Adaptive Filter Bank Time-Frequency Representations

by

Peter C. Weber

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved November 2012 by the
Graduate Supervisory Committee:

Antonia Papandreou-Suppappola, Chair
Cihan Tepedelenlioglu
Narayan Kovvali

ARIZONA STATE UNIVERSITY

December 2012

ABSTRACT

A signal with time-varying frequency content can often be expressed more clearly using a time-frequency representation (TFR), which maps the signal into a two-dimensional function of time and frequency, similar to musical notation. The thesis reviews one of the most commonly used TFRs, the Wigner distribution (WD), and discusses its application in Fourier optics: it is shown that the WD is analogous to the spectral dispersion that results from a diffraction grating, and time and frequency are similarly analogous to a one dimensional spatial coordinate and wavenumber. The grating is compared with a simple polychromator, which is a bank of optical filters. Another well-known TFR is the short time Fourier transform (STFT). Its discrete version can be shown to be equivalent to a filter bank, an array of bandpass filters that enable localized processing of the analysis signals in different sub-bands.

This work proposes a signal-adaptive method of generating TFRs. In order to minimize distortion in analyzing a signal, the method modifies the filter bank to consist of non-overlapping rectangular bandpass filters generated using the Butterworth filter design process. The information contained in the resulting TFR can be used to reconstruct the signal, and perfect reconstruction techniques involving quadrature mirror filter banks are compared with a simple Fourier synthesis sum. The optimal filter parameters of the rectangular filters are selected adaptively by minimizing the mean-squared error (MSE) from a pseudo-reconstructed version of the analysis signal. The reconstruction MSE is proposed as an error metric for characterizing TFRs; a practical measure of the error requires normalization and cross correlation with the analysis signal. Simulations were performed to demonstrate the the effectiveness of the new adaptive TFR and its relation to swept-tuned spectrum analyzers.

ACKNOWLEDGEMENTS

I would like to thank my advisor, Dr. Antonia Papandreou-Suppappola, for her guidance and friendly support. I would also like to thank my wife, and my mom and dad, for all of their support. Finally, I would like to express my thanks to Professor Cihan Tepedelenlioglu and Dr. Narayan Kovvali for their participation on my thesis committee.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

1.1 Motivation

A signal with time-varying frequency content can often be more adequately represented using a time-frequency representation (TFR). This is because a TFR maps the signal into a two-dimensional function of time and frequency, and the value of the TFR indicates how frequency varies with time. Perhaps the best-known and simplest TFR is the short-time Fourier transform (STFT) which is generated by taking the Fourier transform of a windowed portion of the signal and sliding the window in time to obtain the frequency content as a function of time. Another simple and commonly used TFR that does not depend on the use of a window is the Wigner Distribution (WD) which is obtained by computing the Fourier transform of the (non-stationary) cross correlation of the signal and its complex conjugate [5].

There exist many types of TFRs, defined to satisfy different desirable signal properties and thus be useful in different applications. The STFT is a linear TFR that highly depends on the choice of its lowpass window [5]. It is useful in applications such as speech processing since it preserves time and frequency shifts on the analysis signal. The STFT can be implemented in a bandpass filter operation, where the impulse response of the bandpass filter is a modulated version of the lowpass window. The type and length of the window used in the STFT is important for obtaining localized representations in the time-frequency plane. In particular, due to the time-bandwidth uncertainty principle, the STFT suffers from an inherent tradeoff between achieving frequency resolution and time localization since it is not possible to have windows with simultaneously arbitrarily small duration and narrow bandwidth.

The discrete version of the STFT can be shown to be equivalent to an analysis filter bank [3], a system that divides the analysis signal into different sub-bands using an array of bandpass filters. The output of the filter bank from the different sub-bands results in a representation of the signal's time-dependent frequency variation. As a result, the choice of filtering operation in each band can affect the frequency resolution of the signal representation. Another consideration with filter banks is the synthesis problem, which involves reconstructing the original signal. It is important to attempt to minimize errors in the reconstruction that could have occurred, for example, from aliasing distortion in the analysis system [4].

An important property for a TFR to satisfy is the time and frequency marginals, or equivalently, to preserve the signal's energy information. The marginals provide the signal power spectrum and instantaneous energy by integrating the TFR along all times or frequencies, respectively. Although this is a good property for a TFR to possess, some TFRs that have been shown useful in specific applications do not satisfy the marginals. The STFT, and hence, the filter bank decomposition, do not preserve the energy content of the signal. It is therefore not the approach to be used when energy information is required. Still, many applications do not hinge on accurately preserving the energy, and we concentrate largely on these.

1.2 Proposed Work

In this work, we demonstrate that there are applications that benefit from a TFR that quickly distributes frequency content into distinct bands, in such a way that a simple Fourier synthesis sum is capable of perfectly reproducing the input signal. We propose such an adaptive TFR using an analysis filter bank whose parameters are designed by minimizing the mean-squared error (MSE) of the output from a simple synthesis filter bank. The reconstruction MSE is considered as a metric for characterizing the accuracy of a TFR. The proposed adaptive scheme is based on the cross correlation between the anal-

ysis signal and its filter bank reconstructed counterpart. The cross correlation is used to phase correct the synthesis bank output. The adaptive scheme is useful for signal compression and for fast TFR analysis when implemented in hardware, operating on analog signals.

Our adaptive TFR uses a Butterworth filter bank, which is a common approach to the issue of sub-band coding [1, 19]. We proceed from an assumption that a rectangular passband is a desirable property for a filter bank, due to the minimal distortion induced by the flat passband [2–4]. Many audio applications exist that make use of Butterworth filters, and the approach is not unique to this work. The MPEG layer 3 audio compression standard uses sub-band decomposition by means of flat passbands [24], but it differs from our approach in the reconstruction method, and it does not focus on generating a TFR with high time-frequency resolution. Whereas we describe our signal reconstruction as a pseudo-reconstruction, audio coding generally prizes reconstruction accuracy over TFR resolution, and therefore uses a different reconstruction method than is used herein. Discussion of rectangular passband filter bank applications in common use can be found throughout this thesis; they are a well-known approach to spectral decomposition of signals. Our contribution lies in our proposed TFR optimization method, which is predicated on the value of filter banks for signal analysis.

1.3 Thesis Organization

This thesis is organized as follows. Chapter 2 begins by describing TFRs such as the Wigner distribution and an optical analog. We discuss some of the nuances associated with various TFRs, and we briefly describe the ambiguity function. We also discuss the value of analytic signals for certain TFRs.

In Chapter 3, we provide a review of the STFT and demonstrate its use in representing a linear chirp signal. We show that the STFT can be expressed as a filter bank,

and that this approach helps to determine the window function to use for a given signal. The filter bank is generated by a prototype bandpass filter that is heterodyned repeatedly to cover the entire frequency space from DC (zero frequency) to the Nyquist frequency. The infinite impulse response (IIR) filtering operation is described in terms of feedforward and feedback coefficients. The Butterworth filter design process is discussed, and we use the process to design a 4th order lowpass filter. We then perform a time-frequency analysis of a linear chirp using a filter bank, and we compare it to the corresponding STFT analysis.

Chapter 4 discusses the signal reconstruction stage that can follow a filter bank. We discuss limitations to the accuracy of a reconstruction. We introduce the quadrature mirror filter bank (QMF), and we show that it is capable of near-perfect reconstruction. A simulation with a 2-band QMF is performed, showing that although the reconstruction is perfect after correcting for a delay and gain factor, the corresponding TFR is not very useful. An analysis using a QMF tree structure is performed, and the resulting TFR is shown to be distorted due to the passband overlap. A rectangular passband filter bank is then used to analyze the signal, showing that although signal reconstruction is no longer accurate, the TFR provides a more informative representation in the time-frequency plane.

In Chapter 5, we propose new measures to characterize the accuracy of a TFR, and we develop a process for generating a figure of merit[1] that can be used to obtain our adaptive TFR. The process involves normalization, followed by a cross correlation between the analysis signal and the reconstructed signal, and we show that this is necessary for determining the accuracy of any reconstruction, even the near-perfect output from a QMF bank. The cross correlation is used to phase correct the synthesis bank output. The mean-squared error (MSE) of the normalized and phase-corrected reconstruction is computed. The proposed method is applied to a musical signal, showing that the TFR accurately describes the composition by comparing it with the musical notation. The filter

[1] A figure of merit is a compact numerical term that describes the quality, efficiency, or performance of a system.

bank's equivalence to the STFT is shown by analyzing the same data using a generic MATLAB STFT function.

The speed advantage of the STFT is due to the efficiency of the fast Fourier transform (FFT), which we describe in Chapter 6. The filter bank can be implemented with hardware so that it operates on analog signals, resulting in improved performance in execution speed. We discuss the computational efficiencies that result from decimation prior to filtering in the digital domain, and the phase effects associated with the order in which decimation and filtering are performed. We propose new opportunities for signal compression that can be identified by inspecting the filter bank TFR, and downsampling by a factor that is unrelated to the frequency range encoded by the signal, as long as group delay effects are preserved. A signal compression scheme motivated by the TFR is proposed.

Chapter 2

Time-Dependent Spectral Decomposition

The phenomena of visual and auditory perception involve spectral decomposition as a fundamental aspect of the interpretation of signals. Music is interpreted as a time-evolving series of pressure wave frequencies, and images are seen as spatially distributed electromagnetic wave frequencies. These are analogous examples of how signals with time-dependent spectral components can be represented by time-frequency representations. Another example is time-evolving spectroscopic analysis. A spectrometer decomposes the input light into its constituent frequencies, either in a wavelength dispersive or energy dispersive manner. We will briefly discuss the wavelength dispersive techniques in this chapter. The diffraction grating disperses the input light, which has a time-dependent frequency content, into multiple spectral channels in different angles [6]. This is similarly analogous to the filter bank decomposition approach, as discussed in the following chapters.

2.1 Diffraction Interference and Signal Autocorrelation

Diffraction is an interference effect, in which a wave is made to interfere with a shifted version of itself [6]. The shifted version of the incident light can be generated by a grid of rulings etched onto a reflective surface, known as a grating. In addition to reflection gratings, there are also transmission, blazed, and other types of diffraction grating. The resolving power of a grating in optics is analogous to frequency resolution in signal processing. If a grating is to be designed for greater spectral resolution, it is given a finer ruling. The limit to the resolution of the grating is related to the wavelength range of interest. In particular, even if higher diffraction orders can be used, and construction of the grating is not an issue, eventually the grid dimensions approach the wavelength of incident light. This fundamental limit is related to the uncertainty principle [7].

The interference that occurs in diffraction is related to the signal autocorrelation involved in the definition of the Wigner distribution (WD) time-frequency representation, which we describe below. Wave interference is the consequence of different wave amplitudes adding constructively and destructively, while autocorrelation involves the multiplication of signals separated by known time lags. In essence, wave interference is a consequence of the phase coherence properties of the interacting waves, and this coherence is a measure of their degree of correlation. Similarly, the WD signal autocorrelation provides a measure of similarity between time-shifted versions of the signal. The two phenomena yield similar results, in the sense that they both produce peaks representing the relationship between wave periodicities and the separation among interacting signal components.

