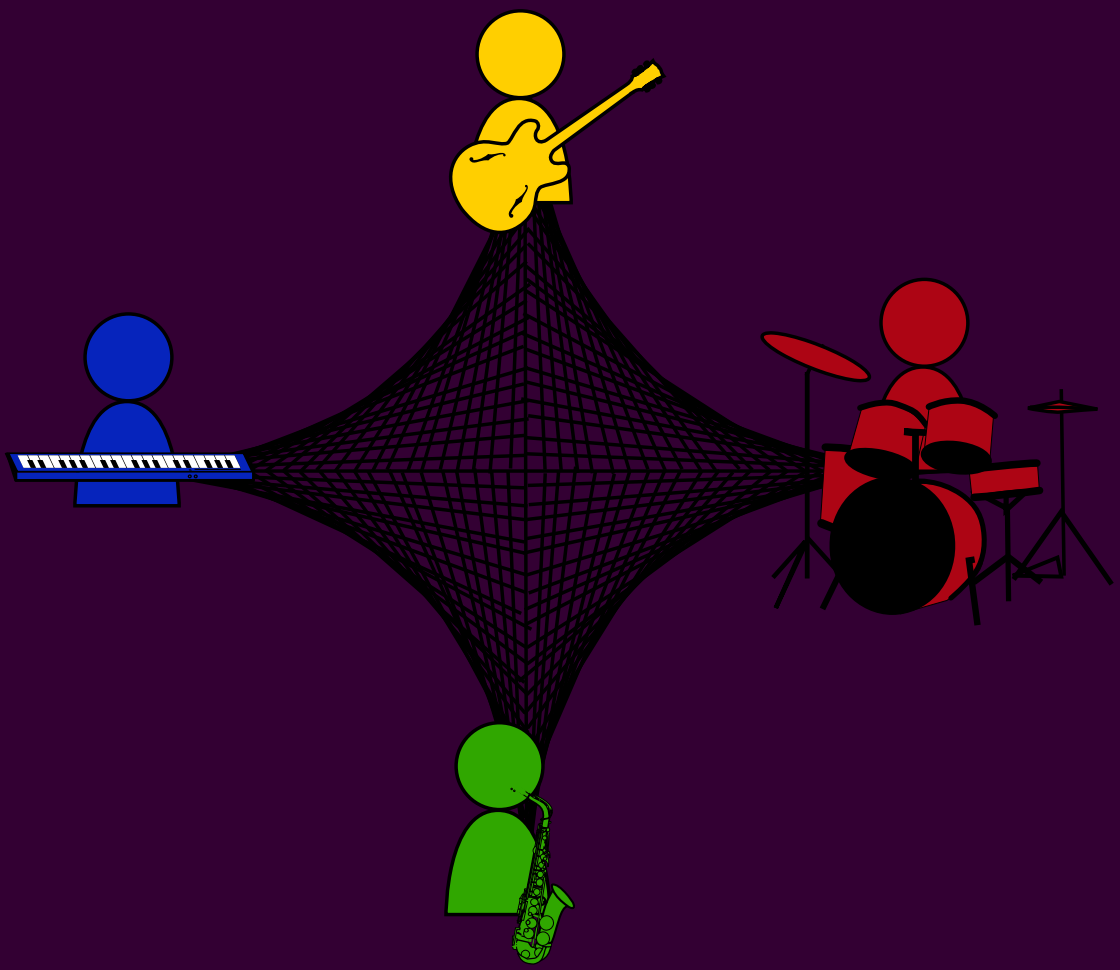
Marco Fink

Enhancements for Networked Music Performances



Enhancements for Networked Music Performances

Von der Fakultät für Elektrotechnik

der Helmut-Schmidt-Universität/Universität der Bundeswehr Hamburg

zur Erlangung des akademischen Grades eines Doktor-Ingenieurs

genehmigte

DISSERTATION

vorgelegt von

Dipl.-Ing. Marco Fink

aus

Nürnberg

Berlin 2018

Table of Contents

Abstract

The availability and capability of today's internet allow several novel challenging interactive multimedia applications like *Networked Music Performances* (NMP). A Networked Music Performance is an online artistic collaboration with musicians located at different geographic locations connected using the internet. While offering manifold artistic possibilities, many technical challenges like the resulting latency and the possibility of packet loss have to be considered. This work depicts three enhancements for NMP applications which improve error robustness, the algorithmic delay, and the spatial listening experience, respectively.

To counteract the possibility of packet loss or network jitter-caused tardy arrival of packets, this work derives two methods to conceal errors during audio replay at the receiver side. The first, auto-regressive model-based variant facilitates concealing the audible impact of missing packets with high quality but is computationally expensive. Several ways of computing the auto-regressive model are presented and compared. The second method, based on wave-form substitution, constitutes an efficient, cheap alternative. The proposed methods are evaluated subjectively with a listening test and objectively with measurements of perceptual quality.

The application of audio codecs in NMP sessions is inevitable in most scenarios due to the restricted data rate and in particular the upload rate of private internet accesses. Besides reducing the data rate the codec must feature a small algorithmic latency to restrict the overall latency to a certain extent. A novel audio coding approach which features smaller delays than widely used low-delay codecs and a clearly reduced data rate in contrast to delay-less codecs is presented. It is constructed using the *Adaptive Differential Pulse Code Modulation* (ADPCM) codec approach in subbands in

combination with a *Vector Quantizer* (VQ) resulting in the *Vector-Quantized Adaptive Differential Pulse Code Modulation* (VQ-ADPCM) codec. The proposed codec is capable of encoding broadband audio with a data rate of 64 kbit/s and algorithmic delay of about 1 ms. The perceptual quality is compared to well-known codecs using perceptually motivated measurements.

The last contribution is intended to improve the acoustic spatial scenery within a NMP. For this purpose, a pseudo stereo conversion method providing a broad stereo panorama for single channel sound sources is derived. The method enhances the spaciousness of the stereo mix at the receiver without adding timbral coloration or reverberation and therefore offers an improved listening experience for NMP participants. The proposed method is based on the design of a complementary filter pair, which can be applied in time- and frequency-domain. Additionally, the integration within a virtual surround mixer based on *Head-Related Impulse Responses* (HRIRs) is demonstrated. Virtual surround mixing allows the arbitrary positioning of several sound sources in a virtual room. The extension with the proposed pseudo-stereo approach even facilitates to define sound sources of a certain size instead of single point sources.

The three proposed enhancements are purely based on digital signal processing and therefore can be implemented in the software layer of any NMP system without demanding any changes to the actual musical performance, the utilized hardware, or the available network structure.

Kurzfassung

Die weitreichende Verfügbarkeit und Leistungsfähigkeit des heutigen Internets erlaubt einige sowohl neuartige als auch herausfordernde interaktive Multimediaanwendungen wie die sogenannte *Networked Music Performance* (NMP). Eine NMP beschreibt eine künstlerische Online-Kollaboration von Musikern, die räumlich getrennt, aber durch das Internet verbunden sind. Dieser Ansatz erlaubt vielfältige künstlerisch Möglichkeiten. Allerdings müssen auch viele technische Schwierigkeiten, wie die Übertragungslatenz und die Möglichkeit von Paketverlusten, in Betracht gezogen werden. Diese Arbeit zeigt drei Erweiterungen für NMP-Anwendungen auf, welche jeweils die Fehlerrobustheit, die Übertragungslatenz und das räumliche Hörerlebnis aufwerten.

Um der Möglichkeit eines Paketverlustes oder dem Netzwerk-Jitter-geschuldetem, verspätetem Eintreffen von Paketen entgegen zu wirken, werden zunächst Methoden vorgestellt, die es erlauben Fehler in der Wiedergabe am Empfänger zu verschleiern. Die erste Variante, basierend auf einem autoregressiven Modell, ermöglicht das Verbergen von hörbaren Beeinträchtigungen durch fehlende Pakete bei hoher Qualität, ist allerdings in der Berechnung sehr aufwendig. Mehrere Ansätze um autoregressive Signalmodelle zu ermitteln werden hierbei aufgezeigt und verglichen. Die zweite Methode, basierend auf der Substitution von Wellenformen, stellt eine bezüglich des Rechenaufwands günstigere Alternative dar. Die Methoden werden mit Hilfe eine Hörtests subjektiv und durch Messungen der wahrgenommenen Qualität objektiv beurteilt.

Die Anwendung von Audiokompressionsverfahren in einer NMP-Session ist in den meisten Szenarien unvermeidlich, da die Datenrate und insbesondere die Uploadrate von privaten Internetzugängen beschränkt ist. Neben der

Reduktion der Datenrate muss der verwendete Audio Codec eine möglichst geringe Latenz aufweisen, um die Gesamtübertragungslatenz einzuschränken. Ein neuartiger Audiokodierungsansatz, der geringere Latenzen als weit verbreitete Niedriglatenz-Codecs und dennoch kleinere Datenraten als latenzfreie Codecs aufweist, wird vorgestellt. Das Kompressionsverfahren basiert auf der Anwendung der *Adaptiven Differentiellen Pulse Code Modulation* (ADPCM) in Teilbändern. In Kombination mit einem *Vektorquantisierer* (VQ) resultiert der *Vektorquantisierte Adaptive Differentielle Pulse Code Modulation* (VQ-ADPCM)-Codec. Der vorgestellte Codec ist imstande breitbandige Audiosignale mit 64 kbit/s und einer algorithmischen Latenz von 1 ms zu enkodieren. Die wahrgenommene Qualität wird mit wohl-bekannten Codecs anhand psychoakustisch-motivierter Messverfahren verglichen.

Der letzte Beitrag ist vorgesehen, die räumliche akustische Szenerie innerhalb einer NMP zu verbessern. Hierfür wird ein Pseudo-Stereo-Verfahren, welches breite Stereopanoramen für einkanalige Klangquellen liefert, hergeleitet. Die Methode verbessert die Räumlichkeit des Stereomixes beim Empfänger ohne das Nutzsignal zu verfärben oder zu verhallen. Dadurch wird dem NMP-Nutzer ein verbessertes Hörerlebnis geboten. Der vorgestellte Ansatz basiert auf dem Entwurf eines komplementären Filterpaares, welches in Zeit- und Frequenzbereich angewendet werden kann. Zusätzlich ist die Integration des Ansatzes in einem virtuellen Surround-Mischer basierend auf kopfbezogenen Impulsantworten (HRIR) veranschaulicht. Virtuelles Surround-Mischen erlaubt das beliebige Platzieren von Klangquellen in einem virtuellen Raum. Die Erweiterung mit der präsentierten Pseudo-Stereo-Methode ermöglicht nun sogar das Platzieren von Quellen verschieden Größe anstatt von Punktquellen.

Die drei vorgestellten NMP-Verbesserungen basieren ausschließlich auf digitaler Signalverarbeitung und können deshalb in der Softwareschicht jeglichen NMP-Systems realisiert werden, ohne dabei Adaptionen der musikalischen Darbietung, der genutzten Hardware oder der verfügbaren Netzwerkstruktur zu erfordern.

1 Introduction

Virtual telepresence is a broadly known phenomenon that allows nearly instant communication and interaction of several individuals without the restriction of being jointly localized. This presence can be based on pure text (E-Mail, SMS, Chat, Instant Messaging), audio (Broadcast, Telephony, Audio Instant Messaging), and video (Video Conferencing, Video Telephony). Most of these communication schemes are widely accepted by the public and therefore, extensively present in our daily life. Additionally, these telepresence methods share the property of being realized with a network infrastructure since classical analog broadcast and telephony system are more and more replaced by their digital IP-based successors. The utilization of a packet-based comunication network even facilitates novel and fascinating applications like *Networked Music Performances* (NMP).

A NMP describes the online musical interaction of several musicians by connecting their instruments to internet-linked devices and utilizing specific software for the task of sending and receiving audio streams to and from other participants. A simplified illustration of a NMP session with three clients A, B, and C is shown in Fig. 1.1. All clients feature a sending and receiving module, respectively. The analog signal of whatever instrument has to be digitized with an Analog-Digital-Converter (A/D), optionally encoded to reduce the data rate of the upload stream, and embedded within a transport protocol to be sent over the internet to all other receivers within the sending module. The receiver module is responsible for extracting the audio data from the packets, optionally decode it, buffer the data, and to render the received streams to the chosen replay format. The last step to replay the data is to apply Digital-Analog-Converter (D/A) to recreate analog signals for feeding it to speakers or headphones. This simplified NMP model is not respecting the synchronization of the musicians and the replay hardware or

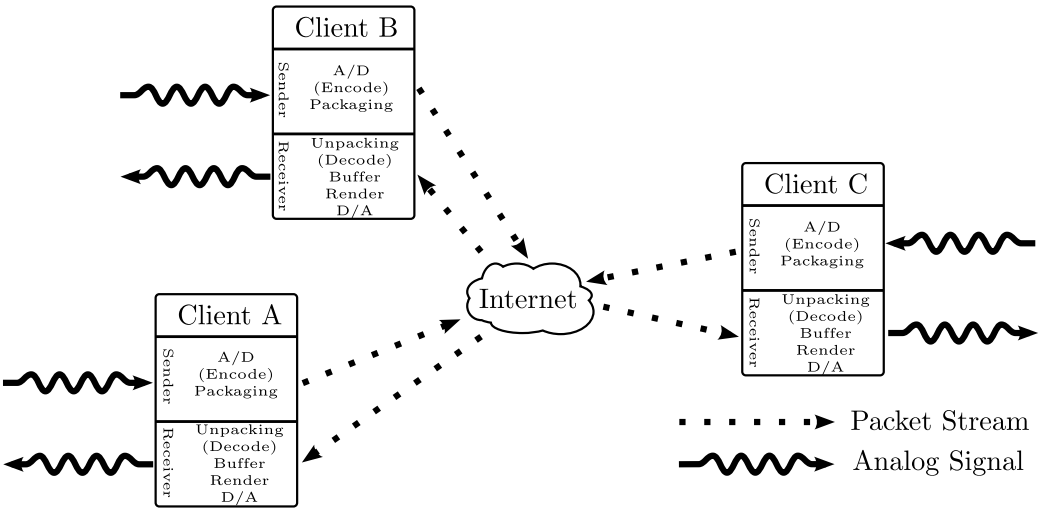


Figure 1.1: Simplified NMP session with 3 participants.

any network specific adaptions since this work is focused on the digital audio signal processing aspects of the described NMP scenario. Furthermore, it shall be noted that the considered NMP scenario takes place in the public *Wide Area Network* (WAN), denoted as the internet. The performance of a NMP session within a private WAN or *Local Area Network* (LAN) would be significantly improved and potentially not require all enhancements proposed in this work.

Several aspects and problems of NMPs were already accurately analyzed. The most significant drawback of NMP sessions is the resulting delay between a sound source and the replay device. Amongst others, the following processing steps significantly contribute to the overall delay [Car09]: A/D conversion, blocking, encoding, packetization, sending, routing, cable propagation, receiving, decoding, buffering, D/A conversion. It is well-known how delay in verbal telecommunication degrades the quality of a conversation since one starts to interrupt other speakers. Musical interaction is drastically more sensitive to delay. Several studies [CGLT04, CSZ+04, CWT09] experimentally determined an upper bound which is acceptable for musicians to perform. Depending on the musical style, values in the range of 30 to 50 ms were measured. Different studies involve prediction of a participants performance to synthesize similar acoustic events at the receiver side [OFF, VC14]. Thereby, the transmission delay can be at least partially compensated. Alexandraki [AB14] suggested to resynthesize a pre-recorded musical piece at a remote location using control data from a live performance to reduce latency and data rate.

Another typical problem of NMPs is the deviation in the sampling frequency of several NMP clients. The sampling frequency of most audio devices is derived from a high-frequency quartz clocks which slightly differ in their nominal frequency due to component tolerances or different local temperatures. Different sampling rates necessarily cause data over- or under-runs and hence audible impairments. Therefore, the so-called clock drift has to be properly compensated by measuring the frequency offset and readjusting one of the clocks [CW09] or resampling adaptively [PEV10]. Integrating visual cues to allow familiar communication between musicians potentially enhances NMP systems. A solution to transmit additional, synchronized video signals within the same transport stream to avoid additional jitter and therefore packet loss is presented in [CS11a]. Another approach to transmit visual cues without the necessity of transmitting video and therefore clearly increasing data rate is to virtually conduct other musicians [CS11b].

Goals

Many commercial and academic NMP systems with different characteristics and features, as listed in Tab. 1.1, were developed and distributed. The plurality of systems suggests that the general NMP technology is already well-engineered in terms of transmission, synchronization, and interfacing. Therefore, the aim of this work shall not be the definition of a new system, providing any further academic value, but the proposal of several enhancements to enrich all existing and future systems in different aspects.

Known NMP systems share a similar software architecture which is sketched in Fig. 1.2. A software module handling the recording of the signal to be sent feeds an audio encoder to compress the recorded audio data before sending it using the network sender module. Peer-to-peer based NMP implementations require multiple network receivers running in parallel to collect the network packet stream of every participant. The streams have to be decoded to be placed in a replay buffer that counteracts inconsistent transmission delays caused by the network. Server-client based implementation where the mixing takes place at the server solely require a single receiver. In case of missing packets the audio concealment module is used to fill the gap in the received audio stream. Before replaying the audio data, the buffered streams have to be rendered to achieve a replayable audio format. The focus of this work lies on the blue-marked software modules Audio Concealment, Audio Encoder and Decoder, and Audio Renderer. The remaining modules are not further discussed. The goals of this study are defined in the following:

Table 1.1: Selection of commercial and academic NMP implementations.

Name	Provider	Connection	Special Feature	Reference
DIAMOUSES	Technological Educational Institute of Crete (TEI)	Server-Client	Video	http://www.teicrete.gr/diamouses
eJamming	eJAMMING AUDIO	Peer-to-Peer		http://www.ejamming.com
Jacktrip	Center for Computer Research in Music and Acoustics (CCRMA)	Peer-to-peer	Open Source	https://github.com/jcacerec/jacktrip
jamBerry	Department of Signal Processing and Communications (HSU)	Peer-to-Peer	Hardware Solution Partly Open Source	Prototype developed at HSU [MFZ14]
jamLink	MusicianLink	Peer-to-Peer	Hardware Solution	https://www.musicianlink.com/
jamkazam	JamKazam, Inc.	Peer-to-Peer		https://www.jamkazam.com/
Jamulus		Server-Client	Open Source	http://llcon.sourceforge.net/
LoLa	Conservatorio di Musica G. Tartini	Peer-to-Peer	Video Room effect	http://www.conservatorio.trieste.it/art/lola-project/
NinJam	Cockos Inc.	Server-Client	Beat-synchronized	http://www.cockos.com/ninjam/
sofasession	Sofasession	Peer-to-Peer		http://www.sofasession.com/
SoundJack	Alexander Carôt	Peer-to-Peer		http://www.soundjack.eu/

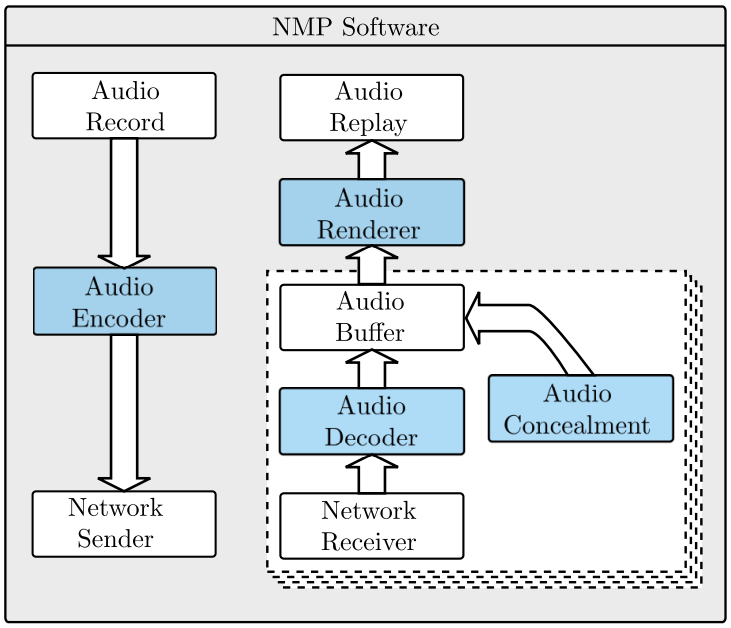


Figure 1.2: Generic NMP software architecture. Modules covered in this work are marked in blue.

- Generic Low-Delay Audio Error Concealment in Chapter 2:
  The IP network as a communication channel is highly non-deterministic and error-prone due to varying load conditions and therefore, the timely arrival of a network packet containing audio data can never be guaranteed. Therefore, packet loss and the resulting audio error must be concealed by replacing the missing data of a lost packet with synthesized audio to prevent a drastic decrease in audio quality. Two error concealment techniques are derived that allow to synthesize audio data generically since they are operating in the time domain and therefore, are not bound to any specific audio codec. The first approach synthesizes a concealment signal based on an auto-regressive model. Multiple approaches to compute the auto-regressive model are analyzed and compared using perceptually motivated measurements. The second approach was developed to have an alternative algorithm featuring very low computational complexity and is based on wave form substitution.

- Low-Delay Audio Coding in Chapter 3:
  Besides the possible data loss caused by the network, one has to consider the limited data rate of most NMP users. Especially typical network links used at home can't exhibit the necessary data rate to send and receive multiple high-quality audio streams. Hence, the application

of an audio codec to compress the audio data is inevitable. A novel audio coding approach is presented that features smaller algorithmic delay than codecs typically applied in the application field of global interactive audio communication and still achieves high data compression. The delayless *Adaptive Differential Pulse Code Modulation* (ADPCM) encoding approach is utilized in subbands and combined with a *Vector Quantizer* (VQ) yielding the *Vector-Quantized Adaptive Differential Pulse Code Modulation* (VQ-ADPCM).

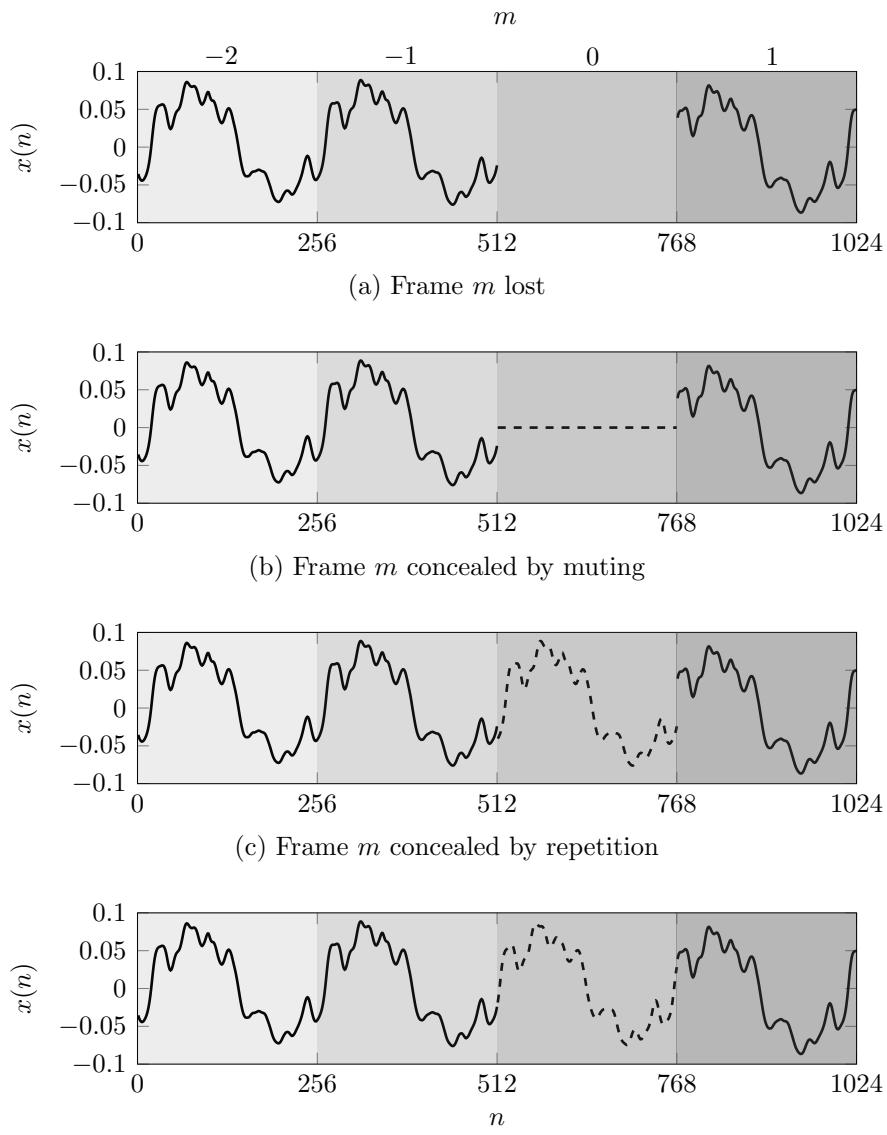
- Efficient Auditory Virtualization in Chapter 4:
  The third goal is to allow the NMP user an enriched acoustic experience. Musicians are used to hear accompanying musicians and themselves in a natural environment, like a rehearsal room or a concert hall. Hearing the dry single-channel recorded instruments of the other musicians leads to an unusual and unpleasant hearing experience. The spaciousness and spatial width of the stereo mix can be increased using the proposed pseudo stereo method which blindly estimates a stereo signal from mono sources using a pair of especially designed filters. The complementary design of the filter prevents timbral coloration of the sound sources and guarantees downmix compatibility to allow the recording and mixing of NMP sessions. The integration of the proposed pseudo stereo method within a virtual surround render method is additionally depicted. Applying the proposed stereo method within the virtual surround renderer allows to define the size of sound sources.

2 Error Concealment

Whenever communication takes place in a packet-switched network instead of a circuit-switched network, the possibility of data arriving too late or not at all has to be considered. This is caused by the dynamic routing process which results in a varying inter-arrival time of network packets called network jitter. Furthermore, transport problems like congestions at a certain node within the routing path can occur and may lead to strong jitter and loss. To prevent an impairment of the signal quality by jitter, large buffers at the receiver are typically applied.

Certainly, an interactive application should limit the receiving buffer to the smallest possible length to allow low-delay communication. Hence, strongly-delayed and lost packets must be handled differently. *Packet Loss Concealment* (PLC) can be applied to fill the gap in the digital audio signal caused by the missing packet. Various simple PLC methods like muting or repetition of previous packets are known. A PLC example is shown in Fig. 2.1. The unconcealed waveform $x(n)$ (Fig. 2.1a), segmented in frames of 256 samples, is incomplete since frame m is missing due to a lost packet. The muting (Fig. 2.1b) and repetition (Fig. 2.1c) are capable of filling the gap to allow replay but due to the clear discontinuity in the waveform, a drastic impact on the audio quality can be expected. In contrast, advanced concealment methods (Fig. 2.1d) allow to produce a well-fitted concealment signal with clearly decreased quality impact.

Several contributions in the area of PLC, especially in the area of *Voice over IP* (VoIP), have been made. Surveys of some basic PLC methods are listed in [PHH98, WSD00]. Most approaches are not directly applicable for NMP due to their additional algorithmic delay, caused by sender-based repair, and the insufficient audio quality for full-range audio, though. Advanced methods like Waveform Similarity Overlap-Add [SYRG96], Linear Prediction



(a) Frame m lost

(b) Frame m concealed by muting

(c) Frame m concealed by repetition

(d) Frame m concealed by model-based resynthesis

Figure 2.1: Packet loss concealment with different techniques.

[GM01], or Hidden Markov Models [RMAJ06] were used to conceal packet losses in VoIP systems. Nevertheless, concealing in the broadband audio case was not investigated. Preihs et al. compared the application of Kalman filters and Linear Prediction to conceal packet loss in broadband audio without delay [SP12].

In the following, two different approaches are investigated that allow a signal-matched synthesis of audio signals. The first method is a model-based synthesis approach whereas the second can be described as waveform substitution. Both methods are solely utilizing previous time-domain audio signals to create concealment signals and therefore can be applied with any audio codec. PLC methods designed for a specific codec and utilizing additional information are expected to show better performance but are not generically applicable.

2.1 Auto-Regressive Modeling (AR)

It is well-known that speech can be authentically reproduced using the source-filter model ([RS78]). The sound of many instruments can be modeled the same way as is shown in [KKZS03, vTSM10]. The source-filter model describes the process of sound production basically as a filtering operation of the source signal, which is typically a combination of a periodic impulse train and white gaussian noise to simulate voiced and unvoiced excitation, respectively. The source-filter model is usually *auto-regressive* (AR) and threfore the model output

$$\begin{aligned} y(n) &= \frac{1}{a_0}\left(a_1 y(n-1) + \cdots + a_p y(n-p)\right) + x(n) = \\ &= \frac{1}{a_0}\left(\sum_{i=1}^{p} a_i y(n-i)\right) + x(n)\,, \end{aligned} \tag{2.1}$$

consists of prior model outputs $y(n-i)$ weighted with p filter coefficients a_i in addition with the excitation signal $x(n)$. The representation in the z-domain

$$H(z) = \frac{1}{\sum_{i=0}^{p} a_i z^{-i}} \tag{2.2}$$

shows that the model is a purely recursive *infinite impulse response* (IIR) filter which can be used to extrapolate a given sequence [KR02]. To conceal a lost frame m, the model-based concealment system, as shown in Fig. 2.2, can be applied as follows:

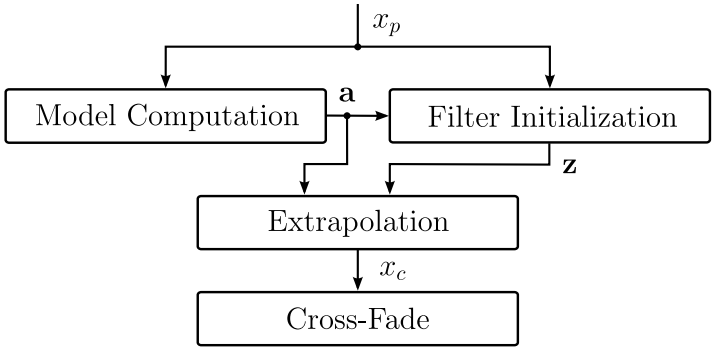


Figure 2.2: AR model-based concealment system overview.

1. Model Computation: Create a signal model from previous data $x_p(n)$ by computing prediction coefficients a_i. Multiple strategies for the computation for different prediction filter realizations are known as shown in the following section.

2. Filter initialization: The recursive extrapolation filter has to be properly initialized to perform extrapolation since the output depends on the filter states as can be seen in Eq. (2.1).

3. Extrapolation: Feed the extrapolation filter with silence to perform the actual extrapolation and obtain the concealment signal $x_c(n)$. Typically, $N + O$ samples are synthesized to allow cross-fading with the following block. N denotes the block length and O the amount of overlapping samples.

4. Cross-Fade: To allow continuous transition from the concealed to the next intact block, cross fading over O samples is applied

2.1.1 Computation of Model Parameters

The process of audio data extrapolation can also be interpreted as the linear prediction of unknown samples

$$\hat{y}(n) = \sum_{i=1}^{p} a_i y(n-i) \tag{2.3}$$

from previous data $y(n-i)$. The difference between predicted signal and original signal is denoted prediction error

$$e(n) = y(n) - \hat{y}(n) = y(n) - \sum_{i=1}^{p} a_i y(n-i)) \tag{2.4}$$

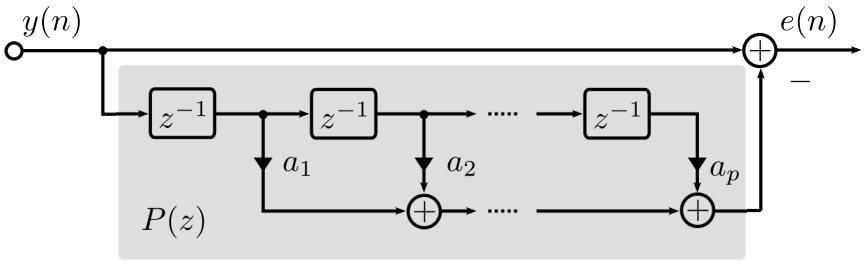


Figure 2.3: Transversal prediction filter realization.

and can be computed using the transversal filter structure of Fig. 2.3. Whenever a cost function $J(n)$ describing the squared prediction error

$$J(n) = E\{e^2(n)\} \tag{2.5}$$

is minimized by a set of prediction coefficients a_i, one can a expect a prediction signal close to the original one. Several techniques to compute the prediction filter coefficients are derived in the following.

