Advances in Perceptual Stereo Audio Coding Using Linear Prediction Techniques

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de Technische Universiteit Eindhoven, op gezag van de Rector Magnificus, prof.dr.ir. C.J. van Duijn, voor een commissie aangewezen door het College voor Promoties in het openbaar te verdedigen op dinsdag 15 mei 2007 om 16.00 uur

door

Arijit Biswas

geboren te Calcutta, India

Dit proefschrift is goedgekeurd door de promotoren:

prof.dr. R.J. Sluijter
en
prof.Dr. A.G. Kohlrausch

Copromotor:
dr.ir. A.C. den Brinker

The work described in this thesis has been carried out under the auspices of Philips Research Europe - Eindhoven, The Netherlands.

CIP-DATA LIBRARY TECHNISCHE UNIVERSITEIT EINDHOVEN

Biswas, Arijit

Advances in perceptual stereo audio coding using linear prediction techniques / by Arijit Biswas. - Eindhoven : Technische Universiteit Eindhoven, 2007.
Proefschrift. - ISBN 978-90-386-2023-7
NUR 959
Trefw.: signaalcodering / signaalverwerking / digitale geluidstechniek / spraakcodering.
Subject headings: audio coding / linear predictive coding / signal processing / speech coding.

Samenstelling promotiecommissie:

prof.dr. R.J. Sluijter, promotor
Technische Universiteit Eindhoven, The Netherlands

prof.Dr. A.G. Kohlrausch, promotor
Technische Universiteit Eindhoven, The Netherlands

dr.ir. A.C. den Brinker, copromotor
Philips Research Europe - Eindhoven, The Netherlands

prof.dr. S.H. Jensen, extern lid
Aalborg Universitet, Denmark

Dr.-Ing. G.D.T. Schuller, extern lid
Fraunhofer Institute for Digital Media Technology, Ilmenau, Germany

dr.ir. S.J.L. van Eijndhoven, lid TU/e
Technische Universiteit Eindhoven, The Netherlands

prof.dr.ir. A.C.P.M. Backx, voorzitter
Technische Universiteit Eindhoven, The Netherlands

This thesis is dedicated to all my well wishers

Contents

Abstract

Advances in Perceptual Stereo Audio Coding Using Linear Prediction Techniques

A wide range of techniques for coding a single-channel speech and audio signal has been developed over the last few decades. In addition to pure redundancy reduction, sophisticated source and receiver models have been considered for reducing the bit-rate. Traditionally, speech and audio coders are based on different principles and thus each of them offers certain advantages. With the advent of high capacity channels, networks, and storage systems, the bit-rate versus quality compromise will no longer be the major issue; instead, attributes like low-delay, scalability, computational complexity, and error concealments in packet-oriented networks are expected to be the major selling factors.

Typical audio coders such as MP3 and AAC are based on subband or transform coding techniques that are not easily reconcilable with a low-delay requirement. The reasons for their inherently longer delay are the relatively long band splitting filters needed to undertake requantization under control of a psychoacoustic model, as well as the buffering required to even out variations in the bit-rate. On the other hand, speech coders typically use linear predictive coding which is compatible with attributes like low-delay, scalability, error concealments, and low computational complexity. Since with predictive coding it is possible to obtain a very low encoding/decoding delay with basically no loss of compression performance, we selected Linear Prediction (LP) as our venturing point.

However, several issues need to be resolved in order to make LP an adequate and attractive tool for audio coding. These stem from the fundamental differences between speech and audio signals. Speech signals are typically band-limited, mono, and stem from a single source. Audio signals are typically multi-channel, broadband, and stem from different instruments (sources). This difference creates some fundamental aspects that need to be addressed; like, choosing an appropriate multi-channel linear prediction system such that the essential single-channel LP properties carry over to this generalized case.

Additionally, LP in speech coding is heavily associated with a source model, which is not adequate for audio in view of the fact that multiple sources appear. Instead, the source model has to be replaced by a receiver model: the psychoacoustic model in standard audio coders. This, together with the higher bandwidth means that an LP system for audio coding tends to become rather complex.

This thesis addresses these issues. A proposal for the 'best' generalization of the single-channel LP system to a stereo and multi-channel linear prediction system, complexity reductions for Laguerre-based linear prediction systems, the quantization scheme for stereo linear prediction parameters, and the concept of perceptually biased linear prediction constitute the most important contributions in this thesis. It thereby gives contributions to the field of low-delay, low-complexity coding of audio by use of linear prediction.

Samenvatting

Verschillende technieken voor het coderen van enkel-kanaals spraak of audio signalen zijn gedurende de laatste decennia ontwikkeld. Naast zuivere redundantiereductie worden bron- en destinatie-modellen ingezet om een lage bit-rate te bereiken. Spraak- en audio-coders zijn ontwikkeld op grond van verschillende principes en daardoor biedt elk zijn specifieke voordelen. Met de opkomst van transmissiekanalen, netwerken en geheugens die over een hoge capaciteit beschikken, zal het noodzakelijke compromis tussen kwaliteit en bit-rate niet langer de overheersende rol spelen in het succes van een coder; in plaats daarvan zullen andere attributen zoals lage vertraging, schaalbaarheid, lage rekenkracht en fouten-correcties in pakket-gebaseerde netwerken een meer bepalende rol gaan spelen.

De gebruikelijke audiocoders (MP3, AAC) zijn gebaseerd op subband- of transform-codeer principes en deze zijn niet eenvoudig te verenigen met de eis van een lage vertraging. De reden hiervan is dat de lengtes van de filters om subband- of transform-domein signalen te genereren relatief lang zijn, alsmede de benodigde tijdelijke opslag om grote variaties in het momentaan benodigde bit budget te reduceren. Daarentegen zijn spraakcoders ontwikkeld vanuit lineaire predictie technieken hetgeen beter aansluit bij de attributen van lage vertraging, schaalbaarheid, fouten-correctie en lage rekenintensiteit. Omdat lineaire predictie (LP) de mogelijkheid biedt van een (zeer) lage codeervertraging met geen of weinig verlies van compressieprestatie, hebben we dit gekozen als ons uitgangspunt.

Desalniettemin moeten er verschillende kwesties beschouwd worden opdat LP een geschikt en aantrekkelijk principe wordt voor audiocodering. Deze kwesties komen voort uit de fundamentele verschillen tussen spraak- en audiosignalen. Spraaksignalen zijn typisch bandbegrensd, enkelkanaals en ontspringen uit een enkele bron. Audiosignalen zijn typisch meerkanaals, breedbandig and voortkomend uit verschillende bronnen (instrumenten). Deze verschillen leiden tot de situatie waar verscheidene fundamentele aspecten van LP opnieuw bekeken dienen te worden, zoals het kiezen van een geschikte uitbreiding naar meerkanaals-signalen. Daarenboven is LP in spraakcodering sterk

geassocieerd met een bronmodel hetgeen, gezien de meerdere bronnen, in audiosignalen niet bruikbaar is. Het bronmodel moet op een of andere manier vervangen worden door een destinatie model; het gebruikelijke psychoacoustische model in audiocoders. Dit, samen met de grotere bandbreedte van audiosignalen, betekent dat LP system voor audiocodering tendeert naar een rekenintensieve procedure.

De hoofdthemas in dit proefschrift gaan in op bovengenoemde zaken en omvatten de definitie van de best denkbare generalisatie van het enkel-kanaals LP principe, complexiteitsreducties voor Laguerre-gebaseerde lineaire predictie systemen, quantisatieprocedures voor meerkanaals lineaire predictie parameters, en het concept van perceptueel-bijgestuurde (biased) lineaire predictie. Daarmee levert dit proefschrift een aantal theoretische en praktische bijdragen tot het gebied van audiocodering op basis van lineaire predictie.

Acknowledgments

I would like to thank my thesis supervisor dr.ir. A.C. den Brinker at Philips Research, for his advice, teaching of basic concepts, technical writing, and scientific ethics. The outcome of this thesis would have been impossible without his guidance, and the free discussions that we had during the course of this research. Not only did he regularly inspire me and give me a different angle for viewing things, but he also gave me the freedom that enabled efficient contribution. Many thanks to ir. J. Geurts and ir. T.P.J. Selten, who showed interest in this research and selected their M.Sc. graduation project in this field. I would like to thank them for numerous collaborations on projects related to this thesis, which provided useful insights.

I am extremely grateful to my thesis promoters, prof.dr. R.J. Sluijter and prof.Dr. A.G. Kohlrausch, for letting me pursue this project for the Ph.D. Degree at the Technische Universiteit Eindhoven (TU/e), and for their valuable suggestions that helped me to improve this thesis considerably. Many thanks to the Chairman of the Signal Processing Systems Group at TU/e, prof.dr.ir. J.W.M. Bergmans, for giving me the opportunity to come over to Eindhoven from Singapore, and also for meeting me in person in Singapore.

I am also grateful to the members of the Doctorate committee, prof.dr. S.H. Jensen, Dr.-Ing. G.D.T. Schuller, dr.ir. S.J.L. van Eijndhoven, and prof.dr.ir. A.C.P.M. Backx, for devoting their valuable time in reading, criticizing, and improving the draft version of this thesis.

This research has been performed at Philips Research Laboratories Eindhoven, The Netherlands. I am therefore grateful to the Board of Directors of the Laboratory. Special thanks goes to the members of the Digital Signal Processing Group, headed by Dr.-Ing. T.P. Eisele. I am especially grateful to the members of the Speech and Audio Signal Processing Cluster, in particular, my office roommates dr. N.H. van Schijndel and V.S. Kot, M.Sc., and my former roommate Dr. F. Riera-Palou, for keeping me motivated.

Thanks to my colleagues from TU/e, in particular, the Ph.D. students, ir. J.J.M. Kierkels, ir. E.A.P. Habets, and C. Rabotti. Thanks to ir. H.J.A. van den Meerendonk for his jokes, and thanks to both S.H. Ypma and Mw. Y.E.M.

Broers for assisting me with the official procedures upon arrival in the Netherlands. Special thanks to my good Indian friend at TU/e, Akash Kumar, for teaching me the basics of Indian vegetarian cooking during the course of this research.

Last but not least, many thanks to my family members, relatives, and good friends living in various parts of the world for their continuous support.

Frequently Used Terms, Abbreviations, and Notations

Terms and Abbreviations

AAC	Advanced Audio Coding
ACs	Arcsine Coefficients
ACF	Autocorrelation Function
ACFW	ACF of the Warped Input Signal
AMR-WB	Adaptive Multi-Rate Wideband
AMR-WB+	Extended Adaptive Multi-Rate Wideband
BCC	Binaural Cue Coding
BMLD	Binaural Masking Level Difference
CD	Compact Disc
CELP	Code Excited Linear Prediction
dB	Decibels
DFT	Discrete Fourier Transform
EVD	Eigenvalue Decomposition
FIR	Finite Impulse Response
HE-AAC-v2	High Efficiency-AAC version 2
Hz	Hertz
IDFT	Inverse Discrete Fourier Transform
IEC	International Electrotechnical Commission
IIR	Infinite Impulse Response
ISC	Intensity Stereo Coding
ISO	International Standardization Organization
kbit/s	kilobits per second
kHz	kilo-Hertz
LARs	Log Area Ratios
LP	Linear Prediction
LPC	Linear Predictive Coding

LPLP	Laguerre-based Pure Linear Prediction
LSFs	Line Spectral Frequencies
LSPLP	Laguerre-based Stereo Pure Linear Prediction
M/S	Mid/Side Coding
MP3	MPEG-1 Layer 3
MPEG	Moving Picture Experts Group
MPP	Minimum-Phase Polynomial
MUSHRA	Multi Stimulus test with Hidden Reference and Anchors
PCA	Principal Component Analysis
PLP	Pure Linear Prediction
PS	Parametric Stereo
PW	Perceptual Weighting
RCs	Reflection Coefficients
RPE	Regular Pulse Excitation
SD	Spectral Distortion
SLP	Stereo Linear Prediction
SNR	Signal to Noise Ratio
SSC	SinuSoidal Coder
SVD	Singular Value Decomposition
ULD	Ultra Low Delay Coder
VQ	Vector Quantization
WLP	Warped Linear Prediction

Notation and Variables

$\boldsymbol{A}_k(z)$	Forward Prediction Polynomial
$\boldsymbol{A}_{K,k}$	Forward Prediction Matrices
$\boldsymbol{B}_k(z)$	Backward Prediction Polynomial
$\boldsymbol{B}_{K,k}$	Backward Prediction Matrices
$\boldsymbol{C}_0$	Zero-Lag Correlation Matrix
$\boldsymbol{C}_k$	Correlation Matrices
$\boldsymbol{\xi}_k$	Forward Normalized Reflection Matrices
$\boldsymbol{\xi}'_k$	Backward Normalized Reflection Matrices
$\boldsymbol{E}_k$	Forward Reflection Matrices
$\boldsymbol{E}'_k$	Backward Reflection Matrices
f	Natural Frequency (Hertz)
F_s	Natural Sampling Frequency
$\boldsymbol{I}$	Identity Matrix
λ	Warping Factor (Laguerre parameter)
$\boldsymbol{M}_k$	Forward Normalizing Matrices
$\boldsymbol{M}'_k$	Backward Normalizing Matrices
$\boldsymbol{R}_k$	Forward Innovation Variance Matrices
$\boldsymbol{R}'_k$	Backward Innovation Variance Matrices
ω	Angular Frequency (radians per second)
$\boldsymbol{y}_k$	Forward Prediction Error Vector
$\boldsymbol{y}'_k$	Backward Prediction Error Vector
$\tilde{\boldsymbol{y}}_k$	Normalized Forward Prediction Error Vector
$\tilde{\boldsymbol{y}}'_k$	Normalized Backward Prediction Error Vector
σ_x^2	Variance of x
$X(z)$	z-transform of $x(n)$
$x(n)$	Discrete-time Signal
$(\cdot)^H$	Matrix Conjugate Transposition
$(\cdot)^T$	Matrix Transposition
$(\cdot)^*$	Complex Conjugate Operation
$(\cdot)*(\cdot)$	Convolution Operation
$\det[\cdot]$	Determinant of a Matrix

Chapter 1

Introduction

1.1 Thesis Motivation

Audio coding is the process of changing the representation of an audio signal for transmission or storage so that it meets the requirements of the transmission or storage media; namely, minimization of transmission resources or cost-efficient storage. Decoding is the process of reconstructing the original audio signal from this coded (compressed) representation so that the quality of the audio signal, with respect to some measure, is not degraded. There are mainly two classes of coding schemes.

Lossless audio coding: In these coding schemes [1],[2],[3],[4],[5],[6], it is possible to perfectly reconstruct the samples of the original signal from the coded representation. Naturally in these coding schemes there is reduction in bit-rate without degrading the quality of the audio signals. A bit-rate reduction is made possible by making use of inherent *redundancy* present in the audio signals. For typical audio signals stored in traditional Compact Disc (CD) format, a reduction by a factor of two is possible.

Perceptual audio coding: In contrast to lossless coding schemes, this coding scheme is incapable of perfect reconstruction of the original signals from the coded representation. The majority of the audio coders belong to this class, where the primary motivation is to achieve higher compression ratios. A perceptual audio coder incorporates a human auditory perception model. Usually, a model for computing the masking threshold [7] is considered. The masking threshold specifies in each time-

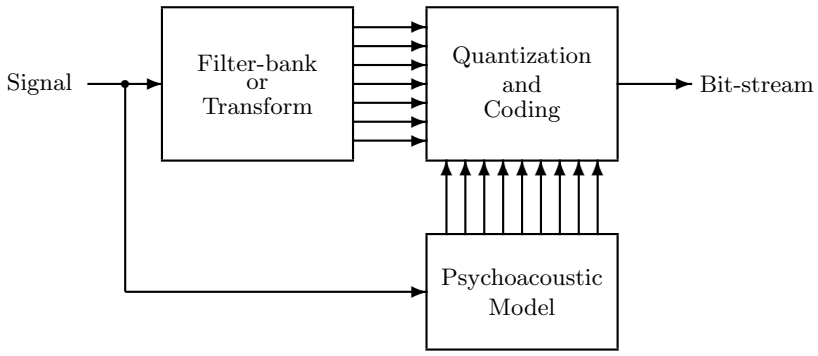


Figure 1.1: *Generic perceptual audio coder (with either a subband or transform encoder) with subband decomposition and a quantization scheme controlled by a psychoacoustic model.*

frequency plane a maximum distortion level such that it is masked by the audio signal to be coded. With a controlled quantization error in the time-frequency plane, the *perceptual irrelevancies* [8] are exploited, which in turn reduces the bit-rate without degrading the perceived quality of the audio signals.

A generic block diagram of a perceptual audio coder is shown in Figure 1.1. The coders typically segment input signals into quasi-stationary frames of 2-50 ms in duration. Then the time-frequency components of each frame are obtained, and from thereon an equivalent time-frequency parameter set is obtained which is more amenable to quantization. A psychoacoustic model computes the masking threshold in the time-frequency plane. The parameters are quantized, then coded, and finally transmitted or stored. The quantization error is controlled such that it is below the computed masking threshold. This coding principle was introduced for speech coding [9] and later applied to audio coding [10]. Today several perceptual audio coding algorithms are in international standards [11],[12],[13],[14] next to various proprietary coding algorithms [15],[16],[17]. For typical CD audio signals, a factor of ten bit-rate reduction is possible without appreciable loss of quality.

Low bit-rate audio coding standards have considerably improved over the last few decades, leading to solutions offering better quality and more flexibility at lower bit-rates than previously available. Research efforts have centered on trying to find compatible ways of adding the enhancements offered by modern developments without penalizing those using

legacy equipment. With the advent of high-capacity channels, networks, and storage systems, the bit-rate versus quality compromise will no longer be the major issue; instead, attributes like low-delay, scalability, computational complexity, and error concealments in packet oriented networks are expected to be the major selling factors.

Traditionally speech and audio coders are based on different principles. Speech coders are based on speech-specific features, whereas audio coders are based on a human hearing model and cannot rely on the characteristics of the input signal. The one-to-one correspondence between the speech production model and the Linear Prediction (LP) analysis-synthesis system has been the major reason for the success of LP in speech coding applications [18]. Moreover, the LP-based speech coding techniques support attributes like low-delay, scalability, error concealments, and they are in general computationally less complex.

In most audio coders it is necessary to use a buffering which delays the processing of the input signal. This yields an algorithmic delay, which is an important attribute in many real-time applications. The main source of this algorithmic delay is related to *bit reservoir* techniques, where more bits are allocated to difficult parts of the input signal, while, for example, pauses in the music can be coded with fewer bits. In applications which use digital transmission of audio signals, like digital microphones in live or studio settings, in-ear monitoring for musicians, wireless digital transmission to loudspeakers, or musician playing together remotely, the tolerable total delay time is less than 10 ms. If the system delay is more than 10 ms, it is difficult for the music ensemble to stay in sync and the tempo of the music can shift. Such a low latency can hardly be attained by means of standard audio coding schemes like MPEG-1 Layer 3 (MP3), MPEG-2 Advanced Audio Coding (AAC), where delays range from 20 ms at 44.1 kHz up to several hundreds of milliseconds [19], which would be too large for the delay-critical applications mentioned above. An advantage of the LP-based audio coders over existing subband and transform coders is that with predictive coding it is possible to obtain a very low encoding/decoding delay [20],[19] with basically no loss of compression performance [21]. Hence it is not necessary to choose between delay and the resulting audio quality. For example, the Ultra Low Delay (ULD) codec [22], which utilizes predictive coding rather than transform coding, is reported to have a delay of about 6-8 ms at a sampling rate ranging from 32-48 kHz, with a bit-rate of 64 kbit/s for each channel [23] at 32 kHz sampling rate.

Naturally, a generic sound coder[1] that combines the strengths of both speech production model and human hearing model into a single coder is desirable. The maturity of the speech coding techniques realized over the last 30 years has fueled the utilization of speech coding knowledge in audio coding. There has been a recent trend of convergence of speech and audio coding applications and many researchers around the world are trying to couple audio with speech technology; examples are, the Adaptive Rate-Distortion Optimized sound codeR (ARDOR) [24] and the AMR-WB+ [25]. LP for audio coding is still not fully explored and, with the exception of the TwinVQ [26], all the LP-based audio coders have remained within the experimental domain.

The aim of this thesis is twofold. Firstly to investigate an important scientific question: whether LP is capable of delivering equally good stereo and/or multi-channel coding schemes such as those currently existing in the market and also understand the problems associated with it. Secondly to investigate the industrial question: whether LP-based coders will be able to generate new applications, such as combined audio and speech coders, offering low-delay, scalability, and preferably a low complexity coding solution. Obviously a reasonable compromise between the quality and bit-rate is also essential. Since it is known from speech coding concepts that LP offers the above-mentioned attributes, we chose LP as our venturing point, especially to arrive at a low-delay coding scheme. As LP is typically associated with the source model of the human speech production system, but not *a priori* with a good model for general audio, we will address this issue first.

1.2 Linear Prediction for Audio Coding

According to information theory [27], a parametric representation for a signal is more efficient than a blind non-parametric representation, provided the parameters are those of an appropriate source model for the signal. Parametric coders are usually based on the source model, and therefore in contrast to perceptual audio coders can achieve higher compression ratios. An example block diagram of such a parametric coder is shown in Figure 1.2. The parameters of the source model are estimated adaptively in time for modeling the input signal. The modeling parameters and often the modeling error are transmitted to the decoder. LP is a

[1] In this thesis, the term "sound coder" always refers to a joint speech and audio coder.

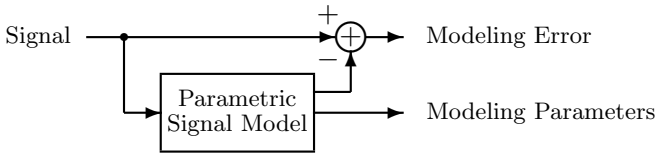


Figure 1.2: *An example of a parametric coder. The parameters of the source model are estimated adaptively in time for modeling the input signal. The modeling parameters and often the modeling error are transmitted to the decoder.*

widely used technique in speech coding and forms one of the classical examples of parametric coders. The modeling parameters are obtained by minimizing the mean square modeling error so that the modeling error is spectrally flat. Thus LP has the spectral whitening property. However, LP-based coders are considered suboptimal for audio signals since these signals do not fit the assumed source model.

As discussed in the previous section, in view of the changing constraints in communication and storage capacity, it is believed that instead of the traditional emphasis on a high compression rate, attributes like low-delay, low complexity and bit-stream scalability will become increasingly important. To handle these issues, LP as used in speech coding is considered to be a more adequate tool than the standard subband or transform coding techniques that are currently in use for audio coding [28],[19].

However, several issues need to be resolved in order to make LP an adequate and attractive tool for audio coding. These stem from the fundamental differences between speech and audio signals. Speech signals are typically band-limited, mono, and stem from a single source. On the other hand, audio signals are typically multi-channel, broadband, and stem from different instruments (sources). These differences are related to fundamental aspects of LP that need to be (re)addressed.

In speech coding, the success of Linear Predictive Coding (LPC) is explained by the fact that an all-pole model is a reasonable approximation of the transfer function of the vocal tract [29]. It is also known that the source model of the LP analysis-synthesis framework does not necessarily model any physical mechanism that generates audio signals. Since LP does not model an ensemble of sources typically encountered in musical signals, intuitively one might think that LP may not be a good tool for audio coding. However, the all-pole model of the LP synthesis filter is

better capable of modeling the spectral peaks than the spectral valleys, and it is known that the spectral peaks play a dominant role in sound perception [30]. Moreover, in speech coding, the LP estimate is also used to derive a perceptual weighting filter [31], which is in fact a perceptual model. The last two observations seem to suggest that there is an intimate relationship between LP and the perceptual model, at least for narrow-band speech. This is only approximately true as the standard method of deriving the weighting filter from LP already fails for broad-band speech and the method to obtain the perceptual weighting filter needs to be modified [32]. Nevertheless, in a rough sense we can conclude that LP is capable of giving a perceptually meaningful parametric representation of the signal.

LP has been proposed for wide-band audio coding [33],[34]. It is also known that by incorporating a frequency warping technique [35] in LP, Warped Linear Prediction (WLP) [36] can be obtained. A conceptually similar method is Pure Linear Prediction (PLP) [37]. In contrast to WLP, PLP has the whitening property just like conventional LP. The modeling capability of both WLP and PLP can be tuned in a psycho-acoustically relevant way [38] making them suitable for speech [36],[39] and audio coding [40],[41].

However, the major challenges of using LP for coding audio signals are the following.

1. In spite of the advantages that PLP offers, its implementation is computationally more intensive than WLP. The major source of complexity resides in the control box that calculates the optimal prediction coefficients.

2. For audio coding applications, one needs to replace the source model used in speech coding by a receiver model (psychoacoustic model) [42], which is commonly used in standard audio coders such that LP is able to model the masking curve. This is feasible by using a WLP or PLP controlled by a (typically complex) psychoacoustic model. This scheme together with the higher bandwidth of audio signals means that an LP system for audio coding tends to be computationally rather intensive [28].

3. Audio is typically (at least) stereo; and there is no existing scheme for the quantization of stereo (or multi-channel) LP parameters.

The audio coding techniques proposed in this thesis focus on a class of LP-based audio coding schemes that provides solutions to the above-

mentioned problems, especially to the class of stereo[2] signals. The techniques discussed in this thesis have been developed for lossy audio coding, where the human ear ultimately judges the degradation of the reconstructed signal.

Before further discussion of the scope of this thesis, Section 1.3 gives an overview of commonly used techniques for stereo audio coding and describes the current state-of-the-art in the field. The contributions of this thesis are briefly described in Section 1.4, which also serves as thesis overview with pointers to the chapters in this thesis.

1.3 Coding of Stereo Audio Signals

Uncompressed stereo audio signals are stored on a CD as two separate channels. The signals are stored as discrete time sampled signals with a sampling frequency of 44.1 kHz. Each sample is represented with a 16-bit precision, which results in a bit-rate of $2 \times 44.1 \times 16 = 1411$ kbit/s. Stereo coding aims at removing redundancy and irrelevancy from the stereo signal to attain lower bit-rates than the sum of the bit-rates of separate channels while maintaining the quality level. In Section 1.1 we mentioned the motivation for selecting LP as a tool for audio coding. In the previous section we discussed the challenges of using LP as a tool for audio coding. In this section, various techniques for the coding of stereo audio signals are reviewed in order to give an insight to the available tools and to identify opportunities to combine these tools with LP-based audio coders.

Mid/Side (M/S) coding [43] is an example of stereo audio coding, which reduces the redundancy between the correlated channel pair by transforming it to a sum/difference channel pair prior to quantization and coding. The masked threshold depends on the inter-aural signal properties of the signal (masker) and the quantization noise (maskee). A perceptual model which describes the inter-aural dependence of the masking threshold is described by a Binaural Masking Level Difference (BMLD) [44].

Compression of stereo audio signals can also be achieved by means of intra- and inter-channel decorrelation methods such as Stereo Linear Prediction (SLP) [45], or by linearly combining the left and the right

[2]In this thesis, the term "stereo audio signal" always refers to two-channel audio signals and the term "multi-channel" refers to more than two channels.

channels into a complex signal and applying WLP to this complex signal [46] as in Complex WLP (CWLP).

Inter-channel redundancy can also be removed with other decorrelation methods such as Karhunen Loève Transform (KLT), which is also known as Principal Component Analysis (PCA). In [47], the time-domain signals are transformed into the frequency domain, and KLT/PCA is applied on the latter domain. In [48], KLT/PCA is applied per frequency band.

Another parametric audio coding technique is based on decomposing an audio signal into three "objects", namely, sinusoids, harmonics, and noise [49],[50] or into sinusoids, transients, and noise [51], and then these "objects"are represented with a suitable set of parameters. This technique was later extended for coding of stereo signals [52].

Apart from perceptual and parametric coding techniques there exist techniques that extend perceptual coders with parametric techniques for improved quality at medium bit-rates. Examples of this class include Intensity Stereo Coding (ISC) [53]. ISC is a joint-channel coding technique that is part of the ISO/IEC MPEG family of standards [11],[12],[13]. It aims at removing cross-channel perceptual irrelevancies. The idea originated from [54] and makes use of the fact that for high frequencies (typically above 2 kHz), the human auditory system is not sensitive to fine-structure phase difference between the two audio channels. Using this technique, a single audio signal is transmitted for the high-frequency range, combined with time- and frequency-dependent scale factors to encode level difference between the channels. The application of ISC is limited, since intolerable distortions can occur if ISC is used for the full bandwidth or for audio signals with highly dynamic and wide spatial image [53].

A parametric stereo coding technique introduced by Faller et al. [55] is called Binaural Cue Coding (BCC) [56],[57],[58],[59]. Some of the limitations of ISC are overcome by BCC [60]. It is based on the assumption that given the sum signal of a number of sources (monophonic signal) and the auditory spatial information contained in a set of parameters (side-information), it is possible to generate a binaural signal by spatially placing the sources contained in the monophonic signal by using the side-information. Mixing the stereo signal down to a mono signal creates the monophonic signal. The parameters containing the auditory spatial information are called sound localization cues and they are extracted from the stereo signal. Thus BCC aims at modeling the most relevant sound source localization cues, while discarding all other spatial

attributes and they can be viewed as an extension to ISC. For the full frequency range, only a mono signal is transmitted, along with spatial parameters. An advantage of creating a mono signal is that a traditional mono audio coder can be used to code this signal. Currently, there exist two important variants of parametric stereo coders, namely BCC and Parametric Stereo (PS) [61]. The most important differences between these two coders can be found in the sound localization cues that are extracted and in the way the mono downmix is created [62]. Both these techniques are lossy techniques, because in the decoder the spatial image is generated using only a few sound localization cues. This means that it is very difficult to attain transparency at high bit-rates. However, because only a few sound localization cues are used, the side information can be transmitted at low to very low bit-rates. This makes these schemes very suitable for low bit-rate coding.

To summarize, the following parametric stereo coding tools already exists.

1. Mid/Side coding;
2. Intensity Stereo Coding;
3. KLT/PCA per frequency band;
4. Parametric stereo coding;
5. Stereo Linear Prediction;
6. Complex Warped Linear Prediction.

The coding tools proposed in 1-4 are associated with processing in the frequency domain, that is, with subband or transform coding. In general, they are not easily reconcilable with a low-delay coder [19]. Furthermore, the coding tools proposed in 2 and 4 offer only a limited scalability, that is, a progression towards a lossless scheme (in the absence of any signal and/or parameter quantization) is not supported. On the other hand, the coding tool proposed in 4 has proved to be an attractive tool because of the creation of a single channel out of two channels, yielding a considerable bit-rate reduction.

The coding tools proposed in 5 and 6 are LP-based schemes. They are scalable and can be modified to operate with a low-delay. With respect to the coding tool proposed in 6, we mention that combining the left and right channels to a complex signal is not equivalent to a general

two-channel structure and therefore is not able to attain maximum redundancy removal. Thus, we limit ourselves in this thesis to the coding tool proposed in 5. There are however several proposals of how to create an SLP. Consequently, this issue is the starting point of this thesis (see Chapter 2).

1.4 Contributions and Overview

In the past, several methodologies have been tried out and optimized for perceptual stereo audio coding research, but none of them is based on a source model and LP. While the current state-of-the-art perceptual stereo audio coders incorporate sophisticated auditory models for computation of the masking thresholds, only a simplistic approach is used to incorporate the effect of spatial hearing. The aim of this thesis is to make contributions in the field of sound coding to arrive at a scheme that can bridge the gap between audio and speech coding concepts and/or techniques to open up new opportunities, specifically for low-delay, scalable, and preferably a low complexity coding solution. We do not address scalability issues nor do we address packet-loss concealment strategies [63]. However, [64] suggests scalability methods that could be incorporated in our proposed scheme. Furthermore, there is abundant literature [18] about error concealments for speech coding which may also be adapted to our coder.

The main contributions of this thesis can be summarized as follows.

1. Review of existing SLP systems with special emphasis on the *symmetric* SLP structure (Chapter 2).

2. Proof of the *stereo all-pass filter* transfer characteristic between the normalized forward and backward prediction error vectors appearing in the symmetric SLP schemes (Chapter 2).

3. Complexity reduced and memory efficient coefficient optimization control box for Laguerre-based PLP (LPLP) systems, and a straightforward way to obtain its reflection coefficients (RCs) associated with the *minimum-phase polynomial* (MPP) (Chapter 3).

4. Quantization strategy for the prediction matrices (Chapters 4 and 5).

5. Complexity reduced method for incorporating a psychoacoustic model in the LP scheme (Chapter 6).

In line with the aims of the thesis, the above list shows that these are the fundamental issues addressed (especially for stereo and multi-channel extensions) along with several complexity reduction proposals for LP-based audio coding system. A more detailed description of these contributions is contained in the following outline of the thesis.

- In **Chapter 2** the proposed SLP scheme for stereo audio signals is described. The stability of the synthesis filter in SLP schemes is investigated. It is experimentally shown that for unequal orders of auto- and cross-predictors, the stability of the synthesis filter cannot be guaranteed. For the symmetric SLP structure with equal orders, the block-Levinson algorithm [65],[66],[67] is applicable, which is the basis for the proof of stability of the symmetric SLP synthesis filter. It is shown that for this particular structure the normalized forward and backward error signal vectors appearing in a normalized two-channel block lattice filter implementation are coupled via a two-channel all-pass filter. This latter finding is used as the basis for an alternative proof of the stability of the synthesis system of the symmetric SLP with equal orders. It also suggests that for the multi-channel prediction, there maybe an analogue of the Line Spectral Frequencies (LSFs) [68].

- In **Chapter 3** we introduce a novel, fast and efficient algorithm for optimizing the LPLP coefficients. In our new algorithm we exploit the symmetries and the redundancies of the Hermitian Toeplitz matrix as well as a newly established relation between the Gram-matrix and the cross-correlation vector. The proposed algorithm calculates all the entries of the Gram-matrix from the data in the cross-correlation vector and the power of the input signal. Compared to the old approach presented in [37], the new method significantly reduces the computational complexity and the memory requirements. The algorithm can be extended to the stereo case, making it possible to incorporate perceptually inspired Laguerre filters in our proposed SLP scheme such that the Laguerre-based Stereo Pure Linear Prediction (LSPLP) system evolves.

  We also present a straightforward way of obtaining the RCs of the minimum-phase polynomial (MPP) associated with the LPLP without actually creating the MPP [41]. This simplification is not only important from a processing point of view, but also serves as a guideline for defining the multi-channel analogue of this mapping.

- In **Chapter 4**, quantization of SLP parameters is discussed. We propose to transmit the (forward) normalized reflection matrices together with the zero-lag correlation matrix. Furthermore, we select a parameterization for these matrices and a quantization strategy per parameter. The parameterization for the normalized reflection matrices is a variant of the singular value decomposition. For the performance evaluation, we use a spectral distortion measure as criterion, which is based on the norm of the transfer matrix of the synthesis filter. The results show that the proposed transformation nicely decouples the effects due to the different quantizers. Simulations also show that the optimal bit allocation for the different parameters as a function of the mean spectral distortion follows very simple rules, implying that a control mechanism of low computational complexity can be designed. Thus, the proposed quantization scheme is a good candidate for SLP systems, where low encoding/decoding delay is desired. This research is extended in **Chapter 5** to evaluate the LSPLP system using stereo audio data to gain insight into the required bit-rates for practical stereo audio coding applications.

- A perceptually biased linear prediction scheme is proposed for audio coding in **Chapter 6**. Using only simple modifications of the coefficients defining the normal equations for a least-squares error, the spectral masking effects are mimicked in the prediction synthesis filter without using an explicit psychoacoustic model. The main advantage of such a scheme is the reduced computational complexity. The proposed approach was implemented in a LPLP scheme and its performance has been evaluated in comparison with an LPLP approach controlled by the ISO MPEG-1 Layer I-II model [12], as well as with one of the latest spectral-integration-based psychoacoustic models [69]. Listening tests clearly demonstrate the viability of the proposed method. Such perceptual biasing rules can also be used in existing speech coders to replace the perceptual weighting filter. The perceptual biasing rules developed for the single-channel case can be simply extended to the stereo or multi-channel case.

  Chapter 7 is an epilogue discussing the future impact of our current research, identifying directions for further scientific (academic) research and avenues for applications.

Chapter 2

Stereo Linear Prediction

2.1 Introduction

Linear Prediction (LP) analysis is by far the most successful technique for removing redundancies from a speech signal and is based on autoregressive (AR) modeling [70]. LP enables us to estimate the coefficients of an AR filter and is closely related to the model of speech production; thus forming a key component of most of the speech coding algorithms [18].

Audio signals typically consist of more than one channel. Redundancies can not only be exploited within each channel of the audio signal, but also across channels by means of intra- and inter-channel decorrelation methods such as Stereo Linear Prediction (SLP) [71],[72],[73].

There is a mixed opinion on whether stereo and multi-channel linear prediction is an effective tool for speech and audio coding. In speech coding, the use of multi-channel linear prediction was reported to produce gains [74], whereas [75] indicates that the use of cross-channel prediction in LP-based stereo coding does not provide much gain with respect to coding the two channels independently. We note, however, that in [75], the cross-channel prediction was applied on the weighted residual signals from a CELP coder, which may not be the optimal way to exploit the inter-channel redundancies. In audio coding, SLP was reported to show very good results, mainly for lossless audio coding [71],[72],[73]. However, also for audio coding, the opinion on the effectiveness of SLP ranges over the full range from positive to negative with intermediate opinion 'maybe only effective for low frequencies' [76]. On the positive side, we note that the optimal SLP parameters are obtained from the statistics of the inter-channel data, an approach that is also exploited in Parametric Stereo [62]

to extract the spatial parameters, yielding very good results.

However relevant it may be, it is not the intention of this chapter to provide an answer to these opinions. In our view, there are fundamental issues that need to be resolved first, before satisfactory answers can be deduced. The fundamental issue that is raised and addressed in this chapter concerns the 'best' generalization of the single-channel LP system to a stereo and multi-channel linear prediction system. Since different generalizations have been proposed in the audio and speech coding community, this issue is apparently not trivial. Our main focus in this chapter will be to define a generalized SLP system that inherits as many as possible attractive properties from the mono LP case. As attractive properties, we regard: the stability of the LP synthesis filter, existence of the concept of RCs (and LSFs), and the existence of fast algorithms (Levinson or Schur) to obtain the optimal prediction coefficients [70].

We start this chapter by considering several intra- and inter-channel prediction schemes that have been proposed for speech and audio coding [74],[75],[71],[72],[73],[77] and consider somewhat older mathematical work [65],[78],[66] within this framework. In the first instance, the differences in the schemes appear to be rather trivial, being a difference in how many taps are used for the prediction of the left channel from left-to-left and right-to-left channel, and the question whether or not to use the current sample in the prediction. Our experiments reveal that arbitrary choices in these issues cannot be made if the stability of the SLP synthesis filter is required. Next, we consider in more detail a particular scheme for which stability is guaranteed. For this scheme we give an alternative proof for the stability along with the filtering interpretation of the mathematics described in [66]. Next, we introduce the SLP coding scheme that is used throughout the rest of this thesis. The stability issue is of prime importance since it is associated with the possibility to devise an efficient quantization strategy for the prediction parameters. This issue is covered in greater detail in the later chapters.

The outline of this chapter is as follows. Section 2.2 includes a review of the existing SLP schemes. In Section 2.3 we present the experimental results, along with the theoretical proof of stability for the symmetric SLP structure for a special case. We also included the means to estimate the optimal prediction coefficients for this special case. In Section 2.4 we present our proposed stereo audio coding scheme which includes the SLP. Then in Section 2.5 we discuss some measures taken to safeguard against some practical problems while calculating the optimal parame-

ters. Finally, we conclude with a discussion in Section 2.6.

2.2 Stereo Linear Prediction

In this section, we discuss the existing SLP schemes [71],[72],[73]. Like in mono LP, SLP consists of an analysis and a synthesis filter. The SLP analysis filter tries to remove the auto- and cross-correlations from the input signals x_1 and x_2. The synthesis SLP filter performs inverse operations of the analysis filter, thus it reconstructs the signals x_1 and x_2 from the signals e_1 and e_2. The general scheme of the analysis filter can be seen in Figure 2.1. The left and the right sound signals, x_1 and x_2 are the inputs to the SLP stage yielding error signals e_1 and e_2.

Prediction of the left channel: For the schemes [71],[72],[73], the prediction $\hat{x}_1$ of x_1 is given by

$$\hat{x}_1(n) = \sum_{k=1}^{K_a} a_k x_1(n-k) + \sum_{k=1}^{K_b} b_k x_2(n-k), \tag{2.1}$$

where a_k and b_k are the auto- and cross-predictor coefficients, respectively, and K_a and K_b are the auto- and cross-predictor orders, respectively. The estimation of the prediction coefficients is done by minimizing

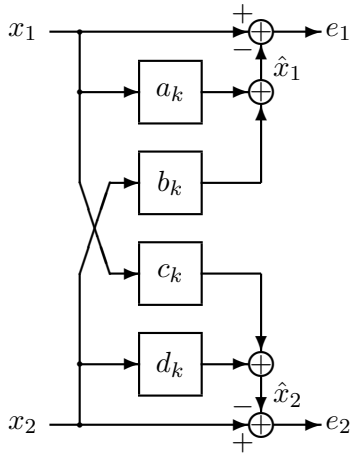


Figure 2.1: *Redundancy removal by a stereo linear prediction analysis filter.*

the mean square of the prediction error e_1. Optimization with respect to a_k and b_k results in the following set of equations

$$r_{11}(k) = \sum_{l=1}^{K_a} a_l r_{11}(k-l) + \sum_{l=1}^{K_b} b_l r_{12}(k-l), \tag{2.2}$$

for $k = 1, 2, \ldots, K_a$, and

$$r_{12}(k) = \sum_{l=1}^{K_a} a_l r_{21}(k-l) + \sum_{l=1}^{K_b} b_l r_{22}(k-l), \tag{2.3}$$

for $k = 1, 2, \ldots, K_b$. These equations involve the correlation function between the channels defined by

$$r_{pq}(k) = \sum_{n} x_p(n) x_q(n-k), \tag{2.4}$$

where p and q can be 1 or 2, and n ranges from $-\infty$ to ∞. The optimal auto- and cross-predictor coefficients a_k and b_k are obtained by solving (2.2) and (2.3).

Prediction of the right channel: For the SLP techniques presented in [71],[72],[73], the current sample of the left channel x_1 is also used to predict the right channel x_2, and the estimate is given by

$$\hat{x}_2(n) = \sum_{k=0}^{K_c} c_k x_1(n-k) + \sum_{k=1}^{K_d} d_k x_2(n-k), \tag{2.5}$$

where d_k and c_k are the auto- and cross-predictor coefficients, respectively, and K_d and K_c are the auto- and cross-predictor orders, respectively. In the SLP schemes described thus far in the literature, the prediction of the current sample of the left channel is based on the past samples of the left and right channels, whereas the prediction of the current sample of the right channel is not only based on the past samples of the left and the right channel, but also on the current sample of the left channel. In that sense, these systems are *asymmetric* since interchanging the left and right channel (that is, an index change) will not lead to only an index change in the predictor.

In contrast to the asymmetric structure, we propose a *symmetric* structure [77] where the prediction $\hat{x}_2$ of x_2 is given by

$$\hat{x}_2(n) = \sum_{k=1}^{K_c} c_k x_1(n-k) + \sum_{k=1}^{K_d} d_k x_2(n-k), \tag{2.6}$$

where $K_b = K_c$ and $K_a = K_d$. For both the symmetric and the asymmetric structures, we can develop similar kind of equations as in (2.2) and (2.3), to calculate the auto- and cross-predictor coefficients d_k and c_k, respectively.

Not only is the symmetric SLP scheme conceptually more appealing, but as will be shown in the next section, it also has advantages with respect to stability. In the special case of the symmetric structure with $K_a = K_b = K_c = K_d = K$, the set of equations for estimating the optimal coefficients a_k, b_k, c_k, and d_k can be merged into a set of equations involving a *block-Toeplitz* structure such that it is also possible to use the *block-Levinson* algorithm [65],[66],[67] for solving these equations. Thus there is only one matrix that needs to be inverted, whereas, in the asymmetric case, two matrices need to be inverted which are symmetric, but not necessarily Toeplitz.

2.3 Stability Analysis

2.3.1 Experimental Observations

In this section, we show experimentally that the stability of the synthesis filter of the SLP is not guaranteed for unequal auto- and cross-predictor orders. If $\boldsymbol{H}(z)$ denotes the transfer matrix of the SLP analysis filter, then the transfer matrix of the synthesis scheme is given by $[\boldsymbol{H}(z)]^{-1}$. Hence, the stability of the synthesis filter is guaranteed if the inverse of the determinant of $\boldsymbol{H}(z)$ is a stable filter. So, all the poles of the synthesis system are defined by the determinant of $\boldsymbol{H}(z)$, and to ensure a stable synthesis filter, all the poles have to lie within the unit circle.

We considered several audio files of CD format (that is, 44.1 kHz, 16 bits/sample stereo) in our experiments. The optimal prediction coefficients were calculated every 23 ms using the autocorrelation method and a Hanning window of length 2048 samples with 50% overlap. The orders K_a and K_d of the auto-predictors were set to 10. Next, we varied the order of the cross-predictors K_b and K_c from 1 to 20 and determined the maximum pole of the synthesis system for each of these settings. The results presented here are from the first 10 seconds of the track *Phenomenon* [79].

Figure 2.2 presents the maximum magnitude of the poles of the synthesis system obtained using different settings of the cross-predictor order. The points underneath the horizontal line at unity indicate the settings

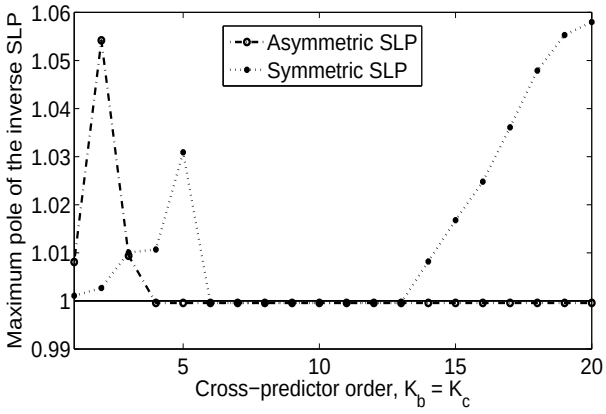


Figure 2.2: *The maximum magnitude of the pole of the synthesis filter as a function of the cross-predictor orders for the symmetric (dotted) and asymmetric (dash-dotted) SLP, for fixed auto-predictors with orders* $K_a = K_d = 10$.

for which the synthesis system is stable. We note that for a fixed auto-predictor order of 10, the synthesis filter of the asymmetric structure is stable for the cross-predictor orders 4-20. The symmetric structure is stable for the cross-predictor orders 6-13. Although the asymmetric structure appears to be much more stable in this scenario, for large prediction orders the variation of the maximum pole of the synthesis system follows similar behavior as the symmetric structure. That is, if we fix the orders of the auto-predictors to a large value say K, and then vary the orders of the cross-predictors from 1 to $2K$, we observe that the synthesis scheme is unstable for low cross-predictor orders ($1 \leq K_b, K_c < K$), stable for mid cross-predictor orders, and unstable for higher cross-predictor orders ($K < K_b, K_c \leq 2K$). This observation indicates that the orders of the auto- and cross-predictors are dependent on each other in order to guarantee the stability of the synthesis system. The experiments using various other tracks show that the synthesis system is always stable when the orders of the auto- and cross-predictors are equal. It is clear that for the case $(K_a = K_d) \neq (K_b = K_c \neq 0)$, the stability is not always guaranteed.

In [72], a method for selecting the optimal prediction orders was presented for the asymmetric SLP structure. There the optimal orders of the

auto-predictors were determined by applying the Levinson-Durbin algorithm [70] independently on the two channels. Then, the cross-predictors were added, and its orders were increased until the bit-rate was minimized. From our experimental observations, it is clear that such an adaptation method might lead to an unstable synthesis system. Backward adaptation [73] is not a solution to the experimentally observed stability issue.

The experimental observation along with the structural nicety of the symmetric structure motivated us to consider the symmetric SLP with equal auto- and cross-predictor orders (a case where the synthesis system is experimentally shown to be stable) in more detail. From now on unless otherwise mentioned, symmetric SLP will mean symmetric SLP with equal auto- and cross-predictor orders.

2.3.2 Symmetric SLP Analysis Filter

Before going to the proof of stability of the symmetric SLP structure, let us consider the symmetric SLP analysis and synthesis filter. We will derive how the optimal prediction coefficients, which are used in both the analysis and synthesis filter, can be calculated. Unless otherwise mentioned, we will consistently use this structure as our SLP coding scheme.

