponent
$r$	impact factor of the masking criterion
$r^{(dB)}$	impact factor in dB

**Other Entities**

$H_{equ}(e^{j\theta})$	frequency response of the equalizer filter
$H_{i100}(e^{j\theta})$	frequency response of the inverse 100-phon filter
$A(z)$	transfer function of a first-order all-pass filter
$\lambda$	pole of a first-order all-pass filter, frequency warping parameter
$\mathbf{T}$	transform of a transform filterbank
$\mathbf{I}$	identity matrix
$\tilde{h}[n]$	prototype low-pass filter
$w_k$	window coefficient
$N_w$	length of the window

**Operators**

$I$	identity operator
$F$	analysis (frame) operator
$F^*$	adjoint analysis (frame) operator
$\mathcal{F}$	Fourier transform operator
$\mathcal{E}\{\}$	expectation operator
$\mathcal{H}\{\}$	Hilbert transform
$\mathcal{EN}\{\}$	Hilbert envelope
$*$	convolution

### 1.3.2 Abbreviations

3AFC procedure	Three-Alternative Forced-Choice procedure
AGC	Automatic Gain Control
AM	Amplitude Modulation
BM	Basilar Membrane
CD	Compact Disc
DC	constant signal (from Direct Current)
DCR	Degradation Category Rating
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
DSP	Digital Signal Processor (or Processing)
ERB	Equivalent Rectangular Bandwidth
FFT	Fast Fourier Transform
FIR	Finite Impulse Response
GUI	Graphical User Interface
IHC	Inner Hair Cell
IIR	Infinite Impulse Response
ISI	Inter-Symbol Interference
JND	Just Noticeable Difference
MIMO system	Multi-Input, Multi-Output system
MPEG	Moving Picture Experts Group
MPLP	Multi-Pulse excitation Linear Prediction
NBN	Narrowband Noise
OME	Outer and Middle Ear
PAM	Pulse Amplitude Modulation
PR	Perfect Reconstruction
PSD	Power Spectral Density
RLE	Run-Length Encoding
RLVQ	Run-Length Vector Quantization
RMS	Root Mean Square
SegSNR	Segmental Signal-to-Noise Ratio
SNR	Signal-to-Noise Ratio
SPL	Sound Pressure Level
XOR	eXclusive OR



## Chapter 2

# Speech and Audio Coding in the Perceptual Domain

In [26] a speech coding paradigm was introduced in which the encoding is performed in a perceptual domain where a simple distortion criterion should form an accurate and meaningful measure for the perceived distortion. In other words, the speech or audio signal is transformed into an auditory representation by passing it through an auditory model. This auditory representation is quantized and encoded and the signal can be reconstructed in the decoder by an inverse of the auditory model.

The proposed paradigm demands a model of the human auditory system that satisfies the following requirements:

1. It provides an accurate quantitative description of perception;
2. It leads to an auditory signal representation with relatively few parameters;
3. It can be inverted with relatively low computational effort.

The first requirement is to ensure that a simple distortion criterion is sufficient in the perceptual domain to reflect actually perceived degradations. The second point requests a compact auditory representation in order to have a good basis for data compression. However, at this point a first contradiction arises because an accurate auditory representation is not compact but highly redundant. This fact should be clear when we keep in mind that about 30,000 auditory neurons [27] encode the stimulus processed by a human cochlea. The third requirement is essential for coding applications since we need to reconstruct the coded signal. Also this point contradicts the first requirement because an accurate auditory model has to account for nonlinear long-term effects and, consequently, the inversion of such nonlinearities could be difficult or even impossible.

The discussed contradictions raise the difficulties associated with the new coding paradigm. An auditory model that satisfies the above listed requirements to some extent was proposed in [26] and is described in more detail in the next sections of this chapter. The model is simple and has therefore limited accuracy, but it is easily invertible and yet achieves a high reconstruction quality. The inversion procedure is described in section 2.3. In chapter 3 we try to satisfy the accuracy requirement by considering nonlinear long-term effects, and we incorporate an adaptation model without jeopardizing the simple inversion procedure (section 3.6). The main focus of chapter 3, however, is the second requirement, that is to obtain a compact auditory representation.

## 2.1 A Physiologically Motivated Subband Coder Based on an Invertible Auditory Model

In [26] an invertible auditory model was proposed, which is based on the pulse-ribbon model [28] introduced by Patterson in the late 1980's. The left side of Fig. 2.1 shows the stages of this simple model. It covers the basilar membrane (BM), inner hair cells (IHC), and first neural stages as neuron ensembles, i.e., it models the cochlea and the auditory nerve in the human inner ear, but skips the outer and the middle ear. The right side of Fig. 2.1 shows the inversion procedure, which is described in more detail in section 2.3.

In this model, the first stage to simulate the motion of the BM caused by acoustic stimulation is a non-decimated analysis filterbank. It is well known that stimuli with different frequencies produce responses with maxima at different locations along the BM. For this purpose, a functional model consists of a bank of band-pass filters with different center frequencies. Note that in a human cochlea about 3,500 inner hair cells [29] are located along the BM and, therefore, this is the actual number of band-pass channels. One reason for this high redundancy is to be robust against damages such as loss of hair cells etc. But this also means that neighboring auditory filters look rather similar and, hence, for modeling purposes or coding applications, it is not necessary (and hardly possible) to implement such a high number of cochlea channels. For the invertible model in [26], the well-known gammatone filterbank [30], implemented using FIR filters, with 20 channels for 8 kHz sampled signals is used.

In each auditory channel, the analysis filterbank is followed by a model of an inner hair cell. The task of the inner hair cells is to convert the displacement of the BM in electrical receptor potentials. These receptor potentials cause a release of neurotransmitters and excite the peripheral terminals of cochlear-afferent neurons [31, 32]. In our model, this transduction process is reproduced in a simplified way

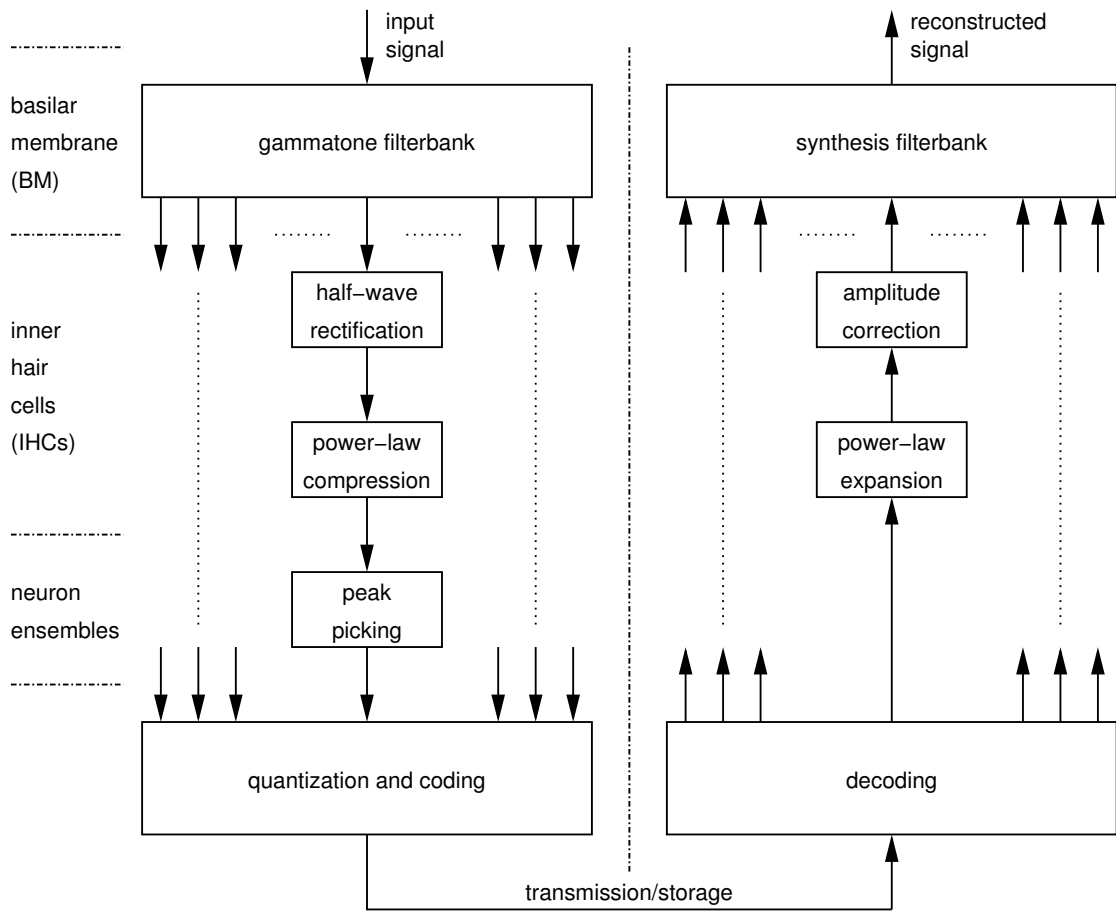


Figure 2.1: The invertible auditory model proposed in [26] used as the transform for a perceptual-domain coder.

using static nonlinearities only, namely a half-wave rectifier and a compressive nonlinearity.

The final stage in our invertible model mimics the behavior of an ensemble of cochlear-afferent neurons in each auditory channel. According to the excitation by neurotransmitters, these neurons produce action potentials (‘firing pulses’) caused by depolarization of an auditory nerve fiber. We model this generation of pulses using a peak-picking procedure. The set of firing pulse trains obtained from all auditory channels is referred to as the auditory representation, which is a perceptual time-frequency representation of the original speech or audio signal.

In the next section, we will describe the components of this auditory model in more detail. We deal with filterbanks whose characteristics are matched to the acoustical and mechanical behavior of the cochlea and basilar membrane. One of these characteristics is that the spectral resolution decreases with increasing frequency. Therefore, warped frequency scales have been introduced long ago where selectivity bandwidths remain approximately constant along the warped frequency axis (auditory scales), e.g., the Bark (critical-band rate) scale [33] or the ERB (equivalent rectangular bandwidth) rate scale [34]. We give a survey of auditory scales and auditory filters. The emphasis is placed on invertibility to allow reconstruction of the input signal. This enables the filterbank pair—analysis and synthesis filterbank—to be used for auditory subband coders or to be used in an invertible auditory model. Furthermore, we will consider important aspects of the implementation of the auditory filterbank, which is the computationally most complex component of the model.

In section 2.3 we present the computationally efficient, non-recursive inversion procedure of the simple auditory model, which allows to reconstruct the input signal at a high quality from the auditory pulse representation.

## 2.2 Auditory Analysis

The selection of the components of the proposed auditory model is based on existing knowledge about the human auditory system. In this section, we give additional details about the motivation of the choices.

### 2.2.1 Basilar Membrane Filterbank

The filterbank to simulate the behavior of the BM is the computationally most complex component of the model. After providing an overview of auditory filters, we consider different aspects of the implementation of an auditory filterbank.

### 2.2.1.1 Brief Overview of Auditory Filters

The frequency selectivity of the human auditory system has been studied by means of psychoacoustic experiments and measurements in the cochlea and on the auditory nerve over many decades. The results of these experiments have led to the concept of auditory filters. For a historical overview, we refer to [35].

Once the bandwidths of these filters are found and expressed as a function of the center frequency, an auditory scale can be defined by integrating the reciprocal of the bandwidth function (the bandwidth function can be seen as the first derivative of the frequency with respect to the unit of the bandwidth). For instance, the equivalent rectangular bandwidth  $\text{ERB}(f_c)$  as a function of the filter's center frequency  $f_c$  in Hz is [34]

$$\text{ERB}(f_c) = 0.1079f_c + 24.7 \quad (2.1)$$

and the corresponding frequency scale, the ERB rate (or ‘number of ERBs’) is then

$$\#\text{ERBs}(f) = \int \frac{df}{\text{ERB}(f)} + \text{const} = 21.4 \log_{10}(1 + 0.00437f), \quad (2.2)$$

where the integration constant has been chosen to make  $\#\text{ERBs}(0) = 0$ .

Auditory frequency scales are related to the frequency-position mapping performed by the cochlea. In Fig. 2.2, the ERB rate from Equ. (2.2) and the Bark scale (critical-band rate) [33, 36, 38] are compared with a position-frequency function, which was derived by Greenwood [37] from measurements of the mechanical motion of the BM. In this comparison, the scales are normalized. At the maximally presented frequency of 4000 Hz, the ERB rate reaches 27.1 ERBs, the Bark scale has 17.1 Bark, and the basilar membrane position is 23.4 mm. The figure reveals that the ERB rate is much closer to the BM position function than the Bark scale. We suppose that the warped auditory frequency scale originates from the frequency-position mapping and prefer therefore to use the ERB rate and ERB-related auditory filters over Bark-based ones. For other comparisons and more details, see also chapter 5 or refer to [39]. In chapter 5, we will approximate the auditory frequency scale by means of an all-pass transform.

The shape of the auditory filters has been obtained by fitting different parametric expressions to experimental data. A simple linear frequency-domain description of auditory filters is the ‘rounded exponential’, or ‘roex( $p, q$ )’ function [40]

$$|H(f)|^2 = (1 - q)(1 + p\xi)e^{-p\xi} + q \quad (2.3)$$

where  $\xi$  is the normalized deviation from the center frequency  $f_c$

$$\xi = \frac{|f - f_c|}{f_c}. \quad (2.4)$$

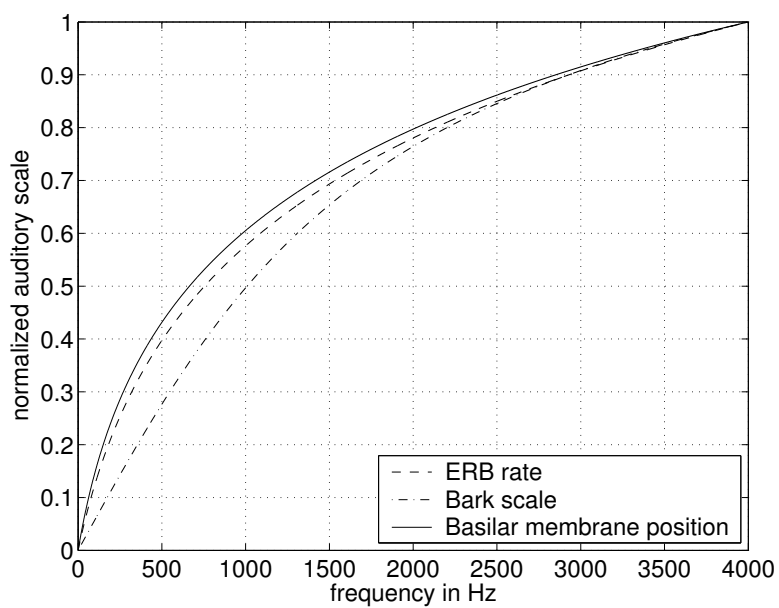


Figure 2.2: Comparison of the ERB rate (Glasberg and Moore, 1990 [34]), the Bark scale (Zwicker and Terhardt, 1980 [36]), and the frequency-position mapping (Greenwood, 1990 [37]).

The parameter  $p$  determines the bandwidth and should be chosen as

$$p = \frac{4f_c}{\text{ERB}(f_c)}. \quad (2.5)$$

The second parameter,  $q$ , flattens the shape outside the passband.

A more recent, time-domain description is the well-known gammatone function [30] for the filter impulse response

$$h(t) = at^{(l-1)}e^{-2\pi bt} \cos(2\pi f_c t) \quad \text{for } t > 0 \quad (2.6)$$

where  $f_c$  is the frequency of the carrier and, therefore, the center frequency of the filter. The envelope of the impulse response is a gamma function with the parameters  $l$  and  $b$ , where  $l$  is the order of the gamma function, and  $b$  largely determines the bandwidth. Patterson [30] determined the choice  $l = 4$  and

$$b = 1.019 \text{ERB}(f_c). \quad (2.7)$$

For our simulations, we use gammatone filters, since the time-domain description allows a straightforward FIR filter design. We discuss issues for the design of a whole bank of gammatone filters and further implementation details in the next subsection.

Finally, it should be mentioned that several nonlinear effects of the BM behavior have been observed. One effect is the dependency on the sound pressure level [41] that causes asymmetric frequency responses of the auditory filters. Especially for low stimulus levels, the shape of the auditory filters becomes considerably narrower. To account for this dependency, descriptions for auditory filters have been extended [34, 42]. The level dependency even may be the reason for two-tone suppression. Auditory filterbanks that are able to account for two-tone suppression can be found in [43, 44]. Sellick et al. [45] found from measurements of the BM motion that most of the nonlinearities such as compression or saturation, which are elsewhere considered to originate from IHC transduction or synaptic transmission, can be observed already on the BM. In order to obtain really accurate BM models, a considerable amount of further research work has to be done. For simplicity, particularly with respect to invertibility of the auditory model, we will only consider linear filters for which the above descriptions are valid for moderate sound pressure levels.

### 2.2.1.2 Implementation Aspects

To obtain the coefficients of a gammatone auditory filter implemented as an FIR filter, we simply sample Equ. (2.6) to get the discrete-time impulse response

$$h[n] = \begin{cases} a(n/f_s)^3 e^{-2\pi bn/f_s} \cos(2\pi f_c n/f_s), & 0 \leq n < N_{gt} \\ 0, & \text{otherwise,} \end{cases} \quad (2.8)$$

where  $f_s$  is the sampling rate and  $N_{gt}$  the chosen length in samples. We choose the amplitude  $a$  of Equ. (2.8) in such a way as to obtain a normalized frequency response at the center frequency

$$H(e^{j\theta})|_{\theta=\theta_c=2\pi f_c/f_s} = 1. \quad (2.9)$$

The Fourier transform of an impulse response of the general form

$$h[n] = e[n] \cos(\theta_c n), \quad (2.10)$$

where  $e[n]$  is the envelope of  $h[n]$  (i.e., the gamma function in our case), evaluated at the carrier frequency  $\theta_c$ , can be expressed as

$$H(e^{j\theta})|_{\theta=\theta_c} = \frac{1}{2}E(e^{j\theta})|_{\theta=0} + \frac{1}{2}E(e^{j\theta})|_{\theta=2\theta_c}. \quad (2.11)$$

Since  $E(e^{j\theta})$ , the Fourier transform of the (smooth) envelope  $e[n]$ , is the frequency response of a low-pass filter, the second term of the last equation becomes small and can be neglected. Thus, we obtain a relation between the frequency response normalized at the center frequency and the envelope that has to sum up to the constant of 2:

$$\sum_{n \in \mathbb{Z}} e[n] \approx 2. \quad (2.12)$$

This relation will be of interest in section 2.3 and again in chapter 3.

An implementation of an auditory filterbank consists of many auditory filters with different center frequencies and different bandwidths in parallel. We denote the number of channels as  $K$  and introduce the channel index  $k \in \{0, \dots, K-1\}$ , which is used as a subscript. We also apply the channel subscript to previously introduced entities such as the center frequency  $f_{c,k}$ , the gammatone parameters  $a_k$  and  $b_k$ , and the impulse response  $h_k[n]$ . For coding applications, we should be able to reconstruct the input signal from the channel signals, and the filter bank should be invertible. We denote the  $z$ -transforms of the analysis filters as  $H_k(z)$  for  $k = 0, \dots, K-1$  and of the synthesis filters as  $G_k(z)$  for  $k = 0, \dots, K-1$ . The corresponding impulse responses of the synthesis filters are denoted as  $g_k[n]$ . We thus obtain the analysis-synthesis structure shown in Fig. 2.3. Filterbank inversion and the design of synthesis filters are described in more detail in section 2.3.2.

A commonly used method to compute the center frequencies for the filters is to transform the minimum and the maximum center frequency of interest ( $f_{min}$  and  $f_{max}$ ) from Hz into ERB rate. This range is divided into  $K-1$  uniform sections and the obtained ERB rates are finally transformed back into Hz:

$$f_{c,k} = \#ERBs^{-1} \left( \#ERBs(f_{min}) + k \frac{\#ERBs(f_{max}) - \#ERBs(f_{min})}{K-1} \right). \quad (2.13)$$

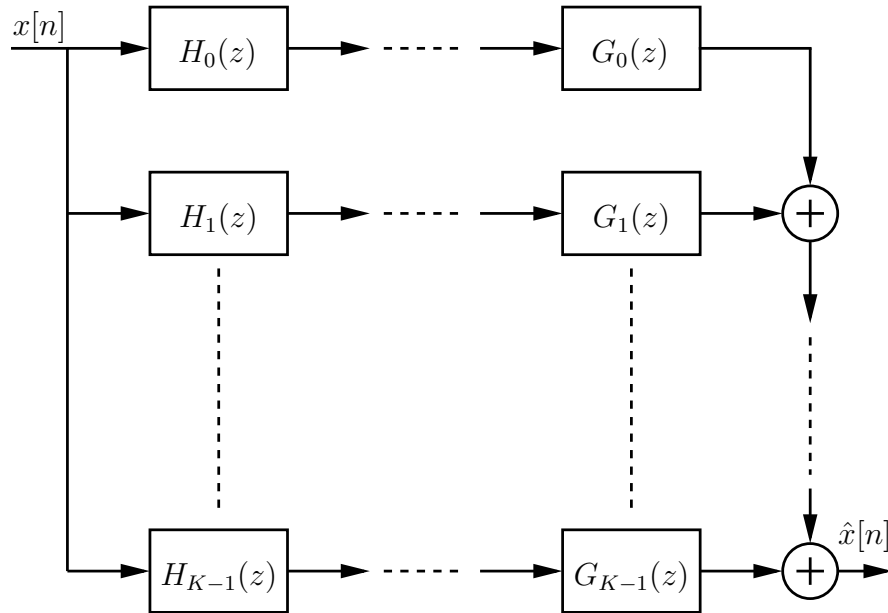


Figure 2.3: General structure of an analysis and synthesis filterbank (without any decimation of the channel signals).

As shown in Equ. (2.8), the discrete-time impulse responses of the gammatone filters can be designed by sampling and windowing the continuous-time infinite-length impulse responses of Equ. (2.6). A problem with direct usage of these impulse responses for FIR implementations is that the impulse responses are rather long. In Fig. 2.4, a gammatone impulse response for a center frequency of 500 Hz is plotted. Its envelope is shown as well and compared with the envelopes obtained for center frequencies of 200 Hz and 80 Hz. As can be seen from this figure, an impulse response with a support of about 30 ms (240 samples at a sampling rate of 8 kHz) is needed for a center frequency  $f_c = 200$  Hz to keep the effects of windowing small and thus, to approximate the frequency response of an ideal gammatone filter accurately. For lower center frequencies, the length increases further (e.g., 60 ms for  $f_c = 80$  Hz). Therefore, the corresponding FIR implementations are computationally expensive and memory consuming.

In chapter 5 we discuss alternative implementation methods, which are computationally less expensive and should therefore be preferred when real-time applications running on a DSP are considered. However, for the experiments and simulations described in the following sections, we use FIR gammatone filters because computational complexity has not been an issue there.

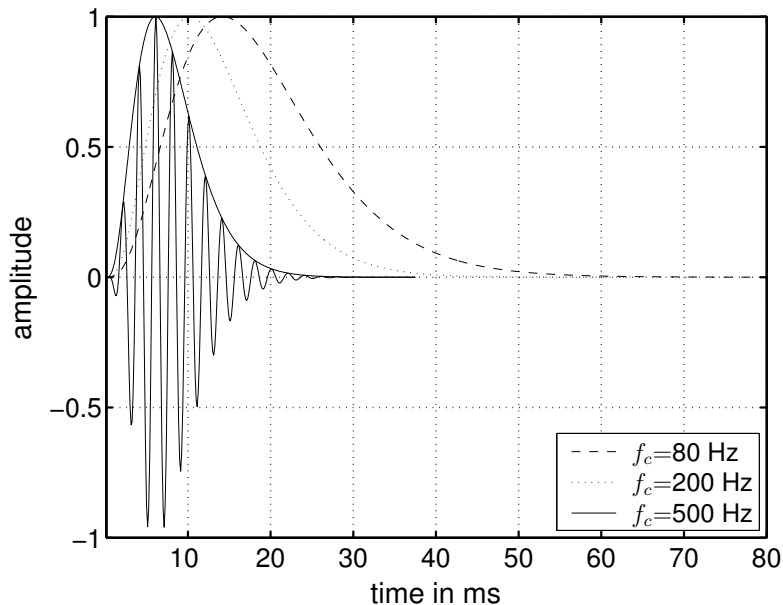


Figure 2.4: Impulse response and impulse response envelopes of gammatone filters for different center frequencies.

### 2.2.2 Inner-Hair-Cell Model

The auditory filterbank is followed by a half-wave rectifier and a power-law compressor, simulating the behavior of inner hair cells. The task of the inner hair cells is the so-called transduction process, i.e., to convert mechanical movements into electrical potentials. It is assumed that the displacement of the cilia of the cells is proportional to BM velocity [41, 45]. Measurements of electrical responses have revealed a directional sensitivity: while displacement in one direction is excitatory, movement in the opposite direction is inhibitory [41]. Thus, the cells mainly react to positive deflection of the BM and, consequently, it is reasonable to model this behavior with a half-wave rectifier. Half-wave rectification is commonly used to model this aspect of physiology, e.g., [19, 46, 47, 48].

The aforementioned measurements also show a compressive response [41]. To model this behavior, we apply a power-law compressor to the half-wave rectified signals. The input  $x[n]$  and the output  $y[n]$  of the IHC model are related by

$$y[n] = \max(x[n], 0)^c \quad (2.14)$$

with the power-law exponent chosen as  $c = 0.4$  [26]. The function  $\max(x[n], 0)$  implements the half-wave rectifier.

The static nonlinearity is a strongly simplified model of the human peripheral

processing. In related literature more sophisticated models of inner hair cells can be found. A physiologically plausible synapse model has been proposed by Meddis [49, 50]. Other examples can be found in [46] and [47] where automatic gain controllers model the synaptic region between the hair cell and the nerve fiber. In [19] a functional model with a compression stage that is able to adapt to the characteristics of the stimulus is used. This compression stage consists of a cascade of five feedback loops with different time constants. The cascade compresses stationary sounds according to a power law whereas rapidly varying stimuli are transformed more linearly, thus modeling the ‘overshoot effect’, i.e., a higher sensitivity at the onset of a stimulus. In appendix B we describe this adaptive compression stage in more detail. In our first implementation of an invertible model we use the simple half-wave rectifier and power-law compressor to avoid difficulties such as stability problems when inverting the gain control loops. Later, in section 3.6, we also discuss the incorporation of dynamic nonlinearities into our coding scheme.

Compressive stages can be found in virtually all functional models of intensity sensation. The Weber-Fechner law relates the sensation to the physical stimulus intensity by a logarithmic function [51]. On the other hand, Stevens’ universal law of intensity sensation across all sensory modalities is a power law [51]. Stevens’ law can for instance be found in widely accepted models of loudness [33]. However, there is a controversy about the general validity of either the one or the other psychophysical law [51, 52] and also about whether the compression happens during the peripheral (e.g., in the cochlea) or the central processing (i.e., in the brain). In [52, 53], based on experiments with cochlear implant listeners, a phenomenological loudness model has been suggested in which the cochlea (or the cochlear nucleus for low frequencies) performs a logarithmic compression while the brain performs an exponential expansion. Both functions together result in a power law. Regardless whether logarithmic or power-law compression should be preferred, for the encoding of amplitude, it is well known that a non-uniform quantization, which can be achieved by a uniform quantization after compression, brings advantages (cf. the companding quantization schemes in ordinary waveform coders such as A-law or  $\mu$ -law [1, 54]).

### 2.2.3 Neuron Model

Contrary to many other auditory models (e.g. [19, 46, 47, 48]), we preserve the temporal fine structure of the signal, i.e., we do not apply time averaging to the subband signals because this would lead to a low reconstruction quality. In our model, the power-law compressor is followed by an adaptive subsampling mechanism (‘peak picking’), which searches for local maxima and sets all other samples to zero. Let the input and the output of the peak-picking stage be denoted by  $x[n]$

and  $y[n]$ , respectively, then the output can be calculated as

$$y[n] = \begin{cases} x[n], & x[n] > x[n-1] \wedge x[n] > x[n+1] \\ 0, & \text{otherwise.} \end{cases} \quad (2.15)$$

This model simulates the firing behavior of an ensemble of auditory neurons. The responses are clusters of high firing activity that are synchronized (phase-locked) with the waveform shape of the input signal.

It is known that a single neuron generally does not fire more often than 250 times per second [31, 32] and, therefore, it is by itself not able to preserve the time structure of high-frequency components. Since in the early human auditory system about 30,000 neurons [27] encode the signals of significantly less hair cells, we can associate several neurons with one hair cell output. Our model of the neurons is physiologically plausible. Each neuron has an internal state that decays exponentially with a relatively large time constant. When it fires, this state is reset to a value that depends on the input signal level. The firing probability increases monotonically with the difference between the neuron's input and its state. So an ensemble of neurons shows a high firing rate at the peak of the input signal. The amplitude of a pulse in our model represents the firing rate, i.e., the number of neurons of the ensemble that fire at the peak location.

The effect of phase locking is known to occur only at frequencies below 4 kHz [31, 32]. So the used model is physiologically plausible for the coding of narrow-band speech signals. For simplicity, we use this neuron model even when we process signals at higher sampling rates, e.g., wide-band speech or general audio signals. However, the fact that the firing behavior is no longer waveform-synchronized in high-frequency channels means that more efficient signal representations and encoding schemes are possible. For instance, a coarse quantization of the pulse locations in these channels should be sufficient to preserve the perceptual signal quality. In chapter 4 we will consider the encoding of the locations by means of vector quantization. The mentioned effect is partially exploited in many existing wide-band speech and audio coders. Examples are parametric models for the high-frequency band [55], where modulated noise is used as the excitation for the prediction-based synthesis filter, and bandwidth extension (either with side information [7, 56] or even without [57]).

The consideration of pulsed neural models where information is carried in the pulse timings is clearly motivated by observations of biological neural networks. In [58] it is well demonstrated that these models should be preferred to classical neuron models such as firing-rate models that average over time or even more simplified ones for many applications of artificial neural networks.

In Fig. 2.5, a pulse representation of a segment of about 30 ms duration taken from voiced speech is shown. For this example, a 50-channel FIR gammatone filter bank was used. The neuron firings are not strictly aligned across the frequency

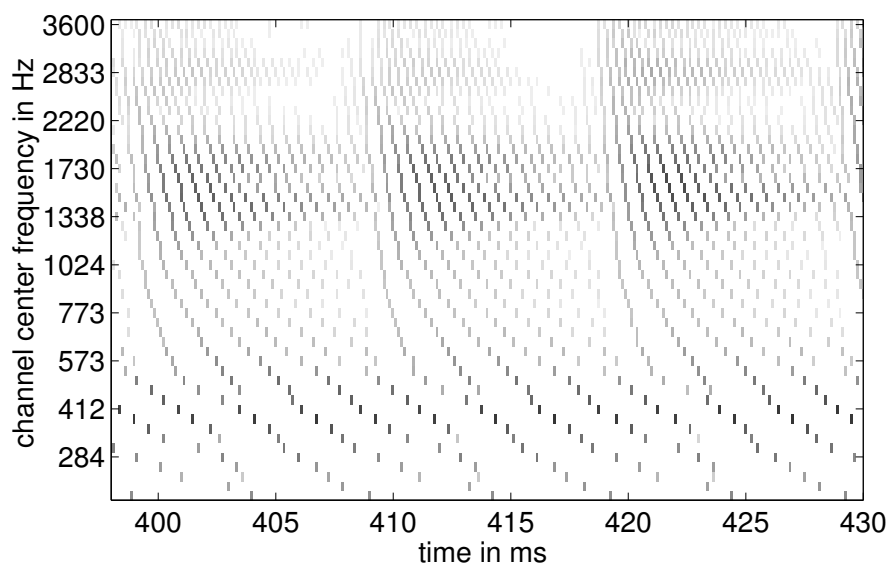


Figure 2.5: Auditory representation (here with 50 channels) of the sound [I] taken from ‘there is’, spoken by a male. Peaks are shown as narrow rectangles with their grey-level representing their amplitude. The time axis covers three pitch periods.

channels due to different delays of the filters. Nevertheless, the phase-locking effect can be seen clearly. Also the formant structure is visible with formants around 400 Hz, 1.7 kHz, and 2.8 kHz.

Weintraub [59] used a similar deterministic model for neural firing in his sound separation system. There is also similarity to Patterson’s pulse ribbon model [28] but we preserve the amplitudes of the pulses in addition to the locations. Contrary to [59] and [28], we are able to resynthesize the original audio signal directly from this neural firing pulses whereas Weintraub uses the (unprocessed) signals from the auditory filterbank for the resynthesis [60], and Patterson does not resynthesize at all.

## 2.3 Auditory Synthesis

The attempts to resynthesize the input signal from an auditory representation are not new. In [61] a historical overview is given. The aim of various model inversions was to understand perception [62, 63, 64], to test the accuracy of the model [65, 66], and to separate speech from noisy backgrounds or interfering speakers [59, 60, 64]. We propose to use an invertible auditory model for coding of speech and audio signals [67].

For most recent models [64, 66], the inversion method is based on projections onto convex sets (or ‘alternate projections’ [68]) and utilizes iterative signal reconstruction algorithms. The resynthesis of our auditory model does not need iterative procedures and is, therefore, computationally very efficient and nevertheless perceptually accurate.

### 2.3.1 Inversion of Neuron and Inner-Hair-Cell Models

The first step in the inversion procedure is to undo the power-law compression using the proper inverse expansion to get the positive peak amplitudes of the original signal:

$$y[n] = x^{1/c}[n]. \quad (2.16)$$

Now, each of the channel signals approximates the situation where a signal is downsampled and then upsampled by means of inserting zeros. This insertion of zeros leads to aliasing which can be removed by band-pass filtering. These band-pass filters are located in the synthesis filterbank. Before they are applied, the amplitude of the pulses has to be corrected to compensate for the loss of energy due to (i) the adaptive downsampling and (ii) the peak amplitude errors at higher frequencies introduced by the finite sampling rate.

#### 2.3.1.1 Adaptive Downsampling

Let us consider one auditory channel. The output of one channel of the analysis filterbank resembles a sinusoid with a period of  $P$  samples that is close to the inverse of the center frequency of the filter<sup>1</sup>. The peak picker transforms the sinusoid into a pulse train with the same peak amplitude and the same fundamental frequency. We can approximate the peak-picking procedure as a cascade of an ordinary downsampler and an upsampler with a fixed decimation/interpolation factor  $P$  for which the Fourier transform relation is

$$Y(e^{j\theta}) = \frac{1}{P} \sum_{k=0}^{P-1} X(e^{j(\theta - k2\pi/P)}). \quad (2.17)$$

The cosine signal with amplitude 1 and angular frequency  $2\pi/P$  with Fourier transform

$$X(e^{j\theta}) = \pi (\delta_{2\pi}(\theta - 2\pi/P) + \delta_{2\pi}(\theta + 2\pi/P)) \quad (2.18)$$

---

<sup>1</sup>This assumption is valid when the input signal is rich in the frequency region around the considered channel’s center frequency. The envelope of the sinusoid is typically modulated, especially in higher-frequency channels due to the wider bandwidths. However, this modulation does not influence the derivation of the correction factor because the envelope is preserved in the pulse amplitudes.

is transformed into the pulse train with Fourier transform

$$Y(e^{j\theta}) = \frac{2\pi}{P} \sum_{k=0}^{P-1} \delta_{2\pi}(\theta - k2\pi/P) \quad (2.19)$$

where  $\delta_{2\pi}(\theta)$  is the  $2\pi$ -periodic delta distribution. All additional frequency components have to be attenuated by the synthesis filter and the remaining components yield the cosine signal with amplitude  $2/P$ . Therefore, the amplitude in this channel has to be corrected by a factor of  $P/2$ . For the auditory channel  $k$  with center frequency  $f_{c,k}$  Hz, we obtain the correction factor

$$\kappa_k = \frac{f_s}{2f_{c,k}}, \quad (2.20)$$

where  $f_s$  is the sampling rate. The channel amplification according to this reciprocal function of the channel's center frequency is strongest for low-frequency channels, decreases with increasing frequency and reaches 1 at half the sampling rate.

We can also derive the amplitude correction factor obtained above by considering the reconstruction of a sinusoid from a pulse train in the time domain. We assume the center frequency of the channel to be  $f_{c,k} = f_s/P$  Hz and the impulse response  $g_k[n]$  of the synthesis filter to be a tone burst with envelope  $e_{g,k}[n]$  and tone frequency  $f_{c,k}$ . In the next subsection we show that this assumption for the synthesis filters is valid. The convolution in the synthesis filter simply overlaps and adds versions of the filter's impulse response shifted by integer multiples of  $P$  samples. The peak amplitude at the output of the filter is then  $\sum_i e_{g,k}[iP]$ . From Equ. (2.12) we know that the envelope of the impulse response has to sum up to 2 to ensure that the frequency response is normalized to 1 at the center frequency, i.e.,  $\sum_n e_{g,k}[n] = 2$ . Using the trapezoidal integration rule, we get  $\sum_n e_{g,k}[n] \approx P \sum_i e_{g,k}[iP]$  and, therefore, for the peak amplitude  $\sum_i e_{g,k}[iP] \approx 2/P$ . To make the peak amplitude equal to 1, we need the amplitude correction factor  $\kappa_k = P/2 = f_s/(2f_{c,k})$ .

The method to use a constant correction factor for each channel is very simple and contributes substantially to good resynthesis results. Another slightly more elaborate correction method is to count the actual number of samples between adjacent pulses which replaces the constant correction factor with an adaptive one. Let  $m$  be the pulse index in a single channel and  $P[m]$  the corresponding pulse distance<sup>2</sup>, then the adaptive correction factor is  $\kappa[m] = P[m]/2$ . This method is also discussed in section 2.3.4 in connection with the frame algorithm and is of

---

<sup>2</sup>For our simulations, we use half the distance between the previous  $(m-1)$ th and the subsequent  $(m+1)$ th pulse as the current distance.

interest in section 3.4 for the resynthesis of a reduced pulse representation where pulses have been omitted. However, for the resynthesis of the complete pulse code, the adaptive correction does not increase the reconstruction quality.

### 2.3.1.2 Peak Amplitude Errors

For the second correction step, we observe that the measurement of the peak amplitude is exact in continuous time only. In discrete time, errors due to the finite sampling interval are inevitable. These errors become significant in particular for those auditory channels whose center frequencies are close to half the sampling frequency. To compensate for these errors, a method based on the assumption of a uniformly distributed random sampling error was proposed in [26]. The method evaluates the average per-cycle maximum amplitude of a sampled sinusoid,  $\alpha$ , which, for the case of a unity amplitude sine wave and a unity sampling period, is given by

$$\alpha = \int_{-1/2}^{1/2} \cos\left(\frac{2\pi t}{P}\right) dt = \frac{P}{\pi} \sin\left(\frac{\pi}{P}\right). \quad (2.21)$$

Thus, the correction factor due to the finite sampling rate for this channel is  $1/\alpha$ .

An improved correction factor was introduced in [67], which is based on least-squares optimization. For a sinusoidal signal with amplitude  $A$  and period  $P$ , we observe the sampled maximum amplitude as  $w_{\max} = A \cos\left(\frac{2\pi t}{P}\right)$  with  $t$  uniform over  $[-1/2, 1/2]$ . The nonlinear least-squares estimate for the amplitude  $\hat{A}$  in terms of the observation  $w_{\max}$  is given by  $\hat{A} = \mathcal{E}\{A|w_{\max}\} = \beta \cdot w_{\max}$  with

$$\beta = \int_{-1/2}^{1/2} \frac{1}{\cos\left(\frac{2\pi t}{P}\right)} dt = \frac{P}{\pi} \ln \left[ \tan \left( \frac{\pi}{4} + \frac{\pi}{2P} \right) \right]. \quad (2.22)$$

It can be shown that the correction factor  $\beta$  keeps the reconstruction error evaluated on the power spectral density function less than 1 dB across the entire frequency range covered by the auditory filterbank [67].

Fig. 2.6 compares the two different correction factors,  $1/\alpha$  and  $\beta$ , as functions of the channel's center frequency. The frequency axis is normalized by the sampling rate here. A correction by  $\beta$  amplifies frequencies close to half the sampling rate clearly stronger than  $1/\alpha$ . As it can be seen from this figure, for oversampled situations with a relatively high sampling rate ( $f_s \gg 2f_c$  or  $f_c/f_s \ll 0.5$ ), the amplification is almost 0 dB. In practice, an oversampling by the factor of 2 (or  $f_c/f_s = 0.25$ ) requires a correction by only 1 dB, i.e., the off-peak errors are small and can be ignored. In such cases only the amplitude correction due to the downsampling described in the previous section has to be considered. For other comparisons of the two gain factors to correct off-peak errors, refer to [69].

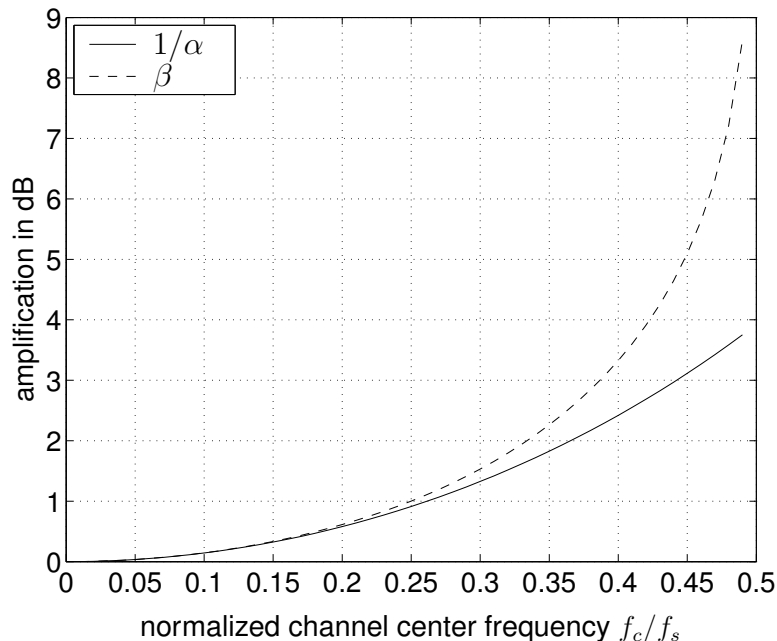


Figure 2.6: Comparison of the two amplitude correction factors to account for peak amplitude errors.

The coder in [70, 71] is also based on our simple, non-iterative reconstruction method. In that work both the amplitude correction due to adaptive downsampling and the correction of peak amplitude errors are ignored and the reconstruction is reduced to band-pass filtering. However, when we evaluate the downsampling correction factor  $\kappa$  of Equ. (2.20) for the frequency range used in [70], which is from 50 Hz to 7 kHz, we recognize a difference in channel weighting of 43 dB. Such an amplification difference cannot be ignored and, therefore, we expect that the coders in [70, 71] produce a substantial signal distortion.

### 2.3.2 Synthesis Filterbank

The last stage is the synthesis filter bank, which should be an inverse of the analysis filterbank. For proper signal reconstruction from the pulse representation, it is essential that the synthesis filters have band-pass characteristics to eliminate aliasing. This also keeps the quantization noise within a local frequency range.

In general, the inverse operator for a non-decimated (i.e., oversampled), invertible filterbank is not unique. A natural method for inverting a non-decimated FIR

filterbank is based on the following condition for perfect reconstruction<sup>3</sup>

$$G_k(z) = \frac{H_k(z^{-1})}{\sum_{i=0}^{K-1} H_i(z)H_i(z^{-1})}. \quad (2.23)$$

For the case that  $\sum_{i=0}^{K-1} H_i(z)H_i(z^{-1}) = 1$ , the synthesis filterbank is the analysis filterbank with time-reversed impulse responses. A delay equal to the length of the analysis filters minus one is needed to make the synthesis filterbank causal. Using long FIR filters, as discussed in section 2.2, causes long delays.

In the general case, when the denominator of Equ. (2.23) is not equal to 1 (e.g., when a low number of auditory channels is used), accurate signal reconstruction can be obtained by an additional linear-phase equalization filter (see [26]) that operates on the sum of all channels synthesized with  $G_k(z) = H_k(z^{-1})$ . This equalizer reduces the remaining magnitude ripple and has to be designed to approximate the frequency response

$$H_{equ}(e^{j\theta}) \stackrel{!}{=} \left[ \sum_k |H_k(e^{j\theta})|^2 \right]^{-1}. \quad (2.24)$$

The magnitude ripple decreases with increasing order of the FIR equalizer. However, an additional delay of half the filter order is introduced. Thus, for the choice of the impulse response length, a suitable compromise must be found. The minimum delay solution without equalization has been used in [64]. We found that, for 20 channels and a sampling rate of 8 kHz, the ripple is about 4 dB. The ripple decreases with a further increase of the number of channels.

As already mentioned for the analysis filters, FIR gammatone filter implementations are memory-consuming. Although the synthesis filters can use the same coefficients as used for the analysis filters, separate ring buffers are needed for every auditory channel in the synthesis filterbank. Consequently, the necessary amount of memory is doubled. For an accurate FIR gammatone filterbank implementation with long impulse responses, the memory of most currently used DSPs is not sufficient. One solution for this problem is to take shorter impulse responses and accept deviations from the ideal frequency responses. In chapter 5 we propose a windowing scheme that can be used to obtain shorter impulse responses and yet accurate frequency responses. Another possibility is to consider alternative filterbank implementations as described in chapter 5.