Least Mean Square (LMS)

A classical optimization technique called steepest descent can be applied to minimize J iteratively. The coefficients a_i are recursively updated by subtracting the weighted derivative of the cost function

$$a_i(n+1) = a_i(n) - \frac{\mu}{2}\frac{\partial J(n)}{\partial a_i}, \quad i \in (1, \ldots, p). \tag{2.6}$$

The weighting $\frac{1}{2}\mu$ allows to control the adaption speed. Inserting the squared error cost function Eq. (2.5) and solving the derivative leads to

$$\frac{\partial J(n)}{\partial a_i} = 2E\left\{e(n)\frac{\partial e(n)}{\partial a_i}\right\} = 2E\{e(n)\, y(n-i)\}. \tag{2.7}$$

Inserting Eq. (2.7) into Eq. (2.6) and neglecting the expectation value results in the LMS update formula

$$a_i(n+1) = a_i(n) + \mu\, e(n)\, y(n-i), \quad i \in (1, \ldots, p). \tag{2.8}$$

Convergence of the LMS is solely guaranteed when

$$0 < \mu < \frac{2}{\sum_{i=0}^{p-1} y(n-i)^2} \tag{2.9}$$

is fulfilled [Hay91]. The LMS algorithm can be extended to perform the filter adaption with a constant speed and hence independent of the signal power. Therefore, a normalized time-variant gradient step size

$$\mu(n) = \frac{\lambda}{\sigma^2(n) + \sigma_{min}} \tag{2.10}$$

is applied instead of the fixed one in Eq. (2.6). The base step size is divided by the squared euclidean norm $\sigma^2(n)$, which is

$$\sigma^2(n) = \sum_{i=0}^{p-1} y(n-i)^2 \tag{2.11}$$

for a transversal filter of length p. Numerical stability is improved by adding the constant σ_{min} in the divisor to avoid divisions by small values, occurring for quiet or silent signals $y(n)$. The resulting *normalized least mean squares* (NLMS) algorithm is adjustable via the base step size λ.

Autocorrelation Method (ACM)

Another way of computing the coefficients of a transversal prediction filter according to the *Minimum Mean Square Error* (MMSE) criterion is to estimate a stochastic process for a limited series of data and replace the expectation value of the product in Eq. (2.5) using an autocorrelation instead of approximating it with least mean squares. Setting the gradient of the cost function to 0 results in

$$\frac{\partial J}{\partial a_i} = \frac{\partial E\{e^2(n)\}}{\partial a_i} = 2E\left\{e(n)\frac{\partial e(n)}{\partial a_i}\right\} \stackrel{!}{=} 0. \tag{2.12}$$

Substituting $e(n)$ with Eq. (2.4) yields

$$-2\left(E\{y(n)y(n-i)\} - \sum_{i=1}^{p} a_k E\{y(n-k)y(n-i)\}\right) \stackrel{!}{=} 0. \tag{2.13}$$

The expression $E\{y(n)y(n-i)\}$ represents the definition of the autocorrelation function $r_{yy}(i)$ and therefore, Eq. (2.13) can be reformulated to

$$\sum_{k=1}^{p} a_k r_{yy}(k-i) = r_{yy}(i), \quad i \in (1, \ldots, p). \tag{2.14}$$

These so-called Yule-Walker equations can be efficiently solved using the Levinson-Durbin recursion [Dur60].

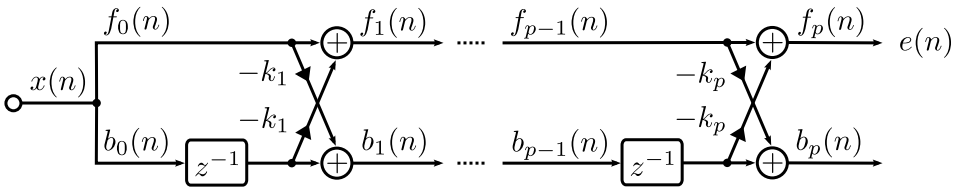


Figure 2.4: Lattice prediction filter realization.

Gradient-Adaptive Lattice (GAL)

Besides transversal filters in form of direct and transposed digital filter structures, the class of lattice filters, as shown in Fig. 2.4, can be utilized in filter and prediction applications [OS09]. The forward and backward prediction error f_i and b_i per lattice stage i are described by

$$\begin{aligned} f_i(n) &= f_{i-1}(n) - k_i\, b_{i-1}(n-1) \\ b_i(n) &= b_{i-1}(n-1) - k_i\, f_{i-1}(n), \quad i \in (1,\ldots,p). \end{aligned} \tag{2.15}$$

The prediction error is iteratively minimized throughout the lattice stages and therefore, the 0_{th} stage $f_0(n) = b_0(n) = x(n)$ is the current input signal $x(n)$. Similar to Eq. (2.6) the method of steepest descent

$$k_i(n+1) = k_i(n) - \frac{1}{2}\mu_i(n)\frac{\partial J_i}{\partial k_i},\ i \in (1,\ldots,p) \tag{2.16}$$

can be applied to iteratively adapt the lattice prediction coefficients to minimize the cost function $J_i = E\{f_i^2(n) + b_i^2(n)\}$ in lattice stage i. Inserting the derivative of the cost function

$$\frac{\partial J_i}{\partial k_i} = 2\,E\left\{ f_i(n)\frac{f_i(n)}{\partial k_i} + b_i(n)\frac{b_i(n)}{\partial k_i} \right\} \tag{2.17}$$

$$= -2\,E\{(f_i(n)\, b_{i-1}(n-1) + b_i(n)\, f_{i-1}(n))\} \tag{2.18}$$

into Eq. (2.16) results in the coefficient update rule

$$k_i(n+1) = k_i(n) + \mu_i(n)\ (f_i(n)\, b_{i-1}(n-1) + b_i(n)\, f_{i-1}(n))\,. \tag{2.19}$$

The stage-dependent adaption step size is power-adaptive

$$\mu_i(n) = \frac{\lambda}{\sigma_i^2(n) + \sigma_{\min}} \tag{2.20}$$

as in the case of LMS (Eq. (2.10)) but individually for every lattice stage. The error power in stage i is computed as the recursive average

$$\sigma_i^2(n) = \lambda\,\sigma_i^2(n-1) + (1-\lambda)\,(f_{i-1}^2(n-1) + b_{i-1}^2(n-1)), \tag{2.21}$$

where λ controls the memory of the averaging as well as the base adaption speed. The direct-form filter coefficients a_i can be obtained from the lattice parcor coefficients k_i using the following algorithm [OS09].

Algorithm 1 *Compute transversal filter coefficients from lattice filter coefficients.*

for i $= [1, \ldots, p]$ **do**
  $a_i^{(i)} = k_i$
  if $i > 1$ **then**
    for $j = [1, \ldots, i-1]$ **do**
      $a_j^{(i)} = a_j^{(i-1)} - k_i \, a_{j-1}^{(i-1)}$
    end for
  end if
end for
$a_j = a_j^{(M)}, \quad j = [1, \ldots, p]$

Burg method

The optimum coefficients for a lattice prediction filter can be obtained in a similar manner. Substituting the prediction error in stage i in the cost function

$$J_i = E\{f_i(n)^2\} + E\{b_i(n)^2\} \tag{2.22}$$

using Eq. (2.15) leads to

$$J_i = E\{(f_{i-1}(n) + k_i \, b_{i-1}(n-1))^2\} + E\{(b_{i-1}(n-1) + k_i \, f_{i-1}(n))^2\} \tag{2.23}$$

$$= E\{f_{i-1}^2(n) + b_{i-1}^2(n-1)\}(1 + k_i^2) + 4 \, k_i \, E\{f_{i-1}(n) b_{i-1}(n-1)\}. \tag{2.24}$$

Computing the derivative with respect to the filter coefficients k_i

$$\frac{\partial J_i}{\partial k_i} = 2 \, k_i \, E\{f_{i-1}^2(n) + b_{i-1}^2(n-1)\} + 4 \, E\{f_{i-1}(n) b_{i-1}(n-1)\} \tag{2.25}$$

and setting it to 0 yields

$$k_i = -\frac{2E\{b_{i-1}(n-1) f_{i-1}(n)\}}{E\{f_{i-1}^2(n) + b_{i-1}^2(n-1)\}}. \tag{2.26}$$

Assuming an ergodic process allows defining the Burg estimator for a block of length N [Hay91]

$$k_i = -\frac{2 \sum_{n=i+1}^{N-1} (f_{i-1}(n) \, b_{i-1}(n-1))}{\sum_{n=i+1}^{N-1} \left(f_{i-1}^2(n) + b_{i-1}^2(n-1)\right)}. \tag{2.27}$$

2.1.2 Initialization, Extrapolation and Fading

After obtaining the prediction filter coefficients from previous samples to extrapolate audio data, one has to prepare the synthesis filter by setting the internal states to meaningful initial values. The initialization depends on the utilized filter implementation. In the case of direct-form I filter structure, using previous values is sufficient. Using the transposed direct-form II implementation requires a reverse filter operation as shown in [OS09].

The actual extrapolation is then performed by feeding $N+O$ zero samples to the initialized recursive synthesis filter and hence let the filter oscillate. The length of the concealment signal should be larger than the block length N to allow cross-fading as explained in the following.

Smooth, transient-less transition from the concealed audio data block m to the next intact block $m+1$ can be guaranteed by cross-fading

$$y_{m+1}(n) = w(n)\, y_{m+1}(n) + w(O-n)\, y_m(N+n), \quad n \in (0, \ldots, O). \tag{2.28}$$

Experiments in [FZ14] have confirmed that constant-amplitude windowing functions

$$w(n) + w(O-n) = 1 \tag{2.29}$$

are perceptually superior to constant-power windowing functions

$$w^2(n) + w^2(O-n) = 1, \tag{2.30}$$

which are typically used in mixing applications. This is due to the strong correlation of original and concealed audio signals. A mathematical derivation of this phenomenon is found in A.3.

2.2 Waveform Substitution (WS)

The discussed model-based concealment is capable of concealing lost packets but features the drawback of being computationally expensive since an autoregressive model has to be computed for every lost frame, before it can be applied. Therefore, another concealment method was developed that allows to restore an audio stream impaired by packet loss in a computationally less complex way [FZ14]. This approach is based on waveform substitution (WS). Multiple periods are extracted from previous data and reshaped to form the concealed frame.

The concealment system's module-wise structure is depicted in Fig. 2.5. Prior data $x_p(n)$ is optionally refined by the pre-processing module before it is fed to the *zero-crossing analysis* (ZCA) module, which delivers the extraction

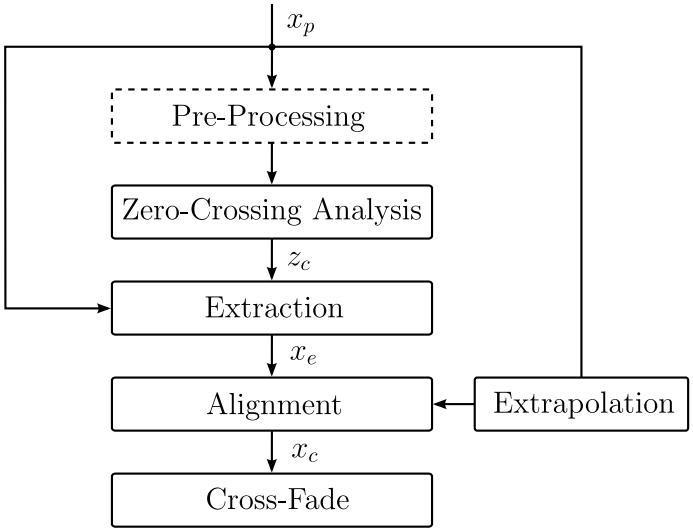


Figure 2.5: WS concealment system overview.

borders $z_c(n)$. Multiple periods of the unaltered signal $x_p(n)$ are cut out using $z_c(n)$ by the extraction module. This step produces the extracted concealment signal $x_e(n)$, which is potentially repeated until the block length $N + O$ is reached. The alignment module is supposed to align the phases of $x_e(n)$ to the tail of $x_p(n)$ and hence assure a smooth phase transition. In addition, the phase transition is enhanced further by extrapolating a few samples using the extrapolation module and cross-fading the extrapolation of $x_p(n)$ with $x_e(n)$. Similar to Sec. 2.1 a cross-fade in the next intact block is performed. Every module is explained in greater detail in the following.

Pre-processing

Two optional enhancement steps are contemplated to improve the following ZCA process. First, non-linear functions are utilized to enhance the ZCA performance as shown in [Hes79]. Two non-linear function

$$f_{\mathrm{NL1}}(x) = \sqrt{|x|} \tag{2.31}$$

$$f_{\mathrm{NL2}}(x) = \sqrt{|x|} \cdot \mathrm{sgn}(x) \tag{2.32}$$

were chosen which solely differ in symmetry as can be seen in Fig. 2.7a).

In addition to the nonlinearities, the analysis band width is restricted using a *finite impulse response* (FIR) lowpass of order 20 with a normalized cut-off frequency $\omega_c = 0.01 \cdot 2\pi$. The FIR filter, plotted in Fig. 2.7b), is applied in forward and backward direction to the buffer of previous data

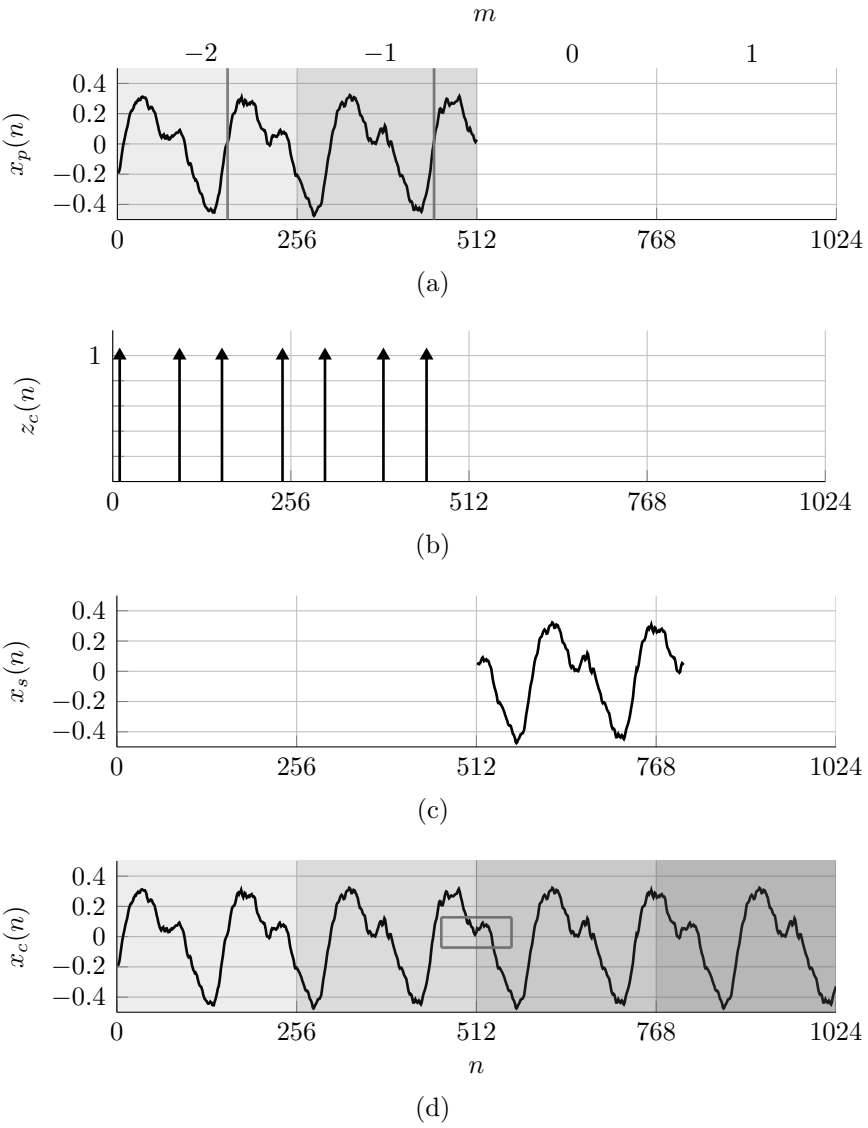
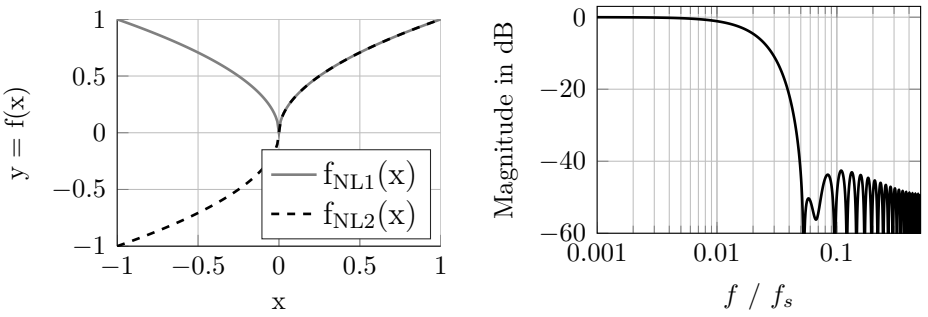


Figure 2.6: Exemplary waveform substitution concealment. Two previous frames $m-2$ and $m-1$ in a). Corresponding zero crossings in b). Aligned, extracted concealment signal in c). Concealed waveform in d).



(a) (b)

Figure 2.7: Pre-processing non-linearities and filters.

$x_p(n)$ to avoid any group delay influences and to maintain the position of the zero crossings [OS09].

Zero-Crossing Analysis

Zero-crossings are chosen to identify a periodic excerpt since they are known to be a useful low-level semantic audio feature [Ler12]. The position of a zero-crossing can easily be identified by comparing the sign of two consecutive samples. Identified zero-crossing are stored in a binary vector

$$z_c(n) = \begin{cases} 1 & \text{if} \quad x(n) \cdot x(n-1) < 0 \\ 0 & \text{else.} \end{cases} \tag{2.33}$$

The resulting vector $z_c(n)$, illustrated in Fig. 2.6b), can be enhanced by imposing a minimum zero-crossing interval

$$\delta_{\text{zc}} = \left\lceil \frac{f_s}{2 \cdot f_{\max}} \right\rceil, \tag{2.34}$$

resulting from the largest assumed fundamental frequency $f_{\max}$ of the input signal and the sampling frequency f_s. On the contrary, the smallest expected fundamental frequency $f_{\min}$ can be respected by analyzing at least

$$N_p = \left\lceil \frac{\beta \cdot f_s}{f_{\min}} \right\rceil \tag{2.35}$$

prior values and hence guarantee that the searched interval is sufficiently large to hold a single period of $f_{\min}$. The interval can be optionally increased by choosing a safety margin $\beta \geq 1$.

Extraction

A single or multiple periods are extracted from the original $x_p(n)$ by cutting out at the boundaries defined by the last entries of $z_c(n)$. For example, the last 5 zero crossings from Fig. 2.6b) were utilized to define the extraction boundaries in Fig. 2.6a).

Alignment

In the next step, $x_e(n)$ is phase-aligned to the end of $x_p(n)$ by circularly shifting it by l samples

$$x_s(n) = x_e(n - l \bmod N_e) \tag{2.36}$$

to yield the aligned concealment signal $x_s(n)$. Several approaches to compute the shifting offset l were analyzed in [FZ14]. The resulting aligned excerpt $x_s(n)$ of length N_e is shown in Fig. 2.6c).

Extrapolation

The continuity from $x_p(n)$ to $x_s(n)$ can be improved by extrapolating N_f values of $x_p(n)$ and cross-fade them with the first samples of $x_s(n)$. Again, several methods to perform the extrapolation were reviewed in [FZ14].

Fade-Out

Similar to the model-based concealment system, the concealment signal is of length $N + O$ and therefore, longer than the actual blocks to allow cross-fading with the next intact block as described in Eq. (2.28). The concealed waveform $x_c(n)$ is depicted in Fig. 2.6d).

2.3 Evaluation

The evaluation of the two proposed concealment strategies is performed in terms of complexity and perceptual quality which is assessed using an automated measurement. Typical measurements like the *Signal-Noise-Ratio* (SNR), *Total Harmonic Distortion* (THD), or any broadband error power measurements are not of interest since the only relevant criterion to evaluate the concealment is the perceived quality.

The first evaluation aspect to be considered is the simulation of lost packets. For this purpose, a test signal $x(n)$ is loaded from a wavefile, downmixed

to a single channel, and segmented into M frames of length N. Next, a random sequence m_{rnd} with an uniform distribution of length M with amplitudes between 0 and 1 is computed. Whenever the packet loss rate r_{drop} exceeds the value of m_{rnd}

$$r_{\text{drop}} \geq m_{\text{rnd}} \tag{2.37}$$

the corresponding frames are assumed to be lost. For every lost frame, the concealment methods from Sec. 2.1 and Sec. 2.2 are applied to conceal the resulting gap in the waveform. The evaluation is performed for every item of the EBU *Sound Quality Assessment Material* (SQAM) [EBU08] and several configurations of the proposed concealment methods. The SQAM dataset consists of 70 various test-signals like synthetic signals in addition to recordings of various instruments, speech, vocals, orchestras, and popular music pieces. The test tracks are sampled with a rate of 44100 Hz due to their distribution in CD form and are also available online in an uncompressed format. The perceptually motivated measurement tool is presented in the following.

2.3.1 Measuring Perceptual Audio Quality With PEAQ

Finding the optimal way of evaluating the audio quality of a signal processing algorithm remains an unsolved problem. Listening to the algorithms output of carefully designed test items within a well-defined spatial environment by several trained listeners and the subsequent formal rating is one of several possibilities. This approach is called a listening test and typically the preferable choice. In contrast, measuring audio quality is a non-trivial, much discussed topic due to the subjective properties of hearing. Therefore, representing perceptual quality in any objective metric seems inappropriate on a first glance. Nevertheless, the International Telecommunication Union (ITU) recommended a method for objective measurements of *perceived audio quality* (PEAQ) [ITU01] that is intended to yield similar results like a listening test according to ITU-R BS.1116-1 [ITU97].

The PEAQ evaluation process takes place as drafted in Fig. 2.8. A test signal is processed with the *device under test* (DUT) which is typically an audio codec but in this case we apply packet loss simulation in addition with the proposed concealment methods. The resulting processed signal and the unprocessed reference signal are fed to the actual evaluation software. The PEAQ algorithm computes several perceptually motivated features based on a psychoacoustical model representation of processed and unprocessed test signals. The features are combined to produce an overall rating called *Objective Difference Grade* (ODG). The ODG values range from -4 to 0 to assess

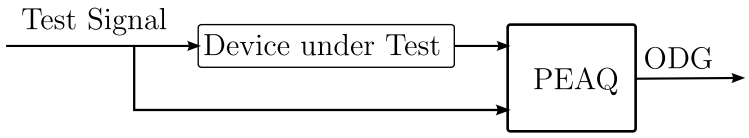


Figure 2.8: PEAQ evaluation process.

the audio impairments of the DUT from "very annoying" to "imperceptible". An open-source implementation [HZ15] is utilized that has proven its usefulness in several studies.

2.3.2 Comparison of the Concealment Quality

Model-based Synthesis

First, it shall be exposed which configurations of the concealment methods are beneficial for the purpose of hiding the audio quality impact caused by lost network packets. Four methods to compute auto-regressive models were presented in Sec. 2.1.1 and applied in the proposed model-based concealment method of Sec. 2.1. The resulting average ODG scores, measured with the PEAQ tool in basic mode on the SQAM data set, using a block size and prediction order of $N = P = 128$ samples and a packet loss rate of $r_{\text{drop}} = 0.01$ are plotted over SQAM items in Fig. 2.9a). Muting lost frames is the chosen reference method and yields an average score of -2.33 corresponding to "Slightly annoying" impairment of audio quality. The unexpectedly good score is caused by the silent parts of the test material. Within speech or percussion instrument tracks, many silent parts occur which are not impaired by a simulated lost packet. It is also noticeable that packet loss in broad band, complex signals like the orchestral and pop music pieces are rated less severe than for single instruments. Applying the AR concealment increases the measured perceptual quality significantly, as it is apparent in the clear gap between the curves in Fig. 2.9a).

Applying the Burg method to perform model-based error concealment improves the average ODG score to -0.44. The improvement of almost 2 ODG scores clearly proves the quality improvement caused by the proposed concealment method. The ACM method yields a slightly worse score of -0.74, followed by GAL yielding a score of -0.75, and at last NLMS with a score -1.38. It is already apparent that the choice of the predictor is essential for the concealment quality and the Burg method is the most promising approach whereas the NLMS can't be advised.

Varying the block length N for a fixed loss rate $r_{\text{drop}} = 0.01$ and prediction order $p = 64$ reveals the advantage of longer blocks as can be seen

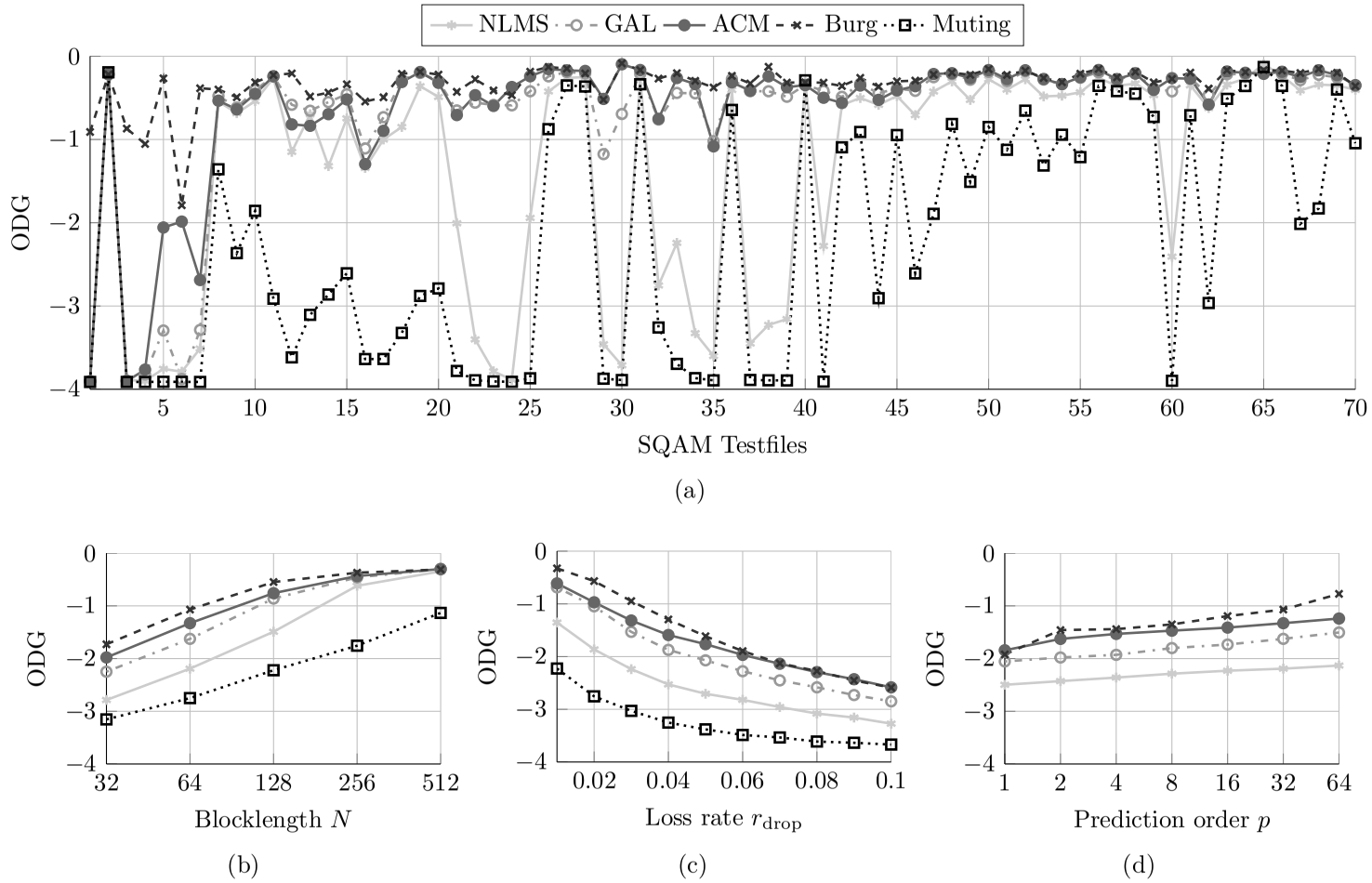


Figure 2.9: ODG scores of AR concealment over SQAM tracks a), block length N b), loss rate r_{drop} c), and prediction order p d).

in Fig. 2.9b). This is caused by the fact that a constant amount of 3 previous frames is utilized to compute the AR models and longer data sequences tend to yield more accurate models. However, the muting concealment is rated superiorly for longer blocks as well. The author assumes that longer but fewer gaps in the audio are rated less disturbing by PEAQ. Figure 2.9c) shows the influence of an increasing packet loss rate. As expected, the perceptual quality is clearly degraded in a similar fashion for all measurements. The superiority of the block-based methods is also visible in form of a clear ODG offset of about 0.5 in comparison to the GAL-based concealment. The order of the predictor for a fixed block length and error rate is visualized in Fig. 2.9d). All predictors benefit from an increased order as shown by the similar rising trend of the curves. Apparently, the block-based methods are advantageous in this application and the Burg method is the most promising one and therefore solely utilized in the following. Unfortunately, the Burg method is also quite expensive in terms of computational costs.

To justify the application of PEAQ for the measurement of audio quality impairments caused by network errors in NMP scenarios, a listening test was performed in addition. The listening test was realized with mushraJS [KZ14] which is a web-based listening test platform according to ITU recommendation BS.1534-2 [ITU14]. Five tracks of the SQAM data set were shortened to 10 seconds and processed similar to the PEAQ measurement. Solely the packet loss rate was increased to $r_{\text{drop}} = 0.02$ to allow an easier identification of the artifacts. The participants were asked to rate the overall audio quality and score it within the range of $[0, \ldots, 100]$. The results of the 23 participants, shown in Fig. 2.10, clearly confirm the automated measurements in terms of predictor preference. For every test item the Burg predictor was rated best, followed by ACM, GAL, NLMS, and the muting reference. The similarity of the measurement and listening test indicate that the PEAQ tool is at least a well-suited quality indicator for PLC.

Waveform Substitution

As well as the model-based concealment approach, the waveform substitution methods allows several configurations. As exposed in Sec. 2.2, different pre-processing, alignment, and extrapolation methods are available. To find the configuration yielding the best average ODG score, all possible configurations were computed using the parameters as before. The evaluation in [FZ14] showed that the pre-processing using a non-linearity and a filter, in addition to the phase alignment based on amplitude and slope matching in combination with linear extrapolation led to the best result.