The proposed SLP analysis filter tries to remove the auto- and cross-correlations from the input signals x_1 and x_2 by estimating the current value of the signals as a linear combination of their past values. First, we introduce the following vector notation for an input stereo signal $\boldsymbol{x}$ as

$$\boldsymbol{x}(n) \;=\; \left[\; x_1(n) \quad x_2(n) \;\right]. \tag{2.7}$$

The predictions $\hat{\boldsymbol{x}}$ of the input signals $\boldsymbol{x}$ are now given by

$$\hat{\boldsymbol{x}}(n) \;=\; \sum_{k=1}^{K} \boldsymbol{x}(n-k)\boldsymbol{A}_{K,k}, \tag{2.8}$$

where K is the order of SLP and with $\boldsymbol{A}_{K,k}$ the k^{th} prediction matrix, given by

$$\boldsymbol{A}_{K,k} \;=\; \begin{bmatrix} a_k & c_k \\ b_k & d_k \end{bmatrix}, \tag{2.9}$$

where a_k and d_k are the auto-predictor coefficients, and b_k and c_k the cross-predictor coefficients. The prediction error vector $\boldsymbol{e}$, the outputs of the analysis filter, are defined as the difference between the original signals and the predicted signals

$$\boldsymbol{e}(n) \;=\; \boldsymbol{x}(n) - \hat{\boldsymbol{x}}(n) = \boldsymbol{x}(n) - \sum_{k=1}^{K} \boldsymbol{x}(n-k)\boldsymbol{A}_{K,k}. \tag{2.10}$$

This leads to the transfer matrix $\boldsymbol{H}(z)$ of the SLP analysis filter, given by

$$\boldsymbol{H}(z) \;=\; \begin{bmatrix} 1 - A(z) & -C(z) \\ -B(z) & 1 - D(z) \end{bmatrix}, \tag{2.11}$$

with the transfer functions of the individual predictors, for example $A(z)$, defined by

$$A(z) \;=\; \sum_{k=1}^{K} z^{-k} a_k. \tag{2.12}$$

The transfer functions $B(z)$, $C(z)$, and $D(z)$ are defined analogous to (2.12).

2.3.3 Symmetric SLP Synthesis Filter

The synthesis filter performs the inverse operations of the analysis filter, thus it reconstructs the signals x_1 and x_2 from the signals e_1 and e_2. This means that the synthesis filter uses the same predictors as the analysis filter, except that they are now in a feedback loop. The transfer matrix of the synthesis filter $\boldsymbol{G}(z)$ is therefore given by the inverse of the transfer matrix of the analysis filter

$$\boldsymbol{G}(z) \;=\; [\boldsymbol{H}(z)]^{-1} = \frac{1}{\det[\boldsymbol{H}(z)]} \begin{bmatrix} 1 - D(z) & C(z) \\ B(z) & 1 - A(z) \end{bmatrix}, \tag{2.13}$$

with the determinant of $\boldsymbol{H}(z)$ given by

$$\det[\boldsymbol{H}(z)] \;=\; [1 - A(z)]\,[1 - D(z)] - B(z)C(z). \tag{2.14}$$

Stability of the SLP scheme is determined by the stability of the synthesis filter. As already mentioned in Section 2.3.1, it is clear from (2.13) that all the poles of $\boldsymbol{G}(z)$, which determine the stability, are determined by $\det[\boldsymbol{H}(z)]$. Next it will be shown how to calculate optimal prediction matrices $\boldsymbol{A}_{K,k}$.

2.3.4 Optimal Symmetric SLP Coefficients

The optimal prediction coefficients are calculated by minimizing the mean squared prediction errors $\sigma_{e_1}^2$ and $\sigma_{e_2}^2$ of the analysis filter, given by

$$\begin{aligned}
\sigma_{e_1}^2 &= \sum_n \left[e_1^2(n)\right] = \sum_n \left[x_1(n) - \hat{x}_1(n)\right]^2 \\
&= \sum_n \left[x_1(n) - \sum_{k=1}^{K} x_1(n-k)a_k - \sum_{k=1}^{K} x_2(n-k)b_k\right]^2, \quad (2.15) \\
\sigma_{e_2}^2 &= \sum_n \left[e_2^2(n)\right] = \sum_n \left[x_2(n) - \hat{x}_2(n)\right]^2 \\
&= \sum_n \left[x_2(n) - \sum_{k=1}^{K} x_1(n-k)c_k - \sum_{k=1}^{K} x_2(n-k)d_k\right]^2, \quad (2.16)
\end{aligned}$$

where the summation extends from $-\infty$ to ∞. This also implies that both auto- and cross-correlations in the input signals are being removed for lags $k = \pm 1, \pm 2, \ldots, \pm K$.

We start with minimizing the mean squared error in the left channel. The minimum of (2.15) with respect to the prediction coefficients is obtained by setting

$$\begin{aligned}
\frac{\partial \sigma_{e_1}^2}{\partial a_l} &= 0, \qquad l = 1, 2, \ldots, K, \quad (2.17) \\
\frac{\partial \sigma_{e_1}^2}{\partial b_m} &= 0, \qquad m = 1, 2, \ldots, K. \quad (2.18)
\end{aligned}$$

We first look at (2.17)

$$\begin{aligned}
\frac{\partial \sigma_{e_1}^2}{\partial a_l} &= \frac{\partial \left\{ \sum_n \left[x_1(n) - \sum_{k=1}^{K} x_1(n-k)a_k - \sum_{k=1}^{K} x_2(n-k)b_k\right]^2 \right\}}{\partial a_l} \\
&= \sum_n \left[x_1(n-l)2 \left(x_1(n) - \sum_{k=1}^{K} x_1(n-k)a_k - \sum_{k=1}^{K} x_2(n-k)b_k \right) \right] \\
&= 0, \quad (2.19)
\end{aligned}$$

which leads to

$$\sum_n \left[\sum_{k=1}^{K} x_1(n-l)x_1(n-k)a_k + \sum_{k=1}^{K} x_1(n-l)x_2(n-k)b_k\right] = \sum_n [x_1(n-l)x_1(n)], \tag{2.20}$$

or

$$\sum_{k=1}^{K}\left[\sum_n x_1(n-l)x_1(n-k)a_k\right] + \sum_{k=1}^{K}\left[\sum_n x_1(n-l)x_2(n-k)b_k\right] = \sum_n [x_1(n-l)x_1(n)]. \tag{2.21}$$

If we now use (2.4), then (2.21) can be written as

$$\sum_{k=1}^{K} r_{11}(l-k)a_k + \sum_{k=1}^{K} r_{12}(l-k)b_k = r_{11}(l), \tag{2.22}$$

for $l = 1, 2, \ldots, K$.

Now, we look at (2.18) and this leads in a similar way as described above to the following equation

$$\sum_{k=1}^{K} r_{21}(m-k)a_k + \sum_{k=1}^{K} r_{22}(m-k)b_k = r_{21}(m), \tag{2.23}$$

for $m = 1, 2, \ldots, K$.

We now want to minimize the mean squared error in the right channel. The minimum of (2.16) with respect to the prediction coefficients is obtained by setting

$$\frac{\partial \sigma_{e_2}^2}{\partial c_l} = 0, \qquad l = 1, 2, \ldots, K, \tag{2.24}$$

$$\frac{\partial \sigma_{e_2}^2}{\partial d_m} = 0, \qquad m = 1, 2, \ldots, K. \tag{2.25}$$

This leads, similar to the left channel, to the following equations

$$\sum_{k=1}^{K} r_{11}(l-k)c_k + \sum_{k=1}^{K} r_{12}(l-k)d_k = r_{12}(l), \tag{2.26}$$

for $l = 1, 2, \ldots, K$, and

$$\sum_{k=1}^{K} r_{21}(m-k)c_k + \sum_{k=1}^{K} r_{22}(m-k)d_k = r_{22}(m), \tag{2.27}$$

for $m = 1, 2, \ldots, K$.
Equations (2.22), (2.23), (2.26) and (2.27) form the stereo Yule-Walker equations, which can be written in matrix form as

$$\begin{bmatrix} r_{11}(0) & \cdots & r_{11}(1-K) & r_{12}(0) & \cdots & r_{12}(1-K) \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ r_{11}(K-1) & \cdots & r_{11}(0) & r_{12}(K-1) & \cdots & r_{12}(0) \\ r_{21}(0) & \cdots & r_{21}(1-K) & r_{22}(0) & \cdots & r_{22}(1-K) \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ r_{21}(K-1) & \cdots & r_{21}(0) & r_{22}(K-1) & \cdots & r_{22}(0) \end{bmatrix} \times \begin{bmatrix} a_1 & c_1 \\ \vdots & \vdots \\ a_K & c_K \\ b_1 & d_1 \\ \vdots & \vdots \\ b_K & d_K \end{bmatrix} = \begin{bmatrix} r_{11}(1) & r_{12}(1) \\ \vdots & \vdots \\ r_{11}(K) & r_{12}(K) \\ r_{21}(1) & r_{22}(1) \\ \vdots & \vdots \\ r_{21}(K) & r_{22}(K) \end{bmatrix}. \tag{2.28}$$

If we now define $\boldsymbol{C}_i$ as a 2×2 block matrix with

$$\boldsymbol{C}_i = \begin{bmatrix} r_{11}(i) & r_{12}(i) \\ r_{21}(i) & r_{22}(i) \end{bmatrix}, \tag{2.29}$$

and use $r_{pq}(i) = r_{qp}(-i)$, then we can rearrange the rows and columns of (2.28), which gives

$$\boldsymbol{\Gamma}_{K-1}\boldsymbol{A}_K = \boldsymbol{P}, \tag{2.30}$$

with the correlation matrix $\boldsymbol{\Gamma}_{K-1}$ a block-Toeplitz matrix given by

$$\boldsymbol{\Gamma}_{K-1} = \begin{bmatrix} \boldsymbol{C}_0 & \boldsymbol{C}_{-1} & \boldsymbol{C}_{-2} & \cdots & \boldsymbol{C}_{-K+1} \\ \boldsymbol{C}_1 & \boldsymbol{C}_0 & \boldsymbol{C}_{-1} & \cdots & \boldsymbol{C}_{-K+2} \\ \boldsymbol{C}_2 & \boldsymbol{C}_1 & \boldsymbol{C}_0 & \cdots & \boldsymbol{C}_{-K+3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{C}_{K-1} & \boldsymbol{C}_{K-2} & \boldsymbol{C}_{K-3} & \cdots & \boldsymbol{C}_0 \end{bmatrix}, \tag{2.31}$$

with $\boldsymbol{A}_K$ the prediction matrix given by

$$\boldsymbol{A}_K = \left[\begin{array}{ccccc} \boldsymbol{A}_{K,1} & \boldsymbol{A}_{K,2} & \boldsymbol{A}_{K,3} & \cdots & \boldsymbol{A}_{K,K} \end{array} \right]^T, \tag{2.32}$$

and $\boldsymbol{P}$ the correlation matrix given by

$$\boldsymbol{P} = \left[\begin{array}{ccccc} \boldsymbol{C}_1 & \boldsymbol{C}_2 & \boldsymbol{C}_3 & \cdots & \boldsymbol{C}_K \end{array} \right]^T, \tag{2.33}$$

with $\boldsymbol{C}_{-k} = \boldsymbol{C}_k^T$. The optimal prediction matrix $\boldsymbol{A}_{K,k}$ can now be calculated by solving (2.30) using the block-Levinson algorithm. For a detailed description of the block-Levinson algorithm, the reader is referred to Appendix A.

For a symmetric SLP scheme of order K, the complexity of the block-Levinson algorithm is of the order $O(K^2)$ matrix operations with $O(2^3)$ operations each [67]. Thus, a total complexity of $O(8K^2)$ to obtain the optimal prediction coefficients. For an asymmetric SLP scheme of order K, two matrices [72] of orders $2K \times 2K$ and $(2K+1) \times (2K+1)$ need to be solved using Cholesky decomposition [80], i.e., a total complexity of $O([2K]^3) + O([2K+1]^3)$. If the asymmetric system does not use equal orders for the auto- and cross-predictors, then K has to be interpreted as an average of the orders of the auto- and cross-predictors. Assuming that K is at least 10, the complexity of determining the optimal coefficients is much less with the symmetric SLP scheme.

2.3.5 Proof of Stability of the Proposed SLP Scheme

As noted in the previous section, the symmetric SLP structure leads to a system of equations which can be solved by the block-Levinson algorithm [65],[66],[67]. The stability of such an inverse system has already been proved [65] and we present an alternative proof based on the block-Levinson algorithm. Here we reconsider it to gain a deeper insight, in particular, the insights relate to interpretations of the variables appearing in the block-Levinson algorithm and the existence of an all-pass relationship between the forward and backward error signals. We will give the proof for complex stereo input signals. The proof can be extended to the multi-channel case in a straightforward way.

For the symmetric SLP scheme we obtain a sequence of complex 2×2 matrices $\boldsymbol{C}_k$ given by

$$\boldsymbol{C}_k = \left[\begin{array}{cc} r_{11}(k) & r_{12}(k) \\ r_{21}(k) & r_{22}(k) \end{array} \right], \tag{2.34}$$

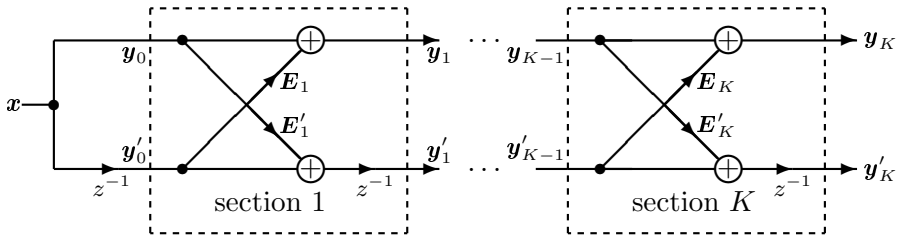


Figure 2.3: *Different sections of the block FIR lattice filter.*

for $k = 0, \pm 1, \cdots, \pm K - 1$. The matrices $\boldsymbol{C}_k$ have the property that $\boldsymbol{C}_{-k} = \boldsymbol{C}_k^H$, and thus all the block-Toeplitz matrices

$$\boldsymbol{\Gamma}_k = \begin{bmatrix} \boldsymbol{C}_0 & \boldsymbol{C}_{-1} & \cdots & \boldsymbol{C}_{-k} \\ \boldsymbol{C}_1 & \boldsymbol{C}_0 & \cdots & \boldsymbol{C}_{1-k} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{C}_k & \boldsymbol{C}_{k-1} & \cdots & \boldsymbol{C}_0 \end{bmatrix}, \tag{2.35}$$

are Hermitian. Here we assume the block-Toeplitz matrix $\boldsymbol{\Gamma}_k$ to be positive definite, that is, not singular.

The block-Levinson algorithm computes the forward and backward predictor coefficients stored in the 2×2 forward and backward reflection matrices $\boldsymbol{E}_k$ and $\boldsymbol{E}'_k$, respectively (see Appendix A for the expression of the reflection matrices). In contrast to the single channel scenario, we note that the information of $\boldsymbol{E}_k$ is not sufficient to construct $\boldsymbol{E}'_k$. However, similar to the classical Levinson algorithm [70], the block-Levinson algorithm also leads to an *FIR lattice filter,* but now a *block FIR lattice filter.* The K different sections of such a structure are shown in Figure 2.3. The vectors $\boldsymbol{y}_k$ and $\boldsymbol{y}'_k$ denote the (k^{th}-order) forward and backward prediction error, respectively. The elements of the input vector $\boldsymbol{x}$ consist of the signals x_1 and x_2 of Figure 2.1.

To prove that the inverse system of the symmetric SLP is stable, we need to prove that the transfer $\boldsymbol{y}_K \rightarrow \boldsymbol{x}$ is stable. Before we proceed, let us mention some useful identities from the block-Levinson algorithm [66]. Firstly, the forward prediction polynomial $\boldsymbol{A}_k(z)$ and the backward pre-

diction polynomial $\boldsymbol{B}_k(z)$ defined as

$$\begin{aligned}\boldsymbol{A}_k(z) &= \boldsymbol{I} - \sum_{i=1}^{k} z^{-i}\boldsymbol{A}_{k,i},\\ \boldsymbol{B}_k(z) &= \boldsymbol{I} - \sum_{i=1}^{k} z^{-i}\boldsymbol{B}_{k,i},\end{aligned}$$

where $\boldsymbol{I}$ denotes the 2×2 identity matrix, are related by the recurrence relations

$$\begin{aligned}\boldsymbol{A}_k(z) &= \boldsymbol{A}_{k-1}(z) + z^{-1}\hat{\boldsymbol{B}}_{k-1}(z)\boldsymbol{E}_k,\\ \hat{\boldsymbol{B}}_k(z) &= z^{-1}\hat{\boldsymbol{B}}_{k-1}(z) + \boldsymbol{A}_{k-1}(z)\boldsymbol{E}'_k,\end{aligned} \tag{2.36}$$

with

$$\hat{\boldsymbol{B}}_k(z) = z^{-k}\boldsymbol{B}_k^H(1/z), \tag{2.37}$$

denoting the reciprocal of the backward prediction polynomial $\boldsymbol{B}_k(z)$. Secondly,

$$\boldsymbol{R}_{k-1}\boldsymbol{E}'_k = \boldsymbol{E}_k^H\boldsymbol{R}'_{k-1}, \tag{2.38}$$

where $\boldsymbol{R}_k$ and $\boldsymbol{R}'_k$ are positive definite 2×2 forward and backward innovation variance matrices, respectively, whose recurrence relations are given by

$$\begin{aligned}\boldsymbol{R}_k &= \boldsymbol{R}_{k-1}(\boldsymbol{I} - \boldsymbol{E}'_k\boldsymbol{E}_k) = \boldsymbol{R}_{k-1} - \boldsymbol{E}_k^H\boldsymbol{R}'_{k-1}\boldsymbol{E}_k,\\ \boldsymbol{R}'_k &= \boldsymbol{R}'_{k-1}(\boldsymbol{I} - \boldsymbol{E}_k\boldsymbol{E}'_k) = \boldsymbol{R}'_{k-1} - \boldsymbol{E}_k'^H\boldsymbol{R}_{k-1}\boldsymbol{E}'_k.\end{aligned} \tag{2.39}$$

It is important to note that $\boldsymbol{R}_0 = \boldsymbol{C}_0$. It can also be shown that

$$\begin{aligned}\boldsymbol{R}_k\boldsymbol{E}'_k &= \boldsymbol{E}_k^H\boldsymbol{R}'_k,\\ \boldsymbol{R}'_k\boldsymbol{E}_k &= \boldsymbol{E}_k'^H\boldsymbol{R}_k.\end{aligned} \tag{2.40}$$

It follows from (2.39) that the eigenvalues λ_1 and λ_2 of the matrices $\boldsymbol{E}'_k\boldsymbol{E}_k$ and $\boldsymbol{E}_k\boldsymbol{E}'_k$ are real and less than unity, and also that both matrices have identical eigenvalues. The property $\lambda_i(\boldsymbol{E}_k\boldsymbol{E}'_k) < 1$ for $1 \leq k \leq K$, together with $\lambda_i(\boldsymbol{R}_0) > 0$, can be used as a criterion for the positive definiteness of the given Hermitian block-Toeplitz matrix $\boldsymbol{\Gamma}_k$.

The forward and backward reflection matrices can be rewritten in the form of *normalized reflection matrices* [66]. Consider factorizing $\boldsymbol{R}_k$ and $\boldsymbol{R}'_k$ in the form

$$\boldsymbol{R}_k = \boldsymbol{M}_k^H \boldsymbol{M}_k, \ \boldsymbol{R}'_k = \boldsymbol{M}_k'^H \boldsymbol{M}'_k, \tag{2.41}$$

for suitable 2×2 *normalizing matrices* $\boldsymbol{M}_k$ and $\boldsymbol{M}'_k$. Generally, $\boldsymbol{M}_k$ and $\boldsymbol{M}'_k$ are calculated within unitary left factors. Then the normalized reflection matrix $\boldsymbol{\xi}_k$ is defined by

$$\boldsymbol{\xi}_k = \boldsymbol{M}'_{k-1} \boldsymbol{E}_k \boldsymbol{M}_{k-1}^{-1} = \left(\boldsymbol{M}_{k-1}'^H\right)^{-1} \boldsymbol{E}_k'^H \boldsymbol{M}_{k-1}^H. \tag{2.42}$$

It is important to note that due to the normalization, the forward and backward normalized reflection matrices $\boldsymbol{\xi}_k$ and $\boldsymbol{\xi}'_k$ are now directly coupled (like the RCs in the single-channel scenario) by the relation $\boldsymbol{\xi}'_k = \boldsymbol{\xi}_k^H$. Using (2.39) in combination with (2.41) and (2.42) gives

$$\left[\left(\boldsymbol{M}_{k-1}^H\right)^{-1} \boldsymbol{M}_k^H\right] \left[\boldsymbol{M}_k \left(\boldsymbol{M}_{k-1}\right)^{-1}\right] = \boldsymbol{I} - \boldsymbol{\xi}_k^H \boldsymbol{\xi}_k. \tag{2.43}$$

From (2.43) one can prove that $\boldsymbol{I} - \boldsymbol{\xi}_k^H \boldsymbol{\xi}_k$ is positive definite or, equivalently, the spectral norm of $\boldsymbol{\xi}_k$ is less than unity *(strictly contractive).*

Now, $\boldsymbol{M}_k$ is defined from $\boldsymbol{M}_{k-1}$ in a way such that $\boldsymbol{M}_k \boldsymbol{M}_{k-1}^{-1}$ is a positive-definite Hermitian matrix satisfying (2.43). Thus we have

$$\begin{aligned} \boldsymbol{M}_k &= \left(\boldsymbol{I} - \boldsymbol{\xi}_k^H \boldsymbol{\xi}_k\right)^{1/2} \boldsymbol{M}_{k-1}, \\ \boldsymbol{M}'_k &= \left(\boldsymbol{I} - \boldsymbol{\xi}_k \boldsymbol{\xi}_k^H\right)^{1/2} \boldsymbol{M}'_{k-1}. \end{aligned} \tag{2.44}$$

The initial values are taken to be $\boldsymbol{M}_0 = \boldsymbol{M}'_0 = \boldsymbol{C}_0^{1/2}$.

Using (2.41)-(2.44), the network of Figure 2.3 is reorganized into the form shown in Figure 2.4. We call this reorganized network as the *normalized block FIR lattice filter.* Figure 2.4 shows the first two sections of such a filter. For better understanding we have zoomed into its $(k+1)^{th}$ section, and it is shown in Figure 2.5. The vectors $\tilde{\boldsymbol{y}}_k$ and $\tilde{\boldsymbol{y}}'_k$ denote the normalized (k^{th}-order) forward and backward prediction errors, respectively. They are normalized in the sense that the covariance matrix is an identity matrix, as indicated in the top and bottom dashed boxes of Figure 2.5.

We will now prove a useful property of such a filtering interpretation. Unlike single-channel LP, where the forward and backward prediction error signals are related via an all-pass transfer, there is no such relation

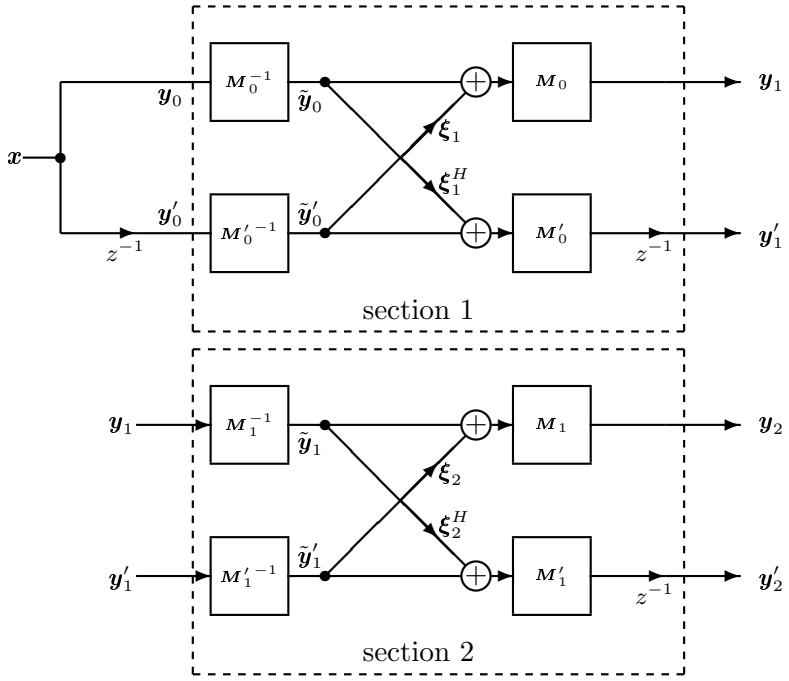


Figure 2.4: *First two sections of the normalized block FIR lattice filter.*

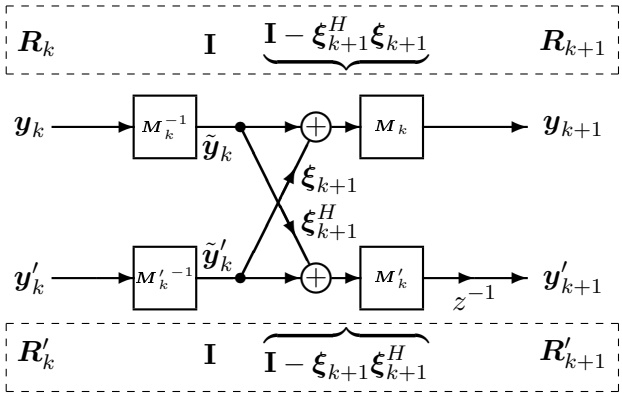


Figure 2.5: $(k+1)^{th}$ *section of the normalized block FIR lattice filter. The matrices in the top and bottom dashed boxes denote the covariance matrices of the consecutive signals in the forward and backward prediction paths, respectively.*

existing between $\boldsymbol{y}_k \rightarrow \boldsymbol{y}'_k$. However, we will show that such a relation does exist between $\tilde{\boldsymbol{y}}_k \rightarrow \tilde{\boldsymbol{y}}'_k$ given by

$$\boldsymbol{P}_{k+1}(z) \;=\; z^{-1}\boldsymbol{M}_k\boldsymbol{A}_k^{-1}\hat{\boldsymbol{B}}_k\left(\boldsymbol{M}'_k\right)^{-1}. \tag{2.45}$$

Note that $\boldsymbol{A}_k$ and $\hat{\boldsymbol{B}}_k$ are a function of z. The variable is dropped for convenience.

For $\boldsymbol{P}_{k+1}(z)$ to be a stereo all-pass filter, we need to prove that

$$\boldsymbol{P}_{k+1}(e^{j\omega})\boldsymbol{P}_{k+1}^H(e^{j\omega}) = \boldsymbol{I},$$

for all ω. Thus, we need to prove that

$$\boldsymbol{I} - \boldsymbol{P}_{k+1}(z)\boldsymbol{P}_{k+1}^H(z) \;=\; \boldsymbol{0} \text{ for } |z| = 1. \tag{2.46}$$

Since $\boldsymbol{A}_k(z)\boldsymbol{M}_k^{-1} \neq \boldsymbol{0}$ at $|z| = 1$, we can prove equivalently that

$$\boldsymbol{A}_k\boldsymbol{M}_k^{-1}\left[\boldsymbol{I} - \boldsymbol{P}_{k+1}\boldsymbol{P}_{k+1}^H\right]\boldsymbol{M}_k^{-H}\boldsymbol{A}_k^H \;=\; \boldsymbol{0}. \tag{2.47}$$

We prove this by induction, first for $k = 0$. Then

$$\begin{aligned}\boldsymbol{P}_1(z) &= z^{-1}\boldsymbol{M}_0\boldsymbol{A}_0^{-1}\hat{\boldsymbol{B}}_0\left(\boldsymbol{M}'_0\right)^{-1}\\ &= z^{-1}\boldsymbol{C}_0^{1/2}\boldsymbol{I}\boldsymbol{I}\boldsymbol{C}_0^{-1/2}\\ &= z^{-1}\boldsymbol{I}.\end{aligned}$$

Thus

$$\begin{aligned}&\boldsymbol{A}_0\boldsymbol{M}_0^{-1}\left[\boldsymbol{I} - \boldsymbol{P}_1\boldsymbol{P}_1^H\right]\boldsymbol{M}_0^{-H}\boldsymbol{A}_0^H\\ &= \boldsymbol{I}\boldsymbol{C}_0^{-1/2}\left[\boldsymbol{I} - |z|^{-2}\boldsymbol{I}\right]\boldsymbol{C}_0^{-1/2}\boldsymbol{I}\\ &= \left[1 - |z|^{-2}\right]\boldsymbol{C}_0^{-1} = \boldsymbol{0},\end{aligned} \tag{2.48}$$

since $1 - |z|^{-2} = 0$ for $|z| = 1$. Thus for $k = 0$, (2.47) is true. Now assume that (2.47) holds for some $k \in \mathbb{N}$. We will now show that it is also true for $k + 1$. Using (2.45) in (2.47), we have for k

$$\boldsymbol{A}_k\boldsymbol{M}_k^{-1}\boldsymbol{M}_k^{-H}\boldsymbol{A}_k^H - \hat{\boldsymbol{B}}_k\left(\boldsymbol{M}'_k\right)^{-1}\left(\boldsymbol{M}'_k\right)^{-H}\hat{\boldsymbol{B}}_k^H \;=\; \boldsymbol{0}. \tag{2.49}$$

We now consider the expression for $k + 1$ given by

$$\boldsymbol{A}_{k+1}\boldsymbol{M}_{k+1}^{-1}\boldsymbol{M}_{k+1}^{-H}\boldsymbol{A}_{k+1}^H - \hat{\boldsymbol{B}}_{k+1}\left(\boldsymbol{M}'_{k+1}\right)^{-1}\left(\boldsymbol{M}'_{k+1}\right)^{-H}\hat{\boldsymbol{B}}_{k+1}^H \tag{2.50}$$

and will prove that it equals a zero matrix.

Using (2.36) and expanding the first term in the left-hand side of (2.50) we get

$$\begin{aligned}
&\boldsymbol{A}_{k+1}\boldsymbol{M}_{k+1}^{-1}\boldsymbol{M}_{k+1}^{-H}\boldsymbol{A}_{k+1}^{H} \\
&= \left[\boldsymbol{A}_k + z^{-1}\hat{\boldsymbol{B}}_k\boldsymbol{E}_{k+1}\right]\boldsymbol{M}_{k+1}^{-1}\boldsymbol{M}_{k+1}^{-H}\left[\boldsymbol{A}_k^H + z^{-H}\boldsymbol{E}_{k+1}^H\hat{\boldsymbol{B}}_k^H\right] \\
&= \boldsymbol{A}_k\boldsymbol{M}_{k+1}^{-1}\boldsymbol{M}_{k+1}^{-H}\boldsymbol{A}_k^H \\
&\quad + z^{-H}\boldsymbol{A}_k\boldsymbol{M}_{k+1}^{-1}\boldsymbol{M}_{k+1}^{-H}\boldsymbol{E}_{k+1}^H\hat{\boldsymbol{B}}_k^H \\
&\quad + z^{-1}\hat{\boldsymbol{B}}_k\boldsymbol{E}_{k+1}\boldsymbol{M}_{k+1}^{-1}\boldsymbol{M}_{k+1}^{-H}\boldsymbol{A}_k^H \\
&\quad + \hat{\boldsymbol{B}}_k\boldsymbol{E}_{k+1}\boldsymbol{M}_{k+1}^{-1}\boldsymbol{M}_{k+1}^{-H}\boldsymbol{E}_{k+1}^H\hat{\boldsymbol{B}}_k^H . \qquad (2.51)
\end{aligned}$$

Similarly, using (2.36) and expanding the second term in the left-hand side of (2.50) we get

$$\begin{aligned}
&\hat{\boldsymbol{B}}_{k+1}\left(\boldsymbol{M}'_{k+1}\right)^{-1}\left(\boldsymbol{M}'_{k+1}\right)^{-H}\hat{\boldsymbol{B}}_{k+1}^H \\
&= \left[z^{-1}\hat{\boldsymbol{B}}_k + \boldsymbol{A}_k\boldsymbol{E}'_{k+1}\right]\left(\boldsymbol{M}'_{k+1}\right)^{-1}\left(\boldsymbol{M}'_{k+1}\right)^{-H}\left[z^{-H}\hat{\boldsymbol{B}}_k^H + \boldsymbol{E}'^H_{k+1}\boldsymbol{A}_k^H\right] \\
&= \hat{\boldsymbol{B}}_k\left(\boldsymbol{M}'_{k+1}\right)^{-1}\left(\boldsymbol{M}'_{k+1}\right)^{-H}\hat{\boldsymbol{B}}_k^H \\
&\quad + z^{-1}\hat{\boldsymbol{B}}_k\left(\boldsymbol{M}'_{k+1}\right)^{-1}\left(\boldsymbol{M}'_{k+1}\right)^{-H}\boldsymbol{E}'^H_{k+1}\boldsymbol{A}_k^H \\
&\quad + z^{-H}\boldsymbol{A}_k\boldsymbol{E}'_{k+1}\left(\boldsymbol{M}'_{k+1}\right)^{-1}\left(\boldsymbol{M}'_{k+1}\right)^{-H}\hat{\boldsymbol{B}}_k^H \\
&\quad + \boldsymbol{A}_k\boldsymbol{E}'_{k+1}\left(\boldsymbol{M}'_{k+1}\right)^{-1}\left(\boldsymbol{M}'_{k+1}\right)^{-H}\boldsymbol{E}'^H_{k+1}\boldsymbol{A}_k^H . \qquad (2.52)
\end{aligned}$$

We will now consider the partial combination of terms (see [81]) from (2.51) and (2.52). The first partial combination is given by

$$\begin{aligned}
&\boldsymbol{A}_k\boldsymbol{M}_{k+1}^{-1}\boldsymbol{M}_{k+1}^{-H}\boldsymbol{A}_k^H - \boldsymbol{A}_k\boldsymbol{E}'_{k+1}\left(\boldsymbol{M}'_{k+1}\right)^{-1}\left(\boldsymbol{M}'_{k+1}\right)^{-H}\boldsymbol{E}'^H_{k+1}\boldsymbol{A}_k^H \\
&= \boldsymbol{A}_k\boldsymbol{M}_k^{-1}\left[\boldsymbol{I} - \boldsymbol{\xi}_{k+1}^H\boldsymbol{\xi}_{k+1}\right]^{-1}\boldsymbol{M}_k^{-H}\boldsymbol{A}_k^H \\
&\quad - \boldsymbol{A}_k\boldsymbol{M}_k^{-1}\boldsymbol{\xi}_{k+1}^H\left[\boldsymbol{I} - \boldsymbol{\xi}_{k+1}\boldsymbol{\xi}_{k+1}^H\right]^{-1}\boldsymbol{\xi}_{k+1}\boldsymbol{M}_k^{-H}\boldsymbol{A}_k^H \\
&= \boldsymbol{A}_k\boldsymbol{M}_k^{-1}\left[\boldsymbol{I} - \boldsymbol{\xi}_{k+1}^H\boldsymbol{\xi}_{k+1}\right]^{-1}\boldsymbol{M}_k^{-H}\boldsymbol{A}_k^H \\
&\quad - \boldsymbol{A}_k\boldsymbol{M}_k^{-1}\left[-\boldsymbol{I} + \left(\boldsymbol{I} - \boldsymbol{\xi}_{k+1}^H\boldsymbol{\xi}_{k+1}\right)^{-1}\right]\boldsymbol{M}_k^{-H}\boldsymbol{A}_k^H \\
&= \boldsymbol{A}_k\boldsymbol{M}_k^{-1}\boldsymbol{M}_k^{-H}\boldsymbol{A}_k^H , \qquad (2.53)
\end{aligned}$$

where we used (2.44) and the identity

$$\boldsymbol{\xi}_{k+1}^H\left[\boldsymbol{I} - \boldsymbol{\xi}_{k+1}\boldsymbol{\xi}_{k+1}^H\right]^{-1}\boldsymbol{\xi}_{k+1} = -\boldsymbol{I} + \left[\boldsymbol{I} - \boldsymbol{\xi}_{k+1}^H\boldsymbol{\xi}_{k+1}\right]^{-1}$$

which can be easily proved using the singular value decomposition of $\boldsymbol{\xi}_k$. Similarly we define the second partial combination

$$\begin{aligned}
&\hat{\boldsymbol{B}}_k \boldsymbol{E}_{k+1} \boldsymbol{M}_{k+1}^{-1} \boldsymbol{M}_{k+1}^{-H} \boldsymbol{E}_{k+1}^H \hat{\boldsymbol{B}}_k^H - \hat{\boldsymbol{B}}_k \left(\boldsymbol{M}'_{k+1}\right)^{-1} \left(\boldsymbol{M}'_{k+1}\right)^{-H} \hat{\boldsymbol{B}}_k^H \\
&= \hat{\boldsymbol{B}}_k \left(\boldsymbol{M}'_k\right)^{-1} \boldsymbol{\xi}_{k+1} \left[\boldsymbol{I} - \boldsymbol{\xi}_{k+1}^H \boldsymbol{\xi}_{k+1}\right]^{-1} \boldsymbol{\xi}_{k+1}^H \left(\boldsymbol{M}'_k\right)^{-H} \hat{\boldsymbol{B}}_k^H \\
&\quad -\hat{\boldsymbol{B}}_k \left(\boldsymbol{M}'_k\right)^{-1} \left[\boldsymbol{I} - \boldsymbol{\xi}_{k+1} \boldsymbol{\xi}_{k+1}^H\right]^{-1} \left(\boldsymbol{M}'_k\right)^{-H} \hat{\boldsymbol{B}}_k^H \\
&= \hat{\boldsymbol{B}}_k \left(\boldsymbol{M}'_k\right)^{-1} \left(-\boldsymbol{I} + \left[\boldsymbol{I} - \boldsymbol{\xi}_{k+1} \boldsymbol{\xi}_{k+1}^H\right]^{-1}\right) \left(\boldsymbol{M}'_k\right)^{-H} \hat{\boldsymbol{B}}_k^H \\
&\quad -\hat{\boldsymbol{B}}_k \left(\boldsymbol{M}'_k\right)^{-1} \left[\boldsymbol{I} - \boldsymbol{\xi}_{k+1} \boldsymbol{\xi}_{k+1}^H\right]^{-1} \left(\boldsymbol{M}'_k\right)^{-H} \hat{\boldsymbol{B}}_k^H \\
&= -\hat{\boldsymbol{B}}_k \left(\boldsymbol{M}'_k\right)^{-1} \left(\boldsymbol{M}'_k\right)^{-H} \hat{\boldsymbol{B}}_k^H, \qquad (2.54)
\end{aligned}$$

where we used (2.44) and the identity

$$\boldsymbol{\xi}_{k+1} \left[\boldsymbol{I} - \boldsymbol{\xi}_{k+1}^H \boldsymbol{\xi}_{k+1}\right]^{-1} \boldsymbol{\xi}_{k+1}^H = -\boldsymbol{I} + \left[\boldsymbol{I} - \boldsymbol{\xi}_{k+1} \boldsymbol{\xi}_{k+1}^H\right]^{-1}$$

which can be easily proved using the singular value decomposition of $\boldsymbol{\xi}_k$. The third partial combination is given by

$$\begin{aligned}
&\boldsymbol{A}_k \boldsymbol{M}_{k+1}^{-1} \boldsymbol{M}_{k+1}^{-H} \boldsymbol{E}_{k+1}^H \hat{\boldsymbol{B}}_k^H - \boldsymbol{A}_k \boldsymbol{E}'_{k+1} \left(\boldsymbol{M}'_{k+1}\right)^{-1} \left(\boldsymbol{M}'_{k+1}\right)^{-H} \hat{\boldsymbol{B}}_k^H \\
&= \boldsymbol{A}_k \left(\boldsymbol{M}_{k+1}^H \boldsymbol{M}_{k+1}\right)^{-1} \boldsymbol{E}_{k+1}^H \hat{\boldsymbol{B}}_k^H - \boldsymbol{A}_k \boldsymbol{E}'_{k+1} \left(\boldsymbol{M}'^H_{k+1} \boldsymbol{M}'_{k+1}\right)^{-1} \hat{\boldsymbol{B}}_k^H \\
&= \boldsymbol{A}_k \boldsymbol{R}_{k+1}^{-1} \boldsymbol{E}_{k+1}^H \hat{\boldsymbol{B}}_k^H - \boldsymbol{A}_k \boldsymbol{E}'_{k+1} \left(\boldsymbol{R}'_{k+1}\right)^{-1} \hat{\boldsymbol{B}}_k^H \\
&= \boldsymbol{0}, \qquad (2.55)
\end{aligned}$$

and, similarly, the fourth partial combination by

$$\begin{aligned}
&\hat{\boldsymbol{B}}_k \boldsymbol{E}_{k+1} \boldsymbol{M}_{k+1}^{-1} \boldsymbol{M}_{k+1}^{-H} \boldsymbol{A}_k^H - \hat{\boldsymbol{B}}_k \left(\boldsymbol{M}'_{k+1}\right)^{-1} \left(\boldsymbol{M}'_{k+1}\right)^{-H} \boldsymbol{E}'^H_{k+1} \boldsymbol{A}_k^H \\
&= \hat{\boldsymbol{B}}_k \boldsymbol{E}_{k+1} \left(\boldsymbol{M}_{k+1}^H \boldsymbol{M}_{k+1}\right)^{-1} \boldsymbol{A}_k^H - \hat{\boldsymbol{B}}_k \left(\boldsymbol{M}'^H_{k+1} \boldsymbol{M}'_{k+1}\right)^{-1} \boldsymbol{E}'^H_{k+1} \boldsymbol{A}_k^H \\
&= \hat{\boldsymbol{B}}_k \boldsymbol{E}_{k+1} \boldsymbol{R}_{k+1}^{-1} \boldsymbol{A}_k^H - \hat{\boldsymbol{B}}_k \left(\boldsymbol{R}'_{k+1}\right)^{-1} \boldsymbol{E}'^H_{k+1} \boldsymbol{A}_k^H \\
&= \boldsymbol{0}. \qquad (2.56)
\end{aligned}$$

The equations (2.55) and (2.56) are proved using

$$\begin{aligned}
&\boldsymbol{M}_{k+1}^{-1} \boldsymbol{M}_{k+1}^{-H} \boldsymbol{E}_{k+1}^H - \boldsymbol{E}'_{k+1} \left(\boldsymbol{M}'_{k+1}\right)^{-1} \left(\boldsymbol{M}'_{k+1}\right)^{-H} \\
&= \boldsymbol{R}_{k+1}^{-1} \boldsymbol{E}_{k+1}^H - \boldsymbol{E}'_{k+1} \left(\boldsymbol{R}'_{k+1}\right)^{-1} \\
&= \boldsymbol{E}_{k+1}^H - \boldsymbol{R}_{k+1} \boldsymbol{E}'_{k+1} \left(\boldsymbol{R}'_{k+1}\right)^{-1} \\
&= \boldsymbol{E}_{k+1}^H \boldsymbol{R}'_{k+1} - \boldsymbol{R}_{k+1} \boldsymbol{E}'_{k+1} \\
&= \boldsymbol{0}, \qquad (2.57)
\end{aligned}$$

where the last equality stems from (2.40).

We can now substitute (2.51) and (2.52) in the expression (2.50) and use the partial combinations (2.53)-(2.56), to yield

$$\boldsymbol{A}_k \boldsymbol{M}_k^{-1} \boldsymbol{M}_k^{-H} \boldsymbol{A}_k^H - \hat{\boldsymbol{B}}_k \left(\boldsymbol{M}'_k\right)^{-1} \left(\boldsymbol{M}'_k\right)^{-H} \hat{\boldsymbol{B}}_k^H \;=\; \boldsymbol{0}, \qquad (2.58)$$

implying $\boldsymbol{P}_k(z)$ as an all-pass filter.

Now, the transfer $\boldsymbol{P}_k(z)$ is stable if $\boldsymbol{I} + z^{-1}\boldsymbol{\xi}_k \boldsymbol{P}_{k-1}(z)$ is *regular* for $|z| > 1$. Since $\boldsymbol{\xi}_k$ is a contracting matrix, and assuming that $\boldsymbol{P}_{k-1}(z)$ is a stable all-pass with the magnitudes of its eigenvalues less than 1 for $|z| > 1$, implies that $\boldsymbol{I} + z^{-1}\boldsymbol{\xi}_k \boldsymbol{P}_{k-1}(z)$ is regular. From this inductive reasoning together with $\boldsymbol{P}_0 = \boldsymbol{I}$, it follows that $\boldsymbol{P}_k(z)$ is a stable all-pass filter.

Now let us consider $\tilde{\boldsymbol{y}}_k$ in the $z-$domain

$$\begin{aligned} \tilde{\boldsymbol{Y}}_k(z) &= \tilde{\boldsymbol{Y}}_{k-1}(z) + z^{-1}\boldsymbol{\xi}_k \tilde{\boldsymbol{Y}}'_{k-1}(z) \\ &= \left[\boldsymbol{I} + z^{-1}\boldsymbol{\xi}_k \boldsymbol{P}_{k-1}(z)\right] \tilde{\boldsymbol{Y}}_{k-1}(z), \end{aligned} \qquad (2.59)$$

where $\tilde{\boldsymbol{Y}}_k(z)$ is the output of the k^{th} section. In view of the fact that $\boldsymbol{\xi}_k$ is a contracting matrix and $\boldsymbol{P}_{k-1}(z)$ is a stable all-pass, the transfer $\boldsymbol{I} + z^{-1}\boldsymbol{\xi}_k \boldsymbol{P}_{k-1}(z)$ is a regular matrix for $|z| > 1$. Consequently, the filtering $\tilde{\boldsymbol{y}}_k \rightarrow \tilde{\boldsymbol{y}}_{k-1}$ is stable. Since this holds for every section k, the total synthesis system $\tilde{\boldsymbol{y}}_K \rightarrow \tilde{\boldsymbol{y}}_0$ is stable. Finally this implies that $\boldsymbol{y}_K \rightarrow \boldsymbol{x}$ is stable. The all-pass character is important since it suggests that for the multi-channel systems an analogue form of LSFs can be defined [81].

2.4 SLP and Rotation

As a drawback of our proposed symmetric SLP scheme, it might be argued that it is inherently incapable of removing the correlations between the current samples of x_1 and x_2. However, this can be easily achieved by cascading a rotator to a symmetric SLP scheme. The rotation block is described in this section. First, it is explained what kind of operation rotation is and next, the function of the rotator is described. It is also described how the optimal rotation angle can be calculated.

The scheme of a rotator can be seen in Figure 2.6(a). The rotator produces a main and side signal m and s, respectively, with the matrix

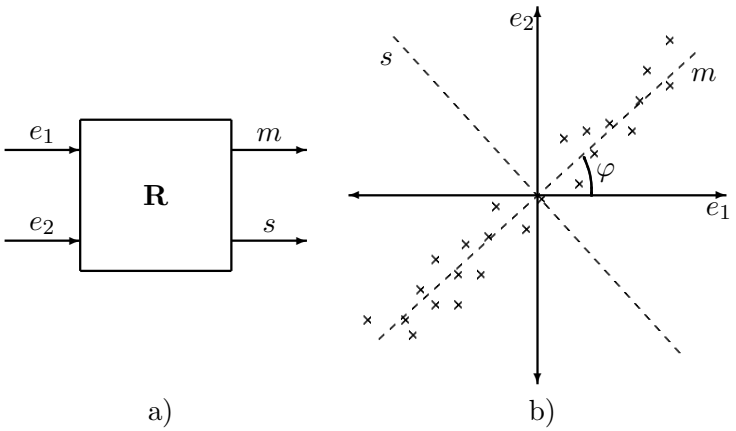


Figure 2.6: *Rotation: a) scheme of a rotator, b) Lissajous plot as an example of rotation. A rotator produces the output signals m and s from the input signals e_1 and e_2 by applying rotation angle φ.*

operation given by

$$\begin{bmatrix} m \\ s \end{bmatrix} = \begin{bmatrix} \cos(\varphi) & \sin(\varphi) \\ -\sin(\varphi) & \cos(\varphi) \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \tag{2.60}$$

where φ is the rotation angle. The rotator uses a variable rotation angle, which is calculated such that the cross-correlation for lag zero is removed. An example of rotation on signals x_1 and x_2 can be seen in Figure 2.6(b), which is a Lissajous plot of the involved signals. The creation of a main and a side signal is similar to the state-of-the-art parametric stereo coding [61], while offering simple means for perfect reconstruction in the absence of quantization. The inverse rotator used in the decoder performs the inverse operations of the rotator in the encoder.

2.4.1 Calculation of Optimal Rotation Angle

The optimal rotation angle is calculated such that the cross-correlation for lag zero is removed, or equivalently, a maximum of the weighted squared sum of the main signal m is produced, which automatically means that a minimum for the weighted squared sum of the side signal s is produced. Calculation of the optimal rotation angle can be done by Principal Component Analysis (PCA) [82].

