Stereo linear predictive coding of audio

Geurts, J.

Award date:
2006

Eindhoven University of Technology
Department of Electrical Engineering
Signal Processing Systems

TU/e technische universiteit eindhoven

Stereo Linear Predictive Coding of Audio

by Job Geurts

Master of Science thesis

Project period: September 2005 – August 2006
Report Number: 28-06

Commissioned by: Prof.dr.ir. J.W.M. Bergmans

Supervisors:
Dr. L.L.M. Vogten and A. Biswas MSc (TU/e).
Dr. A.C. den Brinker (Philips)

Additional Commission members:
Prof. A.G. Kohlrausch

Stereo Linear Predictive Coding of Audio

Job Geurts

28th August 2006

Preface

This thesis is the result of my nine-months graduation project called *Stereo Linear Predictive Coding of Audio.* This project has been performed at Philips Research Europe - Eindhoven, in order to obtain the Master of Science degree in Electrical Engineering from the Eindhoven University of Technology (TU/e). The project was proposed by the Audio and Speech Signal Processing Cluster, which is a part of the Digital Signal Processing Group of Philips Research, and was accepted as graduation project by the research chair of the Signal Processing Systems group, which is a part of the Department of Electrical Engineering of the TU/e.

I would like to thank the people that made it possible to deliver this thesis. To start with, I would like to thank dr. ir. Bert den Brinker for giving me the opportunity to perform my graduation project at Philips Research. I found the High Tech Campus an inspiring working environment. Furthermore, I would like to thank Bert and also Arijit Biswas, MSc for their guidance and support. They always had time to discuss the issues that came up during the project. I found our cooperation very fruitful. I would also like to thank dr. ir. Leo Vogten for his support at the TU/e. Finally, I would like to thank the students who inhabited the student room in the cellar of the DSP building for making my time at Philips such a pleasant stay. The table tennis games we played were great for the necessary relaxation.

Abstract

Audio coders are used to reduce the bit rate of audio data for applications that have limited storage or bandwidth capacities, but at the same time require high-quality audio. Single channel audio coders reduce the bit rate of digital audio data by exploiting intra-channel redundancies and irrelevancies in the data. Stereo audio coders try to remove inter-channel redundancies as well, to attain a lower bit rate than the sum of the bit rates of the separate channels while maintaining the quality level.

As a continuation of the work presented in [1], we propose a stereo audio coder, subject to the following criteria:

- Encoder and decoder form a system allowing perfect signal reconstruction in the absence of signal quantization and thus, near perfect reconstruction at the high bit rate end;
- The encoder constructs a main and a side signal similar to those provided by Optimum Coding of Stereo (OCS), because this is advantageous for low bit rate coding purposes.

The proposed stereo audio coder is called *Stereo Linear Predictive Coding of Audio* (SLP) and uses two rotators and Laguerre-based Pure Stereo Linear Prediction (L-PSLP). The encoder results in two spectrally flat, uncorrelated signals, called the main and side signal, and the decoder reconstructs the original input signal. This reconstruction is perfect in the absence of any quantization. SLP is psycho-acoustically inspired, because L-PSLP uses Laguerre filters instead of delays in the Stereo Linear Prediction analysis and synthesis filters. L-PSLP modifies the uniform frequency resolution, common to most signal processing systems, to a non-uniform frequency resolution. This non-uniform frequency resolution approximately corresponds to a psycho-acoustically relevant scale such as a Bark or an ERB scale. Before the prediction coefficients are quantized, we map them to a suitable representation, resulting in a smaller distortion of the analysis and synthesis filter characteristics. Inspired by informal listening tests, we quantize the main signal and the side signal for stereo input signals, and we quantize the main signal only, thus discarding the side signal, for mono and mono-like input signals. To quantize the main signal, we use Sinusoidal Extraction (SE) and Regular Pulse Excitation (RPE). To quantize the side signal, we also use SE and RPE, but only for frequencies less than 4 kHz. We then use in the decoder a decorrelation filter and a gain factor to construct a synthetic side signal from the main signal for frequencies greater than 4 kHz.

The performance of the SLP coder looks promising for a first version of the coder. We estimate the bit rate of SLP to be 58 kbps for mono or mono-like signals and 80 kbps for stereo signals. The subjective quality of the coded material of SLP, determined by performing a formal listening test, is between fair and good. This should be better, so some work has to be done. The coding delay of SLP is equal to 26 ms. We expect that the computational complexity of SLP will not be an issue if the pulse optimization in RPE is simplified. In addition, SLP can be made bit stream scalable by using two techniques called RPE layer mixing and RPE coding of additional subbands.

We propose to use the SLP coder for applications that require high 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.

Contents

Chapter 1

Introduction

Digital storage and transmission media are reaching higher capacities and are becoming faster. However, there are still many audio and/or speech applications that have limited storage or bandwidth capacities, but at the same time require high-quality audio and/or speech. These applications include portable audio devices and mobile phone systems. Therefore, it is necessary for these applications to reduce the bit rate of high quality digital audio and speech data. Audio [2] and speech coders [3] can be used for this purpose. Single channel audio or speech coders reduce the bit rate of digital audio or speech data by exploiting intra-channel redundancies and irrelevancies in the data. These coders are either lossless or lossy. Lossless coders only exploit redundancies and do not introduce any loss in quality, because the output signal is a perfect reconstruction of the input signal. With lossy coders, which exploit both redundancies and irrelevancies, the output resembles the input in a perceptual way but not necessarily in a waveform sense. Stereo audio or speech coders try to remove inter-channel redundancies as well, to attain a lower bit rate than the sum of the bit rates of the separate channels while maintaining the quality level.

This thesis concerns stereo audio coding. An overview of some well-known stereo audio coding techniques can be found in Section 1.1, followed by the problem description and problem statement of this Masters thesis project in Section 1.2 and the outline of this thesis in Section 1.3.

1.1 Overview of Stereo Audio Coding Techniques

In this section, the following stereo audio coding techniques are described along with their advantages and disadvantages:

- Sum-Difference Stereo;
- Intensity Stereo;
- Parametric Stereo;
- Stereo Linear Prediction.

1.1.1 Sum-Difference Stereo

Sum-Difference Stereo, also known as Mid-Side Stereo, was introduced by Johnston in 1989 [4]. An improved version was published in 1992 [5]. Sum-Difference Stereo was intended as a stereo coding technique to be used in transform coders. A Sum-Difference stereo coder uses both *Left* and *Right* (L and R) and *Sum* and *Difference* (M and S) signals, switched in both frequency and time in a signal dependent fashion. The M and S signals are given by

$$\begin{aligned} M[n] &= \tfrac{1}{2}\left(L[n] + R[n]\right) \\ S[n] &= \tfrac{1}{2}\left(L[n] - R[n]\right) \end{aligned} \quad . \tag{1.1}$$

For every time and frequency partition, a threshold for the left and for the right channel is calculated. The two thresholds are compared, and if they vary less than 2 dB, then the coder is switched into the M/S mode. This results in a S signal with a very low energy for correlated stereo channels, which means that less bits have to be spent. However, the psycho-acoustic model applied to the L and R signals cannot be applied directly to the M and S signals. The bit rate reduction that can be reached with this technique varies nearly between 50% and 0% (when the coder is always in the L/R mode).

The transformation from L/R to M/S is reversible, which means that Sum-Difference Stereo is a lossless technique. This makes it very suitable for high bit rate coding. However, the bit rate reduction is not enough to make it a low bit rate stereo coding technique.

1.1.2 Intensity Stereo

Intensity Stereo, as first described by van der Waal et al. in 1991 [6], but further developed by Herre et al. [7] [8], exploits the irrelevancy of phase cues at higher frequencies. In a first-order approximation for higher frequencies, the human auditory system takes the cues needed for spatial perception from the energy maxima in the left and right channels at each frequency, discarding the phase information, because this information is unreliable at higher frequencies [9].

Intensity Stereo uses a rotator to transform the L and R signals into *Intensity* and *Error* (I and E) signals by maximizing the energy in the I signal. An example of Rotation on signals L and R can be seen in Figure 1.1, which is a Lissajous plot of the involved signals. However, this will not lead to a significant gain for most stereo recordings. Therefore, for higher frequencies (above approximately 2 kHz) the E signal is excluded from transmission and replaced with scaling factors for the L and R signals. Thus, for higher frequencies the Fourier transforms of the reconstructed L and R signals differ in their amplitude, but are identical in their phase information. The bit rate reduction that can be reached with this technique lies between about 20% and 40%.

A less complex implementation of Intensity Stereo generates the I signal from the sum of the L and R signals instead of rotating the signals. This simplifies the procedure, because the calculation of the optimal rotation angle is not needed anymore.

Intensity Stereo is a lossy technique, because the phase information of the left and right channels at higher frequencies is not retained, resulting in a loss of spatial information. This makes it suitable for low bit rate coders, because the loss of spatial information is considered to be less annoying than other coding artifacts that can be created at low bit rates. However, this also means that Intensity Stereo is not suitable for high-quality coders.

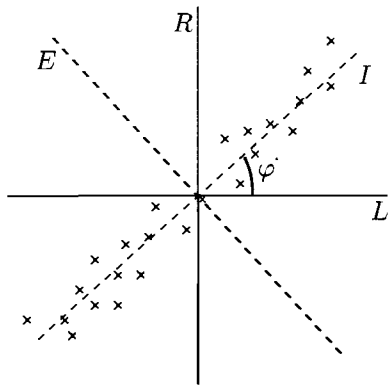

Figure 1.1: Lissajous plot as an example of rotation. The signal with dimensions L and R is rotated with angle φ to dimensions I and E.

1.1.3 Parametric Stereo

Parametric Stereo is a technique introduced by Faller et al. in 2001 [10]. 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. The monophonic signal is created by mixing down the stereo signal to a mono signal. An advantage of creating a mono signal is that a traditional mono audio coder can be used to code this signal. The parameters containing the auditory spatial information are called sound localization cues and they are extracted from the stereo signal. The generalized block diagram of a Parametric Stereo coder can be seen in Figure 1.2. The tildes in this figure denote quantized signals or parameters. We will use this notation throughout this thesis.

Currently, there exist two important Parametric Stereo coders, namely Binaural Cue Coding (BCC) and Optimum Coding of Stereo (OCS). 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.

BCC was presented by Faller et al. in [11] [12]. BCC extracts for every time and frequency partition three cues from the stereo signal. These cues are the Inter-Channel Level Difference (ICLD), the Inter-Channel Time Difference (ICTD) and the Inter-Channel Correlation (ICC). For stationary signals, the ICLD is given by

$$ICLD[b] = \lim_{l\to\infty} 10\log_{10}\left(\frac{\sum_{k=-l}^{l} x_{2,b}^2(k)}{\sum_{k=-l}^{l} x_{1,b}^2(k)}\right), \tag{1.2}$$

where b and k are the frequency subband and the time index, respectively. The ICTD is given by

$$ICTD[b] = \arg\max_d\{\overline{\Phi}_{12,b}(d)\}, \tag{1.3}$$

with the normalized cross-correlation $\overline{\Phi}_{12,b}(d)$ defined as

$$\overline{\Phi}_{12,b}(d) = \lim_{l\to\infty} \frac{\sum_{k=-l}^{l} x_{1,b}(k)x_{2,b}(k+d)}{\sqrt{\sum_{k=-l}^{l} x_{1,b}^2(k)\sum_{k=-l}^{l} x_{2,b}^2(k)}}. \tag{1.4}$$

The ICC is given by

$$ICC[b] = \max_d \left|\overline{\Phi}_{12,b}(d)\right|. \tag{1.5}$$

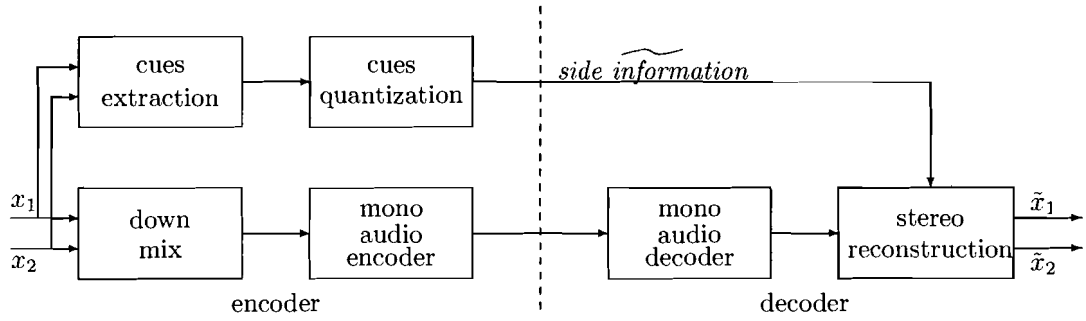

Figure 1.2: Generalized block diagram of a Parametric Stereo coder. In the encoder, sound localization cues are extracted from the input stereo signal, before it is mixed down to a mono signal. In the decoder, the stereo signal is generated by applying the sound localization cues to the mono signal.

The mono downmix is created by adding the two channels of the stereo signal together.

OCS was developed by Breebaart et al. [13] [14] within Philips. OCS extracts for every time and frequency partition four cues from the stereo signal. These cues are the Interchannel Intensity Difference (IID), the Interchannel Phase Difference (IPD), the Interchannel Coherence (IC) and the Overall Phase Difference (OPD). The IID is given by

$$IID[b] = 10 \log_{10} \frac{\sum_k X_1[k] X_1^*[k]}{\sum_k X_2[k] X_2^*[k]}, \tag{1.6}$$

where b and X are the frequency subband and the FFT of the stereo signal, respectively, and the summations extend over all k's within a subband b. The IPD is given by

$$IPD[b] = \angle\left(\sum_k X_1[k] X_2^*[k]\right). \tag{1.7}$$

The IC is given by

$$IC[b] = \frac{\left|\sum_k X_1[k] X_2^*[k]\right|}{\sqrt{\left(\sum_k X_1[k] X_1^*[k]\right)\left(\sum_k X_2[k] X_2^*[k]\right)}}. \tag{1.8}$$

The mono downmix M, created as a linear combination of the two channels of the stereo signal, is given by

$$M[k] = w_1 X_1[k] + w_2 X_2[k], \tag{1.9}$$

where w_1 and w_2 are derived from the IID, IPD and IC. The last cue, OPD, is given by

$$OPD[b] = \angle\left(\sum_k X_1[k] M^*[k]\right). \tag{1.10}$$

The cues are quantized according to perceptual criteria. In addition, the IPD and OPD cues are not transmitted for subbands higher than about 2 kHz, because phase differences are not relevant at high frequencies. This leads to an estimated bit rate of the side information of 7.7 kbps. This bit rate can be decreased to about 1.5 kbps by making the following changes:

- Reducing the number of frequency bands;
- Not transmitting the IPD and OPD cues;
- Increasing the quantization step sizes;
- Decreasing the parameter update rate.

Parametric Stereo is a lossy technique, 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 Parametric Stereo very suitable for low bit rate coding.

1.1.4 Stereo Linear Prediction

Linear Prediction [15] has been widely used as a technique for speech compression for more than thirty years [3]. This is because the behavior of the vocal tract can be very well modelled by the synthesis filter of a linear predictive coder. Despite the fact that audio signals are generally not produced by a vocal tract, Linear Prediction has also been used as a technique in audio coding [16] [17].

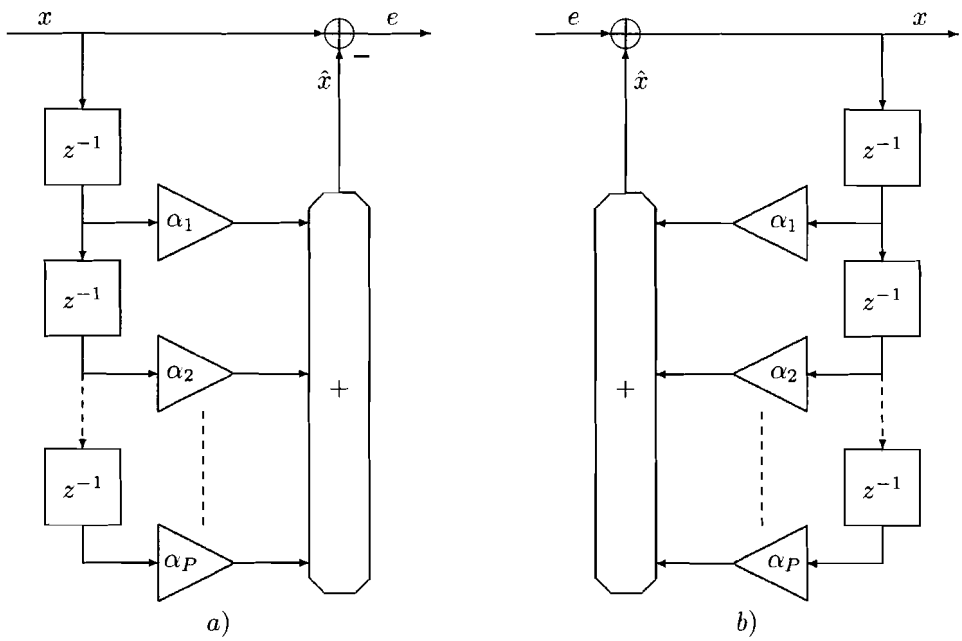

Figure 1.3: Linear Prediction scheme: a) analysis filter, b) synthesis filter. The analysis filter produces the error signal e from signal x by applying the prediction coefficients α_k. The synthesis filter reconstructs x from e and the α's.

Linear Prediction tries to remove redundancies within a signal by estimating the current value of a signal as a linear combination of its previous values. This is possible, because successive samples within a signal are often highly correlated. The prediction $\hat{x}$ of signal x is given by

$$\hat{x}[n] = \sum_{k=1}^{P} x[n-k]\alpha_k, \tag{1.11}$$

where P and α_k are the prediction order and the prediction coefficients, respectively. The hat denotes the prediction of a signal. We will use this notation throughout this thesis. The prediction error e is defined as the difference between the original signal and the predicted signal

$$e[n] = x[n] - \hat{x}[n] = x[n] - \sum_{k=1}^{P} x[n-k]\alpha_k. \tag{1.12}$$

The Linear Prediction analysis filter produces the error signal e from signal x by applying the prediction coefficients α_k. If the optimal α's and a sufficiently high prediction order P are used, then e will have an approximately flat spectral envelope. The Linear Prediction synthesis filter reconstructs x from e and the α's. This reconstruction is perfect in the absence of any parameter and signal quantization. The schemes of the analysis and synthesis filters can be seen in Figure 1.3.

The transfer function F of the Linear Prediction analysis filter is given by

$$F(z) = 1 - \sum_{k=1}^{P} \alpha_k z^{-k}. \tag{1.13}$$

The transfer function H of the Linear Prediction synthesis filter is the inverse of F

$$H(z) = \frac{1}{F(z)} = \frac{1}{1 - \sum_{k=1}^{P} \alpha_k z^{-k}}. \tag{1.14}$$

To have a stable synthesis filter, the poles of H must be within the unit circle. This means that the zeros of F must be within the unit circle.

The optimal prediction coefficients are calculated by minimizing the mean squared prediction error σ_e^2, given by

$$\sigma_e^2 = \sum_n e^2[n] = \sum_n (x[n] - \hat{x}[n])^2, \tag{1.15}$$

where the sum extends from minus infinity to infinity. The minimization of σ_e^2 with respect to the prediction coefficients is obtained by setting

$$\frac{\partial \sigma_e^2}{\partial \alpha_l} = 0, \qquad l = 1, \ldots, P. \tag{1.16}$$

This leads to the following Yule-Walker equations

$$\sum_{k=1}^{P} \rho_{l-k}\alpha_k = \rho_l, \qquad l = 1, \ldots, P, \tag{1.17}$$

where ρ is the auto-correlation function of the input signal, given by

$$\rho_k = \sum_n x[n-k]x[n]. \tag{1.18}$$

Equation (1.17) can be written in matrix form as

$$\mathbf{Q}\underline{\alpha} = \underline{p}, \tag{1.19}$$

with the auto-correlation matrix $\mathbf{Q}$ being a $P \times P$ Toeplitz matrix, given by

$$\mathbf{Q} = \begin{pmatrix} \rho_0 & \rho_{-1} & \rho_{-2} & \cdots & \rho_{-P+1} \\ \rho_1 & \rho_0 & \rho_{-1} & \cdots & \rho_{-P+2} \\ \rho_2 & \rho_1 & \rho_0 & \cdots & \rho_{-P+3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \rho_{P-1} & \rho_{P-2} & \rho_{P-3} & \cdots & \rho_0 \end{pmatrix}, \tag{1.20}$$

which is also a symmetric matrix because $\rho_i = \rho_{-i}$. The vector $\underline{\alpha}$ is given by

$$\underline{\alpha} = \begin{pmatrix} \alpha_1 & \alpha_2 & \alpha_3 & \cdots & \alpha_P \end{pmatrix}^T, \tag{1.21}$$

and the vector $\underline{p}$ is given by

$$\underline{p} = \begin{pmatrix} \rho_1 & \rho_2 & \rho_3 & \cdots & \rho_P \end{pmatrix}^T. \tag{1.22}$$

The optimal prediction coefficients can now be calculated by solving (1.19).

Linear Prediction only removes intra-channel correlations, by using auto-prediction. However, since audio signals usually consist of at least two channels that can be highly correlated, it makes sense to remove inter-channel correlations as well. Therefore, Stereo Linear Prediction, a technique that tries to exploit these inter-channel correlations by using some kind of cross-prediction, can be used for stereo signals.

Stereo linear predictive coders can be divided into two categories. Coders in the first category, conveniently named pseudo-stereo linear predictive coders, try to remove inter-channel correlations without using full cross-prediction. This means that in addition to the auto-predictors, which are also used by linear predictive coders, only one cross-predictor is used, or no explicit cross-predictors are used at all. Coders in the second category, conveniently named full-stereo linear predictive coders, try to remove inter-channel correlations by using full cross-prediction. This means that two cross-predictors are used in addition to the auto-predictors.

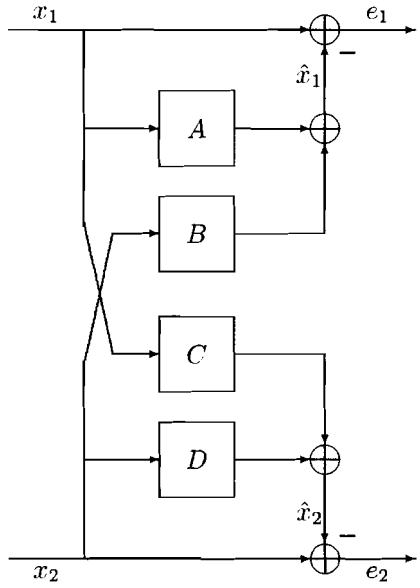

Figure 1.4: Stereo Linear Prediction scheme. It uses two auto-predictors (A and D) and two cross-predictors (B and C).

In 1997 and 1998, two different pseudo-stereo linear predictive coders were proposed. Both of them construct a mono signal from the stereo signal, so that a linear predictive coder can be used for the resulting mono signal. However, the mono signals are constructed in a totally different way. Härmä et al. [18] proposed to combine the left and right channel of a stereo signal into a complex signal. The resulting complex signal is then a linear combination of the left and right channel. Ikeda et al. [19] proposed to use stereo signal interleaving to combine the left and right channel of a stereo signal into a mono signal. This means that the left and right channel samples are alternately interleaved and arranged in a mono signal sequence. Another type of pseudo-stereo linear predictive coder, proposed by Fuchs [20], Mary et al. [21] and Moriya et al. [22], uses only one cross-predictor to exploit inter-channel correlations. The cross-prediction is then done either from the left channel to the right channel or vice versa.

The first full-stereo linear predictive coder was proposed by Cambridge et al. in 1993 [23] as a lossless audio coder that uses two auto-predictors (A and D) and two cross-predictors (B and C). This coder can be seen in Figure 1.4. The predicted value of a sample in each channel is calculated as a linear sum not only of the past samples from the same channel but also of the past samples from the other channel, according to

$$\begin{aligned} \hat{x}_1[n] &= \textstyle\sum_{k=1}^{P_a} x_1[n-k]a_k + \sum_{k=1}^{P_b} x_2[n-k]b_k \\ \hat{x}_2[n] &= \textstyle\sum_{k=0}^{P_c} x_1[n-k]c_k + \sum_{k=1}^{P_d} x_2[n-k]d_k \end{aligned} , \tag{1.23}$$

where $\hat{x}_1$ and $\hat{x}_2$ are the predicted signals for the left and right channel, respectively, a_k and d_k are the auto-prediction coefficients, P_a and P_d are the orders of the auto-predictors, b_k and c_k are the cross-prediction coefficients, and P_b and P_c are the orders of the cross-predictors.

It can be seen from (1.23) that the proposed Stereo Linear Prediction scheme is not symmetrical, because $\hat{x}_2$ is based on the current sample of x_1. This means that the prediction coefficients associated with one output channel must be simultaneously optimized for the auto- and the cross-predictor, but have to be separately optimized per channel. This is also reflected in the sense that the optimal prediction coefficients lead to two separate Yule-Walker equations.

Cambridge et al. [23] compared the prediction gain that can be achieved with a stereo predictor with the prediction gain that can be achieved with a pair of mono predictors with the same total number of taps as the stereo predictor. They showed that for small prediction orders, mono

predictors gave the best result, because there was more to be gained from auto-correlation than from cross-correlation. However, when the prediction order was high, using a stereo predictor (thus exploiting cross-correlation at the expense of auto-prediction) seemed to increase the prediction gain. Since it was not known how the stereo prediction coefficients could be quantized, a comparison between the bit rates of a stereo predictor and a pair of mono predictors with the same total number of taps could be made if only backward Linear Prediction was used. With backward Linear Prediction, the prediction coefficients are estimated from the past decoded signal. The results indicated that the bit rates associated with the error signals only, were slightly lower when Stereo Linear Prediction was used than when Linear Prediction was used.

Two years later, Fuchs [24] suggested to use backward Stereo Linear Prediction to improve MPEG Layer 2 and 3 audio coding. The idea was to apply backward Stereo Linear Prediction to the narrow-band spectral components resulting from the time-frequency mapping. His results showed that the extension of MPEG Layer 2 and 3 audio coding by backward Stereo Linear Prediction provided a significant step towards CD-like quality at 128 kpbs.

Liebchen [25] also proposed a lossless audio coder that uses Stereo Linear Prediction in 2002. His idea was to use an adaptive method for optimization of the prediction coefficients. This method has to be applied to both channels separately. First, the optimal order of the auto-predictor has to be determined by using the Levinson-Durbin algorithm. Then, the order of the cross-predictor is increased until the bit rate has reached a minimum value. Next, the procedure verifies whether a decrement of the order of the auto-predictor leads to an additional bit rate reduction. Finally, the optimal prediction coefficients can be calculated by applying Cholesky decomposition. His results showed that compared to conventional lossless audio coding techniques, the performance was improved for most stereo signals.

A year later, Ghido [26] proposed a generalized stereo decorrelation algorithm and optimal predictor for lossless audio compression, using Stereo Linear Prediction. However, his scheme was still not symmetrical.

In 2004, Garcia et al. [27] proposed a lossless audio coder that uses backward Stereo Linear Prediction. This coder still separates the left and right channel when the optimal prediction coefficients are calculated, but uses the multichannel Levinson-Durbin algorithm for the left channel. This is possible under the restriction of equal orders of the auto- and cross-predictor and because there is no direct feed-through branch in the left channel. The compression ratios obtained by this coder are comparable to the results obtained by two state-of-the-art lossless coders.

The stereo linear predictive coders just described have some common issues:

- The coders are not symmetrical, since interchanging the left and right channel will lead to another system behavior. Thus, a symmetrical structure would be conceptually better;
- The prediction coefficients are not jointly optimized for the left and right channel, which results in an increase in computational costs;
- It is not guaranteed that the synthesis filter is stable;
- The full-stereo coders are all lossless coders, thus not suitable for low bit rate coding.

Therefore, a new stereo audio coder was proposed in 2004 [1] [28] [29] to solve these problems, subject to the following criteria:

- Encoder and decoder form a system allowing perfect signal reconstruction in the absence of signal quantization and thus, near perfect reconstruction at the high bit rate end;
- The encoder constructs a main and a side signal similar to those provided by OCS, because this is advantageous for low bit rate coding purposes.

The proposed coder was called *Stereo Linear Predictive Coding of Audio* (SLP). It uses Stereo Linear Prediction and a single rotator. The encoder scheme of SLP is similar to the scheme

depicted in Figure 1.4 and is followed by a rotator. However, the predictions $\hat{x}_1$ and $\hat{x}_2$ of the left and right channel x_1 and x_2 are now given by

$$\begin{array}{l} \hat{x}_1[n] = \sum_{k=1}^{P_a} x_1[n-k]a_k + \sum_{k=1}^{P_b} x_2[n-k]b_k \\ \hat{x}_2[n] = \sum_{k=1}^{P_c} x_1[n-k]c_k + \sum_{k=1}^{P_d} x_2[n-k]d_k \end{array}, \tag{1.24}$$

where a_k and d_k are the auto-prediction coefficients, P_a and P_d are the orders of the auto-predictors, b_k and c_k are the cross-prediction coefficients, and P_b and P_c are the orders of the cross-predictors. Comparison of (1.23) and (1.24) reveals that SLP uses a symmetric stereo predictor structure. If we now take the special case $P_a = P_b = P_c = P_d$, the prediction coefficients can be jointly optimized for the left and right channel by using the block-Levinson algorithm. Stability of the synthesis filter is also guaranteed, because the auto-correlation method is used to calculate the optimal prediction coefficients [15] [28] [29].

The rotator produces the main signal m and side signal s from the error signals e_1 and e_2 with the matrix operation given by

$$\begin{pmatrix} m \\ s \end{pmatrix} = \begin{pmatrix} \cos(\varphi) & \sin(\varphi) \\ -\sin(\varphi) & \cos(\varphi) \end{pmatrix} \begin{pmatrix} e_1 \\ e_2 \end{pmatrix}, \tag{1.25}$$

where φ is the rotation angle. With the correct φ, the rotator removes correlations between the input signals at lag zero.

It was argued [28] that the proposed scheme might be a proper concept for stereo audio coding, because it combines several attractive features of known coding mechanisms.

1.2 Problem Description and Problem Statement

At Philips Research Europe - Eindhoven, within the Audio and Speech Signal Processing Cluster, which is a part of the Digital Signal Processing Group, one of the current research topics is the development of new techniques for audio coding, such as stereo and multichannel audio coding. This Masters thesis project is a continuation of the work on the SLP coder presented in [1]. It was shown there that the implemented system operated as expected. However, there are still some important issues with the SLP coder that need to be tackled. These issues are:

- Regularization (measures to handle all classes of input signals) of Stereo Linear Prediction and Rotation needs to be considered in more detail;
- SLP has to be made psycho-acoustically inspired by implementing such filters in its scheme;
- An efficient quantization technique for the prediction coefficients has to be developed;
- The main and side signal have to be quantized;
- The subjective quality and bit rate of SLP have to be determined and compared with existing state-of-the-art coding schemes.