Applying the evaluation mechanism from the previous section and using

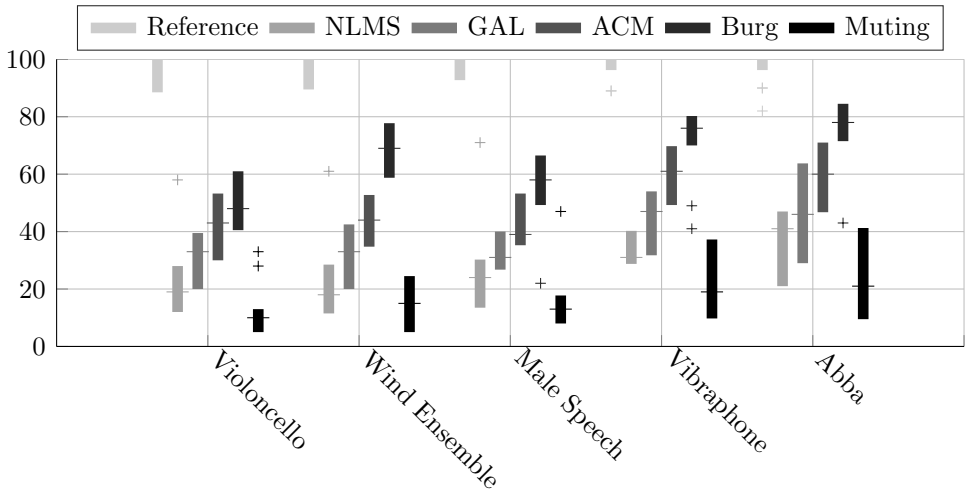


Figure 2.10: Listening test results of AR concealment for 5 SQAM items.

the same parameters of block length $N = 128$ and error rate of $r_{\text{drop}} = 0.01$ leads to an average ODG score of -1.32 and hence an improvement of 1 ODG[1]. As expected, the simple WS approach performs significantly worse than the Burg method but still outperforms the AR concealment using the NLMS predictor when the prediction order is set to $p = 64$. Varying the block length N and the loss rate r_{drop} as shown in Fig. 2.11b+c) results in a similar curve trend as the AR concealment.

2.3.3 Comparison of the Concealment Complexity

The proposed concealment strategies in their best-performing setting are compared in terms of computation cost by analyzing the amount of real multiplications. Remaining operations like additions, comparisons or the computation of absolute values are neglected. The AR concealment consists of three basic steps: The model computation using the Burg method, filter initialization, and the actual filtering. The corresponding amount of multiplications are denoted in Tab. 2.3.3. In contrast, the WS concealment requires only the computation of the zero crossings, the extrapolation, and the optional pre-processing consisting of an FIR filter and the nonlinearity. Besides the extrapolation of a fixed length, all steps depend on the minimal analysis length N_p from Eq. (2.35). The accumulated multiplications of both methods

[1]The measurements of [FZ14] were repeated and led to slightly different results. The superiority of the WS for certain parameter settings in contrast to the AR concealment could not be confirmed.

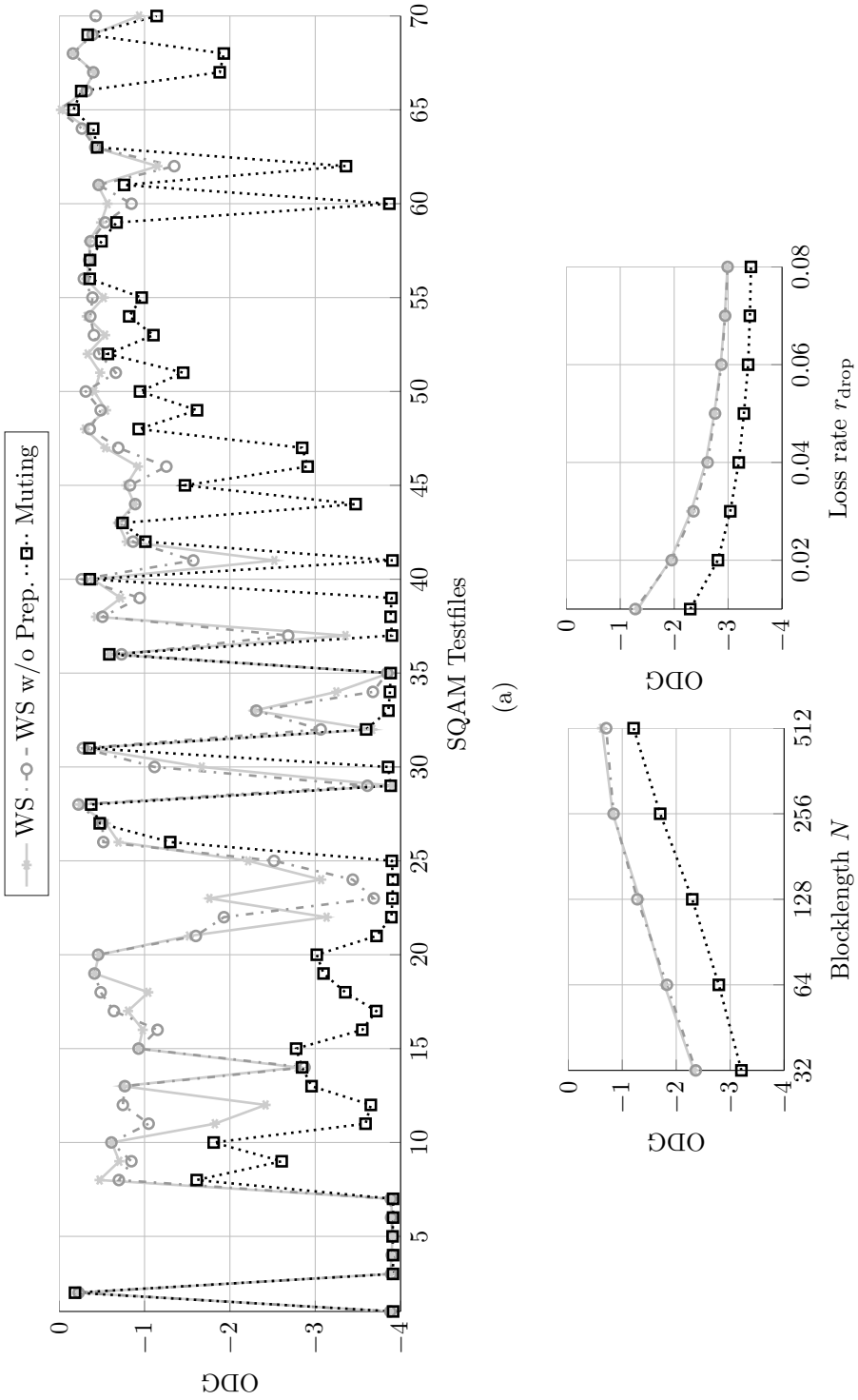


Figure 2.11: ODG scores of WS concealment over SQAM tracks a), block length N b), loss rate r_{drop} c).

Table 2.1: Computational costs of AR and WS concealment denoted in multiplications.

	AR		WS
Burg	$15\,p\,N - \frac{5p^2+p}{2}$	FIR Filter	$42\,N_p$
Filter Initialization	p^2	Nonlinearity	$2\,N_p$
Filtering	$\frac{3}{2}p\,N$	Zero Crossings	N_p
		Extrapolation	5

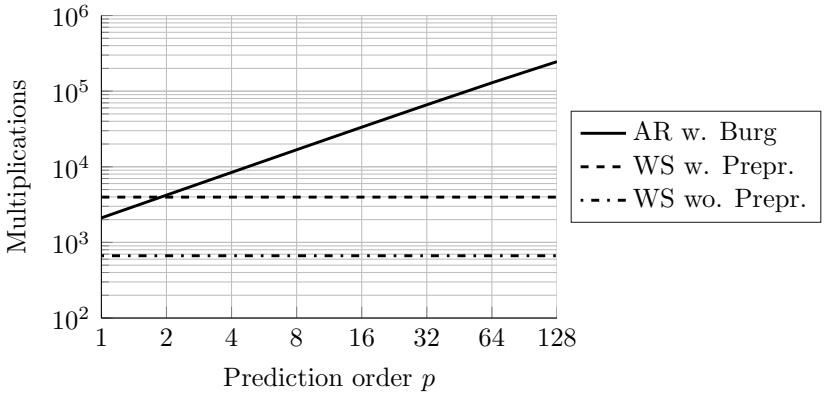


Figure 2.12: Multiplications of AR and WS (with and without pre-processing) concealment plotted against Burg prediction order p for block length $N = 128$ and WS analysis length $N_p = 661$.

for $N = 128$ and $N_p = 661$, corresponding to a minimal frequency of 80 Hz, a sampling frequency of 44.1 kHz and a safety margin $\beta = 1.2$, are plotted in Fig. 2.12 against the prediction order. It can be observed that the AR complexity increases quadratically and intersects the constant curve of the WS concealment at a order of $p = 2$. Significantly more multiplications are required to compute the concealment using higher orders which are required for best perceptual quality. The main complexity of the WS concealment is caused by the pre-processing due to the FIR filter and the nonlinearity. Disabling the pre-processing drastically decreases the complexity and even undershoots the AR complexity for an order of $p = 1$.

2.4 Summary

Any network-based communication technology requires packet loss concealment strategies to retain a certain audio quality in the case of lost network

packets. Two time-domain methods were presented which clearly improve the audio quality in comparison to muting frames in the error case. The first method is based on auto-regressive modeling and delivers physically motivated synthesis signals which are of high quality but expensive in terms of computational costs. For this reason, a second method was developed that mainly substitutes missing samples with a waveform which consists of rearranged previous data. It is drastically simpler to compute but is still able to deliver an enhanced listening experience. Both methods solely require previous time-domain signals and therefore are generically utilizable with any audio codec. The two approaches were evaluated using a perceptually motivated measurement routine, called PEAQ, aiming at predicting the outcome of a listening test. The practicability of the method was demonstrated by additionally performing a listening test yielding results with comparable trends to the measurements. The evaluation on the SQAM dataset with varying method and simulation parameters showed that the auto-regressive concealment using the Burg method is clearly the best performing approach. Improvements of up to 2 ODG in comparison to the muting of lost frames for typical parameters can be achieved. The waveform substitution concealment is only capable to increase the ODG score by 1. Comparing the complexity analytically by evaluating the multiplications depicts that the Burg method with typical parameters is clearly more expensive in terms of computational costs for higher order.

3 Vector-Quantized ADPCM

3.1 Codec Overview

Continuous audio replay in the case of unstable network conditions can be guaranteed with the methods from the previous chapter. However, another property of the utilized network, namely the data rate of internet connection, drastically influences the quality of NMP sessions. Assuming that an NMP participant sends and receives audio streams with a typical sampling frequency $f_s = 48\,\text{kHz}$ and a CD-like quality having $w_b = 16$ bits per sample, the resulting data rate for every up- and down-stream yields $R_b = f_s \cdot w_b = 750\,\text{kbit/s}$. In the case of 4 participants and a non-centralized, server-based communication architecture, each participant requires approximately 2.25 Mbit/s up- and down-stream rate.

A survey of the European Commision in the year 2012 investigated household internet connection and quality in europe. The average upload speed in Germany was identified as 0.74 Mbit/s for DSL connections of any type and 2.9 Mbit/s for cable connection [EC12]. The nationwide survey ordered by the german federal network agency affirmed that the effectively usable down and upload speed are cleary below the advertised offers of internet service providers [BNA13].

Therefore, audio data compression techniques also known as audio coding have to be utilized to allow NMP sessions with typical household internet connections. The development of *MPEG-2 Audio Layer III* (MP3) [ISO93] allowed users to replay their music collections with mobile devices and to exchange music online by significantly reducing the size required for the storage of the media content. This very popular method was further enhanced and led to MPEG-4 *Advanced Audio Coding* (AAC) [ISO97] and MPEG-4 *High-Efficiency Advanced Audio Coding* (HE-AAC) [ISO09]. Amongst oth-

ers, these codecs are massively applied in nowadays entertainment and communication technologies. The core idea of reducing the data rate is to remove irrelevant and redundant information from the audio signal. Irrelevant signal components are identified using a psychoacoustical model that allows to compute a spectral representation of temporarily and instantaneous masked components. Those masked components can be quantized coarsely in contrast to the relevant non-masked components. The analysis and quantization is performed in the time-frequency domain using a *Modified Discrete Cosinus Transform* (MDCT) [Mal90] or a hybrid filter bank based on MDCT and polyphase filter banks. Since the quantized spectrum potentially still features redundancy, entropy coding can be applied to further reduce the data rate of the source signal.

These MPEG audio coding approaches are surely capable of drastically reducing the data rate and still feature a good listening experience in terms of audio quality. In particular, in noisy listening environments the resulting audio quality is sufficient even for lowest bit rates. In the case of interactive communication scenarios like NMP or gaming the bit rate is unfortunately not the most important requirement. These applications are extremely delay-sensitive and hence, the signal processing should feature lowest delays. The MPEG codecs have a significant algorithmic delay of 54 ms for mp3, 55 ms for AAC, and 129 ms for HE-AAC in the best case, respectively [GLS+04]. Therefore, these audio codecs cannot be applied in a NMP scenario. To counteract the problem of missing practicability in low-delay scenarios, the AAC-LD mode was developed. It slightly outperforms mp3 in terms of quality but only features an algorithmic delay of 20 ms [AGHS99]. The delay minimization was achieved by using an adapted filter bank and avoiding block switching.

The algorithmic delay can be significantly decreased by reducing the frame size and the transform length, respectively. The *Constrained Energy Lapped Transform* (CELT) of the OPUS codec [VTMM10, VMTV13] operates with frame sizes of 120 samples corresponding to 2.5 ms in the lowest delay configuration. In addition to the 2.5 ms lookahead, the OPUS codec features an overall algorithmic delay of only 5 ms and still yields audio quality compatible to HE-AAC at a bitrate of 64 kbit/s [Dya11]. It was specifically designed to be utilized in real-time Internet applications like NMP. Nowadays, it is integrated in several browsers, VoIP software, and NMP realizations.

Lower delays are also possible when switching from transform-based approaches based on psychoacoustics to prediction-based methods. The *Ultra-low Delay Audio Coder* (ULD) [HKK+04, HKSW06] is capable of delivering high audio quality while producing only 5.4 ms of algorithmic delay at

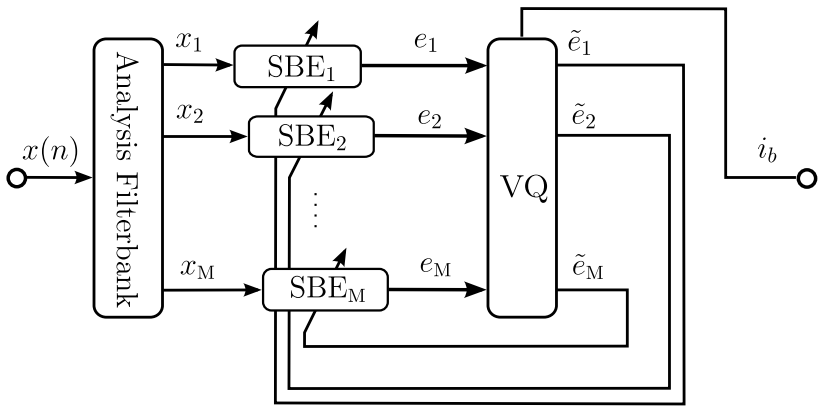


Figure 3.1: Blockscheme of VQ-ADPCM encoder.

$f_s = 48\,\text{kHz}$. It is based on linear prediction in the time-domain and a scalar quantizer combined with pre- and post-filters to shape the quantization noise. The delay is caused by the essential lookahead which is necessary to design the prefilter based on a psychoacoustical analysis. This audio codec already proved its usefulness in NMP scenarios [CKS06].

Another widely used proprietary audio codec is the aptX® codec family. It is based on prediction and quantization as well, but in contrast to the ULD, the processing takes place in subbands. The predictor and quantizer are adapted using quantization error feedback. This scheme is called Adaptive Differential Pulse Code Modulation (ADPCM). The low-latency variant features data rate reduction of factor 4 and an algorithmic delay of 1.89 ms at $f_s = 48\,\text{kHz}$ [APT]. The delay is caused by the filter bank which divides the fullrange input signal in subband signals.

Another subband ADPCM (SB-ADPCM) approach was presented by Keiler [Kei06b]. It claims nearly transparent quality with a minimum bit rate of 128 kbit/s showing an algorithmic delay of $\approx 3\,\text{ms}$ using 8 subbands. The broadband ADPCM approach from Holters et al. [HHZ08] doesn't show any delay at all. The 3 bit and 4 bit variants result in a bitrate of 144 kbit/s and 192 kbit/s, respectively. Noise shaping is performed using adaptive shelving pre-and-post filters [HZ08] or noise feedback [HHZ08].

Since the delay of the lastly described codec is already optimal, this chapter shall reveal how the bit rate of an ADPCM codec can be further decreased when a tiny delay is allowed. Further reducing the bit rate helps to establish NMP sessions with multiple users when the user is bound to a typical household Internet connection. The proposed codec design of this work consists of multiple ADPCM codecs in multiple subbands. In contrast to the mentioned subband ADPCM codecs [Kei06b, APT], a vector quantizer (VQ)

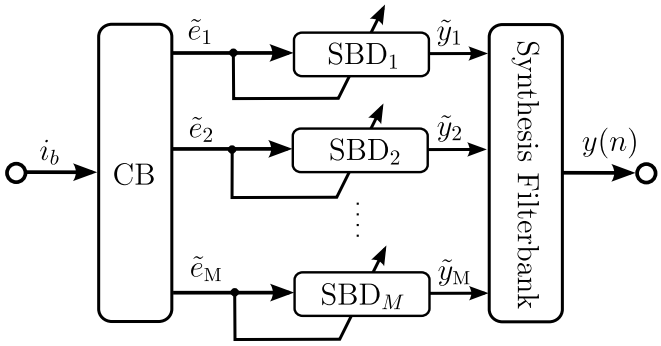


Figure 3.2: Blockscheme of VQ-ADPCM decoder.

will be applied to the subband signals instead of a scalar quantizer. It is well-known that VQ's are taking advantage of non-linear dependencies of the source [MRG85]. In other words, quantizing a non-memoryless source with a VQ is always beneficial and a certain coding gain can be expected since subband signals are still highly correlated.

The combination of a filter bank, ADPCM prediction tools, and a VQ was already attempted in [MZFR00]. But the simple evaluation therein does not allow a classification of this codec within the aforementioned codecs in terms of quality and hence, a new design of the codec structure followed by a global optimization for the purpose of NMP is performed in this study.

In fact, the presented approach, sketched in Fig. 3.1, consists of an *analysis filter bank* (AFB) to divide the input signal $x(n)$ in M critically sampled subband signals $x_1(b), \ldots, x_M(b)$ where b represents the time index of the subband signals. Each subband signal is individually processed by an ADPCM subband encoder SBE$_m$. The outputs of the subband encoders $e_1(b), \ldots, e_M(b)$ are jointly quantized using the VQ resulting in the quantized subband error signals $\tilde{e}_1(b), \ldots, \tilde{e}_M(b)$. The vector of quantized error samples is transmitted solely using the codebook vector i_b without further side information. The encoding process is illustrated in Fig. 3.3, where the signals of every processing step is shown.

The corresponding decoder is shown in Fig. 3.2. The quantized subband encoder outputs $\tilde{e}_1(b), \ldots, \tilde{e}_M(b)$ are reconstructed using a lookup operation utilizing the index i_b. Therefore, the codebook CB has to be known in encoder and decoder. In every band, a subband decoder SBD$_m$ is applied to compute the reconstructed subband signals $\tilde{y}_1(b), \ldots, \tilde{y}_M(b)$. The wideband output signal $y(n)$ is obtained by applying the *synthesis filter bank* (SFB) to the subband signals. In the following sections, all necessary components of the codec structure and the evaluation of the proposed coding approach are presented.

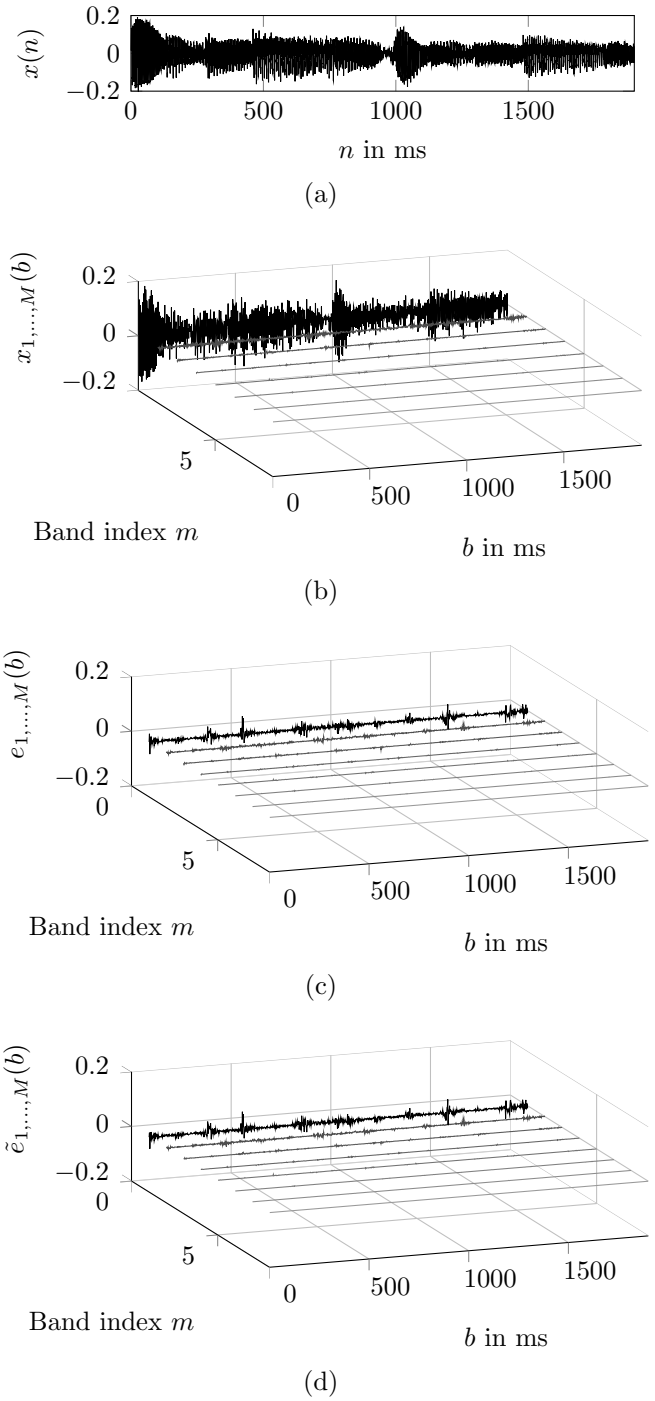


Figure 3.3: Signals involved in the encoding process: a) Broadband input signal $x(n)$, b) subband signals $x_{1,\ldots,M}(b)$, c) subband residuals $e_{1,\ldots,M}(b)$, and d) quantized subband signals $\tilde{e}_{1,\ldots,M}(b)$.

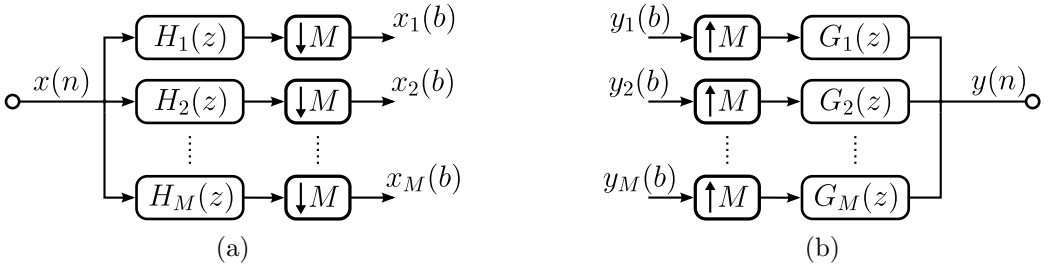


Figure 3.4: Blockscheme of analysis (a) and synthesis (b) filter bank.

3.2 Filter bank

Dividing a broadband signal $x(n)$ into several critically sampled subband signals $x_1(b), \ldots, x_M(b)$ with the help of a filter bank, as shown in Fig. 3.4, to allow individual processing of the corresponding frequency components is a classical digital signal processing task. Many designs and implementations of filter banks are known and utilized in various applications. In the case of SB-ADPCM, subband processing is advantageous since the predictor in every band can be optimally adjusted for characteristics of the audio signal within a certain frequency range. For example, it is expected that low-frequency bands mainly consist of harmonic stationary sounds whereas high-frequencies tend to have a noisier characteristic. Figure 3.5 shows the spectrogram of a snippet from the SQAM viola sample for $M = 4$ subbands. Apparently, the lowest band $m = 1$ shows the strongest tonality whereas the upper bands look noisier.

Therefore, a predictor operating in a low-frequency band is likely to benefit from a high order and a slow adaption. The prediction error energy of a lattice predictor of order $p = 80$ and base step size $\lambda = 0.001$ and a second predictor of order $p = 30$ and base step size $\lambda = 0.3$ is plotted in Fig. 3.5 for the SQAM viola example. In Band $m = 1$, the energy of the higher-order slow predictor E^1_{es} is significantly below the energy of the actual signal E^1_x and hence a clear prediction gain

$$PG = 10 \cdot \log 10 \left(\frac{\sum_{n=0}^{N-1} x^2(n)}{\sum_{n=0}^{N-1} e^2(n)} \right) \tag{3.1}$$

can be achieved. The error energy of the fast predictor E^1_{ef} is even higher than the energy of the actual signal. However, in subband $m = 4$ a clearly decreased error energy can be obtained by applying the fast predictor, whereas the slow predictor does not achieve any prediction gain. Thus, the average

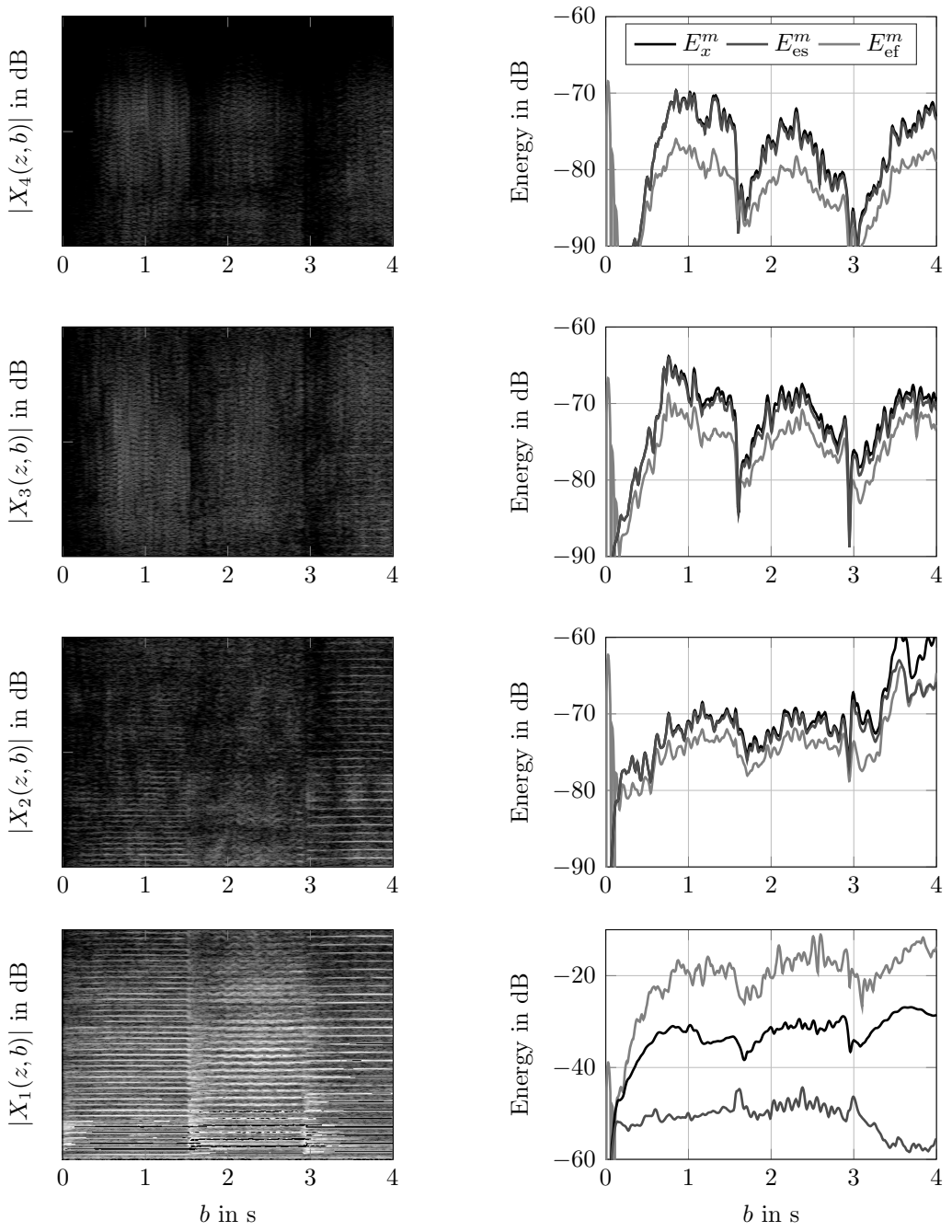


Figure 3.5: Subband spectrograms, subband energy E_x^m, and subband prediction error energies $E_{es/ef}^m$ for the SQAM viola sample for $M = 4$ using a slow and a fast predictor.

prediction gain can be maximized by applying individually adjusted predictors.

If the subbands are equally spaced and hence feature the same normalized bandwidth $\frac{\pi}{M}$, the subband signals can be subsampled without loosing any information. If the subsampling factor is equivalent to the amount of bands M the corresponding filter bank is denoted critcally sampled.

3.2.1 Cosine-Modulated Filter Bank

A popular way to design a filter bank is to derive the analysis filters $H_m(z)$ and synthesis filters $G_m(z)$ by modulating a prototype lowpass filter $H_P(z)$ of length N

$$h_m(n) = 2\,h_p(n)\cos\left((2m+1)\frac{\pi}{2M}\left(n-\frac{N-1}{2}\right)+\frac{\pi}{4}(-1)^m\right) \quad (3.2)$$

$$g_m(n) = 2\,h_p(n)\cos\left((2m+1)\frac{\pi}{2M}\left(n-\frac{N-1}{2}\right)-\frac{\pi}{4}(-1)^m\right). \quad (3.3)$$

The modulation corresponds to a shift in frequency domain and hence M equally spaced bandpass filters are obtained. Analysis and synthesis impulse responses only differ in the sign of the phase offset $\frac{\pi}{4}(-1)^m$ of the cosine. This is equivalent to a time reversal of the impulse response

$$g_m(n) = h_m(N-1-n), \quad (3.4)$$

corresponding to

$$G_m(z) = z^{-(N-1)}H_m\left(z^{-1}\right) \quad (3.5)$$

in the z-Domain. The overall transfer function of band m then reads

$$H_m(z)G_m(z) = z^{-(N-1)}H_m(z)H_m\left(z^{-1}\right). \quad (3.6)$$

It can be seen that the analysis and synthesis filters feature linear phase when combined. In other words, the overall filter bank group delay is constantly $N-1$ samples although $H_m(z)$ and $G_m(z)$ are not necessarily linear phase filters.

The cosine modulation is illustrated in Fig. 3.6, where a lowpass prototype and all derived bandpass filters can be seen in form of their impulse responses and transfer functions.