So, we want to maximize the weighted squared sum J of the main signal, given by

$$J = \sum_n |m(n)|^2 . \tag{2.61}$$

If we now define the input vectors $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$ by

$$\boldsymbol{e}_1 = \left[\begin{array}{cccc} e_1(n) & e_1(n+1) & \cdots & e_1(n+N-1) \end{array} \right]^T , \tag{2.62}$$

$$\boldsymbol{e}_2 = \left[\begin{array}{cccc} e_2(n) & e_2(n+1) & \cdots & e_2(n+N-1) \end{array} \right]^T , \tag{2.63}$$

where N is the length of the signal, and the energies R_{11} and R_{22} and the cross-energy R_{12} of the input signals by

$$R_{11} = \boldsymbol{e}_1^T \boldsymbol{e}_1, \tag{2.64}$$

$$R_{22} = \boldsymbol{e}_2^T \boldsymbol{e}_2, \tag{2.65}$$

$$R_{12} = \boldsymbol{e}_1^T \boldsymbol{e}_2 = \boldsymbol{e}_2^T \boldsymbol{e}_1, \tag{2.66}$$

then J can be written as

$$\begin{aligned} J &= |\boldsymbol{e}_1 \cos(\varphi) + \boldsymbol{e}_2 \sin(\varphi)|^2 \\ &= R_{11} \cos^2(\varphi) + 2R_{12} \cos(\varphi) \sin(\varphi) + R_{22} \sin^2(\varphi) \\ &= \frac{R_{11} + R_{22}}{2} + \frac{R_{11} - R_{22}}{2} \cos(2\varphi) + R_{12} \sin(2\varphi). \end{aligned} \tag{2.67}$$

The maxima and minima of J with respect to the rotation angle are obtained by setting

$$\frac{\partial J}{\partial \varphi} = 0, \tag{2.68}$$

which leads to

$$\frac{\partial J}{\partial \varphi} = -\left(R_{11} - R_{22}\right) \sin(2\varphi) + 2R_{12} \cos(2\varphi) = 0. \tag{2.69}$$

If we now define

$$c = 2R_{12} - j(R_{11} - R_{22}), \tag{2.70}$$

with

$$\phi = \angle c, \tag{2.71}$$

then (2.69) can be rewritten as

$$\frac{\partial J}{\partial \varphi} = |c| \cos(2\varphi - \phi) = 0. \tag{2.72}$$

The solution of (2.72) is given by

$$\varphi = \phi/2 + \pi/4 + k\pi/2, \tag{2.73}$$

with $k \in \mathbb{Z}$. Thus, the values $\hat{\varphi}$ that result in the maxima of J and the values $\check{\varphi}$ that result in the minima of J are given by

$$\hat{\varphi} = \phi/2 + \pi/4 + k\pi, \tag{2.74}$$
$$\check{\varphi} = \phi/2 + 3\pi/4 + k\pi, \tag{2.75}$$

where $\hat{\varphi}$ is the desired rotation angle.

The mean J_{mean}, maximum J_{max} and minimum J_{min} of J are given by

$$J_{mean} = \frac{R_{11} + R_{22}}{2}, \tag{2.76}$$
$$J_{max} = \frac{R_{11} + R_{22}}{2} + \sqrt{\frac{(R_{11} - R_{22})^2}{4} + R_{12}^2}, \tag{2.77}$$
$$J_{min} = \frac{R_{11} + R_{22}}{2} - \sqrt{\frac{(R_{11} - R_{22})^2}{4} + R_{12}^2}, \tag{2.78}$$

and the modulation depth d is defined as the ratio of the mean and the excursion from the mean

$$d = 2\frac{\sqrt{(R_{11} - R_{22})^2/4 + R_{12}^2}}{R_{11} + R_{22}}. \tag{2.79}$$

It will be discussed in Section 2.5.3 that the modulation depth is used to solve practical problems associated while determining the optimal rotation angle. The modulation depth information cannot be obtained by using PCA.

2.5 Practical Problems and Solutions

When calculating the optimal prediction coefficients and optimal rotation angle, which has been described in Sections 2.3.4 and 2.4.1, respectively, some input signals can cause problems. The input signals that cause

problems when calculating the optimal prediction coefficients are signals for which $\mathbf{\Gamma}_{K-1}$ in (2.30) is singular. Examples of such input signals are (nearly) identical left and right signals and one-channel zero signals. One-channel zero signals can be called a specific digital problem, because the parts of an analog signal that have very small amplitude are often quantized such that they result in zero signals. The input signals that can cause problems when calculating the optimal rotation angle are signals with low cross-correlation between the channels and with equal channel powers. Thus, regularization of the calculations in the optimal SLP parameters and the rotation block is needed. The regularization techniques for SLP and rotation are described next.

2.5.1 Regularization of the SLP Optimization

The regularization technique to solve the problems associated with signals for which $\mathbf{\Gamma}_{K-1}$ is singular, is designed for solving the problems associated with (nearly) identical left and right signals and one-channel zero signals, because these signals actually occur. It is expected that the problems associated with the other signals for which $\mathbf{\Gamma}_{K-1}$ is singular, are also solved then. For (nearly) identical left and right signals, auto- and cross-correlations are almost equal. Thus, when the SLP block has these signals as input, it is not defined how to predict the left channel from the right channel and vice versa. In this case, numerical problems can arise when the optimal prediction coefficients are calculated with the block-Levinson algorithm. This is already clear from the fact that the condition number of $\boldsymbol{C}_0$ becomes high, resulting in an inversion of $\boldsymbol{C}_0$ that is difficult to calculate. This inversion is needed in the initialization of the block-Levinson algorithm (see Appendix A). These numerical problems lead to non-uniquely or ill-defined prediction coefficients.

A solution for solving the numerical problems associated with a one-channel zero signals is to add a small amount of noise to the channel with the zero signals. This can be done, because there is a fair chance that the digital one-channel zero signal originally came from an analog signal, which already was a signal with a very small amplitude in one channel. The zero signal in one channel stems from the quantization of the analog signal. Adding noise is then a form of reconstructing the original analog signal. In addition, it is expected that (nearly) identical left and right signals occur more often than one-channel zero signals, so it is more important to have a good method for biasing (nearly) identical left and right signals than to have a good method for biasing one-channel

zero signals. Therefore, a pre-rotator maybe added which uses a rotation angle of $\pi/4$. This results in identical and nearly identical left and right signals rotated to one-channel zero signals and vice versa. Then, noise is added to $\boldsymbol{\Gamma}_{K-1}$ and $\boldsymbol{P}$ in the stereo Yule-Walker equations (2.30), such that they lead towards a biased solution for the optimal prediction coefficients. Thus, the problems associated with (nearly) identical left and right signals are solved by transforming these signals into one-channel zero signals and then applying the regularization technique that is known for these signals (adding noise to $\boldsymbol{\Gamma}_{K-1}$ and $\boldsymbol{P}$). The problems associated with one-channel zero signals are now solved as well, because these signals are transformed into identical left and right signals and adding noise to $\boldsymbol{\Gamma}_{K-1}$ and $\boldsymbol{P}$ also works for identical left and right signals.

Adding noise to $\boldsymbol{\Gamma}_{K-1}$ and $\boldsymbol{P}$ is done by adding noise to each $\boldsymbol{C}_k$ in $\boldsymbol{\Gamma}_{K-1}$ and $\boldsymbol{P}$ according to

$$\begin{aligned} \boldsymbol{C}'_{k,11} &= \boldsymbol{C}_{k,11} + \epsilon_{rel}\boldsymbol{C}_{k,22} \\ \boldsymbol{C}'_{k,22} &= \boldsymbol{C}_{k,22} + \epsilon_{rel}\boldsymbol{C}_{k,11} \end{aligned} , \tag{2.80}$$

where ϵ_{rel} is a factor indicating the amount of noise to be added. Thus, an amount of noise relative to the auto-correlation function of the right channel is added to the auto-correlation function of the left channel and vice versa. In fact, $\boldsymbol{C}'$ may be interpreted as adding some crosstalk to two stochastically independent signals. This improves the condition number of each $\boldsymbol{C}_k$ and therefore decreases the numerical problems in the block-Levinson algorithm. The ϵ_{rel} was set to 10^{-2}.

2.5.2 Robustness of SLP

As mentioned in Section 2.3, all the poles of the SLP synthesis filter are determined by $\det[\boldsymbol{H}(z)]$. This means that the synthesis filter is stable if $\frac{1}{\det[\boldsymbol{H}(z)]}$ is a stable filter. According to Whittle [65], stability of the synthesis filter is guaranteed if the optimal prediction coefficients are calculated with the block-Levinson algorithm. This also means that the auto- and cross-predictors should have the same order, because the block-Levinson algorithm can only be applied in that case.

Although, as observed and proved in Section 2.3, all the poles of the symmetric SLP synthesis filter are within the unit circle, some of them may be very close to unity. To make the prediction coefficients more robust, a technique called spectral smoothing or bandwidth widening, which is a known technique from speech coding [83], can be applied.

Using spectral smoothing in the single-channel case, the calculated prediction coefficients α_k are replaced by α'_k, with

$$\alpha'_k = \gamma^k \alpha_k, \tag{2.81}$$

where γ is the smoothing factor, usually between 0.9 to 1.0. This leads to a new transfer function of the LP analysis filter

$$H'(z) = 1 - \sum_{k=1}^{K} \gamma^k \alpha_k z^{-k} = 1 - \sum_{k=1}^{K} \alpha_k \left(\frac{z}{\gamma}\right)^{-k} = H\left(\frac{z}{\gamma}\right), \tag{2.82}$$

and a new transfer function of the LP synthesis filter

$$G'(z) = \frac{1}{H'(z)} = \frac{1}{1 - \sum_{k=1}^{K} \alpha_k \left(\frac{z}{\gamma}\right)^{-k}}. \tag{2.83}$$

Smoothing shifts the poles of the synthesis filter with a factor γ towards the origin. This improves the stability at the expense of the decorrelation capability of the analysis filter.

If desired, spectral smoothing can also be applied in the stereo case. The transfer matrix of the SLP analysis and synthesis filter is given by (2.11) and (2.13), respectively. If we now apply spectral smoothing to the transfer functions of the individual predictors, the new transfer matrix of the analysis filter is given by

$$\boldsymbol{H}'(z) = \begin{bmatrix} 1 - A\left(\frac{z}{\gamma}\right) & -C\left(\frac{z}{\gamma}\right) \\ -B\left(\frac{z}{\gamma}\right) & 1 - D\left(\frac{z}{\gamma}\right) \end{bmatrix} = \boldsymbol{H}\left(\frac{z}{\gamma}\right), \tag{2.84}$$

and the new transfer matrix of the synthesis filter is given by

$$\boldsymbol{G}'(z) = \frac{1}{\det\left[\boldsymbol{H}\left(\frac{z}{\gamma}\right)\right]} \begin{bmatrix} 1 - D\left(\frac{z}{\gamma}\right) & C\left(\frac{z}{\gamma}\right) \\ B\left(\frac{z}{\gamma}\right) & 1 - A\left(\frac{z}{\gamma}\right) \end{bmatrix} = \boldsymbol{G}\left(\frac{z}{\gamma}\right), \tag{2.85}$$

with

$$\det\left[\boldsymbol{H}\left(\frac{z}{\gamma}\right)\right] = \left[1 - A\left(\frac{z}{\gamma}\right)\right]\left[1 - D\left(\frac{z}{\gamma}\right)\right] - B\left(\frac{z}{\gamma}\right) C\left(\frac{z}{\gamma}\right). \tag{2.86}$$

Thus, applying spectral smoothing to the transfer functions of the individual predictors has an effect similar as that in the single-channel case, since the roots of the determinant are shifted towards the origin.

2.5.3 Regularization of the Rotator

The Lissajous plot of a signal with low cross-correlation between the channels and with equal channel powers can be seen in Figure 2.7. It can be seen that in this plot, there is no clear preferred direction. Thus, when a rotator has this signal as input, problems arise when the optimal rotation angle is calculated, because every rotation angle hardly results in any energy improvement in the main signal. In practice, this means that for slightly different input signals of that kind, the optimal rotation angle can be totally different. In addition, an infinite amount (modulo π) of optimal rotation angles exists always, due to the periodic character of the optimal rotation angle (see (2.74)).

Thus, a strategy has to be devised to choose an optimal rotation angle out of the possible rotation angles. This strategy consists of choosing the optimal rotation angle which is closest to the rotation angle of the previous frame. So, when the optimal rotation angle is calculated, k (in (2.74)) is chosen such that the optimal rotation angle is closest to the rotation angle of the previous frame. If it is difficult to calculate the optimal rotation angle, because every rotation angle hardly results in any energy improvement in the main signal, then the optimal rotation angle is chosen to be equal to the rotation angle of the last frame. Fortunately, this case can be detected with the modulation depth (see (2.79)), because if this is too low, then there is no significant gain to be reached with the rotator for any angle.

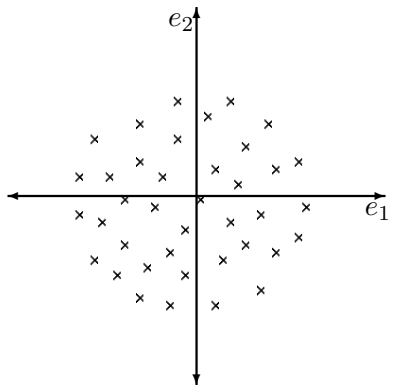


Figure 2.7: *Lissajous plot of a signal with low cross-correlation between the channels and with equal channel powers.*

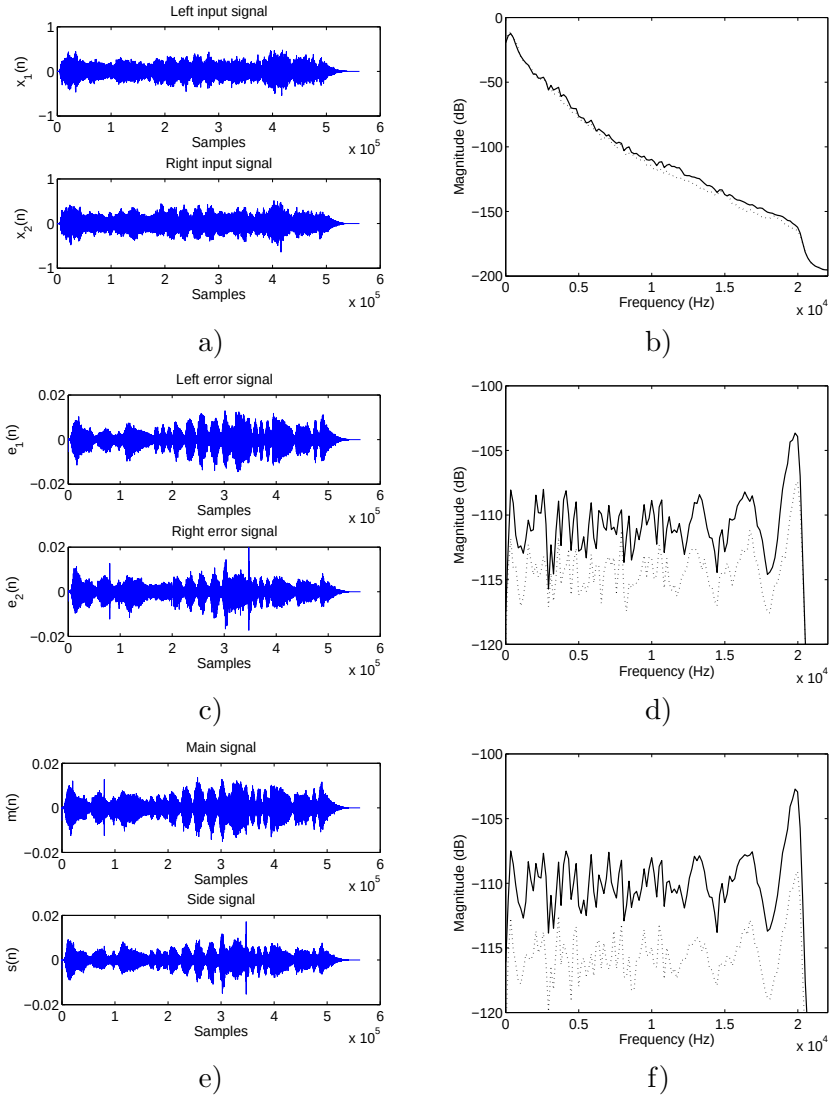


Figure 2.8: *Example of the effect of SLP and rotation: a) Input signals b) Spectrum of the left (solid) and right (dotted) input channel c) Prediction error signals d) Spectrum of the left (solid) and right (dotted) error signals e) Main and side signals f) Spectrum of the main (solid) and the side (dotted) signal.*

Table 2.1: *The MPEG excerpts*

Es01	Suzanne Vega
Es02	Male German Speech
Es03	Female English Speech
Sc01	Haydn Trumpet Concert
Sc02	Orchestra
Sc03	Pop Music
Si01	Harpsichord
Si02	Castanets
Si03	Pitch Pipe
Sm01	Bagpipe
Sm02	Glockenspiel
Sm03	Plucked Strings

A second problem of rotation is that for real audio it was found that the fast varying rotation angle introduces switching artifacts in the output signal. This problem can be solved by filtering the calculated optimal rotation angle with a first-order lowpass filter [84].

Example: As an example, the effect of SLP and rotation is illustrated in Figure 2.8. It depicts the signals from an orchestra piece and the prediction error signals in the time and frequency domain. The effects of rotation are also included in the time and frequency domain. In this example, the prediction order was set to 15, predictor coefficients were calculated every 23 ms using the autocorrelation method and a Hanning window of length 2048 samples with 50% overlap. The SLP analysis filter performs a reduction of the dynamic range in the time domain and spectral flattening (whitening effect) in the frequency domain. The rotators create a spectrally flat main signal of energy higher than the side signal.

The power ratio

$$\begin{aligned} G_l &= \frac{\sigma_{x_1}^2}{\sigma_{e_1}^2} \\ G_r &= \frac{\sigma_{x_2}^2}{\sigma_{e_2}^2}, \end{aligned}$$

of the input signals x_1 and x_2 to its corresponding residual signals e_1 and e_2 given by G_l and G_r, are called the left- and right-channel *predic-*

Table 2.2: *Comparison of mono LP and SLP gains in dB. The numbers inside the brackets indicate the order.*

Track	Channel	Mono (10)	SLP (5)	Mono (20)	SLP (10)	Mono (30)	SLP (15)
Es01	Left	17.5	20.3	18.9	21.7	19.7	22.4
Es01	Right	17.4	24.2	18.7	25.6	19.6	26.3
Es02	Left	24.7	24.3	25.7	25.2	26.6	25.8
Es02	Right	24.7	24.3	25.8	25.3	26.7	25.9
Es03	Left	22.5	24.3	23.3	25.5	23.9	25.9
Es03	Right	22.4	24.5	23.3	25.7	23.9	26.2
Sc01	Left	42.0	41.3	42.5	42.3	42.9	42.7
Sc01	Right	43.4	43.2	43.8	44.1	44.0	44.4
Sc02	Left	33.7	32.7	34.3	33.7	34.6	34.1
Sc02	Right	35.9	35.1	36.5	36.0	36.8	36.4
Sc03	Left	17.5	20.1	18.3	21.7	19.0	22.2
Sc03	Right	16.2	15.6	16.9	16.6	17.6	17.1
Si01	Left	10.6	9.8	11.8	11.0	12.7	11.8
Si01	Right	10.5	9.8	11.5	11.0	12.4	11.8
Si02	Left	6.7	6.6	9.5	8.3	10.2	10.5
Si02	Right	7.2	6.3	9.7	8.2	10.6	10.2
Si03	Left	19.3	19.0	19.9	19.9	20.7	20.6
Si03	Right	16.8	16.7	17.4	17.7	18.2	20.0
Sm01	Left	18.8	15.8	20.1	19.5	20.9	20.8
Sm01	Right	19.6	18.5	20.4	20.3	20.9	21.0
Sm02	Left	26.4	22.3	30.7	29.4	32.0	32.2
Sm02	Right	26.0	21.7	29.9	28.8	31.2	30.8
Sm03	Left	22.3	21.1	23.3	22.7	24.1	23.4
Sm03	Right	16.4	15.1	17.5	16.7	18.4	17.4

tion gains, respectively. Prediction gain achieved by the SLP of order K was compared with the prediction gain achieved by a mono LP of order $2K$ applied independently on the two channels. In this way, the number of optimized parameters for both cases is equal. We used the MPEG excerpts sampled at 44.1 kHz as an input. Table 2.1 explains the acronyms for the excerpts. As an illustration, Table 2.2 gives comparisons of prediction gains (in dB) between SLP, and those using a pair of mono predictors. It appears that for smaller orders the difference between the mono and stereo prediction gains are higher, and the difference gradu-

ally narrows down for higher prediction orders. The general conclusion is that prediction gains are almost equal.

Thus, the advantage of the SLP system is not directly reflected in the prediction gains. The added value of the SLP scheme is studied in more detail in Chapter 3, after we have introduced the Laguerre-based stereo linear prediction scheme. We will show that the coherence is reduced, a measure not reflected in the prediction gains. Since in the SLP scheme a certain amount of stereo information is contained in the prediction matrices, the effects of random errors on the individual channels are less likely to be audible as positional inconsistencies, which is important if one wishes to transmit the main signal and completely (or partly) discard the side signal. We will also show (Chapter 5) that the prediction coefficient bit-rates are in general lower when SLP of order K is used than dual mono LP of order $2K$.

2.6 Conclusions

It was shown experimentally that the stability of the synthesis system in the symmetric and asymmetric SLP is not guaranteed for unequal auto- and cross-predictor orders. For equal orders in a symmetric SLP, a block FIR Lattice filter was introduced to interpret the variables that appear in the block-Levinson algorithm, in particular for the version working with normalized reflection matrices. It was shown that the normalized forward and backward prediction error vectors occurring in the normalized block FIR Lattice filter are coupled via a two-channel all-pass filter. This enables an alternative proof of the stability of the synthesis filter of the symmetric structure. We propose to use this SLP scheme for the coding of stereo audio signals followed by a rotator. We described how the optimal prediction coefficients and the optimal rotation angle can be calculated. We concluded this chapter with regularization of these calculations.

Chapter 3

Low Complexity Laguerre-Based Pure Linear Prediction

3.1 Introduction

It is well known that Linear Prediction (LP) is commonly of interest for applications in speech coding [18]. By incorporating a frequency warping technique in LP, Warped Linear Prediction (WLP) [36] can be obtained, which allows the coder to be tuned to a particular application. WLP whitens the spectrum of the signal in the frequency-warped domain but not in the original frequency domain, and the synthesis filter is not directly realizable using the normal feedback structure [85]. In order to overcome these difficulties associated with WLP, a new technique called Pure Linear Prediction (PLP) [37] was introduced. Two variants of PLP were proposed, that are based on the Laguerre [86] and Kautz [87] filters. We will consider only Laguerre-based PLP (LPLP) throughout the rest of the thesis.

It is known that the modeling capability of the LPLP can be tuned in a psycho-acoustically relevant way [37], making it suitable for lossy speech and audio coding. As an example, LPLP is used in the SinuSoidal Coder (SSC) proposed by Philips [41], which is standardized in MPEG-4 [88]. It is also observed that a moderate reduction in entropy of the residual signal can be obtained if the conventional LP of the recently proposed MPEG-4 Audio Lossless Coding (ALS) [6] is replaced by the LPLP [89].

In spite of the advantages that LPLP offers, its implementation as pro-

posed in [37] is computationally more complex than WLP. There are two sources for this increased computational complexity. One of the sources resides in the control box that estimates the optimal LPLP coefficients for an input signal frame. The second source is that for the interpolation, quantization, and spectral broadening of LPLP coefficients, the LPLP system has to be mapped to a minimum-phase polynomial (MPP) [41] so that standard techniques available in literature for mono LP [68] can be applied.

In this chapter, we introduce a novel, fast, and efficient algorithm that reduces the complexity of the control box of an LPLP system. The algorithm is based on the pertinent relation between the matrix and vector appearing in the set of normal equations defining the optimal prediction coefficients of the LPLP filter. Moreover, we argue that there is a direct way of determining the RCs bypassing the calculation of the MPP. Using these algorithms, the computational complexities of the control boxes in a WLP and LPLP system are essentially equal.

The outline of this chapter is as follows. Section 3.2 includes a short introduction to the PLP scheme, specifically for the Laguerre system, and describes the existing algorithm for obtaining the LPLP coefficients. The fast and efficient algorithm is presented in Section 3.3, where we also discuss the advantages of this algorithm in terms of computational complexity and memory savings. In Section 3.4 we show that the incorporation of the Laguerre filtering in the SLP scheme is advantageous for stereo audio coding. In Section 3.5, we discuss a fast way of obtaining the RCs of the MPP associated with LPLP. Finally, the results of this chapter are summed up in Section 3.6.

3.2 Laguerre-Based Pure Linear Prediction

In PLP, prediction of the current sample is based on the IIR-filtered version of one-sample-delayed input signal. Let us consider a generalized LP scheme as shown in Figure 3.1. We have an input signal x and a set of regression signals y_k for $k = 1, 2, \ldots, K$. We predict $\hat{x}$ of x from the signals y_k by

$$\hat{x} \;=\; \sum_{k=1}^{K} \alpha_k y_k, \tag{3.1}$$

where α_k are the prediction coefficients. These coefficients are usually optimized to minimize the mean square of the prediction error e. The

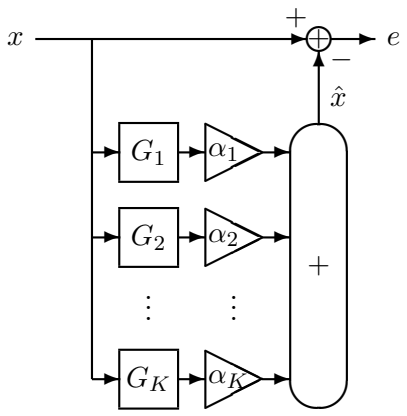


Figure 3.1: *Generalized linear prediction scheme.*

regressor signals can be derived from the input x by linear filtering, thus

$$Y_k(z) \;=\; G_k(z)X(z), \tag{3.2}$$

where $X(z)$ and $Y_k(z)$ are the z-transforms of x and y_k respectively, and $G_k(z)$ is the k^{th} stable transfer function.

For conventional LP we have

$$G_k(z) \;=\; z^{-k}. \tag{3.3}$$

In the PLP scheme [37] a delay has been added explicitly in the analysis and the reconstruction schemes. This restricts the filtering operations $G_k(z)$ to

$$G_k(z) \;=\; z^{-1}H_k(z), \tag{3.4}$$

where $H_k(z)$ are stable and causal IIR filters. To ensure unique optimal prediction coefficients, the filters $H_k(z)$ are typically chosen as linearly independent. The analysis scheme of PLP is shown in Figure 3.2.

PLP has several attractive properties [37]. First, exactly the same predictor can be used in the analysis and synthesis filters, the only difference being that it appears in a feed-forward and feedback structure, respectively. This is necessary in lossless coding applications. Second, it is proved [37] that if $H_k(z)$ is the Laguerre or Kautz filter, and if input data windowing is used for optimization of the prediction coefficients (autocorrelation method), the synthesis filter is stable. Also, if the synthesis filter is stable, the PLP is a spectrum flattening system [37].

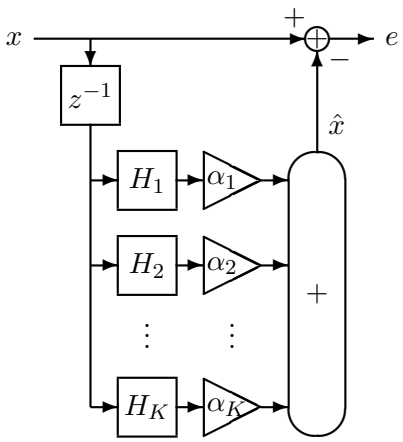


Figure 3.2: *Pure linear prediction scheme.*

Incorporating the Laguerre filters defined as [90]

$$H_k(z) = \frac{\sqrt{1-|\lambda|^2}}{1-z^{-1}\lambda}\left(\frac{z^{-1}-\lambda^*}{1-z^{-1}\lambda}\right)^{k-1}, \tag{3.5}$$

in the PLP scheme, results in a Laguerre-based PLP (LPLP) scheme. The Laguerre parameter (warping factor) $\lambda \in \mathbb{C}$ with $|\lambda| < 1$, and the superscript * denotes complex conjugation. Unequal frequency resolution can be achieved by an appropriate choice of λ, e.g., for 44.1 kHz sampled material, the frequency resolution of LPLP approximates the frequency resolution of the human auditory system (critical band) if $\lambda = 0.756$ [37]. Note that conventional LP is a special case of LPLP where $\lambda = 0$. With the definitions

$$A(z) = \left(\frac{z^{-1}-\lambda^*}{1-z^{-1}\lambda}\right), \tag{3.6}$$

$$C_0(z) = \frac{\sqrt{1-|\lambda|^2}}{1-z^{-1}\lambda}, \tag{3.7}$$

the LPLP can be implemented efficiently as a tapped all-pass line of $A(z)$ preceded by the section $C_0(z)$. This is depicted in Figure 3.3.

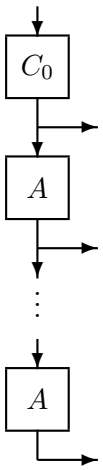


Figure 3.3: *Laguerre system consisting of a pre-filter and a tapped all-pass line.*

3.2.1 Calculation of Optimal LPLP Coefficients

In PLP, a prediction $\hat{x}$ of the value of x is computed using a linear combination of one-sample delayed IIR filtered samples of x

$$\hat{x}(n) = \sum_{k=1}^{K} \alpha_k y_k(n) = \sum_{k=1}^{K} \alpha_k [h_k * x(n-1)], \tag{3.8}$$

where h_k is the impulse response of the k^{th} filter, and $*$ denotes convolution. The optimal prediction coefficients are computed so as to resemble the signal x as much as possible, by minimizing the norm of the residual signal $||e||$, where $e = x - \hat{x}$. We take a deterministic measure J as criterion

$$J = \sum_{n=-\infty}^{\infty} |e(n)|^2. \tag{3.9}$$

It is important to note that the signal x is derived from the original input signal s (either speech or audio) by windowing, i.e., $x(n) = s(n)w(n)$. Typically, a series of overlapping windows are used to find the evolution of prediction coefficients over time. The minimum value of J that can be achieved is called $\hat{J}$, and the value of $\boldsymbol{\alpha}$ for which this minimum is attained is called $\hat{\boldsymbol{\alpha}}$:

$$\hat{J} = \min_{\boldsymbol{\alpha}} J, \tag{3.10}$$

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} J, \tag{3.11}$$

where $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_K]^T$ and $\hat{\boldsymbol{\alpha}} = [\hat{\alpha}_1, \hat{\alpha}_2, \ldots, \hat{\alpha}_K]^T$. Minimizing J with respect to α_k gives

$$\left.\frac{\partial J}{\partial \alpha_k}\right|_{\alpha_k} = 2\sum_n \{x - \sum_{l=1}^{K} \hat{\alpha}_l [h_l * x(n-1)]\}[h_k * x(n-1)] = 0. \tag{3.12}$$

This implies that

$$\sum_{l=1}^{K} \hat{\alpha}_l \sum_n [h_l * x(n-1)][h_k * x(n-1)] = \sum_n [h_k * x(n-1)]x(n). \tag{3.13}$$

If we define

$$Q_{k,l} = \sum_n [h_l * x(n-1)][h_k * x(n-1)] = \sum_n y_l(n) y_k^*(n), \tag{3.14}$$

and

$$P_k = \sum_n x(n)[h_k * x(n-1)] = \sum_n x(n) y_k^*(n), \tag{3.15}$$

Equation (3.13) can be rewritten as a set of K normal equations

$$\sum_{l=1}^{K} \hat{\alpha}_l Q_{k,l} - P_k = 0, \text{ for } k = 1, 2, \ldots, K. \tag{3.16}$$

In matrix notation, the optimal LPLP coefficients are given by the normal equations

$$\boldsymbol{Q}\hat{\boldsymbol{\alpha}} = \boldsymbol{P}, \tag{3.17}$$

where $\hat{\boldsymbol{\alpha}}$ is a vector of optimal prediction coefficients. We note here that the Gram-matrix $\boldsymbol{Q}$ is a Hermitian, positive semi-definite Toeplitz matrix. The vector $\boldsymbol{P}$ is referred to as the cross-correlation vector.

3.2.2 Analysis Algorithm

The optimal prediction coefficients $\hat{\alpha}_k$ and the residual signal e can be computed according to the autocorrelation method described by den Brinker et al. [37]. The flowchart is given here again in Figure 3.4. For convenience we refer to this method as the *Old Method.*

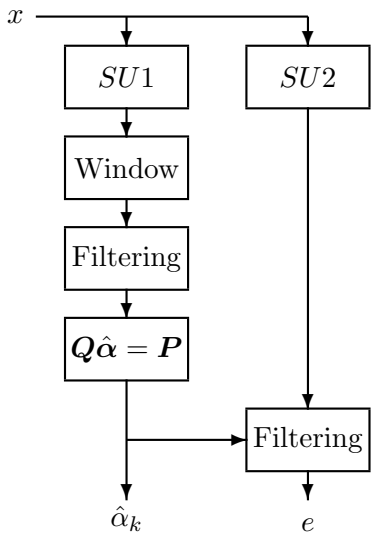


Figure 3.4: *Algorithm for calculating the optimal PLP coefficients and the residual signal. The functions of each of the blocks are described in the text.*

There are two segmentation units *SU1* and *SU2*. The unit *SU1* is used to find the optimal prediction coefficients $\hat{\alpha}_k$. Consecutive segments of *SU1* usually have an overlap of 50%. Then, each segment from *SU1* is windowed (Window box in Figure 3.4) and is applied to the system of filters $G_k(z) = z^{-1}H_k(z)$ (Filtering box in the *SU1* path in Figure 3.4). From the outputs y_k of the filters $H_k(z)$, and together with the input x, the optimal prediction coefficients are obtained for that particular frame in the control box ($\boldsymbol{Q}\hat{\boldsymbol{\alpha}} = \boldsymbol{P}$ box in Figure 3.4).

The unit *SU2* is used to generate the residual signal e from the calculated optimal prediction coefficients. The unit *SU2* produces non-overlapping segments. These signals go through the system of filters $G_k(z) = z^{-1}H_k(z)$. The optimal prediction coefficients obtained from *SU1* is then used to produce the residual signal for this frame. The filtering operation in *SU2* path is given by $F(z) = 1 - \sum_{k=1}^{K} \hat{\alpha}_k G_k(z)$ (Filtering box in the *SU2* path in Figure 3.4). This method guarantees perfect reconstruction in the synthesis scheme.

3.2.3 Problem Statement

For the special case of LPLP where $\lambda = 0$ we have the conventional LP. From (3.14), we obtain the elements of the Gram-matrix for a conventional LP, given by

$$Q_{k,l} = \sum_{n=0}^{N+\min(k,l)-1} x(n-l)x^*(n-k), \tag{3.18}$$

where N is the length of the window that is applied on the input data, and $n = 0$ corresponds to the start position of the window. The elements of the cross-correlation vector are then given by

$$P_l = \sum_{n=0}^{N-1} x(n)x^*(n-l) = Q^*_{1,l+1}, \tag{3.19}$$

i.e., $Q_{1,l} = P^*_{l-1}$, for $l > 1$.

The LPLP system with $\lambda \neq 0$ introduces extra computational complexity when calculating the optimal prediction coefficients when compared to LP. The reason why the LPLP is computationally more demanding is because it exploits IIR filters to produce the regression signals that damp slowly and the Equations (3.14)-(3.16) involve entries of $\boldsymbol{Q}$ and $\boldsymbol{P}$ which are determined by summations over the entire time axis. This can be approximated by observing the signals y_k for a sufficiently long time after the input has become zero. For practical implementation, we may add zero samples to the input signal, so that the responses of the filters are virtually damped out within this additional observation time. The number of zero samples is denoted by Z.

Addition of these extra zero samples introduce extra multiplications. Firstly, the extra multiplications associated with the filtering to determine the tails of y_k. Secondly, the extra multiplications due to the fact that the inner products are defined over longer regression signals. To show the latter, we split the summation defining each element of the Gram-matrix $\boldsymbol{Q}$ in (3.14) into two parts

$$\begin{aligned} Q_{k,l} &= \sum_{n} y_l(n)y_k^*(n) \\ &= \sum_{n=0}^{N-1} y_l(n)y_k^*(n) + \sum_{n=N}^{N+Z-1} y_l(n)y_k^*(n), \end{aligned} \tag{3.20}$$

where N is the length of the window that is applied on the input data, and $n = 0$ corresponds to the start position of the window. The extra term on the right-hand-side of (3.20), contributes to $K^2 \times Z$ extra multiplications to determine the Gram-matrix $\boldsymbol{Q}$ when compared to LP. Since $x(n) = 0$ for $n < 0$ and $n \geq N$, the computational complexity of $\boldsymbol{P}$, given by

$$P_k \;=\; \sum_n x(n)y_k^*(n) = \sum_{n=0}^{N-1} x(n)y_k^*(n),$$

is not affected by this extra addition of zeros.

From the above description follows our problem statement concerning the control box.

1. As shown in (3.18), to determine the elements of the Gram-matrix for LPLP, we don't have an exact truncation as in LP. Additional samples are needed as shown in (3.20) that introduce extra multiplications.

2. As shown in (3.19), for LPLP we do not have a straightforward coupling between the cross-correlation vector and the Gram-matrix as we have for LP. Thus, it is not straightforward to determine the elements of the Gram-matrix just by computing the elements of the cross-correlation vector.

Hence, the main objective of this chapter is to develop a novel algorithm for the calculation of the optimal LPLP coefficients that circumvents these extra multiplications. It is good to mention here that if one could solve the problem 2, the problem 1 is automatically solved in view of the Hermitian Toeplitz structure of the Gram-matrix. In the next section, we will establish a relation between the entries of $\boldsymbol{P}$ and $\boldsymbol{Q}$, and base our new algorithm on this. Later, we discuss the advantages of our proposed algorithm.

3.3 Novel Analysis Method for Obtaining the LPLP Coefficients

In this section we first establish the relation between the elements of $\boldsymbol{P}$ and $\boldsymbol{Q}$. Consequently, we propose a new algorithm which is based on this relation and on the special character of the Gram-matrix. Furthermore, the algorithm is adapted to reduce the memory load. Next, in Section 3.3.2, we consider the performance of the proposed algorithm.

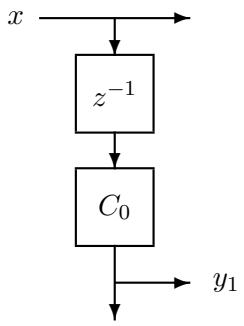


Figure 3.5: *First stage of LPLP*

3.3.1 Proposed Method

Let us now consider the first row of the Gram-matrix $\boldsymbol{Q}$. From (3.14) it is given by

$$Q_{1,l} = \sum_n y_l(n) y_1^*(n), \tag{3.21}$$

where $y_1(n)$ is given by

$$y_1(n) = x(n-1) * c_0(n), \tag{3.22}$$

with $c_0(n)$ being the impulse response of the pre-filter $C_0(z)$ (see Figure 3.5). Substituting (3.22) into (3.21), we get

$$Q_{1,l} = \sum_n y_l(n)[x^*(n-1) * c_0^*(n)]. \tag{3.23}$$

The basis for finding the relation between $\boldsymbol{P}$ and $\boldsymbol{Q}$ is from the following relation between the pre-filter $C_0(z)$ and the all-pass filter $A(z)$

$$z^{-1}C_0(z) = K_1 + K_2 A(z), \tag{3.24}$$

where K_1 and K_2 are the constants to be determined, and $A(z)$ is the all-pass transfer defined by (3.6). Using (3.6) and (3.7) in (3.24), we get

$$z^{-1}\frac{\sqrt{1-|\lambda|^2}}{1-\lambda z^{-1}} = K_1 + K_2\frac{z^{-1}-\lambda^*}{1-\lambda z^{-1}}, \tag{3.25}$$

or

$$z^{-1}\sqrt{1-|\lambda|^2} = K_1(1-\lambda z^{-1}) + K_2(z^{-1}-\lambda^*). \tag{3.26}$$

Solving for the constants, yields

$$K_1 = \frac{\lambda^*}{\sqrt{1-|\lambda|^2}} \tag{3.27}$$

$$K_2 = \frac{1}{\sqrt{1-|\lambda|^2}}. \tag{3.28}$$

Using (3.24), (3.23) can be rewritten in the form

$$\begin{aligned} Q_{1,l} &= \sum_n y_l(n)[x^*(n) * \{K_1\delta(n) + K_2 a(n)\}^*] \\ &= K_1^* \sum_n y_l(n)x^*(n) + K_2^* \sum_n y_l(n)[x^*(n) * a^*(n)] \\ &= K_1^* \sum_n y_l(n)x^*(n) + K_2^* \sum_n y_{l-1}(n)x^*(n), \end{aligned} \tag{3.29}$$

with $\delta(n)$ being the unit impulse, and $a(n)$ being the impulse response of the all-pass filter. Using (3.15), we can reduce (3.29) into

$$Q_{1,l} = K_1^* P_l^* + K_2^* P_{l-1}^*, \text{ for } l > 1. \tag{3.30}$$

From (3.30) we see that with the exception of $Q_{1,1}$, all the elements of the first row of the Gram-matrix $\boldsymbol{Q}$ can be derived from the elements of the vector $\boldsymbol{P}$.

To obtain $Q_{1,1}$ we consider the difference equation for the first-order section (shown in Figure 3.5) given by

$$\sqrt{1-|\lambda|^2}x(n-1) = y_1(n) - \lambda y_1(n-1). \tag{3.31}$$

Multiplying (3.31) by $x^*(n-1)$ and taking summation over all n, we get

$$\begin{aligned} &\sqrt{1-|\lambda|^2}\sum_n x(n-1)x^*(n-1) \\ &= \sum_n y_1(n)x^*(n-1) - \lambda\sum_n y_1(n-1)x^*(n-1) \\ &= \sum_n y_1(n)\left[\frac{y_1^*(n)}{\left(\sqrt{1-|\lambda|^2}\right)^*} - \frac{\lambda^* y_1^*(n-1)}{\left(\sqrt{1-|\lambda|^2}\right)^*}\right] \\ &\quad -\lambda\sum_n y_1(n-1)x^*(n-1). \end{aligned} \tag{3.32}$$

Substituting (3.14) and (3.15) in (3.32), we get

$$\sqrt{1-|\lambda|^2}P_0 = \frac{1}{\sqrt{1-|\lambda|^2}}[Q_{1,1} - \lambda^* \sum_n y_1(n)y_1^*(n-1)] - \lambda P_1^*, \quad (3.33)$$

where P_0 is the power of the windowed input sequence x. Making use of (3.31) in (3.33), we arrive at an expression for $Q_{1,1}$ given by

$$Q_{1,1} \;=\; K_1^* P_1^* + P_0 + K_1 P_1. \quad (3.34)$$

Equations (3.30) and (3.34) express all the elements of the first row of the Gram-matrix $\boldsymbol{Q}$ from the elements of the cross-correlation vector, the signal power P_0, and the two pre-defined constants K_1 and K_2. Since this method computes the elements of the first row of the Gram-matrix $\boldsymbol{Q}$ from the elements of the vector $\boldsymbol{P}$, and since we have *a priori* knowledge that the Gram-matrix is Hermitian Toeplitz, we can derive the whole matrix. In this way we eliminate the extra $K^2 \times Z$ multiplications; where K is the order of the LPLP and Z is the number of extra zero samples as mentioned in Section 3.2.3. Typically the value of K ranges from 10 to 40, and Z is around 500. We name this new method the *Fast Method*.

It is also exciting to observe that the Fast Method avoids the first term on the right-hand-side of (3.20). So, again we save $K^2 \times N$ multiplications; where N is the length of the analysis window (Window box of Figure 3.4), typically around 1000 samples for 44.1 kHz sampled audio. But, in the Fast Method we require N multiplications to compute P_0, and $2K$ multiplications are also consumed to compute the Gram-matrix $\boldsymbol{Q}$. But these additional numbers are extremely small compared to the savings that we get. Hence, overall we save $\approx \{K^2 \times (N+Z)\} - (N+2K)$ multiplications per frame compared to the Old Method, that is, nearly a saving of a factor of K^2.

It is observed for both the Old and Fast Method, we derive all the regression signals y_k at one go (Filtering boxes in Figure 3.4). So, we need $K \times N$ memory locations to store all the regression signals in the case of such an implementation. But, if we calculate the regression signals separately for the different stages of the LPLP; that is, for the first stage we calculate y_1 and derive P_1, and $Q_{1,1}$, and subsequently for each new stage we calculate y_l and derive P_l, and $Q_{1,l}$ for $l = 2, 3, \ldots, K$, we could save memory. More specifically, we save $(K-1) \times N$ memory locations. We name this method as the *Fast & Efficient Method.* For clarity, the proposed algorithm for both the filtering block and the $\boldsymbol{Q}\hat{\boldsymbol{\alpha}} = \boldsymbol{P}$ block of the left-hand side branch of Figure 3.4 is presented below.

compute: P_0;
loop: $l = 1, 2, \ldots, K$
    loop: $n = 0, 1, \ldots, N-1$
        filter: $y_{l-1} \rightarrow y_l$; $y_0 = x$ *(N memories)*
    end
    compute: P_l and $Q_{1,l}$;
end

Typically we know that $N \gg K$. So, if we interchange the inner and the outer loops of the Fast & Efficient Method, we could scale down the memory required to store the regression signals from N to K. The algorithm with the interchanged loops is presented below and the program header can be found in [91].

compute: P_0;
$P_l = 0$; $l = 1, 2, \ldots, K$
loop: $n = 0, 1, \ldots, N-1$
    loop: $l = 1, 2, \ldots, K$
        filter: $y_{l-1}(n) \rightarrow y_l(n)$; $y_0 = x$ *(K memories)*
        $P_l = P_l + x(n)y_l^*(n)$;
    end
end
compute: $Q_{1,l}$; $l = 1, 2, \ldots, K$

For convenience, we name the former Fast & Efficient Method as *Fast & Efficient Method 1,* and the latter as *Fast & Efficient Method 2.* Hence, compared to the Fast and the Old Method, Fast & Efficient Method 1 saves $(K-1) \times N$ memory locations; whereas Fast & Efficient Method 2 saves $(N-1) \times K$ memory locations, that is at the most a factor of N reduction in memory when compared to the Old and the Fast Method.

3.3.2 Results and Discussion

To justify the usage of extra zeros in the Old Method, we used the MPEG excerpts sampled at 44.1 kHz as an input to the LPLP. Table 2.1 explains the acronyms for the excerpts.

We mainly consider three analysis scenarios.

(a) Old Method without additional zeros ($Z = 0$);

(b) Old Method with 500 zeros ($Z = 500$);

(c) Fast Method, and Fast & Efficient Methods 1 and 2.

Experimental Set-Up

The following settings were used in the experiments:

- Segment length in *SU1* is set to 2048 samples;
- A Hanning window is used of length $N = 2048$ samples;
- Segment length in *SU2* (update length) is set to 1024 samples.

The order K of the LPLP was chosen to be 40. The Laguerre parameter was set to $\lambda = 0.756$ to match the psycho-acoustically relevant Bark scale [37].

Observations

Figure 3.6 shows the variation of the log condition number of the Gram-matrix as a function of frame numbers for Sc01. We observe that for Case

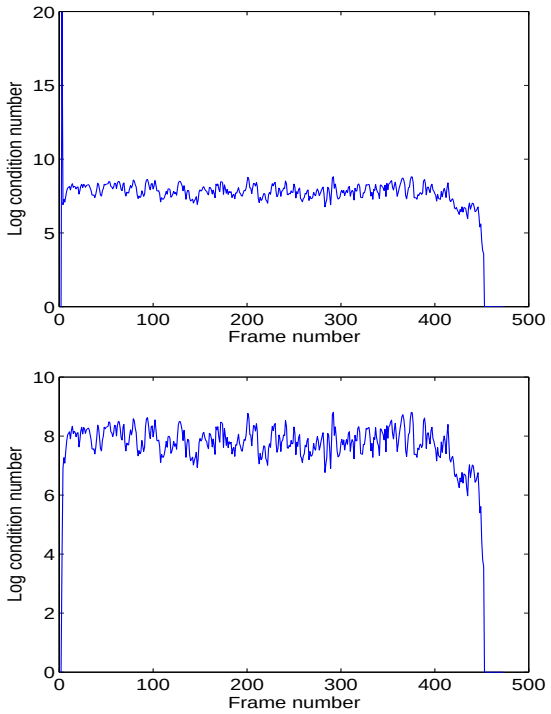


Figure 3.6: *Variation of logarithm (base 10) of the condition number of the Gram-matrix* $\boldsymbol{Q}$ *as a function of frame number for Sc01. Top: Case (a). Bottom: Cases (b) and (c).*

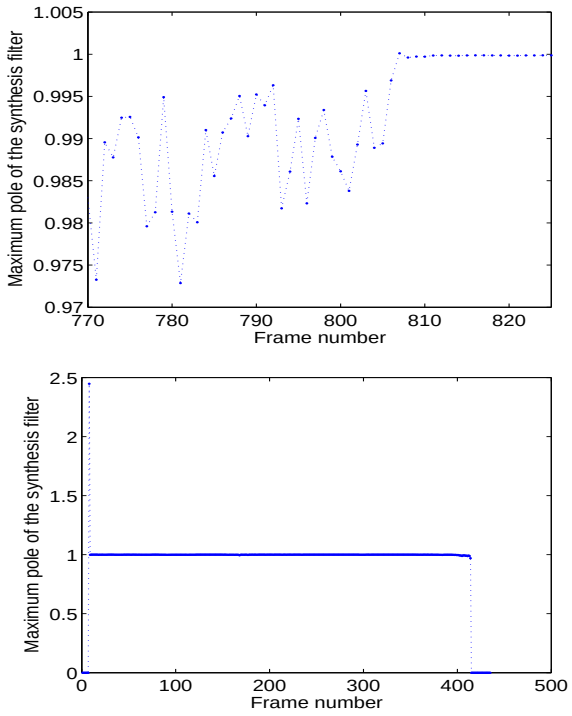


Figure 3.7: *Variation of the maximum pole of the synthesis filter as a function of frame number for Si03 (top) and Sm02 (bottom) for Case (a).*

(a), there is a peak in the log condition number of the Gram-Matrix $\boldsymbol{Q}$ at the first few frames. This behavior is observed for the excerpts Sc01 and Sm02. From this example [1], we can conclude that it is not advisable to remove these extra zeros in order to reduce the computational complexity of the Old Method.

