Suppression of Pitched Musical Sources in Signal Mixtures

by

Carola Behrens
B.A.Sc., University of British Columbia, 1999

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of
MASTER OF APPLIED SCIENCE
in the Department of Electrical and Computer Engineering

Supervisor: Dr. Peter Driessen

Abstract

In this thesis, methods for purification of pitched musical signals recorded by spot microphones are presented and evaluated. Spot microphones capture the sound of a desired instrument in a musical ensemble but also inevitably capture some of the sound from neighbouring musical instruments. The purification methods attempt to suppress the interference from the neighbouring musical instruments.

The interference suppression methods are based on a sinusoidal model for the desired and interfering signals. The sinusoidal model represents a signal as a collection of sinusoids with slowly evolving amplitude, frequency and phase. This model is shown to be valid for signals from pitched musical instruments such as the piano.

Two interference suppression methods that target the sinusoidal (pitched) signal components are proposed and evaluated in this thesis. The filtering method for interference suppression involves the use of time-varying notch filters to suppress the interfering sinusoids. The subtraction method for interference suppression involves synthesising an estimate of the interference and subtracting it from the mixed signal. The filtering method has the advantage that it is not very sensitive to errors in the sinusoidal model, but has the disadvantage that it suppresses any desired signal components that coincide with the notch of the filters. The subtraction method has the advantage that the desired signal is not severely distorted, but has the disadvantage that it is very sensitive to errors in the sinusoidal model.

The sinusoidal model is not a complete model for most musical signals because transient and aharmonic components are not accounted for. The consequence for the interference suppression methods is that these components remain in the recovered signals. A method for suppression of some of the interfering transients in the recovered signals is proposed and evaluated.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ASA	Auditory Scene Analysis
BCI	Blind Channel Identification
BSD	Blind Separation and Deconvolution
BSS	Blind Source Separation
CASA	Computational Auditory Scene Analysis
dB	decibels
DFT	Discrete Fourier Transform
ICA	Independent Component Analysis
IDFT	Inverse Discrete Fourier Transform
PCA	Principal Component Analysis
PDF	Probability Density Function
PSD	Power Spectral Density
SA	Sinusoidal Analysis
SOS	Second-Order Section
STDFT	Short-Time Discrete Fourier Transform
TF	Transfer Function

Acknowledgments

Thank you to my supervisor, Peter Driessen, for his endless patience with me as I repeatedly changed my mind about what research topic to pursue. I appreciate the freedom given to me to find my own way and the positive attitude I encountered with each proposal that I came up with.

I am also very grateful for the help of my co-supervisor, Lynn Kirlin, who provided many interesting suggestions for this work and the encouragement to keep at it.

I would also like to thank colleagues and supervisors at IVL, particularly Glen Rutledge and Peter Lupini for their advice and contributions to discussions related to this work. A special thank-you to Brian Gibson for some frank, motivational words which proved helpful in finally choosing a topic and getting the job done.

Thank you also to my parents, Kai and Hilda, my brother Carl and dear friends Laura, Julie and Alison for their love and moral support that provided much of the fuel I needed to complete this work.

Further, I would like to acknowledge and thank those people and organisations that provided me with financial assistance to complete this work: Lynn Kirlin, IVL Technologies Ltd., the National Sciences and Engineering Research Council (NSERC) and the University of Victoria.

Finally, I would like to express my gratitude to the committee and examiners for taking the time to read this thesis and attend the defense in the middle of summer when there are so many more appealing things to do!

Chapter 1

Introduction

1.1 Problem Description

Recordings of multi-source acoustic events, such as a concert, are most often made with multiple microphones. The raw signals picked up by the microphones are processed then summed together (mixed) to create one or more output signals ready to be rendered by a playback system. The signal flow is illustrated in Figure 1.1 below.

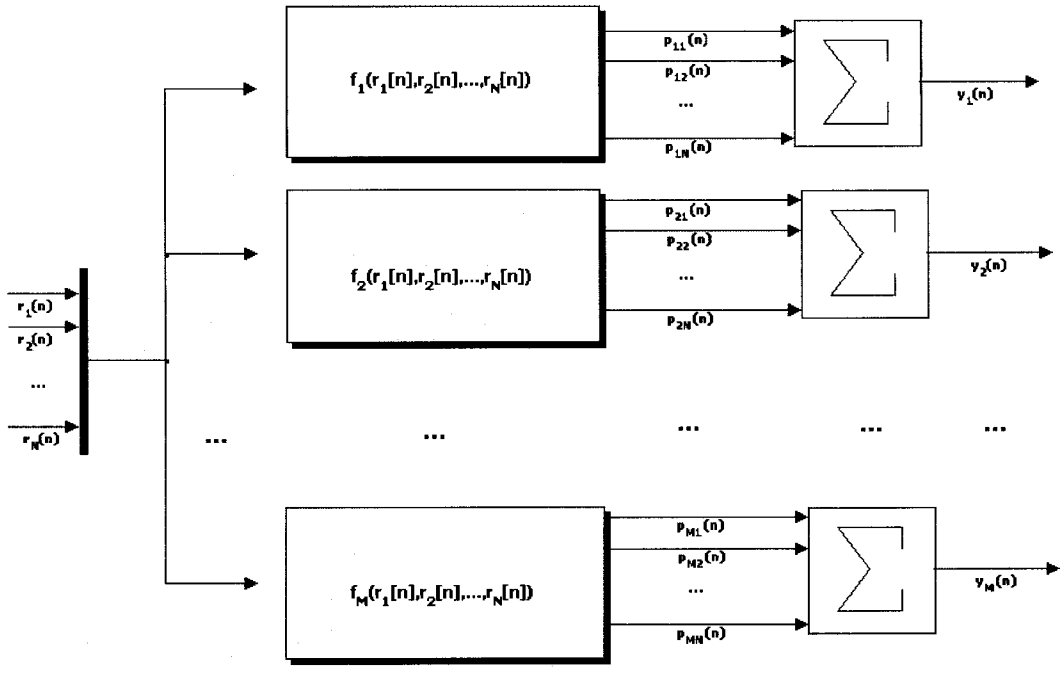


Figure 1.1. Signal Flow Diagram for Mixing of Recorded Audio Signals.

In Figure 1.1 there are N raw recorded signals, $r_k[n]$, from N microphones, and M output signals, $y_m[n]$, to be rendered by an M-channel playback system. The functions, f_m, generate the processed signals, $p_{mk}[n]$, to be mixed into the m^{th} output

signal. Examples of processing include the application of different scaling factors, delays or filters ([1]-[4]). In the case where only a subset of the raw signals contribute to the m^{th} output signal, one can consider the processing function on the non-contributing raw signals to be a scaling factor of zero.

The raw signals are captured by one of two classes of microphones. "Spot" microphones are placed very close to a sound source and are used to capture the direct sound from the target source. Due to the proximity of the spot microphone to the sound source, there is very little reverberation in the captured signal. "Main" microphones are placed at a distance from the ensemble of sources and are used to capture the composite sound of the ensemble. Due to the distance of the main microphone from the sources, the power of the reverberant component of the captured signal is significant, giving an impression of spatial depth [5]. Spot microphone signals are most often used to reinforce specific sound sources in the main microphone signals.

Before spot microphone signals can be used for sound source reinforcement they must be processed, for example by panning, delaying or filtering, so that the spatial image of the processed spot signals is consistent with the spatial image captured at the main microphones. The consequences of inadequate processing of spot microphone signals as well as some ideas for processing techniques are discussed in [2]. The perfect solution to the reinforcement problem is to convolve the ideal spot microphone signal with the impulse response between the sound source and each main microphone. This "ideal" reinforcement scheme is illustrated in Figure 1.2 below.

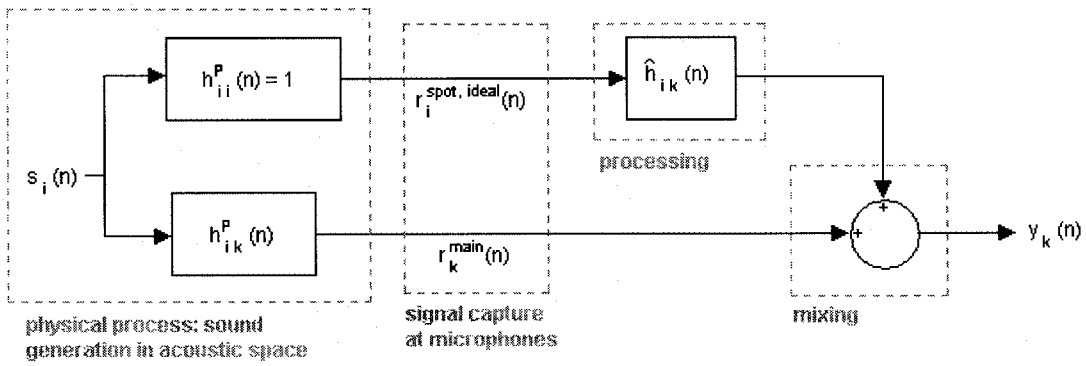


Figure 1.2. Ideal Sound Source Reinforcement in Main Microphone Signal Using Processed Spot Microphone

In Figure 1.2, the sound source *i* is enhanced in the composite signal recorded in main microphone *k*. The impulse responses $h_{ii}^P(n)$ and $h_{ik}^P(n)$ are due to the acoustic characteristics of the room in which the sound source exists; $h_{ii}^P(n)$ is the impulse response between the sound source *i* and spot microphone *i*, which is placed close to the sound source and $h_{ik}^P(n)$ is the impulse response between the sound source *i* and the main microphone *k*. The impulse response $\hat{h}_{ik}(n)$ is the estimate of the room impulse response function $h_{ik}^P(n)$.

To achieve ideal enhancement, there are two requirements:

1. **pure spot microphone signals:**
   the physical impulse response between sound source *i* and its spot microphone, $h_{ii}^P(n)$, is $\delta(n)$ and the physical impulse response between all other sound sources and spot microphone *i* (not shown in Figure 1.2) is 0, so that the signal recorded at the spot microphone is equal to the source signal, $r_i^{spot,\ ideal}(n) = s_i(n)$
2. **knowledge of the physical impulse responses between source signals and main microphones:**
   the estimate of the impulse response between source *i* and the main microphone *k* is equal to the same physical impulse response, $h_{ik}^P(n)$.

In practice, neither requirement is met. Even with careful microphone technique, the spot microphones always pick up some sound from other sources. The leakage

from the other sound sources in the spot microphone signals causes blurring of the spatial image of these interfering sources after mixing [6]. Furthermore, the exact impulse responses between the sound sources and main microphones is unknown, so sound engineers have to resort to burying the spot microphone signals in the reverberation of the main microphone signals so as not to colour the sound of or distort the spatial image of the source [2].

The requirement of the source-to-main microphone impulse responses can be relaxed or omitted altogether if the sound engineer has other goals besides perfect reinforcement of sound sources. A common example of such a goal is reduction of reverberation in the final mix, in which case only the pure spot microphone signals and the delays of the direct wavefront between sound sources and main microphones are required. Regardless of the mixing goals of the sound engineer, the requirement of pure spot microphone signals remains. While it is not possible to acquire such pure signals at the spot microphones, a method to purify the spot microphone signals at the processing stage prior to mixing would greatly improve the final mix of multi-source recordings.

The availability of pure spot microphone signals opens up more signal processing possibilities that would otherwise not be practical due to the effects of leakage from other sources. Some examples of spot signal processing possibilities include:

- convolution with impulse responses of different acoustic spaces to give the impression that the sound sources originated in different locations
- removal of certain sound sources by omitting them in the final mix
- modification of sound source signals (e.g. pitch or time shifting) for error correction or other artistic ends.

Pure spot microphone signals even have applications beyond audio reproduction. Some examples include:

- automatic transcription of live ensemble music without having to rely on problematic polyphonic pitch detectors
- structured audio compression of live ensemble music, where each sound source is individually coded using a model for the source.

This thesis is devoted to developing some processing methods for purifying signals from spot microphones by reducing the amount of interference from other sound sources picked up by said microphones. . The interference suppression problem explored in this thesis is closely related to the source enhancement problem often addressed using microphone arrays, where a desired source is enhanced by spatial filtering of the array signals. Another related problem is that of adaptive noise cancellation. The signal captured by the spot microphone positioned to capture the undesired source may be considered as the reference "noise" signal.

1.2 Contribution and Organisation of the Thesis

In this thesis, two methods for suppressing interfering pitched musical signals in a musical signal mixture are presented and evaluated. The interference suppression methods were designed to attenuate undesired pitched musical sound sources picked up by spot microphones in the recording of ensemble music. The goal of the interference suppression methods is to produce audio signals that are an accurate representation of the desired sound source (musical instrument) playing in isolation. To this end, the interference suppression methods were designed to attenuate the interfering instrument sounds whilst preserving the time-frequency character of the desired instrument sound. The interference suppression methods rely on the well-developed technique known as sinusoidal modeling, first presented by McAulay and Quatieri in [7]. The methods developed in this thesis are based heavily on the work of Tolonen [8] and Virtanen and Klapuri [9].

The work described in this thesis contributes to the important task purification of spot microphone signals discussed in the previous section by suppressing undesired sounds from pitched musical instruments picked up by the spot microphone. The suppression methods described in this thesis do not address interference from non-pitched musical instruments such as the snare drum. With regards to the spot microphone signal purification task, this thesis does not address the issue of a non-unity transfer function between the target instrument and the corresponding signal picked up at the spot microphone. It is assumed that if this is an issue that needs to be

addressed for a particular application, a blind dereverberation method (e.g. [10]-[12]) could be applied after the interference suppression methods described in this thesis.

This thesis is divided into six chapters. In Chapter 2 some general approaches to solving the closely related problem of audio signal separation described in the literature is outlined. Since the approach presented later in this thesis relies on sinusoidal modeling, this technique is described as background information in Chapter 3. The interference suppression methods are described in Chapter 4 and some results are discussed in Chapter 5. The thesis work is summarised and suggestions for future work are presented in Chapter 6.

1.3 Notation

The following describe the general notation used in this thesis.

- All signals and impulse responses are described in the discrete-time domain. The notation used to describe a signal or impulse response is: *x[n]*, where *n* is an integer. Formally, a discrete-time signal or impulse response is denoted *x[nT]* where *T* is the sampling period. To reduce clutter in mathematical equations involving signals and impulse responses the *T* is dropped from the notation, but is always implied. It is assumed that *T* is sufficiently small so as to allow full reconstruction of the discrete-time signals for all times, according to Shannon's sampling theorem:

$$T < \frac{1}{2f_{\max}} \tag{1.1}$$

- vectors and matrices are denoted with bold face, for example **A**
- the expectation operator is denoted as *E[•]*

Chapter 2

Audio Signal Separation Methods

2.1 Introduction

In this chapter some existing methods for audio signal separation are reviewed. The problem of interference suppression in spot microphone signals is closely related to the signal separation problem.

The goal of signal separation is to recover the pure source signals from one or more observations of linear mixtures of the source signals. A set of J observed signals, which are mixtures of K source signals, is given by

$$\begin{bmatrix} x_1[n] \\ \vdots \\ x_j[n] \\ \vdots \\ x_J[n] \end{bmatrix} = \begin{bmatrix} a_{11} & \cdots & a_{1K} \\ \vdots & \ddots & \vdots \\ a_{j1} & \cdots & a_{jk} \\ \vdots & \ddots & \vdots \\ a_{J1} & \cdots & a_{JK} \end{bmatrix} \begin{bmatrix} s_1[n] \\ \vdots \\ s_k[n] \\ \vdots \\ s_K[n] \end{bmatrix} \tag{2.1}$$

where $x_j[n]$ is the j^{th} observed signal, a_{jk} are the mixing coefficients and $s_k[n]$ is the k^{th} source signal. Equation (2.1) can be written succinctly as:

$$\mathbf{x} = \mathbf{A}\mathbf{s} \tag{2.2}$$

where **A** is referred to as the mixing matrix. Note that the time index, n, has been dropped in (2.2) for aesthetic reasons, but it is implied that the signals in **x** and **s** are time-series. The dimensions of the matrices in (2.2) are $Jx1$, JxK and $Kx1$ for **x**, **A** and **s** respectively.

Signal separation methods derive estimates of the source signals, $\hat{s}_k[n]$, from the observed signals $x_j[n]$ without any knowledge of the mixing matrix or the source signals.

The purification of spot microphone signals is a signal separation problem. Consider the case where two instruments are recorded with two spot microphones. The signal model is illustrated in Figure 2.1 below.

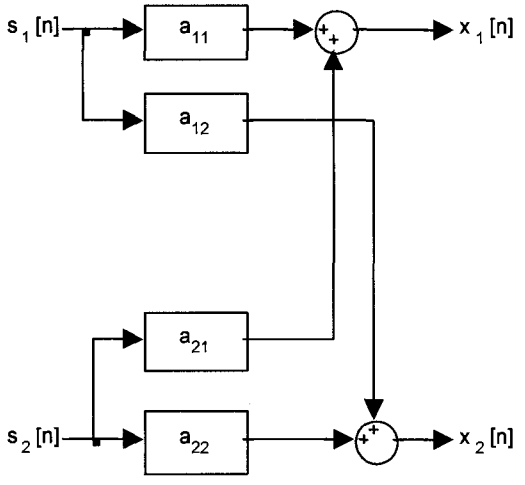


Figure 2.1. Signal Model of 2 Instrument, 2 Spot Microphone Recording Configuration

The signals $s_1[n]$, $s_2[n]$ are the pure signals from the instruments and the signals $x_1[n]$, $x_2[n]$ are the spot microphone signals. The mixing coefficients represent the impulse response between the instruments and the microphones, which may be modeled as a scaling factor or a FIR filter of arbitrary length. The purification of the spot microphone signals involves deriving estimates of the source signals from the mixed signals captured by the microphone.

The interference suppression problem addressed in this thesis is concerned with removing the crosstalk signals, $a_{jk}s_k[n]$, $(j \neq k)$, in the mixed signals $x_j[n]$. The direct impulse responses, a_{jj}, are not inverted in interference suppression. It is assumed that $A_{jj}(z)=Gz^{-D}$, where $A_{jj}(z)$ is the z-transform of the direct impulse response a_{jj}, G is a scalar gain and D is a delay. This is a reasonable assumption if the spot microphones are placed close the their target sources because any echoes will be overwhelmed by the direct signal. For most applications, it is not important to solve for G and D. The

interference suppression problem is a special case of the signal separation problem in which the direct path mixing coefficients, a_{ij}, are assumed to be known and equal to unity for practical purposes.

In the review of signal separation methods below, the methods are sorted and discussed under two general approaches: "blind source separation" (BSS) and "computational auditory scene analysis" (CASA). This chapter will then conclude with a description of the general approach discussed in this thesis and place it into context with other approaches.

2.2 Blind Source Separation Approach

The BSS approach involves the estimation of an unmixing matrix using the observed signal mixtures to recover estimates of the source signals. The BSS problem is illustrated in Figure 2.2 below, where the source signals and mixing matrix belong to a "black box".

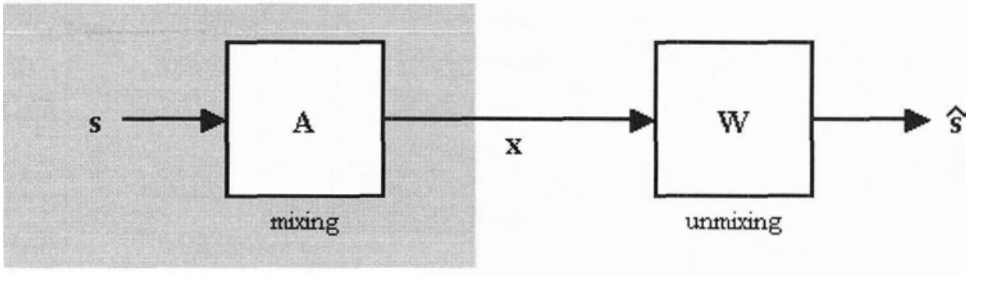


Figure 2.2: Blind Source Separation Problem

The "blind" descriptor in BSS refers to the fact that very little is known or assumed about the linear mixing matrix, **A**, or the source signals, **s**. The matrix **W** is referred to as unmixing matrix and the estimates of the sources are obtained by:

$$\hat{\mathbf{s}} = \mathbf{W}^H \mathbf{x} \qquad (2.3).$$

Substitution of the mixing equation, (2.2), into the unmixing equation, (2.3), reveals that the ideal unmixing matrix has the following property:

$$\mathbf{W}^H \mathbf{A} = \mathbf{I} \qquad (2.4).$$

If a matrix W satisfies (2.4), then the source estimate vector $\hat{\mathbf{s}}$ is equal to the source vector **s**.

BSS techniques seek an unmixing matrix, **W**, that transforms the observed mixed signals, **x**, into statistically independent signals, $\hat{\mathbf{s}}$ via (2.3). The fundamental assumptions of BSS techniques are:

- the unknown source signals, **s**, are realisations of ergodic random processes that are statistically independent
- a matrix **W** that transforms **x** into statistically independent outputs satisfies (2.4).

The assumption of statistical independence and ergodicity of the source signals is required to compute statistics of **s**, which are used either explicitly or implicitly by BSS algorithms to derive **W**. For a brief summary review of properties of random processes relevant to BSS, see Appendix A.

Some BSS algorithms compute the unmixing matrix in one pass from the statistics of **x**. More commonly, the unmixing matrix is derived iteratively based on minimisation of a cost function (or maximisation of a reward function), where the cost function is a function of the statistics of $\hat{\mathbf{s}}$ and is designed to maximise statistical independence of $\hat{\mathbf{s}}$. The iterative approach to derivation of the unmixing matrix is illustrated in Figure 2.3.

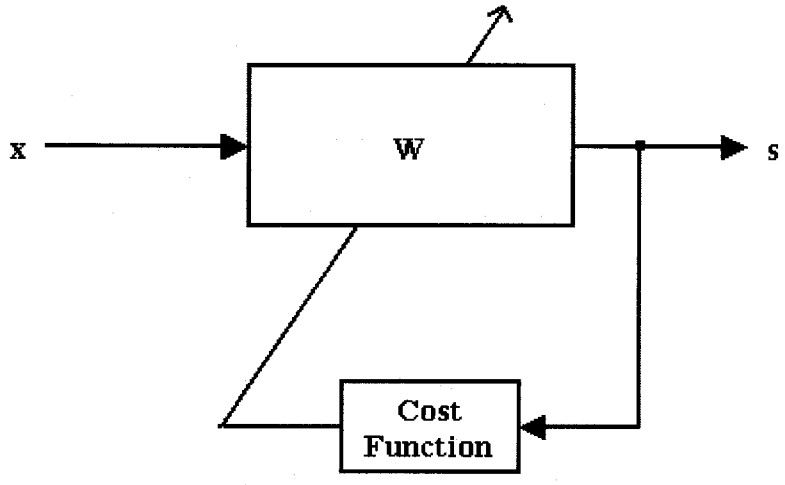


Figure 2.3. Iterative Derivation of Unmixing Matrix

Since nothing is known of **A** or **s** (see equation (2.2)), the estimates $\hat{\mathbf{s}}$ are only known up to a scaling factor and permutation because the energies and order of the signals can be encoded in either **A** or **s**. These uncertainties are often referred to as the "scaling ambiguity" and "permutation ambiguity" in the BSS literature.

While the mixing matrix is unknown, a particular BSS technique will typically make assumptions about the format of its elements. The elements of the mixing matrix are assumed to be either scalar or FIR polynomials. When all elements of the mixing matrix are scalar the mixture is called "instantaneous". When some or all of the elements of a mixing matrix are polynomials the mixture is called "convolutive".

Because so few assumptions are made about the mixing matrix and the source signals, BSS techniques are quite generic and have a myriad of applications. BSS techniques have been used to separate a variety of signal types, including communications, biomedical, image, financial and audio signals.

In the following overview, BSS techniques are classified into those that use only second order statistics and those that use higher order statistics. Techniques based on higher order statistics are by far more popular than those based on second order statistics because the latter produces source signal estimates that are decorrelated, but not necessarily statistically independent.

2.2.1 Techniques Based on Second Order Statistics

An unmixing matrix derived from second order statistics generates decorrelated output signals. Signals that are decorrelated have second order cross-statistics that are zero when the signals have zero-mean:

$$E[x_i(t)x_j(t+\tau)] = 0 \qquad (2.5).$$

Decorrelation is equivalent to second order statistical independence. It is not full statistical independence unless the signals are Gaussian. The higher joint moment statistics of two Gaussian random variables are zero for orders greater than two, provided that the cross-correlation and means are zero:

$$E\left[x_i^u(t)x_j^{*v}(t+\tau)\right]=0, \qquad u+v>2 \tag{2.6}$$

BSS techniques based on second order statistics are therefore suitable for Gaussian signals or when decorrelation is a sufficient criterion for signal separation.

2.2.1.1 Decorrelation

The mixed signals are decorrelated by algorithms that determine a matrix, $\mathbf{Q}^{-1}$, that reduces the off-diagonal elements of the covariance matrix of the mixed signals, $\mathbf{x}$, at lag 0 to zero:

$$\tilde{\mathbf{x}}=\mathbf{Q}^{-1}\mathbf{x} \tag{2.7}$$

The signals $\tilde{\mathbf{x}}$ are decorrelated. The covariance matrix is given by:

$$\mathbf{C}(\tau)=\begin{bmatrix} E[x_1(t)x_1(t+\tau)]-\bar{x}_1^2 & \cdots & E[x_1(t)x_N(t+\tau)]-\bar{x}_1\bar{x}_N \\ \vdots & \ddots & \vdots \\ E[x_N(t)x_1(t+\tau)]-\bar{x}_N\bar{x}_1 & \cdots & E[x_N(t)x_N(t+\tau)]-\bar{x}_N^2 \end{bmatrix} \tag{2.8}$$

In (2.8) stationarity to the second order was assumed. Further, if the means of the signals, $\bar{x}_n$, are assumed to be equal to zero, so then the covariance matrix reduces to the correlation matrix, which contains autocorrelations on the diagonal and crosscorrelations on the off-diagonals at lag τ. The matrix $\mathbf{Q}$ can be estimated using $\mathbf{C}(0)$ with different techniques, one of the most popular being principal component analysis (PCA).

The decorrelation matrix, $\mathbf{Q}^{-1}$, is the unmixing matrix, $\mathbf{W}^{\mathbf{H}}$, only under narrow conditions:

- the source signals are Gaussian
- the mixing matrix is unitary [13].

The above condition on the mixing matrix is very restrictive. In many cases the mixing matrix is not unitary and can be factored according to [14] as follows:

$$\mathbf{A}=\mathbf{QU} \tag{2.9}$$

This means the decorrelation matrix is not the full solution to the separation problem. The unitary matrix, **U**, is still unknown and cannot be determined without additional information or assumptions about the source signals or mixing matrix. Decorrelation, also called "whitening" or "sphering" in the literature is not usually a suitable stand-alone technique for signal separation. However it is a helpful, if not necessary, preprocessing technique used for BSS using higher order statistics [15] that are reviewed in section 2.2.2.

For non-stationary or coloured Gaussian source signals multiple decorrelation-based BSS techniques are used for separation. Constraining the mixing matrix can also provide the additional information required to solve the BSS problem using second order statistics. The remaining subsections will outline BSS techniques based on multiple decorrelations and possible constraints for the mixing matrix.