This leads to the problem statement of this Masters thesis project. The objective of this project is to further enhance SLP. It includes the following tasks:

- Getting familiar with the Mono Linear Prediction techniques;
- Studying existing SLP schemes;
- Implementing and testing input signal conditioning for the case of mono inputs;
- Designing, implementing and testing a regularization technique for the rotator;
- Designing, implementing and testing a quantization scheme for the main and side signal;

- Inclusion of psycho-acoustically inspired filters in the SLP scheme;
- Testing the scheme after removing the side signal;
- Testing the overall scheme using formal listening tests.

1.3 Outline of the Thesis

The enhanced version of the stereo audio coder SLP is described in Chapters 2, 3 and 4. This starts with a detailed description of the proposed coding scheme in Chapter 2. Chapter 3 considers Laguerre-based Pure Stereo Linear Prediction and how Laguerre filters can be incorporated into the proposed coding scheme to make it psycho-acoustically inspired. The description of SLP is concluded in Chapter 4, where it is described how the prediction coefficients and the main and side signal that are constructed in the proposed coding scheme can be quantized. The performance of the SLP coder is evaluated in Chapter 5. This evaluation includes an estimation of the bit rate, the results of a formal listening test to determine the subjective quality of the coded material, and an evaluation of the coding delay, the computational complexity and the scalability of the coder. The thesis finishes with conclusions about the state of SLP and recommendations for further research on SLP in Chapter 6.

Chapter 2

Stereo Linear Predictive Coding of Audio

Our proposal for a stereo audio coder, which we will call SLP, is described in this chapter. It starts with a description of the proposed coding scheme. Thereafter, the two most important blocks in this scheme, Stereo Linear Prediction and Rotation, are described in more detail. The chapter continues with regularization of these two blocks and concludes with a short summary.

2.1 The SLP Coding Scheme

The proposed SLP coding scheme can be seen in Figure 2.1. It consists of two rotators and a Stereo Linear Prediction analysis filter in the encoder and two inverse rotators and a Stereo Linear Prediction synthesis filter in the decoder.

The left and right input signals, x_1 and x_2, first go to the pre-rotator. This pre-rotator uses a fixed rotation angle of $\pi/4$. The reason for this will be explained in Section 2.4.2. The rotated input signals, r_1 and r_2, then go to the Stereo Linear Prediction analysis filter. This filter makes the predictions $\hat{r}_1$ and $\hat{r}_2$ from the rotated input signals with auto-predictors A and D and cross-predictors B and C and subtracts these predictions from the rotated input signals to get the error signals e_1 and e_2. The error signals are fed to the second rotator that produces the main and side signal m and s. This rotator uses a variable rotation angle. The analysis filter removes the auto- and cross-correlations from the input signals, except the cross-correlation for lag zero. This cross-correlation is minimized by the second rotator. Thus, the encoder results in two spectrally flat, uncorrelated signals m and s. The main and side signal are then quantized to get $\tilde{m}$ and $\tilde{s}$ and these are sent to the decoder, along with the quantized parameters. These parameters include the prediction coefficients and the rotation angle of R_2.

The decoder performs the inverse operations of the encoder. This means that the received signals $\tilde{m}$ and $\tilde{s}$ are first inverse rotated to reconstruct the error signals $\tilde{e}_1$ and $\tilde{e}_2$. These signals then go to the Stereo Linear Prediction synthesis filter. This filter uses the predictors A, B, C and D in a feedback loop and has as output the rotated input signals $\tilde{r}_1$ and $\tilde{r}_2$. Finally, these signals are inverse pre-rotated to reconstruct the input signals $\tilde{x}_1$ and $\tilde{x}_2$.

2.2 Stereo Linear Prediction

The Stereo Linear Prediction block is described in this section. This starts with the Stereo Linear Prediction analysis filter and next, the Stereo Linear Prediction synthesis filter. It is also described how the optimal prediction coefficients that are used in both the analysis and synthesis filter can be calculated.

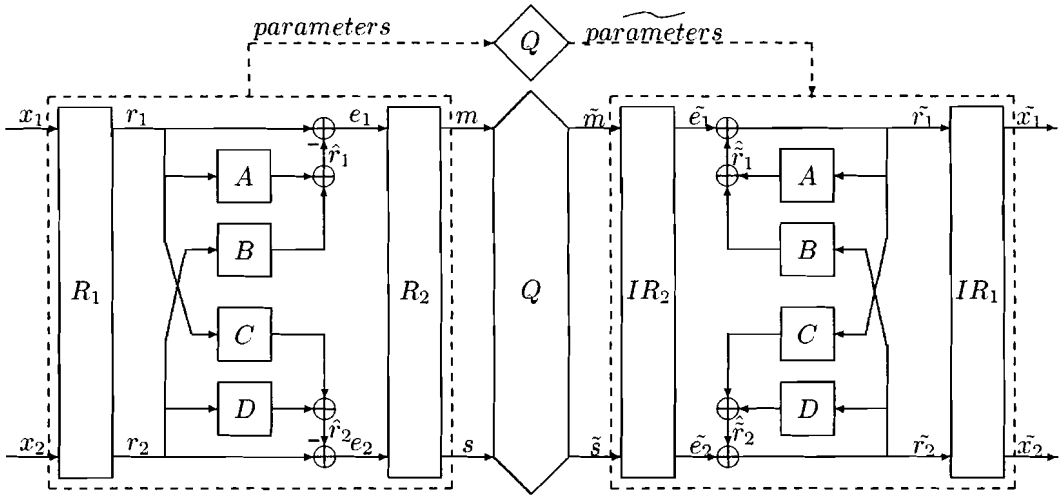

Figure 2.1: SLP coding scheme. It consists of two rotators and a Stereo Linear Prediction analysis filter in the encoder and two inverse rotators and a Stereo Linear Prediction synthesis filter in the decoder.

2.2.1 Analysis Filter

The scheme of the analysis filter can be seen in Figure 2.2. The analysis filter tries to remove the auto- and cross-correlations from the input signals r_1 and r_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 a stereo signal x

$$\underline{x}[n] = \begin{pmatrix} x_1[n] & x_2[n] \end{pmatrix}. \tag{2.1}$$

The predictions $\hat{\underline{r}}$ of the input signals $\underline{r}$ are now given by

$$\hat{\underline{r}}[n] = \sum_{k=1}^{P} \underline{y}_k[n]\mathbf{A_k}, \tag{2.2}$$

where P is the prediction order, where $\underline{y}_k$ is the k-times delayed input signal $\underline{r}$, given by

$$\underline{y}_k[n] = \underline{r}[n-k] \tag{2.3}$$

and with $\mathbf{A_k}$ the kth order prediction coefficient matrix, given by

$$\mathbf{A_k} = \begin{pmatrix} a_k & c_k \\ b_k & d_k \end{pmatrix}, \tag{2.4}$$

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

$$\underline{e}[n] = \underline{r}[n] - \hat{\underline{r}}[n] = \underline{r}[n] - \sum_{k=1}^{P} \underline{y}_k[n]\mathbf{A_k}. \tag{2.5}$$

This leads to the transfer matrix $\mathbf{F}$ of the analysis filter, given by

$$\mathbf{F}(z) = \begin{pmatrix} 1 - A(z) & -C(z) \\ -B(z) & 1 - D(z) \end{pmatrix}, \tag{2.6}$$

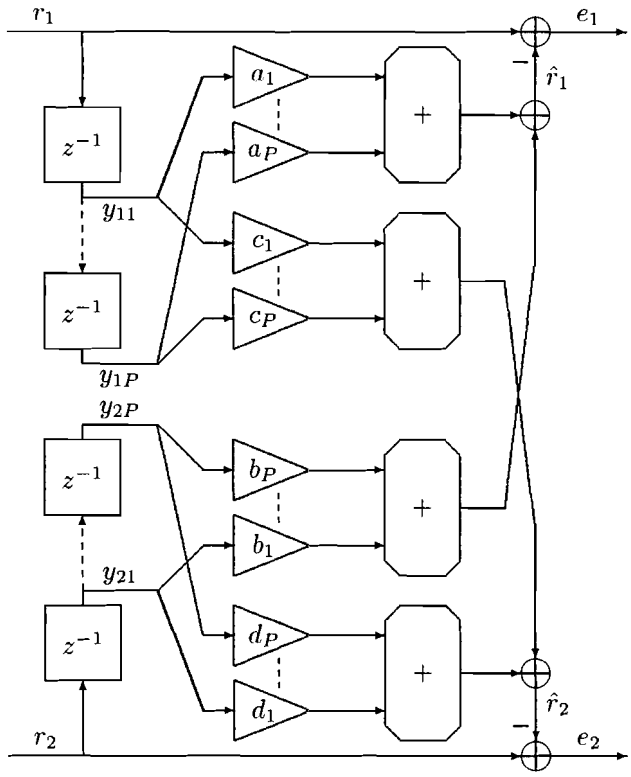

Figure 2.2: Stereo Linear Prediction analysis filter.

with the transfer functions of the individual predictors, for example A, defined by

$$A(z) = \sum_{k=1}^{P} z^{-k} a_k. \tag{2.7}$$

2.2.2 Synthesis Filter

The scheme of the synthesis filter can be seen in Figure 2.3. The synthesis filter performs the inverse operations of the analysis filter, thus it reconstructs the signals r_1 and r_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 $\mathbf{H}$ of the synthesis filter is therefore given by the inverse of the transfer matrix of the analysis filter

$$\mathbf{H}(z) = \{\mathbf{F}(z)\}^{-1} = \frac{1}{\det(\mathbf{F}(z))} \begin{pmatrix} 1 - D(z) & C(z) \\ B(z) & 1 - A(z) \end{pmatrix}, \tag{2.8}$$

with the determinant of $\mathbf{F}$ given by

$$\det(\mathbf{F}(z)) = (1 - A(z))(1 - D(z)) - B(z)C(z). \tag{2.9}$$

Stability of the SLP scheme is determined by the stability of the synthesis filter. It is clear from (2.8) that all the poles of $\mathbf{H}$, which determine the stability, are determined by $\det(\mathbf{F}(z))$. Thus, we can conclude that $\mathbf{H}$ is a stable filter if $\frac{1}{\det(\mathbf{F}(z))}$ is a stable filter. It is discussed in the next section how this can be ensured.

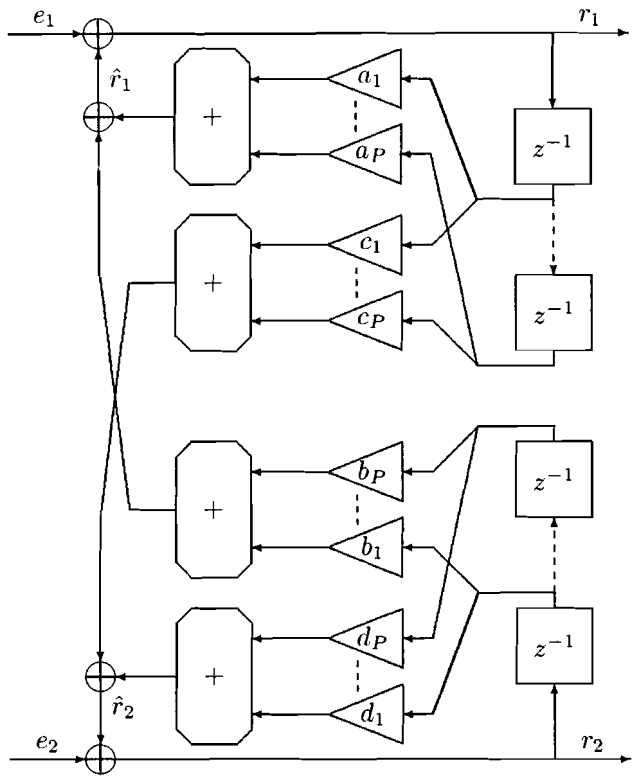

Figure 2.3: Stereo Linear Prediction synthesis filter.

2.2.3 Calculation of the Optimal Prediction Coefficients

We will now derive how the optimal prediction coefficients, which are used by both the analysis filter and the synthesis filter, can be calculated. Furthermore, we will show how the prediction coefficients can be made more robust.

The optimal prediction coefficients are calculated by minimizing the mean squared prediction errors σ_{e1}^2 and σ_{e2}^2 of the analysis filter, given by

$$\sigma_{e_1}^2 = \sum_n [e_1^2[n]] = \sum_n [r_1[n] - \hat{r}_1[n]]^2 = \sum_n \left[r_1[n] - \sum_{k=1}^{P} y_{k1}[n] a_k - \sum_{k=1}^{P} y_{k2}[n] b_k \right]^2, \quad (2.10)$$

$$\sigma_{e_2}^2 = \sum_n [e_2^2[n]] = \sum_n [r_2[n] - \hat{r}_2[n]]^2 = \sum_n \left[r_2[n] - \sum_{k=1}^{P} y_{k1}[n] c_k - \sum_{k=1}^{P} y_{k2}[n] d_k \right]^2, \quad (2.11)$$

where the sum extends from minus infinity to infinity. This also means that the auto- and cross-correlations in the input signals are being minimized.

We start with minimizing the mean squared error in the left channel. The minimum of (2.10) with respect to the prediction coefficients is obtained by setting

$$\frac{\partial \sigma_{e1}^2}{\partial a_l} = 0, \qquad l = 1, \ldots, P, \quad (2.12)$$

$$\frac{\partial \sigma_{e1}^2}{\partial b_m} = 0, \qquad m = 1, \ldots, P. \quad (2.13)$$

We first look at (2.12)

$$\frac{\partial \sigma_{e1}^2}{\partial a_l} = \frac{\partial \left\{ \sum_n \left[r_1[n] - \sum_{k=1}^{P} y_{k1}[n] a_k - \sum_{k=1}^{P} y_{k2}[n] b_k \right]^2 \right\}}{\partial a_l}$$
$$= \sum_n \left[y_{l1}[n] 2 \left(r_1[n] - \sum_{k=1}^{P} y_{k1}[n] a_k - \sum_{k=1}^{P} y_{k2}[n] b_k \right) \right] = 0, \tag{2.14}$$

which leads to

$$\sum_n \left[\sum_{k=1}^{P} y_{l1}[n] y_{k1}[n] a_k + \sum_{k=1}^{P} y_{l1}[n] y_{k2}[n] b_k \right] = \sum_n [y_{l1}[n] r_1[n]], \tag{2.15}$$

or

$$\sum_{k=1}^{P} \left[\sum_n y_{l1}[n] y_{k1}[n] a_k \right] + \sum_{k=1}^{P} \left[\sum_n y_{l1}[n] y_{k2}[n] b_k \right] = \sum_n [y_{l1}[n] r_1[n]]. \tag{2.16}$$

If we now use $\rho_{pq}(i,j)$ as a shorthand for the correlation functions

$$\rho_{pq}(i,j) = \sum_n y_{ip}[n] y_{jq}[n] \tag{2.17}$$

and with $y_{01}[n] = r_1[n]$, then (2.16) can be written as

$$\sum_{k=1}^{P} \rho_{11}(l,k) a_k + \sum_{k=1}^{P} \rho_{12}(l,k) b_k = \rho_{11}(l,0), \qquad l = 1, \ldots, P. \tag{2.18}$$

Now, we look at (2.13) and this leads in a similar way as described above to the following equations

$$\sum_{k=1}^{P} \rho_{21}(m,k) a_k + \sum_{k=1}^{P} \rho_{22}(m,k) b_k = \rho_{21}(m,0), \qquad m = 1, \ldots, P. \tag{2.19}$$

We now want to minimize the mean squared error in the right channel. The minimum of (2.11) with respect to the prediction coefficients is obtained by setting

$$\frac{\partial \sigma_{e2}^2}{\partial c_l} = 0, \qquad l = 1, \ldots, P, \tag{2.20}$$
$$\frac{\partial \sigma_{e2}^2}{\partial d_m} = 0, \qquad m = 1, \ldots, P. \tag{2.21}$$

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

$$\sum_{k=1}^{P} \rho_{11}(l,k) c_k + \sum_{k=1}^{P} \rho_{12}(l,k) d_k = \rho_{12}(l,0), \qquad l = 1, \ldots, P, \tag{2.22}$$
$$\sum_{k=1}^{P} \rho_{21}(m,k) c_k + \sum_{k=1}^{P} \rho_{22}(m,k) d_k = \rho_{22}(m,0), \qquad m = 1, \ldots, P. \tag{2.23}$$

Equations (2.18), (2.19), (2.22) and (2.23) form the Yule-Walker equations, which can be written in matrix form as

$$\begin{pmatrix} \rho_{11}(1,1) & \cdots & \rho_{11}(1,P) & \rho_{12}(1,1) & \cdots & \rho_{12}(1,P) \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \rho_{11}(P,1) & \cdots & \rho_{11}(P,P) & \rho_{12}(P,1) & \cdots & \rho_{12}(P,P) \\ \rho_{21}(1,1) & \cdots & \rho_{21}(1,P) & \rho_{22}(1,1) & \cdots & \rho_{22}(1,P) \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \rho_{21}(P,1) & \cdots & \rho_{21}(P,P) & \rho_{22}(P,1) & \cdots & \rho_{22}(P,P) \end{pmatrix} \begin{pmatrix} a_1 & c_1 \\ \vdots & \vdots \\ a_P & c_P \\ b_1 & d_1 \\ \vdots & \vdots \\ b_P & d_P \end{pmatrix}$$
$$= \begin{pmatrix} \rho_{11}(1,0) & \rho_{12}(1,0) \\ \vdots & \vdots \\ \rho_{11}(P,0) & \rho_{12}(P,0) \\ \rho_{21}(1,0) & \rho_{22}(1,0) \\ \vdots & \vdots \\ \rho_{21}(P,0) & \rho_{22}(P,0) \end{pmatrix}. \tag{2.24}$$

If we now define $\mathbf{R_{i-j}}$ as a 2×2 block matrix with

$$\mathbf{R_{i-j}} = \begin{pmatrix} \rho_{11}(i,j) & \rho_{12}(i,j) \\ \rho_{21}(i,j) & \rho_{22}(i,j) \end{pmatrix}, \tag{2.25}$$

and use $\rho_{pq}(i,j) = \rho_{pq}(i+1, j+1)$ and $\rho_{pq}(i,j) = \rho_{pq}(j,i)$, then we can rearrange the rows and columns of (2.24), which gives

$$\mathbf{QA} = \mathbf{P}, \tag{2.26}$$

with the correlation matrix $\mathbf{Q}$ a block Toeplitz matrix given by

$$\mathbf{Q} = \begin{pmatrix} \mathbf{R_0} & \mathbf{R_{-1}} & \mathbf{R_{-2}} & \cdots & \mathbf{R_{-P+1}} \\ \mathbf{R_1} & \mathbf{R_0} & \mathbf{R_{-1}} & \cdots & \mathbf{R_{-P+2}} \\ \mathbf{R_2} & \mathbf{R_1} & \mathbf{R_0} & \cdots & \mathbf{R_{-P+3}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{R_{P-1}} & \mathbf{R_{P-2}} & \mathbf{R_{P-3}} & \cdots & \mathbf{R_0} \end{pmatrix}, \tag{2.27}$$

$\mathbf{A}$ the prediction coefficient matrix given by

$$\mathbf{A} = \begin{pmatrix} \mathbf{A_1} & \mathbf{A_2} & \mathbf{A_3} & \cdots & \mathbf{A_P} \end{pmatrix}^T, \tag{2.28}$$

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

$$\mathbf{P} = \begin{pmatrix} \mathbf{R_1} & \mathbf{R_2} & \mathbf{R_3} & \cdots & \mathbf{R_P} \end{pmatrix}^T, \tag{2.29}$$

with $\mathbf{R_{-k}} = \mathbf{R_k}^T$. The optimal prediction coefficients can now be calculated by solving (2.26).

Levinson developed a fast algorithm for inversion of a symmetric Toeplitz matrix. This method generalizes to the nonsymmetric and multichannel case, where block matrices and thus matrix operations appear. We can use this block-Levinson algorithm [30] [31] [32] to solve (2.26). The following derivation of the block-Levinson algorithm follows that in [33]. However, it is adapted here for block matrices.

The linear Toeplitz problem is now rewritten as

$$\sum_{j=1}^{P} \mathbf{R_{i-j}} \mathbf{A_j} = \mathbf{y_i}, \qquad i = 1, \ldots, P, \tag{2.30}$$

where the prediction coefficient matrices $\mathbf{A_j}$ are the unknowns to be solved for and with $\mathbf{y_i} = \mathbf{R_i}$. The block-Levinson algorithm recursively solves the M-dimensional Toeplitz problem

$$\sum_{j=1}^{M} \mathbf{R_{i-j}} \mathbf{A_j^{(M)}} = \mathbf{y_i}, \qquad i = 1, \ldots, M, \tag{2.31}$$

for $M = 1, 2, \ldots$ until $M = P$, which is the desired result, is reached. The matrix $\mathbf{A_j^{(M)}}$ is the result at the Mth stage and becomes the desired solution only when P is reached. In following a recursion from step M to $M+1$, the developing solution $\mathbf{A^{(M)}}$ changes from (2.31) to

$$\sum_{j=1}^{M} \mathbf{R_{i-j}} \mathbf{A_j^{(M+1)}} + \mathbf{R_{i-(M+1)}} \mathbf{A_{M+1}^{(M+1)}} = \mathbf{y_i}, \qquad i = 1, \ldots, M+1. \tag{2.32}$$

Eliminating $\mathbf{y_i}$ gives

$$\sum_{j=1}^{M} \mathbf{R_{i-j}} \left(\mathbf{A_j^{(M)}} - \mathbf{A_j^{(M+1)}} \right) \left(\mathbf{A_{M+1}^{(M+1)}} \right)^{-1} = \mathbf{R_{i-(M+1)}}, \qquad i = 1, \ldots, M \tag{2.33}$$

and by letting $i \rightarrow M+1-i$ and $j \rightarrow M+1-j$

$$\sum_{j=1}^{M} \mathbf{R_{j-i}} \mathbf{G_j^{(M)}} = \mathbf{R_{-i}}, \tag{2.34}$$

with

$$\mathbf{G_j^{(M)}} = \left(\mathbf{A_{M+1-j}^{(M)}} - \mathbf{A_{M+1-j}^{(M+1)}} \right) \left(\mathbf{A_{M+1}^{(M+1)}} \right)^{-1}, \tag{2.35}$$

or put in another way

$$\mathbf{A_{M+1-j}^{(M+1)}} = \mathbf{A_{M+1-j}^{(M)}} - \mathbf{G_j^{(M)}} \mathbf{A_{M+1}^{(M+1)}}, \qquad j = 1, \ldots, M. \tag{2.36}$$

This means that if we can use recursion to find the order M quantities $\mathbf{A^{(M)}}$ and $\mathbf{G^{(M)}}$ and the single order $M+1$ quantity $\mathbf{A_{M+1}^{(M+1)}}$, then all of the other $\mathbf{A_j^{(M+1)}}$ will follow. Luckily, $\mathbf{A_{M+1}^{(M+1)}}$ follows from (2.32) with $i = M+1$

$$\sum_{j=1}^{M} \mathbf{R_{M+1-j}} \mathbf{A_j^{(M+1)}} + \mathbf{R_0} \mathbf{A_{M+1}^{(M+1)}} = \mathbf{y_{M+1}}. \tag{2.37}$$

Since

$$\mathbf{G_{M+1-j}^{(M)}} = \left(\mathbf{A_j^{(M)}} - \mathbf{A_j^{(M+1)}} \right) \left(\mathbf{A_{M+1}^{(M+1)}} \right)^{-1}, \tag{2.38}$$

we can substitute the previous order quantities in $\mathbf{G}$ to get $\mathbf{A_j^{(M+1)}}$. This results in

$$\mathbf{A_{M+1}^{(M+1)}} = \left[\sum_{j=1}^{M} \mathbf{R_{M+1-j}} \mathbf{G_{M+1-j}^{(M)}} - \mathbf{R_0} \right]^{-1} \left[\sum_{j=1}^{M} \mathbf{R_{M+1-j}} \mathbf{A_j^{(M)}} - \mathbf{y_{M+1}} \right]. \tag{2.39}$$

The recursion relation for G follows from the left-hand solutions, which we will call $\mathbf{B_i}$, to the original Toeplitz problem. With the left-hand solutions, we deal with the following equations

$$\sum_{j=1}^{M} \mathbf{R_{j-i}} \mathbf{B_j^{(M)}} = \mathbf{y_i}, \qquad i = 1, \ldots, M. \tag{2.40}$$

The same sequence of operations on this set leads to

$$\sum_{j=1}^{M} \mathbf{R}_{\mathbf{i-j}} \mathbf{H}_{\mathbf{j}}^{(\mathrm{M})} = \mathbf{R}_{\mathbf{i}}, \tag{2.41}$$

with

$$\mathbf{H}_{\mathbf{j}}^{(\mathrm{M})} = \left(\mathbf{B}_{\mathbf{M+1-j}}^{(\mathrm{M})} - \mathbf{B}_{\mathbf{M+1-j}}^{(\mathrm{M+1})}\right) \left(\mathbf{B}_{\mathbf{M+1}}^{(\mathrm{M+1})}\right)^{-1}. \tag{2.42}$$

It can be seen from (2.41) that the $\mathbf{H_j}$ satisfy exactly the same equation as the $\mathbf{A_j}$, except for the substitution $\mathbf{y_i} \rightarrow \mathbf{R_i}$ on the right-hand side. Therefore, we can quickly deduce from (2.39) that

$$\mathbf{H}_{\mathbf{M+1}}^{(\mathrm{M+1})} = \left[\sum_{j=1}^{M} \mathbf{R}_{\mathbf{M+1-j}} \mathbf{G}_{\mathbf{M+1-j}}^{(\mathrm{M})} - \mathbf{R_0}\right]^{-1} \left[\sum_{j=1}^{M} \mathbf{R}_{\mathbf{M+1-j}} \mathbf{H}_{\mathbf{j}}^{(\mathrm{M})} - \mathbf{R}_{\mathbf{M+1}}\right]. \tag{2.43}$$

Similarly, the $\mathbf{G_j}$ satisfy the same equation as the $\mathbf{B_j}$, except for the substitution $\mathbf{y_i} \rightarrow \mathbf{R_{-i}}$. This leads to

$$\mathbf{G}_{\mathbf{M+1}}^{(\mathrm{M+1})} = \left[\sum_{j=1}^{M} \mathbf{R}_{\mathbf{j-M-1}} \mathbf{H}_{\mathbf{M+1-j}}^{(\mathrm{M})} - \mathbf{R_0}\right]^{-1} \left[\sum_{j=1}^{M} \mathbf{R}_{\mathbf{j-M-1}} \mathbf{G}_{\mathbf{j}}^{(\mathrm{M})} - \mathbf{R}_{\mathbf{-M-1}}\right]. \tag{2.44}$$

The same substitution can be applied to (2.36) and its partner for $\mathbf{B}$ to get the following equations

$$\mathbf{G}_{\mathbf{j}}^{(\mathrm{M+1})} = \mathbf{G}_{\mathbf{j}}^{(\mathrm{M})} - \mathbf{H}_{\mathbf{M+1-j}}^{(\mathrm{M})} \mathbf{G}_{\mathbf{M+1}}^{(\mathrm{M+1})}, \tag{2.45}$$

$$\mathbf{H}_{\mathbf{j}}^{(\mathrm{M+1})} = \mathbf{H}_{\mathbf{j}}^{(\mathrm{M})} - \mathbf{G}_{\mathbf{M+1-j}}^{(\mathrm{M})} \mathbf{H}_{\mathbf{M+1}}^{(\mathrm{M+1})}. \tag{2.46}$$

We can now start the recursive procedure with the initial values

$$\mathbf{A}_{\mathbf{1}}^{(1)} = \{\mathbf{R_0}\}^{-1} \mathbf{y_1}, \tag{2.47}$$

$$\mathbf{G}_{\mathbf{1}}^{(1)} = \{\mathbf{R_0}\}^{-1} \mathbf{R_{-1}}, \tag{2.48}$$

$$\mathbf{H}_{\mathbf{1}}^{(1)} = \{\mathbf{R_0}\}^{-1} \mathbf{R_1}. \tag{2.49}$$

At each stage M, we use (2.43) and (2.44) to find $\mathbf{H}_{\mathbf{M+1}}^{(\mathrm{M+1})}$ and $\mathbf{G}_{\mathbf{M+1}}^{(\mathrm{M+1})}$ and then (2.45) and (2.46) to find the other components of $\mathbf{H}^{(\mathrm{M+1})}$ and $\mathbf{G}^{(\mathrm{M+1})}$. The prediction coefficient matrices $\mathbf{A}^{(\mathrm{M+1})}$ can then be calculated with (2.39) and (2.36).

As mentioned already in Section 2.2.2, all the poles of the Stereo Linear Prediction synthesis filter are determined by $\det(\mathbf{F}(z))$. This means that the synthesis filter is stable if $\frac{1}{\det(\mathbf{F}(z))}$ is a stable filter. According to Whittle [30], 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. However, this is already the case (see (2.2)). An alternative proof of the stability of the synthesis filter can be found in [28] [29]. It is also shown in [29] that selecting unequal orders of the auto- and cross-predictors gives rise to stability problems.

Although all the poles of the synthesis filter are within the unit circle, some of them may be very close to it. To make the prediction coefficients more robust, a technique called Spectral Smoothing (SS), which is a known technique from speech coding [3], can be applied.

With SS in the one channel case, the calculated prediction coefficients α_k are replaced by α'_k, with

$$\alpha'_k = \gamma^k \alpha_k, \tag{2.50}$$

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

$$F'(z) = 1 - \sum_{k=1}^{P} \gamma^k a_k z^{-k} = 1 - \sum_{k=1}^{P} a_k \left(\frac{z}{\gamma}\right)^{-k} = F\left(\frac{z}{\gamma}\right) \tag{2.51}$$

and a new transfer function of the Linear Prediction synthesis filter

$$H'(z) = \frac{1}{F'(z)} = \frac{1}{1 - \sum_{k=1}^{P} a_k \left(\frac{z}{\gamma}\right)^{-k}} = H\left(\frac{z}{\gamma}\right). \tag{2.52}$$

SS 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.