Figure 3.7 shows the variation of the maximum pole of the synthesis filter for the excerpts Si03 and Sm02 for the Case (a). Figure 3.8 shows the variation of the maximum pole of the synthesis filter for the excerpts Si03 and Sm02 for the Cases (b) and (c). It was observed for Case (a) that the synthesis filter is not always stable for the excerpts Sc01, Si01, Si03, and Sm02. In contrast, for Cases (b) and Case (c),

[1] Additionally, we note that though the peaks occurred relatively rarely in the supplied excerpts, this situation changes when considering other input. For instance, taking an SSC [41] residual signal (i.e., obtained after extracting the transients and the sinusoids from the original) as input produces these kinds of peaks in the log condition number more frequently.

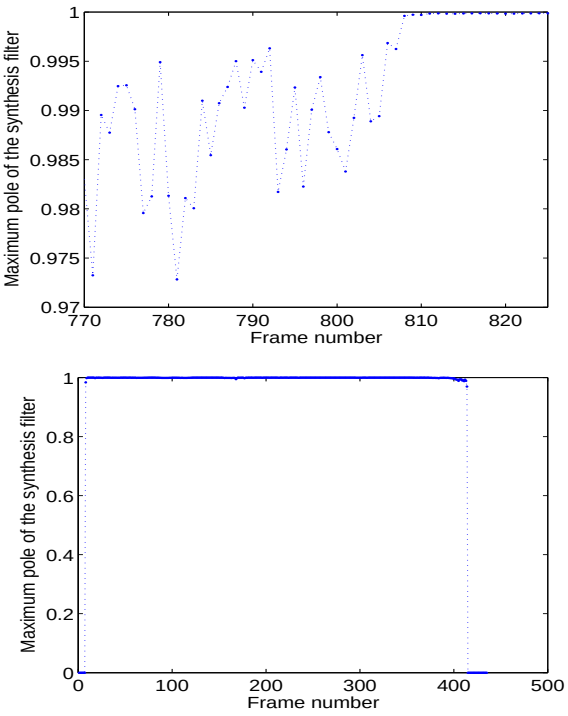


Figure 3.8: *Variation of the maximum pole of the synthesis filter as a function of frame number for Si03 (top) and Sm02 (bottom) for Cases (b) and (c).*

the synthesis filters are always stable. The reason why Case (a) yields unstable synthesis filters is because we truncate the summation in (3.20). Hence it violates the necessary requirement of the input-windowed optimization criterion (autocorrelation method), and therefore the stability of the synthesis filter is not guaranteed [37]. In contrast, Cases (b) and (c) use the autocorrelation method (only avoiding the calculation of the Gram-matrix $\boldsymbol{Q}$ explicitly for Case (c)) hence stability is always guaranteed. When we consider the practical significance of the LPLP scheme, the stability of the synthesis filter is obviously an issue. Therefore Case (a) is not practically feasible and hence ruled out completely.

It was already mentioned previously that Case (c) requires around $\{K^2 \times (N+Z)\} - (N+2K)$ less multiplications when compared to Case (b). So, for the current experimental set-up we save around 4,074,672 multiplications per frame for Case (c). For the Fast & Efficient Method 1 there is a further saving of $(K-1) \times N$ memory locations, implying a sav-

ing of 79,872 memory locations. And, for the Fast & Efficient Method 2, there is a saving of $(N-1) \times K$ memory locations, implying a saving of 81,880 memory locations. If we compare the computational time to generate the residual signal using the MATLAB® simulation tool, we note the following: typically for 10 seconds of audio, Case (b) takes on average 30 seconds, whereas Case (c) with the Fast Method takes 25 seconds, and for the Fast & Efficient methods it takes 16 seconds. So, at best we could get a 45% reduction in computational time using MATLAB. However, for practical applications the algorithms are implemented using the C programming language. Using C, we note the following: Case (b) takes 70 seconds, whereas Case (c) takes 11 seconds, i.e. an 85% reduction in computational time. The reason why Case (b) takes less time in MATLAB than in C is attributed to MATLAB's highly optimized filtering subroutine, in contrast to our non-optimized C programs.

3.4 Laguerre-based Stereo Pure Linear Prediction

The tapped-delay-line of the SLP scheme proposed in Section 2.4 can also be replaced by the Laguerre filters described in Section 3.2, such that the Laguerre-based Stereo Pure Linear Prediction (LSPLP) scheme evolves. The analysis transfer matrix of LSPLP is given by

$$\boldsymbol{F}(z) \;=\; \boldsymbol{I} - z^{-1} \sum_{k=1}^{K} \boldsymbol{A}_{K,k} \frac{\sqrt{1-|\lambda|^2}}{1-\lambda z^{-1}} \left(\frac{-\lambda + z^{-1}}{1-\lambda z^{-1}} \right)^{k-1}, \quad (3.35)$$

where $\boldsymbol{I}$ denotes the 2×2 identity matrix, K the prediction order of the LSPLP system, λ the warping factor (or the Laguerre parameter), and $\boldsymbol{A}_{K,k}$ the k^{th} LSPLP prediction matrix. The novel algorithms described in Section 3.3 can be applied to the LSPLP scheme [84] and thereafter the optimal prediction matrix $\boldsymbol{A}_{K,k}$ can be efficiently obtained by using the block-Levinson algorithm as described in Appendix A. Note that the prediction matrices associated with the LSPLP is different from the prediction matrices associated with the SLP. Since LSPLP can be viewed to consist of four mono LPLP schemes, that is, two mono LPLP as auto-predictors and two mono LPLP as cross-predictors, it is worthwhile at this point to appreciate why the complexity reduced and memory optimized algorithms for LPLP presented in Section 3.3 is so significant for stereo audio coding.

The example in Section 2.5.3 revealed that the performance of the stereo linear prediction in terms of prediction gain is on average equiva-

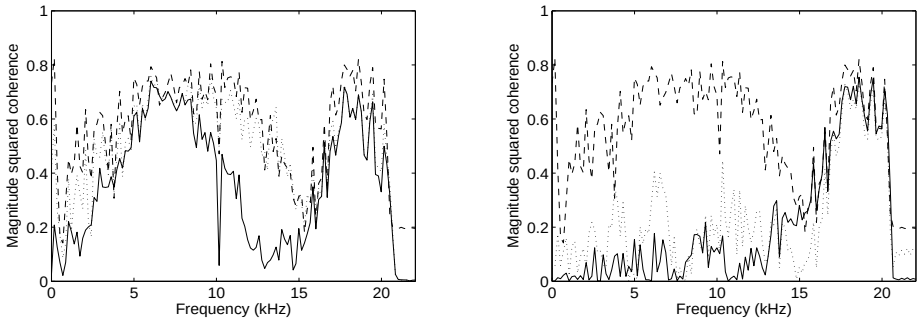
lent to two (mono) linear predictors applied independently on two channels. The SLP system is expected to remove the linear relationship existing between the two input signals and capture this 'coherence' in the prediction parameters. Capturing this coherence information is important [52] to prevent narrowing of the stereo image and spatial instabilities. Therefore, it was proposed to transmit inter-channel coherence as one of the spatial parameters in parametric stereo [62].

An appropriate measure for the degree of linear correlation between two signals x and y as a function of frequency ω is the magnitude squared coherence function [67]. It is defined as

$$\zeta_{xy}^2(\omega) = \frac{|R_{xy}(\omega)|^2}{R_{xx}(\omega)R_{yy}(\omega)},$$

where $R_{xx}(\omega)$ and $R_{yy}(\omega)$ are the power spectral densities of x and y, respectively, $R_{xy}(\omega)$ the cross power spectral density of x and y, and $\zeta_{xy}^2(\omega)$ can be interpreted as the magnitude squared coherence at a specific frequency ω. In practice, $\zeta_{xy}^2(\omega)$ can be estimated using different methods; we will use Welch's averaged, modified periodogram method [70].

We will now compare the LSPLP scheme (applied on stereo input) with the LPLP scheme (applied independently on the two input channels) for the MPEG test excerpts (see Table 2.1). We will test both the schemes with and without the rotator as describe in Section 2.4. The order of



a) b)

Figure 3.9: *Magnitude squared coherence estimate of the stereo input signals (dashed line), error signals (dotted line), and rotated signals (solid line). Plot a) is for LPLP applied independently on the stereo channels and plot b) is for LSPLP.*

LSPLP was set to 15, and the order of LPLP was set to 30. LSPLP predictor matrices and LPLP prediction coefficients were calculated every 6 ms using the autocorrelation method and a Hanning window of length 2048 samples with 50% overlap. Optimal rotation angles for both schemes were calculated with the regularization technique described in Section 2.5.3 [84]. Figure 3.9 illustrates the results for the excerpt Sc03. The figure shows the magnitude squared coherence estimate of the stereo input signals x_1 and x_2. Similarly, the magnitude squared coherence estimate of the error signals obtained from a dual single-channel LPLP system is shown. We note that the coherence is hardly affected. In contrast, when considering the magnitude squared coherence estimate of the error signals obtained from LSPLP system, we observe a large difference with that of the input signal. In particular, there is a large coherence reduction for the low frequencies. The coherence is further reduced after rotation. This effect is encountered in all stereo signals. The reduced coherence is a desired effect if one wishes to transmit the main signal and completely (or partly) discard the side signal.

Thus LSPLP should be preferred over SLP and/or LPLP applied independently on two channels. The reasons to support our statements are the following.

- Similar to LPLP for the single-channel case, LSPLP is able to model the input stereo signals on a warped frequency scale that is closely associated with the perceptually relevant Bark scale.
- LSPLP is able to capture low-frequency coherence more than a dual mono LPLP.

These two advantages make LSPLP a clearly preferred option over both SLP and LPLP. Similar advantages are also expected for Stereo WLP.

3.5 Simplified Mapping of Laguerre Coefficients

In coding applications, the prediction coefficients are usually computed on a frame-by-frame basis and subsequently transmitted and/or stored. In practice, these prediction coefficients are further processed in order to have a better control of the frequency response of the associated filter while performing operations such as quantization, interpolation, and spectral broadening. There is a wealth of information dealing with how these operations should be applied in a conventional LP scheme [68].

In [41] a preprocessing scheme is also presented, which defines a mapping of the LPLP coefficients to a minimum-phase polynomial (MPP). This polynomial can be used as the basis for further mapping to RCs or LSFs to be used for interpolation and/or quantization of LPLP coefficients.

With this mapping, the K^{th} order LPLP transfer function is transformed into a K^{th} order polynomial

$$G(v) = \sum_{k=0}^{K} c_k v^{-k}, \tag{3.36}$$

with the coefficient vector $\boldsymbol{c}$ given by

$$\boldsymbol{c} = \begin{bmatrix} 1 & \lambda & 0 & \cdots & 0 \\ 0 & 1 & \lambda & & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ \vdots & & & 1 & \lambda \\ 0 & \cdots & \cdots & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ -\hat{\alpha}_1/\sqrt{1-|\lambda|^2} \\ -\hat{\alpha}_2/\sqrt{1-|\lambda|^2} \\ \vdots \\ -\hat{\alpha}_K/\sqrt{1-|\lambda|^2} \end{bmatrix}. \tag{3.37}$$

In the case of conventional LP, it is well known that the representation of prediction parameters in the form of RCs or LSFs facilitates control over the frequency characteristics when implementing actions like quantization and interpolation. The transformation from LPLP coefficients to MPP suggest that, after the mapping, the new coefficients can be processed using these known techniques. However, all these methods require the first coefficient c_0 to be equal to 1. This can be done by dividing all c_k by c_0. This mapping from Laguerre coefficients to the MPP together with the normalization [41] is performed in the Mapping to MPP block in Figure 3.10. This normalized polynomial $\hat{\boldsymbol{c}}$ is mapped to its associated RCs or LSFs and the quantization and interpolation is performed in this latter domain. The mapping of the normalized MPP to its associated RCs is depicted in the Polynomial to RC block in Figure 3.10. In Figure 3.10, the scheme is depicted where the data defining the linear Toeplitz problem of the LPLP is mapped to RCs.

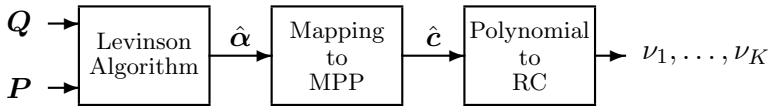


Figure 3.10: *Processing chain for the calculation of reflection coefficients associated with the optimal LPLP.*

However, there is a more straightforward way of obtaining these RCs. First from (3.37) it is easy to show that the last reflection coefficient ν_K can be calculated as

$$\nu_K \;=\; \hat{\alpha}_K/(\sqrt{1-\lambda^2}-\hat{\alpha}_1\lambda), \tag{3.38}$$

since the K^{th} reflection coefficient is the same as the K^{th} coefficient of the normalized polynomial. Furthermore, the Levinson algorithm [70] does not only recursively solves the M-dimensional Toeplitz problem as

$$\sum_{j=1}^{M} Q_{i-j}\hat{\alpha}_j^{(M)} \;=\; P_i, \text{ for } i=1,\ldots,M, \tag{3.39}$$

but implicitly the system of equations

$$\sum_{j=1}^{M} Q_{i-j}\hat{\beta}_j^{(M)} \;=\; Q_i, \text{ for } i=1,\ldots,M \tag{3.40}$$

is solved as well for $M = 1, 2, \ldots$ until $M = K-1$ for calculating the forward and backward RCs. In (3.40), Q_i denotes the *autocorrelation function of the warped input signal* (see Appendix B) and thus the optimal coefficients $\hat{\beta}_k$ are associated with a whitening WLP (or equivalently the MPP). This recursive solution involves calculation of RCs $\nu_1, \nu_2, \ldots, \nu_{K-1}$ [80], [70] which are exactly those associated with the normalized MPP.

This leads to a simplified processing as shown in Figure 3.11. Comparison of this figure with Figure 3.10 clearly shows the reduced complexity: the mappings from the LPLP coefficients to MPP and from this polynomial to RCs have disappeared. Instead, there is only (3.38) to be calculated.

After quantization/interpolation, the inverse process of reconstructing the LPLP coefficients from the associated RCs is shown in Figure 3.12.

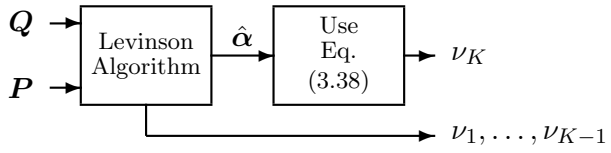


Figure 3.11: *Simplified mapping of Laguerre coefficients.*

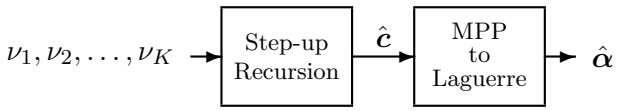


Figure 3.12: *Inverse processing chain for the calculation of LPLP coefficients associated with the reflection coefficients.*

First the RCs are used in the step-up recursion [70] block to generate the normalized polynomial coefficients $\hat{\boldsymbol{c}}$ associated with the MPP. Then the associated Laguerre coefficients are calculated by inverting (3.37) and renormalizing it [41]. This is done in the MPP to Laguerre block in Figure 3.12.

This simplification is not only important from a processing point of view, but also serves as a guideline for the definition of the multi-channel analogue of the Mapping to MPP block. The problem here is that the division $\{\frac{c_k}{c_0}\}$ becomes ambiguous for the multi-channel case. It is unclear whether the analogue division should be, for example, a left or right matrix division. The simplified version also works in the multi-channel case, where the block-Levinson algorithm directly creates the forward and backward reflection matrices $\boldsymbol{E}_1, \boldsymbol{E}_2, \ldots, \boldsymbol{E}_{K-1}$ and $\boldsymbol{E}'_1, \boldsymbol{E}'_2, \ldots, \boldsymbol{E}'_{K-1}$, respectively (see Appendix A). The last forward and backward reflection matrix has to be taken as

$$\begin{aligned} \boldsymbol{E}_K &= \boldsymbol{A}_{K,K}(\sqrt{1-\lambda^2}\boldsymbol{I} - \lambda \boldsymbol{A}_{K,1})^{-1} \\ \boldsymbol{E}'_K &= \boldsymbol{B}_{K,K}(\sqrt{1-\lambda^2}\boldsymbol{I} - \lambda \boldsymbol{B}_{K,1})^{-1}, \end{aligned} \tag{3.41}$$

where $\boldsymbol{A}_{K,k}$ and $\boldsymbol{B}_{K,k}$ are the forward and backward Laguerre-based Stereo Pure Linear Prediction (LSPLP) matrices, respectively. The associated (forward) normalized reflection matrices $\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \ldots, \boldsymbol{\xi}_K$ can be calculated using the normalization defined in Appendix A. The definition in (3.41) also ensures that $\boldsymbol{\xi}'_k = \boldsymbol{\xi}_k^T$. Note that the definition in (3.41) uses a right matrix division. Thus it implies that the matrix division necessary for obtaining the multi-channel analogue of mapping to MPP block should also be a right matrix division. The inverse processing receives the normalized reflection matrices and the zero-lag correlation matrix as an input to the block Step-up recursion (see Appendix A) to construct the multi-channel analogue of the normalized MPP. Then the associated Laguerre Stereo prediction matrices are calculated by inverting the multi-channel analogue of (3.37) and using a right matrix division for renormalization.

3.6 Conclusions

In this chapter, we developed a novel algorithm for calculating the optimal LPLP coefficients. This algorithm is based on a newly established relation between the coefficients of the normal equations and exploits the symmetries and redundancies of the Hermitian Toeplitz Gram-matrix of the LPLP scheme. Compared to the Old Method, the new method significantly reduces the computational complexity and the memory requirements. We are able to reduce the number of multiplications and memories by a factor of K^2 and N, respectively; where K (typically 10-40) is the order of the LPLP, and N is the length of the analysis window (typically 1000). On average, we observe a 45% reduction in computation time to generate the residual signal when compared to the Old Method using the MATLAB simulation tool. More importantly, if the algorithms are implemented using the C programming language, we observe an 85% reduction in computation time, implying that even for relatively high prediction orders like 20 to 30, real-time processing can be achieved. With the proposed simplification, the computational complexity of the LPLP coefficient optimization control box is equal to the complexity of the WLP coefficient optimization control box. In addition, we have found a shortcut in the known processing chain to calculate the RCs associated with the LPLP system. All the algorithms described in this chapter carry over naturally to the stereo (or multi-channel) case where it is shown to capture low-frequency coherence more than mono LPLP applied independently on two channels.

Chapter 4

Quantization of Stereo Linear Prediction Parameters

4.1 Introduction

Linear Prediction (LP) is widely used for low-bit-rate mono speech-coding applications [18]. As stated before, LP removes signal redundancies and captures them in the LP parameters. These parameters are not quantized directly because a stable filter cannot be easily verified after quantization. Instead, the LP parameters are often mapped to Line Spectral Frequencies (LSFs) or to Reflection Coefficients (RCs) which are then mapped to Arcsine Coefficients or Log Area Ratios (LARs) [68].

Compression of stereo audio signals can be achieved by means of intra- and inter-channel decorrelation methods such as SLP (see Chapter 2). An advantage of the LP-based audio coders over existing subband and transform coders is that with predictive coding it is possible to obtain a very low encoding/decoding delay [20],[19] with basically no loss of compression performance [21]. However, for a viable SLP system, we think that an efficient quantization scheme is necessary. To the best of our knowledge, quantization and efficient transmission of stereo (or multi-channel) prediction matrices have not been considered so far. In [72] direct quantization of the prediction coefficients is used. This leads to a high bit-rate (12 bits per prediction coefficient). To circumvent the problems associated with direct quantization of the prediction parameters, [71],[45],[73] proposed backward adaptive quantization. However, backward adaptive systems are not preferred in practice because they do not allow random access and are extremely vulnerable to transmission errors (for example,

packet losses).

In this chapter we develop a quantization strategy for the SLP scheme described in Chapter 2. We restrict ourselves to this scheme, as it appears to be the best generalization of the single-channel LP system to a stereo (and multi-channel) LP system. As already discussed in Chapter 2, it has a symmetric structure (no preference of left channel over right channel or vice versa), its coefficients can be efficiently calculated by the block-Levinson algorithm, and the stability of the synthesis filter can be guaranteed by using an autocorrelation approach. The last issue is especially considered as a prerequisite for developing a quantization strategy, since in the single-channel case the quantization schemes (RC based or LSF based) are built upon the minimum-phase character of the LP analysis filter.

To begin with we recapitulate the notation for the SLP scheme introduced in Chapter 2. The left and the right signals, x_1 and x_2, are the inputs to a SLP stage yielding error signals e_1 and e_2, and the system is described by

$$e_i(n) \;=\; x_i(n) - \sum_{j=1}^{2}\sum_{k=1}^{K} a_{ijk}x_j(n-k), \tag{4.1}$$

where K is the prediction order, a_{11k} and a_{22k} are the auto-predictor coefficients, and a_{21k} and a_{12k} are the cross-predictor coefficients. The transfer matrix $\boldsymbol{H}(z)$ of the analysis filter is given by

$$\boldsymbol{H}(z) \;=\; \boldsymbol{I} - \sum_{k=1}^{K} \boldsymbol{A}_{K,k} z^{-k}, \tag{4.2}$$

where $\boldsymbol{I}$ denotes the 2×2 identity matrix and the prediction matrix $\boldsymbol{A}_{K,k}$ is given by

$$\boldsymbol{A}_{K,k} \;=\; \begin{bmatrix} a_{11k} & a_{12k} \\ a_{21k} & a_{22k} \end{bmatrix}. \tag{4.3}$$

As a drawback of the symmetric SLP scheme, it might be argued that they are inherently incapable of removing the correlations between the current samples of x_1 and x_2. However, as discussed in Section 2.4, this can be easily achieved by cascading a rotator to a symmetric SLP scheme [77].

As described in Chapter 2, the estimation of the prediction matrix $\boldsymbol{A}_{K,k}$ is done by jointly minimizing the mean square error signals e_1 and e_2.

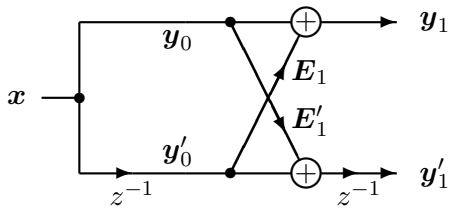


Figure 4.1: *First stage of the block FIR lattice filter.*

The prediction matrices $\boldsymbol{A}_{K,k}$ can be calculated efficiently by the block-Levinson algorithm, where 2×2 forward and backward reflection matrices $\boldsymbol{E}_k$ and $\boldsymbol{E}'_k$, respectively, appear as analogues of forward and backward RCs in the mono case. We will consistently use the superscript $'$ to denote backward signals and matrices. Similar to the classical Levinson algorithm for the mono LP, the block-Levinson algorithm leads to an FIR lattice filter, but now a block FIR lattice filter. The first stage of such a structure is shown in Figure 4.1. The elements of the input vector $\boldsymbol{x} = \boldsymbol{y}_0 = \boldsymbol{y}'_0$ are the signals x_1 and x_2. The vectors $\boldsymbol{y}_1$ and $\boldsymbol{y}'_1$ denote the first-order forward and backward prediction error signals, respectively.

The problems when trying to develop a quantization scheme for SLP matrices are the following.

1. Analogous to mono LP, direct quantization of the entries in the prediction matrices allows little control over the sensitivity of the transfer characteristics; e.g., the determinant of the transfer matrix $\boldsymbol{H}(z)$ could easily change drastically, which can even result in an unstable synthesis system.

2. Quantization of individual polynomials (i.e., the entries of the transfer matrix $\boldsymbol{H}(z)$) by standard procedures [68] is not an option, because these polynomials do not adhere to the minimum-phase requirement.

3. In contrast to the mono LP, where forward and backward RCs are directly coupled [70], for the multi-channel case the information of the forward reflection matrices is not sufficient to construct the backward reflection matrices [67]. The fundamental reason why the forward and backward reflection matrices are not directly coupled in an SLP system can be explained as follows. For the single-channel

LP, we are modeling the autocorrelation function which is symmetric. In contrast, with SLP we are also modeling the cross-correlation function, which is in general not symmetric.

Before we proceed to tackle these problems, let us mention a few useful identities of the block-Levinson algorithm [66]. The input to the block-Levinson algorithm is the 2×2 correlation matrix $\boldsymbol{C}_k$ given by

$$\boldsymbol{C}_k = \begin{bmatrix} r_{11}(k) & r_{12}(k) \\ r_{21}(k) & r_{22}(k) \end{bmatrix}, \tag{4.4}$$

for $k = 0, \pm 1, \cdots, \pm K$. The entries of this matrix involve the correlation function between the channels defined by $r_{pq}(k) = \sum_n x_p(n) x_q(n-k)$, where p and q can be 1 or 2, and n ranges from $-\infty$ to ∞. Along with the forward and backward reflection matrices, the block-Levinson algorithm calculates the 2×2 positive-definite forward and backward error or innovation variance matrices, denoted as $\boldsymbol{R}_k$ and $\boldsymbol{R}'_k$, respectively. It is known [66] that

$$\boldsymbol{R}_{k-1}\boldsymbol{E}'_k \;\; = \;\; \boldsymbol{E}_k^T \boldsymbol{R}'_{k-1}, \tag{4.5}$$

and that a recurrence relation holds for the innovation variance matrices

$$\begin{aligned} \boldsymbol{R}_k &= \boldsymbol{R}_{k-1}(\boldsymbol{I} - \boldsymbol{E}'_k \boldsymbol{E}_k) = \boldsymbol{R}_{k-1} - \boldsymbol{E}_k^T \boldsymbol{R}'_{k-1} \boldsymbol{E}_k, \\ \boldsymbol{R}'_k &= \boldsymbol{R}'_{k-1}(\boldsymbol{I} - \boldsymbol{E}_k \boldsymbol{E}'_k) = \boldsymbol{R}'_{k-1} - \boldsymbol{E}_k'^T \boldsymbol{R}_{k-1} \boldsymbol{E}'_k, \end{aligned} \tag{4.6}$$

where $\boldsymbol{R}_0 = \boldsymbol{C}_0$.

Consider factorizing $\boldsymbol{R}_k$ and $\boldsymbol{R}'_k$ in the form [66]

$$\boldsymbol{R}_k = \boldsymbol{M}_k^T \boldsymbol{M}_k, \;\; \boldsymbol{R}'_k = \boldsymbol{M}_k'^T \boldsymbol{M}'_k, \tag{4.7}$$

for suitable 2×2 normalizing matrices $\boldsymbol{M}_k$ and $\boldsymbol{M}'_k$. The matrices $\boldsymbol{M}_k$ and $\boldsymbol{M}'_k$ can be calculated within unitary left factors. Then the 2×2 normalized reflection matrix $\boldsymbol{\xi}_k$ is defined by

$$\boldsymbol{\xi}_k = \boldsymbol{M}'_{k-1} \boldsymbol{E}_k \boldsymbol{M}_{k-1}^{-1} = \left(\boldsymbol{M}_{k-1}'^T\right)^{-1} \boldsymbol{E}_k'^T \boldsymbol{M}_{k-1}^T, \tag{4.8}$$

and a similar definition exists for $\boldsymbol{\xi}'_k$. Stability of the synthesis filter is only guaranteed if the magnitude of the singular values of $\boldsymbol{\xi}_k$ are less than unity (i.e., $\boldsymbol{\xi}_k$ is *strictly contractive*) [65],[66],[92]. This condition is analogous to the mono LP, where the magnitude of the RCs strictly less than unity guarantees the stability of the synthesis filter.

To resolve the ambiguity in $\boldsymbol{M}_k$ and $\boldsymbol{M}'_k$, we define these matrices as positive-definite symmetric matrices[1]. Since $\boldsymbol{R}_k$ can be decomposed as $\boldsymbol{R}_k = \boldsymbol{U}_k \breve{\boldsymbol{S}}_k \boldsymbol{U}_k^T$ (that is, the standard singular value definition [80] of a positive-definite matrix), this means that we take $\boldsymbol{M}_k = \boldsymbol{U}_k \breve{\boldsymbol{S}}_k^{1/2} \boldsymbol{U}_k^T$. Thus we have

$$\boldsymbol{M}_k = \boldsymbol{R}_k^{1/2}, \ \boldsymbol{M}'_k = \boldsymbol{R}_k^{\prime 1/2}. \tag{4.9}$$

The initial values are taken to be $\boldsymbol{M}_0 = \boldsymbol{M}'_0 = \boldsymbol{C}_0^{1/2}$.

From the previous discussion, we infer the following.

- The reflection matrices $\boldsymbol{E}_k$ and $\boldsymbol{E}'_k$ can be mapped onto the normalized reflection matrices $\boldsymbol{\xi}_k$ and $\boldsymbol{\xi}'_k$, which are directly coupled according to $\boldsymbol{\xi}'_k = \boldsymbol{\xi}_k^T$, thus solving Problem 3.
- For the inverse mapping $\{\boldsymbol{\xi}_k\} \rightarrow \{\boldsymbol{E}_k, \boldsymbol{E}'_k\}$, we need the matrices $\{\boldsymbol{M}_{k-1}, \boldsymbol{M}'_{k-1}\}$. Due to the recursive relation between successive $\boldsymbol{M}_{k-1}$, we only need to transmit $\boldsymbol{M}_0$ (or, alternatively, the zero-lag correlation matrix $\boldsymbol{C}_0$).
- Having the set $\{\boldsymbol{E}_k, \boldsymbol{E}'_k\}$, the prediction matrices $\boldsymbol{A}_{K,k}$ can be generated using the block Step-up recursion (see Appendix A).

This means that transmission of the (forward) normalized reflection matrices together with the zero-lag correlation matrix instead of the forward and backward reflection matrices is much more efficient. In fact, now we need to transmit $4K+4$ data points (matrix entries) instead of $4K+4K$. Later on we will see that for reconstruction purposes we only need two parameters to describe $\boldsymbol{C}_0$, implying that we need to transmit only $4K+2$ data points.

A filter implementation based on the normalized reflection matrices is also possible and is shown in Figure 4.2 for a first-order system. It contains the matrices $\boldsymbol{\xi}_1$, $\boldsymbol{M}_0$, and $\boldsymbol{M}_0^{-1}$ as constituent multipliers. On the top and bottom dashed boxes, we have indicated the variance matrices of the signals at specific positions in the network. The function of the matrix $\boldsymbol{M}_k^{-1}$ is to rotate and scale the incoming signal vector $\boldsymbol{y}_k$ to a signal vector $\tilde{\boldsymbol{y}}_k$ having an identity variance matrix.

In the remainder of the chapter, methods for efficient quantization of the matrices $\boldsymbol{\xi}_k$ (since these are the logical counterparts of the RCs) and

[1] In [93], a different definition of $\boldsymbol{M}_k$ and $\boldsymbol{M}'_k$ was suggested, where they were not necessarily symmetric matrices. However, in our experiments, we found that the current definition yields a slightly lower mean spectral distortion.

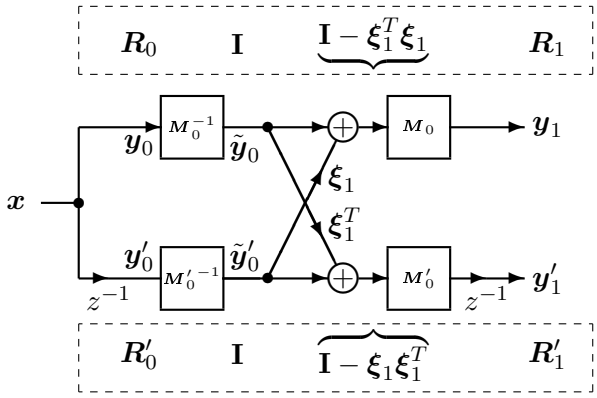


Figure 4.2: *First stage of the normalized block FIR lattice filter. The matrices in the top and bottom dashed boxes denote the variance matrices of the signals at specific positions in the network in the forward and backward prediction paths, respectively.*

$\boldsymbol{C}_0$ (since this is the logical counterpart of filter gain for the mono LP) are considered. In the first instance, we will only consider a first-order SLP system, and later check if the method carries over to higher-order systems.

Consequently, we desire a way of quantizing the data contained in the first normalized reflection matrix $\boldsymbol{\xi}_1$ and in $\boldsymbol{C}_0$, such that a simple way of controlling the distortion as a function of bit-rate is available. More specifically, for the parameters describing the matrices we require that the sensitivities of the individual parameters to quantization should be equal across its range while performing uniform quantization. Secondly, the distortions introduced by quantizing the individual parameters should be decoupled. By the latter we mean that the distortions should add up in an uncorrelated way to the total distortion when quantizing all the parameters simultaneously.

In Section 4.2, the proposal for parameterization and quantization of $\boldsymbol{\xi}_1$ and $\boldsymbol{C}_0$ is given. In Section 4.3, we will first introduce the distortion measure and evaluate the performance of the first-order system based on randomly generated SLP systems. Next, we consider the performance of higher-order systems. In Section 4.4 we shortly discuss the use of another parameterization technique for $\boldsymbol{\xi}_1$ and we point out the reasons why it is not better than our proposed approach. We end with our conclusions in Section 4.5.

4.2 SLP Transmission Parameters

In order to quantize the normalized reflection matrices, it is proposed to decompose these matrices in structures that effectively hold the major matrix characterizations, e.g. Singular Value Decomposition (SVD) and/or Eigenvalue Decomposition (EVD). The rationale behind this approach is that for strictly contractive matrices, the eigenvalues and singular values have similar characteristics as RCs. To clarify this statement, let us assume two linearly independent signals. For this case, the SLP reduces to two single-channel LP systems since the optimal cross-predictors are equal to zero. One can then show that the RCs from the single-channel LP systems are equal in magnitude to the singular values (and eigenvalues) of the normalized reflection matrices associated with the SLP system. Thus, the sensitivities to quantization for the singular values (and eigenvalues) are similar to RCs and, with small adaptations can be quantized using known techniques to quantize RCs [68], e.g., mapping them to LARs or Arcsine Coefficients and performing a scalar quantization, or predictive quantization, or vector quantization (VQ), without deviating from the essential idea. We used the LAR mapping, defined as

$$L = \ln\left(\frac{1+\nu}{1-\nu}\right), \tag{4.10}$$

where ν is the RC, with $|\nu| < 1$.

In addition to the singular values, the decomposition generates two additional parameters. These additional parameters can be quantized efficiently, if their quantization accuracy is adapted according to the singular values and/or eigenvalues.

Several alternative sets of parameterizations of the normalized reflection matrices, such as SVD, EVD, and a combination of both SVD and EVD are investigated as transmission parameters for SLP coding systems. Although each of these parameterizations provides equivalent information about the SLP, their performances under quantization are different. Comparison of various parameterization schemes suggest that a variant of SVD is the best method for parameterization of $\boldsymbol{\xi}_1$. This method is considered in Section 4.2.1. The parameterization of $\boldsymbol{C}_0$ is considered in Section 4.2.2. Disadvantages of parameterizations of the normalized reflection matrices based on EVD are listed in Section 4.4.

The proposed quantization scheme is developed for an SLP system based on a tapped-delay-line. With minor adaptations, it is also applica-

ble to prediction matrices appearing in Warped Stereo Linear Prediction and Laguerre-based Stereo Pure Linear Prediction (LSPLP). The reader is referred to Chapter 3 for the mapping from LSPLP prediction matrices to its associated normalized reflection matrices.

4.2.1 Parameterization of the Normalized Reflection Matrix

Consider the following variant of SVD for decomposing the normalized reflection matrix of a first-order SLP:

$$\boldsymbol{\xi}_1 \;=\; \mathbf{R}(\alpha)\boldsymbol{S}\mathbf{R}(-\beta), \tag{4.11}$$

where $\mathbf{R}$ is 2×2 *rotation matrix* (defined by an angle α or β) and $\boldsymbol{S}$ is a 2×2 real diagonal matrix with

$$\boldsymbol{S} \;=\; \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix}, \tag{4.12}$$

where $0 \leq |\sigma_2| \leq \sigma_1 < 1$. Although we deviate from the standard SVD definition [80], which has $\boldsymbol{\xi}_1 = \boldsymbol{U}\tilde{\boldsymbol{S}}\boldsymbol{V}^T$ and where the diagonal elements of $\tilde{\boldsymbol{S}}$ are non-negative, the elements σ_1 and σ_2 are still referred to as the singular values. The reason for deviating from the standard SVD definition is because it simplifies the parameterization of the matrix $\boldsymbol{\xi}_1$. In the standard SVD definition, the matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ are general orthonormal matrices, meaning a combination of a rotation and a line mirroring operation. In the proposed variant of the SVD, the line mirroring operations are absorbed in the diagonal matrix $\boldsymbol{S}$ and, consequently, the $\boldsymbol{U}$ and $\boldsymbol{V}$ matrices reduce to pre- and post-rotation matrices, respectively (see (4.11)). We rewrite (4.11) to

$$\boldsymbol{\xi}_1 = \mathbf{R}(\gamma)\mathbf{R}(\delta)\boldsymbol{S}\mathbf{R}(\delta)\mathbf{R}(-\gamma), \tag{4.13}$$

with $\gamma = (\alpha + \beta)/2$ and $\delta = (\alpha - \beta)/2$, where the parameters γ and δ can be restricted to the range $(-\pi/2, \pi/2]$.

The parameters after decomposition can be efficiently calculated from the entries of $\boldsymbol{\xi}_1$ (see Appendix C). The rotation angles are given by

$$\gamma \;=\; \frac{1}{2}\tan^{-1}\left(\frac{\xi_{21} + \xi_{12}}{\xi_{11} - \xi_{22}}\right), \quad -\pi/2 < \gamma \leq \pi/2, \tag{4.14}$$

$$\delta \;=\; \frac{1}{2}\tan^{-1}\left(\frac{\xi_{21} - \xi_{12}}{\xi_{11} + \xi_{22}}\right), \quad -\pi/2 < \delta \leq \pi/2. \tag{4.15}$$

The singular values are given by

$$\sigma_1 = \frac{\xi_{11}\cos(\delta-\gamma) - \xi_{12}\sin(\delta-\gamma)}{\cos(\delta+\gamma)} \tag{4.16}$$

or

$$\sigma_1 = \frac{\xi_{21}\cos(\delta-\gamma) - \xi_{22}\sin(\delta-\gamma)}{\sin(\delta+\gamma)} \tag{4.17}$$

and

$$\sigma_2 = \frac{\xi_{11}\sin(\delta-\gamma) + \xi_{12}\cos(\delta-\gamma)}{-\sin(\delta+\gamma)} \tag{4.18}$$

or

$$\sigma_2 = \frac{\xi_{21}\sin(\delta-\gamma) + \xi_{22}\cos(\delta-\gamma)}{\cos(\delta+\gamma)} \tag{4.19}$$

The σ_1 and σ_2 with highest numerical robustness is selected.

As already noted, the character of the singular values is similar to that of an RC and, therefore, we propose to map them to LARs and quantize uniformly in the latter domain. The angle δ can be best quantized depending on the magnitude of the singular values. In particular, we propose to quantize δ such that the *characteristic polynomial* (determining the eigenvalues) of $\boldsymbol{\xi}_1$ is also represented as accurately as possible.

The characteristic polynomial of $\boldsymbol{\xi}_1$ is given by

$$\lambda^2 - (\sigma_1+\sigma_2)\cos(2\delta)\lambda + \sigma_1\sigma_2, \tag{4.20}$$

where the roots of (4.20) are the eigenvalues of the matrix $\boldsymbol{\xi}_1$. Note that (4.20) does not depend on γ, and we assume that γ can be best quantized uniformly. The fact that (4.20) only depends on δ was the reason for introducing γ and δ instead of α and β. Visualizing the characteristic polynomial as a second-order mono LP polynomial, we can calculate the associated RCs (by applying the *Reverse Levinson-Durbin* recursion [70]), which are given by

$$\kappa_1 = -\frac{\sigma_1+\sigma_2}{1+\sigma_1\sigma_2}\cos(2\delta),\ \kappa_2 = \sigma_1\sigma_2. \tag{4.21}$$

Since we already transmit σ_1 and σ_2, we only need to transmit κ_1. For convenience, we refer to κ_1 as κ. We note that $|\kappa| < 1$, hence we can map them to LAR and quantize uniformly in the latter domain.

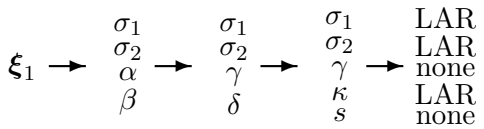


Figure 4.3: *Preprocessing steps before uniform quantization for the normalized reflection matrix.*

In the encoder, κ is calculated using the quantized singular values as this relation has to be inverted in the decoder and, there, only the quantized singular values are available. The decoder mapping $\hat{\kappa} \rightarrow \hat{\delta}$ (where $\hat{}$ denotes quantized parameter in the decoder) is ambiguous because $\cos(2\hat{\delta}) = \cos(-2\hat{\delta})$. To resolve this ambiguity, one extra sign-bit s needs to be transmitted. In Figure 4.3, a sketch of the proposed preprocessing steps before uniform quantization is shown. The parameters σ_1, σ_2, and κ are mapped onto LARs and are then uniformly quantized.

The decoder implements the inverse process. It receives the quantized parameters and reconstructs $\hat{\sigma}_1$ and $\hat{\sigma}_2$. Given these values, the receiver is able to reconstruct $\hat{\delta}$ from $\hat{\kappa}$ and s. From $\hat{\gamma}$ and $\hat{\delta}$, the rotation matrices $\mathbf{R}(\hat{\gamma})$ and $\mathbf{R}(\hat{\delta})$ can be reconstructed. Thus, $\hat{\boldsymbol{\xi}}_1$ can be computed in the decoder.

4.2.2 Parameterization of the Zero-Lag Correlation Matrix

The zero-lag correlation matrix $\boldsymbol{C}_0$ can be expressed as

$$\boldsymbol{C}_0 = \begin{bmatrix} r_{11}(0) & r_{12}(0) \\ r_{21}(0) & r_{22}(0) \end{bmatrix} = \sqrt{r_{11}(0)r_{22}(0)} \begin{bmatrix} \eta & \rho \\ \rho & 1/\eta \end{bmatrix}, \tag{4.22}$$

where $r_{11}(0) \geq 0$, $r_{22}(0) \geq 0$ represent the power of signal x_1 and x_2, respectively, and $r_{12}(0) = r_{21}(0) = \rho\sqrt{r_{11}(0)r_{22}(0)}$ represents the cross-power of the signals x_1 and x_2. The cross-correlation coefficient is given by ρ with $|\rho| < 1$ and should be carefully treated when close to ± 1. This is similar to the situation when quantizing an RC, hence we can map ρ to LAR and quantize uniformly in the latter domain.

The parameter η represents the ratio between the signal powers, and is given by $\eta = \sqrt{r_{11}(0)/r_{22}(0)}$, with $\eta \geq 0$. We now consider the mapping

$$\mu = \frac{\eta - 1/\eta}{\eta + 1/\eta} = \frac{r_{11}(0) - r_{22}(0)}{r_{11}(0) + r_{22}(0)}. \tag{4.23}$$

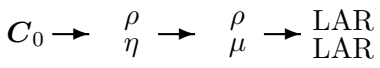


Figure 4.4: *Preprocessing steps before uniform quantization for the zero-lag correlation matrix.*

We note that if the powers are equal, then $\mu = 0$; and if they are largely unequal, then $|\mu| \approx 1$. In the latter case, the matrix $\boldsymbol{C}_0$ becomes numerically ill-conditioned, and therefore, in these ranges μ has to be quantized more accurately. This is again similar to the situation when quantizing an RC, hence μ can be mapped to LAR and quantized uniformly in the latter domain. The factor $\sqrt{r_{11}(0)r_{22}(0)}$ in (4.22) gets cancelled while reconstructing $\hat{\boldsymbol{E}}_1$ from $\hat{\boldsymbol{\xi}}_1$ using (4.8), and therefore need not be transmitted. In Figure 4.4, a sketch of the proposed preprocessing steps before uniform quantization is shown. The parameters ρ and μ are mapped onto LARs and then uniformly quantized.

The decoder implements the inverse process. It receives the quantized parameters $\hat{\rho}$ and $\hat{\mu}$. Given $\hat{\mu}$, the receiver is able to reconstruct $\hat{\eta}$ from

$$\hat{\eta} = \sqrt{\frac{1+\hat{\mu}}{1-\hat{\mu}}}. \tag{4.24}$$

From $\hat{\rho}$, $\hat{\eta}$, and $\hat{\boldsymbol{\xi}}_1$, the reflection matrix $\hat{\boldsymbol{E}}_1$ can be reconstructed using (4.8). Finally, $\hat{\boldsymbol{A}}_{K,1}$ is obtained from $\hat{\boldsymbol{E}}_1$ by using the block Step-up recursion (see Appendix A).

4.3 Sensitivity Analysis

To evaluate our quantization scheme, we introduce an objective measure analogous to the Spectral Distortion (SD) measure for the mono case defined in [68]. In particular, we take the root mean square difference between the matrix norms of the synthesis filter frequency response matrices of the quantized and original (unquantized) prediction schemes; the matrix norm being defined as the larger of the magnitudes of the two singular values. In formula, this reads as

$$D = \sqrt{\frac{1}{F_s}\int_0^{F_s} [20\log_{10}(N(f)) - 20\log_{10}(\hat{N}(f))]^2 df}, \tag{4.25}$$

where F_s is the sampling frequency in Hz, and $N(f)$ and $\hat{N}(f)$ are the matrix norms of $\boldsymbol{H}(e^{j2\pi f/F_s})^{-1}$ and $\hat{\boldsymbol{H}}(e^{j2\pi f/F_s})^{-1}$, respectively, with $\boldsymbol{H}(z)$ and $\hat{\boldsymbol{H}}(z)$ the original and the quantized SLP transfer matrix. The quantized transfer matrix $\hat{\boldsymbol{H}}(z)$ is defined similar to (4.2) as $\hat{\boldsymbol{H}}(z) = \boldsymbol{I} - \sum_{k=1}^{K} \hat{\boldsymbol{A}}_{K,k} z^{-k}$.

We tested the system using randomly generated normalized reflection and zero-lag correlation matrices. The analysis using random systems has the advantage of covering all possible situations that may occur. We first consider first-order systems and individual parameter quantization (Section 4.3.1), continue with simultaneous parameter quantization for first-order systems (Section 4.3.2) and finally discuss the performance of the proposed quantization scheme for higher-order SLP systems (Section 4.3.3).

4.3.1 Single Parameter Quantization of First-Order Systems

In order to determine the sensitivity of the individually quantized parameters, a number of trials was run. For each trial, we generated a random strictly contractive normalized reflection matrix and a random correlation matrix (with $|\rho| < 1$ and $\eta \geq 0$). These matrices were then parameterized according to the method outlined in Section 4.2, and each parameter was then quantized separately. For each quantized parameter, we measured the SD. The number of trials was set to 5000. From this we determined the mean of the SD along with the entropy of the quantized parameter for different step sizes. Note that except for the parameter γ and the sign bit s, all the step sizes are in the LAR domain.

In Figure 4.5, the SD is plotted as a function of the parameters σ_1, σ_2, κ, γ, ρ and μ where we quantized only the LAR of σ_2 with a step size of 2^{-3}. In these plots (so-called scatter plots), each dot denotes an SD for a single trial. Similar plots were made when quantizing any one of the other parameters. These are not shown here because the depicted data are representative of the other cases. The reader is referred to [93] for the scatter plots of the SD when quantizing σ_1. Since Figure 4.5 gives the SD when quantizing σ_2, we see in the plot of the SD as function of σ_2 (top-right in Figure 4.5) triangular shapes. The plot is very similar to the scatter plot when quantizing an RC of a mono LP system [94]. Furthermore, there is no clear indication of dependence between the magnitude of the measured SD and any of the other parameters, suggesting that there is no large correlation. Only if μ is close to ± 1, or if κ is around

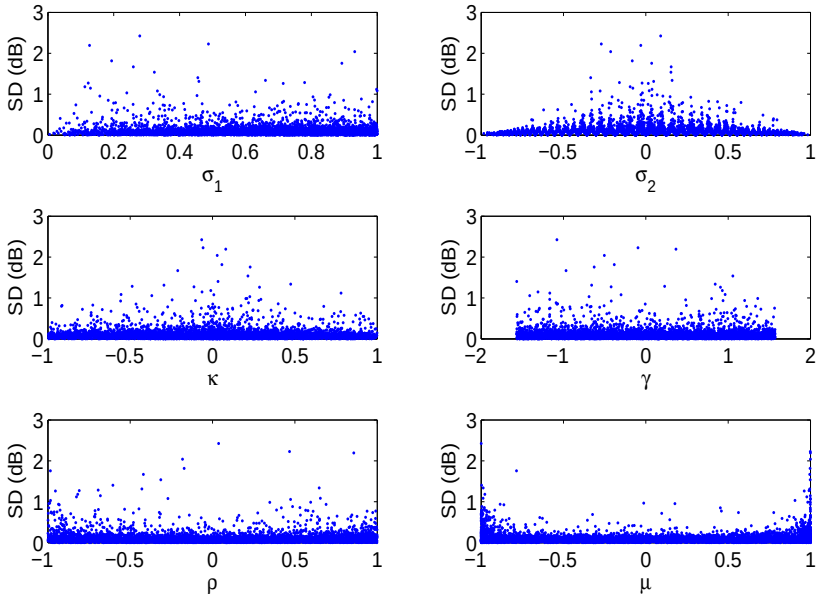


Figure 4.5: *Scatter plots of the SD when quantizing* σ_2.