---

<sup>3</sup>Here, perfect reconstruction refers to processing of the input signal by the analysis and the synthesis filterbank only.

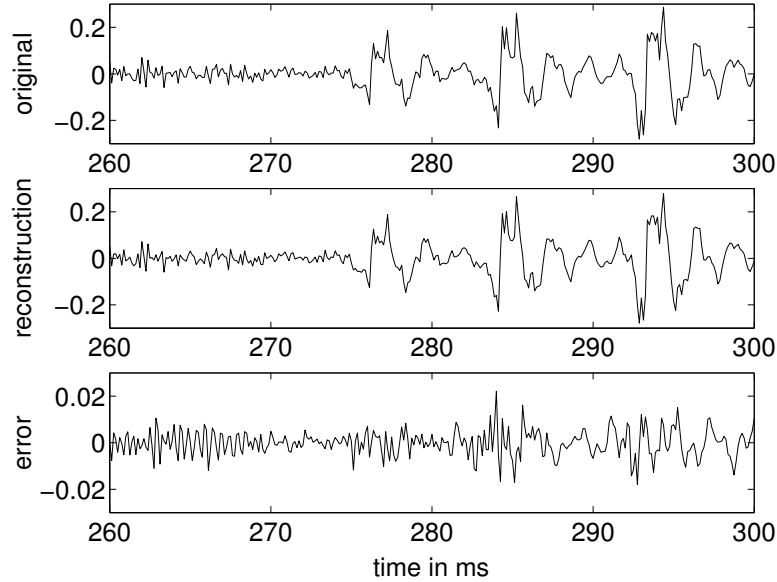


Figure 2.7: Comparison of the original waveform (upper plot), the waveform reconstructed from the auditory pulse representation (middle plot), and the reconstruction error (lower plot, note the finer amplitude scale). Speech segment taken from ‘The source’, spoken by a male speaker.

### 2.3.3 Simulation Results

In Fig. 2.7 a segment of the original waveform with 8 kHz sampling rate is compared with the output of our inverse auditory model with 20 channels. For this simulation, the FIR gammatone filterbank according to Equ. (2.8) is used where the lowest center frequency is  $f_{c,0} = f_{min} = 100$  Hz (order of filter 666) and the highest  $f_{c,19} = f_{max} = 3600$  Hz. The center frequencies of the in-between channels are calculated according to Equ. (2.13), which results in a spacing between the channels of 1.2 ERBs. The auditory representation is left unquantized and uncompressed. The delay of the analysis-synthesis filterbank pair is 83.25 ms. The output of the synthesis filterbank is passed through a linear-phase equalizer<sup>4</sup> with a group delay of 25 ms. Although the segmental<sup>5</sup> signal-to-noise ratio (SegSNR) is typically only between 18 dB and 25 dB, the reconstructed signal is largely without audible distortion. For a subjective evaluation of the reconstruction quality, see section 3.5.2.2.

<sup>4</sup>The equalizer filter is designed using least-squares approximation.

<sup>5</sup>The used segment length is 20 ms.

### 2.3.4 Frame-Theoretic Interpretation of Auditory Synthesis

It is useful to consider the auditory resynthesis from the perspective of *frame theory*. This endorses our choice of synthesis filterbank of section 2.3.2 and provides a bound for the reconstruction error introduced by the analysis-synthesis filterbank pair. Furthermore, it justifies our simple method to reconstruct the signal from the pulse representation and allows us to reduce the number of pulses in the auditory representation. For a compact introduction in frame theory refer to [72, 73].

In practical implementations of the filterbank structure, the analysis and the synthesis filterbank are identical except for a time-reversal of the impulse responses. We first evaluate the validity and implications of this choice. The analysis filterbank maps the input sequence<sup>6</sup>  $x$  to a set of channel sequences, one for each filter. It is essential that the analysis filterbank is invertible and that means it can be interpreted as a frame operator. The operation of the analysis filterbank can be written as  $Fx$  with  $F$  the analysis operator that is defined using a set of inner products  $(Fx)[m] = \sum_{n \in \mathbb{Z}} \psi_m^*[n]x[n]$  with the set of sequences  $\{\psi_m\}_{m \in \mathbb{Z}}$ . Each sequence  $\psi_m$  is a translate of one of the  $K$  time-reversed<sup>7</sup> impulse responses. For notational simplicity, the index  $m$  enumerates the different translates and the different channels (e.g., in an interleaved manner). Invertibility of the filterbank is guaranteed if the frame condition is satisfied:

$$A \sum_{n \in \mathbb{Z}} |x[n]|^2 \leq \sum_{m \in \mathbb{Z}} |(Fx)[m]|^2 \leq B \sum_{n \in \mathbb{Z}} |x[n]|^2 \quad \forall x \in \ell^2(\mathbb{Z}), \quad (2.25)$$

where  $A$  and  $B$  are finite, positive, scalar frame bounds. When the frame condition is fulfilled, the set of sequences  $\{\psi_m\}_{m \in \mathbb{Z}}$  is called a frame. More about the frame bounds and how they can be calculated will be discussed later.

In general, the inverse frame (usually called the ‘dual frame’ and means the filters of the synthesis filterbank) is not unique<sup>8</sup>. We are interested in an inverse that is easy to compute and, what is equally important, that minimizes the effect of quantization errors in  $(Fx)[m]$  on the reconstruction. The so-called *frame algorithm* [72, 73] is an iterative procedure to reconstruct the signal from  $y[m] = (Fx)[m]$ , and it minimizes the effect of quantization errors. Using unit impulses instead of the actual output of the analysis filterbank, the frame algorithm renders the dual frame. The first iteration often provides a useful approximation to the inverse or even the exact inverse. The estimate  $\hat{x}^{(i)}$  of  $x$  at iteration  $i$  of the frame algorithm is

$$\hat{x}^{(i)} = \rho F^* y + (I - \rho F^* F) \hat{x}^{(i-1)}, \quad (2.26)$$

<sup>6</sup>We assume that the input sequence is in the Hilbert space  $\ell^2(\mathbb{Z})$ .

<sup>7</sup>In order to express convolution by an inner product.

<sup>8</sup>The inverse frame is unique if, and only if, the frame forms a basis of the signal space.

where  $\rho$  is a scalar relaxation parameter,  $I$  is the identity operator,  $\hat{x}^{(0)}$  is initialized as a zero sequence, and  $F^*$  is the adjoint analysis operator that maps the  $K$ -channel signal,  $y[m]$ , to a single-channel signal  $(F^*y)[n] = \sum_{m \in \mathbb{Z}} y[m]\psi_m[n]$ . The estimation error at iteration  $i$  is then

$$x - \hat{x}^{(i)} = (I - \rho F^*F)(x - \hat{x}^{(i-1)}) = (I - \rho F^*F)^i x. \quad (2.27)$$

With the optimal selection  $\rho = \frac{2}{B+A}$ , the error is bounded by

$$\begin{aligned} \|x - \hat{x}^{(i)}\| &= \min_{\rho} \|(I - \rho F^*F)^i x\| \\ &\leq \min_{\rho} \max(|1 - \rho A|, |1 - \rho B|)^i \|x\| \\ &= \left(\frac{B - A}{B + A}\right)^i \|x\|. \end{aligned} \quad (2.28)$$

The values  $A$  and  $B$  form the minimum and maximum eigenvalues of the operator  $F^*F$  (i.e., the frame operator), which are precisely the frame bounds of Equ. (2.25).

The first-iteration estimate of  $x[n]$  by the frame algorithm is the expansion  $\rho(F^*y)[n] = \rho \sum_{m \in \mathbb{Z}} y[m]\psi_m[n]$ , which implies that  $\rho F^*$  is the approximation to the inverse operator. It can easily be seen that this corresponds to a synthesis filterbank with impulse responses that are the time-reversed impulse responses of the analysis filterbank, scaled by  $\rho$ . Moreover, we see from Equ. (2.28) that the relative error is bounded by the factor  $\frac{B-A}{B+A}$ . When  $A = B$ , which corresponds to a ‘tight frame’, the first-iteration estimate is exact.

For the cascade of a non-decimated analysis-synthesis filterbank pair, we get  $\hat{x} = \rho F^*F x$ . The discrete-time Fourier transform (which is unitary) simplifies the analysis of the operator  $F^*F$ . In the Fourier domain, the frame operator  $F^*F$  corresponds to [74]

$$\mathcal{F}F^*F\mathcal{F}^{-1} = \sum_{k=0}^{K-1} H_k(e^{j\theta})H_k(e^{-j\theta}), \quad (2.29)$$

where  $\mathcal{F}$  denotes the discrete-time Fourier transform operator. This immediately leads to the inversion formula given in Equ. (2.23) and to the required equalizer frequency response in Equ. (2.24). The same Fourier-domain equivalence shows that the frame bounds then correspond to the essential infimum and supremum of  $\sum_{k=0}^{K-1} H_k(e^{j\theta})H_k(e^{-j\theta})$ .

We can now draw some conclusions for our auditory filterbank based on the frame-theoretical viewpoint. First, the synthesis filterbank based on time-reversing the impulse responses is an approximation to the perfect synthesis filterbank that has minimum sensitivity to quantization errors in the perceptual domain. Second,

the accuracy of this approximation is governed by the relative error  $\frac{B-A}{B+A}$ , where  $A$  and  $B$  can be evaluated as the essential infimum and supremum of the summed responses of the analysis filterbank. For an auditory filterbank implementation based on FIR gammatone filters, the relative error  $\frac{B-A}{B+A}$  is  $-30.7$  dB for 50 channels and  $-5.9$  dB for 20 channels.

Frame theory can also be applied to provide an interpretation of the peak-picking procedure used in our auditory model. It is convenient to look at a single channel first. A frame algorithm that can be used for the reconstruction of continuous low-pass band-limited signals from irregularly spaced samples and their derivatives was presented in [75]. In this case the frame is formed by the translates of the impulse response of an ideal low-pass filter and its derivatives. For our case, the first-order derivative of the selected signal samples is always zero and the reconstruction method is essentially identical to the reconstruction applicable if no derivative is given. However, reconstruction is possible with a larger spacing between the samples than if no information was known about the derivatives (a factor of two for regularly spaced samples). In practice, the first iteration of the frame algorithm consists of ideal low-pass filtering the upsampled (inserting zeros), weighted signal. The weighting of each sample is linear with the distance to the previous sample as already discussed for the adaptive correction factor at the end of section 2.3.1.1. Nearly uniform spacing, as we have in our case, results in nearly uniform weighting, reducing the first iteration of the frame algorithm essentially to a low-pass filter. Moreover, it is easy to see that the frame is tight for the regular sampling case, which means that the first iteration renders the exact inverse.

We note that the frame algorithm of [75] assumes a band-limited signal and a sample spacing that is at most  $2\pi/\theta_m$  for a band-limitation of  $\theta_m$  (in practice the band-limitation is somewhat less). Since the outputs of the auditory filters resemble sinusoids, and since a sinusoid of frequency  $\theta_s$  has its maxima spaced at  $2\pi/\theta_s$ , this implies that the frame algorithm of [75] does not apply to our case without modification. The required modification consists of replacing the impulse response of the ideal low-pass filter by the impulse response of an ideal band-pass filter<sup>9</sup>. For regularly spaced samples the reconstruction algorithm then consists of a simple band-pass filtering. For irregular spacing, the samples must first be weighted appropriately.

In practice, the band-pass filtering operation required for the reconstruction of each of the irregularly sampled channels can be effected by the corresponding synthesis filter within the inverse of the basilar membrane filterbank. In our practical implementation, we then make the following approximations with respect to

---

<sup>9</sup>We note that, in general, sampling rates that are sufficient for low-pass signals may not be so for band-pass signals of identical bandwidth, e.g., [76]. However, this aliasing problem is unlikely to occur for spectra that essentially consist of a single line.

inverting the peak-picking procedure: (i) we use the first iteration of the frame algorithm, which is not exact since the frame is not tight for irregular sampling; (ii) we neglect the sample weightings that are needed to account for irregular sampling; (iii) we assume that the inverse basilar membrane filters, which have narrow-band character, are sufficient to replace the ideal band-pass filters. To quantify the error on the reconstructed signal proves to be difficult since the peak-picking operation is signal dependent. After the peak picker, the signal representation is based on the subframe  $\{\psi_m\}_{m \in \mathbb{J}}$  with  $\mathbb{J}$  the signal-dependently selected subset of  $\mathbb{Z}$ . A direct computation of the frame bounds of this subframe is not possible. The perceptual effect of the three mentioned approximations on auditory synthesis is small as is confirmed by the results provided in section 2.3.3 and in section 3.5.2.2.

The frame interpretation leads directly to a method to reduce the information rate of our basic model. Particularly for the filters of the basilar membrane filterbank with high center frequency, the peak-picking procedure leads to a high rate of peaks. Since the peak locations and amplitudes must be encoded as side information, the resulting parameterization is not an efficient basis for coding. However, we note that the frame-algorithm based reconstruction from irregularly spaced samples as described in [75] only requires that the peaks are not separated by more than a given distance. Importantly, there is no requirement to include all peaks of the signal. As a result, we can downsample the peak sequence in the channels with higher center frequency by a significant factor without losing the ability to reconstruct the signal. The next chapter deals with this downsampling. The amount of downsampling that can be applied to a peak sequence is constrained by the bandwidth of the ideal band-pass filter of the frame. With increasing downsampling of the peak sequence the importance of the sample weighting increases, and it can then not be omitted from the synthesis structure. It is interesting to note that this frame-theoretical viewpoint leads to a new interpretation of the results obtained by [70]. In [70] downsampling of the peak sequence was justified from an auditory modeling argument, which is not physiologically plausible for the auditory representation.



# Chapter 3

## Auditory-Domain Sparsification

The invertible auditory model proposed in the previous chapter allows to resynthesize the input signal with high quality and provides a basis for coding audio signals in the perceptual domain. This chapter describes approaches to reduce the number of firing pulses needed to encode the signal in this domain. We first review a previous approach to sparsify the auditory representation based on a model for temporal forward masking. In section 3.2 we incorporate the final listener into the flow graph of a coding scenario and thus introduce the transmultiplexer point of view to perceptual coding. Furthermore, in subsection 3.2.1 we use digital communication theory to show that the previously chosen synthesis filterbank optimally matches the final receiver. We then show in subsection 3.2.2 that locally dominant pulses in the auditory representation play a constitutive role. Based on the basilar membrane (BM) excitation caused by an isolated pulse, we present a new masking model for both simultaneous and temporal masking in section 3.3. The masking model is used to decide whether a pulse of the auditory representation is needed or not. Since the elimination of pulses removes signal energy, the amplitudes of the sparsified pulse representation need to be corrected. We present a new pulse amplitude correction scheme in section 3.4. Section 3.5 reports on experiments using the proposed masking model and the new amplitude correction method. Furthermore, we compare the decisions of our masking model with psychoacoustical data in subsection 3.5.2. Finally, section 3.6 discusses the incorporation of nonlinear models for the adaptation to the stimulus into our coding method.

### 3.1 How sparse can we make the auditory representation for coding?

The auditory representation provided by our model as presented in chapter 2 is sparse, consisting mostly of zeros. However, it contains more firing pulses in

total than the original input signal has samples (about three times more for the 20-channel case and a sampling rate of 8 kHz). It is natural for sparse data to encode only the non-zero elements by additionally specifying their indices as side information. We apply this method to our firing pulse trains and refer to the pulse locations as the side information even when it makes up more data than the pulse amplitudes.

In [70], a coder based on the physiologically motivated signal representation described in the previous chapter has been proposed, where two separate masking models were added to reduce the overall number of firing pulses: a model of simultaneous masking and a model of temporal masking. For simultaneous masking, a model similar to that of MPEG [6] was used to compute the masking threshold. To model temporal masking, an exponentially decaying forward masking threshold was proposed (see below). It was shown that the consideration of simultaneous masking does not bring a remarkable reduction of the pulses, whereas exploiting temporal masking does. Our own experiments with the model for temporal post-masking adopted from [70] show that an average reduction in the number of pulses by about 50% for 16 kHz-sampled speech does not affect the audible quality of the reconstructed signal [77]. For this model, a temporary masking threshold is computed in each channel. Let  $x[n]$  be the firing pulse train of one auditory channel and let  $T[n]$  be the corresponding masking threshold. Then  $T[n]$  is defined in [70] as

$$T[n] = \begin{cases} x[n], & x[n] > T[n-1]e^{-1/\tau} \\ T[n-1]e^{-1/\tau}, & \text{otherwise.} \end{cases} \quad (3.1)$$

The time constant  $\tau$  is set to 125 samples (7.8 ms) for the lowest-frequency channel and to 33 samples (2 ms) for the highest (according to the empirically determined values from [70]). The authors argue to approximate Zwicker's post-masking threshold [78] in this way. Once the threshold is computed, the output signal of the masking stage is

$$y[n] = \begin{cases} x[n], & x[n] > T[n-1]e^{-1/\tau} \\ 0, & \text{otherwise.} \end{cases} \quad (3.2)$$

The pulse rate is higher in high-frequency channels<sup>1</sup> than in low-frequency channels. Consequently, the reduction in the pulse rate is strongest in high-frequency channels. This finding is in accordance with the frame-theoretic consideration of section 2.3.4. The amount of encoded data should be consistent with the bandwidth of an auditory channel instead of the channel's center frequency.

We have to further reduce the number of pulses significantly, particularly in higher-frequency channels, to achieve a better compression. In our recent work

---

<sup>1</sup>The average number of pulses per second in an auditory channel can be predicted by the channel's center frequency.

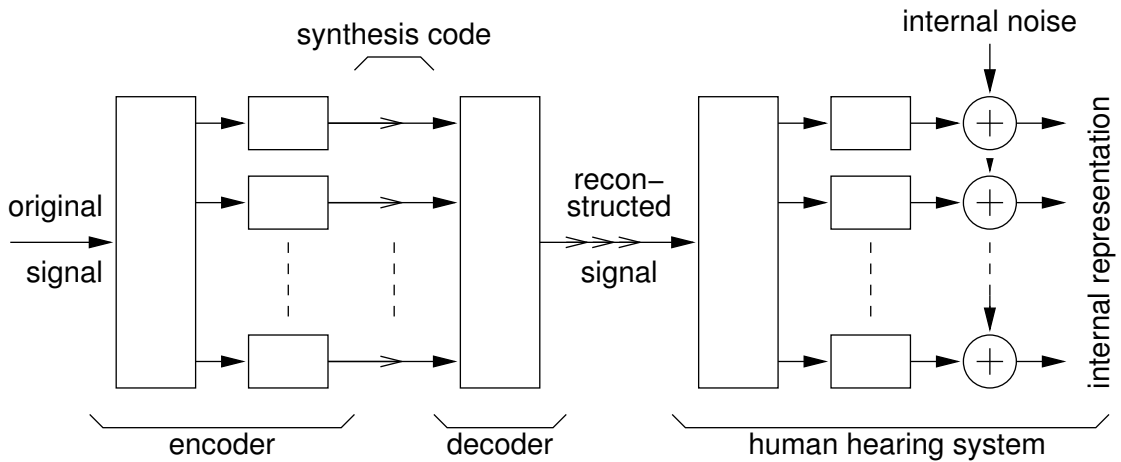


Figure 3.1: The complete signal flow: encoder  $\rightarrow$  decoder  $\rightarrow$  human listener.

[79] we incorporated a combined model for both simultaneous and temporal masking. In section 3.2 we consider the decoder together with the final listener as a transmultiplexer system. Using this transmultiplexer, we derive our new masking model, which is described in section 3.3. Together with another amplitude correction step, which compensates for the loss of energy due to the elimination of pulses, we are able to omit more than 70% of the original pulses of speech signals sampled at both 8 kHz and 16 kHz without noticeably degrading the perceived quality. This result is a step further towards an efficient compression method since it reduces the amount of side information considerably. The amplitude correction method is described in section 3.4. We performed experiments as well as listening tests to show the quality of our masking model. Finally, section 3.5 presents the results of these experiments and listening tests.

## 3.2 The Transmultiplexer View of Perceptual-Domain Coding

To reduce the amount of data, we first want to omit pulses. For this purpose, we should ask how much a single pulse of the encoder's output contributes to the final, actually perceived neural code in a human listener when listening to the reconstructed signal. Then, we should be able to decide whether the pulse is important and, therefore, needed for the synthesis code or not.

In Fig. 3.1 the entire signal flow graph is illustrated, which consists of the encoder, the decoder, and the final receiver—in our case a human listener. The

flow graph omits the influence of the listening environment (e.g., room reflections and background noise) since this is not known a-priori<sup>2</sup>. The aim is to provide the listener with the same perception (or sensation) as the reconstruction from the complete pulse code would do. This can be achieved by exploiting the limited capabilities of perception, which have undoubtedly been shown in psychoacoustical experiments such as just noticeable differences (JNDs), masking thresholds, or absolute thresholds [33]. It is common practice in perceptual coding to utilize the fact that two (slightly) different stimuli (e.g., the original and the quantized signal) can produce the same perception.

In many auditory models (e.g. [19, 80, 81]) the inability of humans to distinguish acoustic stimuli with small differences is modeled by additive noise (‘internal noise’) on the auditory representation (‘internal representation’). This is shown in Fig. 3.1. Usually the internal noise is modeled as independent Gaussian noise with zero mean and a variance that is in accordance with the experimentally found threshold and its probability of being correct<sup>3</sup> (see [19] for more details).

We introduce a second auditory model to represent the human listener in the flow graph of Fig. 3.1, which can perceptually be more accurate than the one used in the encoder (if a more accurate model was used for the encoder, the signal reconstruction could be difficult or even impossible). In this way we introduce a domain where perceptual differences can be detected more accurately (even if the synthesis code is not an auditory representation). This is similar to MPEG audio coders, in which the auditory model operates in a parallel signal path. In contrast to MPEG, our auditory model, which represents the final listener, uses the reconstructed signal as its input. Thus, we are able to directly compare the outputs of the listener model for the signal synthesized from a reduced auditory code and for the signal synthesized from the complete auditory code.

If we consider only the decoder of Fig. 3.1 with the synthesis pulse code as input and the human listener with the actually perceived neural code as output, we get a multi-input, multi-output (MIMO) system generally known as *transmultiplexer*. This is illustrated in Fig. 3.2.

### 3.2.1 Matched-Filter Communication Scenario

The transmultiplexer, which consists of the decoder’s synthesis filterbank and a model of the human listener’s BM, resembles a multiple-access communication channel [82]. The synthesis filters can be seen as pulse-shaping filters and the BM analysis filters correspond to the *matched filters* of the respective channels. In contrast to a frequency-division multiplexed communication system, neighboring

---

<sup>2</sup>The considered flow graph is accurate for headphone listening.

<sup>3</sup>The probability of being correct depends on the method used in the psychoacoustic experiment.

frequency bands in the transmultiplexer do overlap. Contrary to a code-division multiplexed scenario, our pulse-shaping filters are not orthogonal to each other, either. While in digital communication problems usually the design of an optimal receiver for a given transmission scenario is considered, in our case the receiver, which is a human listener, is fixed and does not give any degree of freedom for the coder design. To respect this fact, we should not start with the source (i.e., a speech or audio signal) or the source encoder to design a new perceptual coding method. We should rather start with the final receiver and perform a sink-to-source design. In this subsection we start at the sink and thus show the optimality of the chosen synthesis filterbank with time-reversed gammatone impulse responses.

We model the human listener's BM by a linear gammatone filterbank and the IHCs as non-uniform samplers that detect temporal local maxima. The resulting fixed receiver can thus be seen as a bank of receivers each with a different gammatone filter as the so-called receive filter and a maximum sampler. These receivers represent typical receivers for pulse-amplitude modulation (PAM) [82] but are extended to the case of non-uniform symbol intervals. Consider an isolated amplitude-modulated pulse, a given transmit filter (or pulse-shaping filter), and a delay-free non-dispersive channel with additive noise. For this case it can easily be shown that the optimum receive filter has an impulse response that is a time-reversed version of the impulse response of the transmit filter normalized by its squared norm [82]. Optimality is defined here with respect to decoding the modulated amplitude with a minimum distance. Such a receive filter is called a *matched filter*. Matched filters can be found as front ends in virtually all state-of-the-art receivers. The scenario in our transmultiplexer is even simpler: instead of the communication channel we have a direct, noise-free connection<sup>4</sup> between transmit and receive filter and are therefore able to observe the exact amplitude of the isolated pulse<sup>5</sup>.

When a sequence of amplitude-modulated pulses is transmitted, neighboring shaped pulses may overlap and cause intersymbol interference (ISI) [82]. Since our encoder outputs pulse trains with mean pulse distances close to the inverse of the corresponding channel's center frequency for typical speech or audio input signals, neighboring pulses interfere strongly. On the other hand, ISI is not critical because consecutive pulse amplitudes are marginally different due to the limited bandwidth of the auditory filters. In practice, we do not need to consider a pulse code with rapidly time-varying pulse amplitudes. For the envisaged case when we omit redundant pulses, we generate a scenario with relatively isolated pulses and thus can ignore ISI. But we also have to keep in mind that the demodulated amplitudes are then smaller than the original ones due to the missing constructive interference.

---

<sup>4</sup>In the case of headphone listening.

<sup>5</sup>When the normalization by the squared norm of the impulse response is considered.

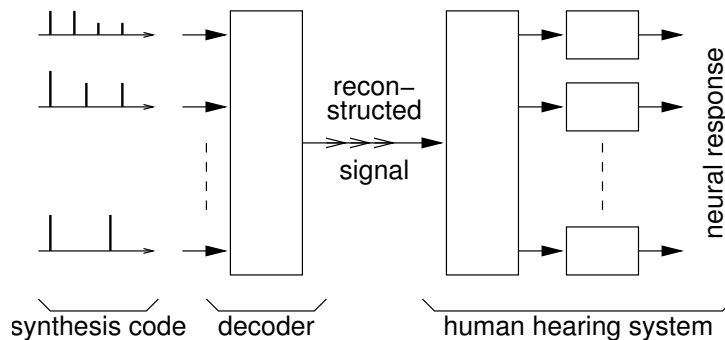


Figure 3.2: The transmultiplexer view of perceptual-domain coding.

Therefore, the deletion of pulses requires a correction of the amplitudes of the remaining pulses. Section 3.4 deals with this issue in more detail.

When we consider the whole transmultiplexer with a bank of matched filters and an isolated pulse at a certain auditory channel's input, we can observe the maximum output signal in the same channel again. Any input signal at the analysis filterbank that differs from the considered channel's time-reversed impulse response, but has the same norm, produces a lower output signal. Thus, we can easily detect the matching channel. For the case of multiple pulse trains, which co-exist in different channels, an interference between pulses of neighboring channels happens in addition to the interference between temporally neighboring pulses. In the context of a multiple-access communication system we usually talk about multi-user interference instead of interchannel interference. For the pulse code generated by our encoder, this interference is usually not destructive due to the similarity of neighboring pulse trains.

Simulations show that locally dominant pulses of the redundant code at the input of the transmultiplexer can be found at the same time-channel positions at the output again (except of the overall delay), whereas the positions and even the amplitudes of less dominant pulses change considerably. The matched filter pairs in our transmultiplexer ensure that the inner hair cells in a cochlea detect the same dominant local maxima as given by the synthesis pulse code. In the next subsection we present examples to underline the importance of locally dominant pulses.

### 3.2.2 The Role of Dominant Pulses

For coding applications, generally orthogonal signal bases are preferred to ensure that signal components stay independent and do not interfere with each other. Using orthogonal bases and transforms does not create redundancy and, therefore,

enables efficient coding. In our approach, we start with the fixed final receiver, which has non-orthogonal channels. For this receiver, digital communication theory suggests us to use a non-orthogonal filterbank also for the decoder's synthesis filterbank (see previous subsection). The non-orthogonal impulse responses or rather the overlapping frequency bands of neighboring auditory channels are not necessarily a drawback. This fact just reflects that signals are spread on the BM. Since the final receiver spreads the incoming signal, we should be allowed to modify the pulse code obtained by our encoder (which is also a spread auditory representation) in terms of focusing it without generating considerable errors in the final internal representation. One possibility to focus the pulse code is to omit the additional pulses caused by spreading in the analysis filterbank of the encoder and thus, to decrease the redundancy of the auditory representation. When we assume that our auditory model (or our encoder) is accurate, we know what the finally received pulse trains (i.e., the internal representation) should look like. But what should a redundancy-reduced synthesis code at the transmultiplexer's input look like?

We present an introductory example to emphasize the important role of dominant pulses. Assume the signal to be encoded is the time-reversed impulse response of a gammatone filter used in channel  $k$  of the auditory filterbank. The pulse representation of the signal obtained from our encoder shows a clear maximum in channel  $k$ . In addition to this pulse with maximum amplitude we can observe a vast number of other pulses with lower amplitudes. This is visualized in Fig. 3.3 where the channel  $k = 4$ . The optimum synthesis code at the input of the transmultiplexer of Fig. 3.2 that generates a neural response in the final listener that is equal to the just described pulse code of Fig. 3.3 consists only of a single pulse in channel  $k$ , which is the dominant pulse. All additional pulses obtained by our encoder need not be used for the synthesis code. For the case considered here, eliminating all extra pulses caused by spreading by the BM analysis filterbank does not at all generate a distortion on the final neural response, but even reduces the reconstruction error to zero<sup>6</sup>. The pulse code in Fig. 3.3 is equivalent to the neural response (predicted by the same auditory model as used in the encoder) when only one pulse from channel 4 is synthesized.

We consider another example and assume the signal to be a sinusoid. According to our model, we obtain pulse trains with the same fundamental frequency as the sinusoidal frequency in all channels. In the channel with center frequency closest to the sinusoidal frequency we can observe the highest pulse amplitudes. The amplitudes in all other channels are lower according to the magnitude of the frequency response of the auditory filters. When we omit all pulse trains with lower amplitudes and resynthesize only the dominant one, we can reconstruct the

---

<sup>6</sup>A constant gain factor has to be considered

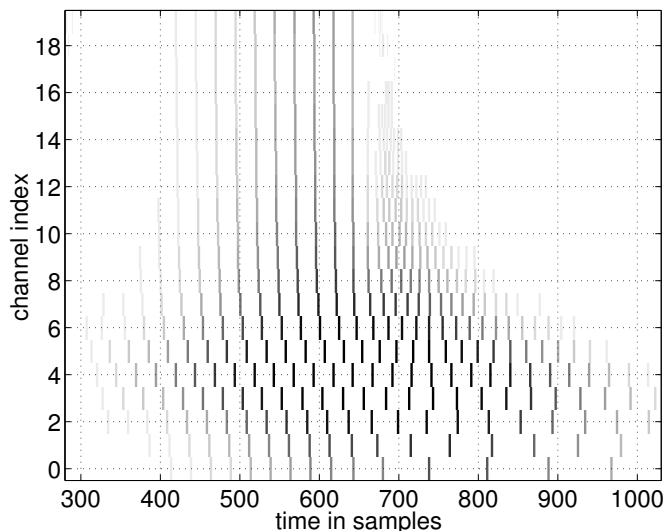


Figure 3.3: The pulse code obtained by our model for a time-reversed impulse response of channel 4 of the gammatone filterbank as the input signal or equivalently, the transmultiplexer’s (simulated) neural response when synthesizing a single impulse of channel 4.

sinusoid sufficiently well. The sinusoidal amplitude might be slightly attenuated on account of a possible damping of the considered channel and the missing contributions of the other channels. It should also be mentioned, that a nonlinear distortion arises due to an imperfect suppression of harmonics of the pulse train (see also section 2.3). However, this problem does not increase when we omit neighboring pulse trains of the original code since these would also contribute to the undesired harmonics.

The shown examples demonstrate that signals can be encoded by considering only locally dominant pulses. Based on this observation, a simple method to eliminate pulses has been proposed in [79] and will be described in more detail in the next section.

The transmultiplexer point of view provides an analysis-by-synthesis framework. It simplifies the optimization of the synthesis code generated by the encoder in terms of minimizing the number of pulses needed at the input of the decoder (or synthesizer) to produce a proper perception. The present problem is similar to finding the optimum multi-pulse excitation signal in linear prediction coding (MPLP) [83] but extended to a multi-channel signal and with implicit perceptual weighting. Another difference is that in MPLP the search starts with an empty signal and pulses are added iteratively while in our problem we start with an overcomplete pulse representation and remove pulses.

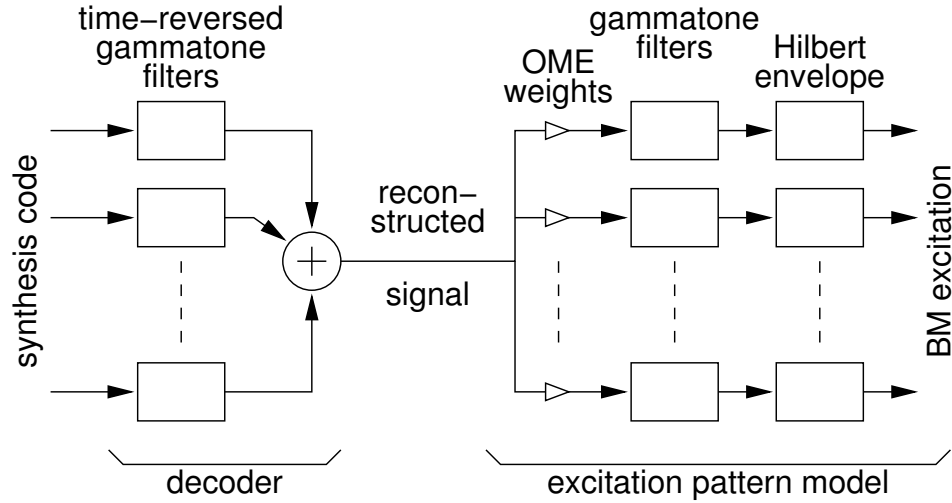


Figure 3.4: Simple excitation pattern model as a replacement for the human listener in the transmultiplexer.

### 3.3 A New Model for Simultaneous and Temporal Masking

#### 3.3.1 Auditory Excitation Pattern Model

Since we simply want to eliminate pulses and do not consider the quantization of pulse locations here (i.e., the temporal fine structure of the auditory representation is preserved) an auditory excitation pattern model is sufficient to model the final human listener. The output of an excitation pattern model is usually the set of envelopes (or average powers<sup>7</sup> [34, 33]) of the band-pass signals from an auditory filterbank. In Fig. 3.4 the human listener has been replaced by a simple excitation pattern model. For the experiments, we use again the same linear FIR gammatone filterbank with 20 channels (for 8 kHz sampling rate) as applied in the encoder for modeling the BM. Additionally, we consider a weighting of the channels according to an inverse 100-phon equal-loudness contour [33], which models the transfer function of the outer and the middle ear (OME). In appendix A more details on the actual implementation of the OME weights are shown. Our simple BM excitation model extracts the Hilbert envelopes in all auditory channels, but, in contrast to classical excitation pattern models, we do not apply a final low-pass filter to get a long-term integrated measure.

<sup>7</sup>Past experiments with classical excitation pattern models have been performed with stationary stimuli mainly.

### 3.3.2 Is exhaustive search feasible?

Usually, analysis-by-synthesis schemes lead to exhaustive search algorithms. Exhaustive search means to try all possible signal representations of the considered coding scheme (e.g., all entries of a vector quantizer's code book) and select the one that minimizes the error of the resynthesis. For the case of searching for an optimal sparse, pulsed signal representation as input for our transmultiplexer, exhaustive search means to try all possible pulse configurations and select the one that produces an transmultiplexer output closest to the one produced by the complete auditory representation. Let  $\mathbb{J}$  denote the set of all pulses of the original auditory representation (i.e., the set of all channel-time locations) of a considered signal segment. The cardinality of  $\mathbb{J}$ , i.e., the number of pulses in the considered segment, is denoted as  $|\mathbb{J}|$ . Since the intention is to minimize the number of pulses, the possible pulse configurations, which we denote as  $\mathbb{S}_i$  with  $i = 0, \dots, N_p - 1$  where  $N_p$  is the number of possible configurations, are all possible subsets of  $\mathbb{J}$ :  $\mathbb{S}_i \subset \mathbb{J}$ . For a fixed pulse rate coder (i.e., the cardinality of  $\mathbb{S}_i$  is given), the number of possible configurations is  $N_p = \binom{|\mathbb{J}|}{|\mathbb{S}_i|}$ . For a variable pulse rate coder, a threshold on the error has to be given. In this case the search is even more complex since also all possible cardinalities have to be considered in general. The number of all possible configurations is  $N_p = \binom{|\mathbb{J}|}{1} + \binom{|\mathbb{J}|}{2} + \dots + \binom{|\mathbb{J}|}{|\mathbb{J}|-1} + \binom{|\mathbb{J}|}{|\mathbb{J}|}$ . However, some configurations are not likely to happen (such as a single pulse only) and can therefore be ignored. To demonstrate that the number of possible configurations reaches unpractically high values, we consider a realistic example with a narrow-band coder setup ( $f_s = 8000$  Hz) that produces  $3 \cdot f_s$  pulses per second after the peak-picking procedure. For a desired segment length of 10 ms and a desired pulse rate of  $1.2 \cdot f_s$  pulses per second after the sparsification (i.e., fixed pulse rate coding and a reduction by 60%), we get  $N_p = \binom{240}{96} \approx 10^{68}$  possible combinations. For a desired segment length of 5 ms we still get  $N_p \approx 10^{33}$ . The presented figures show that an exhaustive search algorithm is not feasible for the sparsification process. The next sections present an alternative method that avoids the exhaustive search.

### 3.3.3 Isolated-Pulse BM Excitation Patterns

In section 3.2.2 the important role of locally dominant pulses has been discussed. Keeping this in mind and also considering that the aim is to find a redundancy-reduced sparse pulse code, which consists of relatively isolated pulses, our masking model treats all pulses independently (*isolated-pulse BM excitation model*). In other words, we do not accumulate an overall excitation pattern produced by more than one pulse. Instead, we compute off-line  $K$  unit excitation patterns for the  $K$  channels of the subband coder and store them in memory. Treating all pulses independently also simplifies the search and substantially saves computational load.

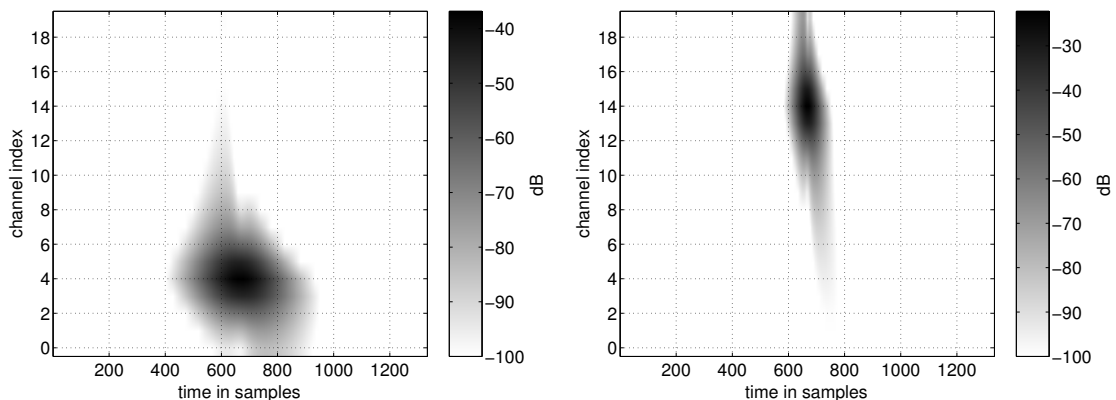


Figure 3.5: BM excitation of a single unit pulse. Left: pulse from channel 4, right: pulse from channel 14. The unit of the time axis is samples at a rate of 8 kHz.

Let us consider a single unit impulse at the transmultiplexer’s input channel  $ch$ . The resulting unit excitation pattern is computed once according to Fig. 3.4 by convolving the impulse response of the corresponding channel’s synthesis filter  $g_{ch}[n]$  with all OME-weighted impulse responses of the listener model’s analysis filterbank  $\varrho_k h_k[n]$  ( $\varrho_k$  denotes the OME weighting) and generating the Hilbert envelopes. We refer to this pattern as  $E_{ch}[n, k]$

$$E_{ch}[n, k] = \mathcal{ENV}\{g_{ch}[n] * (\varrho_k h_k[n])\} \quad (3.3)$$

where  $n$  is time in samples and  $k$  is the output channel index. The operator  $*$  denotes convolution and the operator  $\mathcal{ENV}\{\}$  extracts the Hilbert envelope, which is defined as

$$\mathcal{ENV}\{x[n]\} = |x[n] + j\mathcal{H}\{x[n]\}| \quad (3.4)$$

with  $\mathcal{H}\{x[n]\}$  the Hilbert transform of  $x[n]$ .

In Fig. 3.5 two different unit excitation patterns are plotted, on the left the BM excitation of a unit impulse synthesized from channel 4 and on the right a unit impulse synthesized from channel 14. We can consider such a pattern as an impulse response of the transmultiplexer<sup>8</sup>. Additionally, we call the maximum excitation  $\hat{E}_{ch} = \max_{n,k} E_{ch}[n, k]$ . In our case, where the impulse responses of synthesis and analysis filters are time-reversed (and shifted) versions of each other ( $g_k[n] = h_k[N_{gt} - 1 - n]$  with  $N_{gt}$  the length of the impulse response), the maximum occurs in channel  $k = ch$  delayed by the length  $N_{gt}$  minus 1. Since all used impulse responses have the same length  $N_{gt}$  (667 samples for our simulations with

<sup>8</sup>We should mention that our transmultiplexer is a nonlinear system since it extracts the signal’s envelope, which is a non-additive operation, and therefore, its impulse responses are not sufficient to describe the system. However, for the analysis of single isolated pulses at the input, it is sufficient.

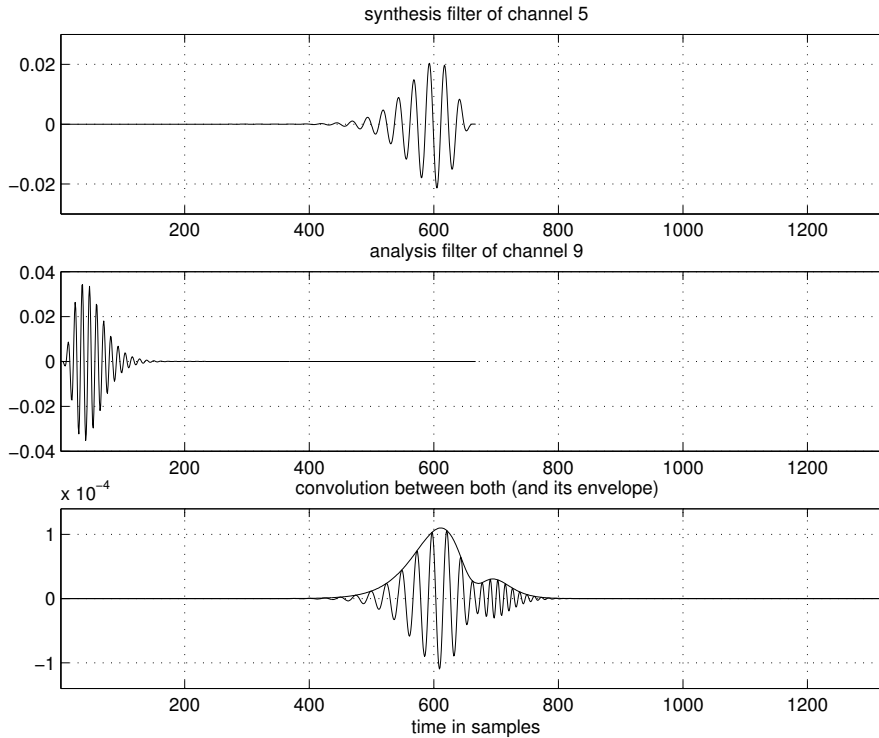


Figure 3.6: Convolution of two gammatones with different carrier frequencies. The envelope can have more than one local maximum due to beating.

8 kHz sampling rate), the peak can always be found at  $\hat{E}_{ch} = E_{ch}[N_{gt} - 1, ch]$ . The amplitude of the peak excitation is the OME-weighted squared norm of the gammatone impulse response  $\hat{E}_{ch} = \varrho_{ch} \sum_n h_{ch}^2[n]$ . The total length of a pattern is  $2N_{gt} - 1$ .

It can be observed that the envelopes that form a unit pulse excitation pattern do not strictly exhibit only a single local maximum along time. This beating phenomenon is a direct result of convolving gammatones with different center frequencies (i.e., different tone carrier frequencies). In Fig. 3.6 this is visualized with the first (time-reversed) gammatone taken from channel 5 (around 320 Hz) and the second from channel 9 (around 700 Hz).