2.2.1.2 Multiple Decorrelations

The source separation problem can be solved for Gaussian signals using multiple decorrelations if the different correlations provide more information to constrain the solution. The additional constraints help in determining the full unmixing solution, rather than just the matrix **Q** in (2.9). When the mixing matrix has polynomial elements, there are even more unknowns than for a simple instantaneous mixing matrix. In this case joint decorrelation of many covariance matrices are required to obtain a solution.

If the source signals are non-stationary then multiple covariance matrices can be estimated at different times in the signal. The unmixing matrix is determined by algorithms that jointly diagonalise all the covariance matrices. Some examples of multiple decorrelation-based BSS techniques using non-stationary signals are found in [16] and [17].

If the source signals are coloured, then the covariance matrices at different time lags are non-zero and therefore provide more information to solve the separation problem. The unmixing matrix is determined by algorithms that jointly diagonalise the multiple covariance matrices obtained from different time lags. Some examples

of multiple decorrelation-based BSS techniques using coloured signals are found in [14] and [18].

2.2.1.3 Constraints on the Mixing Matrix

Another way to deal with the undetermined decorrelation problem is to make assumptions about and constrain the mixing matrix. For example, the authors in [19] suggest the following format for a convolutive, two-source mixing matrix:

$$\mathbf{A}(\omega)=\begin{bmatrix} 1 & \mathbf{A}_{12}(\omega) \\ \mathbf{A}_{21}(\omega) & 1 \end{bmatrix} \qquad (2.10).$$

If the direct path between sources and sensors, $\mathbf{A}_{ii}(\omega)$, is in fact convolutive, the constraint in (2.10) will result in separated signals that have the cross-talk removed, but are not deconvolved. Even with the constraint (2.10), knowledge of one of the cross paths $\mathbf{A}_{ij}(\omega)$ is still required to solve the problem using a single decorrelation.

Constraints or apriori information about the mixing matrix is not always an unreasonable requirement. Often some knowledge of the mixing system can be inferred from source-sensor geometry or measured directly given a single test signal. An example of an application where some knowledge of the mixing system can be obtained is the acoustic signal recording and separation problem that is the topic of this thesis. For sources that are not free to move in space, such as a harp or piano, the room transfer functions can be measured directly or estimated from the instrument-microphone geometry and room modeling.

2.2.2 Techniques Based on Higher Order Statistics

In section 2.2.1 we learned that the information in one covariance matrix was insufficient for solving the BSS problem, especially for convolutive mixtures. Higher order statistics can provide the additional information required to solve the separation problem. Besides, for source signals that are non-Gaussian, second order statistics are insufficient to achieve independent outputs. The probability density functions (PDF) of speech and music signals are super-Gaussian [20], having sharper peaks and

longer tails than Gaussian PDFs. To achieve signal separation for speech and music based on statistical independence, it is essential to consider higher order statistics.

The derivation of a linear transform that converts a set of dependent variables into a set of maximally independent non-Gaussian variables is known as independent component analysis (ICA). Since there are many ways to determine such a linear transform, there are many ICA methods. A nice overview of ICA methods is found in [15]. ICA is the most popular approach for BSS of audio signals, presumably because it is the most effective given the non-Gaussian nature of audio signal PDFs.

ICA methods are distinguished by:

- the cost function used to measure the statistical independence of the transform outputs
- the adaptation rule for updating the transform matrix (for iterative approaches).

The cost function is often referred to as the "contrast function" in the ICA literature. Because of the non-Gaussian PDF of the independent variables, all ICA methods involve explicit or implicit use of higher order statistics in their contrast functions.

Most BSS techniques based on ICA use an iterative approach to deriving the transform (unmixing) matrix, but there are some single-pass approaches as well. In the following section, some non-iterative approaches are reviewed, followed by the more common iterative approaches.

2.2.2.1 Non-iterative ICA

Cardoso describes an example of non-iterative ICA applied to BSS of instantaneous mixtures in [21]. The unmixing matrix is computed in one pass by first decorrelating the mixture and then diagonalising a matrix of fourth-order statistics of the decorrelated signals at zero-lag. This method minimises the second and fourth order cross-statistics of the output signals, achieving independence up the fourth order. This does not achieve full statistical independence, but goes one step further than decorrelation.

Shamsunder and Giannakis extend Cardoso's non-iterative ICA method to convolutive mixtures in [22]. The mixing matrix used by Shamsunder and Giannakis is simplified such that the diagonal elements are scalar and the off-diagonal elements are FIR filters. The *mixing* matrix is solved for in the frequency domain using fourth-order polyspectra, rather than fourth-order statistics used in Cardoso's time-domain method. The unmixing matrix is derived by inversion of the derived mixing matrix.

2.2.2.2 Iterative ICA

The contrast functions used for iterative ICA are designed to assess the statistical independence of the outputs of the current transform matrix. The contrast functions determine how the transform matrix is updated as well as to determine when the outputs are sufficiently independent. Note that the contrast functions are only able to provide an estimate of the degree of statistical independence of the outputs. Any limitations of the contrast functions in providing a true measure of independence will limit the degree of independence in the separated signals. For example, a contrast function based on fourth-order statistics will drive the ICA system to produce outputs that are fourth-order independent at best. This is not equivalent to full statistical independence.

Contrast functions estimate the "non-Gaussianity" of the outputs. Non-Gaussianity is equated with independence [15]. The rationale for this equivalence stems from the central limit theorem, which states that the sum of a large number of random variables, regardless of their PDFs, approaches a Gaussian distribution[1]. This means that the mixed signals, **x**, will be more Gaussian than the source signals assuming, as required for ICA, that the source signals are non-Gaussian. By adjusting the transform matrix to maximise the non-Gaussianity of the outputs, separation of the signals is achieved by increasing their statistical independence.

Contrast functions may be computed from higher order statistics explicitly or implicitly, via information theoretic principles. Examples of contrast functions that

[1] As described in [23], the central limit theorem guarantees that the sum of N random variables approaches a Gaussian *distribution* as N approaches infinity. This does not always guarantee a Gaussian PDF.

use higher order statistics explicitly include kurtosis and negentropy. Kurtosis and negentropy are formally defined in [15]. It suffices to state here that kurtosis and negentropy are estimates of non-Gaussianity, so these functions are maximised in the ICA iterations. An example of a contrast function that makes indirect use of higher order statistics is information maximisation, also defined in [15] and made popular by one of the seminal papers on ICA by Bell and Sejnowski [20]. A review of contrast functions for ICA can be found in [24].

The number of published contributions to ICA is vast. To provide some structure, the review of iterative ICA for BSS below is divided into instantaneous and convolutive mixtures.

2.2.2.2.1 ICA for Instantaneous Mixtures

Bell and Sejnowski [20] contributed one of the groundbreaking publications that described the use of ICA to separate instantaneous mixtures of non-Gaussian signals. Recall that instantaneous mixtures involve only scalars in the mixing matrix. This ICA method made use of the information-theoretic principle of information maximisation in the contrast function. The authors reported success in separating mixtures of up to ten signals. The deficiencies of this ICA method include its limitation to stationary signals, its sensitivity to noise in the signal mixtures and its limitation to instantaneous signal mixtures. The limitation of Bell and Sejnowski's method to instantaneous mixtures makes it impractical for the separation of recorded audio signals. Audio signal mixtures will necessarily involve delays at the very least and possibly more complicated FIR filters if recorded in a reverberant environment.

2.2.2.2.2 ICA for Convolutive Mixtures

In [25] and [26] Torkkola describes an adaptation of Bell and Sejnowski's ICA method to address convolutive signal mixtures consisting of delays in the cross-paths and convolutions in the direct- and cross-paths respectively, as illustrated in Figure 2.4.

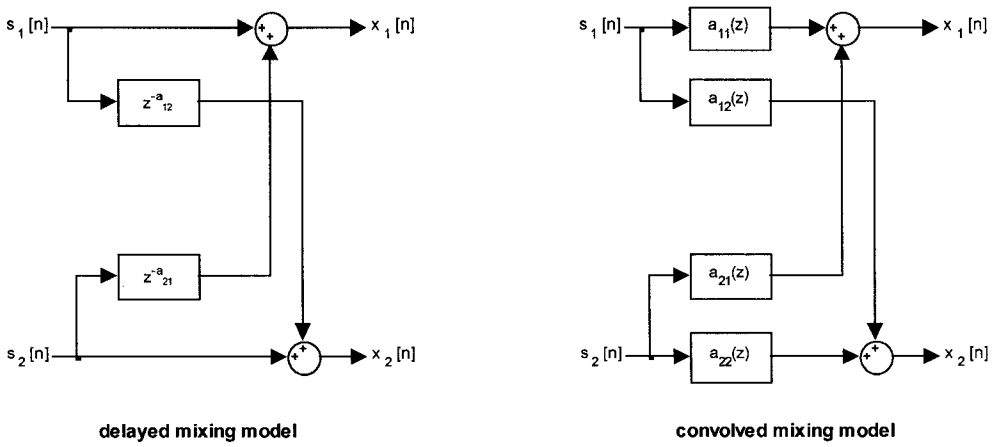


Figure 2.4. Mixing Models for Torkkola's ICA Methods for Convolutive Mixtures

Engebretson describes another example of an ICA method for convolutive mixtures of audio signals in [27]. This method assumes a mixing model similar to Torkkola's delayed mixing model in Figure 2.4 but with arbitrary FIR filters in the cross-paths. Engebretson's ICA method finds an unmixing matrix that minimises fourth-order cross-statistics of the output signals, assuming the source signals are zero-mean.

Some authors have applied instantaneous mixture ICA methods in the frequency domain to reduce the computational complexity required to derive convolutive unmixing transforms [28]-[30]. Convolution in the frequency domain is equivalent to instantaneous mixing at each frequency point. The discrete Fourier transform (DFT) of the mixed signals at bin k is given by:

$$\mathbf{X}(k,n) = \mathbf{H}(k)\mathbf{S}(k,n) \qquad (2.11),$$

where $\mathbf{H}(k)$ is the filter frequency response matrix, an instantaneous mixing matrix with complex scalar elements. Instantaneous ICA methods are applied to each channel k, of the DFT of the signal mixtures. Because of the permutation and scaling ambiguities present in BSS estimates, care must be taken to ensure the same scaling factor and the correct association of all DFT channels in reconstruction of the time-domain signal estimates.

2.3 Computational Auditory Scene Analysis Approach

In the computational auditory scene analysis (CASA) approach to the sound separation problem, certain features in the mixed signals are identified and grouped as belonging to a particular source in the mixture. The source signals are constructed using the information in its assigned features. CASA has an advantage over BSS: it is possible to separate M sources from N signal mixtures, where $M > N$. BSS techniques require $M \geq N$ signal mixtures to successfully separate the sources. In order to extract relevant features for sound separation, CASA techniques make the assumption that the underlying source signals come from sound generators, for example musical instruments. This assumption makes CASA techniques tailored to audio signal separation whereas BSS techniques are more generic with respect to source signal types.

CASA is inspired by our knowledge of how the human hearing system is able to identify and isolate one sound source within a mixture. An example of such sound separation by the human hearing system is the ability to track the speech of one speaker out of several speakers talking simultaneously. Bregman described the sound source separation phenomena in the human hearing system as "auditory scene analysis" (ASA), the aural analogue to object segregation in an image [31]. Bregman hypothesised that a number of principles are used to segregate an auditory scene into different sound objects. Some of these principles are:

1. **Regularity in Harmonic Structure**: frequencies that have a harmonic relationship at a particular point in time belong to the same source
2. **Regularity in Amplitude Trace**: frequencies that have similar trends in amplitude evolution over time belong to the same source
3. **Regularity in Frequency Trace**: frequencies that have similar modulation trends over time belong to the same source.

In CASA, mixed signals are analysed by machine to extract features of the signals that are useful for locating such regularities and these features are grouped and assigned to different sources. The analysis of signals is typically a form of time-

frequency analysis. The most common form of time-frequency analysis for CASA-based separation methods is the short-time discrete Fourier transform (STDFT). Time-frequency regions are then grouped into sources using to Bregman's regularity principles.

The review of CASA-based audio signal separation techniques will begin with a brief discussion of the assumed audio signal model. Once grouping of time-frequency regions based on the signal model and ASA principles is complete there are two general strategies for constructing the source signal estimates: time-varying filtering and signal resynthesis. Examples of each signal construction strategy are reviewed.

2.3.1 Audio Signal Model

A general model for a pitched discrete-time audio signal is given by:

$$s(nT)=\begin{cases} u(nT) & \text{, transient regions} \\ \sum_{k=1}^{N} A_k(nT)\sin\left(2\pi\left(f_k(nT)\right)nT+\phi_k\right)+w(nT) & \text{, steady-state regions} \end{cases} \quad (2.12),$$

where T is the sampling period. A pitched audio signal is one where the sinusoidal components in (2.12) dominate both in time duration as well as power spectral distribution.

The transient regions, $u(nT)$, are short-lived relative to the steady-state regions. The transient regions are not deterministic. They have broadband, continuous spectra that may be due to either impulsive or noise-like time-domain characteristics. Examples of impulsive transients include plosives in speech and the pluck of a stringed instrument. Noise-like transients occur most commonly in speech and singing as sibilance.

The steady-state regions are much longer lived than transient regions, particularly in musical audio signals. They are dominated by the deterministic sinusoidal components that have slowly time-varying amplitudes and frequencies. Typically these sinusoidal components are harmonically related, although it is possible that non-

linearities in the sound generator may distort this harmonic relationship somewhat. The steady-state regions also have an aharmonic component, $w(nT)$, which has a much lower power spectral distribution than the sinusoidal components. The aharmonic component is not deterministic and has noise-like characteristics with a continuous spectrum. Examples of steady-state regions include voiced speech and the sustained oscillations following the plucking of a string of a musical instrument. Voiced speech will have a more energetic aharmonic component, owing the speaker's breath, than the ringing stringed instrument.

CASA-based audio signal separation techniques focus on identifying, grouping and separating the sinusoidal components of steady-state regions of the audio signals. Some reasons for separating only the sinusoidal components include:

- sinusoidal components dominate the signal,
- sinusoidal components are sparse and often don't overlap in time-frequency, simplifying the separation problem to isolating and grouping the appropriate time-frequency cells, and
- the other regions and components are difficult to separate because of their stochastic nature and spectral density.

The sinusoidal components are typically identified using STDFT-based methods such as McAulay and Quatieri's sinusoidal modeling technique [32], which is reviewed in Chapter 3 of this thesis. Grouping of sinusoidal components is done using ASA regularity principles already presented. Separation is achieved either by

- time-varying filters to remove sinusoidal components of undesired sources and/or to enhance sinusoidal components of the desired sources or
- resynthesis of each source from its sinusoidal representation.

Each approach to separation is discussed next under its respective heading.

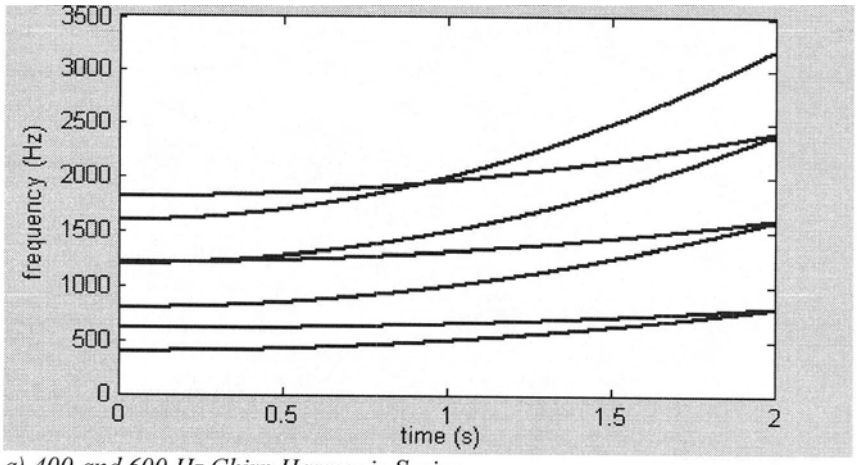
2.3.2 Separation by Time-Varying Filters

Once time-frequency regions have been identified as "desired" or "undesired" using CASA-based techniques, one way of separating out the desired regions is by time-varying filters that are designed to either enhance desired regions and/or attenuate

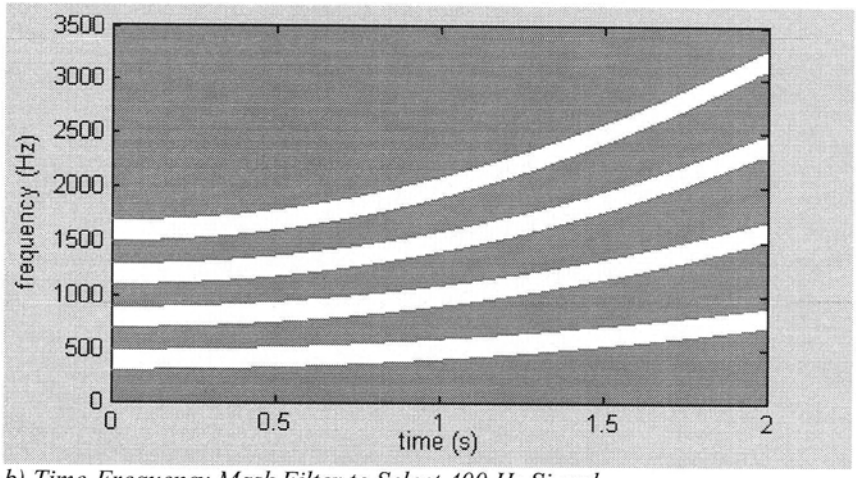
undesired regions. An example of separation by time-varying filters is illustrated in Figure 2.5. It is desired to separate one time-varying harmonic series that begins with a fundamental frequency of 400 Hz from another that begins at 600 Hz. The sinusoidal frequency trajectories of the mixed signals is shown in Figure 2.5 a). The ideal response of the time-varying filter designed to isolate the 400 Hz signal is shown in Figure 2.5 b), where dark areas indicate attenuated time-frequency regions and light areas indicate passed time-frequency regions. Such a filter is referred to as a time-frequency "masking" filter in some of the literature. Figure 2.5 c) shows the masking filter superimposed on the signals. The signal trajectories lying in the dark regions of the filter response are attenuated.

The time-varying masking filter can be implemented in different ways. Roweis describes a system for separating multiple audio signals from one microphone signal in [33] by time-varying gains applied to different sub-bands of the mixed signal. Examples of time-frequency masking by scaling bins of the STDFT are found in [34]-[36]. These references also include interesting methods for the identification of "desired" and "undesired" time-frequency regions.

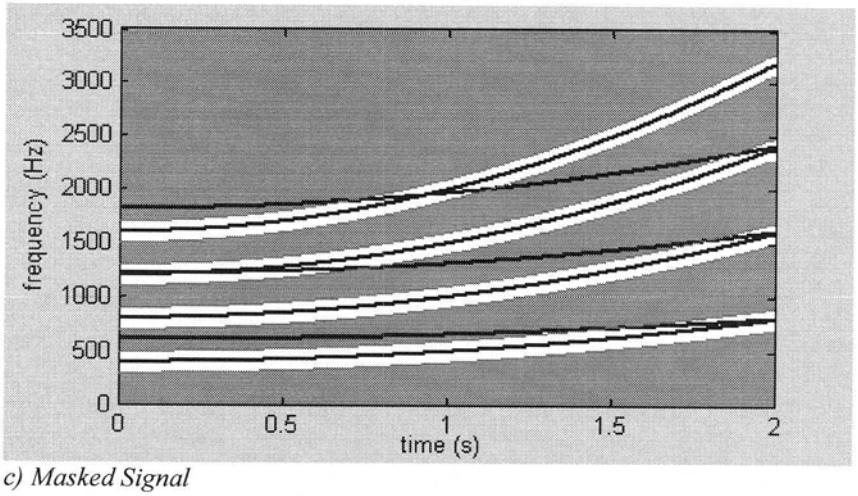
The problem of time-frequency collisions between desired and undesired signals is not elegantly addressed by time-frequency masking separation methods. In the event of a collision, there are two options: include or exclude the region. If the collision region is included, leakage from the undesired signal(s) results. If the collision region is excluded, some of the desired signal is lost. While there are many methods to detect colliding sinusoidal trajectories (discussed in the next section), time-frequency masking methods cannot separate the colliding trajectories because the bandwidth of the filters cannot be made sufficiently narrow. Separation based on signal resynthesis, discussed in the next section, is able to separate colliding time-frequency regions if the collisions are adequately detected.



a) 400 and 600 Hz Chirp Harmonic Series



b) Time-Frequency Mask Filter to Select 400 Hz Signal



c) Masked Signal

Figure 2.5. Time-Frequency "Mask" Filtering

2.3.3 Separation by Sinusoidal Resynthesis

If desired time-frequency entities derived by CASA-based techniques are the time-varying sinusoidal functions in (2.12), the desired signal can be separated from the mixture by reconstruction based on its identified sinusoidal components. Signal resynthesis based on its time-varying sinusoidal representation is illustrated in Figure 2.6.

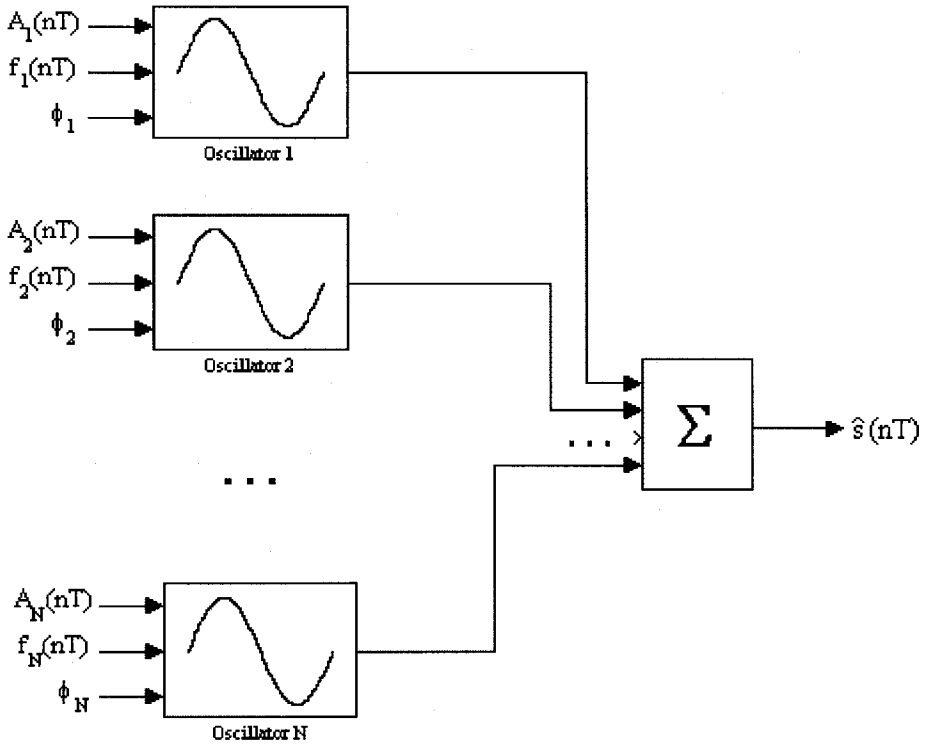


Figure 2.6. Signal Resynthesis from its Sinusoidal Representation

The signal $\hat{s}(nT)$ is the estimate of the desired source signal. The resynthesis method illustrated in Figure 2.6 consists of a time-domain sinusoidal oscillator bank with each oscillator controlled by the time-varying amplitudes and frequencies and initial phase of the desired sinusoidal components identified by the CASA-based analysis and grouping. It is possible to accomplish the sinusoidal resynthesis more efficiently in the frequency domain.

Examples of CASA-based audio signal separation methods using sinusoidal resynthesis are found in [9] and [37]-[39]. The approaches differ in how desired sinusoidal components are identified and the method used for resynthesis.

Because separation by resynthesis does not involve manipulation of the mixed signal, it is possible to separate time-frequency regions that are occupied by more than one source signal if a good estimate of the desired signal's sinusoidal representation is available in these collision regions. The frequency resolution of the DFT is given by:

$$|\Delta f| \geq \frac{n_{-3dB} \times f_s}{N} \qquad (2.13),$$

where Δf is the frequency resolution in Hz, f_s is the sampling frequency in Hz, N is the number of data samples used in computing the DFT and n_{-3dB} is the –3 dB bandwidth of the window function used in computing the DFT in number of bins. The bandwidth, n_{-3dB}, is not restricted to be an integer number of DFT bins. The –3 dB bandwidths for some of the most common window types are given in [40]. If two simultaneously occurring sinusoids are spaced closer than (2.13), they are not distinguishable in the DFT. In order to obtain the required information about the colliding sinusoids, specialised techniques are used. Approaches to determining the parameters of colliding sinusoids include estimation of a demixing matrix for the narrowband collision region ([41]), fitting of models of two sinusoids ([8]), inferring of parameters obscured in collision regions using surrounding data and models ([39], [42], [43]) and narrowband filters ([44]).

A disadvantage of using sinusoidal resynthesis to reconstruct source signal estimates is that the quality of the result is dependent on the quality of the estimated sinusoids and their parameters. Furthermore, even if perfect estimation of the sinusoidal components of the desired source signal is possible, the resynthesised estimate will only contain the sinusoidal part of the true source signal. Transients and the aharmonic components in the signal model of (2.12) are not recovered in the estimated source signals.

2.4 Approach Taken in this Thesis

The review of signal separation methods in the preceding sections shows several different approaches to the problem. In determining a suitable approach to apply to the spot microphone signal purification problem, it is important to consider the problem parameters and requirements. The purification problem parameters and requirements are discussed in the first section below. These provide a rationale for the approach and methods explored in this thesis, which are discussed in the second section.

2.4.1 Problem Parameters and Requirements

In selecting an approach for the problem of the purification of close-miked pitched musical signals a number of factors were considered:

1. the nature of source signals,
2. the nature of mixed signals and
3. the desired properties of the purified (output) signals.

The sound sources are assumed to be pitched musical instruments, including the possibility of the singing voice. The signals generated by such sources are assumed to fit the signal model assumed for CASA-based separation approaches (see section 2.3.1). Pitched musical signals are dominated energetically and temporally by deterministic, slowly evolving sinusoids.

The mixed signals picked up by the spot microphones will most likely be dominated by the desired source signal due to their proximity to the proximity of the spot microphone to the target instrument. Accordingly one would expect typical signal-to-interference ratios greater than 0 dB. This may not always be true in a musical performance because:

- the source instrument may not always be playing when other instruments are playing and
- the interpretation of the musical piece may require that the target instrument is played much quieter than the other instruments.

As stated in the introduction to this chapter (section 2.1), the mixed signals are assumed to consist of the unaltered target instrument and cross-talk from the other instruments convolved by the impulse response of the acoustic space. In BSS terminology, this translates to a mixing matrix where the diagonal elements are unity and the off-diagonal elements are FIR filters.

The intended application of the purified source signals is for mixing the final recording of the musical ensemble. The final mix may consist only of the purified source signals or the purified source signals may be "mixed-onto" the main microphone signals which contain the target instrument. In the mix-onto application, it is important that the purified source signal be a scaled version of the target instrument signal contained in the aggregate main microphone signal. In other words, the phase of all components of the purified source signal must be matched to the phase of the respective components in the aggregate signal. The purified source signal is referred to as "coherent" with respect to the aggregate signal when the phases are matched. The mix-onto application of coherent, purified source signals is illustrated in Figure 2.7.