0, there is a tendency that the error due to quantization of σ_2 increases. Nevertheless, we conclude that decoupling is attained to a large degree; we will return to this issue in the next section.

The mean of the SD along with the entropy of the quantized parameter as a function of the step size for each individually quantized parameter is given in Table 4.1. In the top plot in Figure 4.6, the mean SD is plotted as a function of step size for each individually quantized parameter. For clarity, the mean SD obtained while quantizing γ and κ are scaled-up in the plot by a factor of two and four, respectively. For the same reason, the mean SD obtained while quantizing μ is scaled down by half. We clearly observe a nearly linear relation in all cases, except for κ. The reason for this deviation for κ is due to its dependence on σ_1 and σ_2. Nevertheless, we assume for convenience that for each parameter p_i, the distortion $D(p_i)$ as a function of step size $Q(p_i)$ can be modeled as

$$D(p_i) = d(p_i)Q(p_i), \tag{4.26}$$

where p_i denotes the i^{th} parameter with $i = 1, 2, \ldots, 4K+2$. This relation is depicted in the top plot in Figure 4.6 by the solid lines. The values of

Table 4.1: *Spectral Distortion (SD) performance, along with the entropy of the quantized parameter as a function of quantization step size for individually quantized parameters.*

Step Size	Quantization of σ_1 Avg. SD (dB)	Entropy (bits)	Quantization of σ_2 Avg. SD (dB)	Entropy (bits)	Quantization of κ Avg. SD (dB)	Entropy (bits)
2^{-1}	0.2759	3.18	0.4720	3.06	0.4771	3.48
2^{-2}	0.1320	4.20	0.2478	4.06	0.2709	4.42
2^{-3}	0.0677	5.15	0.1159	5.05	0.1585	5.39
2^{-4}	0.0350	6.12	0.0578	6.04	0.0880	6.39
2^{-5}	0.0172	7.12	0.0347	7.00	0.0474	7.36
2^{-6}	0.0087	8.09	0.0146	7.97	0.0257	8.34
2^{-7}	0.0044	9.04	0.0079	8.90	0.0147	9.24
Step Size	Quantization of γ Avg. SD (dB)	Entropy (bits)	Quantization of ρ Avg. SD (dB)	Entropy (bits)	Quantization of μ Avg. SD (dB)	Entropy (bits)
2^{-1}	0.6154	2.78	0.1729	3.87	0.2209	4.46
2^{-2}	0.2986	3.69	0.0862	4.86	0.1087	5.38
2^{-3}	0.1574	4.66	0.0425	5.83	0.0546	6.38
2^{-4}	0.0819	5.66	0.0215	6.84	0.0281	7.36
2^{-5}	0.0442	6.65	0.0108	7.81	0.0138	8.28
2^{-6}	0.0228	7.62	0.0054	8.75	0.0070	9.21
2^{-7}	0.0109	8.59	0.0027	9.66	0.0035	10.02

$d(p_i)$ are given in Table 4.2.

The middle plot in Figure 4.6 shows how the step sizes $Q(p_i)$ for each quantized parameter are related to the entropy of the quantized parameter measured in number of bits $b(p_i)$. This relation is modeled as

$$Q(p_i) = c(p_i)2^{-b(p_i)} \tag{4.27}$$

and depicted by the solid lines in the middle plot in Figure 4.6. For clarity, the step sizes associated with the entropies of σ_2, σ_1, κ, ρ, and μ are scaled-up in the plot by a factor of two, three, four, five, and six, respectively. The fitted constants $c(p_i)$ are given in Table 4.2.

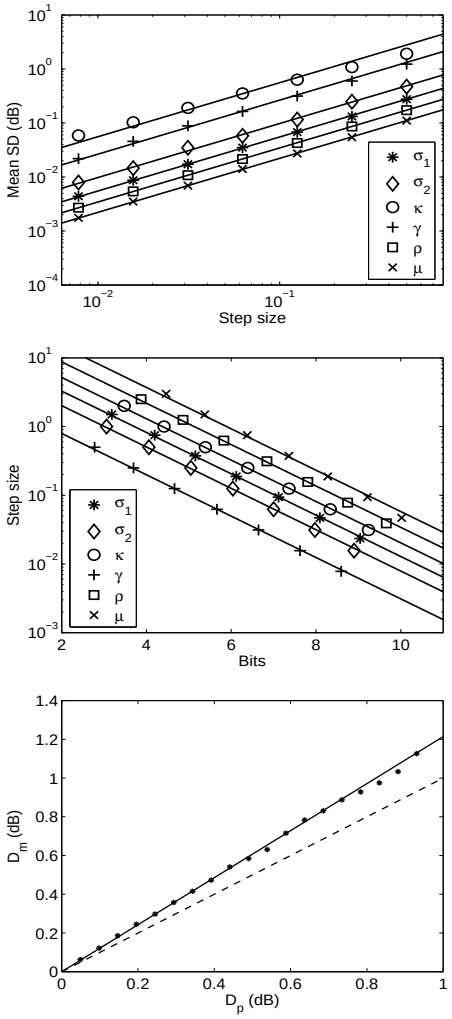


Figure 4.6: *Mean SD as a function of quantization step size for each individually quantized parameter (top plot). Quantization step size Q for each quantized parameter as a function of the number of bits b (entropy of the quantized parameter in Table 4.1) (middle plot). Measured mean SD denoted by D_m versus predicted overall mean distortion D_p (bottom plot). The dashed line represents $D_m = D_p$ and the solid line represents $D_m = 1.21 D_p$.*

Table 4.2: *Regression parameters $c(p_i)$ and $d(p_i)$ for SLP systems with order $K = 1, 2, 3$.*

		c	d					
			$K=1$	$K=2$		$K=3$		
			$k=1$	$k=1$	$k=2$	$k=1$	$k=2$	$k=3$
$\boldsymbol{\xi}_k$	σ_1	4.38	0.55	1.05	0.78	1.21	1.13	0.89
	σ_2	4.03	0.98	1.20	1.49	1.30	1.41	1.69
	κ	5.19	1.39	1.72	1.77	1.95	1.93	1.99
	γ	3.18	1.32	2.22	2.17	2.68	2.80	2.67
$\boldsymbol{C}_0$	ρ	6.99	0.34	0.49		0.59		
	μ	9.89	0.44	0.61		0.71		

4.3.2 Simultaneous Parameter Quantization of First-Order Systems

If the distortions introduced by the individual quantization are decoupled, then the distortion should add up in an uncorrelated way to the total distortion while quantizing all the parameters simultaneously. This means that the variance of the total distortion should be the sum of the separate variances. Therefore, setting the quantization such that the variance due to individual distortions equals D_{eq}^2, and having a total number of $4K + 2$ parameters (K being the prediction order), the predicted mean total distortion D_p becomes $D_p = \sqrt{(4K + 2)}D_{eq}$.

To check this, we used the following procedure. First, an equal mean SD denoted by D_{eq} is chosen for all six parameters of the individual quantization experiments. From the top plot in Figure 4.6, we can determine the corresponding step size for each parameter. With these selected step sizes, we quantize all six parameters at the same time. In case of decoupling we would predict the overall mean distortion D_p as $D_p = \sqrt{6}D_{eq}$.

The bottom plot in Figure 4.6 displays the measured mean SD denoted by D_m obtained by quantizing all six parameters simultaneously, along with D_p. The dashed line represents $D_m = D_p$ and the solid line represents $D_m = 1.21D_p$. This latter line indicates that the measured data are roughly 21% larger than the predicted values. Apparently, the distortions do not add up in a completely uncorrelated way, yet decoupling is achieved to a high extent. Nevertheless, the bottom plot in Figure 4.6 shows that the actual total distortion can be accurately predicted from the individual distortions as indicated by the solid line, and

consequently from the quantization step size or the number of bits. This highly predictable behavior is obviously attractive because a computationally efficient control box for determining the optimal bit allocation over the different parameters is feasible due to the simple relations between distortion and assigned bits.

4.3.3 Quantization of Higher-Order Systems

The proposed scheme can be applied to higher-order systems, where every normalized reflection matrix $\boldsymbol{\xi}_k$ $(k = 1, 2, \cdots, K)$ is quantized by the method proposed for $\boldsymbol{\xi}_1$. The effect of quantizing the parameters of every normalized reflection matrices on the transfer characteristic of a higher-order SLP is investigated next. This needs to be considered separately because it is unclear whether the results obtained for the first-order systems carry over to higher-order systems. The reason why this is *a priori* unclear is the following. Looking at the reconstruction algorithm, we note that from $\boldsymbol{C}_0$ (or $\boldsymbol{R}_0$) and $\boldsymbol{\xi}_1$, the reflection matrices $\boldsymbol{E}_1$ and $\boldsymbol{E}'_1$ and the matrix $\boldsymbol{R}_1$ are constructed; which, in conjunction with $\boldsymbol{\xi}_2$, yields the reflection matrices $\boldsymbol{E}_2$ and $\boldsymbol{E}'_2$ and $\boldsymbol{R}_2$, and so forth. This means that $\boldsymbol{R}_k$ actually depends on $\boldsymbol{R}_0$ and all the normalized reflection matrices of order lower than $k + 1$. This could imply that the algorithm introduces an accumulation of quantization errors and, thus, the higher-order systems are intrinsically more difficult to quantize. In particular, it would suggest that there may be a need to quantize all normalized reflection matrices $\boldsymbol{\xi}_k$ $(k = 1, 2, \cdots, K)$ in a dependent way rather than quantizing them independently.

To consider the quantization effects for higher-order systems, we repeated the experiments from the previous section for the second- and third-order systems, where we generated in each trial one random zero-lag correlation matrix and two (for second-order) or three (for third-order) random strictly contractive normalized reflection matrices. The normalized reflection matrices were drawn from the same distribution as the first-order, i.e., independent of the order. Whereas for the first-order system we were dealing with six parameters (4 for $\boldsymbol{\xi}_1$ and 2 for $\boldsymbol{C}_0$) that had to be quantized, the second- and third-order systems involve ten (8 for $\boldsymbol{\xi}_k$ and 2 for $\boldsymbol{C}_0$) and fourteen parameters (12 for $\boldsymbol{\xi}_k$ and 2 for $\boldsymbol{C}_0$), respectively.

In Figure 4.7 we plotted the mean SD as a function of step size for individually quantized parameters. Like in the first-order case, we observe a nearly linear relationship between step size and mean SD, which can be

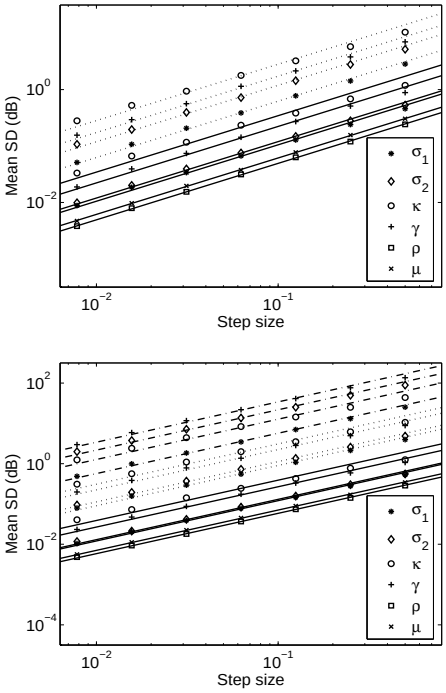


Figure 4.7: *Mean SD as a function of quantization step size for each individually quantized parameter of a second-order (top plot) and a third-order system (bottom plot). Solid lines are associated with the parameters of the first normalized reflection matrix and the zero-lag correlation matrix. The dotted and dash-dotted lines are associated with the parameters of the second and third normalized reflection matrix, respectively. For clarity, vertical shifts have been applied to the data.*

modeled by (4.26). This relation is depicted in the top and the bottom plots in Figure 4.7 by the straight lines. The values of d_i are given in Table 4.2. We note that we need not replicate the measurement of the entropies as a function of step size, since the statistics of $\boldsymbol{C}_0$ and the normalized reflection matrices are exactly equal to those of the first-order system in the chosen experimental setup. However, we note that when working with real (e.g. speech or audio) data, this will presumably not be the case. There the statistics of the normalized reflection matrix are expected to depend on the order just like in the single-channel case [94].

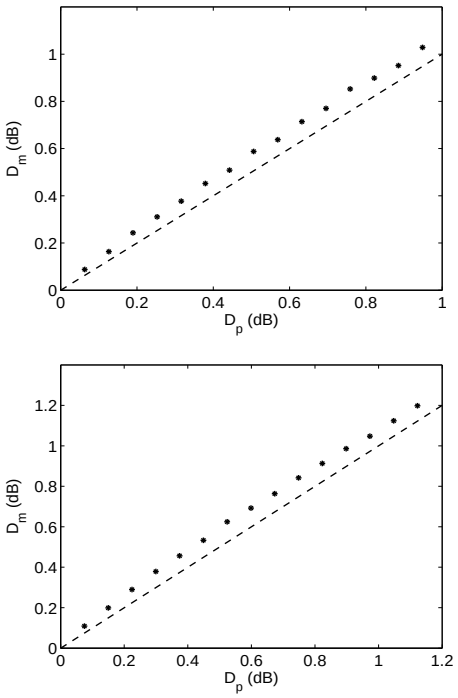


Figure 4.8: *Measured mean SD denoted by D_m versus predicted overall mean distortion D_p for a second-order (top plot) and a third-order SLP system (bottom plot). The dashed line represents $D_m = D_p$.*

Based on the mean SD introduced by the individual quantization, we checked whether the introduced distortions are uncorrelated when quantizing a second- or third-order system. Like in the experiments for the first-order systems, we chose a step size such that the mean individual distortions are equal D_{eq} and, from that, we predicted the overall distortion D_p as $\sqrt{10}D_{eq}$ (for a second-order system) and $\sqrt{14}D_{eq}$ (for a third-order system). From Figure 4.8 we observe that also for higher-order systems D_p and D_m are very much in line, and therefore we conclude that the proposed quantization is such that the distortions add up in an almost uncorrelated way. This implies that also for higher-order systems, the proposed quantization scheme is nicely controllable with respect to mean SD.

Finally, in Figure 4.9 we have plotted the mean SD as a function of

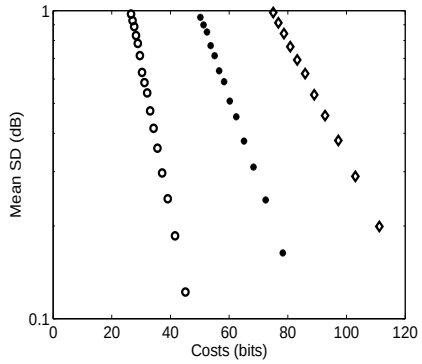


Figure 4.9: *Measured mean SD (in dB) versus the estimated cost (in bits) for the SLP parameter transmission. The circles, asterisks, and diamonds denote the first, second, and third order SLP system, respectively.*

cost (expressed in number of bits) for first-, second- and third-order SLP systems. These graphs were made based on the measured relationship (Table 4.2) between step size versus mean SD (using (4.26)) and step size versus entropy (using (4.27)). We have already seen that the predicted distortion D_p corresponds well to the measured distortion D_m, the latter being only slightly higher. As a consequence of the linear relations depicted in Figures 4.6 to 4.8, a linear relation between mean SD and costs evolves as well. From the graph, we infer that a mean SD of 1 dB would require about 4-5 bits per parameter. It is known that, for 1 dB mean SD, scalar quantization of RCs and LARs associated with mono LP costs 4 bits and 3.2 bits per parameter [68], respectively; and direct quantization of SLP coefficients has been proposed with 11-12 bits per coefficient [72], [73]. This means that the proposed quantization scheme offers significant savings with respect to direct quantization but it appears that it costs more than mono-LP.

We note, however, that the cost depends fundamentally on the statistics of the data. For example, the costs reported in [68] are for a much restricted set of speech data sampled at 8 kHz. For real audio data, the statistics may be quite different from those that we have used in our experiments. As it will be shown in Chapter 5, real audio data exhibit more structure (i.e., a more restricted set of normalized reflection and zero-lag correlation matrices), and the statistics are order-dependent. Furthermore, for real audio data, the necessary bit-rate may be reduced by an

extra lossless coding step, e.g., by differential encoding over frames. In that sense, the costs depicted in Figure 4.9 may be regarded as a worst-case estimate of cost in an actual application.

4.4 Alternative Parameterization Schemes

Several alternative sets of parameterizations of the normalized reflection matrices, such as EVD, and a combination of both SVD and EVD were also investigated as transmission parameters for SLP coding systems. For the detailed description of these parameterizations refer to Appendix D. For both these cases, the transmission parameters consist of the eigenvalue data. We have also investigated experimentally the quantization properties of the different parameterization schemes (see Appendix D). We noted that EVD-based schemes suffer from the following fundamental problems.

- The contractive property of the normalized reflection matrices is not guaranteed when quantizing the parameters, because eigenvalues less than unity do not necessarily imply that singular values are also less than unity.

- Eigenvalues may be real or complex or there could be a possible multiplicity, and therefore this has to be signaled in the bit-stream. On top of that, the character of the eigenvectors is associated with that of the eigenvalues. This implies that when the character of the eigenvalues changes due to the quantization, the eigenvectors have to be adapted as well. Though these problems can be solved, it makes this alternative scheme less straightforward and, therefore, less attractive.

- Finally, experiments with these alternative systems indicate that the nearly uncorrelated addition of quantization errors that we had for the proposed scheme does not hold (or hold to a lesser extent).

Based on these considerations we conclude that the parameterization based on the proposed approach is to be preferred over the mentioned alternatives.

4.5 Conclusions

We have addressed the problem of quantization and transmission of prediction matrices in SLP systems. A quantization scheme was developed and tested using randomly generated normalized reflection matrices and zero-lag correlation matrices. Sensitivities were measured in terms of SD while quantizing each of the parameters individually. Quantitative and qualitative relations between the mean SD, quantization step size, and number of bits were established experimentally. The results show that the proposed transformation nicely decouples the effects due to the different quantizers. In view of the results, it is also clear that the optimal bit distribution among the different parameters can be determined for a given mean SD. Thus, the proposed quantization scheme is a good candidate for SLP systems, where low encoding/decoding delay is desired. We note that the proposed quantization method applies to any two-channel LP system: the input could be stereo audio signals but could equally well represent two lines of an image.

Chapter 5

Quantization of Laguerre-based Stereo Linear Predictors

5.1 Introduction

A quantization scheme for SLP matrices was proposed and evaluated in Chapter 4. However, the scheme was not tested for the warped or Laguerre-based stereo linear prediction schemes. Furthermore, the sensitivity analysis and bit estimation was conducted using randomly generated normalized reflection and zero-lag correlation matrices but not for real stereo audio data as input. Therefore, this analysis is extended in this chapter for the Laguerre-based Stereo Pure Linear Prediction (LSPLP).

The objective of this chapter is to evaluate the LSPLP system using stereo audio data in order to gain insight into the required bit-rates for practical stereo audio coding applications. We tested the scheme for 44.1 kHz stereo audio material and we used the LSPLP only as a full-band spectral flattening system. However, in actual coding applications LSPLP may operate, for instance, on subbands and may also incorporate the stereo perceptual biasing rules (see Chapter 6), and thus the bit-rates indicated in this chapter pertain only to this particular stereo coding scenario.

This chapter is organized as follows. In Section 5.2, we describe the stereo audio database used in our experiments to study the performance of the proposed quantization scheme. Next in this section, the statistical distribution of transmission parameters associated with the normalized

reflection matrices and the zero-lag correlation matrix corresponding to the LSPLP scheme are presented. In Section 5.3, we discuss the evaluation of the LSPLP quantization scheme. We will first introduce a perceptually weighted spectral distortion (SD) measure and evaluate the performance of the LSPLP system with respect to this criterion. Finally we present our conclusions in Section 5.4.

5.2 Distribution of LSPLP Transmission Parameters

Like in mono speech coding applications, it is desired to quantize the prediction parameters with as little distortion as possible. Also, it is required that the synthesis filter remains stable after quantization of the prediction parameters. Since the stability of the LPLP scheme has already been proved [37] along with the stability of the symmetric SLP scheme (see Chapter 2), we can easily prove that LSPLP is also a stable scheme. The stability issue is especially considered as a prerequisite for developing a quantization strategy, since in the single-channel case the quantization schemes (for example, RC based or LSF based) are built upon the minimum-phase character of the LP analysis filter.

In this chapter we develop a quantization scheme for the proposed LSPLP with a transfer matrix given by

$$\boldsymbol{F}(z) \;=\; \boldsymbol{I} - z^{-1}\sum_{k=1}^{K}\boldsymbol{A}_{K,k}\frac{\sqrt{1-|\lambda|^2}}{1-\lambda z^{-1}}\left(\frac{-\lambda+z^{-1}}{1-\lambda z^{-1}}\right)^{k-1}, \qquad (5.1)$$

where $\boldsymbol{I}$ denotes the 2×2 identity matrix, K the prediction order of the LSPLP system, λ the warping factor (or the Laguerre parameter), and $\boldsymbol{A}_{K,k}$ the k^{th} LSPLP prediction matrix. The optimal prediction matrix $\boldsymbol{A}_{K,k}$ can be efficiently obtained by using the block-Levinson algorithm as described in Appendix A.

To test the quantization strategy for the LSPLP prediction matrices, we used nine stereo test signals (stereo, 44.1 kHz sampling frequency, 16 bits/sample, approximately 10 seconds long, and often employed in MPEG listening tests) called: Trumpet, Orchestra, Pop, Harpsichord, Castanets, Pitch Pipe, Bagpipe, Glockenspiel, and Plucked Strings. The order of the LSPLP was set to $K = 15$ and the Laguerre parameter was $\lambda = 0.756$. Our experiments indicated that an order of 15 is a good compromise between prediction orders and obtaining spectrally flat error

signals. This statement is also supported by [95], where an order of 16 is reported as the best compromise. LSPLP prediction matrices $\boldsymbol{A}_{K,k}$ were computed from 2048 sample Hanning windowed frames. The analysis was overlapping such that an analysis frame started after every 1024 samples. To tackle (nearly) identical left and right signals, and one-channel zero signals as input to the LSPLP, a regularization technique similar to the one described in Section 2.5.1 was followed. In fact, the regularization technique (2.80) was applied to the $\boldsymbol{Q}$ matrix of (3.17) that defines the optimal prediction matrices and, indirectly on the $\boldsymbol{P}$ of (3.17) since $\boldsymbol{P}$ and $\boldsymbol{Q}$ are related (see Section 3.3.1). The ϵ_{rel} was set to 10^{-2}.

An efficient strategy for quantization and transmission of the SLP prediction matrices was proposed in Chapter 4, where it was suggested to transmit (either forward or backward) normalized reflection matrices and the zero-lag correlation matrix, as they are the logical counterparts of mono LP transmission parameters. For the SLP scheme, the normalized reflection matrices are obtained from the block-Levinson algorithm as described in [66] and Appendix A. For the LSPLP scheme, the normalized reflection matrices associated with the minimum-phase matrix polynomial were computed using the block-Levinson algorithm followed by a simplified mapping as described in Section 3.5. As described in Section 4.2.1, these normalized reflection matrices $\boldsymbol{\xi}_k$ were then parameterized as $\sigma_1(k)$, $\sigma_2(k)$, $\kappa(k)$, and $\gamma(k)$ and finally quantized. The index k indicates the section index. We parameterized the zero-lag correlation matrix $\boldsymbol{C}_0$ by ρ and μ as described in Section 4.2.2, and quantized these parameters.

The histograms of the transmission parameters $\sigma_1(k)$, ρ, and μ in the LAR domain are plotted in Figure 5.1. The histograms of $\sigma_2(k)$ and $\kappa(k)$ are given in the Appendix E. From the plots it is obvious that there is a clear section index dependence for $\sigma_1(k)$, $\sigma_2(k)$, and $\kappa(k)$. Histograms of the transmission parameter $\gamma(k)$ are also plotted in the Appendix E. Only for $\gamma(k)$ there is no clear section index dependence. A slight hump in the distribution around 4-5 for ρ is contributed by the signals Glockenspiel and Castanets mainly, where there is a very high correlation coefficient of 0.99. The peak in the distribution around 2 for μ is due to the Pitch pipe signal, where there is an inter-channel level difference of about 17 dB.

From these histograms and with a given quantization step, the entropies associated with the parameters can be estimated. For each quantized parameter we obtained the associated entropies for different quan-

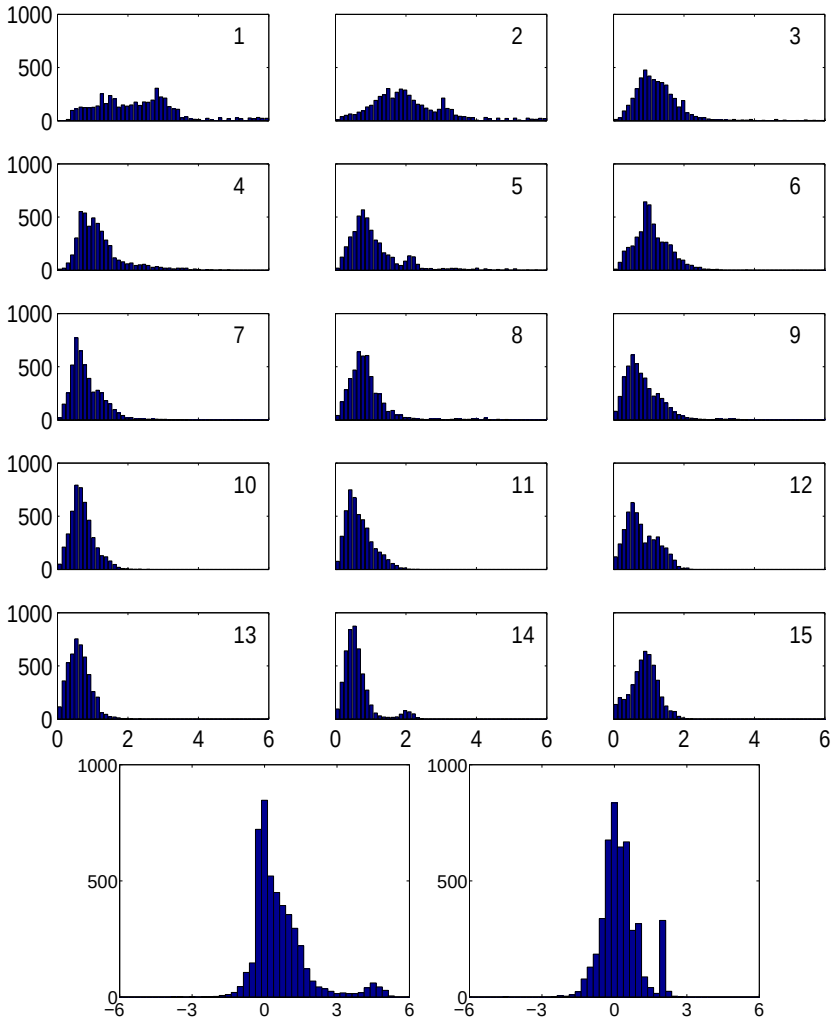


Figure 5.1: *The 15 plots shows the histograms of $\sigma_1(k)$, with k indicated in each plot. The bottom-left plot shows the histogram of ρ, and the bottom-right plot shows the histogram of μ. All the histograms are in the LAR domain.*

tization steps. Since each of the 15 normalized reflection matrices is described by 4 parameters and the zero-lag correlation matrix is described by 2 parameters, the total number of parameters is 62. Lets denote all the different parameters as p_i with $i = 1, 2, \ldots, 62$, where p_i is an element

Table 5.1: *Regression parameters $c(p_i)$ and $d(p_i)$ associated with $\boldsymbol{\xi}_k$.*

	c				d			
k	$\sigma_1(k)$	$\sigma_2(k)$	$\kappa(k)$	$\gamma(k)$	$\sigma_1(k)$	$\sigma_2(k)$	$\kappa(k)$	$\gamma(k)$
1	4.20	3.68	6.76	2.66	2.38	2.94	3.39	1.46
2	3.80	3.05	5.87	2.91	1.48	2.19	2.43	1.65
3	2.32	1.93	4.87	2.76	1.55	2.21	2.65	1.59
4	2.13	2.17	4.13	3.01	1.35	1.54	2.20	1.54
5	2.29	1.69	3.90	2.93	1.14	1.55	1.80	1.49
6	1.90	2.09	3.84	2.96	1.08	1.13	1.83	1.17
7	1.62	1.45	3.40	2.98	0.98	1.21	1.61	1.06
8	1.76	1.66	3.11	3.07	0.89	0.99	1.45	0.90
9	1.84	1.40	3.16	3.06	0.76	0.86	1.20	0.71
10	1.32	1.37	2.92	3.04	0.77	0.80	1.17	0.63
11	1.42	1.23	2.77	3.04	0.67	0.74	0.95	0.50
12	1.67	1.43	2.72	3.05	0.59	0.69	0.96	0.49
13	1.25	1.10	2.62	3.12	0.59	0.59	0.85	0.40
14	1.26	1.12	2.04	3.08	0.53	0.58	0.81	0.42
15	1.56	1.43	2.85	3.01	0.44	0.49	0.85	0.29

Table 5.2: *Regression parameters $c(p_i)$ and $d(p_i)$ associated with $\boldsymbol{C}_0$.*

c		d	
ρ	μ	ρ	μ
3.30	2.55	0.34	0.47

from the set

$$\{\mu, \rho, \sigma_1(k), \sigma_2(k), \kappa(k), \gamma(k) | k = 1, 2, \ldots, 15\}. \tag{5.2}$$

In Figure 5.2 it is shown how the step sizes $Q(p_i)$ for each quantized parameter p_i are related to its entropies measured in number of bits $b(p_i)$ for $k = 1, 5, 10$, and 15. This relation is modeled as

$$Q(p_i) = c(p_i)2^{-b(p_i)}, \tag{5.3}$$

and depicted by the straight lines in Figure 5.2. The fitted constants $c(p_i)$ are given in Tables 5.1 and 5.2. Note that $c(p_i)$ depends on p_i and thus on k, but is independent of the prediction order K.

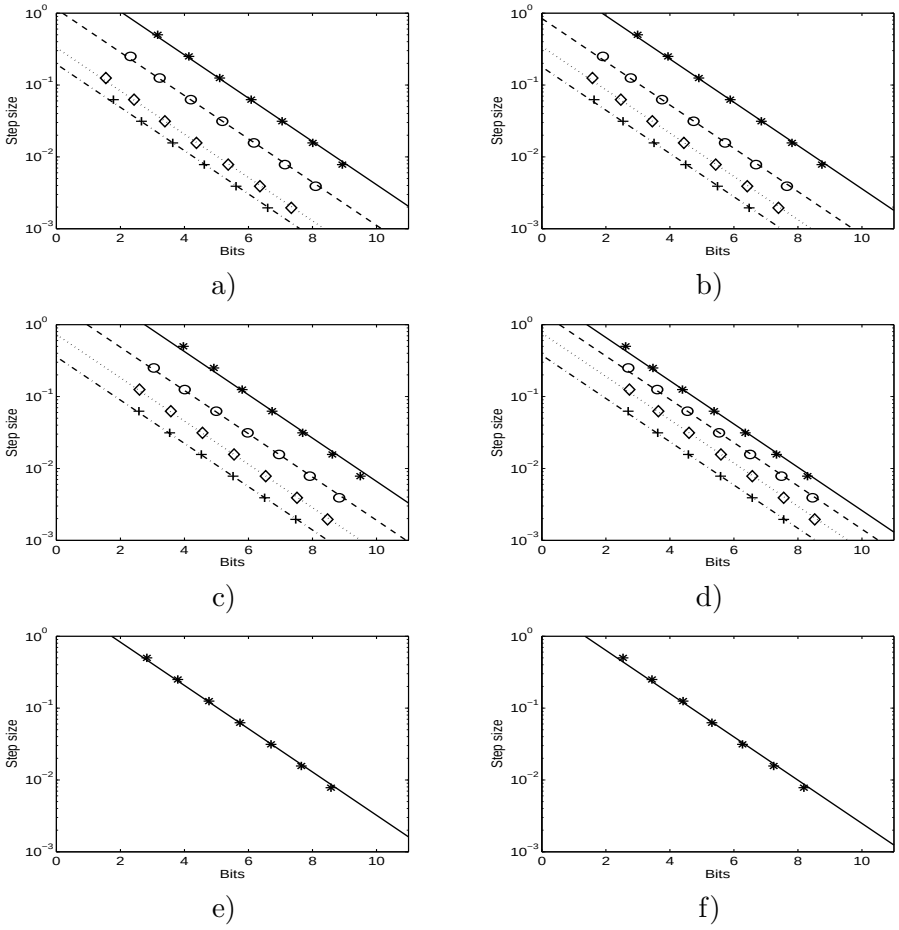


Figure 5.2: *Quantization step size $Q(p_i)$ for each quantized parameter versus the number of bits $b(p_i)$ (entropy of the quantized parameter) for a) $\sigma_1(k)$ b) $\sigma_2(k)$ c) $\kappa(k)$ d) $\gamma(k)$ e) ρ and f) μ. In the plots a), b), c), and d), the lines and symbols correspond to $k = 1$ (solid line, $*$), $k = 5$ (dashed line, $\circ$), $k = 10$ (dotted line, $\diamond$), and $k = 15$ (dash-dotted line, $+$). The step sizes are all in the LAR domain except for the step size of $\gamma(k)$ which is expressed in radians. For clarity, the lines and symbols corresponding to $k = 5$, $k = 10$, and $k = 15$ are shifted down by a factor of 2, 4, and 8, respectively.*

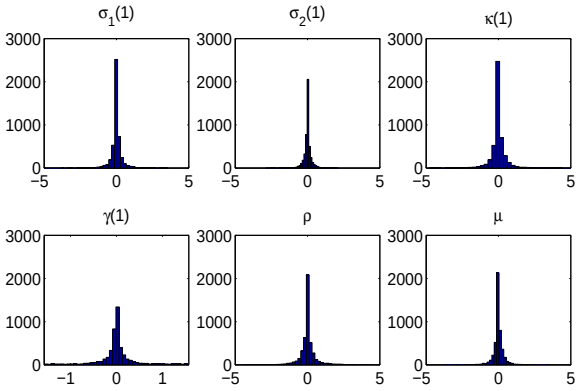


Figure 5.3: *Histograms of the LAR differences over frames for* $\sigma_1(1)$, $\sigma_2(1)$, $\kappa(1)$, $\gamma(1)$, ρ *and* μ.

The histograms of the difference across successive frames for the parameters associated with the first normalized reflection matrix and the zero-lag correlation matrix are plotted in Figure 5.3. As is obvious from these histograms, the distributions are much narrower. This means that substantial improvements can be achieved by differential coding of parameters over the frames. Significant gains in bit-rate are obtained for lower-order parameters and progressively lower gains towards higher order parameters. Gains of 52-60%, 20-40%, 20-40%, 11-30%, 25%, and 25% are obtained for $\sigma_1(k)$, $\sigma_2(k)$, $\kappa(k)$, $\gamma(k)$, ρ, and μ, respectively, for $k = 1, 2, \ldots, 15$. As an example, Figure 5.4 shows how the step sizes are related to the entropies of $\sigma_1(1)$ for differential and non-differential encoding. Note that for differential encoding, the quantization steps and the bits cannot be modeled by the relation (5.3).

In comparison with earlier results using random data discussed in Chapter 4, it is thus clear that the audio data exhibit more structure (that is, a more restricted range for the transmission parameters), that the statistics are section-dependent, and that the necessary bit-rate can be reduced by an extra lossless coding step, for example, by differential encoding over frames.

5.3 Sensitivity Analysis

Ideally, the quantization scheme should be judged by means of subjective listening tests, but there are reasons for not doing so. In short, these

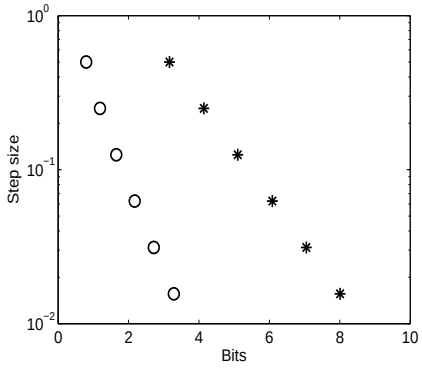


Figure 5.4: *Quantization steps (in LAR domain) versus the number of bits for quantized $\sigma_1(1)$ for non-differential (asterisks) and differential encoding (circles).*

are [68]:

- It is impossible to define a testing setup which is independent of a particular coding scheme.

- Formal listening tests are time consuming and expensive.

In fact, the evaluation is usually performed on an objective measure.

Selection of a proper distortion measure is the most important issue in the design of a quantization scheme. To evaluate the proposed quantization scheme, we make use of the objective measure analogous to the Spectral Distortion (SD) measure for the mono case defined in [68]. The average SD has been used extensively in the past to measure the performance of the single-channel LP parameter quantizers and an average SD of 1 dB along with limited number of outlier frames is considered important for spectral transparency.

In Section 4.3 a SD measure was proposed for SLP scheme. However, LSPLP is associated with a warped frequency scale that is rather close to the Bark-scale. For this reason, the SD measure of Section 4.3 is weighted in a perceptually relevant way. In particular, we take the weighted root mean square difference between the matrix norms of the synthesis filter frequency response matrices of the quantized and original (unquantized) prediction schemes; the matrix norm being defined as the larger of the

magnitudes of the two singular values. In formula, this reads as

$$D_l = \sqrt{\int_0^{F_s} w(f)[20\log_{10}(N_l(f)) - 20\log_{10}(\hat{N}_l(f))]^2 df}, \tag{5.4}$$

where F_s is the sampling frequency in Hz, and $N_l(f)$ and $\hat{N}_l(f)$ are the matrix norms of $\boldsymbol{F}_l(e^{j2\pi f/F_s})^{-1}$ and $\hat{\boldsymbol{F}}_l(e^{j2\pi f/F_s})^{-1}$, respectively, with $\boldsymbol{F}_l(z)$ and $\hat{\boldsymbol{F}}_l(z)$ the original and the quantized LSPLP transfer matrix of the l^{th} frame. The quantized transfer matrix $\hat{\boldsymbol{F}}(z)$ is defined as

$$\hat{\boldsymbol{F}}(z) = \boldsymbol{I} - z^{-1}\sum_{k=1}^{K} \hat{\boldsymbol{A}}_{K,k} \frac{\sqrt{1-|\lambda|^2}}{1-\lambda z^{-1}} \left(\frac{-\lambda + z^{-1}}{1-\lambda z^{-1}}\right)^{k-1}. \tag{5.5}$$

The weights $w(f)$ are defined according to

$$w(f) = \frac{\frac{1}{w_c(f)}}{\int_0^{F_s} \frac{1}{w_c(f)} df}, \tag{5.6}$$

so that

$$\int_0^{F_s} w(f) df = 1. \tag{5.7}$$

In (5.6), $w_c(f)$ denotes the Bark bandwidth. The spectral distortion is evaluated for all the frames in the test database and its average value is computed. This average value represents the distortion associated with the particular quantizer.

We have also tested the spectral distortion measure on a normal frequency scale and we found that it gave roughly 50% lower mean distortion than on a frequency warped scale. The reason for this deviation is because the modeling capability of the LSPLP is tuned more towards the low frequency side, which means that the sensitivity to parameter changes are more in the low frequency regions than in the high frequency regions. Since the distortion measured on the warped scale weights the low-frequency region more than on the normal scale, the distortion associated with the parameter changes are captured in greater detail in the low frequency side compared to the high frequency side.

In the subsequent sections, we will first consider individual parameter quantization (Section 5.3.1), continue with simultaneous parameter quantization (Section 5.3.2) and finally wrap up with a discussion of the performance of the proposed quantization scheme.

5.3.1 Single Parameter Quantization

The 15^{th}-order LSPLP system involves 62 parameters. In order to determine the sensitivity of the individually quantized parameters, we obtained per frame index the normalized reflection matrices $\boldsymbol{\xi}_k$ and a zero-lag correlation matrix $\boldsymbol{C}_0$ for the concatenation of nine MPEG test audio files. These matrices were then parameterized and each parameter was then quantized separately. For each quantized parameter, we determined the mean SD over all the frames for different quantization steps.

In Figure 5.5, the mean SD is plotted as a function of step size for each individually quantized parameters $\sigma_1(k)$, $\sigma_2(k)$, $\kappa(k)$, and $\gamma(k)$ for $k = 1, 5, 10$, and 15. Similarly the mean SD is plotted as a function of step size for each individually quantized parameter ρ and μ. We clearly observe a nearly linear relationship and it holds true for all the 62 parameters. Thus the distortion $D(p_i)$ as a function of step size $Q(p_i)$ can be modeled by

$$D(p_i) = d(p_i)Q(p_i), \tag{5.8}$$

for $i = 1, 2, \ldots, 62$ where i denotes a particular parameter. This relation is depicted in Figure 5.5 by the straight lines for each of the parameters. The values of d are given in Table 5.1 and 5.2. Note that there is a slight deviation from linearity for κ associated with all the orders. We also observe a slight deviation from linearity for σ_1 and σ_2 associated with the first-order. Nevertheless, for convenience, we still assume linearity for the rest of our analysis. Another point worthwhile to note is that, in contrast to $c(p_i)$, $d(p_i)$ is dependent on the prediction order K.

5.3.2 Simultaneous Parameter Quantization

If the distortions introduced by the individual quantization of parameters are decoupled, then the distortion should add up in an uncorrelated way to the total distortion while quantizing all the 62 parameters simultaneously. To check this, we used the following procedure. First, an equal mean SD denoted by D_{eq} is chosen for all 62 parameters of the individual quantization experiments. From the Table 5.1 we can determine the corresponding step size for each parameter. With these selected step sizes, we quantize all 62 parameters simultaneously. In case of decoupling we would predict the overall mean distortion D_p as $D_p = \sqrt{62}D_{eq}$.

Figure 5.6 displays the measured mean SD denoted by D_m obtained by quantizing all 62 parameters simultaneously, along with D_p. It can be observed that at 1 dB mean SD the measured data is 20% above the

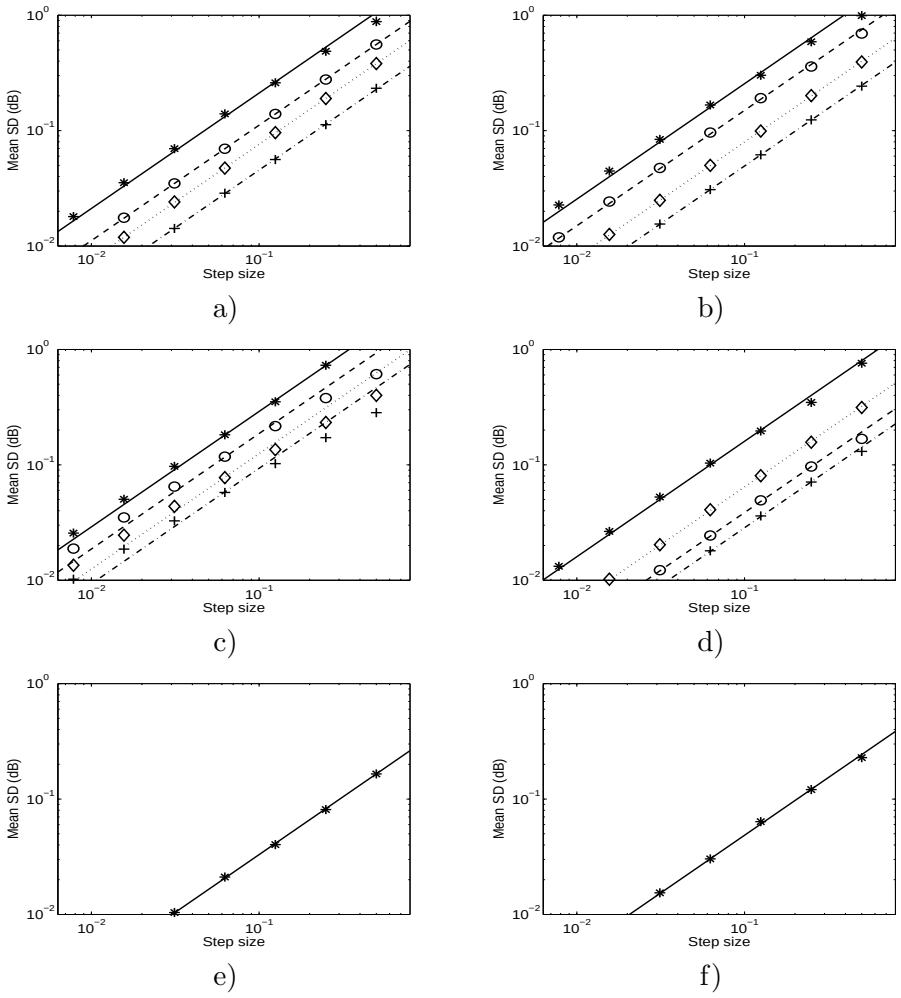


Figure 5.5: *Mean SD as a function of quantization step size for each individually quantized parameters a) $\sigma_1(k)$ b) $\sigma_2(k)$ c) $\kappa(k)$ d) $\gamma(k)$ e) ρ and f) μ. In the plots a), b), c), and d), the lines and symbols correspond to $k = 1$ (solid line, $*$), $k = 5$ (dashed line, $\circ$), $k = 10$ (dotted line, $\diamond$), and $k = 15$ (dash-dotted line, $+$). The step sizes are all in the LAR domain except for the step size of $\gamma(k)$ which is expressed in radians. For clarity, in plot (d) the data corresponding to $k = 5$ is shifted down by a factor of 4.*

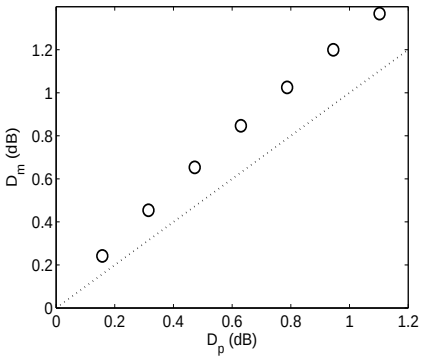


Figure 5.6: *Measured mean SD denoted by D_m versus predicted overall mean distortion D_p for a 15^{th}-order LSPLP system. The dashed line represents $D_m = D_p$.*

predicted data. Apparently, the distortions do not add up in a completely uncorrelated way, nevertheless, decoupling is achieved to a high extent. It also shows that the actual total distortion can be accurately predicted from the individual distortions, and consequently from the quantization step size or the number of bits. This predictable behavior is obviously attractive because the bit allocation per parameter as a function of overall bit-rate can be determined using simple rules.

5.3.3 Performance

In Figure 5.7 we have plotted the mean SD as a function of cost (expressed in number of bits) for the 15^{th}-order LSPLP systems. These graphs were made on the basis of the measured relations between the measured distortion and predicted distortion (Figure 5.6), the measured distortion and individual quantization steps (Figure 5.5), and the quantization steps and the estimated entropies (Figure 5.2). As a consequence of the nearly linear relations depicted in Figures 5.6, 5.5, and 5.2, a linear relation between mean SD and costs evolves as well. From the figure, we infer that a mean SD of 1 dB would require 300 bits/frame, that is about 5 bits per parameter. For this mean SD of 1 dB, there are 4.8% outliers within 2-4 dB distortion, and 0.08% outliers beyond 4 dB distortions.

In Figure 5.7 we have also plotted the mean SD as a function of bits for differential encoding over frames using the measured relationship between entropy of the differential parameters and quantization steps. We observe

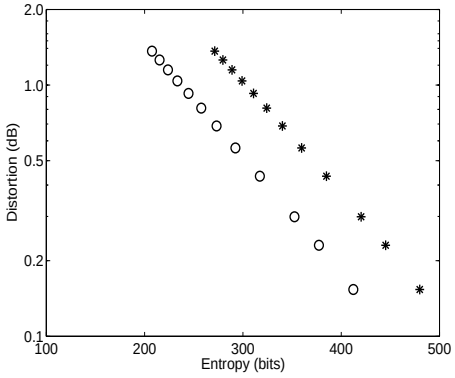


Figure 5.7: *Mean measured spectral distortion using scalar non-differential (asterisks) and differential encoding (circles) as a function of cost (in bits) for a* 15^{th}*-order LSPLP system.*

that a mean SD of 1 dB would require about 230 bits/frame, which is about 4 bits per parameter, i.e., a reduction of about 20% compared to non-differential encoding.