2.2 Wigner Distribution

The WD of an analysis signal $x(t)$ describes the signal's time-evolving frequency content as a function of time. Since the WD is a quadratic time-frequency representation (TFR), it exhibits interference when it is used to analyze multi-component signals [5, 33]. One can better understand the WD by considering diffraction in optics. In geometrical optics, the WD as a function of the position and momentum of an optical ray is sometimes used to characterize an optical system. In the case of a diffraction grating, the WD of the grating describes the grating's effect on an optical input signal, and it provides a compact and efficient method of characterization. The distribution represents wave amplitude as a function of spatial frequency and position. Constant values represent geometrical rays, as described in [8, 9]. Since the WD of the grating describes the angular distribution of input light as a function of spatial frequency, the grating effectively performs a WD on the input optical signal. The process is shown schematically in Figure 2.1; it is closely related to the effect that produces colorful interference patterns in an oily puddle.

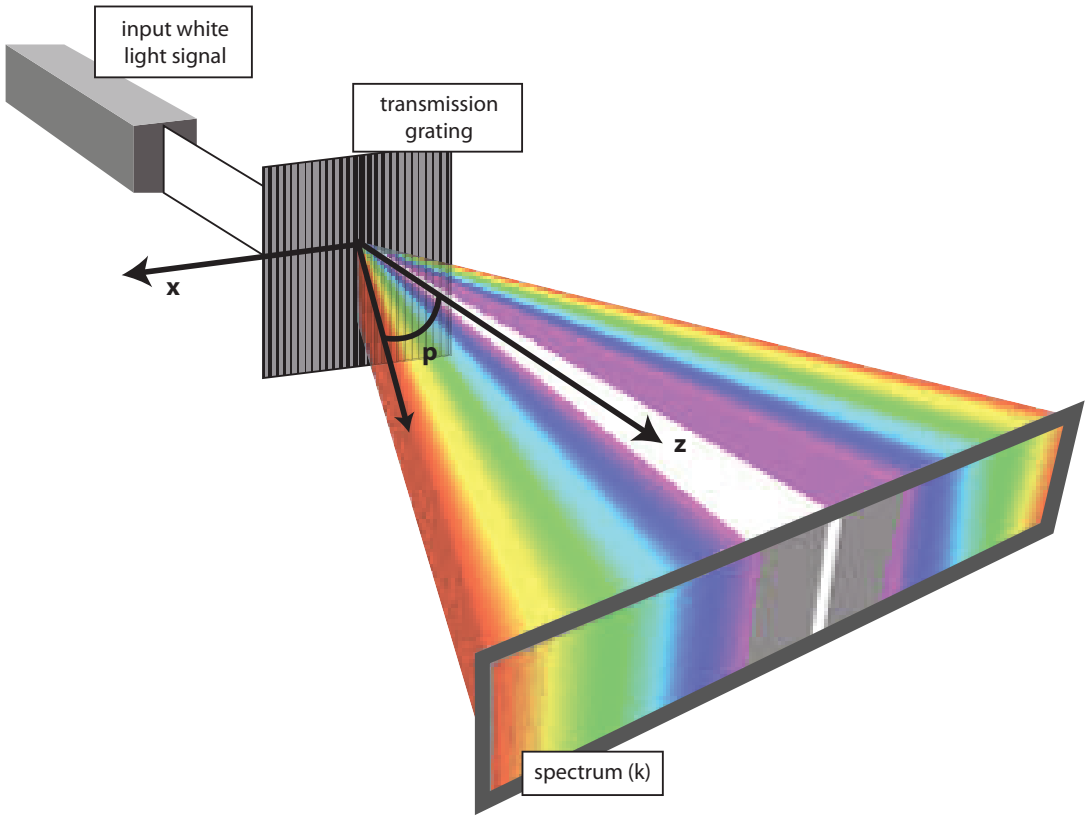

Figure 2.1: Dispersion of an input white light signal into its spectrum by a transmission grating. Colors are given by the wavenumber k, the transmission coefficient $T(x)$ is a function of the x-coordinate of the grating, and the direction of propagation is given by the angle p. Figure adapted from [40].

The WD is the Fourier transform of a function of the original signal and a time lag τ, resulting in an output function of two variables. The WD of a signal $x(t)$, defined as a TFR and thus as a function of time t and frequency ω, is given by:

$$W_x(t,\omega) = \frac{1}{2\pi}\int_{-\infty}^{\infty} x^*(t-\frac{\tau}{2})\, x(t+\frac{\tau}{2})\, e^{-j\omega\tau}\,\mathrm{d}\tau \tag{2.1}$$

When the WD is used to characterize a transmission grating [10], it involves the amplitude transmission coefficient $T(x)$, which describes the effect the grating has on transmitted light as a function of position:

$$W_T(x,p) = \frac{k}{2\pi}\int_{-\infty}^{\infty} T(x-\frac{x'}{2})T^*(x+\frac{x'}{2})\, e^{-jkpx'}\,\mathrm{d}x' \tag{2.2}$$

where k is the wavenumber[1], p is the angle of propagation, and x is the spatial coordinate over which the grating's transmission coefficient varies [10]. Inclusion of a variable p representing the angular direction allows the WD to be extended to arbitrary angles. Comparing Equations (2.1) and (2.2), the time-lag parameter τ is analogous to the position parameter x'; whereas the WD TFR only considers angle 2π, with $\omega\tau = 2\pi f\tau$, the optics WD allows for any angle p. A similar generalization can be applied to the Fourier transform, resulting in the Fractional Fourier transform [11]. When the WD of the light field is computed, projections onto a plane at any angle are always real and non-negative, and in the far-field approximation, in which evanescent waves are negligible[2], they provide accurate descriptions of the intensity of light incident onto a surface orthogonal to the angle of projection [11].

Although the WD produces cross terms in many instances, the marginal energy content is preserved, as shown in Figure 2.2, in which the WD of a signal consisting of four sinusoids is displayed along with the line integrated frequency content, which is the power spectral density. Notably, the power spectrum does not include any cross term contamination. The frequency marginal of the WD accurately depicts the power and frequency of each sinusoid.

2.3 Ambiguity Function

Taking the Fourier transform of the complex autocorrelation function with respect to the time t produces a two-dimensional (2-D) representation known as the ambiguity function (AF), which is a function of the time-lag τ and frequency lag θ, and is defined as:

$$A_x(\theta,\tau) = \frac{1}{2\pi}\int_{-\infty}^{\infty} x^*(t-\frac{\tau}{2})\, x(t+\frac{\tau}{2})\, e^{-j\theta t}\,\mathrm{d}t \tag{2.3}$$

[1]The wavenumber k is standard notation in physics, and is not to be confused with the discrete frequency unit k, used throughout the remainder of this work.

[2]A distance of ten wavelengths or more [11].

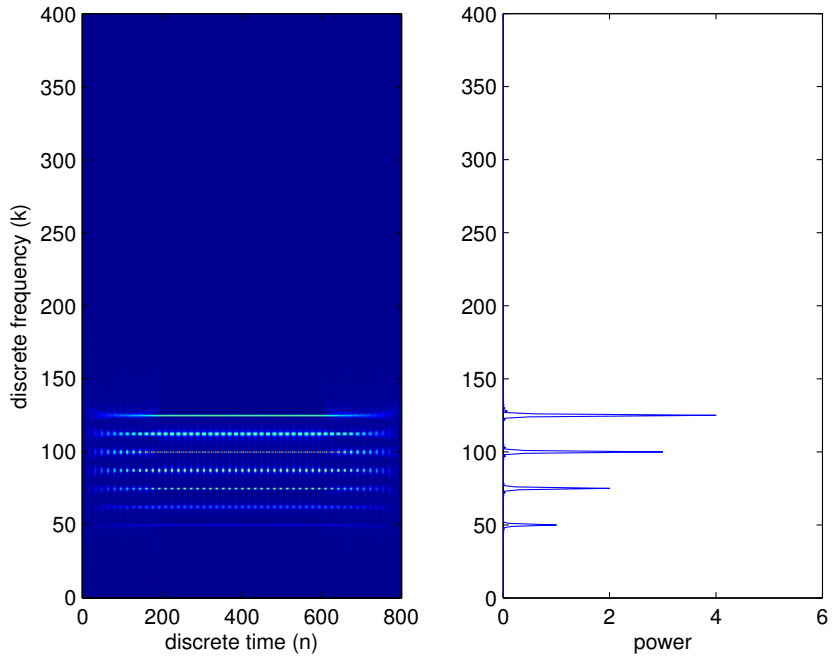

Figure 2.2: Left: Wigner distribution of the signal $x(t) = \sum_{i=1}^{4} A_i \cos 2\pi f_i t$, $A_i = \sqrt{i}$, $f_i = 25\,(i+1)$, corresponding to the sum of 4 sinusoids. Right: The power spectral density of the same signal is obtained by integrating along the time index.

The AF and the WD are 2-D Fourier transform pairs as shown by:

$$A_x(\theta,\tau) = \int\int W_x(t,\omega)\, e^{j(-\theta t+\tau\omega)}\, \mathrm{d}t\, \mathrm{d}\omega \tag{2.4}$$

Whereas the WD provides a representation in the time-frequency plane, the AF is a correlative representation since the AF can be interpreted as a joint time-frequency correlation function [33]. Since the WD and AF are related via Fourier transformations, the AF can be used to reduce the cross terms that result in the WD when multicomponent signals are analyzed. The cross terms appear between any two time-frequency points in the WD, while their 2-D Fourier transform places them at symmetric locations away from the origin. The WD of a signal with two linear chirps, shown in Figure 2.3, and the AF of the same signal, shown in Figure 2.4, demonstrate the effect. The plots are generated by the code "Discrete_Wigner.m" listed in Appendix A. Since the auto terms are at the AF origin, a 2-D lowpass filter can be designed in the AF plane to remove the cross terms. The

relationship between the WD, AF, and the signal product $R_x(t,\tau) = x(t+\tau/2)x^*(t-\tau/2)$ is shown graphically in Figure 2.5, where the operators $\mathscr{F}$ and $\mathscr{F}^{-1}$ denote Fourier and inverse Fourier transformations.

