USC–SIPI REPORT #359

Time-Frequency and Adaptive Signal Processing Methods for Immersive Audio Virtual Acquisition and Rendering

by

Athanasios N. Mouchtaris

May 2003

Signal and Image Processing Institute
UNIVERSITY OF SOUTHERN CALIFORNIA
Department of Electrical Engineering-Systems
3740 McClintock Avenue, Room 400
Los Angeles, CA 90089-2564 U.S.A.

Dedication

This dissertation is dedicated to my parents, Nikolaos and Sophia Mouchtaris.

Acknowledgements

I would like to thank my advisor, Prof. C. Kyriakakis for offering me the opportunity to come and study in such a great place as USC. I will never forget his email asking me if I am interested to work in his Lab, back in May 1997. Chris offered his support during these five years both academically, as an audio expert and enthusiast, but also financially, supporting my studies and providing all of his students with state-of-the-art technology for research. It was truly a privilege being a member of the Immersive Audio Lab and having the opportunity to contribute to its great academic reputation.

I would also like to thank the members of my Dissertation and Guidance committees. Prof. S. Narayanan has been extremely resourceful and profoundly influenced my research methods. I am very grateful for his advice. I consider myself privileged for the opportunity I was offered to interact with Prof. R. Leahy as a student of three of his courses; he is a great educator and scientist. I would also like to thank Prof. K. Jenkins and Prof. C. Papadopoulos for their eagerness to participate in my committees.

During my studies, I had the honor to interact with great educators and researchers, many of them world-renowned for their scientific contributions. I was especially fortunate to have received great assistance from some of them in various personal matters.

I would like to thank Dean M. Nikias, Prof. T. Holman, Prof. A. Petropoulou, and Prof. H. Boussalis for offering their unconditional help. Dean Nikias is partly responsible for me being here as the founder of the Integrated Media Systems Center, an institution that had a great impact in the lives of so many people. It was an honor to have met a person of such vision.

My gratitude goes to Diane Demetras, Jonathan Kotler and Carol Gordon, who have helped me profoundly. Also, I would like to thank Prof. A. Webber who provided great assistance by assuring that all our computers and infrastructure were operating perfectly, and answering all my questions, as well as Linda Varilla, Regina Morton, and Gloria Halfacre, for their help in various matters. All the SIPI and IMSC staff have been very helpful and professional and I thank them for that.

While at USC I met many other students, each one with a truly amazing personality. I would like to thank in particular Panayiotis Georgiou, Panagiotis Reveliotis, Panayotis Tsakalides, Yiota Poirazi, Stergios Roumeliotis, Joanna Giforos, Elias Kosmatopoulos, Andreas Dandalis, Petros Elia, and Dimitris Pantazis, in random order, not only for the numerous coffee-breaks and fun times, but mostly because I learned something from each one of them. A special thank you goes to Panayiotis Georgiou, whose LaTeX advice has been invaluable.

I believe it is also important to thank some people that I met before I came to the United States. I remained sane during times of crisis during the last five years largely by turning back to my friends in Athens and Thessaloniki. Dimitris Mavraganis, Thomas Tsinanis, Antonis Lazaridis, Yiannis Chatzigeorgiou, Petros Ploskas, Nikiforos

Ploskas, Anthi Oikonomou, Kyriakos Barbounakis, Anna Benaki, in random order, are unforgettable friends that will always be in my heart. Also, it is impossible not to mention two great educators who influenced my life to a great extent, Christos Yianibas and Yiannis Tolias.

Fillia Sarri has been my alter ego for the last six years. It has been an adventure, being far, then very far, then very close, then very far again, in absolute distance that is, but always close in every other way. The only thing that comes to mind is that "life is full of surprises, it advertises nothing".

Finally, I would like to thank my parents for their patience and support during all these years. They felt what I felt, pleasure and disappointment, during the different stages of my research. Most of all though, I am grateful to them for teaching me the devotion to the ideals of honesty and hard work, ideals that I always followed strictly. It goes without saying that this dissertation is dedicated to my parents.

Contents

List of Figures

List of Tables

Abstract

This dissertation is concerned with the enhancement of an existing audio acquisition/reproduction system by both providing more loudspeakers for rendering (virtual rendering) as well as multiple input audio channels (virtual acquisition). By providing virtual microphones and virtual loudspeakers, the methods proposed here can transform a given microphone/loudspeaker setting into a truly immersive environment.

Immersive audio virtual rendering systems can be used to render virtual sound sources in three-dimensional space around a listener. This is achieved by simulating the Head-Related Transfer Function (HRTF) amplitude and phase characteristics using digital filters. In this work we examine certain key signal processing considerations in spatial sound rendering over headphones and loudspeakers. We address the problem of crosstalk, inherent in loudspeaker rendering, and examine two methods for implementing crosstalk cancellation and loudspeaker frequency response inversion efficiently. We demonstrate that it is possible to achieve crosstalk cancellation of 30 dB using both methods. Our analysis is extended to non-symmetric listening positions and moving listeners. A method for generating the required crosstalk cancellation filters as the listener moves is developed based on Low-Rank modeling. Using the Karhunen-Loeve

expansion of the crosstalk filters we can interpolate among designed filters to synthesize new ones, for which HRTF measurements are unavailable.

Multichannel audio offers significant advantages for music reproduction that include the ability to provide better localization and envelopment, as well as reduced imaging distortion. Although there are thousands of music recordings available in monophonic or stereophonic form, only a handful have been recorded using microphone techniques that would allow subsequent multichannel rendering. Here, we propose virtual acquisition techniques that are capable of synthesizing the required – for multichannel rendering – multiple microphone signals from a smaller set of existing audio signals. These synthesized "virtual microphone" signals can be used to produce multichannel recordings that accurately capture the acoustics of the venue and the microphone setting of an arbitrary multimicrophone recording. Applications of the proposed system include remastering of existing monophonic and stereophonic recordings for multichannel rendering, as well as efficient transmission of multichannel audio over low-bandwidth networks, such as the current Internet infrastructure.

Chapter 1

An Introduction to Immersive Audio Virtual Acquisition and Rendering

1.1 Problem Statement

Immersive audio is a new area of audio signal processing, which aims at recreating or synthesizing realistic but arbitrary acoustical environments, such as conference rooms and concert halls. There are great benefits from this research for various fields of the industry including telecommunications (teleconferencing), entertainment (audio reproduction and remastering), computer (human-computer interaction, computer based teaching), biotechnology (displays for the visually-impaired) and so forth, all of which are part of the emerging area of multimedia. In the ideal case where a large number of audio channels and loudspeakers is available, audio immersion can be accomplished by using the appropriate channel mixing and rendering. An example of such a system is the 5.1 multichannel audio system or its possible successor, the 10.2 system, currently

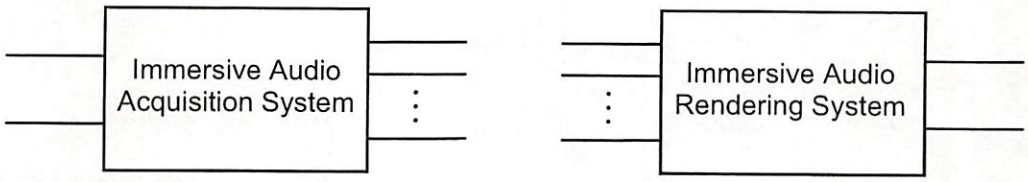

Figure 1.1: An example of immersive audio virtual acquisition and rendering. The two channels of stereophonic sound are converted into the multiple channels of a virtual multichannel recording. A multichannel recording is rendered through only two loudspeakers with the immersive experience unaffected due to the virtual rendering system.

under investigation at the Immersive Audio Laboratory of the University of Southern California.

This dissertation deals with the scenario when an *existing* audio acquisition and/or reproduction system is used for audio immersion. An example of such a system is shown in Fig. 1.1. A given stereophonic recording is enhanced with the immersive audio virtual acquisition system in order to be rendered through a multichannel rendering system. Immersive audio virtual rendering can be utilized for rendering a multichannel audio recording through an existing stereophonic reproduction system. However, the methods described in this work are derived with the more general scenario in mind, when many channels and many loudspeakers may be available but a more realistic recreation of a specific venue is desired. Thus, this work is concerned with two separate procedures, each one with its own challenges and applications:

- Immersive audio virtual acquisition. In many cases the multiple audio channels required for multichannel audio rendering do not exist and must be, in some way, synthesized. More generally, in order to recreate a truly immersive environment

with a given recording, it would be useful to be able to synthesize more channels than those available; this would allow for a more realistic reproduction of the sound. This task can be thought of as recording the sound using "virtual microphones" that synthesize these channels, while preserving the acoustics of the space.

- Immersive audio virtual rendering, which can be defined as the problem of accurately rendering arbitrary sound sources in space using two or more loudspeakers. In this work we concentrate on this problem for the case when two loudspeakers are used, but the methods derived can be easily extended for the general case of an arbitrary number of loudspeakers. Immersive audio virtual rendering can be thought of as rendering the sound through "virtual loudspeakers".

1.2 Dissertation Contribution

In the previous paragraph, a brief statement of the goals of this dissertation was given. A distinction must be made between the long-term system design goals, that are being envisioned and were the motivation behind this work, and the specific contributions of this dissertation towards achieving these goals.

Immersive audio virtual acquisition is envisioned to have the capacity of recreating an immersive multichannel acoustical experience from existing recordings of music made with a small number (as small as one) of microphones. However, this is a very general problem statement; multichannel recordings are being created with a variety of different objectives in mind. For example, a classical music symphony in a large concert hall can

be recorded with many microphones, that aim to capture different parts of the orchestra and/or the reverberation in positions far from the orchestra. Depending on the parts of the orchestra captured, different acoustical results can be obtained. A virtual acquisition system is expected to be capable of recreating multiple channels from a given recording, which would suit the objective of a multichannel microphone setting. For the example described, the objective would be to enhance these certain parts of the orchestra that are considered important. Although deriving a general method for achieving this result is desired, our investigation on this subject suggests that different types of instruments (parts of the orchestra) must be treated in different context, depending on their nature. This dissertation concentrates on a number of different instrument types (as well as on a general solution for the virtual microphones dominated by reverberation). The objective is to provide a specific solution that will, additionally, furnish a framework for further research, in an unexplored subject such as this one.

The ultimate goal of immersive audio virtual rendering research is a system that is capable of rendering virtual sound sources at arbitrary positions in space, around a listener that is possibly moving. This system is expected to render sources in real-time, possibly as a module of a larger system that aims at recreating a fully virtual experience (virtual reality). Potential applications of the immersive audio virtual rendering system are described in the next section. It is clear that there are many difficulties in designing such a system, mainly since the computational complexity is prohibitive and/or the theoretical assumptions do not closely approximate the underlying physical system. This dissertation aims at providing answers at questions regarding the design of

such systems, based on the simplifying (but generalizable) scenario of a two-loudspeaker setting. Specifically, a solution for a non-moving listener, seated symmetrically between two loudspeakers, and virtual sound sources at a particular (fixed) angle with respect to the listener, is initially given. This is a system that has been realized in real-time with an arbitrary audio input, using two FIR filters and a relatively low-cost hardware implementation. An extension is also provided that covers the general case of a moving listener and varying sound sources. This system assumes that accurate real-time tracking of the listener position is available. A problem with the design of this system is its high dependence on specific room acoustics. The performance of a system designed for a specific environment degrades if the acoustic conditions (the room environment) change. This is an inherent problem of the HRTF approach, followed in this work as well as in the majority of the research on this subject.

1.3 Potential Applications of Immersive Audio Virtual Acquisition and Rendering

Fig. 1.1 by no means implies that the systems for virtual acquisition and rendering must coexist. It rather suggests that the two systems are of complementary nature. Each of these systems has its own applications and usefulness, as described next.

1.3.1 Immersive Audio Virtual Acquisition

Immersive audio virtual acquisition is capable of synthesizing the multiple channels of a virtual multichannel recording from a small set of reference channels available in an

existing recording. This is an extremely important application for audio reproduction, given the sense of realism offered by multichannel audio, and would apparently affect many different sectors of technology, from entertainment to consumer electronics and so forth.

As a direct consequence of the immersive audio virtual acquisition system, a system that *resynthesizes* the multiple channels of an existing multichannel recording from only one or two reference channels of this recording can be created, as it will be shown in later chapters of this work. This is of great importance for applications in which bandwidth limitations prohibit transmission of multiple audio channels. In such a case, an alternative would be to transmit only the reference channels and reconstruct the remaining channels at the receiving end.

1.3.2 Immersive Audio Virtual Rendering

Immersive audio virtual rendering is capable of accurate spatial reproduction of sound using two loudspeakers or more (as a direct extension). Applications go well beyond the virtual rendering of multichannel audio. Immersive audio virtual rendering is of importance to applications where either spatial accuracy is of interest or a two loudspeaker system is an insurmountable restriction. Such applications are largely dependent on sound localization relative to visual images and include immersive telepresence; enhanced human-computer interaction; augmented and virtual reality for manufacturing and entertainment; air traffic control, pilot warning, and guidance systems; displays for the visually- or aurally-impaired; home entertainment; and distance learning.

1.4 Dissertation Organization and Overview

1.4.1 Chapter 2

In Chapter 2, the immersive audio virtual rendering [1] problem is examined. Under the assumption of symmetry of the listener with respect to the two loudspeakers used, a solution to this problem is given based on adaptive signal processing algorithms. More specifically, the Least-Mean Squared (LMS, [43]) algorithm is employed and compared to an efficient Least-Squares based algorithm (FTF, [18]) with regard to the required number of iterations and their computational complexity. The problem can be divided in two parts, namely channel equalization and crosstalk cancellation, both important for other applications of audio engineering as well [76, 75]. Although the solution is not given separately for each problem, the methods proposed can be readily used for these two specific problems. This particular problem has been a subject of ongoing investigation [34, 35, 36, 89, 29, 28, 3, 26]. The solution proposed here has the advantage of being easily extendable to the non-symmetric scenario (with the details presented in Chapter 3). The headphone rendering problem is also examined.

1.4.2 Chapter 3

In Chapter 3, the assumption of symmetry in Chapter 2 is relaxed and a generalization of the methods proposed in Chapter 2 is given. This chapter offers an approach to the

[1] The terms immersive audio virtual rendering, immersive audio rendering, spatial audio rendering and virtual loudspeaker rendering are used in the remaining chapters of this dissertation interchangeably, having the meaning of rendering virtual sound sources in space, possibly for multichannel audio applications (but for other applications as well, as described in the previous section), through a pair of loudspeakers (or headphones when stated).

general problem of immersive audio rendering with two loudspeakers when the listener is moving that is very suitable for real-time applications, provided that a mechanism for tracking the listener's position exists [39].

1.4.3 Chapter 4

Chapter 4 serves as an introduction to the remaining part of this work. In this chapter, a brief review of time-frequency signal analysis and synthesis is given. The definition here of time-frequency signal analysis/synthesis covers all signal processing methods that modify in any way the short-term spectral properties of a signal. Thus, in this chapter a unified view is furnished for the various methods suggested in the subsequent chapters for the treatment of the short-term spectral properties of audio signals.

1.4.4 Chapter 5

The "virtual microphones" problem is decomposed into a two step procedure. The first step is multichannel audio resynthesis, which is examined in this chapter. This is the problem of reconstructing an arbitrary channel of an existing multichannel recording from any other channel [77, 74]. Here, both recordings are available but the need is to resynthesize one from the other, the possible application being multichannel audio transmission under bandwidth restrictions in the communication channel (as in the Internet for example). In other words, one or two channels are to be transmitted and the remaining channels to be reconstructed at the receiving end. This is a unique approach to the multichannel audio transmission problem which additionally, as explained in the

next chapter, forms the basis of a solution to the problem of synthesizing a multichannel recording from an existing stereophonic (or even monophonic) recording.

1.4.5 Chapter 6

The second step towards achieving a solution for the "virtual microphones" problem is to use the methods described in Chapter 5 in order to synthesize a non-existing recording. In this case, only one or two reference channels are available and the task is to synthesize all the remaining channels of a *virtual* multichannel recording. A problem of system performance criteria clearly arises in this cases and is also treated at this chapter [73, 59]. This is a novel application in the audio signal processing field. It is anticipated that the time-frequency framework analyzed in Chapter 4 and applied in Chapters 5 and 6 can serve as a starting point for further research on the subject.

1.4.6 Chapter 7

Possible directions for future work on the subject of immersive audio virtual acquisition and rendering are proposed.

Chapter 2

Adaptive Signal Processing Methods for Immersive Audio Rendering

2.1 Overview

Immersive audio systems can be used to render virtual sound sources in three-dimensional space around a listener. This is achieved by simulating the head-related transfer function (HRTF) magnitude and phase characteristics using digital filters. In this chapter we examine certain key signal processing considerations in spatial sound rendering over headphones and loudspeakers. We address the problem of crosstalk, inherent in loudspeaker rendering, and examine two methods for implementing crosstalk cancellation and loudspeaker frequency response inversion efficiently. We demonstrate that it is possible to achieve crosstalk cancellation of 30 dB using both methods, but one of the two (the Fast RLS Transversal Filter method) offers a significant advantage in terms of

computational efficiency. Our analysis is easily extendable to non-symmetric listening positions and moving listeners, as it will be shown in Chapter 3.

2.2 Introduction

Sound perception is based on a multiplicity of cues that include level and time differences, and direction-dependent frequency response effects caused by sound reflection in the outer ear cumulatively referred to as the head-related transfer function (HRTF). The outer ear can be modeled as a linear time-invariant system that is fully characterized by the HRTF in the frequency domain [8].

Using immersive audio techniques it is possible to render virtual sound sources in three-dimensional space using a set of loudspeakers or headphones (for a review see [56]). The goal of such systems is to reproduce the same sound pressure level at the listener's eardrums that would be present if a real sound source was placed in the location of the virtual sound source. In order to achieve this, the key characteristics of human sound localization that are based on the spectral information introduced by the head-related transfer function must be considered [71, 78, 105, 14].

The spectral information provided by the HRTF can be used to implement a set of filters that alter non-directional (monaural) sound in the same way as the real HRTF. Early attempts in this area were based on analytic calculation of the attenuation and delay caused to the soundfield by the head, assuming a simplified spherical model of the head [25, 13]. More recent methods are based on the measurement of individual or averaged HRTF's for each desired virtual sound source direction [105, 5, 108]. In

our implementation we use a pair of HRTF's (one for each ear) that are measured for each desired virtual source direction using a microphone placed in each ear canal of a mannequin (KEMAR). The main advantage of measured HRTF's compared to analytical models is that this method accounts for the pinnae, diffraction from the irregular surface of the human head, and reflections from the upper body.

Several practical problems that arise when attempting to implement digital HRTF filters for immersive audio rendering using headphones or loudspeakers are examined here. In the case of headphone rendering, undesired frequency-dependent distortion is introduced to the binaural signal that is due to anomalies in the headphone frequency response. The inverse filter methods that we present in this chapter can be used to remove these frequency response distortions from the headphones.

When rendering immersive audio using loudspeakers, direction dependent spectral information is introduced to the input signal due to the fact that the sound is generated from a specific direction (the direction of the loudspeakers). In addition, just as in the headphones case, the loudspeakers generally do not have an ideal flat frequency response and therefore must be compensated to reduce frequency response distortion. A key issue in loudspeaker-based immersive audio arises from the fact that each ear receives sound from both loudspeakers resulting in undesirable acoustic crosstalk. We examine the relative advantages of two inverse filter methods for crosstalk cancellation and identify one (the Fast RLS Transversal Filtering method) that is particularly efficient in terms of computational requirements. Adaptive inverse filters for traditional stereophonic reproduction have been studied extensively by Nelson *et al.* [79]. In that

work, the authors examined the general problem of room inversion, but did not specifically address the problem of HRTF-based rendering. The work presented in this chapter is an extension into HRTF-based spatial audio rendering in which the ultimate goal is to achieve real-time filter synthesis for interactive applications.

In this work we refer to monaural sound as non-directional sound. Binaural sound represents sound that has been recorded with a dummy-head or has been generated through filtering with the appropriate HRTF's for the left and right ears.

The rest of this chapter is organized as follows. We first formulate the problem mathematically, in Section 2.3, for both headphone and loudspeaker rendering. The monaural and binaural input cases are treated separately. In Section 2.4 we propose two methods that can be used to address the filter inversion problems that arise due to the non-minimum phase characteristics of the transfer functions involved. Finally, in Section 2.5 we examine the performance of these two methods by comparing the generated HRTF's to the original measured HRTF's as well as the achieved level of crosstalk cancellation.

2.3 Problem Specification: Headphone and Loudspeaker Rendering

Binaural methods attempt to accurately reproduce at each eardrum of the listener the sound pressure generated by a set of sources and their interactions with the acoustic environment [9, 40, 70, 107]. Binaural recordings can be made with specially-designed

probe microphones that are inserted in the listener's ear canal, or by using a dummy-head microphone system that is based on average human characteristics. Sound recorded using binaural methods is then reproduced through headphones that deliver the desired sound to each ear. Alternatively, a monaural sound source can be filtered with the HRTF's for a particular azimuth and elevation angle in order to generate binaural sound. It was concluded from early experiments that in order to achieve the desired degree of realism using binaural methods, the required frequency response accuracy of the transfer function was ± 1 dB [29].

When headphones are used for immersive audio rendering, their frequency response is included in the frequency response of the signal that reaches the eardrums. Ideally, a filter that inverts the frequency response of the headphones is required so that the monaural signal will be processed not only with the HRTF's of the virtual source, but also with this filter. In the frequency domain, if H_p is the frequency response of the headphones and H_L the HRTF for a specific direction and the left ear (the equations for the right ear and channel are analogous), the inversion of the headphones' response can be accomplished in two ways, depending on whether the input to the designed filter is monaural or binaural sound. In the monaural input case we design the inverse filter $H_{inv} = H_L/H_p$. The monaural signal (S) is processed by this filter and then by the headphones' transfer function, so the response E_L at the left eardrum will be

$$E_L = H_p H_{inv} S = H_p \frac{H_L}{H_p} S = H_L S \tag{2.1}$$

which is exactly the desired response (S is the monaural signal to be spatialized).

Alternatively, a filter can be designed whose input is the left binaural signal S_L that already contains the required HRTF information (i.e. $S_L = H_L S$). In this case, it is simply necessary to invert the response of the headphones and so the response of the designed filter should be

$$H_{inv} = \frac{1}{H_p} \tag{2.2}$$

Then, the signal at the left eardrum E_L will be

$$E_L = H_p H_{inv} S_L = H_p \frac{1}{H_p} S_L = H_L S \tag{2.3}$$

A number of methods exist for implementing the filter H_{inv}. We will discuss two of these in a later section of this chapter.

Loudspeakers can also be used to render binaural or HRTF-processed monaural sound. In order, however, to deliver the appropriate binaural sound field to each ear it is necessary to eliminate the crosstalk that is inherent in all loudspeaker-based systems. This limitation arises from the fact that while each loudspeaker sends the desired sound to the same-side (ipsilateral) ear, it also sends undesired sound to the opposite-side (contralateral) ear.

Crosstalk cancellation can be achieved by eliminating the terms H_{RL} and H_{LR} (Fig. 2.1), so that each loudspeaker is perceived to produce sound only for the corresponding ipsilateral ear. Note that the ipsilateral terms (H_{LL}, H_{RR}) and the contralateral terms (H_{RL}, H_{LR}) are just the HRTF's associated with the position of the two loudspeakers with respect to a specified position of the listener's ears. This implies

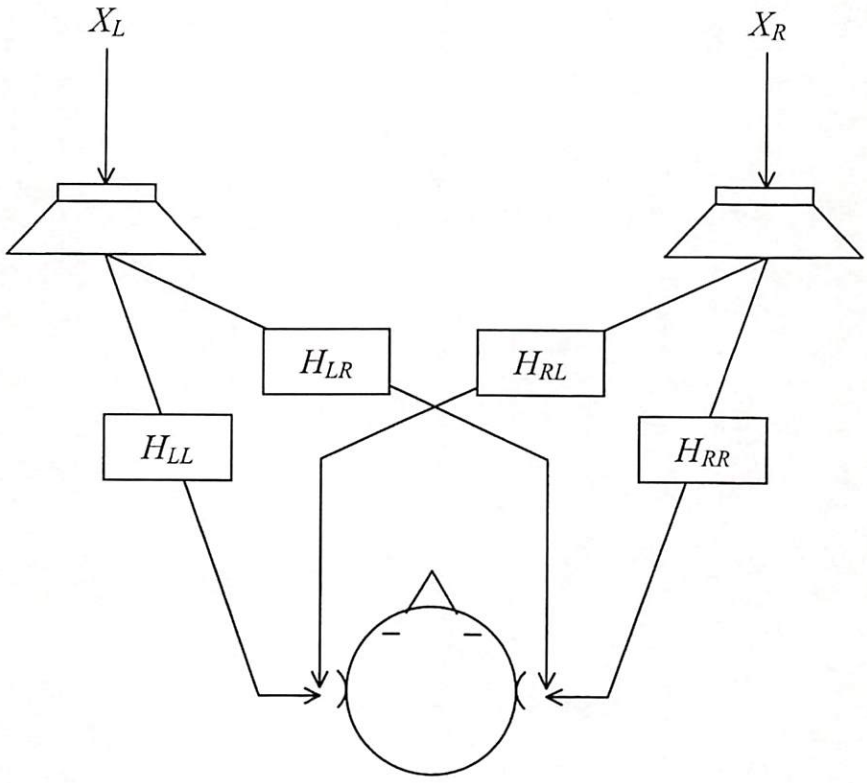

Figure 2.1: Two loudspeaker-based spatial audio rendering system showing the ipsilateral (H_{LL} and H_{RR}) and contralateral (H_{LR} and H_{RL}) terms.

that if the position of the listener changes then these terms must also change so as to correspond to the HRTF's for the new listener position. One of the key limitations of crosstalk cancellation systems arises from the fact that any listener movement that exceeds 75 to 100 mm completely destroys the desired spatial effect. This limitation can be overcome by tracking of the listener's head in three-dimensional space. A prototype system that used a magnetic tracker and adjusted the HRTF filters based on the location of the listener was demonstrated by Gardner [35, 36]. A camera-based system that

does not require that the user to be tethered has been demonstrated for stereophonic reproduction [58, 57].

Several schemes have been proposed to address crosstalk cancellation. The first such scheme was proposed by Atal and Schroeder [89] and later another was published by Damaske and Mellert [29, 28]. A method proposed by Cooper and Bauck modeled the head as a sphere and then calculated the ipsilateral and contralateral terms [3, 26]. They showed that under the assumption of left-right symmetry a much simpler shuffler filter can be used to implement crosstalk cancellation as well as synthesize virtual loudspeakers in arbitrary positions. Another method by Gardner approximates the effect of the head with a low-pass filter, a delay and a gain (less than 1) [34].

While these methods have the advantage of low computational cost, the spherical head approximations can introduce distortions particularly in the perceived timbre of virtual sound sources behind the listener. Furthermore, the assumption that the loudspeakers are placed symmetrically with respect to the median plane (*i.e.* , $H_{LR} = H_{RL}$ and $H_{LL} = H_{RR}$) leads to a solution that uses the diagonalized form of the matrix introduced by the physical system [3, 26]. This solution can *only* work for a non-moving listener seated symmetrically to the loudspeakers. Here, we use a different approach for the analysis that can be easily generalized to the non-symmetric case that arises when the listener is moving (a generalization considered in Chapter 3). While in our analysis we present the symmetric case to make the notation simpler and for facilitating comparison with existing work, the methods that we propose are also valid for the non-symmetric case. A video-based head-tracking algorithm has been developed in which

the listener is tracked and the filters are computed in real time in response to changes in the listener's position [56, 58, 57]. The motivation behind the methods presented in this chapter is the ability to achieve real-time performance so that the necessary filters can be calculated at each listener position.

We can use matrix notation to represent the loudspeaker-ear system as a two input-two output system in which the two outputs must be processed simultaneously. In the frequency domain we define H_i as the ipsilateral term, H_c as the contralateral term, H_L as the HRTF corresponding to a specific virtual sound source for the left ear, H_R as the HRTF corresponding to the same virtual sound source for the right ear, and S as the monaural input sound. Then, in order to accurately recreate a specific sound source in space, the signals E_L and E_R at the left and right eardrums respectively should be

$$\begin{bmatrix} E_L \\ E_R \end{bmatrix} = \begin{bmatrix} H_L & 0 \\ 0 & H_R \end{bmatrix} \begin{bmatrix} S \\ S \end{bmatrix} \tag{2.4}$$

The contralateral and ipsilateral terms from the physical system (the loudspeakers) will introduce an additional transfer matrix

$$\begin{bmatrix} E_L \\ E_R \end{bmatrix} = \begin{bmatrix} H_i & H_c \\ H_c & H_i \end{bmatrix} \begin{bmatrix} H_L & 0 \\ 0 & H_R \end{bmatrix} \begin{bmatrix} S \\ S \end{bmatrix} \tag{2.5}$$

In order to deliver the signals in (2.4), given that the physical system results in (2.5), preprocessing must be performed to the input S. In particular, the required preprocessing introduces the inverse of the matrix associated with the physical system, as shown below

$$\begin{bmatrix} E_L \\ E_R \end{bmatrix} = \begin{bmatrix} H_i & H_c \\ H_c & H_i \end{bmatrix} \begin{bmatrix} H_i & H_c \\ H_c & H_i \end{bmatrix}^{-1} \begin{bmatrix} H_L & 0 \\ 0 & H_R \end{bmatrix} \begin{bmatrix} S \\ S \end{bmatrix} \tag{2.6}$$

It can be seen that equations (2.4) and (2.6) are essentially the same. Solving (2.6) we find

$$\begin{bmatrix} E_L \\ E_R \end{bmatrix} = \begin{bmatrix} H_i & H_c \\ H_c & H_i \end{bmatrix} \frac{1}{H_i^2} \frac{1}{1 - \frac{H_c^2}{H_i^2}} \begin{bmatrix} H_i & -H_c \\ -H_c & H_i \end{bmatrix} \begin{bmatrix} H_L & 0 \\ 0 & H_R \end{bmatrix} \begin{bmatrix} S \\ S \end{bmatrix} \tag{2.7}$$

which can finally be written as

$$\begin{bmatrix} E_L \\ E_R \end{bmatrix} = \begin{bmatrix} H_i & H_c \\ H_c & H_i \end{bmatrix} \begin{bmatrix} 1 & -\frac{H_c}{H_i} \\ -\frac{H_c}{H_i} & 1 \end{bmatrix} \begin{bmatrix} \frac{H_L}{H_i} & 0 \\ 0 & \frac{H_R}{H_i} \end{bmatrix} \begin{bmatrix} S \\ S \end{bmatrix} \tag{2.8}$$

assuming that

$$\frac{1}{1 - \frac{H_c^2}{H_i^2}} \approx 1 \tag{2.9}$$

This assumption is based on the fact that the contralateral term is of substantially less power that the ipsilateral term because of the shadowing caused by the head. The validity of this assumption was examined by plotting the magnitude and phase of the term in (2.9) and comparing them with the corresponding magnitude and phase of an all-pass filter. The term in (2.9) is plotted in Fig. 2.2 for a set of measured HRTF's. It

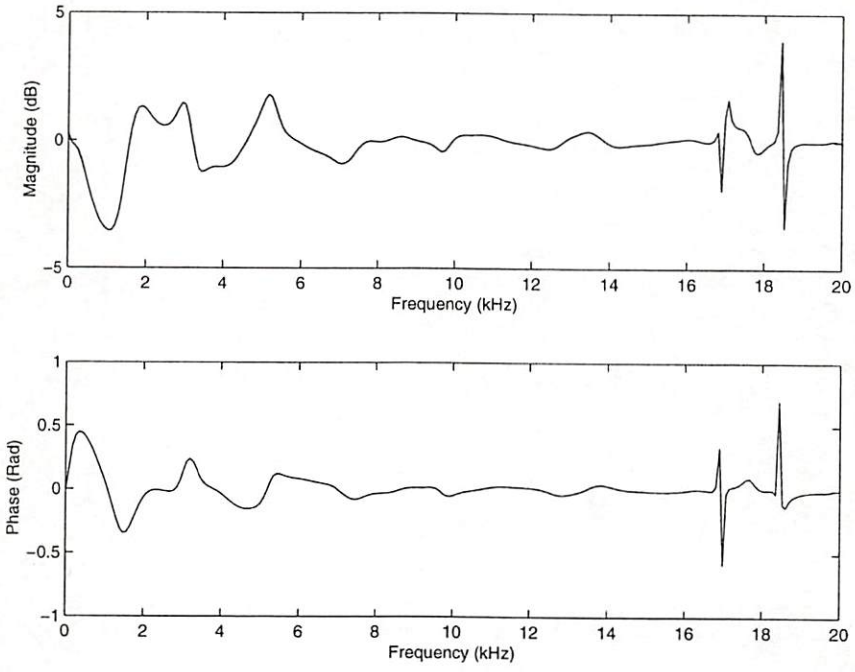

Figure 2.2: Magnitude and phase response of the term $(1-(H_c/H_i)^2)^{-1}$ that is extracted as a common factor from the matrix product that describes the physical system. The assumption that the term is approximately of all-pass response is valid.

can be seen that, indeed, this term can be considered to be of approximately all-pass response.