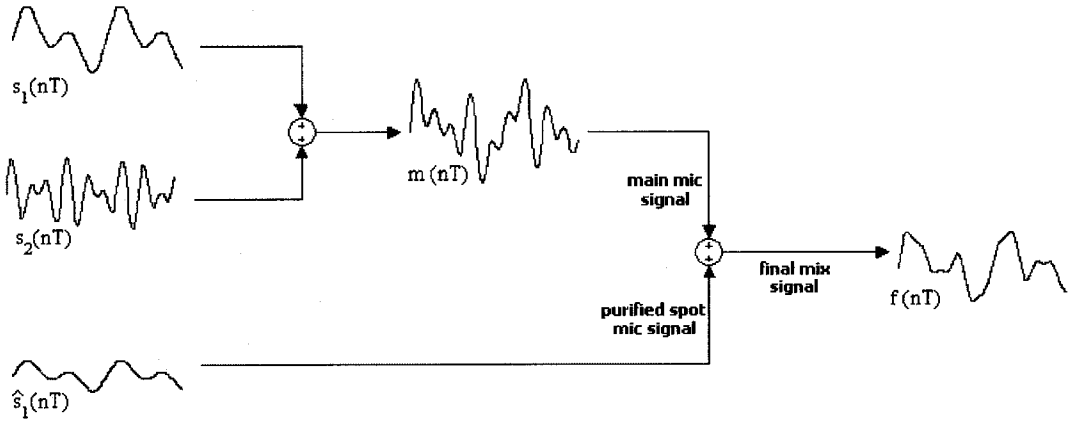


Figure 2.7. Mix-Onto Application Using a Coherent Purified Spot Microphone Signal

The signal $\hat{s}_1(nT)$ is the coherent purified spot microphone signal for source 1, *m(nT)* is the aggregate main microphone signal, consisting of signals from sources 1 and 2 and *f(nT)* is the final mix signal. Note that $\hat{s}_1(nT)$ is phase-matched to the corresponding signal in *m(nT)*; it is a scaled version of the signal from source 1.

In addition to the requirement of coherence in the purified output signals, there are other desirable properties in the output signals:

- purified output signals should contain all parts of the source signal, including transients, aharmonic and sinusoidal parts, and
- purified output signals should not contain processing artefacts that are audible in the final mix.

Due to such strict desired properties in the purified output signals, the interference suppression methods explored in this thesis were designed to be very conservative: the quality of the desired signal should not be compromised by the suppression of the undesired signals.

2.4.2 Approach and Methods

Since the type of source signals addressed in this thesis are dominated by deterministic, slowly evolving sinusoids, it seemed most sensible to take a CASA-based approach to deal with these dominant components rather than a BSS approach, which assumes no knowledge of the source signals. The general CASA-based approach was reviewed in section 2.3 and is summarised in Figure 2.8.

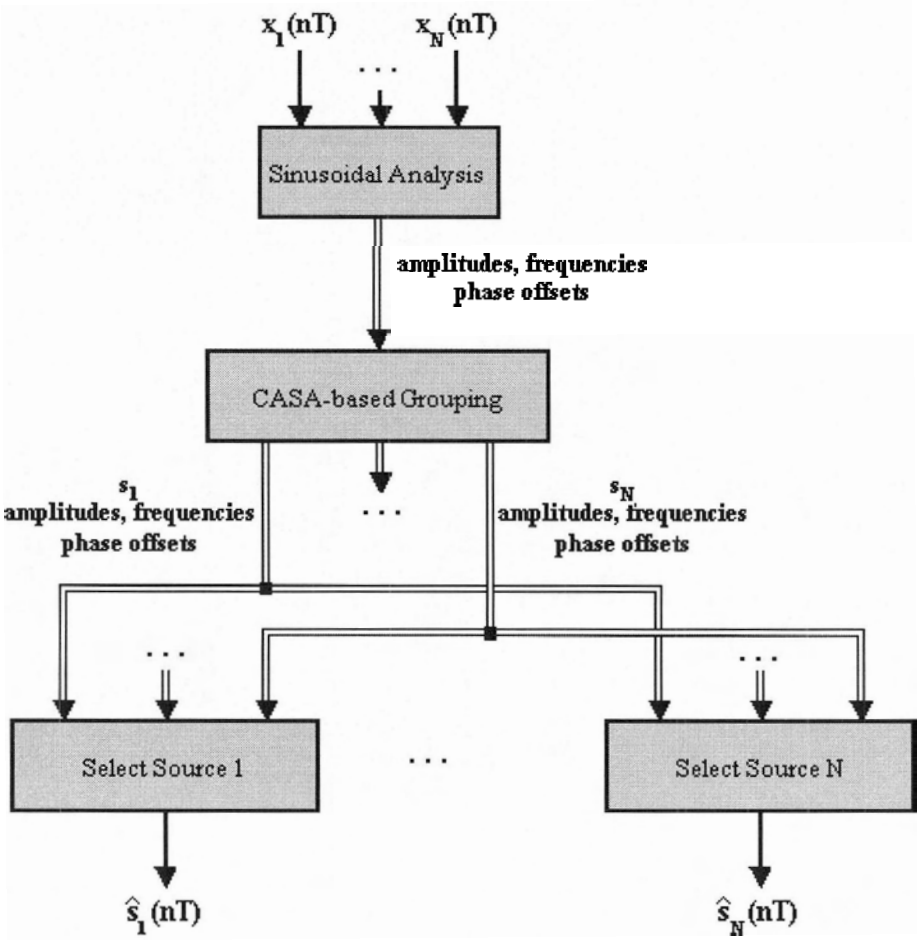


Figure 2.8. CASA-based Interference Suppression

Due to the need for high quality, purified audio signals, $\hat{s}_i(nT)$, a very conservative approach was taken in the source selection step. Two methods for source selection were explored:

1. suppression of undesired sinusoids using time-varying narrowband notch filters and
2. subtraction of resynthesised undesired sinusoids from the mixed signals, $x_i(nT)$.

These methods for source selection are "conservative" because they remove only undesired sinusoidal components, leaving the rest of the signal intact. The reasoning behind the use of this method was that by removing the dominant components of the undesired signals the majority of the interference suppression problem would be

solved. However, from an aesthetic point of view, this solution is incomplete because the transient components of the undesired signals remained in the purified output causing objectionable impulsive bursts. Accordingly a transient suppression mechanism was included in the source selection to remove some of the undesired transients.

The transient suppression mechanism is also based on sinusoidal resynthesis, but this time, the entire composite signal is replaced by the resynthesised, desired sinusoids over the duration of the undesired transient. This method for transient suppression is only useful for cases where undesired transients occur during sinusoidal regions of the desired signal.

A general review of sinusoidal analysis and resynthesis is presented in Chapter 3 as background information. The details of the sinusoidal analysis, CASA-based grouping and source selection are given in Chapter 4. Results of the methods applied to various musical signals are presented in Chapter 5. While the approach and methods explored in this thesis go a long way to providing nice estimates, $\hat{s}_i(nT)$, there are many ways in which these signals may be improved. Possible strategies for improvement may involve the BSS approach for handling outstanding issues with purification of the stochastic components. Suggestions for future directions are presented in Chapter 6.

Chapter 3

Sinusoidal Modeling of Audio Signals

3.1 Introduction

Sinusoidal modeling of a real-valued signal consists of representing the signal in terms of a sum of cosine functions with time-varying amplitudes, frequencies and initial phases. Such a model is rooted in the fact that an arbitrary real-valued signal with finite energy on an interval (t_1, t_2) can be represented by the trigonometric Fourier series:

$$s(t) = \sum_{n=0}^{\infty} c_n \cos\left(2\pi \frac{n}{t_2 - t_1} t + \phi_n\right) \qquad (3.1).$$

A derivation of (3.1) can be found in many textbooks on signals, including [45]. Equation (3.1) is valid for all signals, including non-periodic signals. This means that we can represent all components of our audio signals, transient, aharmonic and harmonic, as a sum of cosines with fixed amplitudes, c_n, and phase offsets, φ_n, over arbitrary time intervals. In practice, the number of cosines required for such a representation is not practical for transient and aharmonic components because they have a very dense frequency distribution. On the other hand, the sinusoidal components of audio signals have a sparse frequency distribution and do not require a very large number of cosine functions for their representation. When $s(t)$ is a harmonic series with period T and band-limited to f_{Max} equation (3.1) simplifies to:

$$s(t) = \sum_{n=0}^{n_{max}} c_n \cos\left(2\pi \frac{n}{T} t + \phi_n\right) \qquad (3.2),$$

where t_2-t_1=T and $n_{max} \leq f_{max}T$ and the coefficient c_n is the amplitudes of the n^{th} harmonic. A sinusoidal model is typically used to represent the slowly-evolving

sinusoidal components of an audio signal because these components are easily represented by a small number of cosine basis functions that are relatively static over short time intervals.

McAulay and Quatieri contributed one of the seminal papers that described a sinusoidal model for speech signals [32]. Shortly thereafter, Smith and Serra published a similar model for musical signals [46]. There have been many extensions to the sinusoidal model to include the aharmonic and transient components of the signal, for example [47] and [48]. This chapter will focus on reviewing the basic sinusoidal model since this is the basis of the interference suppression methods explored in this thesis.

The derivation of the sinusoidal model is referred to as sinusoidal analysis. The synthesis of signals based on the sinusoidal model is referred to as sinusoidal synthesis. This thesis makes use of sinusoidal analysis and synthesis. The sinusoidal analysis is done by a software utility developed by Serra [49]. The remaining sections of this chapter review sinusoidal analysis, synthesis and some of the sinusoidal modeling software programs.

3.2 Sinusoidal Analysis

The sinusoidal model is derived in three steps:

1. computation of the STDFT over short intervals of data (frames),
2. peak detection of the STDFT magnitude for each frame and
3. linking of STDFT magnitude peaks over time

A peak consists of four pieces of information derived from the STDFT: a time stamp of the frame and the frequency, phase and amplitude. Peaks are linked across frames by their amplitude and frequency similarity. A time-series of linked peaks is sometimes referred to as a "peak trajectory" or "track" in the literature and is considered to represent a stable, slowly-evolving sinusoidal component of the signal. The sinusoidal model is the collection of all tracks found in the signal. The three steps of sinusoidal analysis are reviewed in the following sections.

3.2.1 Computation of the STDFT

The process of computing the STDFT and parameters used for computation is illustrated in Figure 3.1 below.

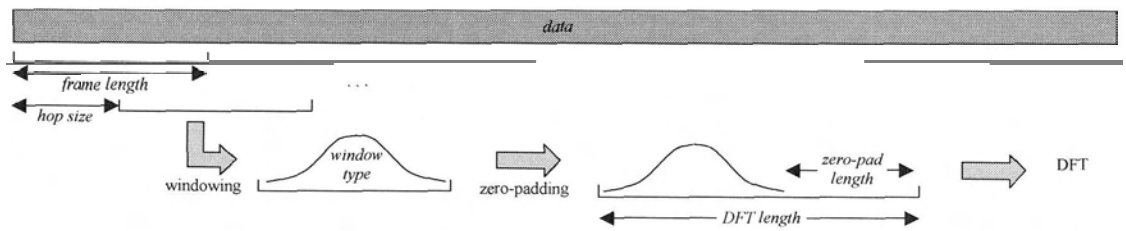


Figure 3.1. Computation of the STDFT with Parameters Shown in Italics

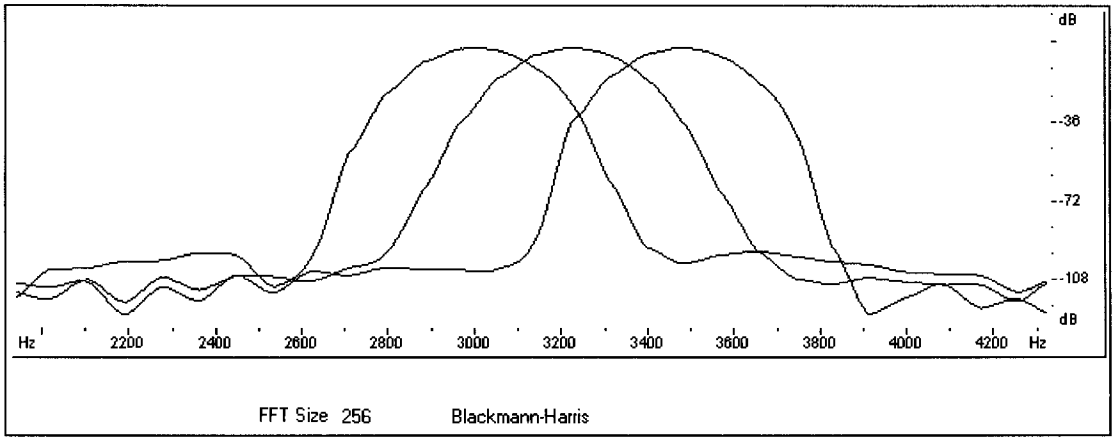
The STDFT parameters and their influence on determining DFT peak parameters is discussed in the sections below.

3.2.1.1 Frame Length

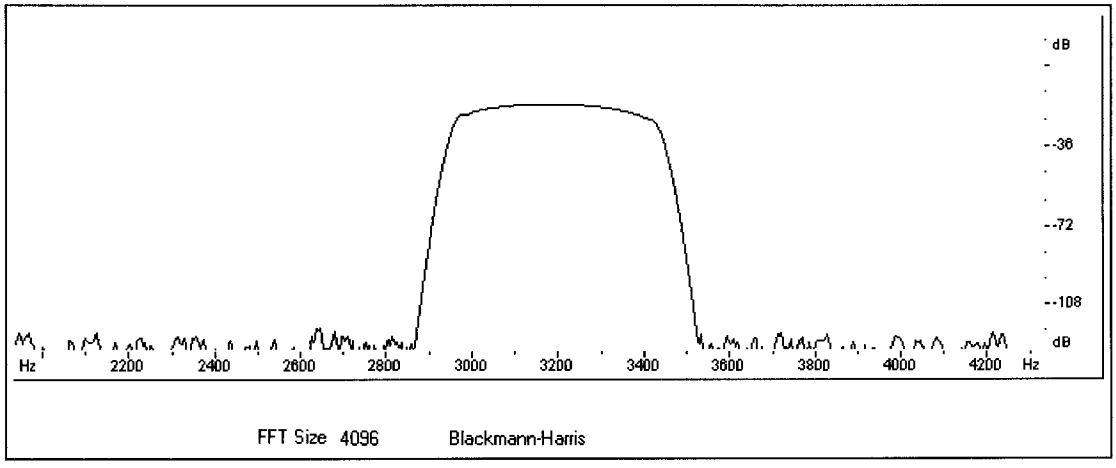
The frame length is the amount of data used to compute the DFT per frame. For time-varying signals use of long frame lengths to obtain high frequency resolution comes at the expense of lower time resolution. If long frames are used, time-varying parameters of the sinusoids, such as frequency and amplitude, are averaged over the duration of the frame. The effect of long versus short frame lengths is illustrated in Figure 3.2 using a vibrato (frequency-modulated) signal, centred at 3200 Hz. The vibrato period is 222 ms. Figure 3.2 a) shows the magnitude of three DFT computed using a frame length of 256 samples around the trough, centre and peak of the signal frequency. The low resolution in frequency is evident in the curves, but the peak of the DFT magnitude clearly follows the evolution of the signal frequency. Figure 3.2 b) shows the DFT computed using 4096 points, which nearly covers an entire vibrato period given the signal sampling rate of 22.05 kHz. The frequency resolution is increased (refer to the noise floor) but the location of the peak is smeared.

A good choice for the frame length is important for sinusoidal analysis: sufficient frequency resolution is necessary to distinguish between closely-spaced sinusoids, but the frame length should not be so large as to smear estimates of the evolving

sinusoidal parameters. McAulay and Quatieri recommend a frame length of at least 2.5 periods of the lowest expected fundamental frequency assuming a Hamming window is used. The data length can be reduced somewhat with special processing techniques, for example [50].



a)



b)

Figure 3.2. DFT Magnitude of Vibrato Signal Sampled at 22.05 kHz Using Different Frame Lengths: a) 256 samples, b) 4096 samples

Sometimes the frame length is adapted to be an integer number of periods of the current estimated fundamental frequency of the signal [51]. This is referred to as "pitch-synchronous analysis" and results in more accurate estimates of the parameters of sinusoidal components that are harmonics of the fundamental frequency. Pitch-

synchronous analysis is useful for monophonic signals that have only one harmonic series.

3.2.1.2 Hop Size

The hop size determines how often the DFT, and consequently sinusoidal parameter estimation is made. The hop size should be short enough to track the frequency and amplitude changes that are typical in musical signals. Some of the most rapidly varying amplitude and frequency changes are found in tremolo and vibrato, where oscillation rates can get as high as 9 Hz. The hop size can be made arbitrarily short (down to as low as one sample) at the cost of increased computations.

3.2.1.3 Window Type

The type of window applied to the frame of data has an impact on the quality of the peaks observed in the DFT magnitude spectrum. The choice of window is always a compromise between main lobe width and sidelobe height. A narrow main lobe width is desirable for distinguishing closely spaced peaks. Low sidelobes are desirable to reduce the effect of spectral "leakage". If the sidelobes are high, then peaks are corrupted by surrounding and distant frequency components. Unfortunately a window cannot have a narrow main lobe and low sidelobes: if the main lobe is narrow, then the sidelobes are high and vice versa. This concept is illustrated in Figure 3.3 where the DFTs of a 400 Hz sine wave were computed using triangular and Blackmann-Harris windows. The triangular window has a narrow main lobe and high sidelobes while the Blackmann-Harris window has a wide main lobe and low sidelobes.

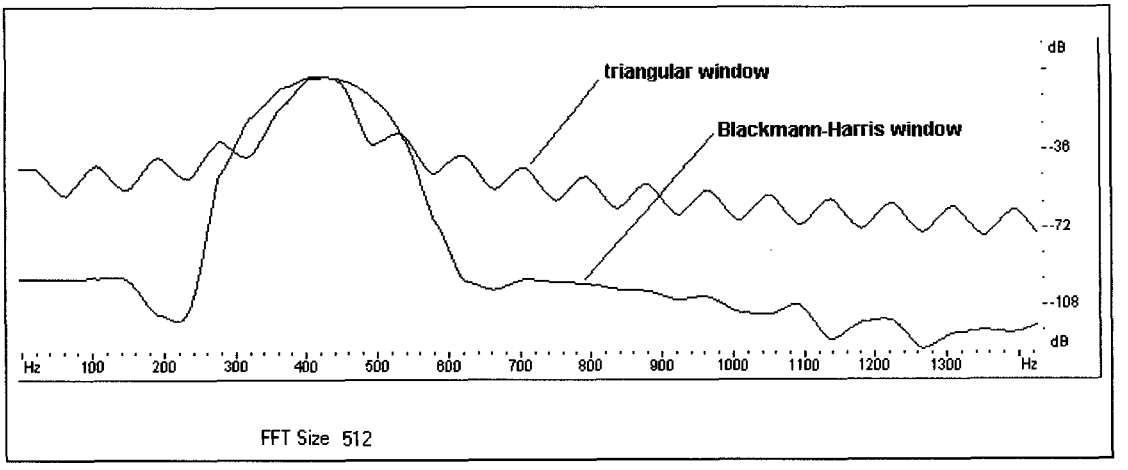


Figure 3.3. DFT Magnitude of a 400 Hz Sine Wave Using Different Window Types

When attempting to find peaks in the DFT magnitude spectrum it is important to consider the effect of the window on the spectral resolution. In particular, the –3 dB bandwidth of the main lobe will influence how close two sinusoids can be before they become indistinguishable in the magnitude spectrum. Harris nicely described the effect of windowing on the DFT and catalogued relevant parameters for many different windows [40].

3.2.1.4 Zero-Padding

The effect of zero-padding is interpolation of the discrete DFT spectrum. Zero-padding increases the number of DFT bins, but not the spectral resolution which is governed by the window type and the number of data points under the window. The effect of zero-padding is illustrated in Figure 3.4 for a DFT computed on a signal containing two closely-spaced sines using a Blackmann-Harris window.

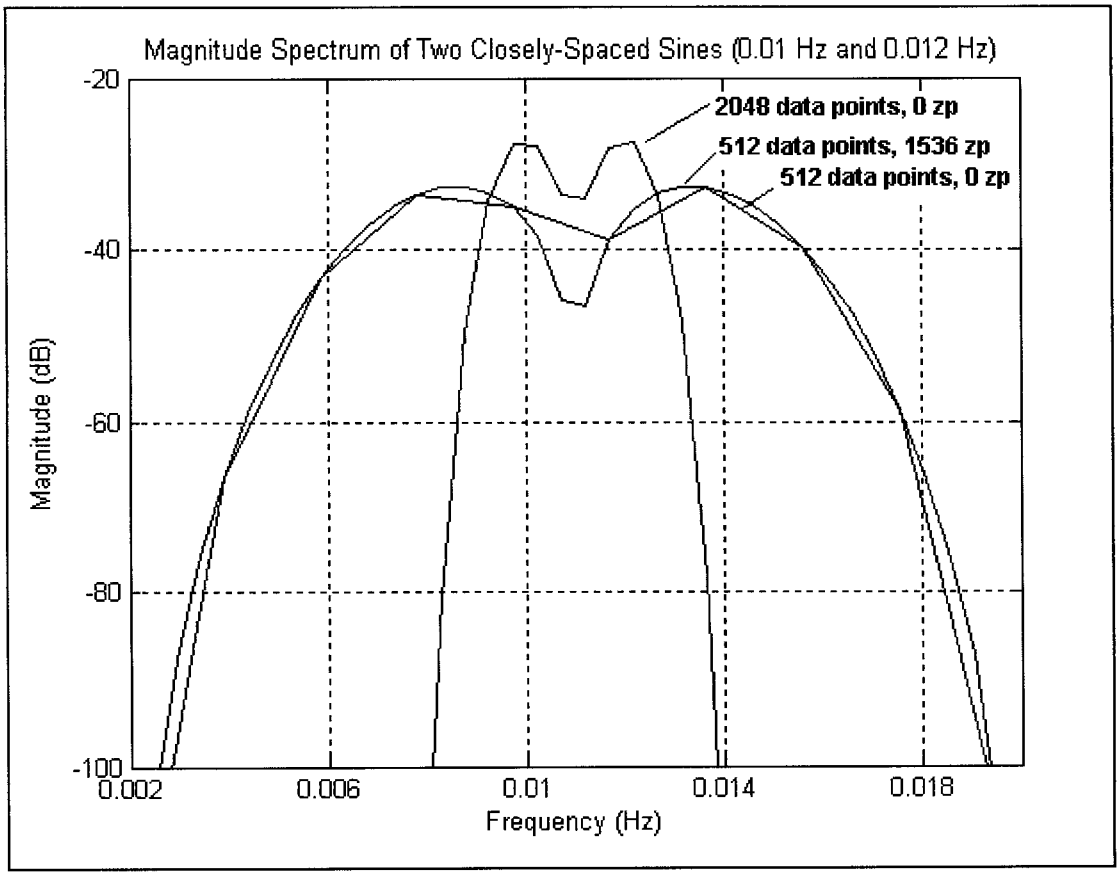
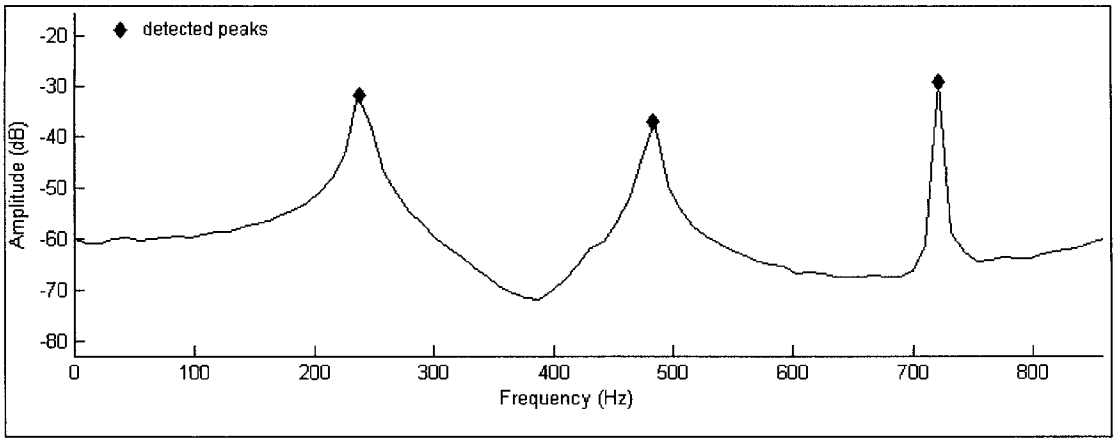


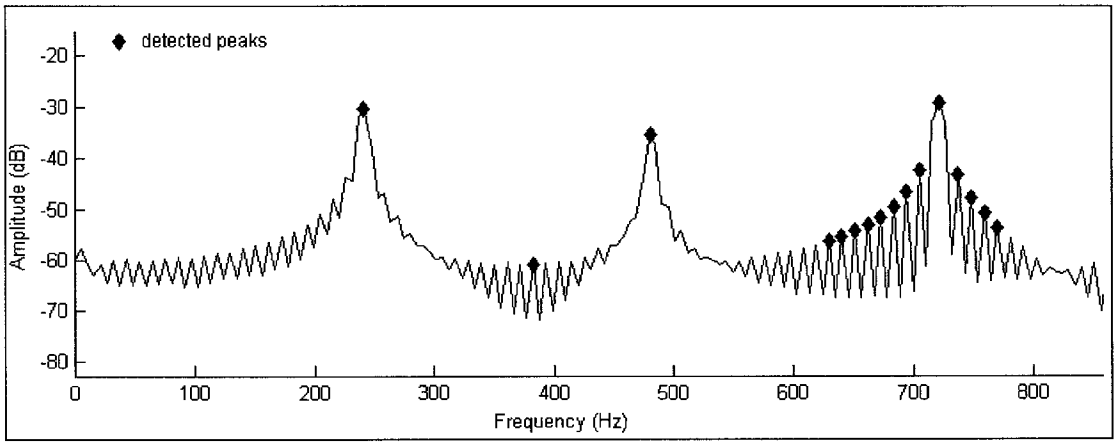
Figure 3.4. DFT Magnitude of Two Closely-Spaced Sines with and without Zero-padding

Zero-padding the windowed 512 data points before computation of the 2048 point DFT results in a magnitude spectrum that is upsampled by a factor of four. The window bandwidth does not change, so the peaks remain broad even with zero-padding. If 2048 points are included under the window, the window bandwidth is reduced and the peaks are sharper.

For sinusoidal analysis, the purpose of computing the DFT is to locate peaks in the magnitude spectrum. For such an application of the DFT, zero-padding can lead to false peaks around the main sinusoidal peak due to the sidelobes of the window. The sidelobe peak structure is exposed by zero-padding, as illustrated in Figure 3.5 below.



a)



b)

Figure 3.5. Peak Detection on DFT Magnitude Spectrum of a Cello Signal Computed using 2048 Data Points and a Hann Window: a) no zero-padding, b) zero-pad length of 2048 samples

Large amounts of zero-padding are unnecessary for detecting magnitude peaks if a good peak interpolation scheme is used, as described in the next section.

3.2.2 Detection of STDFT Magnitude Peaks

Peaks are detected on the DFT magnitude spectra with logarithmic amplitude scaling. A minimum threshold is set for the difference between peaks and neighbouring valleys so that only significant peaks are logged. The parameters estimated from the peak include:

- amplitude: the height of the peak in the magnitude spectrum,
- frequency: the location of the peak along the frequency axis,
- phase: the phase corresponding to the peak's frequency from the phase spectrum and
- time: the time the peak occurred in the signal, set to the time corresponding middle of the analysis frame.