3.2.2 Prototype Design

As mentioned before, a prototype lowpass filter must be designed to realize a cosine-modulated filter bank. Several parameters of the lowpass will

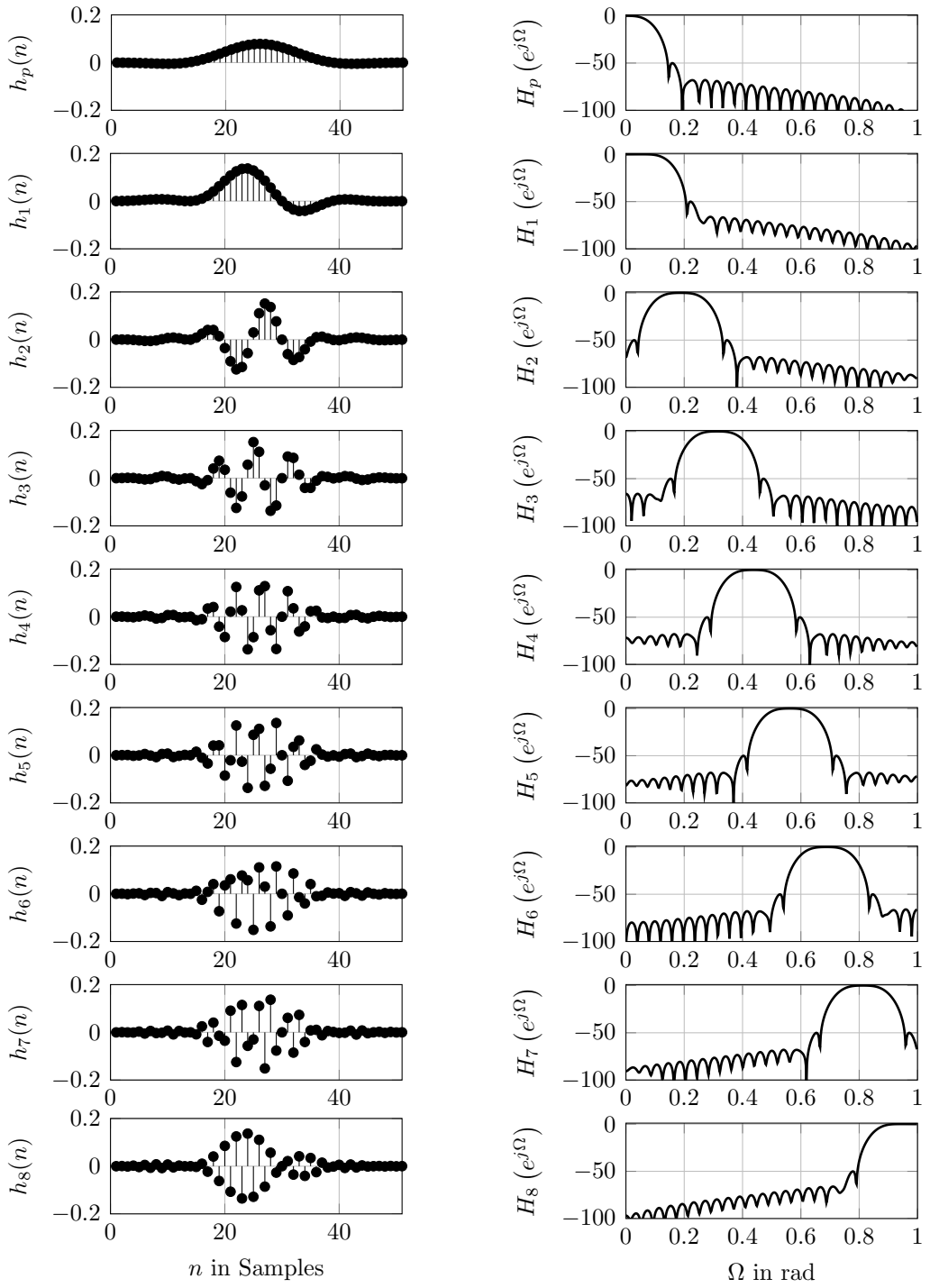


Figure 3.6: Impulse responses and transfer functions of prototype filter $h_p(n)$ and the derived analysis filters $h_m(n)$ for $M = 8$ and $N = 51$.

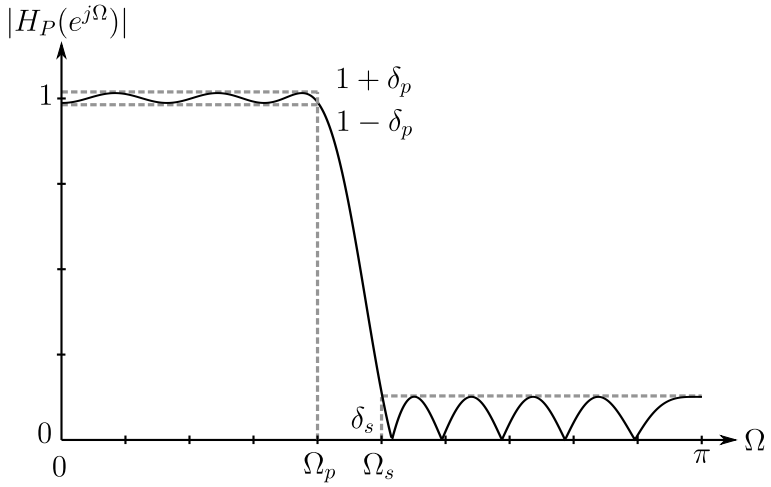


Figure 3.7: Exemplary prototype filter and relevant design parameters.

have a strong impact on the resulting filter bank performance. The amplitude response $H_P(e^{j\Omega})$ is plotted in Fig. 3.7. It consists of 3 major sections: the passband with minor or without attenuation, the transition part, and the stopband where strong attenuation is achieved. The relative transition bandwidth $b = \frac{\Omega_s - \Omega_p}{2\pi}$ is restricted by the passband edge Ω_p and the stopband edge Ω_s. The amplitude deviation from the ideal values is called passband ripple δ_p and stopband ripple δ_s, respectively. Several possibilities for the design shall be compared in the following.

Window Design

The optimal lowpass with an infinitely sharp transition ($\Omega_p = \Omega_s$) and neither passband nor stopband ripple ($\delta_p = \delta_s = 0$) can be described with a rectangle in the frequency domain

$$H_P(e^{j\Omega}) = \text{rect}\left(\frac{\Omega}{2\,\Omega_c}\right) \circ\!\!-\!\!\bullet \quad h(n) = \frac{\sin(\Omega_c n)}{\pi n} \tag{3.7}$$

corresponding to the normalized sinc function in the time domain. The impulse response of infinite length prohibits the utilization of the optimal lowpass. Therefore, the impulse response is limited in its length using a window $w(n)$. Typical windows in audio processing are the Sine-, Hamming, and Blackman-window. All mentioned windows are illustrated in Fig. 3.8a) for a length of $N = 64$ samples. The corresponding frequency responses in Fig. 3.8b) share the same cutoff frequency $\Omega_c = \frac{\pi}{4}$ but differ significantly in their main lobe width and their stopband attenuation.

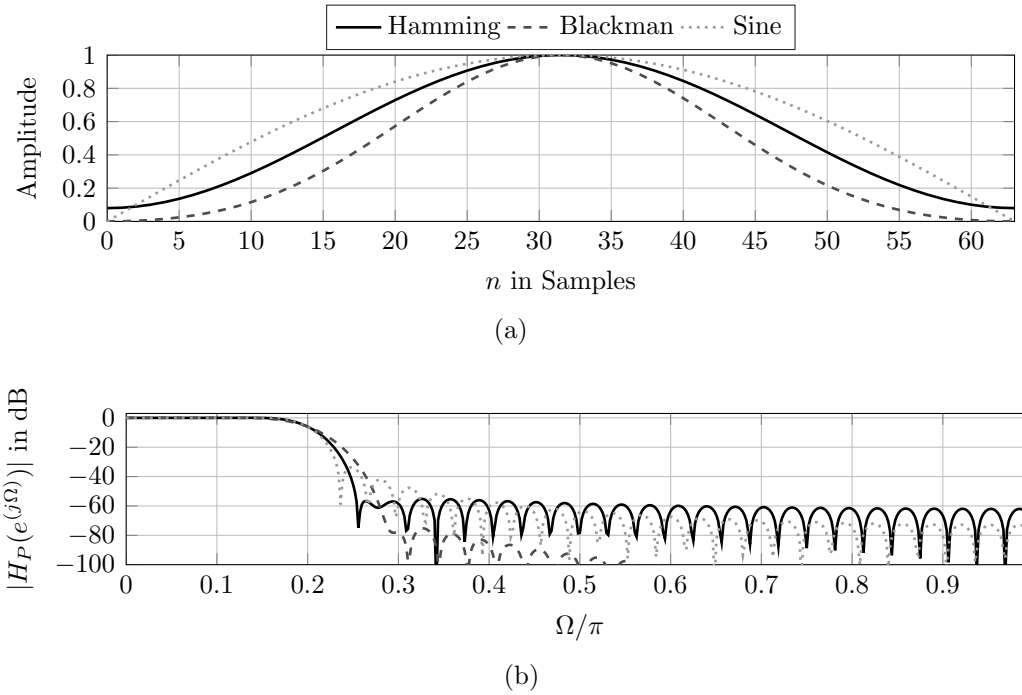


Figure 3.8: Window functions and frequency responses of resulting prototype lowpass filter.

ParksMcClellan Design

Another standard method for FIR filter design is the Parks-McClellan algorithm [PM72]. It obtains the optimum chebyshev approximation, corresponding to an equiripple design, by minimizing the error in passband and stopband iteratively using the Remez exchange algorithm. The algorithm requires the filter order N, passband edge Ω_p, and the stopband edge Ω_s. The frequency response deviations δ_p and δ_s can not be directly respected but weighting factors w_p and w_s are used to priorize smaller deviations in either passband or stopband [CM95].

3.2.3 Power-complementary Filter Bank

A filter bank should be designed in a way to allow perfect reconstruction if no processing occurs in the subbands. Certain design methods are known to achieve reconstruction without amplitude or aliasing distortion but are bound to certain filter lengths [NV90, Vai87]. It is also possible to design near perfect reconstruction filter banks with arbitrary length using optimization techniques.

Creusere and Mitra [CM95] iteratively minimized the cost function

$$\phi = \max_{\Omega}\{\left|H(e^{j\Omega})\right|^2 + \left|H(e^{j\Omega-\frac{\pi}{M}})\right|^2 - 1\}, \quad 0 \leq \Omega \leq \frac{\pi}{M} \tag{3.8}$$

by varying the passband edge of the remez exchange design with a control loop. This approach was modified by Keiler [Kei06a] by extending the control loop with an inner loop to additionally optimize the weighting factor $\frac{w_p}{w_s}$ to further minimize the cost function. The optimization scheme from [CM95] is also applicable to the FIR window design method.

3.2.4 Evaluation of Filter Banks

To describe the performance of a filter bank an evaluation metric is required. Therefore, the distortion and aliasing function from [Fli93] shall be introduced. Applying one of the analysis filters $H_m(z)$ to the input signal $X(z)$ leads to the subband signal

$$X_m(z) = H_m(z) \cdot X(z) \tag{3.9}$$

which is still sampled at the original rate but restricted to a bandwidth of π/M as shown in Fig. 3.9 a+b). The critically sampled subband signal

$$X_m(z^M) = \frac{1}{M}\sum_{l=0}^{M-1} H_m(zW_M^l) \cdot X_m(zW_M^l), \quad W_M = e^{-j\frac{2\pi}{M}} \tag{3.10}$$

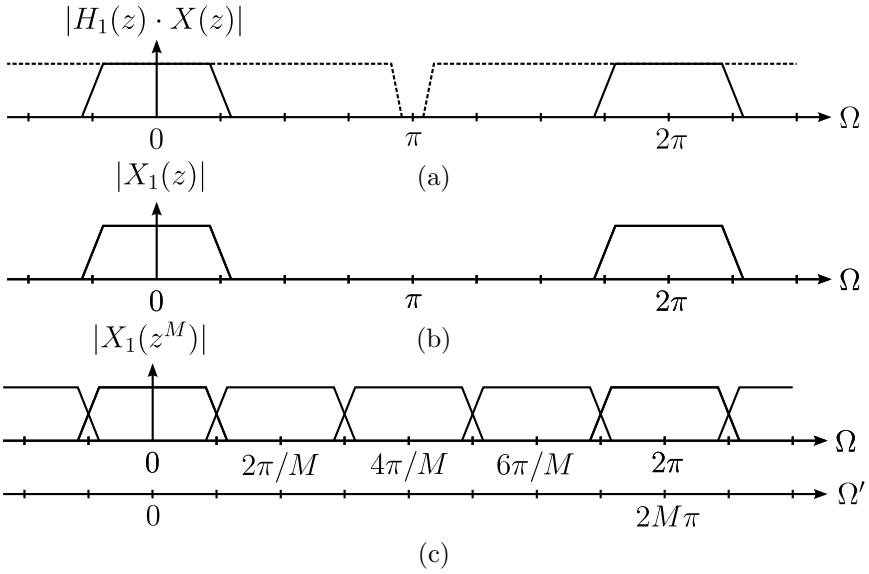


Figure 3.9: Computation of critically sampled subband signals: a) Filter input $X(z)$ with subband filter $H_m(z)$ to get b) high-rate subband signal $X_m(z)$, c) followed by subsampling.

features repetitions of the original spectrum with a periodicity of $\frac{2\pi}{M}$ instead of the previous periodicity of 2π as apparent in Fig. 3.9c). Applying the synthesis filter bank and adding the corresponding outputs leads to the overall filter bank output $Y(z)$, denoted as

$$\begin{aligned} Y(z) &= \frac{1}{M}\sum_{m=1}^{M} G_m(z)\cdot X_m(z^M) \\ &= \frac{1}{M}\sum_{m=1}^{M} G_m(z)\cdot \sum_{l=0}^{M-1} H_m(zW_M^l)\cdot X(zW_M^l) \\ &= \frac{1}{M}\sum_{m=1}^{M}\left[\sum_{l=0}^{M-1} G_m(z)\cdot H_m(zW_M^l)\right]\cdot X(zW_M^l). \end{aligned} \tag{3.11}$$

When the aliasing components are ignored ($l = 0$) the overall filter bank distortion function can be found

$$F_{\text{dist}}(z) = \frac{1}{M}\sum_{m=1}^{M} G_m(z)\cdot H_m(z). \tag{3.12}$$

The remaining aliasing distortion is then defined as

$$F_{\text{alias}}(z) = \sqrt{\sum_{l=1}^{M-1} \left| \frac{1}{M} \sum_{m=1}^{M} G_m(z) \cdot H_m(zW_M^l) \right|^2}. \tag{3.13}$$

In addition to analysis of the reconstruction properties of the filter bank from [Fli93] it is essential to evaluate the filter bank in terms of selectivity of the pass band. This can be achieved by evaluating the stopband attenuation of the prototype filter $H_P(e^{j\Omega})$. Unfortunately, the stopband features an irregular character with ripples as can be seen in Fig. 3.8b). Therefore, the maximum stopband surpression $F_{\text{stop,max}}$ is measured by evaluating the amplitude of $H_P(e^{j\Omega})$ at the maximum of the first ripple. Another way of evaluating the band selectivity is comparing the magnitude ratio

$$F_{\text{ratio}} = \frac{\frac{1}{\pi-\Omega_c} \int_{\Omega=\frac{\pi}{2M}}^{\pi} |H_P(e^{j\Omega})| d\Omega}{\frac{1}{\Omega_c} \int_{\Omega=0}^{\Omega_c} |H_P(e^{j\Omega})| d\Omega}, \quad \Omega_c = \frac{\pi}{2M} \tag{3.14}$$

of the prototype's stop- and pass-band. A value of $-\infty$ dB indicates perfect band separation whereas 0 dB denotes an equal split of energy between pass- and stop-band.

Every introduced evaluation metric shall be applied to the filter bank design according to 3.2.3 using window-designed prototypes. The overall filter bank distortion for a filter bank designed using a Hamming window prototype with $M = 8$ bands and a length of $N = 51$ is plotted in Fig. 3.10. Apparently, the filter bank is close to optimal due to the maximum value of $F_{\text{dist,max}} = 1.0084$. The alias distortion of the filter bank yields a maximum value $F_{\text{alias,max}}$ as low as -50 dB.

To identify advantageous windows and filter lengths for the purpose of designing the analysis and synthesis filter bank, $F_{\text{dist,max}}$ and $F_{\text{alias,max}}$ are plotted against different filter lengths and windows in Fig. 3.11 a+b). The maximum distortion roughly falls smoothly over the order for the Hamming and Blackman window. In contrast the distortion value soars for the sine window for filter lengths $N \geq 50$, where the Hamming windows already converges to the first minimum. The smallest aliasing distortion is obtained for higher-order Blackman windows. However, the $F_{\text{alias,max}}$ values for the Hamming window falls below the Blackman values for $46 \geq N \geq 56$. The maximum stopband attenuation $F_{\text{stop,max}}$, plotted in Fig. 3.11c), is almost constant and therefore can't be used to rate the selectivity of the prototype

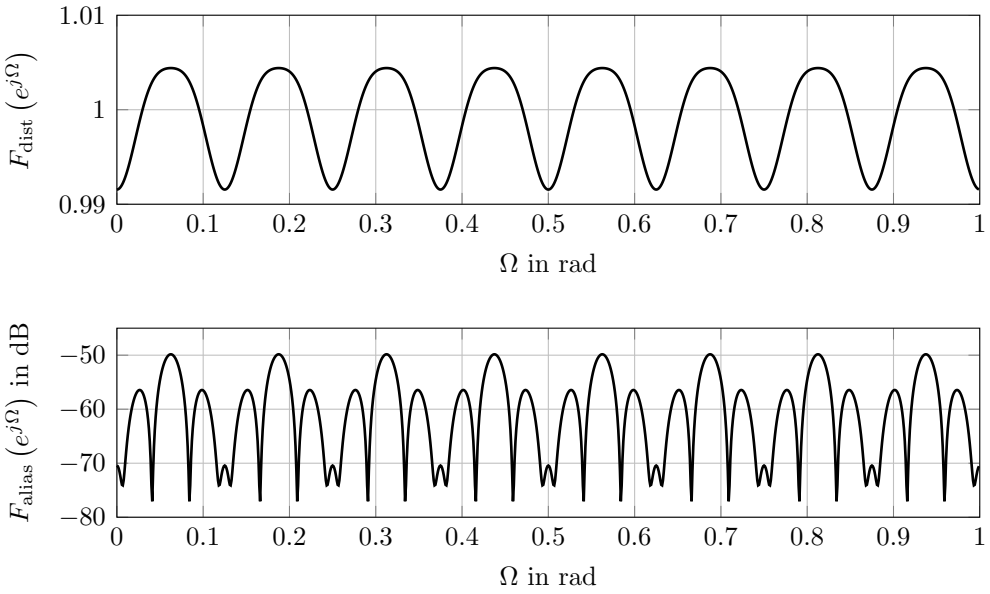


Figure 3.10: Distortion F_{dist} and aliasing distortion F_{alias} of a filter bank with $M = 8$ and $N = 51$.

since the amplitude of the first ripple starts decreasing for higher orders. Therefore, the stop-to-pass-band ratio F_{ratio} is additionally computed and illustrated in Fig. 3.11d). The starting value of $\approx -30\,\text{dB}$ already indicates a strong concentration of energy in the pass band which is increased for increasing filter order. This trend holds true for all utilized windows. In the following, a Hamming-based prototype of length $N = 51$ is used. The Hamming window is chosen due to its low overall and aliasing distortion within the range $N \in [46, \ldots, 56]$ corresponding to a desirable small algorithmic delay of $\approx 1\,\text{ms}$. The stopband attenuation and stopband-to-passband ratio results are in between the results of the Blackman and Sine window and therefore, are the best tradeoff between filter selectivity and stopband attenuation.

3.3 Backward-Adaptive Lattice Prediction

The basic principle of ADPCM is to solely transmit quantized irredundant signal components. The corresponding redundant signal components are extracted using a prediction filter $P(z)$ of order p by first computing the prediction signal

$$\hat{x}(n) = \sum_{i=1}^{p} a_i x(n-i) \tag{3.15}$$

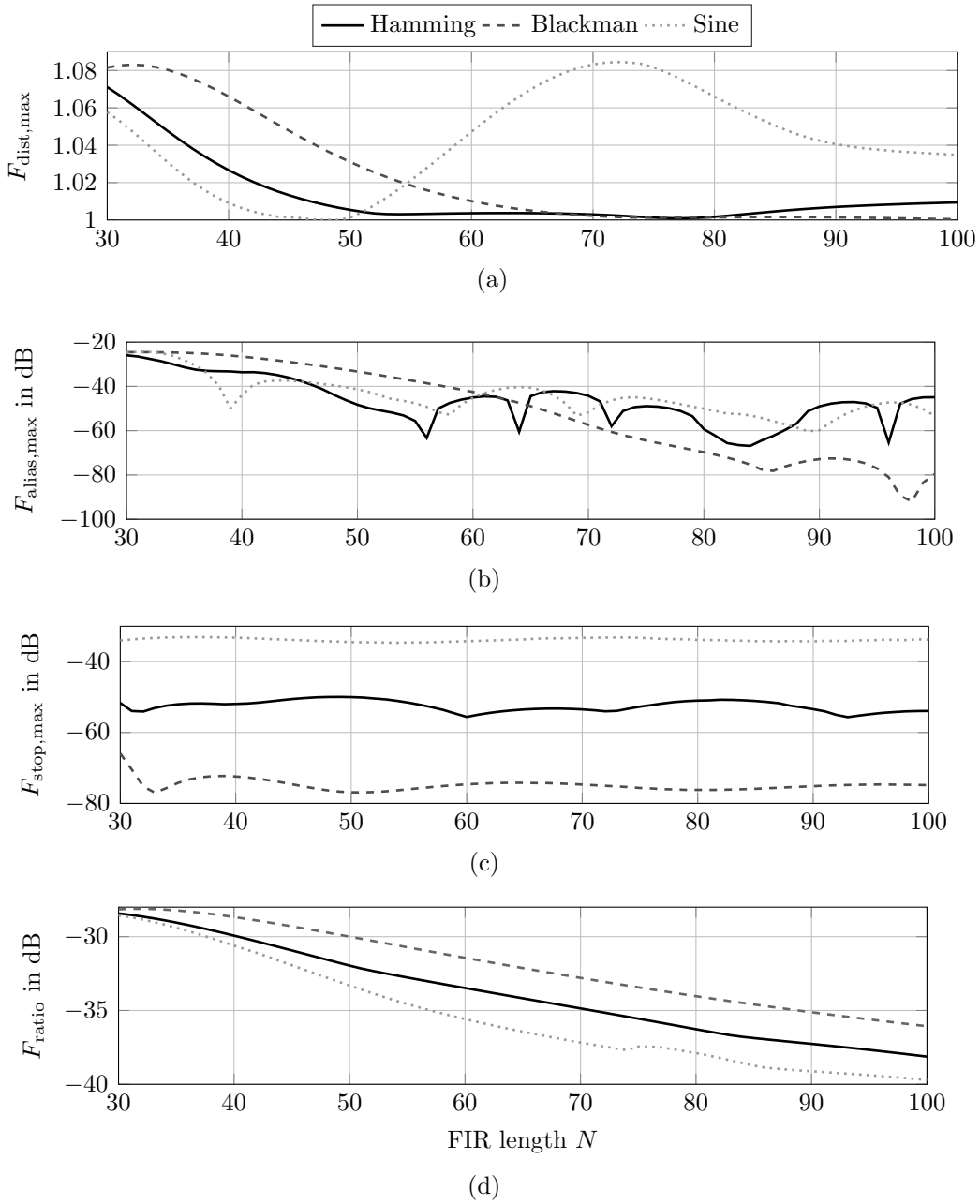


Figure 3.11: Distortion F_{dist}, aliasing distortion F_{alias}, stopband attenuation F_{stop}, and stop-to-pass-band ratio F_{ratio} of window-designed filter bank with $M = 8$ plotted over filter length N.

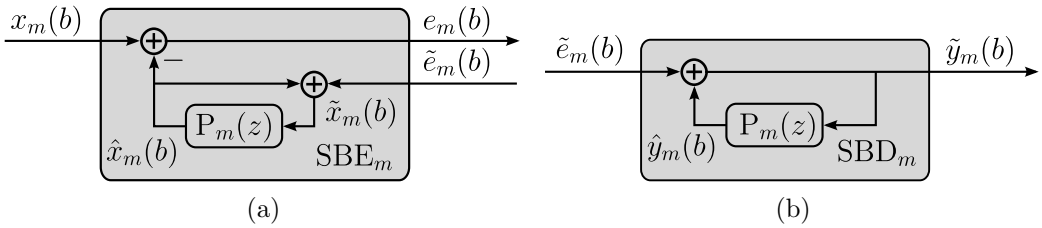


Figure 3.12: ADPCM encoder (a) and decoder (b).

based on prior values of $x(n)$ and then computing the residual

$$e(n) = x(n) - \hat{x}(n) = x(n) - \sum_{i=1}^{p} a_i x(n-i), \tag{3.16}$$

which then is further quantized to reduce the data rate of the signal. The signal reconstruction in the decoder is achieved using the inverse prediction filter $G(z) = \frac{1}{P(z)} = \frac{1}{1-\sum_{i=1}^{p} a_i z^{-1}}$. To guarantee a stable reconstruction filter the prediction filter is required to have minimum phase characteristic. This property is always fulfilled by the class of lattice filters (Fig. 2.4, Eq. (2.15)) when the reflection coefficients are bound to $|k_p| \leq 1$ [Mak78].

In contrast to the concealment method based on lattice filters and the GAL algorithm of Sec. 2.1.1, which is solely applied at the decoder side in the case of transmission problems, the prediction filter is applied in the *subband encoder* (SBE) and in the *subband decoder* (SBD) as shown in Fig. 3.12. The block scheme of the SBE also shows that the prediction is performed using the reconstructed subband value

$$\tilde{x}_m(b) = \hat{x}_m(b) + \tilde{e}_m(b), \tag{3.17}$$

which is the sum of subband prediction signal $\hat{x}_m(b)$ and quantized residual $\tilde{e}_m(b)$ instead of the original subband signal $x_m(b)$. Thus, the SBE and SBD operate synchronously when the prediction and GAL prediction filter adaption is performed in this backward manner. The SBD solely receives the quantized subband residual $\tilde{e}_m(b)$ which is added to the prediction signal $\hat{y}_m$ to compute the reconstructed subband signal $\tilde{y}_m = \tilde{x}_m$, which is identical to the reconstructed subband value in the encoder.

3.4 Vector Quantization

The reduction of data rate achieved by audio codecs is mainly caused by further quantizing the digital audio signal. Quantization can be described

as the process of mapping an (infinite) set of amplitude values to a smaller restricted set. This set is captured in a codebook $\mathbf{C}$ of size $L \times 2^w$ where w denotes the word length. A signal vector $\mathbf{x} = [x_i, \ldots, x_L]$ can therefore be represented using the codebook entry $\mathbf{C_i}$ at index i. Solely the index i is necessary to reconstruct the quantized vector at the receiver side since the codebook is known in encoder and decoder.

The selection of a codebook entry $\mathbf{C_i}$ is typically done by minimizing the error power

$$\min_i \left[(\mathbf{x} - \tilde{\mathbf{x}}_i)^T(\mathbf{x} - \tilde{\mathbf{x}}_i)\right], \quad i \in [1, \ldots, 2^w] \tag{3.18}$$

between $\mathbf{x}$ and every codebook vector $\tilde{\mathbf{x}}_i$. Quantizers are called scalar quantizers for $L = 1$ and vector quantizer for $L > 1$. An increasing value of L is advantageous in several ways. First, vector quantizers allow a finer resolution of the effective word length $\frac{w}{L}$, whereas scalar quantizers are bound to an integer resolution. In addition, a vector quantizer exploits correlation within a data vector and hence converges to the rate-distortion limit [GC83].

The characteristic curve of a uniform scalar 4 bit quantizer is shown in Fig. 3.13a). Apparently, the amplitudes in the interval $x(n) \in [-1, \ldots, 1]$ are mapped to a set of values $\tilde{x}(n)$ with a uniform step size

$$\Delta = 2^{-w+1}. \tag{3.19}$$

A uniform step size is only optimal in the MMSE sense for signals featuring a uniform *probability density function* (PDF). Unfortunately, audio signals are typically modeled and described using the Laplacian and the Gaussian distribution [GZ03]. These distributions are very condensed around their center and hence large amplitudes are less likely to appear. Historically, signals were compressed, uniformly quantized, and expanded to match the signal characteristic to the quantization process. Another solution is to directly adapt the step sizes to the PDF of the signal [Max60]. Figure 3.13b+c) show the characteristic curve of a 4 bit quantizer optimized for a Laplacian distribution with a variance of $\sigma^2 = 0.125$ and for Gaussian distribution with a variance of $\sigma^2 = 0.25$, respectively. The smaller step sizes and therefore higher resolution for small amplitudes can clearly be noticed. An example of a VQ with a dimensionality of $L = 2$ and word length $w = 4$ is shown in Fig. 3.14 in form a Voronoi diagram. The 2D space is partitioned into 2^w cells, also known as Voronoi polygons, enclosing the centroids $\mathbf{C}_i$. It can be seen that every point within a cell is closer to the cell-defining point $\mathbf{C}_i$ than to any other. The already mentioned advantage of a VQ is illustrated in Fig. 3.15a+b). The gray dots are samples of a two-dimensional Laplacian distribution. The orientation along the diagonal axis indicates a certain correlation between the samples of the different dimensions. The quantization levels of the scalar

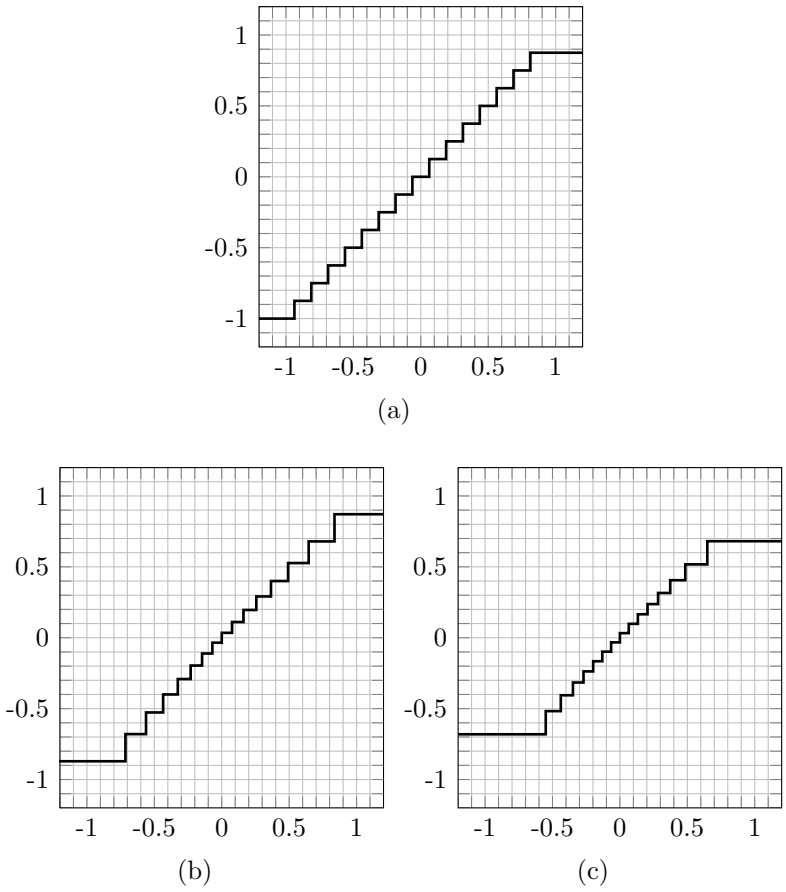


Figure 3.13: Characteristic curves of different scalar $w = 4$ bit quantizers: Uniform quantizer a), Lloyd-Max quantizer optimized for Laplacian distribution with a variance of $\sigma^2 = 0.125$ b), and for Gaussian distribution with a variance of $\sigma^2 = 0.25$ c).

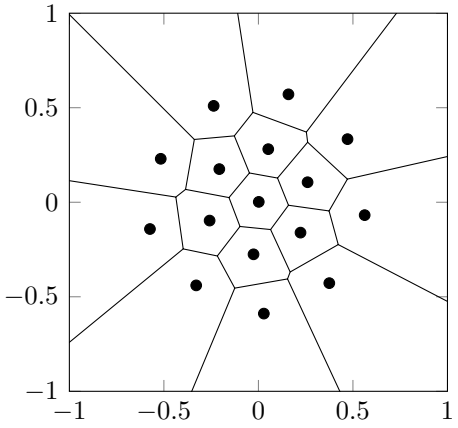


Figure 3.14: Voronoi diagram of two-dimensional vector quantizer with $w = 4\,\text{bit}$.

quantizers which are matched to the PDF of a particular dimension form a regular grid on the x-y plane in Fig. 3.15 a). In contrast, the quantization levels of the VQ in Fig. 3.15 b) represent the shape of the samples superiorly. Therefore, the resulting quantization noise is significantly smaller.