For any SD, the optimal bit allocation over the different parameters can be easily determined. It results in a non-uniform distribution of the bit allocation as a function of section index k, in particular, the normalized reflection matrices with lower section index require more bits than higher indexed ones. An example of this bit allocation is illustrated in Appendix E for a SD of 1 dB.

For an 8 kHz sampled speech database, it is known that for 1 dB mean SD, scalar quantization of RCs and LARs (with nonuniform bit allocation) associated with a mono LP costs 3.4 bits and 3.2 bits per parameter [68], respectively. For 44.1 kHz sampled audio, direct quantization of SLP coefficients has been proposed with 12 bits per coefficient [72]. Results were also reported for a 16^{th}-order mono WLP [95] using 44.1 kHz sampled audio database. There it was claimed that a mean SD of 0.3 dB, with 0.02% of frames with a SD greater than 2 dB, and no frames with a SD greater than 4 dB was necessary for transparency. It was achieved by an 81 bits/frame split multistage VQ of LSFs. However, a 60 bits/frame split multistage VQ of LSFs, delivering an average SD of 0.5 dB, 0.15% outliers within 2-4 dB distortion, and 0.02% outliers beyond 4 dB distortion was deemed to be sufficient. It is very important to take into account at this point that the above measures were made on the normal frequency scale. Thus, for the WLP scheme, the 60 bits/frame quanti-

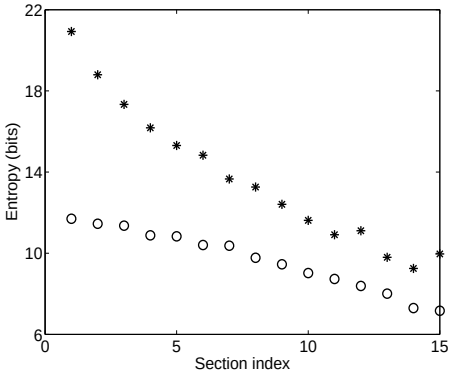


Figure 5.8: *Total bits required per section index k for the scalar quantization (asterisks) of the parameters $\sigma_1(k)$, $\sigma_2(k)$, and $\kappa(k)$ along with the bits required if VQ (circles) is applied on the sets $\{\sigma_1(k), \sigma_2(k), \kappa(k)\}$.*

zation scheme would roughly yield 1 dB distortion on the perceptually weighted spectral distortion introduced in Section 5.3. Therefore, we see that for 1 dB mean SD, multistage VQ of LSFs associated with mono WLP costs 5 bits per parameter. For WLP, other researchers have also used 6 bits per LSF coefficient for the scalar quantization of the LSF differences [96], and 7 bits per LAR coefficient [97]. Thus the prediction coefficient bit-rates are in general lower when LSPLP of order K is used than dual mono LPLP of order $2K$. We can conclude that 5 bits per parameter for the LSPLP scheme is promising.

In speech coding considerable work has been done to develop VQ for the LP parameters. Vector quantizers consider the entire set of LP parameters as an entity and allow for the direct minimization of the quantization distortion. Therefore, the vector quantizers result in smaller distortion than the scalar quantizers at any given bit-rate [98]. It is found from our experiments that there exists some correlation among the parameters $\{\sigma_1(k), \sigma_2(k), \kappa(k)\}$, which can be further exploited in a vector quantizer. The correlation decreases as k increases. Figure 5.8 shows the required number of bits per section index k for scalar quantization of the parameters $\sigma_1(k)$, $\sigma_2(k)$, and $\kappa(k)$ along with the bits required if VQ is applied on the set $\{\sigma_1(k), \sigma_2(k), \kappa(k)\}$. On average a 20% reduction in bits is observed again.

Thus, if we are either using a differential encoding over frames or VQ over the sets $\{\sigma_1(k), \sigma_2(k), \kappa(k)\}$, we require approximately 230 bits/frame

for 1 dB mean SD. In our test setup we used an update rate of 44100/1024 = 43.07 frames/s, implying an LSPLP parameter bit-rate of 10 kbit/s.

5.4 Conclusions

In this chapter, we have provided an indication of the necessary bit-rate for quantizing the LSPLP matrices. The quantization scheme was evaluated on a stereo audio database, using a perceptually weighted spectral distortion as an objective performance criterion. We have shown that LSPLP matrices can be quantized using similar techniques and measures as for SLP matrices. Sensitivities of the parameters were measured in terms of the spectral distortion measure while quantizing each of the parameters individually. Quantitative and qualitative relations between the mean SD, quantization step size, and number of bits were established experimentally, which are in the same order of magnitude as obtained in Chapter 4. Optimal bit allocation across all the parameters was presented. Differential encoding over frames or VQ revealed 20% reduction in bit-rates yielding a LSPLP bit-rate of 10 kbit/s for 1 dB mean SD. The performance of the proposed quantization scheme cannot be precisely benchmarked due to the absence of competitive stereo prediction matrix quantization schemes. However, from the reports on the quantization of mono WLP coefficients, we can conclude that we require fewer bits per parameter.

Chapter 6

Perceptually Biased Linear Prediction

6.1 Introduction

Psychoacoustic models have been used extensively within audio coding over the past decades. The underlying philosophy behind such lossy audio coding schemes is that the quantization errors generated by the audio coding algorithm are masked by the original signal [99]. When the quantization error signal is masked, the modified audio signal generated by the audio coder is perceptually indistinguishable from the original signal. To determine the allowed level of distortion, a psychoacoustic model (see e.g. [12]) is used.

Linear Prediction (LP) is a widely used technique in speech coding and has also been proposed for audio coding. By incorporating a frequency warping technique in LP, *Warped Linear Prediction* (WLP) can be obtained [36], [40]. A conceptually similar method is *Laguerre-based Pure Linear Prediction* (LPLP) [37]. In contrast to WLP, LPLP has the spectral whitening property just as conventional LP. It is well known that the spectral modeling capability for both LPLP and WLP can be tuned in a psycho-acoustically relevant way [38], making it suitable for lossy speech and audio coding. Furthermore, it has been argued that the optimal prediction coefficients should be derived from the masking threshold [42]. In this way, the spectral masking effects are accounted for in the prediction filter. Hermansky et al. also proposed perceptual LP in [100]. There the loudness curve was modeled in the context of speech recognition.

To account for the spectral masking effects in the prediction filter, a psychoacoustic model is required in order to calculate the masking curve. Next, from the masking curve the optimal prediction coefficients have to be calculated. There are several ways in which the masking curve can be converted to the normal equations, which define the optimal prediction filter coefficients. The whole method is rather demanding in the sense that a (complex) psychoacoustic model has to be run, a transformation from the masking curve to the normal equations has to be performed, and only thereafter the normal equations can be solved. The objective in this chapter is to replace the prediction filter based on the masking curve (explicitly obtained from a psychoacoustic model) by a prediction filter based on a *perceptually biased* solution. Thereby the computational complexity associated with the prediction coefficient optimization control box is significantly reduced. The effects of such a replacement are considered in this chapter. Throughout this chapter we will consider only LPLP as our prediction filter.

In summary, the perceptual-biasing method works as follows. The normal equations in the LPLP coefficient optimization involve the *autocorrelation function of the warped input signal* (ACFW). Interested readers are referred to Appendix B for the derivation of the ACFW. Instead of solving the normal equations based on this ACFW, we adapt the normal equations in the following way. We construct a perceptually biased ACFW, which is a sum of the windowed ACFW of the input signal and an ACFW corresponding to the threshold-in-quiet. This effectively means the following. The windowing of the input signal's ACFW corresponds to a spectral convolution. In LPLP (and in WLP), where we are working in the warped frequency domain, this implies that the convolution is actually done on a psycho-acoustically relevant frequency scale (e.g. Bark). Adding the windowed sequence to the ACFW of the threshold-in-quiet implies that we are biasing the solution to a filter with a transfer function that conforms to the threshold-in-quiet.

This chapter is divided into the following sections. In Section 6.2 we discuss the LPLP analysis filter controlled by a psychoacoustic model, followed by the proposed perceptually biased LPLP analysis filter. Section 6.3 gives the comparison of characteristics of these two variants of LPLP, including subjective listening tests to evaluate the performance of the proposed approach. In Section 6.4, we consider applications of perceptual biasing to situations beyond single-channel audio inputs. Specifically, we discuss its application in speech coding and its generalization to

multi-channel coding. Finally, in Section 6.5 we present our conclusions.

6.2 Optimization of LPLP Coefficients

6.2.1 The LPLP Filter

For completeness and clarity, we first review the conventional LPLP coefficient optimization based on a least-squared error criterion. Consider the LPLP analysis filter (LPLPA) as shown in Figure 6.1. It consists of a direct feed-through and a tapped filter line based on all-pass filters A preceded by a pre-filter C_0, with transfers

$$A(z) = \frac{z^{-1} - \lambda}{1 - z^{-1}\lambda} \text{ and } C_0(z) = \frac{z^{-1}\sqrt{1 - |\lambda|^2}}{1 - z^{-1}\lambda}, \tag{6.1}$$

with the Laguerre parameter (warping factor) $|\lambda| < 1$. The effect of λ is to shift the modeling capability to either the low ($\lambda < 0$) or high frequencies ($\lambda > 0$). Note that conventional LP is a special case of LPLP when $\lambda = 0$.

Consider now the optimal prediction coefficients $\hat{a}_l$ for a given input signal x, such that it minimizes the mean-squared error signal e, with e being the difference between the (windowed) input signal x and the predicted signal $\hat{x}$. The optimal prediction coefficients are given by the

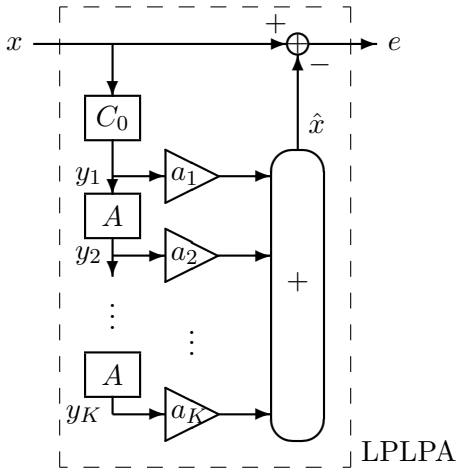


Figure 6.1: *Laguerre-based Pure Linear Prediction analysis filter (LPLPA).*

normal equations according to

$$\sum_{l=1}^{K} Q_{k,l}\hat{a}_l = P_k, \text{ for } k = 1, 2, \ldots, K \tag{6.2}$$

where

$$Q_{k,l} = \sum_n y_l(n)y_k(n) \text{ and } P_k = \sum_n x(n)y_k(n), \tag{6.3}$$

and y_l are the signals in front of the multipliers a_l (see Figure 6.1), with K being the order of the LPLP. We note that the set of equations in (6.2) can be written in the matrix form, where the Gram-matrix $\boldsymbol{Q}$ is a symmetric Toeplitz, positive semi-definite matrix.

It has been shown in Section 3.3 that $Q_{1,l}$ and P_l are related by

$$Q_{1,l} = \begin{cases} P_0 + 2K_1P_1, & \text{for } l = 1 \\ K_2P_{l-1} + K_1P_l, & \text{for } l = 2, 3, \ldots, K, \end{cases} \tag{6.4}$$

where P_0 is the power of the (windowed) input signal x. The constants K_1 and K_2 are given by

$$K_1 = \frac{\lambda}{\sqrt{1-\lambda^2}} \text{ and } K_2 = \frac{1}{\sqrt{1-\lambda^2}}. \tag{6.5}$$

This implies that one only needs to calculate the inner products P_k. The inner products $Q_{k,l}$ can be easily derived from P_k using (6.4), thus the complexity associated with the LPLP prediction coefficient optimization algorithm outlined in [37] can be reduced. Furthermore, from Appendix B and [35] we also note that $Q_{1,l}$ is the ACFW of x.

The LPLP synthesis filter (LPLPS) transfer is just the inverse of the LPLPA transfer. Assuming that the distortions introduced by the quantization of the error signal e can be modeled as additive white noise, the temporal and spectral structure of the noise at the decoder output is fully determined by the characteristics of the LPLPS. In the absence of any quantization, conventional LPLPS generates the spectral envelope of the input signal x from the spectrally flat error signal e generated by the LPLPA. For best performance in coding applications, it is then necessary that the magnitude response of the LPLPS corresponds to the masking threshold.

We note that for 44.1 kHz sampled material, the frequency resolution of LPLP approximates the frequency resolution of the human auditory system (critical band) if $\lambda = 0.756$ [37]. If desired, the spectral resonances in the LPLP synthesis filter responses can be smoothed by using a bandwidth-widening technique [41].

6.2.2 Old Approach: LPLP Controlled by a Psychoacoustic Model

This approach is similar to the one outlined in [42], where first the masking curve is calculated using a psychoacoustic model and next the LP synthesis filter best describing the masking curve is determined. Since we are only interested in a comparison, the particular psychoacoustic model that is used is not important. We decided to use the MPEG-1 Layer I-II (ISO/IEC 11172-3) psychoacoustic model [12] as well as one of the latest psychoacoustic models, in particular a *spectral integration* based psychoacoustic model [101], because it was shown to be more effective than the ISO MPEG model [69].

For the psychoacoustic model, we divide the input signal x into windowed frames of size 1024 (for the MPEG psychoacoustic model) or 2048 samples (for the spectral integration based psychoacoustic model). Then, the output of the psychoacoustic masking model is the frame-dependent masking threshold $M_{mm}(f)$. We will consistently use a subscript or superscript mm to denote signals and transfers associated with the masking model. Now we need to determine an LPLP analysis filter so that it has a transfer function $H_{mm}(e^{j\theta})$ satisfying

$$H_{mm}(e^{j\theta}) = \frac{1}{|M_{mm}(f)|}, \tag{6.6}$$

with $\theta = 2\pi f/F_s$ and F_s being the sampling frequency.

This can be done in several ways. One of the ways would be to start with an Inverse Discrete Fourier Transform (IDFT). The IDFT of $|M_{mm}(f)|^2$ over frequency (with a sampling according to the warping) for a given frame, yields the target ACFW $Q_{1,l}^{mm}$. Since from $Q_{k,l}^{mm}$, the P_k^{mm}s are known, the optimal LPLP coefficients a_l^{mm} can be obtained by solving the normal equations of (6.2). We note that the LPLP synthesis filter only models the shape of the masking curve, but does not incorporate the absolute level. To describe the full masking curve, an additional gain factor is required which represents the average (in dB) of the masked threshold over frequency.

6.2.3 New Approach: Perceptually Biased LPLP

Our approach is based on the following considerations. Firstly, we assume that the shape of the masking curve for a high-intensity input signal is independent of the level of the input. Though this assumption holds

only as a first-order approximation, it creates a natural link with LP, since the estimated optimal coefficients of an LP filter are completely independent of the input signal level. Secondly, we note that for low-intensity input signals, the threshold-in-quiet should be the dominant part of the response of the LP synthesis filter.

For the perceptually biased LPLP filter, we propose the following. From the input signal, we calculate P_l for $l = 0, 1, \cdots, K+1$. From the P_l, we can calculate the ACFW $Q_{1,l}$ for $l = 1, 2, \cdots, K+1$. Next, we construct the perceptually biased ACFW $Q_{1,l}^{pb}$ according to

$$Q_{1,l}^{pb} = Q_{1,l}^{tq} + \beta Q_{1,l} w(l), \text{ for } l = 1, 2, \ldots, K+1. \tag{6.7}$$

The superscript pb denotes the perceptually biased method, β is a constant to be calibrated, $w(n)$ is a window to include spectral smoothing as discussed in Section 6.1, and $Q_{1,l}^{tq}$ is the ACFW corresponding to the threshold-in-quiet. Note that the $Q_{1,l}^{tq}$ is just a constant offset term, therefore, can be calculated off-line (e.g., by using 2048 zero input samples in the old approach as described in Section 6.2.2) and subsequently stored.

We can construct the associated P_l^{pb} in the following way. We remap the sequence $Q_{1,l} w(l)$, using (6.4), to the sequence P_l^{map} for $l = 1, \ldots, K$, assuming $P_{K+1}^{map} = 0$. The perceptually biased P_l^{pb} is then given by

$$P_l^{pb} = P_l^{tq} + \beta P_l^{map}, \text{ for } l = 1, 2, \ldots, K, \tag{6.8}$$

where P_l^{tq} corresponds to the threshold-in-quiet, which can be calculated off-line and subsequently stored. Next, the optimal perceptually biased LPLP coefficients a_l^{pb} for $l = 1, 2, \ldots, K$ can be calculated using (6.2).

For the window, we selected a half-sided Hanning window, namely

$$w(l) = 0.5 \left[1 - \cos\left(\frac{2\pi\,(l+K)}{2K+1} \right) \right], \tag{6.9}$$

for $l = 1, 2, \ldots, K+1$. The window was chosen such that given the sampling frequency and the Laguerre parameter λ, the 3-dB bandwidth of the spectral image of the window corresponds to 1 Bark. This means that we only have to calibrate one modification parameter β. The parameter β can be chosen such that for input signals near the threshold-in-quiet level, $Q_{1,l}^{pb}$ is effectively determined by the $Q_{1,l}^{tq}$ only. The optimal β was found to be 4.0012×10^{-4}.

Similar to the old approach, an appropriate gain factor has to be included in the new approach that represents the average of the masked threshold over frequency. The gain factor G in dB is given by

$$G = 10 \log_{10} \left(P_0^{pb} - \sum_{l=1}^{K} a_l^{pb} P_l^{pb} \right), \tag{6.10}$$

where P_0^{pb} is the power associated with the perceptually biased ACFW $Q_{1,l}^{pb}$, given by

$$P_0^{pb} = Q_{1,1}^{pb} - 2K_1 P_1^{pb}. \tag{6.11}$$

6.3 Experimental Results and Discussion

In this section, we compare the ACFWs and the transfers of the associated LPLP synthesis filters for both the old and new approach. We also discuss the performance results as indicated by formal subjective listening tests in a typical LP based coding set-up. The order of the LPLP for both the approaches was set to $K = 40$ and the Laguerre parameter was $\lambda = 0.756$.

6.3.1 Autocorrelation Function of the Warped Input Signal

We used twelve test signals (mono, 44.1 kHz sampling, 16 bits/sample, approximately 10 seconds long, and often employed in MPEG listening

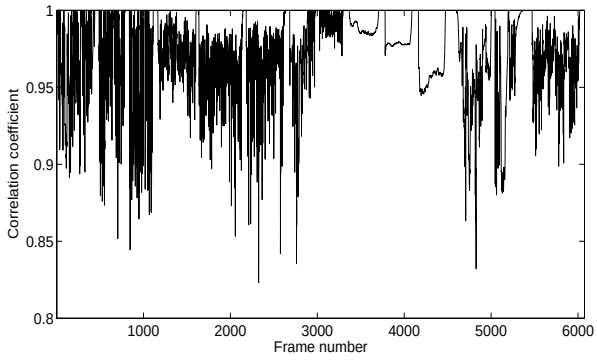


Figure 6.2: *Correlation coefficient between $Q_{1,l}^{mm}$ and $Q_{1,l}^{pb}$ as a function of frame index.*

tests) called: Suzanne Vega, German Male, English Female, Trumpet, Orchestra, Pop, Harpsichord, Castanets, Pitch Pipe, Bagpipe, Glockenspiel, and Plucked Strings. We calculated per frame the ACFW $Q_{1,l}^{mm}$ according to the old approach, where the spectral integration based psychoacoustic model was used and the ACFW $Q_{1,l}^{pb}$ according to the proposed perceptually biased method. In Figure 6.2, we have plotted the correlation coefficient between the sequences $Q_{1,l}^{mm}$ and $Q_{1,l}^{pb}$ as a function of frame index for a concatenation of the twelve test signals. We observe that the correlation coefficient is close to unity with only a small spread. This implies that there is a high degree of resemblance between these two sequences.

6.3.2 LPLP Synthesis Filter Response

Though the ACFWs of both approaches have a high degree of resemblance, this does not necessarily carry over to the LPLP filter, since it involves a highly nonlinear transformation. Therefore, it remains worthwhile to compare the spectra of the LPLP synthesis filters for both approaches.

The frequency responses of the LPLP synthesis filters for the two approaches are compared in Figure 6.3 to the masking thresholds obtained from the spectral integration based psychoacoustic model. The top plot shows the short-term spectrum of a frame from a rock excerpt *Smooth Criminal* [102], the masking threshold from the psychoacoustic model, and the response of the LPLP with the new and the old approach. The old approach was controlled by the spectral integration based psychoacoustic model. The excellent match between the solid and dashed lines demonstrates that LPLP can model the masking curve very accurately. However, there are clear differences between the spectra of the perceptually biased method and the psychoacoustic model based method. In particular, for the frequency range 5-15 kHz, the perceptually biased method gives a lower amplification. Also in the very low frequency range 0-300 Hz, the perceptually biased method is not able to follow the masking threshold. Similar observations hold for the classical music excerpt *Canon* [103] as shown in the bottom plot of Figure 6.3.

In Figure 6.4, the frequency responses of the LPLP synthesis filters for the two approaches were compared to the masking thresholds obtained from the MPEG psychoacoustic model. Now the old approach was controlled by the MPEG psychoacoustic model. The plots in Figure 6.4 for the pop excerpt *Eddie Rabbit* [104] and for the Trumpet shows that the

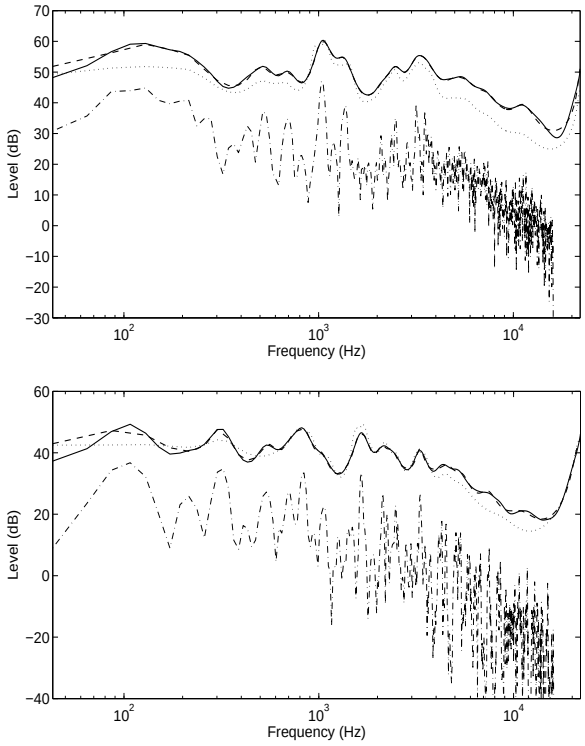


Figure 6.3: *Masking thresholds predicted by the spectral integration model (solid line) and synthesis filter transfers according to the old approach (dashed line) controlled by the spectral integration model, and the new approach (dotted line). The input spectrum (dash-dotted line) is from a short segment of Smooth Criminal (top) and Canon (bottom). For clarity, the input spectrum is shifted downward by 20 dB.*

trends are similar to Figure 6.3, except for the frequency range 5-15 kHz, where the perceptually biased method now gives a higher amplification.

The mean absolute difference between the LPLP synthesis filter response for the old approach (controlled by the spectral integration model) and the new approach per frame index is plotted in Figure 6.5 for the twelve test signals. Thus it can be seen that even though there is a high degree of resemblance between the ACFWs, the actual LPLP synthesis filters substantially differ from each other.

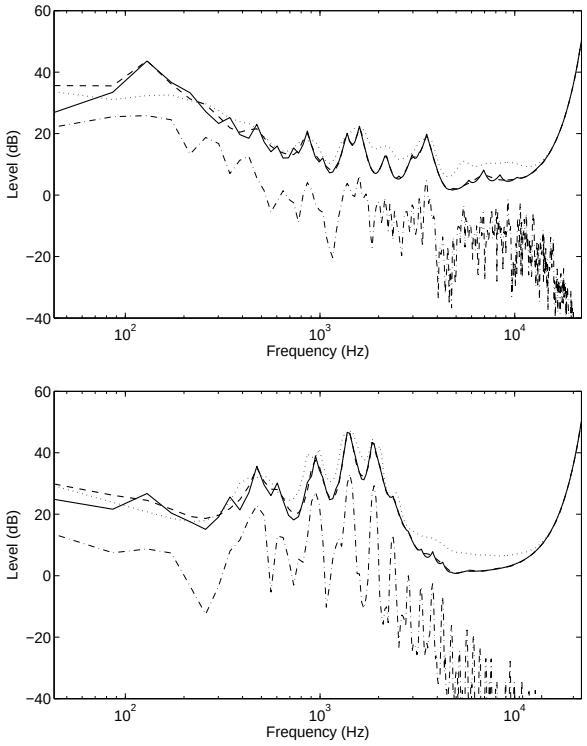


Figure 6.4: *Masking thresholds predicted by the MPEG model (solid line) and synthesis filter transfers according to the old approach (dashed line) controlled by the MPEG model, and the new approach (dotted line). The input spectrum (dash-dotted line) is from a short segment of Eddie Rabbit (top) and Trumpet (bottom). For clarity, the input spectrum is shifted downward by 20 dB.*

6.3.3 Perceptual Evaluation

We assessed the performance of the proposed perceptually biased approach in the context of a typical LP based coding set-up as shown in Figure 6.6, in the form of subjective listening tests. To evaluate the perceptually biased LPLP approach, two listening tests were conducted. The first test aims at establishing the performance of the proposed perceptually biased approach in terms of quality when compared to the old approach controlled by the spectral integration based psychoacoustic model. The second test compares the proposed perceptually biased

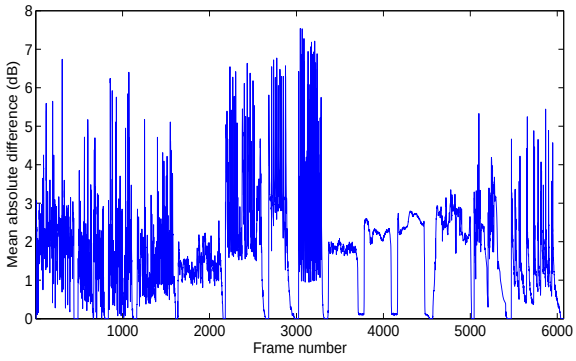


Figure 6.5: *Mean absolute difference between the LPLP synthesis filter response for the old approach (controlled by the spectral integration model) and the new approach per frame index.*

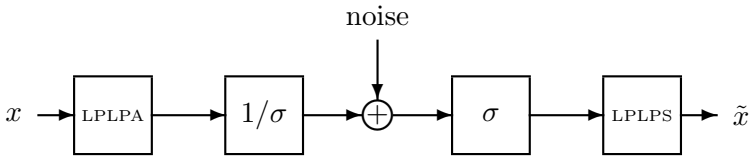


Figure 6.6: *Experimental set-up.*

approach with the old approach controlled by the MPEG psychoacoustic model [12]. The reason for using the MPEG psychoacoustic model is that the model provides a well-known reference and, because of its frequent application, it is still a de facto state-of-the-art model.

Listening Test I

In this test, the coefficients in the LPLPA box in Figure 6.6 were adapted according to the spectral integration model and the perceptually biased LPLP approach. LPLP coefficients for the two methods were computed from 46.4 ms (2048 samples) Hanning windowed frames. The analysis was overlapping such that an analysis frame started after every 5.8 ms interval, thus avoiding the need for interpolation of coefficients for our experimental purpose. The LPLP filter coefficients were not quantized and no bandwidth expansion techniques were applied. The scaling box $1/\sigma$ scales down the residual signal to unit variance and the inverse scaling σ is applied at the decoder. We generated excerpts $\tilde{x}$ for both approaches where we added the same realization of Gaussian white noise with unit

variance, i.e. 0 dB residual *Signal to Noise Ratio* (SNR) at the input of the decoder. For the new approach, we also generated five excerpts with the same realization of noise with SNRs of -1.5, 1.5, 3, 4.5, and 6 dB. The range of SNRs was chosen such that in all cases a clearly audible difference between $\tilde{x}$ and the input signal x was obtained.

In the following, we present the results obtained in the listening test. The test excerpts are mono, 44.1 kHz sampling frequency, and 16 bits per sample, which includes *Phenomenon* [79], *Eddie Rabbit, Smooth Criminal, Canon,* and the twelve MPEG test signals listed in Section 6.3.1

A subjective listening test based on a variant of the MUSHRA [105] methodology was conducted. Unlike the prescribed methodology for the MUSHRA test, no hidden reference and no anchors were presented to the listeners. We presented the seven versions of the noisy excerpt $\tilde{x}$ to the listeners in a "parallel"way, using an interactive tool as an interface to the listeners.

For each excerpt we presented the seven noisy versions in a random order. The original signal was available as a reference. Eight listeners from the Philips Research Laboratories Eindhoven (PRLE) participated in this test. Their ages ranged from 23-39 years. They all have a musical background, have normal hearing, and had undergone a training session before taking part in the experiment. The listeners were requested to rank the seven versions with respect to quality. Moreover, they were instructed to assign a score of 100 to the version with the best quality, 0 to the one with the worst quality, and scores of between 0 and 100 for the remaining five versions. This restriction enables the use of the full scale. It also reduces the difficulty in the ranking task for the listeners. It is important to realize that Mean Opinion Scores are irrelevant for our experiments, as we are only interested in the relative difference between the two approaches. The excerpts were presented through high-quality headphones (Beyer-Dynamic DT990 PRO) in a quiet listening room at PRLE.

Figure 6.7 shows the overall scores of the listening test, averaged across all listeners and excerpts. The asterisks represent the mean scores for the perceptually biased method with SNRs of -1.5, 0, 1.5, 3, 4.5, and 6 dB. The error bars depict the 95% confidence interval. The solid line represents a second-order polynomial fit to these data. We also plotted the mean score (circle) and confidence interval for the performance with the LPLP controlled by the spectral integration based psychoacoustic model. As can be seen from the dashed line, the quality obtained by the pro-

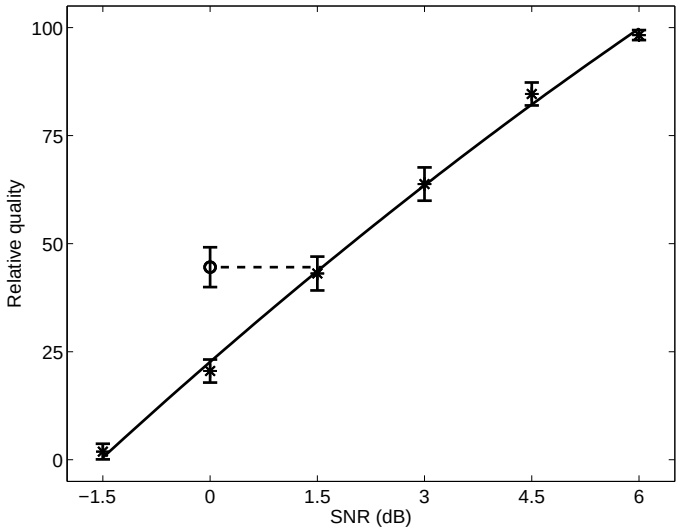


Figure 6.7: *Subjective test results averaged across all listeners and excerpts for the perceptually biased method (asterisks) and the spectral integration model based LPLP approach (circle). The dashed line indicates the difference in residual SNR at equal quality.*

posed perceptually biased LPLP with residual SNR of 1.5 dB is nearly equal to that of the LPLP controlled by the spectral integration model using 0 dB SNR. This means that, on average, the proposed perceptually biased method allows 1.5 dB less quantization noise than the old approach controlled by the spectral intergration based psychoacoustic model. Since the SNR of the residual signal is roughly related to the bit-rate of the residual signal quantizer by

$$SNR \approx 6b, \tag{6.12}$$

where SNR is in dB, and b is the number of bits; it implies that the proposed method requires approximately 0.25 bit/sample more than the old approach. Thus, in terms of bit-rate, the performance of the proposed method is slightly worse than the old approach. On the basis of the test results per excerpt, we also observed that the new approach performs equal to (or better than) the old approach in terms of bit-rate, for musical excerpts (pop, rock, heavy metal, and classical excerpts) and castanets. The largest performance degradation is found in speech (or speech-like) excerpts.

Listening Test II

In this test, the coefficients in the LPLPA box in Figure 6.6 were adapted according to the MPEG psychoacoustic model and the perceptually biased LPLP method. For the MPEG model we made use of the recommendations of MPEG Layer II, which support input frame lengths of 1024 samples. Therefore, for the old approach, the excerpts were segmented into windowed frames of 23.2 ms (1024 samples) with an update rate of 2.9 ms. We generated excerpts $\tilde{x}$ for the old approach where we added Gaussian white noise with 3 dB residual *Signal to Noise Ratio* (SNR) at the input of the decoder. For the proposed approach, we used the same six excerpts with SNRs of -1.5, 0, 1.5, 3, 4.5, and 6 dB as in the previous listening test.

The same set of test signals and exactly the same testing methodology outlined in the previous listening test was used to rank the seven noisy versions per excerpt. Nine listeners (four of them also participated in the previous test) from PRLE participated in this test. Their ages ranged from 25-37 years.

Figure 6.8 shows the overall scores of the listening test, averaged across all listeners and excerpts. The asterisks represent the mean scores for the perceptually biased method with SNRs of -1.5, 0, 1.5, 3, 4.5, and 6 dB. The error bars depict the 95% confidence interval and the solid line represents a second-order polynomial fit to these data. We also plotted the mean score (circle) and confidence interval for the performance with the LPLP controlled by the ISO MPEG psychoacoustic model. As can be seen from the dashed line, the quality obtained by the proposed perceptually biased LPLP with residual SNR of 0 dB is nearly equal to that of the LPLP controlled by the MPEG model using 3 dB SNR. This means that the proposed method allows 3 dB more quantization noise than the old approach when using the MPEG model. On basis of the results per excerpt, the test also revealed that the old approach controlled by the MPEG psychoacoustic model using 3 dB SNR performs better than the new approach using 0 dB SNR for Glockenspiel, Bag pipe, Pitch pipe, Trumpet, and English Female. The new approach performs much better for musical excerpts (pop, rock, heavy metal, and classical excerpts) and castanets. At equal SNRs the new approach outperforms the old approach, especially for the speech signals.

Since in this listening test the frame sizes in the old and new approach were different, one may argue this to be a cause for the difference in performance. So, we have also generated audio files using small frame sizes

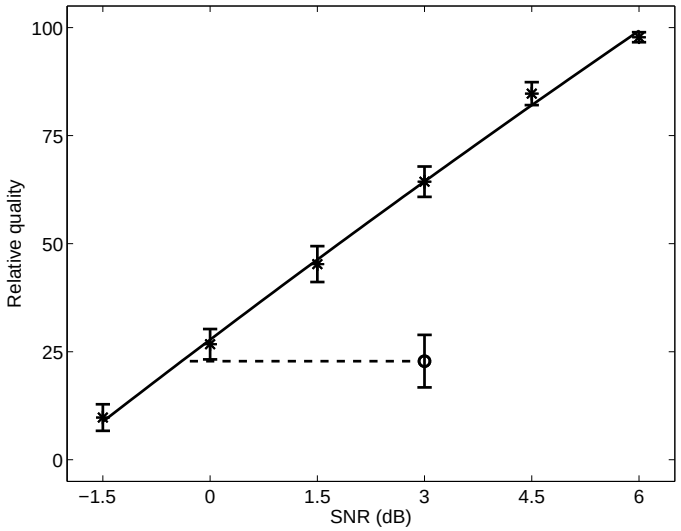


Figure 6.8: *Subjective test results averaged across all listeners and excerpts for the perceptually biased method (asterisks) and the MPEG psychoacoustic model based LPLP approach (circle). The dashed line indicates the difference in residual SNR at equal quality.*

(1024 samples) for the perceptually biased method. Informal listening revealed that there was hardly any difference compared to the audio files generated by using longer frame sizes (2048 samples).

6.3.4 Discussion

The comparison of the two approaches reveals that the old and the new approach generate different LPLP filters. Though the perceptually biased LPLP method does generate a filter that deviates substantially from the masking curve, listening tests using a typical coding set-up reveal that it is still a viable method for audio coding. More specifically, the proposed perceptually biased approach allows 1.5 dB less quantization noise (i.e., roughly requires an additional 0.25 bits/sample) when compared to the old approach controlled by the spectral integration based psychoacoustic model. However, when the old approach is controlled by the MPEG model, the proposed approach allows 3 dB more quantization noise (i.e., roughly gives a reduction of 0.5 bits/sample). Therefore, the performance in terms of the bit rate and quality trade-off of the proposed method is

in the range of that obtained using existing psychoacoustic models.

6.4 Other Applications of Perceptual Biasing

6.4.1 Application to Speech Coding

In order to maximize the speech quality, speech coders (e.g., CELP or ACELP) minimize the mean squared error (noise) between the input speech and synthesized speech in a perceptually weighted domain. Conventionally, for narrowband (8 kHz sampled) speech, a *Perceptual Weighting* (PW) filter $W(z)$ given by

$$W^{'}(z) = \frac{A^{'}(z/\gamma_1)}{A^{'}(z/\gamma_2)},\ 0 < \gamma_2 < \gamma_1 \leq 1 \tag{6.13}$$

is applied to the error signal in the encoder to optimize the codebook search through analysis-by-synthesis procedure. Here $A^{'}(z)$ is the LP filter, where $\gamma_1 = 0.9$ and $\gamma_2 = 0.6$ are the factors that control the amount of perceptual weighting. The spectral shape of the noise tends towards $1/W^{'}(z)$. However, $1/W^{'}(z)$ is only a crude approximation of the auditory masking properties as it was reported that the PW filter controlled by a psychoacoustic model significantly increases the quality [106]. Even though several complexity reduction steps were suggested in [106], still an explicit psychoacoustic model needs to be run, which is not preferred in applications that demand low computational complexity. Therefore, the perceptual biasing method may form a nice alternative.

For wideband (16 kHz sampled) speech, it is known that the conventional PW filter in (6.13) is not suitable when there is a pronounced spectral tilt [32]. To overcome such problems a novel PW filter was proposed for the AMR-WB speech coder that computes the LP filter $A^{''}(z)$ based on a speech filtered through a pre-emphasis filter, $P^{''}(z) = 1-\mu z^{-1}$, with $\mu = 0.68$. The new PW filter is given by

$$W^{''}(z) = \frac{A^{''}(z/\gamma_1)}{1-\gamma_2 z^{-1}},\ 0 < \gamma_2 < \gamma_1 \leq 1. \tag{6.14}$$

If γ_2 in (6.14) is set equal to μ, then the quantization noise is shaped by a filter whose transfer function is $1/A^{''}(z/\gamma_1)$.

Figure 6.9 shows the noise shaped by the conventional PW filter and new PW filter (of AMR-WB). We also included in the plot the noise shaped according to the perceptually biased LPLP for unvoiced speech

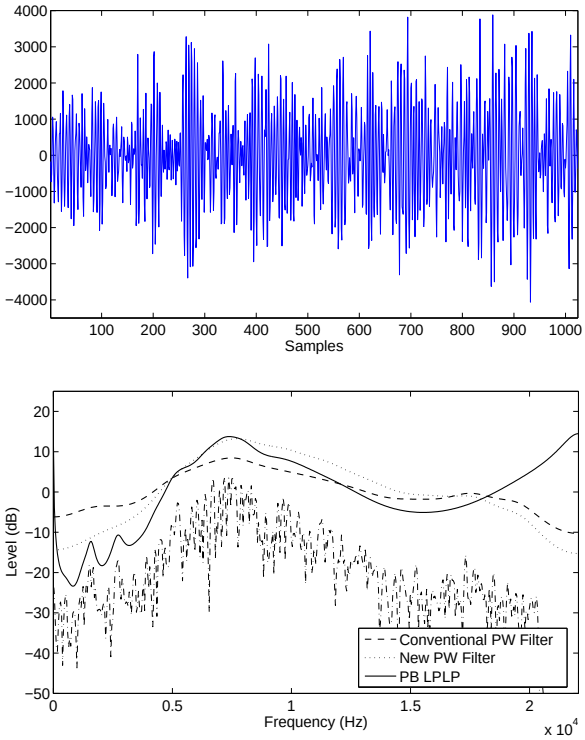


Figure 6.9: *An example of a time domain unvoiced signal (top). Spectrum of the signal in (top) along with the quantization noise envelope after shaping by a conventional PW filter, new PW filter (of AMR-WB), and perceptually biased LPLP (PB LPLP). The responses are plotted for a 0 dB mean level and the spectrum of the signal is shifted down for clarity.*

segment sampled at 44.1 kHz. The order of the prediction filter used in all the three cases was set to 20. As reported in [32], we observe that the new PW filter shapes the noise at low frequencies (which is of relevance for wideband speech coding) in a better way, whereas the conventional filter is not so accurate at low frequencies. We also note that the proposed approach not only shapes the coding noise much better than the method proposed in [32], but it also includes the threshold-in-quite at very low and high frequencies. Therefore, it is also worthwhile to consider the proposed method as an alternative candidate for wideband speech coding.

6.4.2 Binaural Perceptually Biased LPLP

The proposed perceptually biased LPLP scheme was developed and tested for single-channel case only. However, it is known that for the stereo case, the masked threshold depends on the inter-aural signal properties of the signal (masker) and the quantization noise (maskee) [44]. Therefore, when coding stereo (or multi-channel) audio signals, the perceptual model needs to consider this inter-aural dependence of the masking threshold. Since most of the important inter-aural cues (namely IID, ITD, and IC) are already captured in the Laguerre-based Stereo PLP (LSPLP) prediction matrices, it maybe worthwhile to generalize the perceptual biasing rules given in (6.7) and (6.8) to the LSPLP case as well. The generalization to binaural perceptually biased LPLP seems to be straightforward and is given by

$$\boldsymbol{Q}_{1,l}^{pb} = Q_{1,l}^{tq}\boldsymbol{I} + \beta \boldsymbol{Q}_{1,l} w(l), \text{ for } l = 1, 2, \ldots, K+1 \tag{6.15}$$

to construct a 2×2 perceptually biased ACFW matrix sequence; where $\boldsymbol{Q}_{1,l}$ is the 2×2 ACFW matrix sequence for the input signal vector $\boldsymbol{x}$, $\boldsymbol{I}$ is the 2×2 identity matrix. Like before $w(l)$ is the windowing sequence and β is the calibration constant. The diagonal elements of $Q_{1,l}^{tq}\boldsymbol{I}$ represent the ACFW corresponding to the threshold-in-quiet for the two channels. The associated $\boldsymbol{P}_l^{pb}$ sequence can be obtained in a way similar to (6.8), but only extending them to matrices in a way exactly similar to the one described in (6.15). Next, the optimal perceptually biased LSPLP prediction matrices are obtained by solving the block-Levinson algorithm (see Appendix A).

Initial tests seem to suggest that the proposed approach is a possible way to model the binaural masking threshold. Since there exists no such binaural psychoacoustic model that is able to predict the masking thresholds for the different channels taking into consideration the inter-aural dependencies, our initial test results cannot be easily confirmed. Nevertheless, the good news is that the proposed approach for LSPLP gives comparable results when compared to the perceptually biased LPLP applied independently on the two input channels. The order of the LSPLP was set to half of the order of mono perceptually biased LPLP. There is reason to believe that further refinements in the binaural perceptually biased LPLP may lead to a gain over perceptually biased LPLP applied independently on two input channels. More extensive tests need to be done to address this issue.

6.5 Conclusion

In this chapter we presented a novel approach for perceptual audio coding based on Laguerre-based Pure Linear Prediction. In contrast to standard linear prediction based methods, we do not aim at flattening the spectrum of the input signal; neither is the method aiming at directly modeling the masking curve. Instead, the method shapes the input spectrum according to a perceptually biased method by windowing the autocorrelation function of the warped input signal, and adding an offset corresponding to a threshold-in-quiet characteristic. This approach significantly reduces the computational complexity since it is no longer required to execute a separate psychoacoustic model.

Formal subjective listening tests reveal that the proposed method is a promising approach for audio coding. There is a quality reduction when compared to LPLP controlled by one of the latest spectral integration based psychoacoustic models. However, if the LPLP is controlled by the more commonly used MPEG psychoacoustic model, the new approach provides a better quality for the same residual SNR. Therefore, we conclude that the performance of the newly proposed approach in terms of bit-rate and quality trade-off is in the same range as that of existing psychoacoustic models.

From [106], we know that the performance of speech coders can be improved by appropriate design of the weighting filter. However, this comes typically at the price of a significant increase in complexity. The perceptual biasing approach proposed in this chapter may therefore form a nice alternative yielding an improved design of the weighting filter with low additional complexity. The proposed biasing rules can also be extended to the stereo (or multi-channel) case. Initial tests seem to suggest that the proposed approach is a possible way to model binaural masking threshold. Further refinements and tests still needs to be done.

We note that although the results reported here were established for the LPLP scheme, they carry over to the case of WLP. The only relevant difference is that special care has to be taken in view of the fact that the whitening property inherent in the LPLP does not carry over to the WLP case. Lastly, we mention that the perceptual-biasing rules are very simple. Presumably, refinements are possible. Thus, the new approach opens up a new paradigm to psycho-acousticians, who can incorporate further refinements in our proposed system to develop dedicated psychoacoustic rules for LP based coders.

Chapter 7

Epilogue

In this thesis we have addressed several fundamental problems associated with LP-based audio coding schemes. Several complexity reduction proposals were made that could be easily extended to stereo and multi-channel linear predictive coders. The most important contributions of this thesis can be summarized as follows.

- Generalization of the one-channel LP to SLP (and multi-channel prediction) along with the normalized block-lattice filtering implementation, such that the major properties of the one-channel LP are maintained in stereo and multi-channel linear prediction.

- A low complexity and memory efficient implementation of the Laguerre-based Pure Linear Prediction scheme. The algorithm can be extended to SLP (and multi-channel prediction), thus making it possible to incorporate Laguerre filters in SLP.

- A simple and efficient strategy to quantize the prediction matrices was proposed for SLP. This technique can also be extended to multi-channel prediction.

- A low complexity perceptual biasing rule to obtain the masking threshold is presented, such that the spectral masking effects are taken into account in the prediction filter, thus avoiding the need of a separate (complex) psychoacoustic model for irrelevancy reduction. The perceptual biasing rules can be extended to SLP to form a stereo irrelevancy reduction stage. The extension to the multi-channel case needs to be investigated.

Naturally after having all these insights and tools that we developed in this thesis, we would like to discuss in this chapter the impact of our current research. We would like to classify the impact into two separate categories.

1. Directions for further scientific (academic) research;
2. Avenues for applications.

Obviously, this distinction is not strictly disjoint. Nevertheless, we made this distinction for clarity. The bottom line is to identify the key steps that would make the best use of the knowledge that we have gained to build up a coding solution that offers low-delay, low-complexity, error resilience, and scalability, with almost state of the art audio quality.

Let us now elaborate on the key issues that are of scientific interest.

1. (a) Concept of stereo and multi-channel LSFs. Chapter 2 created the basis for the stereo and multi-channel LSF concept. The outcome is possible because our proposed prediction scheme is a true generalization of a one-channel LP scheme. On the basis of this chapter, an attempt towards formulating the stereo LSFs was made in [81]. However, the proposal suffered from two drawbacks. First, in contrast to our expectation, the proposed LSF-like quantization scheme performed worse than the quantization scheme proposed in this thesis. This result is surprising, as intuitively (just like in mono LP) one would expect scalar quantization performance of the stereo LSFs to be at least equal (or better) than the scalar quantization performance of the normalized reflection matrices. Second, the stereo LSF quantization scheme cannot be extended to the multi-channel case. Thus, the above observations suggest that the stereo LSF formulation was not a logical extension of mono LSFs and the right stereo LSFs still need to be formulated.

1. (b) Stereo/Multi-channel perception models based on LP. Parametric stereo and spatial audio coding extracts and quantizes the spatial cues based on statistical data [107],[108]. In this thesis we have shown that simple mono LP biasing rules can be used to build up a psychoacoustic model, which is in fact also based on statistical data. When coding stereo and multi-channel audio signals, the perceptual model needs to consider this inter-aural dependence of the masked threshold. This dependence is often described by means of the Binaural Masking Level Difference (BMLD) [44].

Currently there exists no such binaural psychoacoustic model that is able to predict the masking curves for the different channels taking into consideration the inter-aural dependencies. However, our initial experiments suggest that it is possible to build up such psychoacoustic models by incorporating perceptual biasing rules into SLP (or multi-channel prediction). Furthermore, we have the possibility to build up such a model with very low complexity.

1. **(c) Strategy for residual signal quantization.** One of the major challenges in LP based speech/audio coders is finding a representation for the residual signal that allows efficient and effective quantization. For speech coding, efficient residual signal quantization schemes exist. Linear predictive speech coders typically use pulse coding (such as Regular Pulse Excitation (RPE) [109] or Multi Pulse Excitation (MPE)) to code the error signals [18],[83]. Preliminary investigations into audio coders using LPLP and RPE [64] showed that, in general, this combination yields good results. However, audio signals containing clear tonal components can give problems. RPE is a technique that originally targeted narrow-band speech coding. Therefore, two enhancements were introduced to make RPE more suitable for general audio coding. These enhancements are improved optimization of pulse sequences [110] and extra pulses [111] at the expense of marginally increased computational complexity.