### 3.3.4 Decision for Pulse Deletion

The phenomenon of auditory masking is that one sound stimulus (the ‘masker’) can make another sound stimulus (the ‘probe’ or ‘target’) inaudible. A classical masking pattern for simultaneous masking shows the maximum amplitude level

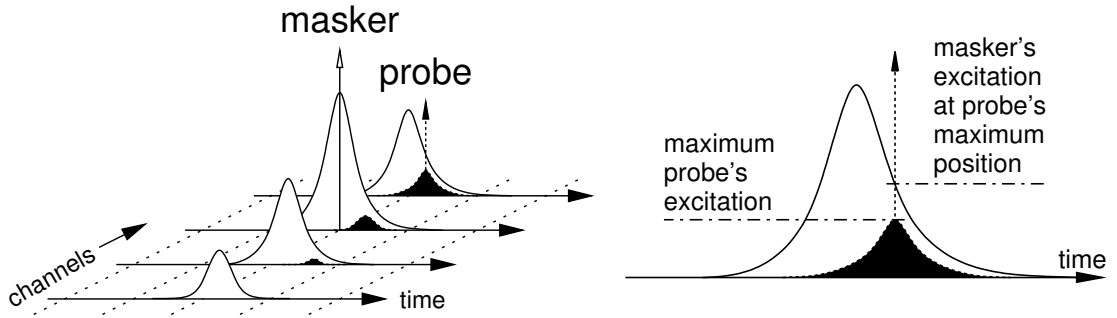


Figure 3.7: Illustration of the BM excitation of an isolated masking pulse (not filled) and the BM excitation of an isolated probe pulse (filled black). On the left hand side, a temporal segment of several auditory channels is shown. The masking pulse and the probe pulse are located in neighboring channels slightly apart in time. The right figure shows a detailed view of the one channel where the probe pulse has its origin. The maximum excitation of the probe and the excitation of the masker at the position of the probe are used in the masking criterion.

of the tone probe that is allowed such that the probe is still inaudible (‘masked threshold’) as a function of the probe frequency. Let us consider the unit pulse BM excitation patterns as masking patterns. Using these patterns, which span a spatio-temporal BM area, we are able to account for both simultaneous masking and temporal masking at the same time. In addition, since these patterns exhibit a smooth attack in a similar way as they decay, we account for forward as well as backward masking.

Since locally dominant pulses carry the major part of the signal information (see previous section), we want to remove less dominant pulses. To find whether a pulse is locally dominant or not, we consider every pulse individually as a masker and all pulses in its neighborhood, again pulse by pulse, as probes. Let us denote the multi-channel pulse trains by  $x[n, k]$  and require that these signals are ready to be resynthesized in the synthesis filterbank, i.e., steps such as power-law expansion and amplitude correction have already been performed. Now, consider the pulse at  $n = n_M$  and  $k = k_M$  as the masker. All other pulses between  $n = n_M - N_{gt} + 1$  and  $n = n_M + N_{gt} - 1$  in all channels are probes. Let us choose the pulse at  $n = n_P$  ( $n_M - N_{gt} + 1 \leq n_P \leq n_M + N_{gt} - 1$ ) and  $k = k_P$  for the current probe. On the left hand side of Fig. 3.7 such a situation with a single masker and a single probe is illustrated (but without delays between pulse and excitation). We now evaluate the worst-case contribution (or interference) of the isolated probe pulse to the isolated masking pulse by relating the maximum of the excitation pattern caused by the probe and the excitation caused by the masker evaluated at the position of the probe. To get the actual excitations we have to multiply the unit pulse

excitation patterns by the pulse amplitudes. For the probe's maximum excitation we get

$$E_{P,P} = x[n_P, k_P] \cdot \widehat{E}_{k_P} \quad (3.5)$$

and for the masker's excitation at the position of the probe's maximum we write

$$E_{M,P} = x[n_M, k_M] \cdot E_{k_M}[n_P - n_M + N_{gt} - 1, k_P]. \quad (3.6)$$

Finally, we can write the criterion for masking as

$$E_{P,P} < r \cdot E_{M,P} \quad (3.7)$$

where  $r$  is a constant usually close to one, that controls the impact of the criterion. We therefore refer to  $r$  as the *impact factor*. The right hand side of Fig. 3.7 illustrates the entities used on both sides of the masking criterion Equ. (3.7): the maximum excitation of the probe and the excitation of the masker at the position of the probe. If the criterion is fulfilled, we can delete the current probe and continue to test the next one. After testing all pulses in the neighborhood of the masking pulse, another pulse has to be selected for the new masking pulse. If  $r = 0$ , the criterion is never fulfilled and no pulses are omitted.

It is also useful to express the masking criterion in logarithmic terms. When we transform the relation of Equ. (3.7) into dB, we get

$$E_{P,P}^{(\text{dB})} < r^{(\text{dB})} + E_{M,P}^{(\text{dB})} \quad (3.8)$$

and the logarithmic impact factor  $r^{(\text{dB})} = 20 \log_{10}(r)$  turns into a maximum distance between the two logarithmic excitations.

The sequence for selecting a pulse to become the next masker plays an important role. Theoretically, an optimum configuration of the pulses of the synthesis code should exist, which produces a proper neural response with a minimum number of pulses to be synthesized, but it would be very difficult to find this configuration. We want to avoid exhaustive search procedures often used in classical analysis-by-synthesis schemes. Therefore, we propose a computationally simple method that finds a solution close to the optimum. For this method we sort all pulses according to their amplitudes and start with the highest one. This sorting is also in agreement with the underlying goal to get a sparse representation consisting of relatively isolated locally dominant pulses. In a real-time coder the sorting and subsequent masking decisions can be accomplished on a block-by-block basis introducing a small delay.

It is obvious that a higher impact factor yields a sparser pulse representation. The sparsification capability of the proposed masking method and its dependency on the impact factor are discussed in section 3.5.1. It is also clear that the deletion of pulses removes signal energy. When too many pulses are deleted, that is, when

the impact factor is relatively high (close to one), spectral distortions can become audible. In the next section we propose methods to correct the amplitudes of the remaining pulses to compensate for these spectral distortions. To determine the impact factor's influence on the overall reconstruction quality when the additional amplitude correction is applied, a three-alternative forced-choice (3AFC) listening test with 30 subjects was performed. The results of this listening test are presented in section 3.5.2.2.

It is worth mentioning that the impact factor  $r$  is the only new empirical parameter in this masking model. Compared to the model of temporal post-masking of [70] according to Equ. (3.1), which needs different time constants for all channels as empirical parameters, this means a considerable simplification.

Fig. 3.8 compares the exponentially decaying forward-masking threshold of Equ. (3.1) with the decay of the isolated-pulse BM excitation patterns for two auditory channels (50 Hz and 7 kHz). For this comparison, the intra-channel excitation has been extracted, i.e., the response of the transmultiplexer in a channel to an input impulse in the same channel. For the low-frequency channel, the BM excitation pattern is slightly wider but shows a comparable decay of  $-1.1$  dB/ms. For the high-frequency channel, the exponential masking threshold is significantly wider and by far flatter ( $-4.2$  dB/ms compared to about  $-20$  dB/ms of the BM excitation pattern, which is steepening even further). The reason for this difference is that the time constants of the exponential masking thresholds have been found empirically [70] by a human listener whereas the unit excitation patterns according to Equ. (3.3) are a direct result assuming a model for the BM only, i.e., just the gammatone filterbank without any further stages such as hair cells or neurons. When further processing such as compression or adaptation is taken into account, the auditory responses will be spread over a considerably wider time interval. This issue indicates that there is still a high potential to widen the patterns along the time axis for auditory channels with higher frequencies. In section 3.6 we discuss the incorporation of hair-cell models to obtain flatter temporal masking curves.

## 3.4 Pulse Amplitude Correction

The sparsification of the auditory representation yields a loss of signal energy. In order to properly resynthesize the sparse pulse code, we need to correct the amplitudes of the remaining pulses to restore the original energy distribution in the time-frequency plane.

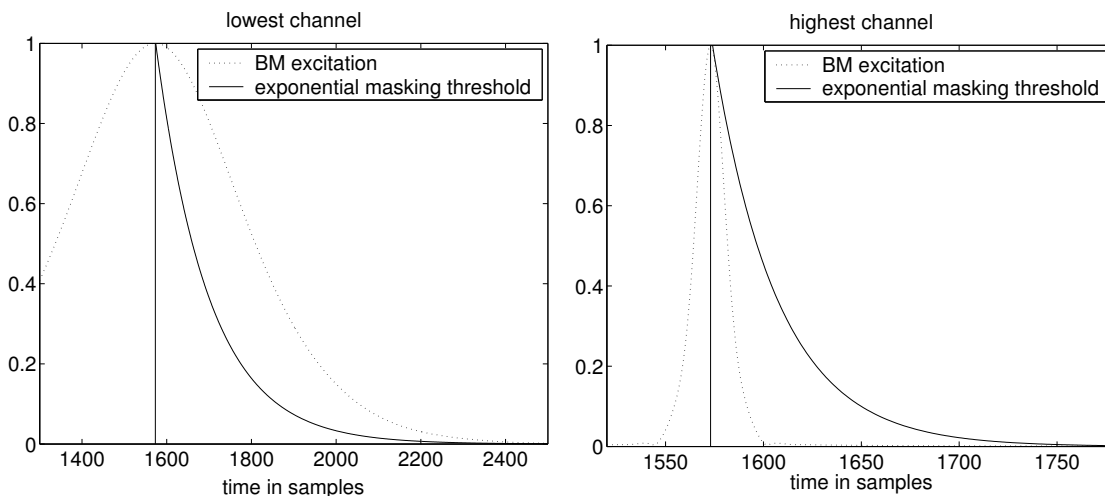


Figure 3.8: Comparison of the (normalized) intra-channel pulse BM excitation pattern and the exponentially decaying forward-masking threshold of [70] for the lowest (50 Hz) and the highest (7 kHz) channel of a wideband speech coder setup.

### 3.4.1 Pulse Distance Based Weighting

The pulse amplitude correction due to sparsification is similar to the correction stage discussed in chapter 2. There, in section 2.3.1, a pulse amplitude correction factor is derived to compensate for the loss of energy due to peak picking (i.e., adaptive downsampling). In the complete pulse representation, that is after peak picking, the distance between pulses (in seconds) is typically close to the inverse of the channel's center frequency. Therefore, constant channel weights according to Equ. (2.20) are sufficient to correct the pulse amplitudes and yield a good resynthesis quality. This is no longer true for synthesizing the sparsified pulse representation. An adaptive correction factor, which is equal to half the actual pulse distance (in samples) as already discussed in section 2.3.1, can be used instead. In section 2.3.4 we have outlined that this pulse distance based weighting is descended from the first iteration of the frame algorithm based reconstruction method in [75]. However, since the amplitudes of the complete pulse code have to be already corrected<sup>9</sup> using Equ. (2.20) before the masking criterion can be tested (see section 3.3), the non-adaptive correction must be undone before the adaptive one can be applied to the sparsified pulse code.

It should be noted that the simple, adaptive correction factor needs to be

<sup>9</sup>The masking model is based on the transmultiplexer that incorporates the decoder incompletely by considering the synthesis filterbank only. Moving the correction due to peak picking from the decoder to the encoder (before the masking model) brings the advantage that this correction has to be performed only once.

limited to a maximum value, which depends on the maximally considered pulse distance. Otherwise it can excessively boost the amplitude after a silence period or disproportionately amplify isolated and ‘loose’ pulses of a noise floor. This limitation has to be in accordance with the support of the impulse response of the corresponding channel’s synthesis filter since, for larger pulse distances, the pulses do no longer contribute to each other to build the output signal, i.e., the shifted impulse responses do not overlap anymore. Note, that the level of a signal with a certain amplitude in dB is increased by less than 1 dB when a second in-phase<sup>10</sup> signal with an amplitude smaller than  $-18$  dB relative to the amplitude of the first signal is added. Keeping this in mind, we can specify an upper limit for pulse distances needed for the computation of the adaptive pulse amplitude correction factor by searching for the  $-18$  dB point of the impulse response envelope with respect to the maximum. Thus, we obtain channel-dependent limits. According to our simulations we get maximum distances between two and six times the period of the channel’s carrier tone for center frequencies between 100 Hz and 3600 Hz. This simple method using such an adaptive correction factor is able to equalize spectral distortion in dominant regions (as it can be seen in Fig. 3.23), but it also amplifies the noise in spectral valleys, which can become audible. Another drawback of this method is that it is not able to take account of interchannel interference (or the absence of interchannel interference). For speech signals and highly sparse pulse representations, no promising results can be obtained, as found by informal listening tests.

### 3.4.2 BM Excitation Based Correction Factor

A method to correct the pulse amplitudes to compensate for the spectral distortion due to pulse removal that is more promising than the simple pulse distance based weighting described in the previous section has been proposed in [79]. Just like the masking model, this correction method is based on a BM excitation model, i.e., we calculate envelopes within all auditory channels and compare the excitation generated by the original (i.e., the unreduced) pulse trains to the one produced by the reduced pulse trains. For the envelope generation, we low-pass filter the pulse trains using linear-phase FIR filters whose impulse responses are already contained in the unit pulse excitation patterns  $E_{ch}[n, k]$ . For the auditory channel  $k$ , we use the impulse response  $l_k[n] = E_k[n, k]$ , i.e., the OME-weighted envelope of the convolution between the impulse responses of the  $k$ th channel of both analysis and synthesis filterbank. Again, we call the original pulse trains  $x[n, k]$  and refer

---

<sup>10</sup>For the in-phase case, the increase in amplitude is a maximum, i.e., the error is bounded by 1 dB in general. An error of 1 dB is in good agreement with JNDs for intensity variations and is often used as a bound for spectral distortion.

to the reduced pulse trains as  $\tilde{x}[n, k]$ . Then, we can generate the envelopes by

$$e_k[n] = x[n, k] * l_k[n] \quad (3.9)$$

and

$$\tilde{e}_k[n] = \tilde{x}[n, k] * l_k[n], \quad (3.10)$$

where  $*$  denotes convolution. Finally, the amplitude of the pulse  $\tilde{x}[n_0, k_0]$  is corrected by multiplication with the ratio of the original to the reduced envelope at the proper location

$$\tilde{y}[n_0, k_0] = \tilde{x}[n_0, k_0] \frac{e_{k_0}[n_0 - N_{gt} + 1]}{\tilde{e}_{k_0}[n_0 - N_{gt} + 1]}. \quad (3.11)$$

The delay of  $N_{gt} - 1$  samples originates from the group delay of the linear-phase low-pass filters, which is the same for all channels. This ratio needs to be computed only for the remaining pulse locations of the sparsified auditory representation, thus, divisions by zero are impossible to occur. We should also mention that the OME-weighting of the low-pass filter impulse responses  $l_k[n]$  may be omitted because it is cancelled in the ratio of Equ. (3.11). The advantage of using  $l_k[n] = E_k[n, k]$ , i.e., the unit pulse excitation patterns of the masking model, over separate low-pass filters is to lower the memory requirement.

The correction scheme proposed in [79] is relatively simple since it computes the excitation for a certain BM position by considering the pulses of a single channel only (the channel that corresponds to that position). When we refer to the transmultiplexer structure of Fig. 3.4, it becomes clear that this is an approximation because also pulses of neighboring channels contribute to the final BM excitation at the considered position. To formulate a more accurate amplitude correction scheme, we have to replace Equ. (3.9) and Equ. (3.10) by

$$e_k[n] = \mathcal{E}\mathcal{N}\mathcal{V} \left\{ \left( \sum_{l=0}^{K-1} x[n, l] * g_l[n] \right) * \varrho_k h_k[n] \right\} \quad (3.12)$$

and

$$\tilde{e}_k[n] = \mathcal{E}\mathcal{N}\mathcal{V} \left\{ \left( \sum_{l=0}^{K-1} \tilde{x}[n, l] * g_l[n] \right) * \varrho_k h_k[n] \right\}, \quad (3.13)$$

respectively. Equ. (3.12) and Equ. (3.13) fully resynthesize the output signal  $(\sum_{l=0}^{K-1} \tilde{x}[n, l] * g_l[n])$  and pass it through the OME-weighted analysis filterbank  $(\varrho_k h_k[n], \text{ for } k = 0, \dots, K - 1)$  before extracting the envelopes. The correction factor for the amplitude of a pulse is calculated as before in Equ. (3.11) by taking the ratio. According to our simulations, this more complex method yields slightly better results than the one proposed in [79], especially for particularly

sparse pulse representations. However, also the computational effort is considerably higher. Using this amplitude correction method we are able to increase the impact factor  $r$  of the masking model even beyond the value of 1 while maintaining transparent resynthesis quality. In section 3.5.2.2 we show the results of a three-alternative forced-choice (3AFC) listening test with 30 subjects to demonstrate the power of this amplitude correction scheme. For the following experiments and the 3AFC listening test, we use the more accurate excitation computation according to Equ. (3.12) and Equ. (3.13).

### 3.4.3 The New Coder Structure

We are now able to incorporate the masking model of section 3.3 together with the pulse amplitude correction scheme into our physiologically motivated subband coder described in the previous chapter. The new block diagram of the encoder is plotted in Fig. 3.9. When we compare this block diagram with Fig. 2.1, we recognize that the sequence of some objects has been changed. The half-wave rectifiers are now followed by the peak pickers directly and the power-law compressors have been moved to the end. Another prominent modification is that the amplitude correction due to peak picking is now performed already in the encoder directly after peak picking and is no longer done in the decoder. The reason for these modifications is that the underlying transmultiplexer of the masking model, which is the next block in the coder, should get the same pulse amplitudes as the synthesis filterbank in the decoder. The new sequence ensures this without the need for considering the different steps multiple times. The masking model is followed by the pulse amplitude correction due to pulse removal. This stage has the reduced (from the masking model) and the unreduced (prior the masking model) pulse trains as inputs to be able to compute the ratio of both excitations. The final stages of the coder are the power-law compressors and the quantization and coding unit. These two elements together operate like a companding quantizer. We will consider quantization and coding in more detail in chapter 4.

Fig. 3.10 shows the modified structure of the decoder, which now has to reconstruct the pulse trains by decoding the incoming bit stream and performing the power-law expansion only, and to filter them in the synthesis filterbank. Finally, the sum is passed through the equalizer filter (see section 2.3.2) to minimize the remaining magnitude ripple.

## 3.5 Experimental Results

In this section we show results from simulations with the proposed masking model and the new pulse amplitude correction scheme described in the previous sections.

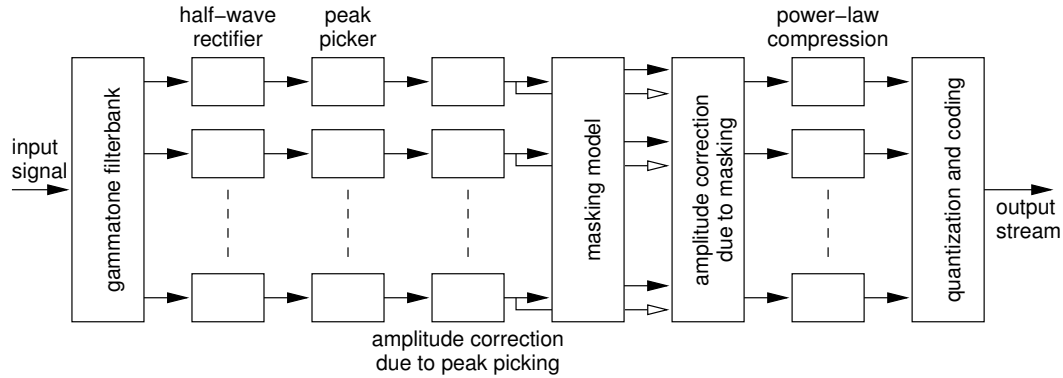


Figure 3.9: New block diagram of the encoder. The masking model and the amplitude correction stage due to masking have been incorporated. The latter has two input signals per channel: the sparsified pulse trains (filled black) and the complete pulse trains (not filled).

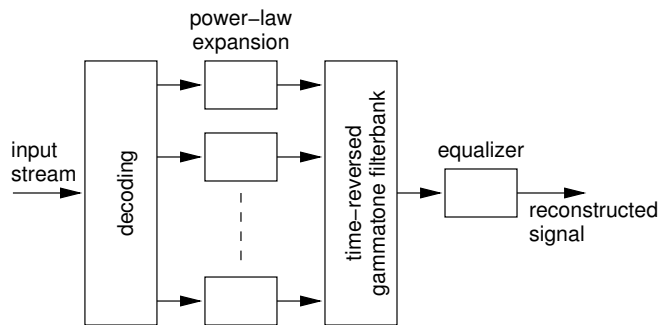


Figure 3.10: Modified block diagram of the decoder. The amplitude correction stage due to peak picking has been moved into the coder (see Fig. 3.9). This simplifies the decoder as well as the encoder because the decoder is (partially) incorporated in the transmultiplexer, which forms the basis of the masking model and the new amplitude correction scheme in the encoder.

First, in section 3.5.1 we demonstrate the sparsification capability of the masking model and the influence of the impact factor on the obtained pulse code. We show the power of the pulse amplitude correction method by comparing the BM excitation caused by the complete pulse representation and the BM excitation caused by the sparsified pulse code. In section 3.5.2 we show that the decisions of the masking model are in accordance with data from psychoacoustical experiments. For the first experiment, we use a simple stationary stimulus that consists of a narrowband noise masker and a tone probe. This experiment provides an insight into the behavior of the new masking model. Furthermore, for the second experiment, a listening test with 30 subjects has been performed to evaluate the subjective quality of narrowband speech when it is reconstructed from its sparsified auditory pulse representation.

### 3.5.1 Sparsification Capability

For the 8 kHz speech signal ‘The juice of lemons makes fine punch’ spoken by a female speaker, the total number of pulses and the number of pulses found in the individual auditory channels are visualized in Fig. 3.11 before applying the masking criterion of section 3.3 with  $r = 1$  and afterwards. We observe a reduction in the overall number of pulses from 76,752 to 19,344 that corresponds to an elimination of 74.8% of the original pulses. If we consider the time unit of a sample we now need only 0.77 pulses per sample instead of 3.05. It also becomes apparent that in higher-frequency channels much more pulses can be omitted than at lower frequencies.

Fig. 3.12 shows a segment of the speech example mentioned above and compares the original and the reconstructed signal. In addition to the masking criterion with  $r = 1$  the amplitude correction method of section 3.4 has been applied. The averaged segmental<sup>11</sup> signal-to-noise ratio is only 12.2 dB, but the error is hardly audible.

The upper plot of Fig. 3.13 shows the auditory representation of the first 100 samples of the voiced signal segment (about three pitch cycles) shown in Fig. 3.12 obtained after peak picking (i.e., the complete pulse code). In the lower plot the pulse representation is shown after the masking criterion with  $r = 1$  (which removes 74% of the pulses) has been applied. The six lowest channels are dominated by the fundamental frequency. This is clearly visible in the sparsified pulse code where only the pulses from channel 3 and 4 do not satisfy the masking criterion and thus survive. In the channels with higher center frequency the sparsified representation exhibits clusters of pulses that are synchronized with the pitch period.

In Fig. 3.14 a 100 ms segment of the pulse train of channel 14 (center frequency

---

<sup>11</sup>The used segment length is 20 ms.

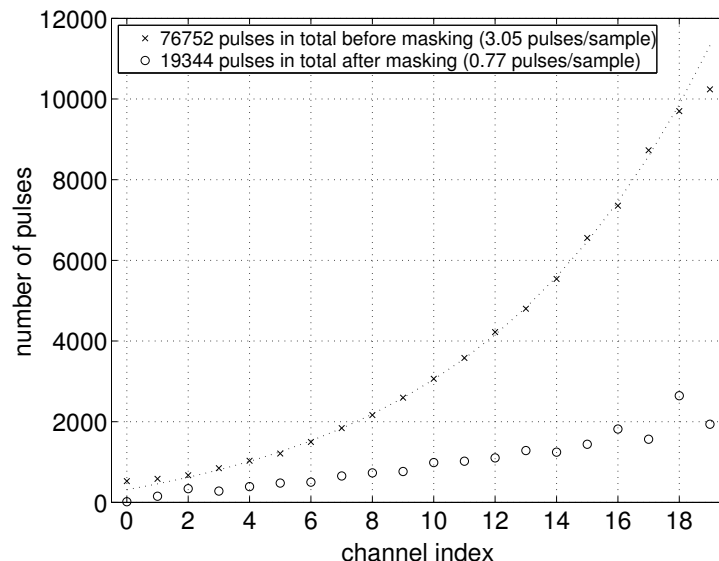


Figure 3.11: Reduction of the number of pulses in the channels. The dotted line, which predicts the number of original pulses, is the channel's center frequency multiplied by the duration of the speech signal.

1778 Hz) is shown for comparison before the masking stage, after the masking stage, and after the pulse amplitude correction stage. The segment is taken from the vowel of the word 'juice' spoken by a female. About 80% of the pulses are deleted in this channel. It can be seen clearly that locally dominant pulses survive the sparsification process. The change in pulse amplitude caused by the amplitude correction method described in section 3.4 can be remarkable. Particularly isolated pulses are amplified, such as the pulse around sample 260, which is amplified by a factor of almost 5. Note the different amplitude scale of the lower plot.

In order to demonstrate the power of the new pulse amplitude correction method, Fig. 3.15 shows the difference in logarithmic BM excitation between the resynthesized sparsified pulse representation and the original one where all pulses have been kept (evaluated at the output of the transmultiplexer shown in Fig. 3.4). The upper plot originates from a reduced pulse code without the amplitude correction, while for the lower plot, the correction has been applied. The difference in logarithmic BM excitation is lowered considerably by the correction factors according to Equ. (3.11). Though sharply localized differences can still occur, the overall area with differences is minimized. For this simulation, the calculation of the correction factors is based on the excitations according to Equ. (3.12) and Equ. (3.13).

The impact factor  $r$  of the masking criterion controls the amount of deleted

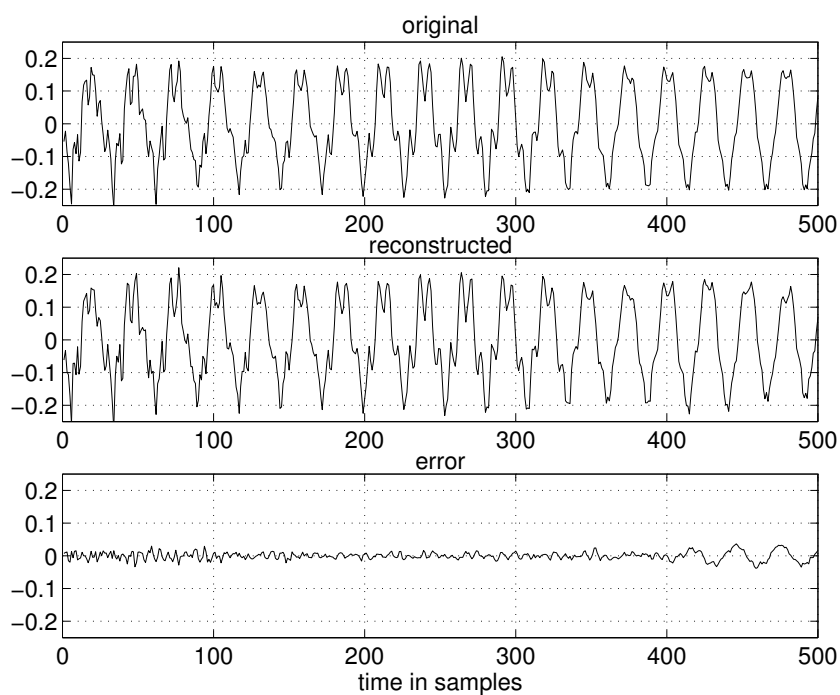


Figure 3.12: Segment of a voiced-speech signal taken from the word ‘juice’ spoken by a female speaker. Original signal (top), signal resynthesized from the remaining 26% of the original pulses (middle) and the difference between both signals (bottom).

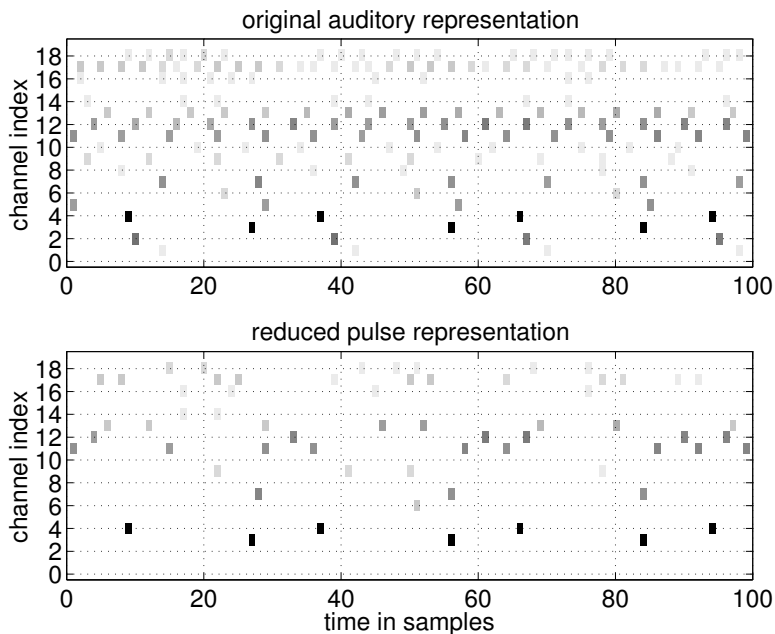


Figure 3.13: Original and sparsified pulse representation.

pulses. A criterion with a higher impact factor deletes more pulses. Fig. 3.16 shows the average number of remaining pulses for different impact factors.  $r$  has been varied from 0.8 ( $-1.9$  dB) to 1.5 ( $+3.5$  dB) for the case of a 20-channel narrowband setup. The figures in this graph have been gathered by processing speech material of about 12 seconds in total duration of both male and female English speakers sampled at 8 kHz. The complete pulse representation has 3.05 pulses per sample on average for this filterbank setup. For the considered range, the number of remaining pulses starts to decrease rather linearly with increasing impact factor. When  $r$  is increased beyond 1 (or 0 dB), a saturation happens. We can no longer observe such a high increase in deletions as at lower impact factors. For factors greater than 0.9 ( $-0.9$  dB), the sparsified signal representation is already undercomplete, i.e., it contains less than 1 pulse per sample on average. The ordinate on the right edge shows the amount of deletions in % where 100% correspond to 3.05 pulses per sample of the original code. This axis is valid for the 20-channel case only.

Interestingly, for a dense filterbank setup with highly overlapping channels, the influence of the impact factor is different. In Fig. 3.16 results from experiments with a 50-channel gammatone filterbank for the same narrowband frequency range as above are presented. The spacing of neighboring filters' center frequencies is only 0.47 ERBs (in contrast to 1.20 ERBs for the 20-channel filterbank). The unreduced

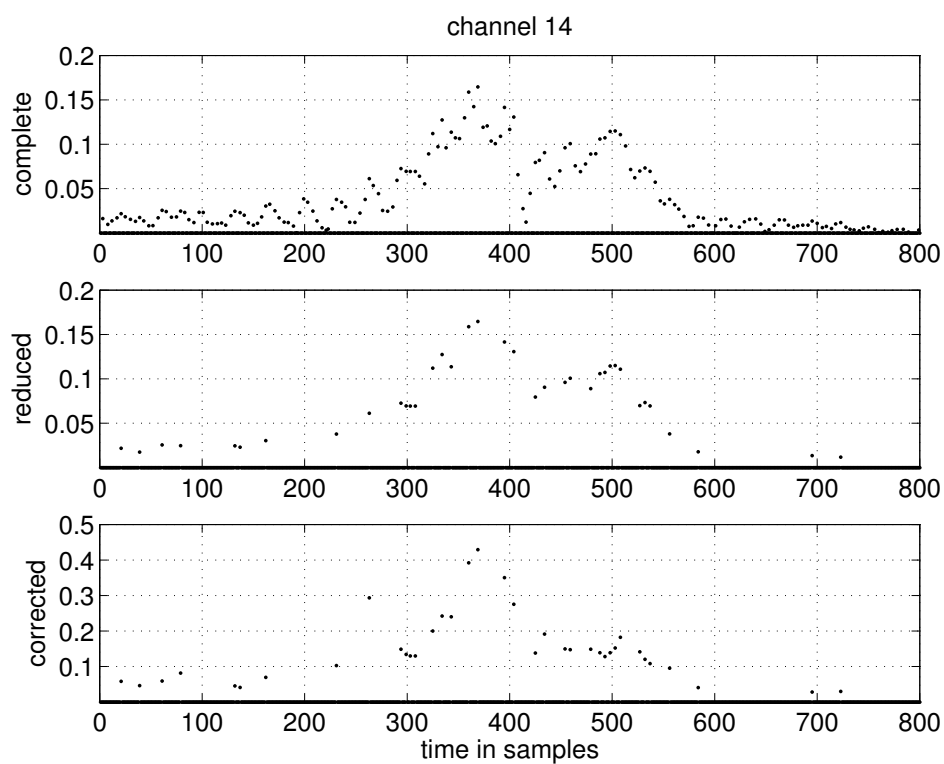


Figure 3.14: Comparison of the original, the sparsified, and the amplitude corrected pulse representations of channel 14. Note that the bottom graph has a different vertical scale.

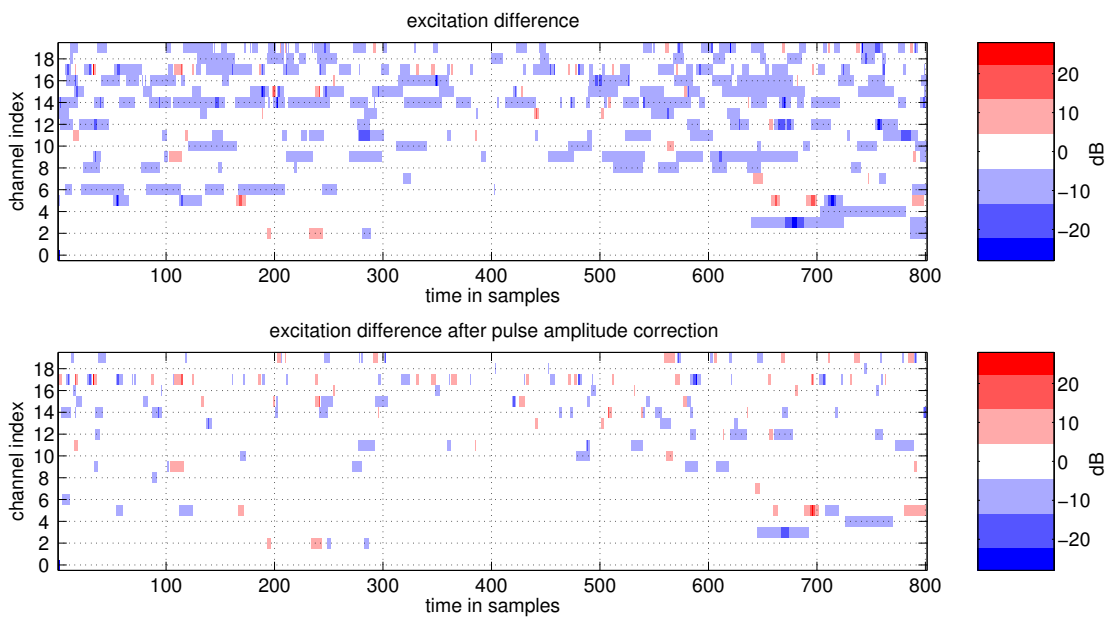


Figure 3.15: Reduction of the difference in BM excitation between the resynthesized sparsified pulse representation and the original one ('n' from the word 'lemons' spoken by a female) without any additional pulse amplitude correction after the pulse reduction (upper plot) and with the proposed amplitude correction scheme (lower plot).

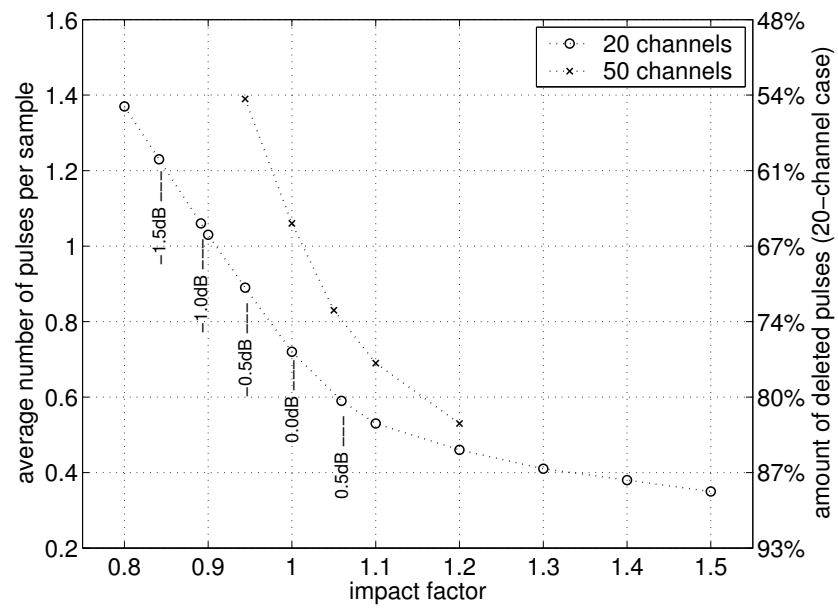


Figure 3.16: Average number of remaining pulses as a function of the impact factor  $r$  for a 20-channel and a 50-channel coder setup for 8 kHz sampling rate. For the 20-channel case, the amount of deletions in percent with respect to 3.05 pulses per sample of the unreduced pulse code is given on the right ordinate.

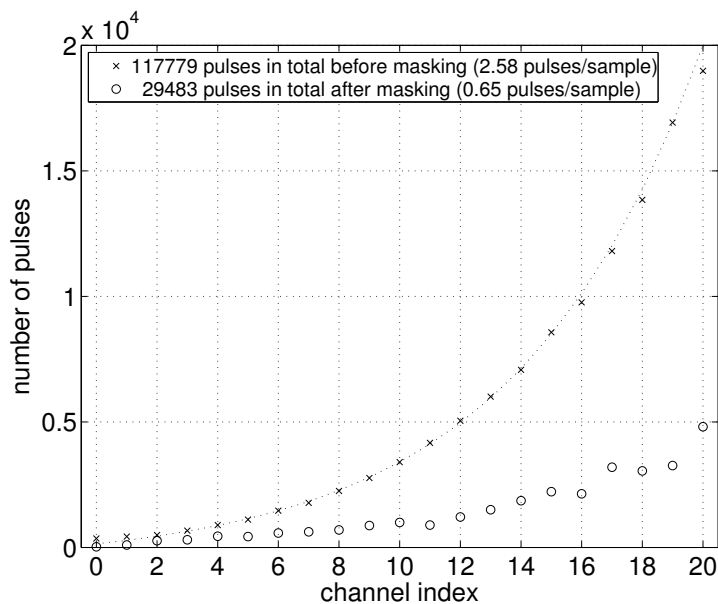


Figure 3.17: Reduction of the number of pulses in the 21 channels for a wideband speech coding experiment.

pulse code has now 7.51 pulses per sample on average. After the masking decisions with a certain impact factor, the number of remaining pulses is higher than in the case with 20 channels. But in terms of deletions we obtain much higher figures: between 81.5% and 92.9% of the original pulses are deleted for the range of  $r$  considered here.

To be able to compare our method with [70] in a fair way, we ran a simulation with a wideband speech signal (i.e., 16 kHz sampling rate) and with a comparable setup of the coder: 21 channels ranging from 50 Hz to 7 kHz. This resulted in a relatively wide spacing of the auditory channels of 1.51 ERBs. With an impact factor of  $r = 1$  and the accompanying amplitude correction, the masking decisions again eliminated 74% of the pulses, what corresponds to merely 0.66 pulses per sample for the reduced representation (see Fig. 3.17), and we achieved a high reconstruction quality as confirmed by informal listening tests. In [70] 1.26 pulses per sample are necessary. Compared to their figures, we are able to almost halve the number of necessary pulses.

As already obvious in Fig. 3.11, exploiting the masking model is much more efficient at higher frequencies. Therefore, we expect to get even better results when signals at higher sampling rate are processed, e.g., general audio signals at 44.1 kHz.

## 3.5.2 Quality of the New Masking Model

### 3.5.2.1 Simple Stationary Stimuli

To test whether the decisions of our new masking model are in accordance with psychoacoustical data, a simulation with a simple auditory stimulus has been performed where also data from a masking experiment has been gathered. The stimulus consists of a narrowband noise (NBN) masker with a bandwidth of 90 Hz centered at a frequency of  $f_M = 410$  Hz (as used in [84, 85]) and a tone probe with a lower frequency  $f_T \leq f_M$ . It is known that masking experiments with this kind of stimuli yield reliable masking patterns since the detection of the probe is not strongly influenced by interfering cues such as the detection of beats or difference tones [33]. The reason for using a probe frequency that is lower than the masker frequency is that masking patterns (or excitation patterns) exhibit the same slope towards lower frequencies for different masker intensities. On the contrary, towards higher frequencies a nonlinear dependency can be observed (‘upward spread of masking’ [35]). Therefore, considering a probe with a lower frequency than the masker simplifies the experimental setup because it does not necessarily require the playback of the stimuli with exactly known sound pressure level (SPL). Furthermore, since our model is currently based on a linear gammatone filterbank, we cannot expect to be able to model any level-dependent effects.

The NBN plus tone stimulus was generated continuously using the real-time audio processing environment ‘pd’ and presented to the subjects by headphones<sup>12</sup> in a silent office. The subjects were asked to change the level of the tone probe (at a fixed probe frequency) until it was just audible (‘method of adjustment’ [33]). The level could be changed by steps<sup>13</sup> of 5 dB and 1 dB.

Fig. 3.18 shows the results of six subjects (one female and five male) ranging in age from 26 to 32. Using these results, we can generate stimuli either with audible probe or with inaudible probe and use them as input signals for the testing of the proposed coding system. In the following the behavior of the new masking model is demonstrated using a probe frequency of  $f_T = 300$  Hz and a probe level of  $-35$  dB for the inaudible probe and  $-25$  dB for the audible probe where 0 dB refers to the level of the NBN masker<sup>14</sup>.

In order to focus on the evaluation of the quality of the masking model, we use a highly redundant coder setup with 50 channels for the frequency range of interest only, which is from  $f_{min} = 200$  Hz to  $f_{max} = 800$  Hz. Additionally, the sampling frequency  $f_s$  is set to 24 kHz. The filterbank settings result in a spacing between the channels of only 0.17 ERBs. Using this high-resolution

<sup>12</sup>AKG K270 studio headphones on an external TASCAM US122 soundcard at maximum output level, at which a moderate SPL is produced.

<sup>13</sup>Steps are implemented as ramps with a duration of 200 ms.

<sup>14</sup>Based on long-term RMS measurement.

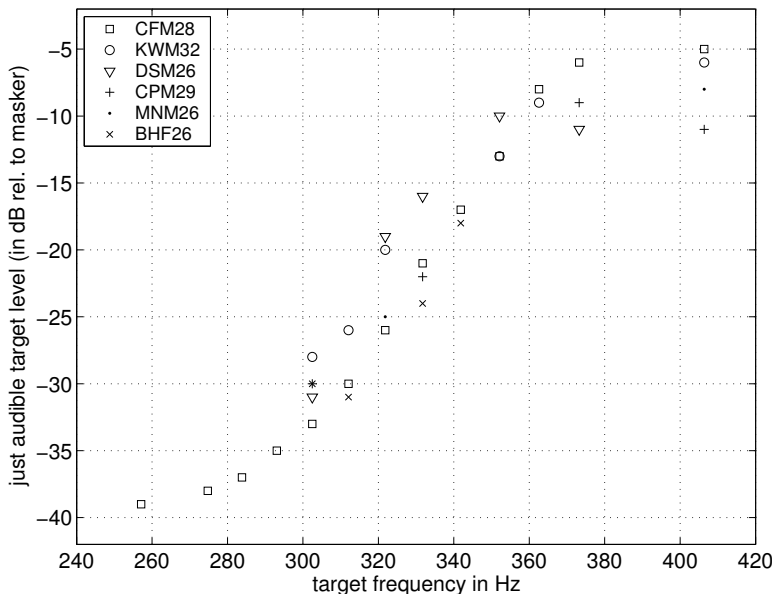


Figure 3.18: Just audible tone probe levels for different probe frequencies in the presence of an NBN masker centered at 410 Hz. Data of six subjects are shown.

filterbank we are able to directly associate a few channels with the tone probe and we can expect that the number of remaining pulses after the masking stage in these channels depends on whether the probe is audible or not. For the case when the tone probe is inaudible, we can expect the pulse rate to be reduced to zero. Using a low-resolution filterbank (such as in the 20-channel coder) we cannot observe the masked scenario on the basis of the pulse rate in the corresponding channels. But this does not automatically mean that an inaudible tone probe is not masked in that case. To show this, we also perform experiments with the 20-channel narrowband setup and present the results at the end of this section.

First, a stimulus is transformed into an auditory representation by filtering the input signal in the gammatone analysis filterbank followed by half-wave rectification and peak picking. The obtained pulse trains are amplitude-corrected by channel-wise multiplication with the weight  $\kappa_k$  according to Equ. (2.20). The correction of off-peak errors as described in section 2.3 can be neglected since the channel signals are highly oversampled for the used sampling frequency.

This pulse representation is then sparsified according to section 3.3: all pulses are sorted in a descending amplitude order to determine the pulse masker sequence, then, masker-by-masker, the masking decision from Equ. (3.7) is made for all pulse probes in the masker's neighborhood using the pre-computed and stored unit pulse BM excitation patterns  $E_{ch}[n, k]$ .

It can be observed that an impact factor in Equ. (3.8) on the order of  $r^{(dB)} =$

−1 dB, which works well for the 20-channel narrowband setup, deletes too many pulses now—even the audible probe is deleted completely. To obtain proper decisions, the impact factor has to be decreased to −5 dB. The need for a lower impact factor than in the experiments with the 20-channel coder setup is a direct consequence of the dense 50-channel filterbank that has highly overlapping filters. For completeness we also present the results of an experiment with the 20-channel narrowband coder setup later in this section.