The terms H_L/H_i and H_R/H_i in (2.8) correspond to the loudspeaker inversion. That is, the HRTF's corresponding to the actual position of the loudspeakers are inverted since they add spectral information that is not in the binaural signal of the virtual source. The matrix

$$\begin{bmatrix} 1 & -\frac{H_c}{H_i} \\ -\frac{H_c}{H_i} & 1 \end{bmatrix}$$

corresponds to the crosstalk cancellation. In the approach described here, the crosstalk cancellation and the inversion of the loudspeakers' response are closely connected, but

it is important to note the difference between these two terms. Finally, the signals X_L and X_R that have to be delivered to the left and right loudspeaker respectively in order to render the virtual source at the desired location are given by

$$\begin{bmatrix} X_L \\ X_R \end{bmatrix} = \begin{bmatrix} \frac{H_L}{H_i} & -\frac{H_c}{H_i}\frac{H_R}{H_i} \\ -\frac{H_c}{H_i}\frac{H_L}{H_i} & \frac{H_R}{H_i} \end{bmatrix} \begin{bmatrix} S \\ S \end{bmatrix} \tag{2.10}$$

which can be written as

$$\begin{aligned} X_L &= \left(\frac{H_L}{H_i} - \frac{H_c}{H_i}\frac{H_R}{H_i} \right) S \\ X_R &= \left(\frac{H_R}{H_i} - \frac{H_c}{H_i}\frac{H_L}{H_i} \right) S \end{aligned} \tag{2.11}$$

This implies that the filters F_L and F_R for the left and right channel should be

$$\begin{aligned} F_L &= \frac{H_L}{H_i} - \frac{H_c}{H_i}\frac{H_R}{H_i} \\ F_R &= \frac{H_R}{H_i} - \frac{H_c}{H_i}\frac{H_L}{H_i} \end{aligned} \tag{2.12}$$

The monaural signal S passes through these filters and then each channel is led to the corresponding loudspeaker.

Similarly to the headphones inversion case described earlier, a filter can be designed for the case that the input are the binaural signals S_L and S_R instead of the monaural S. In this case, filtering with the pair of HRTF's H_L and H_R is not needed since the

binaural signals already contain the directional HRTF information. For the binaural case the matrix

$$\begin{bmatrix} H_L & 0 \\ 0 & H_R \end{bmatrix}$$

is substituted in (2.7) by the identity matrix.

2.4 Theoretical Analysis

The analysis in the previous sections has shown that inversion of the headphones response, crosstalk cancellation, and loudspeaker HRTF inversion, all require the implementation of preprocessing filters of the type $H_{inv} = H_x/H_y$, in which H_x is 1, H_L, H_R or H_c and H_y is the headphones response H_p or the ipsilateral response H_i. There are a number of methods for implementing the filter H_{inv}. The most direct method would be to simply divide the two filters in the frequency domain. However, H_y is in general a non-minimum phase filter, and thus the filter H_{inv} designed with this method will be unstable. A usual solution to this problem is to use cepstrum analysis in order to design a new filter with the same magnitude as H_y but being minimum phase [81]. The drawback is that information contained in the excess phase is lost.

Here, we propose a different procedure that maintains the HRTF phase information. The procedure is to find the non-causal but stable impulse response, which also corresponds to H_x/H_y assuming a different region of convergence for the transfer function, and then add a delay to make the filter causal. The trade-off and the corresponding challenge is to make the delay small enough to be imperceptible to the listener while

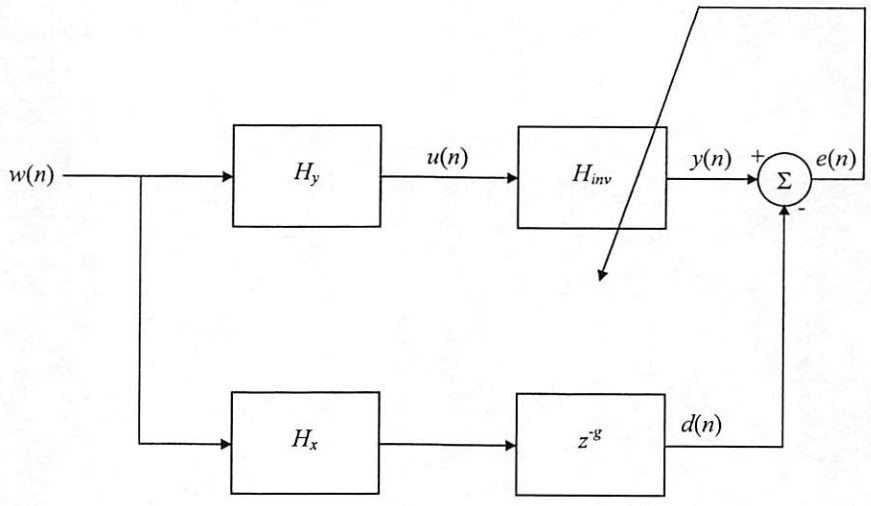

Figure 2.3: Block diagram for the algorithm used to estimate of the inverse filter. The problem of finding the filter H_{inv} such that the mean squared error between $y(n)$ and $d(n)$ is minimized, is a combination of a system identification problem (with respect to H_x) and inverse modeling problem (with respect to H_y) and its solution can be based on standard adaptive methods.

maintaining low computational cost. We describe below two methods for finding this non-causal solution.

2.4.1 Least Mean Squares (LMS) Filter Design Method

Based on the previous discussion and taking into consideration the need for adding a delay in order for the preprocessing filter to be feasible (*i.e.* causal), we conclude that the relationship between the filters H_y, H_x and the preprocessing filter H_{inv} can be depicted as in the block diagram shown in Fig. 2.3.

The problem of defining the filter H_{inv} such that the mean squared error between $y(n)$ and $d(n)$ is minimized, can be classified as a combination of a system identification

problem (with respect to H_x) and inverse modeling problem (with respect to H_y) and its solution can be based on standard adaptive methods such as the LMS algorithm [43]. More specifically, the taps of the filter H_{inv} at iteration n can be computed based on the weight adaptation formula

$$\boldsymbol{h}_{inv}(n+1) = \boldsymbol{h}_{inv}(n) + \mu \boldsymbol{u}(n) e(n) \tag{2.13}$$

in which,

$$e(n) = d(n) - \boldsymbol{h}_{inv}^{H}(n) \boldsymbol{u}(n) \tag{2.14}$$

Lowercase bold notation corresponds to time-domain filters, while filters in the frequency domain are in uppercase notation. Also, ^{H} denotes the Hermitian of a vector. The desired response $d(n)$ can be found from Fig. 2.3 to be

$$d(n) = \boldsymbol{h}_{x}^{H}(n) \boldsymbol{w}(n-g) \tag{2.15}$$

The notation $\boldsymbol{u}(n)$ denotes a vector of samples arranged as

$$\boldsymbol{u}(n) = \begin{bmatrix} u(n) & u(n-1) & \cdots & u(n-M+1) \end{bmatrix}^T \tag{2.16}$$

where, M is the order of the filter $\boldsymbol{h}_{inv}$. This is also true for $\boldsymbol{w}(n)$. The system input $\boldsymbol{w}(n)$ can be chosen arbitrarily, but a usual practice for system identification problems is to use white noise as the input. The reason is that white noise has a flat frequency response so that all frequencies are weighted equally during the adaptation procedure.

The filter length M, as well as the delay g, can be selected based on the minimization of the mean squared error. In this work, we used a variation of the LMS (the Normalized LMS) with a progressive adaptation (decrement) of the step size μ that results in faster convergence as well as smaller misadjustment. The step size μ changes at every iteration, using the update formula

$$\mu(n) = \frac{\beta}{\alpha + \|\boldsymbol{u}(n)\|^2} \tag{2.17}$$

In (2.17) β is a positive constant, usually less than 2, and α is a small positive constant [43].

The resulting filter from this method is $\boldsymbol{h}_{inv}$, which in the frequency domain is equal to H_x/H_y. If the desired output is of the form $1/H_y$, (in the binaural case), $\boldsymbol{h}_x$ can be chosen to be the impulse sequence. The result in either case is an FIR filter.

2.4.2 Least-Squares Filter Design Method

Referring again to Fig. 2.3, another way of approaching the problem is to minimize the sum of squared errors $e(n)$ (instead of the mean squared error as in the LMS method)

$$\min_{h_{inv}(m)} \sum_{n=M}^{N} \left| \sum_{m=0}^{M} u(n-m)h_{inv}(m) - d(n) \right|^2 \tag{2.18}$$

Please note that $h_{inv}(m)$ denotes the m^{th} tap of filter $\boldsymbol{h}_{inv}$, while $\boldsymbol{h}_{inv}(m)$ corresponds to the state of the filter $\boldsymbol{h}_{inv}$ at iteration m. The above equation can be rewritten in matrix notation as

$$\min_{\boldsymbol{h}_{inv}} \|\mathbf{H}\boldsymbol{h}_{inv} - \boldsymbol{h}_x\|^2 \tag{2.19}$$

in which $\mathbf{H}$ is a rectangular Toeplitz matrix that can be easily derived from (2.18). The solution to (2.19) in the Least-Squares sense is

$$\boldsymbol{h}_{inv} = \mathbf{H}^{+}\boldsymbol{h}_{x} \tag{2.20}$$

in which we denote the pseudoinverse of $\mathbf{H}$ as $\mathbf{H}^{+}$. In general, (2.19) describes an overdetermined system for which $\mathbf{H}^{+}$ in (2.20) can be written as

$$\mathbf{H}^{+} = (\mathbf{H}^{H}\mathbf{H})^{-1}\mathbf{H}^{H} \tag{2.21}$$

We denote $\mathbf{P} = \mathbf{H}^{H}\mathbf{H}$ which can be viewed as the time-averaged correlation matrix. The calculation of the pseudoinverse is a computationally demanding operation that is not suitable for real-time implementations. One way to overcome this problem is by calculating the pseudoinverse recursively. Specifically, we can calculate the inverse of $\mathbf{P}$ recursively, using the well-known matrix inversion lemma. This method is known as Recursive Least-Squares (RLS). The advantage of this method is that for most problems it requires M iterations for convergence, where M is the order of the designed filter. On the other hand, LMS usually requires a higher number of iterations for convergence. The number of iterations is a very important issue for real-time implementations, but equally important is the computational complexity of the algorithm (measured in number of multiplies and divides for adaptive systems). Here LMS has a great advantage, requiring only $3M$ operations per iteration whereas RLS requires M^2. This problem of the RLS algorithm has motivated a lot of research to find efficient implementations with reduced

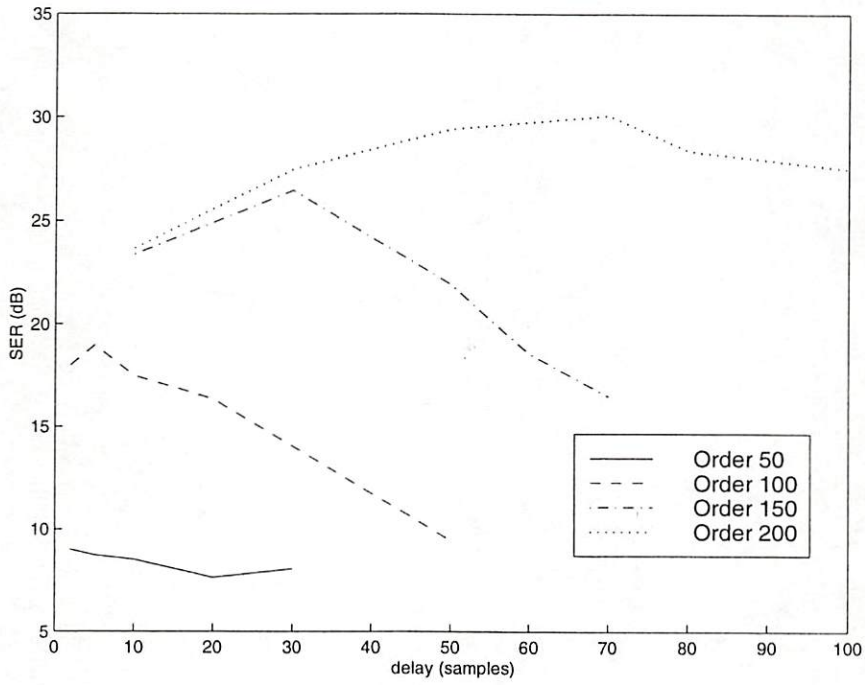

Figure 2.4: SER for several choices of delay and filter order, using the LMS method. The lowest order that gives an SER of 30 dB is 200 with a corresponding optimum delay of 70 samples.

computational complexity. Here, we implemented the FTF method for RLS proposed by Cioffi and Kailath [18]. This algorithm requires $7M$ computations per iteration while it retains the fast convergence property of the RLS algorithm, thus it is highly suitable for real-time implementations. The FTF algorithm decouples the recursive calculation of the inverse matrix of $\mathbf{P}$ into a recursive calculation of three vectors $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{C}$, which is a procedure that requires fewer computations, since no matrix multiplication is involved.

In section 2.5 we describe our findings and show that the FTF algorithm has a significant advantage over the LMS algorithm in terms of convergence rate while incurring only a moderate increase in computational complexity.

2.5 Simulation Results

2.5.1 Loudspeaker Inversion

All of the filters that are of the form H_x/H_y were designed using both the LMS and Least-Squares methods. As discussed above, a delay is introduced to the system to satisfy causality. The coefficients of these FIR filters were designed using Matlab. The delays and lengths for the filters used were optimized to achieve maximum Signal to Error power Ratio (SER) in the time domain between the filter $H_{inv}H_y$ (which we will call the *cascade* filter) and H_x. In our case, the SER is defined by

$$\frac{\sum_{k=1}^{N} h_x^2(k)}{\sum_{k=1}^{N} (h_x(k) - h_{ca}(k))^2} \tag{2.22}$$

in which h_{ca} is the impulse response of the cascade filter.

It is important to evaluate the error in the time-domain because a good approximation is required both in the magnitude and phase responses. Both methods worked successfully with a number of different measured HRTF's corresponding to 128 tap filters. The following simulation results were found using the 0° azimuth and 0° elevation measured HRTF of length 128 taps, corresponding to the term H_x. The HRTF's measurements were performed using a KEMAR dummy-head with Etymotic Research microphones. The playback system consisted of two TMH Corp. loudspeakers placed on a table so that the center of each loudspeaker was at the same height as the center of the KEMAR pinnae for on-axis measurements. The loudspeakers spacing was 50 cm and

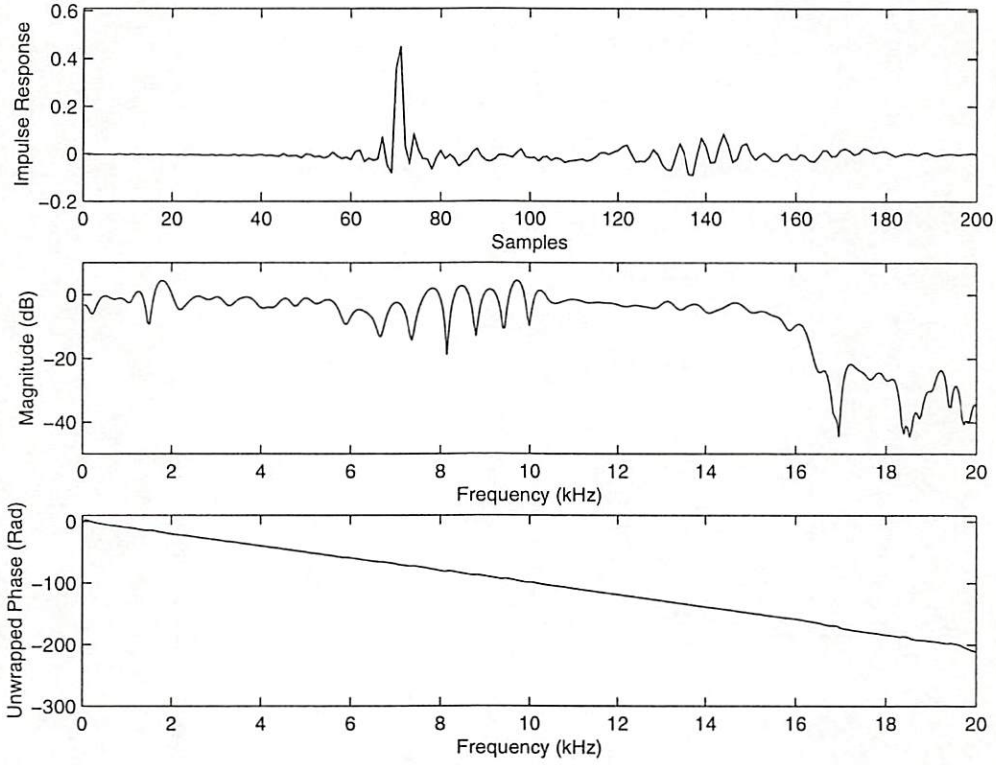

Figure 2.5: Impulse response (top), magnitude response (middle) and phase response (bottom) of the designed filter H_{inv} using the LMS method.

the center of the KEMAR's head was 50 cm from the center point of the loudspeaker baffle plane. The room in which the measurements were performed has dimensions 8.5 m (L) × 7.0 m (W) × 3.5m (H) and the reverberation time was measured using the THX R2 spectrum analyzer and found to be 0.5 seconds from 125 Hz to 4 kHz.

For the monaural input case, an inverse filter of 200 taps was designed, that introduced a delay of 70 samples (1.6 ms at a sampling rate of 44.1 kHz). These were the optimum values of filter length and delay that gave rise to an SER of better than 30 dB. The tradeoffs in SER, filter length, and delay are shown in Figs. 2.4 and 2.7 for the LMS and Least-Squares (RLS) methods respectively. It is interesting to note

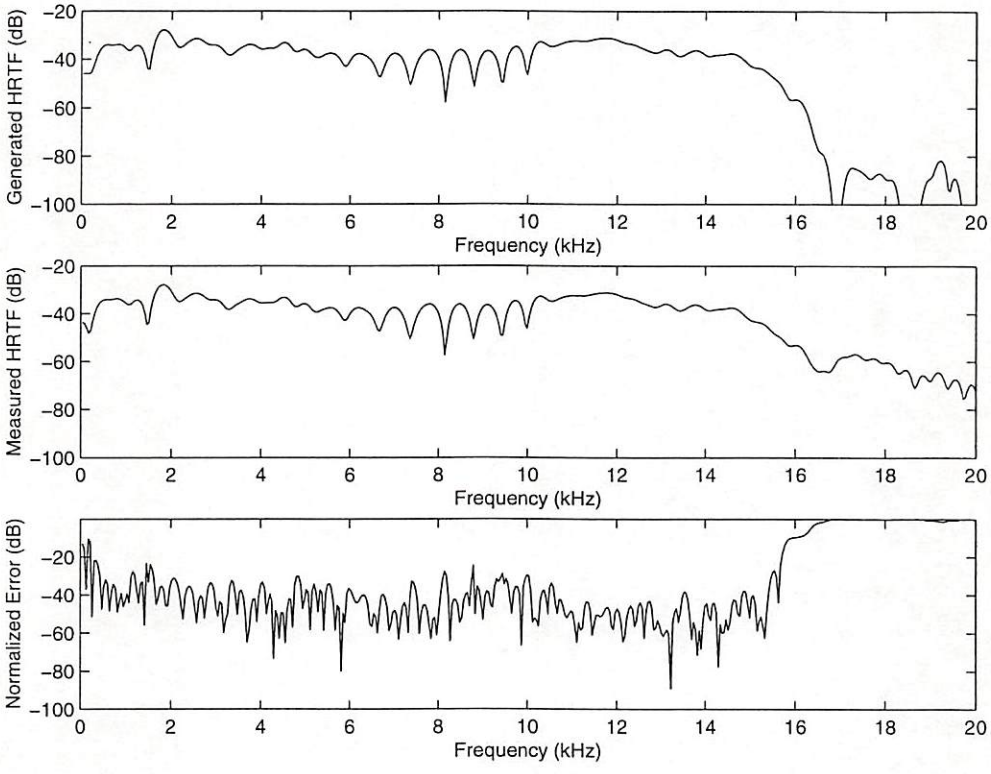

Figure 2.6: The HRTF generated from the inverse filter using the LMS method is shown in the upper plot. The measured HRTF (0° azimuth and 0° elevation) is shown in the middle and the relative error between the two is shown in the bottom plot.

that the optimal choices of filter order and delay are the same for both methods. The filter order can, of course, be chosen arbitrarily, but we found that for a given order, the corresponding delay is the same for both methods. The SER in the time domain for this case was 30.3 dB for the LMS method and 31.5 dB for the Least-Squares method. The results for the LMS method can be seen in Figs. 2.5 and 2.6. In Fig. 2.5 the resulting filter H_{inv} is plotted in both the time and frequency domains. In Fig. 2.6 a comparison is made between the magnitude of the measured HRTF and the HRTF generated using our inverse filter. Because the approximation of the two filters is made in the time

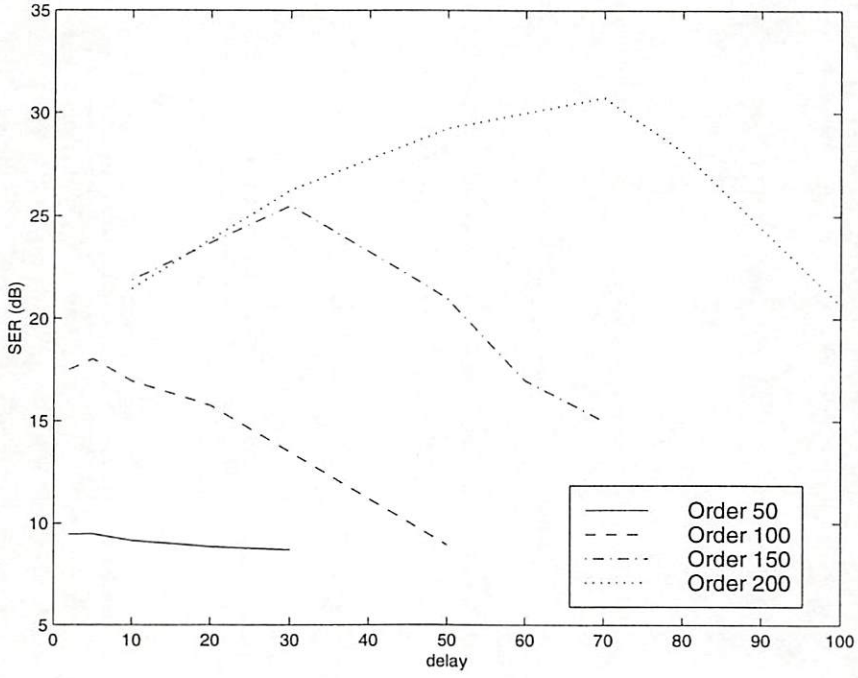

Figure 2.7: SER for several choices of delay and filter order, using the Least-Squares method. As in the LMS case, the lowest order that gives an SER of 30 dB is 200 with corresponding delay of 70 samples.

domain, it was expected that their phase responses would be practically identical. The same plots are shown in Figs. 2.8 and 2.9 for the Least-Squares case. The required number of iterations for the two algorithms is in agreement with what was mentioned in section 2.4. The LMS algorithm required 5000 iterations in order to reach the 30 dB SER criterion, while the Least-Squares method required only 500 iterations for the same error. This result, along with the relatively small increase in computational requirements of the FTF algorithm, justifies the claim that this method is highly suitable for a real-time implementation in which the filter parameters are updated in response to head-tracking information.

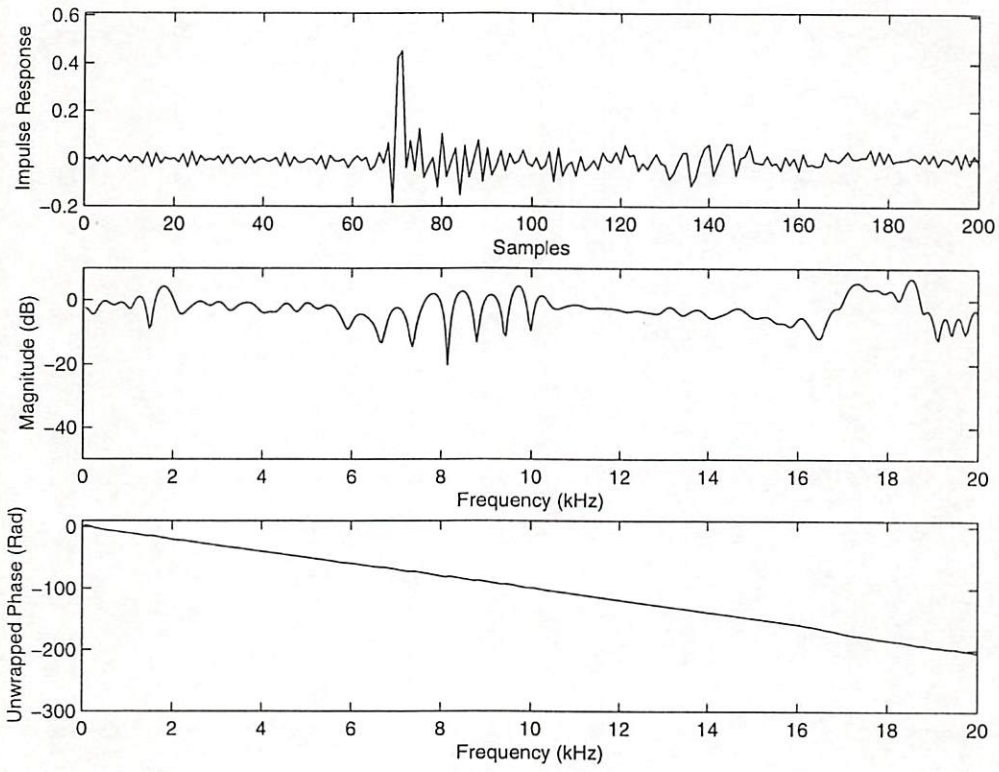

Figure 2.8: Impulse response (top), magnitude response (middle) and phase response (bottom) of the designed filter H_{inv} using the Least-Squares method.

It should be noted that for frequencies above 15 kHz, the associated wavelengths are less than 20 mm. In this range it is practically impossible to accurately place the listener's ears in the desired location for which the filters have been designed. For this reason the degradation of the normalized error above 15 kHz (as seen in Figs. 2.6 and 2.9) is acceptable since listener position errors will dominate.

If inversion of the type $1/H_i$ is required (binaural input), the cascade filter should be of exactly all-pass response. This case proved to be more demanding than the monaural input case. In order to get the desired SER of 30 dB in the time domain we had to increase the filter length to 400 taps (with a corresponding delay of 160 samples).

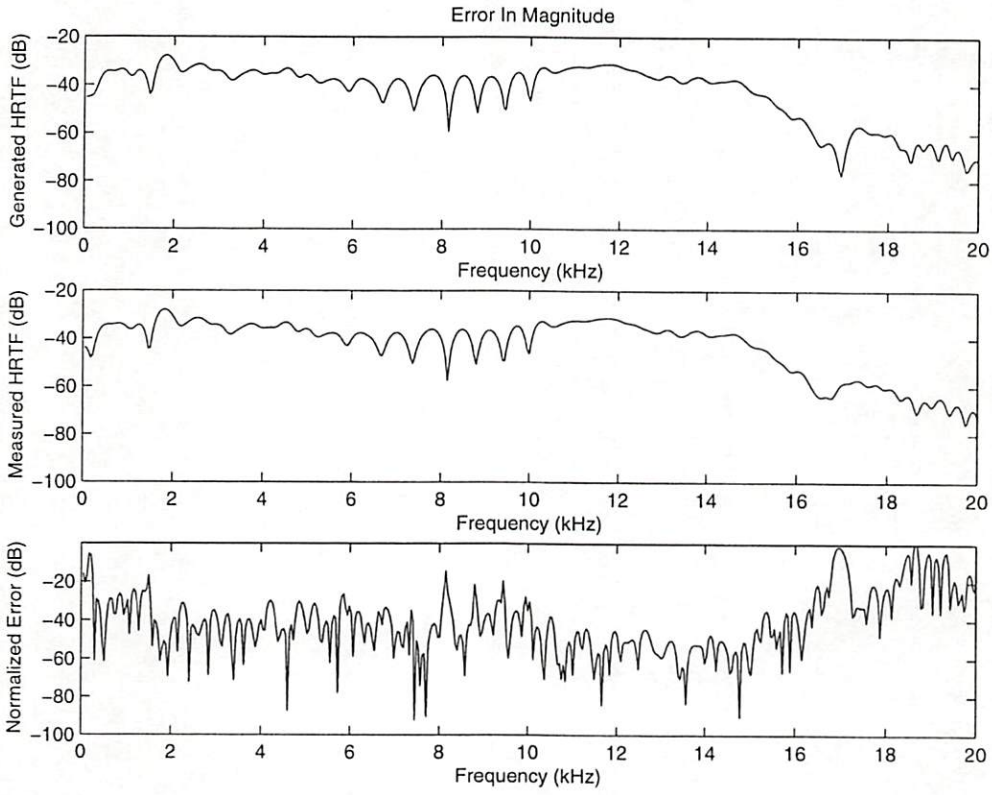

Figure 2.9: The HRTF generated from the inverse filter using the Least-Squares method is shown in the upper plot. The measured HRTF (0° azimuth and 0° elevation) is shown in the middle and the relative error in the bottom plot.

Alternatively, it is possible to design a filter of the form of H_a/H_i where H_a has an all-pass response up to 15 kHz. Using this approximation, we were able to achieve the 30 dB requirement in SER with a filter length of 200 taps and a delay of 70 samples. In listening tests there was no perceptible difference in using this method compared to the full spectrum all-pass.

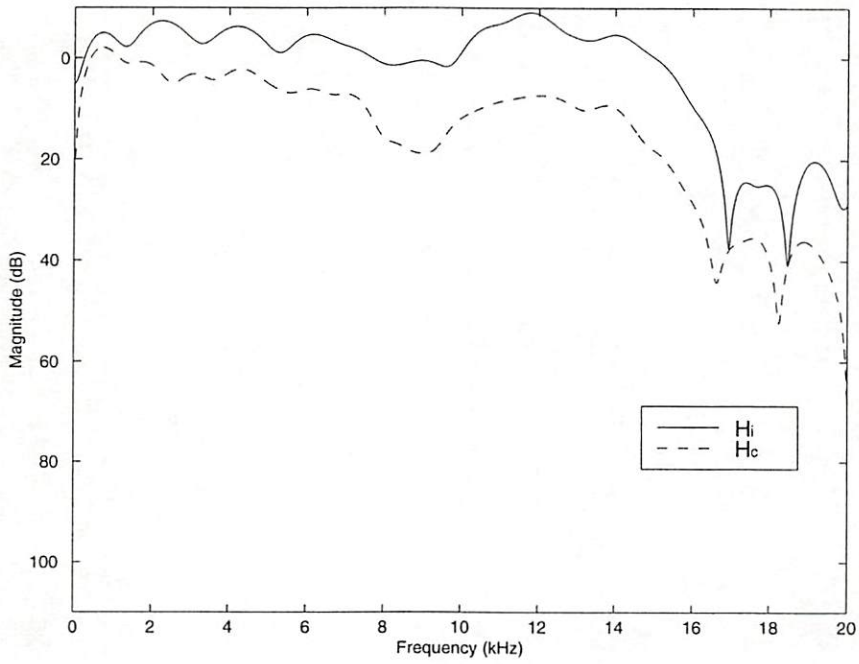

Figure 2.10: The difference in dB between the ipsilateral (H_i) and the contralateral (H_c) terms shows the effect of head shadowing with no crosstalk cancellation. In this set-up the loudspeakers were 50 cm apart and the head was located in the symmetric (center) position at a distance of 50 cm from the loudspeaker baffle plane.