   Another kind of parametric audio coders, called SinuSoidal Coder (SSC) [112] decomposes the audio signal into transients, sinusoids and noise, and describes each component by a set of parameters. It was shown in [112] that SSC attains fair to high audio quality for most audio material. However, the quality is far from transparent for audio signals that are not very well defined in terms of tonal or noise components. The results of SSC and LPLP with RPE indicated that LPLP with RPE is very suitable for those signals that are problematic for SSC and vice versa. This led to the idea [64],[113] to combine SSC with LPLP and RPE. In [64] the SSC block is placed before the LPLP and RPE blocks and in [113] the LPLP block is placed before the SSC and RPE blocks. The scheme of [113] can be seen in Figure 7.1. It uses both RPE and the main block from SSC, which is Sinusoidal Extraction (SE), to code the error signal e from LPLP. The SE block consists of two blocks, which are the Sinusoidal Analyzer (SiA) and the Sinusoidal

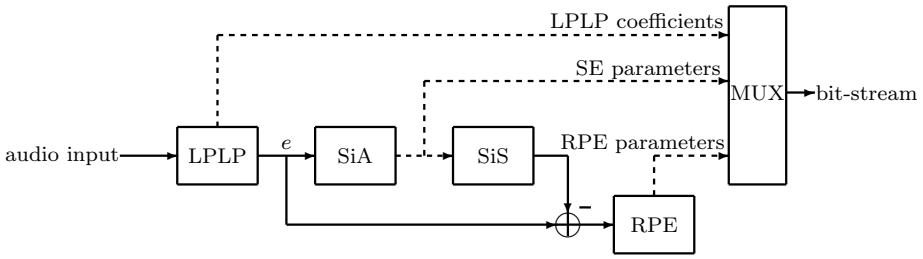


Figure 7.1: *Coding scheme proposed in [113]. It uses both RPE and SE to code the error signal from LPLP.*

Synthesizer (SiS). The SiA extracts the SE parameters from the error signal and the SiS creates a sinusoidal excitation based on the SE parameters. The Multiplexer (MUX) constructs the bit stream from the prediction coefficients, the SE parameters and the RPE parameters. The results of this coder showed that excellent quality could be obtained at relatively low bit-rates. In addition, the audio sounds natural, but also sometimes a bit noisy.

The idea of using both SE and RPE can be used to quantize the error signals from LSPLP. However, for such a residual coding scheme the lower limit of the system coding delay will be dictated by the buffering of samples in the SE block. Therefore, to achieve a low-delay coding solution this issue needs to be solved. Furthermore, the computational complexity will be dictated by the RPE block and thus needs to be optimized. The complexity can be reduced by simplifying the pulse optimization, for example, by changing the combined optimization to a sequential optimization and by making the quantization boundary dependent on the boundary of the previous frame.

1. **(d) Relationship between spatial cues and stereo prediction parameters.** Based on the statistics obtained from the stereo (or multi-channel) perceptual biasing rules, the relationship between the binaural (or spatial) cues and the stereo prediction parameters needs to be explored further. The reason why we think there is an existence of such a relationship is the following. In Parametric Stereo [61], the binaural cues are extracted per Bark band, and then quantized and transmitted as side information. In this thesis we propose to utilize our SLP scheme on a Bark scale by

incorporating Laguerre filters, and we quantize and transmit the prediction matrices, zero-lag correlation matrix, and rotation angle. Thus a certain amount of spatial information is captured in these parameters and there should be an intimate relation between these transmitted parameters and spatial cues. If a one-to-one relationship can be formulated between the transmission parameters and spatial cues, then it would also be possible to manipulate these cues. Such a manipulation would be of interest, for example, in gaming applications.

After discussing directions for scientific research, let us now discuss whether the outcome of all the scientific activities will be fruitful for application scenarios.

2. (a) Improvements of existing speech/audio coding systems.

- Perceptual biasing rules can be used to improve the weighting filters in wideband speech coders (see Chapter 6). Compared to the approach used in AMR-WB [32], we expect the perceptual biasing rule to perform better with a slight increase in computational complexity. On the other hand, we offer much reduced complexity compared to directly using a psychoacoustic model [106] for a comparable performance.
- It was argued that for LP-based audio coding, the optimal prediction parameters should be derived from the masking threshold [42] obtained from the (complex) psychoacoustic model. In this way, the spectral masking effects are accounted for in the prediction filter. Simple perceptual biasing rules (see Chapter 6) offer a low complexity alternative.
- The strategy adopted for prediction matrix quantization can be utilized to quantize the prediction matrices appearing in lossless coding schemes (like in [72],[73]); thus avoiding the need for backward adaptation.

2. (b) Stereo/Multi-channel low-delay audio coders. The motivation for utilizing LP for audio coding is because it is able to offer low-delay coding solution. Currently the state-of-the-art low-delay audio coding solution in the market is the Ultra Low Delay (ULD) coder [22]. The ULD coder has a delay of 6 ms and obtains a perceptual quality comparable to MP3 at 128 kbit/s stereo for bit-rates of 80 kbit/s per channel [22].

The ULD encoder is modularized into two separate stages that perform irrelevancy and redundancy reduction [114]. The irrelevancy reduction stage uses the psycho-acoustically controlled prefilter concept [42] to analyze the input signal in order to obtain the perceptually irrelevant parts of the signal. The redundancy reduction stage following the irrelevancy stage consists of a backward adaptive lossless coding stage. However, the ULD coder has two drawbacks. First, the separate psychoacoustic model adds up to the computational complexity [28]. Second, the ULD coder operates on two channels independently and does not take into account the cross-channel irrelevancies and redundancies.

Thus, as a solution to the above problems, we can offer:

- A computationally efficient stereo perceptually biased prediction scheme to remove both the inter- and intra-channel irrelevancies. This would free the resources required for executing a psychoacoustic model on two independent channels.
- Inter- and intra-channel redundancy removal using a lossless SLP scheme that incorporates the quantization strategy proposed in Chapter 4. Such a scheme is advantageous also because it avoids backward adaptation.

2. (c) Envisioned systems.

- Audio coding using speech technology: The structural nicety of the SLP scheme offers generalized extension from the mono-LP to the stereo and multi-channel prediction; offering us the possibility to code audio using speech technology. Thus, it was felt one of the possible options could be to extend this idea to our proposed SLP scheme to quantize the error signals. The important issue that is not discussed in this thesis is the quantization of the error signals. However, this issue was handled during the course of this thesis by a graduate student and it can be found in [84]. The basic philosophy and motivation behind the quantization is described before (**1. (c)**).

  It was described in [84] how both SE and RPE can be used to quantize the error signals, making SLP very similar to a mono predictive speech coder; except that SE has replaced the Long Term Predictor that is used in speech coding. It is important to note that the error signals in SLP were the

main and the side signal obtained after the rotation. It was found that the important components of the side signal are limited to frequencies less than ≈4 kHz. It was proposed to use SE and RPE to quantize the main signal. For the side signal it was proposed to use SE and RPE to quantize the lower frequency part (less than 4 kHz), and to use a synthetic signal to reconstruct the remaining part of the side signal [115]. For (nearly) identical left and right signals, it was decided to only quantize the main signal, thus discarding the side signal completely.

Formal listening tests showed that SLP at 80 kbit/s stereo is equivalent to HE-AAC-v2 at 48 kbit/s stereo. While HE-AAC-v2 mainly suffered from stereo image problems, the coding artifacts in SLP were mainly attributed to the quantization noise. The proposed SLP scheme lost out in the listening test for two reasons. Firstly, in the SLP coder there was no transient detection and simulation algorithm or something similar. Furthermore, the weighting filters used in the RPE block were not derived from a binaural psychoacoustic model. However, a binaural psychoacoustic model that is able to predict the masking curves for the two channels taking inter-aural dependencies into consideration does not exist, so it may be worthwhile to try the stereo perceptual biasing rules to incorporate an approximation of a binaural psychoacoustic model into the synthesis filter in the RPE scheme. This will also reduce the computational complexity.

Comparing the coding delays of SLP and HE-AAC-v2; the delay of HE-AAC-v2 is 7171 samples or 163 ms. The delay of the SLP coder is 1152 samples or 26 ms, because this is the number of samples that have to be buffered before the first set of prediction coefficients can be computed. It is expected that the coding delay can be even lower, because the ULD Coder, which is based on LP, achieves a delay of only 6 ms. Lowering the number of samples that need to be buffered before the first set of prediction coefficients can be calculated should reduce the coding delay of SLP. This can be achieved by introducing some sort of dynamic windowing into the coding scheme. However, it is expected that a delay of 6 ms cannot be achieved by the SLP coder of [84], because the lower limit of the coding delay

will not be determined by buffering of samples in the LSPLP block, but by buffering of samples in the SE block.

Regarding the computational complexity, the pulse optimization in RPE block was the bottleneck in the SLP coder. The estimated complexity of SLP is somewhat lower than the complexity of the HE-AAC-v2 coder, which is at least 100 MIPS. We expect that the computational complexity of SLP will not be an issue if the pulse optimization in RPE block is simplified. In addition, SLP can be made bit stream scalable by using two additional techniques called RPE layer mixing and RPE coding of additional subbands [64]. Thus we may safely conclude that the performance of the SLP coder looks promising for a first version.

It was found for the ULD coder that sample-wise backward adaptive coding has suboptimal noise shaping capabilities for reduced bit-rate coding (such as, 64 kbit/s for each channel). Thus, a frame-wise forward adaptive predictive coding with backward adaptive quantization step size and clipping was presented [23]. It was claimed that the forward adaptive predictive coding offers advantages like: robustness against high quantization errors, noise shaping capabilities with better ability to maintain constant bit-rate, and improved error resilience. The clipping in the quantization loop introduces quantization noise that is not white, but resembles the pre-filtered signal, and therefore the quantization noise is always smaller than the signal spectrum and its shape is a mixture of signal spectrum and masking threshold. The coder was considered to be a good candidate to bridge the bit-rate gap between high-quality speech coders and low-delay audio coders, thus indeed suggesting the possibility to code audio using speech technology. Since in [84] it was found that the decoded signal sounded noisy, it would be interesting to extend this idea [23] for the SLP as well. For instance, in the encoder we could apply the stereo perceptually biased prediction scheme to remove the irrelevancy and then use our forward adaptive SLP scheme in the closed loop to remove the redundancy. Thus the main challenge would be to design a stereo quantizer with time varying step size that is calculated backward adaptively.

- Utilize SLP on low-frequency bands: In [76] it was concluded

that the time-domain cross-channel prediction is not easily applicable to multi-channel perceptual audio coding. However, there the experiments were conducted on a five-channel AAC encoder, where the left and right channel inputs to the AAC encoder were replaced by the corresponding predicted residual signal from the center channel. The perceptual model that provided the quantization thresholds for each scale factor band was obtained from the original five-channel signal. It was claimed that more bits are required to encode the high-frequency part of the signal and this increase outpaces the bit reduction realized in the low frequency regions. One possible solution suggested in [76] to take advantage of the bit reduction seen in low frequency regions is to utilize cross-channel prediction on low frequency bands. This suggestion seems logical because all the spatial cues are relevant in the low frequency regions and thus they can be efficiently captured in the prediction matrices, covariance matrix, and in the rotation angle. The other option would be to utilize SLP in different subbands along lines similar to the one described in [116]. The other possible solution has already been discussed before, where we suggested to modularize the coder in two separate stages consisting of an irrelevancy removal stage followed by the redundancy reduction stage. The mono version of this idea is presented in [42]. This idea would then free us from using the same quantization threshold as those for the original signal. For the stereo case, a low-complexity modularized version coder could consist of a stereo version of the perceptually biased LP followed by a SLP based lossless coding scheme.

Before concluding, packet-loss concealment strategies [63] were not addressed in this thesis. However, there is abundant literature [18] about error concealments for speech coding which may also be adapted to our coder to guarantee error resilient transmission.

To conclude: if the research goals set in **1. (a)**, **1. (b)**, **1. (c)**, and **1. (d)** are met, we envision a coder with low-delay, low-complexity, error resilient, and scalable with almost state of the art audio quality. Additionally, as a feature it should also offer the possibility for the manipulation of the spatial cues. We propose to use the SLP coder for applications that require competitive quality digital audio with a relatively low-bit-rate, but that also place severe constraints on the coding

delay and/or computational complexity. Applications that require a low coding delay include in-ear monitoring for musicians and wireless digital transmission to loudspeakers. Applications that require an audio coder with a low computational complexity are mostly portable applications such as mobile phones.

Appendix A

Block-Levinson Algorithm

This appendix describes the block-Levinson algorithm and serves as a background material for Chapters 2, 3, and 4. As discussed in Chapter 2, the block-Levinson algorithm is used in the SLP control box to calculate the optimal prediction matrices and is described in Section A.1. In Chapter 4, it was suggested to transmit (forward or backward) normalized reflection matrices and the zero-lag correlation matrix, as they are the logical counterparts of mono LP transmission parameters. The algorithm to generate the normalized reflection matrices is described in Section A.2. As described in Chapter 3, for the LSPLP system, the block-Levinson algorithm can also be used to determine the optimal prediction matrices, and with a slight modification, the normalized reflection matrices (for transmission/storage purpose) associated with the minimum-phase matrix polynomial without actually creating the minimum-phase matrix polynomial. In the decoder, the prediction matrices are reconstructed from the (transmitted) normalized reflection matrices using the block Step-up recursion, and this is described in Section A.3.

A.1 Block-Levinson Algorithm

Levinson developed an efficient recursive algorithm for solving the symmetric Toeplitz problem. The method also generalizes to the nonsymmetrical case, and is derived elaborately in [80]. The method also generalizes to the multi-channel case, but it seems to be less well known. Therefore we give here a derivation that follows from [80], but now we are dealing with matrix operations and hence special care needs to be taken to select the order of matrix operations.

For equal auto- and cross-predictor orders we obtain a sequence of complex 2×2 correlation matrices $\boldsymbol{C}_k$ given by

$$\boldsymbol{C}_k = \begin{bmatrix} r_{11}(k) & r_{12}(k) \\ r_{21}(k) & r_{22}(k) \end{bmatrix}, \tag{A.1}$$

for $k = 0, \pm 1, \cdots, \pm K$. The elements of $\boldsymbol{C}_k$ are correlations as defined in (2.4). The matrices $\boldsymbol{C}_k$ have the property that $\boldsymbol{C}_{-k} = \boldsymbol{C}_k^H$, and thus all the block-Toeplitz matrices

$$\boldsymbol{\Gamma}_k = \begin{bmatrix} \boldsymbol{C}_0 & \boldsymbol{C}_{-1} & \cdots & \boldsymbol{C}_{-(k-1)} \\ \boldsymbol{C}_1 & \boldsymbol{C}_0 & \cdots & \boldsymbol{C}_{-(k-2)} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{C}_{k-1} & \boldsymbol{C}_{k-2} & \cdots & \boldsymbol{C}_0 \end{bmatrix}, \tag{A.2}$$

are Hermitian. Here we assume the block-Toeplitz matrix $\boldsymbol{\Gamma}_k$ to be positive definite, thus not singular.

The linear block-Toeplitz problem can thus be written as

$$\sum_{j=1}^{K} \boldsymbol{C}_{i-j} \boldsymbol{A}_{K,j} = \boldsymbol{P}_i \text{ for } i = 1, \ldots, K, \tag{A.3}$$

where the $\boldsymbol{A}_{K,j}$'s, $j = 1, \ldots, K$ are the *forward prediction matrices.* For convenience, in this section we will use $\boldsymbol{A}_j^{(K)}$ instead of $\boldsymbol{A}_{K,j}$.

The block-Levinson algorithm recursively solves the M-dimensional block-Toeplitz problem as

$$\sum_{j=1}^{M} \boldsymbol{C}_{i-j} \boldsymbol{A}_j^{(M)} = \boldsymbol{P}_i \text{ for } i = 1, \ldots, M \tag{A.4}$$

for $M = 1, 2, \ldots$ until $M = K$, which is the desired result. In following a recursion from step M to $M + 1$, the developing solution $\boldsymbol{A}^{(M)}$ changes from (A.4) to

$$\sum_{j=1}^{M} \boldsymbol{C}_{i-j} \boldsymbol{A}_j^{(M+1)} + \boldsymbol{C}_{i-(M+1)} \boldsymbol{A}_{M+1}^{(M+1)} = \boldsymbol{P}_i \text{ for } i = 1, \ldots, M+1. \tag{A.5}$$

Eliminating $\boldsymbol{P}_i$ gives

$$\sum_{j=1}^{M} \boldsymbol{C}_{i-j} \left(\boldsymbol{A}_j^{(M)} - \boldsymbol{A}_j^{(M+1)} \right) \left(\boldsymbol{A}_{M+1}^{(M+1)} \right)^{-1} = \boldsymbol{C}_{i-(M+1)}, \tag{A.6}$$

for $i = 1, 2, \ldots, M$ and by letting $i \to M + 1 - i$ and $j \to M + 1 - j$

$$\sum_{j=1}^{M} \boldsymbol{C}_{j-i} \boldsymbol{E}_j^{(M)} = \boldsymbol{C}_{-i}, \tag{A.7}$$

with

$$\boldsymbol{E}_j^{(M)} = \left(\boldsymbol{A}_{M+1-j}^{(M)} - \boldsymbol{A}_{M+1-j}^{(M+1)} \right) \left(\boldsymbol{A}_{M+1}^{(M+1)} \right)^{-1}, \tag{A.8}$$

or put in another way

$$\boldsymbol{A}_{M+1-j}^{(M+1)} = \boldsymbol{A}_{M+1-j}^{(M)} - \boldsymbol{E}_j^{(M)} \boldsymbol{A}_{M+1}^{(M+1)} \text{ for } j = 1, 2, \ldots, M. \tag{A.9}$$

This means that if we can use recursion to find the order M quantities $\boldsymbol{A}^{(M)}$ and $\boldsymbol{E}^{(M)}$ and the order $M+1$ quantity $\boldsymbol{A}_{M+1}^{(M+1)}$, then all of the other $\boldsymbol{A}_j^{(M+1)}$ will follow. Luckily, $\boldsymbol{A}_{M+1}^{(M+1)}$ follows from (A.5) with $i = M + 1$

$$\sum_{j=1}^{M} \boldsymbol{C}_{M+1-j} \boldsymbol{A}_j^{(M+1)} + \boldsymbol{C}_0 \boldsymbol{A}_{M+1}^{(M+1)} = \boldsymbol{P}_{M+1}. \tag{A.10}$$

Since

$$\boldsymbol{E}_{M+1-j}^{(M)} = \left(\boldsymbol{A}_j^{(M)} - \boldsymbol{A}_j^{(M+1)} \right) \left(\boldsymbol{A}_{M+1}^{(M+1)} \right)^{-1}, \tag{A.11}$$

we can substitute the previous order quantities in $\boldsymbol{E}$ to get $\boldsymbol{A}_j^{(M+1)}$. This results in

$$\boldsymbol{A}_{M+1}^{(M+1)} = \left[\sum_{j=1}^{M} \boldsymbol{C}_{M+1-j} \boldsymbol{E}_{M+1-j}^{(M)} - \boldsymbol{C}_0 \right]^{-1} \left[\sum_{j=1}^{M} \boldsymbol{C}_{M+1-j} \boldsymbol{A}_j^{(M)} - \boldsymbol{P}_{M+1} \right]. \tag{A.12}$$

The recursion relation for $\boldsymbol{E}$ follows from the left-hand solutions, which we will call the *backward prediction matrices* $\boldsymbol{B}_{K,i}$. Again for convenience, we will use $\boldsymbol{B}_i^{(K)}$ instead of $\boldsymbol{B}_{K,i}$. With the left-hand solutions, we deal with the following set of equations

$$\sum_{j=1}^{M} \boldsymbol{C}_{j-i} \boldsymbol{B}_j^{(M)} = \boldsymbol{P}_i, \text{ for } i = 1, 2, \ldots, M. \tag{A.13}$$

The same sequence of operations on this set as described previously leads to

$$\sum_{j=1}^{M} \boldsymbol{C}_{i-j} \boldsymbol{E}_j'^{(M)} = \boldsymbol{C}_i, \tag{A.14}$$

with

$$\boldsymbol{E}_j'^{(M)} = \left(\boldsymbol{B}_{M+1-j}^{(M)} - \boldsymbol{B}_{M+1-j}^{(M+1)}\right)\left(\boldsymbol{B}_{M+1}^{(M+1)}\right)^{-1}. \tag{A.15}$$

It can be seen from (A.14) that the $\boldsymbol{E}_j'$ satisfy exactly the same equation as the $\boldsymbol{A}_j$ (see A.4), except for the substitution $\boldsymbol{P}_i \rightarrow \boldsymbol{C}_i$ on the right-hand side. Therefore, we can quickly deduce from (A.12) that

$$\boldsymbol{E}_{M+1}'^{(M+1)} = \left[\sum_{j=1}^{M} \boldsymbol{C}_{M+1-j}\boldsymbol{E}_{M+1-j}^{(M)} - \boldsymbol{C}_0\right]^{-1}\left[\sum_{j=1}^{M} \boldsymbol{C}_{M+1-j}\boldsymbol{E}_j'^{(M)} - \boldsymbol{C}_{M+1}\right]. \tag{A.16}$$

Similarly, the $\boldsymbol{E}_j$ satisfy the same equation as the $\boldsymbol{B}_j$, except for the substitution $\boldsymbol{P}_i \rightarrow \boldsymbol{C}_{-i}$. This leads to

$$\boldsymbol{E}_{M+1}^{(M+1)} = \left[\sum_{j=1}^{M} \boldsymbol{C}_{j-M-1}\boldsymbol{E}_{M+1-j}'^{(M)} - \boldsymbol{C}_0\right]^{-1}\left[\sum_{j=1}^{M} \boldsymbol{C}_{j-M-1}\boldsymbol{E}_j^{(M)} - \boldsymbol{C}_{-M-1}\right]. \tag{A.17}$$

The matrices $\boldsymbol{E}_{M+1}^{(M+1)}$ and $\boldsymbol{E}_{M+1}'^{(M+1)}$ are called the forward and backward *reflection matrices,* respectively. The same substitution can be applied to (A.9) and its partner for $\boldsymbol{B}$ to get the following final equations

$$\boldsymbol{E}_j^{(M+1)} = \boldsymbol{E}_j^{(M)} - \boldsymbol{E}_{M+1-j}'^{(M)}\boldsymbol{E}_{M+1}^{(M+1)}, \tag{A.18}$$

$$\boldsymbol{E}_j'^{(M+1)} = \boldsymbol{E}_j'^{(M)} - \boldsymbol{E}_{M+1-j}^{(M)}\boldsymbol{E}_{M+1}'^{(M+1)}. \tag{A.19}$$

We can now start the described recursive procedure with the initial values

$$\boldsymbol{A}_1^{(1)} = \{\boldsymbol{C}_0\}^{-1}\boldsymbol{P}_1, \tag{A.20}$$

$$\boldsymbol{E}_1^{(1)} = \{\boldsymbol{C}_0\}^{-1}\boldsymbol{C}_{-1}, \tag{A.21}$$

$$\boldsymbol{E}_1'^{(1)} = \{\boldsymbol{C}_0\}^{-1}\boldsymbol{C}_1. \tag{A.22}$$

At each stage M, we use (A.16) and (A.17) to find $\boldsymbol{E}_{M+1}'^{(M+1)}$, $\boldsymbol{E}_{M+1}^{(M+1)}$, and then from (A.18) and (A.19) to find the other components of $\boldsymbol{E}_j^{(M+1)}$, $\boldsymbol{E}_j'^{(M+1)}$. From there $\boldsymbol{A}_j^{(M+1)}$ is calculated using (A.12) and (A.9). Similarly, $\boldsymbol{B}_j^{(M+1)}$ can be calculated. The above recursive procedure is called the block-Levinson algorithm.

A.2 Normalization

As described in Chapter 4, for coding applications it is useful to transmit the normalized reflection matrices and zero-lag correlation matrix as they

are the logical counterparts of mono LP transmission parameters. In this section we describe the normalization definition used to generate the normalized reflection matrices.

It is known that along with the forward and backward reflection matrices, the block-Levinson algorithm can be used to generate the positive-definite forward and backward *error* or *innovation variance matrices,* denoted as $\boldsymbol{R}_k$ and $\boldsymbol{R}'_k$, respectively. It is known [66] that

$$\boldsymbol{R}_{k-1}\boldsymbol{E}'_k = \boldsymbol{E}_k^H \boldsymbol{R}'_{k-1}, \tag{A.23}$$

and that a recurrence relation holds for the innovation variance matrices

$$\begin{aligned} \boldsymbol{R}_k &= \boldsymbol{R}_{k-1}(\boldsymbol{I} - \boldsymbol{E}'_k\boldsymbol{E}_k) = \boldsymbol{R}_{k-1} - \boldsymbol{E}_k^H \boldsymbol{R}'_{k-1}\boldsymbol{E}_k, \\ \boldsymbol{R}'_k &= \boldsymbol{R}'_{k-1}(\boldsymbol{I} - \boldsymbol{E}_k\boldsymbol{E}'_k) = \boldsymbol{R}'_{k-1} - \boldsymbol{E}_k'^H \boldsymbol{R}_{k-1}\boldsymbol{E}'_k, \end{aligned} \tag{A.24}$$

where $\boldsymbol{R}_0 = \boldsymbol{C}_0$.

Consider factorizing $\boldsymbol{R}_k$ and $\boldsymbol{R}'_k$ in the form [66]

$$\boldsymbol{R}_k = \boldsymbol{M}_k^H \boldsymbol{M}_k,\ \boldsymbol{R}'_k = \boldsymbol{M}_k'^H \boldsymbol{M}'_k, \tag{A.25}$$

for suitable 2×2 *normalizing matrices* $\boldsymbol{M}_k$ and $\boldsymbol{M}'_k$. The matrices $\boldsymbol{M}_k$ and $\boldsymbol{M}'_k$ can be calculated within *unitary left factors* [66]. Then the 2×2 *normalized reflection matrix* $\boldsymbol{\xi}_k$ is defined by

$$\boldsymbol{\xi}_k = \boldsymbol{M}'_{k-1}\boldsymbol{E}_k\boldsymbol{M}_{k-1}^{-1} = \left(\boldsymbol{M}_{k-1}'^H\right)^{-1} \boldsymbol{E}_k'^H \boldsymbol{M}_{k-1}^H, \tag{A.26}$$

and a similar definition exists for the backward normalized reflection matrices $\boldsymbol{\xi}'_k$ [66].

To resolve the ambiguity in $\boldsymbol{M}_k$ and $\boldsymbol{M}'_k$, we define these matrices as positive-definite symmetric matrices. Since $\boldsymbol{R}_k$ can be decomposed as $\boldsymbol{R}_k = \boldsymbol{U}\breve{\boldsymbol{S}}\boldsymbol{U}^H$ (i.e., the standard singular value definition [80] of a positive-definite matrix), this means that we take $\boldsymbol{M}_k = \boldsymbol{U}\breve{\boldsymbol{S}}^{1/2}\boldsymbol{U}^H$. Thus we have

$$\boldsymbol{M}_k = \boldsymbol{R}_k^{1/2},\ \boldsymbol{M}'_k = \boldsymbol{R}_k^{\prime 1/2}. \tag{A.27}$$

The initial values are taken to be $\boldsymbol{M}_0 = \boldsymbol{M}'_0 = \boldsymbol{C}_0^{1/2}$.

From the above recursion for obtaining the normalized reflection matrices, it is evident that the reflection matrices $\boldsymbol{E}_k$ and $\boldsymbol{E}'_k$ can be mapped onto the normalized reflection matrices $\boldsymbol{\xi}_k$ and $\boldsymbol{\xi}'_k$, which are directly coupled according to $\boldsymbol{\xi}'_k = \boldsymbol{\xi}_k^H$. Thus the associated normalized reflection matrices can also be calculated in place in the block-Levinson algorithm described in the previous section.

A.3 Block Step-up Recursion

As described in Chapter 4, for coding applications it is necessary to synthesize the forward prediction matrix in the decoder from the transmitted (forward) normalized reflection matrices $\boldsymbol{\xi}_k$ and the zero-lag correlation matrix $\boldsymbol{C}_0$. For the inverse mapping $\{\boldsymbol{\xi}_k\} \rightarrow \{\boldsymbol{E}_k, \boldsymbol{E}'_k\}$, we need $\{\boldsymbol{M}_{k-1}, \boldsymbol{M}'_{k-1}\}$. Due to the recursive relation between successive $\boldsymbol{M}_{k-1}$, we only need to transmit $\boldsymbol{M}_0$ (or, alternatively, the zero-lag correlation matrix $\boldsymbol{C}_0$). Having the set $\{\boldsymbol{E}_k, \boldsymbol{E}'_k\}$, the prediction matrices $\boldsymbol{A}_{K,k}$ can be generated using a recursive relation. This is achieved by a process called block Step-up recursion where $\boldsymbol{\xi}_k$ and $\boldsymbol{C}_0$ are translated to forward and backward prediction matrices. The algorithm makes use of the already described recursive rules and it is described next.

Initialize:
$\boldsymbol{M}_0 = \boldsymbol{C}_0^{1/2}$ and $\boldsymbol{M}'_0 = \boldsymbol{C}_0^{1/2}$;
$\boldsymbol{E}_1 = (\boldsymbol{M}'_0)^{-1}\boldsymbol{\xi}_1\boldsymbol{M}_0$ and $\boldsymbol{E}'_1 = (\boldsymbol{M}_0)^{-1}\boldsymbol{\xi}_1^H\boldsymbol{M}'_0$;
$\boldsymbol{A}_{K,1} = \boldsymbol{E}_1$ and $\boldsymbol{B}_{K,1} = \boldsymbol{E}'_1$;
$\boldsymbol{R}_0 = \boldsymbol{C}_0$ and $\boldsymbol{R}'_0 = \boldsymbol{C}_0$;
Store:
$\boldsymbol{E}_1^{(dummy)} = \boldsymbol{E}_1$ and $\boldsymbol{E}_1^{\prime(dummy)} = \boldsymbol{E}'_1$;
loop: $k = 1, 2, \ldots, K-1$
$\boldsymbol{R}_k = \boldsymbol{R}_{k-1} - \boldsymbol{E}_k^{(dummy)H}\boldsymbol{R}'_{k-1}\boldsymbol{E}_k^{(dummy)}$;
$\boldsymbol{R}'_k = \boldsymbol{R}'_{k-1} - \boldsymbol{E}_k^{\prime(dummy)H}\boldsymbol{R}_{k-1}\boldsymbol{E}_k^{\prime(dummy)}$;
update normalizing matrices:
$\boldsymbol{M}_k = \boldsymbol{R}_k^{1/2}$ and $\boldsymbol{M}'_k = \boldsymbol{R}_k^{\prime 1/2}$;
calculate reflection matrices:
$\boldsymbol{E}_{k+1} = (\boldsymbol{M}'_k)^{-1}\boldsymbol{\xi}_{k+1}\boldsymbol{M}_k$;
$\boldsymbol{E}'_{k+1} = (\boldsymbol{M}_k)^{-1}\boldsymbol{\xi}_{k+1}^H\boldsymbol{M}'_k$;
update lower order prediction matrices:
$\boldsymbol{A}_{K,k+1} = \boldsymbol{E}_{k+1}$;
$\boldsymbol{B}_{K,k+1} = \boldsymbol{E}'_{k+1}$;
loop: $j = 1, 2, \ldots, k$
$\boldsymbol{A}_{K,j} = \boldsymbol{A}_{K,j} - \boldsymbol{E}_{k+1-j}^{\prime(dummy)}\boldsymbol{E}_{k+1}$;
$\boldsymbol{B}_{K,j} = \boldsymbol{B}_{K,j} - \boldsymbol{E}_{k+1-j}^{(dummy)}\boldsymbol{E}'_{k+1}$;
end;
$\boldsymbol{E}^{(dummy)} = \boldsymbol{A}$ and $\boldsymbol{E}^{\prime(dummy)} = \boldsymbol{B}$;
end;

Appendix B

Autocorrelation Function of the Warped Signal

The normal equations (3.17) in the LPLP coefficient optimization involve the *autocorrelation function of the warped input signal* (ACFW). In this appendix we derive an expression for the autocorrelation of the warped signal s in terms of the original signal x. We will denote the original frequency θ associated with the z-domain and the warped frequency φ associated with the z'-domain, that is,

$$z = e^{j\theta} \text{ and } z' = e^{j\varphi}.$$

The frequency warping is defined by

$$z' = \frac{1 - \lambda z^{-1}}{z^{-1} - \lambda}. \tag{B.1}$$

Using (B.1), the all-pass transfers used in WLP and LPLP are given by

$$A(z) = \frac{z^{-1} - \lambda}{1 - z^{-1}\lambda} = z'^{-1}. \tag{B.2}$$

If the Discrete Fourier Transforms (DFTs) of the signals $x(n)$ and $s(n)$ are given by $X(e^{j\theta})$ and $S(e^{j\theta})$, respectively, then

$$\begin{aligned} X(e^{j\theta}) &= \sum_n x(n)e^{-j\theta n}, \\ S(e^{j\varphi}) &= \sum_n s(n)e^{-j\varphi n}, \end{aligned} \tag{B.3}$$

and using the Inverse DFT (IDFT)

$$\begin{aligned} x(n) &= \frac{1}{2\pi}\int_{-\pi}^{\pi} X(e^{j\theta})e^{j\theta n}d\theta, \\ s(n) &= \frac{1}{2\pi}\int_{-\pi}^{\pi} S(e^{j\varphi})e^{j\varphi n}d\varphi. \end{aligned} \tag{B.4}$$

Frequency warping maps the unit circle onto the unit circle and does not change the amplitude. Therefore

$$X(e^{j\theta}) = S(e^{j\varphi}). \tag{B.5}$$

The autocorrelation function of the signal s is given by

$$\rho_s(k) = \sum_n s(n)s^*(n-k) = \sum_n s(n+k)s^*(n). \tag{B.6}$$

Using (B.4) in (B.6) and changing the order of summation and integration, we get

$$\begin{aligned} \rho_s(k) &= \sum_n \frac{1}{2\pi}\left[\int_{-\pi}^{\pi} S(e^{j\varphi})e^{j\varphi(n+k)}d\varphi\right] s^*(n) \\ &= \frac{1}{2\pi}\int_{-\pi}^{\pi} S(e^{j\varphi})\left[\sum_n e^{j\varphi(n+k)}s^*(n)\right]d\varphi. \end{aligned} \tag{B.7}$$

Since

$$(e^{-j\varphi})^* = e^{j\varphi}, \text{ for } \varphi \in \mathbb{R} \tag{B.8}$$

and

$$z^{-1} = z^* \text{ and } z'^{-1} = z'^*, \text{ for } z, z' \in \mathbb{C} \text{ if } |z| = |z'| = 1, \tag{B.9}$$

(B.7) reduces to

$$\begin{aligned} \rho_s(k) &= \frac{1}{2\pi}\int_{-\pi}^{\pi} e^{j\varphi k}S(e^{j\varphi})\left[\sum_n e^{-j\varphi n}s(n)\right]^* d\varphi \\ &= \frac{1}{2\pi}\int_{-\pi}^{\pi} z'^k S(z')S^*(z')d\varphi. \end{aligned} \tag{B.10}$$

Differentiating the frequency warping function in (B.1), we obtain

$$d\varphi = \frac{1-\lambda^2}{(1-z\lambda)(1-z^{-1}\lambda)}d\theta. \tag{B.11}$$

Using (B.11), (B.5), and (B.9) in (B.10), we get

$$\begin{aligned}\rho_s(k) &= \frac{1}{2\pi}\int_{-\pi}^{\pi} A^{-k}(z)X(z)X^*(z)\frac{1-\lambda^2}{(1-z\lambda)(1-z^{-1}\lambda)}d\theta \\ &= \frac{1}{2\pi}\int_{-\pi}^{\pi}\left[D_0^{-1}(z)X(z)\right]\left[D_0^{-1}(z)X(z)A^k(z)\right]^* d\theta, \quad \text{(B.12)}\end{aligned}$$

where

$$D_0^{-1}(z) = \frac{\sqrt{1-\lambda^2}}{1-z^{-1}\lambda}. \quad \text{(B.13)}$$

If we now define the filtered signals $Y_k(z)$ as

$$Y_k(z) = D_0^{-1}(z)X(z)A^k(z), \quad \text{(B.14)}$$

for $k = 0, 1, \ldots, K$, we infer from (B.12) that

$$\rho_s(k) = \sum_n y_0(n)y_k^*(n), \quad \text{(B.15)}$$

with $y_k(n)$ the inverse z-transform of $Y_k(z)$. This implies that to obtain the optimal prediction coefficients for a WLP scheme, the autocorrelation function of the warped input signal $\rho_s(k)$ should be used [36]. As can be observed from (B.12), ACFW is obtained after pre-filtering the signal x through $D_0^{-1}(z)$ and then passing the signal through a chain of all-pass filters $A(z)$ [35]. We note that $\rho_s(k)$ is not equal to what is usually referred to as the *warped autocorrelation function* $\tilde{\rho}_s(k)$ defined as [40]

$$\tilde{\rho}_s(k) = \sum_n x(n)\tilde{y}_k^*(n), \quad \text{(B.16)}$$

with

$$\tilde{Y}_k(z) = X(z)A^k(z). \quad \text{(B.17)}$$

The difference between $\rho_s(k)$ and $\tilde{\rho}_s(k)$ is the compensation for the density change due to the frequency warping.

Appendix C

Decomposition of the Normalized Reflection Matrix

In Chapter 4, a specific decomposition of the normalized reflection matrix $\boldsymbol{\xi}_k$ is proposed for $k = 1, 2, \ldots, K$, where k is the section index and K the prediction order. For convenience, if we drop the index k, the matrix $\boldsymbol{\xi}$ is decomposed as

$$\boldsymbol{\xi} = \mathbf{R}(\gamma)\mathbf{R}(\delta)\boldsymbol{S}\mathbf{R}(\delta)\mathbf{R}(-\gamma), \tag{C.1}$$

where $\boldsymbol{\xi}$ contains real entries according to

$$\boldsymbol{\xi} = \begin{bmatrix} \xi_{11} & \xi_{12} \\ \xi_{21} & \xi_{22} \end{bmatrix}. \tag{C.2}$$

$\mathbf{R}$ denotes a rotation matrix according to

$$\mathbf{R}(\gamma) = \begin{bmatrix} \cos(\gamma) & \sin(\gamma) \\ -\sin(\gamma) & \cos(\gamma) \end{bmatrix}, \tag{C.3}$$

with $-\pi/2 < \gamma \leq \pi/2$, and $\boldsymbol{S}$ is a diagonal matrix with

$$\boldsymbol{S} = \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix}, \tag{C.4}$$

with $0 \leq |\sigma_2| \leq \sigma_1$.

In view of the fact that the proposed decomposition is not completely consistent with the standard SVD, we will show how the decomposition parameters $\{\gamma,\ \delta,\ \sigma_1, \sigma_2\}$ can be calculated from the entries of the normalized reflection matrix.

First we note that

$$\begin{aligned}\boldsymbol{\xi} &= \mathbf{R}(\gamma)\mathbf{R}(\delta)\boldsymbol{S}\mathbf{R}(\delta)\mathbf{R}(-\gamma)\\ &= \mathbf{R}(\gamma+\delta)\boldsymbol{S}\mathbf{R}(\delta-\gamma)\\ &= \mathbf{R}(\delta)\mathbf{R}(\gamma)\boldsymbol{S}\mathbf{R}(-\gamma)\mathbf{R}(\delta).\end{aligned}$$

We introduce the auxiliary matrices $\boldsymbol{B}$ and $\boldsymbol{B}'$ according to

$$\begin{aligned}\boldsymbol{B} &= \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} = \mathbf{R}(\delta)\boldsymbol{S}\mathbf{R}(\delta),\\ \boldsymbol{B}' &= \begin{bmatrix} B'_{11} & B'_{12} \\ B'_{21} & B'_{22} \end{bmatrix} = \mathbf{R}(\gamma)\boldsymbol{S}\mathbf{R}(-\gamma),\end{aligned}$$

and note that $\boldsymbol{B}$ is a skew-symmetric matrix and that $\boldsymbol{B}'$ is a symmetric matrix: $B_{12} = -B_{21}$ and $B'_{12} = B'_{21}$.

From the definition of $\boldsymbol{B}$ and (C.1) follows

$$\boldsymbol{B} = \mathbf{R}(-\gamma)\boldsymbol{\xi}\mathbf{R}(\gamma),$$

which yields

$$\begin{aligned}B_{12} &= -\cos(\gamma)\sin(\gamma)\xi_{11} - \sin^2(\gamma)\xi_{21}\\ &\quad + \cos^2(\gamma)\xi_{12} + \cos(\gamma)\sin(\gamma)\xi_{22},\\ B_{21} &= -\cos(\gamma)\sin(\gamma)\xi_{11} + \cos^2(\gamma)\xi_{21}\\ &\quad - \sin^2(\gamma)\xi_{12} + \sin(\gamma)\cos(\gamma)\xi_{22}.\end{aligned}$$

Since $B_{12} = -B_{21}$, it can be shown that

$$\gamma = \frac{1}{2}\tan^{-1}\left(\frac{\xi_{21}+\xi_{12}}{\xi_{11}-\xi_{22}}\right), \quad -\pi/2 < \gamma \leq \pi/2. \tag{C.5}$$

Similarly, from the definition of $\boldsymbol{B}'$ and (C.1) follows

$$\boldsymbol{B}' = \mathbf{R}(-\delta)\boldsymbol{\xi}\mathbf{R}(-\delta),$$

which yields

$$\begin{aligned}B'_{12} &= \cos(\delta)\sin(\delta)\xi_{11} + \sin^2(\delta)\xi_{21}\\ &\quad + \cos^2(\delta)\xi_{12} + \cos(\delta)\sin(\delta)\xi_{22},\\ B'_{21} &= -\sin(\delta)\cos(\delta)\xi_{11} + \cos^2(\delta)\xi_{21}\\ &\quad + \sin^2(\delta)\xi_{12} - \sin(\delta)\cos(\delta)\xi_{22}.\end{aligned}$$

Since $B'_{12} = B'_{21}$, it can be shown that

$$\delta = \frac{1}{2}\tan^{-1}\left(\frac{\xi_{21} - \xi_{12}}{\xi_{11} + \xi_{22}}\right), \quad -\pi/2 < \delta \leq \pi/2. \tag{C.6}$$

Having expressions for γ and δ, we only require expressions for σ_1 and σ_2. For convenience, we introduce the shorthand notations

$$\begin{aligned}
\boldsymbol{v}_1 &= [\cos(\delta+\gamma) \quad \sin(\delta+\gamma)]^T, \\
\boldsymbol{v}_2 &= [-\sin(\delta+\gamma) \quad \cos(\delta+\gamma)]^T, \\
\boldsymbol{w}_1 &= [\cos(\delta-\gamma) \quad -\sin(\delta-\gamma)]^T, \text{ and} \\
\boldsymbol{w}_2 &= [\sin(\delta-\gamma) \quad \cos(\delta-\gamma)]^T.
\end{aligned}$$

For any vector $\boldsymbol{z}$ we have

$$\boldsymbol{\xi}\boldsymbol{z} = \begin{bmatrix} \boldsymbol{v}_1 & \boldsymbol{v}_2 \end{bmatrix} \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} \begin{bmatrix} \boldsymbol{w}_1 & \boldsymbol{w}_2 \end{bmatrix}^T \boldsymbol{z}.$$

Taking $\boldsymbol{z}$ such that

$$\begin{bmatrix} \boldsymbol{w}_1 & \boldsymbol{w}_2 \end{bmatrix}^T \boldsymbol{z} = \begin{bmatrix} 1 \\ 0 \end{bmatrix},$$

or, equivalently, $\boldsymbol{z} = \boldsymbol{w}_1$ yields

$$\boldsymbol{\xi}\boldsymbol{w}_1 = \boldsymbol{v}_1\sigma_1. \tag{C.7}$$

This leads to various expressions for σ_1, for example, we can multiply (C.7) with $\boldsymbol{v}_1^T$ to obtain an explicit expression: $\sigma_1 = \boldsymbol{v}_1^T\boldsymbol{\xi}\boldsymbol{w}_1$. However, we can also look element-wise to the vectors of (C.7). In this way, we find

$$\sigma_1 = \frac{\xi_{11}\cos(\delta-\gamma) - \xi_{12}\sin(\delta-\gamma)}{\cos(\delta+\gamma)}, \tag{C.8}$$

and

$$\sigma_1 = \frac{\xi_{21}\cos(\delta-\gamma) - \xi_{22}\sin(\delta-\gamma)}{\sin(\delta+\gamma)}. \tag{C.9}$$

For numerical robustness, the equation with largest absolute denominator can be selected.

Similarly, taking $\boldsymbol{z}$ such that

$$\begin{bmatrix} \boldsymbol{w}_1 & \boldsymbol{w}_2 \end{bmatrix}^T \boldsymbol{z} = \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

or, equivalently, $\boldsymbol{z} = \boldsymbol{w}_2$ yields

$$\boldsymbol{\xi}\boldsymbol{w}_2 = \boldsymbol{v}_2\sigma_2. \tag{C.10}$$

Considering the vector of (C.10) element-wise, we have

$$\sigma_2 = \frac{\xi_{11}\sin(\delta-\gamma) + \xi_{12}\cos(\delta-\gamma)}{-\sin(\delta+\gamma)}, \tag{C.11}$$

and

$$\sigma_2 = \frac{\xi_{21}\sin(\delta-\gamma) + \xi_{22}\cos(\delta-\gamma)}{\cos(\delta+\gamma)}. \tag{C.12}$$

For numerical robustness the equation with largest absolute denominator can be selected.

In conclusion, given the matrix $\boldsymbol{\xi}$, we can calculate the decomposition parameters γ, δ, σ_1 and σ_2 using (C.5), (C.6), (C.8) or (C.9), and (C.11) or (C.12), respectively.

Appendix D

Alternative Parameterizations of the Normalized Reflection Matrix

In Chapter 4 it was already proposed that in order to quantize the normalized reflection matrices, it has to be decomposed into structures that effectively hold the major matrix characterizations, for example, Singular Value Decomposition (SVD) and/or Eigenvalue Decomposition (EVD). The rationale behind the approach is that for strictly contractive normalized reflection matrices, the eigenvalues and singular values have similar characteristics as RCs (for example, they have the same range) and, with small adaptations, can be quantized using known techniques [68]. Furthermore, the additional parameters in these decompositions can be quantized efficiently, if their quantization accuracy is adapted according to the singular values and/or eigenvalues. In this appendix we present alternative sets of parameterizations of the normalized reflection matrices, based on EVD and a combination of SVD and EVD.

D.1 Parameterization Using EVD

Consider EVD for decomposing the normalized reflection matrix $\boldsymbol{\xi}_k$ for $k = 1, 2, \ldots, K$, where k is the section index and K the prediction order. For convenience, we drop the index k. The eigenvalue analysis is based on determining a matrix $\boldsymbol{D}$ and a matrix $\boldsymbol{W}$ such that [80]

$$\boldsymbol{\xi W} = \boldsymbol{W D}, \tag{D.1}$$

where $\boldsymbol{D}$ a diagonal matrix given by

$$\boldsymbol{D} = \begin{bmatrix} d_1 & 0 \\ 0 & d_2 \end{bmatrix}, \tag{D.2}$$

with d_1 and d_2 called the eigenvalues, and $\boldsymbol{W}$ a matrix with

$$\boldsymbol{W} = (\boldsymbol{v}_1 \ \boldsymbol{v}_2), \tag{D.3}$$

with $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ called the eigenvectors.

For subsequent use, we review a few basic properties of the eigenvalues and eigenvectors where we assume that $\boldsymbol{\xi}$ is a real matrix as a consequence of the fact that they stem from real stereo audio data.

1. The eigenvalues d_1 and d_2 maybe either real and distinct, or they may appear as a complex-conjugated pair.

2. If $\boldsymbol{v}_1$ is an eigenvector then $c\boldsymbol{v}_1$ is an eigenvector for any $c \in \mathbb{C}$ with $c \neq 0$.

3. If the eigenvalues are distinct (or both zero), then the eigenvectors are (or can be taken as) linearly independent. That is, $\boldsymbol{v}_1 \neq c\boldsymbol{v}_2$ for any $c \in \mathbb{C}$ and $c \neq 0$.

4. If $\boldsymbol{W}$ is regular (that is, if the eigenvalues are distinct), then we can write

   $$\boldsymbol{\xi} = \boldsymbol{W}\boldsymbol{D}\boldsymbol{W}^{-1}. \tag{D.4}$$

   Thus $\boldsymbol{\xi}$ is described by two matrices $\boldsymbol{D}$ and $\boldsymbol{W}$. The eigenvalues (either real or complex-conjugated) can be described by two real parameters. The two eigenvectors can also be described by two real parameters. In case the eigenvalues are identical, the description (D.4) does not hold; we will return to this issue later.