SS can also be applied in the stereo case [1]. The transfer matrix $\mathbf{F}$ of the Stereo Linear Prediction analysis filter was given by

$$\mathbf{F}(z) = \begin{pmatrix} 1 - A(z) & -C(z) \\ -B(z) & 1 - D(z) \end{pmatrix}, \tag{2.53}$$

with the transfer functions of the individual predictors, for example A, defined by

$$A(z) = \sum_{k=1}^{P} z^{-k} a_k. \tag{2.54}$$

The transfer matrix $\mathbf{H}$ of the Stereo Linear Prediction synthesis filter was given by

$$\mathbf{H}(z) = \{\mathbf{F}(z)\}^{-1} = \frac{1}{\det(\mathbf{F}(z))} \begin{pmatrix} 1 - D(z) & C(z) \\ B(z) & 1 - A(z) \end{pmatrix}. \tag{2.55}$$

If we now apply SS to the transfer functions of the individual predictors, the new transfer matrix of the analysis filter is given by

$$\mathbf{F}'(z) = \begin{pmatrix} 1 - A\left(\frac{z}{\gamma}\right) & -C\left(\frac{z}{\gamma}\right) \\ -B\left(\frac{z}{\gamma}\right) & 1 - D\left(\frac{z}{\gamma}\right) \end{pmatrix} = \mathbf{F}\left(\frac{z}{\gamma}\right), \tag{2.56}$$

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

$$\mathbf{H}'(z) = \{\mathbf{F}'(z)\}^{-1} = \frac{1}{\det\left(\mathbf{F}\left(\frac{z}{\gamma}\right)\right)} \begin{pmatrix} 1 - D\left(\frac{z}{\gamma}\right) & C\left(\frac{z}{\gamma}\right) \\ B\left(\frac{z}{\gamma}\right) & 1 - A\left(\frac{z}{\gamma}\right) \end{pmatrix} = \mathbf{H}\left(\frac{z}{\gamma}\right), \tag{2.57}$$

with the determinant of $\mathbf{F}\left(\frac{z}{\gamma}\right)$ given by

$$\det\left(\mathbf{F}\left(\frac{z}{\gamma}\right)\right) = \left(1 - A\left(\frac{z}{\gamma}\right)\right)\left(1 - D\left(\frac{z}{\gamma}\right)\right) - B\left(\frac{z}{\gamma}\right) C\left(\frac{z}{\gamma}\right). \tag{2.58}$$

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

2.3 Rotation

The Rotation block is described in this section. First, it is explained what kind of operation Rotation is and next, the functions of the two rotators and inverse rotators are described. It is also described how the optimal rotation angle, which is used by the second rotator, can be calculated.

The scheme of a rotator can be seen in Figure 2.4(a). A rotator produces the output signals r_1 and r_2 from the input signals x_1 and x_2 with the matrix operation given by

$$\begin{pmatrix} r_1 \\ r_2 \end{pmatrix} = \begin{pmatrix} \cos(\varphi) & \sin(\varphi) \\ -\sin(\varphi) & \cos(\varphi) \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \tag{2.59}$$

where φ is the rotation angle. An example of rotation on signals x_1 and x_2 can be seen in Figure 2.4(b), which is a Lissajous plot of the involved signals.

The first rotator or pre-rotator in the SLP scheme uses a fixed rotation angle of $\pi/4$. As was mentioned already, the reason for this will be explained in Section 2.4.2. The second rotator produces the main and side signal m and s. This rotator uses a variable rotation angle, which is calculated such that the cross-correlation for lag zero is minimized.

The inverse rotators used in the decoder, perform the inverse operations of the rotators in the encoder. This means of course rotation over the negative rotation angle. Thus, the first inverse rotator uses the negation of the variable rotation angle and the second inverse rotator or inverse pre-rotator uses $-\pi/4$.

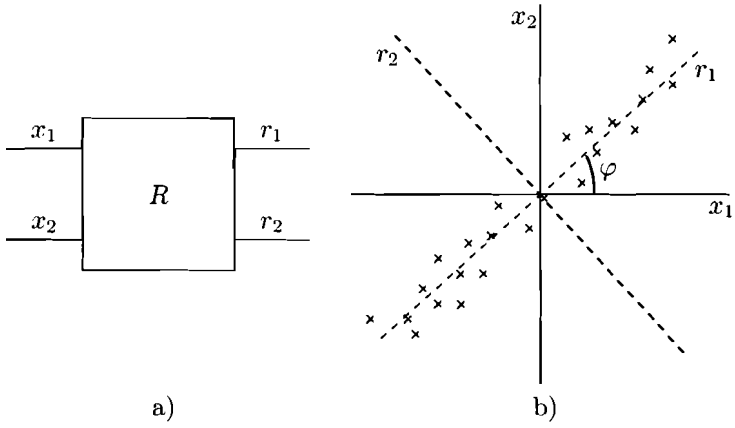

Figure 2.4: Rotation: a) scheme of a rotator, b) Lissajous plot as an example of rotation. A rotator produces the output signals r_1 and r_2 from the input signals x_1 and x_2 by applying rotation angle φ.

2.3.1 Calculation of the Optimal Rotation Angle

The optimal rotation angle, used by the second rotator, is calculated such that the cross-correlation for lag zero is minimized, 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) [34].

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

$$J = \sum_n |m[n]|^2 . \tag{2.60}$$

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

$$\underline{e}_1 = \begin{pmatrix} e_1[n] & e_1[n+1] & \cdots & e_1[n+N-1] \end{pmatrix}^T, \tag{2.61}$$

$$\underline{e}_2 = \begin{pmatrix} e_2[n] & e_2[n+1] & \cdots & e_2[n+N-1] \end{pmatrix}^T, \tag{2.62}$$

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

$$R_{11} = \underline{e}_1^T \cdot \underline{e}_1, \tag{2.63}$$

$$R_{22} = \underline{e}_2^T \cdot \underline{e}_2, \tag{2.64}$$

$$R_{12} = \underline{e}_1^T \cdot \underline{e}_2 = \underline{e}_2^T \cdot \underline{e}_1, \tag{2.65}$$

then J can be written as

$$\begin{aligned} J = |\underline{e}_1 \cos(\varphi) + \underline{e}_2 \sin(\varphi)|^2 &= R_{11}\cos^2(\varphi) + 2R_{12}\cos(\varphi)\sin(\varphi) + R_{22}\sin^2(\varphi) \\ &= \frac{R_{11}+R_{22}}{2} + \frac{R_{11}-R_{22}}{2}\cos(2\varphi) + R_{12}\sin(2\varphi). \end{aligned} \tag{2.66}$$

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

$$\frac{\partial J}{\partial \varphi} = 0, \tag{2.67}$$

which leads to

$$\frac{\partial J}{\partial \varphi} = -(R_{11}-R_{22})\sin(2\varphi) + 2R_{12}\cos(2\varphi) = 0. \tag{2.68}$$

If we now define

$$c = 2R_{12} - j(R_{11}-R_{22}), \tag{2.69}$$

with

$$\phi = \angle c, \tag{2.70}$$

then (2.68) can be rewritten as

$$\frac{\partial J}{\partial \varphi} = |c|\cos(2\varphi - \phi) = 0. \tag{2.71}$$

The solution of (2.71) is given by

$$\varphi = \phi/2 + \pi/4 + k\pi/2. \tag{2.72}$$

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

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

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

where $\hat{\varphi}$ are the values we were looking for.

The mean J_{mean}, maximum J_{max} and minimum J_{min} of J can now be given by

$$J_{mean} = \frac{R_{11} + R_{22}}{2}, \tag{2.75}$$

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

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

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

$$d = 2\frac{\sqrt{(R_{11} - R_{22})^2/4 + R_{12}^2}}{R_{11} + R_{22}}. \tag{2.78}$$

2.4 Regularization

When calculating the optimal prediction coefficients and optimal rotation angle, which has been described in Sections 2.2.3 and 2.3.1, respectively, some input signals can cause problems. Thus, regularization of the calculations in the Stereo Linear Prediction and the Rotation blocks is needed. The input signals that cause problems when calculating the optimal prediction coefficients are signals for which $\mathbf{Q}$ in (2.26) is singular. Examples of these input signals are mono, mono-like and one-channel zero signals (in this context, a stereo signal where the left and right signal are identical is referred to as a mono signal). One-channel zero signals can be called a specific digital problem, because the parts of an analog signal that have a 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. A regularization technique for Stereo Linear Prediction and a regularization technique for Rotation will be described next.

2.4.1 Regularization Technique for Stereo Linear Prediction

The regularization technique to solve the problems associated with signals for which $\mathbf{Q}$ is singular, is designed for solving the problems associated with mono, mono-like and one-channel zero signals, because these signals actually occur. It is expected that the problems associated with the other signals for which $\mathbf{Q}$ is singular, are also solved then. For mono and mono-like signals, auto- and cross-correlations are almost equal. Thus, when the Stereo Linear Prediction block has these signals as input, it is not defined how to predict the left channel from the right channel and vice versa. For one-channel zero signals, it is not defined how to predict the zero channel from itself and how to predict the other channel from the zero channel. In these cases, 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 $\mathbf{R_0}$ becomes high, resulting in an inversion of $\mathbf{R_0}$ that is difficult to calculate. This inversion is needed in the initialization of the block-Levinson algorithm (see (2.47), (2.48) and (2.49)). 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 signal is to add a small amount of noise to the channel with the zero signal while calculating the optimal prediction coefficients. 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. However, it is expected that mono and especially mono-like signals occur more often than one-channel zero signals, so it is more important to have a good method for biasing mono and mono-like signals than to have a good method for biasing one-channel zero signals. Therefore, a pre-rotator is added

which uses a rotation angle of $\pi/4$. This results in mono and mono-like signals rotated to one-channel zero signals and vice versa. We can now add a small amount of noise to the one-channel zero signals that are obtained by applying the pre-rotator to mono and mono-like signals. Thus, noise is added to $\mathbf{Q}$ and $\mathbf{P}$ in the Yule-Walker equations (2.26), such that they lead towards a biased solution for the optimal prediction coefficients. This means that the problems associated with mono and mono-like signals are solved by transforming these signals into one-channel zero signals and subsequently, by applying the regularization technique that is known for these signals (adding noise to $\mathbf{Q}$ and $\mathbf{P}$).

Adding noise to $\mathbf{Q}$ and $\mathbf{P}$ is done by adding noise to each $\mathbf{R_k}$ in $\mathbf{Q}$ and $\mathbf{P}$ according to

$$\begin{aligned} \mathbf{R'_{k,11}} &= \mathbf{R_{k,11}} + \epsilon_{rel} \cdot \mathbf{R_{k,22}} \\ \mathbf{R'_{k,22}} &= \mathbf{R_{k,22}} + \epsilon_{rel} \cdot \mathbf{R_{k,11}} \end{aligned}, \tag{2.79}$$

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, $\mathbf{R}'$ may be interpreted as adding some crosstalk to two stochastically independent signals. This improves the condition number of each $\mathbf{R_k}$ and therefore decreases the numerical problems in the block-Levinson algorithm.

The problems associated with one-channel zero signals are solved as well with this method, because these signals are transformed into mono signals and adding noise to $\mathbf{Q}$ and $\mathbf{P}$ also works for mono signals. However, noise is now added to both channels, resulting in a distortion of both channels. This is not the case with mono and mono-like signals. These signals are transformed into one-channel zero signals and the noise is then added to the zero channel only, keeping the channel with the signal to be predicted noise-free. In this way, we see that mono and mono-like signals are less disturbed than one-channel zero signals, due to the use of a pre-rotator.

2.4.2 Regularization Technique for Rotation

The Lissajous plot of a signal with low cross-correlation between the channels and with equal channel powers can be seen in Figure 2.5. 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 (2.73)).

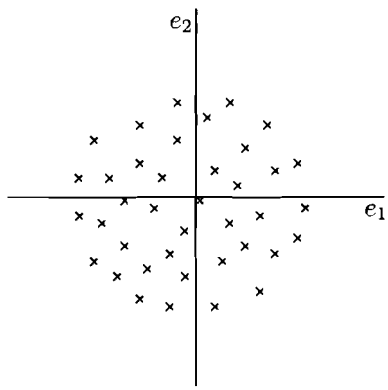

Figure 2.5: Lissajous plot of a signal with low cross-correlation between the channels and with equal channel powers.

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 (2.73)) 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 (2.78)), because if this is too low, then there is no significant gain to be reached with the rotator for any angle.

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

2.5 Summary

We described in this chapter our proposal for a stereo audio coder, which was called SLP. First, we gave a short description of the proposed coding scheme. We also stated that the encoder results in two spectrally flat, uncorrelated signals and that the decoder reconstructs the original input signal. Thereafter, we described the two most important blocks in the coding scheme, which are Stereo Linear Prediction and Rotation, in more detail. In addition, we described how the optimal prediction coefficients and the optimal rotation angle can be calculated. We concluded this chapter with regularization of these calculations.

In the next two chapters, we will describe the two remaining issues of the SLP scheme. The first issue, described in Chapter 3, is making SLP more psycho-acoustically inspired by incorporating Laguerre filters into its scheme. The second issue, described in Chapter 4, is quantization of the prediction coefficients and quantization of the main and side signal.

Chapter 3

Laguerre-Based Pure Stereo Linear Prediction

The human auditory system has a frequency resolution which is denser at lower frequencies and sparser at higher frequencies. Frequency warping, as introduced by Oppenheim et al. in 1971 [35], is a technique that modifies the uniform frequency resolution, common to most signal processing systems, to a non-uniform frequency resolution. If designed well, this non-uniform frequency resolution approximately corresponds to a psycho-acoustically relevant scale such as a Bark or an ERB scale [36].

Frequency warping is a linear transformation between the original sequence $f(n)$ and the transformed sequence $g(k)$

$$f(n) = \sum_{k=-\infty}^{+\infty} g(k)\psi_k(n), \tag{3.1}$$

where $\psi_k(n)$ is a set of linearly independent sequences. We want the Fourier transform of $f(n)$ and the Fourier transform of $g(k)$ to be related by a change of variables

$$F(e^{j\Omega}) = G(e^{j\omega}), \tag{3.2}$$

with $\omega = \theta(\Omega)$. This leads to the following requirement on the set of functions $\psi_k(n)$

$$\Psi_k(e^{j\Omega}) = e^{-jk\theta(\Omega)}, \tag{3.3}$$

which means that the functions $\psi_k(n)$ must have an allpass characteristic. The simplest, implementable form of $\Psi_k(z)$ can now be given by

$$\Psi_k(z) = \left(\frac{z^{-1} - \lambda}{1 - \lambda z^{-1}}\right)^k, \tag{3.4}$$

with the relation between the frequency variables ω and Ω given by

$$\omega = \theta(\Omega) = \tan^{-1}\left[\frac{(1-\lambda^2)\sin\Omega}{(1+\lambda^2)\cos\Omega - 2\lambda}\right], \tag{3.5}$$

where λ is the warping factor with $|\lambda| < 1$. For $\lambda > 0$ the frequency resolution increases for lower frequencies, for $\lambda < 0$ the frequency resolution increases for higher frequencies and for $\lambda = 0$ no warping occurs.

Warped Linear Prediction (WLP) was introduced by Strube [37] in 1980 as a technique that combines Linear Prediction with frequency warping. With WLP, the Linear Prediction scheme can be better matched to the human auditory system. However, WLP whitens the input signal in the warped frequency domain, but not in the original frequency domain. The synthesis filter

of WLP is also not directly realizable with the normal feedback structure, because of delay-free loops that are introduced. As a solution to these problems, Pure Linear Prediction (PLP) was introduced by Voitishchuk and den Brinker et al. in 2002 [38] [39] [40]. PLP whitens the input signal in the original frequency domain and has a directly realizable synthesis filter, because the predicted signal relies only on past samples. In addition, PLP can be tuned in such a way, that it produces a frequency mapping very similar to that of WLP.

WLP and Laguerre-based PLP (L-PLP) are described in more detail in the first two sections of this chapter. This will be done only for the one channel case. However, to go from the one channel to the stereo case is straightforward with Section 2.2 in mind. Section 3.3 gives the relation between the correlation sequences of WLP and L-PLP in the stereo case and Section 3.4 describes how the regularization technique for Stereo Linear Prediction, given in Section 2.4.1, can be adapted for Laguerre-based Pure Stereo Linear Prediction. The chapter concludes with a short summary.

3.1 Warped Linear Prediction

Strube [37] showed that frequency warping can be included into the Linear Prediction scheme by replacing the tapped-delay lines in the analysis and synthesis filters with transfers z^{-k}, by allpass filter lines with transfers $G_k(z)$, given by

$$G_k(z) = \left(\frac{z^{-1} - \lambda}{1 - \lambda z^{-1}}\right)^k. \tag{3.6}$$

Equation (1.12) can now be rewritten for WLP to

$$e[n] = x[n] - \hat{x}[n] = x[n] - \sum_{k=1}^{P} y_k[n]\alpha_k, \tag{3.7}$$

where P and α_k are the prediction order and the prediction coefficients, respectively, and y_k are the outputs of the allpass filters or regressor signals, given by

$$y_k[n] = g_k[n] * x[n], \tag{3.8}$$

where g_k is the impulse response of the kth filter given by (3.6). The optimal prediction coefficients can still be calculated by solving

$$\mathbf{Q}\underline{\alpha} = \underline{p}, \tag{3.9}$$

with

$$\mathbf{Q}_{\mathbf{k},\mathbf{l}} = \rho_{y,k-l}, \tag{3.10}$$

and

$$\underline{p}_k = \rho_{y,k}, \tag{3.11}$$

but the autocorrelation function ρ_y (the warped autocorrelation function of the input signal) is now given by

$$\rho_{y,k} = \sum_n y_k[n]x[n], \tag{3.12}$$

where the sum extends from minus infinity to infinity. The schemes of the WLP analysis and synthesis filters can be seen in Figure 3.1.

WLP has been successfully applied in speech and audio coding [41] [42] [43]. It was shown in [44] that WLP can lead to a bit rate reduction of one bit per sample compared to traditional Linear Prediction, while retaining the same quality level. However, there is a direct feed-through

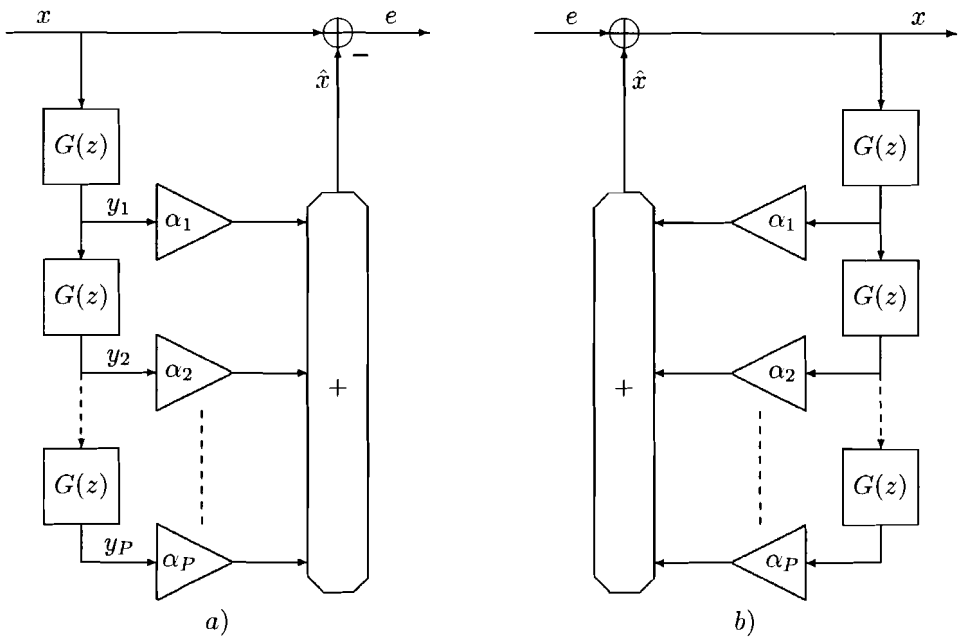

Figure 3.1: Warped Linear Prediction scheme: a) analysis filter, b) synthesis filter. The analysis filter produces the error signal e from signal x by applying the prediction coefficients α_k. The synthesis filter reconstructs x from e and the α's.

in the allpass filter line in the analysis filter. Thus, information from the current sample is present in the outputs of the allpass sections, as opposed to traditional Linear Prediction. This results, as already stated, in the problem that WLP whitens the input signal in the warped frequency domain, but not in the original frequency domain. In addition, the synthesis filter of WLP is not directly realizable with the normal feedback structure, because of the delay-free loops that are introduced.

A solution for the whitening problem is the following. Strube [37] showed that WLP minimizes an error signal weighted with

$$D_0(z) = \frac{\sqrt{1-\lambda^2}}{1-\lambda z^{-1}}. \tag{3.13}$$

Thus, to achieve a spectrally flat error signal, the error signal has to post-filtered in the analysis filter with the filter given by

$$D_0^{-1}(z) = \frac{1-\lambda z^{-1}}{\sqrt{1-\lambda^2}}. \tag{3.14}$$

In addition, $D_0(z)$ should be applied in the synthesis filter.

Another solution for the same problem was given by Voitishchuk et al. [38]. It was stated that the optimal prediction coefficients should be calculated on the warped input signal s. This leads to a new autocorrelation function ρ_s (the autocorrelation function of the warped input signal), to be used in (3.9), given by

$$\rho_{s,k} = \sum_n s[n-k]s[n]. \tag{3.15}$$

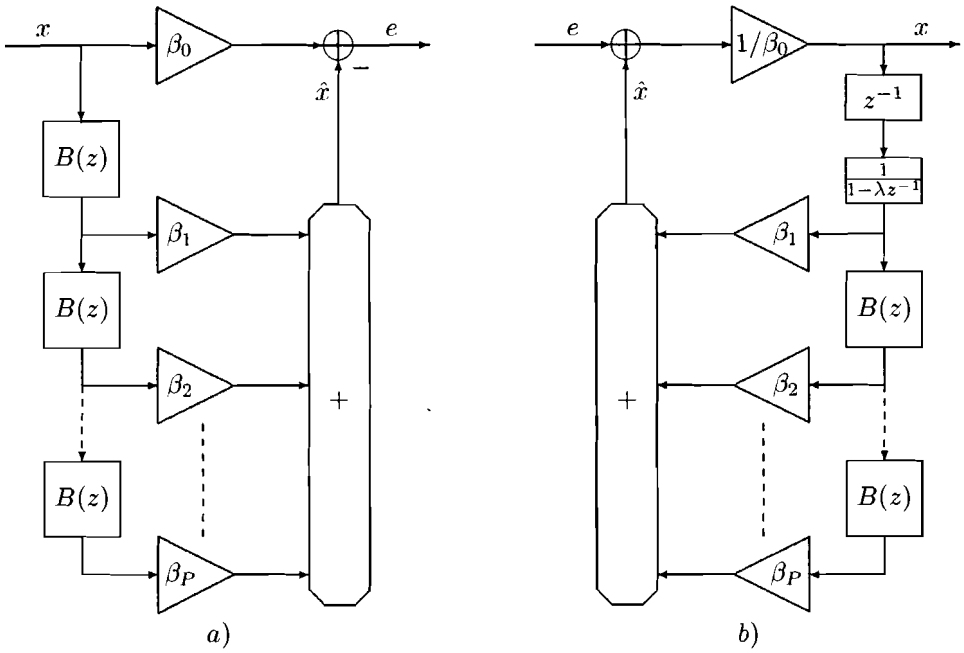

Figure 3.2: Warped Linear Prediction scheme implemented by Gamma filters: a) analysis filter, b) synthesis filter.

It was also shown that

$$\rho_{s,k} = \sum_n s[n-k]s[n] = \frac{1}{2\pi}\int_{<2\pi>} \left\{ \frac{\sqrt{1-\lambda^2}}{1-\lambda z^{-1}} \cdot X(z) \cdot G^{-k}(z) \right\} \left\{ X^*(z) \cdot \left(\frac{\sqrt{1-\lambda^2}}{1-\lambda z^{-1}} \right)^* \right\} d\theta, \tag{3.16}$$

with $z = e^{j\theta}$, and

$$\rho_{y,k} = \sum_n y_k[n]x[n] = \frac{1}{2\pi}\int_{<2\pi>} X(z) \cdot G^{-k}(z) \cdot X^*(z) d\theta, \tag{3.17}$$

where X is the Fourier transform of signal x. Thus, the optimal prediction coefficients can be calculated by minimizing the error signal shown in Figure 3.1, where an extra pre-filter is used to filter the input signal. The pre-filter is given by

$$D_0(z) = \frac{\sqrt{1-\lambda^2}}{1-\lambda z^{-1}}. \tag{3.18}$$

The error signal is then obtained with the analysis filter in Figure 3.1 and with the calculated prediction coefficients, but without the extra pre-filter. This results in a spectrally flat error signal.

A solution for the second problem, which is that the synthesis filter of WLP is not directly realizable, has also been given by Strube [37]. He proposed to map the calculated prediction coefficients to a new structure that uses Gamma filters. This new structure introduces a delay in the synthesis filter. It can be seen in Figure 3.2 and uses the following filters $B(z)$

$$B(z) = \frac{z^{-1}}{1-\lambda z^{-1}}. \tag{3.19}$$

However, when the prediction coefficients are mapped to Gamma filters, numerical problems can arise, which result in significant inaccuracies [40].

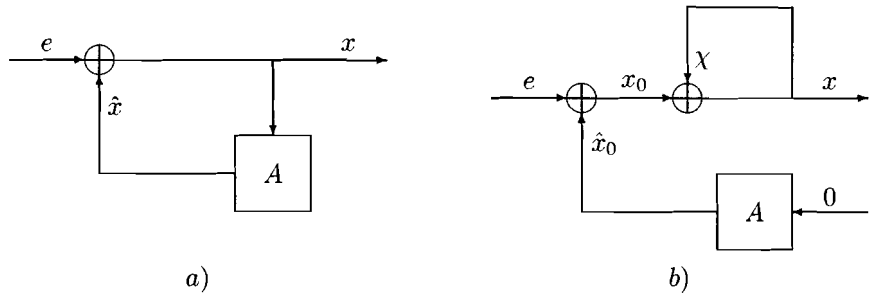

Figure 3.3: Synthesis filter structures: a) original structure, b) modified structure, with A a linear prediction filter with delay-free paths.

Another solution for the second problem was proposed by Härmä [45]. He suggested two approaches. In the first approach, he used a technique to implement the delay-free loops in the synthesis filter as a separate branch, without modifying the structure of the original prediction coefficients. This technique changes the synthesis filter from the circuit in Figure 3.3(a) to the circuit in Figure 3.3(b), where A is a linear prediction filter with delay-free paths. Thus, the difference equation of the synthesis filter changes from

$$x[n] = e[n] + \hat{x}[n], \tag{3.20}$$

where $\hat{x}[n]$ is obtained by passing $x[n]$ through A, to

$$x[n] = \frac{e[n] + \hat{x}_0[n]}{1 - \chi}, \tag{3.21}$$

where χ is a coefficient equal to the pure delay-free structure of A (which means that all its unit delay elements are disconnected) and it is assumed that $\chi \neq 1$. Thus, the computation of the output x is separated from updating the states of filter A.

In the second approach, Härmä used a technique to eliminate the delay-free loops in the synthesis filter, producing a modified filter structure with a new set of coefficients. This technique does not change the structure of A, but its output nodes and feedback coefficients are changed, which leads to the following difference equation of the synthesis filter

$$x[n] = \frac{1}{1 - \chi} \cdot \left(e[n] + \sum_{i=1}^{P} c_i r_i \right), \tag{3.22}$$

where the signals r_i are selected such that they are outputs of unit delay elements of A and do not depend on the current value of x, c_i are coefficients to be calculated, and it is still assumed that $\chi \neq 1$.

Both methods of Härmä change the structure of the synthesis filter, resulting in a synthesis filter which is not exactly the inverse of the analysis filter. The second approach changes the filter coefficients as well. These issues result in the problem that the synthesis filter is not able to perfectly reconstruct the input signal anymore, because now different inaccuracies arise in the analysis filter than in the synthesis filter. In addition, it is clear that both methods of Härmä have problems when χ approaches 1, which can lead to numerical sensitivity problems.

3.2 Laguerre-Based Pure Linear Prediction

Pure Linear Prediction was introduced as a solution to the problems associated with WLP. It is based on IIR filters and relies only on past samples of the input signal, hence traditional Linear Prediction is a special case of PLP, however WLP is not. The synthesis filter of PLP can be

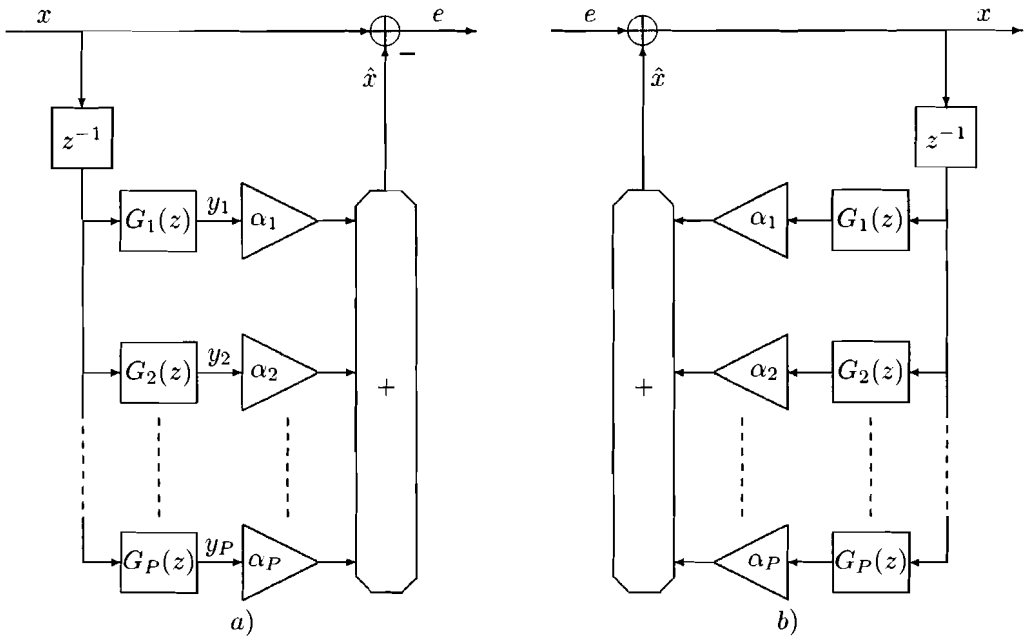

Figure 3.4: Pure Linear Prediction scheme: a) analysis filter, b) synthesis filter, where G_k are stable and causal IIR filters. The analysis filter produces the error signal e from signal x by applying the prediction coefficients α_k. The synthesis filter reconstructs x from e and the α's.

directly realized because of the explicit delay in the feedback loop. In addition, a suitable choice for the IIR filters can produce a frequency warping similar to that of WLP, while the whitening property of the input signal still holds. Furthermore, stability of the synthesis filter for this specific class of IIR filters is also guaranteed, if the optimal prediction coefficients are calculated using input data windowing [46].