Fig. 3.19 shows the number of pulses in the sparsified pulse representation of the two different stimuli of one second duration for an impact factor of  $r^{(dB)} = -5$  dB. At frequencies far from the probe at 300 Hz only small differences between the two stimuli can be recognized. These small differences arise from the fact that the continuously generated NBN masker signal is slightly different in the two stimuli, i.e., it has not been generated once and stored (or ‘frozen’) and reused for both stimuli. On the other hand, at frequencies around 300 Hz (the tone probe), big differences in the remaining number of pulses can be observed. In the case of the audible probe, roughly 100 pulses per second and per channel are spent on the tone probe whereas in the case of the inaudible probe almost no pulse survives the sparsifying process. This indicates that the decisions of the new masking model are in accordance with psychoacoustical data for this simple stimulus.

In order to gain insight into the functionality of the proposed masking model, we examine some more statistics about the masking decisions for the simple NBN plus tone stimulus. Fig. 3.20 shows the averaged pulse amplitudes in the individual auditory channels of the complete pulse representation. These contours resemble classical excitation patterns<sup>15</sup> such as described by Glasberg and Moore in [34, 86]. When the excitation pattern produced by the masker alone is considered as a vertically shifted version [35] of the masking pattern from Fig. 3.18 we do not get a satisfactory match. Better results are obtained by classical masking models that evaluate the differences of the excitation patterns produced by different stationary stimuli [85]. When the difference at any channel<sup>16</sup> exceeds a certain threshold (usually 1 dB is believed to be the ‘quantization step’ in intensity perception [33]) also a human listener should be able to detect a difference. In Fig. 3.20 the NBN masker alone and the NBN plus inaudible tone stimulus produce virtually the same patterns. Whereas in the case of the audible tone small differences around the probe frequency of 300 Hz and below can be observed. The maximum difference of about 2.5 dB arises at channel 8 (266 Hz). Also according to a classical masking model, the tone should be audible.

The problem of masking models based on classical excitation patterns is that

---

<sup>15</sup>It should be mentioned that the OME-weighting is missing in this representation. However, for the narrow frequency region considered here, the weighting can be neglected.

<sup>16</sup>This way ‘off-frequency listening’ is taken into account.

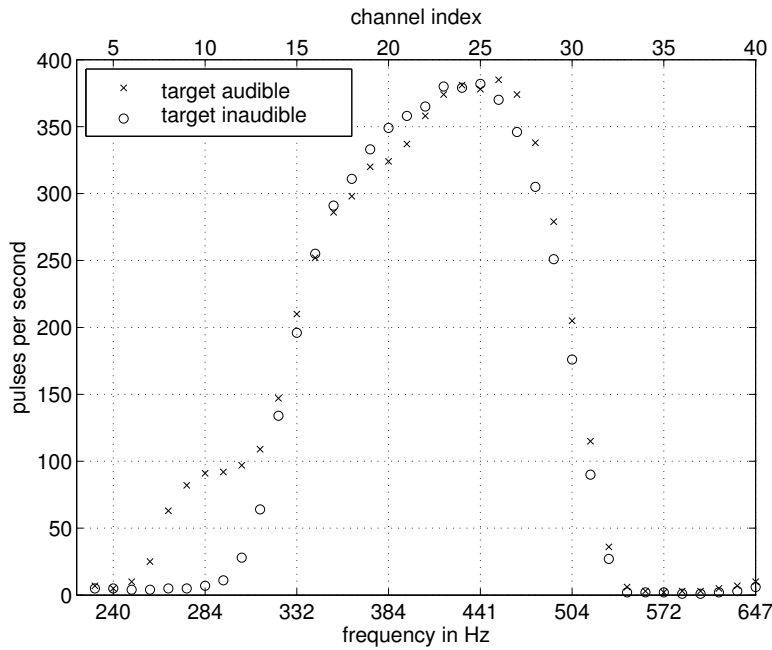


Figure 3.19: Number of remaining pulses in the sparsified representation of one second of two different NBN plus tone stimuli. Comparison between audible and inaudible 300 Hz tone probe.

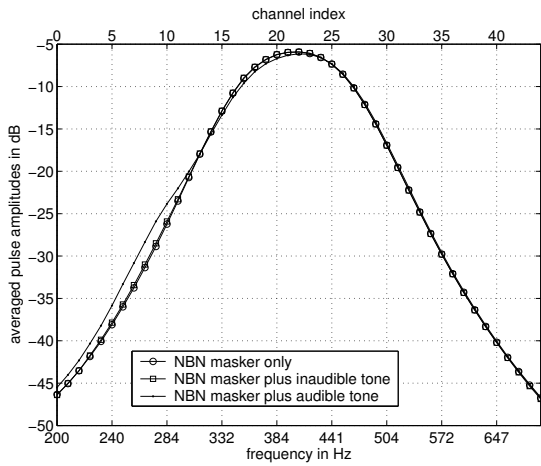


Figure 3.20: Averaged pulse amplitudes of the complete pulse representation for three NBN-plus-tone stimuli.

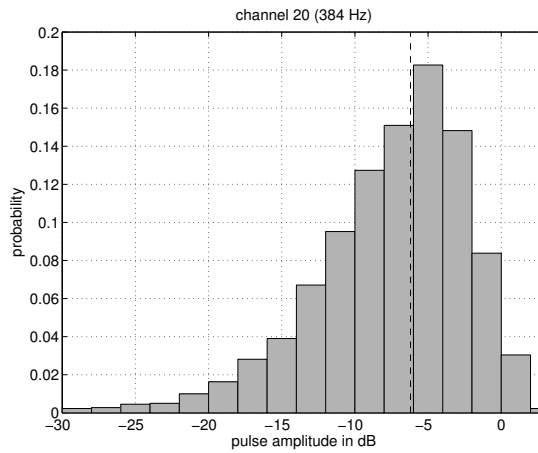


Figure 3.21: Pulse amplitude distribution (logarithmic) in channel 20 for the NBN masker alone. The dashed vertical line shows the averaged value plotted in Fig. 3.20.

they are restricted to stationary stimuli, whereas the proposed isolated pulse masking model does not rely on such a strong requirement since it is able to adapt to envelope variations. The 90-Hz-bandwidth NBN stimulus described above shows very strong envelope fluctuations<sup>17</sup>. Fig. 3.21 shows the distribution of the logarithmic pulse amplitudes in channel 20 with a center frequency  $f_c = 384$  Hz produced by the NBN masker. It reveals that amplitudes close ( $\pm 1$  dB) to their averaged value, which is shown in Fig. 3.20, occur only around 18% of the time. It also shows that envelope valleys with very small amplitudes are possible.

The proposed model treats even simple stimuli (such as NBN-plus-tone stimuli) as non-stationary signals. This behavior is well demonstrated in the case of the audible tone where only about a third of the original pulses survives in the channels close to the probe frequency, i.e., one pulse per three tone periods is found on average in the sparsified representation. Thus, the model illustrates that the detection of a probe starts by partial detection in valleys of the masking envelope.

The non-stationarity makes it difficult to derive simple masking or excitation patterns (i.e., a single decaying threshold curve starting at the edge of a masker) such as the RMS-based ones [33, 86, 34]. In addition to the envelope fluctuations discussed above, another fuzziness results from the two-dimensional isolated-pulse BM excitation patterns, on which probes will be deleted on varying temporal positions, and also the varying masking channel contributes. Fig. 3.22 shows the effective masking channels for deleted probe pulses from channels 10–12 (around the tone frequency). Masking pulses from channel 20, which is close to the low-frequency edge of the NBN masker<sup>18</sup> (384 Hz), cause the most deletions. For higher channels, which at first achieve a higher excitation by the NBN masker but which are also more distant from the probes, the number of deletions decreases. But also channels below the nominal band of the NBN masker cause a considerable amount of pulse deletions. Fig. 3.21 and Fig. 3.22 point out the difficulty of deriving a global decision whether the tone is audible or not based on the decisions of individual pulses. The experiment nicely illustrates that there is no hard transition between audible and inaudible—the transition is rather soft. Many people with experience in psychoacoustic experiments may agree on this point.

It should be noted that the correctness of the masking decisions depends directly on the accuracy of the underlying auditory filterbank. In general, an extension to level-dependent filterbanks (such as the gammachirp auditory filters [44]) is possible but that would also require level-dependent excitation patterns needed for the masking decisions.

Finally, we compare the resynthesized signals from the sparsified pulse represen-

---

<sup>17</sup>In general, any band-pass noise stimulus evokes envelope fluctuations, which are limited by the bandwidth. Even wideband noise becomes band-pass noise after filtering in an auditory filter.

<sup>18</sup>The centers of channels 19–26 lie in the nominal frequency band of the NBN masker (365–455 Hz).

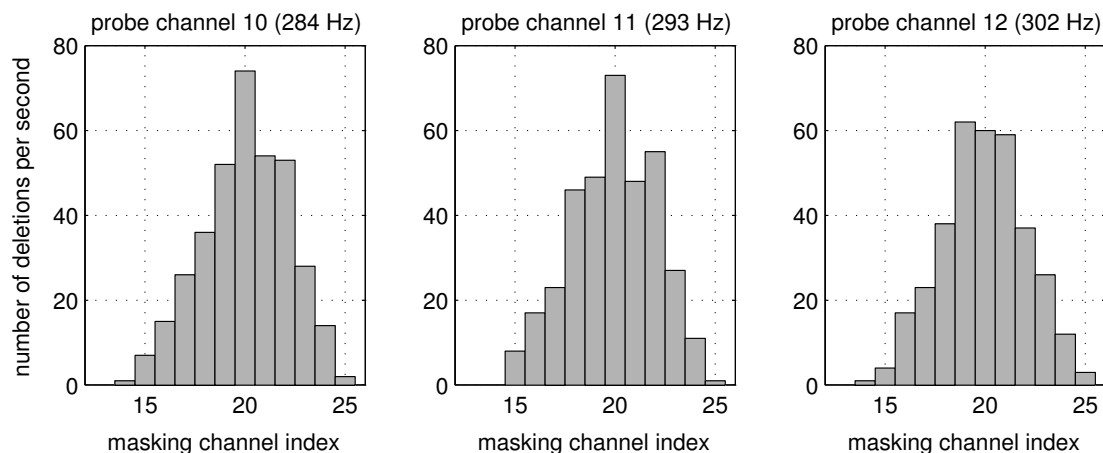


Figure 3.22: Origin and occurrence of masking pulses for deleted probes from channels 10–12 (around the tone frequency) for the NBN plus inaudible tone stimulus.

tations for the NBN plus audible tone and the NBN plus inaudible tone. Fig 3.23 shows the power spectral density for the original signals, the signals resynthesized from the sparsified and not amplitude-corrected pulse trains, and the resynthesis of the sparsified and adaptively corrected representations. In the case of the audible probe (upper plot) the 300 Hz tone is distinctly present. Even without further pulse amplitude correction it is attenuated only by 3 dB. It can be seen that the resynthesized signals from the sparse pulse trains exhibit amplified spectral valleys. But these valleys are masked by the dominant neighboring parts. In the case of the inaudible probe (lower plot), no tone can be detected after the resynthesis but only an increased noise floor.

As mentioned above, the masking model does its job properly even in a filterbank setup with a much wider spacing. The reason for using the hyper-dense 50-channel settings in the experiment above is to be able to assign certain channels to the tone probe. In this way, we are able to observe almost no remaining pulses in these channels for an inaudible probe. For completeness and to underline the potential of the proposed masking model, we perform another experiment with the NBN plus tone signal together with the 20-channel narrowband coder setup. Now, the filterbank covers a frequency range from 100 Hz to 3600 Hz, which results in a channel spacing of 1.2 ERBs that is seven times wider than in the 50-channel case above. The impact factor of the masking model is set to  $-1$  dB. Fig. 3.24 plots the number of remaining pulses of individual channels in the sparsified pulse code. The two channels that are closest to the 300 Hz tone probe are the channels 3 and 4 with center frequencies 256 Hz and 323 Hz, respectively. In contrast to the 50-channel experiment of Fig. 3.19, this graph does not exhibit vital differences in

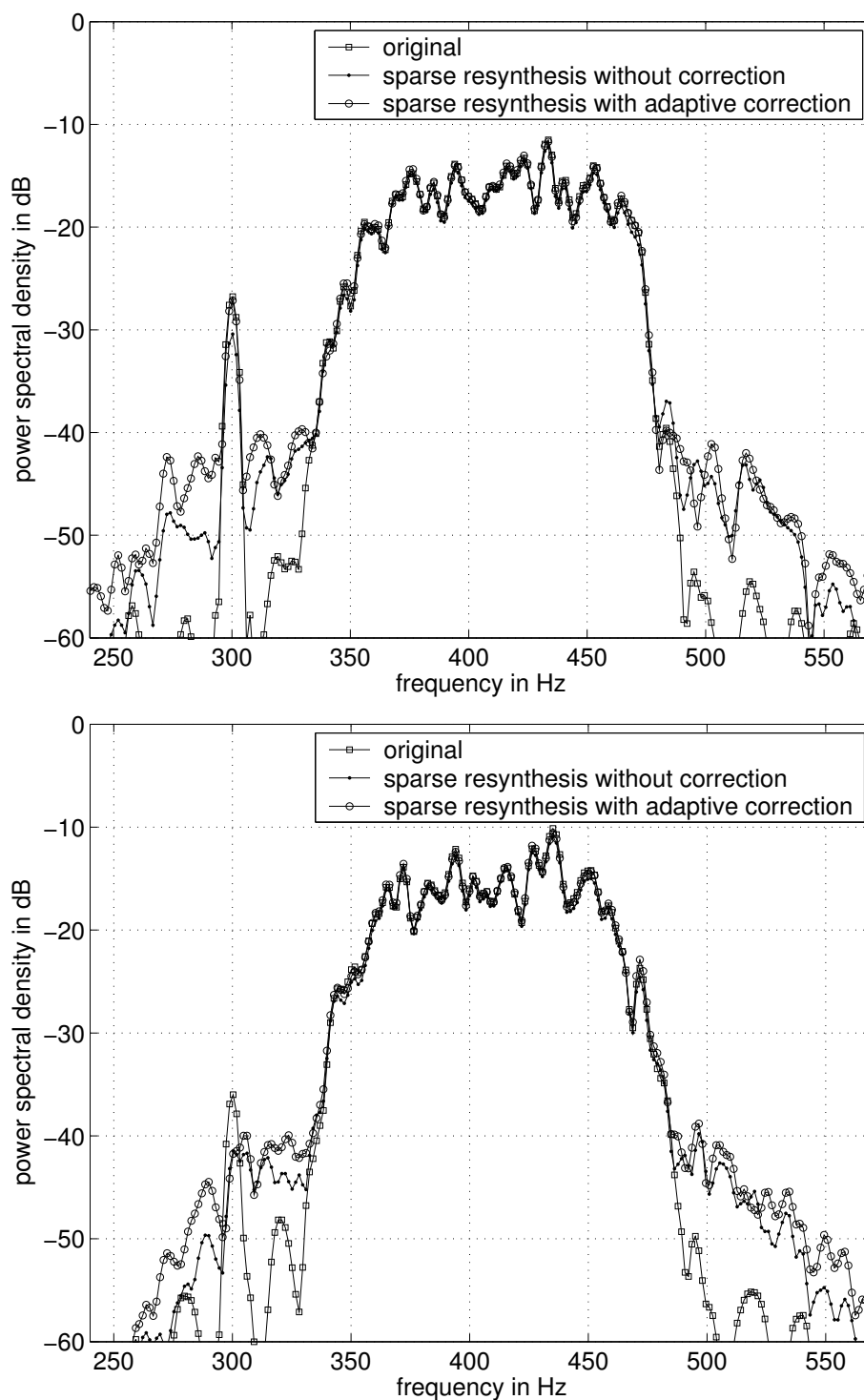


Figure 3.23: Power spectra of NBN masker plus tone probe. The upper plot shows the audible probe and the lower plot the inaudible probe. Comparison between the original signal, the resynthesized signal from the sparsified pulse representation without pulse amplitude correction, and with the simple adaptive amplitude correction method.

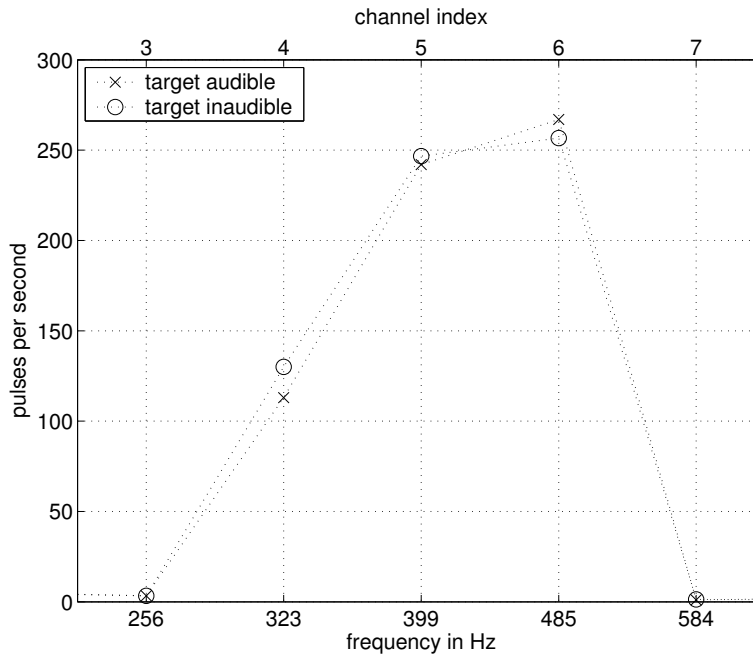


Figure 3.24: Number of remaining pulses in the sparsified 20-channel representation per second. Comparison between NBN plus audible 300 Hz tone and NBN plus inaudible tone.

these two channels between the two different stimuli. In channel 3, the number of pulses is almost zero in both cases. While in channel 4 the number of pulses is surprisingly high. The reason for this is that channel 4 is also needed to encode the close NBN signal, while channel 3 is already too far away. It should be mentioned here that the small difference between the plots for the two different stimuli in Fig. 3.24 originates from the fact that random noise has been used for both NBN signals and not frozen noise.

The fact that virtually no difference in the number of pulses between the stimuli with either audible or inaudible probe can be seen does not mean the masking model fails. We synthesize the sparse pulse code and are then able to recognize the difference. For the simulation, the improved pulse amplitude correction method from section 3.4 is used. In Fig. 3.25 we compare the power spectra of both resynthesized signals with the power spectra of the original signals. The upper plot shows the case with the audible probe. It can be seen that the resynthesized signal has a certain amount of reconstruction error. However, the tone is encoded sufficiently to be clearly present in the reconstructed signal. In the lower plot the PSDs for the inaudible 300 Hz tone are shown. The inaudible probe is almost completely hidden in the reconstruction error. But since the tone is not audible,

also the reconstruction error is hardly audible, as verified by an informal listening test.

It should be noted that the resynthesis of the complete (i.e., unreduced) pulse code yields virtually the same PSD as the original signal. That means that steep spectral edges can be well approximated (even with the gammatone filters used here, which do not have steep edges at all). This is possible because properly linearly combined shifted impulse responses of the synthesis filterbank can interfere constructively and of course also destructively. For the sparsified synthesis code, deletions of pulses result in a spreading of spectral contents since this interference is disturbed. The decisions of the masking model are made without knowing how the impulse responses interfere with each other. A pulse simply must be sufficiently dominant in a certain neighborhood to survive the sparsifying process. Thus, also dominant spectral components survive. On the other hand, spectral valleys are ‘flooded’ with the reconstruction error (cf. Fig. 3.25). The used masking and synthesis method ensures that the error remains masked by the dominant components.

### 3.5.2.2 **Narrowband Speech Stimuli**

The subjective quality obtained by resynthesis from the complete pulse representation and the influence of the used resynthesis method has never been evaluated systematically before. In this section we evaluate the reconstruction quality of the underlying analysis-synthesis method. Furthermore, we perform a subjective evaluation of the quality when the new masking model is incorporated into the system and the resynthesis is based on a sparsified pulse representation. This is to find out whether the deletion of pulses degrades the signal markedly or not. We present the results of a comprehensive listening test with narrowband speech stimuli.

The design of the listening test started with the question ‘Is there any perceptual difference between the original and the resynthesized signals?’. Therefore, a truly sensitive procedure, the ‘three-alternative forced-choice (3AFC) procedure with feedback’ was chosen. In this method, a subject was presented with three speech samples and was allowed to listen to them repeatedly. Two out of the three samples were the original speech signal and one had been processed by the proposed method. The task of the subject was to detect the processed version. The position of the processed sample was randomly selected. After each decision, the subject was immediately informed about the correctness of his/her choice. The high sensitivity of this procedure originates from the fact that a subject picks out the processed sample by excluding the two identical ones. In the case of only marginal perceptual differences between the original and the processed signal, where it is difficult or impossible to perceive the difference as a degradation, this procedure

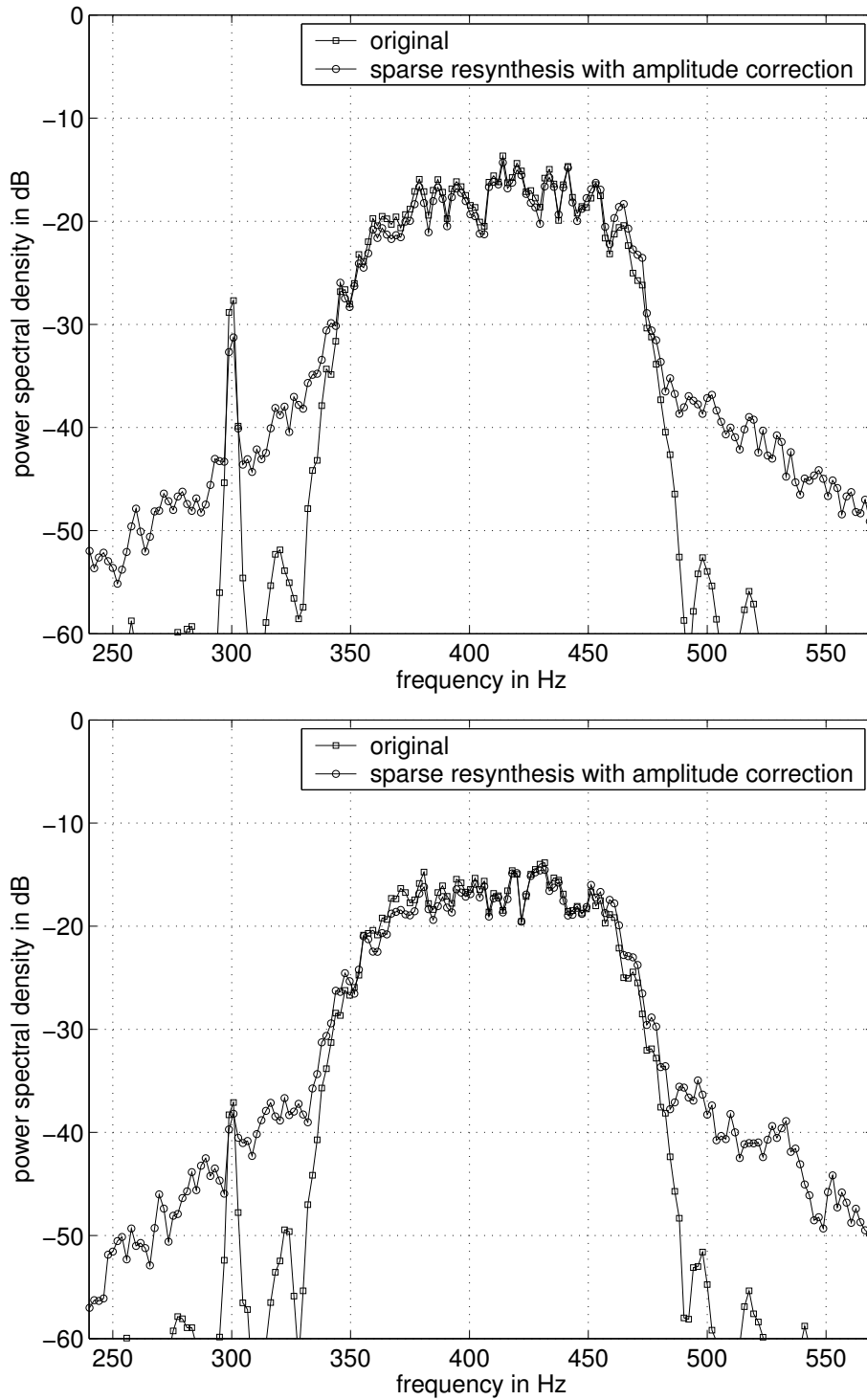


Figure 3.25: Power spectra of NBN masker plus tone probe. The upper plot shows the audible probe and the lower plot the inaudible probe. Comparison between the original signal and the resynthesized signal from the sparsified and amplitude-corrected pulse representation for the case of a 20-channel narrowband coder setup.

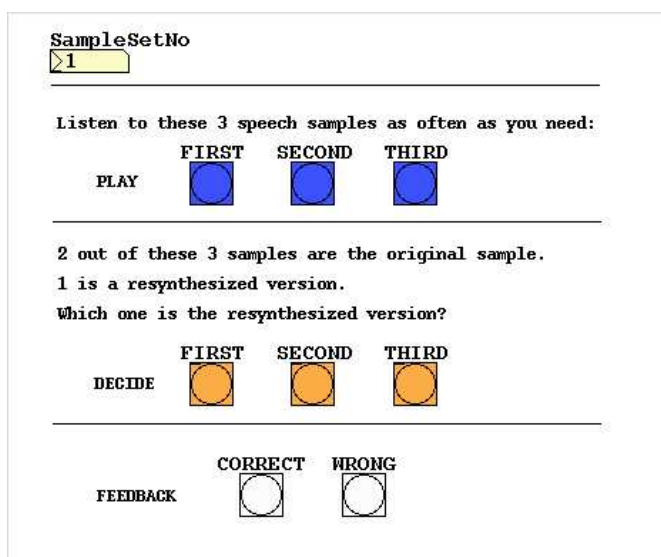


Figure 3.26: GUI of the three-alternative forced-choice listening test with feedback.

still yields useful decisions. Otherwise, an A/B comparison without knowledge of the reference (‘two-alternative forced-choice test’) would already result in a guessing game in such a situation. Another point that increases the sensitivity is the provided feedback, which results in a training of the subject. A positive feedback affirms possible cues a subject has already found and thus helps making further decisions.

The test was performed using the real-time audio processing environment ‘pd’ and the speech signals were presented to the subjects over headphones<sup>19</sup> in a silent office. Fig. 3.26 shows the graphical user interface (GUI) of the performed listening test. In addition to the control by mouse, keyboard control has been implemented to enable the subjects to keep their eyes closed<sup>20</sup>.

For the speech material 12 English sentences spoken by six native male and six native female speakers were chosen. The material had been recorded at a sampling rate of 16 kHz and with 16 bits accuracy. The duration of these speech samples varied from 2.2 to 4 seconds. The amplitude of all samples had been normalized to a maximum amplitude of 25% of full scale range. Since the listening test was intended to reflect the quality of the masking model and the accompanying pulse amplitude correction method, other possibly degrading factors were excluded. To achieve this, the whole system operated in an oversampled mode (as already done

<sup>19</sup>AKG K270 studio headphones on an external TASCAM US122 soundcard at maximum output level, at which a moderate SPL is produced.

<sup>20</sup>After the test, some subjects reported to achieve a higher level of concentration when they had their eyes closed while listening.

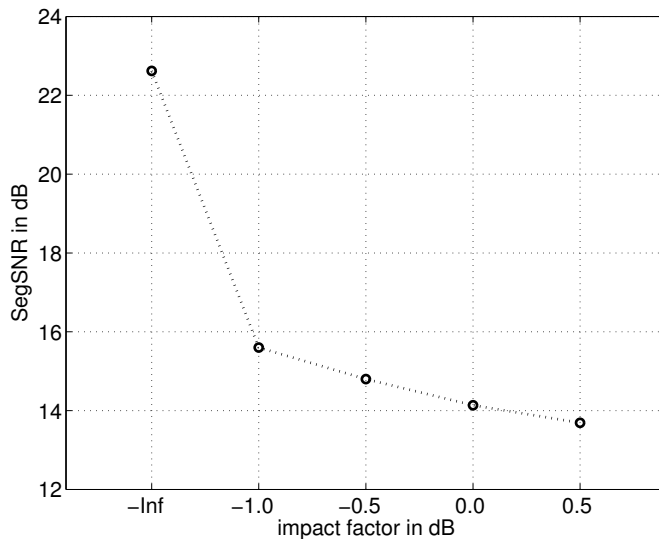


Figure 3.27: Averaged segmental signal-to-noise ratio (SegSNR) versus impact factor. ‘-Inf’ refers to the case where no masking model has been applied.

in section 3.5.2.1), and the sampling rate was chosen to be 16 kHz for narrowband-filtered speech. This also required that the speech material recorded at a rate of 16 kHz was preprocessed by narrowband filtering. This was done by passing the wideband speech signals through the cascade of the system’s analysis filterbank, synthesis filterbank, and linear-phase equalizer. The number of filterbank channels was 20 ranging from 100 Hz to 3600 Hz and the order of the FIR equalizer was 1024. For the resynthesized speech samples, five different cases were considered: the impact factor of the masking model was set to  $-1.0$  dB,  $-0.5$  dB,  $0.0$  dB, and  $0.5$  dB. The fifth case was a resynthesis from the complete pulse representation with non-adaptive pulse amplitude correction, i.e., without any masking model at all. Or equivalently, with a masking model that operated with an impact factor of  $-\infty$  dB.

Fig. 3.27 shows the segmental SNR averaged over all 12 speech samples for the five different resynthesis cases, i.e., for the five different impact factors. A considerable loss in SNR can be observed when the masking model is applied compared to the resynthesis of the complete pulse representation. But for the here-considered range of actual impact factors, a relatively linear relation between  $r^{(dB)}$  and the segmental SNR is obvious.

The listening test was performed by 32 subjects (five female and 27 male) ranging in age from 22 to 40. 18 subjects can be considered to be ‘experienced listeners’ because they are either professionally involved in speech and audio processing,

are sound engineering students<sup>21</sup> or graduated sound engineers with experience in recording and mixing, musicians or composers, or linguists.

26 out of the 32 subjects accomplished the complete test with all five coder settings corresponding to 60 comparisons (five coder settings times 12 speech samples). The other six subjects evaluated only four coder settings (impact factors  $-1.0$  dB,  $-0.5$  dB,  $0.0$  dB, and  $0.5$  dB), i.e., 48 comparisons without the signals resynthesized from complete pulse trains because these signals have been included in the test later.

The time the subjects required for the complete test varied between 35 and 60 minutes. The guideline value was 40 minutes. Two experienced subjects (one female and one male) had to be disqualified since they spent disproportionately more time (75 min and 120 min) for the test and consequently, their data is not included in the following statistics.

A true guessing game, which occurs when there is absolutely no perceptual difference between the original and the processed signal, should yield an average hit rate of  $1/3$  for the 3AFC procedure. Or equivalently, the probability for a wrong decision is  $2/3$ . In the case of 12 presented speech samples per coder setup this would correspond to 8 wrong decisions on average for a perfect system. The visualization of the number of wrong decisions gives a good representation of the subjective quality of the signal representation and resynthesis scheme. In Fig. 3.28 individual results of all 30 subjects are presented. Though considerably high variations are obvious among individual subjects, a clear tendency as a function of the impact factor can be seen. The results for the speech signals resynthesized from the complete pulse representation are shown in Fig. 3.29. Data from 24 subjects is shown here. For completeness, we also present the outstandingly good hit rates of the two disqualified subjects (see above): four and two wrong decisions in the categories ‘-Inf dB’ and ‘ $-1$  dB’ only.

Numbers of wrong decisions averaged on a sufficiently large set of subjects lie between 0 and 8 in our case of 12 decisions per class. As already mentioned, 8 can be reached only in case of an ideal system that does not cause any perceptual differences. On the other hand, 0 would only occur in case of a clearly degrading processing scheme. In that case a degradation category rating (DCR) [87] listening test should be performed. We use half the guessing threshold (4 in our case) to divide the range between 0 (‘always detectable’) and 8 (‘never detectable’) in two categories. The category below half the guessing threshold represents cases where detectability is probable by more than 50% while the category above shows cases with a detectability of less than 50%.

Fig. 3.30 shows the subject-averaged results of the 3AFC test. The number of

---

<sup>21</sup>Sound engineering students at our university have to pass a qualifying examination including a Seashore listening test.

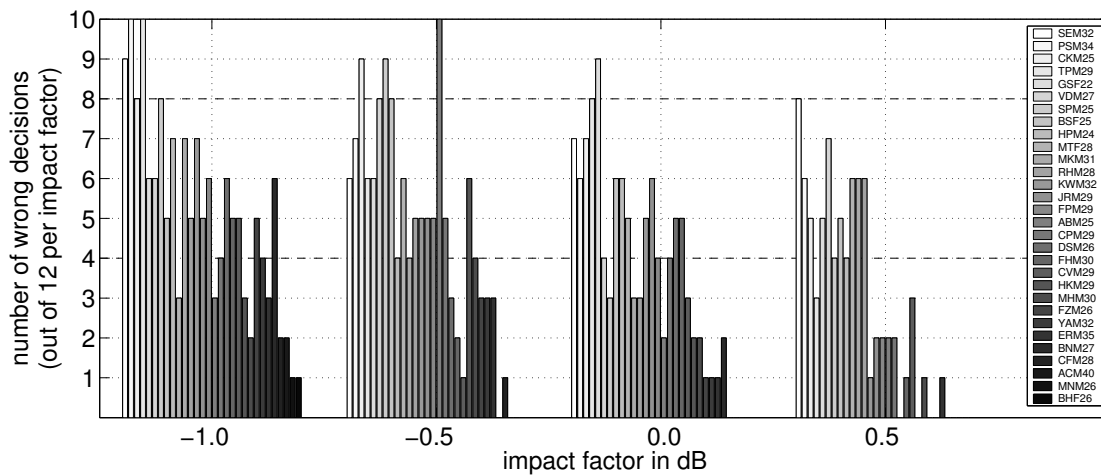


Figure 3.28: Number of wrong decisions of all individual subjects in the 3AFC listening test for the sparsified signal representation obtained by the proposed masking model with four different impact factors. The 30 subjects are distinguished by different shading and are ordered according to their overall number of wrong decisions (over the four categories shown here). The two dashed lines show  $2/3$  of the number of samples (=the 3AFC guessing threshold) and  $1/3$  (= half the guessing threshold).

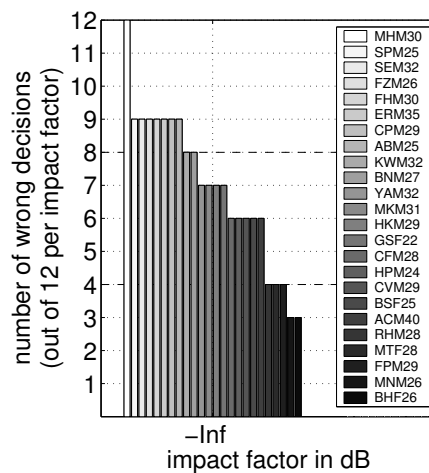


Figure 3.29: Number of wrong decisions of individual subjects in the 3AFC listening test for the case of a not sparsified signal representation (i.e., without the masking model). Here, only data from 24 subjects has been gathered (see text). The two dashed lines show  $2/3$  of the number of samples (=the 3AFC guessing threshold) and  $1/3$  (= half the guessing threshold).

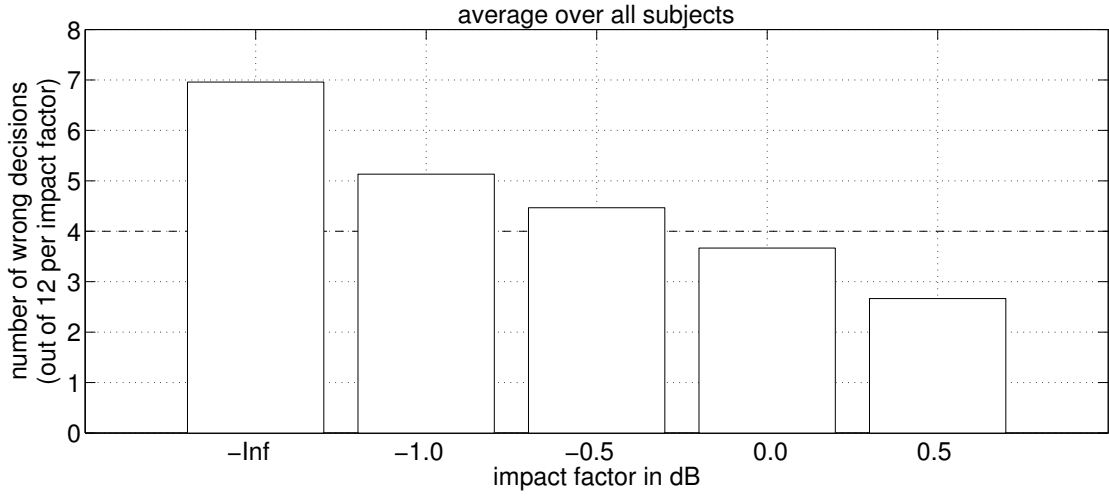


Figure 3.30: Average number of wrong decisions for all five cases. The number of subjects is 24 in the '-Inf' case and 30 in all other cases. The dashed line shows  $1/3$  of the number of samples per category (= half the guessing threshold).

subjects is 24 in the  $-\infty$  case and 30 in all other cases. The bar at  $-\infty$  (resynthesis from complete pulse trains) reaches 7 and approaches almost the guessing threshold of 8 wrong decisions. Therefore, we can consider the quality of a resynthesis from complete pulse trains as virtually identical to the quality of the original signal. The other four bars show a decrease with increasing impact factor. Similar as in the segmental SNR plot of Fig. 3.27, the number of wrong decision, which is an excellent measure for subjective quality, decreases rather linearly with an increasing logarithmic impact factor in the range considered here. But in contrast to the segmental SNR, the number of wrong decision does not exhibit such a noticeable step between  $-\infty$  and  $-1$  dB. The bars at  $-\infty$ ,  $-1$ , and  $-0.5$  dB lie above half the guessing rate (dashed line at 4). Therefore, we can consider the accuracy of the masking model as sufficiently high for practical coding applications up to an impact factor of  $-0.5$  dB. The bars at  $0.0$  and  $0.5$  dB lie already below half the guessing threshold.

The fact that the guessing threshold of eight wrong decisions cannot be reached, not even for the case of an unreduced pulse representation, implies that a (slight) perceptual difference between the original and the processed signals exists. However, we should keep in mind that this guessing threshold is an asymptotic limit, which would be obtained only for identical signals and an infinite number of decisions. We should also note the influence of possible systematic errors of the performed listening test. One such systematic error is that the processed signals have been passed through the cascade of analysis-synthesis filterbank and equal-

izer filter twice, while the original signals have been filtered only once (see above). This can result in differences especially in frequency regions outside the considered band of the speech signals.

For individual speech samples, a moderate spread can be observed. This can be seen in Fig. 3.31, where for the 12 speech samples reconstructed from four differently sparsified pulse representations the number of subjects with wrong decisions is visualized. The spread exists for a single impact factor as well as for the average over the four impact factors (shown by dashed bars). This means that a certain speech material is more or less vulnerable to degradations by the signal representation scheme. The guessing threshold in this statistical representation is 20 since the total number of subjects is 30.

The first six samples are spoken by females and the latter six by males. Some subjects reported after their listening test that they experienced greater differences when listening to speakers of a certain gender. However, the average over speech samples of either female or male speakers shows this tendency only for a very high impact factor of +0.5 dB where female speech undergoes a greater degradation. This is visualized in Fig. 3.32.

After the listening test, the subjects were asked to verbally describe the noticed differences or artifacts. The most common answers were that the resynthesized speech samples (or just parts of them) sounded rough, modulated (such as jitter in pitch), reverberant, dirty, or simply not as clean as the original. Also most subjects reported that the test in general was difficult and that it demanded exceedingly elevated concentration. Some mentioned that a series of decisions had been made according to some feeling and not according to a consciously perceived difference.

### 3.5.2.3 Justification for the Isolated-Pulse Simplification

The masking decision in the isolated-pulse BM masking model described in section 3.3 is always based on the excitation pattern produced by a single pulse of the code, the current masker or the current probe. This represents a coarse simplification of the actual model according to Fig. 3.4 since we do not compute complete convolutions with the real pulse trains before extracting the envelopes. In other words, we do not accumulate the BM vibrations caused by all<sup>22</sup> pulses of the synthesis code but rather ignore the contributions of all other pulses. On the other hand, this simplification enormously saves computational load. Since the extraction of the envelope in our model is a nonlinear operation, computing or predicting the response of the non-additive system for all possible different pulse train configurations as the input signal would be a highly time-consuming problem. Using isolated pulses, we are able to bypass this problem. Furthermore,

---

<sup>22</sup>Or at least all pulses in a sufficiently big neighborhood.

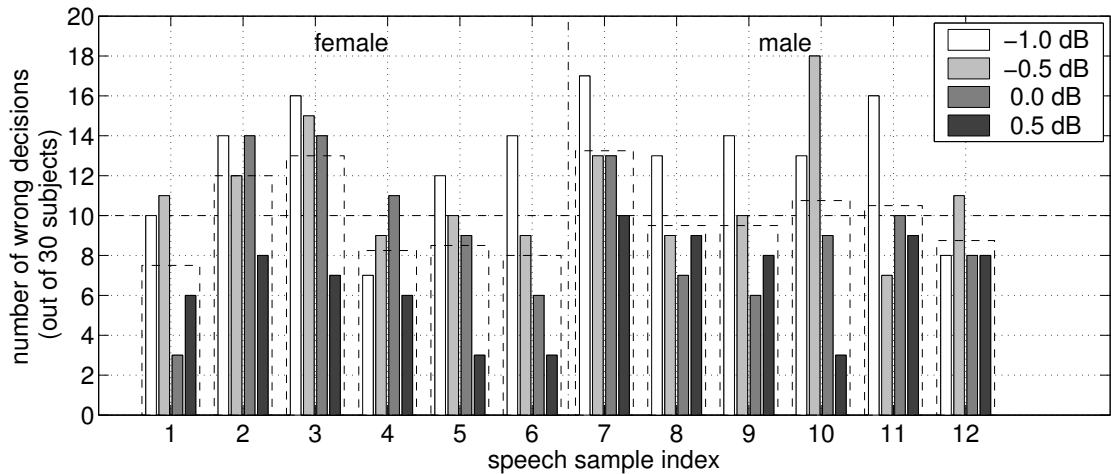


Figure 3.31: Speech sample dependency of the reconstruction quality. The bars are differently shaded for the four impact factors. The dashed bars show the average over all four impact factors. The dashed line shows  $1/3$  of the number of subjects (= half the guessing threshold).

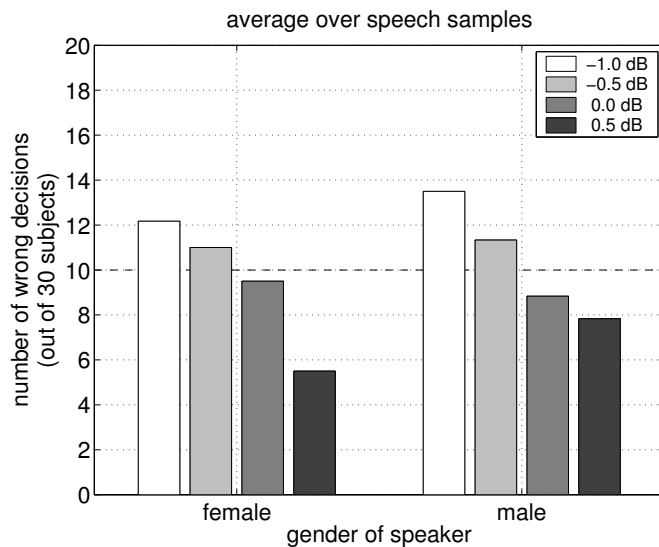


Figure 3.32: Speaker gender dependency of the reconstruction quality. The bars are differently shaded for the four impact factors. The dashed line shows  $1/3$  of the number of subjects (= half the guessing threshold).

it enables us to process the pulses in a reordered and sorted sequence (see section 3.3). The isolated-pulse simplification reduces complexity since it does not demand an exhaustive search algorithm to find a sparse pulse code.

To answer the question whether the proposed method, which is based on such a strong simplification, can still be accurate, we first can consult the results of the listening tests. These definitely show that the introduced perceptual distortions are small and often not audible for several listeners. For further analysis, we can calculate excitation patterns using the model of the transmultiplexer in Fig. 3.4 without simplifications, i.e., we do not treat pulses as isolated here. We calculate the excitation pattern caused by the complete synthesis code and compare it with the excitation pattern caused by the reduced synthesis code obtained by our isolated-pulse masking model. The masking model with an impact factor of  $r^{dB} = -1.0$  dB is used to sparsify the auditory representation. For the reduced synthesis code, the amplitude correction due to masking is omitted for this representation since the decisions of the isolated-pulse masking model are made before the correction stage. The two excitation patterns are converted into dB<sup>23</sup> before we compare them. The result of this experiment is presented in Fig. 3.33. The first plot shows the response of the transmultiplexer (which is a BM excitation pattern) caused by the complete pulse representation of the initial and central part (270 ms) of the word ‘rag’ taken from a sentence spoken by a male. The difference between the two logarithmic excitation patterns is presented in the second plot (i.e., the complete excitation has been subtracted from the reduced one). As revealed in this plot, local differences in excitation can reach more than 50 dB! A positive difference means that the reduced synthesis code yields a higher excitation than the complete one. Sometimes local over-excitation arises, but under-excitation dominates the second plot. While over-excitation usually occurs in envelope valleys, it is more difficult to deduce such a connection for under-excitation. Regions of under-excitation often appear temporally less concentrated than over-excitations and are found where a series of pulses has been deleted. This plot shows furthermore almost no differences at constantly excited regions, i.e., in channels without envelope fluctuations, such as in channels 0, 2, 5, and 7.