To improve the estimate of the peak amplitude and location, a parabola is fit to the peak DFT magnitude sample and its neighbours to the left and right [46]. The magnitude and location of the parabola is taken as the interpolated peak amplitude and frequency. The peak phase is interpolated linearly from the phase spectrum. Parabolic peak interpolation is illustrated in Figure 3.5.

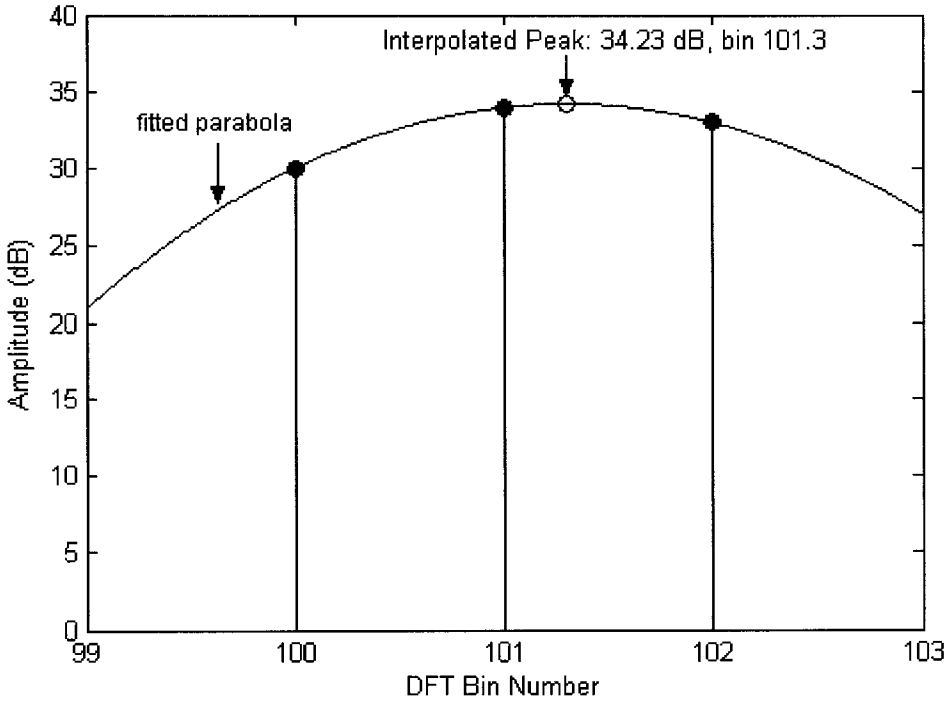


Figure 3.6. Parabolic Interpolation of a DFT Magnitude Peak

3.2.3 Peak Linking

The peaks detected in neighbouring DFT frames are linked across time based on similarity in amplitude and frequency. Some peak linking algorithms also expect a

harmonic structure for the peaks and do the linking based on maximising global harmonicity, not just local similarity of peaks across the frames. The peak linking algorithms in [32] and [46] are simple and generic in that they don't assume a harmonic model for the peak trajectories. Other more complex peak-linking algorithms are found in [51], [52] and [53].

3.3 Sinusoidal Synthesis

Synthesis of a signal from its sinusoidal components is referred to as additive synthesis. Additive synthesis of a signal from its peak trajectories is done using one of two basic approaches:

1. time-domain oscillators or
2. inverse DFT (IDFT).

Each approach will be reviewed under its own section below. A nice review of these approaches as well as other sound synthesis methods can be found in [54].

3.3.1 Additive Synthesis by Oscillators

The signal is synthesised by summing the outputs of a sinusoidal oscillator bank with time-varying parameters (see Figure 2.6). The time-varying sinusoidal parameters come from the amplitude, frequency and phase components of the peak trajectories found by sinusoidal analysis.

The frequency and phase trajectories are not independent. For a sinusoid the instantaneous frequency and phase are related:

$$\theta(t) = \int_0^t \omega(\tau)\,d\tau + \theta_0 \quad \Leftrightarrow \quad \omega(t) = \frac{d\theta}{dt} \tag{3.3}$$

where $\theta(t)$ is the instantaneous phase and $\omega(t)$ is the instantaneous angular frequency. The oscillators may be driven by the frequency of phase trajectories. The equation for the k^{th} frequency-driven oscillator is given by:

$$A_k(n)\cos\left(2\pi F_k(n)\right), \quad F_k(n) = F_k(n-1) + f_k(n)T_{hop} \tag{3.4}$$

The equation for the k^{th} phase-driven oscillator is given by:

$$A_k(n)\cos(\theta_k(n)) \qquad (3.5).$$

In (3.4) and (3.5), $A_k(n)$ is the amplitude trajectory, $f_k(n)$ is the frequency trajectory and $\theta_k(n)$ is the phase trajectory of the k^{th} peak trajectory and T_{hop} is the hop interval.

Unless sinusoidal analysis is run with a hop size of 1 sample, the peak trajectories do not contain amplitude, frequency and phase values for every sample in the signal. Per-sample parameters are obtained by upsampling the parameter trajectories with different interpolation schemes. McAulay and Quatieri proposed two interpolation strategies in [32]:

1. implicit interpolation by overlapping and adding windowed frames of the synthesised signal generated using sinusoidal parameters that were fixed for the hop size interval and
2. explicit interpolation of the parameter trajectories that could be used to control oscillators on a sample-by-sample basis.

The implicit strategy works well when the hop size is small. For the explicit strategy, linear interpolation of the amplitude trajectory was suggested. For the phase trajectory a cubic interpolation due to Almeida and Silva ([55]) with an extension to include smooth unwrapping was proposed.

The major disadvantage to using additive synthesis is the computational complexity involved in running all the oscillators. The IDFT method described in the next section is computationally more economic for generating a large number of simultaneous sinusoids.

3.3.2 Additive Synthesis by Inverse DFT

When a signal to be synthesised has a large number of simultaneous sinusoids, additive synthesis via the IDFT method is more efficient than via a bank of oscillators. The IDFT method, proposed in [56], is the reverse of the computation of the STDFT shown in Figure 3.1. The IDFT synthesis procedure is described by the following steps:

1. synthesis of a complex line spectrum for each frame,
2. convolution of the line spectra with the DFT of a window function,
3. computation of the IDFT of the spectra to generate time-domain signal and
4. overlapping and adding the IDFT frames.

The line spectra are generated using amplitude, frequency and phase trajectories for the signal to be synthesised. The frame length is dictated by the chosen DFT size. The amount of overlapping of time-domain frames is dictated by the interval between sinusoidal parameters in the trajectories (hop size). Note this method does not require explicit interpolation of the sinusoidal parameter trajectories. The parameters are assumed static over each frame and the interpolation between frames is accomplished implicitly by the overlap/add mechanism.

The computational cost of the IDFT is fixed for a given DFT size, so computational complexity does not change with the number of sinusoidal components. The computational cost is increased as the interval between sinusoidal parameters in the trajectories is decreased and/or the DFT size is increased. Some further optimisations for the IDFT additive synthesis method are suggested in [56].

3.4 Sinusoidal Modeling Software

Sinusoidal modeling is a mature technology and has been applied to many different signal processing and musical problems. Several sinusoidal modeling software packages have been developed and are available free of charge. Two popular choices are Lemur ([57], [58]) for the Macintosh computer, and SMSTools ([49]), for the IBM personal computer. The work in this thesis made use of the results of sinusoidal analysis produced by SMSTools.

SMSTools is a sound analysis, transformation and synthesis software package. For the work in this thesis, only the results of sinusoidal analysis (the peak trajectories) were used. SMSTools provides user control of relevant sinusoidal analysis parameters including frame length, hop size, window type, zero-padding length, peak detection and peak linking parameters. Sound analysis results are exported by SMSTools either as text-based Extensible Markup Language (XML,

[59]) or binary Sound Description Interchange Format (SDIF, [60]) files. Due to the large size of XML files, SDIF files were used exclusively for the work in this thesis.

Chapter 4

Pitched Musical Interference Suppression Methods

4.1 Introduction

The methods described in this chapter were designed to enhance the desired musical instrument signal in spot microphone signals by suppressing the interfering signals from neighbouring instruments. As described in the introduction to this thesis (Chapter 1), the purpose of a spot microphone is to capture the unreverberated signal of the desired instrument. In the context of an ensemble performance spot microphones unavoidably capture some of the signals due to neighbouring instruments. The signal due to the target instrument is referred to as the desired signal while the signals due to neighbouring instruments are collectively referred to as interference.

The signal composition of the spot microphone signals is illustrated in Figure 4.1 for the case of two instruments and two spot microphones.

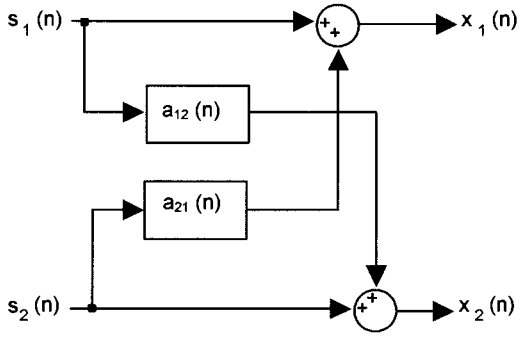


Figure 4.1. Signal Model of 2 Instrument, 2 Spot Microphone Recording Configuration

For spot microphone signal $x_1(n)$, the desired signal is $s_1(n)$, the signal from the target instrument, and the interfering signal is $s_2(n)*a_{21}(n)$, the signal from the neighbouring

instrument convolved with the room impulse response between the location of $s_2(n)$ and $s_1(n)$. Note that the interference suppression of sinusoidal components does not involve solving for the impulse responses $a_{ij}(n)$; all that matters is to suppress the interference signal $s_i(n)*a_{ij}(n)$. In this chapter, the case of two instruments and two spot microphones is considered, but the interference suppression methods can be extended to address more than two instruments and microphones.

The general approach to musical interference suppression is based on sinusoidal analysis. The goal is to identify the sinusoidal components of the interference and to remove said components. This approach, illustrated in Figure 4.2, is typical of CASA-based approaches outlined in section 2.3.

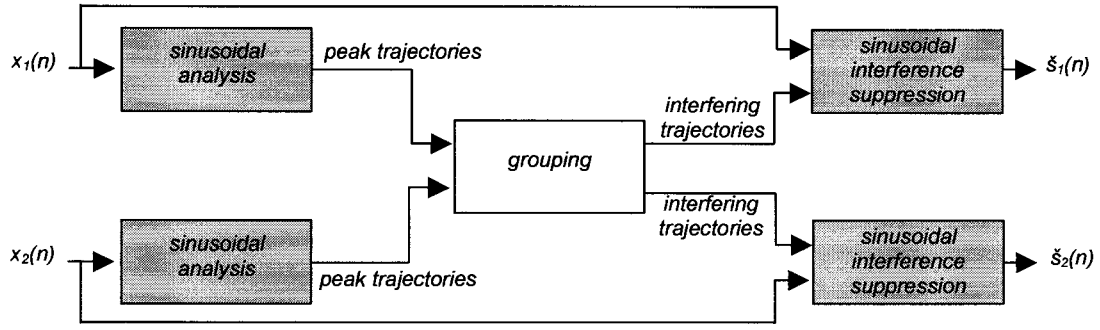


Figure 4.2. Sinusoidal Interference Suppression Framework

The signals $x_i(n)$ are the mixed spot microphone signals and the signals $\check{s}_1(n)$ and $\check{s}_2(n)$ are estimates of the desired signals $s_1(n)$ and $s_2(n)$. However, since the interference suppression approach attempts to remove only sinusoidal interfering components, $\check{s}_1(n)$ and $\check{s}_2(n)$ will still contain aharmonic and transient components of the interference (refer to the signal model, section 2.3.1). Interfering transients that occur while the desired signal is dominated by sinusoidal components are removed by replacing the time segment of $\check{s}_i(n)$ that contains the interfering transient with an estimate of the desired signal generated by sinusoidal synthesis, as illustrated in Figure 4.3.

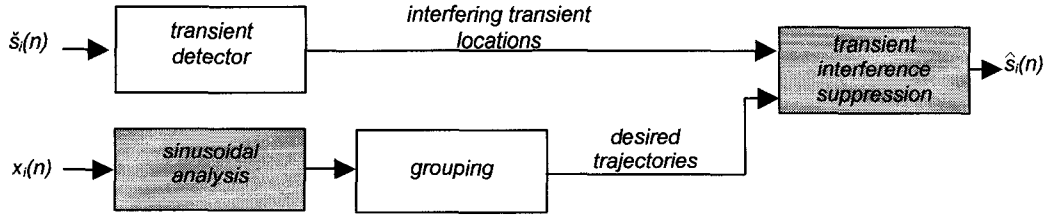


Figure 4.3. Transient Interference Suppression Framework

The sinusoidal trajectories used to synthesise the desired signal in the transient suppression are derived from sinusoidal analysis of the mixed spot microphone signals, $x_i(n)$, and the CASA-based grouping (also illustrated in Figure 4.2 for sinusoidal interference suppression).

The components of the interference suppression framework illustrated in Figure 4.2 and Figure 4.3 are discussed in the next sections of this chapter. The components in the white boxes of Figure 4.2 and Figure 4.3 were not implemented due to time constraints, but ideas for their implementation are presented under the respective sections. This chapter will conclude with a section discussing the known limitations of and problems associated with the interference suppression methods. Results of the interference suppression methods applied to some musical signals are presented in Chapter 5.

4.2 Sinusoidal Analysis

Sinusoidal analysis was done using the software program SMSTools [49]. SMSTools was chosen because it ran on this author's IBM computer and generated amplitude, frequency and phase trajectories for the sinusoidal components. Lemur, a sinusoidal analysis program for the Macintosh computer, does not generate phase information. The phase information is critical for sinusoidal synthesis used for both sinusoidal and transient interference suppression (see sections 4.4 and 4.6). It turned out that the phase information from SMSTools was not useful for the synthesis task and the phase trajectories had to be recomputed. The first section below describes the parameters used for the sinusoidal analysis. The second section describes how the phase trajectories were recomputed.

4.2.1 Parameters

The sample rate of all audio files was 22.05 kHz. The following parameters were used for the sinusoidal analysis of all audio files:

Parameter	Value	Meaning
Frame Size	825 samples (37.4 ms)	see section 3.2.1.1
Hop Size	32 samples (1.45 ms)	see section 3.2.1.2
Window Type	Blackmann-Harris	see section 3.2.1.3
Zero-Padding	1223 samples	see section 3.2.1.4
Peak Detection Threshold	-80 dB	minimum magnitude of a peak
Peak Detection Maximum Frequency	10000 Hz	maximum frequency of a peak
Sine Tracking Frequency Deviation	20 Hz	maximum frame-to-frame frequency deviation of a peak (for peak linking)
Harmonic Analysis	Off	does not force peaks to be a multiple of the estimated fundamental frequency

Table 4.1. Sinusoidal Analysis Parameters

The frame size was chosen based on McAulay and Quatieri's recommendation that the amount of data for each DFT frame be at least 2.5 periods of the fundamental frequency ([32]). A 37.4 ms frame size meets this recommendation for frequencies as low as 67 Hz, a lower frequency limit suitable for most musical instruments played in their typical ranges.

The hop size was chosen to provide dense sinusoidal parameter estimates, reducing the error in the interpolation of these parameters.

The Blackmann-Harris window has low side lobes, but a wide main lobe. This reduces the parameter estimation error introduced by frequency components far away. However, if there are strong frequency components situated close together, they will not be distinguishable. The Blackmann-Harris window is a good choice for sinusoidal analysis on an audio file with a single fundamental frequency because harmonics are guaranteed to be spaced wide enough apart to be distinguishable. All audio files analysed in the work reported in this thesis had a single fundamental

frequency, but this will not be the case in general and a different window will most likely be required.

Zero-padding was kept low to prevent the detection of false peaks due to window sidelobes.

Peak detection parameters were set to allow for the detection of sinusoids all the way up to 10 kHz. In typical musical instrument signals, high frequency sinusoids will have low amplitude. This is why the peak threshold was set so low.

The peak linking parameter (sine tracking frequency deviation) was left at the default value and was found to be adequate. It would be nice if this parameter were frequency dependent, so that one could allow for more frame-to-frame frequency deviation as the frequency of the sinusoid increased. In harmonic signals, if the fundamental changes by Δf Hz from one frame to the next, the n^{th} harmonic of the fundamental will change by $n\Delta f$ Hz, so a larger tolerance for frequency deviation for higher frequencies makes sense.

Harmonic analysis was turned off because when on, the frequency trajectories are forced to be a multiple of the estimated fundamental frequency. This makes the peak trajectories subject to errors in fundamental frequency estimation.

4.2.2 Phase Computation

The phase trajectories reported by SMSTools did not appear to be correctly time-stamped. This was discovered when sinusoidal synthesis using a phase-driven oscillator was attempted. It was found that the frequency trace of the generated signal was correct, but the phase of the signal synthesised signal did not match that of the original signal, as illustrated in Figure 4.4.

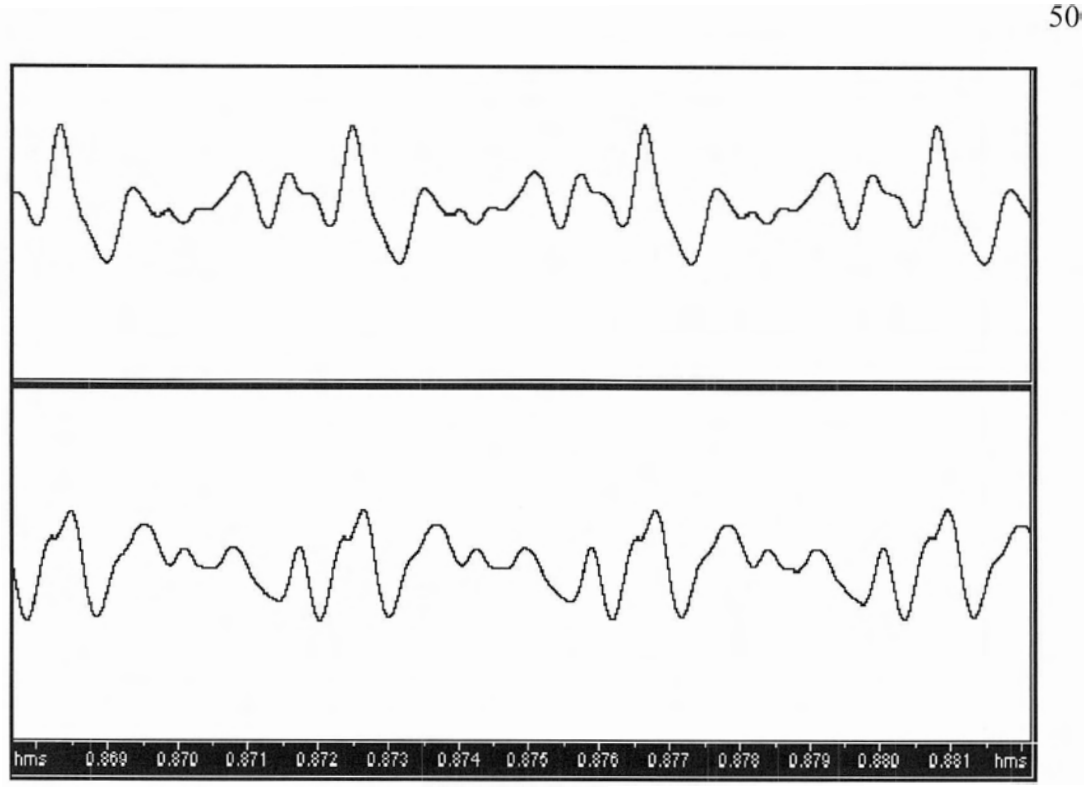


Figure 4.4. Cello Signal Waveforms: Original (Top) and Synthesised By Additive Synthesis using a Phase-Driven Oscillator (Bottom)

Although the original and synthesised signals sound the same (except for the missing aharmonic component in the synthesised version), phase fidelity is important for interference suppression methods, as explained in sections 4.4 and 4.6.

To compute the correct phases of a trajectory, the frequency information from sinusoidal analysis is used. To compute the phase of a trajectory at a specific point in time t_0, a fragment of the signal beginning at t_0 and containing 2 periods of the target frequency is isolated. This fragment is then forward-backward filtered with a 4^{th} order Butterworth bandpass filter, centred at the target frequency. This isolates the sinusoidal component we are currently interested in computing the phase for. The forward-backward filtering ensures that the filter doesn't introduce any phase distortion (which would sabotage the task at hand). The filtered signal is then upsampled to achieve the desired phase resolution. The upsample factor is computed as follows:

$$R = \max\left(1, \textit{round}\left(\frac{2\pi T_s f_0}{err}\right)\right) \qquad (4.1),$$

where *err* is the error tolerance in the phase estimate, T_s is the sampling period and f_0 is the trajectory frequency at t_0. Note even at low frequencies when there is lots of time resolution in the trajectory, the signal is never downsampled ($R>=1$). A 9 point sinc interpolation kernel is used in the upsampling to get a good quality reconstruction of the cosine wave.

To determine the phase of this upsampled cosine wave a single-period prototype function with zero phase offset is generated and correlated to the 2 periods upsampled signal. The prototype signal is given by:

$$p(nT_{s1}) = \cos(2\pi f_0 nT_{s1}), \quad n = [0, T_0) \tag{4.2}$$

where T_{s1} is the upsampled sampling period ($T_{s1} = RT_s$) and T_0 is the period of the trajectory. Two periods of the signal are required in order to guarantee that the maximum positive correlation value will be observable. The phase of the signal at t_0 is then given by:

$$\theta(t_0) = -lagMax \times T_{s1} \times 2\pi f_0 \tag{4.3}$$

where *lagMax* is the lag at which the maximum of the correlation is found.

The actual phase is so estimated for the trajectory at each analysis timestamp and replaces SMSTools' estimates.

4.3 Grouping of Sinusoidal Components

Grouping of the peak trajectories into those that are desired and those that are interference was not automated as part of the work for this thesis due to time constraints. Grouping was done manually for the experimental results discussed in Chapter 5.

A grouping method based on CASA principles such as harmonicity and similarity in amplitude and frequency evolution (see section 2.3) is recommended. The availability of multiple spot microphone signals provides further information useful for performing grouping. Consider the case of two spot microphones and two sources. The peak trajectories of source 1 will appear in both spot microphone signals. In spot microphone signal 1 the peak trajectories due to source 1 will be

strong. In spot microphone signal 2 the peak trajectories due to source 1 will be weak and delayed relative to their counterparts in spot microphone signal 1. Therefore the grouping task can be made easier by searching for matching peak trajectories in the two spot microphone signals and considering their relative amplitude and delay differences.

4.4 Sinusoidal Interference Suppression

The suppression of sinusoidal interference is done using knowledge of the undesired sinusoidal components, detected by sinusoidal analysis of the spot microphone signals. Two strategies for suppression of the undesired sinusoidal components are proposed:

1. suppression of undesired sinusoidal components by time-varying, narrowband notch filters and
2. suppression of undesired sinusoidal components by synthesis and subtraction from the spot microphone signal.

For the sake of succinctness, the sinusoidal interference suppression strategies shall be referred to as the "filtering method" and "subtraction method" respectively.

Each sinusoidal interference strategy has its own set of advantages and disadvantages. The filtering method has the advantage that highly accurate sinusoidal parameters are not required for effective interference suppression. Indeed only frequency trajectories are required for controlling the notch filters' centre frequencies. The disadvantages of the filtering method include:

- introduction of wide time-frequency "holes" in the resulting signal
- distortion of the phases of the remaining (desired) components and
- suppression of desired components that collide with the undesired components.

The advantage of the subtraction method is that only a minimal amount of distortion to the desired components of the signal is introduced, provided that accurate parameters for the undesired components are available. The disadvantages of the subtraction method are:

- requirement for a phase trajectory in the peak trajectories and
- sensitivity to errors in the parameterised undesired sinusoidal components, particular the phases.

The impact of the first disadvantage is just an increase in computational complexity to compute the phases (see section 4.2.2). If the errors in sinusoidal parameters are severe enough, the subtraction method has the potential to amplify the undesired components and introduce new undesired sinusoidal parameters.

The implementations of the filtering and subtraction methods are discussed in the following subsections.

4.4.1 Filtering Method

Once the trajectories representing the stationary sinusoidal components of the interferer are identified, they are removed by a bank of time-varying notch filters, centred about the (slowly) time-varying frequency of the trajectories. Since this method is non-real-time, there is no need to do the filtering all in one pass. Instead it is done one trajectory at a time over multiple passes (there is only one notch filter active at a time).

The time-varying notch filter is based on parametric second-order sections, described in [61]. This filter structure allows for direct control via intuitive independent parameters: notch gain (attenuation), centre frequency and bandwidth. In order to get a good, very narrow response from such a filter, a cascade of five of them is used, with each second-order section (SOS) having identical coefficients at any given time. A cascade of SOS filters is used in lieu of a single 10^{th} order filter because the SOS are often borderline stable (poles very close to the unit circle). Convolution of the coefficients in MATLAB sometimes yields an unstable 10^{th} order filter due to finite numerical precision. The frequency response of the filter cascade centred about 5000 Hz is illustrated in Figure 4.5. The bandwidth and attenuation of each filter section is 10 Hz and –10 dB, respectively.

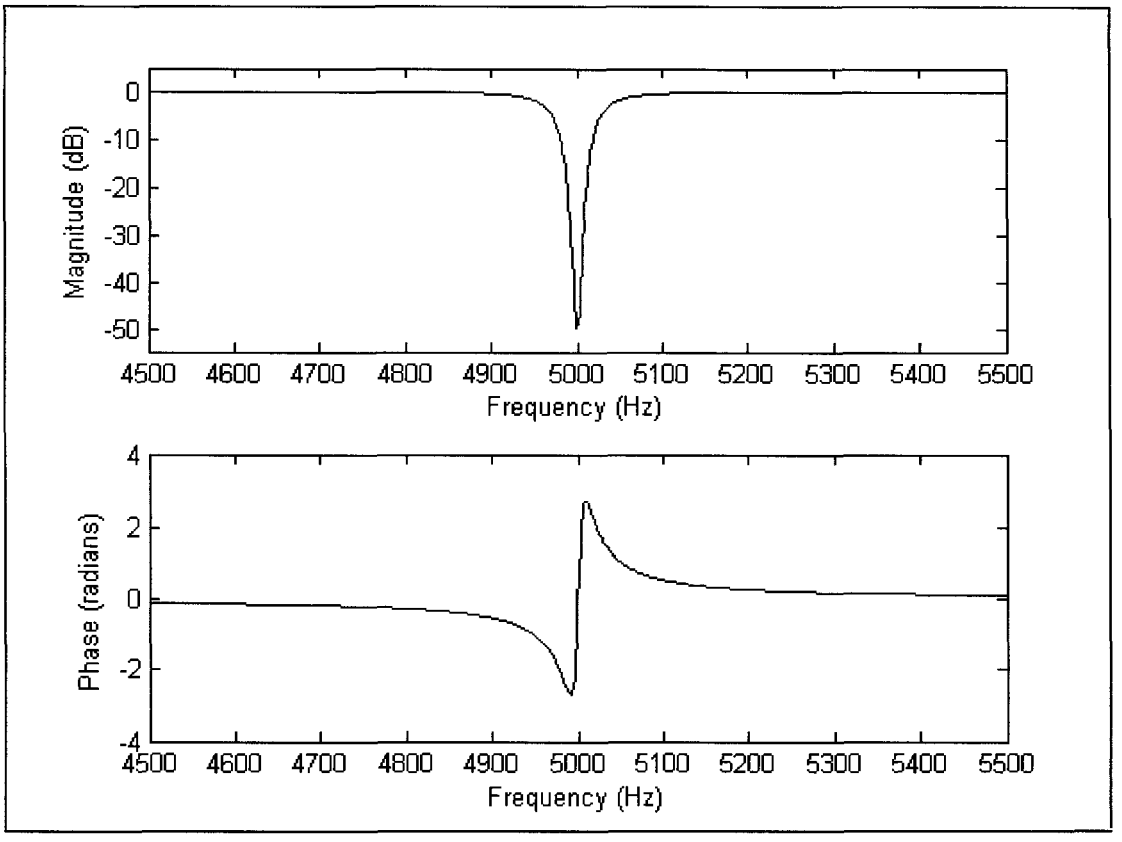


Figure 4.5. Frequency Response of Time-Varying Notch Filter Bank, with 5000 Hz Centre Frequency, –10 dB Attenuation and 10 Hz Bandwidth for Each Section

The following sections describe how the filter parameters, attenuation, bandwidth and centre frequency were configured.