2.5.2 Crosstalk Cancellation

If we denote in the upper equation of (2.12) the delay introduced by H_c/H_i as d_1 and the delay introduced by H_R/H_i as d_2 then, in the z-domain, we find that the filter can be written as

$$F_L = \frac{H_L}{H_i} z^{-(d_1+d_2)} - \frac{H_c}{H_i} z^{-d_1} \frac{H_R}{H_i} z^{-d_2} \tag{2.23}$$

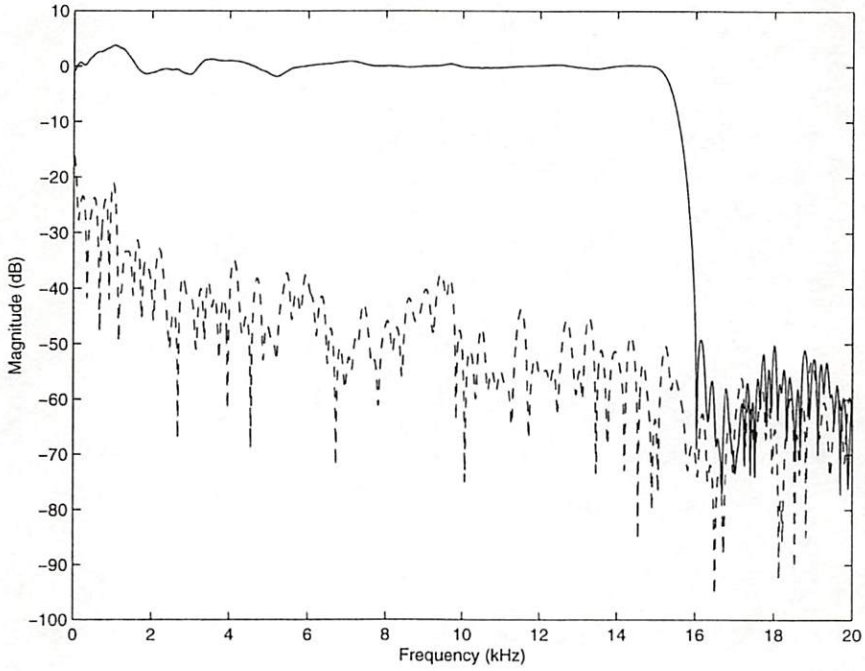

Figure 2.11: Measured HRTF data from the loudspeakers (H_i and H_c) were used to simulate the physical system and design a set of filters to eliminate the crosstalk. The resulting diagonal (solid line) and off-diagonal (dotted line) terms of (2.27) produced by our simulation using the LMS method are plotted above. The diagonal term is very close to 1 (0 dB) from 2 kHz to 15 kHz and deviates only slightly in the region below 1 kHz. The off-diagonal term starts at -15 dB and remains below -30 dB from 1 kHz to 15 kHz.

Note that the delay for H_L/H_i in (2.23) must be equal to the sum of d_1 and d_2. The delay introduced by the filter F_R should also be equal to $d_1 + d_2$. In the time domain (2.23) becomes

$$\begin{aligned} \boldsymbol{f}_l &= \boldsymbol{h}_{li} - \boldsymbol{h}_{ci} * \boldsymbol{h}_{ri} \\ \boldsymbol{f}_r &= \boldsymbol{h}_{li} - \boldsymbol{h}_{ci} * \boldsymbol{h}_{li} \end{aligned} \tag{2.24}$$

in which $*$ denotes convolution.

In order to design the filter for each channel, each of the three filters $\boldsymbol{h}_{li}$,$\boldsymbol{h}_{ci}$ and $\boldsymbol{h}_{ri}$ can be designed separately, and then be combined using (2.24) to obtain the desired final filter. This method is preferable when H_L, H_c and H_R are given in the time domain (*e.g.* from a measurement). In this case note that the delay introduced by $\boldsymbol{h}_{li}$ in $\boldsymbol{f}_l$ is $d_1 + d_2$ while in $\boldsymbol{f}_r$ it is d_2. A similar argument holds for $\boldsymbol{h}_{ri}$. This means that the filters $\boldsymbol{h}_{li}$ and $\boldsymbol{h}_{ri}$ required for $\boldsymbol{f}_l$ will be different from the filters $\boldsymbol{h}_{li}$ and $\boldsymbol{h}_{ri}$ required for $\boldsymbol{f}_r$. Although in this case it is possible to take advantage of the equality of these terms, it should be stressed that in the non-symmetrical case these filters will be different both in the magnitude and phase domains. The advantage of designing two FIR filters, one for each channel, is that these filters implement all the required functions of the virtual rendering system while their order can be kept at a computationally feasible level. However, these filters are useful for applications where the virtual sound source and listener position remain constant. Other possible implementations of our method can be found at Chapter 3. It is also of interest to note that filter lengths should be chosen accordingly, since convolution of two filters with lengths l and p results in a filter with length $l + p - 1$ and in order to subtract two filters they should be of the same length.

An interesting test of the performance of the methods described is to measure the crosstalk cancellation that is achieved. That is, when both loudspeakers produce sound, the sound pressure level at the contralateral ear must be very low compared with the sound pressure level at the ipsilateral ear. A certain degree of crosstalk cancellation is achieved even with no filtering due to the head shadowing, particularly at higher

frequencies (Fig. 2.10). Toole [99, 100] and Walker [103] studied the psychoacoustic effects of early reflections and found that in order to remain inaudible they must be at least 15 dB below the direct sound in spectrum level. A successful crosstalk cancellation scheme should therefore result in at least a 15 dB attenuation of the crosstalk term.

For the symmetric positioning of the listener that we have examined, we saw that for the binaural input case we can set $H_L = H_R = 1$ in (2.8) since the virtual source HRTF's are already contained in the binaural signal. Then, (2.8) becomes

$$\begin{bmatrix} E_L \\ E_R \end{bmatrix} = \begin{bmatrix} H_i & H_c \\ H_c & H_i \end{bmatrix} \begin{bmatrix} 1 & -\frac{H_c}{H_i} \\ -\frac{H_c}{H_i} & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{H_i} & 0 \\ 0 & \frac{1}{H_i} \end{bmatrix} \begin{bmatrix} S_L \\ S_R \end{bmatrix} \tag{2.25}$$

in which ideally $E_L = S_L$ and $E_R = S_R$. If we define the filters $F_{ii} = 1/H_i$ and $F_{ci} = -H_c/H_i^2$, then (2.25) can be written as

$$\begin{bmatrix} E_L \\ E_R \end{bmatrix} = \begin{bmatrix} H_i & H_c \\ H_c & H_i \end{bmatrix} \begin{bmatrix} F_{ii} & F_{ci} \\ F_{ci} & F_{ii} \end{bmatrix} \begin{bmatrix} S_L \\ S_R \end{bmatrix} \tag{2.26}$$

which finally becomes

$$\begin{bmatrix} E_L \\ E_R \end{bmatrix} = \begin{bmatrix} H_iF_{ii} + H_cF_{ci} & H_iF_{ci} + H_cF_{ii} \\ H_iF_{ci} + H_cF_{ii} & H_iF_{ii} + H_cF_{ci} \end{bmatrix} \begin{bmatrix} S_L \\ S_R \end{bmatrix} \tag{2.27}$$

In order to deliver the desired binaural signal to each ear (*i.e.* , $E_L = S_L$ and $E_R = S_R$) the diagonal terms $H_iF_{ii} + H_cF_{ci}$ must be 1 (this would mean that the loudspeaker

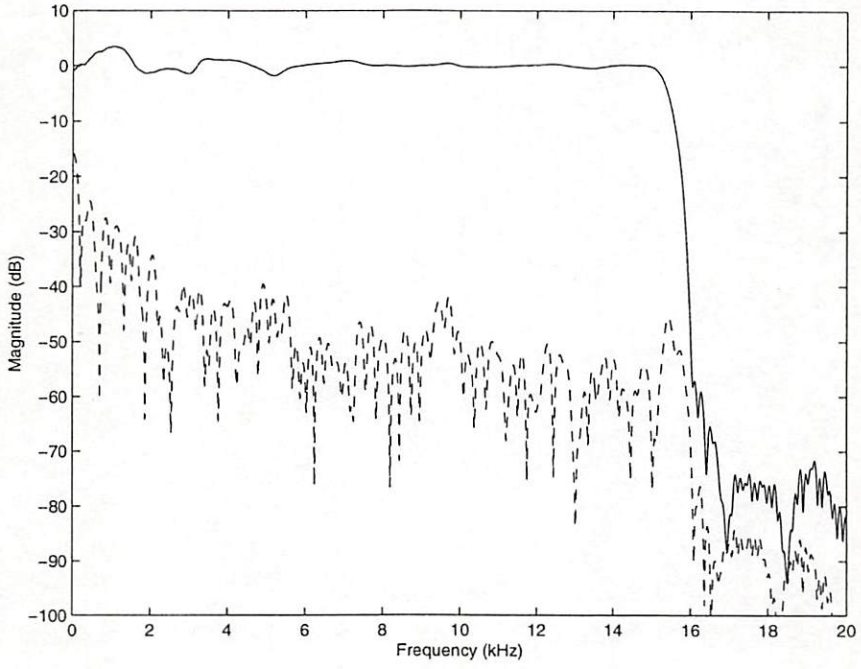

Figure 2.12: Measured HRTF data from the loudspeakers (H_i and H_c) were used to simulate the physical system and design a set of filters to eliminate the crosstalk. The resulting diagonal (solid line) and off-diagonal (dotted line) terms of (2.27) produced by our simulation using the Least-Squares method are plotted above. The diagonal term is very close to 1 (0 dB) from 2 kHz to 15 kHz and deviates only slightly in the region below 1 kHz. The off-diagonal term starts at -15 dB and remains below -30 dB from 1 kHz to 15 kHz.

frequency response inversion has succeeded) and the off-diagonal term $H_iF_{ci} + H_cF_{ii}$ must be 0 (this would mean that the crosstalk cancellation has succeeded).

We designed the filters F_{ii} and F_{ci} using both LMS and Least-Squares methods. For the LMS method, we designed the filter $\boldsymbol{f}_{ii}$ using a length of 349 taps, introducing a delay of 140 samples and an SER of 44.1 dB. For the filter $\boldsymbol{f}_{ci}$ we designed a filter of 150 taps length, delay of 70 samples and a resulting SER of 31.4 dB with frequency response H_c/H_i, and a filter of 200 taps length, delay of 70 samples and SER of 31.6

dB with frequency response $1/H_i$, and then convolved their time domain responses. As mentioned earlier, this procedure is preferable when the HRTF's are given in the time domain. We used the measured HRTF data from the loudspeakers (H_i and H_c) to simulate the physical system and designed a set of filters to eliminate the crosstalk. The resulting diagonal and off-diagonal terms produced by our simulation are plotted in Fig. 2.11, in which the diagonal term is plotted as a solid line and the off-diagonal term as a dotted line. As can be seen in the plot, the diagonal term is very close to 1 (0 dB) from 2 kHz to 15 kHz and deviates only slightly in the region below 1 kHz. The off-diagonal term starts at -15 dB and remains below -30 dB from 1 kHz to 15 kHz. For the Least-Squares method, we designed the filter $\boldsymbol{f}_{ii}$ using a length of 349 taps, introducing a delay of 140 samples and an SER of 44.9 dB. The filter $\boldsymbol{f}_{ci}$ was designed using a filter of 150 taps length, a delay of 70 samples and SER of 31.6 dB with frequency response H_c/H_i, and a filter of frequency response of 200 taps length, delay of 70 samples and SER of 33 dB and then convolved their time domain responses. The resulting diagonal and off-diagonal terms are plotted in Fig. 2.12, in which the diagonal term is plotted as a solid line and the off-diagonal term as a dotted line. As in the LMS case, the diagonal term is near 1 (0 dB) in the range of 20 Hz to 15 kHz and the off-diagonal term starts at -15 dB and remains below -30 dB up to 15 kHz.

2.6 Conclusions

Several theoretical and practical aspects regarding the implementation of immersive audio rendering were discussed in this chapter. They include inversion of non-minimum

phase filters and crosstalk cancellation that is an inherent problem in loudspeaker-based rendering. Two methods were examined to implement a set of filters that can be used to generate the necessary inverse filters required for rendering virtual sound sources, namely the Least-Squares and LMS algorithms. Our simulations have shown that both methods provide good crosstalk cancellation results using various HRTF's. It should be noted that there are still some unanswered questions that can only be addressed by psychoacoustic evaluation. Although mathematical measures such as the SER give an indication of relative performance among different methods, the final validation should be performed using the human ear. Such a study would require an anechoic chamber or equalized room response so that reflections and reverberation frequency alterations can be minimized. This was not treated in this work (although informal listening tests proved the validity of our methods) and can be the subject of future work.

One of the main advantages of the FTF implementation of the Least-Squares algorithm is that it is highly suitable for real-time implementations. This is of particular importance for the case of a moving listener in which a different set of HRTF's must be implemented for every listener position.

Chapter 3

Asymmetry and Real-Time Considerations for Immersive Audio Rendering

3.1 Overview

In the previous chapter, it was shown that it is possible to render virtual sound sources in space using an existing two-loudspeaker audio reproduction system. The solution was given for a listener symmetrically positioned with respect to the loudspeakers, for ease of comparison of our methods with previous work. The methods described, however, were claimed to be easily extendable to the non-symmetric scenario. This chapter shows how this extension can form a basis for a real-time rendering system with a moving listener, assuming that accurate localization of the listener's ears is available [39].

3.2 Introduction

In implementing a sound virtual rendering system, one is faced with three main challenges. First the rendering of virtual sound sources using the head-related transfer functions (HRTF's), second the cancellation of the crosstalk terms that is necessary for loudspeaker-based rendering, and third the localization of the listener's ears in order to dynamically adjust both the HRTF's and crosstalk cancellation filters as the listener moves.

For a given sound direction, localization can be accomplished by filtering an audio signal with the appropriate pair of HRTF's (one for each ear). This is true, though, only when using headphones for sound reproduction. When loudspeakers are used, it is clear that the physical setting introduces cross-terms, since each ear receives sound from both loudspeakers. These cross-terms need to be canceled, so that we can exert sufficient control on what reaches each of the listener's ears, as in the headphones case. Additionally, the frequency response of the loudspeakers in practice is not flat and needs to be equalized (this is also true for the response of the headphones). A solution for these issues was given in Chapter 2 and the extension of these methods will be given in this chapter, in Section 3.3. In addition, it is attempted in this chapter to form a solution that is of relatively low computational complexity, in order to design a system that will be capable of rendering virtual sound sources for a moving listener in real-time. The approach followed here is described in Section 3.4 and realizes acoustic crosstalk cancellation based on Karhunen-Loeve expansion of precalculated filters. KLE has the

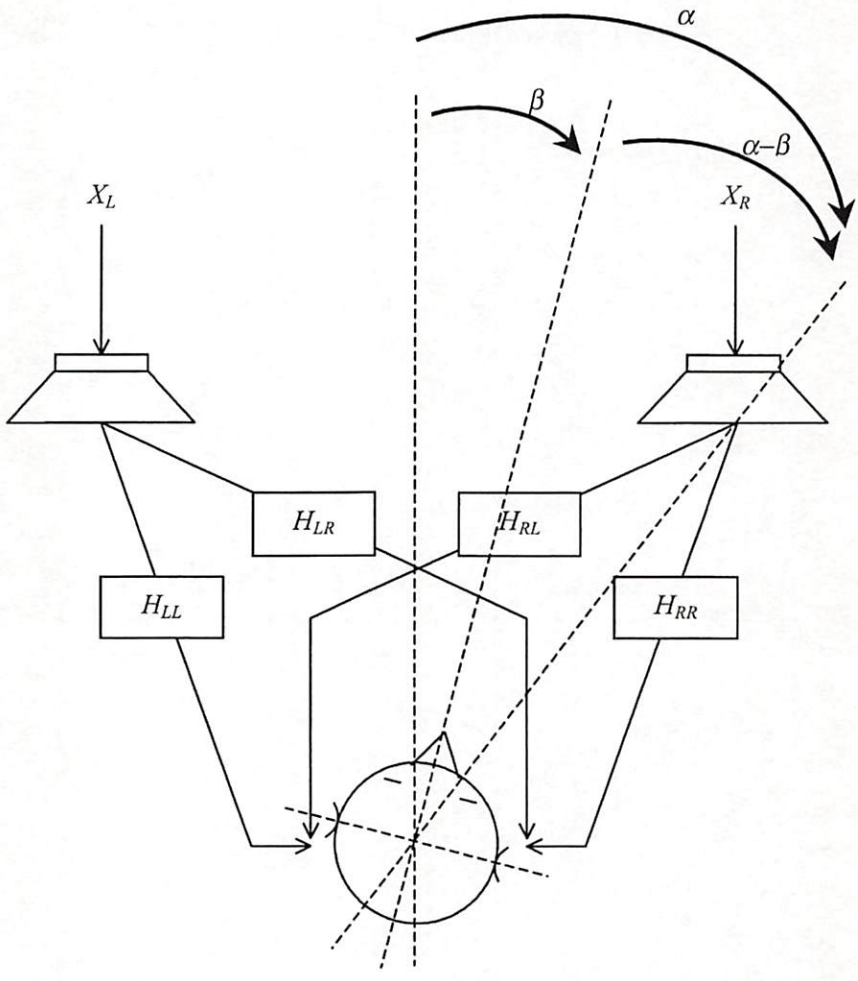

Figure 3.1: Two loudspeaker-based spatial audio rendering system showing the ipsilateral (H_{LL} and H_{RR}) and contralateral (H_{LR} and H_{RL}) terms for a rotating listener.

additional advantage of permitting interpolation for the purpose of deriving filters for every listener position from a smaller number of HRTF measurements.

3.3 Asymmetry Considerations

As explained in Chapter 2, in order to deliver the appropriate binaural sound field to each ear through loudspeakers, it is necessary to eliminate the crosstalk that is inherent in all loudspeaker-based systems.

Crosstalk cancellation can be achieved by eliminating the terms H_{RL} and H_{LR} (Fig. 3.1), so that each loudspeaker is perceived to produce sound only for the corresponding ipsilateral ear. Again, the ipsilateral terms (H_{LL}, H_{RR}) and the contralateral terms (H_{RL}, H_{LR}) are the HRTF's associated with the position of the two loudspeakers with respect to a specified position of the listener's ears. This implies that if the position of the listener changes then these terms must also change in order to correspond to the HRTF's for the new listener position.

In our analysis we present the non-symmetric case for a listener placed at the center between two loudspeakers but being able to do rotational movement[1] (assuming the ears of the listener are at the same level as the loudspeakers for simplicity purposes only). For this case, the ipsilateral and contralateral terms are not equal, however they still correspond to the HRTF's of a specific angle. This angle can be found based on the angle of the loudspeaker placement with respect to the listener and to the angle of rotation of the listener with respect to the fully symmetric case. If the angle of the loudspeakers with respect to the median plane is α° referring to the setting as in Fig. 3.1, in which the listener is seated symmetrically with respect to the two loudspeakers but

[1]This practically covers the general case of movement along any dimension. This is true since a non-symmetrical – with respect to the loudspeakers – listener position, can be treated as rotational movement in a symmetrical setting by applying an appropriate delay and attenuation to the signal that is rendered by the loudspeaker that is closer to the listener [36].

has rotated $\beta°$ clockwise, then the required HRTF's will be as follows: H_{RR} will be the ipsilateral HRTF for the angle $(\alpha - \beta \mod 360)°$ and H_{RL} will be the corresponding contralateral HRTF, while the same is true for H_{LL} and H_{LR} but for the angle $(\alpha + \beta \mod 360)°$ (mod stands for the modulo operation). For the system described, the range of β can be from 0° to 90° for both clockwise and counter-clockwise rotation. For angles greater than these, front-back confusions will dominate since the listener will not be in visible contact with the loudspeakers. A solution for this problem is described in [34], where two additional loudspeakers, opposite to the original ones, are utilized when the listener rotates more than 90°.

As in Chapter 2, matrix notation is used to represent the loudspeaker-ear system. The following analysis corresponds to the frequency domain. We define H_L as the virtual sound source HRTF for the left ear, H_R as the virtual sound source HRTF for the right ear, H_{RR} , H_{RL}, H_{LL} and H_{LR} as described above, and S as the monaural input sound. Then the signals E_L and E_R at the left and right eardrums respectively should, ideally, be equal with the HRTF-processed monaural sound S_L and S_R (the input to the crosstalk canceling system) and are given by

$$\begin{bmatrix} E_L \\ E_R \end{bmatrix} = \begin{bmatrix} S_L \\ S_R \end{bmatrix} = \begin{bmatrix} H_L & 0 \\ 0 & H_R \end{bmatrix} \begin{bmatrix} S \\ S \end{bmatrix} \tag{3.1}$$

The contralateral and ipsilateral terms from the loudspeakers will introduce an additional transfer matrix

$$\begin{bmatrix} E_L \\ E_R \end{bmatrix} = \begin{bmatrix} H_{LL} & H_{RL} \\ H_{LR} & H_{RR} \end{bmatrix} \begin{bmatrix} S_L \\ S_R \end{bmatrix} \tag{3.2}$$

Comparing (3.1) and (3.2), it is apparent that the required preprocessing requires inversion of the physical system matrix

$$\begin{bmatrix} E_L \\ E_R \end{bmatrix} = \begin{bmatrix} H_{LL} & H_{RL} \\ H_{LR} & H_{RR} \end{bmatrix} \begin{bmatrix} H_{LL} & H_{RL} \\ H_{LR} & H_{RR} \end{bmatrix}^{-1} \begin{bmatrix} S_L \\ S_R \end{bmatrix} \tag{3.3}$$

Solving (3.3) we find

$$\begin{bmatrix} E_L \\ E_R \end{bmatrix} = \begin{bmatrix} H_{LL} & H_{RL} \\ H_{LR} & H_{RR} \end{bmatrix} \frac{1}{H_{RR}H_{LL}} \frac{1}{\left(1 - \frac{H_{RL}}{H_{RR}}\frac{H_{LR}}{H_{LL}}\right)} \begin{bmatrix} H_{RR} & -H_{RL} \\ -H_{LR} & H_{LL} \end{bmatrix} \begin{bmatrix} S_L \\ S_R \end{bmatrix} \tag{3.4}$$

which can finally be written as

$$\begin{bmatrix} E_L \\ E_R \end{bmatrix} = \begin{bmatrix} 1 & -\frac{H_{RL}}{H_{LL}} \\ -\frac{H_{LR}}{H_{RR}} & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{H_{LL}} & 0 \\ 0 & \frac{1}{H_{RR}} \end{bmatrix} \begin{bmatrix} S_L \\ S_R \end{bmatrix} \tag{3.5}$$

assuming that

$$\frac{1}{\left(1 - \frac{H_{RL}}{H_{RR}}\frac{H_{LR}}{H_{LL}}\right)} \approx 1 \tag{3.6}$$

As explained in Chapter 2, this assumption is based on the fact that the contralateral term is of substantially less power than the ipsilateral term because of the shadowing caused by the head. Finally, the signals X_L and X_R that have to be presented to the left

and right loudspeaker respectively in order to render the virtual source at the desired location are given by

$$\begin{bmatrix} X_L \\ X_R \end{bmatrix} = \begin{bmatrix} \frac{1}{H_{LL}} & -\frac{H_{RL}}{H_{RR}}\frac{1}{H_{LL}} \\ -\frac{H_{LR}}{H_{LL}}\frac{1}{H_{RR}} & \frac{1}{H_{RR}} \end{bmatrix} \begin{bmatrix} S_L \\ S_R \end{bmatrix} \tag{3.7}$$

which can be written as

$$\begin{aligned} X_L &= \left(S_L - \frac{H_{RL}}{H_{RR}}S_R\right)\frac{1}{H_{LL}} \\ X_R &= \left(S_R - \frac{H_{LR}}{H_{LL}}S_L\right)\frac{1}{H_{RR}} \end{aligned} \tag{3.8}$$

This implies that four different filters should be designed as follows:

$$\begin{aligned} F_{LL} &= \frac{1}{H_{LL}} \\ F_{LR} &= -\frac{H_{LR}}{H_{LL}} \\ F_{RR} &= \frac{1}{H_{RR}} \\ F_{RL} &= -\frac{H_{RL}}{H_{RR}} \end{aligned} \tag{3.9}$$

The binaural signals pass through these filters, which should form a lattice structure as in Fig. 3.2, and then each channel is led to the corresponding loudspeaker. The delays introduced in this figure are to imply that special care should be taken for the direct and cross-terms of this diagram to introduce the same amount of delay to the signals. For example, if all filters of (3.9) are designed with same amount of modeling delay, then a delay of k samples – equal to the modeling delay in the filter design –

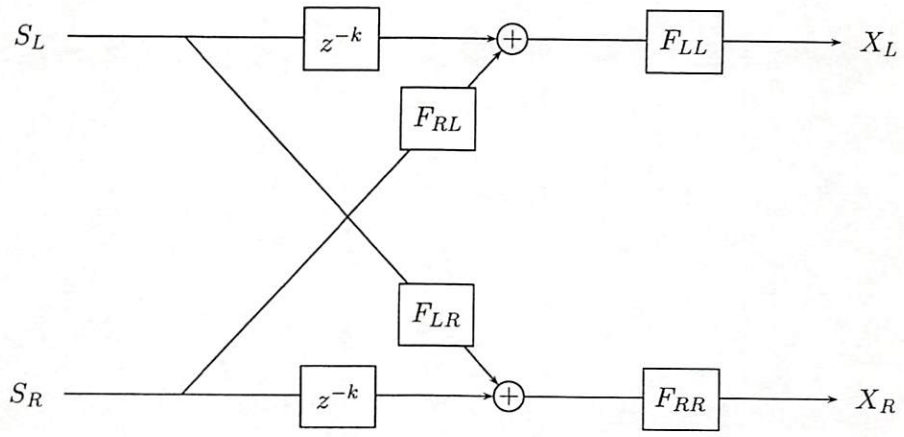

Figure 3.2: Lattice structure for implementing crosstalk cancellation using the filters defined in (3.9).

should be introduced so that the addition in the diagram is meaningful. This structure has been the immediate result of the assumption that the module under consideration was designed for the task of crosstalk cancellation exclusively. It was assumed that filtering with the appropriate HRTF's of the monaural sound has been implemented in a prior stage from a different module. If the particular application is such that the virtual position of the sound source does not vary with time, then the system proposed in Chapter 2 is more appropriate since the designed FIR filters incorporate the HRTF information as well.

3.4 Real-Time Considerations

The analysis in the previous section has shown that crosstalk cancellation requires, as in Chapter 2, the implementation of preprocessing filters of the type $H_{\text{inv}} = H_x/H_y$. In Chapter 2 we introduced two methods that maintain the HRTF phase information.

Although one of these methods has been chosen because of its advantage of combining computational efficiency and fast convergence, it is still quite demanding for a real-time implementation, especially in the case it has to be combined with other modules, such as head tracking and HRTF modeling. The solution that we present here is based on precomputing the crosstalk filters for all the possible angles with the methods of Chapter 2, and then use low-rank modeling for the purpose of data reduction and position interpolation.

3.4.1 Low-Rank Modeling

We employ the Karhunen-Loeve Expansion (KLE) [43] for the purpose of modeling the resulting filters in a low-dimensional space and, additionally, for interpolating between the available listener positions (the listener positions for which the crosstalk filters have been calculated). For this purpose, each crosstalk filter is treated as a vector of measurements and is denoted by $\boldsymbol{h}_j$, where j ranges from 1 to P and P is the number of all the crosstalk filters used for all the desired listener rotation angles (4 filters for each angle as in (3.9)).

In KLE, a vector of measurements can be expanded into an orthonormal basis, which actually consists of the eigenvectors of the covariance matrix that describes the measurement process. In the case of multiple vectors that we examine, a possible procedure [111] is to define a time-averaged covariance matrix such as

$$\mathbf{R} = \frac{1}{P}\sum_{j=1}^{P} \left(\boldsymbol{h}_j - \boldsymbol{h}_{\text{av}}\right)\left(\boldsymbol{h}_j - \boldsymbol{h}_{\text{av}}\right)^T \tag{3.10}$$

where,

$$\boldsymbol{h}_{\text{av}} = \frac{1}{P}\sum_{j=1}^{P}\boldsymbol{h}_j \tag{3.11}$$

is the average vector of all the vectors used. Then, the vector $\boldsymbol{h}_j$ can be represented as an expansion of orthonormal vectors as

$$\boldsymbol{h}_j = \mathbf{Q}\boldsymbol{w}_j + \boldsymbol{h}_{\text{av}} \tag{3.12}$$

In (3.12) $\boldsymbol{Q}$ is a matrix whose columns are the eigenvectors of $\boldsymbol{R}$, and $\boldsymbol{w}_j$ are the corresponding coefficients, given by

$$\boldsymbol{w}_j = \mathbf{Q}^T\left(\boldsymbol{h}_j - \boldsymbol{h}_{\text{av}}\right) \tag{3.13}$$

For the listener rotation angles that there do not exist any HRTF measurements and, therefore, no crosstalk filters can be designed, it is possible to calculate corresponding coefficients by linearly interpolating between the two closest angles for which coefficients can be found by using (3.13). Additionally, low rank modeling of the crosstalk filters is possible by using only K eigenvectors of R that correspond to the K largest eigenvalues ($K < P$). Then $\mathbf{Q}$ contains only these K eigenvectors and its dimensions become P by K instead of P by P. This is especially effective for the case that the remaining $P - K$ eigenvalues are very close to zero, since the modeling error between the vector $\boldsymbol{h}_j$ and the reconstructed $\boldsymbol{h}_{rj}$ from the low-rank model is analogous to the summation of the $P - K$ eigenvalues that were considered small. As it will be shown in the next

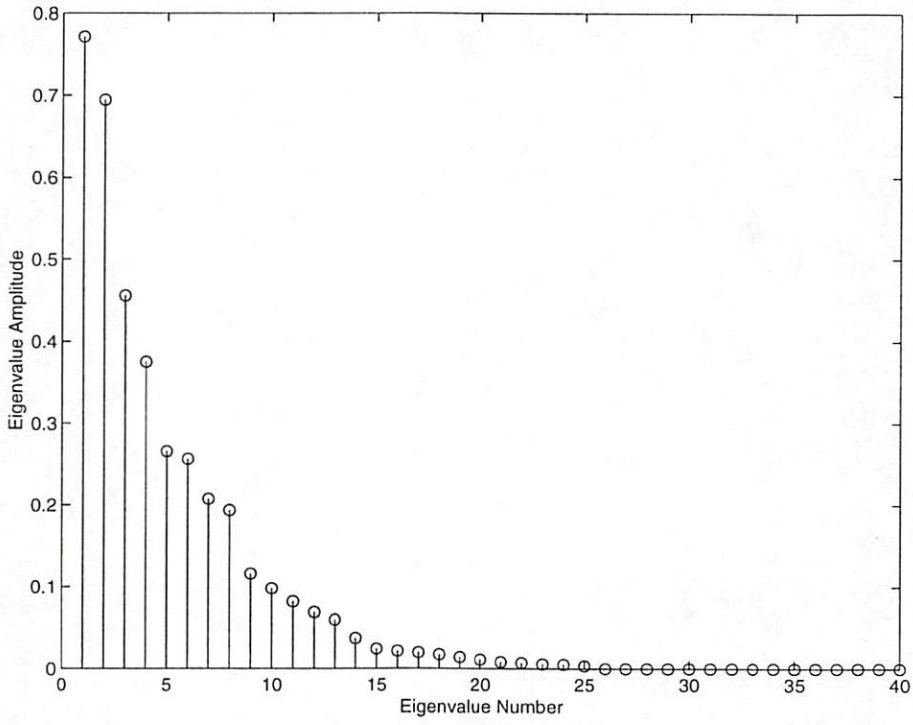

Figure 3.3: The 40 largest of the 2000 eigenvalues of the covariance matrix formed from the designed crosstalk filters. It is clear that a Karhunen-Loeve expansion of these vectors based on the first 25 eigenvalues of this matrix will involve minimal modeling error.

paragraph, this is true for the application that we examine and, for the specific filters designed, only a very small number of eigenvalues were substantially greater than zero.