The high local differences in BM excitation between the complete and the sparsified pulse representation reveals that the isolated treatment of pulses yields a rather poor approximation of the underlying excitation pattern model. The impact factor ( $-1$  dB in this experiment), which is the maximum excitation difference for a pair of isolated pulses, is definitely not reflected in the actual (i.e., properly calculated) excitation difference. However, the results of the listening test show something different: the speech sample chosen for the experiment described above

---

<sup>23</sup>Here, the conversion to dB can be seen as an approximation of the compressive behavior in the cochlea.

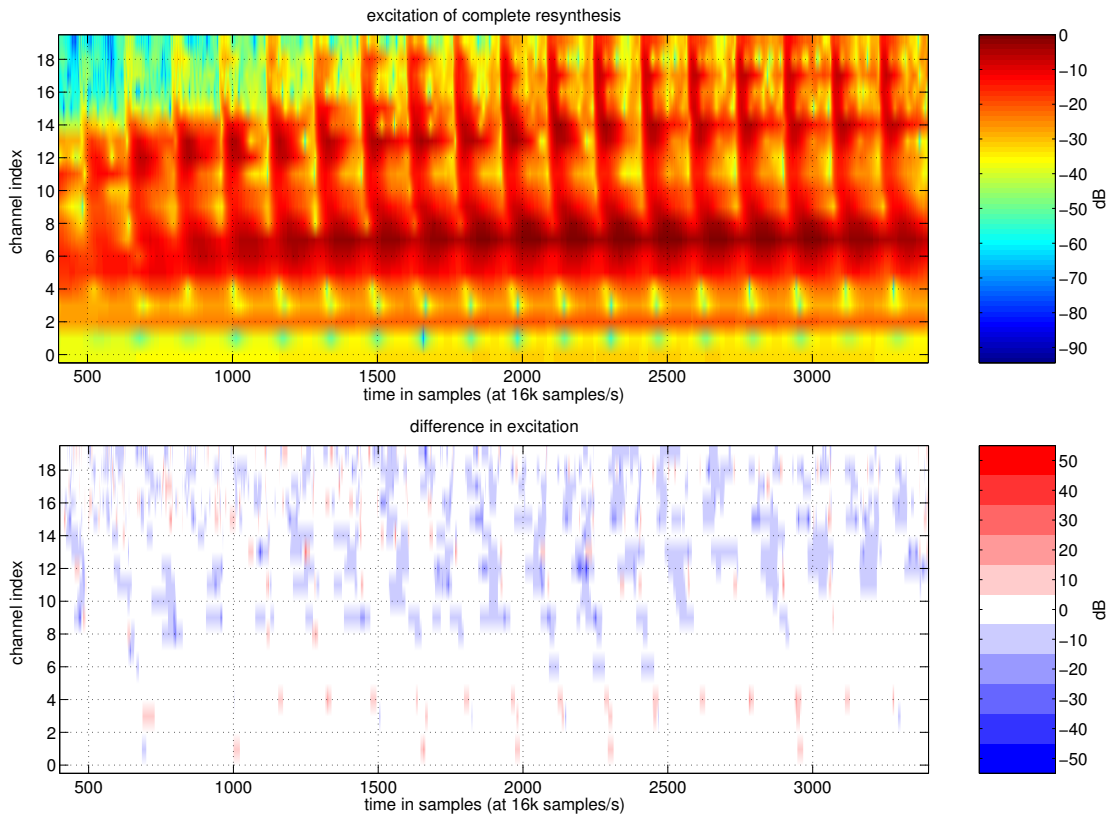


Figure 3.33: Top: Transmultiplexer response (BM excitation) to the complete pulse representation (taken from a segment of the word ‘rag’ spoken by a male). Bottom: Difference between the logarithmic BM excitation of the complete pulse representation and the logarithmic BM excitation of the reduced and not amplitude-corrected pulse representation.

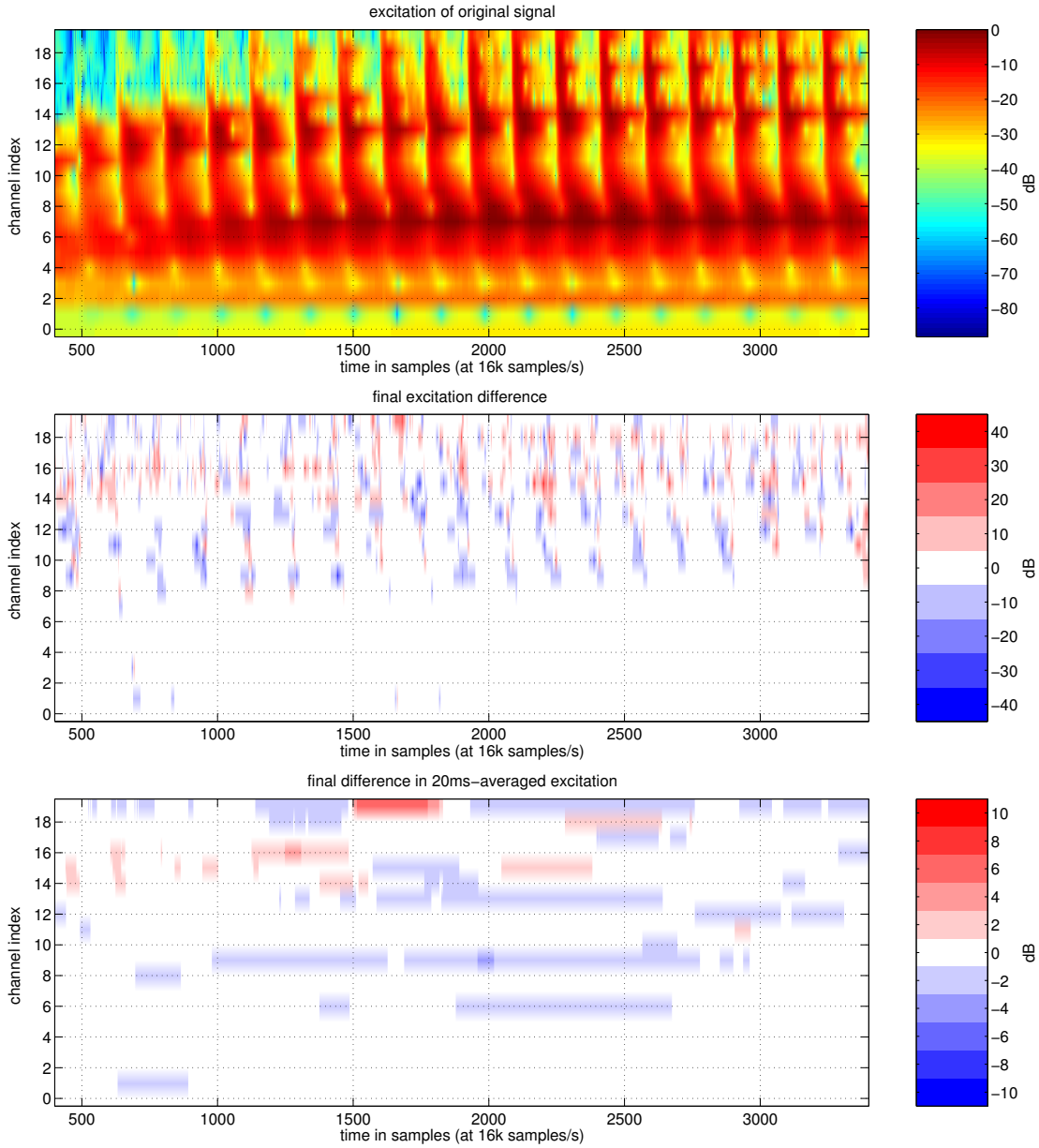


Figure 3.34: Top: BM excitation of the original signal (same segment as in Fig. 3.33). Middle: Difference in logarithmic BM excitation between the original signal and the signal resynthesized from reduced and amplitude-corrected pulse trains and finally equalized. Bottom: Difference between the logarithmic, 20 ms-averaged BM excitations.

is the sample with index 7 of the 3AFC listening test. As shown in Fig. 3.31, 17 subjects out of 30 could not correctly identify the resynthesized version among the three speech samples under test. There seems to be a discrepancy between the presented differences in BM excitation and the results of the listening test.

The above demonstrated excitation differences do not consider the pulse amplitude correction that follows the masking stage since the decisions for deletions in the masking model are made without modifying the pulse amplitudes. Also the final equalizer of the decoder has not been incorporated into the transmultiplexer. For the 3AFC listening test, listeners were of course presented with signals that had been resynthesized from amplitude-corrected pulses and finally filtered by the FIR equalizer. Additionally, listeners were presented with the original signal for the reference and not with a resynthesis from complete pulse trains. For completeness and also for providing difference measures that are better in accordance with results of the listening test, Fig. 3.34 shows the excitation produced by the original signal in the first plot. The second plot shows the difference between the logarithmic excitation of the original signal and the logarithmic excitation of the signal resynthesized from a sparsified pulse code where also the aforementioned two stages are considered. Local differences in excitation can still reach values of almost  $\pm 40$  dB but an overall reduction can be seen when compared with Fig. 3.33.

Usually, the assumed criterion for masking, when classical excitation pattern models are used, is a maximum excitation difference of only 1 dB in all filterbank channels. Admittedly, stationary stimuli are primarily applied to these models and long-term averages<sup>24</sup> are considered. The envelope extraction in our excitation model can be seen as a kind of low-pass filtering. However, its passband is much wider<sup>25</sup> than that of a corresponding low-pass filter for computing long-term averages in classical excitation pattern models. To get a difference measure that is closer to classical excitation pattern models, we adapt our transmultiplexer by adding low-pass filters after the envelope extraction stages (for comparison purposes only). For simplicity, we choose a moving average FIR filter that averages over 20 ms<sup>26</sup>. When these averaged excitations generated by the complete and the reduced pulse code are compared (again on a dB scale), we naturally obtain much smaller differences as shown in the third plot of Fig. 3.34. The averaged excitation shows many areas without a noteworthy difference with the exception of the local difference of around 9 dB in channel 19. The differences presented here are still higher than the classical 1 dB threshold.

In the first instance, an explanation for the still present discrepancy seems to be that the generation of Hilbert envelopes is not realistic to be performed in the

---

<sup>24</sup>Such as in the *power-spectrum model of masking* [34], where the long-term integration is inherent in the power spectrum.

<sup>25</sup>The effective cut-off frequency originates from the bandwidth of the auditory filter.

<sup>26</sup>Cf. the final low-pass filter in the model of [19].

auditory pathway since it requires filters with long memory<sup>27</sup>. On the other hand, it has been shown in [88] that AM signal envelopes can be well approximated using the nonlinear Teager-Kaiser energy operator, which only needs two samples of memory. Even if the Hilbert envelope extractor of our model in Fig. 3.4 is replaced by a half-wave rectifier and a low-pass filter<sup>28</sup> (as in many other auditory models, e.g. [47, 19]), still local differences up to almost 40 dB can be observed.

The relatively high differences in logarithmic BM excitation shown in Fig. 3.33 (and again in Fig. 3.34) on the one hand, and the high subjective, perceptual quality as confirmed by the listening test on the other hand, suggest that difference measures based on classical excitation pattern models are not suitable for stimuli that cause envelope fluctuations, at least not when these measures are used to derive decisions for detectability on the basis of fixed thresholds (such as the classical 1 dB threshold). Such complex stimuli (e.g., speech or music) can locally mask a considerable amount of reconstruction errors—much more than expected. Our pulsed signal representation contains the temporal fine structure of the input signal accurately, which enables us to make masking decisions on a localized basis. In this way, we are allowed to introduce high but localized reconstruction errors. The proposed isolated-pulse masking model exploits this point in a simple way.

For coding applications, the obvious drawback of the rather classical perceptual difference measures that incorporate averaging is that they cannot tell us where precisely we can hide quantization errors. The temporal fine structure of the signal would still be observable in the envelopes of the auditory filter outputs, but no longer in averaged excitations. So, quantization errors with different temporal distributions could result in the same average distortion but would yield different actually perceived distortions. However, the signal representations used in existing coding schemes do not possess a resolution that is high enough to exploit local envelope maxima. Another important issue is that these classical distortion criteria lead to iterative analysis-by-synthesis procedures, which are computationally expensive.

## 3.6 Consideration of Nonlinear Effects of Perception

It is well known that there are several highly nonlinear mechanisms involved in human auditory perception. Temporal effects play an important role when nonstationary signals (such as speech or music) are processed and should be considered carefully in coding. One observed nonlinear effect is the dependency of the masking

---

<sup>27</sup>The ideal Hilbert transformer is acausal and has an infinite impulse response.

<sup>28</sup>A cut-off frequency of 1 kHz is used.

efficiency on the delay of a short probe after the onset of a long masker (*overshoot effect* [33]). Another observation is the dependency of temporal post masking on the duration of the masker [18]. The reason for these dependencies seems to be the adaptation of the hearing system, especially of the inner hair cells, to the presented stimulus. Physiologically plausible hair-cell or rather synapse models have been proposed by Meddis [49, 50] or by Seneff [47].

In the functional model proposed by Dau et al. in [19], cascaded adaptation loops with different time constants are used after the envelope generation in the auditory channels to model the adaptation effect. They also show that their model predictions are in good agreement with psychoacoustic data for several masking scenarios [80, 89, 90]. These loops are automatic gain controllers (AGCs), where the gain factor is obtained from the charging state of a capacitor. In appendix B we summarize the implementation of these adaptation loops. We should note that in the adaptation circuit of [19, 89] a low-pass filter with a relatively long time constant of 20 ms is used as the final stage. In Dau's later work [80, 90], this low-pass filter has been removed<sup>29</sup>. Therefore, we also use the loops without the final low-pass filter for our simulations.

The response of the adaptation loops can be interpreted as the *sensitivity* of the hearing system. Fig. 3.35 shows an example generated from an English sentence spoken by a female. The shown segment of 150 ms duration contains 'to carry' (starting in the closure interval of the stop consonant and ending while the tongue moves backwards). The enhancement of onsets can clearly be observed. During the more stationary segments, the adaptation happens and the output of the adaptation circuit is reduced.

We can use the sensitivity measure to derive a correction factor that is incorporated in the masking model to adapt the efficiency of masking according to the temporal characteristics of the signal. The definition of the term 'correction factor' postulates that it has to be equal to 1 when there is nothing to correct. For the temporal adaptation, we do not want to correct the masking efficiency in stationary situations. Therefore, we need to normalize the sensitivity measure by the stationary response. For a constant input signal, the cascaded adaptation loops asymptotically behave like a static power-law nonlinearity (see appendix B). The generation of the correction factor is illustrated in Fig. 3.36. The so normalized correction factor is bounded between the values  $-10.5$  dB and  $6.5$  dB for real speech data.

Fig. 3.37 shows how the adaptation circuit based amplitude correction is incorporated into the coding system. Since the adaptation loops need the signal envelope of the filterbank channel as input signal, we create a second signal path

---

<sup>29</sup>Actually, the low-pass filter has been replaced by a filterbank covering a frequency range up to 1 kHz to analyze the envelope modulations

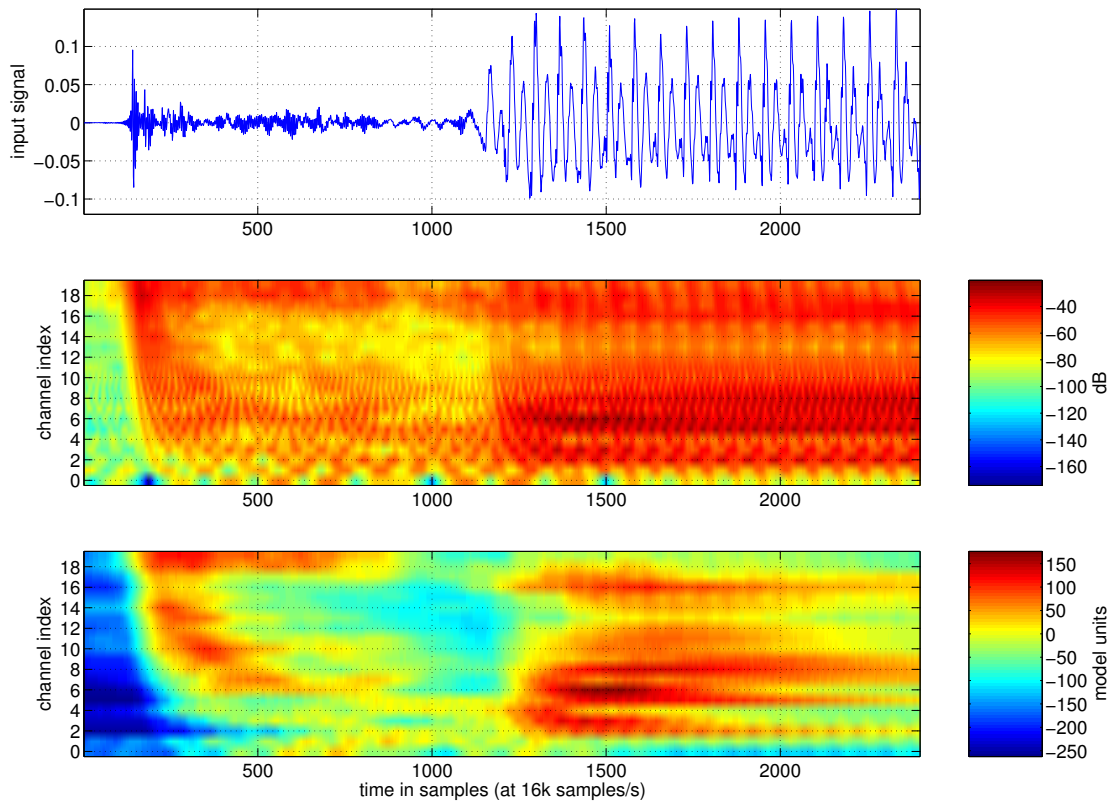


Figure 3.35: Top: Segment of the speech sample ‘to carry’ spoken by a female. Middle: BM excitation (half-wave rectified and low-pass filtered outputs of the auditory filterbank). Bottom: Response of the adaptation circuits to the BM excitation.

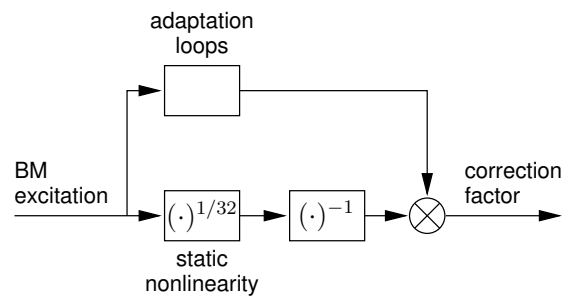


Figure 3.36: Computation of the correction factor to account for the adaptation to the stimulus. The output of the cascaded adaptation loops is normalized by the output of the static power-law nonlinearity to build the correction factor.

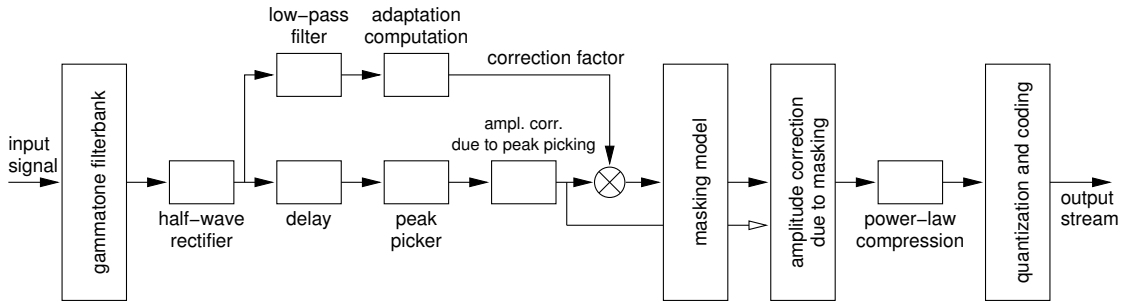


Figure 3.37: Incorporation of the adaptation loops in the coder.

per auditory channel in addition to the signal path where the peak picker generates the pulses. The half-wave rectification stage can be shared by both branches. In order to compute the envelope in each filterbank channel, a low-pass filter<sup>30</sup> with a cut-off frequency of 1 kHz is applied as in [19]. Since the filterbank of the coder does not account for the transmission characteristics of the outer and middle ear, the OME weighting is performed here in the additional signal branch. The OME-weighted envelope (BM excitation) is used as the input for the adaptation circuit to compute the correction factor as shown in Fig. 3.36. In the peak picker signal branch the half-wave rectified signal is delayed<sup>31</sup> by the group delay of the 1 kHz low-pass filter before the pulses are generated. This ensures synchrony between the pulses and the correction factor. Finally, the pulses are weighted by the correction factor.

The weighting yields a different behavior of the masking model according to the temporal adaptation to the signal. A first change is that the weighting alters the level difference between a masking pulse and a probe pulse. This can especially be observed when the signal level varies abruptly such as at onsets or at offsets. For instance after the offset of a strong signal that was long enough to charge the capacitors of the AGCs, the correction factor becomes less than one, i.e., it will further attenuate the anyway small pulses after the offset. Thus, the effective temporal masking patterns are widened compared to the original pulse BM excitation patterns. This widening and, thereby, flattening of the patterns along the time axis is especially needed for the channels with higher frequencies as it has already been discussed in section 3.3. Fig. 3.38 shows an example where the effective widening of the masking patterns can be observed. For this experiment, the 20-channel narrowband setup of the filterbank is used in an oversampled mode

<sup>30</sup>The impulse response of this filter has to be purely positive to guaranty purely positive envelopes.

<sup>31</sup>For a low-delay implementation, the usage of the Teager-Kaiser energy operator [88] can be considered to compute the envelopes.

(16 kHz sampling rate) as used for the listening test in section 3.5.2.2. A sinusoid with a frequency of 1 kHz and a total duration of 1 second is used as the input signal. After 500 ms, the amplitude of the sinusoid is changed by  $-60$  dB. The figure shows a 50 ms segment of the input signal that covers the amplitude change (upper plot). In the middle plot the sparsified pulse representation obtained by the isolated-pulse masking model with an impact factor  $r^{(dB)} = -0.5$  dB without the nonlinear correction factor (i.e., obtained by the system of Fig. 3.9) is shown. For comparison, the lower plot presents the sparsified pulse representation obtained by the masking model with a prior weighting of the pulse amplitudes using the nonlinear correction factor (i.e., obtained by the system of Fig. 3.37. It can be observed that in channel 10 and 11 the interval of deleted pulses is widened by about 6 ms.

Another change in the behavior of the masking model is a slightly different order after sorting the pulses according to their amplitudes to determine the sequence of masking pulses. The incorporation of the nonlinear adaptation helps to hide the reconstruction error even better in terms of accurate temporal noise shaping.

The inversion of highly nonlinear circuits is known to be very challenging and often even impossible when only their output signal and no side information is available. In [64, 66], AGC inversion is based on projections onto convex sets. But it should be mentioned that there is no necessity to invert the AGCs in the decoder. The nonlinear weighting changes the output of the masking model that acts as a bit-allocation algorithm since it tells us where in the time-frequency domain we have to spend bits (surviving pulses) and where not (deleted pulses). Actually, there is no further benefit in keeping the weighting after this bit-allocation stage. Therefore, the additional weighting can already be undone in the encoder. Note, that the pulse amplitude correction stage after the masking model, which has been described in section 3.4, restores the energy distribution of the unsparsified pulse trains. Thus, also the nonlinear weighting can be undone by that stage. We notice that the decoder does not need to be modified when the nonlinear adaptation is incorporated into the encoder in the proposed way.

We performed experiments with narrowband-filtered speech signals and the 20-channel filterbank setup with center frequencies from 100 Hz to 3600 Hz. For the speech material, the 12 original samples of the 3AFC listening test of section 3.5.2.2 are used. The sparsified pulse representation obtained by the isolated-pulse masking model with prior nonlinear weighting according to Fig. 3.37 is compared with the pulse representation obtained without the nonlinear weighting. The impact factor of the masking model is set to  $r^{(dB)} = -0.5$  dB. Depending on the chosen input speech sample, the benefit of incorporating the adaptation circuit is a reduction in the number of pulses between 15% and almost 21% while producing the same perceptual quality after the resynthesis as confirmed by an informal listening test.

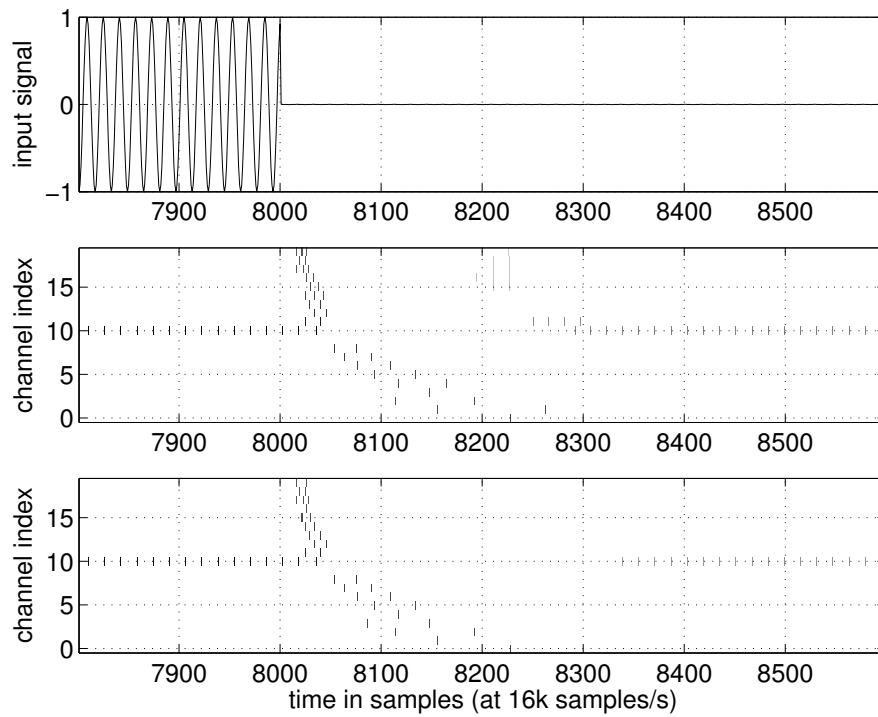


Figure 3.38: Demonstration of the changed behavior of the masking model. Top: 1 kHz tone that changes its amplitude abruptly by  $-60$  dB after 500 ms. 50 ms around this change are shown. Middle: Sparsified pulse representation without the nonlinear correction factor. Bottom: Sparsified pulse representation with the nonlinear correction factor.

For comparison, we also performed simulations where the final low-pass filter of the adaptation circuit was not omitted. The weighting according to this smoothed sensitivity does not yield a noteworthy increase in sparsification performance. For impact factors  $r^{(dB)} = 0$  dB and  $r^{(dB)} = 0.5$  dB, the benefit is only a slight reduction in the number of pulses of the sparsified representation by about 3% to 4% compared with the output of the masking model without the nonlinear correction factor. For a smaller impact factor of  $r^{(dB)} = -0.5$  dB, the number of pulses is reduced by only 1% to 3%. This finding underlines the importance of preserving the signal envelopes in the auditory channels. When the envelopes, which are band-limited by the bandwidth of the corresponding auditory filter, are low-pass filtered with a cut-off frequency lower than this bandwidth, important signal features are lost.

The simulations show that the incorporation of dynamic nonlinearities that model the temporal adaptation of the hearing system to the stimulus increases the masking efficiency of the proposed masking model. The so obtained pulsed signal representation is appreciable sparser than without considering the nonlinear behavior. The proposed method to exploit the nonlinear behavior in terms of a more accurate temporal noise shaping for the sparsification stage does not require an inversion of the nonlinearities in the decoder—the proposed amplitude correction scheme that restores the energy distribution of the original, unsparsified pulse representation is also able to undo the signal-dependent weighting.

### 3.7 Conclusions

In this chapter, the so-called transmultiplexer, which is formed by the decoder and the human listener’s hearing system, has been introduced to perceptual-domain coding. This transmultiplexer provides a perceptual analysis-by-synthesis framework. However, to bypass the problem of computationally expensive or even not feasible exhaustive search algorithms to find an optimal, sparse synthesis code, we have suggested a masking model that is based on the transmultiplexer with isolated pulses as input. This masking model yields a sparse synthesis code, which consists of locally dominant pulses of the signal’s auditory representation. In terms of sparsification capability, the proposed masking model outperforms the comparable model of [70]—we are able to almost halve the number of remaining pulses of [70]. To ensure the sparsified synthesis code causes a basilar membrane excitation in the human listener close to the one caused by the unreduced auditory representation, a non-iterative pulse amplitude correction scheme, which is also based on the transmultiplexer, has been proposed.

To evaluate the subjective quality of the reconstructed signals, a three-alternative forced-choice listening test has been performed. It has been found that a

resynthesis from the complete pulsed auditory representation is practically not distinguishable from the original narrow-band speech signal. The subjective quality of speech signals reconstructed from a sparsified synthesis code decreases rather linearly with an increasing impact factor of the proposed masking model. The reconstructed signals are indistinguishable from the originals in more than 50% of the cases for impact factors  $r^{(dB)} \leq -0.5$  dB. Therefore, the perceptual quality is sufficient for practical high-quality coding applications.

We can conclude that the independent treatment of dominant pulses in the proposed isolated-pulse excitation pattern model is not at all a coarse simplification of human auditory perception as it might have appeared initially, when we started to approximate a conventional excitation pattern model. Considering the temporal fine structure of a signal is important since it allows us to mask a considerable amount of reconstruction errors. The experiments have shown that these local errors evaluated on the modeled basilar membrane excitation can reach about 40 dB and are still inaudible for more than half of the subjects of the listening test. This masking capability can be exploited well by the proposed sparse pulsed signal representation based on the simple isolated-pulse masking model. The main benefit of the proposed method is that a single control variable, the impact factor, well reflects the finally perceived distortion. Thus, it enables an efficient coding scheme that is not based on an iterative and computationally costly analysis-by-synthesis procedure.



# Chapter 4

## Auditory-Domain Quantization: First Attempts

### 4.1 The Encoding of Sparse Signals

The auditory representation obtained by the physiologically motivated subband coder described in chapter 2 consists of pulse trains. Pulse trains are by nature sparse signals. When we incorporate the masking model described in chapter 3 to eliminate most of the less dominant pulses, the auditory representation becomes even sparser. To efficiently describe sparse signals, we consider only the non-zero samples and represent them as position-amplitude pairs.

The pulse amplitude is originally a continuous variable (or at least finely quantized, e.g., with 16 bits). On the other hand, the position variable is an integer since it is specified as a number of samples. It is important to mention that errors of position and amplitude depend on each other. This causes difficulties for lossy encoding of the position-amplitude pairs, since position errors cause more distortion when the pulse amplitudes are large. Therefore, we cannot quantize the position and the amplitude information independently of each other. For an encoding of the pulse trains at a low bit rate, the amplitude undoubtedly needs to be quantized (or re-quantized). When the position information is kept unchanged, the amplitude can be quantized independently. However, to decrease the bit rate further, the position information also has to be (re-)quantized. The joint position-amplitude quantization requires an extension of classical rate-distortion theory [91].

This chapter analyzes the auditory pulse representation, gathers statistics, and presents entropy considerations. Section 4.2 deals with the independent encoding of the pulse position information. The focus is on lossless coding of the positions using distance encoding in section 4.2.1. Section 4.2.2 considers the independent

quantization of the positions by means of downsampling. For the quantization of the pulse amplitudes, it is important to investigate amplitude statistics. This is done in section 4.3 in order to give hints for the design of block-scalar quantizers. Section 4.4 deals with vector quantization. We propose a new similarity measure for sparse signals that reflects the dependencies between position and amplitude errors well. Furthermore, we suggest to combine distance encoding and vector quantization to achieve a variable block length coding in order to efficiently encode the pulse train signals. We also discuss results of experiments to show whether the sparsification of the auditory representation brings an advantage for its encoding or not. This chapter does not present a complete encoder with finalized bit allocation and evaluated subjective quality, but summarizes different approaches and their difficulties to provide a basis for further research work.

## 4.2 Pulse Positions

### 4.2.1 Lossless Encoding

The frequency of a firing pulse train is close to the center frequency of the basilar-membrane filter of the corresponding auditory channel for typical input signals. Therefore, the distance between two adjacent pulses will usually be close to the reciprocal of the center frequency (quantized at integer multiples of the unit of a sampling interval  $1/f_s$ ). This suggests to encode the distances between adjacent pulses instead of their absolute time indices. Let us consider a single channel and let  $m \in \mathbb{N}_0$  be the pulse index. Then,  $n_m$  denotes the pulse location in samples at the sampling rate  $f_s$ . The pulse distances are defined as  $d_m = n_m - n_{m-1}$  with  $n_{-1} = 0$  for the initialization. The distance in samples is equal to the number of zeros between two consecutive pulses plus 1. Therefore, the distance encoding is similar to *run-length encoding* (RLE). In contrast to RLE, the number of consecutive non-zero samples need not be encoded since it is always 1 (i.e., the width of a pulse).

Plotting the histograms over the pulse distances shows monomodal distributions when the pulse trains prior to the reduction by the masking model are evaluated (see Fig. 4.1, upper plots). Hence, the obtained pulse distance codes have a very low entropy, which can be exploited by an additional entropy coder (e.g. Huffman or arithmetic coder). When we analyze the pulse trains after the pulse deletion by the masking model, the histograms become multimodal (see Fig. 4.1, lower plots) since now, not only distances according to the reciprocal of the center frequency occur, but also integer multiples of it.

The deletion of pulses results in a higher entropy of the distance codes. This is visualized in Fig. 4.2. Particularly in high-frequency channels the entropy is

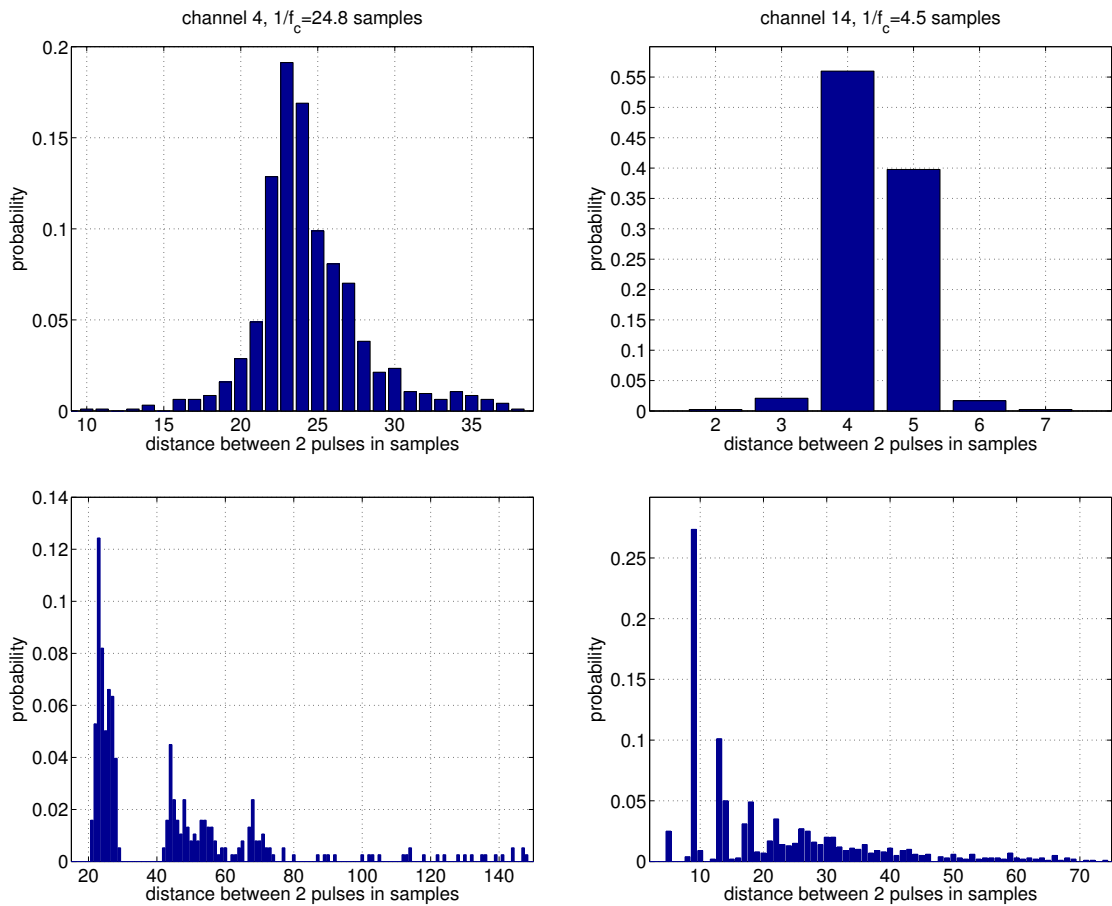


Figure 4.1: The probability of a certain distance between two adjacent firing pulses in channel 4 with a center frequency  $f_c = 322.5$  Hz (left) and in channel 14 with  $f_c = 1,778$  Hz (right). The diagrams in the upper row analyze the unreduced pulse trains, which were obtained without the masking model. In the lower row the pulse trains after the masking model are evaluated. The impact factor  $r$  has been set to 1.05 which results in a reduction by 60% (from 327 to 131 pulses/s) in channel 4 and by 81% (from 1,813 to 347 pulses/s) in channel 14.

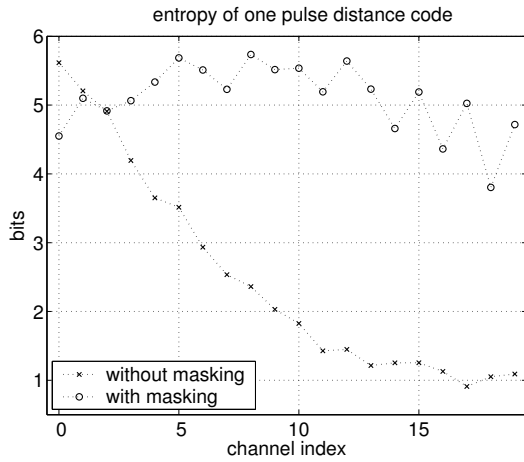


Figure 4.2: Entropy in bits of a distance code for the 20-channel coder for 8 kHz sampling rate. The impact factor  $r$  has been set to 1.05.

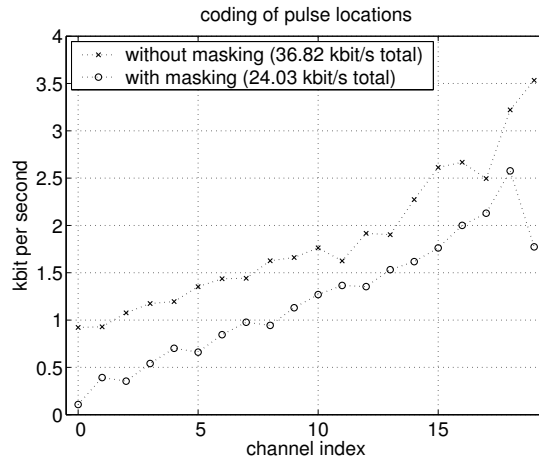


Figure 4.3: The corresponding number of bits per second needed to losslessly encode the pulse positions using distance encoding and ideal entropy coding (see Fig. 4.2) in the individual channels. The sum over all channels is about 24 kbit/s with masking compared to 36.8 kbit/s without masking.

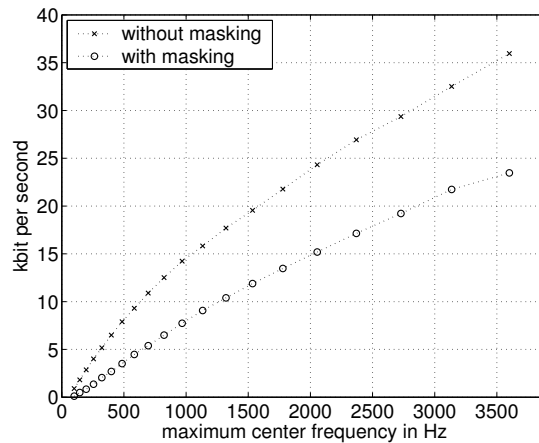


Figure 4.4: Bitrate necessary to losslessly encode the pulse positions (distance encoding and ideal entropy coding) as a function of the considered bandwidth (cumulative sum over data from Fig. 4.3). The center frequencies in Hz of the 20 channels from 100 Hz to 3600 Hz are shown on a linear axis.

significantly increased (from 1 bit to about 5 bit per distance symbol). Nevertheless, as demonstrated in Fig. 4.3, the overall amount of data has been reduced since the length of the sequences containing all distance codes has been decreased considerably. For the 20-channel narrowband coder, the optimally entropy-coded pulse distances produce an average data rate of about 24 kbit/s when the masking criterion with an impact factor of  $r = 1.05$  (0.42 dB) is used. Compared to the data rate of 36.8 kbit/s of the unreduced pulse trains, masking achieves an improvement of about 35%. It should be noted that these rates are needed for a lossless encoding of the pulse positions.

Since the overcompleteness of the auditory representation exists especially in high-frequency channels, we should investigate how much additional information needs to be encoded when the signal bandwidth is increased towards higher frequencies. For a uniformly sampled signal, the data rate increases linearly with increasing bandwidth. Fig. 4.4 presents the cumulative sum over the bit rate data from Fig. 4.3 on a linear frequency scale. The curve for the complete auditory representation (without masking) starts relatively steep at very low frequencies, flattens towards 1000 Hz, and continues to increase rather linearly beyond 1000 Hz with about 9 bit/s per Hz. On the other hand, the curve for the reduced pulse representation (with masking) starts rather linearly with about 9 bit/s per Hz right from the beginning and becomes flatter above 1500 Hz. This shows that the position side information becomes less important with increasing frequency.

### 4.2.2 Lossy Encoding

It has to be noted that the above investigations are for lossless encoding of the pulse locations only. For efficient encoding, the pulse positions can be (re-) quantized, which introduces additional distortion. One difficulty of the related rate-distortion behavior is that the individual distortions have to be weighted by the corresponding pulse amplitude [92]. Another difficulty is the yet unclear perceptual relevance of position errors. For high-frequency channels, we can expect human listeners to be less sensitive to position errors than for low-frequency channels, because neurons of the auditory nerve do no longer show a phase-locked firing behavior above 4 kHz (see section 2.2.3). We therefore estimate only a minor increase in necessary bit rate when audio signals at higher sampling rates (e.g. 48 kHz) are coded. However, to affirm this is a matter of further research. In section 4.4 we address the pulse position encoding of high-frequency channels by means of vector quantization. In this section we consider to gain coding efficiency by means of a simple downsampling procedure.

For low-frequency channels, a (regular) downsampling can be performed to reduce the density of the sampling grid even before the peak-picking operation (e.g., using a decimated filterbank). This is especially useful when the coder operates

on a high sampling rate such as 48 kHz for general audio signals. Though the reconstruction from the pulse trains works well when the signal frequency is clearly below half the sampling rate, the reconstruction error can already be neglected for oversampling by a factor 2, as shown by the listening test in section 3.5.2.2. For two times oversampling, even the correction of peak amplitude errors according to section 2.3.1.2 can be neglected. When no oversampling is provided, this amplitude correction is needed.

To avoid the problem of peak amplitude errors, it is better to introduce the coarse sampling grid after the peak-picking procedure by means of re-quantization of the pulse positions. This re-quantization can be implemented as a simple rounding operation of the pulse locations. However, we have to keep in mind that the reconstruction sampling points (or the centers of the quantization cells) have to be a subset of the original sample positions, i.e., they have to be integers. For instance, we cannot choose the reconstruction points to be  $\{0.5, 2.5, 4.5, \dots\}$  for the case of a quantization cell size of 2. As a consequence, the introduced location errors are always integer multiples of the sampling interval. For the desired decimation factor  $L \in \mathbb{N}$  in the considered channel (which corresponds to the quantization step size), the re-quantized pulse location is

$$\hat{n}_m = \left\lfloor \frac{L-1}{2} \right\rfloor + L \cdot \text{round} \left( \frac{n_m - \frac{L-1}{2}}{L} \right) \quad (4.1)$$

where  $\text{round}(x)$  rounds  $x \in \mathbb{R}$  to its nearest integer. The pulse distances are then computed as  $\hat{d}_m = \hat{n}_m - \hat{n}_{m-1}$ . It is natural for a non-decimated analysis filterbank that channels with lower frequencies are more oversampled than high-frequency channels. We can therefore choose higher decimation factors  $L$  for the channels with lower frequencies.

We performed experiments with the 20-channel narrow-band speech coder that operates at a sampling rate of  $f_s = 16000$  Hz. The masking model was used with an impact factor of  $r^{(dB)} = -0.5$  dB to sparsify the pulse representation. The channel-dependent decimation factor  $L_k$  was chosen to ensure  $\frac{f_s}{L_k} \geq 5f_{c,k}$ . This resulted in decimation factors as shown in the first row of Tab. 4.1. The six upper most channels were left undecimated. After the sparsification, the pulse location re-quantization according to Equ. (4.1) was computed. It should be noted that a considerably better reconstruction quality can be obtained when the amplitude correction due to masking is performed after the re-quantization of the pulse locations. We used the amplitude correction scheme of section 3.4.2 with Equ. (3.12) and Equ. (3.13) and the corrected amplitudes were left unquantized. The location re-quantization did not cause noteworthy distortions on the reconstructed signal, as confirmed by an informal listening test.