4.4.1.1 Attenuation

To try to keep the system stable, get a narrow bandwidth and a good attenuation, each section is configured to yield a constant gain of –10 dB, except when the filter is being switched on and off, for a total gain of -50 dB for the cascade at the notch centre frequency. When the filter is being switched on at the start of a trajectory the gain is ramped down from 0 to –10 dB on each section over time. Likewise, when the filter is switched off at the end of a trajectory the gain is ramped up from –10 dB to 0 dB. This gradual fade in and out is meant to reduce the occurrence of audible artefacts in the filtered signal.

4.4.1.2 Bandwidth

The bandwidth is defined as the width of the notch in Hz at –3 dB attenuation. The bandwidth of each section is made very narrow to avoid affecting other frequencies that may contain desired signal. The bandwidth parameter is fixed at 28 Hz for each filter section for the whole duration that the filter is active and for any trajectory frequency. This results in a bandwidth of 77 Hz for the filter cascade.

4.4.1.3 Centre Frequency

The filter centre frequency is time-varying and is set by the frequency trajectory. The filter centre frequency is updated at the frequency trajectory interval. Since the filter bandwidth is 77 Hz and the maximum frame-to-frame frequency deviation set for sinusoidal analysis is 20 Hz (see section 4.2.1), there is no need to update the notch frequency any more often than the analysis interval. The sinusoidal frequency will always fall in the notch over the duration of the analysis interval.

4.4.2 Subtraction Method

The subtraction method for sinusoidal interference suppression involves synthesis of an estimate of the interference and subtracting it from the spot microphone signal. This method is inspired by the work of Tolonen ([8]). The estimate of the interference is generated by sinusoidal synthesis using the interference trajectories. The sinusoidal synthesis is done using phase-driven oscillators (see section 3.3.1). This synthesis method was chosen because it provides for direct manipulation of the phase, which is essential for generating the interference estimate that is phase matched to the interference in the spot microphone signal. A phase mismatch larger than $\pi/3$ causes interference amplification when the unmatched interference estimate is subtracted.

The amplitude and phase parameters of the oscillator used for synthesis are updated on a per-sample basis. This requires upsampling the amplitude and phase trajectories to the sampling period. The upsampling of the amplitude and phase trajectories is discussed in the first section below. Post-processing of the synthesised

sinusoidal components of the interference estimate is discussed in the second section below.

4.4.2.1 Upsampling of Peak Trajectories

The amplitude and phase parameters of the phase-driven oscillator are updated on a per-sample basis. This requires that the amplitude and phase trajectories be upsampled from the analysis interval (hop size) to the sampling period. Following McAulay and Quatieri, the amplitude is upsampled using linear interpolation and the phase is upsampled using a cubic polynomial [32].

The upsampled amplitude between the kth and kth+1 analysis interval using linear interpolation is given by:

$$\tilde{A}(n) = \hat{A}^k + \frac{\left(\hat{A}^{k+1} - \hat{A}^k\right)}{T} n, \quad n = 0,1,\ldots,T-1 \qquad (4.4),$$

where $\hat{A}^k$ and $\hat{A}^{k+1}$ are the k^{th} and $k^{th}+1$ amplitude estimates in decibels (dB) from the amplitude trajectory and T is the analysis interval (hop size) in samples at the sampling period.

The upsampled phase between the kth and kth+1 analysis interval using cubic interpolation is given by:

$$\tilde{\theta}(n) = \zeta + \gamma n + \alpha n^2 + \beta n^3, \quad n = 0,1,\ldots,T-1 \qquad (4.5),$$

where T is the analysis interval (hop size) in samples at the sampling period. The coefficients of the cubic polynomial function are given by:

$$\zeta = \hat{\theta}^k$$

$$\gamma = \hat{\omega}^k$$

$$\begin{bmatrix} \alpha(M) \\ \beta(M) \end{bmatrix} = \begin{bmatrix} \frac{3}{T^2} & \frac{-1}{T} \\ \frac{-2}{T^3} & \frac{1}{T^2} \end{bmatrix} \begin{bmatrix} \hat{\theta}^{k+1} - \hat{\theta}^k - \hat{\omega}^k T + 2\pi M \\ \hat{\omega}^{k+1} - \hat{\omega}^k \end{bmatrix} \quad (4.6),$$

where $\hat{\theta}^k$ and $\hat{\theta}^{k+1}$ are the k^{th} and $k^{th}+1$ phase estimates in radians from the phase trajectory and $\hat{\omega}^k$ and $\hat{\omega}^{k+1}$ are the k^{th} and $k^{th}+1$ angular frequency estimates in radians from the frequency trajectory. The expression $2\pi M$ unwraps the phase, and the integer M is chosen such that the unwrapped phase function has as little variation as possible. The value of M is given by the closest integer to x^*:

$$x^* = \frac{1}{2\pi}\left[\left(\hat{\theta}^k + \hat{\omega}^k T - \hat{\theta}^{k+1}\right) + \left(\hat{\omega}^{k+1} - \hat{\omega}^k\right)\frac{T}{2}\right] \quad (4.7).$$

The derivation of the cubic polynomial coefficients stated above is found in [32].

4.4.2.2 Post-processing of Synthesised Sinusoids

Two forms of post-processing are applied to each sinusoid following synthesis:

- fading in at onsets and fading out at offsets and
- bandpass filtering the sinusoid.

To avoid artefacts that may be caused by subtraction of a signal with sudden onsets and offsets, the synthesised signal is faded in at its onset and faded out at its offset. The fading function used is linear, ramped up from 0 to 1 in 20 ms for the fade in case and from 1 to 0 in 20 ms for the fade out case. The fading function is a gain modulation applied to the signal. Fading is illustrated in Figure 4.6

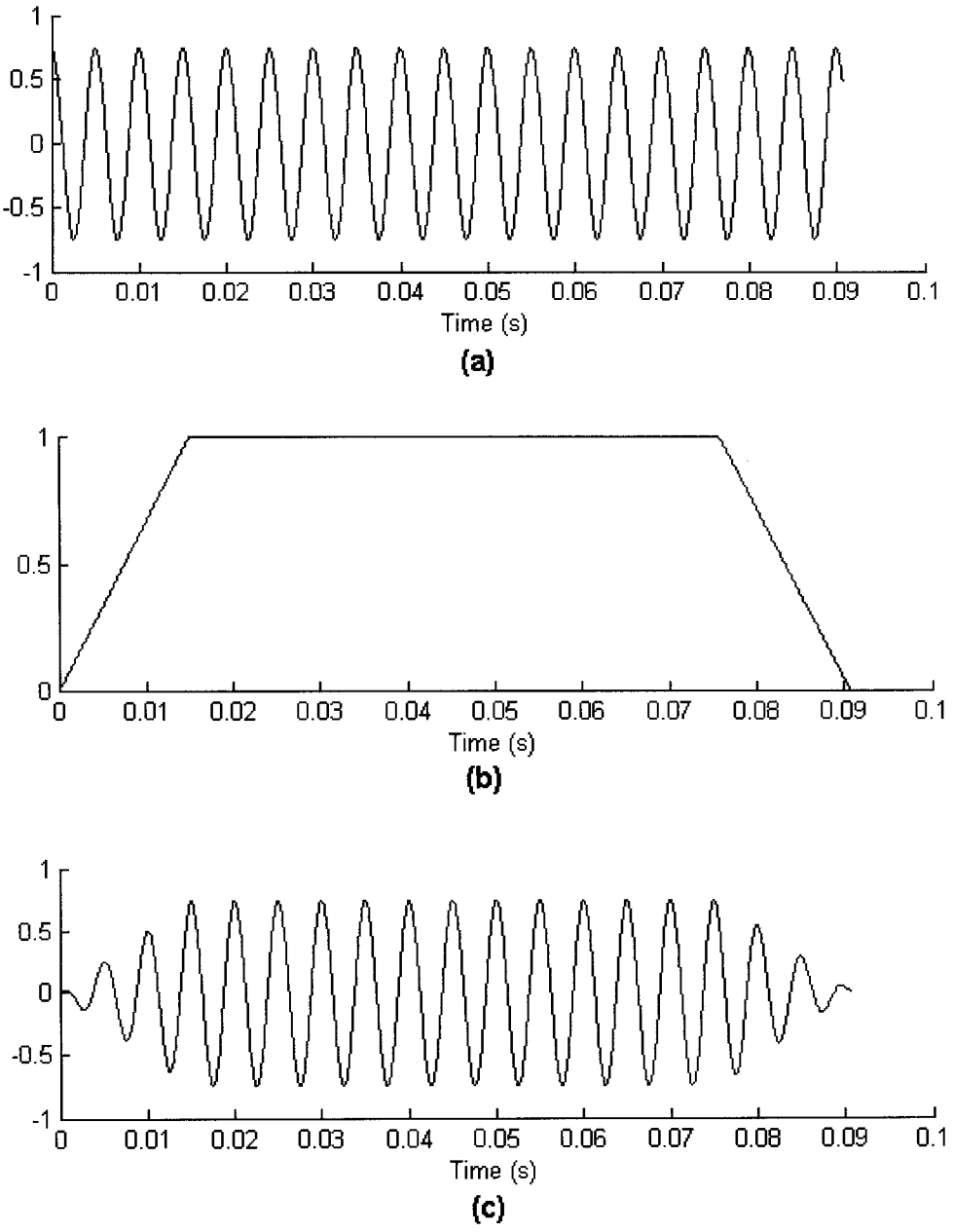


Figure 4.6. Fading a Signal In and Out Using a 15 ms Fade Time. (a) Original Signal, (b) Fading Function, (c) Faded Signal

Noisy amplitude and phase trajectories used to control the oscillator introduce noise into the synthesised signal. To reduce this noise, the resynthesised sinusoidal trajectory is post-filtered with a narrowband, time-varying bandpass filter that tracks the frequency of the trajectory. To eliminate startup effects due to the filter memory being initialised with zeros, the trajectory is first extended back in time from its start.

The segment of the trajectory that is backward-extended is referred to as the "leader sequence". The leader sequence is only temporary and ensures the filter memory is initialised appropriately by the time it reaches the true start of the trajectory. The extended trajectory is filtered and then the leader sequence, which contains the startup transient, is removed. The phase shift introduced by the bandpass filter is always 2π at the centre frequency, so there is very little phase distortion introduced by this filter, and there is no need to compensate by forward-backward filtering.

The frequency trajectory is used to control the bandpass filter. Since there is usually noise in the frequency trajectory reported by SMSTools, the frequency trajectory itself is first smoothed with a 128 point FIR averaging filter before it is applied to control the bandpass filter centre frequency.

The time-averaged power spectral density (PSD) of a resynthesised fragment of a single-tone vibrato signal before and after post-processing is shown in Figure 4.7.

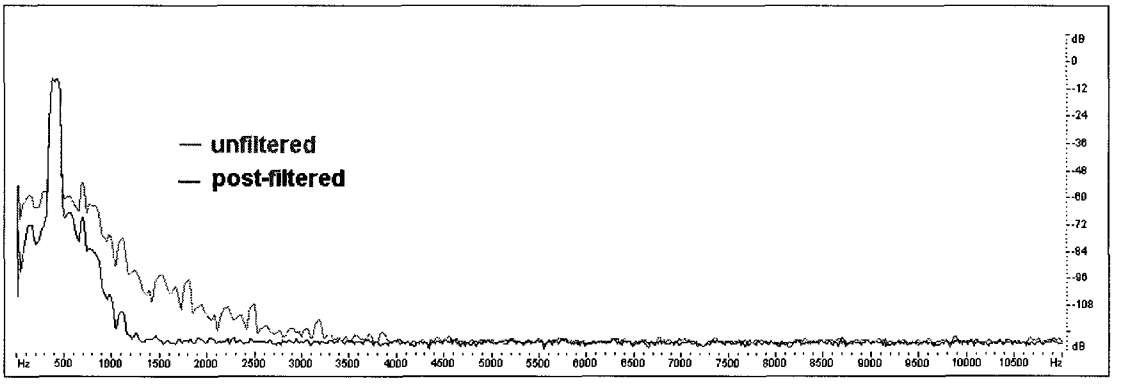


Figure 4.7. Time-Averaged PSD of a Synthesised Single Tone Vibrato Signal Before and After Post-Filtering

4.5 Transient Detector

Sinusoidal interference suppression, described in section 4.4, only removes sinusoidal components of the interference. Interfering transients that occur during time regions when the desired signal is sinusoidal are easily removed, as described in section 4.6. The method in section 4.6 relies on information about when such interfering transients occur. Ideally, an automated transient detector would provide this information. Due to time constraints, a transient detector was not implemented as

part of the work described in this thesis. Transient regions were manually identified for the experimental results discussed in Chapter 5.

Some methods for detecting transients are suggested in [62] and [63]. As for sinusoidal grouping, described in section 4.3, the redundancy in the multiple spot microphone signals could be exploited to improve the transient detector.

4.6 Transient Interference Suppression

When interfering transients occur in sinusoidal regions of the desired signal, the transients can be removed by:

1. applying a gain of zero to the time segment containing the transient then
2. resynthesising the desired signal over this time segment.

This method requires knowledge of the time boundaries of the segment containing the interfering transient (see section 4.5) and the peak trajectories of the desired signal over this segment (see sections 4.2 and 4.3). This method cannot be applied to removal of all interfering transients. The interfering transients that occur at the same time as desired transients couldn't be removed by this method because it is not possible to resynthesise the desired transient from sinusoidal components. As discussed in section 3.1, transients are not adequately represented by a sinusoidal model.

To avoid the potential for introducing artefacts, the segment of the signal containing the transient region is gradually faded to zero before the start of the transient and is gradually faded back to full amplitude after the end of the transient. The resynthesised signal receives the complementary fading: it is faded in starting at the same time as the original signal is starting to be faded out. The resynthesised signal is faded out starting at the same time as the original signal is starting to be faded back in. The faded original and resynthesised signals are then summed together to get a signal with the transient removed. To preserve energy in the summed signal, the fading functions are linear. This process is known as linear cross-fading and is described by the following equations:

$$\hat{s}(n) = s(n) \times f_2(n) + r(n) \times f_1(n)$$

$$f_2(n) = \begin{cases} 1 & , n < n_s , n > n_t \\ 1 - \dfrac{n - n_s}{T} & , n_s \le n \le n_s + T \\ 0 & , n_s + T \le n \le n_t \\ \dfrac{n - n_t}{T} & , n_t \le n \le n_t + T \end{cases} \qquad (4.8),$$

$$f_1(n) = 1 - f_2(n)$$

where $s(n)$ and $r(n)$ are the original and resynthesised signals, $f_2(n)$ is the fade function applied to the original signal and $f_1(n)$ is the fade function applied to the resynthesised signal. The transient segment is defined as the samples between n_s and n_t. The cross-fade length, T, is the number of samples over which the fade occurs. Note that if the phase of the resynthesised signal is perfectly matched to that of the original signal, gradual cross-fading is unnecessary; one could just cut and paste the signal fragments without linear fading at the edges. The cross-fade was implemented to minimise artefacts when the phases are not perfectly matched. Cross-fading is illustrated in Figure 4.8 using a cross-fade time of 20 ms.

The resynthesised desired signal is synthesised by sinusoidal synthesis from the desired signal trajectories, in the same manner as the undesired sinusoidal interference signal is synthesised, described in section 4.4.2.

This transient suppression method completely removes the interfering transient but also results in the loss of the aharmonic component of the desired signal over the transient segment. This is because the aharmonic component is not represented in the sinusoidal trajectories used to resynthesise the desired signal, as explained in section 3.1.

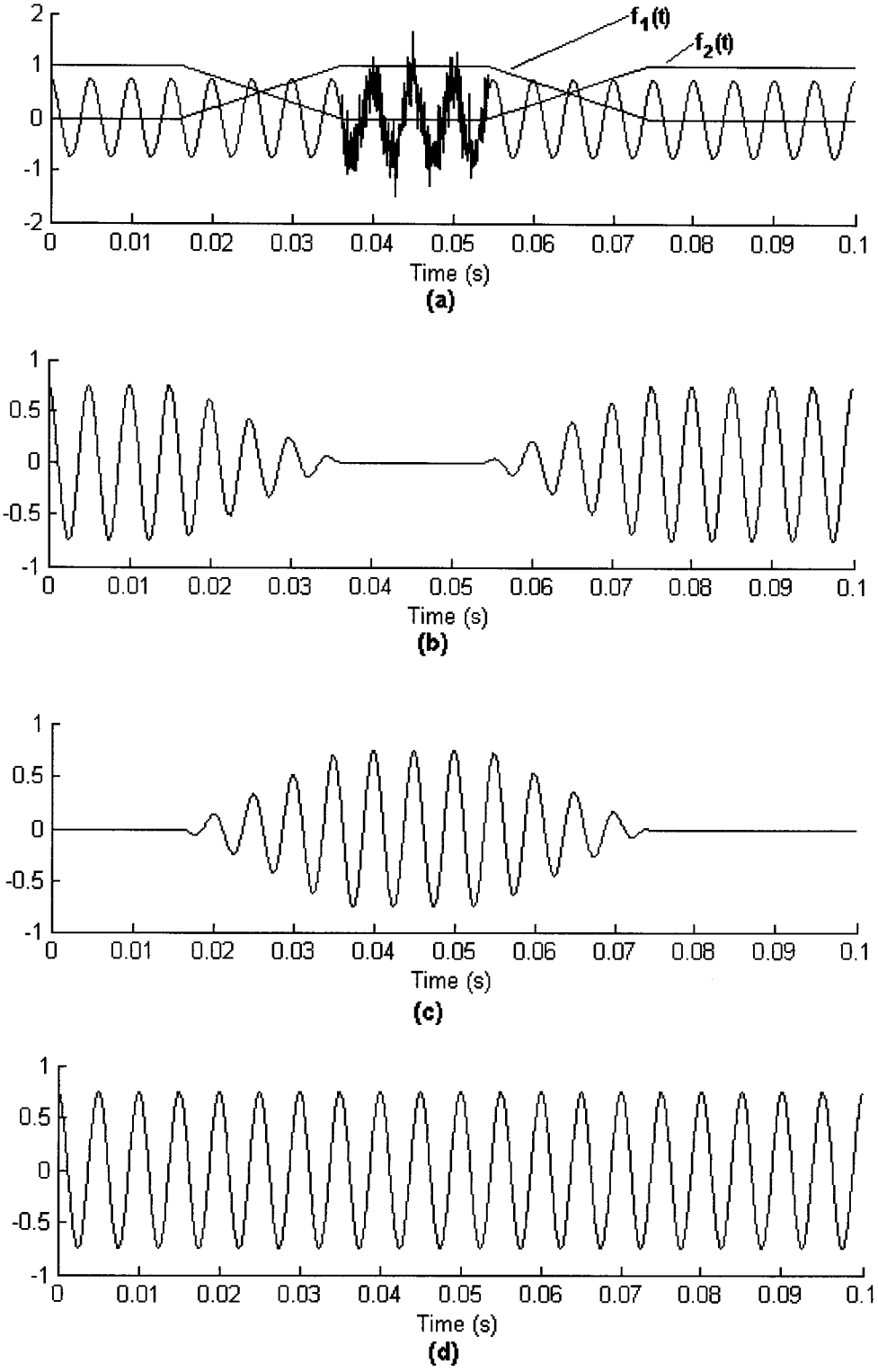


Figure 4.8. Transient Suppression. (a) Signal Containing Transient and Fade Functions, $f_1(t)$: for Resynthesised Signal, $f_2(t)$ for Original Signal, (b) Original Signal Multiplied by $f_2(t)$, (c) Resynthesised Signal Multiplied by $f_1(t)$, (d) Resulting Signal with Transient Removed

4.7 Limitations of the Methods

The methods outlined in the previous sections of this chapter have the following limitations:

1. transient segments of the interference that overlap with transient segments in the desired signal are not suppressed,
2. aharmonic components of the interference are not suppressed.
3. time-frequency collisions result in loss of desired signal components when the filtering method is used and
4. the methods do not address interference suppression of non-pitched musical instruments (e.g. snare drum).

Further to the general limitations outlined above, other problems may arise from the interference suppression methods, including:

1. artefacts introduced by the notch filters for the filtering method,
2. enhancement of interference or addition of new interfering sinusoids if poor estimates of the interfering sinusoidal parameters are made in the subtraction method.

For the filtering method, the notch filters may introduce artefacts if the coefficients are changed too quickly. Artefacts due to desired sinusoidal components entering and leaving the notch zones might also be introduced.

For the subtraction method, if the absolute value of the error in the phase estimate is larger than $\pi/3$ the subtraction actually increases the magnitude of the interfering sinusoidal component. The subtraction method can also introduce new, undesired sinusoidal components if the error in phase and frequency estimates is large.

Chapter 5

Results

5.1 Introduction

The pitched musical interference suppression methods described in Chapter 4 were applied to a number of musical signals. The metrics used to evaluate the success of the methods are described in the first section of this chapter. Following the introduction of the evaluation metrics, the performance of sinusoidal analysis and synthesis is assessed using these metrics. Assessing the ability to reconstruct a signal by sinusoidal synthesis using the results of sinusoidal analysis is important because the interference suppression methods are dependent on the success of sinusoidal modeling. This chapter will conclude with an evaluation of the interference suppression methods described in Chapter 4 using several different musical signals.

5.2 Evaluation Metrics

Two metrics were used to evaluate the similarity of two signals:

1. error of spectral magnitude and
2. ordinary error.

The error of spectral magnitude has the following definition:

$$errM = 10\log_{10}\left(\frac{\sum_{k=0}^{N-1}\left(|S_1(k)|-|S_2(k)|\right)^2}{N^2}\right) \qquad (5.1),$$

where $S_1(k)$ is the value of the DFT of signal 1 at bin k and $S_2(k)$ is the value of the DFT of signal 2 at bin k. The N-point DFT is computed using a rectangular window. Equation 5.1 was derived assuming the following definition for the DFT:

$$S(k) = \sum_{n=0}^{N-1} s(n) \exp^{\frac{-j2\pi nk}{N}} \qquad (5.2),$$

where $s(n)$ are the samples of the signal. The error of spectral magnitude provides information about the power of the error between signals 1 and 2 when only spectral magnitude is considered. This metric does not penalise for phase mismatch between the signals. The error of spectral magnitude metric is useful for evaluation of the filtering method for sinusoidal interference suppression because the notch filters distort the phase of the desired signal around the notch frequency.

The ordinary error has the following definition:

$$err = 10\log_{10}\left(\frac{\sum_{n=0}^{N-1}\left(s_1(n)-s_2(n)\right)^2}{N}\right) \qquad (5.3),$$

where $s_1(n)$ is the value of the n^{th} sample of signal 1 and $s_2(n)$ is the value of the n^{th} sample of signal 2. The error of spectral magnitude in (5.1) will be referred to as *errM* while the ordinary error in (5.3) will be referred to as *err*. The *err* (5.3) can also be defined in terms of the DFT of the signals:

$$err = 10\log_{10}\left(\frac{\sum_{k=0}^{N-1}\left|S_1(k)-S_2(k)\right|^2}{N^2}\right) \qquad (5.4).$$

Unlike *errM*, *err* penalises for phase mismatches. The *err* is important for assessing the phase fidelity between the signals.

Since the interference suppression methods target the sinusoidal regions of the desired signal, the error metrics, *err* and *errM*, between the actual desired signal and the estimated desired signal were expected to vary with time. Accordingly, the error metrics were computed over non-overlapping short time intervals of the signals and

time-stamped to the middle of the analysis interval. The evaluation of the difference between the true desired signal and estimated desired signal (following interference suppression) consists of error metric curves that vary over time.

5.3 Evaluation of Signal Reconstruction

The interference suppression methods, particularly the subtraction method, rely on the assumption that a pitched musical signal can be reconstructed by sinusoidal synthesis using peak trajectories determined from sinusoidal analysis. In this section, this assumption is validated using error metrics described in section 5.2. The following steps summarise the validation procedure:

1. analyse the signal and extract peak trajectories using SMSTools, supplemented by a unique phase estimator
2. use peak trajectories to reconstruct the signal by sinusoidal synthesis
3. compute error metrics between original signal and reconstructed signal.

The parameters and phase trajectory computation for sinusoidal analysis were described in section 4.2. The sinusoidal synthesis was done according to the method described in section 4.4.2.

Six different signals were used for the validation. The signals and their properties are summarised in Table 5.1. The guitar signals are the same, except that the latter were filtered with an FIR reverberation filter.

Signal	Description
flat	machine-generated, single unmodulated 400 Hz tone
vibrato	machine-generated tone, 400 Hz centre frequency, frequency-modulated by a sine with amplitude = 30 Hz, frequency = 4.5 Hz
flatSeries	machine-generated, unmodulated harmonic series with fundamental frequency = 400 Hz, maximum frequency = 8800 Hz (22 sines), all harmonics have the same amplitude
cello	single bowed cello note
guitar	time-series of 7 monophonic notes
guitar with reverberation	time-series of 7 monophonic notes, with long reverberation

Table 5.1. Signals used for Resynthesis Tests

Each of the signals in Table 5.1 were sinusoidally analysed then resynthesised. The error metrics between original signal and resynthesised signals were computed.

Some statistics of the errors are listed in Table 5.2. The error curves are illustrated in Figure 5.1 and Figure 5.2.

Signal	Mean *err*	Median *err*	Mean *errM*	Median *errM*
flat	-32.83 dB	-48.89 dB	-33.42 dB	-52.71 dB
vibrato	-30.04 dB	-34.17 dB	-34.54 dB	-46.19 dB
flatSeries	-35.54 dB	-37.06 dB	-39.58 dB	-49.15 dB
cello	-49.86 dB	-50.09 dB	-51.71 dB	-52.24 dB
guitar	-24.78 dB	-45.31 dB	-27.42 dB	-49.19 dB
guitar with reverberation	-25.31 dB	-37.05 dB	-28.18 dB	-39.25 dB

Table 5.2. Error Statistics for Original and Resynthesised Signals

As can be seen in Figure 5.1 and Figure 5.2, the error is typically larger around onsets and offsets. These error maxima are short-lived and can be attributed to:

1. sudden onsets and offsets (machine-generated signals, Figure 5.1) and
2. transients (guitar signals, Figure 5.2).