The schemes of the PLP analysis and synthesis filters can be seen in Figure 3.4, where G_k are stable and causal IIR filters. The delay z^{-1} is here explicitly shown. The transfer function F of the analysis filter is given by

$$F(z) = 1 - z^{-1} \sum_{k=1}^{P} \alpha_k G_k(z), \tag{3.23}$$

and the transfer function H of the synthesis filter is given by

$$H(z) = \frac{1}{F(z)} = \frac{1}{1 - z^{-1} \sum_{k=1}^{P} \alpha_k G_k(z)}. \tag{3.24}$$

The error signal is still given by

$$e[n] = x[n] - \hat{x}[n] = x[n] - \sum_{k=1}^{P} y_k[n] \alpha_k, \tag{3.25}$$

where the regressor signals y_k are the outputs of the filters G_k, given by

$$y_k[n] = g_k[n] * x[n], \tag{3.26}$$

with g_k the impulse response of G_k. In addition, the optimal prediction coefficients can still be calculated by solving

$$\mathbf{Q}\underline{\alpha} = \underline{p}, \tag{3.27}$$

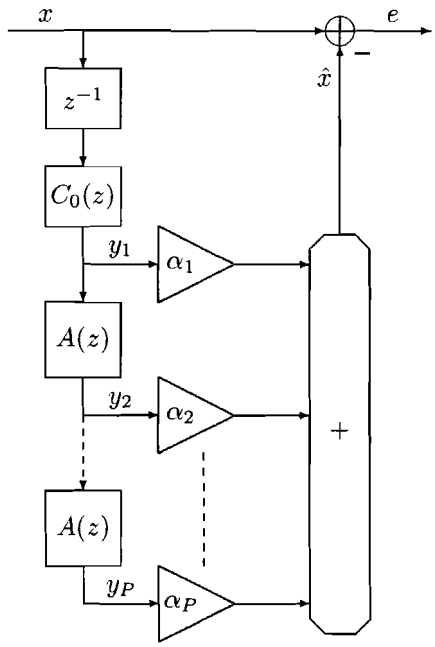

Figure 3.5: Analysis filter of a L-PLP sytem. The Laguerre filters are implemented as a pre-filer followed by a tapped allpass line.

but now

$$\mathbf{Q_{k,l}} = \sum_n y_k[n]y_l[n], \tag{3.28}$$

and

$$\underline{p}_k = \sum_n y_k[n]x[n]. \tag{3.29}$$

We now take Laguerre filters for the filters G_k. This leads to a special case of PLP, Laguerre-based PLP. Laguerre filters [47] [48] are given by

$$G_k(z) = \frac{\sqrt{1-\lambda^2}}{1-\lambda z^{-1}} \left(\frac{z^{-1}-\lambda}{1-\lambda z^{-1}} \right)^{k-1}, \tag{3.30}$$

with $\lambda \in \mathbb{R}$ and $|\lambda| < 1$. For $\lambda > 0$ the frequency resolution increases for lower frequencies, for $\lambda < 0$ the frequency resolution increases for higher frequencies and $\lambda = 0$ leads to traditional Linear Prediction. Thus, a behaviour similar to WLP is obtained. We also remark that for L-PLP, $\mathbf{Q_{k,1}}$ is equal to the autocorrelation function of the warped input signal

$$\mathbf{Q_{k,1}} = \rho_{s,k} = \sum_n s[n-k]s[n]. \tag{3.31}$$

The Laguerre filters can be implemented efficiently as a tapped allpass line with filters A, preceded by the section C_0, with A given by

$$A(z) = \frac{z^{-1}-\lambda}{1-\lambda z^{-1}}, \tag{3.32}$$

and C_0 given by

$$C_0(z) = \frac{\sqrt{1-\lambda^2}}{1-\lambda z^{-1}}. \tag{3.33}$$

The simplest implementation of the analysis filter of a L-PLP system can be seen in Figure 3.5.

3.3 Correlation Sequences

In this section, we derive the relations between the $\mathbf{Q_W}$ and $\mathbf{P_W}$ in the Yule-Walker equations of Warped Stereo Linear Prediction (WSLP) and the $\mathbf{Q_L}$ and $\mathbf{P_L}$ in the Yule-Walker equations of Laguerre-based Pure Stereo Linear Prediction (L-PSLP).

First, we rewrite equations (3.10), (3.11), (3.28) and (3.29) for the stereo case to

$$\mathbf{Q_{W,k,l}} = \sum_n \underline{y}_{W,k-l}^T[n]\underline{x}[n], \tag{3.34}$$

$$\mathbf{P_{W,k}} = \sum_n \underline{y}_{W,k}^T[n]\underline{x}[n], \tag{3.35}$$

$$\mathbf{Q_{L,k,l}} = \sum_n \underline{y}_{L,k}^T[n]\underline{y}_{L,l}[n], \tag{3.36}$$

$$\mathbf{P_{L,k}} = \sum_n \underline{y}_{L,k}^T[n]\underline{x}[n], \tag{3.37}$$

where $\underline{x}$, $\underline{y}_{W,k}$ and $\underline{y}_{L,k}$ are the input signal in the stereo case, the regressor signals of WSLP and the regressor signals of L-PSLP, respectively. From (3.34) and (3.35) it directly follows that

$$\mathbf{Q_{W,k+1,1}} = \mathbf{P_{W,k}}. \tag{3.38}$$

Before we can establish the relation between $\mathbf{Q_W}$ and $\mathbf{P_L}$, we first have to establish a relation between the allpass filter G, which is used in WSLP, and the first section $z^{-1}C_0$ of the Laguerre filter, which is used in L-PSLP. This relation can be written as

$$G(z) = L_1 + L_2 z^{-1} C_0(z), \tag{3.39}$$

which leads to

$$\frac{z^{-1} - \lambda}{1 - \lambda z^{-1}} = L_1 + L_2 z^{-1} \frac{\sqrt{1-\lambda^2}}{1 - \lambda z^{-1}}, \tag{3.40}$$

and

$$z^{-1} - \lambda = L_1 \left(1 - \lambda z^{-1}\right) + L_2 z^{-1} \sqrt{1-\lambda^2}. \tag{3.41}$$

Thus, L_1 and L_2 are given by

$$L_1 = -\lambda, \tag{3.42}$$

$$L_2 = \sqrt{1-\lambda^2}. \tag{3.43}$$

If we now look at $\mathbf{Q_W}$

$$\mathbf{Q_{W,k,1}} = \sum_n \left(g_{k-1}[n] * \underline{x}^T[n]\right) \underline{x}[n], \tag{3.44}$$

and use (3.39) to replace the $(k-1)$th g, then (3.44) becomes

$$\mathbf{Q_{W,k,1}} = \sum_n \left\{\left(L_1 + L_2 z^{-1} C_0\right) * g_{k-2}[n] * \underline{x}^T[n]\right\} \underline{x}[n], \tag{3.45}$$

which leads to the following relation between $\mathbf{Q_W}$ and $\mathbf{P_L}$

$$\begin{array}{lcll} \mathbf{Q_{W,k,1}} & = & L_1\mathbf{Q_{W,k-1,1}} + L_2\mathbf{P_{L,k-1}}, & \text{for } k > 1 \\ \mathbf{Q_{W,1,1}} & = & \mathbf{P_{L,0}}, & \text{for } k = 1 \end{array}, \tag{3.46}$$

where $\mathbf{P_{L,0}}$ is the energy of the input signal x, given by

$$\mathbf{P_{L,0}} = \sum_n \underline{x}^T[n]\underline{x}[n]. \tag{3.47}$$

Thus, $\mathbf{P_L}$ can be computed from $\mathbf{Q_W}$ with the following set of equations

$$\begin{array}{rcl} \mathbf{P_{L,0}} & = & \mathbf{Q_{W,1,1}} \\ \mathbf{P_{L,1}} & = & \frac{1}{L_2}\mathbf{Q_{W,2,1}} - \frac{L_1}{L_2}\mathbf{Q_{W,1,1}} \\ \vdots & = & \vdots \\ \mathbf{P_{L,P}} & = & \frac{1}{L_2}\mathbf{Q_{W,P+1,1}} - \frac{L_1}{L_2}\mathbf{Q_{W,P,1}} \end{array}. \tag{3.48}$$

To find the relation between $\mathbf{P_L}$ and $\mathbf{Q_L}$, we follow the same reasoning as in [49], but now for the stereo case. First, we have to establish a relation between the first section $z^{-1}C_0$ and the allpass filter A of the Laguerre filter, which are used in L-PSLP. This relation can be written as

$$z^{-1}C_0(z) = K_1 + K_2A(z), \tag{3.49}$$

which leads to

$$z^{-1}\frac{\sqrt{1-\lambda^2}}{1-\lambda z^{-1}} = K_1 + K_2\frac{z^{-1}-\lambda}{1-\lambda z^{-1}}, \tag{3.50}$$

and

$$z^{-1}\sqrt{1-\lambda^2} = K_1\left(1-\lambda z^{-1}\right) + K_2\left(z^{-1}-\lambda\right). \tag{3.51}$$

Thus, K_1 and K_2 are given by

$$K_1 = \frac{\lambda}{\sqrt{1-\lambda^2}}, \tag{3.52}$$

$$K_2 = \frac{1}{\sqrt{1-\lambda^2}}. \tag{3.53}$$

Note that $K_1 = -L_1/L_2$ and $K_2 = 1/L_2$. We now look at $\mathbf{Q_L}$, which is given by

$$\mathbf{Q_{L,1,1}} = \sum_n \underline{y}^T_{L,1}[n]\underline{y}_{L,l}[n], \tag{3.54}$$

where $\underline{y}_{L,1}$ is equal to

$$\underline{y}_{L,1}[n] = c_0[n] * \underline{x}[n-1], \tag{3.55}$$

with c_0 the impulse response of pre-filter C_0. Substituting (3.55) into (3.54) results in

$$\mathbf{Q_{L,1,1}} = \sum_n \left[c_0[n] * \underline{x}[n-1]\right]^T \underline{y}_{L,l}[n]. \tag{3.56}$$

If we now use (3.49) to replace $z^{-1}c_0[n]$, then (3.56) becomes

$$\mathbf{Q_{L,1,1}} = \sum_n \left[\{K_1 + K_2a[n]\} * \underline{x}[n]\right]^T \underline{y}_{L,l}[n], \tag{3.57}$$

where a is the impulse response of the allpass filter. This leads to the following relation between $\mathbf{P_L}$ and $\mathbf{Q_L}$

$$\mathbf{Q_{L,1,l}} = K_1\mathbf{P}^{\mathrm{T}}_{\mathbf{L,1}} + K_2\mathbf{P}^{\mathrm{T}}_{\mathbf{L,l-1}}, \qquad \text{for } l > 1. \tag{3.58}$$

To find an expression for $\mathbf{Q_{L,1,1}}$, we consider the difference equation of the first section of the Laguerre filters, which is given by

$$\sqrt{1-\lambda^2}\underline{x}[n-1] = \underline{y}_{L,1}[n] - \lambda\underline{y}_{L,1}[n-1]. \tag{3.59}$$

Multiplying (3.59) with $\underline{x}^T[n-1]$ and taking the summation over all n results in

$$\begin{aligned}\sqrt{1-\lambda^2}\textstyle\sum_n \underline{x}^T[n-1]\underline{x}[n-1] &= \textstyle\sum_n \underline{x}^T[n-1]\underline{y}_{L,1}[n] - \lambda\sum_n \underline{x}^T[n-1]\underline{y}_{L,1}[n-1] \\ &= \textstyle\sum_n \left[\frac{\underline{y}^T_{L,1}[n]}{\sqrt{1-\lambda^2}} - \frac{\lambda\underline{y}^T_{L,1}[n-1]}{\sqrt{1-\lambda^2}}\right]\underline{y}_{L,1}[n] - \lambda\sum_n \underline{x}^T[n-1]\underline{y}_{L,1}[n-1]\end{aligned}. \tag{3.60}$$

By substituting (3.36), (3.37) and (3.47) into (3.60), we get

$$\sqrt{1-\lambda^2}\mathbf{P_{L,0}} = \frac{1}{\sqrt{1-\lambda^2}}\left[\mathbf{Q_{L,1,1}} - \lambda\sum_n \underline{y}^T_{L,1}[n-1]\underline{y}_{L,1}[n]\right] - \lambda\mathbf{P}^{\mathrm{T}}_{\mathbf{L,1}}, \tag{3.61}$$

and using (3.59) results in an expression for $\mathbf{Q_{L,1,1}}$, given by

$$\mathbf{Q_{L,1,1}} = K_1\mathbf{P}^{\mathrm{T}}_{\mathbf{L,1}} + \mathbf{P_{L,0}} + K_1\mathbf{P_{L,1}}. \tag{3.62}$$

Thus, $\mathbf{Q_L}$ can be computed from $\mathbf{P_L}$ with the following set of equations

$$\begin{aligned}\mathbf{Q_{L,1,1}} &= K_1\mathbf{P}^{\mathrm{T}}_{\mathbf{L,1}} + \mathbf{P_{L,0}} + K_1\mathbf{P_{L,1}} \\ \mathbf{Q_{L,1,2}} &= K_1\mathbf{P}^{\mathrm{T}}_{\mathbf{L,2}} + K_2\mathbf{P}^{\mathrm{T}}_{\mathbf{L,1}} \\ &\vdots \\ \mathbf{Q_{L,1,P}} &= K_1\mathbf{P}^{\mathrm{T}}_{\mathbf{L,P}} + K_2\mathbf{P}^{\mathrm{T}}_{\mathbf{L,P-1}}\end{aligned}. \tag{3.63}$$

3.4 Regularization Technique for Laguerre-Based Pure Stereo Linear Prediction

As was the case when calculating the optimal prediction coefficients for Stereo Linear Prediction, signals for which $\mathbf{Q}$ in (3.27) is singular, for example mono, mono-like and one-channel zero signals, also cause problems when calculating the optimal prediction coefficients for L-PSLP. Thus, a regularization technique, as described in Section 2.4.1 for Stereo Linear Prediction, is needed for L-PSLP as well. In addition, Spectral Smoothing, as described in Section 2.2.3 for Stereo Linear Prediction, has to be applied, to make the calculated prediction coefficients more robust. In this section, both the regularization technique for Stereo Linear Prediction and Spectral Smoothing will be adapted for L-PSLP.

The approach that was followed in Section 2.4.1 for the regularization technique for Stereo Linear Prediction will also be followed for L-PSLP. However, problems arrive when the noise has to be added to $\mathbf{Q_L}$ and $\mathbf{P_L}$, because for L-PSLP, $\mathbf{Q_L}$ and $\mathbf{P_L}$ do not have the same elements $\mathbf{R_k}$. Therefore, for L-PSLP the noise has to be added in a different way. For Stereo Linear Prediction, the noise was added to the auto-correlation function of the input signal. This means that for L-PSLP, the noise has to be added to the auto-correlation function of the warped input signal. Thus, the noise has to be added to $\mathbf{Q_L}$, according to

$$\begin{aligned}\mathbf{Q'_{L,1,1,11}} &= \mathbf{Q_{L,1,1,11}} + \epsilon_{rel}\cdot\mathbf{Q_{L,1,1,22}} \\ \mathbf{Q'_{L,1,1,22}} &= \mathbf{Q_{L,1,1,22}} + \epsilon_{rel}\cdot\mathbf{Q_{L,1,1,11}}\end{aligned}, \tag{3.64}$$

where ϵ_{rel} is a factor indicating the amount of noise to be added. Equation (3.58) can be used then to compute the biased $\mathbf{P_L}$ from the biased $\mathbf{Q_L}$, assuming that $\mathbf{P_{L,k}} = 0$ for $k = P+1, P+2, \ldots$. However, this assumption can lead to stability problems in the synthesis filter and hence, the assumption is not valid.

Therefore, another, safer, method for adding the noise is to add it to $\mathbf{Q_W}$ in the warped domain, where $\mathbf{Q_W}$ and $\mathbf{P_W}$ have the same elements. Then, (3.48) can be used to go from $\mathbf{Q_W}$ to $\mathbf{P_L}$ and (3.63) to go to $\mathbf{Q_L}$.

Another interesting application of biasing of L-PLP was given by Biswas et al. [50]. It was described how L-PLP can be perceptually biased when the optimal prediction coefficients are calculated. It has to be noted that this technique can also be used for L-PSLP. However, the details of it are beyond the scope of this report and still need to be investigated.

It is not known how SS can be applied to the prediction coefficients of L-PSLP directly. However, it was shown in [39] that mapping the Laguerre filter coefficients to FIR filter coefficients makes the use of SS for Stereo Linear Prediction, as described in Section 2.2.3, possible. This procedure is now described for the one channel case, because to go from the one channel to the stereo case is straightforward.

The mapping from the L-PLP analysis filter

$$F(z) = 1 - z^{-1} \sum_{k=1}^{P} \alpha_k G_k(z), \tag{3.65}$$

onto the FIR filter

$$G(v) = \sum_{k=0}^{P} c_k v^{-k}, \tag{3.66}$$

with the restriction that

$$\sum_{k=0}^{P} c_k (-\lambda)^k = 1, \tag{3.67}$$

is given by

$$v^{-1} = \frac{-\lambda + z^{-1}}{1 - \lambda z^{-1}}. \tag{3.68}$$

SS can now be applied to the coefficients c_k. However, to yield a valid set of mapped Laguerre filter coefficients, condition (3.67) must be reinstated by calculating a normalization factor g_c, given by

$$g_c = \sum_{k=0}^{P} \gamma^k c_k (-\lambda)^k, \tag{3.69}$$

and applying this factor to the smoothed prediction coefficients, which results in

$$c'_k = \gamma^k c_k / g_c. \tag{3.70}$$

The smoothed Laguerre filter coefficients can then be obtained by applying the inverse mapping.

3.5 Summary

We described WSLP and L-PSLP in this chapter. We showed that L-PSLP is preferred, because it produces a frequency warping similar to that of WSLP, while it solves the problems associated with it. We also described the relation between the correlation sequences of WSLP and L-PSLP and gave a regularization technique for L-PSLP.

We already described our proposal for a stereo audio coder, which we called SLP, and how SLP can be made more psycho-acoustically inspired by incorporating Laguerre filters into its scheme. In the next chapter, we will describe the remaining issues of the SLP scheme, which are quantization of the prediction coefficients and quantization of the main and side signal.

Chapter 4

Quantization

Quantization, which is the remaining issue of the SLP scheme, is described in this chapter. This starts with quantization of the stereo prediction coefficients, followed by quantization of the main and side signal. The chapter concludes with a short summary.

4.1 Quantization of the Stereo Prediction Coefficients

Prediction coefficients are preferably not quantized directly, because the frequency responses of the analysis and synthesis filters are extremely sensitive to such quantization. This can even lead to stability problems of the synthesis filter. Therefore, the prediction coefficients have to be mapped to a different representation before they can be quantized. This should result in a smaller distortion of the analysis and synthesis filter characteristics.

The approach usually taken for one channel Linear Prediction is to map the prediction coefficients to Line Spectral Frequencies (LSFs) or to Reflection Coefficients (RCs) and then to Arcsine Coefficients (ACs) or Log Area Ratios (LARs) [3].

The LSFs are defined as the roots of polynomials P and Q, which are given by

$$\begin{aligned} P(z) &= F(z) + z^{-(P+1)}F(z^{-1}) \\ Q(z) &= F(z) - z^{-(P+1)}F(z^{-1}) \end{aligned}, \tag{4.1}$$

where F and P are the transfer function of the analysis filter and the prediction order, respectively. The roots of P and Q are on the unit circle and are interlaced.

The RCs are intermediate variables of the Levinson algorithm [33], which was described in Section 2.2.3 for block matrices. The synthesis filter is unconditionally stable if the RCs are of magnitude less than one. The RCs can be quantized directly. However, the frequency responses of the analysis and synthesis filters are more sensitive to distortions when the magnitude of the RCs is close to one than when it is close to zero. Therefore, RCs can be better mapped to ACs or LARs. ACs (Θ_i) are given by

$$\Theta_i = \sin^{-1} RC_i, \tag{4.2}$$

and LARs (Γ_i) are given by

$$\Gamma_i = \log \frac{1 + RC_i}{1 - RC_i}. \tag{4.3}$$

However, mapping the prediction coefficients to LSFs or RCs is not straightforward for Stereo Linear Prediction. In addition, it requires an extra step to map the prediction coefficients of L-PSLP to LSFs or RCs. Therefore, as described in [39], the Laguerre filter coefficients should be mapped to FIR filter coefficients, to make the mapping to LSFs or RCs possible. This mapping from Laguerre filter coefficients to FIR filter coefficients was described in Section 3.4.

Thus, the following method was proposed [51] to quantize Stereo Linear Prediction coefficients. First, the L-PSLP coefficients are mapped to FIR filter coefficients. Next, these FIR filter coefficients are mapped to Reflection Matrices (RMs), which are the analogues of the RCs in the mono case. The RMs can be quantized in the following way. This derivation follows that in [51].

First, we denote the forward and backward RMs by $\mathbf{E_k}$ and $\mathbf{E'_k}$, respectively. The forward and backward Innovation Variance Matrices (IVMs) $\mathbf{V_k}$ and $\mathbf{V'_k}$, respectively, can be given by

$$\begin{aligned} \mathbf{V_k} &= \mathbf{V_{k-1}}\left(\mathbf{I} - \mathbf{E'_k E_k}\right) = \mathbf{V_{k-1}} - \mathbf{E_k^T V'_{k-1} E_k} \\ \mathbf{V'_k} &= \mathbf{V'_{k-1}}\left(\mathbf{I} - \mathbf{E_k E'_k}\right) = \mathbf{V'_{k-1}} - \mathbf{E_k'^T V_{k-1} E'_k} \end{aligned}, \tag{4.4}$$

with $\mathbf{I}$ the 2×2 identity matrix and $\mathbf{V_0} = \mathbf{R_0}$, where $\mathbf{R_0}$ is the correlation matrix defined in Section 2.2.3. The following relation holds for the RMs and IVMs

$$\mathbf{V_{k-1} E'_k} = \mathbf{E_k^T V'_{k-1}}. \tag{4.5}$$

If $\mathbf{V_k}$ and $\mathbf{V'_k}$ are factorized in the form

$$\begin{aligned} \mathbf{V_k} &= \mathbf{M_k^T M_k} \\ \mathbf{V'_k} &= \mathbf{M_k'^T M'_k} \end{aligned}, \tag{4.6}$$

for suitable 2×2 normalizing matrices $\mathbf{M_k}$ and $\mathbf{M'_k}$, then the 2×2 normalized Reflection Matrices (nRMs) $\xi_\mathbf{k}$ are given by

$$\xi_\mathbf{k} = \mathbf{M'_{k-1} E_k M_{k-1}^{-1}} = \left(\mathbf{M_{k-1}'^T}\right)^{-1} \mathbf{E_k'^T M_{k-1}^T}, \tag{4.7}$$

with a similar definition for $\xi'_\mathbf{k}$. If we now use (4.4) combined with (4.6) and (4.7), we get

$$\left[\left(\mathbf{M_{k-1}^T}\right)^{-1} \mathbf{M_k^T}\right] \left[\mathbf{M_k}\left(\mathbf{M_{k-1}}\right)^{-1}\right] = \mathbf{I} - \xi_\mathbf{k}^\mathrm{T} \xi_\mathbf{k}. \tag{4.8}$$

If we now define $\mathbf{M_k}$ from $\mathbf{M_{k-1}}$ such that $\mathbf{M_k M_{k-1}^{-1}}$ is a positive definite symmetric matrix satisfying (4.8), then we have

$$\begin{aligned} \mathbf{M_k} &= \left(\mathbf{I} - \xi_\mathbf{k}^\mathrm{T} \xi_\mathbf{k}\right)^{1/2} \mathbf{M_{k-1}} \\ \mathbf{M'_k} &= \left(\mathbf{I} - \xi_\mathbf{k} \xi_\mathbf{k}^\mathrm{T}\right)^{1/2} \mathbf{M'_{k-1}} \end{aligned}, \tag{4.9}$$

where the initial values are taken to be $\mathbf{M_0} = \mathbf{M'_0} = \mathbf{R_0^{1/2}}$.

Thus, the RMs $\mathbf{E_k}$ and $\mathbf{E'_k}$ can be mapped onto nRMs $\xi_\mathbf{k}$ and $\xi'_\mathbf{k}$, which are related according to $\xi'_\mathbf{k} = \xi_\mathbf{k}^\mathrm{T}$. For the inverse mapping, we need $\mathbf{R_0}$. This means that we only need to quantize and transmit the forward nRMs $\xi_\mathbf{k}$ and the correlation matrix $\mathbf{R_0}$. We now consider quantization of $\xi_\mathbf{1}$ and $\mathbf{R_0}$.

To quantize $\xi_\mathbf{1}$, we use a variant of Singular Value Decomposition to write $\xi_\mathbf{1}$ as

$$\xi_\mathbf{1} = \mathbf{R}(\alpha)\mathbf{S}\mathbf{R}(-\beta), \tag{4.10}$$

where $\mathbf{R}$ is a 2×2 rotation matrix with angles α and β, respectively and $\mathbf{S}$ is a 2×2 real diagonal matrix with

$$\begin{pmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{pmatrix}, \tag{4.11}$$

where $0 \leq |\sigma_2| \leq \sigma_1 < 1$. We now rewrite (4.10) to

$$\xi_\mathbf{1} = \mathbf{R}(\gamma)\mathbf{R}(\delta)\mathbf{S}\mathbf{R}(\delta)\mathbf{R}(-\gamma), \tag{4.12}$$

with $\gamma = (\alpha + \beta)/2$ and $\delta = (\alpha - \beta)/2$. The character of σ_1 and σ_2 is similar to that of an RC. Therefore, σ_1 and σ_2 can be quantized using LARs. The angles γ and δ are quantized such that the characteristic polynomial p of $\xi_\mathbf{1}$, given by

$$p = \lambda^2 - (\sigma_1 + \sigma_2)\cos(2\delta)\lambda + \sigma_1\sigma_2, \tag{4.13}$$

is represented as accurately as possible. We observe that p does not depend on γ, therefore, it is presumed that γ can be best quantized uniformly. We now define k_1 and k_2 as the RCs associated with p, given by

$$\begin{array}{c} k_1 = -\frac{\sigma_1+\sigma_2}{1+\sigma_1\sigma_2}\cos(2\delta) \\ k_2 = \sigma_1\sigma_2 \end{array} . \tag{4.14}$$

Since we already quantized σ_1 and σ_2, we only need to quantize k_1, with $|k_1| < 1$. Therefore, the LAR can be used to quantize k_1, because a RC also has as property $|RC| < 1$. In addition, we need a sign bit s to reconstruct δ from k_1, because $\cos(2\delta) = \cos(-2\delta)$. This strategy for quantization of ξ_1 can also be used for quantization of $\xi_\mathbf{k}$ of any order k [52].

To quantize $\mathbf{R_0}$, we write it as

$$\mathbf{R_0} = \begin{pmatrix} \rho_{11}(0,0) & \rho_{12}(0,0) \\ \rho_{21}(0,0) & \rho_{22}(0,0) \end{pmatrix} = \sqrt{\rho_{11}(0,0)\rho_{22}(0,0)} \begin{pmatrix} \eta & r \\ r & 1/\eta \end{pmatrix}, \tag{4.15}$$

where $\rho_{11}(0,0) \geq 0$, $\rho_{22}(0,0) \geq 0$ and $\rho_{12}(0,0) = \rho_{21}(0,0) = r\sqrt{\rho_{11}(0,0)\rho_{22}(0,0)}$ are the energy of signal x_1, the energy of signal x_2 and the cross-energy of the signals x_1 and x_2, respectively. The cross-correlation coefficient r has as property $|r| < 1$ and should be treated carefully when close to ± 1. Therefore, the LAR can be used to quantize r.

The ratio between the signal powers is given by η and is equal to

$$\eta = \sqrt{\frac{\rho_{11}(0,0)}{\rho_{22}(0,0)}}, \tag{4.16}$$

with $\eta \geq 0$. We now map the η to μ, which is given by

$$\mu = \frac{\eta - 1/\eta}{\eta + 1/\eta} = \frac{\rho_{11}(0,0) - \rho_{22}(0,0)}{\rho_{11}(0,0) + \rho_{22}(0,0)}, \tag{4.17}$$

with $|\mu| < 1$. The μ has to be treated carefully when $|\mu| \simeq 1$ and therefore, the LAR can also be used to quantize μ. The factor $\sqrt{\rho_{11}(0,0)\rho_{22}(0,0)}$ does not have to be quantized, because it gets cancelled while reconstructing $\mathbf{E_k}$ from $\xi_\mathbf{k}$.

It was shown in [52] that the strategy just described to quantize ξ_1 and $\mathbf{R_0}$, is a good strategy.

4.2 Quantization of the Main and Side Signal

Linear predictive speech coders usually use pulse coding (such as Regular Pulse Excitation (RPE) or Multi Pulse Excitation (MPE)) to code the error signal [3]. Preliminary investigations into audio coders using L-PLP and RPE, done by Riera-Palou et al. [53] [54], showed that, in general, this combination yields good results. However, audio signals containing clear tonal components can give problems.

Another kind of audio coders called parametric coders (such as the Sinusoidal Coder (SSC) [55]) decomposes the audio signal into transients, sinusoids and noise, and describes each component by a set of parameters. It was shown in [55] 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 L-PLP with RPE indicated that L-PLP with RPE is very suitable for those signals that are problematic for SSC and vice versa. This led to the idea of Riera-Palou et al. [53] [54] to combine SSC with L-PLP and RPE. Therefore, both Riera-Palou and Peter [56] implemented an audio coder according to this idea, the difference being that Riera-Palou placed the SSC block before the L-PLP and RPE blocks and Peter placed the L-PLP block before the SSC and RPE blocks. The scheme of Peter can be seen in Figure 4.1. His coder uses both RPE and the main block from SSC, which is Sinusoidal Extraction (SE), to code the error signal from L-PLP. The SE block consists of two blocks, which are the Sinusoidal Analyzer (SiA) and the Sinusoidal

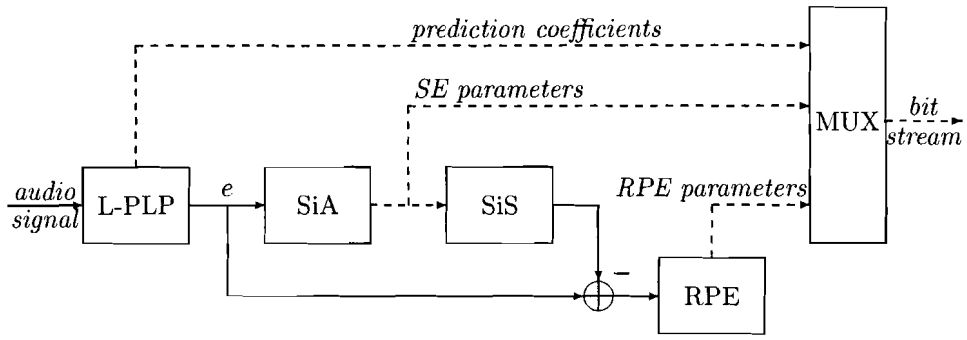

Figure 4.1: Coding scheme proposed by Peter [56]. It uses both RPE and SE to code the error signal from L-PLP.