The downsampling or re-quantization reduces the number of possible pulse

$k$	0	1	2	3	4	5	6	7	8	9	10	11	12	13
$L_k$	32	22	16	12	9	8	6	5	4	3	3	2	2	2
$H_{d,k}$	6.5	6.3	6.3	6.0	6.0	6.0	5.7	5.5	5.5	5.0	4.9	4.9	4.4	4.4
$H_{\hat{d},k}$	3.0	3.1	3.5	3.5	3.9	3.9	3.9	4.2	4.2	4.0	4.1	4.2	3.8	3.8
$f_{p,k}$	66	81	82	128	121	144	174	174	231	295	311	340	438	499

Table 4.1: Figures and statistics of the location re-quantization experiment.  $k$  is the channel index,  $L_k$  denotes the decimation factor of channel  $k$ ,  $H_{d,k}$  is the entropy of a distance code without re-quantization in bits,  $H_{\hat{d},k}$  is the reduced entropy after re-quantization in bits, and  $f_{p,k}$  denotes the pulse rate in pulses/s.

positions and possible pulse distances. Thus, the size of the distance ‘alphabet’ is decreased, which results also in a decrease in entropy. The second row in Tab. 4.1 presents the entropy of the original (i.e., not re-quantized) distance codes for the individual channels  $H_{d,k}$  in bits. In the third row the reduced entropy of the distance codes after the location re-quantization  $H_{\hat{d},k}$  is shown. The pulse rates do not change due to the re-quantization. However, since the (averaged) pulse rates  $f_{p,k}$  in pulses/s are low in low-frequency channels (as shown in the last row of Tab. 4.1), the achievable reduction in bit rate is low. The savings in bit rate to encode the distances of all 20 channels with ideal entropy coding are 3.71 kbit/s (which corresponds to 10.7% of the original bit rate of 34.54 kbit/s without the re-quantization).

While the discussed pulse location re-quantization works only for low-frequency channels, other encoding strategies are necessary to reduce the bit rate in high-frequency channels. A quantization of the pulse locations can also be accomplished by means of vector quantization. This has the advantage that the pulse positions and the pulse amplitudes can be regarded jointly. We consider vector quantization in more detail in section 4.4.

### 4.2.3 Discussion

The original (i.e., unreduced) auditory representation is highly redundant due to (i) the overlapping auditory filters used for the analysis and (ii) the high pulse rates obtained by peak picking, which depend on the filters’ center frequencies instead of their bandwidths. The obtained pulse distances  $d_m$  of a certain channel do not vary markedly along time, i.e, their standard deviation is relatively small compared to their mean value. However, we should note that the mean value does not carry information. When we subtract the mean value of the pulse distances and analyze their (normalized) autocorrelation sequence, as presented in Fig. 4.5 (dashed line) for channel 4 of the narrow-band coder, we can observe that consecutive distances

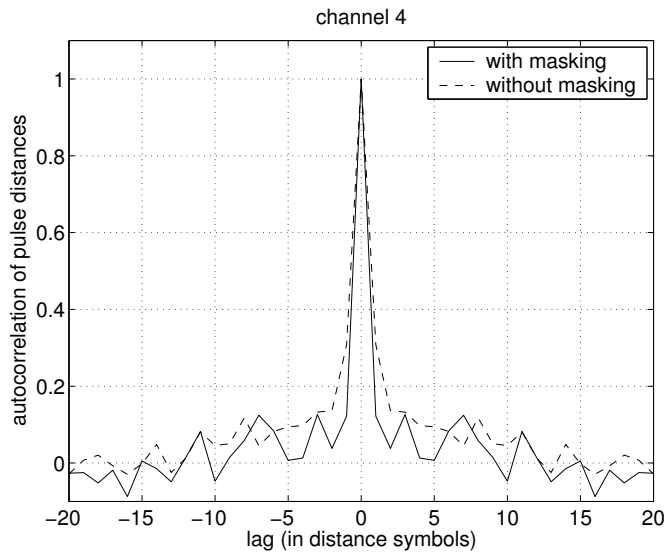


Figure 4.5: Autocorrelation sequence of the pulse distances of channel 4 of the 20-channel narrow-band coder with masking (impact factor  $r^{(dB)} = -0.5$  dB) and without masking.

are rather uncorrelated. Experiments with pulse trains from other channels show that the autocorrelation sequence looks similar for all auditory channels. The corresponding power spectra are relatively flat, which means a low prediction gain. Therefore, we cannot expect to gain coding efficiency considerably by means of linear prediction.

The analysis of the pulse distance dependencies between neighboring channels proves to be more difficult than considering only single channels. The problem is that distance symbols occur at different time instances and with different rates in different channels. This point makes it even difficult to define a multi-variate random process to analyze the channel dependencies.

The sparsification achieved by our masking model reduces the pulse rates. When we analyze the (normalized) autocorrelation sequence of the mean-subtracted pulse distances after the masking model (see Fig. 4.5, solid line), we can observe that the pulse distances are slightly less correlated as without masking, but the change is rather small. However, the variance of the pulse distances has been increased considerably due to the sparsification: from around 50 without masking to almost 20000 with masking. Similar to the trade-off between reduced pulse rate and increased distance entropy caused by masking, as discussed in section 4.2.1, we get a trade-off between reduced pulse rate, reduced prediction gain, and increased distance variance. The here shown analysis reveals that linear prediction does not

facilitate the encoding of the pulse positions. The discussed difficulties and the yet unclear strategies to analyse channel dependencies raise the necessity of further investigations.

### 4.3 Pulse Amplitudes

In [26] a block-scalar quantizer, which operates on 20 ms segments, is used to quantize the amplitudes of the pulse trains. The positions are left unquantized. The maximum amplitude of a block has to be transmitted as side information for each channel. Six bits per value have been found to suffice for this purpose [77]. The amplitude detail information within the block is quantized spending 1 bit per pulse. In fact, quantizing the pulse amplitudes with 1 bit enables us to refer to three amplitude values—high, middle, and zero—since zero means that there is no pulse at all (also no pulse position transmitted as side information). New experiments using the same quantizer configuration reveal clearly audible distortions.

To be able to optimize a block-scalar quantizer, we have to carefully investigate the statistics of the pulse amplitudes. For the following analysis, we use pulse trains obtained by the 20-channel coder for 8 kHz-sampled signals with an impact factor of  $r = 1.05$  (0.42 dB) for the masking criterion and with the improved amplitude correction scheme described in section 3.4.2. These pulse trains are ready to be synthesized by the synthesis filterbank, i.e., the amplitudes are not power-law compressed (cf. the block diagram in Fig. 3.9). The sentence ‘These days, a chicken leg is a rare dish’ spoken by a female was chosen as input signal. The block duration was chosen as 8 ms. Consider a single block in a single channel and let  $\mathbb{J}$  denote the set of all pulse positions in the considered segment. The number of pulses in the block is  $|\mathbb{J}|$  and the pulse amplitudes are  $x[n] > 0$ ,  $n \in \mathbb{J}$ . For blocks with more than one pulse  $|\mathbb{J}| > 1$ , we can define the dynamic range of the pulse amplitudes as  $20 \log_{10}(\max_{n \in \mathbb{J}}(x[n]) / \min_{n \in \mathbb{J}}(x[n]))$  in dB. Fig. 4.6 shows histograms of the number of pulses observed in a block (upper diagrams) and histograms of the dynamic range found in blocks with more than one pulse. The left column analyzes the pulse train of channel 4 (around 320 Hz) and the right column channel 14 (around 1.8 kHz). The statistics have been gathered using a sliding block that advances in steps of a single sample instead of the entire block length. This is equivalent to performing the experiment repeatedly with shifted versions of the same input signal together with a block step of the entire block duration (i.e., with non-overlapping segments). In this way we ensure that all possible segments can be covered.

For our auditory representation, which consists of unequally spaced pulses, a block quantizer should distinguish between three different situations. When a block

without a pulse in a channel arises, certainly no amplitude information needs to be transmitted for that channel. When a block contains a single pulse in a channel, then only the single pulse amplitude has to be encoded as the block maximum for that channel. The third case happens when two or more pulses are located within a block. Then, in addition to the maximum pulse amplitude, the individual pulse amplitudes should be encoded. The information needed to distinguish between the different cases can fully be recovered from the pulse-position side information.

The encoding of amplitude information in low-frequency channels does not raise difficulties. As revealed by Fig. 4.6, in channel 4 with a relatively low center frequency, the two cases with either a single pulse per block or no pulse at all are predominant (together with a probability of 0.69) for the chosen block duration of 8 ms. Therefore, the transmission of within-block amplitude information is necessary only for 31% of all frames. Fig. 4.6 also shows the dynamic range of the pulse amplitudes within the duration of a block for the cases with at least two pulses per frame in the lower diagram. For this histogram, the amplitude variation has been converted to dB and rounded to integers. For channel 4, the probability of a within-block dynamic range less than 0.5 dB is 0.72 and for a range less than 1.5 dB even 0.91. Therefore, a coarse quantization with only 1 bit does not cause noteworthy distortions in this channel. In lower channels, the probability that more than one pulse occurs per block is even less than 31%. The inverse of the chosen block length of 8 ms, i.e., the block rate, is 125 Hz. For channels with center frequencies lower than the block rate, naturally the situation with empty blocks predominates and the occurrence of multiple pulses is no longer possible for typical input signals.

In high-frequency channels, the pulse amplitude encoding proves to be more difficult. As visible in Fig. 4.6, for channel 14 the case with two and more pulses per block is predominant. Blocks with even 8 pulses can be observed. Consequently, the dynamic range is wider than for low-frequency channels and spending only 1 bit per pulse is definitely no longer sufficient. At least 3 bits should be used for the amplitude detail information for channel 14 to avoid noteworthy distortions. For the channels with even higher center frequencies, the situation gets worse. The number of pulses in a block and also the dynamic range increases and, as a consequence, we should spend more bits per pulse. This explains the audible distortions caused by the 1-bit quantizer used in [26]. Furthermore, the block duration used in [26] is 20 ms, whereas the statistics shown here are gathered for a block duration of only 8 ms. Therefore, it is better to use shorter blocks for the amplitude quantization for the high-frequency channels. This reduces the probability of relatively high dynamic ranges within the blocks and enables the encoding of the amplitude detail information with less bits. On the other hand, shorter blocks naturally increase the data rate for the side information.

Shorter blocks for higher channel frequencies are used in [70]. However, in that



work 0 bits are spent on the amplitude detail information, i.e., all amplitudes are set to the same level. The side information, which in [70] is the mean of the pulse amplitudes, is calculated on blocks with a duration of 32 ms for frequencies below 1 kHz, 16 ms for frequencies up to 2.5 kHz, 8 ms for frequencies up to 4.8 kHz, and 4 ms up to 7 kHz. The choice of these block lengths is inspired by the work of Krasner in [93]. Keeping the dynamic-range histogram of channel 14 ( $f_c = 1778$  Hz) shown in Fig. 4.6 in mind, casts doubts on the ‘high quality’ resynthesis capability of the quantization scheme as it is claimed in [70]. Interestingly, in a later work [71] these authors use 1 bit for the amplitude detail information. In the work of Krasner [93], between 3 and 4 bits are used for the amplitude details per sample<sup>1</sup>.

Fig. 4.7 shows statistics of high-frequency channels gathered for a block duration of only 2 ms. Now the histograms look similar to those of low-frequency channels for the longer block duration of 8 ms shown in Fig. 4.6. Consequently, a coarse quantization of the amplitude detail information is now possible without noteworthy distortions. The consideration of such short block lengths is important for the usage of vector quantizers. Vector quantizers can also be used to encode the amplitude detail information. The next section deals with vector quantization in more detail.

Finally, we should note that the usage of fixed block lengths brings a crucial drawback. It nullifies the whole advantage of having a non-decimated signal representation that carries the temporal fine structure. Therefore, it is advantageous to use signal-dependent block lengths. In section 4.4.2, we will consider an encoding with variable block lengths.

## 4.4 Vector Quantization

The nature of the multi-channel pulse representation suggests to use individual vector quantizers for each channel. One reason is that the sparsification process makes the pulse trains of neighboring auditory channels relatively independent due to simultaneous masking. Another reason is that the codebook can be optimized for certain pulse distances that are typical for the individual channel.

Vector quantizers are typically embedded in analysis-by-synthesis schemes [1] to find the optimum codebook entry. When a classical analysis-by-synthesis scheme is preferred for our coder, we should include the power-law expansion and the channel’s synthesis filter with the time-reversed gammatone impulse response (see the decoder structure in the last chapter in Fig. 3.10) to compute the channel’s reconstruction error. A more accurate analysis-by-synthesis scheme should be in

---

<sup>1</sup>In [93] the subbands carry regularly downsampled signals obtained by a quadrature mirror filterbank.

accordance with the transmultiplexer structure (see section 3.2) and would incorporate even another analysis stage before the reconstruction error is computed. However, using such analysis-by-synthesis schemes, we lose the whole advantage of coding in the perceptual domain where a simple single-letter distortion criterion is assumed to be directly applicable and yet perceptually meaningful.

We can search for the optimum codebook entry in the perceptual domain directly by evaluating a distance measure (or a similarity measure) between all codebook entries and the actual pulse train segment. The entry that minimizes the distance measure (or maximizes the similarity measure) has to be selected and its index has to be encoded. In [70] a vector quantizer is used for the pulse trains in the auditory channels with center frequencies above 1 kHz to encode the pulse positions exclusively<sup>2</sup>. The codebook size varies from  $2^2 = 4$  entries for the highest channel (7 kHz) to  $2^6 = 64$  entries for 1.17 kHz. The entries have a length of 2 ms.

#### 4.4.1 Distance/Similarity Measure for Sparse Vectors

The lossy encoding of sparse sources (i.e., spike signals) involves the quantization of position-amplitude pairs. The proper bit allocation between position and amplitude information has turned out to be difficult. A simple way is to losslessly encode the positions and quantize the amplitudes only. In order to achieve lower bit rates, the position information also needs to be quantized. Recent work on the rate-distortion behavior of sparse sources can be found in [92, 94, 95]. However, in these contributions the Hamming distance measure is used that does not account for the actual extent of the deviations from the ideal pulse positions. This section proposes a similarity measure that is better suitable for sparse vectors.

##### 4.4.1.1 Position Encoding

Let us assume to have pulse train segments where the amplitude information has been discarded, i.e., we can treat the segments as binary vectors. We will consider segments with different pulse amplitudes later. A commonly used distance measure for binary vectors is the Hamming distance [94, 95]. Let  $\mathbf{x}$  be a vector with the elements  $x_k \in \{0, 1\}$  and  $\tilde{\mathbf{x}}$  an approximation of  $\mathbf{x}$ . The Hamming distance is defined as

$$d_H(\mathbf{x}, \tilde{\mathbf{x}}) = \sum_k [1 - \delta[x_k - \tilde{x}_k]]. \quad (4.2)$$

It can be seen that  $d_H(\mathbf{x}, \tilde{\mathbf{x}})$  counts the number of elements that are different between the two vectors. In [70] a Hamming distance is used for the amplitude-normalized pulse train segments and they suggest to use an XOR operation to simplify the distance computation.

---

<sup>2</sup>In [70] the amplitude detail information is neglected.

The Hamming distance can be related to an inner vector product. While the Hamming distance counts the wrong vector elements, the inner vector product counts the correct non-zero elements for binary vectors. Thus, an inner product yields a similarity measure. An inner vector product is related to the cosine of the angle between two vectors. It yields 0 when the two vectors are orthogonal and reaches a maximum when they are identical.

The simple Hamming distance measure and also the inner vector product similarity measure need to be modified because they involve ambiguities (i.e., different approximations can have the same distance/similarity). This is because sparse vectors are likely to be orthogonal to each other. When the codebook contains sparse pulse train segments, we have to consider that a minimum shift of a pulse by one sample can change the measure either considerably or not at all, since only congruent pulses contribute. The deviation of a pulse from its desired position is not reflected in these measures in a continuous way.

To obtain an optimality measure that considers the amount of deviation from the desired pulse position, we should use widened pulses (e.g., triangular or rectangular shape) either for the approximation  $\tilde{\mathbf{x}}$  (e.g., an entry of the codebook) or for the desired segment  $\mathbf{x}$  or for both before the inner product is computed. Using widened pulses can be interpreted as convolving the pulse trains with a pulse shaping filter, e.g.,  $\mathbf{H}_p\mathbf{x}$  and/or  $\mathbf{H}_p\tilde{\mathbf{x}}$  with  $\mathbf{H}_p$  the convolution matrix of the pulse shaping filter. In this way we are able to distinguish between almost congruent pulses and more distant ones. The width of a shaped pulse has to be chosen in accordance with the auditory channel's center frequency, i.e., the width should be limited by the typical pulse distance of the channel (see previous chapters). The optimality measure should also reflect whether the overall number of pulses is right or not. This can be achieved by adding negative samples at positions where no pulse is desired either for the desired segment  $\mathbf{x}$  or for the approximation  $\tilde{\mathbf{x}}$ . Thus, a pulse far off a desired position or an additional pulse results in a penalty.

Fig. 4.8 illustrates an example of building an enhanced similarity measure based on the above discussion. The upper row shows the pulse train segment to be encoded (i.e., the desired segment  $\mathbf{x}$ ). Before the inner product is computed, the desired pulse train segment is convolved with the pulse shaping filter, the constant 0.5 is subtracted, and the segment is multiplied by 2 (as shown in the second row). Also the approximation  $\tilde{\mathbf{x}}$  is convolved with the same pulse shaping filter before the inner product. The three lower rows of Fig. 4.8 represent three processed approximations. We use a rectangular pulse shape with a width of  $W_P = 5$  samples for this example. The resulting similarity measure can be written as

$$s(\mathbf{x}, \tilde{\mathbf{x}}) = \langle 2 \cdot (\mathbf{H}_p\mathbf{x} - 0.5 \cdot \mathbf{1}), \mathbf{H}_p\tilde{\mathbf{x}} \rangle \quad (4.3)$$

where  $\mathbf{1}$  is a vector with 1 for all elements. It should be noted that for a vector quantizer with a fixed codebook, the convolution of the approximation candidates

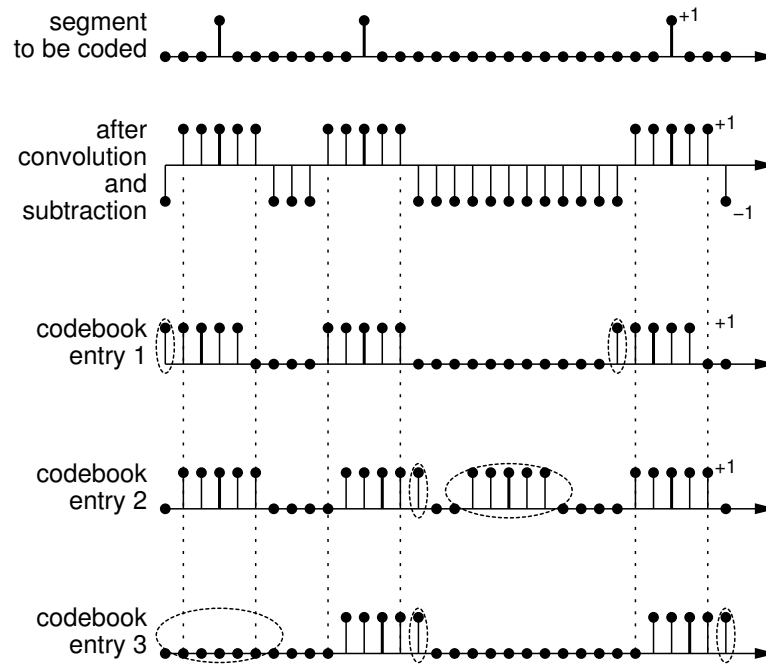


Figure 4.8: Example of building a similarity measure that reflects proper pulse positions. The thick lines indicate the actual pulse positions and the ellipses mark penalty terms. The rectangularly shaped pulses penalize small deviations linearly up to a shift by 4 samples. Larger deviations (or additional pulses) yield a constant penalty.

penalty reason	penalty	in % for $W_P = 5$
shift by 1	2	40%
shift by 2	4	80%
shift by 3	6	120%
shift by 4	8	160%
missing pulse	$W_P$	100%
additional pulse	$W_P$	100%

Table 4.2: The effective penalties in the inner product similarity measure with rectangular pulses of width  $W_P$  samples. The relative penalty in % with respect to the penalty for a missing or an additional pulse refers to a pulse width of  $W_P = 5$  samples (as used in Fig. 4.8).

$\mathbf{H}_p \tilde{\mathbf{x}}$  can be performed already offline, and the widened pulses can be stored in the codebook. This saves computational load to search for the optimal codebook entry in the encoder. The three lower rows of Fig. 4.8 can therefore be considered as codebook entries of the encoder. The shown segments have a length of 32 samples (2 ms at a sampling rate of 16 kHz). The figure considers the quantization of the pulse positions only by treating all pulse amplitudes as binary. For 2 ms segments, the neglect of amplitude detail information seems to be reasonable. The thick lines indicate the actual pulse positions, which exclusively build the entries in the codebook used in the decoder. The rectangular pulses result in a penalty that increases linearly with the deviation from the ideal position. The measure is also able to penalize additional pulses at wrong locations and missing pulses. For the similarity measure shown in this example, an optimum match between a desired segment with  $N_P$  pulses and a codebook entry yields an inner vector product of  $N_P W_P$ . This maximum value is decreased in terms of penalties according to table 4.2. The penalty for a missing pulse can be explained by the fact that a missing pulse does not contribute positively either. For the three different codebook entries of Fig. 4.8, we get an optimality measure of 11 for the first, 8 for the second, and 6 for the third. The achievable maximum is 15 since the desired segment contains 3 pulses. The first codebook entry is clearly the best one.

#### 4.4.1.2 Joint Position-Amplitude Encoding

So far only the pulse positions have been considered for the distance (or actually the similarity) measure. All pulse amplitudes are set to 1, which means that the amplitude detail information is discarded. This has also been proposed in [70]

for a codebook entry duration of 2 ms<sup>3</sup>. For such short segments, the amplitude detail information can be quantized coarsely or even ignored as is shown in the histograms in Fig. 4.7. If the reconstruction quality requires the encoding of the detail information, the codebook entries can contain pulses with different amplitudes. The inner vector product criterion described above is still applicable, but the size of the codebook is increased considerably in this case to account for all likely combinations.

Independent of the fact whether the codebook contains pulses with different amplitudes or not, the desired segment should keep the amplitude information to weight the contributions (and/or penalties) of the individual pulses according to their amplitudes. Thus, the optimality measure reflects that position errors are more costly when the pulse amplitudes are large. This requires a slightly different preprocessing of the pulse train of the desired segment before the inner product is computed. In the example with binary vectors shown in Fig. 4.8 the preprocessing consists of the pulse shaping, the subtraction by 0.5, and a normalization afterwards. For vectors that contain pulses with different amplitudes, one possibility is to subtract  $0.5 \min_{k \in \mathcal{P}} x_k$  with  $\mathcal{P}$  the set of pulse positions, i.e., to subtract half the minimum pulse amplitude of the segment. In this way a codebook entry with a pulse at the position where the desired segment has a large pulse contributes more than a match at a position of a small pulse. For the operation of a vector quantizer, the normalization can be neglected since we are simply searching for the entry that maximizes the similarity measure. The reason for the normalization in the binary case above is to simplify the specification of penalties with respect to an optimally matching vector.

#### 4.4.2 Variable Frame Length Vector Quantization

The size of the codebook, i.e., the number of codebook entries, is an important issue since it determines the number of bits that needs to be transmitted. A method to obtain small codebooks is called pruning. Pruning eliminates entries that are not likely to occur. For the pulse distances of an auditory channel, some values are more likely than others (see Fig. 4.1). This can be exploited for the design of a codebook. However, for frames with a fixed length, the phase of a pulse train occurs randomly. In order to account for the most likely pulse constellations, the different phases also have to be considered and consequently, the codebook size explodes. In this section a coding strategy that avoids codebook entries with different phases is proposed.

In order to use a codebook that does not contain pulse trains with differ-

---

<sup>3</sup>Note, that in [70] the amplitude side information is even transmitted only for superframes with durations between 4 ms and 32 ms.

ent phases, we need to synchronize the frames to certain pulse positions. This naturally requires the transmission of side information. A straightforward way to achieve a synchronization is to combine distance (or run-length) encoding and vector quantization and we get *run length vector quantization* (RLVQ). This means an interleaved transmission of codebook indices and pulse distances. The codebook entries can still have a fixed length. The side information specifies the variable number of additional zero samples before the next frame starts. This results in a variable frame length encoding where we can ensure that the desired segment has always a pulse at the first position.

The codebook design becomes significantly easier when all segments start with a pulse. The probability of the placement of additional pulses at a particular position within a frame depends on the probability distribution of the pulse distances (see Fig. 4.1). More precisely, when we assume consecutive pulse distances to be independent<sup>4</sup>, the probability distribution of the location of the second pulse equals the pulse distance distribution (truncated after the considered segment length). The probability of the location of the third pulse is equal to the convolution of the pulse distance distribution with itself. The probability of the location of the fourth pulse is equal to the three-fold convolution of the pulse distance distribution and so forth.

We performed an experiment with the 20-channel narrow-band coder operating at  $f_s = 8000$  Hz and an English sentence spoken by a female as input. The masking model described in the previous chapter with an impact factor of  $r = 1.05$  ( $r^{(dB)} = 0.42$  dB) and the pulse amplitude correction stage described in section 3.4.2 were applied. Histograms for the probabilities of the pulse locations within first-pulse synchronous 2 ms segments were gathered and the obtained numbers were normalized by the number of observed frames. The comparison of these histograms with the frame location probabilities computed from the distance distributions as described above shows that the latter predict the histograms well for sparsified pulse trains. Therefore, the so predicted location histograms can be used to simplify the codebook design. The upper plot of Fig. 4.9 shows the location distribution for synchronous 2 ms segments of channel 19 (3.6 kHz). For the codebook design, some positions can clearly be ignored: 1, 2, 3, 4, 6, and 8. More important are the positions 5, 7, 12, and of course 0. The contributions of the second, third, and fourth pulse are shaded differently. Note that the sum over all positions (including position 0) yields generally more than 1 (around 2.0 for the shown histogram of Fig. 4.9). Naturally, there is always a first pulse (as a consequence of the synchronization). This also means that the probability of finding one or more pulses per segment is 100%. Similarly, the sum of the contributions

---

<sup>4</sup>Consecutive pulse distances become less dependent with increasing impact factor of the masking model.

of placing a second pulse, which gives 71% in the shown example, tells us that the probability of finding two or more pulses is 71%. As a consequence, the probability of finding only one pulse is  $100\% - 71\% = 29\%$ . The probability to find a third pulse (= three or more pulses) is 28%, which gives a probability to find exactly two pulses per frame of  $71\% - 28\% = 43\%$ . The probability to find a fourth pulse is only 1%. To find more than four pulses is not possible. Consequently, to have three pulses in a frame occurs with a probability of  $28\% - 1\% = 27\%$ .

Experiments with the complete pulse representation (i.e., without the masking stage) reveal that the synchronization does not yield sparse placement probability distributions. Fig. 4.10 shows the gathered location histogram for the unreduced pulse train of channel 19. Now, up to eight pulses can be found in a frame. As visible in the figure, only the position 1 is not populated for the complete pulse representation. As a consequence, we cannot design small codebooks. Therefore, the sparsification process is considerably assisting for the RLVQ encoding scheme.

In the lower row of Fig. 4.9, a scatter plot of the pulse amplitudes in synchronous 2 ms segments of channel 19 is shown. The amplitudes are normalized by the RMS value of the pulse amplitudes (i.e., non-zero amplitudes) of a frame and converted to dB. This is to visualize the relative errors when all amplitudes within a segment are set equal to the RMS value, which is transmitted as side information in this case, in the decoder. Even for short segments of 2 ms, these relative amplitude errors can reach about 12 dB. However, the perceptual effect of these errors in high-frequency channels is small (see later). In [70] the pulse amplitudes are replaced by the mean of the non-zero amplitudes. However, using the RMS value instead of the mean, we are able to preserve the energy of the subband signals, which is preferable.

To gain coding efficiency over RLE (which is lossless), the RLVQ codebook size has to be small so that the transmission of the indices requires markedly less bits per pulse than the entropy of the pulse distances (see Fig. 4.2). Let the average number of pulses per synchronized frame in a channel be denoted by  $\bar{N}$ . Since the synchronization is equivalent to the encoding of the first pulse of the frame, the codebook index effectively encodes only  $\bar{N} - 1$  pulses on average. When we denote the entropy of the pulse distances of a channel by  $H_d$  (from Fig. 4.2 we get for instance an entropy of 4.7 bits for channel 19 and 3.8 bits for channel 18) we can obtain an upper bound for the length of the codebook index as  $H_d \cdot (\bar{N} - 1)$  bits. The experiments with the 20-channel narrowband coder with an impact factor of  $r = 1.05$  and a vector length of 2 ms show an average number of 2.0 pulses per synchronous frame for channel 19 and 2.3 pulses for channel 18. These figures suggest to spend less than 4.7 bits for the codebook index for channel 19 and less than 4.9 bits for channel 18 in order to be more efficient than RLE.

Experiments with synchronous 2-bit codebooks with a segment length of 2 ms for the three uppermost channels (2.72 kHz–3.6 kHz) of the 20-channel narrow-

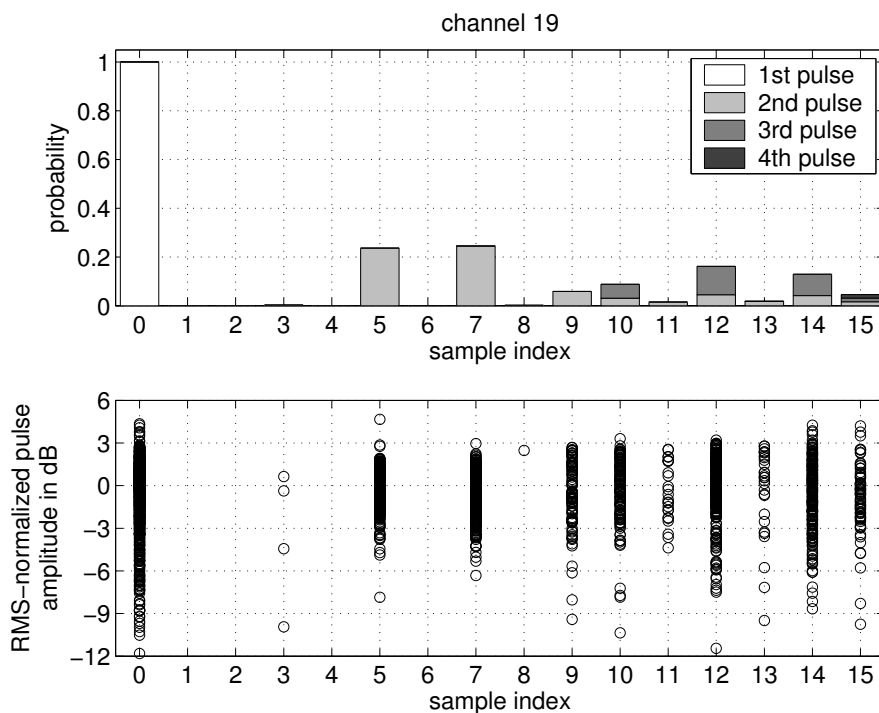


Figure 4.9: Upper plot: Pulse location probabilities in first-sample synchronous segments with a maximum support of 2 ms for channel 19 (3.6 kHz) of the 20-channel 8 kHz coder. Lower plot: Scatter plot of synchronous segments where the pulse amplitudes are normalized by the RMS value of the non-zero amplitudes of the segment.

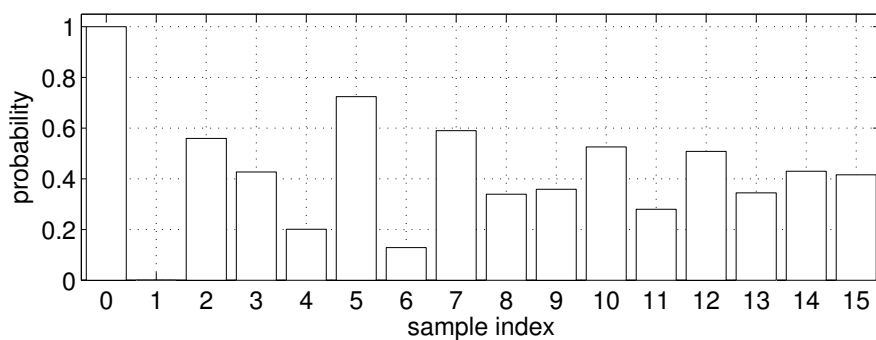


Figure 4.10: Pulse location probabilities in first-sample synchronous segments with a maximum support of 2 ms for channel 19 (3.6 kHz) of the 20-channel 8 kHz coder without masking.

band coder with masking model (impact factor  $r^{(dB)} = -0.5$  dB) show promising results. The similarity measure described in the previous section with a rectangular pulse shape of width  $W_P = 3$  is applied for the vector quantizer. For the three channels, the codebooks are simply chosen to contain only 2-pulse entries. First, pulse placement histograms such as in Fig. 4.9 are gathered using training speech material. The four positions with the highest probabilities (except position 0) are selected for the second pulse. Table 4.3 summarizes some figures of the experiment. The computation of the entropy of the pulse distance codes  $H_d$ , which is shown in the second column, is based on histograms as shown in Fig. 4.1. The computation of  $H_d$  is needed to determine the bit rate of the side information as well as to be able to compare the overall bit rate of the hybrid encoding with RLE. The third column shows the average number of pulses in synchronous frames  $\bar{N}$ . The pulse positions of the chosen 2-pulse entries of the three individual 2-bit codebooks are given in the fourth column. Since the four different codebook entries do not match actual segments with the same probability, an additional entropy coding can increase the coding efficiency. The fifth column shows the entropy of the codebook indices  $H_I$ . For the overall bit rate, we have to combine the rate of the side information and the rate required for the codebook indices:  $H_d + H_I$  bits/frame. To be able to compare with  $H_d$ , we compute the overall rate on the basis of a pulse:  $H_{RLVQ} = (H_d + H_I) / \bar{N}$  bits/pulse. The comparison of RLVQ and RLE shows savings in bit rate of more than 40% when we assume that ideal entropy coding can be achieved. However, the proposed synchronized vector quantization is a lossy encoding whereas RLE is lossless. Here, the codebook index can be interpreted as a 2-bit quantized pulse distance. The position errors occur at irregular intervals and are interleaved with correctly encoded pulse positions. For these experiments, the pulse amplitudes within a segment are replaced by the RMS value of the non-zero amplitudes of each segment. The quality of the reconstructed speech material was evaluated in an informal listening test where no further degradation due to the quantization was observed.

## 4.5 Conclusions

The encoding of sparse signals such as the neural firings in an auditory representation is a challenging problem. In this chapter the statistics of the pulsed signal representations have been analyzed and different approaches towards efficient encoding have been discussed. The entropy of the pulse location information, which is needed as side information, has been analyzed. A lossless encoding of the locations can be achieved by distance encoding. Downsampling or location re-quantization and vector quantization have been discussed as lossy encoding schemes. However, to evaluate the perceptual effect of pulse location errors, formal listening tests

channel index	$H_d$ $\frac{\text{bits}}{\text{pulse}}$	$\bar{N}$ $\frac{\text{pulses}}{\text{frame}}$	codebook entries 2 <sup>nd</sup> pulse position	$H_I$ $\frac{\text{bits}}{\text{frame}}$	$H_{RLVQ}$ $\frac{\text{bits}}{\text{pulse}}$	saving %
19	3.83	2.39	5, 7, 10, 12	1.71	2.3	-40
18	3.11	2.70	5, 8, 10, 13	1.81	1.8	-42
17	3.82	2.31	3, 6, 9, 14	1.23	2.2	-42

Table 4.3: Statistics of the 2 ms RLVQ for the three uppermost channels of the narrowband coder.  $H_{RLE}$  is the entropy of pulse distances.  $\bar{N}$  denotes the average number of pulses per synchronized frame.  $H_I$  is the entropy of the codebook indices.  $H_{RLVQ}$  is the ideal overall mean rate per pulse. The saving in bit rate is with respect to RLE.

are necessary in future work. For lossy encoding of both amplitude and position, classical rate-distortion theory has to be extended to account for the dependencies between position and amplitude errors. In this chapter a similarity measure has been proposed that is suitable for vector quantizers that deal with sparse signals. Furthermore, a lossy encoding scheme that is a hybrid of distance encoding and vector quantization has been investigated. This scheme has been shown to gain considerable savings in bit rate in high-frequency channels without causing audible distortions. The simulations also show that the sparsification of the auditory representation using the masking model described in chapter 3 facilitates the encoding, both lossless using distance encoding and lossy using the proposed hybrid encoding. However, schemes for the efficient encoding of the amplitude information such as linear prediction have to be investigated in further research work. In order to find a proper bit allocation, further analysis as well as further listening tests are necessary.

# Chapter 5

## Alternative Filterbank Implementation Methods

The coding method this thesis deals with is based on an invertible auditory model that provides non-decimated subband signals (see chapter 2). The auditory filterbank is the computationally most complex component in this model. As already discussed in section 2.2.1.2, FIR implementations of non-decimated filterbanks are computationally expensive and memory consuming. To achieve proper frequency responses, the lengths of the finite impulse responses have to be sufficiently long. Generally speaking, the length of the gammatone impulse response of Equ. (2.6) is determined by the gamma envelope, which becomes wider for auditory filters with lower center frequencies (refer to section 2.2 for more details). The overall memory requirement is made up of the storage for the impulse responses and the memory for the delay lines. For the case when the synthesis filterbank uses time-reversed versions of the analysis filters (as done in section 2.3), the storage requirement can be halved. While an analysis filterbank needs only one delay line (because there is only one input signal), this is not true for the synthesis filterbank, which has as many different input signals as it has channels. A synthesis filterbank with  $K$  channels needs therefore  $K$  separate delay lines, which result in the same memory requirement as for the coefficient storage. The computational complexity is reflected by the overall number of coefficients of the analysis and the synthesis filterbank, since each coefficient means one multiplication and in general one addition.

This chapter deals with more efficient implementation methods for non-decimated auditory filterbanks for analysis and synthesis. We briefly discuss IIR filterbanks in section 5.1. Section 5.2 considers a frequency-warped transform filterbank, which constitutes a computationally efficient and memory saving implementation method. We also propose a design recipe to ideally approximate a gammatone filterbank using a frequency-warped transform filterbank in section 5.2.2.

## 5.1 IIR Filterbank

IIR filters are in general computationally less expensive than FIR filters on account of considerably less filter coefficients. In related literature, several physiologically motivated basilar membrane filterbanks, which result in IIR implementations, have been suggested. Examples are Lyon's cascade/parallel filterbank [46, 96], which consists of notch and resonator filters, or the more recent cascade filterbank by Baumgarte [97].

IIR implementations for gammatone auditory filters have been suggested (e.g., [98]). These are based on usual transforms from continuous-time transfer functions to discrete-time transfer functions (e.g., impulse invariance transformation [99]), which result in filters with an order of 8. In terms of approximation quality, we can regard these filters as ideal. The computational cost is 17 multiplications and 16 additions. In [98], a computationally even less expensive all-pole approximation has been proposed, which requires only 9 multiplications and 8 additions.

### Filterbank Inversion

A causal, straight-forward inversion based on time reversion of the impulse responses according to Equ. (2.23) is not possible for filters with infinite impulse responses. Instead of time-reversing the analysis filter's impulse response to obtain a synthesis filter, the subband signal can be time-reversed before filtering in the synthesis filter that is equal to the channel's analysis filter. The output of the synthesis filter has to be time-reversed again to achieve a zero-phase (i.e., noncausal) filter for the overall analysis-synthesis system within one filterbank channel. For real-time applications, a block-based computation is possible based on this method if the filters' initial states are set properly at the block boundaries (see [100] for more details).

Another possibility is to use FIR synthesis filters that approximate the needed characteristics [101]. In this case we have the high computational load and the high memory consumption anew.

Although for non-decimated filterbanks the direct channel-by-channel inversion of minimum-phase analysis filters seems possible with stable and causal synthesis filters, this is not advisable since the frequency response of the inverse is complementary, i.e., the inverse of a band-pass filter gives a band-stop. Thus, even without quantization or processing of the channel signals, numerical problems occur. Moreover, for coding applications, it is essential to have band-pass filters also for the synthesis filters to keep the quantization noise within a local frequency band. Particularly for the perceptual-domain coder described in chapter 2, it is vital to have band-pass filters to eliminate the aliasing caused by the pulse train representation. In this work we do not provide more detail on further inversion

possibilities for IIR filter banks. Instead, we refer to [101]. We can conclude that IIR implementations for auditory filterbanks are computationally efficient, but their inversion leads to difficulties and is therefore not straightforward.

## 5.2 Frequency-Warped Transform Filterbank

A computationally efficient approximation of an auditory filterbank is to take a frequency-warped transform filterbank. We first explain the general principle of a frequency-warped transform filterbank and then provide a design strategy to approximate a gammatone filterbank. Finally, we describe a simple inversion method for frequency-warped transform filterbanks.

### 5.2.1 Principle

In the early 1970's Oppenheim and co-workers [102, 103] introduced the technique of computing non-uniform resolution Fourier transforms. Their method first transforms the input sequence into a frequency-warped version by time-reversing and passing it through a cascade of all-pass filters. The all-pass cascade works as a dispersive delay line. Upon completion of the all-pass operations, an FFT of the samples along this all-pass cascade is performed. This is a computationally efficient method for a constant relative-bandwidth spectral analysis for finite-length signals.

In the late 1970's Vary [104] suggested a frequency-warped transform filterbank obtained by simply replacing the unit delay elements in the signal flow graph representation of a sliding window with general all-pass filters. This is illustrated in Fig. 5.1. In this chapter we consider a non-decimated filterbank where the sliding window advances by one sample at a time. The transform  $\mathbf{T}$  has to be calculated for every sample and non-decimated subband signals are obtained at the outputs of the transform. Note that in this flow graph the transform gets the most recent input sample at the upper most input. When we compare this flow graph with the buffer of a sliding window, where the most recent input sample is at the end of the buffer, we recognize that our transform needs to be used upside down. We therefore start the numbering of the window coefficients  $w_0, \dots, w_{N_w-1}$  at the bottom.

The general idea of a transform filterbank (left side of Fig. 5.1) is to use a prototype low-pass filter represented by a smooth window sequence. The basis sequences of the transform (e.g., a DFT or a DCT) are used to modulate the prototype low-pass filter to the different frequencies of the basis sequences and thus a bank of band-pass filters is built. All band-pass filters have the same bandwidth, which is twice the bandwidth of the prototype low-pass filter. Therefore, and

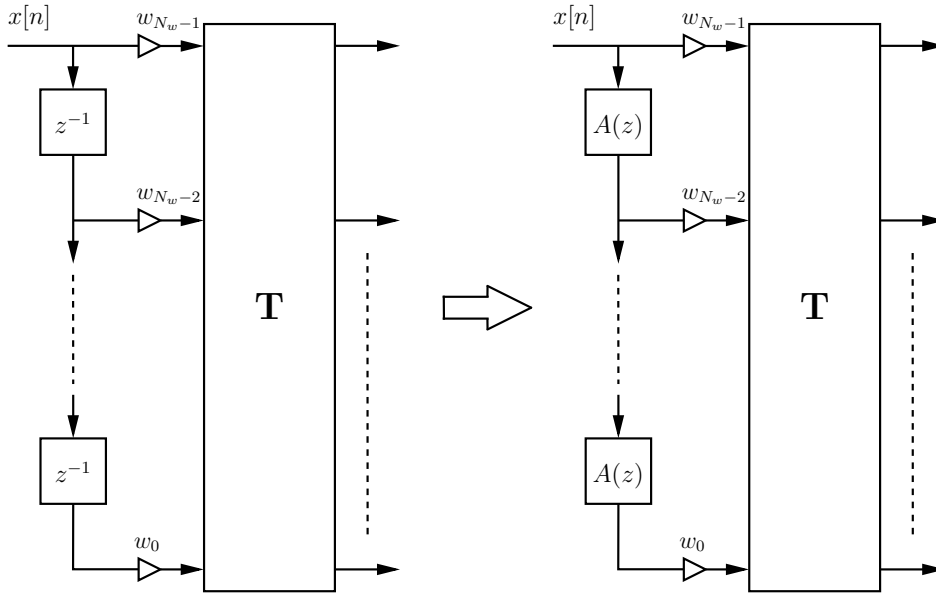


Figure 5.1: Modification of a non-decimated transform filterbank to obtain a frequency-warped version. The unit delay elements  $z^{-1}$  are replaced by all-pass filters  $A(z)$ .

since the frequencies of the basis sequences are usually equally spaced, we obtain a uniform filterbank.

It should be noted that the impulse response of the prototype low-pass filter has to be provided in a time-reversed version. This can easily be recognized when the DC output of a DFT is considered, which is simply the sum of all transform inputs:  $\sum_{i=0}^{N_w-1} w_{N_w-1-i} x[n-i]$ . To make this sum a convolution sum with the (finite) prototype impulse response  $\tilde{h}[n]$ , the window coefficients need to be chosen as  $w_i = \tilde{h}[N_w - 1 - i]$  for  $i = 0, \dots, N_w - 1$ . The window length  $N_w$  does not necessarily have to be equal to the number of channels  $K$  (see [105] or [106] for more details). Thus, a longer FIR prototype filter can be designed to better approximate the desired frequency responses (e.g., gammatone or roex auditory filter).