The sudden onsets and offsets of machine-generated signals are not well localised in the DFT. This causes inaccurate amplitude estimates and timestamps in the sinusoidal analysis near the onsets and offsets. By contrast, the cello signal has very gradual onsets and offsets that are easier to localise with the DFT. This results in less error in the resynthesised signal at the onset and offset.

Since transients are not modeled by the sinusoidal representation, the error in the resynthesised signals will be large at note onsets. This is evident in the error curves for the guitar signals (Figure 5.2 (b) and (c), which have a significant transient component due to the plucking of the string. By contrast, the bowed cello signal does not have a significant transient, so the error is not large at the onset.

The median error statistics are presented in Table 5.2 because the onset and offset error spikes do not as strongly influence it as the mean error statistics. The mean and median of the cello signal error metrics are nearly identical because there are no large error spikes due to the smooth onset and offset of the cello signal.

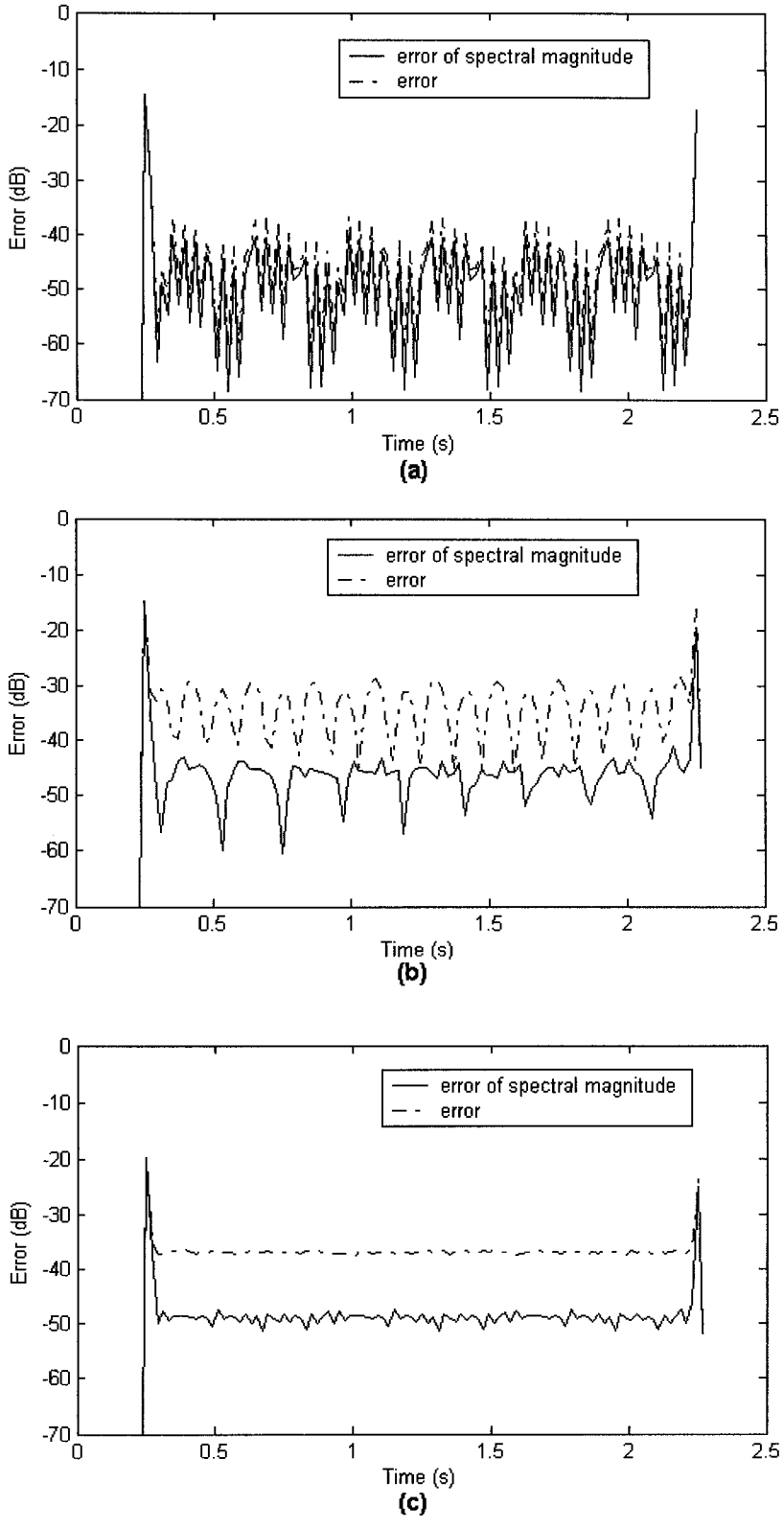


Figure 5.1. Error in Resynthesised Signals. (a) flat, (b) vibrato, (c) flatSeries

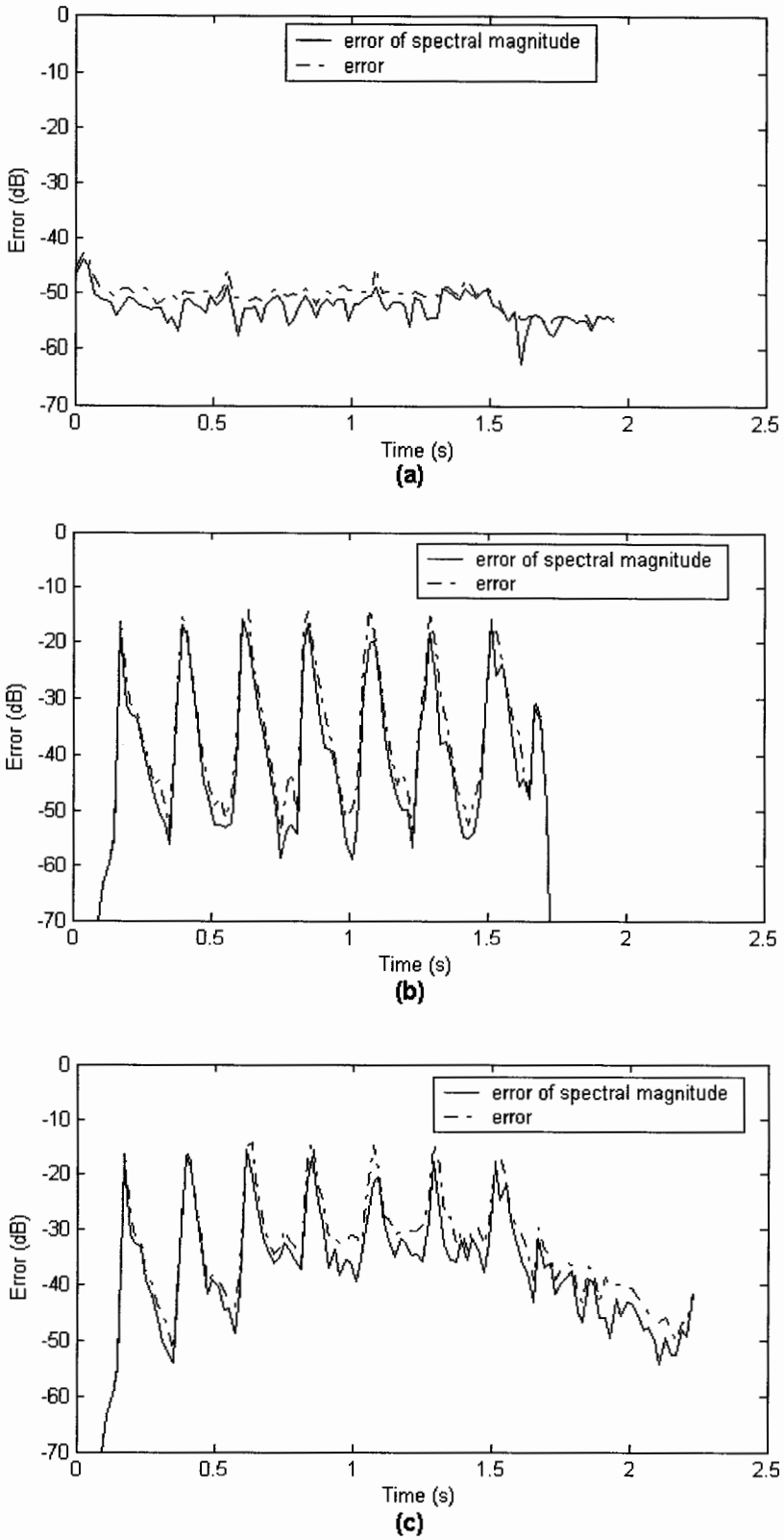


Figure 5.2. Error in Resynthesised Signals. (a) cello, (b) guitar, (c) guitar with reverberation

In general, the *err* and *errM* are very similar, indicating that the phase of the resynthesised signal sinusoids are fairly well-matched to the original signal. This is a nice validation of the phase computation method described in section 4.2.2. There are two exceptions to this trend: the vibrato signal and the flatSeries signal (Figure 5.1). The vibrato signal has a rapidly time-varying frequency, which can make the phase difficult to estimate accurately. This situation will be even worse for higher sinusoids of a vibrato signal, because the frequency changes even more rapidly. The flatSeries signal has energetic, high-frequency sinusoids. These high-frequency sinusoids have fewer samples per period, which reduces the accuracy of the phase estimate. Because these high-frequency sinusoids have the same amplitude as the low-frequency sinusoids in the flatSeries signal, inaccuracies in the phase estimates of the high-frequency sinusoids will contribute significantly to the *err*. Fortunately this scenario is not typical of most natural musical instruments. Natural musical signals will have a strong spectral tilt: the high-frequency sinusoids will have much lower amplitude than low frequency sinusoids, as illustrated in Figure 5.3. The errors in phase estimates for the high-frequency sinusoids will not have a significant impact on the overall *err*. This is evident by the similarity in *err* and *errM* curves for the natural musical signals in Figure 5.2.

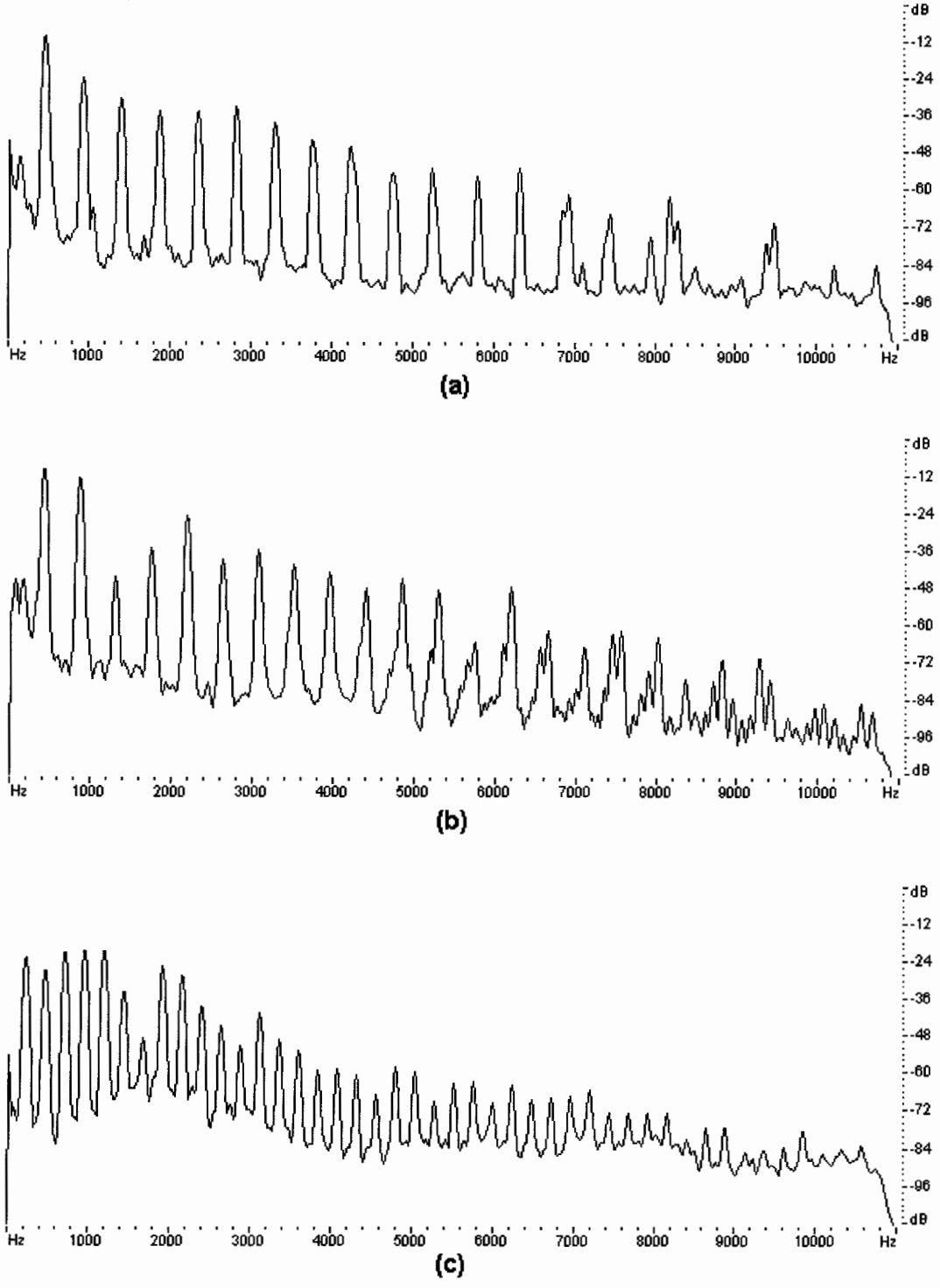


Figure 5.3. PSD Averaged over 100 ms of Stable Notes from Various Musical Instruments. (a) piano, (b) guitar, (c) cello

The mean error statistics in Table 5.2 for the guitar signals are similar for the dry and the reverberated signal. The median statistics for the reverberated signal are higher than those for the dry signal due to the reverberation tail after the final note (t = 1.75-2.25 s, Figure 5.2 (c)). The valleys of the error curves for the guitar signals (Figure 5.2 (b), (c)) are higher for the reverberated signal, particular as time increases. This can be attributed to:

1. smearing of the transients over time and
2. smearing of the sinusoids from previous notes over time.

Both smearing effects are due to the action of the FIR reverberation filter. When the sinusoids from previous notes are smeared into the sinusoids of the next note, the signal is no longer monophonic and there is a greater probability of time-frequency collisions. These collisions will reduce the accuracy of sinusoid detection and parameterisation in sinusoidal analysis, leading to less fidelity in the resynthesised signal.

Despite the detrimental effects of sudden onsets and offsets, transients, rapid frequency modulations such as vibrato, lower accuracy for parameterisation of high-frequency sinusoids and reverberation, the signals are still well-represented by their sinusoidal trajectories. Resynthesis based on these trajectories leads to a reasonably accurate reproduction of the original signal, as demonstrated by the low error metrics presented in Table 5.2 and illustrated in Figure 5.1 and Figure 5.2.

5.4 Evaluation of Interference Suppression Methods

The musical interference suppression methods were applied using the following procedure to recover the desired signal from a two-signal mixture:

1. mix two signals with 1:1 mix ratio (i.e. $xMix = x1 + x2$)
2. sinusoidally analyse each individual signal using SMSTools and the phase computation method described in section 4.2.2
3. use sinusoidal analysis results from undesired signal to suppress undesired signal in the mix using filtering and subtraction methods described in sections 4.4.1 and 4.4.2
4. manually identify time-regions in the undesired signal corresponding to transients
5. use sinusoidal analysis results from desired signal to remove undesired transients in sinusoidal regions of the desired signal as described in section 4.6.

The parameters for sinusoidal analysis and phase trajectory computation were described in section 4.2. The results of the interference suppression were evaluated by comparing the processed mixed signals to the original desired signal and computing the error metrics described in section 5.2.

Twelve different signals were used for testing the interference suppression methods. The test signals are described in Table 5.3. Musical signals have time-varying amplitude and power, so it is not possible to state a global signal-to-interference ratio for the mixed signals. However, peak amplitude and average RMS power of each test signal are given in Table 5.4.

The twelve test signals were used to create seven different mixed signals. The mixed signals are specified in Table 5.5. The last three mixed signals in Table 5.5 simulate signals that might be picked up by spot microphones set up to record the two instruments playing the Bach Prélude. In fact, the power of the interfering signal in the mix is higher than what might be expected in real spot microphone signals. The interference suppression tests are dealing with a tougher suppression scenario than what might normally be encountered with real signals.

Signal	Description
flat400	machine-generated, single unmodulated 400 Hz tone
flat600	machine-generated, single unmodulated 600 Hz tone
vibrato400	machine-generated tone, 400 Hz centre frequency, frequency-modulated by a sine with amplitude = 30 Hz, frequency = 4.5 Hz
vibrato450	machine-generated tone, 450 Hz centre frequency, frequency-modulated by a sine with amplitude = 30 Hz, frequency = 4.5 Hz; frequency modulation is 180 out of phase from frequency modulation of vibrato400 signal, so that frequency troughs from vibrato450 signal overlap with frequency peaks of vibrato400 signal
flatSeries400	machine-generated, unmodulated harmonic series with fundamental frequency = 400 Hz, maximum frequency = 8800 Hz (22 sines), all harmonics have the same amplitude
flatSeries600	machine-generated, unmodulated harmonic series with fundamental frequency = 600 Hz, maximum frequency = 9000 Hz (15 sines), all harmonics have the same amplitude
cello	single bowed cello note
oboe	single oboe note
bachPrelude1	time-series of 7 monophonic guitar notes; lead part of a Bach Prélude
bachPrelude1 with reverberation	same as bachPrelude1 signal above, but with a long reverberation
bachPrelude2	time-series of 4 monophonic piano notes; harmony part of a Bach Prélude
bachPrelude2 with reverberation	same as bachPrelude2 signal above, but with a long reverberation

Table 5.3. Signals used for Interference Suppression Tests

Signal	Peak Amplitude	Average RMS Power
flat400	-9 dB	-12 dB
flat600	-9 dB	-12 dB
vibrato400	-9 dB	-12 dB
vibrato450	-9 dB	-12 dB
flatSeries400	-9 dB	-18 dB
flatSeries600	-9 dB	-17 dB
cello	-12 dB	-24 dB
oboe	-12 dB	-21 dB
bachPrelude1	-7 dB	-18 dB
bachPrelude1 with reverberation	-7 dB	-18 dB
bachPrelude2	-7 dB	-20 dB
bachPrelude2 with reverberation	-7 dB	-20 dB

Table 5.4. Peak Amplitude and Average RMS Power of Signals used for Interference Suppression Tests

Mixed Signal	Signal 1	Signal 2
flat400/600	flat400	flat600
vibrato400/450	vibrato400	vibrato450
flatSeries400/600	flatSeries400	flatSeries600
cello/oboe	cello	oboe
bachPrelude1/2	bachPrelude1	bachPrelude2
bachPrelude1/2rev	bachPrelude1	bachPrelude2 with reverberation
bachPrelude1rev/2	bachPrelude1 with reverberation	bachPrelude2

Table 5.5. Mixed Signals used for Interference Suppression Tests and their Composition

For each mixed signal in Table 5.5, both component signals were recovered by suppressing the other signal component in turn. The exception was for the mixed signals that contained a reverberated signal (bachPrelude1/2rev and bachPrelude1rev/2). These mixed signals were created to simulate the reverberation that might occur on the interfering signal in a real spot microphone signal. For these mixed signals, only the dry component signal was recovered by suppression of the reverberated interfering signal.

Steps 1 to 3 outlined at the start of this section constitute sinusoidal interference suppression. Steps 4 and 5 constitute transient interference suppression. Results of sinusoidal interference suppression alone (steps 1 to 3) as well as sinusoidal and transient interference suppression (steps 1 to 5) were evaluated. This allowed for assessment of the improvement in performance obtained when interfering transients were also suppressed. The results of sinusoidal interference suppression are discussed in the next section below. Performance improvement due to transient suppression is discussed in the following section.

5.4.1 Results of Sinusoidal Interference Suppression

The error statistics for the signals recovered from the mixed signals are presented in Table 5.6 for the filtering and subtraction sinusoidal interference suppression methods. The error curves and discussion for each mixed signal is given in the subsections to follow. However some general statements apply to all test cases:

- signals processed with the filtering method have a higher error than those processed with the subtraction method, which can be attributed to:
    1. attenuation of desired signal components if they coincide with the notch regions of the filters and
    2. phase distortion of the desired signal components around the notch frequencies (evident in *err* rather than *errM*)
- *errM* in filtered signals is much lower than *err* because the former does not penalise for the phase distortion introduced by the notch filters

- in most cases, the *err* and the *errM* for the signals processed using the subtraction method are similar, because there is very little phase distortion introduced by this method
- the signals processed using the subtraction method sound better than those processed using the filtering method.

The results of the sinusoidal interference suppression on each signal mixture are discussed in the following subsections. Each subsection will contain a discussion of quantitative results as well as a general description of the qualitative results (i.e. a description of what the recovered signals sounded like).

Mixed Signal	Recovered Signal	Method	Mean *err* (dB)	Median *err* (dB)	Mean *errM* (dB)	Median *errM* (dB)
flat400/600	flat400	filter	-26.14	-25.35	-41.29	-52.40
		subtract	-38.87	-52.48	-40.20	-52.74
	flat600	filter	-22.87	-21.98	-41.02	-45.98
		subtract	-38.85	54.90	-40.51	-54.93
vibrato400/450	vibrato400	filter	-14.36	-15.56	-18.47	-29.29
		subtract	-28.42	-29.36	-31.68	-35.14
	vibrato450	filter	-13.99	-14.64	-18.38	-28.23
		subtract	-36.06	-40.19	-39.93	-46.69
flatSeries400/600	flatSeries400	filter	-23.72	-22.78	-24.48	-23.49
		subtract	-39.84	-41.36	-40.78	-41.58
	flatSeries600	filter	-20.48	-19.53	-21.10	-20.10
		subtract	-41.56	-43.08	-42.71	-43.32
cello/oboe	cello	filter	-27.38	-28.27	-35.70	-36.05
		subtract	-51.72	-57.51	-52.99	-59.89
	oboe	filter	-24.15	-23.16	-32.98	-32.23
		subtract	-55.87	-56.10	-57.33	-58.13
bachPrelude1/2	bachPrelude1	filter	-19.80	-19.35	-21.59	-25.76
		subtract	-34.10	-49.26	-37.20	-52.19
	bachPrelude2	filter	-22.90	-24.78	-27.24	-30.05
		subtract	-30.80	-51.33	-34.37	-54.54
bachPrelude1/2rev	bachPrelude1	filter	-19.77	-19.41	-21.61	-25.97
		subtract	-33.65	-41.53	-36.60	-44.93
bachPrelude1rev/2	bachPrelude2	filter	-22.52	-23.88	-26.66	-29.31
		subtract	-30.45	-38.78	-33.89	-42.77

Table 5.6. Error Statistics for Recovered Signals

5.4.1.1 Results for flat400/600 Mix

5.4.1.1.1 Quantitative Analysis

The error curves for recovery each of the 400 Hz and 600 Hz tones from the flat400/600 mixed signal are shown in Figure 5.4.

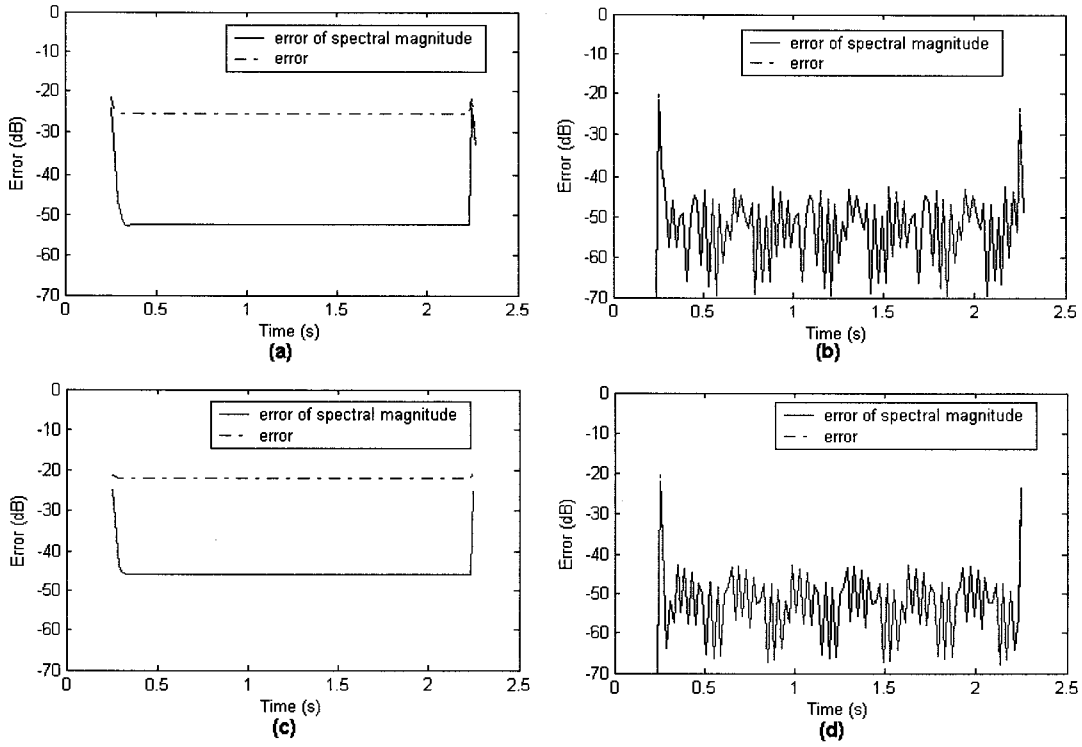


Figure 5.4. Error Curves for Signals Recovered from flat400/600 Mix. (a) flat400 using Filtering Method, (b) flat400 using Subtraction Method, (c) flat600 using Filtering Method, (d) flat600 using Subtraction Method.

The 400 Hz tone is recovered about equally well by the filtering and subtraction method, when only the spectral magnitude is considered. While the desired tone does not coincide with the notch of the filter, its phase is still distorted by the filter, accounting for the high error in Figure 5.4 (a) and (c).

5.4.1.1.2 Qualitative Analysis

In all signals other than the flat600 recovered by the subtraction method, the interfering tone could be heard (though it was very quiet). The flat600 signal recovered by the subtraction method had a low-frequency rumble that was suppressed

by filtering below 500 Hz. This rumble was supposedly introduced by subtraction of an imperfect interferer.

5.4.1.2 Results for vibrato400/450 Mix

5.4.1.2.1 Quantitative Analysis

The error curves for recovery each of the 400 Hz and 450 Hz vibrato tones from the vibrato400/450 mixed signal are shown in Figure 5.5.