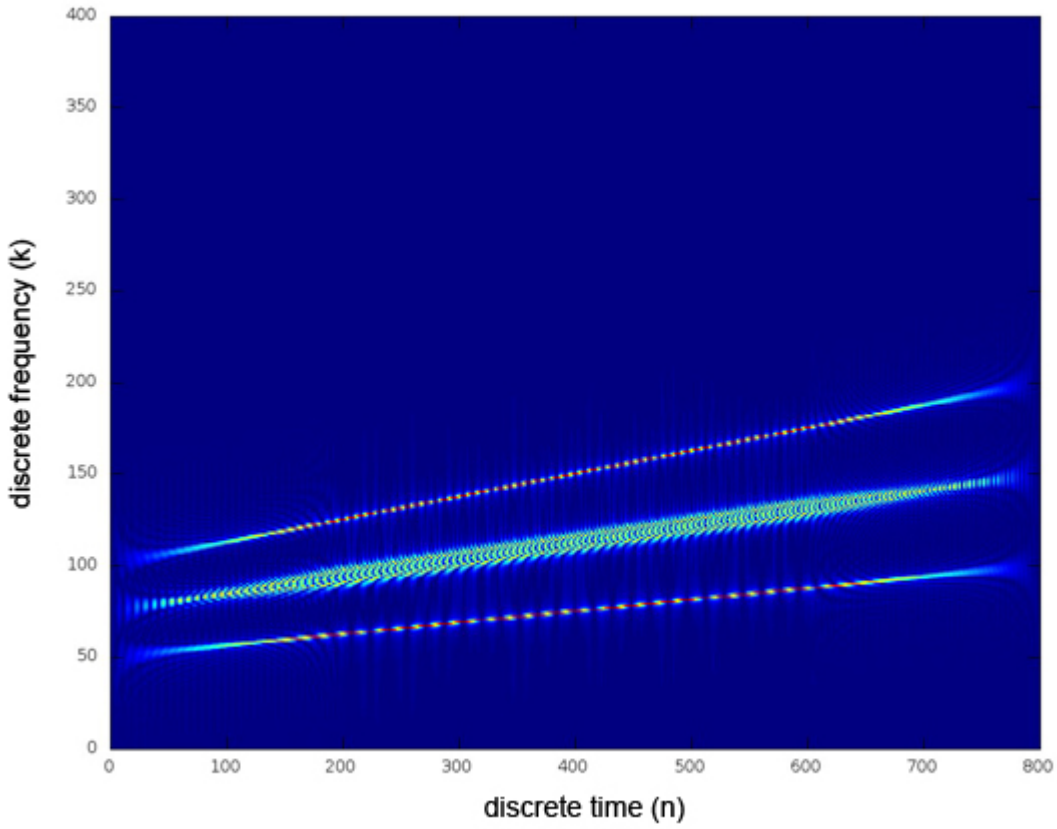

Figure 2.3: Wigner distribution of two linear chirps. Note the cross terms that occur halfway between any two TF points.

In addition to the WD, many other TFRs have been proposed in the literature [5]. Depending on the application, a TFR can be chosen to preserve many signal properties while simultaneously having high localization in the time-frequency plane. The WD is among the best TFRs at preserving information, but it is often difficult to interpret, because it produces cross terms at frequencies and times not present in the original signal. TFRs satisfying some of the properties that the WD satisfies while not suffering as severely from the presence of cross terms are the subject of continued research.

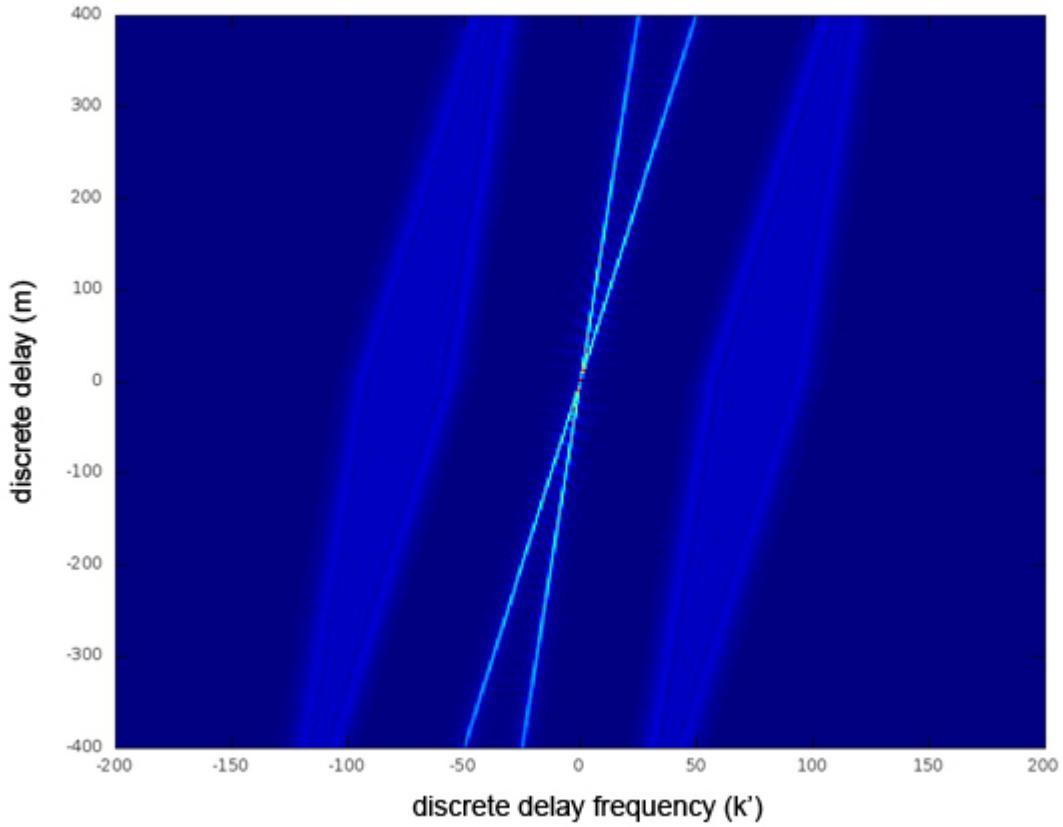

Figure 2.4: Ambiguity function of two linear chirps. Note the cross terms are located away from the origin.

2.4 Analytic Signals

In order to reduce cross terms in the WD, analytic signals are often used in analysis as they do not have negative frequency components. The analytic signal $x_a(t)$ adds to the original signal $x(t)$ an imaginary component which is the Hilbert transform $H[x(t)]$ of the signal [5]. Specifically, it is given by:

$$x_a(t) = x(t) + jH[x(t)] = x(t) + \frac{j}{\pi}\int \frac{x(t')}{t-t'}\,\mathrm{d}t' \tag{2.5}$$

This operation is equivalent to eliminating the negative frequencies and doubling the amplitude of the positive frequencies. As a result, taking the WD of the analytic signal results in removing the negative frequencies from the Wigner distribution. In the next chapter, we describe another common TFR, called the short-time Fourier transform

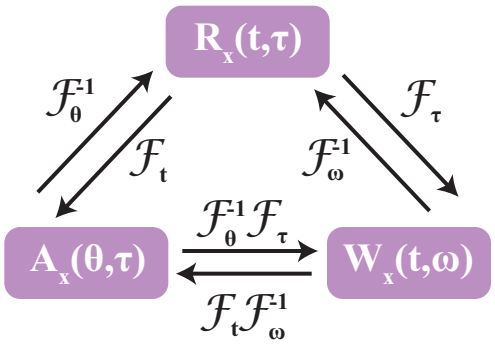

Figure 2.5: Relationship between Wigner Distribution, Ambiguity Function, and signal product.

(STFT), whose interpretation as a TFR is easier to understand. The relationship between the STFT and an optical analog called a "polychromator", similar to the relationship between the WD and diffraction, is discussed.

Chapter 3

Short time Fourier Transform and Filter Bank Decomposition

3.1 Short-Time Fourier Transform

The short-time Fourier transform (STFT) is a linear time-frequency representation (TFR) obtained by taking the Fourier transform of windowed signal segments, using a lowpass window $h(t)$. Specifically, it is given by:

$$S_x(t,\omega;h) = \int_{-\infty}^{\infty} h(\tau - t)\, x(\tau)\, e^{-j\omega\tau}\, \mathrm{d}\tau \tag{3.1}$$

The STFT can be interpreted as a TFR by assuming that the analysis time-varying signal does not have frequencies that vary with time over the duration of the window. Thus, decreasing the duration of the windowed signal segment can yield higher resoleuton in time, though at the tradeoff of lower frequency resolution. The squared magnitude of the normalized STFT is called the spectrogram, and is defined as:

$$S_x(t,\omega) = \left| \frac{1}{\sqrt{2\pi}} \int e^{-j\omega\tau}\, x(\tau)\, h(\tau - t)\, \mathrm{d}\tau \right|^2 \tag{3.2}$$

The spectrogram is a quadratic TFR, like the WD; however, unlike the WD, it does not suffer from any cross terms. The discrete version of the STFT is obtained by sampling both time, $t = nT_s$ and frequency $\omega = 2\pi k/N$, where T_s is the sampling period, N is the number of samples in time and frequency, and the range of discrete time n and discrete frequency k is $n,k = 0,1,\ldots,N-1$. Thus, the discrete STFT is given by:

$$S_x[n,k;h] = \sum_{m=0}^{N-1} h[m-n]\, x[m]\, e^{-j2\pi\frac{k}{N}m} \tag{3.3}$$

The choice of window function is important. A rectangular window results in "switching noise" injected into the resulting time-frequency representation due to the sharp cutoff at the window edges. The abrupt amplitude decrease over a short time at the window edges appears as a high frequency signal component that is spuriously added to the STFT. A tapered window is preferable for this reason, as will be discussed later in the thesis.

3.1.1 Kernel Method

Both the WD of Chapter 2 and the spectrogram can be represented as members of a general class of quadratic TFRs, that preserve time shifts and frequency shifts on the analysis signal [5]. This class is often called Cohen's class of quadratic TFRs, $C_x(t,\omega)$, and each TFR in this class is characterized by a unique kernel function $\phi(\theta,\tau)$. In particular, any TFR in Cohen's class can be formulated as:

$$\begin{aligned} C_x(t,\omega) &= \frac{1}{4\pi^2}\int\int\int x^*(u-\frac{\tau}{2})x(u+\frac{\tau}{2})\ \phi(\theta,\tau)\ e^{j(\theta t-\tau\omega-\theta u)}\ \mathrm{d}u\ \mathrm{d}\tau\ \mathrm{d}\theta \\ &= \frac{1}{2\pi}\int\int A_x(\theta,\tau)\ \phi(\theta,\tau)e^{j(-\tau\omega+\theta t)}\ \mathrm{d}\tau\ \mathrm{d}\theta \end{aligned} \tag{3.4}$$

With this formulation, the WD kernel is in its simplest form, $\phi_{WD}(\theta,\tau)=1$. By selecting a different function, the kernel can be considered as a 2-D lowpass filter in the AF domain that can be used to suppress the cross terms that often appear in the WD. This topic is explored in detail in [5]. The kernel of the spectrogram depends on the window function $h(t)$ and is $\phi(\theta,\tau)=\int h^*(u-\frac{\tau}{2})h(u+\frac{\tau}{2})\ e^{-j\theta u}\ \mathrm{d}u$. The relationship between the WD and the spectrogram can be found in [13].

3.1.2 STFT Simulations

We can observe the effect of the rectangular window on the STFT by performing simulations. A good place to start is with a simple stationary sinusoidal signal. We will follow this with simulations involving a time-varying (TV) signal, that is a signal whose frequency content changes with time, to demonstrate the usefulness of the STFT as a TFR.

3.1.2.1 Stationary Signal

A stationary signal is one whose frequency does not vary with time. Consider the discrete-time signal $x[n]=\sin(2\pi\frac{k_0}{N}n)$, where $n=0,1,\ldots,N-1$, the data record length is $N=1000$ samples, and k_0/N is the normalized frequency of the sinusoid. The maximum

frequency that can be identified without aliasing is given by the Nyquist criterion, and is $k_{max} = N/2 = 500$, or in normalized frequency units $\hat{f}_{max} = \frac{1}{2}$. The signal with frequency $k_0 = 50$ is shown along with its discrete Fourier transform (DFT) in Figure 3.1. The frequency of the signal does not change with time, and we expect that the STFT will be a continuous straight line of constant frequency.