In section 3.5 we describe our findings by showing simulation results of the performance of the filter design method described, comparing the designed filters with the filters produced by the low-rank modeling method.

3.5 Simulation Results

This section is concerned with the modeling error of the previously described KLE modeling of the crosstalk filters. The HRTF's that were used for the results mentioned in this section were the ones made available by researchers of the Massachusetts Institute of Technology (MIT) [37]. The reason for this was that measurements were available every 5° (assuming that the listener's ears were at the same elevation as the loudspeakers). For a more general application, the given measurements for all different elevation angles could be used (that is, for the case that the listener's level with respect to the loudspeakers is not constant). It is certainly expected that, in most cases, KLE will offer a substantial improvement in performance, as well as a means for interpolation.

Using these measurements, crosstalk filters were created for a listener rotating 180° (90° left to 90° right) for every 5°. The idea, then, was to use KLE so that the listener rotation angles that were not available by measurements could be calculated by means of interpolation, using the available measurements. Instead of using synthetic HRTF's by applying KLE to the available measurements and then designing the required filters as is the usual case in the existing literature, our approach was different. The filters for the available angles were designed and then KLE was applied to those filters. This approach is by far more appropriate for real-time applications than redesigning the required filters every time that the listener moves.

For the listener positions that were mentioned, two different basis sets were designed for the two different types of filters that were to be designed using the methods described in Chapter 2. That is, KLE was applied to filers of the type $1/H_i$ and to filters H_c/H_i

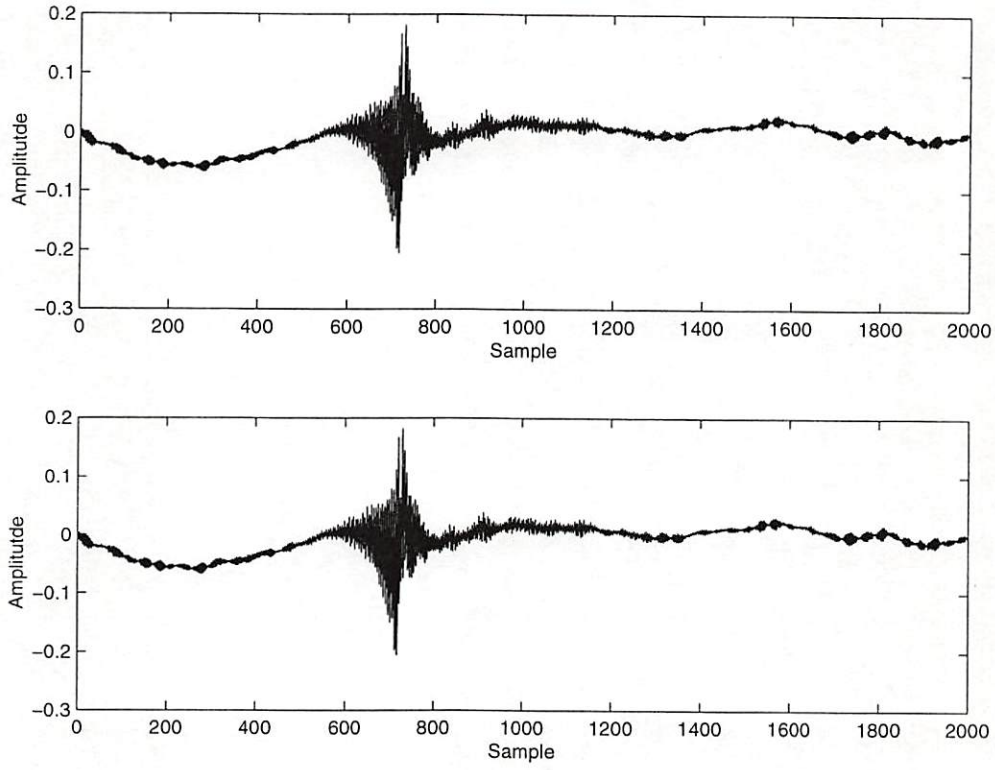

Figure 3.4: Impulse responses of the designed filter (top) and the KLE-modeled filter (bottom) for the particular HRTF corresponding to 10° rotation (crosstalk filters of type $1/H_i$).

resulting in two different basis sets. The reason for this separation was that filters of the same type were quite similar in the time domain, which means that a smaller number of basis vectors would be required for modeling the filters with the least error. The discussion that follows describes the results obtained for the filters of the type $1/H_i$.

All the filters that were designed were of 2000 taps length and 700 samples of delay. Thus, the covariance matrix $\boldsymbol{R}$ was of dimension 2000-by-2000. From its 2000 eigenvalues, the largest 40 are shown in Fig. 3.3. It is obvious from this figure that except from the largest 25 eigenvalues, all the remaining ones can be considered as being

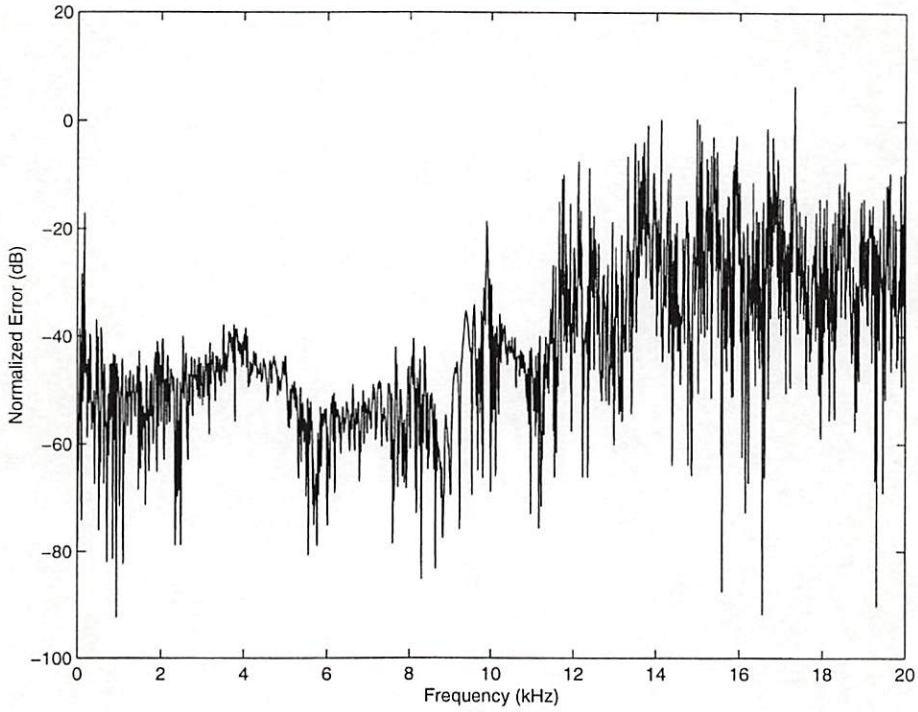

Figure 3.5: Normalized error between the magnitude responses of the designed filter and the KLE-modeled filter that are represented in Fig. 3.4.

approximately zero, resulting in a basis of 25 eigenvectors of 2000 samples each, the eigenvectors of $\mathbf{R}$ that correspond to these 25 eigenvalues. By using these 25 eigenvectors, a very low modeling error was achieved. In Fig. 3.4 one of the filters corresponding to 10° clockwise rotation is shown (upper plot) with the filter that resulted by using the low-rank model (bottom plot). The two filters are obviously very close and their time-domain error (SER) of 45 dB (defined in Chapter 2) verifies this. In Fig. 3.5 their error in frequency domain is shown (normalized with the desired response). For a wide range of the frequency band it is obvious that the error is quite acceptable. For the reasons mentioned in Chapter 2, for frequencies above 15 kHz, the degradation of the performance is not considered important.

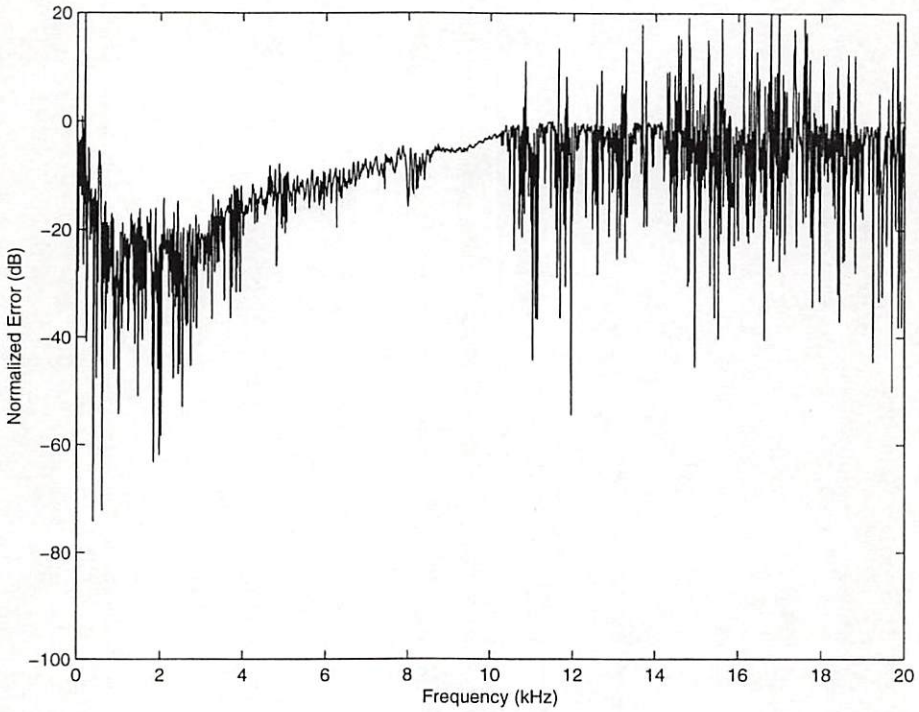

Figure 3.6: Normalized error between the magnitude responses of the designed filter and the KLE-modeled filter based on linear interpolation of the KL coefficients of the crosstalk filters corresponding to the two closest angles.

Finally, an example is given of the modeling error for the case of filter interpolation. In Fig. 3.6 the normalized error in the frequency domain is depicted between a crosstalk filter designed for a clockwise rotational angle of 25° and its modeled version based on linearly interpolating the KL coefficients of the crosstalk filters for angles 20° and 30°, calculated on a 10° grid. As expected, the error in this case increases significantly, but still remains at an acceptable level in the most important region, below 10 kHz.

3.6 Conclusions

A method for generating the required crosstalk cancellation filters as the listener moves was developed based on Low-Rank modeling. Using Karhunen-Loeve expansion we can interpolate among listener positions from a smaller number of HRTF measurements. A set of corresponding crosstalk cancellation filters was precomputed for the available (measured) HRTF angles and then KLE was applied to these filters. This approach significantly reduces the required computational resources and is more appropriate for integration in a real-time implementation with head tracking. From our results we found that the KL expansion allows only a small number of eigenvalues to be retained with the remaining eigenvalues discarded. The resulting modeled transfer functions are shown to have an SER in the order of 45 dB compared to the measured data, thus the improvement in computational performance comes at a very low cost in model accuracy. In addition, linear interpolation of the KL coefficients of existing filters offers a means of designing crosstalk filters for positions that measurements are not available, with an acceptable modeling error.

Chapter 4

Time-Frequency Analysis and Synthesis of Audio Signals

4.1 Overview

In this chapter, a brief introduction is given to time-frequency signal analysis and synthesis. Such methods are very useful for the treatment of non-stationary signals (for example speech and audio signals), whose spectral properties vary significantly with time. The most common of such methods is the Short-Time Fourier Transform (STFT), however, other less commonly known methods will also be described. Bilinear time-frequency distributions, as such, will be described in more detail. Especially for audio signal processing, these distributions are advantageous for both analysis and synthesis [82, 83]. The interest is on discrete-time signals. For the case of bilinear distributions, the continuous-time case will also be briefly covered, since the definitions of the discrete case can be better understood when they are related with their continuous-time

counterparts. This chapter does not offer a complete coverage of all time-frequency distributions; it merely attempts to provide a unified treatment of such distributions with respect to audio signal analysis/synthesis and offer the required background for the tools used in the subsequent chapters of this work.

4.2 STFT Analysis and Synthesis

The term time-frequency analysis corresponds to many different signal representations that have a common objective: to give an accurate description of a signal's time-varying spectrum. The most commonly used model for achieving such analysis is to divide the signal in short segments and compute the Fourier transform of each segment, resulting in a time-frequency representation known as the short-time Fourier transform (STFT). Consider a signal $s(m)$, then its STFT $F_s(n,\omega)$ is defined as[1]

$$F_s(n,\omega) = \sum_m s(m)w(n-m)e^{-j\omega m} \tag{4.1}$$

where $w(m)$ is a window whose main purpose is to divide the signal in small segments. It is apparent that when the length of the window is small compared to the length of the signal, the STFT gives a time-varying analysis of the spectral properties of the signal. The choice of window is important given the trade-off between resolution in time and frequency. A detailed analysis of the importance of the window type and length can be found for example in [85]. STFT analysis is of very low computational complexity since it involves calculating the DFT (hence the FFT) for relatively small signal segments.

[1] All summations and integrations in this chapter are from $-\infty$ to ∞ unless stated otherwise.

Synthesis of signals using the STFT involves modifying the STFT (or its magnitude) at each time point according to some predefined rules. This modification cannot be arbitrary; in most cases overlapping windows are used in the analysis which restricts the synthesis part as well. Exact reconstruction after modification is usually not possible since some restrictions have to be utilized. The Gabor transform [33] offers a means for exact synthesis from the STFT. The drawback is implementation complexity (due to use of infinite-length windows) and restriction in the choice of window. Any attempt to use windows of compact support in the Gabor expansion, results in least-squared (non-exact) synthesis, analogous to the overlap-add methods described next [32]. A more popular procedure is to relax the requirement of exact reconstruction. The most common procedure is to use the Inverse DFT of the modified DFT of each segment and then combine all the segments using overlap-add techniques (*i.e.* adding the (windowed) segments with overlapping of adjacent segments according to the overlapping of the analysis windows, [41, 84, 2]). The algorithm given in [41] is briefly described. Assuming that the STFT $F_s(n,\omega)$ is the result of arbitrary modification of a valid STFT, the objective is to estimate the signal $\hat{s}(m)$ whose STFT $F_{\hat{s}}(n,\omega)$ is close in some sense to $F_s(n,\omega)$. This approximation is the result of the fact that an arbitrary modification of an STFT does not, in general, result in a valid STFT (*i.e.* no sequence exists with such an STFT). Least-squares minimization between the two STFT's, *i.e.*,

$$\min_{\hat{s}} \sum_n \frac{1}{2\pi} \int_{-\pi}^{\pi} |F_s(n,\omega) - F_{\hat{s}}(n,\omega)|^2 \, \mathrm{d}\omega \tag{4.2}$$

yields

$$\hat{s}(m) = \frac{\sum_n w(n-m)s(n,m)}{\sum_m w^2(n-m)} \tag{4.3}$$

where, $s(n,m)$ is the inverse Fourier transform of F_s, that is,

$$s(n,m) = \frac{1}{2\pi}\int_{-\pi}^{\pi} F_s(n,\omega)e^{j\omega m}\,\mathrm{d}\omega \tag{4.4}$$

This overlap-add procedure is apparently very efficient for estimating the desired signal from its modified STFT.

4.2.1 Models for Analysis/Synthesis of Audio Signals

Modifying the STFT of a signal assumes that specific rules have been defined regarding the objective of such modification. It is often the case that use of a model for each signal segment can serve as an intermediate step for defining such rules. Two different models that are particularly suitable for audio signals are the residual/LP (Linear Predictive) and sinusoidal models.

The residual/LP model assumes that each segment of the signal can be considered as stationary. It then applies linear predictive analysis at each segment, modeling the segment samples as a random autoregressive (AR) process and calculating the coefficients of this model (linear predictive coefficients, LPC, [43]). The analysis that follows

refers to samples $s^{(i)}(0), \cdots, s^{(i)}(M-1)$ that form block i of signal $s(m)$. Consider the linear combination of p past samples of the process $s^{(i)}(m)$

$$s_t^{(i)}(m) = \sum_{k=1}^{p} a(k) s^{(i)}(m-k) \tag{4.5}$$

Assume $s_t^{(i)}(m)$ is the linear prediction of the process at time m, with prediction error

$$e(m) = s^{(i)}(m) - s_t^{(i)}(m) \tag{4.6}$$

The transfer function of the prediction coefficients is given by

$$A(z) = \frac{E(z)}{S_{s^{(i)}}(z)} = 1 - \sum_{k=1}^{p} a(k) z^{-k} \tag{4.7}$$

Mean-squared minimization of the error $e(n)$ produces the coefficients $a(i)$. Linear prediction is a special case of linear optimum filtering thus the principle of orthogonality holds. Accordingly, minimization of the error is equivalent to the error $e(m)$ being orthogonal to all the input samples $s^{(i)}(m-l)$ from which the error at time m is calculated (l will be in the interval $[1, p]$), *i.e.*

$$\mathrm{E}\left\{ s^{(i)}(m-l) e(m) \right\} = 0 \tag{4.8}$$

$$\mathrm{E}\left\{ s^{(i)}(m-l) \left[s^{(i)}(m) - \sum_{k=1}^{p} a(k) s^{(i)}(m-k) \right] \right\} = 0 \tag{4.9}$$

$$r(-l) = \sum_{k=1}^{p} a(k) r(k-l) \tag{4.10}$$

in which $r(m)$ is the autocorrelation function of $s^{(i)}(m)$ and $\mathrm{E}\{\cdot\}$ denotes the expectation operator. Finally, since the autocorrelation function is symmetric, we can rewrite equation (4.10) in matrix form as

$$\begin{bmatrix} r(0) & r(1) & \cdots & r(p-1) \\ r(1) & r(0) & \cdots & r(p-2) \\ \vdots & \vdots & \ddots & \vdots \\ r(p-1) & r(p-2) & \cdots & r(0) \end{bmatrix} \begin{bmatrix} a(1) \\ a(2) \\ \vdots \\ a(p) \end{bmatrix} = \begin{bmatrix} r(1) \\ r(2) \\ \vdots \\ r(p) \end{bmatrix} \tag{4.11}$$

The coefficients $a(k)$ can be found from the above equation by inverting the correlation matrix $\mathbf{R}$. This can be done very efficiently using the Levinson-Durbin algorithm, which is a recursive method of solving equation (4.11), based on the special structure of matrix $\mathbf{R}$.

In effect this is an approximation of the spectrum of the particular segment $S_{s^{(i)}}(\omega)$ by an all-pole minimum-phase transfer function with frequency response $G(\omega) = \frac{1}{A(\omega)}$. By filtering the segment samples with the inverse of this transfer function, the residual signal is obtained (thus the model being referred to as the residual/LP model). The residual signal represents the modeling error between the AR approximation of the sample sequence and the exact sequence. Then, modifications are possible either on the LP coefficients or the residual signal, as explained in later chapters of this work. The advantage of this model is that it allows for high-quality synthesis (important in

audio signal processing) since the model incorporates the modeling error (the residual), thus the simplicity of using the model does not come at a cost regarding quality. Note that the model is used for signal enhancement as opposed to reduction of the signal information rate (coding), in which case it is often sensible to have a trade-off between signal quality and model simplicity. It also important to mention that the residual/LP model can be considered as a special case of STFT modification. In other words, this model can be used as an intermediate step following STFT analysis and followed by STFT synthesis by overlap-add methods.

Sinusoidal models have been used extensively in speech processing for high quality speech modification [67, 94, 38]. For audio signals, the sinusoidal model of [90] has been widely known to offer advantages for high-quality modifications. In this approach, the audio signal is again divided in segments modeled as a sum of a deterministic harmonic signal (expressed as a summation of R sinusoids) and a stochastic signal, in other words for block i

$$s^{(i)}(m) = \sum_{r=1}^{R} A_r(m) \cos\left(\theta_r(m)\right) + e(m) \tag{4.12}$$

where $A_r(m)$ is the instantaneous amplitude and $\theta_r(m)$ is the instantaneous phase of the r^{th} sinusoid. The error $e(m)$ is the difference between the original signal and its deterministic harmonic approximation, modeled as filtered white noise. The magnitude of this filter is estimated by subtracting the magnitude of the deterministic part from the magnitude of the original signal and is modeled as a piecewise linear function. The phase is taken to be a random sequence with uniform distribution in the interval $[-\pi,\pi]$. Other modules of this algorithm include peak detection in the magnitude domain for

estimating the harmonics of each segment and peak continuation for estimating the intra-frame trajectories of the harmonics. This model has been proved to be capable of modeling high-quality audio signals and has been applied to timbral modifications.

4.3 Bilinear Distributions for Signal Analysis

Bilinear time-frequency distributions include the spectrogram, the Wigner distribution and many others. A general class of bilinear time-frequency representations is Cohen's class [23], which is a method that generates all existing *bilinear* time-frequency representations and allows for the definition of infinite new ones. This class can be obtained from

$$C(t,\omega) = \iiint s(u+\frac{1}{2}\tau)s^*(u+\frac{1}{2}\tau)\phi(\theta,\tau)e^{-j\theta t - j\tau\omega + j\theta u}\,\mathrm{d}u\,\mathrm{d}\tau\,\mathrm{d}\theta \tag{4.13}$$

in which $\phi(\theta,\tau)$ is called the kernel. A mathematically equivalent expression with (4.13) offering better insight is

$$C(t,\omega) = \int R(t,\tau)e^{-j\omega\tau}\,\mathrm{d}\tau \tag{4.14}$$

where

$$R(t,\tau) = \int r(t-u,\tau)s(u+\frac{1}{2}\tau)s^*(u-\frac{1}{2}\tau)\,\mathrm{d}u \tag{4.15}$$

and

$$r(t,\tau) = \int \phi(\theta,\tau)e^{-jt\theta}\,\mathrm{d}\theta \tag{4.16}$$

In other words, the time-frequency representation in (4.13) is the Fourier transform of the (time-dependent and time-averaged) autocorrelation of the signal $s(t)$ in (4.14). This view offers insight in the motivation for using these distributions, however note that this definition of the time-averaged autocorrelation is slightly different than the common one, so usually this function is called the generalized local autocorrelation function. The difference lies in the use of $s(t+\frac{\tau}{2})s(t-\frac{\tau}{2})$ instead of $s(t)s(t-\tau)$ and also in the use of $r(t,\tau)$, which is an arbitrary window, the Fourier Transform of the kernel. The reasons for using this window are many: either for suppressing the cross-terms that appear in (4.13) (this is briefly explained later), or for defining computationally efficient representations (*i.e.* that do not require infinite summations), or for defining positive distributions (in general, the distributions obtained by (4.13) are not always positive which is counterintuitive since these transformations are informally viewed as energy distributions). This latter view is based on the fact that because of (4.14), these distributions can be considered as a generalization of the power spectrum.

The most important property that leads to new definitions of bilinear transformations is the improved resolution compared to the spectrogram. Other important properties of several bilinear distributions can be found in [23]. However, these transformations are computationally expensive and they also produce cross-terms in multi-component signal analysis because of their bilinear form. The spectrogram, which is the most frequently used time-frequency transformation, is computationally inexpensive and does not produce such artifacts. In order to reduce the computational burden of

these bilinear transformations, a window of compact support can be used in the definition of (4.15) in both t and τ. However, this kind of smoothing comes with a tradeoff regarding resolution. The particular choice of the kernel $\phi(\theta, \tau) = 1$, leads to the Wigner distribution [23, 19], which is defined as

$$W(t, \omega) = \int s(t + \frac{1}{2}\tau) s^*(t - \frac{1}{2}\tau) e^{-j\omega\tau} \, \mathrm{d}\tau \tag{4.17}$$

implying that in this case the autocorrelation becomes

$$R(t, \tau) = s(t + \frac{1}{2}\tau) s^*(t - \frac{1}{2}\tau) \tag{4.18}$$

The Cross-Wigner distribution (CWD), whose usefulness will be made clear later, is the extension

$$W_{s_1 s_2}(t, \omega) = \int s_1(t + \frac{1}{2}\tau) s_2^*(t - \frac{1}{2}\tau) \, \mathrm{d}\tau \tag{4.19}$$

The preceding definitions correspond to deterministic signals. For random signals, the extension of the power spectrum is the Wigner-Ville distribution (see for example [65])

$$W(t, \omega) = \int R(t + \frac{1}{2}\tau, t - \frac{1}{2}\tau) e^{-j\omega\tau} \, \mathrm{d}\tau \tag{4.20}$$

where the autocorrelation of $x(t)$ is defined as

$$R(t_1, t_2) = \mathrm{E}\{x(t_1) x^*(t_2)\} \tag{4.21}$$

It also of interest, for reasons that will be clear later in this chapter, to describe an alternate definition of Cohen's class. Given a function $H(t_1, t_2)$, the Weyl Symbol of H, L_H is defined as:

$$L_H(t, \omega) = \int H(t + \frac{1}{2}\tau, t - \frac{1}{2}\tau) e^{-j\omega\tau} \, \mathrm{d}\tau \tag{4.22}$$

and it is trivial to define the inverse mapping as well (which can be referred to as the inverse Weyl symbol):

$$H(t_1, t_2) = \frac{1}{2\pi} \int L_H(\frac{t_1 + t_2}{2}, \omega) e^{j\omega(t_1 - t_2)} \, \mathrm{d}\omega \tag{4.23}$$

It is obvious that the definition (4.14) can be modified so that every time-frequency distribution belonging to Cohen's class can be considered to be the Weyl symbol of a function $H(t_1, t_2)$. A change of variables connects the autocorrelation in (4.14) with this function with the relation:

$$H(t_1, t_2) = R(\frac{t_1 + t_2}{2}, t_1 - t_2) \tag{4.24}$$

For the case of the Wigner Distribution, by inspection we have

$$H(t_1, t_2) = x(t_1) x^*(t_2) \tag{4.25}$$

Other distributions belonging to Cohen's class are for example the Born-Jordan distribution, the Choi-Williams distribution, the Rihaczek distribution, the spectrogram, *etc.* [23]. The Discrete-Time Wigner Distribution (DTWD) is defined in [20] as

$$W(n,\omega) = 2\sum_{m} s(n+m)s^*(n-m)e^{-j2\omega m} \tag{4.26}$$

that is, the autocorrelation function now is

$$R(n,m) = s(n+m)s^*(n-m) \tag{4.27}$$

This definition results in a sampled continuous-time WD. The problem is that the distribution is π-periodic which means that this definition results in aliasing unless the signal $x(n)$ is restricted to be halfband (to have non-zero spectrum only for frequencies inside the interval $[-\frac{\pi}{2}, \frac{\pi}{2}]$) or oversampled by at least a factor of 2 [10, 22]. In a case of a real signal, its analytic version can be used since only half of its spectrum is enough to recover the signal and no aliasing occurs [10].

4.4 Signal Synthesis from Bilinear Distributions

In this chapter, the Wigner distribution (WD) and smoothed versions of the WD will be examined, serving as a reference on how bilinear transformations can be utilized for signal synthesis. Signal synthesis is defined as follows. Given an arbitrary function $W_s(t,\omega)$, find a signal $\hat{s}(t)$ whose distribution $W_{\hat{s}}(t,\omega)$ is as close as possible to the given function. The problem arises from the fact that if the distribution of a signal

is arbitrarily modified, it no longer corresponds to a valid distribution, so a simple inversion is not possible. The solution of this problem in all the references given is based in least-squared minimization, that is

$$\min_{\hat{s}} \iint |W_s(t,\omega) - W_{\hat{s}}(t,\omega)|^2 \, \mathrm{d}\omega \, \mathrm{d}t \tag{4.28}$$

The applications of such synthesis algorithms are mainly isolating signals from noise or, in general from other unwanted components, as well as creating time-frequency filter banks. Most of the references herein regarding synthesis, give examples explaining where and how these methods can be applied. A tutorial about time-frequency analysis and synthesis applications is [45]. Another important reference, describing the relation of the WD with other time-frequency distributions is [21]. Finally, many of the synthesis algorithms outlined here along with interesting applications can be found in [44].

Signal synthesis from WD is a subject many different authors have dealt with. Here, the most important results will be briefly presented. The initial motivation for Wigner synthesis was given in [97]. This paper describes synthesis from the Ambiguity Function. The Ambiguity Function (AF) is the two-dimensional Fourier Transform of the Wigner Distribution and appears in applications of the Doppler radar. In [97], the author solves the synthesis problem applied to the AF by minimizing the error between the given (non-valid) and the desired (valid) AF by least-squared minimization by expanding the AF in an orthonormal basis. This basis is proved to be the (cross) AF of the orthonormal basis in which the signal itself is expanded. The problem then reduces in finding the eigenvalues and eigenvectors of the matrix of the expansion coefficients (the

inner product of the modified AF and the *induced* basis) and choosing the eigenvector corresponding to the largest eigenvalue. In reality, this is a result to be expected because, as a consequence of the definition of the AF (as well as the WD), the matrix containing the expansion coefficients is a rank-one matrix (considering the WD, this corresponds to the fact that the autocorrelation (4.18) is a separable function). The problem is also solved in the discrete-time domain, but more about this case will be discussed in Section 4.4.1.

The application of the above procedure for WD signal synthesis can be found in [112, 88]. The procedure in [47] is important for gaining insight in the algorithms for DTWD synthesis. It is well known that the minimization problem in (4.28) can be transformed in the autocorrelation domain by using the Parseval relation of the Fourier transform. However, in [47] the problem is transformed in the domain of $H(t_1, t_2)$. The reason is that this function, in contrast with the autocorrelation function (as defined in (4.14)), is hermitian symmetric thus the minimization procedure is equivalent to finding the eigenvalues of $H(t_1, t_2)$ (which because of the symmetry property are real and the corresponding eigensignals are orthonormal) and choosing the eigensignal corresponding to the largest eigenvalue as the solution. This is an intuitive result since $H(t_1, t_2)$ is a

separable function. The algorithm is generalized for all bilinear transformations T_s [2] that are unitary *i.e.* satisfy the Moyal formula [3]

$$\langle T_{ab}, T_{cd} \rangle = \langle a, d \rangle \langle c, b \rangle \tag{4.29}$$

This unitarity constraint ascertains that the solution in the inverse Weyl domain is equivalent to the solution of (4.28). This constraint is dependent on the form of the kernel and the conditions that the kernel must satisfy can be found in [47]. It is also of interest to mention that a synthesis procedure has been defined for the Wigner-Ville distribution in [47]. In this case, the inverse Weyl symbol corresponds to the true autocorrelation of the signal, and the synthesis problem is solved by restricting this function to be positive definite (instead of separable as previously).

4.4.1 Discrete-Time Wigner Synthesis

Most of the algorithms described for WD synthesis cannot be readily used for DTWD synthesis. In DTWD synthesis the induced basis, as defined previously in this chapter, is never orthonormal and can be orthogonal only for very few choices of the original basis. This implication prohibits the adaptation of the algorithm in [112] in the discrete-time domain, as it was pointed out in [55, 106]. It is also true that the inverse Weyl

[2] T_s represents the time-frequency distribution of $s(t)$ and T_{ab} represents the cross- time-frequency distribution of $a(t)$ and $b(t)$ *i.e.*

$$T_{ab}(t,\omega) = \iiint a(u+\frac{1}{2}\tau) b^*(u-\frac{1}{2}\tau) \phi(\theta,\tau) e^{-j\theta t - j\tau\omega + j\theta u} \, \mathrm{d}u \, \mathrm{d}\tau \, \mathrm{d}\theta$$

[3] where $\langle x, y \rangle = \int x(t) y^*(t) \, \mathrm{d}t$ and $\langle T_{ab}, T_{cd} \rangle = \iint T_{ab}(t,\omega) T^*_{cd}(t,\omega) \, \mathrm{d}t \, \mathrm{d}\omega$ *i.e.* inner product relations for 1-D and 2-D functions

symbol of the DTWD as defined in (4.23) cannot be defined for discrete-time signals. A way to overcome this problem is to consider separately even and odd samples of the signals, as is described in [11]. This is an extremely influential work in the context of Wigner synthesis, which again solves the problem by minimizing in the least-squared sense. It is important to mention that the solution of this problem is not unique since if the signal $x(n)$ is a solution to the problem then it is easy to see that the signal $e^{j\phi}x(n)$ with arbitrary ϕ will also be a solution (this is true in general for WD synthesis based on least-squared minimization). The solution given in [11] also results in a similar eigenvalue-eigenvector solution as in [97]. The difference here is that, for the reason stated earlier, the problem is decoupled in odd and even samples of the desired signal, which has the disadvantage that the phase ambiguity will be a separate factor for even and odd samples, a more serious issue compared to phase ambiguity regarding the signal as a whole. A method to overcome this ambiguity under some assumptions is described in [11]. Another reference regarding the issue of the phase ambiguity is [48].