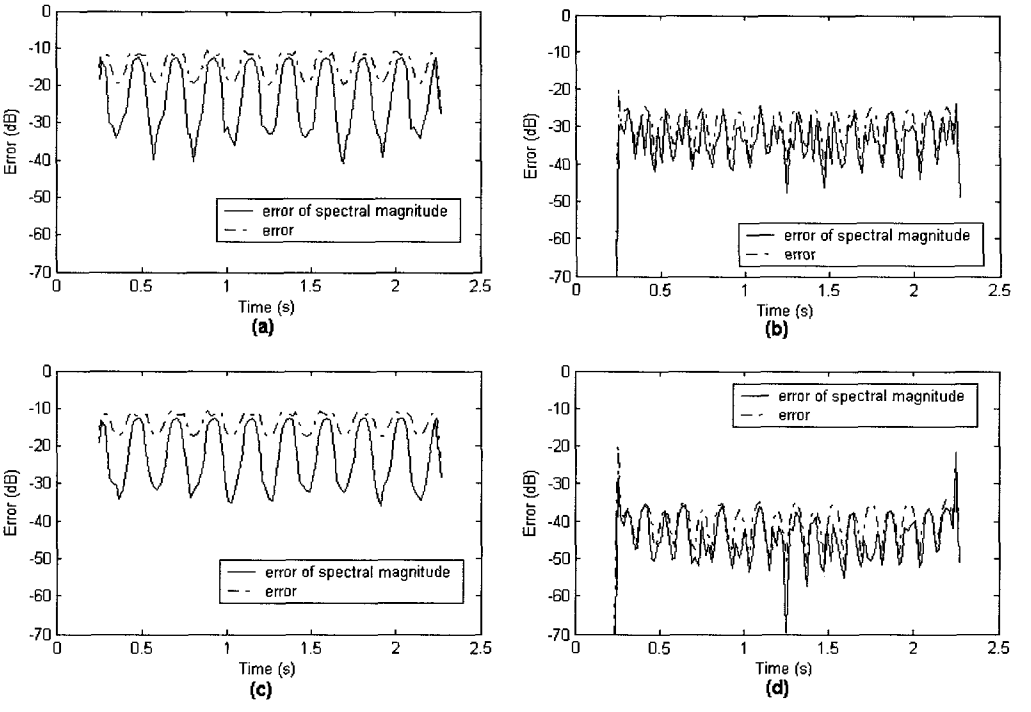


Figure 5.5. Error Curves for Signals Recovered from vibrato400/450 Mix. (a) vibrato400 using Filtering Method, (b) vibrato400 using Subtraction Method, (c) vibrato450 using Filtering Method, (d) vibrato450 using Subtraction Method.

The *err* and *errM* has a periodicity related to the vibrato frequency. Due to the rapidly varying frequency and phase of the vibrato signals, the accuracy of the sinusoidal parameter estimation is reduced, affecting both filtering and subtraction methods. Furthermore, there is a periodic time-frequency collision of the two vibrato tones, as illustrated in Figure 5.6. The desired vibrato tone periodically coincides with the notch of the filter designed to remove the interfering vibrato tone, causing the periodic peaks of the error metrics in Figure 5.5 (a) and (c).

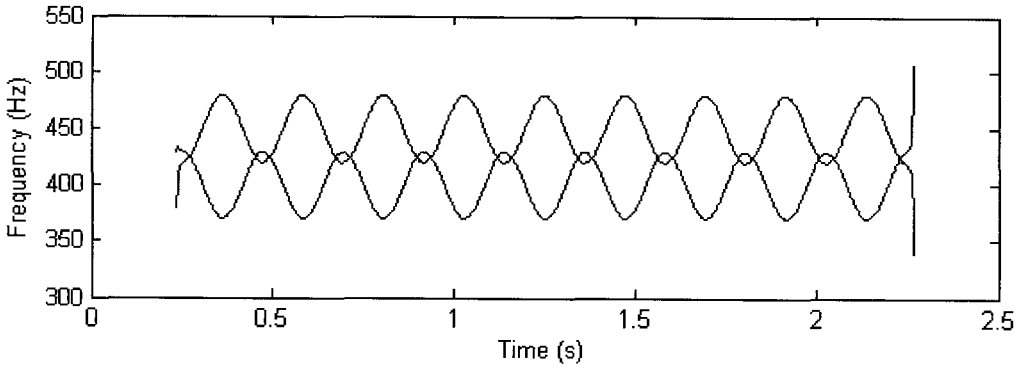


Figure 5.6. Frequency Trajectories for 400 Hz and 450 Hz Vibrato Signals

5.4.1.2.2 Qualitative Analysis

Due to the time-frequency collisions of the desired and undesired signal components, the filtering method introduced a very audible amplitude modulation. Despite this amplitude modulation, the signals processed using the filtering method generated a purer-sounding signal (i.e. there was less of the interfering signal audible in the result than the signals processed using the subtraction method). The signals processed using the subtraction method did not have amplitude modulation. However the interfering signals were more audible in the result, creating a reverberant sound.

5.4.1.3 Results for flatSeries400/600 Mix

5.4.1.3.1 Quantitative Analysis

The error curves for recovery each of the 400 Hz and 450 Hz harmonic series from the flatSeries400/600 mixed signal are shown in Figure 5.7.

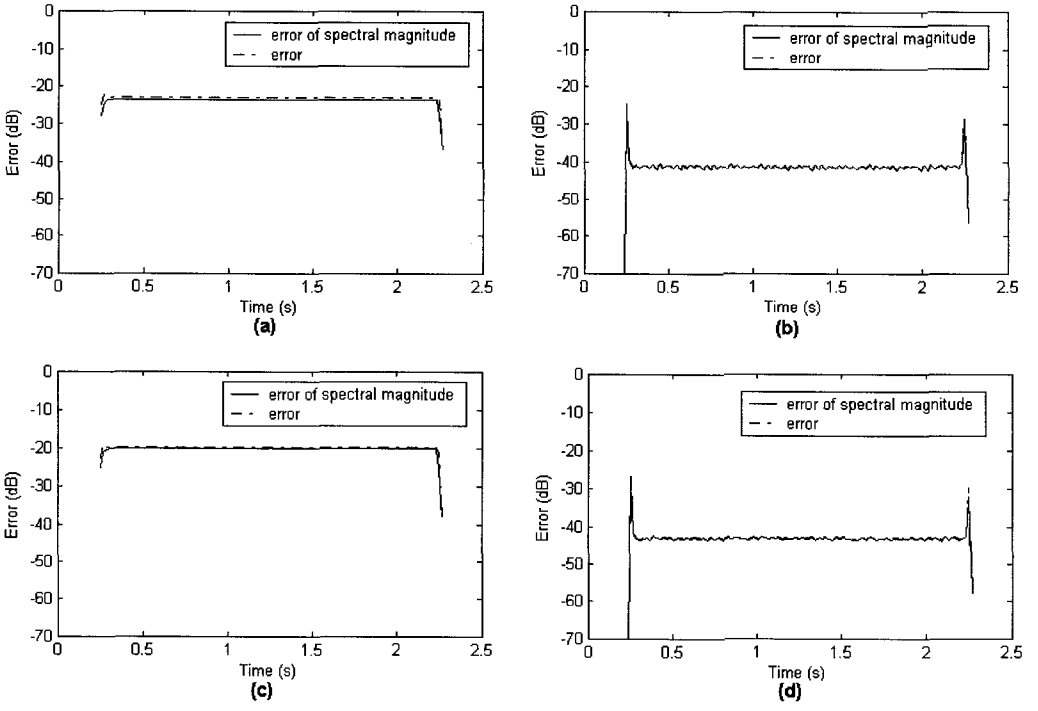


Figure 5.7. Error Curves for Signals Recovered from flatSeries400/600 Mix. (a) flatSeries400 using Filtering Method, (b) flatSeries400 using Subtraction Method, (c) flatSeries600 using Filtering Method, (d) flatSeries600 using Subtraction Method.

For the 400 Hz harmonic series, every third harmonic overlaps with a harmonic of the 600 Hz signal. For the 600 Hz harmonic series, every other harmonic overlaps with a harmonic of the 400 Hz signal. All of these time-frequency collisions of the two signals results in a large error in the filtered signals. The recovered 600 Hz series is more severely affected than the 400 Hz because more of the former's energy is lost due to the collisions. Half of the 600 Hz series' components coincide with the notches designed to suppress the 400 Hz series, while a third of the 400 Hz series' components coincide with the notches designed to suppress the 600 Hz series.

The subtraction method does not suffer as much from all these collisions because the sinusoidal parameters were estimated on each harmonic series in isolation.

5.4.1.3.2 Qualitative Analysis

The loss of energy of some of the harmonics in the signals processed using the filtering method produced very noticeable distortions in timbre when compared to the original signals. There was very little timbral distortion introduced into the signals processed with the subtraction method.

5.4.1.4 Results for cello/oboe Mix

5.4.1.4.1 Quantitative Analysis

The error curves for recovery of the cello and oboe signals from the cello/oboe mixed signal are shown in Figure 5.8.

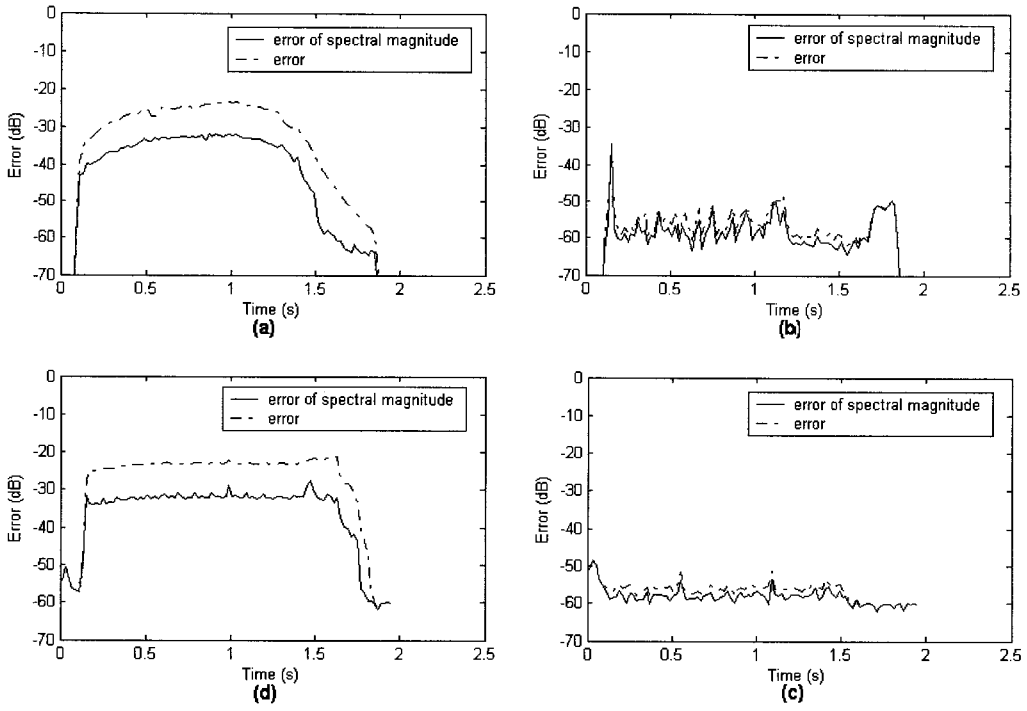


Figure 5.8. Error Curves for Signals Recovered from cello/oboe Mix. (a) cello using Filtering Method, (b) cello using Subtraction Method, (c) oboe using Filtering Method, (d) oboe using Subtraction Method.

The original cello signal ended at about 1.5 seconds. This explains the drop in error after 1.5 s for the signals processed using the filtering method (Figure 5.8 (a) and (c)). When the cello signal was isolated, the notch filters stop damaging the desired components of the cello after these components die out. When the oboe signal was isolated the filters terminate shortly after the end of the undesired cello signal dies out, leaving the remainder of the oboe signal untouched.

The subtraction method did a very good job with this test case (Figure 5.8 (b) and (d)), indicating that both signals were well modeled by their sinusoidal components. This is not surprising since both oboe and cello signals had stable frequencies. The error spike at the beginning of the cello signal error curve is due to inaccurate

sinusoidal parameter estimation and localisation caused by the rapid onset of the oboe tone. Recall that the cello signal has a gradual onset, allowing for accurate parameterisation around its onset, and keeping the error in the resynthesised signal low.

5.4.1.4.2 Qualitative Analysis

The error spike due to the onset of the oboe is audible in the recovered cello signals. Since this occurs after the onset of the cello note and therefore well into the sinusoidal region of the cello, this error transient is easily removed by the transient suppression strategy described in section 4.6.

The cello signal recovered by the filtering method has a noticeable timbral distortion when compared to the original signal. Timbral distortion is not noticeable in the cello signal recovered by the subtraction method. Further, timbral distortion is also not very obvious in either of the recovered oboe signals. However, the oboe signal recovered using the filtering method does have very obvious artefacts, which are not heard in the same signal recovered using the subtraction method.

5.4.1.5 Results for Bach Prélude Mixes

5.4.1.5.1 Quantitative Analysis

The error curves for recovery of each part of the bachPrelude1/2 mixed signal are shown in Figure 5.9.

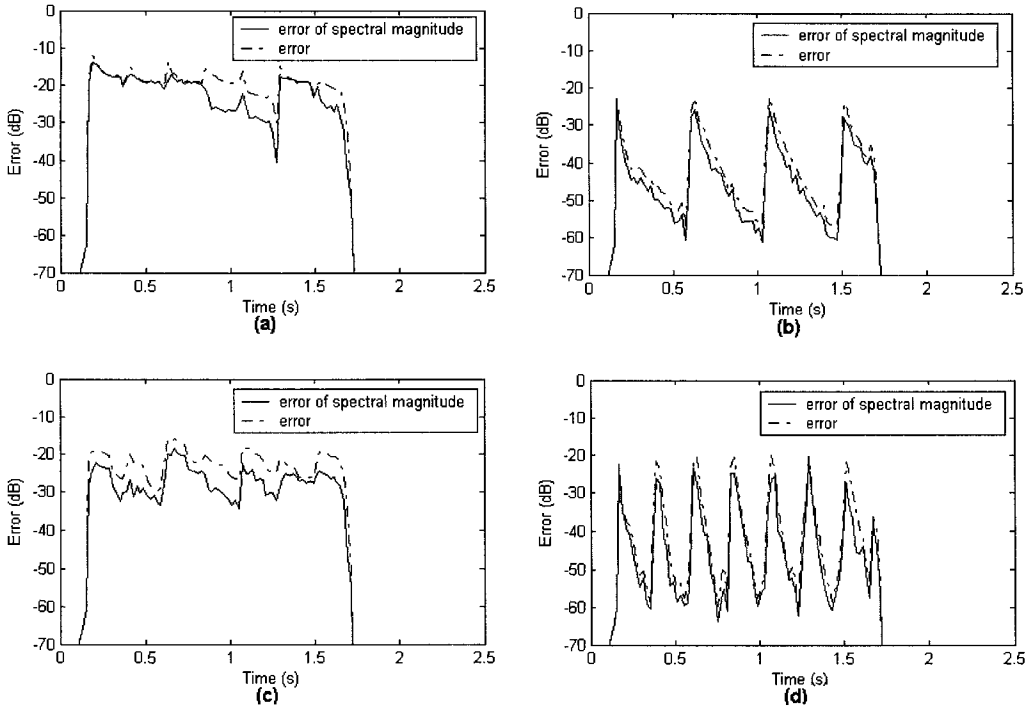


Figure 5.9. Error Curves for Signals Recovered from bachPrelude1/2 Mix. (a) bachPrelude1 using Filtering Method, (b) bachPrelude1 using Subtraction Method, (c) bachPrelude2 using Filtering Method, (d) bachPrelude2 using Subtraction Method.

The results in Figure 5.9 are explained using knowledge of the fundamental frequencies of each note in each part. The fundamental frequencies and duration of the notes of both parts of the Bach Prélude are illustrated in Figure 5.10 and Figure 5.11 and listed in Table 5.7.

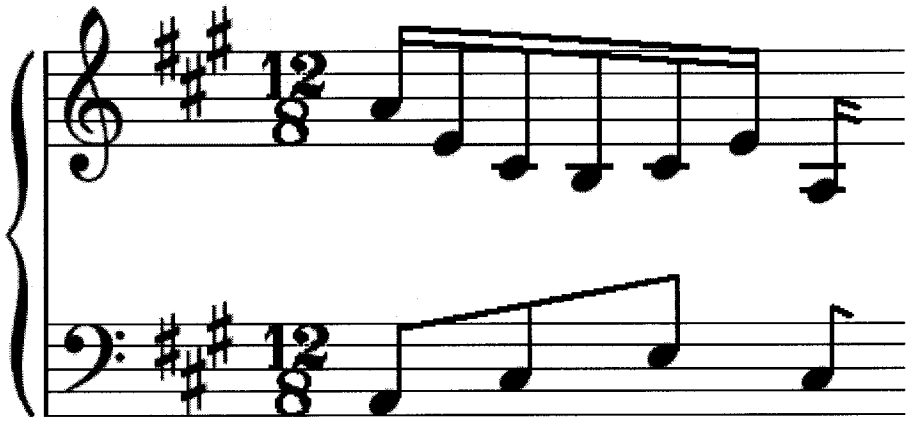


Figure 5.10. Notes for Two-Part Bach Prelude in Common Musical Notation

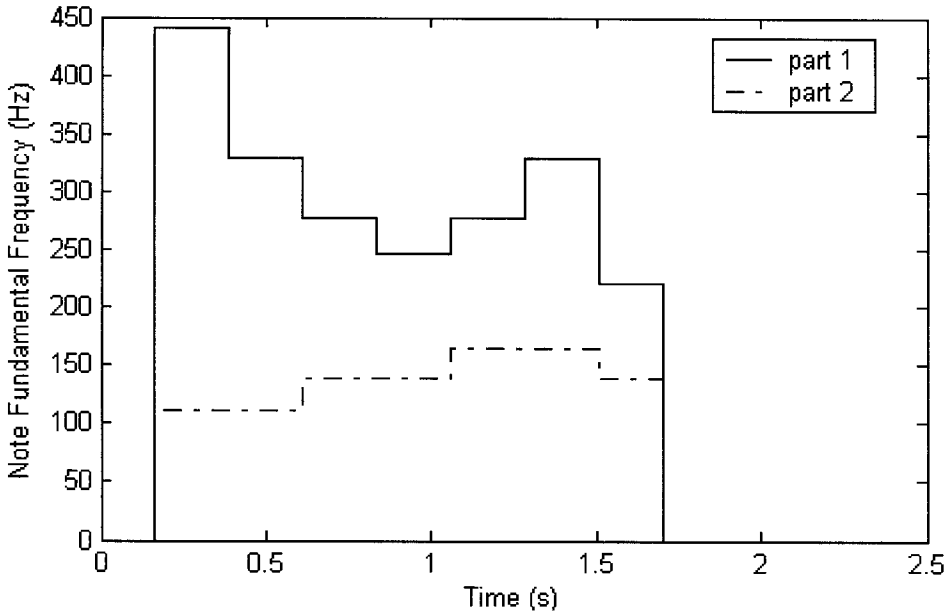


Figure 5.11. Notes for Two-Part Bach Prelude

Time Interval	Part 1 Fundamental	Part 2 Fundamental
0.16 - 0.39 s	440.0 Hz	110.0 Hz
0.39 - 0.61 s	329.7 Hz	110.0 Hz
0.61 - 0.83 s	277.2 Hz	138.6 Hz
0.83 - 1.06 s	246.9 Hz	138. 6 Hz
1.06 - 1.28 s	277.2 Hz	164.8 Hz
1.28 - 1.51 s	329.7 Hz	164.8 Hz
1.51 - 1.70 s	220.0 Hz	138.6 Hz

Table 5.7. Fundamental Frequencies of Notes in Bach Prelude

From Figure 5.11, there are seven notes for part 1 and four notes for part 2. The notes for part 1 change twice as frequently as those for part 1. The sinusoids of notes 1, 2, 3 and 6 of part 1 collide with sinusoids of notes 1,2 and 3 of part 2.

The collision of nearly all of the part 1 sinusoids with the part 2 sinusoids results in the loss of most of the energy of the recovered part 1 signal over notes 1-3 and 6. This is indicated by the high error in Figure 5.11 (a) during the affected intervals. There is also some energy loss (and higher error) for the recovered part 2 signal during the same intervals (Figure 5.11 (c)). The energy loss is not as severe because some of the sinusoids of the part 2 signal do not collide with sinusoids of the part 1 signal.

The subtraction method performs much better than the filtering method. The error peaks correspond to transient components of the undesired signals that are not subtracted out of the mixed signals. The transients of the part 2 signal overlap with those of the part 1 signal, so the transient suppression method proposed in section 4.6 cannot be applied to reduce the error peaks in Figure 5.11 (b)). Since the notes of part 1 occur twice as frequently as those of part 2, every odd transient of part 1 occurs during a sinusoidal region of part 2. In these cases, the transient suppression method can be applied to reduce every odd error peak in Figure 5.11 (d)) and improve the quality of the recovered part 2 signal. The result of applying the transient suppression method to this case is discussed in section 5.4.2

The error curves for recovery of part 1 from the bachPrelude1/2rev mixed signal are shown in Figure 5.12. The error curves for recovery of part 2 from the bachPrelude1rev/2 mixed signal are shown in Figure 5.13. The error curves show that the performance of the filtering method is similar for suppression of a dry versus a reverberant interferer. The performance of the filtering method is poor and the error is dominated by the energy loss of desired signal components. The performance of the subtraction method when suppressing a reverberant signal is degraded somewhat from the performance when suppressing the dry signal (Figure 5.11). In particular, the error in the valleys is larger and grows larger over time, for the reasons discussed in section 5.3.

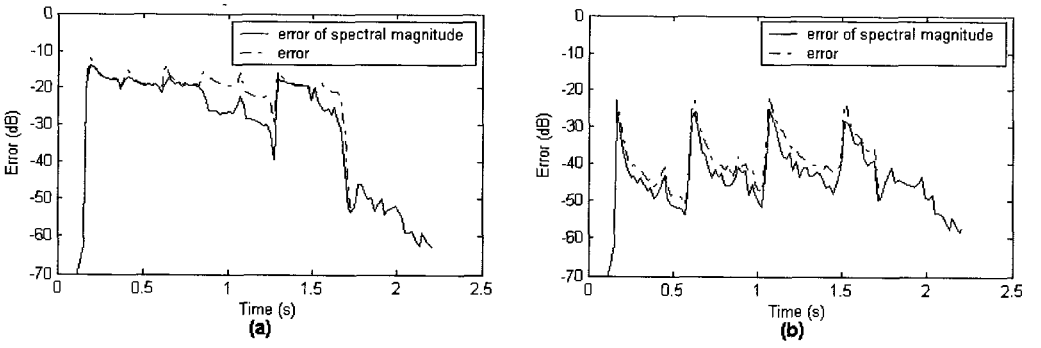


Figure 5.12. Error Curves for Signals Recovered from bachPrelude1/2rev Mix. (a) bachPrelude1 using Filtering Method, (b) bachPrelude1 using Subtraction Method

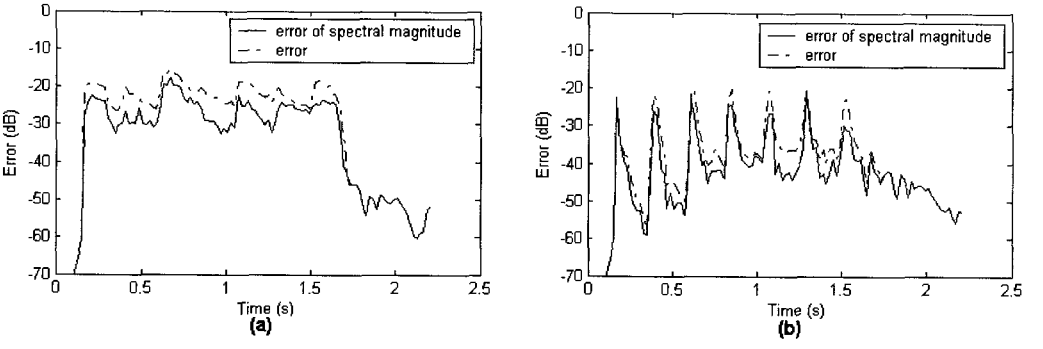


Figure 5.13. Error Curves for Signals Recovered from bachPrelude1rev/2 Mix. (a) bachPrelude2 using Filtering Method, (b) bachPrelude2 using Subtraction Method

5.4.1.5.2 Qualitative Analysis

The loss of energy in notes 1-3 and 6 of the part 1 signals recovered using the filtering method is very noticeable. The recovered signals in these regions are very distorted in both level and timbre when compared with the original part 1 signal. The level and timbre of the part 1 signals recovered using the subtraction method sound very close to the original part 1 signal. Significant level and timbre differences were also heard in the part 2 signals recovered using the filtering method, but these differences were not as drastic as those of the filtered part 1 signals. This is because only a subset of the part 2 sinusoids collided with the part 1 signal sinusoids, while all part 1 sinusoids collided with part 2 sinusoids over the problem regions.

In the part 1 signals recovered using the subtraction method, notes 1, 3, 5 and 7 sounded much noisier than notes 2, 4 and 6. This is because the odd numbered notes coincided with the beginning of the piano notes in part 2. Piano notes are noisy after

an onset, with the noise decaying over time. This is illustrated in Figure 5.14 by two PSDs of a piano note, averaged over 100 ms: one computed after the onset, the other computed 0.3 s after the onset.

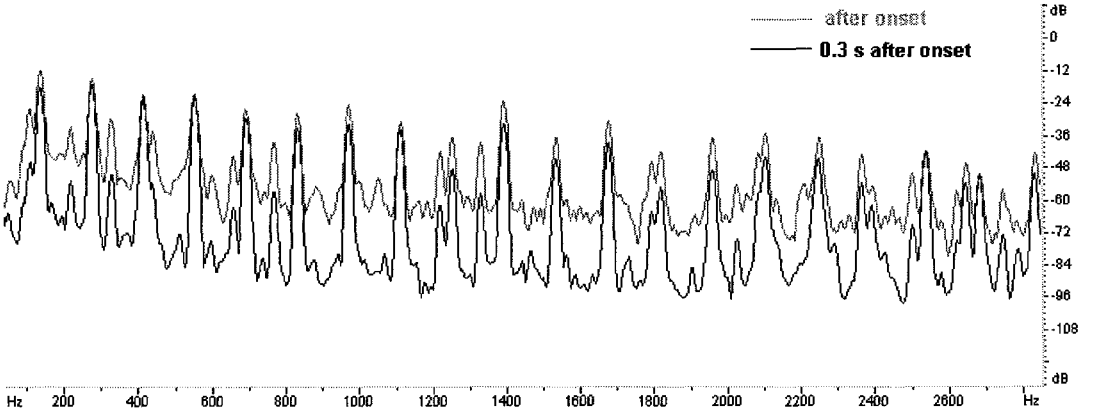


Figure 5.14. Time-Averaged PSDs of a Piano Note

The noise after the onset is broadband and therefore not modeled by sinusoidal analysis. Therefore, this noise is not removed from the mix by the interference suppression methods.

The interfering transients are audible in the recovered signals. When interfering transients coincide with desired transients, the interfering transients are somewhat masked by the desired transients, making them less bothersome. However, the interfering transients of the part 1 signal that don't coincide with the transients of the part 2 signals are very noticeable and bothersome in the recovered part 2 signals. The transient interference suppression method described in section 4.6 and evaluated in section 5.4.2 was designed to deal with these most annoying cases.

The signals recovered from the mixtures where the interfering signals were reverberated sounded noisier than those recovered from the mixture with a dry interferer. The reverberation tail was also noticeable in the signals recovered from the mixtures where the interfering signals were reverberated. The level of the tail was low and did not sound pitched and it was assumed that the tail was due to reverberation of the aharmonic component of the interfering signal.