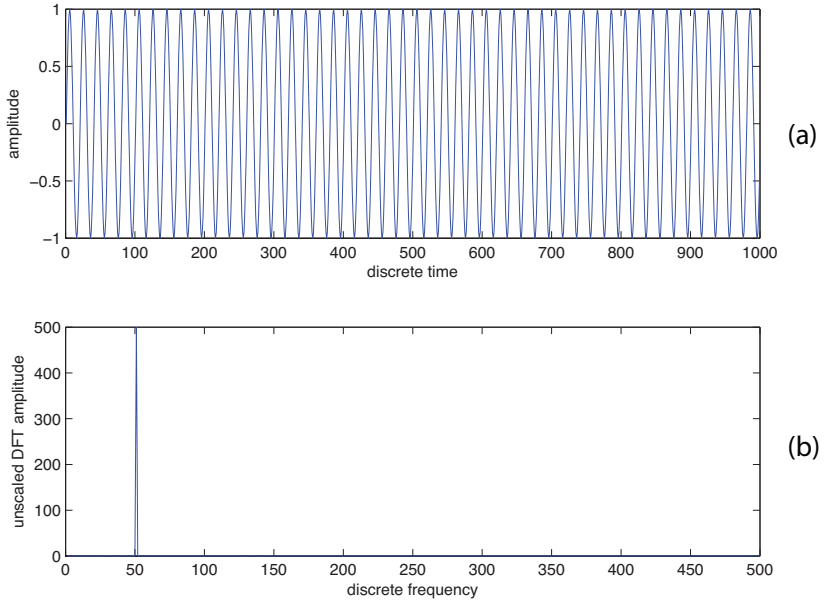

Figure 3.1: Simple sinusoidal signal with discrete frequency $k = 50$. (a) Signal in the time-domain; (b) DFT of the signal.

3.1.2.2 TV Signal

We now consider a linear TV signal, whose discrete frequency increases from $k_0 = 50$ to $k_1 = 400$. We note at this point that the window is typically symmetric about time zero, so that for a signal that originates at time index $n = 0$, the window must be shorter than its maximum length as it "rolls onto" the signal[1]. The STFT performed here, using the MATLAB code "STFT.m" in Appendix A, involves a window that wraps around the

[1] Similarly, it will "roll off" at the end.

signal, resulting in an artifact at the end. We will later use symmetric windows and a non-wrapping STFT (by zero-padding the signal) to avoid these artifacts. The signal and its STFT are shown in Figure 3.2.

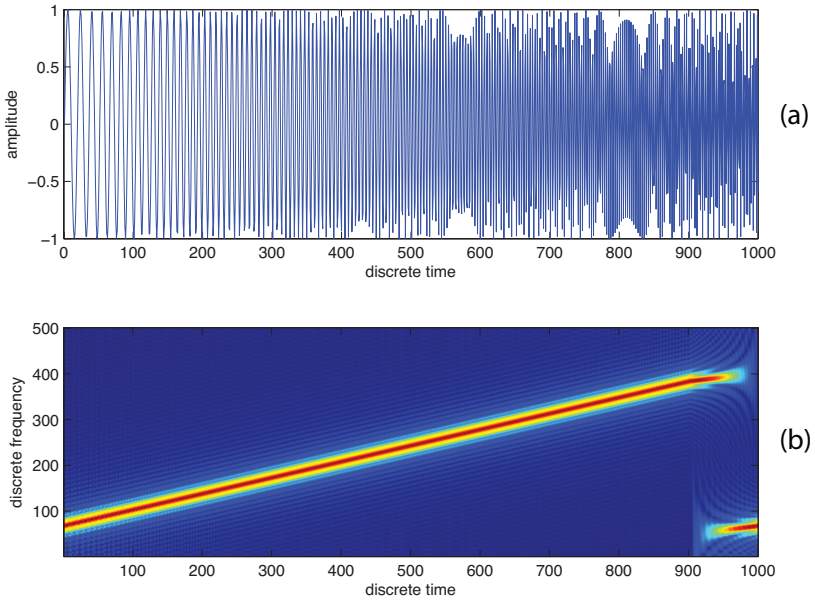

Figure 3.2: STFT of linear TV sinusoid with discrete frequency linearly increasing from $k_0 = 50$ to $k_1 = 400$. (a) Time-domain signal; (b) STFT TFR.

3.2 Bandpass Representation of the STFT

In Equation (3.1), we interpret the STFT as a Fourier transform (FT) of windowed signal segments. We can also interpret the STFT as a lowpass filtering operation by fixing the output frequency. Specifically, $S_x(t,\omega,h) = \int h(t-\tau[x(\tau)e^{j\omega_0\tau}]d\tau = (x(t)e^{j\omega_0 t}) * h(t)$, where $*$ denotes the convolution operator. An equivalent frequency domain expression of the STFT shows that it can also be interpreted as the inverse FT of the signal spectrum, windowed to the analysis frequency or as a bandpass filtering operation of the analysis signal before frequency shifting it back to low frequency [14, 15]. When discretizing the STFT, the filtering operation becomes a combination of multiple bandpass filters, constituting a filter bank. In order to demonstrate the bandpass nature of the filter bank in detail,

we first consider the discrete formulation of the STFT in Equation (3.3) using a real and even window $h[n] = h^*[-n]$:

$$\begin{aligned} S_x[n,k;h] &\triangleq \sum_{m=0}^{N-1} x[m]\ h[n-m]\ e^{-j2\pi\frac{k}{N}m} \\ &= \sum_{m=0}^{N-1} x[m]\ h[n-m]\ e^{-j2\pi\frac{k}{N}m}\ e^{-j2\pi\frac{k}{N}n}\ e^{j2\pi\frac{k}{N}n} \\ &= \sum_{m=0}^{N-1} x[m]\ h[n-m]\ e^{-j2\pi\frac{k}{N}(m-n)}\ e^{-j2\pi\frac{k}{N}n} \\ &= e^{-j2\pi\frac{k}{N}n} \sum_{m=0}^{N-1} x[m]\ h[n-m]\ e^{j2\pi\frac{k}{N}(n-m)} \end{aligned} \tag{3.5}$$

If we define a new impulse response sequence $h_{BP}[n;k] = h[n]e^{j2\pi\frac{k}{N}n}$, the STFT can be expressed as:

$$\begin{aligned} S_x[n,k;h] &= e^{-j2\pi\frac{k}{N}n} \sum_{m=0}^{N-1} x[m]\ \underbrace{h[n-m]\ e^{j2\pi\frac{k}{N}(n-m)}}_{h_{BP}[n-m;k]} \\ &= e^{-j2\pi\frac{k}{N}n} \sum_{m=0}^{N-1} x[m] h_{BP}[n-m;k] \\ &= e^{-j2\pi\frac{k}{N}n} \left(x[n] * h_{BP}[n;k]\right) \end{aligned} \tag{3.6}$$

We see that the window that is convolved with the signal is the impulse response of a heterodyned filter, which we call h_{BP}, since it is a bandpass filter. The convolution is repeated for each value of k, which is the discrete frequency index. Following each convolution, the result is down converted by multiplication with a complex exponential. A simple schematic of the filtering process described in Equation (3.6) below is shown in Figure 3.3, with the downconverted outputs from each filter stored as the sequences $y_i[n]$. This corresponds to separating the original signal into several frequency bands so that it can be transmitted over a bank of filters, each having its own frequency response, with the outputs from each filter stored separately.

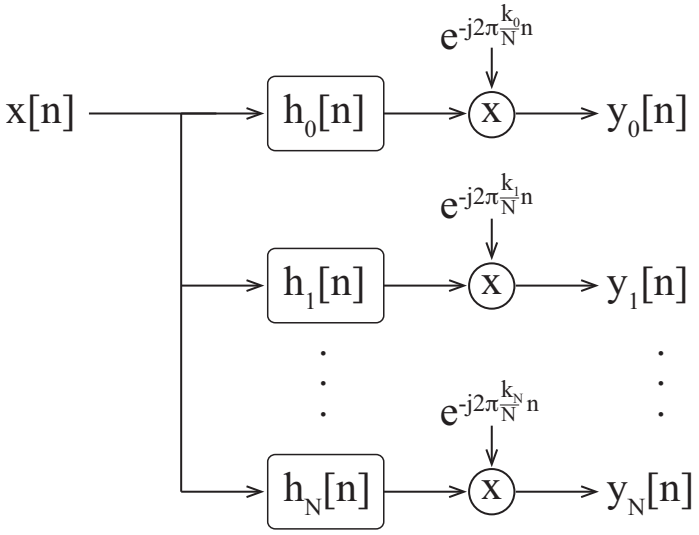

Figure 3.3: Simple filter bank.

3.2.1 Bandpass Filters

When discretizing frequency to $f/f_s = \ell/N$, where f_s is the sampling frequency and N is the number of discrete frequency points, then the discrete Fourier transform (DFT) of the lowpass STFT window $h[n]$, $n = 0,\ldots,N-1$ is given by

$$H(\ell) = \sum_{n=0}^{N-1} h[n]e^{-j2\pi\frac{\ell}{N}n}, \quad \ell = 0,\ldots,N-1. \tag{3.7}$$

An ideal bandpass filter is one whose magnitude spectrum is zero everywhere except at a specified band $f_{min} < f < f_{max}$. Considering the bandpass STFT interpretation of Equation (3.6), the bandpass filter $h_{BP}[n;k]$ is obtained by multiplying the lowpass filter $h[n]$ by a complex exponential that has the effect of up converting, or heterodyning, the sequence of impulse coefficients. As a result, the DFT of the bandpass filter is given by:

$$\begin{aligned} H_{BP}[\ell;k] &= \sum_{n=0}^{N-1} \left[h[n]e^{j2\pi\frac{k}{N}n}\right] e^{-j2\pi\frac{\ell}{N}n} \\ &= \sum_{n=0}^{N-1} \left[h[n]e^{-j2\pi\frac{\ell-k}{N}n}\right] = H[\ell-k] \end{aligned} \tag{3.8}$$

Comparing Equations (3.8) and (3.6), it can be seen that the DFT $H_{BP}[\ell;k]$ of the bandpass filter is a shifted version of the DFT $H[\ell]$ of the lowpass filter. If we fix

$k = k_0$ in the discrete frequency STFT $S_x[n,k;h]$ in Equation (3.2), then $S_x[n,k_0;h] = e^{-j2\pi(k_0/N)n}(h_{BP}[n;k_0] * x[n])$ involves a lowpass filter, shifted to discrete frequency k_0/N or, equivalently, a bandpass filter at k_0/N. As a result, the STFT can be considered as a set of N bandpass filters, each of which is separated from the next in frequency by $1/N$.

We can isolate frequency content into distinct bands if we use non-overlapping passbands. This requires that the 1/N separation between filters is replaced by a value determined by the filter bandwidths. We anticipate that a rectangular passband will minimize distortion in the frequency domain because it will allow the least stopband leakage (due to steep rolloff), and passband distortion (due to flat passband). We can begin the filter design process by assuming a rectangular time window, observing the resulting frequency response, and using the duality property of the Fourier transform to obtain the desired rectangular passband in th frequency domain.