Two algorithms for DTWD synthesis that inspired some methods described in Chapter 6 can be found in [12] and [68]. In [68] an algorithm based on the discrete-time CWD (DTCWD) is given. The algorithm is iterative and is based on the fact that if the reference signal $y(n)$ is an approximation of the desired signal $x(n)$, then we can use their CWD as an approximation of the DTWD of $x(n)$. The advantage is that the problem is decoupled in solving $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}$ where $\mathbf{A}$ is a known matrix and $\boldsymbol{b}$ a known vector. The solution to this problem is based on the Singular Value Decomposition (SVD) of $\mathbf{A}$. In practice, $y(n)$ is not available and this is the reason that the algorithm is iterative: at

each iteration, $y(n)$ is the synthesized $x(n)$ of the previous iteration. In [12], a different approach of DTCWD synthesis is outlined. Using a reference signal $\tilde{x}(n)$ as previously, it is possible to isolate $x(n)$ from a multicomponent signal $s(n)$ under the observation that the DTCWD of two signals is oscillatory whereas the DTWD of a signal is relatively smooth. Under this assumption, taking the DTCWD of $\tilde{x}(n)$ and $s(n)$ and then choosing the part of the distribution that exhibits non-oscillatory properties will result in an approximate DTWD of $x(n)$. This distribution can be used as the modified DTWD that is the input to a DTWD synthesis algorithm as the ones described here.

4.4.2 Smoothed Wigner Synthesis

Smoothed Wigner distributions are obtained by substituting the autocorrelation (4.27) with

$$R(n,m) = \sum_{n'} r(n-n',m)s(n'+m)s^*(n'-m) \tag{4.30}$$

where $r(n,m)$ is, as usual, the Fourier transform of the kernel. DTWD synthesis will be discussed here since the interest is in practical implementation and not theoretical derivations. Different choices of kernel lead to different distributions, the most well known being the Choi-Williams distribution [17] (exponential kernel) and the pseudo-Wigner distribution [20] (finite length window kernel). The former kernel is designed for suppressing the interference terms, while the latter is a kernel that leads to a less computationally expensive distribution. Smoothed Wigner distributions require a different approach for synthesis because of the form of (4.30). The minimization procedure described *e.g.* by [11] leads now to third-order equations because of the use of the kernel.

In [54] this problem is solved by applying an iterative procedure inspired by the DTWD synthesis methods in [11]. The algorithm, called the Quasi Power Algorithm (QPA), is an iterative application of the Wigner synthesis solution, using at each iteration the signal calculated at the previous iteration. A slightly modified version of this algorithm is outlined in [46], where the error minimization is employed only in the region of interest (cf. [68], where a similar modification is proposed).

Algorithms have been designed for the special case of the pseudo-Winger distribution. The interested reader should examine [52, 53, 113] in which methods analogous to the STFT overlap-add procedure in context and complexity are described. There exists a significant trade-off in these methods between complexity and solution optimality.

Chapter 5

Multichannel Audio Resynthesis by Subband-Based Spectral Conversion

5.1 Overview

Multichannel audio offers significant advantages for music reproduction that include the ability to provide better localization and envelopment, as well as reduced imaging distortion. On the other hand, multichannel audio is one of the most demanding media types in terms of transmission requirements. A novel architecture was previously proposed [77], allowing delivery of uncompressed multichannel audio over high-bandwidth communications networks. In most cases, however, bandwidth limitations prohibit transmission of multiple audio channels. In such cases, an alternative would be to transmit only one or two reference channels and recreate the rest of the channels at the receiving end. In this chapter, we propose a system that is capable of synthesizing the required signals from a smaller set of signals recorded in a particular venue.

These synthesized "virtual" microphone signals can be used to produce multichannel recordings that accurately capture the acoustics of the particular venue. Applications of the proposed system include transmission of multichannel audio over the current Internet infrastructure and, as an extension of the methods proposed here, remastering of existing monophonic and stereophonic recordings for multichannel rendering, a topic discussed in Chapter 6.

5.2 Introduction

Multichannel audio can enhance the sense of immersion for a group of listeners by reproducing the sounds that would originate from several directions around the listeners, thus simulating the way we perceive sound in a real acoustical space. On the other hand, multichannel audio is one of the most demanding media types in terms of transmission requirements. A novel architecture allowing delivery of uncompressed multichannel audio over high-bandwidth communications networks was presented in [77]. As suggested there, for applications in which bandwidth limitations prohibit transmission of multiple audio channels, an alternative would be to transmit only one or two channels (denoted as *reference* channels or recordings in this work, *e.g.* the left and right signals in a traditional stereo recording) and reconstruct the remaining channels at the receiving end. The system proposed in this chapter provides a solution for reconstructing the channels of a specific recording from the reference channels and is particularly suitable for live concert hall performances. The proposed method is based on information of

the acoustics of a specific concert hall and the microphone locations with respect to the orchestra, information that can be extracted from the specific multichannel recording.

Before proceeding to the description of the method proposed, a brief outline of the basis of our approach is given. A number of microphones are used to capture several characteristics of the venue, resulting in an equal number of *stem recordings* (or *elements*). Fig. 5.1, provides an example of how microphones may be arranged in a recording venue in a multichannel recording. These recordings are then mixed and played back through a multichannel audio system that attempts to recreate the spatial realism of the recording venue. Our objective is to design a system based on available stem recordings that is able to recreate all of these recordings from the reference channels at the receiving end of a communications channel (thus, stem recordings are also referred to as *target* recordings here). The result would be a significant reduction in transmission requirements, while enabling mixing at the receiving end. Consequently, such a system would be suitable for completely resynthesizing any number of channels in the initial recording (*i.e.* no information needs to be transmitted about the target recordings other than the conversion parameters). This is different than what commercial systems accomplish today. In addition, the system proposed in this chapter is a structured representation of multichannel audio that lends itself to other possible applications such as multichannel audio synthesis which is briefly described later in this section. By examining the acoustical characteristics of the various stem recordings, the distinction of microphones is made into reverberant and spot microphones.

Spot microphones are microphones that are placed close to the sound source (*e.g.* G in Fig. 5.1). These microphones introduce a very challenging situation. Because the source of sound is not a point source but rather distributed such as in an orchestra, the recordings of these microphones depend largely on the instruments that are near the microphone and not so much on the acoustics of the hall. Synthesizing the recordings of these microphones, therefore, involves enhancing certain instruments and diminishing others, which in most cases overlap both in the time and frequency domains. The algorithm described here that focuses on this problem is based on spectral conversion (SC). The special case of percussive drum-like sounds is separately examined since these sounds are of impulsive nature and cannot be addressed by spectral conversion methods. These sounds are of particular interest however, since they greatly affect our perception of proximity to the orchestra.

Reverberant microphones are the microphones placed far from the sound source, for example C and D in Fig. 5.1. These microphones are treated separately as one category because they mainly capture reverberant information (that can be reproduced by the surround channels in a multichannel playback system). The recordings captured by these microphones can be synthesized by filtering the reference recordings through linear time-invariant (LTI) filters, designed using the methods that will be described in later sections of this chapter. Existing reverberation methods use a combination of comb and all-pass filters to effectively add reverberation to the existing monophonic or stereophonic signal. Our objective is to estimate the appropriate filters that capture the concert hall acoustical properties from a given set of stem microphone recordings. We

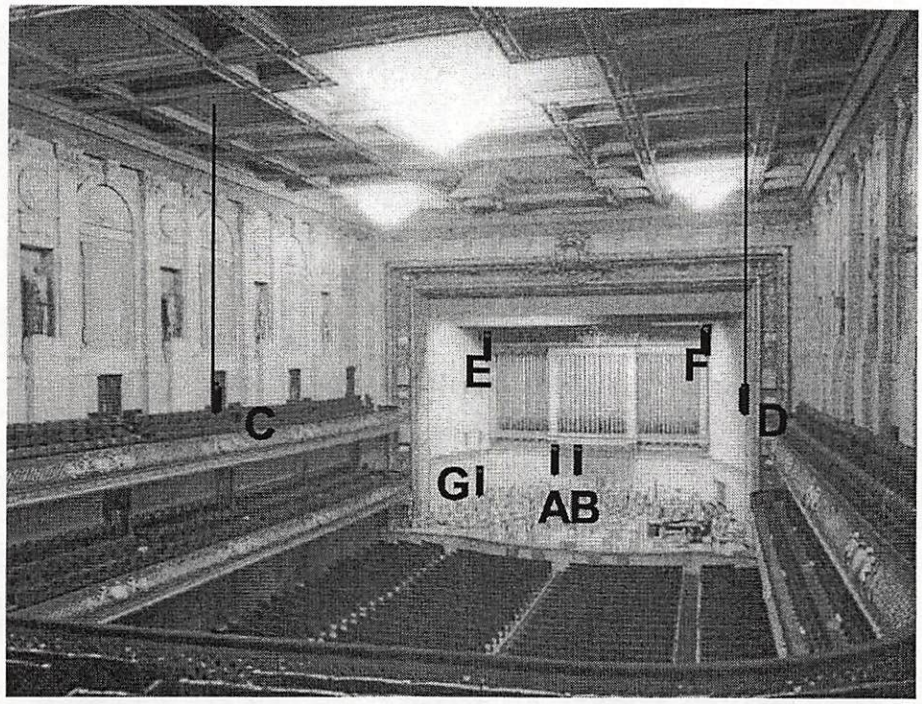

Figure 5.1: An example of how microphones may be arranged in a recording venue for a multichannel recording. In the virtual microphone resynthesis algorithm, microphones A and B are the main reference pair from which the remaining microphone signals can be derived. Virtual microphones C and D capture the hall reverberation, while virtual microphones E and F capture the reflections from the orchestra stage. Virtual microphone G can be used to capture individual instruments such as the tympani. These signals can then be mixed and played back through a multichannel audio system that recreates the spatial realism of a large hall.

describe an algorithm that is based on a spectral estimation approach and is particularly suitable for generating such filters for large venues with long reverberation times. Ideally, the resulting filter implements the spectral modification induced by the hall acoustics.

We have obtained such stem microphone recordings from two orchestra halls in the USA by placing microphones at various locations throughout the hall. By recording a performance with a total of sixteen microphones we then designed a system that *recreates*

these recordings (thus named *virtual microphone* recordings) from the main microphone pair. It should be noted that the methods proposed here intend to provide a solution for the problem of resynthesizing *existing* multichannel recordings from a smaller subset of these recordings. The problem of completely synthesizing multichannel recordings from stereophonic (or monophonic) recordings, thus greatly augmenting the listening experience, is not addressed here. The synthesis problem is a topic of related research to be discussed in the next chapter. However, it is important to distinguish the cases where these two problems (synthesis and resynthesis) differ. For reverberant microphones, since the result of our method is a group of LTI filters, both problems are addressed at the same time. The filters designed are capable of recreating the acoustic properties of the venue where the specific recordings took place. If these filters are applied to an arbitrary (non-reverberant) recording, the resulting signal will contain the venue characteristics at the particular microphone location. In such manner, it is possible to completely synthesize reverberant stem recordings and synthesize a multichannel recording. In contrast, this will not be possible for the stem microphone methods. As it will be clear later, the algorithms described here are based on the specific recordings that are available. The result is a group of spectral conversion functions that are designed by estimating the unknown parameters based on training data that are available from the target recordings. These functions cannot be applied to an arbitrary signal and produce meaningful results. This is an important issue when addressing the synthesis problem and will be the topic of Chapter 6.

The remainder of this chapter is organized as follows. In Section 5.3 the spot microphone resynthesis problem is addressed. Spectral conversion methods are described and applied to the problem in different subbands of the audio signal. The special case of percussive sounds is also examined. In Section 5.4 the reverberant microphone resynthesis problem is examined. The issue of defining an objective measure of the method's performance arises which is addressed by defining a normalized mutual information measure. Finally, a brief discussion of the results is given in Section 5.5 and possible directions for future research on the subject are proposed.

5.3 Spot Microphone Resynthesis

5.3.1 Spectral Conversion

The goal is to modify the short-term spectral properties of the reference audio signal in order to recreate the desired one. The short-term spectral properties are extracted by using a short sliding window with overlapping (resulting in a sequence of signal segments or frames). Each frame is modeled as an autoregressive (AR) filter excited by a residual signal. The AR filter coefficients are found by means of linear predictive analysis (LPC, [43]) and the residual signal is the result of inverse filtering the audio signal of the current frame by the AR filter. The LP coefficients are modified in a way to be described later in this section and the residual is filtered with the designed AR filter to produce the desired signal of the current frame. Finally, the desired response is synthesized from the designed frames using overlap-add techniques [41].

In order to obtain the desired response for each frame, an algorithm is required for converting the LP coefficients into the desired ones. Although the target coefficients in the application examined can be found by applying the same residual/LP analysis described (assuming that the reference and target waveforms are time-aligned), our intention is to design a mapping function based on the reference and target responses whose parameters will remain constant. The result will be a significant reduction of information as the target response can be reconstructed using the reference signal and this function.

Such a mapping function can be designed by following the approach of voice conversion algorithms [1, 95, 50]. The objective of voice conversion is to modify a speech waveform so that the context remains as is but appears to be spoken by a specific (target) speaker. Although the application is completely different, the approach followed is very suitable for our problem. In voice conversion pitch and time-scaling need to be considered, while in the application examined here this is not necessary. This is true since the reference and target waveforms come from the same excitation recorded with different microphones and the need is not to modify but to *enhance* the reference waveform. However, in both cases, there is the need to modify the short-term spectral properties of the waveform. The method to do that is briefly described next.

Assuming that a sequence $[\boldsymbol{x}_1\boldsymbol{x}_2\ldots\boldsymbol{x}_n]$ of reference spectral vectors (*e.g.* line spectral frequencies (LSF's), cepstral coefficients, *etc.*) is given, as well as the corresponding sequence of target spectral vectors $[\boldsymbol{y}_1\boldsymbol{y}_2\ldots\boldsymbol{y}_n]$ (training data from the reference and target recordings respectively), a function $\mathcal{F}(\cdot)$ can be designed which, when applied

to vector $\boldsymbol{x}_k$, produces a vector close in some sense to vector $\boldsymbol{y}_k$. Many algorithms have been described for designing this function (see [1, 95, 50, 4] and the references therein). Here the algorithms based on vector quantization (VQ, [1]) and Gaussian mixture models (GMM, [95, 50]) were implemented and compared.

Spectral Conversion based on VQ

Under this approach, the spectral vectors of the reference and target signals (training data) are vector quantized using the well-known modified K-means clustering algorithm (see for example [85] for details). Then, a histogram is created indicating the correspondences between the reference and target centroids. Finally, the function $\mathcal{F}$ is defined as the linear combination of the target centroids using the designed histogram as a weighting function. It is important to mention that in this case the spectral vectors were chosen to be the cepstral coefficients so that the distance measure used in clustering is the truncated cepstral distance.

Spectral Conversion based on GMM's

In this case, the assumption made is that the sequence of spectral vectors $\boldsymbol{x}_k$ is a realization of a random vector $\boldsymbol{x}$ with a probability density function (pdf) that can be modeled as a mixture of M multivariate Gaussian pdf's. Thus, the pdf of $\boldsymbol{x}$, $\mathrm{g}(\boldsymbol{x})$, can be written as

$$\mathrm{g}(\boldsymbol{x}) = \sum_{i=1}^{M} p(\omega_i)\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_i^x, \boldsymbol{\Sigma}_i^{xx}) \tag{5.1}$$

where, $\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the normal multivariate distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ and $p(\omega_i)$ is the prior probability of class ω_i. The parameters of the GMM, *i.e.* the mean vectors, covariance matrices and priors, can be estimated using the expectation maximization (EM) algorithm [87].

As already mentioned, the function $\mathcal{F}$ is designed so that the spectral vectors $\boldsymbol{y}_k$ and $\mathcal{F}(\boldsymbol{x}_k)$ are close in some sense. In [95], the function $\mathcal{F}$ is designed such that the error

$$\mathcal{E} = \sum_{k=1}^{n} \|\boldsymbol{y}_k - \mathcal{F}(\boldsymbol{x}_k)\|^2 \tag{5.2}$$

is minimized. Since this method is based on least-squares estimation, it will be denoted as the LSE method. This problem becomes possible to solve under the constraint that $\mathcal{F}$ is piecewise linear, *i.e.*

$$\mathcal{F}(\boldsymbol{x}_k) = \sum_{i=1}^{M} p(\omega_i|\boldsymbol{x}_k) \left[\boldsymbol{v}_i + \boldsymbol{\Gamma}_i \boldsymbol{\Sigma}_i^{xx^{-1}} (\boldsymbol{x}_k - \boldsymbol{\mu}_i^x)\right] \tag{5.3}$$

where the conditional probability that a given vector $\boldsymbol{x}_k$ belongs to class ω_i, $p(\omega_i|\boldsymbol{x}_k)$ can be computed by applying Bayes' theorem

$$p(\omega_i|\boldsymbol{x}_k) = \frac{p(\omega_i)\mathcal{N}(\boldsymbol{x}_k; \boldsymbol{\mu}_i^x, \boldsymbol{\Sigma}_i^{xx})}{\sum_{j=1}^{M} p(\omega_j)\mathcal{N}(\boldsymbol{x}_k; \boldsymbol{\mu}_j^x, \boldsymbol{\Sigma}_j^{xx})} \tag{5.4}$$

The unknown parameters ($\boldsymbol{v}_i$ and $\boldsymbol{\Gamma}_i$, $i = 1, \ldots, M$) can be found by minimizing (5.2) which reduces to solving a typical least-squares equation.

A different solution for function $\mathcal{F}$ results when a different function than (5.2) is minimized [50]. Assuming that $\boldsymbol{x}$ and $\boldsymbol{y}$ are jointly Gaussian for each class ω_i, then, in mean-squared sense, the optimal choice for the function $\mathcal{F}$ is

$$\begin{aligned} \mathcal{F}(\boldsymbol{x}_k) &= \mathrm{E}(\boldsymbol{y}|\boldsymbol{x}_k) \\ &= \sum_{i=1}^{M} p(\omega_i|\boldsymbol{x}_k) \left[\boldsymbol{\mu}_i^y + \boldsymbol{\Sigma}_i^{yx} {\boldsymbol{\Sigma}_i^{xx}}^{-1} (\boldsymbol{x}_k - \boldsymbol{\mu}_i^x)\right] \end{aligned} \tag{5.5}$$

where $\mathrm{E}(\cdot)$ denotes the expectation operator and the conditional probabilities $p(\omega_i|\boldsymbol{x}_k)$ are given again from (5.4). If the source and target vectors are concatenated, creating a new sequence of vectors $\boldsymbol{z}_k$ that are the realizations of the random vector $\boldsymbol{z} = [\boldsymbol{x}^T \boldsymbol{y}^T]^T$ (where ^{T} denotes transposition), then all the required parameters in the above equations can be found by estimating the GMM parameters of $\boldsymbol{z}$. Then,

$$\boldsymbol{\Sigma}_i^{zz} = \begin{bmatrix} \boldsymbol{\Sigma}_i^{xx} & \boldsymbol{\Sigma}_i^{xy} \\ \boldsymbol{\Sigma}_i^{yx} & \boldsymbol{\Sigma}_i^{yy} \end{bmatrix}, \boldsymbol{\mu}_i^z = \begin{bmatrix} \boldsymbol{\mu}_i^x \\ \boldsymbol{\mu}_i^y \end{bmatrix} \tag{5.6}$$

Once again, these parameters are estimated by the EM algorithm. Since this method estimates the desired function based on the joint density of $\boldsymbol{x}$ and $\boldsymbol{y}$, it will be referred to as the Joint Density Estimation (JDE) method.

5.3.2 Subband Processing

Audio signals contain information over a larger bandwidth than speech signals. The sampling rate for audio signals is usually 44.1 or 48 kHz compared to 16 kHz for speech. Moreover, since high acoustical quality for audio is essential, it is important to consider

the entire spectrum in detail. For these reasons, the decision to follow an analysis in subbands seems natural. Instead of warping the frequency spectrum using the Bark scale as is usual in speech analysis, the frequency spectrum was divided in subbands and each one was treated separately under the analysis presented in the previous section. Perfect reconstruction filter banks, based on wavelets [93], provide a solution with acceptable computational complexity as well as the appropriate, for audio signals, octave frequency division. The choice of filter bank was not a subject of investigation but steep transition from passband to stopband is desirable. The reason is that the short-term spectral envelope is modified separately for each band thus frequency overlapping between adjacent subbands would result in a distorted synthesized signal.

5.3.3 Transient Sounds Consideration

The spectral conversion methods described earlier will not produce the desired result in all cases. Transient sounds cannot be adequately processed by altering their spectral envelope and must be examined separately. An example of an analysis/synthesis model that treats transient sounds separately and is very suitable as an alternative to the subband-based residual/LP model that we employed, is described in [62]. It is suitable since it also models the audio signal in different bands, in each one as a sinusoidal/residual model [67, 90]. The sinusoidal parameters can be treated in the same manner as the LP coefficients during spectral conversion [16]. We are currently considering this model for improving the produced sound quality of our system. However, no structured model is proposed in [62] for transient sounds. In the remainder of this section, the special case of percussive sounds is addressed.

The case of percussive drum-like sounds is considered of particular importance. It is usual in multichannel recordings to place a microphone close to the tympani as drum-like sounds are considered perceptually important in recreating the acoustical environment of the recording venue. For percussive sounds, a similar model to the residual/LP model described here can be used [61] (see also [96, 64, 60]), but for the enhancement purposes investigated here, the emphasis is given to the residual instead of the LP parameters. The idea is to extract the residual of an instance of the particular percussive instrument from the recording of the microphone that captures this instrument and then recreate this channel from the reference channel by simply substituting the residual of all instances of this instrument with the extracted residual. As explained in [61], this residual corresponds to the interaction between the exciter and the resonating body of the instrument and lasts until the structure reaches a steady vibration. This signal characterizes the attack part of the sound and is independent of the frequencies and amplitudes of the harmonics of the produced sound (after the instrument has reached a steady vibration). Thus, it can be used for synthesizing different sounds by using an appropriate all-pole filter. This method proved to be quite successful and further details are given in the next section. The drawback of this approach is that a robust algorithm is required for identifying the particular instrument instances in the reference recording. A possible improvement of the proposed method would be to extract all instances of the instrument from the target response and use some clustering technique for choosing the residual that is more appropriate in the resynthesis stage. The reason is that the residual/LP model introduces modeling error which is larger in the spectral

Band Nr.	Frequency (kHz)		LPC Order	GMM Centroids
	Low	High		
1	0.0000	0.1723	4	4
2	0.1723	0.3446	4	4
3	0.3446	0.6891	8	8
4	0.6891	1.3782	16	16
5	1.3782	2.7563	32	16
6	2.7563	5.5125	32	16
7	5.5125	11.0250	32	16
8	11.0250	22.0500	32	16

Table 5.1: Parameters for the chorus microphone resynthesis example.

valleys of the AR spectrum; thus, better results would be obtained by using a residual which corresponds to an AR filter as close as possible to the resynthesis AR filter. However, this approach would again require robustly identifying all the instances of the instrument.

5.3.4 Resynthesis Performance

The three spectral conversion methods outlined in Section 5.3.1 were implemented and tested using a multichannel recording which we obtained as described in Section 5.2 of this chapter. The objective was to recreate the channel that mainly captured the chorus of the orchestra (residual processing for percussive sound resynthesis is also included). Acoustically, the primary emphasis was on the male and female voices. At the same time, it was clear that some instruments, inaudible in the target recording but particularly audible in the reference recording, needed to be attenuated. A database of about 10,000 spectral vectors for each spectral band was created so that only parts of the recording where the chorus is present are used, with the choice of spectral vectors

SC Method	Ceps. Distance		Centroids per Band
	Train	Test	
LSE	0.6451	0.7144	Table 5.1
JDE	0.6629	0.7445	Table 5.1
VQ	1.2903	1.3338	1024

Table 5.2: Normalized distances for LSE-, JDE- and VQ-based methods.

being the cepstral coefficients. Parts of the chorus recording were selected so that there were no segments of silence included. Results were evaluated through informal listening tests and through objective performance criteria. The spectral conversion methods were found to provide promising enhancement results. Formal listening tests are currently underway and will be available in the near future. For this work, objective test results were performed, which manifest that the spectral conversion methods can be used successfully for the enhancement purposes investigated here. The experimental conditions are given in Table 5.1. The number of octave bands used was 8, a choice that gives particular emphasis on the frequency band 0-5 kHz and at the same time does not impose excessive computational demands. The frequency range 0-5 kHz is particularly important for the specific case of chorus recording resynthesis since this is the frequency range where the human voice is mostly concentrated. For producing better results, the entire frequency range 0-20 kHz must be considered. The order of the LPC filter varied depending on the frequency detail of each band and for the same reason the number of centroids for each band was different.

In Table 5.2, the average quadratic cepstral distance (averaged over all vectors and all 8 bands) is given for each method, for the training data as well as for the data used for testing (9 sec. of music from the same recording). The cepstral distance is

normalized with the average quadratic distance between the reference and the target waveforms (*i.e.* without any conversion of the LPC parameters). The improvement is large for both the GMM-based algorithms, with the LSE algorithm being slightly better, for both the training and testing data. The VQ-based algorithm, in contrast, produced a deterioration in performance which was audible as well. This can be explained based on the fact that the GMM-based methods result in a conversion function which is continuous with respect to the spectral vectors. The VQ-based method, on the other hand, produces audible artifacts introduced by spectral discontinuities because the conversion is based on a limited number of existing spectral vectors. This is the reason why a large number of centroids was used for the VQ-based algorithm as seen in Table 5.2 compared to the number of centroids used for the GMM-based algorithms. However, the results were still unacceptable both from the objective and subjective perspectives.

The algorithm described in Section 5.3.3 considering the special case of percussive sound resynthesis was tested as well. Fig. 5.2 shows the time-frequency evolution of a tympani instance using the Choi-Williams distribution [17], a distribution that achieves the high resolution needed in such cases of impulsive signal nature. Fig. 5.2 clearly demonstrates the improvement in drum-like sound resynthesis. The impulsiveness of the signal at around samples 60-80 is observed in the desired response and verified in the synthesized waveform. The attack part is clearly enhanced, significantly adding naturalness in the audio signal, as our informal listening tests clearly demonstrated.

The methods described in this section can be used for synthesizing recordings of microphones that are placed close to the orchestra. Of importance in this case were the

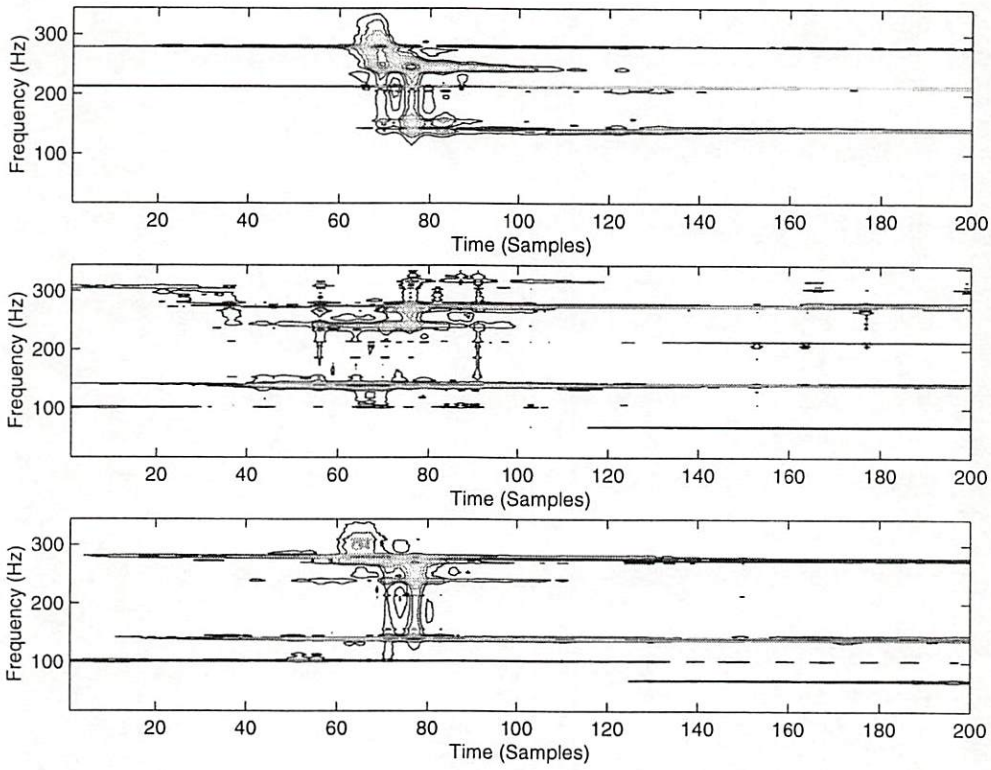

Figure 5.2: Choi-Williams distribution of the desired (top), reference (middle) and resynthesized (bottom) waveforms at the time points during a tympani strike (samples 60-80). This high resolution time-frequency analysis is necessary for understanding the evolution of the audio signal spectrum and identifying the correct approach for signal synthesis. The impulsiveness of the signal is observed in the desired response and verified in the resynthesized waveform.

short-term spectral properties of the audio signals. Thus, linear time-invariant filters were not suitable and the time-frequency properties of the waveforms had to be exploited in order to obtain a solution. In the next section, we focus on microphones placed far from the orchestra and thus contain mainly reverberant signals. As we demonstrate, the desired waveforms can be synthesized by taking advantage of the long-term spectral properties of the reference and the desired signals.

5.4 Reverberant Microphone Resynthesis

The problem of resynthesizing a virtual microphone signal from a signal recorded at a different position in the room can be described as follows. Given two processes s_1 and s_2, determine the optimal filter H that can be applied to s_1 (the reference microphone signal) so that the resulting process s_2' (the virtual microphone signal) is as close as possible to s_2. The optimality of the resulting filter H is based on how "close" s_2' is to s_2. For the case of audio signals, the distance between these two processes must be measured in a way that is psychoacoustically valid. We can treat this as a typical system identification problem. However, there are several unique aspects that need to be considered, the most important being that the physical system is characterized by a long impulse response. For a typical large symphony hall the reverberation time is approximately 2 sec., which would require a filter of more than 96000 taps to describe the reverberation process (for a typical sampling rate of 48 kHz).

5.4.1 IIR Filter Design

There are several possible approaches to the problem. One is to use classical estimation theoretic techniques such as least-squares or Wiener filtering based algorithms to estimate the hall environment with a long finite-duration impulse response (FIR) or infinite-duration impulse response (IIR) filter. Adaptive algorithms such as LMS [43] can provide an acceptable solution in such system identification problems while least-squares methods suffer prohibitive computational demands. For LMS the limitation lies

in the fact that the input and the output are non-stationary signals making its convergence quite slow. In addition, the required length of the filter is very large so such algorithms would prove to be inefficient for this problem. Although it is possible to prewhiten the input of the adaptive algorithm (see for example [43, 66] and references therein), so that convergence is improved, these algorithms still did not prove to be efficient for this problem.

An alternative to the aforementioned methods for treating system identification problems, is to use spectral estimation techniques based on the cross-spectrum [63]. These methods are divided into parametric and non-parametric. Non-parametric methods, based on averaging techniques such as the averaged periodogram (Welch spectral estimate) [7, 24, 104] are considered more appropriate for the case of long observations and for non-stationary conditions since no model is assumed for the observed data (a different approach based on the cross-spectrum which, instead of averaging, solves an overdetermined system of equations can be found in [91]). After the frequency response of the filter is estimated, an IIR filter can be designed based on that response. The advantage of this approach is that IIR filters are a more natural choice of modeling the physical system under consideration and can be expected to be very efficient in approximating the spectral properties of the recording venue. In addition an IIR filter would implement the desired frequency response with a significantly lower order compared to an FIR filter. Caution must, of course, be taken in order to ensure the stability of the filters.