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 can be obtained at relatively low bit rates. In addition, the audio sounds natural, but also sometimes a bit noisy.

Thus, inspired by the work of Riera-Palou and Peter, it was decided to use both SE and RPE to quantize the main and side signal. This makes SLP very similar to a linear predictive speech coder, except the Long Term Predictor (LTP, which function will be explained in Section 4.2.2) has been replaced by SE. SE and RPE are described in more detail in the next two sections and Section 4.2.3 describes the final design and settings of the system that quantizes the main and side signal.

4.2.1 Sinusoidal Extraction

Sinusoidal Extraction estimates the frequency, amplitude and phase of sinusoids in the input signal. A waveform is then generated based on these parameters, which is subtracted from the input signal, resulting in a residual signal. A detailed description of SE can be found in [57]. However, this technique is not adequate for estimating sinusoids in the spectrally flat error signals that result from L-PSLP. Therefore, Peter [56] implemented a redesigned SE, which is capable of selecting the right sinusoids and right number of sinusoids from a spectrally flat signal. This redesigned SE will now be described.

The redesigned SE, from now on simply called SE, uses three different algorithms to estimate the SE parameters. The first algorithm is used to estimate the low frequencies (less than 400 Hz), the second algorithm to estimate the high frequencies (greater than 400 Hz) and to select the right number of sinusoids, and the third algorithm to estimate the amplitudes and phases. Two different algorithms are used to estimate the low and the high frequencies, because otherwise, the low-frequency sinusoids cannot be estimated as a consequence of the compromise between time and frequency resolution [58]. The whole estimation procedure for the SE parameters can be seen in Figure 4.2. The three estimation algorithms will now be described. But first, we derive the Fourier Transform and the derivative of the Fourier Transform of the error signal, now denoted by r. With this and the description of the algorithms, we follow the same reasoning as in [56].

The Fourier Transform R_w of the windowed signal r_w, with

$$r_w[n] = r[n] \cdot w[n], \tag{4.18}$$

and w the window, is given by

$$R_w\left[e^{j\theta_k}\right] = \sum_n r_w[n] e^{-jn\theta_k}, \tag{4.19}$$

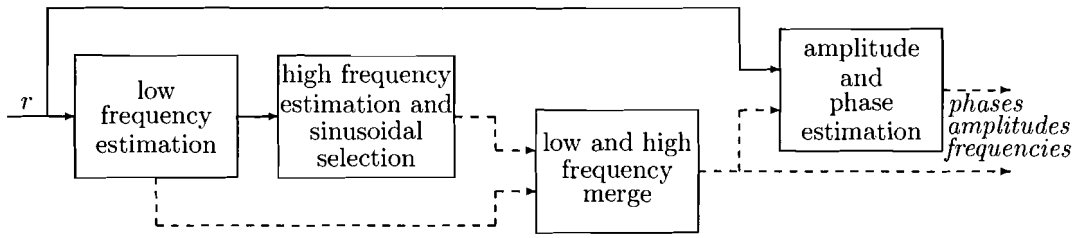

Figure 4.2: Procedure to estimate the SE parameters, which uses an algorithm to estimate the low frequencies, an algorithm to estimate the high frequencies and to select the right number of sinusoids, and an algorithm to estimate the amplitudes and phases.

where θ_k is a discrete set of frequencies. The derivative of R_w with respect to θ_k is given by

$$\frac{dR_w\left[e^{j\theta_k}\right]}{d\theta_k} = \frac{d\sum_n r_w[n]e^{-jn\theta_k}}{d\theta_k} = -j\sum_n nr_w[n]e^{-jn\theta_k} = -jY\left[e^{j\theta_k}\right], \tag{4.20}$$

where Y is the Fourier Transform of y, with

$$y[n] = nr_w[n]. \tag{4.21}$$

Thus, we have

$$Y\left[e^{j\theta_k}\right] = j\frac{dR_w\left[e^{j\theta_k}\right]}{d\theta_k}. \tag{4.22}$$

We can rewrite (4.20) to

$$\frac{dR_w\left[e^{j\theta_k}\right]}{d\theta_k} = e^{j\phi_i}\frac{d|R_w|}{d\theta_k} + jR_w\left[e^{j\theta_k}\right]\frac{d\phi_i}{d\theta_k}, \tag{4.23}$$

which leads to

$$Y\left[e^{j\theta_k}\right] = -R_w\left[e^{j\theta_k}\right]\frac{d\phi_i}{d\theta_k} + je^{j\phi_i}\frac{d|R_w|}{d\theta_k}. \tag{4.24}$$

Taking the complex conjugation and multiplying by $R_w\left[e^{j\theta_k}\right]$ gives

$$R_w\left[e^{j\theta_k}\right]Y^*\left[e^{j\theta_k}\right] = -|R_w|^2\frac{d\phi_i}{d\theta_k} - j|R_w|\frac{d|R_w|}{d\theta_k}, \tag{4.25}$$

where * denotes complex conjugation. We now define D as

$$D[k] = 2\cdot\Im\{R_w\left[e^{j\theta_k}\right]\cdot Y^*\left[e^{j\theta_k}\right]\}, \tag{4.26}$$

thus D can be written as

$$D[k] = -2|R_w|\frac{d|R_w|}{d\theta_k} = -\frac{d|R_w|^2}{d\theta_k}. \tag{4.27}$$

This means that searching for maxima of $|R_w|$ is equivalent to searching for zero-crossings of D. From now on, we write $R_w\left[e^{j\theta_k}\right]$ as $R_w[k]$.

The algorithm that estimates the low frequencies is the peak picking algorithm from [57]. It works as follows:

1. Determine R_w, Y, $|R_w|$ and D as described above.

2. Search for the roughly estimated candidate frequencies, which are the maxima of $|R_w|$. Index k indicates a maximum if

$$\begin{aligned}&(|R_w[k-1]| < |R_w[k]|) \wedge (|R_w[k+1]| < |R_w[k]|)\\ &\wedge\{(|R_w[k-2]| < |R_w[k-1]|) \vee (|R_w[k+2]| < |R_w[k+1]|)\},\end{aligned} \tag{4.28}$$

where $k = 0, 1, \ldots, N_{fft}$, with N_{fft} the length of the FFT.

3. A candidate is accepted if the derivative of $|R_w|$ has a zero crossing, thus

$$D[k] > 0 \wedge D[k+1] < 0. \tag{4.29}$$

4. Calculate a high-precision frequency estimate $\theta_{k,h}$ for the frequencies in θ_k that are accepted candidates. This high-precision frequency estimate is given by linear interpolation according to

$$\theta_{k,h} = \theta_k - \frac{D[k] \cdot (\theta_{k+1} - \theta_k)}{D[k+1] - D[k]}. \tag{4.30}$$

5. The high-precision frequency estimate is accepted if

$$(\theta_k - r_f < \theta_{k,h}) \wedge (\theta_k + r_f > \theta_{k,h}), \tag{4.31}$$

where r_f is the frequency resolution defined by $r_f = 2\pi/N$, with N the frame length of the input signal.

6. Order the accepted high-precision frequency estimates according to decreasing amplitude and select a number of frequencies according to a given threshold.

The algorithm that estimates the high frequencies and selects the right number of sinusoids is a redesigned version of the algorithm that estimates the low frequencies. It works as follows, where the first five steps are the same as above:

6. Calculate the second derivative D^2 for the frequencies in θ_k that are accepted high-precision frequency estimates. This second derivative is given by

$$D^2[k] = \frac{D[k+1] - D[k]}{\theta_{k+1} - \theta_k}. \tag{4.32}$$

7. Calculate for all the accepted high-precision frequency estimates the equivalent Bark bandwidth BW with the empirical formula [59]. The BW is given by

$$BW[k] = 25 + 75 \cdot \left(1 + 1.4 \cdot 10^{-6} \cdot \theta_{k,h}^2\right)^{0.69}. \tag{4.33}$$

8. Calculate the normalized second derivative D^2_{norm} around each accepted high-precision frequency estimate, given by

$$D^2_{norm}[k] = \frac{D^2[k]}{BW[k]}, \tag{4.34}$$

and order the accepted high-precision frequency estimates according to decreasing normalized second derivative.

9. Search for left side and right side valley data points for every k where an accepted high-precision frequency estimate exists. Data point $k - l$ is considered a left side valley at the vicinity of k if

$$(|R_w[k]| > |R_w[k-1]| > \cdots > |R_w[k-l]|) \wedge (|R_w[k-l]| < |R_w[k-l-1]|)\,. \tag{4.35}$$

Data point $k + l$ is considered a right side valley at the vicinity of k if

$$(|R_w[k]| > |R_w[k+1]| > \cdots > |R_w[k+l]|) \wedge (|R_w[k+l]| < |R_w[k+l+1]|)\,. \tag{4.36}$$

The general valley point around data point k is now the point having the maximum amplitude V, with V given by

$$V = \max\{|R_w[k-l]|, |R_w[k+l]|\}. \tag{4.37}$$

10. Determine the height H of the frequency peaks (in dB) for every accepted high-precision frequency estimates, given by

$$H[k] = 20 \cdot \log_{10} |R_w[k]| - 20 \cdot \log_{10} V. \tag{4.38}$$

11. Select the frequencies from the ordered accepted high-precision frequency estimates that satisfy $H[k] > H_{thres}$, where H_{thres} is a given threshold.

Before we describe the algorithm that estimates the amplitudes and phases, we first have to derive some relations for them. If we consider the signal r, we want to find an approximation $\hat{r}$ of this signal on the interval $[-(N-1)/2, (N-1)/2]$, where N is the length of the interval. The approximation is given by

$$\hat{r}[n] = A\cos(\omega_0 n + \phi), \tag{4.39}$$

where A, ω_0 and ϕ are the amplitude, frequency and phase, respectively. This approximation can be rewritten as

$$\hat{r}[n] = \Re\left\{a_0 e^{j\theta n}\right\}, \tag{4.40}$$

with $a_0 = Ae^{j\phi}$ and $\theta = \omega_0$. We now introduce the pattern p, given by

$$p[n] = e^{j\theta n}, \tag{4.41}$$

and its windowed version p_w, given by

$$p_w[n] = e^{j\theta n} \cdot w[n]. \tag{4.42}$$

To compute the complex amplitude a_0, we consider the approximation of signal r_w on the approximation interval given above by a linear combination of pattern functions p_w. This leads to the following optimization criterion E, given by

$$E = \sum_n |r_w[n] - (a_0 p_w[n] + a_0^* p_w^*[n])\,|^2. \tag{4.43}$$

Minimizing E with respect to a_0 leads to the normal equation

$$\left(\mathbf{P}_\mathbf{w}^\mathrm{T}\mathbf{P}_\mathbf{w}\right)\underline{A}_0 = \mathbf{P}_\mathbf{w}^\mathrm{T}\underline{r}_w, \tag{4.44}$$

with

$$\mathbf{P_w} = \begin{pmatrix} p_w[-(N-1)/2] & p_w^*[-(N-1)/2] \\ \vdots & \vdots \\ p_w[0] & p_w^*[0] \\ \vdots & \vdots \\ p_w[(N-1)/2] & p_w^*[(N-1)/2] \end{pmatrix}, \tag{4.45}$$

$$\underline{A}_0 = \begin{pmatrix} a_0 \\ a_0^* \end{pmatrix}, \tag{4.46}$$

and

$$\underline{r}_w = \begin{pmatrix} r_w[-(N-1)/2] \\ \vdots \\ r_w[0] \\ \vdots \\ r_w[(N-1)/2] \end{pmatrix}. \tag{4.47}$$

Solving (4.44) results in the complex amplitude a_0, and the amplitude A and phase ϕ are now given by

$$A = |a_0|, \tag{4.48}$$

and

$$\phi = \tan^{-1}\left(\frac{\Im\{a_0\}}{\Re\{a_0\}}\right), \tag{4.49}$$

respectively.

The algorithm that estimates the amplitudes and phases now works as follows:

1. Calculate p_w using (4.42).
2. Calculate a_0 by solving (4.44).
3. Calculate A and ϕ from (4.48) and (4.49), respectively.
4. Repeat the aforementioned steps for all selected frequencies.

In principle, joint optimization is required for all selected frequencies. However, to keep the computational complexity low, the aforementioned algorithm calculates the amplitudes and phases for one frequency at a time. We expect that this way of calculating will not lead to significant deviations to the calculated amplitudes and phases if the selected frequencies are distinct enough.

4.2.2 Regular Pulse Excitation

Regular Pulse Excitation [60] is a quantization technique that tries to model the input signal as an excitation that consists of equally spaced pulses with zeros in between. An important parameter of RPE is the decimation factor, which is equal to the number of zeros in between the pulses minus one. The pulses can have arbitrary amplitudes, or can be quantized by the RPE algorithm. RPE works best on spectrally flat signals. The optimal excitation, which will be sent to the decoder, is calculated such that the perceptual least-squares error at the decoder is minimized. For this purpose, an Analysis-by-Synthesis (AbS) scheme is used, that incorporates an encoder, a decoder and a weighting filter. The purpose of the weighting filter is to incorporate certain characteristics of the human auditory system into the least-squares error. The RPE AbS encoder scheme that

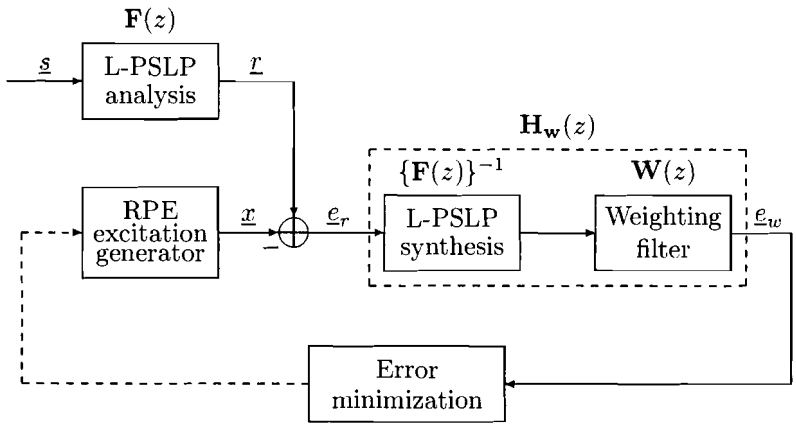

Figure 4.3: RPE AbS encoder scheme. It incorporates an encoder, a decoder and a weighting filter to calculate the optimal excitation, which will be sent to the decoder. The stereo input signal is denoted by $\underline{s}$, the stereo L-PSLP error signal by $\underline{r}$, the stereo RPE excitation by $\underline{x}$, the difference between $\underline{r}$ and $\underline{x}$ by $\underline{e}_r$ and the stereo perceptual error at the decoder by $\underline{e}_w$.

is used in SLP can be seen in Figure 4.3, where the rotators and inverse rotators are removed for convenience. The stereo input signal is denoted in this figure by $\underline{s}$, the stereo L-PSLP error signal by $\underline{r}$, the stereo RPE excitation by $\underline{x}$, the difference between $\underline{r}$ and $\underline{x}$ by $\underline{e}_r$ and the stereo perceptual error at the decoder by $\underline{e}_w$. The RPE AbS encoder scheme works on a frame-by-frame basis, with the frame length N chosen such that stationarity over the whole frame can be assumed.

We now want to derive an expression for the optimal excitation. First, we write the stereo perceptual error at the decoder $\underline{e}_w$ as

$$\underline{e}_w = \underline{e}_0 + \mathbf{H_w}\underline{e}_r, \tag{4.50}$$

where $\underline{e}_w$, $\underline{e}_0$ and $\underline{e}_r$ are vectors of the form

$$\underline{e}_w = \begin{pmatrix} e_{w,1}[0] \\ \vdots \\ e_{w,1}[N-1] \\ e_{w,2}[0] \\ \vdots \\ e_{w,2}[N-1] \end{pmatrix}, \tag{4.51}$$

$\underline{e}_0$ is a vector corresponding to the responses of filter $\mathbf{H_w}$ due to its initial filter states, truncated after N samples, $\underline{e}_r$ is a vector containing the difference between $\underline{r}$ and $\underline{x}$, and $\mathbf{H_w}$ is an $2N \times 2N$ matrix representing the impulse responses of filters $\mathbf{H_{w,11}}$, $\mathbf{H_{w,12}}$, $\mathbf{H_{w,21}}$ and $\mathbf{H_{w,22}}$, truncated to N samples. The matrix $\mathbf{H_w}$ is given by

$$\mathbf{H_w} = \begin{pmatrix} \mathbf{H_{w,11}} & \mathbf{H_{w,12}} \\ \mathbf{H_{w,21}} & \mathbf{H_{w,22}} \end{pmatrix}, \tag{4.52}$$

where the impulse responses $\mathbf{H_{w,11}}$, $\mathbf{H_{w,12}}$, $\mathbf{H_{w,21}}$ and $\mathbf{H_{w,22}}$ are of the form

$$\mathbf{H_{w,ij}} = \begin{pmatrix} h_{w,ij}[0] & 0 & \cdots & 0 & 0 \\ h_{w,ij}[1] & h_{w,ij}[0] & \ddots & \vdots & \vdots \\ \vdots & h_{w,ij}[1] & \ddots & 0 & \vdots \\ \vdots & \vdots & \ddots & h_{w,ij}[0] & 0 \\ h_{w,ij}[N-1] & h_{w,ij}[N-2] & \cdots & h_{w,ij}[1] & h_{w,ij}[0] \end{pmatrix}. \tag{4.53}$$

The filter $\mathbf{H_w}$ will be specified more precisely in the next section. If we now choose a decimation factor for the left channel of D_1, which means that only J_1 equidistant pulses are allowed per frame in the left channel ($D_1 = N/J_1$) and a decimation factor for the right channel of D_2, which means that only J_2 equidistant pulses are allowed per frame in the right channel ($D_2 = N/J_2$), then the pulse amplitudes $\underline{x}_p$ minimizing $||\underline{e}_w||^2$ are given by [60]

$$\underline{x}_p = \left(\mathbf{M}^T\mathbf{H}_\mathbf{w}^\mathrm{T}\mathbf{H_w}\mathbf{M}\right)^{-1}\left(\mathbf{M}^T\mathbf{H}_\mathbf{w}^\mathrm{T}\right)\left(\underline{e}_0 + \mathbf{H_w}\underline{r}\right), \tag{4.54}$$

where $\underline{x}_p$ is a vector of the form

$$\underline{x}_p = \begin{pmatrix} x_{p,1}[0] \\ \vdots \\ x_{p,1}[J_1 - 1] \\ x_{p,2}[0] \\ \vdots \\ x_{p,2}[J_2 - 1] \end{pmatrix}, \tag{4.55}$$

and $\mathbf{M}$ is a $2N \times (J_1 + J_2)$ location matrix containing zeros and ones, where ones indicate a position in the excitation sequence that is non-zero. The $2N$-sample excitation vector $\underline{x}$ can now be computed as

$$\underline{x} = \mathbf{M} \cdot Q[\underline{x}_p], \tag{4.56}$$

where Q denotes the quantization procedure. Thus, quantization of the pulses is embedded into the pulse optimization procedure, as presented in [53]. Then, different quantization techniques or grids can be evaluated and the one resulting in the lowest distortion $||\underline{e}_w||^2$ is selected.

There are two additional degrees of freedom when calculating the optimal pulse amplitudes, which are both related to the location matrix. This is because the first pulses of excitation sequences $\underline{x}_{p,1}$ and $\underline{x}_{p,2}$ can have an arbitrary position in the location matrix, ranging from the first to the D_1th and D_2th position, respectively. The optimal offsets are usually calculated by computing the optimal pulse amplitudes $\underline{x}_p$ and their associated distortions $||\underline{e}_w||^2$ for every possible offset combination and then choosing the optimal pulse amplitudes with the lowest distortion.

RPE is a technique that originally targeted narrow-band speech coding. Therefore, Riera-Palou et al. [53] introduced two enhancements to make RPE more suitable for general audio coding. These enhancements are improved optimization of pulse sequences and extra pulses, and will be described next.

As described above, the optimal RPE excitation for the current frame is calculated by minimizing the error $||\underline{e}_w||^2$ in the current frame. This results in expression (4.54) for the optimal pulse amplitudes. However, the error in the next frame is not independent from the error in de current frame, which is indicated by the term $\underline{e}_0$ in (4.50). This means that due to the initial filter states, the optimal excitation for frame k can induce a large error component in frame $k + 1$.

A solution for this problem is to define a new expression for the stereo perceptual error at the decoder [61], given by

$$\tilde{\underline{e}}_w = \mathbf{W}\left(\tilde{\underline{e}}_0 + \tilde{\mathbf{H}}_\mathbf{w}\tilde{\underline{e}}_r\right), \tag{4.57}$$

where $\mathbf{W}$ is a $2(N+F) \times 2(N+F)$ weighting matrix, used to weigh the importance of the induced error in future frames, with F indicating the additional length over which the error is measured. Variables $\tilde{\underline{e}}_0$, $\tilde{\mathbf{H}}_\mathbf{w}$ and $\tilde{\underline{e}}_r$ are identical to those in (4.50), except $\tilde{\underline{e}}_0$ and the filter responses in $\tilde{\mathbf{H}}_\mathbf{w}$ are now truncated after $N + F$ samples and $\tilde{\underline{e}}_r$ is now padded with F additional zeros. If we now minimize $||\tilde{\underline{e}}_w||^2$, then the optimal pulse amplitudes $\tilde{\underline{x}}_p$ are given by

$$\tilde{\underline{x}}_p = \left(\tilde{\mathbf{M}}^T\tilde{\mathbf{H}}_\mathbf{w}^\mathrm{T}\mathbf{W}^T\mathbf{W}\tilde{\mathbf{H}}_\mathbf{w}\tilde{\mathbf{M}}\right)^{-1}\left(\tilde{\mathbf{M}}^T\tilde{\mathbf{H}}_\mathbf{w}^\mathrm{T}\right)\mathbf{W}^T\mathbf{W}\left(\tilde{\underline{e}}_0 + \tilde{\mathbf{H}}_\mathbf{w}\tilde{\underline{r}}\right), \tag{4.58}$$

where extra zeros are inserted into variables $\tilde{\mathbf{M}}$ and $\underline{\tilde{r}}$. This expression is very similar to (4.54). The $4N$-sample excitation vector $\underline{\tilde{x}}$ can now be computed as

$$\underline{\tilde{x}} = \tilde{\mathbf{M}} \cdot Q[\underline{\tilde{x}}_p]. \tag{4.59}$$

It has to be noted that still only $J_1 + J_2$ pulse amplitudes are computed and that $\underline{\tilde{x}}$ has $2N$ non-zero samples that can be removed. This makes the improved optimization only marginally more complex.

The L-PSLP analysis filter is very effective in removing short-term correlations from the input signal, which results in a spectrally flat error signal. However, the input signal often exhibits also long-term correlations, which are not removed from the input signal. This results in a quasi-periodic pulse train in the error signal, which can be the reason for a poor compromise excitation. Therefore, a Long Term Predictor is usually used in linear predictive speech coders to pre-filter the error signal before the RPE stage. The LTP is a Linear Prediction analysis filter with a long tapped delay line.

In contrast to its success in speech coding, the LTP gain is low for audio signals. This is due to the presence of high frequencies, which obscure the quasi-periodic pulse train. Thus, the LTP cannot be used for audio signals. As a solution, the RPE enhancement extra pulses was proposed by Riera-Palou et al. [62], to model the pulse train. These pulses have a free gain, but are constrained to the RPE grid and are placed in the positions of the largest pulse amplitudes. This means that the RPE excitation is now given by

$$\underline{\tilde{x}}_{ext} = \tilde{\mathbf{M}} \cdot Q[\underline{\tilde{x}}_p] + \sum_{i=1}^{R_1} Q[g_{r,1}]\underline{p}(d_{r,1}) + \sum_{j=1}^{R_2} Q[g_{r,2}]\underline{p}(d_{r,2}), \tag{4.60}$$

where R_1 and R_2 are the number of extra pulses in the left and right channel, respectively, $\underline{p}$ is a $2(N+F)$-length vector of zeros, except at positions $d_{r,1}$ and $d_{r,2}$ for the left and right channel, respectively, where it has unity value, and $g_{r,1}$ and $g_{r,2}$ are the gains associated with the extra pulses in the left and right channel, respectively. It has to be noted that this RPE excitation can be seen as a combination of RPE and multi-pulse excitation.

4.2.3 System Design and Settings

Before the final design and settings of the system that quantizes the main and side signal could be made, several informal listening tests were performed to establish which parts of the main and side signal are essential to maintain a good quality of the reconstructed input signal. The starting point of these tests were the results from the informal listening tests performed by Selten [1]. These results showed that the side signal cannot be discarded completely, because there are some important components within the lower frequency region. In addition, they showed that Spectral Band Replication (SBR) can be applied to both the main and the side signal.

The first informal listening test in this project was performed to validate the statement that the side signal cannot be discarded completely. The results showed that this is indeed the case. They also showed that the important components of the side signal are limited to frequencies less than $\approx$4 kHz. In addition, the results showed that the side signal can be discarded completely for mono and mono-like signals. The second test was performed to find the best method to reconstruct the remaining part (greater than 4 kHz) of the side signal. The methods tried were SBR and calculating a synthetic side signal from the main signal by applying a Schröder filter (a decorrelation filter, which will be described more precisely later in this section) and a gain to the main signal, resulting in a synthetic side signal uncorrelated with the main signal and with the energy of the original side signal. The results showed that the second method gives a more stable stereo image in the reconstructed input signal. The last test was performed to validate the statement that SBR can be applied to the main signal. Again, the results showed that this is indeed the case. They also showed that if SBR is used to transmit only the lower frequency part, then the quality of the reconstructed input signal is still reasonably good, but not transparent anymore.

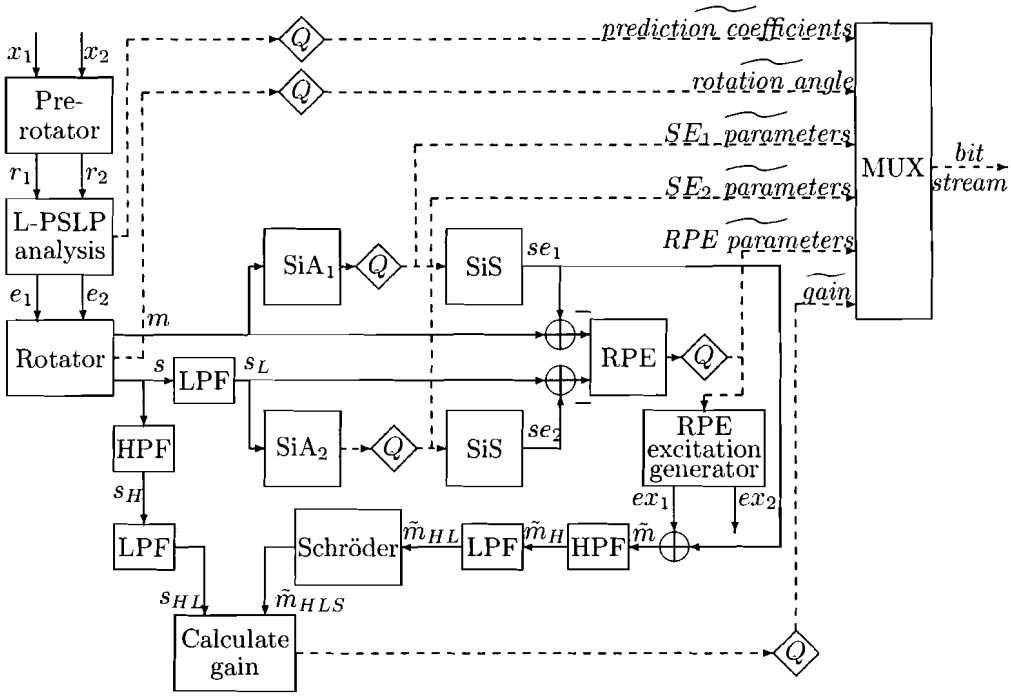

Figure 4.4: Full scheme of the SLP encoder, which uses two rotators and a L-PSLP analysis filter to construct the main and side signal, and SE, RPE and a gain factor to quantize the main and side signal.

Thus, with the results of the listening test in mind, it was decided to use SE and RPE to quantize the main signal, and for stereo signals to use SE and RPE to quantize the lower frequency part of the side signal (less than 4 kHz) and to use a synthetic side signal to reconstruct the remaining part of the side signal. For mono and mono-like signals, it was decided to only quantize the main signal, thus discarding the side signal completely. This resulted in the full scheme of the SLP encoder depicted in Figure 4.4 and the full scheme of the SLP decoder depicted in Figure 4.5. SE and RPE are combined in the encoder in the same way as in the scheme of Peter [56], depicted in Figure 4.1, but with some modifications. The full schemes of the encoder and decoder will now be described, concentrating on the part that quantizes the main and side signal in the encoder, and on the corresponding part in the decoder.