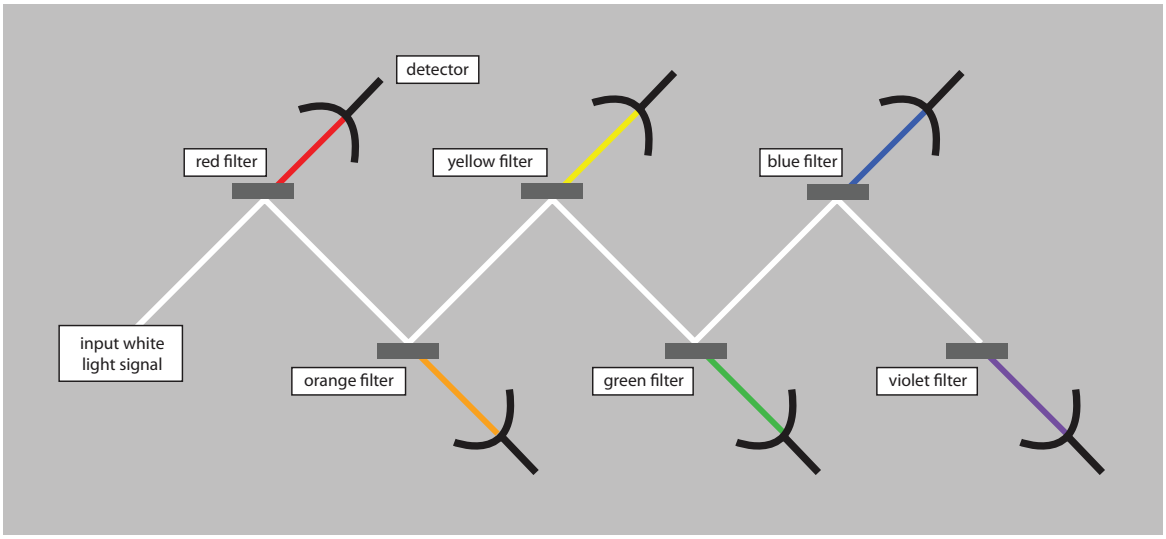

Figure 3.4: Schematic of a polychromator, which is the optical analog of the filter bank described in Section 3.2.

As we have shown, the STFT defined above can be equivalently described in the frequency domain as a series of bandpass filters. In effect, it is the same as the polychromator shown in Figure 3.4, which is the optical analog of the STFT filter bank. With this

change in perspective, some light is shed on the issue of window function choice, and we address this issue in section 3.3.

3.2.2 Window Heterodyning

In order to show the effect of the heterodyning process, we begin with the short duration rectangular window $h[n] = 1$, $n = 0, 1, \ldots, N-1$, for $N = 5$. The window is plotted in Figure 3.5, and its frequency response is computed with the code "DFT.m" in Appendix A. The resulting plot of the real part of the DFT $H[\ell]$ of the window, as a function of discrete frequency, is shown in Figure 3.6.

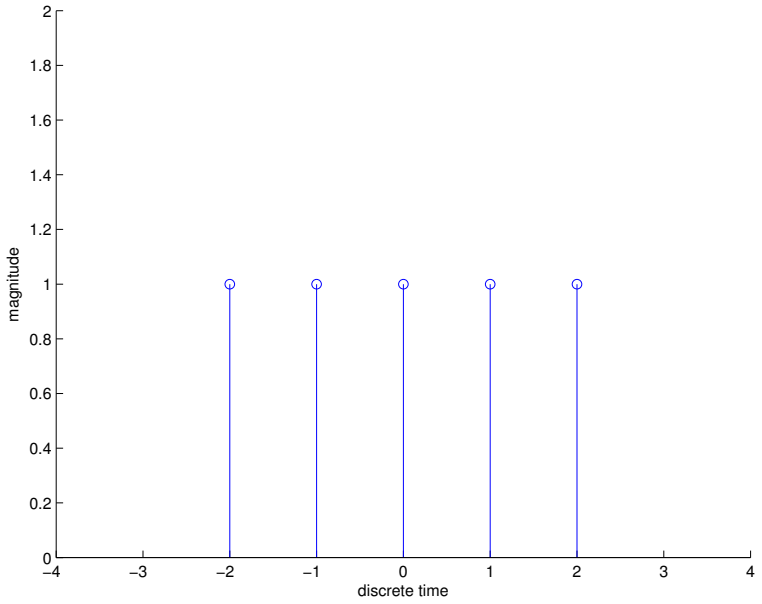

Figure 3.5: Symmetric rectangular window of duration $N = 5$ samples.

The window function involves five samples of the signal, so that for each time step m, the discrete STFT multiplies five of the samples by 1, and the rest by 0, prior to computing the FT of the modified sequence. The result is the FT of a short windowed segment of the signal as a function of the window's *position* along the signal sequence. The simple rectangular window in time is a sinc function in the frequency domain.

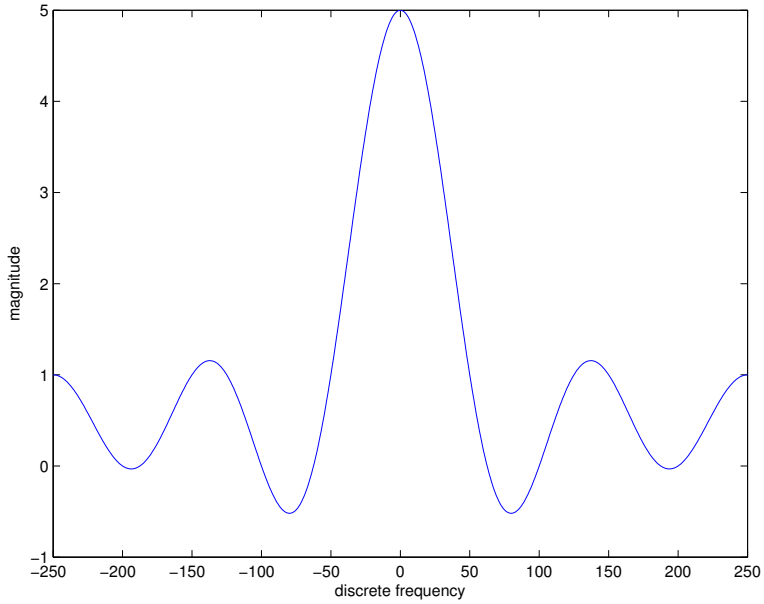

Figure 3.6: Real part of frequency response of symmetric rectangular window.

After multiplication with the complex exponential, the new filter $H_{BP}[\ell;k]$ is formed. The code "heterodyning.m" listed in Appendix A carries out the process. The resulting plot with the prototype filter plus four of the upconverted bandpass filters is shown in Figure 3.7. The entire frequency space is spanned by 5 of the filters, and the pass band of the prototype filter is quite wide, which is not surprising given how few samples are included in the time window.

3.3 Filter Design

The rectangular time window used by the STFT in Section 3.1.2 resulted in a filter with a sinc function for its frequency magnitude response. As a result of this filter shape, the STFT is expected to introduce spurious content when the analysis signal has multiple frequencies present at the same time. We note that, by the duality property of the FT, since the FT of a rectangular time window is a sinc function, a rectangular passband will have a sinc function for its time window [14, 15].

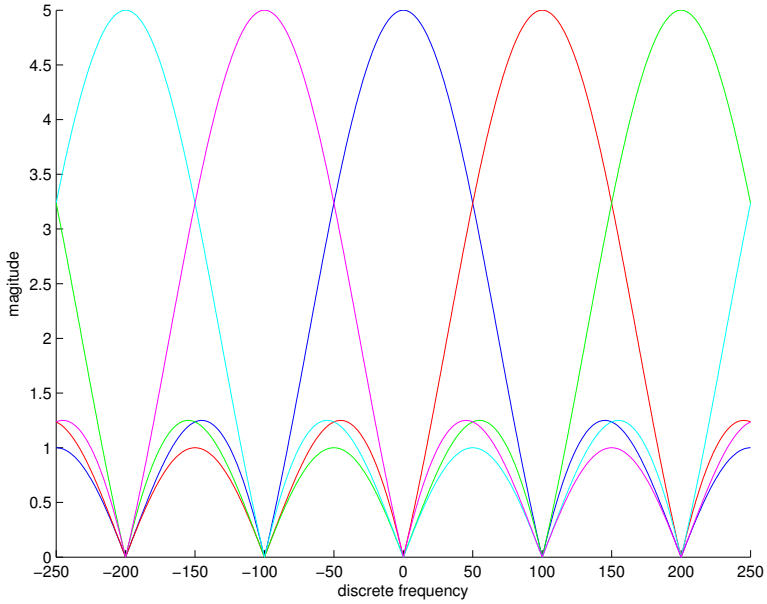

Figure 3.7: Five bandpass filters (magnitude plot).

3.3.1 Filter Representations

In order to realize the advantages of using a filter bank to produce the STFT, it is necessary to carry out the operation represented in compact form in (3.6), reproduced below:

$$S_x[n,k;h] = e^{-j2\pi\frac{k}{N}n}\left(x[n] * h_{BP}[n;k]\right) \tag{3.9}$$

The process involves a series of convolutions between the signal and a bandpass filter's impulse response. Each of these convolutions involves a different filter, differing only from the prototype filter by a frequency upconversion. Each of the resulting convolutions is then downconverted in a following step. The convolution is equivalent to a filtering operation. Under certain initial conditions, the filter can be represented by a difference equation, in

which the signal is multiplied by a series of feedback and feedforward coefficients:[2]

$$y[n] = h[n] * x[n] = \sum_{p=1}^{P} y[n-p]a_p + \sum_{q=0}^{Q} x[n-q]b_q \tag{3.10}$$

In order to make use of Equation (3.10), we need to obtain the coefficients using z-transform properties. Specifically, convolution in the time domain is equivalent to multiplication in the z domain:

$$y[n] = h[n] * x[n] \leftrightarrow H(z)X(z) = Y(z) \tag{3.11}$$

where $Y(z) = \sum_{r=0}^{R} y[r]z^{-r}$ is the z-transform of $y[n]$. We can express the z-transform $H(z)$ of the impulse response in terms of $X(z)$ and $Y(z)$ by noting that $H(z) = \frac{Y(z)}{X(z)}$. Given the definition of the z-transform, and its time-shift property, which states that $x[n-k] \leftrightarrow z^{-k}X(z)$, the difference equation in (3.10) can be equivalently written as:

$$Y(z) = \sum_{p=1}^{P} z^{-p}a_p Y(z) + \sum_{q=0}^{Q} z^{-q}b_q X(z) \tag{3.12}$$

We can therefore express the transfer function in the z-domain as:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{q=0}^{Q} b_q z^{-q}}{1 - \sum_{p=1}^{P} a_p z^{-p}} \tag{3.13}$$

Since the z-transform of the impulse response of the filter is equal to the frequency response if it is evaluated at $z = e^{j\omega}$, then we can shape the frequency response by choosing the a and b coefficients in (3.13) to produce poles and zeros at appropriate places in the complex z-plane.