When the unit delays in the signal flow graph are replaced with general all-pass filters, which have nonlinear phase functions, the characteristics of the uniform transform filterbank are modified (right-hand side of Fig. 5.1). Let the transfer function of a first-order all-pass be denoted by

$$A(z) = \frac{z^{-1} - \lambda}{1 - \lambda z^{-1}} \quad (5.1)$$

with the single so-called frequency warping parameter  $\lambda$ . The replacement for

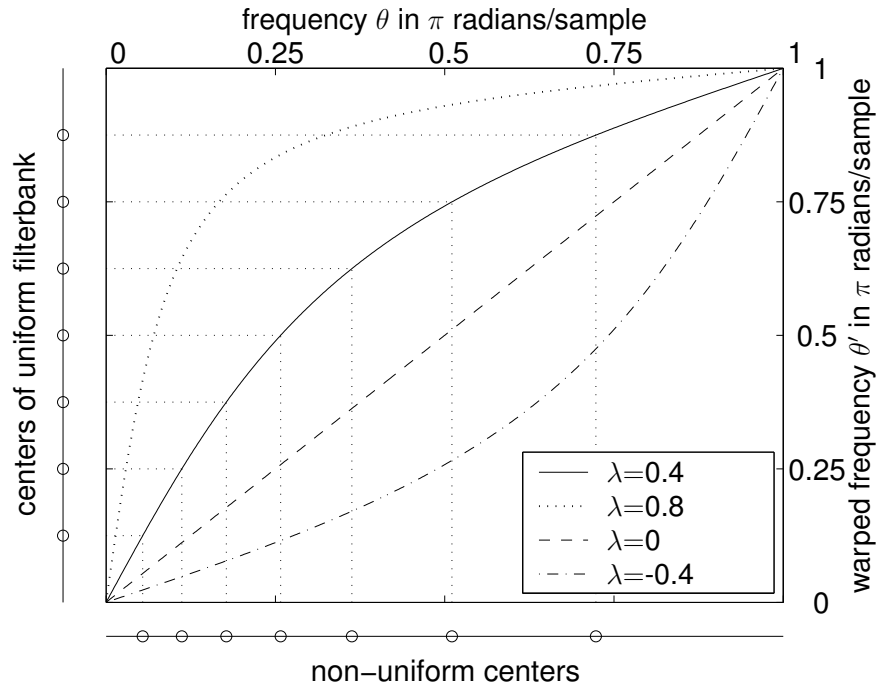


Figure 5.2: Frequency warping according to the phase function of the first-order all-pass for four different values of the warping parameter  $\lambda$ . The transform of the center frequencies of a uniform filterbank to a non-uniform filterbank is illustrated with  $\lambda = 0.4$ .

the unit delays corresponds to a substitution of  $A(z)$  for  $z^{-1}$  and thus, a bilinear transform of the complex  $z$ -plane is applied. This transform results in warping the frequency axis corresponding to the phase function of the all-pass filter

$$\theta'(\theta) = \arctan \left( \frac{(1 - \lambda^2) \sin(\theta)}{(1 + \lambda^2) \cos(\theta) - 2\lambda} \right), \quad (5.2)$$

where  $\theta$  and  $\theta'$  are the frequency variables (in radians per sample) before and after the warping, respectively. In Fig. 5.2, this function is plotted for different warping parameters  $\lambda$ . With  $\lambda = 0$  the transfer function of the all-pass is simplified to a unit delay and the frequency axis is not warped  $\theta' = \theta$ . For a negative  $\lambda$ , the function is mirrored at the line  $\theta' = \theta$  compared to the function with  $-\lambda$ . This property suggests a simple rule to obtain the inverse warping function<sup>1</sup>.

<sup>1</sup>Note, that the inverse warping function is not needed to invert the filterbank. We need this function to obtain the effective center frequencies of the frequency-warped filterbank.

Since the transform, which usually corresponds to a uniform filterbank, is applied to the frequency-warped signal, the effective center frequencies are mapped according to the inverse frequency-warping function  $\theta(\theta')$ . This is illustrated in Fig. 5.2. Utilizing the above mentioned property, the inverse frequency-warping function can be expressed as

$$\theta(\theta') = \arctan \left( \frac{(1 - \lambda^2) \sin(\theta')}{(1 + \lambda^2) \cos(\theta') + 2\lambda} \right). \quad (5.3)$$

Frequency-warped filterbanks mimic an important property of the cochlea. The warping by the all-pass-chain corresponds to the frequency-position mapping. The uniform frequency analysis of the warped signal corresponds to the approximately equally spaced inner hair cells along the basilar membrane.

## 5.2.2 Design Recipe for an Auditory Filterbank

This section presents a design strategy to obtain a non-decimated frequency-warped transform filterbank that approximates a gammatone auditory filterbank well. We choose the gammatone filters because they are based on the more recent and more accurate ERB and not on the classical critical bandwidth (see section 2.2.1) and because their time-domain description constitutes the ideal starting point for the design. The gammatone impulse response of Equ. (2.6) used in the previous chapters has a cosine carrier and is real-valued. We therefore use a discrete cosine transform (DCT) for the modulating transform  $\mathbf{T}$  and perform the simulations using a DCT of type 4 (DCT-4). The following design recipe can be used for other transforms as well. The design consists of two major steps: (i) the selection of a proper frequency warping parameter  $\lambda$  and (ii) the design of the ideal transform window sequence.

### 5.2.2.1 How to choose $\lambda$ ?

Smith and Abel [107] proposed an analytical expression for choosing a proper  $\lambda$  to achieve a frequency warping close to that of the Bark scale for a given sampling frequency. For a sampling frequency of 8 kHz, this expression yields  $\lambda = 0.4$ . It can be seen in Fig. 5.3 that the all-pass transform with  $\lambda = 0.4$  fits the Bark scale well. Therefore, warping a uniform filterbank with a cascade of first-order all-pass filters yields a good approximation of auditory filterbanks for critical band spectral analysis.

Since roex and gammatone auditory filters are based on the ERB rate, we are interested in modeling the ERB rate rather than the critical-band rate. Smith and Abel also proposed an expression for approximating the ERB rate scale, which suggests a warping parameter of  $\lambda = 0.58$  for 8 kHz sampling rate. The comparison

in Fig. 5.3 shows that the ERB rate cannot be approximated as well as the Bark scale. A fair compromise can be obtained with  $\lambda = 0.46$  that yields a curve in-between the ERB rate and the Bark scale.

It makes sense to compare also the resulting normalized bandwidth-mapping functions, i.e., the first derivative of the inverse frequency mapping functions with respect to the normalized warped frequency. In Fig. 5.4 this comparison is done for the same warped frequency scales as shown in Fig. 5.3. The figure shows the numerical derivative with respect to an increment in warped frequency that corresponds to 0.1 ERBs. A bandwidth mapping close to the ERB cannot be achieved but the curve obtained by the all-pass transform with  $\lambda = 0.46$  is again in-between the ERB and the critical bandwidth. For the further filterbank design, we select the warping parameter  $\lambda = 0.46$  for 8 kHz sampling rate.

### 5.2.2.2 How to choose the prototype low-pass filter?

We want to approximate a gammatone filterbank that has channel centers equally spaced on the ERB rate frequency scale (see Equ. (2.13)) and channel bandwidths proportional to the ERB (see Equ. (2.7)). Consequently, all impulse responses have different carrier frequencies and different gamma functions as the envelopes. Let the center frequency (in Hz) of one channel be  $f_c$ . Then, the envelope of the continuous-time filter's impulse response is

$$\gamma(t) = t^3 e^{-2\pi 1.019 \text{ERB}(f_c)t}, \quad \text{for } t > 0. \quad (5.4)$$

An approximate match of the carrier frequencies can be achieved by selecting the right warping parameter. Finally, we have to answer which prototype low-pass filter needs to be selected for the window coefficients.

For a uniform transform filterbank, the prototype low-pass impulse response specifies the envelope of all channel's impulse responses. When the all-pass transform is applied to obtain a non-uniform transform filterbank, the envelopes are modified: for low-frequency channels with narrow bandwidths, the impulse response envelopes are widened and for high-frequency channels with broad bandwidths, the impulse response envelopes are temporally compressed. What envelope should the prototype filter for such a warped transform filterbank have? The answer is to search for the frequency  $\theta_p$  so that the warping function  $\theta'(\theta_p)$  of Equ. (5.2) has a unity slope, i.e.,

$$\left. \frac{d\theta'(\theta)}{d\theta} \right|_{\theta=\theta_p} = 1. \quad (5.5)$$

As discussed above, the derivative specifies the bandwidth mapping, and a unity slope means that the bandwidth of a filter is not modified by the all-pass transform.

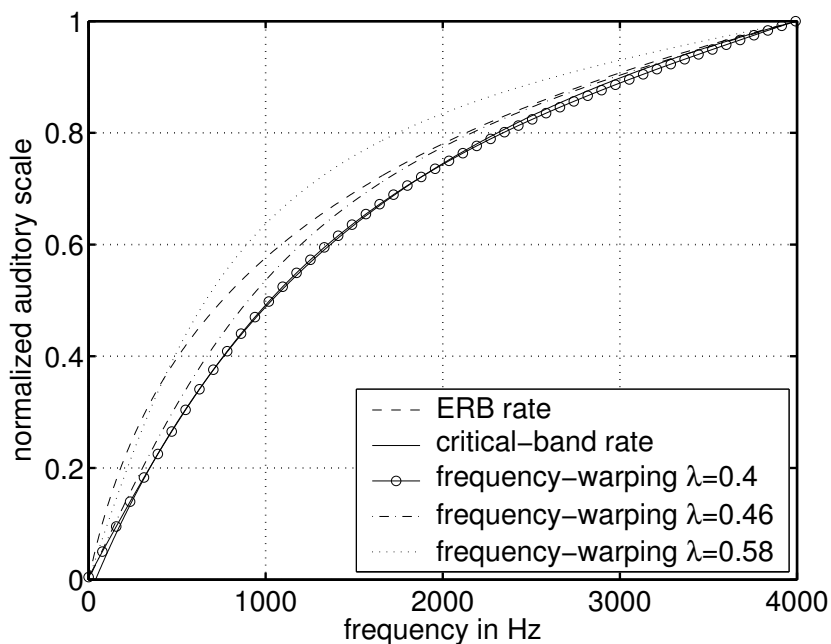


Figure 5.3: Comparison of the ERB rate (Glasberg and Moore, 1990 [34]), the Bark scale (Traunmüller, 1990 [38]), and the frequency warping with three different values for  $\lambda$ .

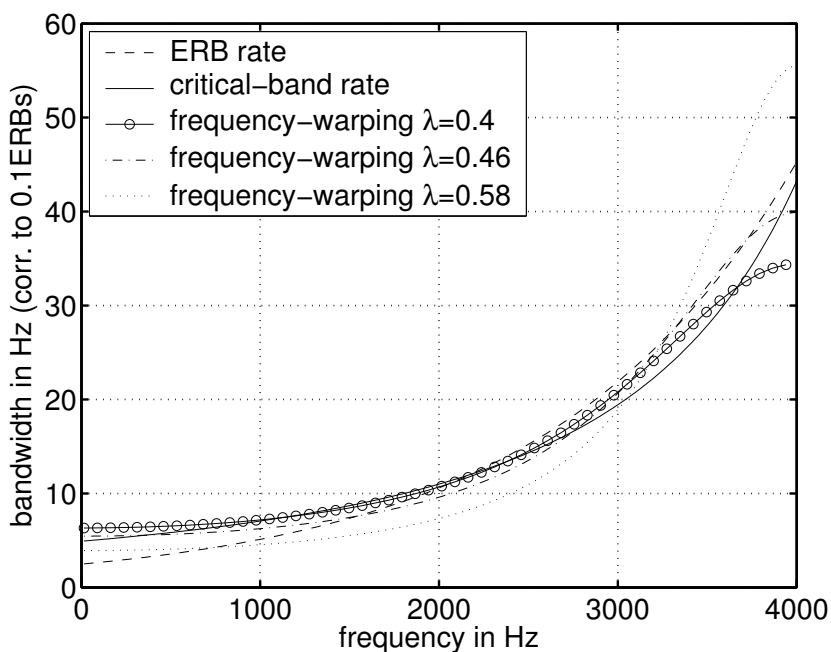


Figure 5.4: Comparison of the normalized bandwidth mapping of ERB rate, Bark scale, and frequency warping with three different values for  $\lambda$ .

When we specify a prototype gamma envelope in accordance with the center frequency  $f_c = \theta_p f_s / (2\pi)$  Hz, which is a sampled version of the gamma function of Equ. (5.4), we can ensure that the impulse response envelope for a filter with center frequency  $f_c$  is ideally approximated. For all other center frequencies we rely on a proper envelope widening or compression performed by the all-pass transform. From the discussion above we know that the ERB bandwidth mapping can only be approximated by the all-pass transform. Therefore, we have to face deviations from the ideal bandwidths. A distribution of the deviations over the whole frequency range enables further optimization. For the proposed design strategy, we accept these errors. Setting the derivative of Equ. (5.2) equal to 1 yields

$$\theta_p = \arccos(\lambda) \quad (5.6)$$

and consequently

$$\gamma(t) = t^3 e^{-2\pi \cdot 1.019 \text{ ERB}(\arccos(\lambda) f_s / (2\pi)) t}. \quad (5.7)$$

To obtain the window coefficients, we need to window and sample the continuous gamma function  $\gamma(t)$ . We assume to have the length of the window  $N_w$  as a constraint for the design. The number of points should be a power of 2 to allow a fast computation of the DCT (refer to [108] for a fast DCT-4 algorithm). To get an optimum prototype filter, the effect of windowing has to be minimized. This requires that the windowed sequence has to be a smooth function. Another requirement for our design is that the final windowed sequence exhibits maximum similarity to the gamma function to obtain gammatone filters. A straight-forward method to cut out a smooth function is to clip the decaying tail as well as the rising part of the gamma function by the same amount, i.e., we need to search for the shift  $t_0$  so that

$$\gamma(t_0) = \gamma(t_1), \quad t_1 > t_0, \quad (5.8)$$

where  $t_1 - t_0$  is specified by the number of points  $N_w$  of the window and the sampling rate  $f_s$ . In Fig. 5.5, the search for the optimum shift is shown. Note that the optimal shift  $t_0$  can generally be found only before the continuous function is sampled since  $t_0$  is not an integer multiple of the sampling interval in general. We therefore search for the optimal shift first and sample the interval of interest afterwards.

Once the shift is determined, we subtract the value  $\gamma(t_0)$  to obtain a function that starts and ends with a value of zero. For the final window we should not use coefficients that are zero because this effectively reduces the number of points from  $N_w$  to  $N_w - 2$ . To avoid this problem and yet ensure smoothness, we can divide the interval  $t_1 - t_0$  into  $N_w + 1$  sample intervals instead of  $N_w - 1$  and discard the first and the last zero samples. Fig. 5.5 also illustrates this method for sampling a gamma function. According to this way, the optimum shift has to be derived to

ensure:

$$\gamma(t_0) = \gamma(t_0 + (N_w + 1)/f_s). \quad (5.9)$$

We can solve Equ. (5.9) and obtain a closed-form solution for  $t_0$ :

$$t_0 = \frac{(N_w + 1)/f_s}{e^{2\pi b(N_w + 1)/(3f_s)} - 1}, \quad (5.10)$$

where  $b = 1.019 \text{ ERB}(\arccos(\lambda)f_s/(2\pi))$ . Finally, the finite impulse response of the prototype low-pass filter is calculated as

$$\tilde{h}[n] = a(\gamma(t_0 + (n + 1)/f_s) - \gamma(t_0)), \text{ for } n = 0, \dots, N_w - 1 \quad (5.11)$$

and the corresponding window coefficients as the time-reversed response  $w_n = \tilde{h}[N_w - 1 - n]$  as discussed earlier. The coefficient  $a$  can be used when the window needs to be scaled (e.g., for normalization).

The proposed procedure for windowing and sampling a gamma function is not only applicable for the design of a prototype low-pass filter of a transform filterbank. It can also be used for the design of regular FIR gammatone filters as applied in the previous chapters. However, the provided finite impulse responses have to be longer than the prototype low-pass of a frequency-warped transform filterbank to achieve a comparable approximation quality. This is particularly important for low-frequency channels with wide gamma envelopes. For a frequency-warped transform filterbank, the envelope widening is done automatically.

### 5.2.2.3 Computational Complexity

Let us assume that the length of the window  $N_w$ , which is equal to the size of the transform, is a power of 2. When the DCT-4 is computed according to [108],  $N_w \log_2(N_w)$  real operations (multiplications plus additions) are required per transform.  $N_w$  multiplications are needed for the window. One all-pass filtering operation can be performed using 2 additions and 1 multiplication<sup>2</sup>. For the all-pass filter cascade, we need  $3(N_w - 1)$  operations. Therefore, the overall computational effort is  $N_w(\log_2(N_w) + 4) - 3$  operations to calculate  $N_w$  non-decimated auditory filter output samples (one sample per channel).

### 5.2.2.4 Simulation Results

In Fig. 5.6, the frequency responses of four effective analysis filters of a frequency-warped 64-point DCT-4 filterbank are plotted and compared with ideal gammatone and roex auditory filters. The sampling rate is 8 kHz. The shown DCT-4 channels

---

<sup>2</sup>The multiplication with the single coefficient  $\lambda$  can be performed after the summation in the difference equation:  $y[n] = x[n - 1] + \lambda(y[n - 1] - x[n])$ .

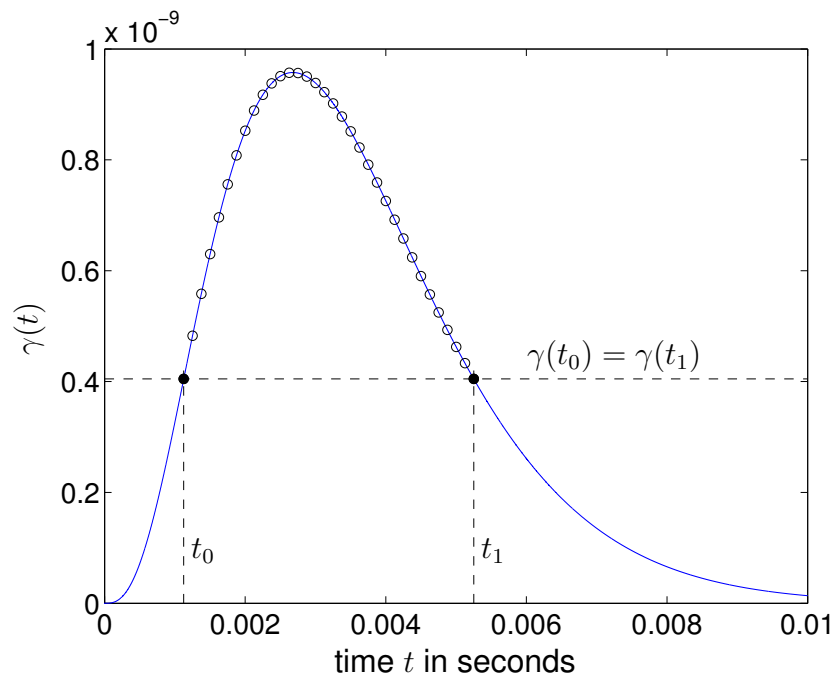


Figure 5.5: The design of a finite-length window sequence for the prototype low-pass filter with  $N_w = 32$  points from a gamma function. The shift by  $t_0$  ensures to maintain maximal similarity to the gamma function. The value of the start and the end point (indicated by the two filled circles) is subtracted from the samples for the final window sequence, and the start and the end point are not used. In this way, smoothness can be ensured, and the effects of windowing are minimized.

have the indices 6, 27, 41, and 55 with the corresponding unwarped carrier frequencies  $f_{c,k} = (k + 1/2)f_s/(2N_w)$  of 406.3 Hz, 1718.8 Hz, 2593.8 Hz, and 3468.8 Hz. The warping parameter has been selected as  $\lambda = 0.46$ . Since the transform is applied to an all-pass-transformed version of the input signal, the effective carrier frequencies are determined by the inverse frequency warping function  $\theta(\theta')$ , which gives 151.4 Hz, 732.9 Hz, 1376.6 Hz, and 2676.2 Hz. The window has been designed as described in the last section. The center frequency with unity bandwidth mapping that is used for the prototype gamma function is  $f_c = 1391.4$  Hz. The ideal shift  $t_0$  corresponds to roughly 3.3 samples. As revealed by the figure, the ideal filters can be approximated with high accuracy, particularly in the frequency range from 500 Hz to 2000 Hz. At very low frequencies and above 2000 Hz the resulting bandwidths are wider than the ideal filters.

In Fig. 5.7 two effective impulse responses of the frequency-warped 64-point DCT-4 filterbank are shown. We consider the channels with indices 27 and 41, which are also shown in Fig. 5.6 (the two middle filters). Their Hilbert envelopes are compared with the ideal gamma functions according to their actual carrier frequencies. As shown by the figure, we are able to approximate the ideal gammatone impulse responses (and their envelopes) well by the warped transform filterbank. Note that the length of the effective impulse responses is longer than the support of the prototype low-pass filter  $N_w$ . This is a direct consequence of the all-pass transform, which results in infinite-length impulse responses.

A window length of only 64 samples at a sampling rate of 8 kHz yields reasonable frequency responses and practical gammatone impulse responses. Therefore, the usage of a frequency-warped transform filterbank constitutes a computationally efficient and memory saving option for an auditory filterbank implementation on a DSP for real-time applications. The computational complexity to compute one output sample in all 64 channels is only 637 operations (real multiplications plus additions). For comparison, when all 64 output channels are of interest, the all-pole IIR implementation of [98] requires already  $64 \cdot 17 = 1088$  operations. The IIR implementation achieves a much better approximation of the ideal gammatone filters, but the inversion is easier for the frequency-warped transform filterbank, as described in the next section.

The described filterbank implementation method is not only attractive for the usage in perceptual-domain coders but also in the auditory models of classical perceptual coders. For these models, often decimated analysis filterbanks are used. A decimated frequency-warped transform filterbank can be obtained by downsampling the input signals of the transform  $\mathbf{T}$ . In this case, the transform  $\mathbf{T}$  is computed at a lower rate<sup>3</sup>, and the computational complexity is decreased even further. When we consider a decimation factor  $N$ , the computational cost is

---

<sup>3</sup>The all-pass cascade has to be operated at the original rate.

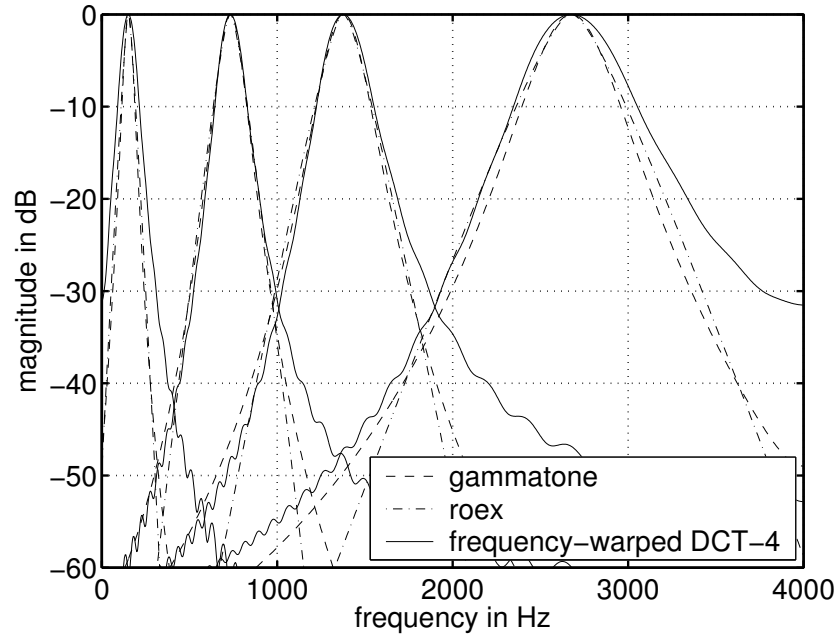


Figure 5.6: Normalized frequency responses of 4 channels of auditory filterbanks with center frequencies of 151 Hz, 733 Hz, 1377 Hz, and 2676 Hz. Comparison between gammatone filters, rounded exponentials, and a frequency-warped DCT-4 filterbank ( $\lambda=0.46$ ,  $f_s=8$  kHz, 64-point gamma window, channels 6, 27, 41, and 55).

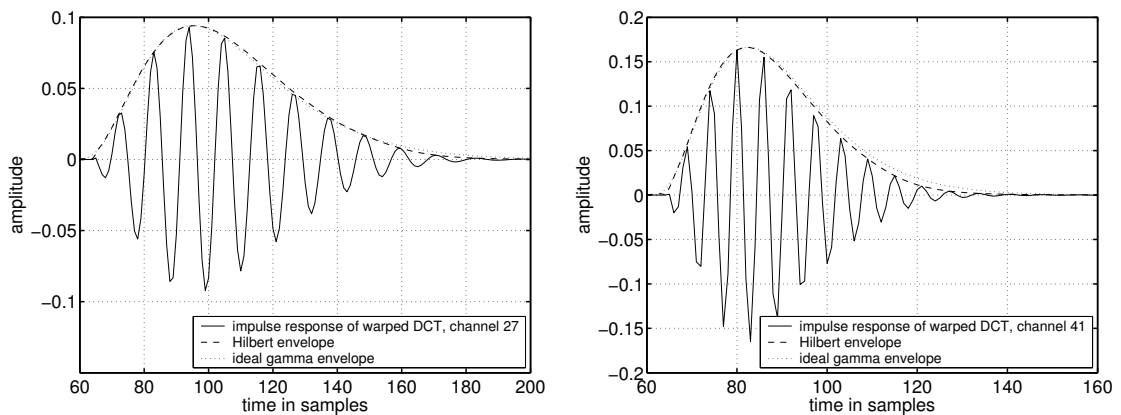


Figure 5.7: The resulting impulse responses of two channels (27 and 41) of the frequency-warped DCT-4 filterbank. The ideal gamma envelopes for the corresponding center frequencies (733 Hz and 1377 Hz) are shown for comparison.

$N_w(\log_2(N_w) + 1 + 3N) - 3N$  operations per frame of length  $N$  samples.

### 5.2.3 Filterbank Inversion

As applied in [105], an inverse filterbank (or synthesis filterbank) can be obtained by generalizing the overlap-and-add procedure that is well known from the inverse short-term Fourier transform in the same way as the sliding window—by replacing the unit-delay line with an all-pass cascade (see Fig. 5.8). While the uniform frequency resolution analysis-synthesis filterbank achieves perfect reconstruction under the condition that the overlapping windows sum up to a constant, the frequency-warped version does not.

For the application in a perceptual-domain coder, we are interested in a non-decimated filterbank. Therefore, we only consider the inversion of a non-decimated frequency-warped transform filterbank here. We assume that the transform is invertible, i.e.,  $\mathbf{T}\mathbf{T}^{-1} = \mathbf{I}$  with  $\mathbf{I}$  the identity matrix. For the simple case, when the window length  $N_w$  equals the number of channels  $K$ , we can normalize the window to satisfy  $\sum_{i=0}^{N_w-1} w_i^2 = 1$ , and we obtain for the output signal

$$\hat{X}(z) = X(z)A^{N_w-1}(z). \quad (5.12)$$

This equation shows that a phase distortion is introduced. The distortion increases with the number of all-pass filters in the cascade. In [109, 110], FIR filters are used to compensate for this phase distortion to get a near-perfect-reconstruction filterbank. This compensation filter introduces an additional delay, which increases with decreasing compensation error. However, it is not necessary to equalize for perfect linear phase since small phase distortions are inaudible. The case where a longer prototype filter is used without a higher number of auditory channels, i.e.,  $N_w > K$ , is considered in [105, 106].

In a recent development [111], we have shown that an FIR synthesis filterbank exists for a critically sampled frequency-warped transform filterbank which achieves perfect reconstruction. In Appendix C the closed-form solution for the synthesis filters is derived. However, these synthesis filters amplify any quantization noise introduced in the subband signals and do not exhibit band-pass characteristics. For this reason, they are not recommended for coding applications.

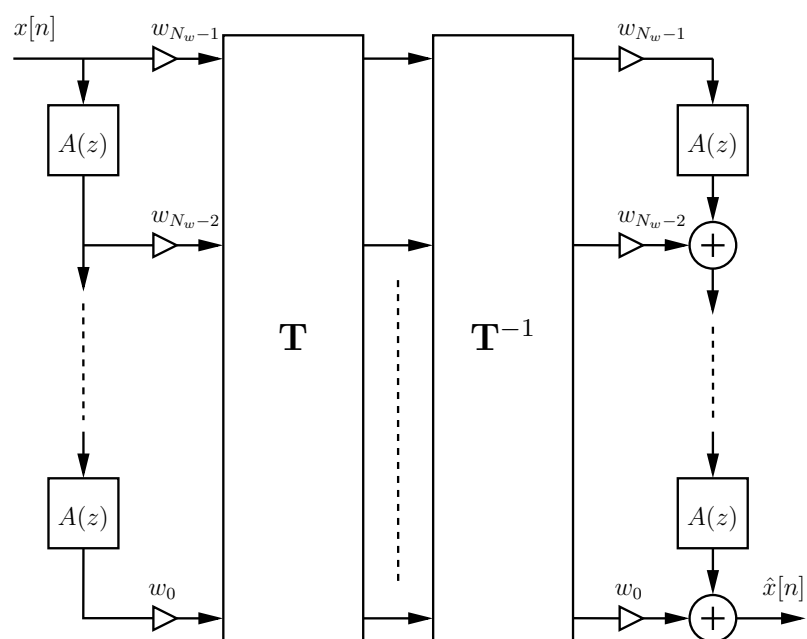


Figure 5.8: Non-decimated frequency-warped analysis-synthesis filterbank with perfect overall magnitude frequency response but distorted overall phase response.



# Chapter 6

## Summary and General Conclusions

Chapter 1 of this thesis has clearly motivated the consideration of the approach of ‘coding in the perceptual domain’ but calls attention to the trade-off between the advantages of a simple and yet accurate distortion measure and the problem of redundancy. This thesis is dedicated to mitigate the limiting effects caused by this trade-off.

We have reviewed an invertible auditory model and its usage for coding of speech and audio signals in chapter 2. The inversion procedure to reconstruct the original signal from its auditory representation does not need computationally expensive iterative algorithms and produces reconstructed audio signals with transparent quality. The frame-theoretical interpretation of the inversion procedure in section 2.3.4 as well as the communication-theoretical considerations in section 3.2 support the application of the used invertible model for coding. For the subjective quality assessment, a listening test has been performed in section 3.5.2, and it has been found that signals processed by the invertible model are practically indistinguishable from the original when the peak-picking procedure is performed at a twice oversampled rate.

An overview of auditory frequency scales, auditory filterbanks, and their inverse filterbanks has been given. This overview includes physiological and psychophysical facts and frame-theoretical considerations given in chapter 2. In chapter 5 we have shown that a frequency-warped transform filterbank constitutes a computationally efficient and memory saving implementation method for an auditory filterbank. Furthermore, a design recipe has been proposed in section 5.2.2 to well approximate a gammatone filterbank using a frequency-warped transform filterbank based on a first-order all-pass transform. This design strategy results in a relatively short prototype filter and, nevertheless, obtains an approximation with high accuracy.

Motivated by the fact that the auditory pulse representation obtained by our invertible model is highly redundant, special attention has been paid to reducing the number of pulses. In chapter 3 a joint time-frequency masking model has been proposed to decide whether a pulse is needed or not. The so-called impact factor used for this decision controls the finally perceived quality approximately linearly as has been confirmed by a listening test. The sparsification procedure is fast since it avoids computationally expensive analysis-by-synthesis schemes that may result in exhaustive search routines. In addition to the sparsification method, a pulse amplitude correction scheme has been proposed in section 3.4 to compensate for the loss of signal energy and thus to ensure proper signal reconstruction. The sparsified pulse representation can contain less pulses than the original signal has samples. The resulting signal reconstruction is basically the summation of rather isolated time-reversed basilar membrane impulse responses. The experiments have shown that the proposed coding method is able to hide a considerable amount of quantization noise due to the temporal fine structure preserved by the used signal representation. From the comparison with another temporal masking model and from the experiments using dynamic nonlinearities to model the adaptation of the hearing system to the stimulus in section 3.6, it can be expected that the sparsification method allows further optimization to obtain even sparser signal representations, especially in high-frequency channels.

The experiments of chapter 4 show that the sparsification process facilitates indeed efficient encoding. A lossless encoding of the pulse positions using pulse distance encoding has been investigated in section 4.2 as well as a lossy encoding using a new variable frame length vector quantizer in section 4.4. Contrary to existing coders, this vector quantizer does not need to be embedded in computationally costly analysis-by-synthesis loops, but employs a new similarity measure suitable for sparse vectors proposed in section 4.4.1. Furthermore, the codebook size can be kept small due to the signal-dependent synchronization of frames. Experiments have shown promising results. However, to obtain a proper bit allocation and yet transparent reconstruction quality, further experimental work is needed to ascertain limits on the quantization of both pulse amplitudes and pulse positions. Furthermore, the problem of efficiently encoding pulse trains by means of a joint position-amplitude quantization calls for a careful extension of classical rate-distortion theory to be applicable for sparse sources. The proposed similarity measure represents a first approach towards this direction.

# Appendix A

## Outer and Middle Ear (OME) Weighting

Glasberg and Moore [34] have suggested to use the inverse shape of equal-loudness contours at high levels to model the transmission characteristics of the outer and the middle ear. Also in [85] an inverse 100-phon contour has been used. In this thesis the modified inverse 100-phon contour from [112] is used, which has been modeled by a linear shift-invariant system with frequency response

$$H_{i100}(e^{j\theta}) = \frac{\sum_k b_k e^{-jk\theta}}{\sum_k a_k e^{-jk\theta}}. \quad (\text{A.1})$$

The coefficients of the numerator polynomial  $b_k$  and of the denominator polynomial  $a_k$  are listed in Table A.1. The underlying sampling rate is 44100 Hz. The magnitude of the frequency response  $|H_{i100}(e^{j\theta})|$  is shown in Fig. A.1.

The computation of the unit pulse BM excitation patterns in section 3.3 only needs simple channel weights. These weights are calculated according to

$$\varrho_k = |H_{i100}(e^{j2\pi f_{c,k}/44100})| \quad (\text{A.2})$$

where  $f_{c,k}$  is the center frequency in Hz of the  $k$ th auditory channel.

$k$	$a_k$	$b_k$
0	1.0	0.2
1	-6.37962882026536	-0.80938476441234
2	18.22371448121400	0.90389694815738
3	-31.05610768291973	0.61821572006517
4	35.88179603101454	-2.16411897387062
5	-30.86430161016611	1.27744916284003
6	21.49293566172255	1.08590479315555
7	-12.65921215504449	-1.99173553537713
8	6.06493896644346	0.81639655260282
9	-2.11971588743927	0.55582353541564
10	0.46178621000873	-0.78415534135712
11	-0.04620481009893	0.34963188146863
12	0.0	-0.05792397868800

Table A.1: Coefficients of the OME-weighting filter at a sampling rate of 44100 Hz.

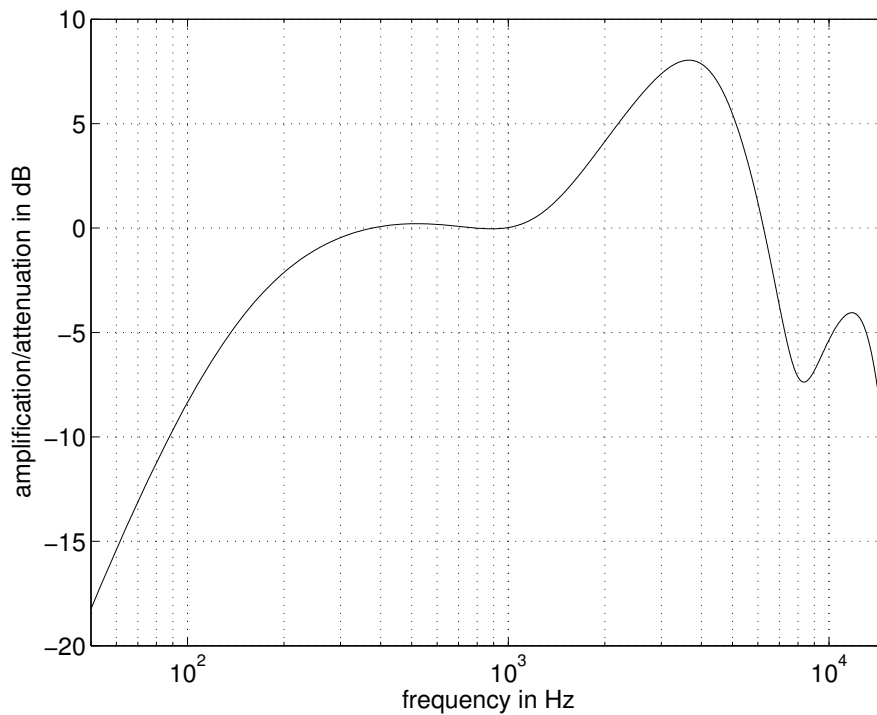


Figure A.1: Magnitude of the frequency response of the OME-weighting filter.

# Appendix B

## Modeling Temporal Adaptation

The functional model proposed by Dau et al. in [19] uses cascaded adaptation loops with different time constants to model the temporal adaptation to the stimulus. Fig. B.1 shows the adaptation circuit that is used in each auditory filterbank channel after the envelope calculation<sup>1</sup>. The circuit consists of a minimum threshold, a cascade of five automatic gain controllers (AGCs), and a final low-pass filter.

For our simulations, the threshold was chosen to be  $MIN = 10^{-5}$  as in [113]. This minimum models the absolute threshold. Together with an assumed maximum input signal of  $MAX = 1$ , a dynamic range of 100 dB is considered (as in [113]).

The input signal of an AGC stage is divided by the charging state of the low-pass filter. In Fig. B.1 the low-pass filters are presented as capacitors that are charged through a resistor. The time constants  $\tau_k$  of the first-order low-pass filters are 5 ms, 50 ms, 129 ms, 253 ms, and 500 ms (taken from [19]) with the smallest for the first ( $k = 1$ ) stage and the largest for the last ( $k = 5$ ) stage. The final low-pass filter has a time constant of 20 ms. In our simulations, all first-order low-pass filters are implemented in a discrete-time version as:

$$divisor_k[n + 1] = a_k \cdot divisor_k[n] + b_k \cdot y_k[n] \quad (\text{B.1})$$

with

$$a_k = e^{-1/(\tau_k f_s)} \quad \text{and} \quad b_k = 1 - a_k \quad (\text{B.2})$$

for  $k = 1, \dots, 5$ . The same difference equation with corresponding input and output is also used for the final low-pass filter. In Dau's later work [80] the final low-pass filter is omitted and further processing is performed by a filterbank that covers frequencies up to 1 kHz ('modulation filterbank'). For most of the simulation experiments reported in this thesis, the final low-pass filter is also omitted<sup>2</sup>.

---

<sup>1</sup>In [19] the envelopes are computed by half-wave rectification and 1 kHz-low-pass filtering.

<sup>2</sup>Actually it is replaced by a 2-tap moving average filter that eliminates the oscillation at half

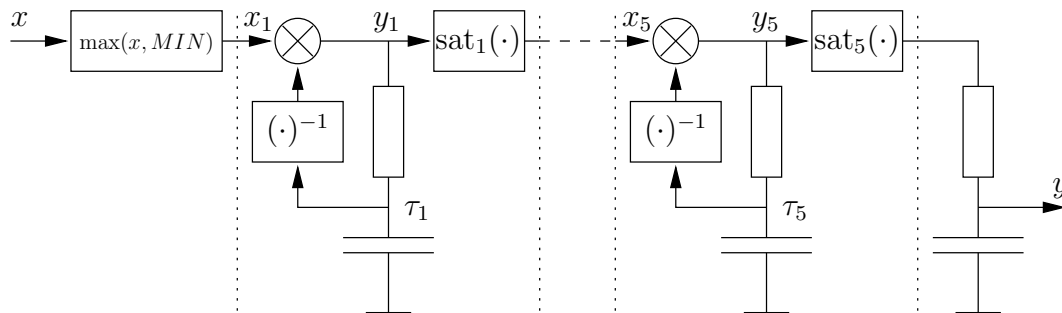


Figure B.1: The adaptation circuit of [19]: minimum threshold, five cascaded adaptation loops, and a final low-pass filter.

For constant input signals  $x$  greater than  $MIN$ , the adaptation circuit asymptotically behaves like a static nonlinearity. One AGC maps its input as the square root to the output:  $y_k = \sqrt{x_k}$ . Consequently, the cascade of 5 AGCs asymptotically transforms stationary input signals according to  $y = x^{1/32}$ . This power law approximates a logarithmic conversion to dB ( $20 \cdot \log_{10}(x)$ ) in the considered input range when scaled and shifted according to  $330 \cdot y - 330$ . In this work the unit of  $330 \cdot y - 330$  is given in 'model units' (cf. [19]).

On the other hand, rapidly varying input signals are transformed more linearly since the capacitors cannot be charged to follow the signal in the given time. Thus, the gain remains relatively constant. As discussed in [89], the feedback loops show an extremely high sensitivity at signal onsets. To better model forward masking, some modifications are proposed in [80]. These modifications provide each adaptation loop with a saturation stage according to Münkner [114]. This saturation limits the output of the  $k$ th stage to ten times the steady-state output range of the loop:  $10 \cdot (MAX^{1/2^k} - MIN^{1/2^k})$ . Münkner suggests to use a soft saturation curve, which starts to compress values greater than one:

$$\text{sat}_k(x) = \begin{cases} x, & x \leq 1 \\ C_k \left( \frac{2}{1+e^{-2(x-1)/C_k}} - 1 \right) + 1, & x > 1 \end{cases} \quad (\text{B.3})$$

The resulting limit of this function  $\lim_{x \rightarrow \infty} \text{sat}_k(x) = C_k + 1$ . In Fig. B.2 this function is plotted for the actually used limits of the five AGC stages. The first stage has the highest steady-state output range and therefore the highest limit.

---

the sampling rate caused by the discrete-time implementation of the feedback loop: a change in charging state results in a change of the gain factor, which yields a change in the output signal and again a change in charging state. The 2-tap moving average filter, which has a zero at  $z = -1$  in the complex  $z$ -plane, is sufficient to suppress this oscillation.

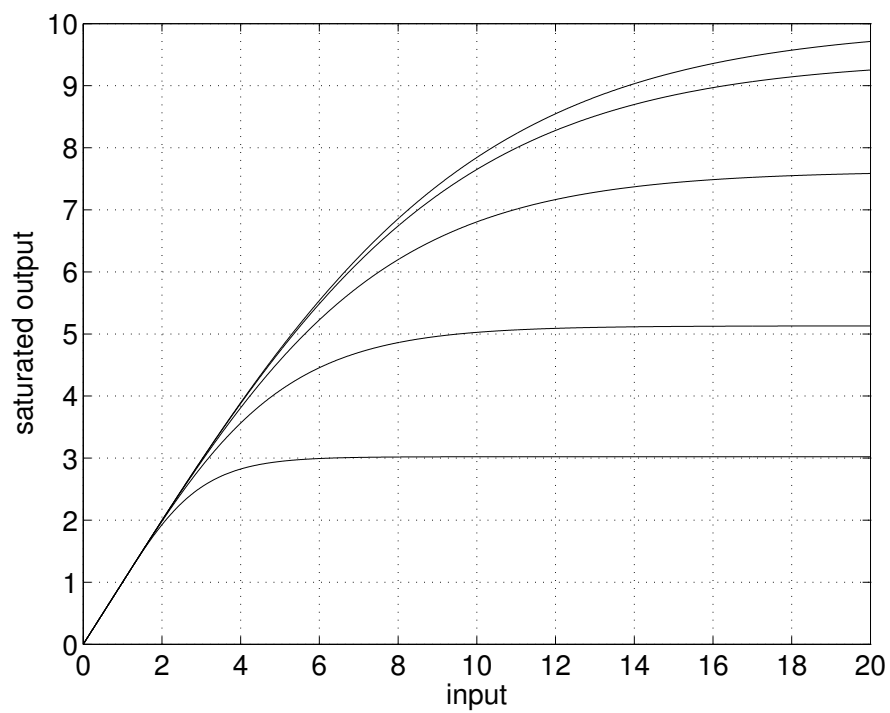


Figure B.2: Saturation functions to compress values greater than one for the five adaptation loops.

Likewise the last stage has the smallest one. Also the range of the overall output  $y$  is limited by the last saturation stage.



# Appendix C

## Critically Sampled Frequency-Warped PR Filterbank

Chapter 5 is dedicated to alternative implementation methods for non-decimated auditory filterbanks, and a frequency-warped transform filterbank has been shown to be a computationally efficient option. This appendix deals with a critically sampled frequency-warped DCT and DFT filterbank, and it summarizes our work proposed in [111]. Contrary to many earlier observations, we prove the existence of stable and causal perfect reconstruction (PR) synthesis filters for the case of non-singular frequency-warping all-pass filters. Furthermore, we prove that for first-order frequency warping these synthesis filters have a finite impulse response, and we provide a closed-form solution. We also apply the PR solution to oversampled filterbanks and conclude with simulation experiments and the observation that these filters do not exhibit well concentrated band-pass characteristics.