The main and side signal, m and s, respectively, are constructed by passing the stereo input signal through a pre-rotator, the L-PSLP analysis filter and a rotator, successively. These blocks were described already in Chapters 2 and 3. The L-PSLP analysis filter and the rotator use a window size of 2048 samples and an update rate of 256 to calculate the optimal coefficients. Because of bit rate constraints, the update rate should be raised to 1024 by interpolating the coefficients for the missing three frames in between, but it was not known yet how to interpolate the prediction coefficients. The order of the auto- and cross-predictors of L-PLSLP is equal to 15. Although setting $\lambda = 0.7564$ in the Laguerre filters, which were given by (3.30), results in a frequency scale which is closest to the Bark scale for a sample frequency of 44.1 kHz, this setting was not chosen, because numerical problems can occur when calculating the error signal for λ close to, or higher than 0.8. Therefore, it was proposed to use the setting of $\lambda = 0.7$ [56]. The factor ϵ_{rel} that is used in the regularization technique for L-PSLP in (3.64) is equal to 10^{-2}, and the smoothing factor γ in (3.70), used by Spectral Smoothing, is equal to 0.97. The first-order lowpass filter that is used to smooth the calculated optimal rotation angle uses a filter coefficient in the feedback loop of 0.95. After the rotator, the side signal is filtered by a lowpass filter with cutoff frequency 4 kHz, to make sure that only this part of the side signal is quantized. This results in

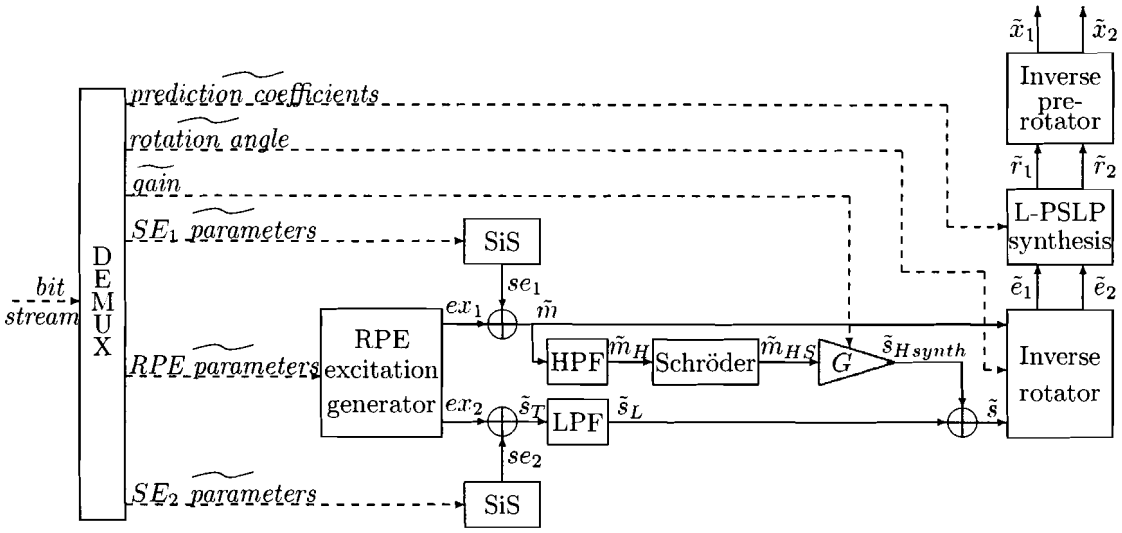

Figure 4.5: Full scheme of the SLP decoder, which uses the sinusoidal excitation, RPE excitation and gain factor to reconstruct the main and side signal, and two inverse rotators and the L-PSLP synthesis filter to reconstruct the original signal.

the lowpass filtered side signal s_L.

Now, the main signal goes through Sinusoidal Analyzer 1 (SiA_1), which extracts sinusoids with a maximum frequency of 16 kHz. The resulting sinusoidal parameters should then be quantized, for example as in the SSC coder [63] [64], which results in the quantized SE_1 parameters called $\widetilde{SE_1\ parameters}$. A Sinusoidal Synthesizer (SiS) then creates a sinusoidal excitation based on the quantized SE_1 parameters, which is subtracted from the main signal, resulting in a sinusoidal residual of the main signal. The lowpass filtered side signal goes through SiA_2, which extracts sinusoids with a maximum frequency of 4 kHz. The resulting sinusoidal parameters are quantized, which results in the quantized SE_2 parameters called $\widetilde{SE_2\ parameters}$. Now, an SiS creates a sinusoidal excitation based on the quantized SE_2 parameters, which is subtracted from the lowpass filtered side signal, resulting in a sinusoidal residual of the lowpass filtered side signal.

The sinusoidal residuals of the main and lowpass filtered side signal then go to the RPE block. This block was described already in Section 4.2.2, but the filter $\mathbf{H_w}$ will now be specified more precisely. The filter $\mathbf{H_w}$ implements the filters of the decoder, joined in $\{\mathbf{F}\}^{-1}$, followed by the weighting filter $\mathbf{W}$, where $\{\mathbf{F}\}^{-1}$ is equal to the inverse of the pre-rotator followed by the L-PSLP analysis filter, followed by the rotator, thus equal to the inverse rotator, followed by the L-PSLP synthesis filter, followed by the inverse pre-rotator. In addition, the side signal reconstructed from SE and RPE is lowpass filtered in the decoder, because only the frequency part less than 4 kHz is used there. This operation also has to be included in $\{\mathbf{F}\}^{-1}$. The weighting filter $\mathbf{W}$ implements certain characteristics of the human auditory system. That is, its frequency response is the inverse of the spectral masking curve, calculated by a Psycho-Acoustic Model (PAM). However, a Stereo Psycho-Acoustic Model (S-PAM) does not exist currently. Therefore, an approximation was used that assumes that the S-PAM for the left ear depends on the left channel of the input signal and not on the right channel of the input signal and thus, is independent from the S-PAM for the right ear, which depends on the right channel of the input signal and not on the left channel of the input signal. It is known that this approximation is not very close to the practical situation, but it is the best that could be done. Thus, the spectral masking curve for the left channel is derived from the envelope of the left channel of the input signal and the threshold in quiet, and the spectral masking curve for the right channel is derived from the envelope of the right channel

of the input signal and the threshold in quiet, where the threshold in quiet T_q is given by [2]

$$T_q(f) = 3.64(f/1000)^{-0.8} - 6.5e^{-0.6(f/1000-3.3)^2} + 10^{-3}(f/1000)^4. \tag{4.61}$$

However, the expression for the threshold in quiet that was used in the final weighting filter was changed to

$$T_q(f) = \begin{cases} 3.64[(f+48.51)/1000]^{-0.8} - 6.5e^{-0.6(f/1000-3.3)^2} \\ \qquad +10^{-3}(f/1000)^{3.5} & \text{if } 0 < f \leq 13867; \\ 3.64[(f+48.51)/1000]^{-0.8} - 6.5e^{-0.6(f/1000-3.3)^2} \\ \qquad +10673 \cdot 10^{-3}[(f-13230)/1000]^{0.16} & \text{if } 13867 < f \leq 22050. \end{cases} \tag{4.62}$$

Otherwise, certain frequency ranges will be completely ignored in the modelling error. This can result in an RPE excitation with relatively much energy in the lower and higher frequencies. The final filter $\mathbf{H_w}$ can now be given by

$$\begin{aligned} \mathbf{H_w} = & \begin{pmatrix} \mathbf{W_1} & \varnothing \\ \varnothing & \mathbf{W_2} \end{pmatrix} \cdot \begin{pmatrix} \cos(-\pi/4)\cdot\mathbf{I} & \sin(-\pi/4)\cdot\mathbf{I} \\ -\sin(-\pi/4)\cdot\mathbf{I} & \cos(-\pi/4)\cdot\mathbf{I} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{S_{11}} & \mathbf{S_{12}} \\ \mathbf{S_{21}} & \mathbf{S_{22}} \end{pmatrix} \\ & \cdot \begin{pmatrix} \cos(-\varphi)\cdot\mathbf{I} & \sin(-\varphi)\cdot\mathbf{I} \\ -\sin(-\varphi)\cdot\mathbf{I} & \cos(-\varphi)\cdot\mathbf{I} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{I} & \varnothing \\ \varnothing & \mathbf{LP} \end{pmatrix}, \end{aligned} \tag{4.63}$$

where $\mathbf{W_1}$ and $\mathbf{W_2}$ are the weighting filters for the left and right channel, respectively, $\mathbf{I}$ is the identity matrix, $\mathbf{S_{11}}$, $\mathbf{S_{12}}$, $\mathbf{S_{21}}$ and $\mathbf{S_{22}}$ are the L-PSLP synthesis filters for the left to the left channel, for the right to the left channel, for the left to the right channel and for the right to the right channel, respectively, and $\mathbf{LP}$ is the lowpass filter already mentioned. All matrices just mentioned are $(N+F) \times (N+F)$ matrices and $\mathbf{W_1}$, $\mathbf{W_2}$, $\mathbf{S_{11}}$, $\mathbf{S_{12}}$, $\mathbf{S_{21}}$, $\mathbf{S_{22}}$ and $\mathbf{LP}$ are of the form given by (4.53), but the filter responses are now truncated after $N+F$ samples. The result of the RPE block is a set of parameters which describe the RPE excitation. These parameters are then quantized, which results in the quantized RPE parameters called *RPE parameters*.

The quantization procedure in (4.59) was chosen to be the same as in [53] and [56]. This means that the pulses can only have three possible levels: +1, 0 and -1, and a gain is computed for each frame, which scales the quantized pulses such that the error with respect to the signal to be modelled is minimized in a least-squares sense.

The RPE block uses a decimation factor for the main signal of 2 and a decimation factor for the side signal of 5. The number of extra pulses is equal to 5 for the main signal and equal to 1 for the side signal. These numbers of extra pulses were chosen, because less extra pulses had a negative effect on the quality of the output signal, but more extra pulses did not lead to a big improvement of the quality of the output signal. Parameter F, which indicates the additional length over which the error $||\tilde{e}_w||^2$ is measured, is equal to N and the induced error in the next frame is taken to be as important as the induced error in the current frame. Thus, the weighting matrix $\mathbf{W}$ in (4.57) can be removed.

It has to be noted that the RPE block calculates the optimal excitation such that the perceptual least-squares error at the decoder is minimized. Thus, the RPE block looks at the output signal of the decoder and not at the input signal of the RPE block (the main and side signal) when calculating the optimal excitation signal. This means that it can happen that for one non-zero and one zero input channel (for example a main and a zero side signal), the optimal excitation can have pulses in both channels. Thus, for mono and mono-like signals, where the side signal is removed, it is not sufficient to make the side signal zero before it is sent to the RPE block, because the output of the RPE block can still have pulses in both channels. A better solution is to explicitly forbid the RPE block to put pulses in the side signal, by changing the filter $\mathbf{H_w}$.

Now, the quantized RPE parameters are used to make a RPE excitation for the main signal ex_1, which, combined with the sinusoidal excitation for the main signal, results in the reconstructed main signal $\tilde{m}$. This signal is used, together with the side signal, to calculate the gain that is

needed to preserve the original energy in the higher frequencies of the reconstructed side signal, when calculating the synthetic side signal from the main signal. Both signals are first highpass filtered with a cutoff frequency of 4 kHz and then lowpass filtered with a cutoff frequency of 18 kHz, resulting in two signals with the right frequency range to calculate the gain. The filtered reconstructed main signal $\tilde{m}_{HL}$ then goes through a Schröder filter, which is a decorrelation filter, because the synthetic side signal will be calculated from the main signal with this filter in the decoder. The Schröder filter will be described more precisely later in this section, when the full scheme of the decoder is described. It is also explained then why this filter is used in the decoder. The next step is to calculate the gain g by

$$g = \sqrt{\frac{E[s_{HL}]}{E[\tilde{m}_{HLS}]}}, \tag{4.64}$$

where E denotes the signal energy. The final step in the encoder is to quantize the gain with an uniform or non-uniform quantizer.

The decoder uses the quantized RPE parameters to make the RPE excitation and the quantized SE parameters to make the sinusoidal excitations. These excitations are added, which results in the reconstructed main signal $\tilde{m}$ and the temporary reconstructed side signal $\tilde{s}_T$. The temporary reconstructed side signal is then lowpass filtered to obtain $\tilde{s}_L$, because it is only used for frequencies less than 4 kHz. Now, the synthetic side signal is made by highpass filtering the main signal, because it is only used for frequencies greater than 4 kHz. The resulting signal then goes through a decorrelation filter. This filter is needed, because the synthetic side signal has to be uncorrelated with the main signal, from a fine-structure waveform point of view. This is necessary for the stability of the stereo image in the resulting output signal. The filter used is a Schröder filter, which is an allpass decorrelation filter with a frequency-dependent delay. It exhibits low autocorrelation at nonzero lags. Therefore, it is suitable to construct the synthetic side signal. A Schröder filter consists of a single period of a positive Schröder complex [65]. Its impulse response h_d for $0 \leq n \leq N_s - 1$ is given by

$$h_d[n] = \sum_{k=0}^{N_s/2} \frac{2}{N_s} \cos\left(\frac{2\pi kn}{N_s} + \frac{2\pi k(k-1)}{N_s}\right), \tag{4.65}$$

where N_s is the length of the period, which is equal to 640. The last step in constructing the synthetic side signal $\tilde{s}_{Hsynth}$ is to apply the gain to the Schröder filtered signal, to preserve the original energy in the higher frequencies of the reconstructed side signal. This reconstructed side signal $\tilde{s}$ is now the sum of $\tilde{s}_L$ and $\tilde{s}_{Hsynth}$. The reconstructed main and side signal are then passed through the inverse rotator, the L-PSLP synthesis filter and the inverse pre-rotator, which results in the reconstructed input signals $\tilde{x}_1$ and $\tilde{x}_2$. These three blocks were also described already in Chapters 2 and 3.

4.3 Summary

We showed in this chapter how the stereo prediction coefficients can be mapped to a different representation before they are quantized. This results in a smaller distortion of the analysis and synthesis filter characteristics. Thereafter, we described how both SE and RPE can be used to quantize the main and side signal, making SLP very similar to a linear predictive speech coder. We chose this combination, because Riera-Palou et al. [53] [54] and Peter [56] showed that RPE with SE is a good combination to code the error signal from L-PLP, and that excellent quality can be obtained with this combination at relatively low bit rates.

We have now described our proposal for a stereo audio coder, which we called SLP, how SLP can be made more psycho-acoustically inspired by incorporating Laguerre filters into its scheme, and how the prediction coefficients and the main and side signal can be quantized. In the next chapter, we will describe the remaining issue of this thesis, which is evaluation of the performance

of SLP. This evaluation includes an estimation of the bit rate, the results of a formal listening test to determine the subjective quality of the coded material, and an evaluation of the coding delay, the computational complexity and the scalability of the coder.

Chapter 5

Performance of the Coder

When evaluating the performance of a coder, several attributes are important. These attributes include bit rate, subjective quality of the coded material, coding delay, computational complexity and scalability. This chapter evaluates the performance of the SLP coder by considering the aforementioned attributes. The first section gives a rough estimate of the bit rate of the SLP coder. The second section determines the subjective quality of the coded material by using a formal listening test. The last section considers the three other attributes, which are coding delay, computational complexity and scalability.

5.1 Bit Rate

The bit rate of the SLP coder consists of three parts: the quantized prediction coefficients and rotation angle, the quantized main signal, and the quantized side signal. Thus, to estimate the bit rate of the whole SLP coder, the bit rates of the three different parts should be analyzed. These bit rate estimations are based on entropy-coding of the quantized data. The resulting bit rates were not measured and it was not the purpose of this project to find the optimal quantization method for the different parameters. Therefore, only a rough estimate of the bit rates can be given.

Unfortunately, it is not possible yet to give an accurate estimate of the bit rate needed by the quantized prediction coefficients, because no quantization experiments have been performed for L-PSLP coefficients of order 15 yet. However, we do know that the bit rate needed by the rotation angle will be insignificant compared to the overall bit rate. This is because the rotation angle is lowpass filtered, resulting in a slowly-varying rotation angle. This means that 2 bits should be sufficient to code one rotation angle. We have approximately $44100/256 = 172$ frames per second, which results in a bit rate of approximately $172 \cdot 2 = 344$ bps for the rotation angle. To make a rough estimate of the bit rate needed by the quantized prediction coefficients, we can use data that we know from one channel linear prediction. These data indicate that 3 bits suffice for quantizing one L-PLP coefficient, when entropy coding is applied [66]. We use a total of $15 \cdot 4 = 60$ L-PSLP coefficients and have approximately $44100/1024 = 43$ frames per second, which gives a bit rate of approximately $43 \cdot 60 \cdot 3 = 8$ kbps. We estimate the bit rate of the overhead (the header) to be 2 kbps, which results in 10 kbps as a very rough estimate of the bit rate needed by the quantized prediction coefficients.

The bit rate needed by the quantized main signal consists of two components: the bit rate needed by the Sinusoidal Excitation parameters and the bit rate needed by the Regular Pulse Excitation parameters. Peter [56] made an estimate of these bit rates for his one channel L-PLP audio coder. Because the main signal can be seen as a normal one channel Linear Prediction error signal, we can use his method to give an estimate for the bit rates needed by the SE parameters and the RPE parameters.

The bit rate needed by the SE parameters is a function of the number of sinusoids extracted

Code	Name	Duration (s)	Stereo Character	N_{sins} Main	Side
es01	Suzanne Vega	10.7340	Mono-like	9917	-
es02	German male	8.5998	Mono	7620	-
es03	English female	7.6043	Mono	4489	-
sc01	Trumpet	10.9688	Stereo	10557	5123
sc03	Pop	11.5520	Stereo	17544	7442
si01	Harpsichord	7.9954	Stereo	21735	7467
sm01	Bagpipe	11.1487	Stereo	24494	13022

Table 5.1: The number of sinusoids N_{sins} in seven excerpts.

per second. Three different sinusoids can be distinguished: absolute births, relative births and continuations. Births are sinusoids that are new in a particular frame, thus were not present in the frame before, where an absolute birth is the birth in a frame with the lowest frequency and the relative births are all other births in that frame. Continuations are sinusoids that are linked to another sinusoid in the previous frame. Every sinusoid is characterized by three parameters, which are the frequency, amplitude and phase. This means that the bit rate needed for encoding the sinusoids BR_{sin} is given by

$$BR_{sin} = \frac{n_{ab}(b_{af} + b_{aa} + b_{ap})}{t_{excerpt}} + \frac{n_{rb}(b_{rf} + b_{ra} + b_{rp})}{t_{excerpt}} + \frac{n_c(b_{cf} + b_{ca} + b_{cp})}{t_{excerpt}}, \tag{5.1}$$

where n_{ab}, n_{rb} and n_c are the total number of absolute births, relative births and continuations, respectively; b_{af}, b_{rf} and b_{cf} are the average number of bits needed for encoding the frequency of one absolute birth, one relative birth and one continuation, respectively; b_{aa}, b_{ra} and b_{ca} are the average number of bits needed for encoding the amplitude of one absolute birth, one relative birth and one continuation, respectively; b_{ap}, b_{rp} and b_{cp} are the average number of bits needed for encoding the phase of one absolute birth, one relative birth and one continuation, respectively; $t_{excerpt}$ is the length of the audio excerpt. The overall bit rate needed by the SE parameters BR_{SE} can now be given by

$$BR_{SE} = BR_{sin} + BR_{header}, \tag{5.2}$$

where BR_{header} is the bit rate of the header, which transmits the number of births occurring in every frame. This bit rate is given by

$$BR_{header} = \frac{n_{frames} \cdot b_{births}}{t_{excerpt}}, \tag{5.3}$$

where n_{frames} and b_{births} are the total number of frames and the average number of bits needed for encoding the number of births in a frame, respectively.

To give an estimate of the bit rate needed by the SE parameters in his coder, Peter [56] calculated all parameters involved for seven excerpts typically used in MPEG listening tests. To give an estimate of the bit rate needed by the SE parameters in the SLP coder, the number of sinusoids N_{sins} was calculated for the same excerpts. The only difference being that these excerpts were now stereo. The results obtained can be seen in Table 5.1. The number of sinusoids in the main signal is in the same range as the number of sinusoids that Peter calculated for all excerpts. Therefore, we assume that the other parameters are also in the same range and we can use his estimations for the bit rates of the SE parameters. If we average these estimations over all excerpts involved, then we arrive at the bit rate needed by the SE parameters for the main signal, which equals 9 kbps.

Similar to the bit rate needed by the SE parameters, the bit rate needed by the RPE parameters BR_{RPE} is given by

$$BR_{RPE} = BR_{amp} + BR_{header}, \tag{5.4}$$

where BR_{amp} and BR_{header} are the bit rates needed to encode the amplitudes of the RPE pulses and the bit rate of the header, respectively. The pulses can have only three different amplitudes (+1, 0 and -1), which means that the BR_{amp} is given by

$$BR_{amp} = F_s \cdot \frac{1.5}{D}, \tag{5.5}$$

where F_s and D are the sampling frequency and the decimation factor, respectively and 1.5 is the average number of bits needed to represent the amplitude of one pulse [67]. The bit rate of the header is given by

$$BR_{header} = BR_{gain} + BR_{offset} + BR_{epulses}, \tag{5.6}$$

where BR_{gain}, BR_{offset} and $BR_{epulses}$ are the bit rate needed to encode the gain applied to the pulses, the bit rate needed to encode the offset of the pulses and the bit rate needed to encode the extra pulses, respectively. The BR_{gain} is given by

$$BR_{gain} = F_s \cdot \frac{6}{N_f}, \tag{5.7}$$

with 6 the average number of bits needed to represent one gain [67], and N_f the frame length. The BR_{offset} is given by

$$BR_{offset} = F_s \cdot \frac{\log_2(D)}{N_f}, \tag{5.8}$$

and the $BR_{epulses}$ is estimated to be 1 kbps per extra pulse [56].

RPE works on the main signal with a decimation factor of 2, with a frame length of 256 and uses five extra pulses, which results in a bit rate needed by the RPE parameters for the main signal of 39 kpbs. Thus, the total bit rate needed by the quantized main signal equals 48 kbps.

The bit rate needed by the quantized side signal consists of three components: the bit rate needed by the SE parameters, the bit rate needed by the RPE parameters, and the bit rate needed by the extra gain, which is used for the synthetic side signal. This gain is assumed to have a similar characteristic as the gain applied to the pulses in RPE and is also transmitted once per frame, which has as size of 256 samples. Therefore, the bit rate of the extra gain is assumed to be equal to the bit rate of the gain applied to the pulses in RPE, which was 1 kbps. The side signal can also be seen as a normal one channel Linear Prediction error signal and therefore, we use the method of Peter again to give an estimate for the bit rates needed by the SE parameters and the RPE parameters.

To give an estimate of the bit rate needed by the SE parameters for the side signal, the number of sinusoids in the side signal was calculated for the same excerpts as were used to calculate the number of sinusoids in the main signal, with the exception of excerpts Suzanne Vega, German male and English female, because these excerpts do not use a side signal. The results can be seen in Table 5.1. The average number of sinusoids in the side signal is approximately half the average number of sinusoids in the main signal. For the main signal, we had on average 1400 sinusoids per second. This means that for the side signal, we have on average 700 sinusoids per second. This number is in between the average number of sinusoids per second of excerpts Suzanne Vega and English female in the results of Peter [56]. Therefore, we use a bit rate that is also in between the bit rates of these excerpt, as the bit rate needed by the SE parameters for the side signal. This bit rate is now estimated to be approximately 5 kbps.

It was described above how the bit rate needed by the RPE parameters can be calculated. RPE works on the side signal with a decimation factor of 5, with a frame length of 256 and uses

one extra pulse, which results in a bit rate needed by the RPE parameters for the side signal of 16 kpbs. Thus, the total bit rate needed by the quantized side signal equals 22 kbps.

The estimate of the total bit rate of the SLP coder can now be given as the sum of the bit rates of the three different parts, which results in a bit rate of 80 kbps for stereo signals and a bit rate of 58 kbps for mono or mono-like signals. The calculations described above to arrive at these bit rates, are summarized in Table 5.2.

Part			Bit Rate (kbps)	Remarks	Reference
Param.			**10**		
	Rot. angle		0.3	$N_f = 256$ 2 b/angle	
	Pred. coeff.		10	Order = 60 $N_f = 1024$ 3 b/coeff.	[66]
Main			**48**		
	SE		9	N_{sins} same as Peter	[56]
	RPE		39		
		RPE_{amp}	33	$D = 2$ Levels = 3 1.5 b/amplitude	[67]
		RPE_{gain}	1	$N_f = 256$ 6 b/gain	[67]
		$\text{RPE}_{\text{offset}}$	0.2	$N_f = 256$ 1 b/offset	
		$\text{RPE}_{\text{epulses}}$	5	5 Pulses 1 kb/pulse	[56]
Side			**22**		
	Gain		1	$N_f = 256$ 6 b/gain	
	SE		5	N_{sins} half of Peter	[56]
	RPE		16		
		RPE_{amp}	13	$D = 5$ Levels = 3 1.5 b/amplitude	[67]
		RPE_{gain}	1	$N_f = 256$ 6 b/gain	[67]
		$\text{RPE}_{\text{offset}}$	0.4	$N_f = 256$ 2.3 b/offset	
		$\text{RPE}_{\text{epulses}}$	1	1 Pulse 1 kb/pulse	[56]
Total			**80** **58**	**Stereo** **Mono**	

Table 5.2: Summary of the bit rate calculations.

5.2 Subjective Quality

One of the criteria that was used to design SLP was that the coder should deliver a subjective audio quality which is (almost) equal to the subjective audio quality delivered by state-of-the-

art low bit rate mono coders, which use OCS for the stereo image. We choose Advanced Audio Coding (AAC) as low bit rate mono coder to make this comparison, because it is one of the most popular audio formats available. AAC aims for International Telecommunications Union (ITU)-R indistinguishable quality at 64 kbps per mono channel. If the stereo image is coded with 8 kbps OCS, then the total stereo bit rate equals 72 kbps. This means that the bit rate of SLP is only slightly higher. Whether the subjective quality of the coded material of SLP is also similar to that of AAC still remains to be seen of course. This question will be answered in the next three sections. The first section describes the setup of the formal listening test that was performed to determine the subjective quality of the coded material, the second section gives the results of this listening test and the last section discusses these results.

5.2.1 Listening Test Setup

The purpose of the formal listening test was to have an indication of the subjective quality of the coded material of SLP with respect to the subjective quality of the coded material of AAC. The test was performed in a quiet listening room and the excerpts were presented through high-quality headphones (Beyer-Dynamic DT990 PRO). The test employed MUSHRA methodology (**MU**lti **S**timulus test with **H**idden **R**eference and **A**nchors) [68]. In a MUSHRA test, the listener is presented with a known original version (which can be listened to at any time) and several blind versions, for each excerpt to be tested. The blind versions include a hidden reference, anchors, and versions coded by the different coders to be tested. The order of the excerpts is different for each listener and the order of the blind versions is different for each excerpt and for each listener. The listener has to rank each blind version on a quality scale from 0 to 100. The software tool used to perform the test was the Integrated Listening Test Tool (ILTT) from Philips Digital Systems Labs.

Six listeners participated in the listening test. Their ages ranged from 24-39 years and they all had a musical background and a normal hearing.

The test excerpts were eleven excerpts typically used in MPEG listening tests. They can be seen in Table 5.3. All items were stereo, 16 bits per sample, at a sampling frequency of 44.1 kHz. The MPEG excerpt Castanets was not used, because it was known in advance that SLP does not produce an output signal with a high subjective quality for this excerpt. This can be attributed to the fact that Castanets contains a lot of transients and a transient detection and simulation algorithm or something similar is not implemented in SLP.

The blind versions coded by different coders were a version coded by SLP and two versions coded by AAC. The version coded by SLP was made by quantizing the RPE parameters and the

Code	Name	Duration (s)	Stereo Character	Contents
es01	Suzanne Vega	10.7340	Mono-like	Speech
es02	German male	8.5998	Mono	Speech
es03	English female	7.6043	Mono	Speech
sc01	Trumpet	10.9688	Stereo	Multiple instruments
sc02	Orchestra	12.7322	Stereo	Multiple instruments
sc03	Pop	11.5520	Stereo	Multiple instruments
si01	Harpsichord	7.9954	Stereo	Single instrument
si03	Pitch pipe	27.8871	Stereo	Single instrument
sm01	Bagpipe	11.1487	Stereo	Single instrument
sm02	Glockenspiel	10.0954	Stereo	Single instrument
sm03	Plucked strings	13.9857	Stereo	Single instrument

Table 5.3: Excerpts used in the listening test.

gain, but without quantizing the prediction coefficients, the rotation angle and the SE parameters, because it was not known how to do this, or the code was not available to do this. However, the subjective quality of the output signal is mainly determined by RPE, because this block quantizes everything that has not been filtered out by the L-PSLP block and the SE blocks. Thus, it is not expected that the subjective quality of the output signal is much lower when all parameters are quantized. If the excerpt was mono or mono-like, then the side signal was removed. The two versions coded by AAC were coded by a High Efficiency AAC (HE-AAC) coder from Coding Technologies. The HE version of AAC was used, because this version uses Spectral Band Replication (SBR) when coding the audio signals. This technique is also used by SLP (in the RPE block, because the decimation factors are greater than 1). To be able to use OCS as the stereo coding technique in HE-AAC, the V2 version of HE-AAC has to be used. However, the maximum bit rate that is possible with this version is 48 kbps. Therefore, the first AAC version was coded by HE-AAC-V2 at 48 kbps. The second AAC version was coded by HE-AAC using a normal stereo coding technique at the same bit rate as SLP for stereo signals, which is 80 kbps. The anchors were 7 and 10 kHz lowpass filtered versions of the original signal.

5.2.2 Listening Test Results

The results of the listening test, averaged across all listeners and all mono and mono-like excerpts per coder, are given in Figure 5.1. The asterisks denote the mean MUSHRA scores and the error bars represent the 95% confidence intervals. It can be seen that the scores of HE-AAC at 80 kbps (HEAAC@80) and HE-AAC-V2 at 48 kbps (HEAACV2@48) are almost equal, which means that these coders result in almost equal subjective quality for mono or mono-like signals. It can also be seen that the score of the hidden reference (Hid.Ref.) is not much higher and that the confidence interval is not overlapping with 100, indicating that the hidden reference was fairly often confused with one of the coded versions. This means that HEAAC@80 and HEAACV2@48 are almost transparent for mono or mono-like signals. The score of SLP at 58 kbps (SLP@58) is equal to approximately 60, indicating that the subjective quality of SLP@58 for mono or mono-like signals is between fair and good. The last thing that can be seen is that the score of the 10 kHz anchor (Anc.10kHz) is almost equal to the scores of HEAAC@80 and HEAACV2@48, but that the score of the 7 kHz anchor (Anc.7kHz) is even lower than the score of SLP@58.

The results of the listening test, averaged across all listeners and all stereo excerpts per coder, are given in Figure 5.2. It can be seen that compared to the previous figure, the score of HEAACV2@48 has dropped substantially, indicating that HEAACV2@48 has problems with coding the stereo image. The score of HEAAC@80 has also dropped, but only slightly and the score of the hidden reference is now a little bit higher. This means that now more listeners spotted the hidden reference and that HEAAC@80 is less transparent for stereo signals. It can also be seen that the score of SLP at 80 kbps (SLP@80) is equal to the score of SLP@58 from the previous figure. This indicates that for mono or mono-like signals, the side signal can be removed without any loss in subjective quality, as was expected. In addition, the score of SLP@80 is now equal to the score of HEAACV2@48, because of the drop in subjective quality of HEAACV2@48 for stereo signals. However, listeners indicated that these equal scores were not caused by equal types of coding artifacts. While HEAACV2@48 mainly suffered from stereo image problems, the coding artifacts in SLP@80 were mainly attributed to quantization noise. The last thing that can be seen is that the score of the 10 kHz anchor is now a little bit lower, compared to the previous figure, but that the score of the 7 kHz anchor is now a little bit higher. However, the 7 kHz anchor is still rated lower than SLP@80.