Due to the contractive nature of the normalized reflection matrices, the absolute values of its eigenvalues are always less than one. In the case of real eigenvalues they can be treated as RCs, that is, we could apply a non-uniform quantization in the range (-1,1) or map them to arcsine coefficients or LARs, and perform an uniform quantization in the latter domains. For complex eigenvalues, different strategies can be used. For example, we could take the radius of the complex number and map them in a way similar to a real eigenvalue, and take the angle of the complex number and quantize them with accuracy dependent on the magnitude

of the radius. However, this is not a preferred strategy, because in that case we need to signal in the bit stream whether we are dealing with real or complex numbers. Instead, it is easier to construct from the two eigenvalues (either complex-conjugated or real), a real second-order polynomial $P_2(z)$, the so-called characteristic polynomial of the matrix $\boldsymbol{\xi}$, given by

$$P_2(z) = (1 - z^{-1}d_1)(1 - z^{-1}d_2). \tag{D.5}$$

Since eigenvalues are in absolute value less than one, $P_2(z)$ is a minimum-phase polynomial. Consequently, $P_2(z)$ can be transmitted in standard ways as RCs (e.g., mapped to arcsine coefficients or LAR representation) or as LSFs. We denote the RCs of $P_2(z)$ as κ_1 and κ_2.

For the eigenvectors, we also have to distinguish between the cases of real (both distinct and identical) and complex-conjugated eigenvalues. The starting point for this distinction is the following.

1. If we have two real eigenvalues, then the eigenvectors can be written as real vectors.

2. If we have two complex-conjugated eigenvalues, then the eigenvectors form a complex-conjugated pair, that is $\boldsymbol{v}_2 = \boldsymbol{v}_1^*$.

As evident from the above discussion, the description of the eigenvector data depends on the three possible characters (real and distinct, complex-conjugated pair, and real identical) of the eigenvalues. We will now discuss the parameterization and quantization of the eigenvector data associated with these three cases separately.

Real and distinct eigenvalues: In this scenario, we have two real eigenvectors in $\boldsymbol{W}$. Due to the earlier properties, we can scale the eigenvectors to unit norm. Therefore the matrix $\boldsymbol{W}$ can be written as

$$\boldsymbol{W} = \begin{bmatrix} \cos(\alpha) & \cos(\beta) \\ \sin(\alpha) & \sin(\beta) \end{bmatrix}, \tag{D.6}$$

described by two angles with $-\pi < \alpha, \beta \leq \pi$ and $\beta \neq \alpha$ and $\beta \neq \alpha \pm \pi$.

A more convenient description will be to write the matrix $\boldsymbol{W}$ as a product of two matrices, each described by a single parameter

$$\boldsymbol{W} = \boldsymbol{W}_1 \boldsymbol{W}_2, \tag{D.7}$$

with $\boldsymbol{W}_1$ an orthogonal matrix according to

$$\boldsymbol{W}_1 = \begin{bmatrix} \cos(\gamma) & -\sin(\gamma) \\ \sin(\gamma) & \cos(\gamma) \end{bmatrix}, \tag{D.8}$$

and $\boldsymbol{W}_2$ given by

$$\boldsymbol{W}_2 = \begin{bmatrix} \cos(\delta) & \cos(\delta) \\ \sin(\delta) & -\sin(\delta) \end{bmatrix}, \tag{D.9}$$

where without loss of generality we can take $0 < |\delta| < \pi/2$ and $0 < |\gamma| < \pi/2$. The relation between α, β and γ, δ is simply a transformation of coordinates. The angle γ is essentially halfway between α and β. Twice the angle δ is the difference between the angles α and β. The determinant of $\boldsymbol{W}_1$ is equal to 1 and the determinant of $\boldsymbol{W}_2$ is equal to $-\sin(2\delta)$. In other words, all possible ill-conditioning of the matrix $\boldsymbol{W}$ resides in $\boldsymbol{W}_2$. Therefore, quantization of δ can be done in such a way that the relative variation in the determinant is roughly constant. Substituting $\boldsymbol{W} = \boldsymbol{W}_1\boldsymbol{W}_2$ in (D.4) shows that $\boldsymbol{W}_1$ operates merely as a pre- and a post-rotation. It is therefore assumed that the parameter γ describing this matrix can be uniformly quantized with γ being restricted to the range $(-\pi/2, \pi/2]$. The parameter δ determines how close the vector $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ are. It is assumed that the quantization of δ can best be done on a non-uniform way, where, possibly, the quantization is dependent on the parameters d_1 and d_2.

Complex-conjugated eigenvalues: In this case, the complex eigenvectors can be described by two angles as well, though the interpretation of these angles is obviously different than in the case of real and distinct eigenvectors. For an efficient data transmission, the accuracy of these angles is preferably coupled to the complex eigenvalues, in particular to its radius. Thus the matrix $\boldsymbol{W}$ can be described by

$$\boldsymbol{W} = \begin{bmatrix} r_1 e^{j\phi} & r_1 e^{-j\phi} \\ r_2 e^{-j\phi} & r_2 e^{j\phi} \end{bmatrix}, \tag{D.10}$$

with radii r_1, r_2, and angle ϕ with $0 < |\phi| < \pi$. Since scaling of eigenvectors is allowed, (D.10) can be rewritten to

$$\boldsymbol{W} = \boldsymbol{W}_3\boldsymbol{W}_4, \tag{D.11}$$

with

$$\boldsymbol{W}_3 = \begin{bmatrix} sc & 0 \\ 0 & 1/c \end{bmatrix}, \tag{D.12}$$

with $c \in \mathbb{R}^+$ and $s = \pm 1$ and

$$\boldsymbol{W}_4 = \begin{bmatrix} e^{j\phi} & e^{-j\phi} \\ e^{-j\phi} & e^{j\phi} \end{bmatrix}. \tag{D.13}$$

We note that the determinant of $\boldsymbol{W}_3$ equals ± 1 and the determinant of $\boldsymbol{W}_4$ equals $2j\sin(2\phi)$. The parameter c can be best quantized uniformly on a logarithmic scale. The angle ϕ can be treated similarly as the parameters δ.

Substituting $\boldsymbol{W} = \boldsymbol{W}_3\boldsymbol{W}_4$ in (D.4) shows that $\boldsymbol{W}_3$ operates merely as a pre- and post-scaling. A proposal would be to write $sc = \tan(\psi)$ (with $c = |\tan(\psi)|$) and to quantize ψ with $\pi/2 < \psi \leq \pi/2$. Another proposal would be to quantize c uniformly on a decibel scale. Comparing the determinants of $\boldsymbol{W}_2$ and $\boldsymbol{W}_4$ suggests that the parameter ϕ acts similarly as δ.

Real and identical eigenvalues: If the eigenvalues are real and identical, the decomposition in (D.4) does not generally hold. Instead, we use the decomposition

$$\boldsymbol{\xi} = d\left(\boldsymbol{I} + \tau \begin{bmatrix} \cos(\alpha) \\ \sin(\alpha) \end{bmatrix} [-\sin(\alpha)\ \ \cos(\alpha)]\right), \tag{D.14}$$

where $d = d_1 = d_2$ is the eigenvalue, $\boldsymbol{I}$ the identity matrix, α the angle associated with the eigenvector, and τ is a constant. As before, the eigenvalues can be mapped to a second-order polynomial $P_2(z)$, then to RCs and quantized uniformly in the LAR domain. The angle α can be efficiently quantized uniformly. The parameter τ is a ratio indicating the weight of the matrix defined by α in comparison to the identity matrix $\boldsymbol{I}$. The parameter d can be quantized uniformly in the logarithmic domain.

A problem that remains at the decoder is the interpretation of the received parameters of the eigenvector data. This interpretation depends on the character of the eigenvalues. This may cause confusion if due to quantization the eigenvalues change their character (real-distinct, complex-conjugated, and real identical). Different strategies can be used to solve this. An option would be to indicate in the bit stream the original character of the eigenvalues and restore that in the decoder when changed due to quantization. Another option would be to control the eigenvalue quantization in such a way that the character of the quantized eigenvalues remains unaltered. Yet another option would to check in the encoder if the character of the eigenvalues has changed due to the quantization and choose appropriate parameters corresponding to the new character. An example of the latter procedure is as follows. If in the quantization plane of the eigenvalues, complex-conjugated pairs and real eigenvalues are mapped onto the same representation, then presumably, there is also an eigenvalue pair with $d_1 = d_2$ which is mapped to this representation.

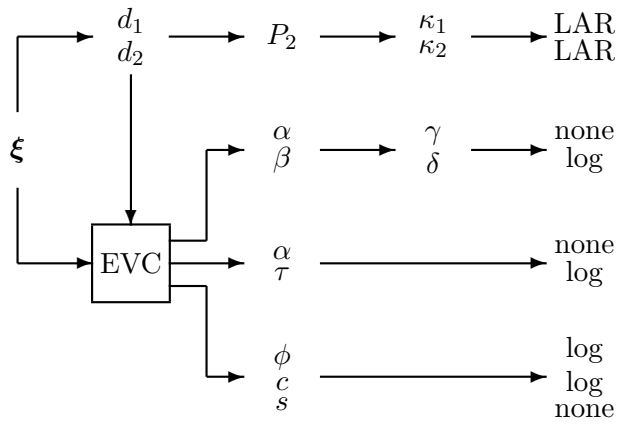


Figure D.1: *Preprocessing steps before uniform quantization for the normalized reflection matrices using EVD*

Roughly, we should have for this quantization tile that the product $d_1 d_2$ is constant. We then omit the parameters γ, δ or ϕ, c and find the best parameters τ, α instead. Since the quantization strategy is known at the decoder, the decoder knows for each quantized eigenvalue pair whether the quantization strategy could have changed the eigenvalue character. In that case the two eigenvalues are taken as the geometric mean of the received eigenvalues (real and identical) and the eigenvector information is correctly interpreted as being τ and α.

In Figure D.1, a sketch of possible preprocessing steps before uniform quantization is given. Obviously, variations are possible, for example, the LAR mappings can be replaced by arcsine mappings without deviating from the essential idea. For the eigenvectors, we need to distinguish between real (both distinct and identical) and complex-conjugated eigenvalues. This is taken care by the EVC (Eigenvalue Classifier) box. The EVC box generates from the normalized reflection matrix the corresponding eigenvector parameters associated with the character of the eigenvalues d_1 and d_2. Possible confusions in the character (real/complex) of these values due to quantization have to be taken into account when generating the eigenvector parameters. More elegantly, this can be done on basis of the character of the quantized eigenvalues, since this information is also available at the decoder.

The decoder implements the inverse process. It receives the quantized parameters and reconstructs d_1 and d_2. Given these values, the receiver knows which eigenvector data are contained in the bit stream: either γ, δ

or s, c, ϕ or α, τ. The matrix $\boldsymbol{\xi}$ can then be reconstructed accordingly.

D.2 Parameterization Using Combined EVD and SVD

The normalized reflection matrix can be decomposed using both EVD and SVD. Combining the eigenvalues in a second-order polynomial $P_2(z)$, and quantizing the associated RCs belonging to this polynomial gives an accurate control over the characteristic equation. The singular values σ_1 and σ_2 can be mapped onto the ratio $c = \sigma_1/\sigma_2$. Such a ratio can be efficiently quantized uniformly on a logarithmic scale. The two additional parameters α and β associated with SVD can be combined to $\gamma = (\alpha + \beta)/2$ (like in the method proposed in Chapter 4) and quantized uniformly.

In Figure D.2, a sketch of possible preprocessing steps before uniform quantization is given. Obviously, variations are possible, for example, the LAR mappings can be replaced by arcsine mappings without deviating from the essential idea.

The decoder implements the inverse process. It receives the quantized parameters and reconstructs d_1 and d_2. Given these values and c, the receiver is able to reconstruct the singular values. From $d_1, d_2, \sigma_1, \sigma_2$, the parameter δ can be reconstructed. Like in the approach proposed in Chapter 4, an ambiguity appears which is resolved by an extra sign-bit s. From δ and γ, the accompanying matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ can be constructed.

In all the cases discussed so far in this thesis, we have for each normalized reflection matrix two coefficients (eigenvalues in $\boldsymbol{D}$ or singular values in $\boldsymbol{S}$), which with some adaptation can be treated like RCs appearing in mono LP systems. The accompanying matrices ($\boldsymbol{V}$ and $\boldsymbol{U}$, or $\boldsymbol{W}$) can be encoded with an accuracy (number of bits) depending on the eigenvalues or singular values.

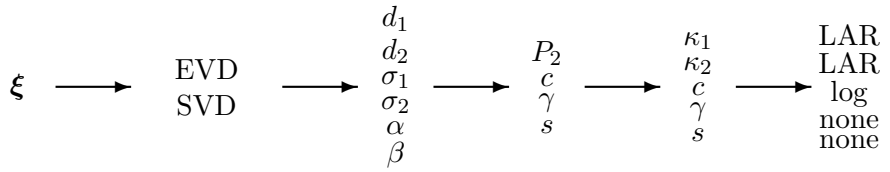


Figure D.2: *Preprocessing steps before uniform quantization for the normalized reflection matrices using a combined EVD and SVD method.*

D.3 Sensitivity Analysis

In this section we evaluate the performance of the alternative quantization schemes for a first-order SLP system along similar lines as described in Chapter 4 (see Section 4.3). We compared the parameterization of the normalized reflection matrix $\boldsymbol{\xi}_1$ using the combined EVD and SVD method described in Section D.2 with the variant of the SVD method described in Chapter 4 (see Section 4.2.1). The zero-lag correlation matrix is parameterized according to Chapter 4 (see Section 4.2.2) for both the cases. The reason for ignoring the parameterization technique using EVD as described in Section D.1 in the present evaluation is due to the following. Firstly, it has been suggested in [117] that the quantization of the normalized reflection matrix is sensitive to both eigenvalues and singular values and therefore both of them need to be treated with care. Since the parameterization technique using EVD provides direct control only over the eigenvalues but not on the singular values, it is expected to perform worse. Secondly, the parameterization technique using EVD needs to make a distinction between three separate cases (real, complex, and real-identical) of eigenvalues and is thus more complex.

For the combined EVD and SVD method, the mean of the SD as a function of step size for each individually quantized parameter is shown in Figure D.3. Unlike the top plot in Figure 4.6, we clearly observe that the mean SD as a function of step size cannot be modeled by a simple linear relations defined by (4.26) for the parameters κ_1, κ_2, and c. The best fits according to (4.26) are indicated by the straight lines in Figure D.3.

We now select an equal mean SD denoted by D_{eq} for all the six parameters of the individual quantization, and determine the corresponding step size for each parameter. With these selected step sizes, we quantize all the six parameters at the same time. The plot in Figure D.4 displays the measured mean SD denoted by D_m obtained by quantizing all the six parameters simultaneously, along with $D_p = \sqrt{6}D_{eq}$. The dashed line represents $D_m = D_p$. We observe that on average the mean distortion $D_m = 1.31D_p$ for the parameterization using the combined EVD and SVD method, implying that the measured data are roughly 31% larger than the predicted values. For the parameterization technique using the variant of SVD, the measured data were roughly 21% larger than the predicted value. Thus using the combined EVD and SVD method, decorrelation is achieved to a lesser extent compared to the method proposed in Chapter 4.

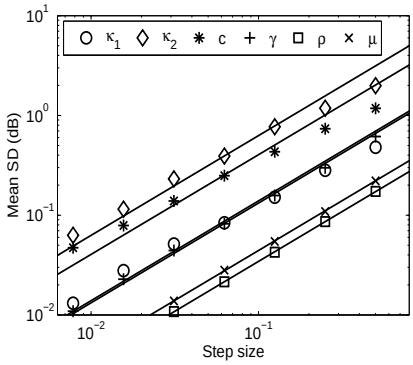


Figure D.3: *Mean SD as a function of the quantization step size for each individually quantized parameter κ_1, κ_2, c, γ, ρ, and μ.*

For a mean distortion D_m of 1 dB, it is found that the standard deviation of the distortions associated with the combined EVD and SVD method and the variant of the SVD method are 1.69 and 1.22, respectively. This suggests that the combined EVD and SVD method has a broader distribution and, therefore, a more pronounced tail that would result in more outliers in practice. In Figure D.5, we have plotted the probability (estimated from the measurements with an average distortion

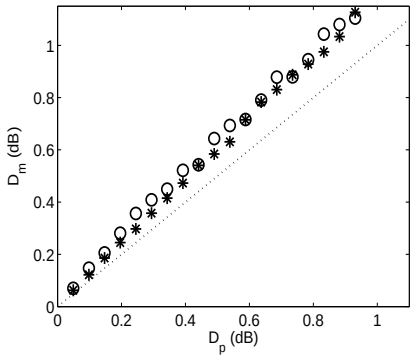


Figure D.4: *Measured mean SD denoted by D_m associated with the combined EVD and SVD method (circles) and the variant of SVD method (asterisks) versus predicted overall mean distortion D_p.*

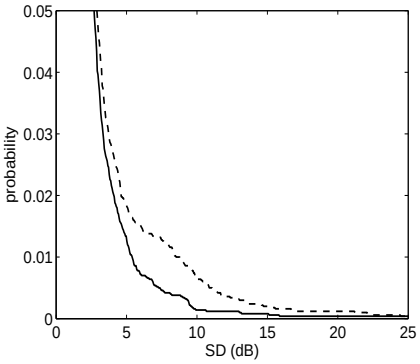


Figure D.5: *Probability that the distortion exceeds the SD level for the combined EVD and SVD method (dashed line) and the variant of SVD method (solid line). The average distortion in both cases is 1 dB.*

of 1 dB) that the SD is above a certain level. Indeed we clearly observe that the combined EVD and SVD method has a higher probability of generating data with distortions substantially higher than the average value.

D.4 Concluding Remarks

Several alternative sets of parameterizations of the normalized reflection matrices, such as EVD, and a combination of SVD and EVD were suggested and investigated as transmission parameters for stereo linear predictive coding systems. Compared to the parameterization proposed in Section 4.2.1, the alternatives schemes are more directly coupled to an EVD analysis than an SVD analysis.

We note that these two schemes suffer from the following fundamental problems.

- The contractive property of the normalized reflection matrices is not guaranteed when quantizing the parameters, because eigenvalues less than unity not necessary imply that singular values are also less than unity.
- Eigenvalues may be real or complex or there could be possible multiplicity for the alternative method proposed in Section D.1. This has to be signaled in the bit stream. On top of that, the character of

the eigenvectors is associated with that of the eigenvalues. This implies that when the character of the eigenvalues changes due to the quantization, the eigenvectors have to be adapted as well. Though these problems can be solved, it makes this alternative scheme less straightforward and, therefore, less attractive.

- Finally, experiments with the alternative scheme proposed in Section D.2 indicate that the nearly uncorrelated addition of quantization errors that we had for the proposed scheme does not hold (or hold to a lesser degree) and there is a higher probability of outliers. It is expected that for the method proposed in Section D.1, these effects will be even larger since it deviates even more from the proposal in Section 4.2.1

Based on these considerations we conclude that the parameterizations proposed in Chapter 4 is to be preferred over the alternatives presented in this appendix.

Appendix E

Additional data to Chapter 5

In Chapter 5, we considered quantization of Laguerre-based stereo linear predictors based on data from stereo audio files. This appendix contains supplementary plots concerning the quantization strategy.

In Figures E.1 and E.2, we plotted the histograms with respect to parameters σ_2, κ and γ which complement the histograms of σ_1, ρ and μ (Figure 5.1).

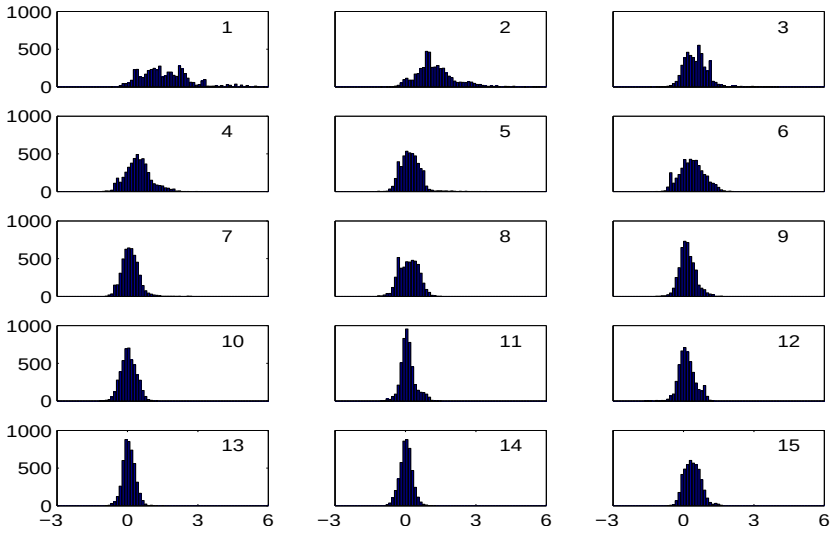


Figure E.1: *Histograms of $\sigma_2(k)$, with k indicated in each plot. All the histograms are in the LAR domain.*

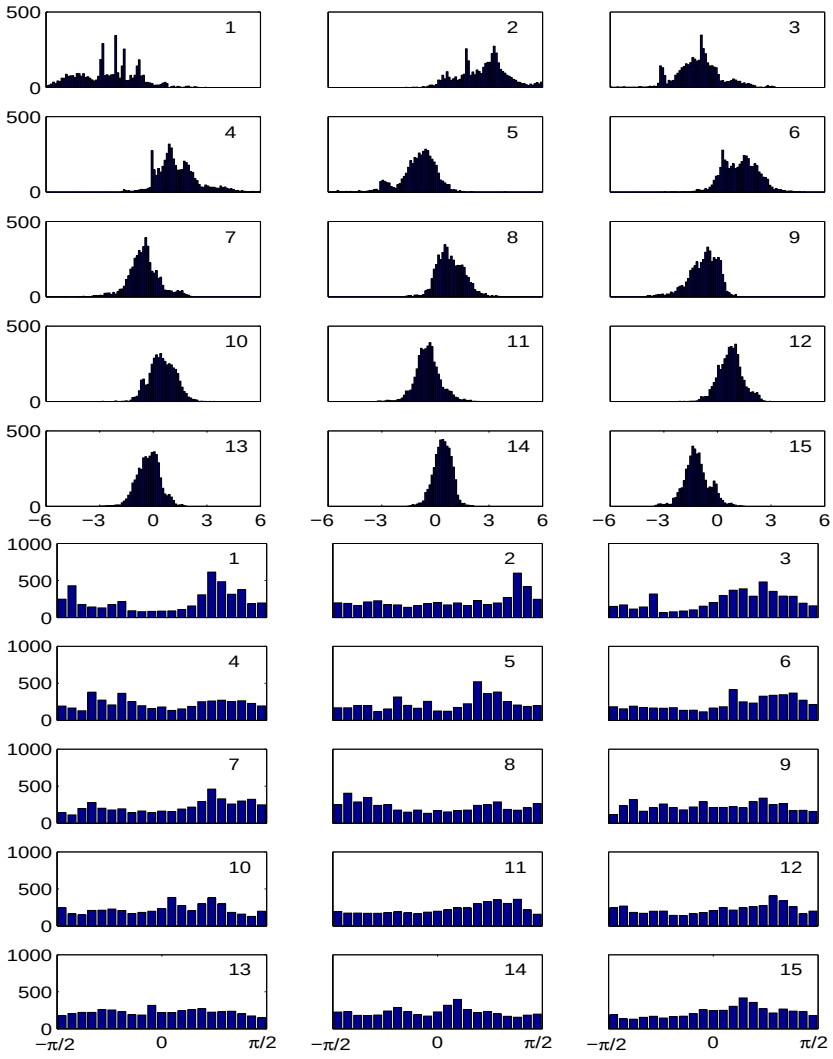


Figure E.2: *Histograms of $\kappa(k)$ (top 15 plots) in the LAR domain and the histograms of $\gamma(k)$ (bottom 15 plots) in radians, with k indicated in each plot.*

In Figure E.3 we have plotted the bit allocation over the different parameters as a function of section index k. This allocation achieves an equal average distortion per individually quantized parameter and a measured overall distortion of 1 dB for simultaneous quantization. The plot reveals that the bit allocation is not equal over the different sections.

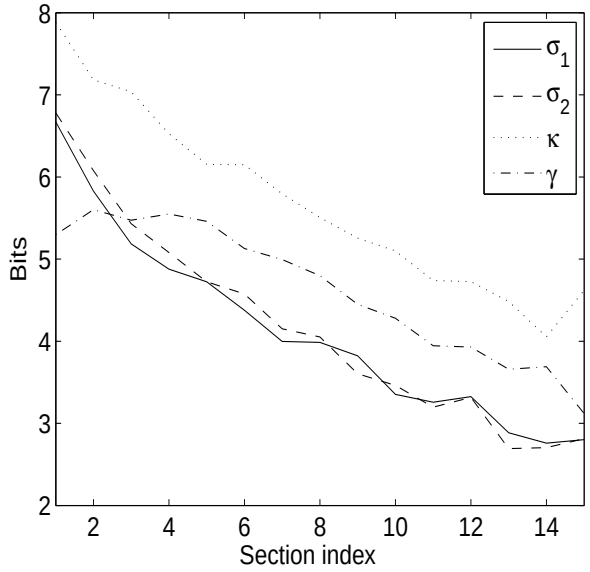


Figure E.3: *Distribution of bits for $\sigma_1(k)$, $\sigma_2(k)$, $\kappa(k)$, and $\gamma(k)$ as a function of section index k for a mean SD of 1 dB.*

Instead, more bits are allocated to sections having a lower index, which is in line with our expectation based on single channel data [68].

References

[1] C. Cellier, P. Chenes, and M. Rossi. Lossless audio data compression for real-time applications. In *Proc. 95th AES Conv.*, New York, USA, 7-10 Oct. 1993. Preprint 3780.

[2] T. Robinson. Simple lossless and near-lossless waveform compression. Technical Report CUED/F-INFENG/TR.156, Cambridge University, Cambridge, UK, 1994.

[3] A. A. M. L. Bruekers, A. W. J. Oomen, R. J. van der Vleuten, and L. M. van de Kerkhof. Lossless coding for DVD audio. In *Proc. 101st AES Conv.*, Los Angeles, CA, USA, 8-11 Nov. 1996. Preprint 4358.

[4] M. Purat, T. Liebchen, and P. Noll. Lossless transform coding of audio signals. In *Proc. 102nd AES Conv.*, Munich, Germany, 22-25 Mar. 1997. Preprint 4414.

[5] M. A. Gerzon, P. G. Craven, J. R. Stuart, M. J. Law, and R. J. Wilson. The MLP lossless compression system. In *Proc. AES 17th Int. Conf.: High-Quality Audio Coding*, pages 61–75, Florence, Italy, 2-5 Sept. 1999.

[6] T. Liebchen. MPEG-4 lossless coding for high-definition audio. In *Proc. 115th AES Conv.*, New York, USA, 10-13 Oct. 2003. Preprint 5872.

[7] E. Zwicker and H. Fastl. *Psychoacoustics Facts and Models.* Springer-Verlag, Berlin, Germany, 1990.

[8] B. C. J. Moore. *An Introduction to the Psychology of Hearing.* Academic Press, New York, USA, 1997.

[9] R. Zelinski and P. Noll. Adaptive transform coding of speech signals. *IEEE Trans. Acoust., Speech, Signal Processing*, 25(4):299–309, 1977.

[10] K. Brandenburg, G. G. Langenbucher, H. Schramm, and D. Seitzer. A digital signal processor for real time adaptive transform coding of audio signal up to 20 kHz bandwidth. In *Proc. ICCC*, pages 474–477, Sept. 1982.

[11] K. Brandenburg and G. Stoll. ISO/MPEG-1 Audio: A generic standard for coding of high-quality digital audio. *J. Audio Eng. Soc*, 42(10):780–792, Oct. 1982.

[12] IISO/MPEG Committee. Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s - part 3: Audio, 1993.

[13] G. Stoll. *Collected Papers on Digital Audio Bit-rate Reduction*, chapter ISO-MPEG-2 Audio: A Generic Standard for the Coding of Two-Channel and Multichannel Sound, pages 43–53. Audio Engineering Society Inc., New York, USA, 1996.

[14] B. Grill. The MPEG-4 general audio coder. In *Proc. AES 17th Int. Conf.: High-Quality Audio Coding*, pages 147–156, Florence, Italy, 2-5 Sept. 1999.

[15] G. Davidson. *The Digital Signal Processing Handbook*, chapter Digital Audio Coding: Dolby AC-3, pages 41.1–41.23. CRC Press, Boca Raton, FL, USA, 1998.

[16] D. Sinha, J. D. Johnston, S. Dorward, and S. Quackenbush. *The Digital Signal Processing Handbook*, chapter The Perceptual Audio Coder (PAC), pages 42.1–42.20. CRC Press, Boca Raton, FL, USA, 1998.

[17] K. Akagiri, M. Katakura, H. Yamauchi, E. Saito, M. Kohut, M. Nishiguchi, and K. Tsutsui. *The Digital Signal Processing Handbook*, chapter Sony Systems, pages 43.1–43.22. CRC Press, Boca Raton, FL, USA, 1998.

[18] P. Vary and R. Martin. *Digital Speech Transmission: Enhancement, Coding and Error Concealment.* Wiley, New York, USA, 2006.

[19] M. Lutzky, G. Schuller, M. Gayer, U. Krämer, and S. Wabnik. A guideline to audio codec delay. In *Proc. 116th AES Conv.*, Berlin, Germany, 8-11 May 2004. Preprint 6062.

[20] S. Dorward, D. Huang, S. A. Savari, G. Schuller, and B. Yu. Low delay perceptually lossless coding of audio signals. In *Proc. Data Compression Conference*, pages 312–320, Snowbird, UT, USA, 27-29 Mar. 2001.

[21] N. S. Jayant and P. Noll. *Digital Coding of Waveforms: Principles and Applications to Speech and Video.* Prentice-Hall, Englewood Cliffs, NJ, USA, 1984.

[22] U. Krämer, G. Schuller, S. Wabnik, J. Klier, and J. Hirschfeld. Ultra low delay audio coding with constant bit rate. In *Proc. 117th AES Conv.*, San Francisco, CA, USA, 28-31 Oct. 2004. Preprint 6197.

[23] S. Wabnik, G. Schuller, J. Hirschfeld, and U. Krämer. Reduced bit rate ultra low delay audio coding. In *Proc. 120th AES Conv.*, Paris, France, 20-23 May 2006. Preprint 6747.

[24] N. H. van Schijndel and S. van de Par. Rate-distortion optimized hybrid sound coding. In *Proc. WASPAA*, pages 235–238, New Paltz, NY, USA, 16-19 Oct. 2005.

[25] R. Salami, R. Lefebvre, A. Lakaniemi, K. Kontola, S. Bruhn, and A. Taleb. Extended AMR-WB for high-quality audio on mobile devices. *IEEE Commun. Mag.*, 44(5):90–97, May 2006.

[26] T. Moriya, N. Iwakami, K. Ikeda, and S. Miki. Extension and complexity reduction of TwinVQ audio coder. In *Proc. ICASSP*, volume 2, pages 1029–1032, Atlanta, GA, USA, 7-10 May 1996.

[27] T. Berger and J. D. Gibson. Lossy source coding. *IEEE Trans. Inform. Theory*, 44(6):2693–2723, 1998.

[28] Staff technical writer. Next generation of audio communications. *J. Audio Eng. Soc*, 54(9):865–867, Sep. 2006.

[29] B. S. Atal and S. L. Hanauer. Speech analysis and synthesis by linear prediction of the speech wave. *J. Acoust. Soc. Amer.*, 50(2):637–655, Aug. 1971.

[30] M. R. Schroeder. Linear prediction, extremal entropy and prior information in speech signal analysis and synthesis. *Speech Commun.*, 1(1):9–20, May 1982.

[31] M. R. Schroeder and B. S. Atal. Code-excited linear prediction (CELP): High-quality speech at very low bit rates. In *Proc. ICASSP*, volume 10, pages 937–940, Tampa, FL, USA, 26-29 Mar. 1985.

[32] B. Bessette, R. Salami, R. Lefebvre, M. Jelinek, J. Rotola-Pukkila, J. Vainio, H. Mikkola, and K. Jarvinen. The adaptive multirate wideband speech codec (AMR-WB). *IEEE Trans. Speech Audio Processing*, 10(8):620–636, 2002.

[33] S. Singhal. High quality audio coding using multipulse LPC. In *Proc. ICASSP*, pages 1101–1104, Albuquerque, NM, USA, 3-6 Apr. 1990.

[34] S. Boland and M. Deriche. High quality audio coding using multipulse LPC and wavelet decomposition. In *Proc. ICASSP*, volume 5, pages 3067–3069, Detroit, MI, USA, 9-12 May 1995.

[35] A. Oppenheim, D. Johnson, and K. Steiglitz. Computation of spectra with unequal resolution using the fast Fourier transform. *Proc. IEEE*, 59(2):299–301, 1971.

[36] H. W. Strube. Linear prediction on a warped frequency scale. *J. Acoust. Soc. Amer.*, 68(4):1071–1076, Oct. 1980.

[37] A. C. den Brinker, V. Voitishchuk, and S. J. L. van Eijndhoven. IIR-based pure linear prediction. *IEEE Trans. Speech Audio Processing*, 12(1):68–75, 2004.

[38] J. O. Smith III and J. S. Abel. Bark and ERB bilinear transforms. *IEEE Trans. Speech Audio Processing*, 7(6):697–708, 1999.

[39] E. Krüger and H. W. Strube. Linear prediction on a warped frequency scale. *IEEE Trans. Acoust., Speech, Signal Processing*, 36(9):1529–1531, 1988.

[40] A. Härmä, U. K. Laine, and M. Karjalainen. Warped linear prediction (WLP) in audio coding. In *Proc. NorSig-96*, pages 447–450, Espoo, Finland, 24-27 Sept. 1996.

[41] A. C. den Brinker and F. Riera-Palou. Pure linear prediction. In *Proc. 115th AES Conv.*, New York, USA, 10-13 Oct. 2003. Preprint 5924.

[42] B. Edler and G. Schuller. Audio coding using a psychoacoustic pre- and post-filter. In *Proc. ICASSP*, volume 2, Istanbul, Turkey, 22-26 May 2000.

[43] J. D. Johnston and A. J. Ferreira. Sum-difference stereo transform coding. In *Proc. ICASSP*, volume 2, pages 569–572, San Francisco, CA, USA, 23-26 Mar. 1992.

[44] J. Blauert. *Spatial Hearing: The Psychophysics of Human Sound Localization*. The MIT Press, Cambridge, Massachusetts, USA, 1997.

[45] H. Fuchs. Improving MPEG audio coding by backward adaptive linear stereo prediction. In *Proc. 99th AES Conv.*, New York, USA, 6-9 Oct. 1995. Preprint 4086.

[46] A. Härmä, U. K. Laine, and M. Karjalainen. An experimental audio codec based on warped linear prediction of complex valued signals. In *Proc. ICASSP*, volume 1, pages 323–326, Munich, Germany, 20-24 Apr. 1997.

[47] D. Yang, H. Ai, C. Kyriakakis, and C.-C. J. Kuo. High-fidelity multichannel audio coding with Karhunen-Loeve transform. *IEEE Trans. Speech Audio Processing*, 11(4):365–380, Jul. 2003.

[48] M. Briand, D. Virette, and N. Martin. Parametric coding of stereo audio based on Principal Component Analysis. In *Proc. (DAFx-06)*, pages 291–296, Montreal, Quebec, Canada, 18-20 Sept. 2006.

[49] B. Edler, H. Purnhagen, and C. Ferekidis. ASAC-Analysis/synthesis audio codec for very low bit rates. In *Proc. 100th AES Conv.*, Copenhagen, Denmark, 11-14 May 1996. Preprint 4179.

[50] H. Purnhagen and N. Meine. HILN-The MPEG-4 parametric audio coding tools. In *Proc. ISCAS*, volume 3, pages 201–204, Geneva, Switzerland, 28-31 May 2000.

[51] A. C. den Brinker, E. Schuijers, and A. W. J. Oomen. Parametric coding for high-quality audio. In *Proc. 112th AES Conv.*, Munich, Germany, 10-13 May 2002. Preprint 5554.

[52] W. Oomen, E. Schuijers, B. den Brinker, and J. Breebaart. Advances in parametric coding for high-quality audio. In *Proc. 114th AES Conv.*, Amsterdam, The Netherlands, 22-25 Mar 2003. Preprint 5852.

[53] J. Herre, K. Brandenburg, and D. Lederer. Intensity stereo coding. In *Proc. 96th AES Conv.*, Amsterdam, The Netherlands, 26 Feb. - 01 Mar. 1994. Preprint 3799.

[54] R. G. van der Waal and R. N. J. Veldhuis. Subband coding of stereophonic digital audio signals. In *Proc. ICASSP*, pages 3601–3604, Toronto, Ontario, Canada, 14-17 May 1991.

[55] C. Faller and F. Baumgarte. Efficient representation of spatial audio using perceptual parametrization. In *Proc. WASPAA*, pages 199–202, New Paltz, NY, USA, 21-24 Oct. 2001.

[56] C. Faller and F. Baumgarte. Binaural cue coding: a novel and efficient representation of spatial audio. In *Proc. ICASSP*, volume 2, pages 1841–1844, Orlando, FL, USA, 13-17 May 2002.

[57] C. Faller and F. Baumgarte. Binaural cue coding applied to stereo and multi-channel audio compression. In *Proc. 112th AES Conv.*, Munich, Germany, 10-13 May 2002. Preprint 5574.

[58] C. Faller and F. Baumgarte. Binaural cue coding-part II:schemes and applications. *IEEE Trans. Speech Audio Processing*, 11(6):520–531, Nov. 2003.

[59] C. Faller. Parametric multichannel audio coding: synthesis of coherence cues. *IEEE Trans. Audio Speech Language Processing*, 14(1):299–310, Jan. 2006.

[60] F. Baumgarte and C. Faller. Why binaural cue coding is better than intensity stereo coding. In *Proc. 112th AES Conv.*, Munich, Germany, 10-13 May 2002. Preprint 5575.

[61] J. Breebaart, S. van de Par, A. Kohlrausch, and E. Schuijers. High-quality parametric spatial audio coding at low bit rates. In *Proc. 116th AES Conv.*, Berlin, Germany, 8-11 May 2004. Preprint 6072.

[62] J. Breebaart, S. van de Par, A. Kohlrausch, and E. Schuijers. Parametric coding of stereo audio. *EURASIP Journal on Applied Signal Processing*, 2005(9):1305–1322, 2005.

[63] S. Lin and D. J. Costello Jr. *Error Control Coding: Fundamentals and Applications*. Prentice-Hall, Englewood Cliffs, NJ, USA, 1983.

[64] F. Riera-Palou, A. C. den Brinker, and A. J. Gerrits. A hybrid parametric-waveform approach to bit stream scalable audio coding. In *Proc. ASILOMAR SSC*, volume 2, pages 2250–2254, Pacific Grove, CA, USA, 7-10 Nov. 2004.

[65] P. Whittle. On the fitting of multivariate autoregressions, and the approximate canonical factorization of a spectral density matrix. *Biometrika*, 50:129–134, 1963.

[66] P. Delsarte and Y. V. Genin. Multichannel singular predictor polynomials. *IEEE Trans. Circuits Syst.*, 35(2):190–200, 1988.

[67] S. L. Marple. *Digital Spectral Analysis: With Applications*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1986.

[68] K. K. Paliwal and W. B. Kleijn. *Speech Coding and Synthesis*, chapter 12, pages 433–466. Elsevier, Amsterdam, The Netherlands, 1995.

[69] S. van de Par, A. Kohlrausch, R. Heusdens, J. Jensen, and S. H. Jensen. A perceptual model for sinusoidal audio coding based on spectral integration. *EURASIP Journal on Applied Signal Processing*, 2005:1292–1304, 2005.

[70] M. H. Hayes. *Statistical Digital Signal Processing and Modeling*. Wiley, New York, USA, 1996.

[71] P. Cambridge and M. Todd. Audio data compression techniques. In *Proc. 94th AES Conv.*, Berlin, Germany, 16-19 Mar. 1993. Preprint 3584.

[72] T. Liebchen. Lossless audio coding using adaptive multichannel prediction. In *Proc. 113th AES Conv.*, Los Angeles, CA, USA, 5-8 Oct. 2002. Preprint 5680.

[73] P. Gournay, J.-L. Garcia, and R. Lefebvre. Backward linear prediction for lossless coding of stereo audio. In *Proc. 116th AES Conv.*, Berlin, Germany, 8-11 May 2004. Preprint 6076.

[74] T. Chonavel and S. Saoudi. Multi-channel linear predictive coding of audio signals. In *Proc. EUROSPEECH*, pages 53–56, Madrid, Spain, 18-21 Sept. 1995.

[75] S. A. Ramprashad. Stereophonic CELP coding using cross channel prediction. In *Proc. IEEE Workshop on Speech Coding*, pages 136–138, Delavan, WI, USA, 17-20 Sep. 2000.

[76] S.-S. Kuo and J. D. Johnston. A study of why cross channel prediction is not applicable to perceptual audio coding. *IEEE Signal Processing Lett.*, 8(9):245–247, Sep. 2001.

[77] A. Biswas, T. Selten, and A. C. den Brinker. Stability of the synthesis filter in stereo linear prediction. In *Proc. 15th ProRISC*, pages 230–237, Veldhoven, The Netherlands, 25-26 Nov. 2004.

[78] P. Delsarte, Y. Genin, and Y. Kamp. Orthogonal polynomial matrices on the unit circle. *IEEE Trans. Circuits Syst.*, 25(3):149–160, Mar. 1978.

[79] Thousand Foot Krutch. Phenomenon. CD, 2003.

[80] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge Univ. Press, Cambridge, MA, USA, 1992.

[81] J. Smid, H. Trentelman, A. C. den Brinker, and E. Verbitskiy. LSF stereo coding. Technical Note PR-TN-2006/00686, Philips Research Eindhoven, Eindhoven, The Netherlands, Aug. 2006.

[82] T.-W. Lee. *Independent Component Analysis: Theory and Applications*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1998.

[83] R. J. Sluijter. *The Development of Speech Coding and the First Standard Coder for Public Mobile Telephony*. Ph.D. thesis, Technische Universiteit Eindhoven, Eindhoven, The Netherlands, 2005.

[84] J. Geurts. Stereo linear predictive coding of audio. M.Sc. thesis, Technische Universiteit Eindhoven, Eindhoven, The Netherlands, 2006.

[85] A. Härmä. Implementation of frequency-warped recursive filters. *Signal Process.*, 80(3):543–548, 2000.

[86] Y. W. Lee. Synthesis of electrical networks by means of Fourier transforms of Laguerre functions. *J. Math. Phys.*, 11:83–113, 1932.

[87] W. Kautz. Transient synthesis in the time domain. *IRE Trans. Circuit Theory*, 1(3):29–39, Sep. 1954.

[88] ISO/IEC. Information technology - coding of audio-visual objects. part 3: Audio, amendment 2: Parametric coding of high-quality audio, July 2004.

[89] A. Biswas and A. C. den Brinker. Lossless compression of digital audio using Laguerre based pure linear prediction. In *Proc. IEEE Benelux Signal Processing Symposium*, pages 49–52, Hilvarenbeek, The Netherlands, 15-16 Apr. 2004.

[90] P. W. Broome. Discrete orthonormal sequences. *J. Association for Computing Machinery*, 12:151–165, 1965.

[91] A. Biswas and A. C. den Brinker. Fast and efficient Laguerre based pure linear prediction. Technical Note PR-TN-2004/01053, Philips Research Eindhoven, Eindhoven, The Netherlands, Jan. 2005.

[92] A. Biswas and A. C. den Brinker. Stability of the stereo linear prediction schemes. In *Proc. 47th Int. Symp. ELMAR-2005 Focused on Multimedia Systems and Applications*, pages 221–224, Zadar, Croatia, 8-10 Jun. 2005.

[93] A. Biswas and A. C. den Brinker. Quantization of transmission parameters in stereo linear predictive systems. In *Proc. Data Compression Conference*, pages 262–271, Snowbird, UT, USA, 28-30 Mar. 2006.

[94] A. Gray Jr. and J. Markel. Quantization and bit allocation in speech processing. *IEEE Trans. Acoust., Speech, Signal Processing*, 24(6):459–473, 1976.

[95] M. Deriche and D. Ning. A novel audio coding scheme using warped linear prediction model and the discrete wavelet transform. *IEEE Trans. Audio Speech Language Processing*, 14(6):2039–2048, Nov. 2006.

[96] Rongshan Yu and C. C. Ko. A warped linear-prediction-based subband audio coding algorithm. *IEEE Trans. Speech Audio Processing*, 10(1):1–8, Jan. 2002.

[97] K. Palomäki, A. Härmä, and U. K. Laine. Warped linear predictive audio coding in video conferencing application. In *Proc. EUSIPCO*, volume 2, pages 1433–1436, Rhodes, Greece, 8-11 Sept. 1998.

[98] J. Makhoul, S. Roucos, and H. Gish. Vector quantization in speech coding. *Proc. IEEE*, 73(11):1551–1588, 1985.

[99] T. Painter and A. Spanias. Perceptual coding of digital audio. *Proc. IEEE*, 88(4):451–515, 2000.

[100] H. Hermansky, B. Hanson, and H. Wakita. Perceptually based linear predictive analysis of speech. In *Proc. ICASSP*, volume 10, pages 509–512, Tampa, FL, USA, 26-29 Mar 1985.

[101] S. van de Par, A. Kohlrausch, G. Charestan, and R. Heusdens. A new psychoacoustical masking model for audio coding applications. In *Proc. ICASSP*, volume 2, pages 1805–1808, Orlando, FL, USA, 13-17 May 2002.

[102] Alien Ant Farm. ANThology. CD, 2001.

[103] J. Pachelbel. Pachelbel's greatest hit: Canon in D. CD, 1991.

[104] SQAM (Sound Quality Assessment Material). CD 422 204-2, 1988.

[105] ITU. ITU-R BS 1534. Method for subjective assesment of intermediate quality level of coding systems, 2001.

[106] J.-M. Valin and C. Montgomery. Improved noise weighting in CELP coding of speech-applying the Vorbis psychoacoustic model to Speex. In *Proc. 120th AES Conv.*, Paris, France, 20-23 May 2006. Preprint 6746.

[107] J. Breebaart. *Modeling Binaural Signal Detection*. Ph.D. thesis, Technische Universiteit Eindhoven, Eindhoven, The Netherlands, 2001.

[108] C. Faller. *Parametric Coding of Spatial Audio*. Ph.D. thesis, Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland, 2004.

[109] P. Kroon, E. Deprettere, and R. Sluyter. Regular-pulse excitation–a novel approach to effective and efficient multipulse coding of speech. *IEEE Trans. Acoust., Speech, Signal Processing*, 34(5):1054–1063, Oct. 1986.

[110] F. Riera-Palou, A. C. den Brinker, A. J. Gerrits, and R. J. Sluijter. Improved optimisation of excitation sequences in speech and audio coders. *Electronics Letters*, 40(8):515–517, Apr. 2004.

[111] F. Riera-Palou, A. C. den Brinker, and A. J. Gerrits. Modelling long-term correlations in broadband speech and audio pulse coders. *Electronics Letters*, 41(8):508–509, Apr. 2005.

[112] Audio subgroup. Report on the verification test of MPEG-4 parametric coding of high-quality audio, 2004.

[113] H. C. Peter. Combined sinusoidal and pulse coding for audio compression. Technical Note PR-TN-2005/00183, Philips Research Eindhoven, Eindhoven, The Netherlands, 2005.

[114] G. Schuller and A. Härmä. Low delay audio compression using predictive coding. In *Proc. ICASSP*, volume 2, pages 1853–1856, Orlando, FL, USA, 13-17 May 2002.

[115] M. R. Schroeder. Synthesis of low-peak-factor signals and binary sequences with low autocorrelation. *IEEE Trans. Inform. Theory*, 16(1):85–89, Jan. 1970.

[116] F. Keiler. Real-time subband-ADPCM low-delay audio coding approach. In *Proc. 120th AES Conv.*, Paris, France, 20-23 May 2006. Preprint 6748.

[117] B. van Zweden and A. Willems. Quantization of two-channel prediction coefficients. Modeling Report 04.20, Technische Universiteit Eindhoven, Eindhoven, The Netherlands, Apr. 2005.

Curriculum Vitae

Arijit Biswas was born in Calcutta, India, on 17 September 1977. He received a Bachelor of Engineering (B.E.) degree in electronics and communication engineering from Bangalore University, India, and an M.Sc. degree in signal processing from the School of Electrical and Electronic Engineering at the Nanyang Technological University, Singapore, in 2001 and 2002, respectively. From April 2003 he is a Ph.D. student in the Signal Processing Systems Group at the Technische Universiteit Eindhoven, The Netherlands. His Ph.D. project was carried out in collaboration with the Digital Signal Processing Group at Philips Research Laboratories, Eindhoven. His research interests are mainly in speech and audio signal processing, specifically generic coding of sound (convergence of speech and audio coding), multi-channel audio coding, and digital signal processing in general.

Mr. Biswas has been awarded the Postgraduate Manpower Programme scholarship sponsored by the Economic Development Board, Singapore, and DAAD-ABB scholarship sponsored by ABB, Germany. He was also awarded the Best Student Technical Paper Award at the 121^{st} Audio Engineering Society Convention, San Francisco, USA in October 2006.