In chapter 5 a brief history of frequency-warped filterbanks is given. Different approaches to obtain a critically sampled analysis-synthesis filterbank have been proposed. In [115], Laine suggested a critically sampled non-uniform analysis and synthesis filterbank employing a block-recursive algorithm based on his ‘FAMlet’ transform. Here critical is understood as ‘critical on average’, i.e., the subbands do not all have the same decimation factors, but the total number of samples per time unit is preserved when summing over all channels. As a drawback, this method does not achieve perfect reconstruction.

Evangelista and Cavaliere [116] worked on frequency-warped wavelets based on warped quadrature mirror filters. Such a filterbank is able to achieve perfect reconstruction but cannot be used for real-time applications due to the time reversal of the entire input signal as done in [102].

Makur and Mitra [117] introduced the warped DFT and its inverse. Here, we do not frequency-warp the signal, rather, we directly non-uniformly sample the  $z$ -transform of a finite-length signal on the unit circle. This yields filterbanks with

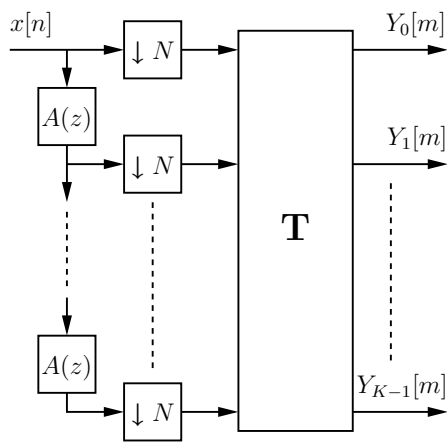


Figure C.1: Frequency-warped analysis filterbank.

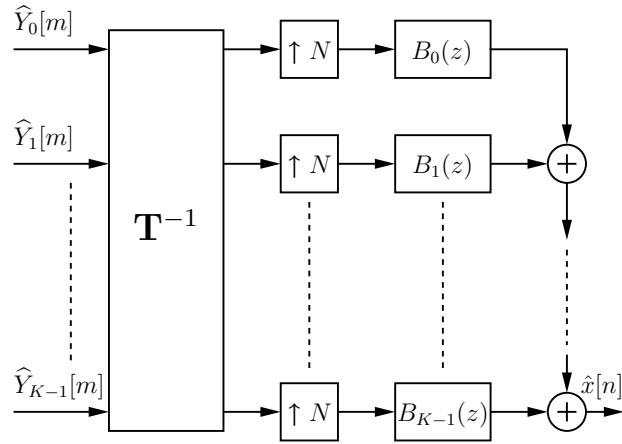


Figure C.2: Synthesis filterbank.

non-uniformly spaced center frequencies but constant bandwidths. Therefore, the method cannot be used for auditory filter approximation.

In [39] an extensive overview of applications of frequency warping in audio signal processing is given. Furthermore, it has been stated that for warped filterbanks critical subsampling with perfect reconstruction is impossible in most cases. Similarly in [106] it is asserted that a warped transform filterbank leads to unstable synthesis filters when perfect reconstruction is desired. In the next section we show that stable synthesis filters exist and even that they have finite impulse responses for the case of frequency warping with first-order all-pass filters.

As we noticed after the publication of [111], Shankar and Makur had reported in [118] that an FIR PR synthesis filterbank exists for frequency warping using first-order all-pass filters. However, our closed-form solution results in a proof of this statements, whereas [118] is lacking of similar derivations.

## C.1 Exact Solution for the Synthesis Filters

In this section we consider a critically sampled filterbank, i.e., the number of channels  $K$  equals the decimation factor  $N$ . The analysis filterbank is shown in Fig. C.1 with  $A(z)$  the transfer function of an all-pass filter. The downsampling is performed before the transform  $\mathbf{T}$  (cf. Fig. 5.1 where no decimation is done). For such an analysis filterbank, we want to compute the transfer functions of the synthesis filters  $B_k(z)$  for  $k = 0, \dots, N - 1$  of the synthesis filterbank shown in Fig. C.2 such that  $\hat{x}[n]$  is a delayed version of  $x[n]$ , i.e.,  $\hat{x}[n] = x[n - \Delta]$ .

We directly connect the outputs of the analysis stage to the inputs of the synthesis stage  $\hat{Y}_k[m] = Y_k[m]$ , i.e., we do not consider quantization or coding yet.

The cascade of the transform  $\mathbf{T}$ , e.g. a DFT or a DCT, and its inverse  $\mathbf{T}^{-1}$  can be simplified since  $\mathbf{T}\mathbf{T}^{-1} = \mathbf{I}$ . After expressing  $\widehat{X}(z)$  in terms of  $X(z)$ , we have to require that all aliasing terms  $X(W_N^k z)$  disappear for  $k = 1, \dots, N-1$  where  $W_N = e^{-j2\pi/N}$ . Furthermore, we have to ensure that the overall system is a pure delay.

For convenience, we put all synthesis filters  $B_k(z)$  into the vector  $\mathbf{b}(z) = [B_0(z) B_1(z) \dots B_{N-1}(z)]^T$  and rewrite the problem as

$$\mathbf{A}(z)\mathbf{b}(z) = \mathbf{v}(z) = \begin{bmatrix} Nz^{-\Delta} \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \quad (\text{C.1})$$

The matrix  $\mathbf{A}(z)$ , which is referred to as the alias component matrix [119], has the form

$$\mathbf{A}(z) = \begin{bmatrix} 1 & A(z) & \dots & A^{N-1}(z) \\ 1 & A(W_N z) & \dots & A^{N-1}(W_N z) \\ \vdots & \vdots & \ddots & \vdots \\ 1 & A(W_N^{N-1} z) & \dots & A^{N-1}(W_N^{N-1} z) \end{bmatrix}. \quad (\text{C.2})$$

### C.1.1 Existence and Uniqueness

We can obtain the solution for  $B_k(z)$  using Cramer's rule

$$B_k(z) = \frac{D_k(z)}{D(z)} \quad (\text{C.3})$$

where  $D(z) = \det \mathbf{A}(z)$  and  $D_k(z)$  is the determinant obtained from  $\mathbf{A}(z)$  by replacing in  $\mathbf{A}(z)$  the  $(k+1)$ th column by  $\mathbf{v}(z)$ .

The square matrix  $\mathbf{A}(z)$  has the form of a Vandermonde matrix for which the determinant has the simplified structure [120]

$$D(z) = \prod_{\substack{k,l=0 \\ k>l}}^{N-1} (A(W_N^k z) - A(W_N^l z)). \quad (\text{C.4})$$

We can see that  $D(z) \neq 0$  as long as all elements  $A(W_N^k z)$  for  $k = 0, \dots, N-1$  are distinct. This is true for several classes of  $A(z)$  and also for the first-order all-pass

$$A(z) = \frac{z^{-1} - \lambda}{1 - \lambda z^{-1}}, \quad (\text{C.5})$$

which is also used in chapter 5 (see Equ. (5.1)), and a unique solution exists in these cases. For detailed requirements on  $A(z)$  refer to [118].

### C.1.2 FIR Property

To show the FIR property, it is necessary to examine Equ. (C.3) more closely.  $D_k(z)$  can be factored as

$$D_k(z) = Nz^{-\Delta} \tilde{D}(z) \bar{D}_k(z) \quad (\text{C.6})$$

where  $\tilde{D}(z)$  is a minor, which contains a subset of the factors of Equ. (C.4)

$$\tilde{D}(z) = \prod_{\substack{k,l=1 \\ k>l}}^{N-1} (A(W_N^k z) - A(W_N^l z)) \quad (\text{C.7})$$

and  $\bar{D}_k(z)$  is given by

$$\bar{D}_k(z) = (-1)^k \sum_{m=1}^{M_k} \prod_{l=1}^{N-1-k} \mathcal{C}_{kA}[m, l]. \quad (\text{C.8})$$

$\mathcal{C}_{kA}[m, l]$  is the  $l$ th element of the  $m$ th<sup>1</sup> combination of size  $(N-1-k)$  from the set  $\{A(W_N z), \dots, A(W_N^{N-1} z)\}$  and  $M_k$  is the number of combinations  $M_k = \binom{N-1}{N-1-k}$ . For illustration, for  $N=4$  and  $k=1$  we get

$$\bar{D}_1(z) = -1(A(W_N z)A(W_N^2 z) + A(W_N z)A(W_N^3 z) + A(W_N^2 z)A(W_N^3 z)).$$

We also need to define  $\prod_{l \in \Omega} a_l = 1$ . After canceling the common factors, we can rewrite Equ. (C.3) as

$$B_k(z) = \frac{Nz^{-\Delta} \bar{D}_k(z)}{\prod_{l=1}^{N-1} (A(W_N^l z) - A(z))}. \quad (\text{C.9})$$

Now, we use  $A(z) = \frac{P(z)}{Q(z)}$  with  $P(z) = 1 - \lambda z$  and  $Q(z) = z - \lambda$  defined as polynomials in  $z$  and write the double-rational expression as a rational function

$$B_k(z) = \frac{Nz^{-\Delta} (-1)^k Q^{N-1}(z) \sum_{m=1}^{M_k} \prod_{l=1}^{N-1-k} \mathcal{C}_{kP}[m, l] \prod_{l=1}^k \mathcal{C}_{kQ}[M+1-m, l]}{\prod_{l=1}^{N-1} (P(W_N^l z)Q(z) - P(z)Q(W_N^l z))}. \quad (\text{C.10})$$

---

<sup>1</sup>This implies that the combinations are ordered according to a rule, e.g. ascending indices, cf. the MATLAB function `nchoosek`.

$\mathcal{C}_{kP}[m, l]$  is the  $l$ th element of the  $m$ th combination of length  $(N-1-k)$  from the set  $\{P(W_N z), \dots, P(W_N^{N-1} z)\}$  and similarly,  $\mathcal{C}_{kQ}[m, l]$  is an element of a combination of length  $k$  from the set  $\{Q(W_N z), \dots, Q(W_N^{N-1} z)\}$ .

For the case that  $A(z)$  is the first-order all-pass of Equ. (C.5), the denominator of Equ. (C.10) reduces to  $N((1-\lambda)(1+\lambda)z)^{N-1}$  and we get

$$B_k(z) = \frac{(-1)^k Q^{N-1}(z) \sum_{m=1}^{M_k} \prod_{l=1}^{N-1-k} \mathcal{C}_{kP}[m, l] \prod_{l=1}^k \mathcal{C}_{kQ}[M_k + 1 - m, l]}{z^\Delta ((1-\lambda)(1+\lambda)z)^{N-1}}. \quad (\text{C.11})$$

From this equation it can easily be seen that all poles of  $B_k(z)$  are at  $z = 0$  and consequently, we obtain FIR filters. The degree of the numerator polynomial (which determines the number of filter coefficients) is  $2(N-1)$ .

### C.1.3 Causality and Delay

For an  $A(z)$  of Equ. (C.5), the degrees of the polynomials of numerator and denominator of Equ. (C.11) are equal for an overall delay of  $\Delta = N-1$  so the synthesis filters  $B_k(z)$  are causal for  $\Delta \geq N-1$ . Hence this generalized critically sampled transform filterbank produces the same delay as a non-overlapping short-term Fourier transform with inverse transform.

## C.2 Oversampled Filterbanks

Let the number of channels be greater than the decimation factor, i.e.,  $K > N$ . Now, the solution for the synthesis filters is no longer unique. One simple solution is to set  $B_k(z) = 0$  for  $k = N, \dots, K-1$  and the first  $N$  synthesis filters are obtained from Equ. (C.11) in the same way as in the critically sampled case, and we still achieve perfect reconstruction. Note that this is possible for any oversampled transform-like filterbank as soon as a solution for the critically sampled instance exists. For the extreme case when no decimation is done ( $N = 1$ ), we get  $B_0(z) = 1$  and consequently  $\hat{x}[n] = x[n]$ , i.e., a delay-free filterbank.

## C.3 Discussion and Experiments

In Fig. C.3 pole/zero diagrams are plotted for a critically sampled 4-channel synthesis filterbank with a warping parameter  $\lambda = 0.5$ . All filters share the common factors  $Q^{N-1}(z)$ , which can be extracted and implemented only once. These common factors are a high-pass filter for  $\lambda > 0$  and a low-pass filter for  $\lambda < 0$  and are

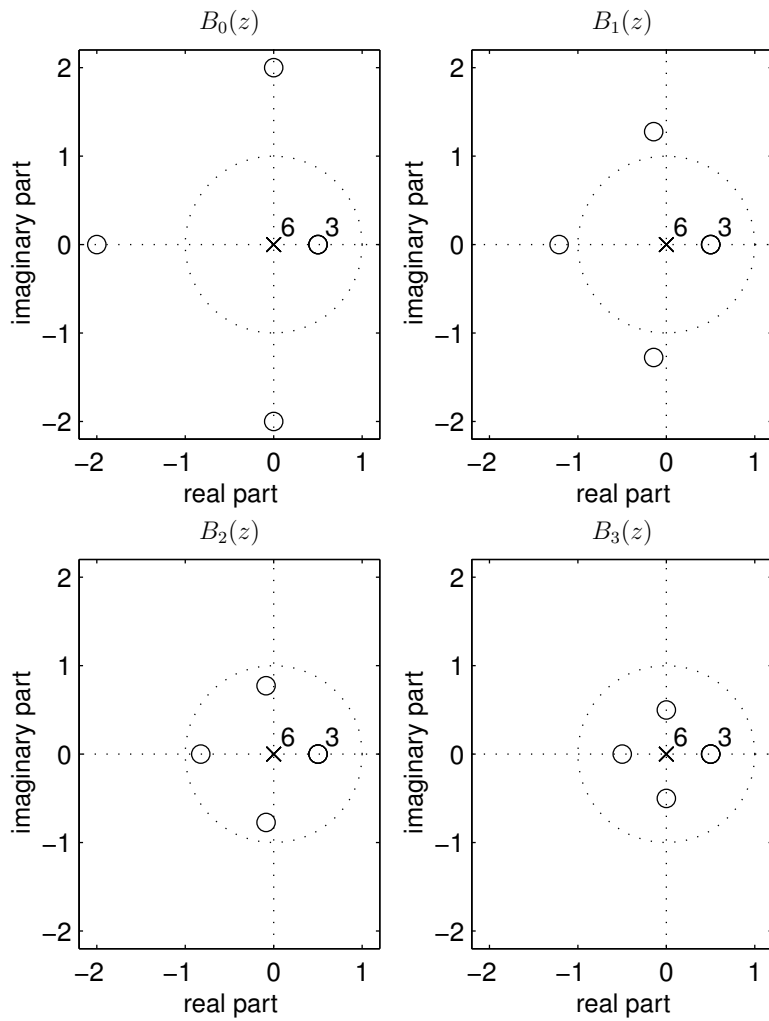


Figure C.3: Pole/zero diagrams of the synthesis filters for a 4-channel critically sampled filterbank with  $\lambda = 0.5$ . Crosses mark poles and circles mark zeros. Associated numbers denote multiple occurrences.

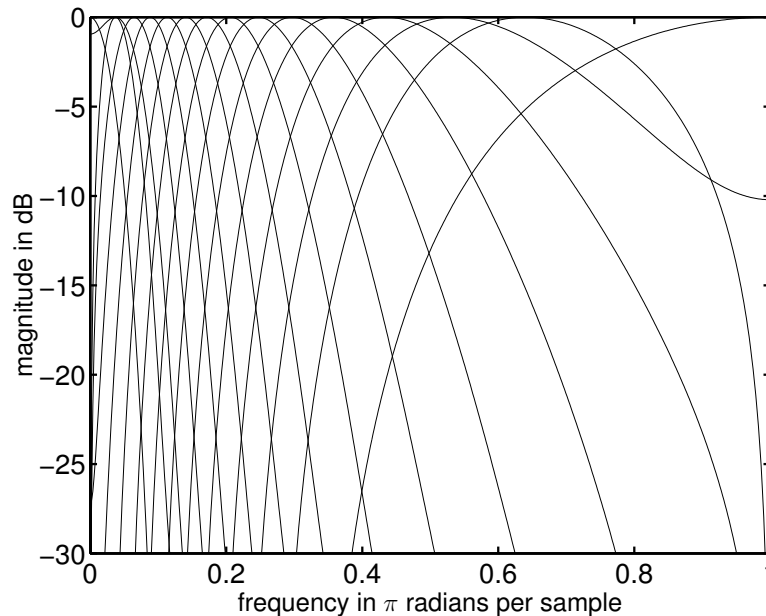


Figure C.4: Normalized frequency responses of the effective analysis filters of a 16-channel frequency-warped DCT filterbank with  $\lambda = 0.5$ . Here the signals are weighted by a Hamming window before the transform.

only responsible for equalizing the overall frequency response and do not cancel any aliasing. For  $B_{N-1}(z)$  we get the simplest expression because  $\overline{D}_{N-1}(z)$  reduces to  $(-1)^{N-1}$  and all zeros are located at  $W_N^l \lambda$ . Also the solution of  $B_0(z)$  is simple since  $\overline{D}_0(z) = \prod_{l=1}^{N-1} A(W_N^l z)$  and all non-common zeros are located at  $W_N^l / \lambda$ . Note that the zeros of the other synthesis filters  $B_1(z), \dots, B_{N-2}(z)$  are not on a circle. We found for the magnitude of the frequency responses the following relation:  $|B_k(e^{j\theta})| = |B_{N-1-k}(e^{j\theta})|$ .  $B_k(z)$  for  $k \geq N/2$  are minimum-phase filters and  $B_k(z)$  for  $k < (N-1)/2$  are non-minimum-phase versions where all non-common zeros are reflected outside the unit circle.

In Fig. C.4 the frequency responses of the effective non-decimated analysis filters of a 16-channel warped ( $\lambda = 0.5$ ) DCT filterbank are plotted. For a positive warping parameter  $\lambda$ , both the distance between adjacent center frequencies and the bandwidth increase with increasing frequency. To get a better side lobe attenuation, a Hamming window was applied prior the DCT. In such a case when a non-rectangular analysis window  $w_k$  is used, we need to weight the output of the inverse transform by  $1/w_k$  for a critically sampled filterbank. In Fig. C.5 simulation results for the critically sampled 16-channel filterbank described above can be seen. The delay  $\Delta$  is 15 samples or equivalently 1.875 ms for 8 kHz sampling rate.

As a drawback, the frequency responses of the synthesis filters  $B_k(z)$  do not

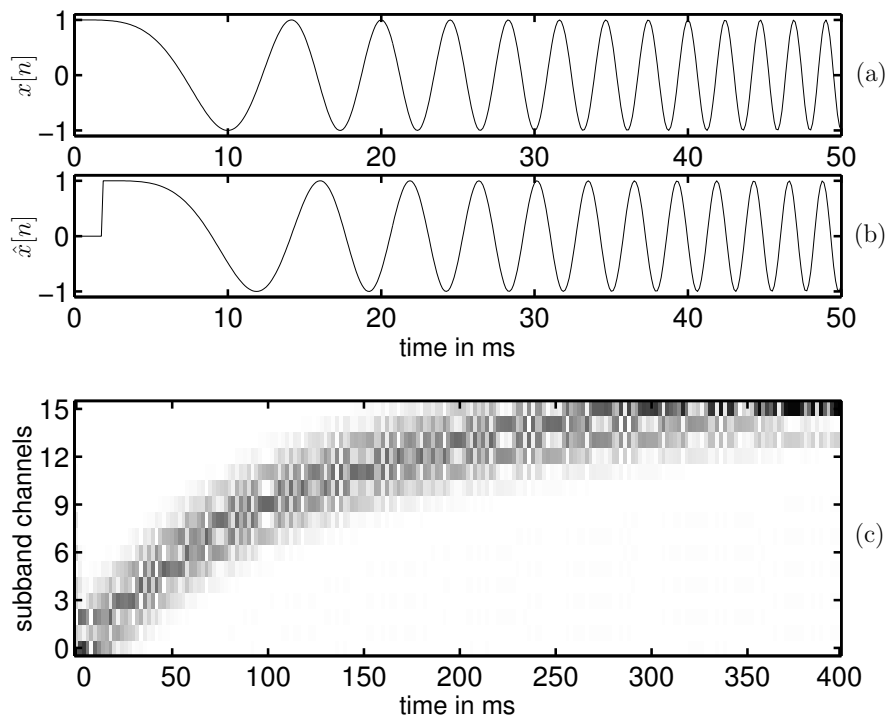


Figure C.5: Experiment with the 16-channel filterbank of Fig. C.4 and a chirped cosine with frequency linearly increasing from 0 Hz to 4 kHz in 400 ms. Sampling rate is 8 kHz. (a) First 50 ms of the input signal  $x[n]$ . (b) First 50 ms of the resynthesized signal  $\hat{x}[n]$ . (c) Entire 400 ms of the maximally decimated subband signals  $|Y_k[m]|$  as grayscale image.

have band-pass characteristics. Therefore, quantization noise injected in a sub-band channel does not only affect a local frequency area, but is rather spread all over the spectrum. Moreover, the noise gain of the filters, i.e., the sum of the squared impulse response samples, is unusually high and grows exponentially with an increasing decimation factor  $N$ . Therefore, these synthesis filters are not suitable for coding applications yet.

## C.4 Conclusions

We have shown that an exact and stable solution for the synthesis filters of a critically sampled or oversampled frequency-warped transform filterbank exists and achieves perfect reconstruction. Furthermore, we have proved that for first-order warping all-pass filters these synthesis filters are FIR filters, and we have derived a closed-form solution. However, these synthesis filters amplify any quantization noise introduced in the subband signals and do not exhibit band-pass characteristics. Since the perfect-reconstruction solution for an oversampled filterbank is not unique, further investigations to improve the band-pass characteristics and to reduce the noise gain are necessary to make this filterbank suitable for coding applications.



# Bibliography

- [1] A. M. Kondo, *Digital Speech Coding for Low Bit Rate Communication Systems*. Chichester, England: John Wiley & Sons Ltd, 1994.
- [2] R. Veldhuis and A. Kohlrausch, “Waveform coding and auditory masking,” in *Speech Coding and Synthesis*, W. B. Kleijn and K. K. Paliwal, Eds. Elsevier Science, 1995.
- [3] D. T. Yang, C. Kyriakakis, and C. J. Kuo, *High-Fidelity Multichannel Audio Coding*. Cairo, Egypt: Hindawi Publishing Corporation, 2004.
- [4] N. Jayant, J. Johnston, and R. Safranek, “Signal compression based on models of human perception,” *Proceedings of the IEEE*, vol. 81, no. 10, pp. 1385–1422, Oct. 1993.
- [5] J. D. Johnston, “Transform coding of audio signals using perceptual noise criteria,” *IEEE Journal on Selected Areas in Communications*, vol. 6, no. 2, pp. 314–323, Feb. 1988.
- [6] K. Brandenburg and G. Stoll, “ISO-MPEG-1 Audio: A generic standard for coding of high-quality digital audio,” *Journal of the Audio Engineering Society*, vol. 42, no. 10, pp. 780–792, Oct. 1994.
- [7] P. Ekstrand, “Bandwidth extension of audio signals by spectral band replication,” in *Proceedings of the IEEE Benelux Workshop on Model based Processing and Coding of Audio, MPCA’02*, Leuven, Belgium, Nov. 2002, pp. 53–58.
- [8] T. Ziegler, M. Dietz, J. Röden, S. Meltzer, and A. Ehret, “aacPlus—Highest efficient audio coding for broadcast applications,” in *Proceedings of the NAB 2003 Broadcast Engineering Conference*, Las Vegas, Apr. 2003, pp. 293–299.
- [9] T. Painter and A. Spanias, “Perceptual coding of digital audio,” *Proceedings of the IEEE*, vol. 88, no. 4, pp. 451–515, Apr. 2000.

- [10] *Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s—Part 3: Audio*, International Standard ISO/IEC 11172-3:1993, 1993.
- [11] J. D. Johnston, “Estimation of perceptual entropy using noise masking criteria,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP’88*, vol. 5, New York, Apr. 1988, pp. 2524–2527.
- [12] B. Tang, A. Shen, A. Alwan, and G. Pottie, “A perceptually based embedded subband coder,” *IEEE Transactions on Speech and Audio Processing*, vol. 5, no. 2, pp. 131–140, Mar. 1997.
- [13] B. S. Atal and M. R. Schroeder, “Predictive coding of speech signals and subjective error criteria,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 27, no. 3, pp. 247–254, June 1979.
- [14] R. Geiger, J. Herre, G. Schuller, and T. Sporer, “Fine grain scalable perceptual and lossless audio coding based on INTMDCT,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP’03*, vol. 5, 2003, pp. 445–448.
- [15] F. Baumgarte, “Improved audio coding using a psychoacoustic model based on a cochlear filter bank,” *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 7, pp. 495–503, Oct. 2002.
- [16] R. Der, P. Kabal, and W.-Y. Chan, “Towards a new perceptual coding paradigm for audio signals,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP’03*, vol. 5, Hong Kong, Apr. 2003, pp. 457–460.
- [17] M. Hansen and B. Kollmeier, “Using a quantitative psychoacoustical signal representation for objective speech quality measurement,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP’97*, vol. 2, Apr. 1997, pp. 1387–1390.
- [18] E. Zwicker, “Dependence of post-masking on masker duration and its relation to temporal effects in loudness,” *Journal of the Acoustical Society of America*, vol. 75, no. 1, pp. 219–223, Jan. 1984.
- [19] T. Dau, D. Püschel, and A. Kohlrausch, “A quantitative model of the “effective” signal processing in the auditory system. I. Model structure,” *Journal of the Acoustical Society of America*, vol. 99, no. 6, pp. 3615–3622, June 1996.

- [20] H. Su and P. Mermelstein, "Delayed decision coding of pitch and innovation signals in code-excited linear prediction coding of speech," in *Speech and audio coding for wireless and network applications*, B. S. Atal, V. Cuperman, and A. Gersho, Eds. Boston: Kluwer Academic Publishers, 1993, pp. 69–76.
- [21] R. Fandos Marin, "Delayed decision CELP speech coding using squared and perceptual error criteria," Master's thesis, KTH (Royal Inst. of Technology), Dept. of Signals, Sensors and Systems, July 2003.
- [22] J. H. Plasberg, D. Y. Zhao, and W. B. Kleijn, "The sensitivity matrix for a spectro-temporal auditory model," in *Proceedings of the European Signal Processing Conference, EUSIPCO'04*, Vienna, Austria, Sept. 2004, pp. 1673–1676.
- [23] J. H. Plasberg and W. B. Kleijn, "The sensitivity matrix: Using advanced auditory models in speech and audio processing," *IEEE Transactions on Speech and Audio Processing*, 2005, submitted for publication.
- [24] W. Gardner and B. Rao, "Theoretical analysis of the high-rate vector quantization of LPC parameters," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 5, pp. 367–381, Sept. 1995.
- [25] R. Der, P. Kabal, and W.-Y. Chan, "Rate-distortion allocation for time-frequency dependent audio coding," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'05*, vol. 3, Philadelphia, Pennsylvania, Mar. 2005, pp. 197–200.
- [26] G. Kubin and W. B. Kleijn, "On speech coding in a perceptual domain," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'99*, vol. 1, Phoenix, Arizona, Mar. 1999, pp. 205–208.
- [27] J. B. Allen, "Cochlear modeling," *IEEE Acoustics, Speech and Signal Processing (ASSP) Magazine*, vol. 2, pp. 3–29, Jan. 1985.
- [28] R. D. Patterson, "A pulse ribbon model of monaural phase perception," *Journal of the Acoustical Society of America*, vol. 82, no. 5, pp. 1560–1586, Nov. 1987.
- [29] J. B. Allen, "Nonlinear cochlear signal processing," in *Physiology Of The Ear*, A. John and J. Santos-Sacchi, Eds. Singular Thomson Learning, 2001, pp. 393–442.

- [30] R. Patterson, K. Robinson, J. Holdsworth, D. McKeown, C. Zhang, and M. Allerhand, “Complex sounds and auditory images,” in *Auditory Physiology and Perception*, Y. Cazals, L. Demany, and K. Horner, Eds. Oxford: Pergamon Press, 1992, pp. 429–446.
- [31] S. Greenberg, “Acoustic transduction in the auditory periphery,” *Journal of Phonetics*, vol. 16, pp. 3–17, 1988.
- [32] M. A. Ruggero, “Physiology and coding of sound in the auditory nerve,” in *The Mammalian Auditory Pathway: Neurophysiology*, A. Popper and R. Fay, Eds. New York: Springer-Verlag, 1992, pp. 34–93.
- [33] E. Zwicker and H. Fastl, *Psychoacoustics. Facts and Models*, 2nd ed. Springer-Verlag, 1999.
- [34] B. R. Glasberg and B. C. Moore, “Derivation of auditory filter shapes from notched-noise data,” *Hearing Research*, vol. 47, pp. 103–138, 1990.
- [35] B. C. J. Moore, *An Introduction to the Psychology of Hearing*, 4th ed. London: Academic Press, 1997.
- [36] E. Zwicker and E. Terhardt, “Analytical expressions for critical-band rate and critical bandwidth as a function of frequency,” *Journal of the Acoustical Society of America*, vol. 68, no. 5, pp. 1523–1525, Nov. 1980.
- [37] D. Greenwood, “A cochlear frequency-position function for several species—29 years later,” *Journal of the Acoustical Society of America*, vol. 87, no. 6, pp. 2592–2605, June 1990.
- [38] H. Traunmüller, “Analytical expressions for the tonotopic sensory scale,” *Journal of the Acoustical Society of America*, vol. 88, no. 1, pp. 97–100, 1990.
- [39] A. Härmä, M. Karjalainen, L. Savioja, V. Välimäki, U. Laine, and J. Huopaniemi, “Frequency-warped signal processing for audio applications,” *Journal of the Audio Engineering Society*, vol. 48, no. 11, pp. 1011–1029, Nov. 2000.
- [40] R. Patterson, I. Nimmo-Smith, D. Weber, and R. Milroy, “The deterioration of hearing with age: Frequency selectivity, the critical ratio, the audiogram, and speech threshold,” *Journal of the Acoustical Society of America*, vol. 72, no. 6, pp. 1788–1803, Dec. 1982.

- [41] P. Dallos, “Overview: Cochlear neurobiology,” in *The Cochlea*, P. Dallos, A. Popper, and R. Fay, Eds. New York: Springer Verlag, 1996, vol. 8, pp. 1–43.
- [42] T. Irino and R. D. Patterson, “A time-domain, level-dependent auditory filter: The gammachirp,” *Journal of the Acoustical Society of America*, vol. 101, no. 1, pp. 412–419, Jan. 1997.
- [43] J. M. Kates, “Two-tone suppression in a cochlear model,” *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 5, pp. 396–406, Sept. 1995.
- [44] T. Irino and R. D. Patterson, “A dynamic, compressive gammachirp auditory filterbank,” *IEEE Transactions on Speech and Audio Processing*, 2005, submitted for publication.
- [45] P. Sellick, R. Patuzzi, and B. Johnstone, “Measurement of basilar membrane motion in the guinea pig using the Mössbauer technique,” *Journal of the Acoustical Society of America*, vol. 72, no. 1, pp. 131–141, July 1982.
- [46] R. F. Lyon, “A computational model of filtering, detection, and compression in the cochlea,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP’82*, Paris, France, May 1982, pp. 1282–1285.
- [47] S. Seneff, “A joint synchrony/mean-rate model of auditory speech processing,” *Journal of Phonetics*, vol. 16, pp. 55–76, 1988.
- [48] F. Baumgarte, “A psychoacoustic model for audio coding based on a cochlear filter bank,” in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, New York, Oct. 2001, pp. 139–142.
- [49] R. Meddis, “Simulation of mechanical to neural transduction in the auditory receptor,” *Journal of the Acoustical Society of America*, vol. 79, no. 3, pp. 702–711, 1986.
- [50] ———, “Simulation of auditory–neural transduction: Further studies,” *Journal of the Acoustical Society of America*, vol. 83, no. 3, pp. 1056–1063, Mar. 1988.
- [51] K. Johnson, S. Hsiao, and T. Yoshioka, “Neural coding and the basic law of psychophysics,” *The Neuroscientist*, vol. 8, no. 2, pp. 111–121, 2002.
- [52] F.-G. Zeng and R. V. Shannon, “Psychophysical laws revealed by electric hearing,” *NeuroReport*, vol. 10, no. 9, pp. 1931–1935, June 1999.

- [53] ———, “Loudness-coding mechanisms inferred from electric stimulation of the human auditory system,” *Science*, vol. 264, no. 5158, pp. 564–566, Apr. 1994.
- [54] P. Vary, U. Heute, and W. Hess, *Digitale Sprachsignalverarbeitung*. Stuttgart, Germany: B.G. Teubner, 1998.
- [55] A. McCree, “A 14 kb/s wideband speech coder with a parametric highband model,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP’00*, vol. 2, Istanbul, Turkey, June 2000, pp. 1153–1156.
- [56] J.-M. Valin and R. Lefebvre, “Bandwidth extension of narrowband speech for low bit-rate wideband coding,” in *Proceedings of the IEEE Workshop on Speech Coding2000*, Delavan, WI, Sept. 2000, pp. 130–132.
- [57] P. Jax and P. Vary, “Feature selection for improved bandwidth extension of speech signals,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP’04*, vol. 1, Montreal, Canada, May 2004, pp. 697–700.
- [58] W. Maass and C. M. Bishop, Eds., *Pulsed neural networks*. MIT Press, 1999.
- [59] M. Weintraub, “The GRASP sound separation system,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP’84*, vol. 9. IEEE, Mar. 1984, pp. 69–72.
- [60] ———, “A computational model for separating two simultaneous talkers,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP’86*, vol. 11. IEEE, Apr. 1986, pp. 81–84.
- [61] M. Slaney, “Pattern playback from 1950 to 1995,” in *Proceedings of the IEEE Systems, Man, and Cybernetics Conference*, vol. 4, Vancouver, Canada, Oct. 1995, pp. 3519–3524.
- [62] F. Cooper, “Acoustics in human communication: Evolving ideas about the nature of speech,” *Journal of the Acoustical Society of America*, vol. 68, no. 1, pp. 18–21, 1980.
- [63] T. Irino and H. Kawahara, “Signal reconstruction from modified auditory wavelet transform,” *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3549–3554, Dec. 1993.

- [64] M. Slaney, D. Naar, and R. F. Lyon, "Auditory model inversion for sound separation," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'94*, vol. 2, Adelaide, Australia, Apr. 1994, pp. 77–80.
- [65] R. Hukin and R. Damber, "Testing an auditory model by resynthesis," in *Proceedings of the European Conference on Speech Communication and Technology, EUROSPEECH'89*, vol. 1, Paris, France, Sept. 1989, pp. 243–246.
- [66] X. Yang, K. Wang, and S. A. Shamma, "Auditory representations of acoustic signals," *IEEE Transactions on Information Theory*, vol. 38, no. 2, pp. 824–839, Mar. 1992.
- [67] G. Kubin and W. B. Kleijn, "Multiple-description coding (MDC) of speech with an invertible auditory model," in *Proceedings of the IEEE Workshop on Speech Coding*, Porvoo, Finland, June 1999, pp. 81–83.
- [68] S. Mallat and S. Zhong, "Characterization of signals from multiscale edges," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 7, pp. 710–732, July 1992.
- [69] C. Feldbauer, G. Kubin, and W. B. Kleijn, "Anthropomorphic coding of speech and audio: A model inversion approach," *EURASIP Journal on Applied Signal Processing*, vol. 2005, no. 9, pp. 1334–1349, June 2005.
- [70] 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'01*, vol. 2, Salt Lake City, Utah, May 2001, pp. 773–776.
- [71] L. Lin, E. Ambikairajah, and W. Holmes, "Perceptual domain based speech and audio coder," in *Proceedings of the sixth International Symposium on Digital Signal Processing for Communication Systems, DSPCS'02*, Sydney, Australia, Jan. 2002, pp. 6–11.
- [72] I. Daubechies, "The wavelet transform, time-frequency localization and signal analysis," *IEEE Transactions on Information Theory*, vol. 36, no. 5, pp. 961–1005, Sept. 1990.
- [73] H. Bölcskei, "Oversampled filter banks and predictive subband coders," Ph.D. dissertation, Vienna University of Technology, Austria, 1997.
- [74] H. Bölcskei, 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.

- [75] H. N. Razafinjatovo, “Iterative reconstructions in irregular sampling with derivatives,” *Journ. Fourier Anal. Appl.*, vol. 1, no. 3, pp. 281–295, 1995.
- [76] B. Foster and C. Herley, “Exact reconstruction from periodic nonuniform samples,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP’1995*, vol. 2, Detroit, MI, May 1995, pp. 1452–1455.
- [77] M. Stocker, “Efficient coding methods for a perceptual speech coder,” Master’s thesis, Graz University of Technology, Institute of Communications and Wave Propagation, June 2003.
- [78] E. Zwicker and U. Zwicker, “Audio engineering and psychoacoustics: Matching signals to the final receiver, the human auditory system,” *Journal of the Audio Engineering Society*, vol. 39, no. 3, pp. 115–126, Mar. 1991.
- [79] C. Feldbauer and G. Kubin, “How sparse can we make the auditory representation of speech?” in *Proceedings of the International Conference on Spoken Language Processing, ICSLP’04*, Korea, Oct. 2004.
- [80] T. Dau, B. Kollmeier, and A. Kohlrausch, “Modeling auditory processing of amplitude modulation. I. Detection and masking with narrow-band carriers,” *Journal of the Acoustical Society of America*, vol. 102, no. 5, pp. 2892–2905, Nov. 1997.
- [81] B. R. Glasberg, B. C. Moore, and R. W. Peters, “The influence of external and internal noise on the detection of increments and decrements in the level of sinusoids,” *Hearing Research*, vol. 155, pp. 41–53, 2001.
- [82] J. R. Barry, E. A. Lee, and D. G. Messerschmitt, *Digital Communication*, 3rd ed. Boston: Kluwer Academic Publishers, 2004.
- [83] S. Singhal and B. Atal, “Amplitude optimization and pitch prediction in multipulse coders,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 3, pp. 317–327, Mar. 1989.
- [84] J. Egan and H. Hake, “On the masking pattern of a simple auditory stimulus,” *Journal of the Acoustical Society of America*, vol. 22, pp. 622–630, 1950.
- [85] M. van der Heijden, “A comparison of masking by tones and noise,” Ph.D. dissertation, Technische Universiteit Eindhoven, 1995.

- [86] B. C. J. Moore and B. R. Glasberg, “Suggested formulae for calculating auditory-filter bandwidths and excitation patterns,” *Journal of the Acoustical Society of America*, vol. 74, no. 3, pp. 750–753, Sept. 1983.
- [87] *Methods for subjective determination of transmission quality*, ITU-T Recommendation P.800, Rev. 08/96, 1996.
- [88] P. Maragos, J. F. Kaiser, and T. F. Quatieri, “Energy separation in signal modulations with application to speech analysis,” *IEEE Transactions on Signal Processing*, vol. 41, no. 10, pp. 3024–3051, Oct. 1993.
- [89] T. Dau, D. Püschel, and A. Kohlrausch, “A quantitative model of the “effective” signal processing in the auditory system. II. Simulations and measurements,” *Journal of the Acoustical Society of America*, vol. 99, no. 6, pp. 3623–3631, June 1996.
- [90] T. Dau, B. Kollmeier, and A. Kohlrausch, “Modeling auditory processing of amplitude modulation. II. Spectral and temporal integration,” *Journal of the Acoustical Society of America*, vol. 102, no. 5, pp. 2906–2919, Nov. 1997.
- [91] R. E. Blahut, “Computation of channel capacity and rate distortion functions,” *IEEE Transactions on Information Theory*, vol. 18, no. 4, pp. 410–421, July 1972.
- [92] R. Castro, M. Wakin, and M. Orchard, “On the problem of simultaneous encoding of magnitude and location information,” in *Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, California, Nov. 2002, pp. 1–5.
- [93] M. Krasner, “The critical band coder—digital encoding of speech signals based on the perceptual requirements of the auditory system,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP’80*, vol. 5, Denver, Colorado, Apr. 1980, pp. 327–331.
- [94] C. Weidmann and M. Vetterli, “Rate-distortion analysis of spike processes,” in *Proceedings of the Data Compression Conference, DCC’99*, Mar. 1999, pp. 82–91.
- [95] ———, “Rate distortion behavior of sparse sources,” *Subm. IEEE Transaction on Information Theory*, Oct. 2001.
- [96] M. Slaney, “Lyon’s cochlea model,” Apple Computer, Inc, Tech. Rep. 13, 1988.

- [97] F. Baumgarte, “A computationally efficient cochlear filter bank for perceptual audio coding,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP’01*, vol. 5, Salt Lake City, Utah, May 2001, pp. 3265–3268.
- [98] M. Slaney, “An efficient implementation of the Patterson-Holdsworth auditory filter bank,” Apple Computer, Inc., Tech. Rep. 35, 1993.
- [99] A. Oppenheim, R. Schaffer, and J. Buck, *Discrete-Time Signal Processing*, 2nd ed. Upper Saddle River, New Jersey: Prentice Hall, Inc., 1999.
- [100] S. Mitra, C. Creusere, and H. Babic, “A novel implementation of perfect reconstruction QMF banks using IIR filters for infinite length signals,” in *Proceedings of the IEEE International Symposium on Circuits and Systems, ISCAS’92*, San Diego, May 1992, pp. 2312–2315.
- [101] L. Lin, W. Holmes, and E. Ambikairajah, “Auditory filter bank inversion,” in *Proceedings of the IEEE International Symposium on Circuits and Systems, ISCAS’01*, vol. 2, Sydney, Australia, May 2001, pp. 537–540.
- [102] A. Oppenheim, D. Johnson, and K. Steiglitz, “Computation of spectra with unequal resolution using the fast Fourier transform,” in *Proceedings of the IEEE*, vol. 59, Feb. 1971, pp. 299–301.
- [103] C. Braccini and A. Oppenheim, “Unequal bandwidth spectral analysis using digital frequency warping,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 22, no. 4, pp. 236–244, Aug. 1974.
- [104] P. Vary, “Ein Beitrag zur Kurzzeitspektralanalyse mit digitalen Systemen,” in *Ausgewählte Arbeiten über Nachrichtensysteme*, ser. 32, H. Schüßler, Ed. Universität Erlangen, Germany, 1978.
- [105] T. Gülzow, A. Engelsberg, and U. Heute, “Comparison of a discrete wavelet transformation and a nonuniform polyphase filterbank applied to spectral-subtraction speech enhancement,” *Signal Processing*, vol. 64, no. 1, pp. 5–19, Jan. 1998.
- [106] E. Galijašević, “Design of allpass-based non-uniform oversampled DFT filter banks,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP’02*, vol. 2, Orlando, Florida, May 2002, pp. 1181–1184.
- [107] J. Smith and J. Abel, “Bark and ERB bilinear transforms,” *IEEE Transactions on Speech and Audio Processing*, vol. 7, no. 6, pp. 697–708, Nov. 1999.

- [108] P. Duhamel, Y. Mahieux, and J. Petit, “A fast algorithm for the implementation of filter banks based on “Time Domain Aliasing Cancellation”,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'91*, vol. 3, Toronto, Ont., Apr. 1991, pp. 2209–2212.
- [109] E. Galijašević and J. Kliever, “Non-uniform near-perfect-reconstruction oversampled DFT filter banks based on allpass-transforms,” in *Proceedings of the Ninth IEEE DSP Workshop*, Hunt, Texas, Oct. 2000, pp. 1–6.
- [110] M. Parfieniuk and A. Petrovsky, “Reduced complexity synthesis part of non-uniform near-perfect-reconstruction DFT filter bank based on all-pass transformation,” in *Proceedings of the European Conference on Circuit Theory and Design, ECCTD'03*, vol. 3, Krakow, Poland, Sept. 2003, pp. 5–8.
- [111] C. Feldbauer and G. Kubin, “Critically sampled frequency-warped perfect reconstruction filterbank,” in *Proceedings of the European Conference on Circuit Theory and Design, ECCTD'03*, vol. 3, Krakow, Poland, Sept. 2003, pp. 109–112.
- [112] M. Pflüger, “Modelle des peripheren Gehörs am Beispiel der menschlichen Lautheitsempfindung,” Ph.D. dissertation, Graz University of Technology, Austria, Sept. 1997.
- [113] T. Dau, “Modeling auditory processing of amplitude modulation,” Ph.D. dissertation, University Oldenburg, 1996.
- [114] S. Münkner, “Modellentwicklung und Messungen zur Wahrnehmung nicht-stationärer Signale,” Ph.D. dissertation, University of Göttingen, 1993.
- [115] U. Laine, “Block-recursive, multirate filterbanks with arbitrary time-frequency plane tiling,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'99*, vol. 3, Phoenix, Arizona, Mar. 1999, pp. 1461–1464.
- [116] G. Evangelista and S. Cavaliere, “Frequency-warped filter banks and wavelet transforms: A discrete-time approach via laguerre expansion,” *IEEE Transactions on Signal Processing*, vol. 46, no. 10, pp. 2638–2650, Oct. 1998.
- [117] A. Makur and S. Mitra, “Warped discrete-Fourier transform: Theory and applications,” *IEEE Trans. Circuits Systems I: Fund. Theory Applic.*, vol. 48, no. 9, pp. 1086–1093, Sept. 2001.

- [118] B. Shankar and A. Makur, "Allpass delay chain-based IIR PR filterbank and its application to multiple description subband coding," *IEEE Transactions on Signal Processing*, vol. 50, no. 4, Apr. 2002.
- [119] P. Vaidyanathan, *Multirate Systems and Filter Banks*. Englewood Cliffs, New Jersey: Prentice Hall, Inc., 1993.
- [120] T. Moon and W. Stirling, *Mathematical Methods and Algorithms for Signal Processing*. Upper Saddle River, New Jersey: Prentice Hall, Inc., 2000.