The listening test results for the SLP coder, averaged across all listeners per excerpt, are given in Figure 5.3. This figure shows fairly broad confidence intervals, due to the fact that different listeners rate the same artifacts with different scores. This is not unusual when testing low bit rate coders. It can be seen that SLP@80 performs very well for Trumpet and especially Orchestra, which are complex excerpts with multiple instruments. However, the resulting subjective quality of Pop is not good, which is also a complex excerpt with multiple instruments. It can also be seen that the results of Harpsichord, Pitch pipe and Bagpipe, which are excerpts with a single

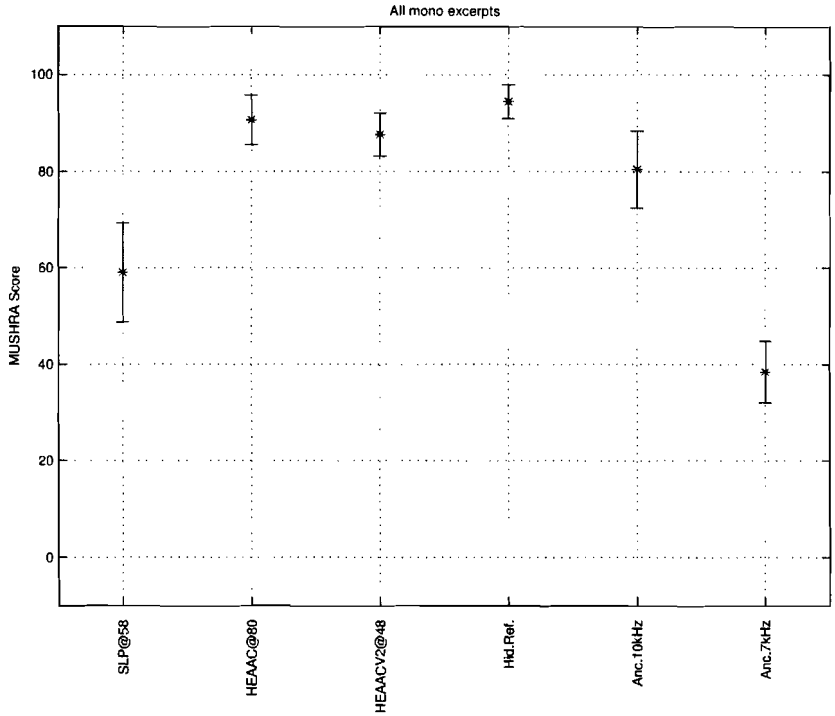

Figure 5.1: Listening test results per coder for all mono excerpts.

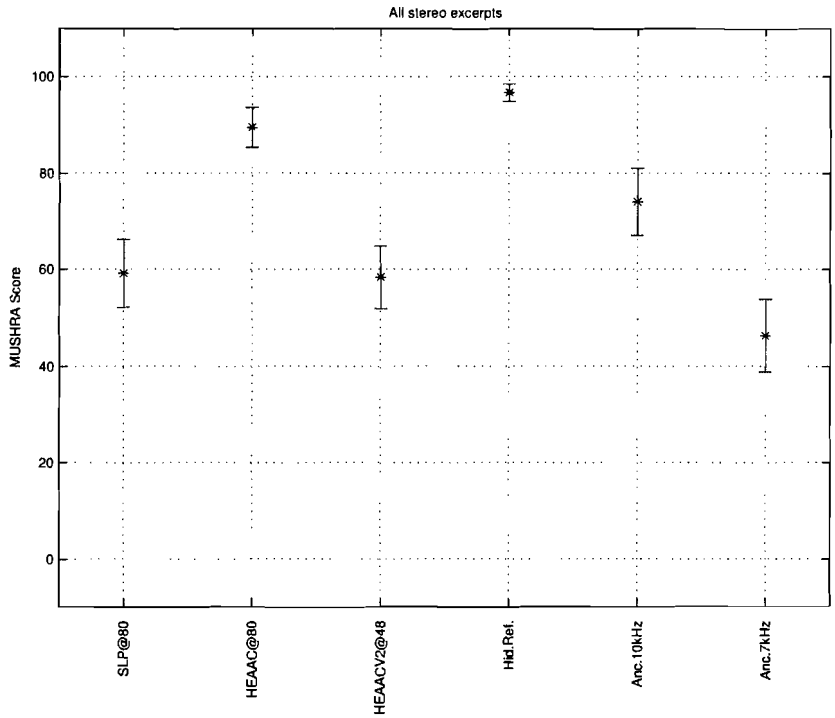

Figure 5.2: Listening test results per coder for all stereo excerpts.

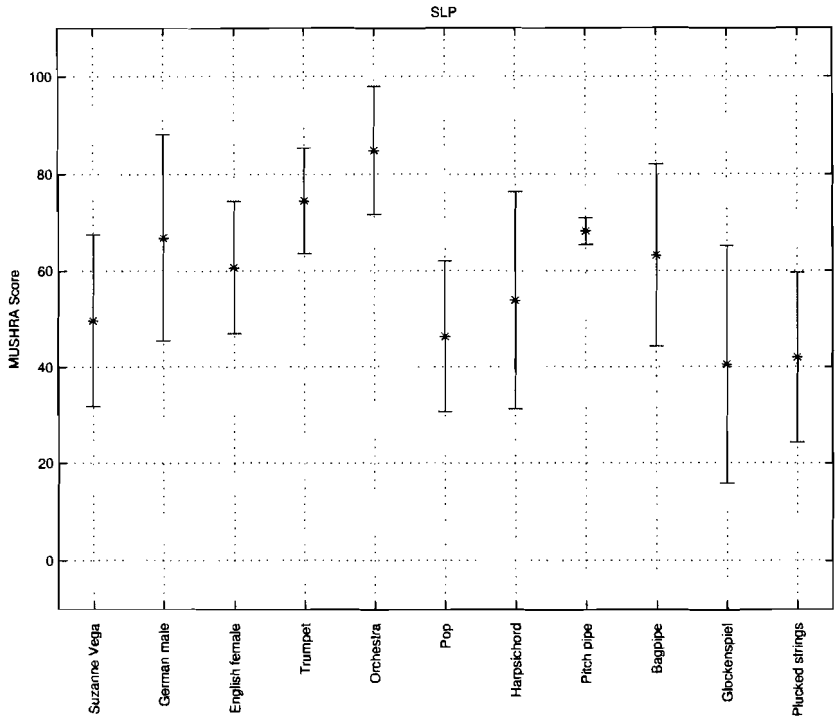

Figure 5.3: Listening test results per excerpt for SLP. The bit rate of the first three excerpts is 58 kbps, the bit rate of the other excerpts 80 kbps.

instrument, range from average to a little bit above average. Yet, the the results of Glockenspiel and Plucked strings, which are also excerpts with a single instrument, are way below average. Thus, it can be concluded that SLP@80 has problems with single instrument excerpts in general, but with Glockenspiel and Plucked strings in particular and, apart from that, that it has a lot of problems with Pop.

In addition to the listening test results shown in this section, Appendix B shows the listening test results per individual excerpt, averaged across all listeners.

5.2.3 Discussion

In the mono case, the score of SLP@58 does not come very close to the scores of either HEAAC@80, HEAACV2@48 or the hidden reference, which means that some work has to be done to improve this. It is expected that if SLP for mono or mono-like signals uses the same bit rate as HEAAC@80 and as SLP for stereo signals, which means a bit rate of 80 kbps, then its subjective quality will be a lot closer to the subjective quality of HEAAC@80 and HEAACV2@48 for mono or mono-like signals. Thus, an additional 23 kbps can be spent then on quantizing the main signal. Suggestions for this are to use a decimation factor of 1 or to use five levels to quantize the pulses in the RPE block.

We now concentrate on the stereo case, where the score of SLP@80 does not come very close to the score of HEAAC@80. This means that also for the stereo case some work has to be done to improve this. However, it is not possible now to use a higher bit rate, in contrast to SLP@58 for mono and mono-like signals. This is because the bit rate is already reasonably high. Thus, other measures have to be taken. We will now give some suggestions.

In the first place, it is known that transients are a problem currently, because a listener noticed pre-echoes in some of the SLP@80 coded files, such as Harpsichord and Glockenspiel. This could

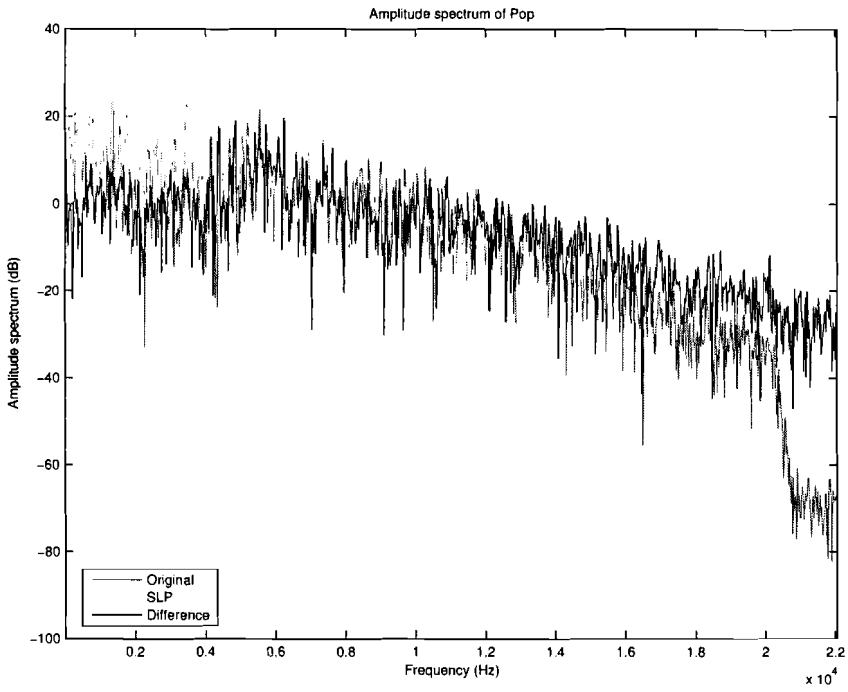

Figure 5.4: Amplitude spectrum of a windowed portion of the left channel of Pop.

be expected, because as already mentioned, a transient detection and simulation algorithm or something similar was not implemented. Thus, transients need to be addressed in SLP. The biggest improvement is expected for Harpsichord, Glockenspiel and possibly Plucked strings and the speech excerpts, but it is also possible that the other excerpts will improve as well. The total bit rate does not need to increase with this extra block, because the optimal number of extra pulses in RPE was determined before the SE block was added to the scheme. So, it is possible that the number of extra pulses can be lowered, making up to 6 kbps available.

In addition, in order to obtain insight into the other causes, we can use the results of the SLP coder per excerpt. It was already stated that Pop, Glockenspiel and Plucked strings have the biggest problems. From these, Pop immediately stands out, because SLP performs very well for the other complex excerpts with multiple instruments. Therefore, Pop should be analyzed thoroughly. This analysis should result in a few key areas which need to be improved. Preliminary investigations already showed that two key areas can be identified in the frequency domain. This is depicted in Figure 5.4, which shows the amplitude spectrum of a windowed portion of the left channel of Pop. Three lines can be seen, which are the original, the SLP coded version and the difference between the coded and the original version.

It can be seen that in the frequency region from 1 Hz to 4 kHz, the amplitude spectrum of the coded version is very close to that of the original version. Therefore, the difference is relatively small there. The first key area that is noticeable is the frequency region from 4 kHz to approximately 8 kHz, where the difference between the coded and the original version is larger than in the previous region. The difference is also larger than it was for HEAACV2@48 in the same region (4 - 8 kHz). A possible solution for this problem is to use RPE and SE to quantize the side signal up to 8.8 kHz. However, the decimation factor of RPE for the side signal still remains 5. This means that Spectral Band Replication is now used for the 4.4 – 8.8 kHz part of the side signal. The synthetic side signal is then used for frequencies greater than 8.8 kHz.

In the frequency region from approximately 8 kHz to approximately 15 kHz, the difference

between the coded and the original version is also relatively large. However, this was also the case for HEAACV2@48. The second problem can be found in the frequency region of approximately 15 kHz and higher. It can be seen that the SLP coded version has more energy in that region. Possibly, this is consistent with the noted extra quantization noise in the SLP coded excerpts. Therefore, it is suggested to filter the SLP coded version with a filter that attenuates the signal in the higher frequencies.

5.3 Other Attributes

The three other attributes, which are the coding delay, computational complexity and scalability, are now considered, starting with the coding delay.

Several applications that use digital transmission of audio signals, such as in-ear monitoring for musicians or wireless digital transmission to loudspeakers, require minimal delay. However, standard perceptual audio coders such as AAC or AAC Low Delay suffer from the algorithmic delay inherent to subband coding. An advantage of the Linear Prediction based audio coders over existing subband and transform coders is that with predictive coding, it is possible to obtain a very small encoding/decoding delay [69] [70], with basically no loss of compression performance [71]. Comparing the coding delays of the coders used in the listening test, the delay of HEAAC@80 is equal to 4276 samples or 97 ms at a sampling frequency of 44.1 kHz, while the delay of HEAACV2@48 is 7171 samples or 163 ms, which is even higher. The delay of the SLP coder is currently 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 Codec from Fraunhofer, which is based on one-channel Linear Prediction, achieves a delay of only 6 ms [72]. However, it is expected that a delay of 6 ms cannot be achieved by the SLP coder, because the lower limit of the coding delay will not be determined by buffering of samples in the L-PSLP block, but by buffering of samples in the SE block.

Computational complexity is an important issue, because it determines the physical size and capabilities of the digital signal processor the coder will be implemented on. A lower computational complexity usually means a cheaper processor and a lower power usage. Especially the latter is very important for portable applications such as mobile phones. Currently, the pulse optimization in RPE is the bottleneck in the SLP coder, because for every pulse offset in the main signal (of which there are two), five different pulse offsets in the side signal are tried. Then, for every offset combination, 20 different quantization boundaries are tried. These boundaries determine whether the amplitude of a pulse is either +1, 0, or -1. If the pulse optimization is simplified, 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, then we can estimate the computational complexity as follows. The complexity of the GSM RPE-LTP coder, which is a one-channel Linear Prediction based speech coder that uses a sampling frequency of 8 kHz, is 5 to 6 MIPS [73]. The SLP coder is very similar to the GSM RPE-LTP coder, the differences being that SLP is a stereo coder, that it uses a sampling frequency of 44.1 kHz and that the LTP is replaced by SE. We now assume that doubling the number of channels results in doubling the number of MIPS and doubling the sampling frequency also results in doubling the number of MIPS. This means that we have to multiply the complexity of the GSM RPE-LTP coder by a factor of 2 and then by a factor of 5.5. This results in a complexity of 60 MIPS. However, this is the complexity of a Stereo Linear Prediction based coder with an LTP, but without SE. Removing the LTP and adding SE results in an increase in complexity, but the extent of this increase is unfortunately unknown. However, it is expected that the total computational complexity of the SLP coder will be below 100 MIPS, because it is not expected that the complexity of SE will be equal to the complexity of the whole SLP coder without SE. The estimated complexity of SLP is somewhat lower than the complexities of the HE-AAC coder and the HE-AAC-V2 coder, which are both at least 100 MIPS [74].

A coder is bit rate scalable when it is capable of encoding audio material at different bit rates,

and consequently, different qualities. A coder is bit stream scalable when the end user can decide the bit rate and quality at which the audio material is decoded. This property is very attractive for applications where the audio material should offer the possibility of being accessed at different qualities/bit rates. This is often the case in music distribution or internet radio. Riera-Palou et al. [53] [54] proposed a bit stream scalable SSC-RPE audio coder, by using RPE layer mixing and RPE coding of additional subbands, when coding the Linear Prediction error signal. RPE layer mixing means that several RPE stages can be used per subband and mixing factors are used then to mix these stages. RPE coding of additional subbands means that for every higher layer, an additional subband is coded by RPE. These ideas can also be used in the SLP coder, resulting in a bit stream scalable audio coder. In addition, if enough bits are used to quantize the main and side signal, then the output of SLP is a perfect reconstruction of the input signal, indicating transparent coding.

Chapter 6

Conclusions and Recommendations

6.1 Conclusions

As a continuation of the work presented in [1], we proposed a stereo audio coder, subject to the following criteria:

- Encoder and decoder form a system allowing perfect signal reconstruction in the absence of signal quantization and thus, near perfect reconstruction at the high bit rate end;
- The encoder constructs a main and a side signal similar to those provided by OCS, because this is advantageous for low bit rate coding purposes.

The proposed stereo audio coder was called SLP and uses two rotators and Laguerre-based Pure Stereo Linear Prediction (L-PSLP). The encoder results in two spectrally flat, uncorrelated signals, called the main and side signal, and the decoder reconstructs the original input signal. This reconstruction is perfect in the absence of any signal or parameter quantization. We described how the optimal prediction coefficients and the optimal rotation angle can be calculated and we gave regularization techniques for both calculations.

We showed that SLP is psycho-acoustically inspired, because L-PSLP was implemented. This form of Stereo Linear Prediction uses Laguerre filters instead of delays in the Stereo Linear Prediction analysis and synthesis filters. L-PSLP modifies the uniform frequency resolution, common to most signal processing systems, to a non-uniform frequency resolution. This non-uniform frequency resolution approximately corresponds to a psycho-acoustically relevant scale such as a Bark or an ERB scale. We implemented L-PSLP, because it produces a frequency warping similar to that of Warped Stereo Linear Prediction, while solving the problems associated with it. These problems were discussed in Section 3.1.

Prediction coefficients are preferably not quantized directly, because the frequency responses of the analysis and synthesis filters are extremely sensitive to such quantization. Therefore, we described how the stereo prediction coefficients can be mapped to a different representation before they are quantized. This results in a smaller distortion of the analysis and synthesis filter characteristics.

Inspired by informal listening tests, we quantized the main signal and the side signal for stereo input signals, and we quantized the main signal only, thus discarding the side signal, for mono and mono-like input signals. To quantize the main signal, we used Sinusoidal Extraction (SE) and Regular Pulse Excitation (RPE). To quantize the side signal, we also used SE and RPE, but only for frequencies less than 4 kHz. We then used in the decoder a Schröder filter and a gain factor to construct a synthetic side signal from the main signal for frequencies greater than 4 kHz. The combination of SE and RPE makes SLP very similar to a linear predictive speech coder.

The performance of the SLP coder looks promising for a first version of the coder. We estimated the bit rate of SLP to be 58 kbps for mono or mono-like signals and 80 kbps for stereo signals. The estimated bit rate for stereo signals is already not very high. However, it should be possible to achieve a lower bit rate than 80 kbps, because this bit rate was estimated without using optimal quantization methods for the different parameters. This estimate was made by using safe error margins and without using advanced techniques such as vector quantization and differential coding. We showed that the subjective quality of the coded material of SLP, which was determined by performing a formal listening test, is between fair and good. This should be better, so some work has to be done to improve the subjective quality. In the mono and mono-like case, it is expected that if SLP for mono and mono-like signals uses the same bit rate as HE-AAC at 80 kbps (HEAAC@80) and as SLP for stereo signals, which means a bit rate of 80 kbps, then its subjective quality will be a lot closer to the subjective quality of HEAAC@80 and HE-AAC-V2 at 48 kbps (HEAACV2@48) for mono and mono-like signals. In the stereo case, where the subjective quality of HEAACV2@48 is equal to the subjective quality of SLP at 80 kbps (SLP@80), it is known that transients are a problem, because an experienced listener noticed pre-echoes in some of the SLP@80 coded files. Thus, the subjective quality of SLP will improve if transients are addressed, by implementing a transient detection and simulation algorithm or something similar. In addition, the results of the SLP coder per excerpt can be used to improve the subjective quality of coded stereo signals. SLP@80 has the most problems with excerpt Pop, so if Pop is analyzed thoroughly, then this analysis should result in a few other key areas which should be improved as well.

Regarding the other attributes, we showed that the coding delay of the SLP coder is equal to only 26 ms. This delay is caused by buffering of samples in the L-PSLP block. It is expected that a delay lower than 26 ms can be achieved and that the lower limit of the coding delay will not be determined by buffering of samples in the L-PSLP block, but by buffering of samples in the SE block. We also expect that the computational complexity of SLP will not be an issue if the pulse optimization in RPE is simplified. In addition, we argued that the SLP coder can be made bit stream scalable by using two techniques called RPE layer mixing and RPE coding of additional subbands.

We propose to use the SLP coder for applications that require high 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.

6.2 Recommendations

In line with the conclusions given above, we give the following recommendations for further research on the SLP coder:

- All parameters, including the prediction coefficients, the rotation angle and the SE parameters should be quantized and coded by using advanced techniques such as vector quantization and differential coding;
- The total bit rate of the SLP coder should be measured instead of estimated;
- Some aspects of the quantization scheme for the main and side signal should be reviewed. These aspects are:
  - The optimal number of extra pulses in the RPE block. This is because the optimal number that is used now was determined without the SE block in the quantization scheme;

  - The weighting filter in the RPE block, because it is not derived from a Stereo Psycho-Acoustic Model (S-PAM). However, a S-PAM does not exist currently, so a solution for this problem still has to be found. An interesting suggestion was given by Biswas et al. [50]. It was described how the analysis and synthesis filters of L-PLP can be perceptually biased, thus incorporating a weighting filter into the analysis and synthesis filters. It may be possible with this method to incorporate an approximation of a S-PAM into the synthesis filter in the RPE scheme;
  - The frequency that determines which part of the side signal is quantized by SE and RPE and which part is reconstructed by using a synthetic side signal. Currently, this frequency is 4 kHz, but analysis of the coded version of Pop showed that in the frequency region from 4 kHz to approximately 8 kHz, the difference between the coded and the original version is relatively large. Thus, as described in Section 5.2.3, it should be investigated whether changing this frequency to for example 8.8 kHz results in an improvement of the subjective quality of the coded material;

- It should be investigated whether the inclusion of a filter that attenuates the output signal in the higher frequencies improves the subjective quality of the coded material;
- A transient detection and simulation algorithm or something similar should be implemented in the coding scheme;
- The subjective quality of the coded material should be further improved by analyzing the results of the SLP coder per excerpt, particularly Pop. This analysis should result in a few other key aspects of the coding scheme which are to be improved;
- The coding delay of SLP should be reduced by lowering the number of samples that need to be buffered before the first set of prediction coefficients can be calculated. This can be achieved by introducing some sort of dynamic windowing into the coding scheme;
- The computational complexity of the whole SLP scheme, but especially of the RPE block should be reduced. The computational complexity in the RPE block 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;
- The computational complexity of the SLP coder should be measured when the scheme has been fully optimized;
- The SLP coder should be made bit stream scalable by using the two techniques proposed by Riera-Palou et al. [53] [54]: RPE layer mixing and RPE coding of additional subbands. In addition, an extra layer for mono and mono-like signals should be added that improves the subjective quality of these signals substantially.

Bibliography

[1] Selten, T.P.J.: "Stereo coding by two-channel linear prediction and rotation". Royal Philips Electronics N.V., Philips Research, Eindhoven, The Netherlands, 2004. Technical note TN-2004-00803.

[2] Painter, T. and A. Spanias: "Perceptual coding of digital audio". Proc. of the IEEE, Vol. 88 (2000), No. 4, pp. 451-515.

[3] Kleijn, W.B. and K.K. Paliwal: "Speech coding and synthesis". Elsevier Science B.V., Amsterdam, The Netherlands, 1995. ISBN 0-4448-2169-4.

[4] Johnston, J.D.: "Perceptual transform coding of wideband stereo signals". Proc. IEEE ICASSP-89, Glasgow, Scotland, 23-26 May 1989. Vol. 3, pp. 1993-1996.

[5] Johnston, J.D. and A.J. Ferreira: "Sum-difference stereo transform coding". Proc. IEEE ICASSP-92, San Francisco, USA, 23-26 March 1992. Vol. 2, pp. 569-572.

[6] van der Waal, R.G. and R.N.J. Veldhuis: "Subband coding of stereophonic digital audio signals". Proc. IEEE ICASSP-91, Toronto, Canada, 14-17 April 1991. Vol. 5, pp. 3601-3604.

[7] Herre, J. and E. Eberlein, K. Brandenburg: "Combined stereo coding". Proc. 93rd AES Conv., San Francisco, USA, October 1992. Preprint 3369.

[8] Herre, J. and K. Brandenburg, D. Lederer: "Intensity stereo coding". Proc. 96th AES Conv., Amsterdam, The Netherlands, February 1994. Preprint 3799.

[9] Blauert, J.: "Spatial hearing; The psychophysics of human sound localization". The MIT Press, Cambridge, UK, 1997. ISBN 0-262-02413-6.

[10] Faller, C. and F. Baumgarte: "Efficient representation of spatial audio using perceptual parametrization". Proc. IEEE WASPAA-01, New Paltz, USA, 21-24 October 2001. Pp. 199-202.

[11] Baumgarte, F. and C. Faller: "Binaural cue coding-part I: Psychoacoustic fundamentals and design principles". IEEE Trans. on Speech and Audio Processing, Vol. 11 (2003), No. 6, pp. 509-519.

[12] Faller, C. and F. Baumgarte: "Binaural cue coding-part II: Schemes and applications". IEEE Trans. on Speech and Audio Processing, Vol. 11 (2003), No. 6, pp. 520-531.

[13] Breebaart, J. and S. van de Par, A. Kohlrausch, E. Schuijers: "High-quality parametric spatial audio coding at low bit rates". Proc. 116th AES Conv., Berlin, Germany, May 2004. Preprint 6076.

[14] Breebaart, J. and S. van de Par, A. Kohlrausch, E. Schuijers: "Parametric coding of stereo audio". EURASIP Journal on Applied Signal Processing, Vol. 2005 (2005), No. 9, pp. 1305-1322.

[15] Makhoul, J.: "Linear prediction: A tutorial review". Proc. of the IEEE, Vol. 63 (1975), No. 4, pp. 561-580.

[16] Singhal, S.: "High quality audio coding using multipulse LPC". Proc. IEEE ICASSP-90, Albuquerque, USA, 3-6 April 1990. Vol. 2, pp. 1101-1104.

[17] Lin, X. and R.A. Salami, R. Steele: "High quality audio coding using analysis-by-synthesis technique". Proc. IEEE ICASSP-91, Toronto, Canada, 14-17 April 1991. Vol. 5, pp. 3617-3620.

[18] Härmä, A. and U.K. Laine, M. Karjalainen: "An experimental audio codec based on warped linear prediction of complex valued signals". Proc. IEEE ICASSP-97, Munich, Germany, 21-24 April 1997. Vol. 1, pp. 323-326.

[19] Ikeda, K. and T. Mori, T. Moriya, N. Iwakami, T. Kaneko: "Audio transfer system on PHS using error-protected stereo twin VQ". IEEE Trans. on Consumer Electronics, Vol. 44 (1998), No. 3, pp. 1032-1038.

[20] Fuchs, H.: "Improving joint stereo audio coding by adaptive inter-channel prediction". Proc. IEEE WASPAA-93, Mohonk, USA, 17-20 October 1993. Pp. 39-42.

[21] Mary, D. and D.T.M. Slock: "Multistage integer-to-integer multichannel prediction for scalable lossless coding". Proc. IEEE Asilomar-02, Pacific Grove, USA, 3-6 November 2002. Vol. 1, pp. 200-205.

[22] Moriya, T. and D.T. Yang, T. Liebchen: "Extended linear prediction tools for lossless audio coding". Proc. IEEE ICASSP-04, Montreal, Canada, 17-21 May 2004. Vol. 3, pp. 1008-1011.

[23] Cambridge, P. and M. Todd: "Audio data compression techniques". Proc. 94th AES Conv., Berlin, Germany, March 1993. Preprint 3584.

[24] Fuchs, H.: "Improving MPEG audio coding by backward adaptive linear stereo prediction". Proc. 99th AES Conv., New York, USA, October 1995. Preprint 4086.

[25] Liebchen, T.: "Lossless audio coding using adaptive multichannel prediction". Proc. 113th AES Conv., Los Angeles, USA, October 2002. Preprint 5680.

[26] Ghido, F.: "An asymptotically optimal predictor for stereo lossless audio compression". Proc. IEEE DCC'03, Snowbird, USA, 25-27 March 2003. P. 429.

[27] Garcia, J-L. and P. Gournay, R. Lefebvre: "Backward linear prediction for lossless coding of stereo audio". Proc. 116th AES Conv., Berlin, Germany, May 2004. Preprint 6076.

[28] Biswas, A. and T. Selten, A.C. den Brinker: "Stability of the synthesis filter in stereo linear prediction". Proc. ProRISC 2004, Veldhoven, The Netherlands, 25-26 November 2004. Pp. 230-237.

[29] Biswas, A. and A.C. den Brinker: "Stability of the stereo linear prediction schemes". Proc. ELMAR 2005, Zadar, Croatia, 8-10 June 2005. Pp. 221-224.

[30] Whittle, P.: "On the fitting of multivariate autoregressions, and the approximate canonical factorization of a spectral density matrix". Biometrika, Vol. 50 (1963), No. 1/2, pp. 129-134.

[31] Delsarte, P. and Y.V. Genin, Y.G. Kamp: "Orthogonal polynomial matrices on the unit circle". IEEE Trans. on Circuits and Systems, Vol. 25 (1978), No. 3, pp. 149-160.

[32] Delsarte, P. and Y.V. Genin: "Multichannel singular predictor polynomials". IEEE Trans. on Circuits and Systems, Vol. 35 (1988), No. 2, pp. 190-200.

[33] Press, W.H. and B.P. Flannery, S.A. Teukolsky, W.T. Vetterling: "Numerical recipes in C: The art of scientific computing". Cambridge University Press, Cambridge, UK, 1988. ISBN 0-521-43108-5.

[34] Lee, T-W.: "Independent component analysis: Theory and applications". Kluwer Academic Publishers, Dordrecht, The Netherlands, 1998. ISBN 0-7923-8261-7.

[35] Oppenheim, A.V. and D.H. Johnson, K. Steiglitz: "Computation of spectra with unequal resolution using the fast fourier transform". Proc. of the IEEE, Vol. 59 (1971), No. 2, pp. 299-301.

[36] Smith, J.O. and J.S. Abel: "Bark and ERB bilinear transforms". IEEE Trans. on Speech and Audio Processing, Vol. 7 (1999), No. 6, pp. 697-708.

[37] Strube, H.W.: "Linear prediction on a warped frequency scale". J. Acoust. Soc. Am., Vol. 68 (1980), No. 4, pp. 1071-1076.

[38] Voitishchuk, V. and A.C. den Brinker, S.J.L. van Eijndhoven: "Pure linear prediction and its application to speech and audio coding". Royal Philips Electronics N.V., Philips Research, Eindhoven, The Netherlands, 2002. Technical note 2002/062.

[39] den Brinker, A.C. and F. Riera-Palou: "Pure linear prediction". Proc. 115th AES Conv., New York, USA, October 2003. Preprint 5924.

[40] den Brinker, A.C. and V. Voitishchuk, S.J.L. van Eijndhoven: "IIR-based pure linear prediction". IEEE Trans. on Speech and Audio Processing, Vol. 12 (2004), No. 1, pp. 68-75.