Considering the block scheme of the proposed encoder in Fig. 3.1 it becomes apparent that the proposed encoder feeds the vector of subband residuals $\mathbf{e}(b) = [e_1(b), \ldots, e_M(b)]$ for a certain block b to the vector quantizer to obtain the quantized residual vector $\tilde{\mathbf{e}}(b)$. Hence, the dimensionality of the vector quantizer is assumed to be equal to the amount of subbands $L = M$.

3.4.1 Adaptive Vector Quantization

The fixed optimization of a quantizer to a certain PDF is not optimal due to varying signal characteristics of typical input signals. Instead of directly quantizing the residual vectors, the author decided to quantize the normalized residual

$$\bar{e}_m(b) = \frac{e_m(b)}{v_m(b)}, \tag{3.20}$$

which is computed using an recursive envelope estimate [HHZ08]

$$v_m(b) = \sqrt{(1 - \lambda + \lambda \cdot \tilde{e}_m^2(b-1)) \cdot v_m^2(b-1)} \tag{3.21}$$

$$\lambda = \begin{cases} \lambda_{\text{AT}}, & \text{if } \tilde{e}_m^2(b-1) > 1 \\ \lambda_{\text{RT}}, & \text{else} \end{cases} \tag{3.22}$$

to realize an almost constant signal variance [CM75]. The envelope estimation is signal-adaptive since it involves two smoothing coefficients λ_{AT} and

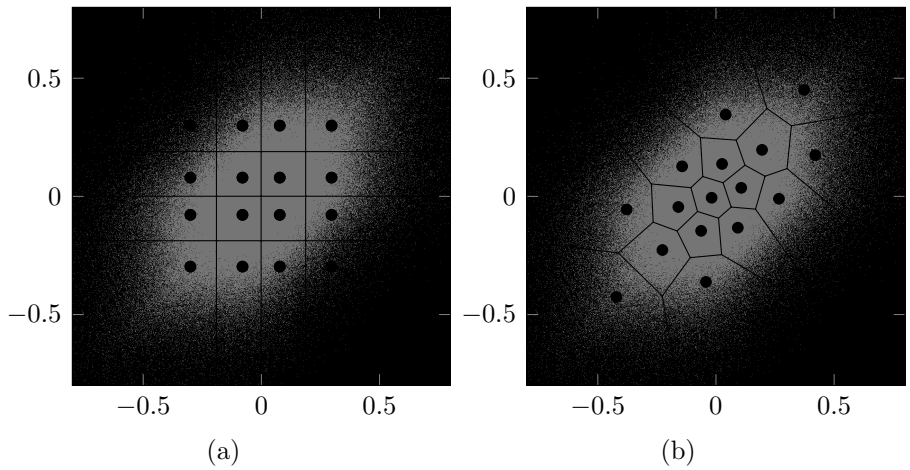


Figure 3.15: Quantization levels of two-dimensional scalar and vector quantizer with $w = 4\,\text{bit}$ on top of two-dimensional correlated Laplacian distribution.

λ_{RT} for the attack and release case, respectively. Envelope amplitudes smaller than $v_m < v_{\min}$ are clipped to $v_{\min}$.

The codec implementation benefits from the computed envelope twice. In addition to the adaptive quantization, the codebook search can be optimized by weighting the cost function according to the subband envelopes

$$\min_i \left[(\mathbf{e} - \tilde{\mathbf{e}}_i)^T \operatorname{diag}(\mathbf{v})(\mathbf{e} - \tilde{\mathbf{e}}_i)\right]. \tag{3.23}$$

The weighting emphasizes subbands with large amplitudes and hence psychoacoustically relevant parts which are typically the lower frequency bands. The effect of the coarse quantization in high-frequency bands is a rise of quantization noise resulting in a degraded subband SNR. However, the improved subband SNR in lower frequency bands is expected to be perceptually more relevant. The band-wise SNR after vector quantization of the SQAM viola example using a white noise codebook with and without weighting the cost function is shown in Fig. 3.16. The SNR in band $m = 1$ rises almost $30\,\text{dB}$ whereas the gain in band $m = 2$ still yields $14\,\text{dB}$. The SNR in the higher bands decreases by up to $-15\,\text{dB}$.

3.4.2 Nearest Neighbor Search

The major drawback of vector quantization is the large complexity of searching the codebook for the best-fitting codebook entry for the current data to

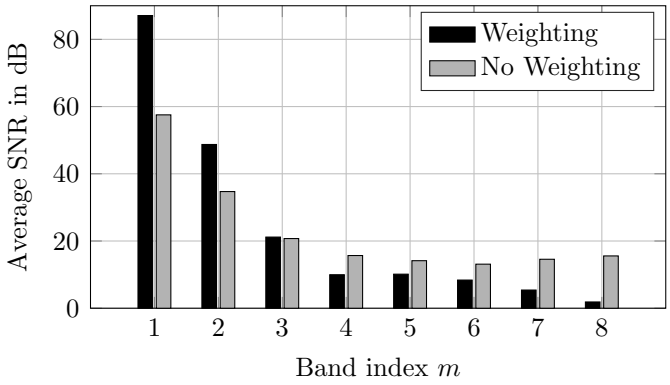


Figure 3.16: Bandwise average SNR for the SQAM viola example w/o weighting the cost function.

be quantized. Hence, the choice of the codebook size suffers from the trade-off between the SNR gain for higher dimensionality and the corresponding increased searching complexity. A linear search in the codebook requires the computation of 2^w vector differences of length L. This corresponds to $L \cdot 2^w$ subtractions, multiplications, and additions in the case of the plain euclidean distance. In the case of $M = 8$ subbands and $w = 16$, corresponding to 2 bits per sample, more than 1.5 million arithmetic instructions are necessary.

The searching complexity can be massively reduced using the *Nearest Neighbor Search* (NNS) algorithm. The pseudo code of the general procedure can be seen in Alg. 2. Instead of linearly searching the codebook entry yielding the smallest distortion, the NNS approach intends to compare K neighbors with index i of the current entry $C_{i_{min}}$ as can be seen in $\mathrm{l.}\,4-7$. The neighbors have to be available in form of a reference book $\mathbf{N_n}(k, i_{min})$. The neighbor $\mathbf{N_n}(k, i_{nn})$ showing the smallest distortion d_{nn} is then used as the current codebook index i_{min} (l. 13). The globally smallest distortion d_{min} is also replaced (l. 12). The routine terminates after l_{max} repetitions or in the case of no further smallest neighbors (l. 9). The NNS is initialized with the first codebook entry and the corresponding distortion is set as the globally smallest distortion $(\mathrm{l.}\,1-2)$.

The algorithm is visualized in Fig. 3.17. The codebook $\mathbf{C}$ of size 2×2048 is shown in form of dots on an euclidean plane. The vector to be quantized at $[-1.3, -1.9]$ is shown as a black diamond. The initial codebook entry $\mathbf{C}_{i_{min}}$ (white diamond) is apparently surrounded by the $K = 100$ neighbors (grey squares). Initially at $l = 0$, the nearest neighbors are focused around the center of the sphere. The cloud of nearest neighbors is further traveling in direction of the given vector for increasing iterations until the best-fitting codebook entry is found after $l = 7$ iterations. The distortion is iteratively

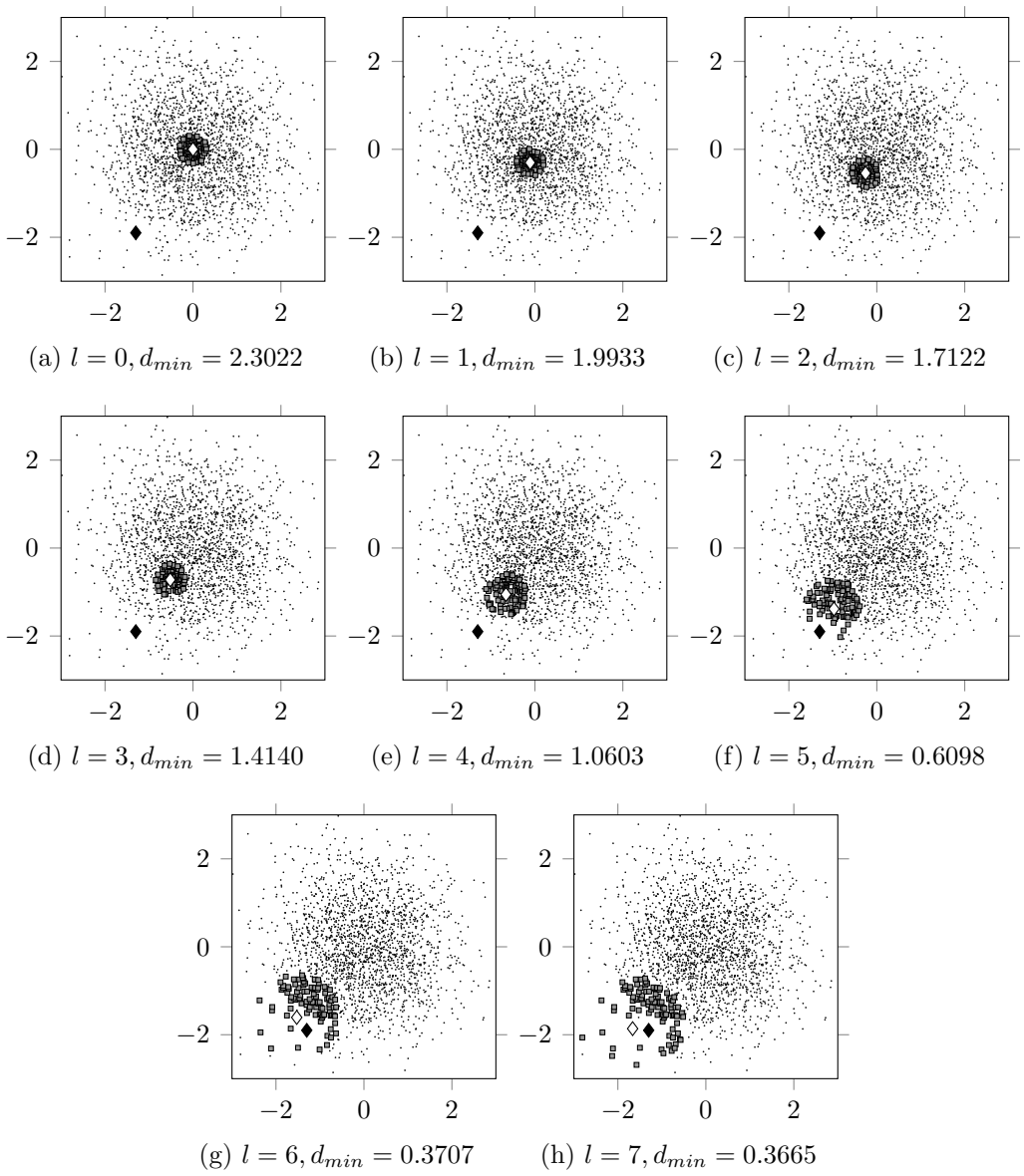
reduced from 2.3022 to 0.3365. Only 801 vector comparisons instead of 2048 had to be computed in this example. The saving of computations is increasing for a rising dimensionality of the codebook. Encoding the entire SQAM data set using the proposed encoder and a vector quantizer with a codebook size of $8 \times 2^{2.8}$ and $K = 100$ averages to 3.27 iterations. The actual discrete probability distribution is shown in Fig. 3.18. Almost all NNS operations require less than $l = 5$ iterations and hence less than 500 comparisons instead of 65536. This massive reduction of complexity allows the coding approach to be performed in real time.

Algorithm 2 *Nearest neighbor search algorithm.*

$i_{min} = 1$
$d_{min} = \left[(\mathbf{e} - \mathbf{C}_{i_{min}})^T \mathrm{diag}(\mathbf{v})(\mathbf{e} - \mathbf{C}_{i_{min}})\right]$
for $\mathrm{l} = [1, \ldots, \mathrm{l}_{max}]$ **do**
    for $k = [1, \ldots, K]$ **do**
        $i = \mathbf{N}_\mathbf{n}(k, i_{min})$
        $d(k) = \left[(\mathbf{e} - \mathbf{C}_i)^T \mathrm{diag}(\mathbf{v})(\mathbf{e} - \mathbf{C}_i)\right]$
    end for
    $[d_{nn}, i_{nn}] = \min(d)$
    if $d_{nn} > d_{min}$ **then**
        break
    else
        $d_{min} = d_{nn}$
        $i_{min} = i_{nn}$
    end if
end for

3.4.3 Entropy Coding

The bitrate of the proposed encoder amounts to 88.2 kbit/s for $f_s = 44.1$ kHz and $w = 2$ bits per sample. But it can be further reduced by applying entropy coding. Whenever the input signal shows a certain amount of stationarity and the subband predictors are properly adjusted to remove the majority of the subband signals redundancy, the resulting subband residuals are expected to be very small. These small residuals are typically represented with codebook entries with a small index since the codebook is sorted according to its euclidean norm. In other words, codebook entries, located in the center of the codebook sphere, are more likely to be utilized. This entropy can be exploited as shown in the following.



(a) $l = 0, d_{min} = 2.3022$ (b) $l = 1, d_{min} = 1.9933$ (c) $l = 2, d_{min} = 1.7122$

(d) $l = 3, d_{min} = 1.4140$ (e) $l = 4, d_{min} = 1.0603$ (f) $l = 5, d_{min} = 0.6098$

(g) $l = 6, d_{min} = 0.3707$ (h) $l = 7, d_{min} = 0.3665$

Figure 3.17: Illustration of the NNS search procedure requiring 8 iterations for a codebook size of 2×2048.

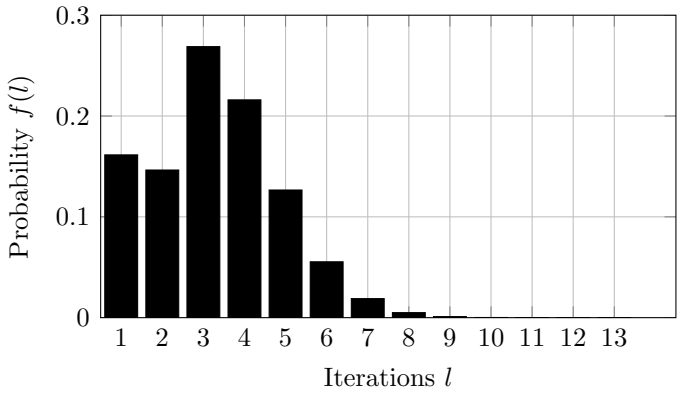


Figure 3.18: Empirical probability of required NNS iterations l when encoding the entire SQAM data set.

The SQAM data set is encoded using a white noise codebook. The resulting empirical codeword probability distribution p_i in Fig. 3.19 clearly shows the aforementioned characteristic. The corresponding source entropy

$$H = -\sum_i p_i \log_2 p_i \tag{3.24}$$

yields 11.54 bits whereas direct indexing of the codebook requires 16 bits. Based on the probabilities of Fig. 3.19, a Huffman encoder [Huf52] can be designed. The average word length of the resulting variable length code yields $\sum_i p_i \cdot w_i = 11.5727$ bits, where w_i is the word length for the code word representing codebook entry i. Figure 3.20 illustrates the exact word length distribution. As expected, it shows the inverse trend of the code word probabilities of Fig. 3.19. Apparently, this source encoding scheme is almost optimal for the probability distribution of the utilized codebook entries. Without modifying the core encoder or degrading the audio quality of the proposed encoder, the Huffman encoder reduces the average word length from 2 to 1.45 bits per sample. This corresponds to an average bitrate of about 64 kbit/s.

The actual resulting average bitrate for all SQAM items is plotted in Fig. 3.21. The bitrate is varying between 30.24 and 78.74 kbit/s and averages to 58.46 kbit/s. The lowest bitrates are obtained for simple sinusoid signals where as the highest bitrate is caused by noise. Hence, the bitrate is apparently dependent on the redundancy of the source signal.

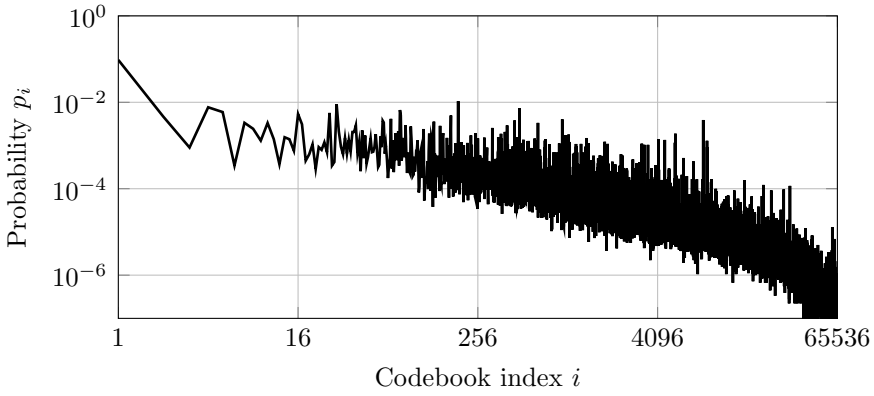


Figure 3.19: Empirical probability of codebook entry i appearance when encoding the entire SQAM data set.

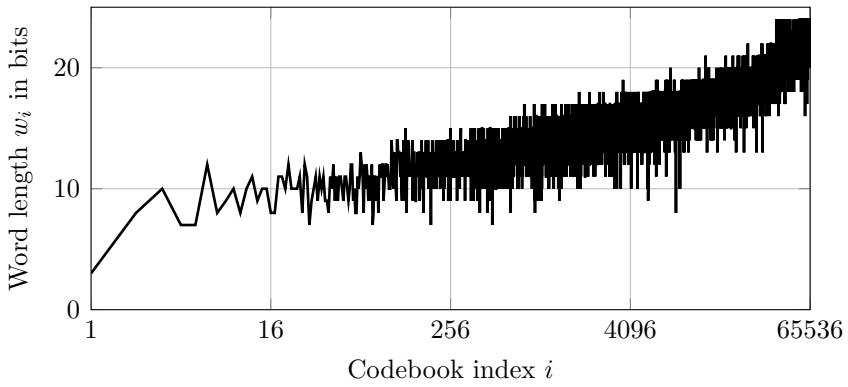


Figure 3.20: Word length w_i of codebook entry i.

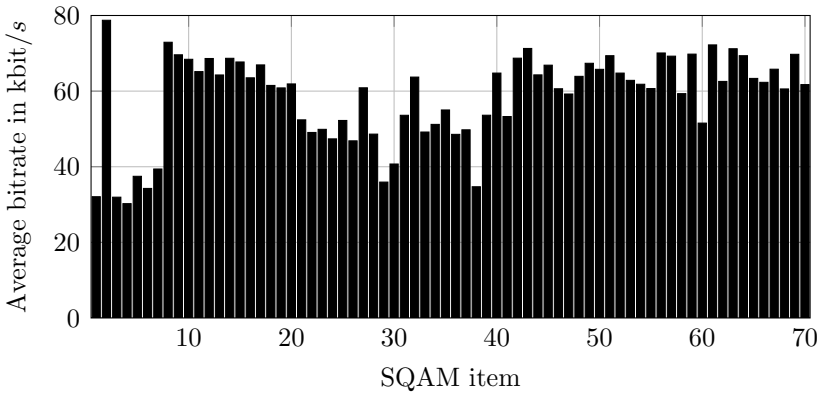


Figure 3.21: Average bitrate of SQAM items.

Table 3.1: Non-optimized codec parameters for the simple single parameter optimization of order p and base step size λ.

$v_{min} = \sigma_{min}$	λ_{AT}	λ_{RT}
$1.46494 \cdot 10^{-5}$	0.835779	0.0987227

3.5 Parameter Optimization

Before assembling the individual components of the proposed codec to evaluate the audio quality one has to find a set of parameters for the individual components. The major focus hereby is on adjusting the subband predictors which need to be parameterized in terms of prediction order p, base step size λ, stability constants σ_{min}, v_{min}, and smoothening coefficients $\lambda_{\mathrm{AT/RT}}$. Since the signal characteristics and hence prediction requirements differ drastically throughout the bands, as demonstrated in Sec. 3.2, different parameter settings are required in every subband.

3.5.1 Simple Iterative optimization

The optimization of parameters was performed in two steps. At first, the author wanted to gain insight in the range of the individual parameters in a simple way. Therefore, the cost function

$$J_{so}^{m}(p_m) = \frac{1}{F} \sum_{f=1}^{F} E_f^m, \quad \text{with} \quad E_f^m = \sum_{n=0}^{N_f-1} e_m^2(n) \tag{3.25}$$

describing the sum of squared prediction errors $e_m^2(n)$ for every test track f of the SQAM data set and a given prediction order p_m, is minimized to identify the optimal base step size λ_m for every subband m. The author utilized an adapted implementation of the optimization routine from [CM95] where the base step size λ_m is varied instead of the passband edge. The remaining codec parameters are set to the values for the ADPCM optimization using 3 bits from [HZ09] listed in Tab. 3.1. The cost function for the first band $m = 1$ and a set of predictor orders $p = [20, \ldots, 200]$ is illustrated in Fig. 3.22. The individual cost functions tend to be parabolically shaped and hence the simple optimization is always converging to the optimal solution. The optimal base step sizes $\lambda_1(p_1)$ are shown in form of the black curve whereas the globally best-performing combination of order 156 and base step size $1.45 \cdot 10^{-3}$ is indicated with the white marker. Apparently, higher order predictors tend to require smaller base step sizes to perform optimally and the prediction gain for predictors of large order are decreasing for $p > 156$.

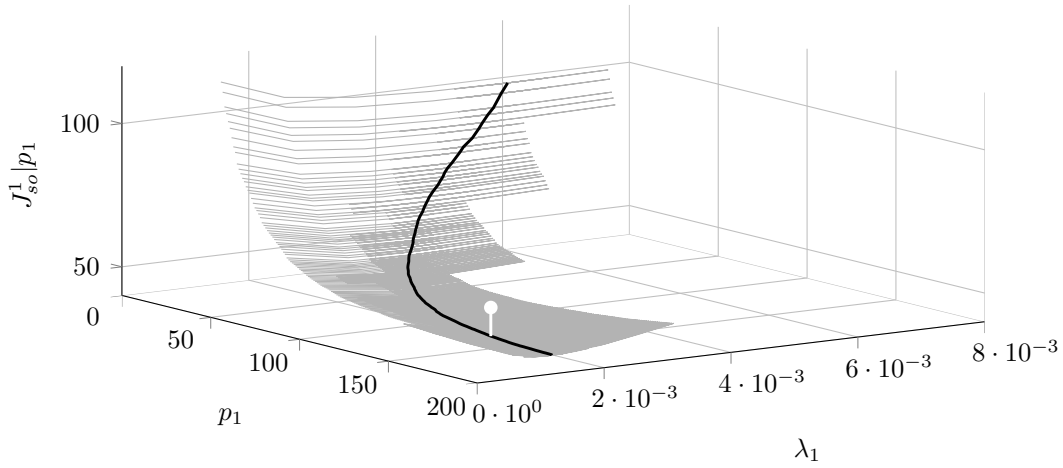


Figure 3.22: Simple optimization cost function $J^1_{so}(p_1)$ for band $m = 1$ and order $p_1 = [20, \ldots, 200]$. Optimal base step sizes are shown in form of the black curve whereas the overall minimum is indicated with the white marker.

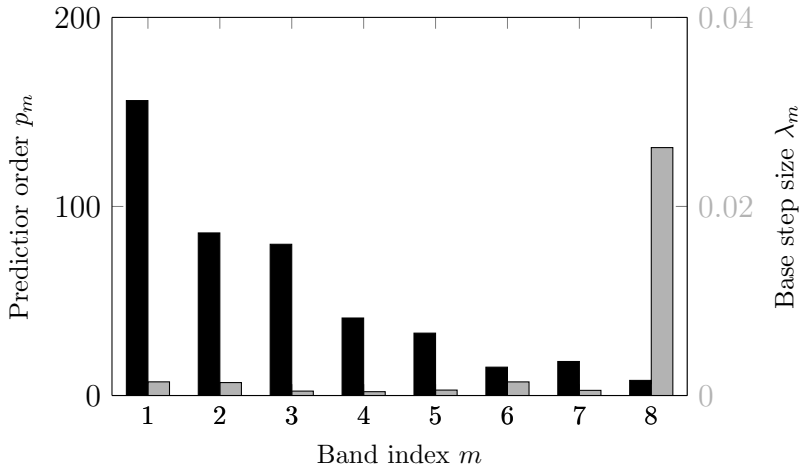


Figure 3.23: Best-performing order p and base step size λ from the simple optimization over subbands m.

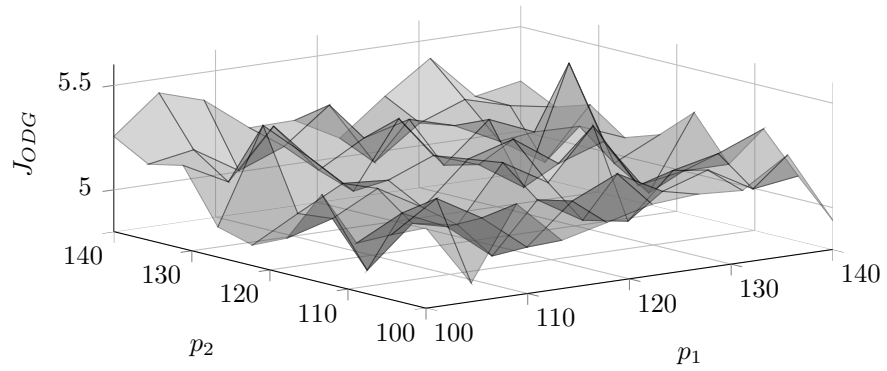


Figure 3.24: Perceptually motivated cost function J_{ODG} plotted over prediction order of first and second band p_1, p_2.

Figure 3.23 shows the overall best-performing order and base step size values for every band m. As expected, the prediction order is decreasing over the bands due to the increasing noisy character of the subband signals. In other words, the lower subbands contain more predictable signal components.

3.5.2 Simulated Annealing

Although the obtained predictor settings are optimal in terms of the resulting prediction error, further optimizations offer the potential of enhancing the audio quality of the proposed codec drastically. The major drawback of the undertaken optimization is that the utilized cost function is based on the prediction error energy which is not reflecting characteristics of the human hearing. Therefore, the cost function for the final optimization

$$J_{ODG} = \frac{1}{F} \sum_{f=1}^{F} \mathrm{ODG}_f^4 \tag{3.26}$$

utilizes the ODG score from the PEAQ algorithm (see Sec. 2.3.1) to assess the codec in terms of perceptual quality instead of empirical error energy measures as proposed in [HZ09]. The cost function emphasizes strongly degraded items due to the fourth power. This intends to force the optimization to an overall pleasant quality instead of very good quality with strong outliers for certain input signals. In addition, the cost function is not order-dependent and hence the optimization routine shall jointly adapt the order p, base step size λ, stability constants σ_{min}, v_{min}, and smoothing coefficients $\lambda_{\mathrm{AT/RT}}$ for all subbands. The optimization is again performed on the SQAM dataset. The very complex-shaped cost function, illustrated in Fig. 3.24, and the parameter dimensionality provoke to utilize another optimization algorithm than

the simple optimzation routine from the previous chapter. As proposed in [HZ09], the simulated annealing optimization routine [KGV83, Čer85] is chosen. It is advantageous for this task due to its heuristic capabilities, simplicity, and the hill-climbing capability. Hill-climbing in this context describes escaping from local minima to allow global optimization.

Simulated annealing is loosely based on the thermodynamic process of annealing solid matter. Slow and controlled annealing leads to an optimized, low-energy state of the particles within the solid. In contrast, the particles of the melted solid feature high energy and circulate in a random manner. This idea is projected on the problem of minimizing a cost function $J(\mathbf{X}_i)$ parametrized by the parameter vector $\mathbf{X}_i$. Similar to the randomly moving particles, the parameter vector is modified for every time step i

$$\mathbf{X}_{i+1} = \mathbf{X}_i + \mathbf{r}_i. \tag{3.27}$$

by adding a random sequence $\mathbf{r}_i$. The resulting new state of the parameter vector is accepted if

$$J(\mathbf{X}_{i+1}) < J(\mathbf{X}_i) \tag{3.28}$$

and hence the resulting cost function decreased. To incorporate the possibility of a high-energy particle breaking through the current lowest-energy grid, a parameter vector can also be accepted with a certain probability

$$P = e^{-\frac{J(\mathbf{X}_{i+1}) - J(\mathbf{X}_i)}{T_i}} \tag{3.29}$$

which decreases for the decreasing Temperature T_i which is continuously lowered for every new time step

$$T_{i+1} = \gamma \cdot T_i. \tag{3.30}$$

In the following, the simulated annealing routine of MATLAB® using the standard parameters $\gamma = 0.95$ and $T_0 = 100$ is utilized and led to the results listed in Tab. 3.2.

The prediction order p_m and base step size λ_m differ quite clearly from the initial optimization of the previous chapter when optimizing them jointly using a perceptually motivated cost function. The overall trend of the predictor order remains similar though. The stability constant $v_{min,m}$ features a strong deviation which can be expected due to varying statistical distributions for every subband m. Surprisingly, the envelope smoothing coefficients $\lambda_{\mathrm{AT/RT,m}}$ are almost constant. One possible explanation could be that the temporal signal characteristic do not differ drastically between the subbands.

Table 3.2: Optimized codec parameters after Simulated Annealing optimization for $M = 8$ bands.

Band m	Order p_m	Base step size λ_m	Constant $v_{min,m}$	$\lambda_{\mathrm{AT,m}}$	$\lambda_{\mathrm{RT,m}}$
1	119	$8.1380 \cdot 10^{-3}$	$3.6502 \cdot 10^{-7}$	$8.0005 \cdot 10^{-1}$	$1.0005 \cdot 10^{-1}$
2	112	$5.9780 \cdot 10^{-3}$	$3.8094 \cdot 10^{-5}$	$8.0048 \cdot 10^{-1}$	$9.9988 \cdot 10^{-2}$
3	88	$6.1173 \cdot 10^{-3}$	$5.5866 \cdot 10^{-5}$	$8.0046 \cdot 10^{-1}$	$1.0017 \cdot 10^{-1}$
4	75	$3.5272 \cdot 10^{-3}$	$7.8944 \cdot 10^{-5}$	$7.9988 \cdot 10^{-1}$	$9.9973 \cdot 10^{-2}$
5	41	$3.2309 \cdot 10^{-3}$	$5.4090 \cdot 10^{-5}$	$7.9973 \cdot 10^{-1}$	$1.0078 \cdot 10^{-1}$
6	26	$1.1997 \cdot 10^{-3}$	$1.2565 \cdot 10^{-5}$	$8.0054 \cdot 10^{-1}$	$9.9990 \cdot 10^{-2}$
7	26	$2.7557 \cdot 10^{-3}$	$1.9712 \cdot 10^{-6}$	$8.0000 \cdot 10^{-1}$	$9.9976 \cdot 10^{-2}$
8	19	$8.9343 \cdot 10^{-3}$	$1.5693 \cdot 10^{-6}$	$8.0042 \cdot 10^{-1}$	$9.9947 \cdot 10^{-2}$

Note, that the author decided to neglect the codebook design in the optimization process to limit the parameter space and to use a codebook consisting of noise featuring a gaussian distribution. Jointly optimizing the codebook using more sophisticated methods, like k-means clustering [LBG80], and the remaining codec parameters is believed to perform even better.