3.3.2 Butterworth Filter Design

The general filter design process consists of choosing feedforward and feedback coefficients according to the desired passband. Once a passband is selected, it remains to decide how many coefficients to include in the difference equation in (3.10). It is possible

[2]We use an infinite impulse response due to its ability to produce flatter pass bands for a given filter order compared with finite impulse response (FIR) filters, and because the infinite impulse response (IIR) is the more general case.

to construct filters with flat passbands and sharp cutoffs, such as rectangular filters, by using several methods including Chebyshev I/II, elliptic, and Bessel designs. For a given filter order, however, the Butterworth design process, described in [16, 17], results in the maximally flat passband. The Butterworth filter is an analog filter design; it achieves its passband by placing the poles in a circle in the s-plane, and using the bilinear transformation $s = \frac{z-1}{z+1}$ to determine their location in the z-plane. The purpose of the bilinear transformation is to discretize a continuous time filter through a conformal mapping from the analog to the digital plane, and is discussed in [18]. The transform is a first order approximation to the relationship $z = e^{sT_s}$, where T_s is the sampling period.

As an example of the Butterworth design process, we choose a 4th order all-pole lowpass filter. The passband is chosen such that the frequency space is divided into 10 bands, hence the passband edge is at the normalized frequency $\Omega_p = 0.1\pi$, and we allow the gain at the passband edge to be -1dB. Having selected the filter order, passband edge, and the gain at the passband edge, we leave the stopband edge and its associated gain as free parameters.

The Butterworth filter that results from the design parameters above has order $M = 4$ and $2M$ s-plane poles. The eight poles are symmetrically distributed about the origin in the s-plane, at a distance given by our chosen cutoff frequency, mapped from the z-plane frequency unit Ω to the s-plane frequency ω. The relationship between the two frequencies is given by: The relationship between ω and Ω is:

$$\frac{e^{j\Omega}-1}{e^{j\Omega}+1} = \frac{2j\sin(\Omega)}{2+2\cos(\Omega)} = j\omega \tag{3.14}$$

This can be shown to simplify to:

$$\omega = \tan(\Omega/2) \tag{3.15}$$

With this frequency transformation, we calculate our filter pole radius in the s-plane, and find $\omega_c = \tan(\Omega_c/2) = \tan(0.1\pi/2) = 0.158$. Since there are 8 poles, the loca-

tions are separated by $2\pi/8$ radians, and $p_1 = \pi/8$, $p_2 = 2\pi/8$, and so on. The pole-zero plot in the s-plane is shown in Figure 3.8.

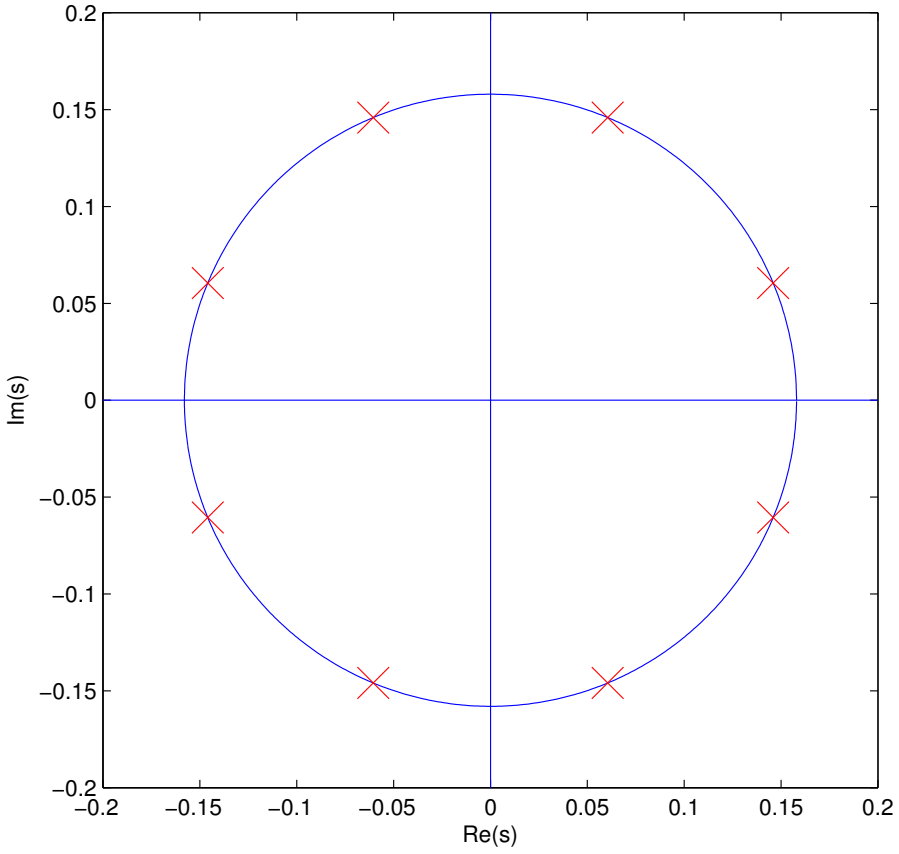

Figure 3.8: s-plane pole-zero plot of 4th order Butterworth filter.

Because the bilinear transformation maps the exterior of the z-plane unit circle to the positive real half of the s-plane, we only use the poles that are in the left half of the s-plane since right side poles result in unstable filters. Returning to the z-plane gives:

$$H(z) = \Pi_{m=1}^{M} \frac{s_m}{(s - s_m)}\bigg|_{s=\frac{z-1}{z+1}} \tag{3.16}$$

where s_m are the s-domain poles. The zeros in the z-domain are equal in number to the poles, but they are all real-valued and located on the unit circle at $\Omega = \pi$. The poles are

conjugate symmetric pairs in the shape of an arc as shown in Figure 3.9. They are found by the inverse bilinear transform $z = \frac{1+s}{1-s}$.

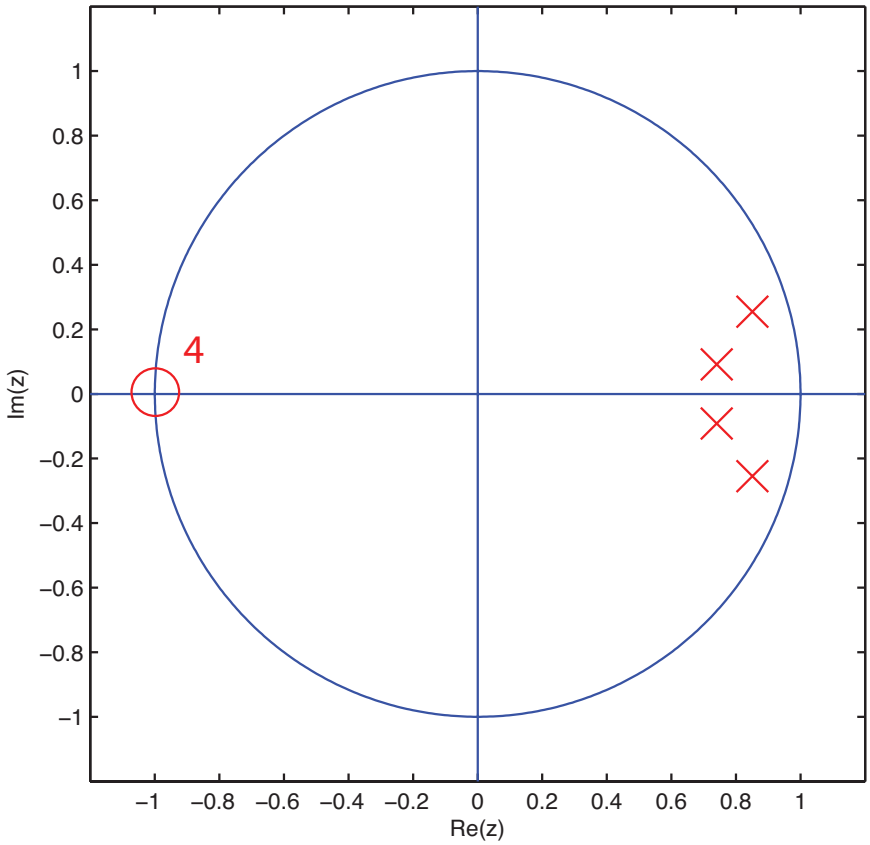

Figure 3.9: z-plane pole-zero plot of 4th order Butterworth filter.

The filter designed above is a lowpass Butterworth filter with a flat passband and a relatively sharp cutoff at $\Omega = 0.1\pi$. The phase response is very nearly linear in the passband, and nonlinear in the stopband. Since the magnitude is low in the stopband, we are not concerned by the nonlinear phase response in that region. The frequency response is computed by measuring the vectors from the zeros and poles to the unit circle in the z-domain. The magnitude of the frequency response is given by:

$$|H(\omega)| = G\,\Pi_{l=1}^{L} \frac{|e^{j\omega} - q_l|}{|e^{j\omega} - p_l|} \tag{3.17}$$

The phase, denoted by $\angle$, is given by:

$$\angle H = \sum_{l=1}^{L} \angle(e^{j\omega} - q_l) - \angle(e^{j\omega} - p_l) \tag{3.18}$$

where q_l and p_l are the z-domain zeros and poles, and G is a normalizing gain factor. The frequency response magnitude and phase are shown in Figure 3.10.

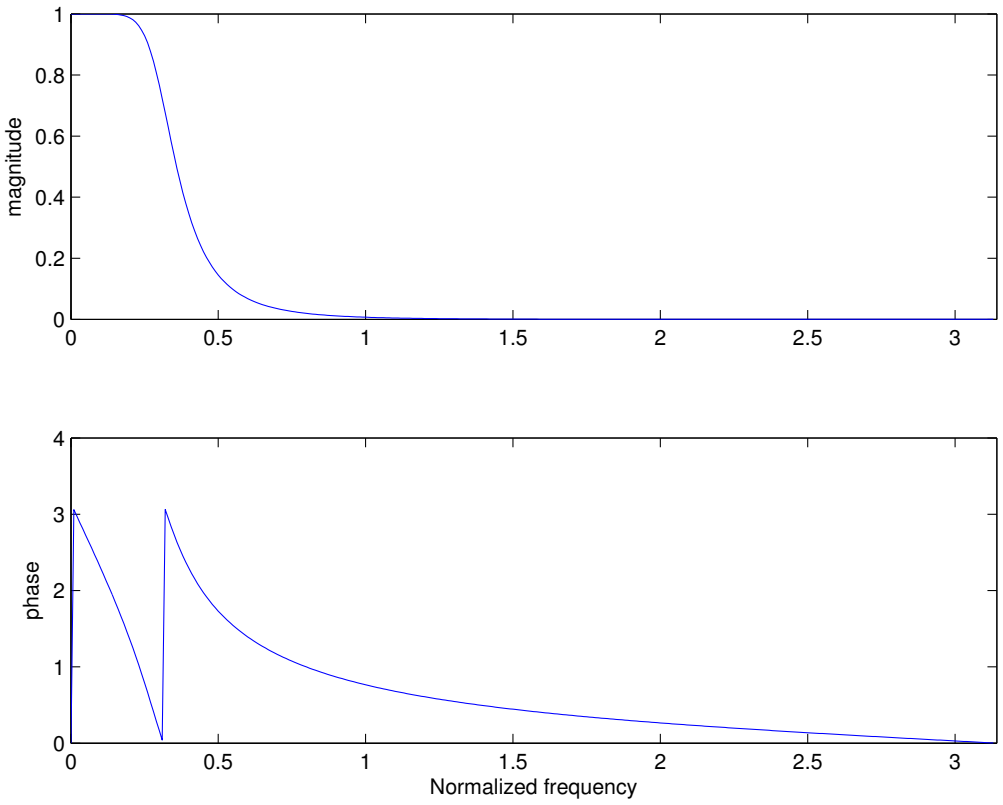

Figure 3.10: Magnitude and phase response of 4th order low-pass Butterworth filter.