To summarize, if we could define a power spectral density $S_{s_1}(\omega)$ for signal s_1 and $S_{s_2}(\omega)$ for signal s_2, then it would be possible to design filter $H(\omega)$ that can be applied to process s_1 resulting in process s_2', which is intended to be an estimate of s_2. The filter $H(\omega)$ can be estimated by means of spectral estimation techniques. Furthermore, if $S_{s_1}(\omega)$ is modeled by an all-pole approximation $|1/A_{p1}|^2$ and $S_{s_2}(\omega)$ similarly as $|1/A_{p2}|^2$ then $H = A_{p1}/A_{p2}$, if H is restricted to be the minimum phase spectral factor of $|H(\omega)|^2$. This results in a stable IIR filter that can be designed efficiently but is minimum phase. The analysis that follows provides the details for designing H.

The estimation of $H(\omega)$ is based on computing the cross-spectrum $S_{s_2 s_1}$ of signals s_2 and s_1 and the auto spectrum S_{s_1} of signal s_1. It is true that if these signals were stationary then

$$S_{s_2 s_1}(\omega) = H(\omega) S_{s_1}(\omega) \tag{5.7}$$

The difficulties arising in the design of filter H are due to the non-stationary nature of audio signals. This issue can be partly addressed if the signals are divided into segments short enough that can be considered of approximately stationary nature. It must be noted, however, that these segments must be large enough so that they can be considered long compared to the length of the impulse response that must be estimated, in order to avoid edge effects (as explained in [98], where a similar procedure is followed for the case of blind deconvolution for audio signal restoration).

For interval i, composed from M (real) samples $s_1^{(i)}(0), \ldots, s_1^{(i)}(M-1)$, the empirical transfer function estimate (ETFE, [63]) is computed as

$$\hat{H}^{(i)}(\omega) = \frac{S_2^{(i)}(\omega)}{S_1^{(i)}(\omega)} \tag{5.8}$$

where

$$S_1^{(i)}(\omega) = \sum_{n=0}^{M-1} s_1^{(i)}(n) e^{-j\omega n} \tag{5.9}$$

is the Fourier transform of the segment samples. This cannot be considered an accurate estimate of $H(\omega)$ though, since the filter $H^{(i)}(\omega)$ will be valid only for frequencies corresponding to the harmonics of segment i (under the valid assumption of quasi-periodic nature of the audio signal for each segment). An intuitive procedure would be to obtain the estimate of the spectral properties of the recording venue $\hat{H}(\omega)$ by averaging all the estimates available. Since the ETFE is the result of frequency division, it is apparent that in frequencies where $S_{s_1}(\omega)$ is close to zero, the ETFE would become unstable, so a more robust procedure would be to estimate H using a weighted average of the K segments available [63], *i.e.*

$$\hat{H}(\omega) = \frac{\sum_{i=0}^{K-1} \beta^{(i)}(\omega) H^{(i)}(\omega)}{\sum_{i=0}^{K-1} \beta^{(i)}(\omega)} \tag{5.10}$$

A sensible choice of weights would be

$$\beta^{(i)}(\omega) = |S_1^{(i)}(\omega)|^2 \tag{5.11}$$

It can be easily shown that estimating H under this approach is equivalent to estimating the auto-spectrum of s_1 and the cross-spectrum of s_2 and s_1 using the Cooley-Tukey spectral estimate [24] (in essence Welch spectral estimation with rectangular windowing of the data and no overlapping). In other words, defining the power spectrum estimate under the Cooley-Tukey procedure as

$$S_{s_1}^{CT}(\omega) = \frac{1}{K}\sum_{i=0}^{K-1} |S_1^{(i)}(\omega)|^2 \tag{5.12}$$

where $S(\omega)$ is defined as previously, and a similar expression for the cross-spectrum

$$S_{s_2 s_1}^{CT}(\omega) = \frac{1}{K}\sum_{i=0}^{K-1} S_2^{(i)}(\omega) S_1^{(i)*}(\omega) \tag{5.13}$$

then, it holds that

$$\hat{H}(\omega) = \frac{S_{s_2 s_1}^{CT}(\omega)}{S_{s_1}^{CT}(\omega)} \tag{5.14}$$

which is analogous to (5.7). Thus, for a stationary signal, the averaging of the estimated filters is justifiable. A window can additionally be used to further smooth the spectra.

The method described is meaningful for the special case of audio signals, despite their non-stationarity. It is well known that the averaged periodogram provides a smoothed version of the periodogram. Considering that it is true even for non-stationary (but of finite length) signals that

$$S_2(\omega) S_1^*(\omega) = H(\omega)|S_1(\omega)|^2 \tag{5.15}$$

then averaging in essence smoothes the frequency response of H. This is justifiable since it is true that a non-smoothed H will contain details that are of no acoustical significance. Further smoothing can yield a lower order IIR filter, by taking advantage of AR modeling. Considering signal s_1, the inverse Fourier transform of its power spectrum $S_{s_1}(\omega)$ derived as described earlier will yield the sequence $r_{s_1}(m)$. If this sequence is viewed as the autocorrelation of s_1 and samples $r_{s_1}(0), \cdots, r_{s_1}(p+1)$ are inserted in the Wiener-Hopf equations for linear prediction (with the AR order p being significantly smaller than the number of samples of each block M, for smoothing the spectra)

$$\begin{bmatrix} r_{s_1}(0) & r_{s_1}(1) & \cdots & r_{s_1}(p-1) \\ r_{s_1}(1) & r_{s_1}(0) & \cdots & r_{s_1}(p-2) \\ \vdots & \vdots & \ddots & \vdots \\ r_{s_1}(p-1) & r_{s_1}(p-2) & \cdots & r_{s_1}(0) \end{bmatrix} \begin{bmatrix} a_{p1}(1) \\ a_{p1}(2) \\ \vdots \\ a_{p1}(p) \end{bmatrix} = \begin{bmatrix} r_{s_1}(1) \\ r_{s_1}(2) \\ \vdots \\ r_{s_1}(p) \end{bmatrix} \tag{5.16}$$

then, the coefficients $a_{p1}(i)$ result in an approximation of $S_{s_1}(\omega)$ (omitting the constant gain term which is not of importance in this case)

$$S_{s_1}(\omega) = \left| \frac{1}{A_{p1}(\omega)} \right|^2 \tag{5.17}$$

where

$$A_{p1}(\omega) = 1 + \sum_{l=1}^{p} a_{p1}(l) e^{-j\omega l} \tag{5.18}$$

A similar expression holds for $S_{s_2}(\omega)$. S_{s_1} and S_{s_2} can be computed as in (5.12). Using the fact that

$$S_{s_2}(\omega) = |H(\omega)|^2 S_{s_1}(\omega) \tag{5.19}$$

and restricting H to be minimum phase, we find from the spectral factorization of (5.19) a solution for H is

$$H(\omega) = \frac{A_{p1}(\omega)}{A_{p2}(\omega)} \tag{5.20}$$

Filter H can be designed very efficiently even for very large filter orders following this method since equation (5.16) can be solved using the Levinson-Durbin recursion. This filter will be IIR and stable.

A problem with the aforementioned design method is that the filter H is restricted to be of minimum phase. It is of interest to mention that in our experiments the minimum phase assumption proved to be perceptually acceptable. This can be possibly attributed to the fact that if the minimum phase filter H captures a significant part of the hall reverberation, then the listener's ear will be less sensitive to the phase distortion [86]. It is not possible, however, to generalize this observation and the performance of this last step in the filter design will possibly vary depending on the particular characteristics of the venue captured in the multichannel recording.

5.4.2 Mutual Information as a Spectral Distortion Measure

As previously mentioned, we need to apply the above procedure in blocks of data of the two processes s_1 and s_2. In our experiments, we chose signal block lengths of 100,000 samples (long blocks of data are required due to the long the reverberation time of

the hall as explained earlier). We then experimented with various orders of filters A_{p1} and A_{p2}. As expected, relatively high orders were required to reproduce s_2 from s_1 with an acceptable error between s_2' (the resynthesized process) and s_2 (the target recording). The performance was assessed through blind A/B/X listening evaluation. An order of 10,000 coefficients for both the numerator and denominator of H resulted in an error between the original and resynthesized signals that was not detectable by listeners. We also evaluated the performance of the filter by resynthesizing blocks from a part of the signal other than the one that was used for designing the filter. Again, the A/B/X evaluation showed that for orders higher than 10,000 the resynthesized signal was indistinguishable from the original. Although such high order filters are impractical for real-time applications, the performance of our method is an indication that the model is valid and therefore motivating us to further investigate filter optimization. This method can be used for off-line applications such as remastering of old recordings. A real-time version was also implemented using the Lake DSP Huron digital audio convolution workstation. With this system we are able to resynthesize 12 virtual microphone stem recordings from a monophonic or stereophonic compact disc (CD) in real time.

To obtain an objective measure of the performance it is necessary to derive a mathematical measure of the distance between the resynthesized and the original processes. The difficulty in defining such a measure is that it must also be psychoacoustically valid. This problem has been addressed in speech processing where measures such as the log spectral distance and the Itakura-Saito distance are used [49]. In our case, we need to compare the spectral characteristics of long sequences with spectra that contain

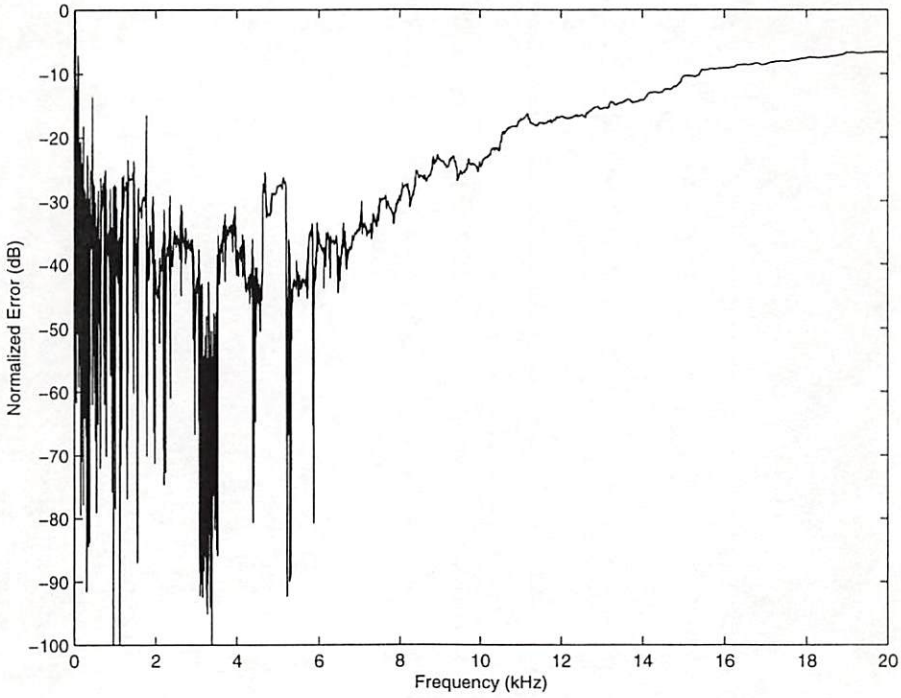

Figure 5.3: Normalized error between original and resynthesized microphone signals as a function of frequency.

a large number of peaks and dips that are narrow enough to be imperceptible to the human ear. In other words, the focus is on the long-term spectral properties of the audio signals, while spectral distortion measures have been developed for comparing the short-term spectral properties of signals. To overcome comparison inaccuracies that would be mathematical rather than psychoacoustical in nature, we chose to perform 1/3 octave smoothing [72] and compare the resulting smoothed spectral cues. The results are shown in Fig. 5.3 in which we compare the spectra of the original (measured) microphone signal and the resynthesized signal. The two spectra are practically indistinguishable below 10 kHz. Although the error increases at higher frequencies, the

listening evaluations show that this is not perceptually significant. One problem that was encountered while comparing the 1/3 octave smoothed spectra was the fact that the average error was not reduced with increasing filter order as rapidly as the results of the listening tests suggested. To address this inconsistency we experimented with various distortion measures.

These measures included the RMS log spectral distance, the truncated cepstral distance, and the Itakura distance (for a description of all these measures see for example [85]). The results, however, were still not in line with what the listening evaluations indicated. This led us to a measure that is commonly used in pattern comparison and is known as the mutual information (see for example [27]). By definition, the mutual information of two random variables X and Y with joint probability density function (pdf) $p(x, y)$ and marginal pdf's $p(x)$ and $p(y)$ is the relative entropy between the joint distribution and the product distribution, *i.e.*

$$I(X;Y) = \sum_{x\in\mathcal{X}} \sum_{y\in\mathcal{Y}} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \tag{5.21}$$

It is easy to prove that

$$\begin{aligned} I(X;Y) &= H(X) - H(X|Y) && (5.22) \\ &= H(Y) - H(Y|X) && (5.23) \end{aligned}$$

and also

$$I(X;Y) = H(X) + H(Y) - H(X,Y) \tag{5.24}$$

where $H(X)$ is the entropy of X

$$H(X) = -\sum_{x \in \mathcal{X}} p(x) \log p(x) \tag{5.25}$$

similarly, $H(Y)$ is the entropy of Y. $H(X|Y)$ is the conditional entropy defined as

$$\begin{aligned} H(X|Y) &= \sum_{y \in \mathcal{Y}} p(y) H(X|Y = y) && (5.26) \\ &= -\sum_{y \in \mathcal{Y}} p(y) \sum_{x \in \mathcal{X}} p(x|y) \log p(x|y) && (5.27) \end{aligned}$$

while $H(X, Y)$ is the joint entropy defined as

$$H(X, Y) = -\sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log p(x, y) \tag{5.28}$$

The mutual information is always positive. Since our interest is in comparing two vectors X and Y with Y being the desired response, it is useful to use a modified definition for the mutual information, the Normalized Mutual Information (NMI) $I_N(X; Y)$ which can be defined as

$$\begin{aligned} I_N(X; Y) &= \frac{H(Y) - H(Y|X)}{H(Y)} && (5.29) \\ &= \frac{I(X; Y)}{H(Y)} && (5.30) \end{aligned}$$

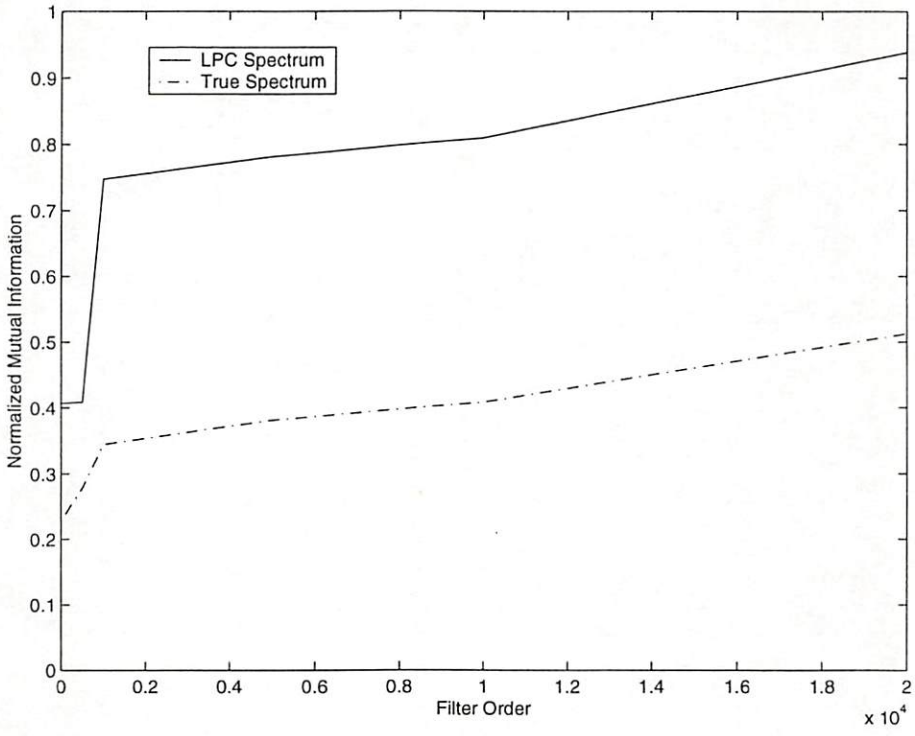

Figure 5.4: Normalized Mutual Information between original and resynthesized microphone signals as a function of filter order.

This version of the mutual information is mentioned in [27, p. 47] and has been applied in many applications as an optimization measure (*e.g.* radar remote sensing applications [101]). Obviously,

$$0 \leq I_N(X;Y) \leq 1$$

The NMI obtains its minimum value when X and Y are statistically independent and its maximum values when $X = Y$. The NMI does not constitute a metric since it lacks symmetry. On the other hand, the NMI is invariant to amplitude differences [92], which is a very important property especially for comparing audio waveforms.

The spectra of the original and the resynthesized responses were compared using the NMI for various filter orders and the results are depicted in Fig. 5.4. The NMI increases with filter order both when considering the raw spectra, as well as when we used the spectra that were smoothed using AR modeling (spectral envelope by all-pole modeling with Linear Predictive coefficients). We believe that the NMI calculated using the smoothed spectra is the measure that closely approximates the results we achieved from the listening tests. As can be seen from the figure, the NMI for a filter order of 20,000 is 0.9386 (*i.e.*, close to unity which corresponds to indistinguishable similarity) for the LPC spectra while the NMI for the same order but for the raw spectra is 0.5124. Furthermore, the fact that both the raw and smoothed NMI measures increase monotonically in the same fashion indicates that the smoothing is valid since it only reduces the "distance" between the two waveforms in a proportionate way for all the resynthesized waveforms (order 0 in the diagram corresponds to no filtering – it is the distance between the original and the reference waveforms).

5.5 Conclusions

Multichannel audio resynthesis is a new and important application that allows transmission of only one or two channels of multichannel audio and resynthesis of the remaining channels at the receiving end. It offers the advantage that the stem microphone recordings can be resynthesized at the receiving end, which makes this system suitable for many professional applications and, at the same time, poses no restrictions on the number of channels of the initial multichannel recording. The distinction was made of the

methods employed, depending on the location of the "virtual" microphones, namely spot and reverberant microphones. Reverberant microphones are those that are placed at some distance from the sound source (*e.g.* the orchestra) and therefore, contain more reverberation. On the other hand, spot microphones are located close to individual sources (*e.g.*, near a particular musical instrument). This is a completely different problem because placing such microphones near individual sources with varying spectral characteristics results in signals whose frequency content will depend highly on the microphone positions.

Spot microphones were treated separately by applying spectral conversion techniques for altering the short-term spectral properties of the reference audio signals. Spectral conversion algorithms that have been used successfully for voice conversion can be adopted for the task of multichannel audio resynthesis quite favorably. Three of the most common spectral conversion methods have been compared and our objective results, in accordance with our informal listening tests, have indicated that GMM-based spectral conversion can produce extremely successful results. Residual signal enhancement was also found to be essential for the special case of percussive sound resynthesis.

For the reverberant microphone recordings, we have described a method for resynthesizing the desired audio signals, based on spectral estimation techniques. The emphasis in this case is on the long-term spectral properties of the signals since the reverberation process is considered to be long in duration (*e.g.* 2 seconds for large concert halls). An IIR filtering solution was proposed for addressing the long reverberation-time problem,

with associated long impulse responses for the filters to be designed. The issue of objectively estimating the performance of our methods arose, which was treated by proposing the normalized mutual information as a measure of spectral distance that was found to be very suitable for comparing the long-term spectral properties of audio signals. It should be noted that the IIR filters designed are of high order thus not suitable for real-time applications.

Chapter 6

Maximum Likelihood Parameter Adaptation for Multichannel Audio Synthesis

6.1 Overview

Multichannel audio can immerse a group of listeners in a seamless aural environment. Previously, we proposed a system capable of synthesizing the multiple channels of a virtual multichannel recording from a smaller set of reference recordings. This problem was termed multichannel audio *resynthesis* and the application was to reduce the excessive transmission requirements of multichannel audio. In this chapter, we address the more general problem of multichannel audio *synthesis*, *i.e.* how to completely synthesize a multichannel audio recording from a specific stereophonic or monophonic recording, which would significantly enhance the recording's acoustic impression. We approach

this problem by extending the model employed for the resynthesis problem. This is accomplished by adapting the resynthesis conversion parameters to the statistical properties of the recording that we wish to enhance. This parameter adaptation is similar to the task adaptation employed in speech recognition, when a specific model is applied to a different environment (speaker, language or channel). One particular approach to this problem is shown here to be quite advantageous towards solving the multichannel audio synthesis problem as well.

6.2 Introduction

The advantages of multichannel audio over conventional stereophonic sound are well-known and have been mentioned repeatedly in this work. However, several key issues must be addressed. Multichannel audio imposes excessive requirements to the transmission medium. A system we proposed in the previous chapter, attempted to address this issue by offering the alternative to synthesize the multiple channels of a multichannel recording from a smaller set of signals (denoted as *reference* channels or recordings in this work, *e.g.* the left and right channels in a traditional stereophonic recording). The solution, termed multichannel audio *resynthesis*, focused on the problem of enhancing a concert hall recording and divided the problem in two different parts, depending on the characteristics of the recording to be synthesized. The approach followed is next briefly reviewed, since in this chapter these methods are extended to address the synthesis problem. Given the microphone recordings from several locations in a venue (*stem* recordings, see Fig. 5.1 for an example of how microphones may be arranged in

a recording venue for a multichannel recording), our objective was to design a system that can resynthesize these recordings from the reference recordings. For this reason, stem recordings are also referred to as *target* recordings in this work. As explained in Chapter 5, we have obtained such stem microphone recordings from two orchestra halls in the USA by placing microphones at various locations throughout the hall. By recording a performance with a total of sixteen microphones, we then designed a system that *recreates* these recordings (thus named *virtual microphone* recordings) from the main microphone pair. These resynthesized stem recordings are then mixed in order to produce the final multichannel audio recording. The distinction of the recordings is made depending on the location of the microphone in the venue, thus resulting in two different categories, namely *reverberant* and *spot* microphone recordings.

Reverberant microphones are the microphones placed far from the sound source, for example C and D in Fig. 5.1. These microphones are treated separately as one category because they mainly capture reverberant information (that can be reproduced by the surround channels in a multichannel playback system). For simulating recordings of such microphones, infinite impulse response (IIR) filters were designed from existing multichannel recordings made in a particular concert hall [73]. Our objective was to estimate the appropriate filters that capture the concert hall acoustical properties from a given set of stem microphone recordings. These IIR filters were shown in Chapter 5 to be capable of recreating the acoustical properties of the venue at specific locations.

Spot microphones are microphones that are placed close to the sound source (*e.g.* G in Fig. 5.1). Synthesizing the recordings of these microphones, therefore, involves enhancing certain instruments and diminishing others, which in most cases overlap both in the time and frequency domains. The algorithm described in the previous chapter and in [74], focuses on this problem and is based on spectral conversion.

In this chapter, we address the more general problem of multichannel audio *synthesis*. The goal is to convert existing stereophonic or monophonic recordings into multichannel. The significance of this application is evident since, although there are consumer media that allow the delivery of multiple channels of audio, only a limited set of multichannel music recordings have been made to-date. The same approach is followed as in the resynthesis problem. Based on existing multichannel recordings, we decide which microphone locations must be synthesized. For reverberant microphones, the filters designed for the resynthesis problem can be readily applied to arbitrary recordings. Their time-invariant nature offers the advantage that these filters can be applied to any recording although having been designed based on a specific recording. In contrast, the time-varying nature of the methods designed for spot microphone resynthesis, prohibits us from applying them in an arbitrary recording. This is the problem that we address here.

The block diagram of Fig. 6.1 can serve as a guide to the methods examined in this chapter. The part of the diagram to the left of the dotted line corresponds to an existing multimicrophone recording. Multichannel audio resynthesis allows us to reconstruct the stem recordings (target channels) from the reference channel. The part of the

diagram to the right of the dotted line, corresponds to multichannel audio synthesis, which is used to fully synthesize stem recordings from the reference channel of a stereo recording. Our approach is to take advantage of the resynthesis parameters that have been derived based on an existing target channel. In order to achieve that, the stereo and multimicrophone recordings are related with the GMM constrained estimation method that is analyzed later in this chapter. The adaptation assumption is also needed that relates the (unavailable) target response of the stereo recording with the target response of the multimicrophone recording.

The remainder of this chapter is organized as follows. In Section 6.3 spot microphone resynthesis is revisited, since it is an inherent part of the synthesis problem. Here, the implementation of the GMM-based algorithms of the previous chapter is examined. More specifically, the focus is on the diagonal implementation of the LSE and JDE algorithms, *i.e.* when all conversion matrices are restricted to be diagonal. Diagonal implementations are of significance when combined with model adaptation techniques for addressing the synthesis problem, as explained later in this chapter. In Section 6.4, we test the effectiveness of diagonal conversion as compared with full conversion (*i.e.* with no structural restrictions) that was examined in Chapter 5. Model adaptation schemes are discussed in Section 6.5, and the performance of these methods is examined in Section 6.6. Finally, a brief discussion of the results is given in Section 6.7 and possible directions for future research are proposed.

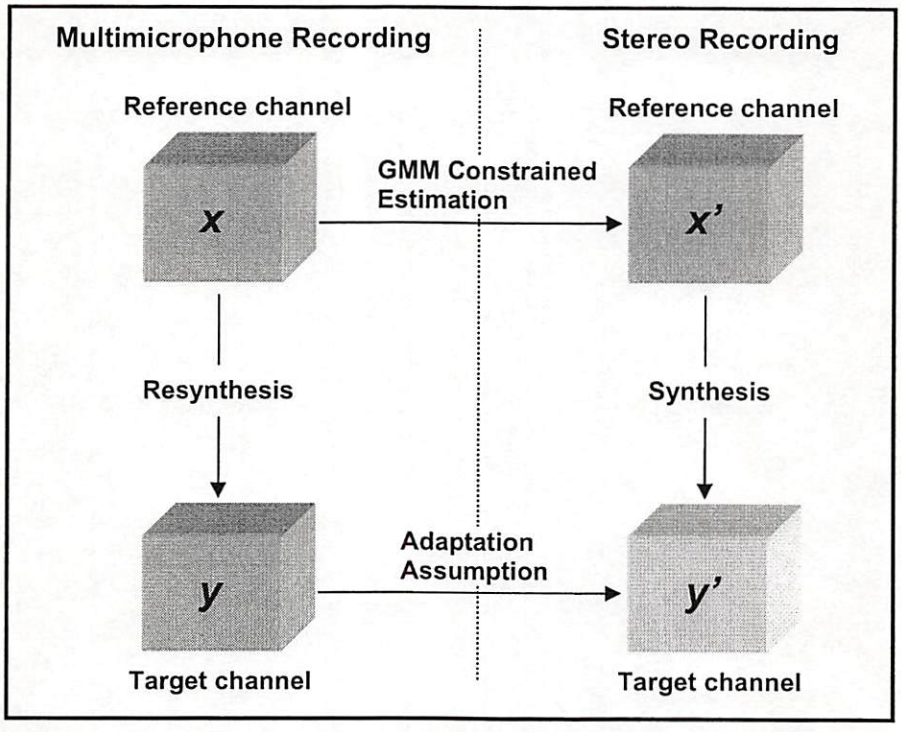

Figure 6.1: Block diagram outlining multichannel audio resynthesis and synthesis. Resynthesis corresponds to existing multichannel audio recordings while synthesis corresponds to stereo recordings. The objective of resynthesis is to recreate the multiple channels of the recording (target channels) from a smaller set of reference channels. The objective of synthesis is to completely synthesize target channels from one or two reference channels, thus converting the stereo recording for multichannel rendering. Resynthesis parameters can be used for the synthesis task, by adapting them through GMM constrained estimation and the adaptation assumption explained in the text.

6.3 Spot Microphone Resynthesis Revisited

6.3.1 Spectral Conversion

As explained in Chapter 5, the objective of spot microphone resnythesis is to modify the short-term spectral properties of the reference audio signal in order to recreate the desired signal. By analyzing these recordings with the residual/LP model, we obtain a

sequence $[\boldsymbol{x}_1\boldsymbol{x}_2\ldots\boldsymbol{x}_n]$ of reference spectral vectors (*e.g.* line spectral frequencies (LSF's), cepstral coefficients, *etc.*), as well as the corresponding sequence of target spectral vectors $[\boldsymbol{y}_1\boldsymbol{y}_2\ldots\boldsymbol{y}_n]$ (training data from the reference and target recordings respectively). It was then that it is possible to design a function $\mathcal{F}(\cdot)$, which, when applied to vector $\boldsymbol{x}_k$, produces a vector close in some sense to vector $\boldsymbol{y}_k$. GMM-based algorithms, namely the LSE and JDE methods, were found to favorably perform this task. These methods are now briefly reviewed for the convenience of the reader, since they form the basis for successfully addressing the synthesis task as well. In the following two sections, we examine some implementation issues of these algorithms, which are implicitly related with their application to multichannel audio synthesis. The synthesis problem is then addressed in Sections 6.5 and 6.6.

Spectral Conversion based on GMM's

The conversion function $\mathcal{F}$ is designed so that the spectral vectors $\boldsymbol{y}_k$ and $\mathcal{F}(\boldsymbol{x}_k)$ are close in some sense. In the LSE case [95], the function $\mathcal{F}$ is designed such that the error

$$\mathcal{E} = \sum_{k=1}^{n} \|\boldsymbol{y}_k - \mathcal{F}(\boldsymbol{x}_k)\|^2 \tag{6.1}$$

is minimized. For LSE, $\mathcal{F}$ is assumed to be piecewise linear, *i.e.*

$$\mathcal{F}(\boldsymbol{x}_k) = \sum_{i=1}^{M} p(\omega_i|\boldsymbol{x}_k)\left[\boldsymbol{v}_i + \boldsymbol{\Gamma}_i\boldsymbol{\Sigma}_i^{xx^{-1}}(\boldsymbol{x}_k - \boldsymbol{\mu}_i^x)\right] \tag{6.2}$$

where the conditional probability that a given vector $\boldsymbol{x}_k$ belongs to class ω_i, $p(\omega_i|\boldsymbol{x}_k)$ can be computed by applying Bayes' theorem

$$p(\omega_i|\boldsymbol{x}_k) = \frac{p(\omega_i)\mathcal{N}(\boldsymbol{x}_k; \boldsymbol{\mu}_i^x, \boldsymbol{\Sigma}_i^{xx})}{\sum_{j=1}^{M} p(\omega_j)\mathcal{N}(\boldsymbol{x}_k; \boldsymbol{\mu}_j^x, \boldsymbol{\Sigma}_j^{xx})} \tag{6.3}$$

The parameters of the GMM, *i.e.* the mean vectors, covariance matrices and priors, can be estimated using the expectation maximization (EM) algorithm. The remaining unknown parameters ($\boldsymbol{v}_i$ and $\boldsymbol{\Gamma}_i$, $i = 1, \ldots, M$) can be found by minimizing (6.1) which reduces to solving a typical least-squares equation.

For the JDE method [50], it is assumed that $\boldsymbol{x}$ and $\boldsymbol{y}$ are jointly Gaussian for each class ω_i. Then, in mean-squared sense, the optimal choice for the function $\mathcal{F}$ is

$$\begin{aligned} \mathcal{F}(\boldsymbol{x}_k) &= \mathrm{E}(\boldsymbol{y}|\boldsymbol{x}_k) \\ &= \sum_{i=1}^{M} p(\omega_i|\boldsymbol{x}_k) \left[\boldsymbol{\mu}_i^y + \boldsymbol{\Sigma}_i^{yx}\boldsymbol{\Sigma}_i^{xx^{-1}}(\boldsymbol{x}_k - \boldsymbol{\mu}_i^x)\right] \end{aligned} \tag{6.4}$$

where $\mathrm{E}(\cdot)$ denotes the expectation operator and the conditional probabilities $p(\omega_i|\boldsymbol{x}_k)$ are given again from (6.3). In essence, the JDE method assumes a GMM for the random vector $\boldsymbol{z} = [\boldsymbol{x}^T\boldsymbol{y}^T]^T$, where the random vector $\boldsymbol{z}$ corresponds to the sequence of concatenated vectors, $\boldsymbol{z}_k = [\boldsymbol{x}^T\boldsymbol{y}^T]^T$. The parameters of the GMM of $\boldsymbol{z}$ are

$$\boldsymbol{\Sigma}_i^{zz} = \begin{bmatrix} \boldsymbol{\Sigma}_i^{xx} & \boldsymbol{\Sigma}_i^{xy} \\ \boldsymbol{\Sigma}_i^{yx} & \boldsymbol{\Sigma}_i^{yy} \end{bmatrix}, \boldsymbol{\mu}_i^z = \begin{bmatrix} \boldsymbol{\mu}_i^x \\ \boldsymbol{\mu}_i^y \end{bmatrix} \tag{6.5}$$

Once again, these parameters are estimated by the EM algorithm.