3.5.3 Instrument-Class Specific Parameter Optimization

The global optimization of codec parameters for a non-homogeneous mix of input signals could be clearly achieved. Nevertheless, it is likely that an NMP participant is playing a single instrument and hence the codec could be further optimized for specific instruments. To evaluate this concept of instrument class-dependent codec presets the SQAM data set was partitioned into the instrument classes listed in Tab. A.1. For every listed instrument class, the optimization routine from Sec. 3.5 is repeated but on the restricted data set. The average ODG scores for the denoted instrument classes for the generic and the specific optimization are illustrated in Fig. 3.25. The average ODG score can be improved for every class of instruments except the string instruments. Especially the speech and percussion classes benefit from individual optimization. Although, it is somehow obvious that the two instrument classes featuring the strongest deviation to typical harmonic instrument sound benefit most. Hence, codec presets for certain instrument classes are beneficial in the context of NMP to increase audio quality without increasing the complexity or bitrate of the corresponding audio stream. Solely, the predictor settings have to be saved in encoder and decoder and an instrument class index has to be transmitted in form of meta data to realize this concept.

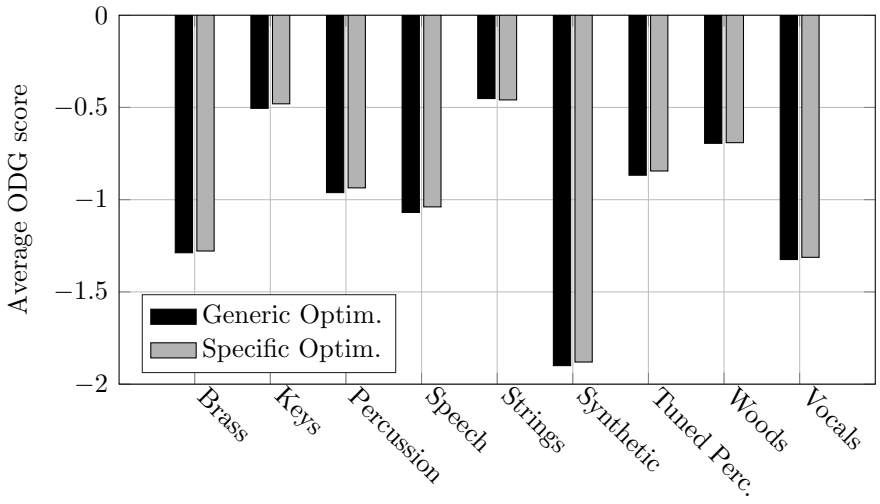


Figure 3.25: Average ODG score for individually optimized instrument classes.

3.6 Evaluation

The proposed codec structure is evaluated using the PEAQ algorithm from Sec. 2.3.1 and the SQAM data set as priorly used. At first it shall be shown, how the VQ-ADPCM codec performs in contrast to a broadband reference codec from [HHZ08] using parameters from [HZ09] for the scalar 3 bit quantizer and without the noise shaping functionality. Unfortunately, the bitrates of 64 and 132.3 kbit/s can't be aligned without further changes. The proposed codec is operated with the optimized parameters from Sec. 3.5 and a $M = 8$ band window-based filter bank design using $N = 51$ coefficients.

The resulting ODG scores for all SQAM items are plotted in Fig. 3.26. The broadband variant achieves a constantly good quality in the range of $[-1.4, \ldots, -0.075]$ averaging to -0.46. Only a few items (castanets (27), triangle (32), accordion (42), and organ (56)) are rated below -1 corresponding to a perceptible, but not annoying audio quality degradation. The proposed codec structure running at half the bit rate performs similar for the majority of SQAM items. Nevertheless, the synthetic (1,3-7), horn (23), tuba (24), castanets (27), vibraphone (37), and bass (47) items are massively degraded. The average ODG score constitutes -0.76. In other words, the proposed codec structure allow to save about half of the bit rate for the cost of 0.3 ODG.

The cause for the quality loss shall be analyzed with the help of selected examples. Figure 3.28 shows an excerpt of SQAMs castanet (27) example.

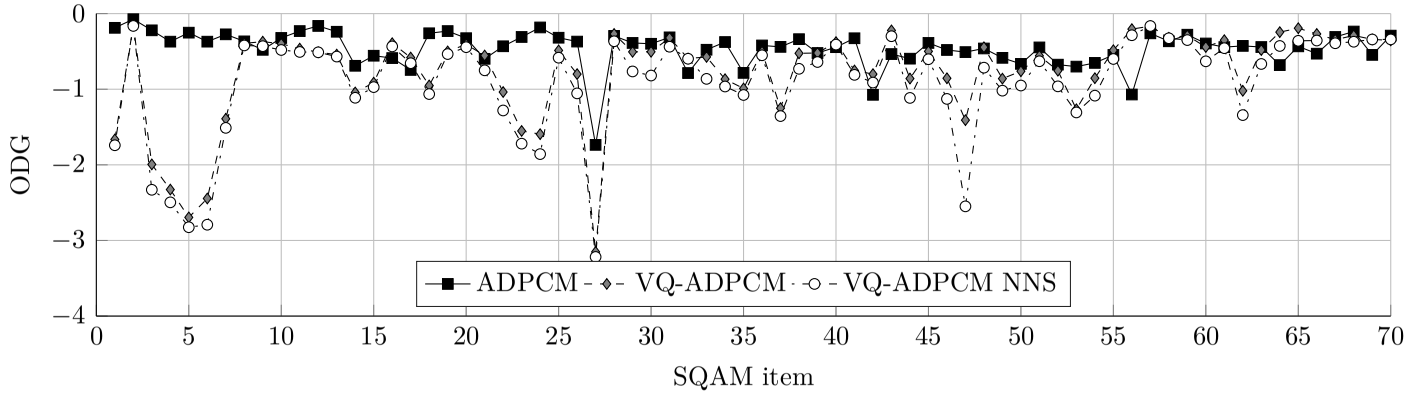


Figure 3.26: ODG score of the proposed codec, the codec using NNS, and a 3 bit broadband reference using the SQAM data set.

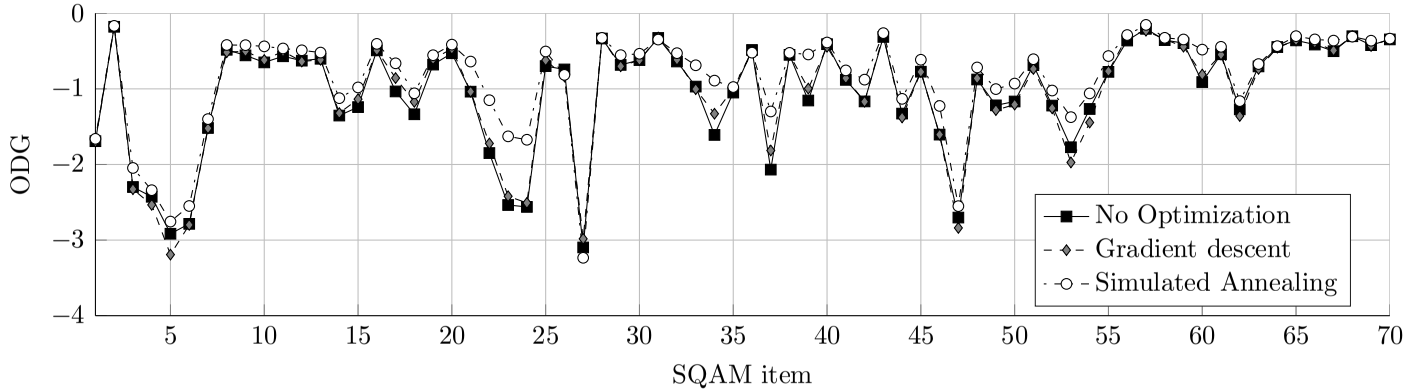


Figure 3.27: ODG score of the proposed codec without optimization, after simple optimization, and after Simulated Annealing optimization using the SQAM data set.

The en- and -decoded, delay-compensated codec output signal is plotted on top. It can be clearly seen how the first part of the transient is not gathered in the output signal. This is caused by tardy reacting predictors which are tuned to perform well in general. However, these extremely fast transients constitutes a rather rare signal characteristic and therefore only slightly influences the optimization process. Another quality-degrading artifact is visualized in Fig. 3.29. The spectrogram of the tuba (24) example shows a certain amount of aliasing at $\frac{fs}{8} = 5512.5\,\text{Hz}$. The amplitude of the aliasing components is about 60 dB smaller than the ones of the fundamental frequencies. Nevertheless, the artifacts are audible since the tuba example mainly consists of low-frequency content. Therefore, the artifacts are not masked by the actual signal. This explanations also holds true for the synthetic (1,3-7) SQAM tracks.

Figure 3.26 shows the ODG score of the proposed codec when the NNS codebook search is applied. It is apparent, that a minor quality degradation to an average value of 0.85 occurs since the NNS does not guarantee to always find the optimal codebook entry. On the other hand, it massively reduces the complexity to allow real-time applicability.

The effect of the codec parameter optimization on the perceptual quality is shown in Fig. 3.27. The basic trend of ODG scores is similar for the non-optimized and optimized codec parameters. This leads to the assumption that the codec parameters of the ADPCM broadband variant perform already well within the subband ADPCM approach. Another explanation is that the aliasing distortion of the FIR filter bank is the major contributor to the quality degradation. Applying the simple parameter optimization improved the average ODG sore from -1.04 to -1.03. The following simulated annealing optimization increased the average ODG score further to -0.85. The improvement of the second optimization is apparently clearly superior. Hence, the optimization of the stability constant v_{min}, which was omitted for the simple optimization to restrict the parameter room, is crucial to achieve a strong quality improvement.

In the following, the proposed codec shall be ranked in terms of quality respecting the algorithmic delay. Therefore, the SQAM data set was encoded using several different codecs operating at 64 kbit/s. Besides VQ-ADPCM, an implementation of mp3, two different AAC and HE-AAC variants, and the Opus codec in all possible frame size variants were tested. The software versions of the utilized codecs are listed in Tab. A.2. Figure 3.30 illustrates the average ODG score of the mentioned codecs over its algorithmic delay. The delay values for the MPEG codec variants from [GLS+04] are utilized. All MPEG variants (mp3, AAC, and HE-AAC) clearly outperform the VQ-ADPCM. But the lowest encoding delay is 58 ms at $f_s = 48\,\text{kHz}$ and hence

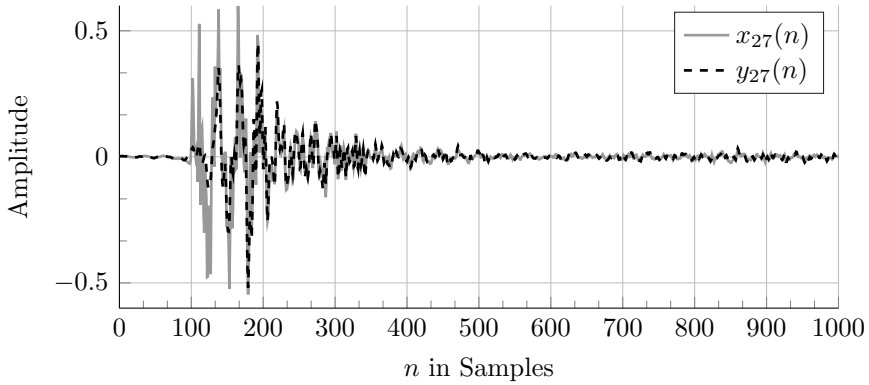


Figure 3.28: Codec output and reference signal for single transient of SQAM castanets track.

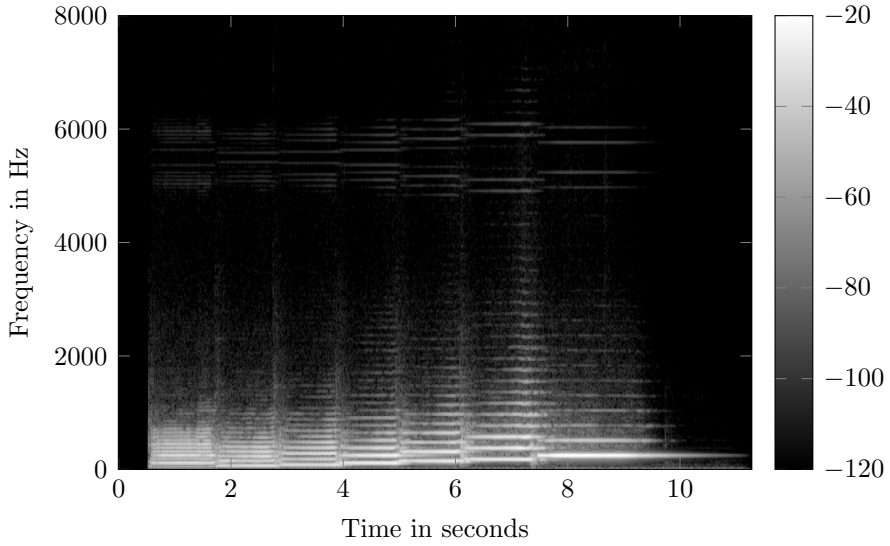


Figure 3.29: Spectrogram of codec output for SQAM tuba track.

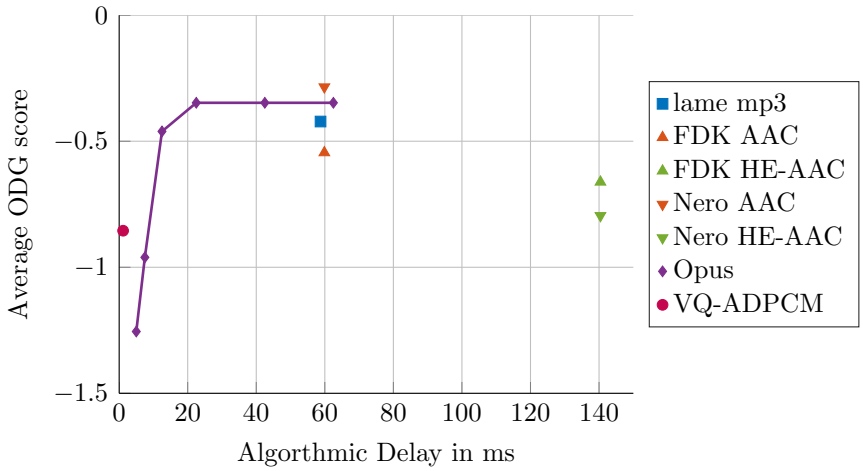


Figure 3.30: Average ODG score of different audio codecs using a bitrate of 64 kbit/s in relation to their algorithmic delay using the SQAM data set.

these codecs are not suited for the NMP application. The Opus codec was especially designed for low latency applications and hence the delay scales from 62.5 to 5 ms. The quality for delay settings above 12.5 ms is clearly superior. But for lowest delays the VQ-ADPCM outperforms the Opus codec. The comparison to other low-delay codecs like apt-X and ULD could not be performed due to restricted availability of these codecs.

3.7 Summary

A novel audio codec structure for the application of NMP was presented in this section. The initial intention of designing an ADPCM variant with lowest delays but competitive bit rates featuring good perceptual quality was achieved. The proposed design utilizes a broadband ADPCM design based on adaptive lattice prediction filters and a scalar quantizer that is made signal-adaptive with the help of a normalization operation beforehand. This ADPCM codec is applied in subbands obtained by a window-designed, critically-sampled FIR filter bank. The scalar quantizer is replaced with a vector quantizer that jointly quantizes the prediction residuals of all subbands to a single codebook index. The normalization procedure of the broadband variant is used in subbands to obtain a signal-adaptive vector quantizer. Every quantization operation requires the search of the codebook entry that represents the subband prediction residuals with the lowest distortion. The distortion is computed using the euclidean norm weighted with an estimate

of the subband residuals envelope. The weighting allows to prioritize relevant subbands in the quantization process. In other words, the band-wise SNR is rearranged to achieve less noise in the relevant bands to obtain a higher perceptual quality. To circumvent the heavy computational load, caused by a linear codebook search, the nearest neighbor search algorithm is applied. Since the codebook is sorted in terms of vector radius and the subband predictors tend to produce small residuals during stationary signal conditions, it was observed that the codebook index probability is unequally distributed. This allowed the application of a Huffman encoder to reduce the average word length of the codebook index from 2 to 1.475 bits per sample. Hence, the VQ-ADPCM operates at about 64 kbit/s for a sampling frequency of 44.1 kHz. Although, well-known codecs like mp3, AAC, and HE-AAC outperform the VQ-ADPCM in terms of perceptual quality at the same bit rate, the proposed codec only features an algorithmic delay of 1.15 ms which is solely caused by the FIR filter bank. The proposed codec also outperforms the Opus codec regarding perceptual quality and algorithmic delay although Opus was designed for the very same application.

Although the result is already pleasing, it must be stated that the VQ-ADPCM should be considered as an initial design with manifold optimization possibilities. For instance, analyzing low-latency filter banks leading to smaller aliasing distortion could massively enhance the perceptual quality. Other codebook designs and enhanced perceptually motivated cost functions for the codebook search are expected to achieve further perceptual enhancements.

4 Enhancing Listening Experience

The previous chapters describe very technical concepts to enhance the audio quality of an NMP session by concealing lost packets and encoding audio data with less delay than typically applied systems but ensuring good quality at low bit rates. These proposed enhancements are purely focused on the data transmission aspects. However, also the audio replay through headphones can be massively enhanced with algorithms illustrated in this section. The headphone replay scenario is assumed since it is the simplest and most direct way of monitoring the signals in an NMP session. Furthermore, replaying the incoming sound with speakers would clearly increase the overall perceived delay. Assuming the speakers are located 2 meters apart from the listener, the corresponding replay delay is $t_{\text{delay}} = \frac{2\,\text{m}}{343.2\,\frac{\text{m}}{\text{s}}} \approx 5.8\,\text{ms}$.

Musicians in a rehearsal room or on a stage experience a completely different acoustic environment than musicians participating in an NMP session. Common NMP systems only allow to adjust the replayed stereo signal in terms of volume and panning for S incoming sources $x_1, \ldots, x_S$ using a

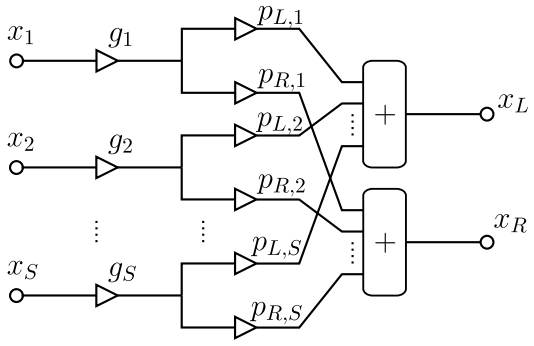


Figure 4.1: Simple stereo mixer.

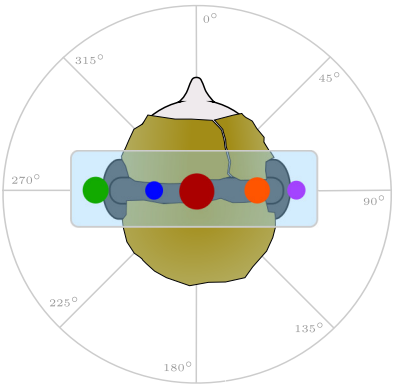


Figure 4.2: Source placement options for stereo panning on headphones.

software mixer. Figure 4.1 shows a corresponding stereo mixer that allows to adapt the volume of source i using the gain factor g_i and the panning by changing the complementary pan coefficient set p_L, p_R to mix the overall output channels

$$x_L(n) = \sum_{s=1}^{S} g_s \, p_L \, x_s(n) \tag{4.1}$$

$$x_R(n) = \sum_{s=1}^{S} g_s \, p_R \, x_s(n). \tag{4.2}$$

Note, that the actual values of the panning coefficients depend on the applied panning law. Typical panning laws are the amplitude-complementary linear, power-complementary square-root, and the power-complementary sinusoidal [BSI+12].

The disadvantage of the simple amplitude panning of Eq. (4.1) and Eq. (4.2) is that incoming sound sources can solely be rendered in form of point sources in a narrow stereo panorama as shown in Fig. 4.2. The colored points represent differently panned sources and the radius illustrates the volume of the sources. To counteract this limited replay scenario, the following sections present alternative replay approaches for incoming mono sources in an NMP session.

First, a blind mono to stereo conversion is presented that allows to create stereo sound sources of arbitrary stereo width. Using this technique enhances the listening experience by providing a certain amount of spaciousness and width. In the following, a way to render audio in a virtual surround space is shown which can also be enhanced by the proposed mono to stereo conversion.

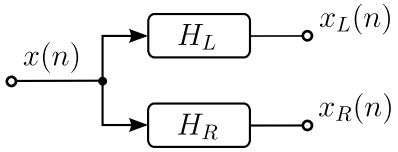


Figure 4.3: Basic blockscheme of the pseudo-stereo system.

4.1 Frequency-Dependent Amplitude Panning

A significant part of recordings is still done in mono for several reasons. Simple broadcast commentaries produced outside the usual studio environment are often done using a single channel for example. In the case of NMP, simple hardware setups using a single microphone or an instrument's direct output are desirable to facilitate the NMP setup for users. However, having a stereo recording is likely to produce enhanced listening experiences due to its spatial character. Estimating a stereo signal from mono sources is a surprisingly old problem and therefore, it is well-known in literature.

Early attempts to realize a pseudo-stereo conversion used a delayed version of the input signal to provide a second channel [Lau54]. The same author came up with the idea of applying complementary comb filters which were later extended to be phase-aligned in [Sch58]. Alternatively, in [Bau69, Orb70] different allpass network designs were proposed to obtain a strong decorrelation and to achieve a wide and also scalable stereo image. Another extension allowing more control is shown in [Ger92]. For even stronger decorrelation, a frequency-domain filter design method is suggested in [Ken95]. The above methods either impose a strong timbral coloration or they are not downmix compatible and reversible. Both are important features, though.

A completely different approach granting possibilities to design a specific auditory image is explained in [Fal05]. However, it is not a pseudo stereo algorithm in the narrower sense as it requires complex user input to explicitly define pan positions for certain frequency bands and does not allow a fully automatic conversion. The same holds true for upmixing based on Directional Audio Coding (DirAC) [PPP12]. However, the decorrelation mechanism in the DirAC synthesis is similar to the proposed approach to a certain extent.

The basic idea of the proposed pseudo-stereo system is to apply two filters $H_L(e^{j\Omega})$ and $H_R(e^{j\Omega})$ on the input signal $x(n)$ to produce two output channels $x_L(n)$ and $x_R(n)$ as illustrated in Fig. 4.3. Strong decorrelation and hence spatiality is achieved when the filters differ clearly.

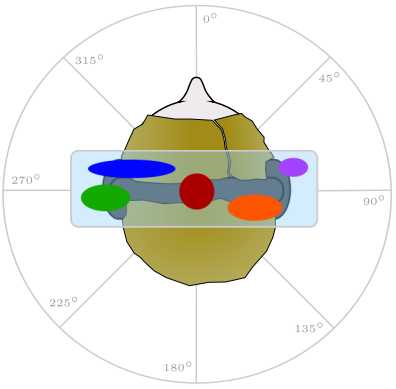


Figure 4.4: Source placement and scalability options for stereo panning on headphones using the proposed pseudo-stereo technique.

4.1.1 Constraints

The proposed pseudo-stereo method is supposed to allow the spatial enhancement without degrading the actual source signal. Therefore, certain design constraints have to be fulfilled in the filter design. First, it shall be guaranteed that the overall sound quality of a sound source remains unmodified. This can be realized by forcing the mono sum

$$\begin{aligned} x_M(n) &= x_L(n) + x_R(n) = \\ &= x(n) * h_L(n) + x(n) * h_R(n) \stackrel{!}{=} x(n-D), \end{aligned} \tag{4.3}$$

of the pseudo-stereo signal to be the original signal which is only altered in form of a delay D caused by the filters. Examining Eq. (4.3) in the frequency domain

$$\begin{aligned} X_M(e^{j\Omega}) &= X(e^{j\Omega}) \cdot H_L(e^{j\Omega}) + X(e^{j\Omega}) \cdot H_R(e^{j\Omega}) = \\ &= X(e^{j\Omega}) \cdot \left(H_L(e^{j\Omega}) + H_R(e^{j\Omega})\right) = \\ &\stackrel{!}{=} X(e^{j\Omega}) \cdot e^{-j\Omega D} \end{aligned} \tag{4.4}$$

leads to the constraint of linear phase for the sum of the pseudo-stereo filters

$$H_L(e^{j\Omega}) + H_R(e^{j\Omega}) \stackrel{!}{=} e^{-j\Omega D}. \tag{4.5}$$

Even more important is that the algorithm doesn't alter the original timbre of the sound source which is ensured by forcing the sum of the filters to be neutral

$$|H_L(e^{j\Omega})| + |H_R(e^{j\Omega})| \stackrel{!}{=} 1. \tag{4.6}$$

To allow the implementation of the proposed pseudo-stereo method in the time-domain it is crucial to have conjugate symmetric frequency responses

$$H_{L/R}(e^{j\Omega}) = H^*_{L/R}(e^{-j\Omega}) \tag{4.7}$$

to achieve real-valued impulse responses $h_{L/R}(n)$. The claim for constant magnitude in Eq. (4.6) and for linear phase in Eq. (4.5) implies that pure amplitude panning is performed. Otherwise, no downmix compatibility could be realized. Since the amplitude panning varies over the spectrum, this method is denoted *Frequency-Dependent Amplitude Panning* (FDAP) in the following.

4.1.2 Filter Design

It is well-known that a regular frequency magnitude pattern, as achieved by higher-order complementary comb filters, provides a significant amount of decorrelation but still sounds synthetic and unnatural. Therefore, the digital filter design should lead to a diffuse frequency response. The proposed filter design is in fact based on a random sequence $R(k)$ where k denotes the frequency bin index. The noise sequence, characterized by its gaussian distribution with a variance $\sigma = 25$ and mean $\mu = 0$, is scaled with the squared width parameter w and then nonlinearly clamped with the arctan function. At last, the resulting sequence is scaled to the range $[0, \ldots, 1]$ and supplied with the linear phase term to yield the pseudo-stereo filter for the left channel

$$H_L(k) = \left(\frac{1}{2} + \frac{1}{\pi}\arctan(w^2 \cdot R(k))\right) e^{-j\frac{2\pi k D}{N}}. \tag{4.8}$$

The corresponding right channel can be computed as the complement

$$H_R(k) = 1 - H_L(k). \tag{4.9}$$

An example frequency response is shown in Fig. 4.5. When the width parameter is set to $w = 0$, the filter features a constant magnitude of 0.5 and hence, no panning is performed. For increasing values of w, the frequency response converges to its limits. Some instruments have a typical placement in a stereo mix. For example, most listeners are conditioned to hear singer and bass instruments from the center, whereas cymbals and horn sections are expected to appear clearly panned. To approximate this hearing experience, certain frequency regions can be kept in the center by setting

$$|H_{L/R}(k)| = 0.5, \quad \text{for} \quad k \in [k_{\text{lo}}, \ldots, k_{\text{hi}}], \tag{4.10}$$

where $k_{\text{hi/lo}}$ denotes the corresponding cut-off frequencies defining the band that is actually processed.

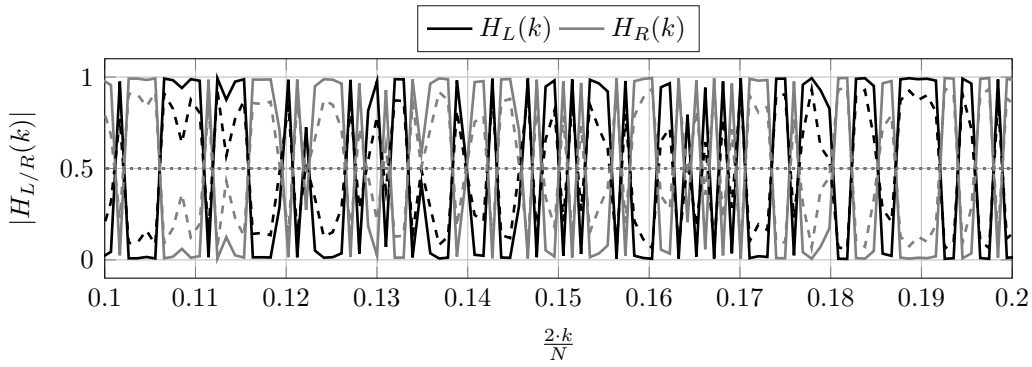


Figure 4.5: Example frequency response detail of the pseudo-stereo filters for stereo width parameter $w = [0, 0.3, 1]$.

4.1.3 Realizations

The proposed pseudo-stereo method can be applied in time- and frequency-domain. The application in the time domain requires the impulse response of the left pseudo-stereo filter of Eq. (4.8) which can be obtained with the inverse discrete Fourier transform

$$h_L(n) = \mathcal{F}^{-1}\{H_L(k)\}. \tag{4.11}$$

The left channel

$$x_L(n) = x(n) * h_L(n) \tag{4.12}$$

is obtained by FIR filtering the mono input signal with the impulse response whereas the right channel can be computed by subtracting the left output channel from the time delayed input signal

$$x_R(n) = x(n - D) - x_L(n), \tag{4.13}$$

where $D = \frac{N-1}{2}$ is the group delay of the FIR filter of length N. The obvious advantage of the time-domain realization is the utilization of a single filter. The drawback of this approach is the complexity of the FIR filtering operation for longer impulse responses. This effort can be significantly reduced when the fast convolution in the frequency domain as shown in Fig. 4.7 is applied. After transferring the input signal to the time-frequency domain using the Short-time Fourier Transform (STFT) the left and right pseudo-stereo channel

$$X_{L/R}(b, k) = X(b, k) \cdot H_{L/R}(k) \tag{4.14}$$

can be computed as the element-wise product. The corresponding time-domain signals are synthesized with the following inverse STFT. Similar to

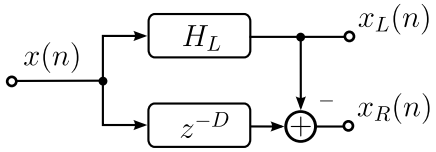


Figure 4.6: Blockscheme of the time-domain realization.

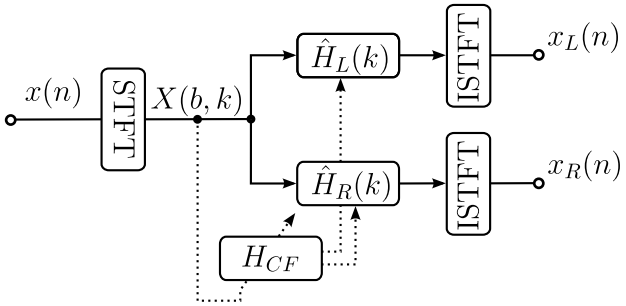


Figure 4.7: Blockscheme of the frequency-domain realization.

the time-domain realization the right channel can also be computed by subtracting the left channel from the original signal. The additional filter H_{CF} in Fig. 4.7 will be introduced in Sec. 4.3.

Besides reducing complexity of the filtering operation, the frequency-domain processing also allows the dynamic adaption of the pseudo-stereo filters without the need to continuously compute the impulse response for time-domain processing.