3.4 Filter Bank Simulation

In the following simulation, a signal is fed into a filter bank consisting of flat pass bands, and a TFR is generated. The filter bank is constructed using the MATLAB function butter.m which produces the a and b coefficients for a butterworth filter of order and bandwidth determined by the arguments supplied to the function. The prototype filter

is heterodyned, and the filtering operation is repeated until the entire frequency space is spanned. The usefulness of Butterworth filters in filter banks is discussed in [19].

In order to represent the filter outputs as the amplitudes associated with the frequencies described by the vertical axis in a TFR plot, some additional processing is needed. The filtered signal will ideally be unaffected for frequency content that is within the passband of the filter. For a sinusoidal input, the filtered outputs will also be sinusoidal, while the TFR plot is typically understood to be an amplitude plot. Another way of describing this is to observe that the STFT output is a series of FTs associated with different time indexes. The STFT displays one FT after another, so that the vertical, or frequency, distribution consists of all the information from a particular FT. On the other hand, the outputs from a filter bank are time domain signals associated with a given filter. The horizontal, or time, distribution consists of the output from a given bandpass filter. Because these distributions are constructed differently, the meaning of an individual grid value is also different. For the STFT, each value is the amplitude of a FT, which is the associated amplitude for a given frequency component in the original signal. In contrast, the filter bank TFR grid values are the instantaneous amplitudes of signals that are not attenuated by the filter bank.

If the filter bank generated TFR is to be interpreted in the same way as the STFT, the outputs therefore must be rectified and smoothed. This can be done by taking the magnitude of the complex output, and smoothing with a lowpass filter, prior to displaying it as a TFR. The heterodyning that follows the filtering does not need to be carried out explicitly. In other words, we produce the following 2-D TFR:

$$\begin{aligned} S_x[n,k;h] &= e^{-j2\pi\frac{k}{N}n}\left(x[n] * h_{BP}[n;k]\right) \\ &\approxeq h_{LP}[n;k] * |x[n] * h_{BP}[n;k]| \end{aligned} \tag{3.19}$$

where $h_{LP}[n;k]$ is a low pass filter operating on the absolute value of the sub-band filter outputs. We note that the bandpass filter outputs are associated with the passband of that

filter, and we place the results into a matrix at a position given by the center frequency of the passband. The overall effect is similar to multiplying by the downconversion factor $e^{-j2\pi\frac{k}{N}n}$. The entire filtering process is carried out by the code "Subbands.m", listed in Appendix A.

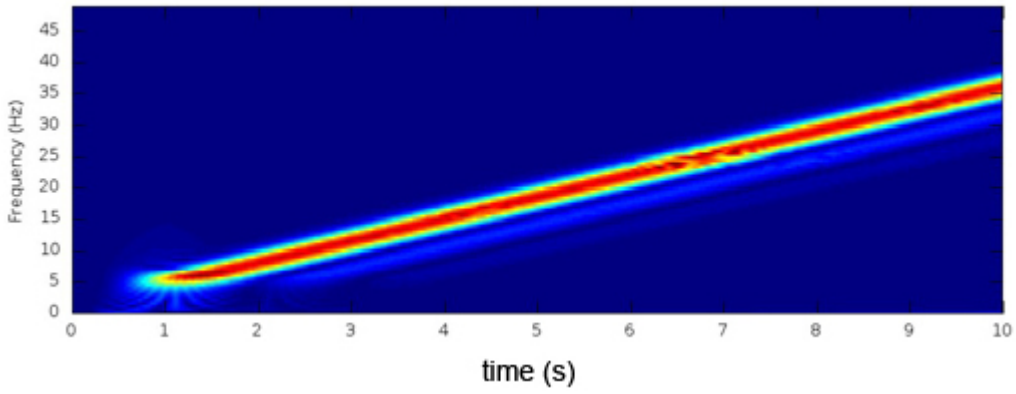

Figure 3.11: Filter bank TFR of a linear chirp; this should be compared with the corresponding one in Figure 3.2.

We demonstrate the rectified and smoothed TFR with the linear chirp of Section 3.1.2, and find that it is similar to the plot obtained earlier. In Figure 3.11, we show the TFR of a TV sinusoid with linearly increasing frequency from 5 Hz to 40 Hz over 10 seconds, with a sampling rate of 100 Hz. The output from the filter bank is rectified and then smoothed by a low-pass Butterworth filter with a passband that includes the frequency of the corresponding sub-band. We are satisfied with the result, and we proceed to more interesting signals in the following chapters, in which we will discuss the accuracy of the filter bank TFR. In particular, we wish to know how well the TFR has captured the important features contained in the signal. In order to quantify the error, we reproduce the signal. A variety of approaches are available to us for this reconstruction, which we now discuss.

Chapter 4

Signal Reconstruction

The output of the analysis filter bank is a 2-D array of output functions of time and frequency. Subject to the inherent low value limitation on the time-bandwidth product due to the uncertainty principle, we expect that we can otherwise reproduce the original signal with arbitrary precision given arbitrarily many-ordered bandpass filters. Even when the filter order is restricted to a relatively small number, the reconstructed signal can be made very accurate through the use of specially chosen reconstruction filters. We will henceforth refer to the filters as *analysis* and *synthesis* stages. In order to minimize relative group delay errors, it is necessary to use filters with linear phase response.

When using an analysis filter bank, a sub-band filter may still result in non-zero output even when the signal segment does not have a frequency present. In other words, there are artifacts in the filter bank time-frequency representation (TFR) due to the non-zero magnitude of the stopband of the bandpass filter. These spurious terms are unavoidable, but they can be minimized if the filter is designed to be sufficiently rectangular. We will show that this is not necessary if perfect reconstruction is the goal. Instead, it is possible to use filters in the synthesis filter bank with properties that "mirror" those in the analysis filter bank.

4.1 Synthesis Filter Banks

The simplest example of a reconstruction bank is the Fourier synthesis sum, in which the short-time Fourier transform (STFT) output value at a given time-frequency location is multiplied by a complex exponential and the reconstructed signal at a given discrete time is the magnitude of the linear combination of the resulting sinusoids, weighted by their respective amplitudes, which are allowed to vary over time. To distinguish from

perfect reconstruction techniques which are discussed in Section 4.2, we call the real result the peudo-reconstruction:

$$x_r[n] = \sum_{k=0}^{N_f} S_x[n,k;h] cos(2\pi \frac{k}{N} n) \tag{4.1}$$

where N_f is the number of filters in the analysis bank. From Chapter 3, the TFR is obtained from the analysis filter bank as:

$$\begin{aligned} S_x[n,k;h] &= h_{LP}[n;k] * |h_{BP}[n;k] * x[n]| \\ &\approx e^{-j2\pi \frac{k}{N} n} (h_{BP}[n;k] * x[n]) \end{aligned} \tag{4.2}$$

where $h_{BP}[n;k] = h[n] e^{j2\pi \frac{k}{N} n}$ and $h_{LP}[n;k]$ is a lowpass smoothing filter with a passband that includes the center frequency of the bandpass filter $h_{BP}[n;k]$ and a stopband that includes all of the bandpass filters $h_{BP}[n;\ell]$ for $\ell \geq k+1$. The synthesis sum adds together a number of sinusoids with frequencies given by the center of their associated filters, and whose amplitudes are varying. The phase of each sinusoid is fixed by its chosen phase at time index $n = 0$. The pseudo-reconstructed signal $x_r[n]$ can be very different from the original analysis signal $x[n]$ asa a result of two possible error types. These errors are:

(a) Error type 1: The bandwidth of a filter may be wide enough to pass multiple frequencies from the signal, but the reconstruction ascribes these frequencies to a single sinusoid at the center frequency of the filter.

(b) Error type 2: The nonzero magnitude of the stopbands can introduce artifacts in the reconstruction, as discussed earlier.

These two sources of error can be eliminated by replacing the Fourier synthesis sum with a synthesis filter bank. To compensate for errors of type 1, we can make the synthesis filter bank bandwidths the same as those of the analysis filter bank. Another possibility is to plot the TFR differently in this case, foregoing the rectification and smoothing steps, since that will blend together different frequency sinusoids that appear within the same passband, and storing the filter outputs at the location of the center frequency of the

passband, but without forcing all of the frequencies to be the same. However, this approach is not compatible with obtaining a highly localized TFR, and we must make compromises when constructing a TFR for visual interpretation.

To correct for errors of type 2, we must make the synthesis and analysis filters complementary so that the cascade combination is unity. In the z-domain, we can represent the analysis and synthesis stages as a cascade combination of parallel analysis filters $H_0(z)$, $H_1(z)$, ... , $H_N(z)$ and parallel synthesis filters with transfer functions $S_0(z)$, $S_1(z)$, ... , $S_N(z)$. For the ith passband, the series combination would then give the overall transfer function $T_i(z) = H_i(z)S_i(z)$, and for perfect reconstruction we require that $T_i(z) = 1$, $i = 0,\ldots,N$.

Attempting to isolate the individual passbands and obtain perfect reconstruction for each of these, however, is not the best approach to take. This is because this approach does not address the leakage from one passband to another, other than by forcing the stopband outputs to be zero, which can only be done by constructing perfect filters. The goal is to obtain perfect reconstruction without recourse to perfect filters. This requires that the distortion due to the stopbands is self-canceling. We achieve this by considering the following two-band filter bank, shown in Figure 4.1.

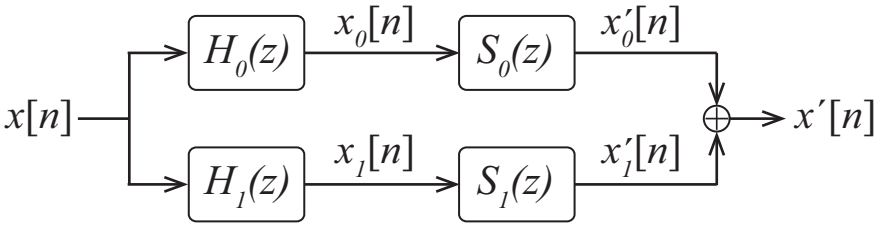

Figure 4.1: Two-band filterbank.

The z-domain output $X'(z)$ is related to the input $X(z)$ by:

$$\begin{aligned} X'[z] &= X[z]H_0(z)S_0(z) + X(z)H_1(z)S_1(z) \\ &= X(z)[H_0(z)S_0(z) + H_1(z)S_1(z)] \\ &= X(z)T(z) \end{aligned} \tag{4.3}$$

where $T(z) = H_0(z)S_0(z) + H_1(z)S_1(z)$. Perfect reconstruction requires that:

$$T(z) = H_0(z)S_0(z) + H_1(z)S_1(z) = 1 \tag{4.4}$$

A common reconstruction filter approach is the so-called quadrature mirror filter (QMF) bank [1, 3, 16, 20, 21]. These filter banks are frequently used in audio sub-band coding, and are therefore particularly concerned with signal reconstruction [3,4]. In many cases, the QMF bank is not intended to be an implementation of the STFT, and we would point out that it is common practice to design analysis filters that are not mere heterodyned versions of a prototype lowpass filter, but rather, consist of qualitatively different filters operating on the different frequency bands. These minor differences do not obscure the primary point of the foregoing discussion, in which the goal of perfect reconstruction is contrasted with our intention of generating meaningful TFRs.