5.4.2 Improvement due to Transient Interference Suppression

When interfering transients occur during sinusoidal regions of the desired signal, the transients can be eliminated using the method described in section 4.6. The recovered bachPrelude2 (part 2 of Bach Prélude) signals are good candidates for this method. The guitar notes of part 1, the interfering signal, occur at twice the rate as the piano notes of part 2. This means that the onsets of notes 2, 4 and 6 of part 1 occur over sustained, sinusoidal regions of part 2 (Figure 5.11). The transients at the start of notes 2, 4 and 6 are removed from the recovered part 2 signals by the transient interference suppression method described in section 4.6.

As discussed in section 4.5, an automatic transient detector was not automated as part of the work discussed in this thesis. The transient regions of notes 2, 4 and 6 of part 1 were manually identified and are listed in Table 5.8.

Note Number	Start of Transient	End of Transient
2	0.37 s	0.42 s
4	0.82 s	0.87 s
6	1.27 s	1.32 s

Table 5.8. Transient Locations for Selected Notes in Bach Prélude Part 1

The part 2 signal recovered from the bachPrelude1/2 mix (no reverberation on interferer) using the subtraction method was used to demonstrate the results of transient interference suppression. The error curves for the recovered part 2 signal before and after transient interference suppression around the times listed in Table 5.8 are illustrated in Figure 5.15. The error in the regions listed in Table 5.8 is reduced by 15-20 dB.

The aesthetic result of the transient suppression is significant. The onsets of notes 2, 4 and 6 of part 1 are barely audible in the recovered part 2 signal processed to remove the transients. The sinusoidal components of the interfering notes are somewhat noticeable around the note onsets, but the level of this interference is very low.

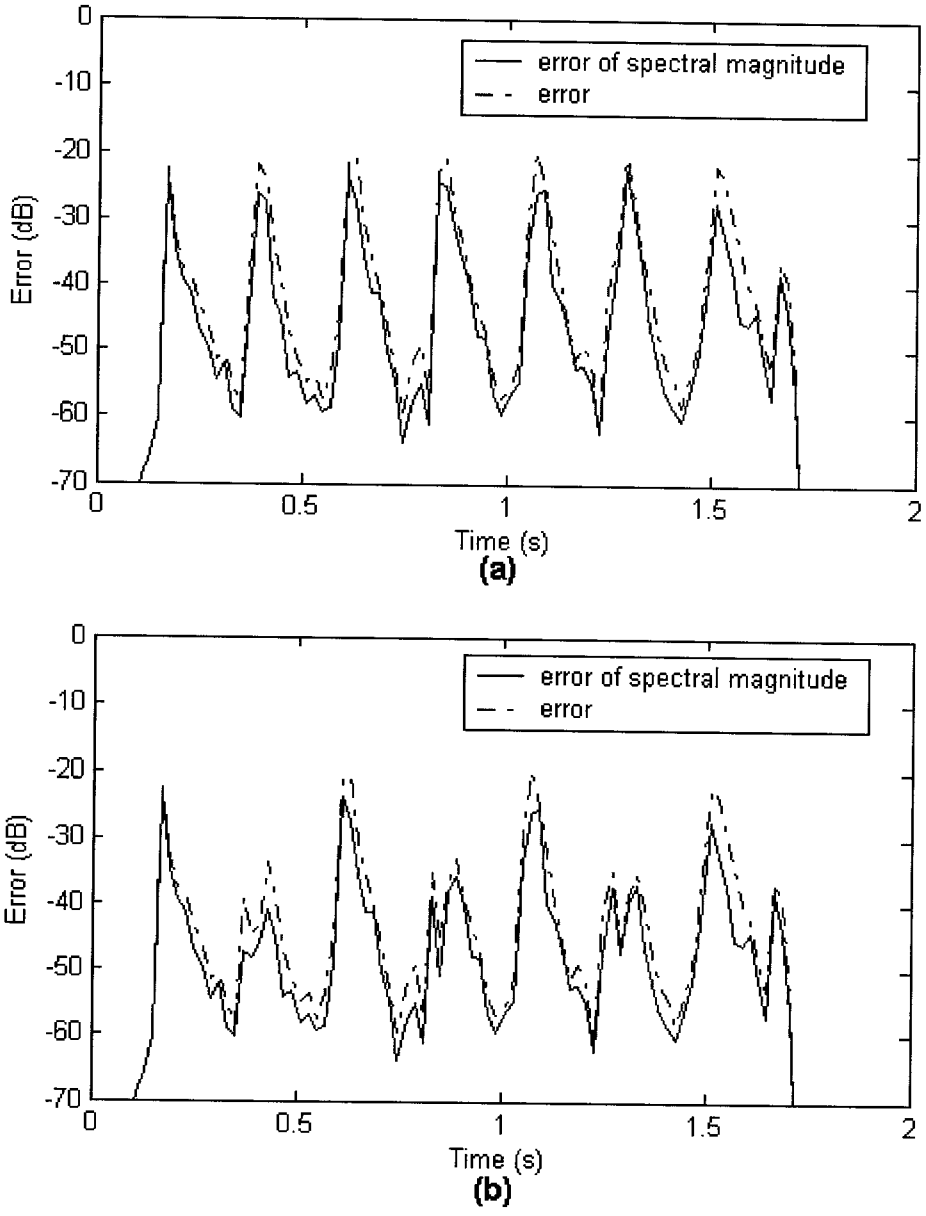


Figure 5.15. Error Curves for Part 2 Signal Recovered from bachPrelude1/2 Mix. (a) Before Transient Suppression, (b) After Transient Suppression.

Chapter 6

Conclusions

6.1 Summary

In this thesis, methods for purification of musical signals recorded by a spot microphone were presented and evaluated. Spot microphones are used to record the sound of one musical instrument playing in an ensemble. In addition to the desired sound, the spot microphone inevitably captures some of the sound from neighbouring musical instruments. The purification methods attempt to suppress the signals due to undesired pitched musical instruments, collectively termed the interference. The purification of spot microphone signals has a number of applications in musical recording engineering.

The interference suppression methods are based on a sinusoidal model for the desired and interfering signals. The sinusoidal model represents a signal as a collection of sinusoids with slowly evolving amplitude, frequency and phase. This model is valid for pitched musical signals as pitched musical signals are dominated by slowly-evolving sinusoidal components. Typically these sinusoidal components form a harmonic series, but due to non-linearities in the musical instrument, a perfect harmonic relationship does not always hold. Examples of instruments that generate pitched musical signals include pianos, guitars, cellos, and oboes. It was shown in this thesis that the sinusoidal model is valid for such instruments. The validity of the model was proved by the low error between the original musical signal and the signal resynthesised from sinusoids detected from the original signal.

While pitched musical instruments are dominated by their sinusoidal components, they also tend to have short transients at note onsets as well as a broadband, aharmonic component during the sustained part of the note. The sinusoidal model

accurately represents the sinusoidal component, but it does not represent the transient and aharmonic components. This is because these components require a very large number of sinusoids for their representation. Since the sinusoidal model does not represent the transient and aharmonic components of the musical signals, the interference suppression methods that are based on this model do not address these components. The consequence is that the interfering transients and aharmonic signals remain in the recovered desired signal.

Two sinusoidal interference suppression methods were proposed and evaluated in this thesis. Both methods rely on the sinusoidal representation of the interfering signal. The sinusoidal representation was derived by SMSTools, a free software program that performs sinusoidal analysis of a signal. In sinusoidal analysis, the slowly-evolving sinusoidal components are detected and their time-varying amplitudes, frequencies and phases are estimated from short-time discrete Fourier transform frames. An independent phase estimation algorithm was developed to supplement SMSTools because the phase estimates produced by SMSTools were not sufficiently accurate.

The filtering method for interference suppression involves the use of time-varying notch filters to suppress the interfering sinusoids. Each parametric notch filter was set to track the time-varying frequency trajectory of a sinusoid over its lifetime. The notch has a fixed attenuation of 50 dB at its centre frequency and a very narrow bandwidth to prevent attenuation of nearby desired signal components. The advantage of the filtering method is that it relies on accurate parameterisation of the interfering frequency trajectories only. Amplitude and phase trajectories of the interfering sinusoids are not required by the filtering method. The disadvantage of the filtering method is that it does not gracefully address time-frequency collisions of desired and interfering sinusoids. If a desired sinusoid collides with an interfering sinusoid, the desired sinusoid is suppressed by the filter, causing energy loss and timbral changes in the recovered desired signal.

The subtraction method for sinusoidal interference suppression involves synthesising an estimate of the interference and subtracting it from the mixed signal. The subtraction method follows from [8], where it was proposed for the related task

of signal separation. The estimate of the interference is generated by sinusoidal synthesis from the sinusoidal model of the interference produced by SMSTools. The sinusoidal synthesis method is based on phase-driven oscillators. The time-varying amplitude and phase parameters for the oscillator are derived from the amplitude, frequency and phase trajectories of the interfering sinusoids. The subtraction method is dependent on producing an estimate of the interference that is phase-matched to the interference in the mixed signal. Unfortunately, the phase information reported by SMSTools does not result in a phase-matched estimate of the interference. A successful correlation-based phase estimator was designed to produce proper phase trajectories. The sensitivity of the subtraction method to errors in sinusoidal parameters is a notable disadvantage. The method is particularly sensitive to phase, where large errors actually increase the level of the interference rather than suppress it. On the other hand, if accurate sinusoidal parameters are available, the subtraction method causes far less distortion to the desired signal than the filtering method, particularly when desired and interfering sinusoids are very close in time-frequency.

In the evaluation of the sinusoidal interference suppression methods, good estimates of the interfering sinusoidal parameters were available. Therefore, it was not surprising that the subtraction method out-performed the filtering method. As expected, energy loss and timbral changes were a common problem in the signals recovered by the filtering method. A problem common to both methods is the interfering transients and aharmonic signals remaining in the recovered desired signals. This was also expected because of the known limitations to the sinusoidal interference suppression methods.

From an aesthetic perspective, the most objectionable consequences of the limitations of the sinusoidal interference suppression methods are the interfering transients remaining in sinusoidal regions of the desired signal. To address this most annoying limitation, a transient interference suppression method was proposed and evaluated. The transient interference suppression method involves deletion of the signal segment containing the interfering transient and replacing the signal segment with an estimate of the desired signal. The estimate of the desired signal over the transient regions is derived by sinusoidal synthesis of the desired signal from its

sinusoidal model. This method cannot be applied to the removal of interfering transients coinciding with desired transients because the desired transients are not represented by the sinusoidal model and therefore cannot be resynthesised. In circumstances where the transient interference suppression method can be applied, it greatly improves the aesthetic quality of recovered desired signal.

6.2 Future Work

The musical interference suppression methods described in this thesis show promise for the application to purification of spot microphone signals. However, there are many issues to be addressed before the methods may be considered complete and practical. Some of these issues are very interesting and challenging and offer the possibility to those that take them on to make some useful and unique contributions. Some of outstanding issues are enumerated in this section.

1. The automatic classification of sinusoids detected by sinusoidal analysis into desired and interference has not been implemented. This is a very challenging problem that has received much attention in the literature. The unique properties of a collection of spot microphone signals from the same performance will likely be very useful in development of a classification strategy uniquely tailored to this application.
2. Related to issue 1. above, the challenge of identifying sinusoids in mixed signals has not been addressed. The experiments described in this thesis involved sinusoidal analysis on monophonic, reference signals. Detection of sinusoids due to multiple sources is bound to be error-prone. To cope with this scenario, some have suggested iterative sinusoidal analysis ([8], [64]-[66]), where the most predominant sinusoid in the signal is identified then subtracted from the signal before the next iteration of analysis.
3. The automatic identification of interfering transient regions has not been implemented, but the suggested process was manually applied with good results.

4. The suppression of interfering transients that coincide with desired transients was not investigated. The suppression of interfering aharmonic components was also not investigated. These components have traditionally been overlooked by CASA-based signal separation strategies. An interesting option would be to remove all sinusoidal components, desired and interference, from the mixed signal using the subtraction method. The residual signal, containing desired and interfering transients and aharmonic components could be separated by independent component analysis. The desired sinusoidal components could then be added to the desired transient and aharmonic components using sinusoidal synthesis.
5. The destruction of desired sinusoidal components by the notch filters is not addressed in the filtering method. One possibility is to not filter over time-frequency regions that contain a desired and interfering sinusoid. Due to source/microphone geometry, the desired sinusoid is likely to be more energetic than the undesired sinusoid; the undesired sinusoid may not even be perceived due to the effects of auditory masking. Alternatively, it is theoretically possible to restore desired sinusoids by resynthesis, but this would involve careful estimates of the amount of attenuation and phase distortion in the recovered signal so that the resynthesised sinusoid is added with the correct phase and amplitude.
6. A perceptually based evaluation metric would be nice to have for quantifying the aesthetic properties of the recovered signals.

Appendix A

Some Properties of Random Processes

In this section some basic properties of random processes and variables are reviewed as background for understanding BSS techniques. The definitions and explanations below come mostly from [23], but any textbook on random processes should cover the same principles.

A.1 Random Processes and Random Variables

A random process is an ensemble of time functions:

$$X(t,s) \qquad (A.1),$$

where s is the outcome index and t is the time index. When t is fixed, the collection of numbers over all s represents samples of a random variable. Estimates of the statistics of the random variable can be made using this collection of numbers. In the BSS problem, each mixed signal represents only one time function of the ensemble in a random process. In order to estimate statistics of the random variable at a particular time t_1, we require the signal to be ergodic over an interval surrounding t_1.

A.2 Stationarity

A random process is considered stationary to the n^{th} order if the n^{th} order joint probability density function (PDF), given by

$$f_X(x_1,\ldots,x_n;t_1,\ldots,t_n) = \frac{\partial^n F_X(x_1,\ldots,x_n;t_1,\ldots,t_n)}{\partial x_1 \ldots \partial x_n} \qquad (A.2),$$

does not change with time. The function $F_X(x_1,\ldots,x_n;t_1,\ldots,t_n)$ is the nth order joint distribution function. The nth order stationarity condition implies that all statistics up to the nth order do not vary over absolute time.

A.3 Ergodicity

A random process that is nth order ergodic must also be nth order stationary. When a random process is nth order ergodic all statistics up to the nth order may be estimated by time averages. Ergodicity is required to estimate statistics when we have only one time function of a random process, as is the case for the observed signal mixtures in the BSS problem.

A.4 Statistical Independence

Statistical independence will be defined here between two random variables rather than two random processes to simplify notation. Formally, M random variables are statistically independent if their joint PDF are factorable:

$$f(x_1,\ldots x_M)=\prod_{i=1}^{M} f(x_i) \qquad (A.3).$$

In practical problems, cross-statistics are more often used than PDF, so the implications of statistical independence on the cross-statistics of two signals is of great interest:

$$E\left[x_i^u(t)x_j^{*v}(t+\tau)\right]=E\left[x_i^u(t)\right]E\left[x_j^{*v}(t+\tau)\right], \qquad i\neq j \qquad (A.4).$$

In (A.4) time signals have replaced random variables. The complex designator, *, is a formality. Most signals, including acoustic signals, are real-valued. The time signal may be considered a random variable for the purposes of extracting statistics up to the nth order if the random process, of which the signal is one realisation, is ergodic to the nth order. The order of the cross-statistic in (A.4) is given by $n = u+v$. The nth order cross-statistics are zero when all of the following conditions hold:

- the random variables are statistically independent up to at least order n
- the random variables are stationary up to at least order n
- the random variables have zero mean.

Bibliography

[1] G. Theile. Natural 5.1 Music Recording Based on Psychoacoustic Principals. In *Proceedings of the Audio Engineering Society 19th International Conference*, 2001.

[2] M. Wohr et al. Room-Related Balancing Technique: A Method for Optimizing Recording Quality. *Journal of the Audio Engineering Society*, vol. 38, no. 9, 1991.

[3] T. A. Leonard. Time-Delay Compensation of Distributed Multiple Microphones in Recording: An Experimental Example. In *Proceedings of the 95th Audio Engineering Society Convention, preprint 3710*, Sept. 1993

[4] J. Wuttke. General Considerations on Audio Multichannel Recording. In *Proceedings of the Audio Engineering Society 19th International Conference*, 2001.

[5] D. Griesinger. The Psychoacoustics of Listening Area, Depth and Envelopment in Surround Recordings and their Relationship to Microphone Technique. In *Proceedings of the Audio Engineering Society 19th International Conference*, 2001.

[6] J. A. Moorer and J. H Vad. Towards a Rational Basis for Multichannel Music Recording. In *Proceedings of the 104th Audio Engineering Society Convention, preprint 4680*, May 1998.

[7] R. McAulay and T. Quatieri. Speech Analysis/Synthesis Based on a Sinusoidal Representation. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 34(4):744-754, August 1986.

[8] T. Tolonen. Methods for Separation of Harmonic Sound Sources Using Sinusoidal Modeling. In *Proceedings of the 103rd Audio Engineering Society Convention, preprint 4958*, April 1999.

[9] T. Virtanen and A. Klapuri. Separation of Harmonic Sound Sources Using Sinusoidal Modeling. In *Proceedings IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 2, pages 765-768, June 2000.

[10] M. Tohyama, R. H. Lyon and T. Koike. Source Waveform Recovery in a Reverberant Space by Cepstrum Dereverberation. In *Proceedings IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 1, pages 157-160, April 1993.

[11] L. Jong-Hwan and L. Soo-Young. Blind Dereverberation of Speech Signals Using Independence Transform Matrix. In *Proceedings of the Joint International Conference on Neural Networks*, vol. 2, pages 1453-1457, July 2003.

[12] T. Nakatani and M. Miyoshi. Blind Dereverberation of Single Channel Speech Signal Based on Harmonic Structure. In *Proceedings IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 1, pages 92-95, April 2003.

[13] L. Parra and P. Sajda. Blind Source Separation via Generalized Eigenvalue Decomposition. *Journal of Machine Learning Research*, vol. 4, pages 1261-1269, 2003

[14] A. Belouchrani, K. Abed-Meraim, J.-F. Cardoso and E. Moulines. A Blind Source Separation Technique Using Second-Order Statistics. *IEEE Transactions on Signal Processing*, vol. 45, pages 434-444, February 1997.

[15] A. Hyvarinen and E. Oja. Independent Component Analysis: Algorithms and Applications. *Neural Networks*, vol. 13, pages 411-430, 2000.

[16] L. Parra and C. Spence. Convolutive blind separation of non-stationary sources, *IEEE Transactions on Speech and Audio Processing*, vol.8, no. 3, pages 320-332, May 2000

[17] R. Zhang and M.K. Tsatsanis. A Second-Order Method for Blind Separation of Non-Stationary Sources. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing,* vol. 5, pages 2797-2800, 2001

[18] D.C.B. Chan, P.J.W. Rayner and S.J. Godsill. Multi-Channel Signal Separation. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 2, pages 649-652, 1996.

[19] E. Weinstein, M. Feder and A.V. Oppenheim. Multi-channel signal separation by decorrelation. *IEEE Transactions on Speech and Audio* Processing, vol.1, no. 4, pages 405-413, Oct 1993.

[20] A. J. Bell and T. J. Sejnowski. An Information Maximisation Approach to Blind Separation and Blind Deconvolution. *Neural Computation*, vol. 7, no. 6, pages 1129-1159, 1995.

[21] J.-F. Cardoso. Source Separation Using Higher Order Moments. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 4, pages 2109-2112, 1989.

[22] S. Shamsunder and G.B. Giannakis. Multichannel Blind Signal Separation and Reconstruction. *IEEE Transactions on Speech and Audio Processing*, vol. 5, no. 6, pages 515-528, 1997.

[23] P.Z. Peebles Jr., *Probability, Random Variables and Random Signal Principles (4^{th} Edition)*. McGraw-Hill Companies Inc., New York, 2001.

[24] J.-F. Cardoso. High-order Contrasts for Independent Component Analysis. *Neural Computation*, vol. 11, pages 157-192, 1999.

[25] K. Torkkola. Blind Separation of Delayed Sources Based on Information Maximization. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 6, pages 3509-3512, 1996.

[26] K. Torkkola. Blind Separation of Convolved Sources Based on Information Maximization. In *Proceedings of the IEEE Signal Processing Society Workshop on Neural Networks for Signal Processing*, pages 423-432, 1996.

[27] A.M. Engebretson. Acoustic Signal Separation of Statistically Independent Sources Using Multiple Microphones. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing,* vol. 2, pages 343-346, 1993.

[28] P. Smaragdis. Blind Separation of Convolved Mixtures in the Frequency Domain. *Neurocomputing*, vol. 22, pages 21-34, 1998.

[29] V. Capdevielle, C. Serviere and J.L. Lacoume. Blind Separation of Wide-Band Sources in the Frequency Domain. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing,* vol. 3, pages 2080-2083, 1995.

[30] M. Knaak, S. Araki and S. Makino. Geometrically Constraint ICA for Convolutive Mixtures of Sound. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 2, pages 725-728, 2003.

[31] A. S. Bregman. *Auditory Scene Analysis: the perceptual organisation of sound*. MIT Press, Cambridge, MA, 1990.

[32] R. McAulay and T. Quatieri. Speech Analysis/Synthesis Based on a Sinusoidal Representation. *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 34, no. 4, pages 744-754, 1986.

[33] S. Roweis. One Microphone Source Separation. *Advances in Neural Information Processing Systems*, vol. 13, pages 793-799, 2000.

[34] O. Yilmaz and S. Rickard. Blind separation of speech mixtures via time-frequency masking. *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 52, no. 7, pages 1830-1847, 2004.

[35] H. Srinivasan and M. Kankanhalli. Harmonicity and Dynamics-based Features for Audio. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing,* vol.4, pages: 321-324, 2004.

[36] D.P. Morgan, E.B. George, L.T. Lee and S.M. Kay. Cochannel Speaker Separation by Harmonic Enhancement and Suppression. *IEEE Transactions on Speech and Audio Processing,* vol. 5, no. 5, pages 407-424, 1997.

[37] T.F. Quatieri and R.G. Danisewicz. An Approach to Co-channel Talker Interference Suppression Using a Sinusoidal Model for Speech. *IEEE Transactions on Signal Processing*, vol. 38, no. 1, pages 56-69, 1990.

[38] D.P.W. Ellis. Hierarchic Models of Hearing for Sound Separation and Reconstruction. In *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics,* pages 157-160, 1993.

[39] R. Maher. Evaluation of a Method for Separating Digitized Duet Signals. *Journal of the Audio Engineering Society,* vol. 38, no. 12, pages 956-979, 1990.

[40] F.J. Harris. On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform. In *Proceedings of the IEEE,* vol. 66, pages 51-83, 1978.

[41] H. Viste and G. Evangelista. Separation of Harmonic Instruments with Overlapping Partials in Multi-channel Mixtures. In *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 25-28, 2003.

[42] P. Jinachitra. Constrained EM Estimates for Harmonic Source Separation. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 6, pages 609-612, 2003.

[43] A.S. Master. Bayesian Two Source Modeling for Separation of N Sources from Stereo Signals. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 4, pages 281-284, 2004.

[44] S.W. Foo and E.W.T.L. Lee. Application of FRM Filters for Musical Notes Separation. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 1, pages 731-734, 2003.

[45] F.G. Stremler. *Introduction to Communication Systems (Third Edition)*. Addison-Wesley Publishing Company, Massachusetts, 1990.

[46] J.O. Smith III and X. Serra. PARSHL: an Analysis/Synthesis Program for Non-harmonic Sounds based on a Sinusoidal Representation. In *Proceedings of the International Computer Music Conference*, pages 290-297, 1987.

[47] X. Serra. *A System for Sound Analysis/Transformation/Synthesis Based on a Deterministic Plus Stochastic Decomposition*, PhD thesis, Stanford University, U.S.A., 1989.

[48] S. Levine and J.O. Smith III. A Sines+Transient+Noise Audio Representation for Data Compression and Time/Pitch Scale Modifications. In *Proceedings of the 105th Convention of the Audio Engineering Society, preprint 4781*, Sept. 1998.

[49] X. Serra. *Spectral Modeling Synthesis Homepage [Online]*, Available: http://www.iua.upf.es/~sms.

[50] P. Depalle and T. Hélie. Extraction of Spectral Peak Parameters using a Short-Time Fourier Transform Modeling and no Sidelobe Windows. In *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 4 pages, 1997.

[51] X. Serra. Musical Sound Modeling with Sinusoids plus Noise, In G. D. Poli, A. Picialli, S. T. Pope, and C. Roads, *Musical Signal Processing*. Swets & Zeitlinger Publishers, 1997.

[52] P. Depalle, G. Garcia and X. Rodet. Tracking of Partials for Additive Sound Synthesis Using Hidden Markov Models. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 1, pages 225-228, 1993.

[53] M. Lagrange, S. Marchaud and J.-B. Rault. Partial Tracking Based on Forward Path Exploration. In *Proceedings of the 116th Convention of the Audio Engineering Society*, May 2004.

[54] T. Tolonen, V. Välimäki and M. Karjalainen. *Evaluation of modern synthesis methods*, Technical Report 48, Laboratory of Acoustics and Audio Signal Processing, Dept. of Electrical and Communications Engineering, Helsinki University of Technology, March 1998. Available from http://www.acoustics.hut.fi/~ttolonen/sound_synth_report.html.

[55] L.B. Almeida and F.M. Silva. Variable-Frequency Synthesis: An Improved Harmonic Coding Scheme. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing,* vol. 9, pages 437-440, 1984.

[56] X. Rodet and P. Depalle. Spectral Envelopes and Inverse FFT Synthesis. In *Proceedings of the 93rd Convention of the Audio Engineering Society, preprint 3393*, Sept. 1992.

[57] K. Fitz and L. Haken. Sinusoidal modeling and manipulation using Lemur. *Computer Music Journal*, vol. 20, no. 4, pages 44-59, 1996.

[58] K. Fitz, L. Haken. *Lemur 4.0.1 On-line User's Guide [Online]*, Available from http://www.cerlsoundgroup.org/Lemur/LemurDocIndex.html.

[59] *OpenOffice.org XML File Format 1.0, Technical Reference Manual*, version 2, Sun Microsystems Inc., December 2002. Available from http://xml.openoffice.org/xml_specification.pdf

[60] M. Wright. *SDIF Specification [Online]*, 1999. Available from http://www.cnmat.berkeley.edu/SDIF/Spec.html.

[61] U. Zölzer and T. Boltze. Parametric Digital Filter Structures. In *Proceedings of the 99th Convention of the Audio Engineering Society, preprint 4099*, Sept. 1995.

[62] M. Kahrs. Audio Applications of the Teager Energy Operator. In *Proceedings of the 111th Convention of the Audio Engineering Society, preprint 5473*, Nov. 2001.

[63] B. Kostek, A. Czyzewski and S. Zielinski. Artificial Intelligence Approach to the Detection of Events in a Musical Signal. In *Proceedings of the 96th Convention of the Audio Engineering Society, preprint 3822*, Jan. 1994.

[64] B. Edler, H. Purnhagen and C. Ferekidis. ASAC – Analysis/Synthesis Audio Codec for Very Low Bit Rates. In *Proceedings of the 102nd Convention of the Audio Engineering Society, preprint 4376*, 1996.

[65] E.B. George and M.J.T. Smith. Analysis-by-Synthesis Overlap-Add Sinusoidal Modeling Applied to the Analysis and Synthesis of Musical Tones. *Journal of the Audio Engineering Society,* vol. 40, no. 6, pages 497-516, 1992.

[66] E.B. George and M.J.T. Smith. Speech Analysis/Synthesis and Modification using an Analysis-Synthesis/Overlap-Add Sinusoidal Model. *IEEE Transactions on Speech and Audio Processing,* vol. 5, no. 5, pages 389-406, 1997.