6.3.2 Diagonal Implementation

The GMM-based LSE spectral conversion algorithm can be implemented with the covariance matrix having no structural restrictions or restricted to be diagonal [95], denoted as full and diagonal conversion respectively. Full conversion is of prohibitive computational complexity when combined with the adaptation algorithm for the synthesis problem examined in the second part of this chapter. As explained in [31, 30], the adaptation methods described are less computationally demanding when applied to GMM's with diagonal covariance matrices. Thus, it was apparent that it would be more efficient to combine these methods with the diagonal conversion algorithm of [95] for LSE and the diagonal conversion for JDE implemented in this chapter, as explained next.

It is important to note that the covariance matrix for the JDE method cannot be diagonal because this method is based on the cross-covariance of $\boldsymbol{x}$ and $\boldsymbol{y}$ which is found from (6.5). This will be zero if the covariance of $\boldsymbol{z}$ is diagonal. In order to obtain the same structure as in the diagonal LSE conversion, we must restrict the matrices $\boldsymbol{\Sigma}_i^{xx}$, $\boldsymbol{\Sigma}_i^{yy}$, $\boldsymbol{\Sigma}_i^{xy}$, and $\boldsymbol{\Sigma}_i^{yx}$ in (6.5) to be diagonal. For achieving this restriction, we slightly modified the EM algorithm, with the most noteworthy modification being that of obtaining the inverse of $\boldsymbol{\Sigma}_i^{zz}$ by taking advantage of its structure. It is very easy to show [69], that the inverse of $\boldsymbol{\Sigma}_i^{zz}$ will be

$$\boldsymbol{\Sigma}_i^{zz^{-1}} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{C} \end{bmatrix} \tag{6.6}$$

where

$$\begin{aligned}
\mathbf{A} &= \left(\Sigma_i^{xx} - \Sigma_i^{xy}\Sigma_i^{yy^{-1}}\Sigma_i^{yx}\right)^{-1} \\
&= \Sigma_i^{xx^{-1}} + \Sigma_i^{xx^{-1}}\Sigma_i^{xy}\mathbf{C}\Sigma_i^{yx}\Sigma_i^{xx^{-1}} \\
\mathbf{B} &= -\mathbf{A}\Sigma_i^{xy}\Sigma_i^{yy^{-1}} = -\Sigma_i^{xx^{-1}}\Sigma_i^{xy}\mathbf{C} \\
\mathbf{C} &= \left(\Sigma_i^{yy} - \Sigma_i^{yx}\Sigma_i^{xx^{-1}}\Sigma_i^{xy}\right)^{-1} \\
&= \Sigma_i^{yy^{-1}} + \Sigma_i^{yy^{-1}}\Sigma_i^{yx}\mathbf{A}\Sigma_i^{xy}\Sigma_i^{yy^{-1}}
\end{aligned} \tag{6.7}$$

In the above equations, all matrices, thus their products, sums, and differences are diagonal, so the inversions will be of very low computational demands. Based on this structure for the inverse of $\mathbf{\Sigma}_i^{zz}$, the joint pdf of $\boldsymbol{x}$ and $\boldsymbol{y}$ can be written as

$$\mathrm{g}(\boldsymbol{x},\boldsymbol{y}) = \frac{\exp\left(-\frac{1}{2}\left(\boldsymbol{x}^T\mathbf{A}\boldsymbol{x} + \boldsymbol{y}^T\mathbf{C}\boldsymbol{y} + 2\boldsymbol{x}^T\mathbf{B}\boldsymbol{y}\right)\right)}{(2\pi)^K\sqrt{|\mathbf{\Sigma}_i^{zz}|}} \tag{6.8}$$

K being the dimensionality of $\boldsymbol{x}$, and the determinant of $\mathbf{\Sigma}_i^{zz}$ equals

$$|\mathbf{\Sigma}_i^{zz}| = |\mathbf{\Sigma}_i^{yy}||\mathbf{\Sigma}_i^{xx} - \mathbf{\Sigma}_i^{xy}\mathbf{\Sigma}_i^{yy^{-1}}\mathbf{\Sigma}_i^{yx}| \tag{6.9}$$

6.3.3 Subband Processing

For the reasons explained in the previous chapter, the spectral conversion algorithms were implemented independently in different subbands. This is also the case for the synthesis algorithms, as explained in Section 6.6.

Band Nr.	Frequency (kHz)		LPC Order	Mixtures	
	Low	High		Full	Diag
1	0.0000	0.1723	4	4	8
2	0.1723	0.3446	4	4	8
3	0.3446	0.6891	8	8	16
4	0.6891	1.3782	16	16	32
5	1.3782	2.7563	32	16	64
6	2.7563	5.5125	32	16	64
7	5.5125	11.0250	32	16	64
8	11.0250	22.0500	32	16	64

Table 6.1: Parameters for the chorus microphone **resynthesis** example (full and diagonal conversion).

6.3.4 Transient Sounds Consideration

Transient sounds resynthesis was considered in the previous chapter. Please note that for the synthesis case the methods described must be modified, since no exact desired response (thus excitation signal) will be available. In Section 6.6.2 of this chapter, transient sound synthesis for percussive drum-like sounds is considered.

6.4 Diagonal Resynthesis Performance

The two GMM spectral conversion methods outlined in Section 6.3.1 (LSE and JDE) were implemented and tested using a multichannel recording, in the same manner as in Section 5.3.4 of the previous chapter. The results for full conversion are repeated here for reference, since the objective is to test the performance of diagonal conversion compared to full conversion. Informal listening tests were conducted to validate the objective performance criteria. Diagonal and full conversion were found to produce comparable results. The experimental conditions are given in Table 6.1.

SC Method	Covari-ance	Ceps. Distance		Centroids per Band
		Train	Test	
LSE	Full	0.6451	0.7144	Table 6.1
	Diag	0.5918	0.7460	Table 6.1
JDE	Full	0.6629	0.7445	Table 6.1
	Diag	0.6524	0.7508	Table 6.1

Table 6.2: Normalized distances for LSE- and JDE-based methods, for full and diagonal conversion.

In Table 6.2, the average quadratic cepstral distance (averaged over all vectors and all 8 bands) is given for each method, for the training data as well as for the data used for testing (9 sec. of music from the same recording, the same data used in Section 5.3.4). The two cases tested were the JDE and LSE spectral conversion algorithms with full and diagonal covariance matrices, as explained in Section 6.3.2. The difference lies in the fact that in the second case, the covariance matrix for all Gaussians is restricted to be diagonal. This restriction provides a more efficient conversion algorithm in terms of computational requirements, but at the same time requires more GMM components for producing comparable results with full conversion. This is evident from Table 6.2. However, diagonal conversion greatly simplifies the synthesis implementation and, consequently, will be the method chosen for this task.

The algorithms described in this section can be used for resynthesizing recordings of microphones that are placed close to the orchestra. For this case, the desired responses (stem recordings) are available and are required in order to derive the conversion functions. In the synthesis problem, the desired responses are not available. In the next section, we attempt to address this lack of training data by adapting the parameters

derived for the resynthesis problem, based on the derived statistics of the available reference recording of the synthesis problem. As we demonstrate, the desired waveforms can be synthesized by taking advantage of techniques developed for speech recognition parameter adaptation.

6.5 ML Constrained Adaptation

The above approach offers a possible solution to the issue of multichannel audio transmission by allowing transmission of only one or two reference channels along with the filters that can subsequently be used to recreate the remaining channels at the receiving end (virtual microphone resynthesis). Here, we are interested to address the issue of virtual microphone synthesis, *i.e.*, applying these filters to arbitrary monophonic or stereophonic recordings in order to enhance particular instrument types and completely synthesize a multichannel recording. This step requires an algorithm that generalizes these filters. In the synthesis case, no training target data will be available so some assumptions must be explicitly made about the target recording. Our approach is to derive a transformation between the reference recording used in the training step of the resynthesis algorithm and the reference recording to be used for the synthesis algorithm, that in some way represents the statistical correspondence between these two recordings. We then assume that the same transformation holds for the two corresponding target recordings and practically test this hypothesis. Techniques for deriving such transformations have been successfully applied in the task of speaker adaptation for speech recognition. In this work we applied the maximum-likelihood constrained

adaptation method [31, 30], which offers the advantage of a simple probabilistic linear transformation leading to a mathematically tractable solution for the synthesis problem.

As in the resynthesis case, we obtain a sequence of spectral vectors from the reference channel of an available multimicrophone recording. These vectors are considered as realizations of a random vector $\boldsymbol{x}$, which is modeled with a GMM as in (5.1). From the reference channel of the *stereo* recording we also obtain a sequence of spectral vectors, considered as realizations of random vector $\boldsymbol{x}'$. In this manner, we also obtain random vector $\boldsymbol{y}$ from the desired response of the multimicrophone recording, and we denote as $\boldsymbol{y}'$ the random vector that corresponds to the (not available) desired response of the stereo recording. Instead of applying a GMM for $\boldsymbol{x}'$, we attempt to relate the random variables $\boldsymbol{x}'$ and $\boldsymbol{x}$, the motivation being to derive a transformation that relates $\boldsymbol{y}'$ with $\boldsymbol{y}$. We assume that the target random vector $\boldsymbol{x}'$ is related to reference random vector $\boldsymbol{x}$ by a probabilistic linear transformation

$$\boldsymbol{x}' = \begin{cases} \mathbf{A}_1\boldsymbol{x} + \boldsymbol{b}_1 & \text{with probability } p(\lambda_1|\omega_i) \\ \mathbf{A}_2\boldsymbol{x} + \boldsymbol{b}_2 & \text{with probability } p(\lambda_2|\omega_i) \\ \vdots & \vdots \\ \mathbf{A}_N\boldsymbol{x} + \boldsymbol{b}_N & \text{with probability } p(\lambda_N|\omega_i) \end{cases} \tag{6.10}$$

This equation corresponds to the GMM constrained estimation that relates $\boldsymbol{x}'$ with $\boldsymbol{x}$ in the block diagram of Fig. 6.1. In the above equation, $\mathbf{A}_j$ denotes a $K \times K$ dimensional matrix (K is the number of components of vector $\boldsymbol{x}$), and $\boldsymbol{b}_j$ is a vector of the same

dimension with $\boldsymbol{x}$. Each of the component transformations j is related with a specific Gaussian i of $\boldsymbol{x}$ with probability $p(\lambda_j|\omega_i)$ which satisfy the constraint

$$\sum_{j=1}^{N} p(\lambda_j|\omega_i) = 1, \quad i = 1, \ldots, M \tag{6.11}$$

where M is the number of Gaussians of the GMM that corresponds to the reference vector sequence $\boldsymbol{x}$. Clearly,

$$\mathrm{g}(\boldsymbol{x}'|\omega_i, \lambda_j) = \mathcal{N}(\boldsymbol{x}'; \mathbf{A}_j\boldsymbol{\mu}_i^x + \boldsymbol{b}_j, \mathbf{A}_j\boldsymbol{\Sigma}_i^{xx}\mathbf{A}_j^T) \tag{6.12}$$

resulting in the pdf of $\boldsymbol{x}'$

$$\mathrm{g}(\boldsymbol{x}') = \sum_{i=1}^{M}\sum_{j=1}^{N} p(\omega_i)p(\lambda_j|\omega_i)\mathcal{N}(\boldsymbol{x}'; \mathbf{A}_j\boldsymbol{\mu}_i^x + \boldsymbol{b}_j, \mathbf{A}_j\boldsymbol{\Sigma}_i^{xx}\mathbf{A}_j^T) \tag{6.13}$$

which is a GMM of $M \times N$ mixtures. The matrices $\mathbf{A}_j$, the vectors $\boldsymbol{b}_j$ and the conditional probabilities $p(\omega_i)$ and $p(\lambda_j|\omega_i)$ can be estimated using maximum likelihood estimation techniques. The EM algorithm can be applied to this case in a similar manner to estimating the parameters of a GMM from observed data. In essence, it is a linearly constrained maximum-likelihood estimation of the GMM parameters.

The purpose of adopting the transformation (6.10) is to use it in order to obtain a target training sequence for the synthesis problem. The assumption is that this function represents the statistical correspondence between the two available recordings. It is then justifiable to apply the same function to the target recording of the multichannel recording to obtain a reference recording for the synthesis problem. The synthesis

problem then can be simply solved if the conversion methods mentioned in the previous section are employed. In other words, the assumption made is that the target vector $\boldsymbol{y}'$ for the synthesis problem can be obtained from the available target vector $\boldsymbol{y}$ by

$$\boldsymbol{y}' = \begin{cases} \mathbf{A}_1\boldsymbol{y} + \boldsymbol{b}_1 & \text{with probability } p(\lambda_1|\omega_i) \\ \mathbf{A}_2\boldsymbol{y} + \boldsymbol{b}_2 & \text{with probability } p(\lambda_2|\omega_i) \\ \vdots & \vdots \\ \mathbf{A}_N\boldsymbol{y} + \boldsymbol{b}_N & \text{with probability } p(\lambda_N|\omega_i) \end{cases} \tag{6.14}$$

This equation corresponds to the adaptation assumption that relates $\boldsymbol{y}'$ with $\boldsymbol{y}$ in the block diagram of Fig. 6.1.

It is now possible to derive the conversion function for the synthesis problem, based entirely on the parameters derived during the resynthesis stage that correspond to a completely different recording. A solution is provided for adapting the parameters of both the JDE and LSE resynthesis methods. This derived conversion function for synthesis will allow the synthesis of the target response from the reference channel of the stereo recording as depicted in Fig. 6.1

6.5.1 LSE Parameter Adaptation

Since it is not clear what parameters $\boldsymbol{v}_i$ and $\boldsymbol{\Gamma}_i$ represent, we follow the analysis of [95], where the form of the conversion function proposed is explained by examining the limit-case of a single class GMM for $\boldsymbol{x}$ (*i.e.* a Gaussian distribution). In that case, and

assuming the source and target vectors are jointly Gaussian, the optimal conversion function in mean-squared sense will be

$$\begin{aligned} \mathcal{F}(\boldsymbol{x}_k) &= \mathrm{E}(\boldsymbol{y}|\boldsymbol{x}_k) \\ &= \boldsymbol{\mu}^y + \boldsymbol{\Sigma}^{yx}\boldsymbol{\Sigma}^{xx^{-1}}(\boldsymbol{x}_k - \boldsymbol{\mu}^x) \\ &= \boldsymbol{v} + \boldsymbol{\Gamma}\boldsymbol{\Sigma}^{xx^{-1}}(\boldsymbol{x}_k - \boldsymbol{\mu}^x) \end{aligned} \tag{6.15}$$

where $\mathrm{E}(\cdot)$ denotes the expectation operator. So, in the limit-case, it holds that

$$\boldsymbol{v} = \boldsymbol{\mu}^y,\ \boldsymbol{\Gamma} = \boldsymbol{\Sigma}^{yx} \tag{6.16}$$

We also examine the simple case where (6.10) and (6.14) become

$$\boldsymbol{x}' = \mathbf{A}\boldsymbol{x} + \boldsymbol{b},\ \boldsymbol{y}' = \mathbf{A}\boldsymbol{y} + \boldsymbol{b} \tag{6.17}$$

Since under these conditions

$$\boldsymbol{\mu}^{x'} = \mathbf{A}\boldsymbol{\mu}^x + \boldsymbol{b},\ \boldsymbol{\mu}^{y'} = \mathbf{A}\boldsymbol{\mu}^y + \boldsymbol{b} \tag{6.18}$$

and

$$\boldsymbol{\Sigma}^{x'x'} = \mathbf{A}\boldsymbol{\Sigma}^{xx}\mathbf{A}^T,\ \boldsymbol{\Sigma}^{y'x'} = \mathbf{A}\boldsymbol{\Sigma}^{yx}\mathbf{A}^T \tag{6.19}$$

it is then apparent that the parameters $\boldsymbol{v}'$ and $\boldsymbol{\Gamma}'$ for the conversion function for the synthesis case will be

$$\boldsymbol{v}' = \mathbf{A}\boldsymbol{v} + \boldsymbol{b},\ \boldsymbol{\Gamma}' = \mathbf{A}\boldsymbol{\Gamma}\mathbf{A}^T \tag{6.20}$$

The conversion function for the limit-case becomes

$$\begin{aligned} \mathcal{F}(\boldsymbol{x}_k') &= \mathrm{E}(\boldsymbol{y}'|\boldsymbol{x}_k') \\ &= \boldsymbol{\mu}^{y'} + \boldsymbol{\Sigma}^{y'x'}\boldsymbol{\Sigma}^{x'x'^{-1}}\left(\boldsymbol{x}_k' - \boldsymbol{\mu}^{x'}\right) \\ &= \mathbf{A}\boldsymbol{v} + \boldsymbol{b} + \mathbf{A}\boldsymbol{\Gamma}\boldsymbol{\Sigma}^{xx^{-1}}\mathbf{A}^{-1}\left(\boldsymbol{x}_k' - \mathbf{A}\boldsymbol{\mu}^x - \boldsymbol{b}\right) \end{aligned} \tag{6.21}$$

By analogy then, it is justifiable to conclude that the conversion function for synthesis will be

$$\begin{aligned} \mathcal{F}(\boldsymbol{x}_k') = \sum_{i=1}^{M}\sum_{j=1}^{N} p(\omega_i|\boldsymbol{x}_k')p(\lambda_j|\boldsymbol{x}_k', \omega_i)\Big[\mathbf{A}_j\boldsymbol{v}_i + \boldsymbol{b}_j + \\ \mathbf{A}_j\boldsymbol{\Gamma}_i\boldsymbol{\Sigma}_i^{xx^{-1}}\mathbf{A}_j^{-1}\left(\boldsymbol{x}_k' - \mathbf{A}_j\boldsymbol{\mu}_i^x - \boldsymbol{b}_j\right)\Big] \end{aligned} \tag{6.22}$$

where

$$p(\omega_i|\boldsymbol{x}_k') = \frac{p(\omega_i)\sum_{j=1}^{N} p(\lambda_j|\omega_i)\mathrm{g}(\boldsymbol{x}_k'|\omega_i, \lambda_j)}{\sum_{i=1}^{M}\sum_{j=1}^{N} p(\omega_i)p(\lambda_j|\omega_i)\mathrm{g}(\boldsymbol{x}_k'|\omega_i, \lambda_j)} \tag{6.23}$$

and

$$p(\lambda_j|\boldsymbol{x}_k', \omega_i) = \frac{p(\lambda_j|\omega_i)\mathrm{g}(\boldsymbol{x}_k'|\omega_i, \lambda_j)}{\sum_{j=1}^{N} p(\lambda_j|\omega_i)\mathrm{g}(\boldsymbol{x}_k'|\omega_i, \lambda_j)} \tag{6.24}$$

and $g(\boldsymbol{x}^{'}|\omega_i, \lambda_j)$ is given from (6.12). Thus, all the parameters of the conversion function (6.22) are known from the resynthesis stage of the algorithm and the GMM constrained estimation step.

6.5.2 JDE Parameter Adaptation

Given the linearity of the transformations (6.10) and (6.14) and the fact that for a particular class ω_i, $\boldsymbol{x}$ and $\boldsymbol{y}$ will be jointly Gaussian, $\boldsymbol{x}^{'}$ and $\boldsymbol{y}^{'}$ will also be jointly Gaussian for a particular class ω_i and λ_j. Thus,

$$\begin{aligned} \mathrm{E}(\boldsymbol{y}^{'}|\boldsymbol{x}_k^{'}, \omega_i, \lambda_j) &= \boldsymbol{\mu}_i^{y^{'}} + \boldsymbol{\Sigma}_i^{y^{'}x^{'}} \boldsymbol{\Sigma}_i^{x^{'}x^{'}-1} \left(\boldsymbol{x}_k^{'} - \boldsymbol{\mu}_i^{x^{'}}\right) \\ &= \mathbf{A}_j \boldsymbol{\mu}_i^{y} + \boldsymbol{b}_j + \mathbf{A}_j \boldsymbol{\Sigma}_i^{yx} \boldsymbol{\Sigma}_i^{xx-1} \mathbf{A}_j^{-1} \\ &\quad \left(\boldsymbol{x}_k^{'} - \mathbf{A}_j \boldsymbol{\mu}_i^{x} - \boldsymbol{b}_j\right) \end{aligned} \tag{6.25}$$

since

$$\boldsymbol{\Sigma}_i^{x^{'}x^{'}} = \mathbf{A}_j \boldsymbol{\Sigma}_i^{xx} \mathbf{A}_j^{T}, \; \boldsymbol{\Sigma}_i^{y^{'}x^{'}} = \mathbf{A}_j \boldsymbol{\Sigma}_i^{yx} \mathbf{A}_j^{T} \tag{6.26}$$

and

$$\boldsymbol{\mu}_i^{x^{'}} = \mathbf{A}_j \boldsymbol{\mu}_i^{x} + \boldsymbol{b}_j, \; \boldsymbol{\mu}_i^{y^{'}} = \mathbf{A}_j \boldsymbol{\mu}_i^{y} + \boldsymbol{b}_j \tag{6.27}$$

It is also true that under the above analysis, the pdf of $\boldsymbol{y}^{'}$ will be

$$g(\boldsymbol{y}^{'}) = \sum_{i=1}^{M} \sum_{j=1}^{N} p(\omega_i) p(\lambda_j|\omega_i) \mathcal{N}(\boldsymbol{y}^{'}; \mathbf{A}_j \boldsymbol{\mu}_i^{y} + \boldsymbol{b}_j, \mathbf{A}_j \boldsymbol{\Sigma}_i^{yy} \mathbf{A}_j^{T}) \tag{6.28}$$

Band Nr.	LPC Order	GMM Classes	Components			
			M-1	M-2	M-3	M-4
1	4	4	1	2	2	4
2	4	4	1	2	2	4
3	8	8	1	2	4	8
4	16	16	1	2	8	16
5	32	16	1	2	8	16
6	32	16	1	2	8	16
7	32	16	1	2	8	16
8	32	16	1	2	8	16

Table 6.3: Parameters for the chorus microphone **synthesis** example (diagonal conversion).

Finally, the conversion function for synthesis will be

$$\begin{aligned} \mathcal{F}(\boldsymbol{x}_k^{'}) &= \mathrm{E}(\boldsymbol{y}^{'}|\boldsymbol{x}_k^{'}) \\ &= \sum_{i=1}^{M}\sum_{j=1}^{N} p(\omega_i|\boldsymbol{x}_k^{'})p(\lambda_j|\boldsymbol{x}_k^{'},\omega_i)\Big[\mathbf{A}_j\boldsymbol{\mu}_i^{y} + b_j + \\ &\quad \mathbf{A}_j\boldsymbol{\Sigma}_i^{yx}{\boldsymbol{\Sigma}_i^{xx}}^{-1}\mathbf{A}_j^{-1}\left(\boldsymbol{x}_k^{'} - \mathbf{A}_j\boldsymbol{\mu}_i^{x} - b_j\right)\Big] \end{aligned} \tag{6.29}$$

where $p(\omega_i|\boldsymbol{x}_k^{'})$ and $p(\lambda_j|\boldsymbol{x}_k^{'},\omega_i)$ are given from (6.23) and (6.24) respectively, and $\mathrm{g}(\boldsymbol{x}^{'}|\omega_i,\lambda_j)$ is given from (6.12). Again, all the parameters of the conversion function (6.29) are known from the resynthesis stage of the algorithm and the GMM constrained estimation step. It is of interest to note that the conversion function derived for the JDE synthesis problem is optimal in mean-squared sense while the conversion function for LSE synthesis is not optimal in any sense.

6.6 Synthesis Results

6.6.1 Adaptation Performance

The experimental conditions for the synthesis example (spectral conversion followed by parameter adaptation) are given in Table 6.3. Given that the methods for spectral conversion as well as for model adaptation were originally developed for speech signals, the decision to follow an analysis in subbands seemed natural. The number of GMM components for the synthesis problem is smaller than those of the resynthesis problem due to the increased computational requirements of the described algorithm for adaptation (diagonal conversion is applied for the synthesis problem as explained in Section 6.3.2).

In Tables 6.4 and 6.5, the average quadratic cepstral distance for the synthesis example is given, for the LSE and JDE methods respectively. The objective was to test the performance of the adaptation method for two different cases. The first case was when the GMM parameters correspond to a database obtained from a recording of similar nature with the recording that is attempted to be synthesized. Referring to the chorus example, the GMM parameters are derived as explained in the resynthesis algorithm, by applying the conversion method to a multichannel recording for which the chorus microphone (desired response) is available. If these parameters are applied to another recording of similar nature (*e.g.* both of classical music) the error is quite large as it appears in the second column of Tables 6.4 and 6.5 (denoted as "Same"), in the row denoted as "None" (*i.e.* no adaptation). It should be noted that the error is measured exactly as in the resynthesis case. In other words, the desired response is available for the synthesis case as well but only for measuring the error and not for

Adaptation Method	Ceps. Distance		Components per Band
	Same	Other	
None	0.9454	1.3777	Table 6.3
M-1	1.1227	1.1482	Table 6.3
M-2	1.0034	1.1348	Table 6.3
M-3	0.8794	1.0995	Table 6.3
M-4	0.8589	1.0728	Table 6.3

Table 6.4: Normalized distances for LSE method without adaptation ("None") and with several components adaptation (M-1 to M-4) for diagonal conversion.

estimating the conversion parameters. Because of limited availability of such multi-microphone orchestra recordings, the similarity of recordings was simulated by using only a small portion of the available training database (about 5%) for obtaining the GMM parameters. For testing we used the same recordings that were used for testing in the resynthesis example. The results in the second column of Tables 6.4 and 6.5 show a significant improvement in performance by increasing the number of component transformations. It is interesting to note, however, the performance degradation for small numbers of component transformations (more evident for LSE synthesis cases M-1 and M-2). This can be possibly attributed to the fact that the GMM parameters were obtained from the same recording thus, even with such a small database, they can be expected to capture some of the variability of the cepstral coefficients. On the other hand, adaptation is based on the assumption of the same transformation for the reference and target recordings, which becomes very restrictive for such a small number of transformations. The fact that larger numbers of transformation components yield significant reduction of the error, validate the methods derived here and support the assumptions that were made in the previous section.

Adaptation Method	Ceps. Distance		Components per Band
	Same	Other	
None	0.9900	1.2792	Table 6.3
M-1	0.9938	1.2341	Table 6.3
M-2	0.9303	1.1865	Table 6.3
M-3	0.9011	1.1615	Table 6.3
M-4	0.8786	1.1019	Table 6.3

Table 6.5: Normalized distances for JDE method without adaptation ("None") and several components adaptation (M-1 to M-4) for diagonal conversion.

The second case examined was when the GMM parameters corresponded to a database obtained from a recording completely different from the recording that is attempted to be synthesized. For this case, we utilized a multimicrophone recording which we obtained from a live modern music performance as explained in Section 6.2. The GMM parameters were derived from a database constructed from this recording, again the focus being on the vocals of the music. These GMM parameters were applied to the chorus testing recording of the previous examples and the results are given in the third column of Tables 6.4 and 6.5 (denoted as "Other"). An improvement in performance is apparent by increasing the number of transformation components, however this case proved to be, as expected, more demanding. The results show that adaptation is very promising for the synthesis problem, but must be applied to a database that corresponds to recordings of nature as diverse as possible.

6.6.2 Percussive Sound Synthesis

In Section 5.3.3, percussive drum-like sounds were considered as an example of transient sounds, which cannot be adequately addressed with the spectral conversion resynthesis

methods examined. It is of interest to consider possible methods to manipulate such sounds for the synthesis case as well. In this section, we examine this special case from the synthesis perspective, showing how time-frequency distributions can provide a possible alternative in some cases. The method described for percussive sound resynthesis must be modified for the case of synthesis, where the exact excitation signal is not available. One possible solution to this problem is to use the excitation signal of the same type of instrument obtained from another recording and modify it so as to provide an acoustically closer result to the desired. Here we examine the use of the Wigner distribution (WD) to this problem. A brief introduction is given for both Wigner analysis and synthesis as well as the application of these methods to the problem under consideration here. More information regarding time-frequency distributions can be found in Chapter 4. Here we state the most important results of that chapter for reference.

The Discrete-Time Wigner Distribution (DTWD) is defined in [20] as

$$W(n,\omega) = 2 \sum_{m=-\infty}^{\infty} s(n+m)s^*(n-m)e^{-j2\omega m} \tag{6.30}$$

and a similar definition exists for the discrete-time cross-Wigner distribution (DTCWD). For real signals, their analytic version can be used to avoid aliasing [10]. Smoothed Wigner distributions are obtained by using some type of window (or *kernel*) in (6.30). Different choices of kernel lead to different distributions, the most well known being the Choi-Williams distribution [17] (exponential kernel), which was designed for suppressing the interference terms. The importance of the Wigner distribution in audio signal analysis and its smoothed versions (especially the Choi-Williams distribution) lies in

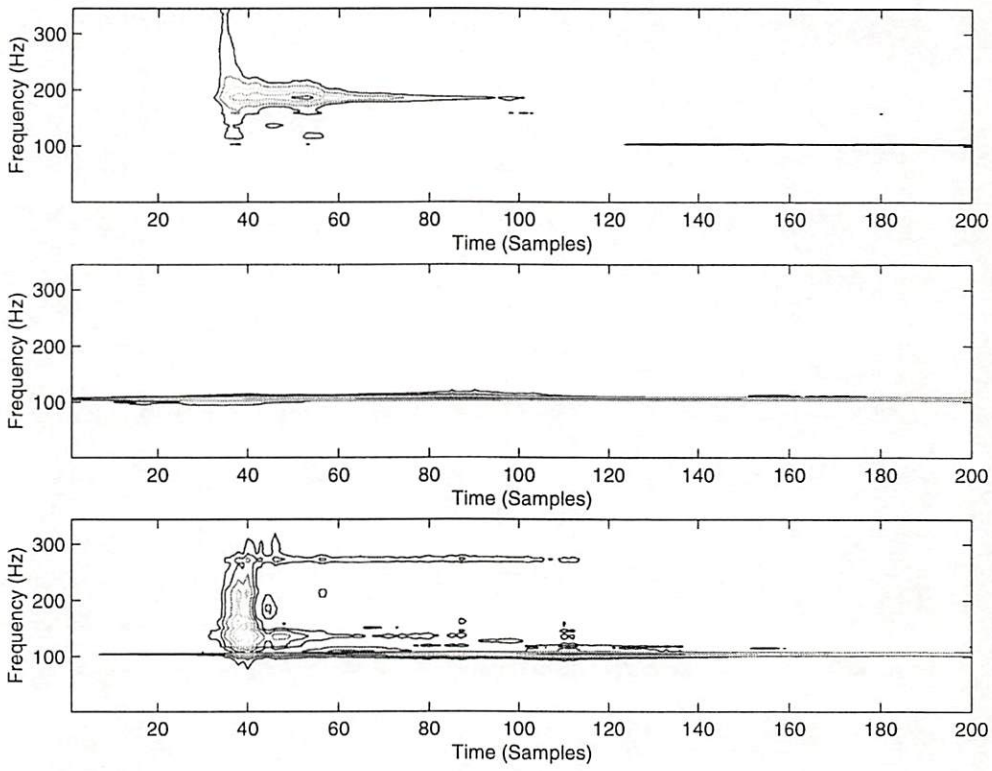

Figure 6.2: Choi-Williams distribution of the desired (top), reference (middle) and synthesized (bottom) waveforms at the time points during a tympani strike (samples 35-55).

their improved resolution compared to the spectrogram [23]. This is obvious in Fig. 5.2 and Fig. 6.2, where the distribution around a time point of a tympani strike is plotted (contour plots) along with the respective intervals of the reference and the synthesized waveforms. As mentioned in the resynthesis example, the objective is to recreate the impulsiveness which is apparent in the desired signal and absent from the reference signal.