4.2 Evaluation

As mentioned in Sec. 2.3.2 measuring audio quality is a very non-trivial task. In the context of concealment and coding it is at least possible to measure and interpret the acoustic impairments of processed data relative to a reference. In contrast, evaluating the audio quality of a novel audio effect in a similar manner is not possible due to missing references. Although the rating of quality of the proposed method remains purely subjective, the spaciousness can be objectively measured. A typical measurement in the field of room acoustics is the *Interaural Crosscorrelation Coefficient* (IACC). It allows to rank a room in terms of envelopment and spaciousness. This measure can be adapted to evaluate signals resulting in the *Interchannel Crosscorrelation Coefficient* (ICC)

$$\mathrm{ICC} = \max |r_{LR}(\tau)|, \tag{4.15}$$

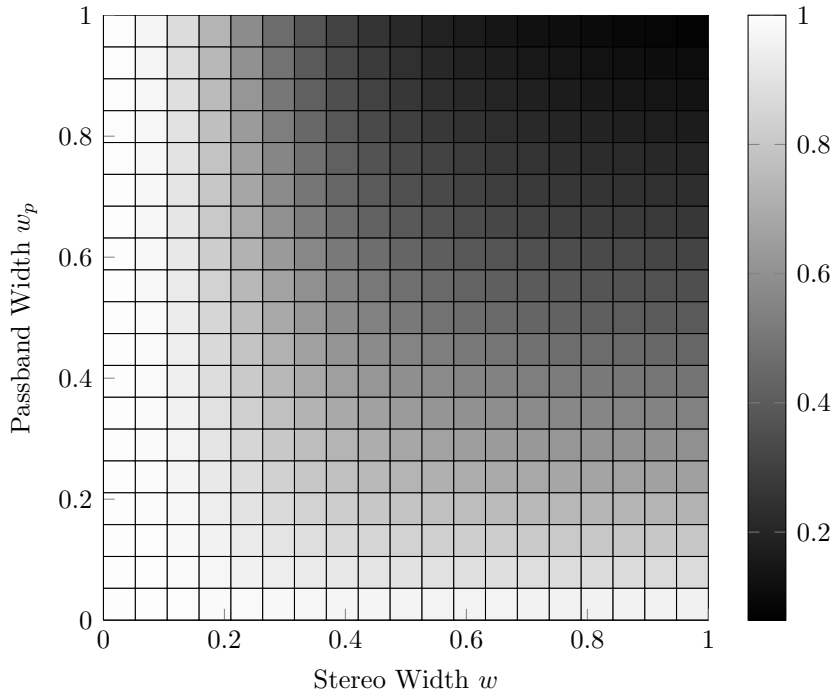


Figure 4.8: ICC for FDAP filtered white noise using different values for the stereo width w and passband width w_p.

which is defined as the maximum value of the normalized crosscorrelation function

$$r_{LR}(\tau) = \frac{\sum x_L(n) x_R(n-\tau)}{\sqrt{\sum x_L^2(n) \sum x_R^2(n)}}. \tag{4.16}$$

The spaciousness and hence correlation of FDAP-processed signals is validated by feeding the same white noise signal to the left and right decorrelation filter $h_{L/R}(n)$ correspondingly and computing the ICC in the following. The filters are designed with Eq. (4.8) and Eq. (4.9) using varying values for the stereo width w and passband width $w_p = \frac{k_{\mathrm{hi}} - k_{\mathrm{lo}}}{N}$. The result is illustrated in Fig. 4.8. It can be clearly seen, how the ICC value falls from 1 to almost 0 for increasing values of w and w_p. Hence, FDAP is capable to achieve almost full decorrelation for extreme settings. In other words, FDAPs resulting spaciousness can be adjusted using different design parameters.

4.3 Center-Focusing Enhancement

As previously mentioned most listeners are used to typical scenarios and pannings. The appearance of the dominant sound source in the center of

the stereo panorama is common. For example, the voice of a reporter in a documentary or the singing voice in most musical pieces is expected to be unpanned and the corresponding pseudo-stereo signal would deliver an unusual, discomforting listening experience. Therefore, the proposed pseudo-stereo method is extended with a so-called center-focusing filter $H_{CF}(b,k)$ as shown in Fig. 4.7. The filter $H_{CF}(b,k)$ forces the dominant spectral components back to the center.

One way of describing dominant spectral components is to estimate the normalized spectral energy. It can be computed using the amplitude-normalized magnitude spectrum

$$X_n(b,k) = \frac{|X(b,k)|}{\max\limits_k |X(b,k)|}. \tag{4.17}$$

In a next step, the squared magnitude spectrum is recursively averaged to compute the center-focusing filter

$$H_{CF}(b,k) = (1-\alpha)\, H_{CF}(b-1,k) + \alpha\, X_n^2(b,k). \tag{4.18}$$

The filter tends to feature values close to 1 for stationary components with high amplitudes whereas the remaining components are close to 0. Other features like tonalness [KLZ13] are expected to work as well to compute the center-focusing filter.

In a next step, a weighted pseudo-stereo filter

$$\hat{H}_{L/R}(b,k) = H_{L/R}(k)\, H_{CF}(b,k) - \frac{1}{2}\,(1 - H_{CF}(b,k)) \tag{4.19}$$

is computed which forces the magnitude of strong components to be close to 0.5 and hence those components are panned to the center.

4.4 Application in Virtual Surround

The proposed pseudo-stereo technique demonstrates a convenient way of increasing the spaciousness of a mono signal in a simple and controllable way. Nevertheless, the perceived virtual acoustic space is bounded by the headphones since no externalization of sound sources can be achieved. The externalization of sound sources can be achieved when sound sources are placed in a virtual acoustic space using *Head-Related Transfer Functions* (HRTFs) and their time-domain correspondents *Head-Related Impulse Responses* (HRIRs) [MHJS95]. HRTFs describe the transfer function of the outer ear, pinna, and torso of a subject in dependence of the incoming sound source azimuth

and elevation angle. They are obtained using measurements with inear microphones or using generic modeling [BD98]. Since the size and shape of different persons pinna differ significantly also the corresponding HRTFs vary drastically [WAKW93]. Hence, measured HRTFs achieve best localization effect for individual persons but are impracticable for the general application in contrast to modeled or dummy-head measured HRTFs. Figure 4.9 shows HRIRs and the corresponding HRTFs from the FABIAN dummy head database [LW07] for different azimuth angles. Besides the expected level difference for left and right channels at angles $\phi \neq 0°$ it can be seen that strong spectral notches at high frequencies are essential for the perception of direction.

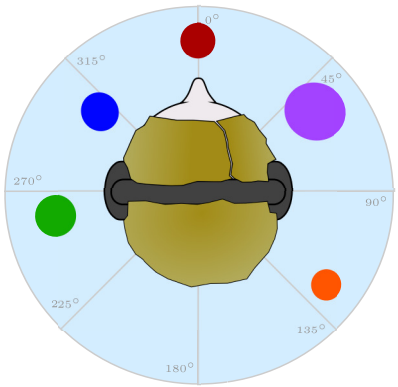


Figure 4.10: Source placement and scalability options for stereo panning on headphones using HRIRs.

An example of 5 sound sources which are located in a virtual acoustic space is shown in Fig. 4.10. In contrast to the simple panning in Fig. 4.2 and the pseudo-stereo panning in Fig. 4.4 the sound sources are perceived outside of the head at arbitrary azimuth angles. Due to its manifold design possibilities and the potential of designing convincing listening experiences, the rendering with HRIRs is used in several platforms. Amongst others, it is applied in generic spatial audio engines [AGS08], virtual acoustic space rendering [OCD+13], virtual room-acoustic enhanced teleconference systems[1] and spatial in-ear monitoring[2].

The basic processing that is required to realize a virtual space similar to the previous examples is illustrated in Fig. 4.11. In contrast to the simple stereo mixer of Fig. 4.1 every sound source $x_s(n)$ is filtered with the HRIR $g_{L/R}^{\phi_s}(n)$ for an azimuth angle ϕ_s for left and right headphone channel instead

[1] https://www.symonics.com/
[2] http://www.klang.com/

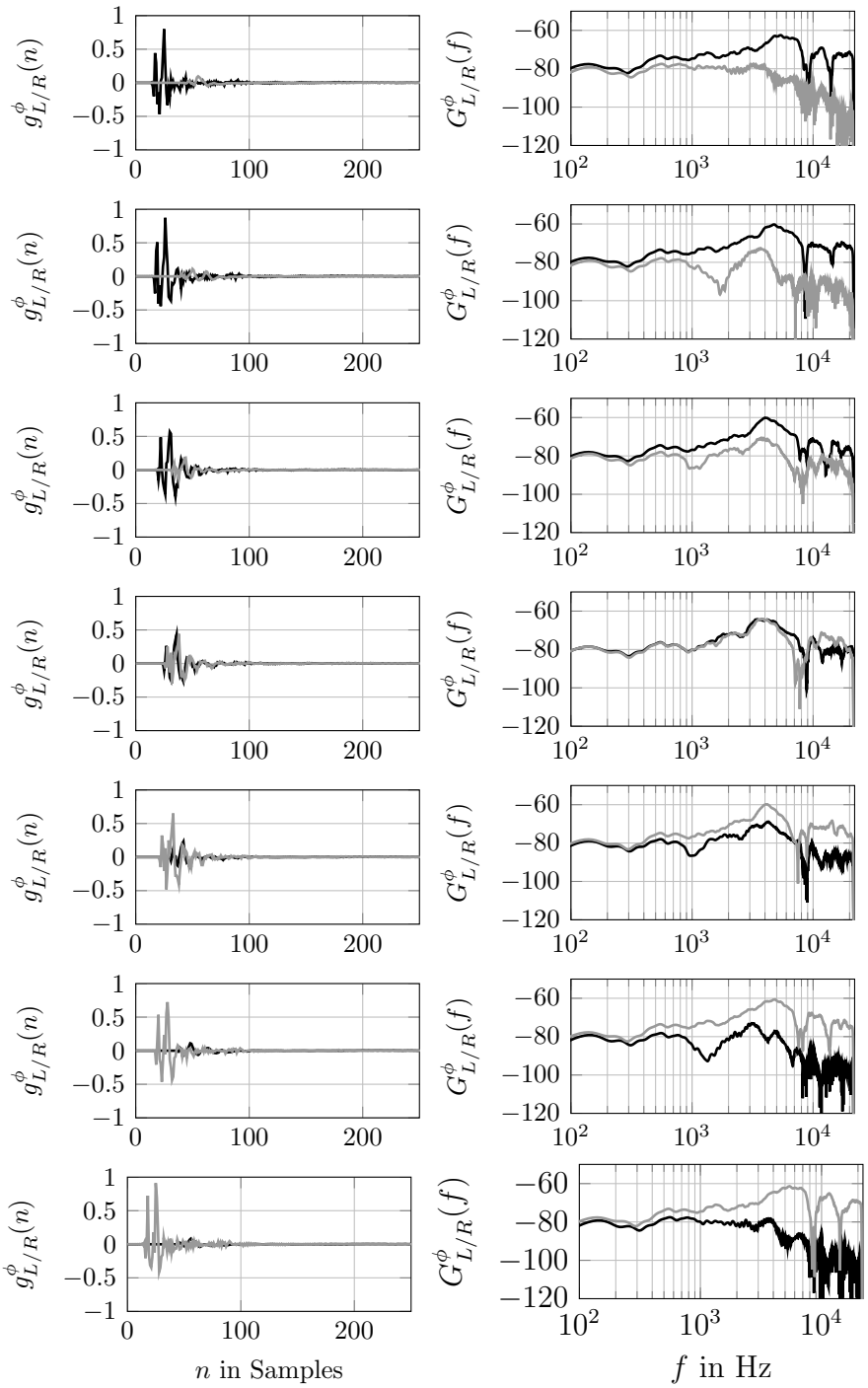


Figure 4.9: Measured HRIRs and HRTFs for azimuth angles $\phi = [-90°, -60°, -30°, 0°, 30°, 60°, 90°]$ for left (black) and right channel (grey) correspondingly.

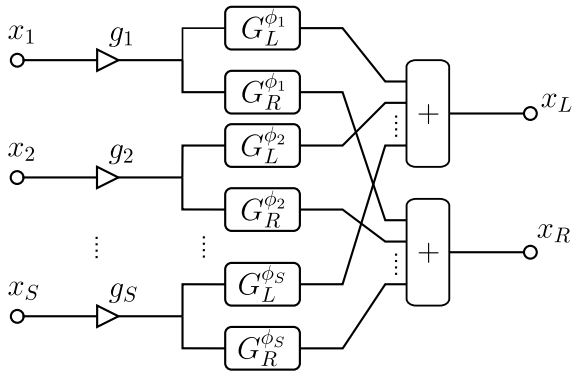


Figure 4.11: Virtual surround mixer using HRIRs.

of simple weighting. The corresponding output channels are defined as

$$x_L(n) = \sum_{s=1}^{S} g_s \cdot (g_L^{\phi_s}(n) * x_s(n)) \tag{4.20}$$

$$x_R(n) = \sum_{s=1}^{S} g_s \cdot (g_R^{\phi_s}(n) * x_s(n)). \tag{4.21}$$

Although the virtual surround panning already allows a flexible and immersive replay of sound sources it can be further enhanced with the proposed pseudo-stereo method. The application of HRIRs to single mono sources solely define a point source perceived from a certain azimuth and elevation angle. However, when the source is preprocessed with the pseudo-stereo method to obtain a stereo signal, both signals can be individually filtered with HRIRs for similar but unequal angles $\phi_{s,a}$ and $\phi_{s,b}$. The point source is therefore expanded to a sound source with a radial size of $\Delta\phi_s = \phi_{s,a} - \phi_{s,b}$.

Different approaches can be used to select the HRIRs $g_{L/R}^{\phi_{s,a/s,b}}(n)$:

1. Using single HRIRs at angle $\phi_{s,a}$ and $\phi_{s,b}$

2. Averaging the M available HRIRs within a certain radial area
   $\frac{1}{M} \sum_{\Phi=\phi_{s,a/s,b}-\Delta\phi_s}^{\phi_{s,a/s,b}+\Delta\phi_s} g_{L/R}^{\Phi}(n)$

3. Spatial windowing of HRIRs within a certain area
   $\sum_{\Phi=\phi_{s,a/s,b}-\Delta\phi_s}^{\phi_{s,a/s,b}+\Delta\phi_s} w(\Phi)\, g_{L/R}^{\Phi}(n)$

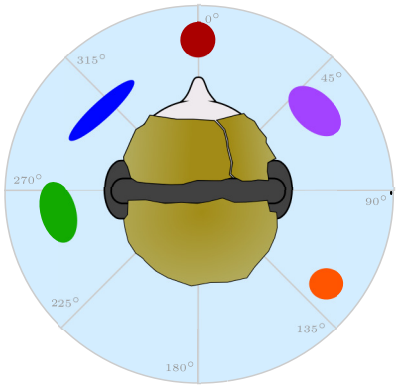


Figure 4.12: Source placement and scalability options for stereo panning on headphones using the HRIRs in combination with the proposed pseudo-stereo technique.

The advantage of the combined FDAP-HRTF panning is visualized in Fig. 4.12. This panning approach allows to position a sound source at an arbitrary angular position and additionally allows to define the size of the sound source instead of rendering every sound source as a point source. Especially larger instruments like pianos and drums benefit from this processing and can be experienced more naturally.

The major drawback of the FDAP-HRTF panning is the increased complexity. The computation of the output channels

$$x_L(n) = \sum_{s=1}^{S} g_s \cdot x_s(n) * (h_{s,L} * g_L^{\phi_{s,a}}(n) + h_{s,R} * g_L^{\phi_{s,b}}(n)) \tag{4.22}$$

$$x_R(n) = \sum_{s=1}^{S} g_s \cdot x_s(n) * (h_{s,L} * g_R^{\phi_{s,a}}(n) + h_{s,R} * g_R^{\phi_{s,b}}(n)). \tag{4.23}$$

requires the application of the FDAP filter pair and the convolution with 4 HRIRs per sound source as visualized in Fig. 4.13. In contrast, the simple HRTF panning of Eq. (4.21) solely requires two convolutions.

4.5 Summary

Many NMP systems offer only limited tools to create acoustically pleasing stereo mixes of an NMP session. The attempts to recreate acoustical environments, which musicians are used to, are also very limited. One way of enhancing the audio replay is to mix stereo sources instead of mono sources. A method to create convincing stereo signals from mono signals is derived

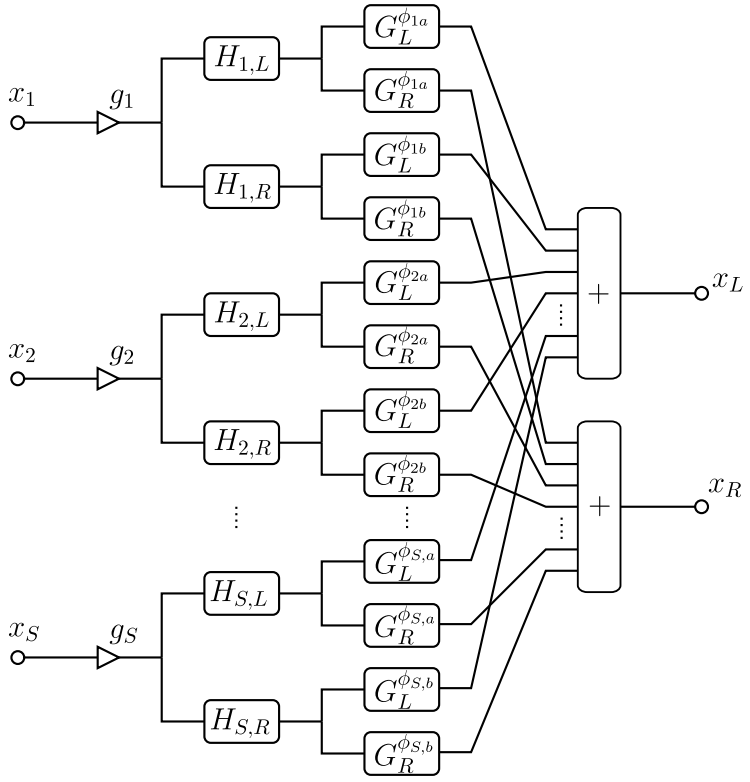


Figure 4.13: Virtual surround mixer featuring size-scalable sources.

in this chapter. The proposed method is based on a pair of complementary filters which realize frequency-dependent amplitude panning with a high frequency resolution. Since the phase relations are preserved the resulting stereo signal is downmix-compatible and does not add any timbral coloration. The filter design, including various design parameters like the stereo width or band width of the signal to be processed, is done in the frequency domain using random noise sequences to achieve very diffuse frequency responses which tend to give a pleasing, natural listening experience. Furthermore, the filter approach is extended to keep dominant spectral components and hence sources in the center of the stereo panorama. The integration of the proposed approach in a virtual surround render engine based on HRTFs is shown. The extension allows to define the size of sound sources in the azimuth domain and hence convincing representations of larger sound sources which appear unrealistic when rendered as point sources.

5 Conclusion

Artistic processes have been evolving ever since. In the domain of music, this evolution has mainly been pushed by the development of new instruments, artistic collaborations, and cultural exchange. Nowadays, technology complements the set of musical evolution catalysts. The availability and structural quality of the internet allows online collaborations of musicians in real-time which are called Networked Music Performances (NMP). Several commercial and academic NMP platforms have been established to help musicians in global artistic collaborations. The wide availability of NMP providers indicates the technology to be well-engineered. However, several technical key questions are still to be resolved. Three critical aspects, which potentially impair NMP sessions significantly, were discussed in this work.

A major drawback of NMP is the unreliability of the transport medium. The packet-segmented audio stream can be interrupted by lost or belated network packets anytime and hence the application of an error concealment strategy is indispensable. Several restrictions should be respected in the context of NMP. First, the concealment strategy must not contribute further delay to the overall NMP delay and therefore any interpolation of frames in time or frequency-domain can't be applied. Second, the concealment strategy should be applied to the time-domain signal to be applicable with any audio data compression technique. Two concealment methods with different focuses were analyzed in this work to provide application-matched algorithms.

The first proposed concealment strategy is based on auto-regressive modeling of the transmitted signal. Whenever the concealment strategy is triggered the auto-regressive model is created by computing prediction coefficients based on previous audio frames. A recursive filter is initialized with these prediction filter coefficients to realize a smooth transition from previous audio data to the concealment data. The recursive filter is then fed

with silence to synthesize the concealment signal. The length of the concealment signal exceeds the system's block size to allow cross-fading with the next received audio frame. Multiple methods to compute the auto-regressive model were investigated. In this study the author used the *Least Mean Square* (LMS), *Gradient-Adaptive Lattice* (GAL), Autocorrelation, and the Burg method. Additionally the influence of block lengths, prediction order, and cross-fading window curve was investigated. An objective measurement based on the *Perceptual Evaluation of Audio Quality* (PEAQ) algorithm and a subjective listening test showed that an auto-regressive model computed with the Burg method clearly outperforms the other methods. Nevertheless, all methods were rated better than simple muting concealment.

The major drawback of the proposed auto-regressive modeling approach for error concealment is its algorithmic complexity. A second concealment method was developed to counteract the problem of high computational cost. This approach is based on the extraction and looping of previous periodic audio data segments. The first step to implement the concealment method is to perform a zero-crossing analysis of previous data to detect periodic segments. The analysis is optionally enhanced by restricting the bandwidth of the signal to be analyzed and by intensifying the first harmonic with help of a non-linearity. Depending on the amount of zero-crossings detected, a segment of one or multiple periods is cut out at the borders defined by the zero-crossings. Next the extracted segment is phase-aligned to previous data to allow a smooth transition. Additionally, previous data is extrapolated and cross-faded with the phase-aligned extracted segment. The extracted segment is then repeated to create a sequence of sufficient length. Similar to the first method the last step is to cross-fade the concealment signal with the next intact audio data segment. A clear improvement of the audio quality in contrast to muting could be experimentally affirmed. However, the perceptual quality of the auto-regressive modeling approach could not be reached. The actual intention of reducing the complexity was clearly fulfilled since the amount of multiplications is significantly smaller, especially for higher orders of the auto-regressive model which led to the best results. Omitting the optional preprocessing reduces the complexity even further.

In addition to pointing out the necessity of error concealment in an NMP session and the proposal of two methods to perform concealment in the time-domain, the major contribution of this work is an audio data compression technique especially fitted to NMP requirements. A crucial key requirement of a successful, satisfactory NMP session is a very low latency between the musicians to provide a near real-time experience. Hence, the applied audio compression technique should feature as little delay as possible. Well-known block-based encoding techniques likes MP3 or AAC are based upon time-

frequency transforms of long blocks which result in delays above 50 ms and hence can't be applied. Low delay codecs like ULD and OPUS still feature significant algorithmic delay. Utilizing delay-less prediction-based codecs like *Adaptive Differential Pulse Code Modulation* (ADPCM) is not feasible in most scenarios due to their high data rate. Apparently, there is a strong demand for an audio encoding technique that is capable of delivering high audio quality with small data rates and smallest delays. The proposed codec strategy combines the delay-less encoding possibilities of ADPCM encoders with the advantage of a vector quantizer. Multiple ADPCM encoders are applied in critically sampled subbands to compute subband prediction residuals which are jointly quantized with a vector quantizer. The subband signals are obtained using a critically-sampled cosine-modulated FIR filterbank. The filterbank prototype filter is designed using the window method and a hanning window. The most-promising design with eight bands and an impulse response length of 51 samples, corresponding to small delay of 1.1 ms at a sampling rate of 44.1 kHz, is utilized in the audio codec. Although the filter bank is not perfectly reconstructing, it outperforms perfectly reconstructible designs due its superior transition steepness and stopband attenuation.

The subband ADPCM encoders are implemented using *Gradient Adaptive Lattice* (GAL) prediction filters which were already successfully applied in several ADPCM broadband codecs. GAL filters are advantageous due to the guaranteed minimal phase characteristic and hence a stable inverse prediction filter when the filter coefficients are limited to a certain amplitude range. The prediction is performed in a backward manner using the quantized prediction residuals to allow synchronous adaption of prediction filter coefficients in encoder and decoder. All subband residuals, computed by subtracting the predicted signal from the subband signals, are normalized using recursively estimated envelope estimates per subband. The normalization results in an almost unit variance and hence an optimized utilization of the quantizers amplitude range. The vector quantizer codebook is constructed using noise featuring a gaussian distribution and a size corresponding to 2 bits per sample. The cost function to search the best fitting code book entry for to the subband residual vector is based on the euclidean distance but weighted with the subband envelope estimates. This weighting of the cost function corresponds to a rearrangement of subband *Signal-to-Noise Ratios's* (SNR's) from higher to lower bands. Hence the perceptually relevant lower bands feature less quantization noise which results in a significantly increased perceptual quality of the codec.

The *Vector-Quantized Adaptive Differential Pulse Code Modulation* (VQ-ADPCM) is capable of delivering perceptually pleasant results at a bitrate of 88.2 kbit/s. Unfortunately, the linear search in the codebook prohibits the

real-time usage. Therefore, the linear search is replaced with the *Nearest Neighbor Search* (NNS), which iteratively searches the code book entry with the smallest cost function deviation within a set of neighboring codebook entries surrounding the current codebook entry. The routine terminates after a certain amount of iterations or whenever code book entries with smaller distortion can't be found. The implementation of NNS with 100 neighbors reduces the code book search complexity from 65536 cost function computations to 327 in average. This significant reduction allows real-time usage although the perceptual quality is slightly reduced since NNS doesn't guarantee to find a global minimum.

Analyzing the distribution of actually transmitted code book indexes revealed the preference of code book entries featuring small euclidean norms. This trend is caused by the predominant harmonic and partly stationary components of music material which tend to produce small subband prediction residuals. On the basis of the measured code book entry distribution, an almost optimal Huffman encoder was designed which reduces the average word length to 1.447 bit per sample corresponding to a bit rate of 64 kbit/s at a sampling rate of 44.1 kHz.

The proposed codec structure and especially the subband predictors need to be carefully adjusted to the characteristics of the subband signals to achieve the best possible quality. The predictors have to be optimized in terms of prediction order, base step size, and a stability constant. Additionally, the envelope estimation is parametrized with a smoothing coefficient for the attack and release case, respectively. A two step optimization approach was used to identify these parameters. First, an estimate of the prediction order and base step size was determined by applying the gradient descent algorithm for every subband on a given set of prediction orders using a cost function based on the prediction error energy. The optimization was performed using the *Sound Quality Assessment Material* (SQAM) dataset. The results were used as initial parameters for the second optimization step which jointly optimizes all previously mentioned parameters. In contrast to the first optimization approach, the cost function is based on PEAQ measurements to optimize the perceptual quality of the codec. Simulated annealing is applied as an optimization routine due to its heuristic character and its capability of hill climbing to escape local minima. Applying the identified parameter set leads to acoustically pleasant results for most SQAM items. The results indicate that the VQ-ADPCM approach is capable of providing good audio quality of -0.85 ODG at a low bitrate of about 64 kbit/s at a very small delay of 1.1 ms. These characteristic values imply that the VQ-ADPCM is an attractive alternative coding approach for NMP systems. Although the audio stream can be concealed using the methods from the first chapter, a

real-world NMP implementation using the proposed codec must be extended to synchronize the predictor coefficients in encoder and decoder in the case of lost packets.

The last contribution of this work is a method for the spatial enhancement of NMP platforms which are based on single channel audio streams. Typical NMP platforms allow to define the volume and panning of the NMP participants sound signals. Thus, solely the definition of point sources within a stereo panorama is possible. Neglecting the spatial character of sound sources, especially of large instruments, is likely to limit the realism and therefore the user experience of the audio replay. This restriction can be resolved by applying a pseudo-stereo algorithm which blindly estimates a pair of stereo signals from a mono signal. A novel pseudo-stereo method is proposed which is capable of delivering a listening experience close to a real stereo signal by applying a set of complementary linear-phase filters which can be applied in time- or frequency-domain. The filter pair implements diffuse amplitude panning with a high frequency resolution to create decorrelated signals providing spatial width and depth. In contrast to other well-known pseudo-stereo techniques, the novel approach doesn't add any timbral coloration or reverberation. Furthermore, the complementary design guarantees downmix compatibility to allow replay with mono loudspeakers in a stereo setup or subsequent mixing of pseudo-stereo processed material.

The proposed pseudo stereo can also be integrated in a virtual surround render engine. The well-known method of rendering mono sources to a arbitrary positions in space using *Head-Related Impulse Responses* (HRIRs) results in point sources. First computing a pseudo-stereo signal and panning it to slightly different positions in space allows to define the size of the sound source and hence potentially yields a listening environment with enhanced naturalness.

The three proposed NMP enhancements are not bound to any specific NMP platform and therefore can be integrated in any existing NMP framework to improve the error robustness, algorithmic latency caused by audio coding, and spaciousness of the audio replay.

A

Appendix

A.1 Partitioning SQAM into instrument classes

Table A.1: Partition of SQAM dataset into instrument subclasses.

Track	Brass	Keys	Percussion	Speech	Strings	Synthetic	Tuned Percussion	Woods	Vocals
01 Sine 1KHz -20,-10,0dB						✓			
02 Band-ltd. pink noise						✓			
03 Electr. gong 100 Hz						✓			
04 Electr. gong 400 Hz						✓			
05 Electr. gong 5 kHz						✓			
06 Electr. gong 500 Hz vib.						✓			
07 Electr. tune						✓			
08 Violin					✓				
09 Viola					✓				
10 Violoncello					✓				
11 Double-bass					✓				
12 Piccolo								✓	
13 Flute								✓	
14 Oboe								✓	
15 Cor anglais								✓	
16 Clarinet								✓	
17 Bass-clarinet								✓	
18 Bassoon								✓	

Track	Brass	Keys	Percussion	Speech	Strings	Synthetic	Tuned Percussion	Woods	Vocals
19 Contra-bassoon								✓	
20 Saxophone								✓	
21 Trumpet	✓								
22 Trombone	✓								
23 Horn	✓								
24 Tuba	✓								
25 Harp		✓							
26 Claves st,rhythm			✓						
27 Castanets			✓						
28 Side drum			✓						
29 Bass drum			✓						
30 Kettle-drums			✓						
31 Cymbal soft,hard stick			✓						
32 Triangles			✓						
33 Gong forte,piano			✓						
34 Tubular bells							✓		
35 Glockenspiel							✓		
36 Xylophone							✓		
37 Vibraphone							✓		
38 Marimba							✓		
39 Grand piano		✓							
40 Harpsichord		✓							
41 Celesta		✓							
42 Accordion		✓							
43 Organ		✓							
44 Soprano									✓
45 Alto									✓
46 Tenor									✓
47 Bass									✓
48 Quartet									✓
49 Fem. speech English				✓					
50 Male speech English				✓					
51 Fem. speech French				✓					
52 Male speech French				✓					
53 Fem. speech German				✓					
54 Male speech German				✓					

Track	Brass	Keys	Percussion	Speech	Strings	Synthetic	Tuned Percussion	Woods	Vocals
55 Trumpet Haydn									
56 Organ Handel		✓							
57 Organ Bach		✓							
58 Guitar Sarasate									
59 Violin Ravel									
60 Piano Schubert		✓							
61 Soprano Mozart									
62 Soprano Spiritual									✓
63 Soloists Verdi									
64 Choir Orff									
65 Orchestra R Strauss									
66 Wind ens. Stravinsky									
67 Wind ens. Mozart									
68 Orchestra Baird									
69 Abba									
70 Eddie Rabbitt									

A.2 Utilized codec libraries

Table A.2: Utilized codec libraries.