[41] Laine, U.K. and M. Karjalainen, T. Altosaar: "WLP in speech and audio processing". Proc. IEEE ICASSP-94, Adelaide, Australia, April 1994. Vol. 2, pp. 349-352.

[42] Härmä, A. and U.K. Laine, M. Karjalainen: "Backward adaptive warped lattice for wideband stereo coding". Proc. EUSIPCO-98, Rhodes, Greece, September 1998. Vol. 2, pp. 729-732.

[43] Härmä, A. and M. Karjalainen, L. Savioja, V. Valimaki, U.K. Laine, J. Huopaniemi: "Frequency-warped signal processing for audio applications". J. Audio Eng. Soc., Vol. 48 (2000), No. 2, pp. 1011-1031.

[44] Härmä, A. and U.K. Laine: "A comparison of warped and conventional linear predictive coding". IEEE Trans. on Speech and Audio Processing, Vol. 9 (2001), No. 5, pp. 579-588.

[45] Härmä, A.: "Implementation of frequency-warped recursive filters". Signal Processing, Vol. 80 (2000), No. 2, pp. 543-548.

[46] den Brinker, A.C.: "Stability of linear predictive structures based on IIR filters". Proc. ProRISC 2001, Veldhoven, The Netherlands, 29-30 November 2001. Pp. 317-320.

[47] Lee, Y.W.: "Synthesis of electrical networks by means of Fourier transforms of Laguerre functions". J. Math. Phys., Vol. 11 (1932), pp. 83-113.

[48] Broome, P.W.: "Discrete orthonormal sequences". J. Association for Computing Machinery, Vol. 12 (1965), pp. 151-165.

[49] Biswas, A. and A.C. den Brinker: "Fast and efficient Laguerre based pure linear prediction". Royal Philips Electronics N.V., Philips Research, Eindhoven, The Netherlands, 2005. Technical note PR-TN-2004/01053.

[50] Biswas, A. and A.C. den Brinker: "Perceptually biased linear prediction". Proc. 121st AES Conv., San Francisco, USA, October 2006.

[51] Biswas, A. and A.C. den Brinker: "Quantization of transmission parameters in stereo linear predictive systems". Proc. IEEE DCC'06, Snowbird, USA, 28-30 March 2006. Pp. 262-271.

[52] Biswas, A. and A.C. den Brinker: "Quantization of stereo linear prediction systems". Submitted to IEEE Trans. on Audio, Speech and Language Processing.

[53] Riera-Palou, F. and A.C. den Brinker, A.J. Gerrits: "The SSC-RPE audio coder: A hybrid parametric-waveform approach to scalable audio coding". Royal Philips Electronics N.V., Philips Research, Eindhoven, The Netherlands, 2004. Technical note TN-2004/00274.

[54] Riera-Palou, F. and A.C. den Brinker, A.J. Gerrits: "A hybrid parametric-waveform approach to bit stream scalable audio coding". Proc. IEEE Asilomar-04, Asilomar, USA, 7-10 November 2004. Vol. 2, pp. 2250-2254.

[55] Audio subgroup: "Report on the verification test of MPEG-4 parametric coding for high-quality audio". ISO/IEC JTC/SC29/WG11 N6675, 2004.

[56] Peter, H.C.: "Combined sinusoidal and pulse coding for audio compression". Royal Philips Electronics N.V., Philips Research, Eindhoven, The Netherlands, 2005. Technical note TN-2005-00182.

[57] den Brinker, A.C. and A.J. Gerrits: "Sinusoidal analysis in the SSC coder". Royal Philips Electronics N.V., Philips Research, Eindhoven, The Netherlands, 2004. Technical note PR-TN-2004/00183.

[58] Oomen, A.W.J. and A.C. den Brinker: "Sinusoids plus noise modelling for audio signals". Proc. AES 17th Int. Conf., Florence, Italy, 2-5 September 1999. Pp. 226-232.

[59] Zwicker, E. and R. Feldtkeller: "Das Ohr als Nachrichtenempfänger". S. Hirzelverlag, Stuttgart, Germany, 1967.

[60] Kroon, P. and E.D.F. Deprettere, R.J. Sluyter: "Regular-pulse excitation - a novel approach to effective and efficient multipulse coding of speech". IEEE Trans. on Acoustics, Speech and Signal Processing, Vol. 34 (1986), pp. 1054-1063.

[61] Riera-Palou, F. and A.C. den Brinker, A.J. Gerrits, R.J. Sluijter: "Improved optimisation of excitation sequences in speech and audio coders". IEE Electronics Letters, Vol. 40 (2004), pp. 515-517.

[62] Riera-Palou, F. and A.C. den Brinker, A.J. Gerrits: "Modelling long-term correlations in broadband speech and audio coders". IEE Electronics Letters, Vol. 41 (2005), pp. 508-509.

[63] den Brinker, A.C. and E.G.P. Schuijers, A.W.J. Oomen: "Parametric coding for high-quality audio". Proc. 112th AES Conv., Munich, Germany, May 2002. Preprint 5554.

[64] den Brinker, A.C. and A.J. Gerrits, R.J. Sluijter: "Phase transmission in a sinusoidal audio and speech coder". Proc. 115th AES Conv., New York, USA, October 2003. Preprint 5983.

[65] Schröder, M.R.: "Synthesis of low-peak-factor signals and binary sequences with low autocorrelation". IEEE Trans. Inform. Theory, Vol. 16 (1970), No. 1, pp. 85-89.

[66] den Brinker, A.C. and A.J. Gerrits: "The noise module in the SSC audio and speech coder". Royal Philips Electronics N.V., Philips Research, Eindhoven, The Netherlands, 2003. Technical note 2003/00010.

[67] Riera-Palou, F. and A.C. den Brinker: "Pulse excitation coding for broadband audio". Proc. 4th Philips DSP Conf., Veldhoven, The Netherlands, 16-16 November 2005.

[68] Stoll, G. and F. Kozamernik: "EBU listening tests on internet audio codecs". EBU Technical Review, No. 283 (2000).

[69] Jayant, N.S. and P. Noll: "Digital coding of waveforms". Prentice-Hall, New Jersey, USA, 1984. ISBN 0-13-211913-7.

[70] Dorward, S. and D. Huang, S.A. Savari, G. Schuller, B. Yu: "Low delay perceptually lossless coding of audio signals". Proc. Data Compression Conference, Snowbird, USA, 27-29 March 2001. Pp. 312-320.

[71] Manfred, L. and G. Schuller, M. Gayer, U. Krämer, S. Wabnik: "A guideline to audio codec delay". Proc. 116th AES Conv., Berlin, Germany, 8-11 May 2004. Preprint 6062.

[72] Fraunhofer IDMT: "Audio coding with ultra low encoding/decoding delay". Http://www.idmt.fraunhofer.de/eng/press_media/download/product_information/uld_eng_-web.pdf, 2006.

[73] Spanias, A.S.: "Speech coding: A tutorial review". Proc. of the IEEE, Vol. 82 (1994), No. 10, pp. 1541-1582.

[74] Coding Technologies: "MPEG-4 aacPlus fixed-point reference implementations". Http://www.codingtechnologies.com/products/assets/productsheets/aacPlusFixedPointReferenceImplementations_v3.1.pdf, 2006.

Appendix A

List of Abbreviations

AAC	Advanced Audio Coding
AbS	Analysis-by-Synthesis
ACs	Arcsine Coefficients
Anc.7kHz	7 kHz Anchor
Anc.10kHz	10 kHz Anchor
BCC	Binaural Cue Coding
HE-AAC	High Efficiency Advanced Audio Coding
HEAAC@80	HE-AAC at 80 kbps
HEAACV2@48	HE-AAC-V2 at 48 kbps
Hid.Ref.	Hidden Reference
IC	Interchannel Coherence
ICC	Inter-Channel Correlation
ICLD	Inter-Channel Level Difference
ICTD	Inter-Channel Time Difference
IID	Interchannel Intensity Difference
ILTT	Integrated Listening Test Tool
IPD	Interchannel Phase Difference
ITU	International Telecommunications Union
IVMs	Innovation Variance Matrices
LARs	Log Area Ratios
L-PLP	Laguerre-based Pure Linear Prediction
L-PSLP	Laguerre-based Pure Stereo Linear Prediction
LSFs	Line Spectral Frequencies
LTP	Long Term Predictor
MPE	Multi Pulse Excitation
MUSHRA	Multi Stimulus test with Hidden Reference and Anchors
MUX	Multiplexer
nRMs	normalized Reflection Matrices
OCS	Optimum Coding of Stereo
OPD	Overall Phase Difference
PAM	Psycho-Acoustic Model
PCA	Principal Component Analysis
PLP	Pure Linear Prediction
RCs	Reflection Coefficients
RMs	Reflection Matrices
RPE	Regular Pulse Excitation
SBR	Spectral Band Replication
SE	Sinusoidal Extraction
SiA	Sinusoidal Analyzer

SiS	Sinusoidal Synthesizer
SLP	Stereo Linear Predictive Coding of Audio
SLP@58	SLP at 58 kbps
SLP@80	SLP at 80 kbps
S-PAM	Stereo Psycho-Acoustic Model
SS	Spectral Smoothing
SSC	Sinusoidal Coder
TU/e	Eindhoven University of Technology
WLP	Warped Linear Prediction
WSLP	Warped Stereo Linear Prediction

Appendix B

Listening Test Results per Excerpt

Chapter 5 described a listening test that was performed to have an indication of the subjective quality of the coded material of SLP with respect to the subjective quality of the coded material of AAC. This appendix shows the listening test results per individual excerpt, averaged across all listeners.

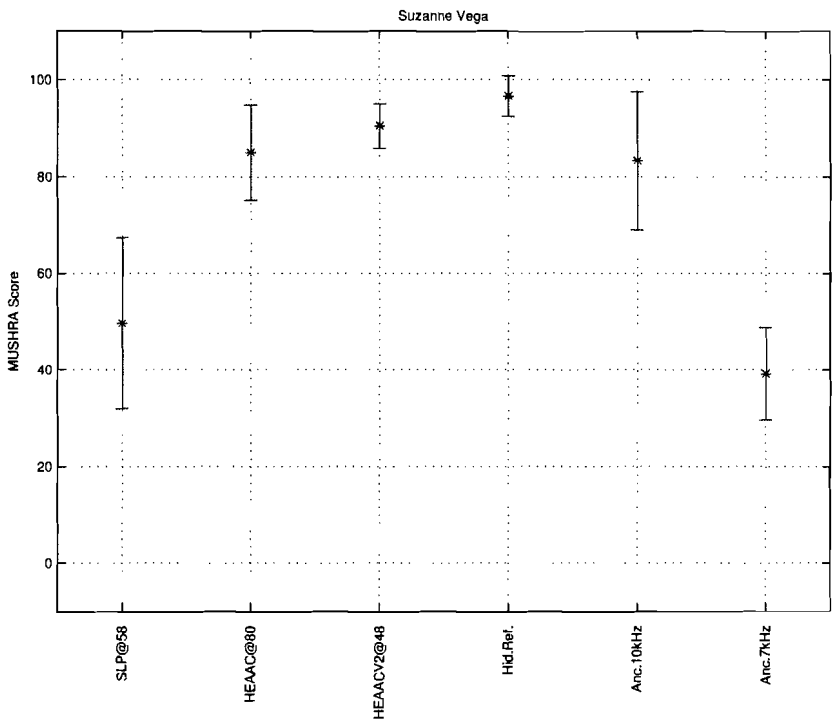

Figure B.1: Listening test results for Suzanne Vega.

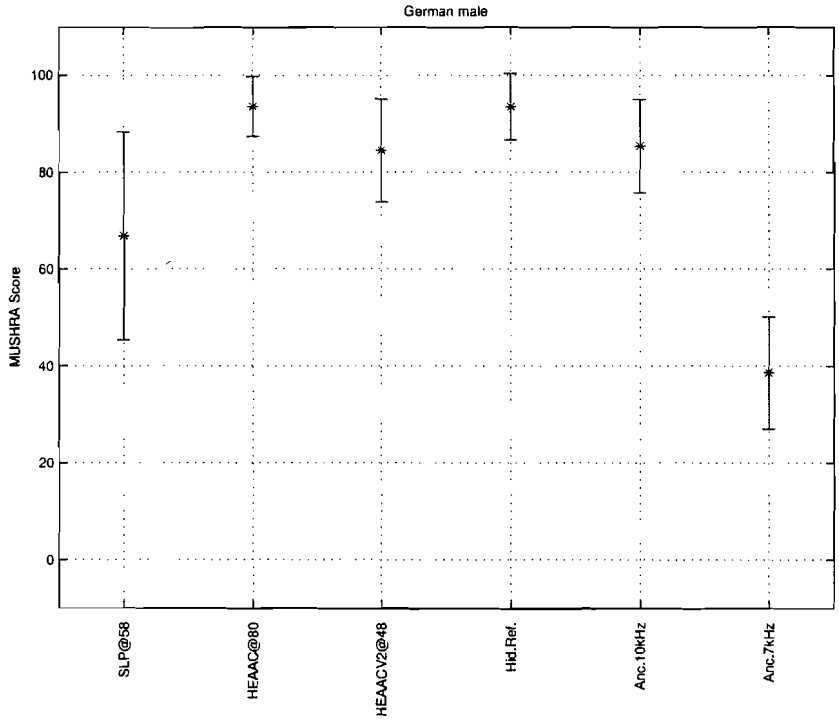

Figure B.2: Listening test results for German male.

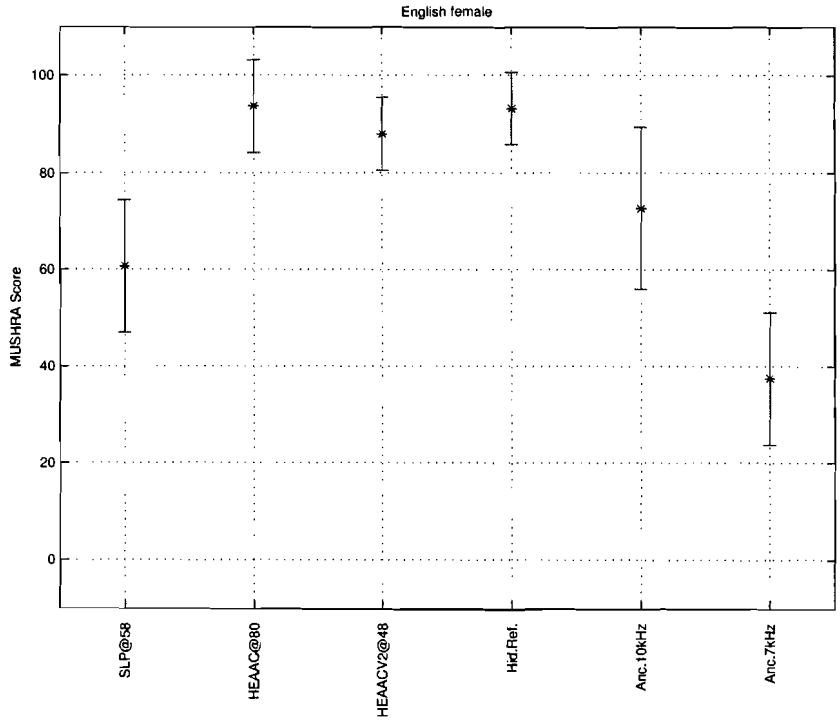

Figure B.3: Listening test results for English female.

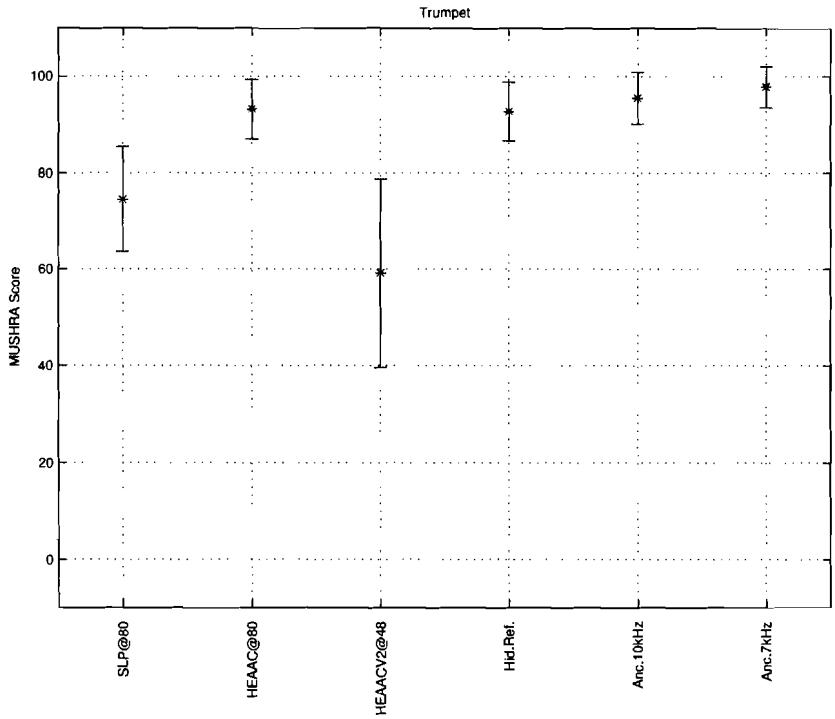

Figure B.4: Listening test results for Trumpet.

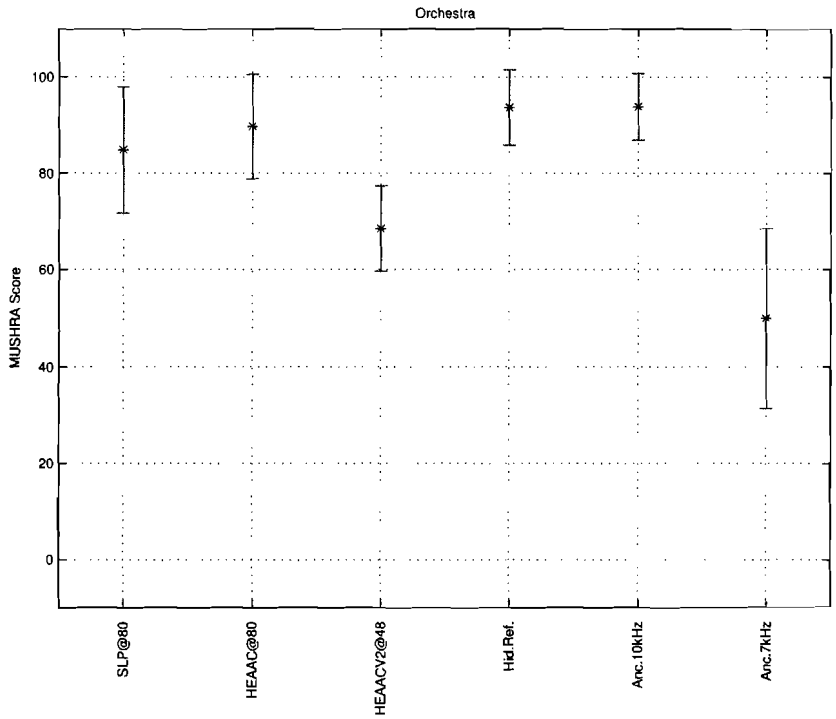

Figure B.5: Listening test results for Orchestra.

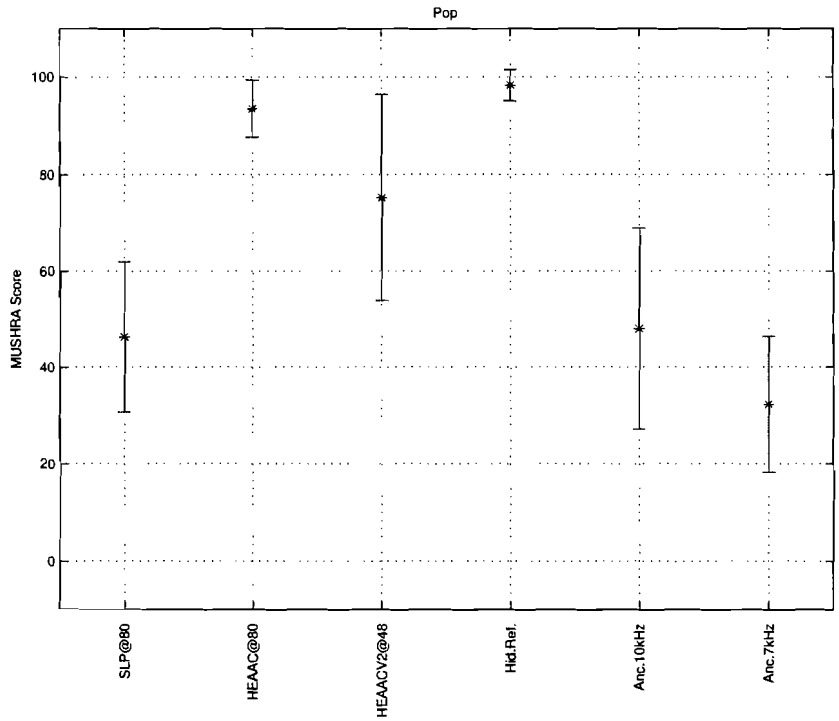

Figure B.6: Listening test results for Pop.

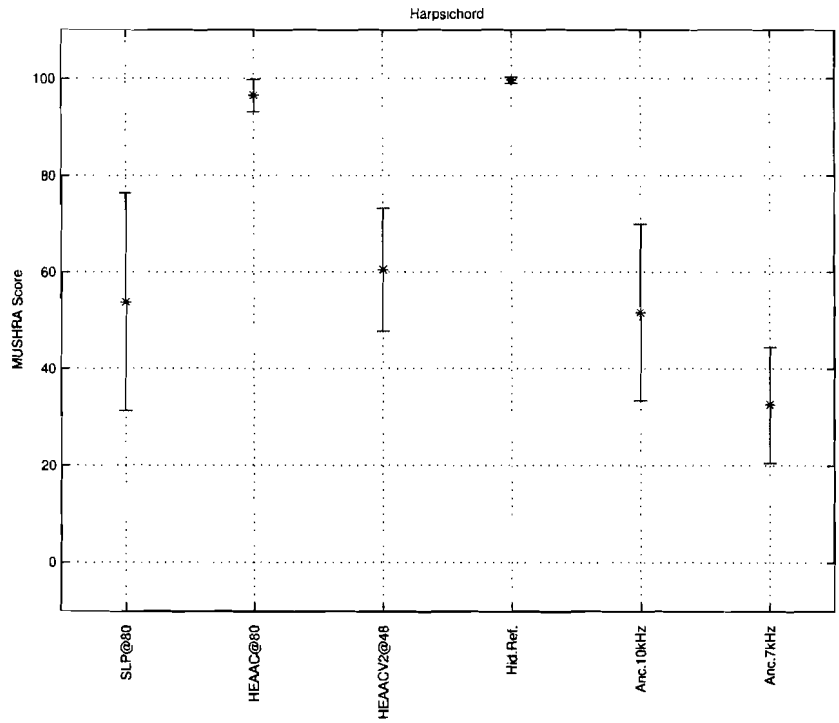

Figure B.7: Listening test results for Harpsichord.

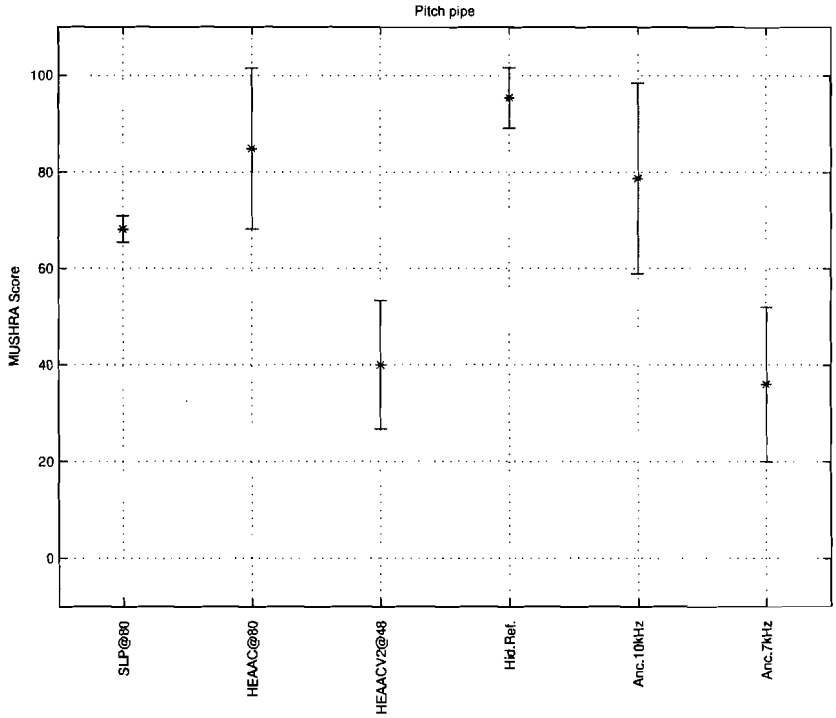

Figure B.8: Listening test results for Pitch pipe.

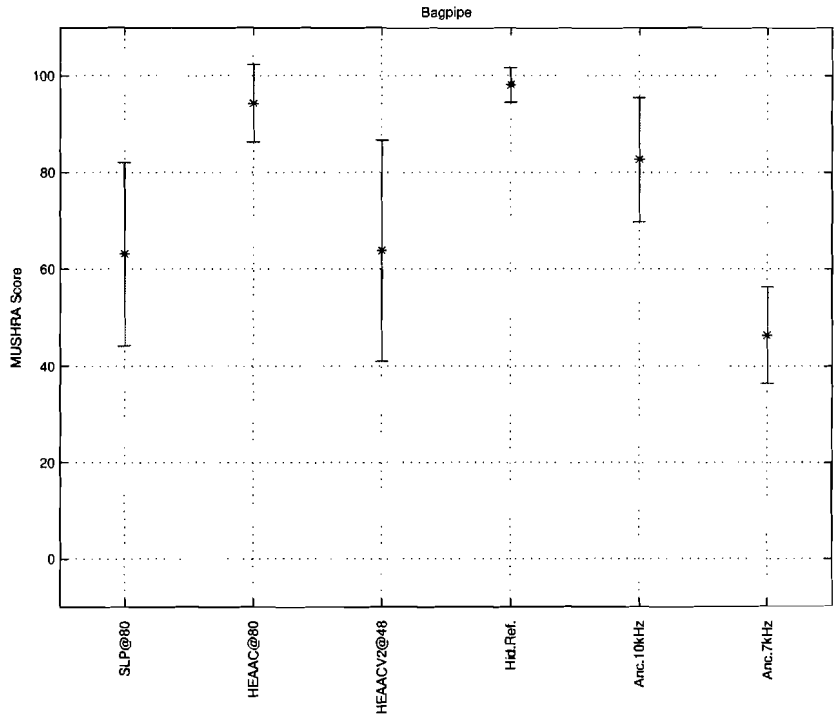

Figure B.9: Listening test results for Bagpipe.

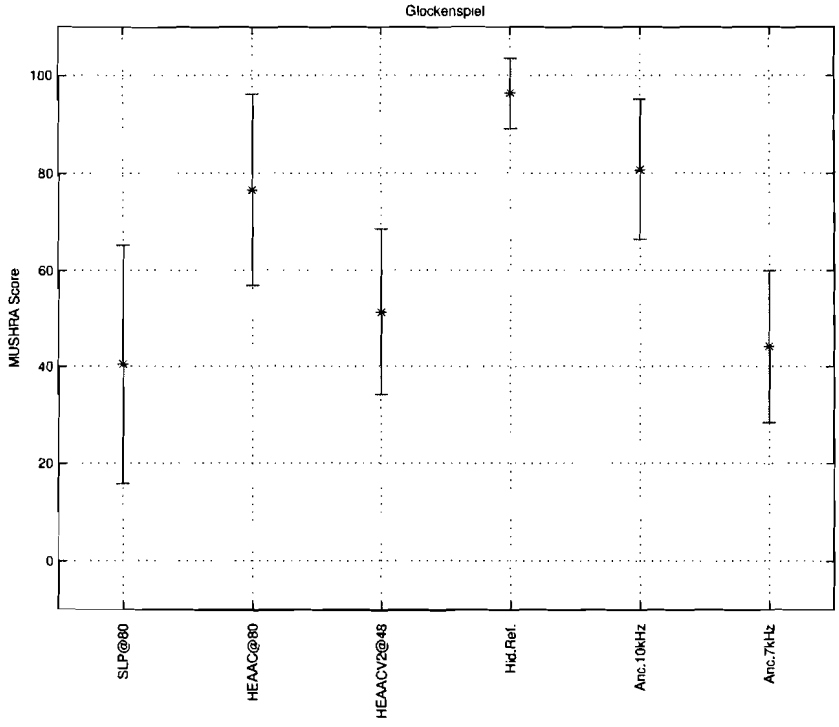

Figure B.10: Listening test results for Glockenspiel.

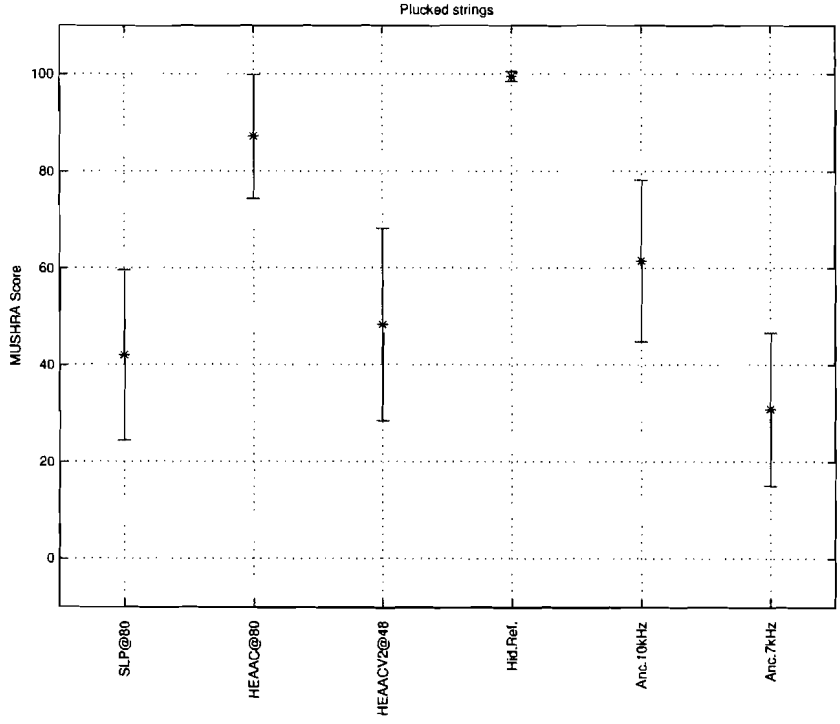

Figure B.11: Listening test results for Plucked strings.