Signal synthesis from the Wigner distribution is a subject that has been extensively examined in the literature and is defined as follows. Given an arbitrary function

$W_s(n,\omega)$, find a signal $\hat{s}(n)$ whose distribution $W_{\hat{s}}(n,\omega)$ is as close as possible to the given function. The problem arises from the fact that if the distribution of a signal is arbitrarily modified, it no longer corresponds to a valid distribution, so a simple inversion is not possible. The solution of this problem in the references given is based in least-squared minimization, that is

$$\min_{\hat{s}} \sum_{n=-\infty}^{\infty} \int_{-\pi}^{\pi} |W_s(n,\omega) - W_{\hat{s}}(n,\omega)|^2 \, \mathrm{d}\omega \tag{6.31}$$

A way to solve the WD synthesis problem is to consider even and odd samples of the signals separately, as described in [11]. This method was applied here in order to solve the percussive sound synthesis problem. The approach that was followed was to improve the excitation signal that was derived as explained in percussive sound resynthesis, from a recording that contained a similar type of instrument alone. Instead of using this excitation signal along with the AR filter extracted from the reference waveform, a synthesized excitation signal was derived by DTWD synthesis. The algorithm followed was to approximate the DTWD of the desired excitation signal with the DTCWD of the two excitation signals available (the excitation of the instrument recording and of the reference recording). Then, the desired excitation signal was derived from this distribution according to the methods of DTWD synthesis described (*e.g.* as in [11]). This method is similar to the one of [68].

6.7 Conclusions

We termed as multichannel audio resynthesis the task of recreating the multiple microphone recordings of an existing multichannel audio recording, from a smaller set of reference signals. Our motivation was to provide a scheme that allows for efficient transmission of multichannel audio through low-bandwidth networks. At the same time, the resynthesis problem arises as a first step towards solving the multichannel audio synthesis problem. Multichannel audio synthesis is the more complex task of completely synthesizing these multiple microphone recordings from an existing monophonic or stereophonic recording, thus making it available for multichannel rendering.

In this chapter, we applied spectral conversion and adaptation techniques, originally developed for speech synthesis and recognition, to the multichannel audio synthesis problem. The approach was to adapt the GMM parameters developed for the resynthesis problem (where the desired response is available for training the model) to the synthesis problem (no available desired response) by assuming that the reference and target recordings are related with a number of probabilistic linear transformations. The results we obtained were quite promising. Further research is needed in order to validate our methods by employing subjective evaluation tests in addition to the objective measures (cepstral distance) that were utilized here. It is also of interest to verify the validity of our assumption, that a more diverse collection of recordings would result in easily generalizable GMM parameters. A large number of multimicrophone recordings of various types of music performances is required in this case, which is far from trivial to acquire.

It should be noted the methods described in this chapter will not yield acceptable results for all types of sounds. Transient sounds in general cannot be adequately processed by simply modifying their short-term spectral envelope. The special case of percussive drum-like sounds was examined because of their acoustical significance and because models for these sounds are available. More work is also needed in this area for identifying other types of sounds which these methods cannot adequately address and possible alternative solutions for these cases.

Finally, other algorithms developed for task adaptation for speech recognition should be examined for the multichannel audio synthesis problem. The algorithm that was examined here, while advantageous due to the simple probabilistic linear model it assumes for relating the two different data sets, has the disadvantage of increased computational complexity.

Chapter 7

Future Research Directions

As mentioned repeatedly in this dissertation, the objective of this work is twofold: virtual rendering of sound using a stereophonic audio reproduction system and virtual synthesis of multichannel audio from a stereophonic audio recording. A complete solution of the first problem has been detailed in Chapters 2 and 3. The problem of virtual synthesis has been treated in Chapters 5 and 6. The cases covered are those of reverberant and spot microphone synthesis and resynthesis. The last step for spot microphones, *i.e.* proceeding from resynthesis to synthesis, proved to be the most challenging, as one would expect. The fact that the desired response is not available for the synthesis problem introduced, as we explained in Chapter 6, a restriction regarding the extent of the applicability of the parameter adaptation methods to different types of recordings. A question that arises from the same perspective is that of evaluating the performance of the system, since no desired result is available for comparison. The next two sections intend to provide an outline of a possible answer to those questions.

7.1 Data-Driven Approach for Virtual Microphone Signal Synthesis

For multichannel audio synthesis, the desired signal will not be available in the design of the conversion function, in contrast to multichannel audio resynthesis. An approach for overcoming this fundamental restriction was to adapt the conversion parameters, derived during the resynthesis stage, to the recording that is to be enhanced with the synthesis algorithm. In Chapter 6, it was shown that the performance of this approach is limited by the statistical proximity of the two recordings: the parameters derived based on a modern music multimicrophone recording cannot be adapted to synthesize a multichannel classical music recording. This is a point where future work on the subject can focus. It is expected that a diverse collection of multimicrophone recordings for training the GMM models of the resynthesis algorithms will provide better adaptation results. However, a related issue is how these recordings should be combined in order to achieve the best possible performance. Some possible answers to this question are suggested next.

When the number of the multimicrophone recordings in the training dataset is large, it will be difficult to associate the stereo recording to be enhanced with only one of the available recordings. One solution might be to use existing methods that relate a given recording with a large collection of available recordings based on their statistical properties. For example, a method described in [6], results in different probabilities, each one corresponding to the "distance" of a specific recording and a recording in the database. For the synthesis problem, these probabilities could then weight the different

conversion parameters for each of the recordings in the training data set, resulting in a new conversion function that will, hopefully, take advantage of the diversity of the collection of recordings in our database.

Based on the description of the previous paragraph, the following model is proposed for multichannel audio synthesis. Given, at a particular time frame, spectral vector $\boldsymbol{x}_k$, a conversion function $\mathcal{G}$ can be heuristically chosen as

$$\mathcal{G}(\boldsymbol{x}_k) = \sum_{i=1}^{Q} b_i \mathcal{F}_i(\boldsymbol{x}_k) \tag{7.1}$$

where Q is the number of multimicrophone recordings in the training dataset, each one related with the stereo reference recording with inversely proportionate distance b_i (*e.g.* as in [6]), and $\mathcal{F}_i$ is the spectral conversion function as defined in Chapter 5 and Chapter 6, for the i^{th} recording. This method in essence uses a weighted average of the Q available conversion parameters, after these are adapted to the stereo recording. The alternative to this approach would be to combine all of the recordings during the training phase. In other words, the GMM conversion parameters would be derived based on the cepstral vectors from all the recordings. The conversion function from these parameters could be viewed as a generalized conversion function. It is expected however that this method would produce inferior results compared to the method of (7.1), since it does not take into account the statistical correspondence between the stereo recording and the available multimicrophone recordings, and in effect, weighs all of the available recordings equally.

The performance of the synthesis algorithm described in the previous paragraph, as well as in the previous chapter, can be practically evaluated by means of objective performance tests, such as those described in this work. It is equally important, however, to test whether the designed conversion functions for performing the resynthesis and synthesis tasks will be applicable to recordings of different nature and to several different types of instruments. Subjective evaluation tests can be utilized for this task, if an available desired response is given. This can be accomplished by applying the conversion algorithms to existing multichannel recordings treated as conventional recordings. In other words, the available desired responses of these recordings will not be used during the training phase of the algorithms. Practically, the restriction that such recordings are currently limited exists, so these tests cannot be considered exhaustive. The listening tests that are explained in more detail in Section 7.2 can be adapted for the case when the desired response is not available, in a manner to be explained.

7.1.1 Quality Improvement by Sinusoidal Audio Signal Models

The model that was utilized for the spectral conversion methods of Chapter 5 and Chapter 6 was the residual/LP model, which is explained in Chapter 4. Speech signal modification methods have been developed based on this model as well as sinusoidal models, with the latter performing better in terms of acoustical quality. For audio signals, the sinusoidal model described in Chapter 4 has been successfully used mainly for time-scale modifications of audio signals [90]. An interesting topic would be to test the performance of this model in the resynthesis and synthesis algorithms of this work compared to the residual/LP model already implemented. It is possible to use the exact

methods of spectral conversion for the two models, since algorithms exist for estimating the cepstral coefficients from the sinusoidal model [15] as well as using the two models interchangeably [110]. The objective measures (cepstral distance) and the subjective tests (described in the next section) will reveal the more appropriate of the two models for multichannel audio resynthesis and consequently for the synthesis problem as well.

7.2 Performance Evaluation

The performance of the methods proposed can be evaluated under objective tests as well as subjective tests (listening tests). Objective tests include, for example the cepstral distance, the normalized mutual information, *etc.* The lack of availability of a diverse training database for the spectral conversion methods proposed in this work can be addressed by using specially designed listening tests that will provide a suggestion as to whether – and to what extent – the conversion functions trained on a specific recording can be used in different recordings as well. In addition, listening tests are useful for evaluating the resynthesis methods performance. Two methods are proposed to achieve such performance evaluation.

The design of the listening tests is very important for obtaining a meaningful outcome. For the synthesis and resynthesis methods (spot microphones), the experiments should involve judging their performance for enhancing various musical instruments, including the case of percussive sounds. The synthesis and resynthesis methods for reverberant microphones should be evaluated as well. The procedure suggested is based on listening tests performed for the evaluation of speech synthesis and conversion

[50, 51, 1, 109, 110, 94, 102, 80, 42], since there are great similarities in the motivation and objective of these methods with the ones proposed in this work.

7.2.1 ABX Listening Tests

These tests can provide an evaluation of each of the proposed methods separately. The listeners are presented with a large number of groups of three waveforms, A,B and finally X, and asked whether X is *perceptually* closer to A or B. These waveforms can be short segments of three different audio recordings (about 30 sec., so that there is enough time for the listener to establish a firm opinion but at the same time they must be short so that we can choose many segments of different context). A and B should be chosen from the target and reference recordings (with balanced choice, *i.e.* there must be no preference in the order these recordings are presented). It should be noted that waveforms A and B will be obtained from the original recordings using the residual/LP analysis/synthesis system (excluding the reverberant microphone methods that are not based on any model assumption). This modification will provide a performance evaluation of the proposed approach compared to the best achievable outcome possible under the restriction of the particular analysis/synthesis model. There must also be a balanced choice in the testing of the methods for the various different cases examined (various instruments, spot and reverberant synthesis and resynthesis). For the synthesis and resynthesis methods evaluation, it is expected that the desired response will be available, meaning that the testing waveforms will belong to multichannel recordings, so that the tests for both procedures will be similar. The only difference will be that for the resynthesis case, the conversion will be based on the exact desired response, while for the synthesis case

this will not hold. It is also possible for the waveforms A and/or B to be chosen from a different recording than the one corresponding to X, in order to depict an example of the perceptually desired response of the methods. This will be the case when the desired response is not available as in the synthesis problem. Since the interest is on enhancing the recording and not exactly synthesizing the desired response, it is important to test whether a method results in a recording that moves towards the desired direction (*e.g.* enhances the voices in the reference recording as opposed to exactly synthesizing the chorus microphone recording). The perceptual preference test of the following section is very useful for achieving such an evaluation.

The listening tests are expected to demonstrate the amount of degradation (if any) due to this important difference between the two procedures. Spectral conversion methods are expected to be affected by the lack of training data. It is also interesting to test whether there will be any noticeable degradation of performance in the case of reverberant sound synthesis compared to the resynthesis, which could be attributed to the fact that the filter design for this case was based on the recording that took place in a specific concert hall. These filters can be applied in an arbitrary recording, given their time-invariant nature, however, they are dependent on the specific hall acoustics which might differ significantly compared to the characteristics of the venue where the testing recording occurred.

7.2.2 Perceptual Preference Tests

In this case, the listeners are presented with two recordings, one corresponding to the conventional stereophonic recording and the other corresponding to the multichannel

synthesis enhanced version of the same recording. The listeners are asked to choose the best recording between the two, based on the criterion of which was more pleasant in listening. Again different parts of many recordings should be presented with a balanced choice regarding the order of presentation. The objective of this test is to demonstrate whether the methods proposed here provide a truly enhanced version of the conventional recording and how significant, perceptually, this enhancement appears.

Bibliography

[1] M. Abe, S. Nakamura, K. Shikano, and H. Kuwabara. Voice conversion through vector quantization. In *IEEE Proc. Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, pages 655–658, New York, NY, April 1988.

[2] J. B. Allen. Short-term spectral analysis, synthesis, and modification by discrete Fourier transform. *IEEE Trans. Acoust., Speech, and Signal Process.*, ASSP-25:235–238, June 1977.

[3] J. Bauck and D. H. Cooper. Generalized transaural stereo and applications. *Journal of the Audio Engineering Society*, 44:683–705, 1996.

[4] G. Baudoin and Y. Stylianou. On the transformation of the speech spectrum for voice conversion. In *IEEE Proc. Int. Conf. Spoken Language Processing (ICSLP)*, pages 1405–1408, Philadephia, PA, October 1996.

[5] D. R. Begault. Challenges to the successful implementation of 3-D sound. *Journal of the Audio Engineering Society*, 39:864–870, 1991.

[6] A. Berenzweig, D. P. W. Ellis, and S. Lawrence. Anchor models for classification and similarity measurement of music. Submitted *IEEE Int. Conf. Multimedia and Expo (ICME)*, 2002.

[7] R. B. Blackman and J. W. Tukey. *The Measurement of Power Spectra.* Dover Publications, 1958.

[8] J. Blauert. *Spatial Hearing: The Psychophysics of Human Sound Localization, Revised Edition.* MIT Press, 1997.

[9] J. Blauert, H. Lehnert, W. Pompetzki, and N. Xiang. Binaural room simulation. *Acustica*, pages 295–296, 1990.

[10] B. Boashash. Note on the use of the Wigner distribution for time-frequency signal analysis. *IEEE Trans. Signal Processing*, 36(9):1518–1521, September 1988.

[11] G. F. Boudreaux-Bartels and T. W. Parks. Time-varying filtering and signal estimation using Wigner distribution synthesis techniques. *IEEE Trans. Acoust., Speech, and Signal Process.*, ASSP-34(3):442–451, June 1986.

[12] G. F. Boudreaux-Bartels and P. J. Wiseman. Wigner distribution of acoustic well logs. In *IEEE Proc. Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, pages 2237–2240, Dallas, TX, April 1987.

[13] C. P. Brown and R. O. Duda. A structural model for binaural sound synthesis. *IEEE Trans. on Speech and Audio*, 6:476–488, 1998.

[14] R. A. Butler. Spatial hearing: the psychophysics of human sound localization. *J. Acoust. Soc. Am.*, 77:334–335, 1985.

[15] O. Cappe, J. Laroche, and E. Moulines. Regularized estimation of cpestrum envelope from discrete frequency points. In *IEEE ASSP Workshop on App. of Sig. Proc. to Audio and Acoust.*, Mohonk, NY, October 1995.

[16] O. Cappe and E. Moulines. Regularization techniques for discrete cepstrum estimation. *IEEE Signal Processing Letters*, 3(4):100–102, April 1996.

[17] H.-I. Choi and J. Williams. Improved time-frequency representation of multicomponent signals using exponential kernels. *IEEE Trans. Acoust., Speech, and Signal Process.*, 37(6):862–871, June 1989.

[18] J. M. Cioffi and T. Kailath. Fast, recursive least-squares transversal filters for adaptive filtering. *IEEE Trans. Acoust., Speech, and Signal Process.*, ASSP-32:304–307, 1984.

[19] T. A. C. M. Claasen and W. F. G. Mecklenbrauker. The Wigner distribution – a tool for time-frequency signal analysis; part I: Continuous-time signals. *Philips J. Res.*, 35(5):217–250, January 1980.

[20] T. A. C. M. Claasen and W. F. G. Mecklenbrauker. The Wigner distribution – a tool for time-frequency signal analysis; part II: Discrete-time signals. *Philips J. Res.*, 35(5):276–300, January 1980.

[21] T. A. C. M. Claasen and W. F. G. Mecklenbrauker. The Wigner distribution – a tool for time-frequency signal analysis; part III: Relations with other time-frequency signal transformations. *Philips J. Res.*, 35(5):372–389, January 1980.

[22] T. A.C.M. Claasen and W. F. G. Mecklenbrauker. The aliasing problem in discrete-time Wigner distributions. *IEEE Trans. Acoust., Speech, and Signal Process.*, ASSP-31(5):1067–1072, October 1983.

[23] L. Cohen. *Time-Frequency Analysis.* Prentice Hall, 1995.

[24] J. W. Cooley and J. W. Tukey. An algorithm for the machine calculation of complex Fourier series. *Math. Comp.*, 19(90):297–301, April 1965.

[25] D. H. Cooper. Calculator program for head-related transfer functions. *Journal of the Audio Engineering Society*, 30:34–38, 1982.

[26] D. H. Cooper and J. Bauck. Prospects for transaural recording. *Journal of the Audio Engineering Society*, 37:3–19, 1989.

[27] T. M. Cover and J. A. Thomas. *Elements of Information Theory.* Wiley, 1991.

[28] P. Damaske. Head related two channel stereophony with loudspeaker reproduction. *J. Acoust. Soc. Am.*, 50:1109–1115, 1971.

[29] P. Damaske and V. Mellert. A procedure for generating directionally accurate sound images in the upper half-space using two loudspeakers. *Acustica*, 22:154–162, 1969.

[30] V. D. Diakoloukas and V. V. Digalakis. Maximum-likelihood stochastic-transformation adaptation of Hidden Markov Models. *IEEE Trans. Speech and Audio Processing*, 7(2):177–187, March 1999.

[31] V. V. Digalakis, D. Rtischev, and L. G. Neumeyer. Speaker adaptation using constrained estimation of Gaussian mixtures. *IEEE Trans. Speech and Audio Processing*, 3(5):357–366, September 1995.

[32] S. Farkash and S. Raz. Linear systems in Gabor time-frequency space. *IEEE Trans. Signal Processing*, 42(3):611–617, March 1994.

[33] H. G. Feichtinger and T. Strohmer, editors. *Gabor Analysis and Algorithms.* Birkhauser, 1998.

[34] W. G. Gardner. Transaural 3-D audio. Technical Report 342, MIT, January/February 1995.

[35] W. G. Gardner. Head-tracked 3-D audio using loudspeakers. In *WASPAA '97*, New Palz, New York, 1997.

[36] W. G. Gardner. *3-D Audio Using Loudspeakers.* Kluwer Academic Publishers, 1998.

[37] W. G. Gardner and K. Martin. HRTF measurements of a KEMAR dummy-head microphone. Technical Report 280, MIT, May 1994.

[38] E. B. George and M. J. T. Smith. Speech analysis/synthesis and modification using an analysis-by-synthesis/overlap-add sinusoidal model. *IEEE Trans. Speech and Audio Processing*, 5(5):389–406, September 1997.

[39] P. G. Georgiou, A. Mouchtaris, S. I. Roumeliotis, and C. Kyriakakis. Immersive sound rendering using laser-based tracking. In *Proc.* 109^{th} *Convention of the Audio Engineering Society (AES), preprint No. 5227*, Los Angeles, CA, September 2000.

[40] H. W. Gierlich. The application of binaural technology. *Applied Acoustics*, 36:219–243, 1992.

[41] D. W. Griffin and J. S. Lim. Signal estimation from modified short-time Fourier transform. *IEEE Trans. Acoust., Speech, and Signal Process.*, ASSP-32(2):236–242, April 1984.

[42] D. W. Griffin and J. S. Lim. Multiband excitation vocoder. *IEEE Trans. Acoust., Speech, and Signal Process.*, 36(8):1223–1235, August 1988.

[43] S. Haykin. *Adaptive Filter Theory.* Prentice Hall, 1996.

[44] F. Hlawatsch. *Time-Frequency Analysis and Synthesis of Linear Signal Spaces.* Kluwer Academic Publishers, 1998.

[45] F. Hlawatsch and G. F. Boudreaux-Bartels. Linear and quadratic time-frequency signal representations. *IEEE Signal Processing Magazine*, 35:21–67, April 1992.

[46] F. Hlawatsch, A. H. Costa, and W. Krattenthaler. Time-frequency signal synthesis with time-frequency extrapolation and don't-care regions. *IEEE Trans. Signal Processing*, 42(9):2513–2520, September 1994.

[47] F. Hlawatsch and W. Krattenhaler. Bilinear signal synthesis. *IEEE Trans. Signal Processing*, 40(2):352–363, February 1992.

[48] F. Hlawatsch and W. Krattenthaler. Phase matching algorithms for Wigner-distribution signal synthesis. *IEEE Trans. Signal Processing*, 39(3):612–619, March 1991.

[49] F. Itakura and S. Saito. A statistical method for estimation of speech spectral density and formant frequencies. *Electronics and Communications in Japan*, 53A:36–43, 1970.

[50] A. Kain and M. W. Macon. Spectral voice conversion for text-to-speech synthesis. In *IEEE Proc. Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, pages 285–289, Seattle, WA, May 1998.

[51] A. Kain and M. W. Macon. Design and evaluation of a voice conversion algorithm based on spectral envelope mapping and residual prediction. In *IEEE Proc. Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, pages 813–816, Salt Lake City, UT, May 2001.

[52] W. Krattenthaler and F. Hlawatsch. Two signal synthesis algorithms for pseudo-Wigner distribution. In *IEEE Proc. Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, pages 1550–1553, New York, NY, April 1988.

[53] W. Krattenthaler and F. Hlawatsch. Improved signal synthesis from pseudo-Wigner distribution. *IEEE Trans. Signal Processing*, 39(2):506–509, February 1991.

[54] W. Krattenthaler and F. Hlawatsch. Time-frequency design and processing of signals via smoothed Wigner distributions. *IEEE Trans. Signal Processing*, 41(1):278–287, January 1993.

[55] B. V. K. Vijaya Kumar, C. P. Neuman, and K. J. DeVos. Discrete Wigner synthesis. *Signal Processing*, 11(3):277–304, 1986.

[56] C. Kyriakakis. Fundamental and technological limitations of immersive audio systems. *Proc. IEEE*, 86:941–951, 1998.

[57] C. Kyriakakis and T. Holman. Video-based head tracking for improvements in multichannel loudspeaker audio. In *105 Meeting of the Audio Engineering Society*, San Francisco,California, 1998.

[58] C. Kyriakakis, T. Holman, J.-S. Lim, H. Hongand, and H. Neven. Signal processing, acoustics, and psychoacoustics for high quality desktop audio. *Journal of Visual Communication and Image Representation*, 9:51–61, 1997.

[59] C. Kyriakakis and A. Mouchtaris. Virtual microphones for multichannel audio applications. In *Proc. IEEE Int. Conf. Multimedia and Expo (ICME)*, volume 1, pages 11–14, New York, NY, July 2000.

[60] J. Laroche. A new analysis/synthesis system of musical signals using Prony's method-application to heavily damped percussive sounds. In *IEEE Proc. Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, pages 2053–2056, Glasgow, UK, May 1989.

[61] J. Laroche and J.-L. Meillier. Multichannel excitation/filter modeling of percussive sounds with application to the piano. *IEEE Trans. Speech and Audio Processing*, 2:329–344, 1994.

[62] S. N. Levine, T. S. Verma, and J. O. Smith III. Multiresolution sinusoidal modeling for wideband audio with modifications. In *IEEE Proc. Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, pages 3585–3588, Seattle, WA, May 1998.

[63] L. Ljung. *System Identification: Theory for the User.* Prentice Hall, 1987.

[64] M. W. Macon, A. McCree, Lai Wai-Ming, and V. Viswanathan. Efficient analysis/synthesis of percussion musical instrument sounds using an all-pole model. In *IEEE Proc. Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, pages 3589–3592, Seattle, WA, May 1998.

[65] W. Martin and P. Flandrin. Wigner-Ville spectral analysis of nonstationary processes. *IEEE Trans. Acoust., Speech, and Signal Process.*, ASSP-33(6):1461–1470, December 1985.

[66] M. Mboup, M. Bonnet, and N. Bershad. LMS coupled adaptive prediction and system identification: A statistical model and transient mean analysis. *IEEE Trans. Signal Processing*, 42(10):2607–2615, October 1994.

[67] R. J. McAulay and T. F. Quatieri. Speech analysis/synthesis based on a sinusoidal representation. *IEEE Trans. Acoust., Speech, and Signal Process.*, ASSP-34(4):744–754, August 1986.

[68] T. J. McHale and G. F. Boudreaux-Bartels. An algorithm for synthesizing signals from partial time-frequency models using the cross-Wigner distribution. *IEEE Trans. Signal Processing*, 41(5):1986–1990, May 1993.

[69] J. M. Mendel. *Lessons in Estimation Theory for Signal Processing, Communications and Control.* Prentice Hall, 1995.

[70] H. Moller. Fundamentals of binaural technology. *Applied Acoustics*, 36:171–218, 1992.

[71] H. Moller, M. F. Sorensen, and D. Hammershoi. Head-related transfer functions of human subjects. *Journal of the Audio Engineering Society*, 43:300–321, 1995.

[72] B. C. J. Moore. *An Introduction in the Psychology of Hearing.* Academic Press, 1989.

[73] A. Mouchtaris and C. Kyriakakis. Time-frequency methods for virtual microphone signal synthesis. In *Proc.* 111^{th} *Convention of the Audio Engineering Society (AES), preprint No. 5416*, New York, NY, November 2001.

[74] A. Mouchtaris, S. S. Narayanan, and C. Kyriakakis. Multiresolution spectral conversion for multichannel audio resynthesis. In *IEEE Proc. Int. Conf. Multimedia and Expo (ICME)*, volume 2, pages 273–276, Lausanne, Switzerland, August 2002.

[75] A. Mouchtaris, P. Reveliotis, and C. Kyriakakis. Non-minimum phase inverse filter methods for immersive audio rendering. In *IEEE Proc. Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, pages 3077–3080, Phoenix, AZ, March 1999.

[76] A. Mouchtaris, P. Reveliotis, and C. Kyriakakis. Inverse filter design for immersive audio rendering over loudspeakers. *IEEE Trans. Multimedia*, 2(2):77–87, 2000.

[77] A. Mouchtaris, Z. Zhu, and C. Kyriakakis. High-quality multichannel audio over the Internet. In *Conf. Record of the Thirty-Third Assilomar Conf. Signals, Systems and Computers*, volume 1, pages 347–351, Pacific Grove, CA, October 1999.

[78] A. D. Musicant and R. A. Butler. The influence of pinnae-based spectral cues on sound localization. *J. Acoust. Soc. Am.*, 75:1195–1200, 1984.

[79] P. A. Nelson, H. Hamada, and S. J. Elliott. Adaptive inverse filters for stereophonic sound reproduction. *IEEE Trans. Signal Processing*, 40:1621–1632, 1992.

[80] D. O'Brien and A. I. C. Monaghan. Concatenative synthesis based on a harmonic model. *IEEE Trans. Speech and Audio Processing*, 9(1):11–20, January 2001.

[81] A. V. Oppenheim and R. W. Shafer. *Discrete Time Signal Processing.* Prentice Hall, 1989.

[82] W. J. Pielemeier and G. H. Wakefield. A high-resolution time-frequency representation for musical instrument signals. *J. Acoust. Soc. Am.*, 99(4):2382–2396, April 1996.

[83] W. J. Pielemeier, G. H. Wakefield, and M. H. Simoni. Time-frequency analysis of musical signals. *Proc. IEEE*, 84(9):1216–1230, September 1996.

[84] M. R. Portnoff. Time-frequency representation of digital signals and systems based on short-time Fourier analysis. *IEEE Trans. Acoust., Speech, and Signal Process.*, ASSP-28:55–69, February 1980.

[85] L. Rabiner and B.-H. Juang. *Fundamentals of Speech Recognition.* Prentice Hall, 1993.

[86] B. D. Radlovic and R. A. Kennedy. Nonminimum-phase equalization and its subjective importance in room acoustics. *IEEE Trans. Speech and Audio Processing*, 8(6):728–737, November 2000.

[87] D. A. Reynolds and R. C. Rose. Robust text-independent speaker identification using Gaussian mixture speaker models. *IEEE Trans. Speech and Audio Processing*, 3(1):72–83, January 1995.

[88] B. E. A. Saleh and N. S. Subotic. Time-variant filtering of signals in the mixed time-frequency domain. *IEEE Trans. Acoust., Speech, and Signal Process.*, ASSP-33(6):1479–1485, December 1985.

[89] M. R. Schroeder and B. S. Atal. Computer simulation of sound transmission in rooms. In *IEEE International Convention Record*, volume 7, 1963.

[90] X. Serra and J. O. Smith III. Spectral modeling sythesis: A sound analysis/synthesis system based on a deterministic plus stochastic decomposition. *Computer Music Journal*, 14(4):12–24, Winter 1990.

[91] O. Shalvi and E. Weinstein. System identification using nonstationary signals. *IEEE Trans. Signal Processing*, 44(8):2055–2063, August 1996.

[92] C. Shekhar and R. Chellappa. Experimental evaluation of two criteria for pattern comparison and alignment. In *Proc. Fourteenth International Conference on Pattern Recognition*, volume 1, pages 146–153, Brisbane, Australia, August 1998.

[93] G. Strang and T. Nguyen. *Wavelets and Filter Banks.* Wellesley-Cambridge, 1996.

[94] Y. Stylianou. Applying the harmonic plus noise model in concatenative speech synthesis. *IEEE Trans. Speech and Audio Processing*, 9(1):21–29, January 2001.

[95] Y. Stylianou, O. Cappe, and E. Moulines. Continuous probabilistic transform for voice conversion. *IEEE Trans. Speech and Audio Processing*, 6(2):131–142, March 1998.

[96] R. B. Sussman and M. Kahrs. Analysis and resynthesis of musical instrument sounds using energy separation. In *IEEE Proc. Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, pages 997–1000, Atlanta, GA, May 1996.

[97] S. Sussman. Least-square synthesis of radar ambiguity functions. *IRE Trans. on Information Theory*, IT-8:246–254, April 1962.

[98] Jr. T. G. Stockham, T. M. Cannon, and R. B. Ingebretsen. Blind deconvolution through digital signal processing. *Proc. IEEE*, 63(4):678–692, April 1975.

[99] F. E. Toole. Loudspeaker measurements and their relationship to listener preferences. *Journal of the Audio Engineering Society*, 34:227–235, 1986.

[100] F. E. Toole and S. E. Olive. The modification of timbre by resonances: perception and measurement. *Journal of the Audio Engineering Society*, 36:122–142, 1988.

[101] D. B. Trizna, C. Bachmann, M. Sletten, N. Allan, J. Topokovand, and R. Harris. Projection pursuit classification methods applied to multiband polarimentric SAR imagery. In *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS) 2000*, volume 1, pages 105–107, Honolulu, HI, July 2000.

[102] F. Violaro and O. Boeffard. A hybrid model for text-to-speech synthesis. *IEEE Trans. Speech and Audio Processing*, 6(5):426–434, September 1998.

[103] R. Walker. Early reflections in studio control rooms: The results from the first controlled image design installations. In *96 Meeting of the Audio Engineering Society*, Amsterdam, 1994.

[104] P. W. Welch. The use of fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms. *IEEE Trans. Audio and Electroacoustics*, AU-15:70–73, June 1967.

[105] E. M. Wenzel, M. Arruda, and D. J. Kistler. Localization using nonindividualized head-related transfer functions. *J. Acoust. Soc. Am.*, 94:111–123, 1993.

[106] J. Wexler and S. Raz. Synthesis of discrete-time signals from distributions. *Electronics Letters*, 25(2):93–95, January 1989.

[107] F. L. Wightman and D. J. Kistler. The dominant role of low-frequency interaural time differences in sound localization,. *J. Acoust. Soc. Am.*, 91:1648–1661, 1992.

[108] F. L. Wightman, D. J. Kistler, and M. Arruda. Perceptual consequences of engineering compromises in synthesis of virtual auditory objects. *J. Acoust. Soc. Am.*, 101:1050–1063, 1992.

[109] J. W. Wouters and M. W. Macon. Spectral modification for concatenative speech synthesis. In *IEEE Proc. Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, pages 941–944, Istanbul, Turkey, June 2000.

[110] J. W. Wouters and M. W. Macon. Control of spectral dynamics in concatenative speech synthesis. *IEEE Trans. Speech and Audio Processing*, 9(1):30–38, January 2001.

[111] Z. Wu, F. H. Y. Chan, F. K. Lam, and J. C. K. Chan. A time domain binaural model based on spatial feature extraction for the head-related transfer function. *J. Acoust. Soc. Am.*, 102(4):2211–2218, 1997.

[112] K-B Yu and S. Cheng. Signal synthesis from Wigner distribution. In *IEEE Proc. Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, pages 1037–1040, Tampa, FL, March 1985.

[113] K.-B. Yu and S. Cheng. Signal synthesis from pseudo-Wigner distribution and applications. *IEEE Trans. Acoust., Speech, and Signal Process.*, ASSP-35(9):1289–1302, September 1987.