Library	Supplier	Version
libmp3lame	The LAME project	3.99.5
Fraunhofer FDK AAC	Fraunhofer IIS	0.6.1
Nero AAC Encoder	Nero AG	1.5.4.0
Opus tools	Xiph.Org Foundation	0.18 (libopus 1.1)

A.3 Relation of correlation and cross-fading curves

The cross-fading of two signals $x_1(n)$ and $x_2(n)$ can be described as

$$x_{\text{mix}}(n) = w(n)\, x_1(n) + w_i(n)\, x_2(n), \quad n \in [0, \ldots, N-1], \tag{A.1}$$

where $w(n)$ and $w_i(n)$ are the fading and the inverse fading curve. Assuming signals with the same power and hence same signal variance

$$\sigma_x^2 = \mathrm{E}\left[x_1^2\right] = \mathrm{E}\left[x_2^2\right]$$

and identical mean value

$$\mu_x = E\left[x_1\right] = E\left[x_2\right] = 0$$

allows to estimate the variance of the faded signal

$$\begin{aligned}
\mathrm{E}\left[x_{\text{mix}}^2\right] &= \mathrm{E}\left[(w\, x_1 + w_i\, x_2)^2\right] \\
&= \mathrm{E}\left[w^2 x_1^2 + 2\, w\, w_i\, x_1\, x_2 + w_i^2 x_2^2\right] \\
&= w^2 \mathrm{E}\left[x_1^2\right] + 2\, w\, w_i\, \mathrm{E}\left[x_1 x_2\right] + w_i^2 \mathrm{E}\left[x_2^2\right] \\
&= w^2 \sigma_x^2 + 2\, w\, w_i\, \mathrm{Cov}(x_1, x_2) + w_i^2 \sigma_x^2.
\end{aligned} \tag{A.2}$$

Expressing the covariance as the product of the signal variances and the correlation coefficient $\mathrm{Cov}(x_1, x_2) = r_{x_1,x_2}\, \sigma_x^2$ yields

$$\begin{aligned}
\mathrm{E}\left[x_{\text{mix}}^2\right] &= w^2 \sigma_x^2 + 2\, w\, w_i\, r_{x_1,x_2} \sigma_x^2 + w_i^2 \sigma_x^2 \\
&= \sigma_x^2 (w^2 + 2\, w\, w_i\, r_{x_1,x_2} + w_i^2).
\end{aligned} \tag{A.3}$$

It is typically desirable that the cross-faded signal features the same power and hence variance as the input signals

$$\mathrm{E}\left[x_{\text{mix}}^2\right] = \sigma_x^2 (w^2 + 2\, w\, w_i\, r_{x_1,x_2} + w_i^2) = \sigma_x^2.$$

Canceling the variance holds

$$w^2 + 2\, w\, w_i\, r_{x_1,x_2} + w_i^2 = 1. \tag{A.4}$$

Therefore, the fading curve depends on the correlation coefficient

$$r_{x_1,x_2} = \frac{1 - (w^2 + w_i^2)}{2\, w\, w_i} \tag{A.5}$$

and can be implicitly described as

$$w_i(n) = -w(n)\, r_{x_1,x_2} + \sqrt{w^2(n)\, r_{x_1,x_2}^2 - w^2(n) + 1}.$$

A.3.1 Amplitude-complementary fading curve

Substituting the inverse fading curve with the amplitude-complementary curve

$$w_i = 1 - w$$

in Eq. (A.5) holds

$$\begin{aligned} r_{x_1,x_2} &= \frac{1 - (w^2 + w_i^2)}{2\,w\,w_i} = \\ &= \frac{1 - w^2 - 1 + 2\,w - w_i^2}{2\,w - 2\,w^2} = \\ &= \frac{-2\,w^2 + 2\,w}{-2\,w^2 + 2\,w} = 1. \end{aligned}$$

Hence, correlated signals featuring a correlation coefficient $r_{x_1,x_2} \approx 1$ should be cross-faded using amplitude-complementary curves like the linear fading curve $w(n) = \frac{n}{N-1}, \quad n \in [0, \ldots, N-1]$.

A.3.2 Power-complementary fading curve

Applying the power-complementary curve

$$w_i(n) = \sqrt{1 - w^2(n)}$$

into Eq. (A.5) holds

$$\begin{aligned} r_{x_1,x_2} &= \frac{1 - (w^2 + w_i^2)}{2\,w\,w_i} = \\ &= \frac{1 - (w^2 + 1 - w^2)}{2\,w\,w_i} = 0. \end{aligned}$$

Consequently, cross-fading uncorrelated signals, which are characterized by $r_{x_1,x_2} \approx 0$, must be realized with power-complementary curves to achieve power preservation. Typical power complementary curves are the square-root curve $w_i(n) = \sqrt{\frac{n}{N-1}}$ and the cosine curve $w_i(n) = \cos\left(\frac{n\,\pi}{2(N-1)}\right)$.

A.3.3 Correlation-based fading curve design

Exact power-complementary cross-fading of partially correlated is also possible with analytically designed fading curves. In the following, the amplitude ratio of w and w_i shall be described with the function

$$f(\alpha) = \frac{w(\alpha)}{w_i(\alpha)}, \tag{A.6}$$

where $\alpha = \frac{n}{N-1}$ denotes the normalized time index. Rewriting Eq. (A.6) to $w_i(\alpha) = f(\alpha)\, w(\alpha)$ and inserting it into Eq. (A.4) holds

$$w_i^2(\alpha) = \frac{1}{1 + 2\, f(\alpha)\, r_{x_1,x_2} + f^2(\alpha)}. \tag{A.7}$$

Multiple assumptions concerning $f(\alpha)$ can be made:

1. $f(0) = 0$ since the fading curve w starts with an amplitude of 0
2. $f(1) = \infty$ since the inverse fading curve w_i end with an amplitude of 0
3. $f(0.5) = 1$ since the fading curves w and w_i are symmetric and hence feature the same amplitude in the center of the curves.

Several functions fulfill these requirements. In the following, the tangent function $f_{\tan}(\alpha) = \tan(\frac{\pi\,\alpha}{2})$ and the function $f_{\text{scl}} = \frac{\alpha}{1-\alpha}$ are utilized and applied in Eq. (A.7) to hold the inverse fading curves

$$\begin{aligned} w_{i,\tan}(\alpha) &= \frac{1}{\sqrt{1 + 2\, \tan(\frac{\pi\,\alpha}{2})\, r_{x_1,x_2} + \tan^2(\frac{\pi\,\alpha}{2})}} \\ &= \frac{\cos(\frac{\pi\,\alpha}{2})}{\sqrt{1 + 2\, r_{x_1,x_2}\, \sin(\frac{\pi\,\alpha}{2})\, \cos(\frac{\pi\,\alpha}{2})}} \end{aligned} \tag{A.8}$$

and

$$\begin{aligned} w_{i,\text{scl}}(\alpha) &= \frac{1}{\sqrt{1 + 2\, \frac{\alpha}{1-\alpha}\, r_{x_1,x_2} + \frac{\alpha^2}{(1-\alpha)^2}}} \\ &= \frac{1-\alpha}{\sqrt{1 - 2\,(1 - r_{x_1,x_2})\,\alpha\,(1-\alpha)}}. \end{aligned} \tag{A.9}$$

Correspondingly, the fading curves are derived as

$$w_{\tan}(\alpha) = \frac{\sin(\frac{\pi\,\alpha}{2})}{\sqrt{1 + 2\, r_{x_1,x_2}\, \sin(\frac{\pi\,\alpha}{2})\, \cos(\frac{\pi\,\alpha}{2})}} \tag{A.10}$$

and

$$w_{\text{scl}}(\alpha) = \frac{\alpha}{\sqrt{1 - 2\,(1 - r_{x_1,x_2})\,\alpha\,(1-\alpha)}}. \tag{A.11}$$

The resulting curves functions are illustrated in Fig. A.1. Apparently, the $w_{\tan}$ curve evolves from a sine curve to a slightly S-shaped curve for decreasing correlation values. The second fading curve w_{scl} changes from a non-symmetric S-shaped curve to the linear curve. It should also be noted,

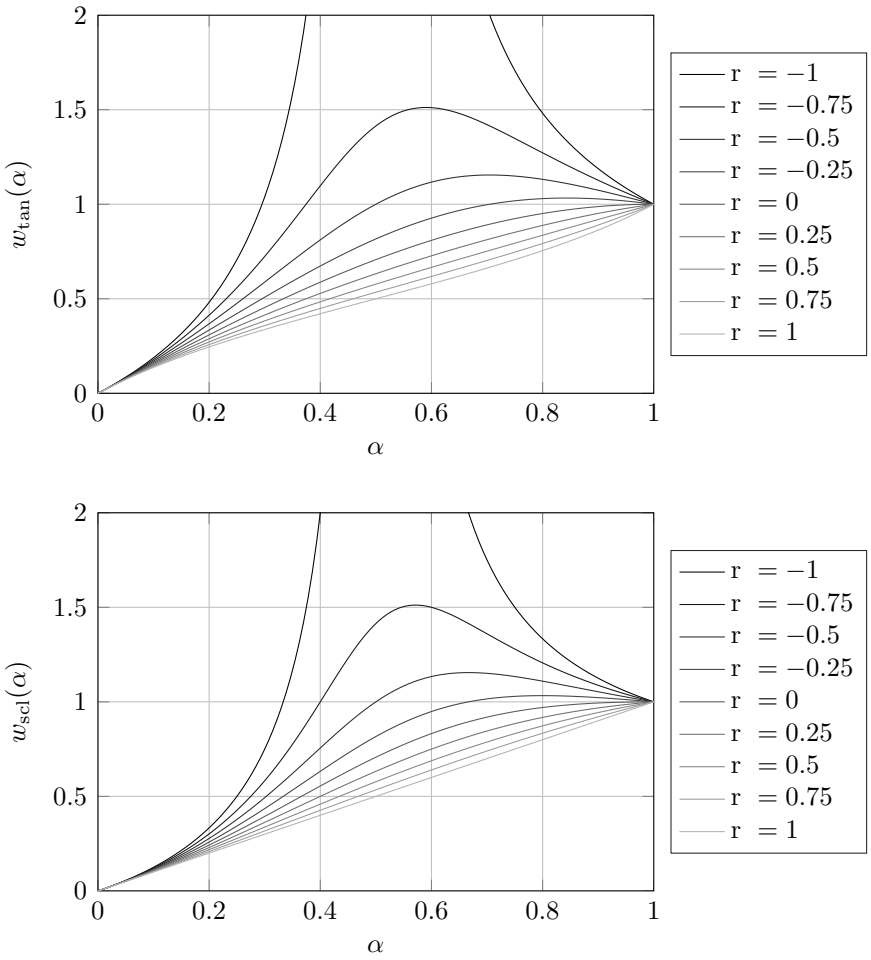


Figure A.1: Power-complementary cross-fading curves plotted over correlation coefficient r.

that the proposed cross-fading curve design also works flawlessly for negatively correlated signals. Certainly the amplitude of the cross-fading curves increases to compensate the power loss caused by destructive interference. However, cross-fading completely negatively correlated ($r_{x_1,x_2} = -1$) remains an undesirable scenario since infinite amplification is required.

List of Selected Symbols

Chapter 2:

N	Block length
M	Amount of blocks
n	Discrete time index
m	Block index
$x(n)$	Discrete time-domain input signal
$y(n)$	Discrete time-domain output signal
$\mathbf{a}$	Auto-regressive filter coefficients
$\hat{x}(n)$	Estimation of $x(n)$
p	Prediction order
$e(n)$	Prediction residual
$f_i(n)$	Forward prediction error in lattice stage i
$b_i(n)$	Backward prediction error in lattice stage i
k_i	Reflection coefficient of lattice stage i
μ/μ_i	Gradient step size / Gradient step size in lattice stage i
σ^2/σ_i^2	Error power / Error power in lattice stage i
λ	Base gradient step size
$J(\mathbf{X})$	Cost function for parameter set $\mathbf{X}$
$\mathbf{y}_m(n)$	m_{th} block
$x_p(n)$	Previous data
$x_c(n)$	Concealment signal
$\mathbf{z}$	Filter states
O	Overlap
$w(n)$	Cross-fading curve

Chapter 3:

f_s	Sampling frequency
w_b	Word length
R_b	Data rate
M	Amount of subbands
m	Subband index
b	Discrete time index in subbands
$x_m(b)$	Subband signals
$e_m(b)$	Subband residuals
$v_m(b)$	Subband envelope estimate
$\bar{e}_m(b)$	Normalized subband residuals
$i(b)$	Code word index of frame b
$\mathbf{C}$	Codebook
$\tilde{e}_m(b)$	Quantized subband residuals
$H_m(z)/G_m(z)$	Analysis / Synthesis Subbandfilter

Chapter 4:

S	Number of Sources
s	Source index
g_s	Gain factor of source s
x_s	Time-domain signal of source s
$x_{L/R}(n)$	Left / Right time-domain signal
$H_{L/R}(z)$	Left / Right decorrelation filter
ϕ_s	Azimuth angle of source s
$G_{L/R}^{\phi_s}(z)$	HRTF at azimuth angle ϕ_s

List of Figures

List of Tables

Bibliography

[AB14] Chrisoula Alexandraki and Rolf Bader. Using Computer Accompaniment to Assist Networked Music Performance. In *Audio Engineering Society Conference: 53rd International Conference: Semantic Audio*, London, UK, Jan 2014.

[AGHS99] Eric Allamanche, Ralf Geiger, Juergen Herre, and Thomas Sporer. MPEG-4 Low Delay Audio Coding Based on the AAC Codec. In *Audio Engineering Society Convention 106*, Munich, Germany, May 1999.

[AGS08] Jens Ahrens, Matthias Geier, and Sascha Spors. The SoundScape Renderer: A Unified Spatial Audio Reproduction Framework for Arbitrary Rendering Methods. In *Audio Engineering Society Convention 124*, Amsterdam, Netherlands, May 2008.

[APT] aptX® Low Latency. Online: `http://www.csr.com/products/aptx-low-latency` [Accessed: 03-03-2016].

[Bau69] Benjamin B. Bauer. Some Techniques Toward Better Stereophonic Perspective. *Journal Audio Engineering Society*, 17(4):410–415, 1969.

[BD98] C. Philip Brown and Richard O. Duda. A Structural Model for Binaural Sound Synthesis. *IEEE Transactions on Speech and Audio Processing*, 6(5):476–488, 1998.

[BNA13] Dienstequalität von Breitbandzugängen II. Technical report, Bundesnetzagentur, 2013.

[BSI+12] Øyvind Brandtsegg, Sigurd Saue, John Pål Inderberg, Victor Lazzarini, Axel Tidemann, Jan Tro, Jøran Rudi, and Notto JW Thelle. The Development of an Online Course in DSP Eartraining. In *Proceedings of the 17th International Conference on Digital Audio Effects (DAFx-12), September 17-21, 2012*, York, UK, 2012.

[Car09] Alexander Carôt. *Musical Telepresence - A Comprehensive Analysis Towards New Cognitive and Technical Approaches*. PhD thesis, Universität zu Lübeck, 2009.

[Čer85] Vladimír Černỳ. Thermodynamical Approach to the Traveling Salesman Problem: An Efficient Simulation Algorithm. *Journal of optimization theory and applications*, 45(1):41–51, 1985.

[CGLT04] Chris Chafe, Michael Gurevich, Grace Leslie, and Sean Tyan. Effect of Time Delay on Ensemble Accuracy. In *International Symposium on Musical Acoustics*, Nara, Japan, March 2004.

[CKS06] Alexander Carôt, Ulrich Krämer, and Gerald Schuller. Network Music Performance (NMP) in Narrow Band Networks. In *Audio Engineering Society Convention 120*, Paris, France, May 2006.

[CM75] David L. Cohn and James L. Melsa. The Relationship between an Adaptive Quantizer and a Variance Estimator. *IEEE Transactions on Information Theory*, 21(6):669–671, Nov 1975.

[CM95] Charles D. Creusere and Sanjit K. Mitra. A Simple Method for Designing High-quality Prototype Filters for M-band Pseudo QMF Banks. *Signal Processing, IEEE Transactions on*, 43(4):1005–1007, Apr 1995.

[CS11a] Alexander Carôt and Gerald Schuller. Applying Video to Low Delayed Audio Streams In Bandwidth Limited Networks. In *Audio Engineering Society Conference: 44th International Conference: Audio Networking*, San Diego, USA, November 2011.

[CS11b] Alexander Carôt and Gerald Schuller. Towards a Telematic Visual-conducting System. In *Audio Engineering Society Conference: 44th International Conference: Audio Networking*, San Diego, USA, November 2011.

[CSZ+04] Elaine Chew, Alexander Sawchuk, Roger Zimmermann, Ilia Tosheff, Christos Kyriakakis, Christos Papadopoulos, Alexandre

François, and Anja Volk. Distributed Immersive Performance. In *In National Association of Schools of Music*, 2004.

[CW09] Alexander Carôt and Christian Werner. External Latency-optimized Soundcard Synchronization for Applications in Wide-area Networks. In *AES 14th regional convention*, Tokio, Japan, July 2009.

[CWT09] Alexander Carôt, Christian Werner, and Fischinger Timo. Towards a Comprehensive Cognitive Analysis of Delay-influenced Rythmical Interaction. In *International Computer Music Confrernce (ICMC) 2009*, Montreal, Canada, August 2009.

[Dur60] James Durbin. The Fitting of Time-series Models. *Revue de l'Institut International de Statistique*, 28(3):233–244, Jan 1960.

[Dya11] Igor Dyakonov. Results of the Public Multiformat Listening Test @ 64 kbps. Online: `http://listening-tests.hydrogenaud.io/igorc/results.html` [Accessed: 03-03-2016], March 2011.

[EBU08] EBU – TECH 3253, Sound Quality Assessment Material Recordings for Subjective Tests. Technical report, European Broadcasting Union, September 2008.

[EC12] Quality of Broadband Services in the EU March 2012. Technical report, European Commission DG Communications Networks, Content & Technology, 2012.

[Fal05] Christof Faller. Pseudostereophony Revisited. In *Audio Engineering Society Convention 118*, Barcelona, Spain, May 2005.

[FHZ13] Marco Fink, Martin Holters, and Udo Zölzer. Comparison of Various Predictors for Audio Extrapolation. In *Proceedings of the 16th International Conference on Digital Audio Effects (DAFx-13), September 2-5, 2013*, Maynooth, Ireland, 2013.

[Fli93] Norbert Fliege. *Multiraten-Signalverarbeitung: Theorie und Anwendungen*. Teubner Informationstechnik. Teubner, 1993.

[FZ14] Marco Fink and Udo Zölzer. Low-Delay Error Concealment with Low Computational Overhead for Audio over IP Applications. In *Proceedings of the 17th International Conference on Digital Audio Effects (DAFx-14), September 1-5, 2014*, Erlangen, Germany, 2014.

[GC83] Allen Gersho and Vladimir Cuperman. Vector Quantization: A Pattern-matching Technique for Speech Coding. *Communications Magazine, IEEE*, 21(9):15–21, December 1983.

[Ger92] Michael A. Gerzon. Signal Processing for Simulating Realistic Stereo Images. In *Audio Engineering Society Convention 93*, San Francisco, USA, Oct. 1992.

[GLS+04] Marc Gayer, Manfred Lutzky, Gerald Schuller, Ulrich Kraemer, and Stefan Wabnik. A Guideline to Audio Codec Delay. In *Audio Engineering Society Convention 116*, Berlin, Germany, May 2004.

[GM01] Emre Gündüzhan and Kathryn Momtahan. Linear Prediction Based Packet Loss Concealment Algorithm for PCM Coded Speech. *Speech and Audio Processing, IEEE Transactions on*, 9(8):778–785, Nov 2001.

[GZ03] Saeed Gazor and Wei Zhang. Speech Probability Distribution. *Signal Processing Letters, IEEE*, 10(7):204–207, July 2003.

[Hay91] Simon Haykin. *Adaptive Filter Theory*. Prentice Hall, Englewood Cliffs, NJ, USA, 2_{nd} edition, 1991.

[Hes79] Wolfgang Hess. Time-domain Pitch Period Extraction of Speech Signals using Three Nonlinear Digital Filters. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '79.*, Washington D.C., USA, April 1979.

[HHZ08] Martin Holters, Christian R. Helmrich, and Udo Zölzer. Delay-free Audio Coding based on ADPCM and Error Feedback. In *Proceedings of the 11th International Conference on Digital Audio Effects (DAFx-08), Sep 1-4, 2008*, Espoo, Finland, 2008.

[HKK+04] Jens Hirschfeld, Juliane Klier, Ulrich Kraemer, Geral Schuller, and Stefan Wabnik. Ultra Low Delay Audio Coding with Constant Bit Rate. In *Audio Engineering Society Convention 117*, San Francisco, USA, Oct. 2004.

[HKSW06] Jens Hirschfeld, Ulrich Kraemer, Gerald Schuller, and Stefan Wabnik. Reduced Bit Rate Ultra Low Delay Audio Coding. In *Audio Engineering Society Convention 120*, Paris, France, May 2006.

[Huf52] David A. Huffman. A Method for the Construction of Minimum-Redundancy Codes. *Proceedings of the IRE*, 40(9):1098–1101, Sept 1952.

[HZ08] Martin Holters and Udo Zölzer. Delay-free Lossy Audio Coding using Shelving Pre- and Post-filters. In *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, Las Vegas, USA, March 2008.

[HZ09] Martin Holters and Udo Zölzer. Automatic Parameter Optimization for a Perceptual Audio Codec. In *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, Taipei, Taiwan, April 2009.

[HZ15] Martin Holters and Udo Zölzer. GstPeaq - An Opensource Implementation of the PEAQ Algorithm. In *Proceedings of the 18th International Conference on Digital Audio Effects (DAFx-15), Nov 30 - Dec 3, 2015*, Trondheim, Norway, 2015.

[ISO93] Information Technology – Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1,5 Mbit/s – Part 3: Audio. Technical report, International Organization for Standardization, 1993.

[ISO97] Information Technology – Generic Coding of Moving Pictures and Associated Audio Information – Part 7: Advanced Audio Coding (AAC). Technical report, International Organization for Standardization, 1997.

[ISO09] Information Technology – Coding of Audio-visual Objects – Part 3: Audio. Technical report, International Organization for Standardization, 2009.

[ITU88] ITU Recommendation ITU-R G.721,32 kbit/s Adaptive Differential Pulse Code Modulation (ADPCM). Technical report, International Telecommunications Union, November 1988.

[ITU97] ITU Recommendation ITU-R BS.1116-1, Methods for the Subjective Assessment of Small Impairments in Audio Systems including Multichannel Sound Systems. Technical report, International Telecommunications Union, October 1997.

[ITU01] ITU Recommendation ITU-R BS.1387-1, Method for Objective Measurements of Perceived Audio Quality. Technical report, International Telecommunications Union, November 2001.

[ITU14] ITU Recommendation ITU-R BS.1534-2, Method for the Subjective Assessment of Intermediate Quality Levels of Coding Systems. Technical report, International Telecommunications Union, June 2014.

[Kei06a] Florian Keiler. *Beiträge zur Audiocodierung mit kurzer Latenzzeit*. PhD thesis, Helmut-Schmidt-Universität, 2006.

[Kei06b] Florian Keiler. Real-Time Subband-ADPCM Low-Delay Audio Coding Approach. In *Audio Engineering Society Convention 120*, Paris, France, May 2006.

[Ken95] Gary S. Kendall. The Decorrelation of Audio Signals and its Impact on Spatial Imagery. *Computer Music Journal*, 19(4):72–87, 1995.

[KGV83] Scott Kirkpatrick, C. Daniel Gelatt, and Mario P. Vecchi. Optimization by Simulated Annealing. *Science*, 200(4598):671–680, May 1983.

[KKZS03] Florian Keiler, Can Karadogan, Udo Zölzer, and Albrecht Schneider. Analysis of Transient Musical Sounds by Autoregressive Modeling. In *Proceedings of the 6th International Conference on Digital Audio Effects (DAFx-03), September 8-11, 2003*, London, UK, 2003.

[KLZ13] Sebastian Kraft, Alexander Lerch, and Udo Zölzer. The Tonalness Spectrum: Feature-Based Estimation of Tonal Components. In *Proc. of the 16th Int. Conference on Digital Audio Effects (DAFx-13), September 2-5, 2013*, Maynooth, Ireland, 2013.

[KR02] Ismo Kauppinen and Kari Roth. Audio Signal Extrapolation - Theory and Applications. In *Proc. of the 5th Conference on Digital Audio Effects (DAFx-02), September 26-28, 2002*, Hamburg, Germany, 2002.

[KZ14] Sebastian Kraft and Udo Zölzer. BeaqleJS: HTML5 and JavaScript based Framework for the Subjective Evaluation of

Audio Quality. In *Linux Audio Conference, LAC 2014*, Karlsruhe, Germany, May 2014.

[Lau54] Holger Lauridsen. Nogle forsog reed forskellige former rum akustik gengivelse. *Ingeniøren*, 47:906, 1954.

[LBG80] Y. Linde, A. Buzo, and R. Gray. An Algorithm for Vector Quantizer Design. *IEEE Transactions on Communications*, 28(1):84–95, Jan 1980.

[Ler12] Alexander Lerch. *An Introduction to Audio Content Analysis: Applications in Signal Processing and Music Informatics*. A John Wiley & Sons, Inc., 1_{st} edition, 2012.

[LW07] Alexander Lindau and Stefan Weinzierl. Fabian - Schnelle Erfassung binauraler Raumimpulsantworten in mehreren Freiheitsgraden. In *Fortschritte der Akustik: Tagungsband d. 33. DAGA*, Stuttgart, 2007.

[Mak78] John Makhoul. A Class of All-zero Lattice Digital Filters: Properties and Applications. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 26(4):304–314, 1978.

[Mal90] Henrique S. Malvar. Modulated QMF Filter Banks with Perfect Reconstruction. *Electronics Letters*, 26(13):906–907, June 1990.

[Max60] Joel Max. Quantizing for Minimum Distortion. *IRE Transactions in Information Theory*, 6:7–12, March 1960.

[MFZ14] Florian Meier, Marco Fink, and Udo Zölzer. The JamBerry - A Stand-Alone Device for Networked Music Performance Based on the Raspberry Pi. In *Linux Audio Conference, LAC 2014*, Karlsruhe, Germany, May 2014.

[MHJS95] Henrik Møller, Dorte Hammershøi, Clemen Boje Jensen, and Michael Friis Sørensen. Transfer Characteristics of Headphones Measured on Human Ears. *J. Audio Eng. Soc*, 43(4):203–217, 1995.

[MRG85] John Makhoul, Salim Roucos, and Herbert Gish. Vector Quantization in Speech Coding. *Proceedings of the IEEE*, 73:1551–1588, 1985.

[MZFR00] D. Martinez Munoz, M. Rosa Zurera, F. Lopez Ferreras, and N. Ruiz Reyes. Low Delay Audio Coder Using Adaptive Vector Quantization. In *10th European Signal Processing Conference, Eusipco 2000*, Tampere, Finaland, Sept 2000.

[NV90] Truong Q. Nguyen and Palghat P. Vaidyanathan. Structures for M-channel Perfect-reconstruction FIR QMF Banks which Yield Linear-phase Analysis Filters. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 38(3):433–446, Mar 1990.

[OCD+13] André Oliveira, Guilherme Campos, Paulo Dias, Damian Thomas Murphy, José Viera, Catarina Mendonça, and Jorge Santos. Real-time Dynamic Image-source Implementation for Auralisation. In *Proc. of the 16th Int. Conference on Digital Audio Effects (DAFx-13), September 2-5, 2013*, Maynooth, Ireland, 2013.

[OFF] Reid Oda, Adam Finkelstein, and Rebecca Fiebrink. Towards Note-Level Prediction for Networked Music Performance. In *Proceedings of the 13th International Conference on New Interfaces for Musical Expression (NIME)*.

[Orb70] Robert Orban. A Rational Technique for Synthesizing Pseudo-Stereo from Monophonic Sources. *Journal Audio Engineering Society*, 18(2):157–164, 1970.

[OS09] Alan V. Oppenheim and Ronald W. Schafer. *Discrete-Time Signal Processing*. Prentice Hall Press, Upper Saddle River, NJ, USA, 3_{rd} edition, 2009.

[PEV10] Matthias Pawig, Gerald Enzner, and Peter Vary. Adaptive Sampling Rate Correction for Acoustic Echo Control in Voice-over-IP. *Signal Processing, IEEE Transactions on*, 58(1):189–199, 2010.

[PHH98] Collin Perkins, Orion Hodson, and Vicky Hardman. A Survey of Packet Loss Recovery Techniques for Streaming Audio. *Network, IEEE*, 12(5):40–48, Sept 1998.

[PM72] Thomas W. Parks and James H. McClellan. Chebyshev Approximation for Nonrecursive Digital Filters with Linear Phase. *Circuit Theory, IEEE Transactions on*, 19(2):189–194, Mar 1972.

[PPP12] Archontis Politis, Tapani Pihlajamäki, and Ville Pulkki. Parametric Spatial Audio Effects. In *Proc. of the 15th Int. Conference on Digital Audio Effects (DAFx-12), September 17-21, 2012)*, York, UK, 2012.

[RMAJ06] Christoffer A. Rødbro, Manohar N. Murthi, Søren V. Andersen, and Søren H. Jensen. Hidden Markov Model-based Packet Loss Concealment for Voice over IP. *Audio, Speech, and Language Processing, IEEE Transactions on*, 14(5):1609–1623, Sept 2006.

[RS78] Lawrence R. Rabiner and Ronald W. Schafer. *Digital Processing of Speech Signals*. Prentice Hall Press, Englewood Cliffs, NJ, USA, 1978.

[Sch58] Manfred R. Schroeder. An Artificial Stereophonic Effect Obtained from a Single Audio Signal. *Journal Audio Engineering Society*, 6(2):74–79, 1958.

[SP12] Jörn Ostermann Stephan Preihs, Fabian-Robert Stöter. Low Delay Error Concealment for Audio Signals. In *Audio Engineering Society Conference: 46th International Conference: Audio Forensics*, Denver, USA, Jun 2012.

[SYRG96] Alexander Stenger, Khaled B. Younes, Richard Reng, and Bernd Girod. A New Error Concealment Technique for Audio Transmission with Packet Loss. In *8th European Signal Processing Conference, 1996. EUSIPCO 1996*, Trieste,Italy, Sept 1996.

[Vai87] Palghat P. Vaidyanathan. Theory and Design of M-channel Maximally Decimated Quadrature Mirror Filters with Arbitrary M, Having the Perfect-reconstruction Property. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 35(4):476–492, Apr 1987.

[VC14] Bogdan Vera and Elaine Chew. Towards Seamless Network Music Performance: Predicting an Ensemble's Expressive Decisions for Distributed Performance. In *Proceedings of the 15th International Society for Music Information Retrieval Conference, ISMIR 2014, Taipei, Taiwan, October 27-31, 2014*, Taipei, Taiwan, 2014.

[VMTV13] Jean-Marc Valin, Grefory Maxwell, Timothy Terriberry, and Koen Vos. High-Quality, Low-Delay Music Coding in the Opus

Codec. In *Audio Engineering Society Convention 135*, New York, USA, May 2013.

[VTMM10] Jean-Marc Valin, Timothy Terriberry, Christopher Montgomery, and Gregory Maxwell. A High-Quality Speech and Audio Codec With Less Than 10-ms Delay. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(1), Jan 2010.

[vTSM10] Friedrich v. Türckheim, Thorsten Smit, and Robert Mores. String Instrument Body Modeling Using FIR Filter Design and Autoregressive Paramter Estimation. In *Proceedings of the 13th International Conference on Digital Audio Effects (DAFx-10), September 6-10, 2010*, Graz, Austria, 2010.

[WAKW93] Elizabeth M. Wenzel, Marianne Arruda, Doris J. Kistler, and Frederic L. Wightman. Localization using Nonindividualized Head-related Transfer Functions. *The Journal of the Acoustical Society of America*, 94(1):111–123, 1993.

[WSD00] Benjamin W. Wah, Xiao Su, and Lin Dong. A Survey of Error-concealment Schemes for Real-time Audio and Video Transmissions over the Internet. *Proceedings International Symposium on Multimedia Software Engineering*, 2000.

[ZF99] Eberhard Zwicker and Hugo Fastl. *Psychoacoustics - Facts and Models*. Springer-Verlag, Berlin, 2_{nd} edition, 1999.

Curriculum Vitae

Personal Data

Marco Fink,
born on February 24_{th} 1987 in Nuremberg

Education

1997 - 2006	Abitur Adam-Kraft-Gymnasium, Schwabach
2006 - 2011	Diplom Ingenieur (M. Sc.) Friedrich-Alexander-University, Erlangen

Bachelor Thesis

"Modeling of Tube Amplifiers using Wave Digital Filters"

Supervised by Prof. Dr.-Ing. Rudolf Rabenstein

Diploma Thesis

"Chromagram Computation in the MDCT Domain based on Psychoacoustic Model"

Supervised by Prof. Dr.-Ing. Walter Kellermann, Dr.-Ing. Arijit Biswas

Vocational

Mar. 2007 - Jan. 2009	Working Student Elektrobit Automotive GmbH, Erlangen
Mar. 2009 - Mar. 2010	Working Student Institute for Information Transmission, Erlangen
April 2010 - October 2010	Internship Dolby Laboratories, Nürnberg
Nov. 2010 - May 2011	Working Student Dolby Laboratories, Nürnberg
June 2011 - Nov. 2011	Graduand Dolby Laboratories, Nürnberg
Dec. 2011 - Jan. 2012	Design Engineer Dolby Laboratories, Nürnberg
Jan. 2012 - March 2016	Research Assistant Dept. of Signal Processing and Communications, Helmut-Schmidt-University, Hamburg
April 2016 - now	Software Engineer Ableton AG, Berlin