4.1.1 QMF bank design

QMF banks are designed to implement a filter bank by splitting the signal into two bands, lowpass and passband bands. It is designed such that the linear combination of the squared magnitude of the transfer functions of the two filters is equal to one [20,21,45].The analysis filters $H(z)$ are followed by synthesis filters $S(z)$ that mirror the opposite passband of the analysis filter. The result is a two-band system that can be extended by further

subdivisions in a tree structure, or by a series of downsampling operations in a uniform parallel structure. The analysis filters of the fundamental two-band QMF bank satisfy the following relationships:

$$\begin{aligned} |H_0(z)|^2 + |H_1(z)|^2 &= 1 \\ |H_0(z)| &= |H_1(-z)| \end{aligned} \tag{4.5}$$

where $H_0(z)$ is a lowpass filter and $H_1(z)$ is a bandpass filter. The synthesis filters are related to the analysis filters by:

$$\begin{aligned} S_0(z) &= H_1(-z) \\ S_1(z) &= -H_0(-z) \end{aligned} \tag{4.6}$$

With these filter relationships, (4.3) becomes:

$$\begin{aligned} T(z) &= H_0(z)S_0(z) + H_1(z)S_1(z) \\ &= H_0(z)H_1(-z) - H_1(z)H_0(-z) \\ &= H_0(z)H_0(z) - H_1(z)H_1(z) \\ &= [H_0(z)]^2 - [H_1(z)]^2 \end{aligned} \tag{4.7}$$

This can be expressed in terms of the lowpass filter $H_0(z)$ only:

$$T(z) = [H_0(z)]^2 - [H_0(-z)]^2 \tag{4.8}$$

If we relax the constraint on the system transfer function, recognizing that for many purposes, a scaled and shifted output is essentially a perfect reconstruction, we obtain the new constraint:

$$T(z) = gz^{-M} \tag{4.9}$$

where the system gain g does not have to be unity and the system delay z^{-M} can be as long as the filter order M. A solution to Equations (4.7) and (4.8) is the following first-order FIR filter:

$$H_0(z) = 1 + z^{-1} \tag{4.10}$$

By calculating the transfer function, we see that $H_0(z)$ obeys the necessary relationships for perfect reconstruction:

$$\begin{aligned} T(z) &= H_0(z)H_1(-z) - H_1(z)H_0(-z) \\ &= H_0(z)H_0(z) - H_0(-z)H_0(-z) \\ &= (1+z^{-1})(1+z^{-1}) - (1-z^{-1})(1-z^{-1}) \\ &= 4z^{-1} \end{aligned} \tag{4.11}$$

4.2 QMF Reconstruction

In the following simulation, a linear frequency-modulated (FM) chirp signal with sampling rate $f_s = 100$ Hz, duration 10 s, varying linearly from $f_0 = 5$ Hz to $f_1 = 40$ Hz is analyzed with a two-band QMF filter bank using first order Haar filters $H_i(z)$, $i = 0, 1$. The a and b coefficients are found by inspection, and shown in Table 4.1. We note these coefficients for use with the MATLAB function filter.m using the [b,a] syntax for filter definition.

Filter	a	b
$H_0(z)$	1	[1,1]
$H_1(z)$	1	[1,-1]

Table 4.1: Two-band QMF analysis filter coefficients.

We immediately run into a problem when attempting to plot the output of the two-band analysis stage. With the filter bank analysis discussed in Chapter 3, the output of each passband filter was placed at the center frequency of the filter. We can attempt to follow the same procedure in the two-band analysis case by determining the two passbands. Figure 4.2 shows the magnitude frequency response of both the lowpass and bandpass filters. We observe that the filter $H_0(z)$ has a normalized center frequency of $\hat{f}_0 = 0$ and filter $H_1(z)$ has a normalized passband center frequency of $\hat{f}_1 = 0.5$. We therefore plot the output of the lowpass filter on the low frequencies of the TFR, and that of the passband filter on

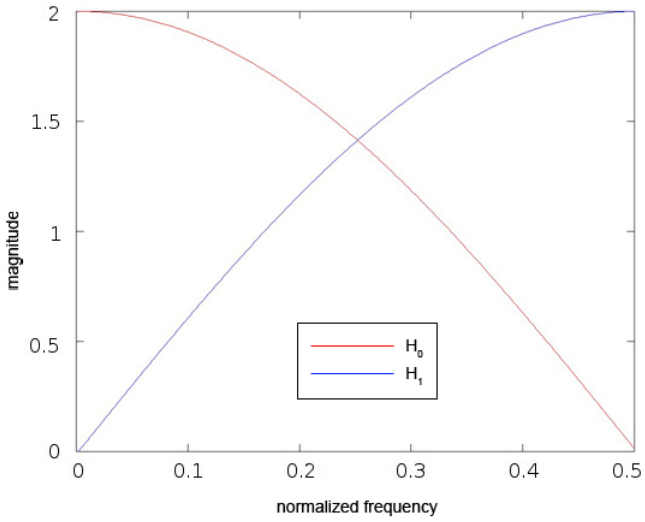

Figure 4.2: First-order FIR filter passbands.

the high frequencies of the TFR, as shown in Figure 4.3. Note that the frequencies are concentrated into narrow bands and displaced from their actual values for simplicity.

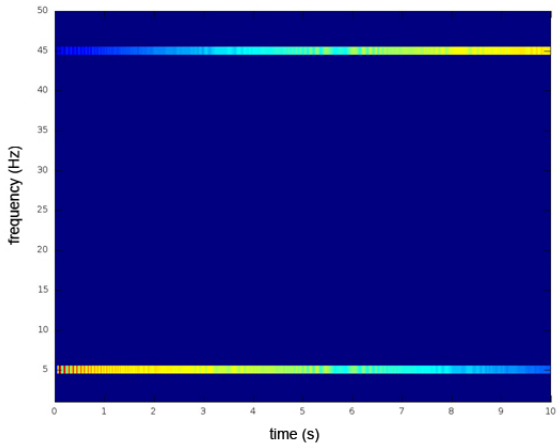

Figure 4.3: TFR of linear FM chirp when a two-band QMF filter bank is used.

As Figure 4.3 demonstrates, the two-band analysis does not provide us with any information on the time-varying nature of the signal. The low frequency band has a high amplitude at the beginning of the signal and it decreases in amplitude over time whereas the high frequency band behaves exactly the opposite. Although the information contained in the TFR is very limited, the signal can nevertheless be perfectly reconstructed.

Specifically, for the reconstruction, the two-band analysis output can be processed with the "mirrored" reconstruction filters whose transfer functions, following (4.5) and (4.6) are given by:

$$\begin{aligned} S_0(z) &= H_1(-z) = 1-(-z)^{-1} = 1+z^{-1} \\ S_1(z) &= -H_0(-z) = -[1+(-z)^{-1}] = -1+z^{-1} \end{aligned} \tag{4.12}$$

The result of the reconstruction stage is shown in Figure 4.4, and we see that the reconstruction includes a gain of 4. A closeup view of the two signals, shown in Figure 4.5, reveals a delay of one sample in the otherwise perfect reconstruction. As demonstrated by this simple example, if we measure the quality of a TFR by the resolution of its representation of the time-varying nature of a signal, then the quality of a TFR is not determined solely by the accuracy of the reconstructed signal. In this example, the this two-band analysis stage produced a TFR that is essentially useless while still allowing the signal to be reconstructed with minimum error.

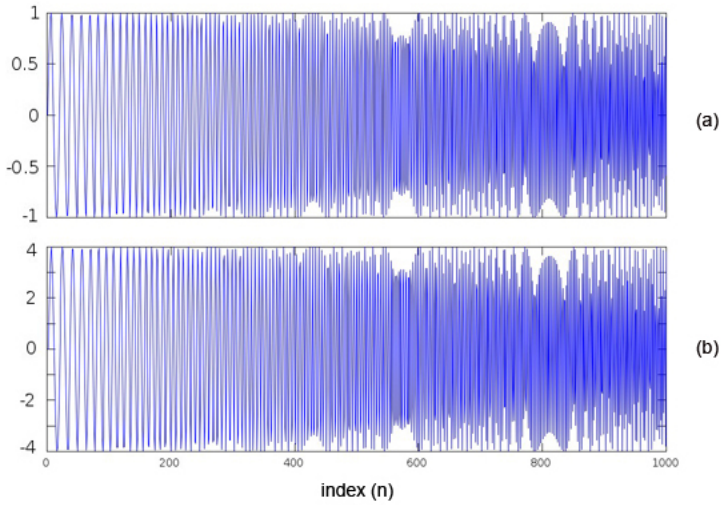

Figure 4.4: (a) Input signal to the analysis filter bank; (b) reconstructed signal using a two-band QMF.

Although our goal is to design an analysis bank to optimize the quality of a TFR, we expect that the reconstruction accuracy will necessarily be an important feature used in

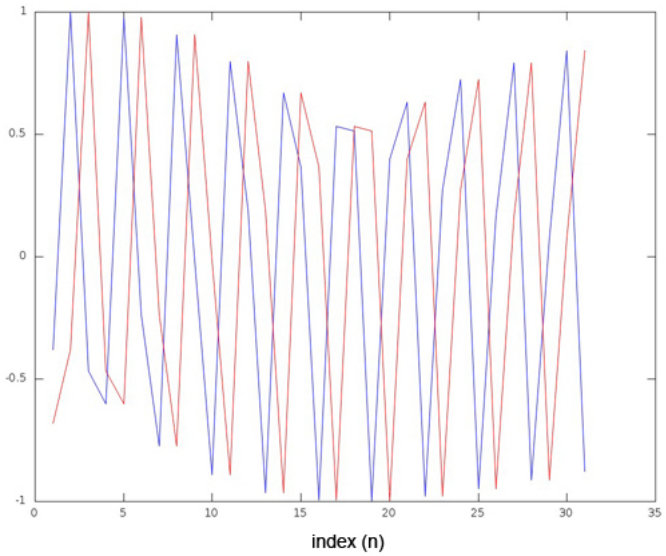

Figure 4.5: Zoomed region demonstrating a segment of the input signal (blue) superimposed with the two-band QMF reconstructed signal (red).

the process. In order to automate any process we may develop, we will calculate the mean-squared error (MSE) of the reconstructed signal as a means of determining the accuracy of the synthesis stage signal output, as we will discuss in the next chapter.

4.3 M-Band QMF bank

In Section 4.2, the TFR obtained using a two-band QMF was shown to have low quality. Next we want to investigate whether the TFR quality can improve when the number of bands increase. Having found FIR filters with perfect mirroring qualities, we use these in a tree structure to further isolate the different frequency bands. Each of the analysis stages has a corresponding mirror in the synthesis bank, and the overall structure is shown in Figure 4.6.

This approach is especially useful for multi-rate coding, allowing signal compression by lowering the sampling rate for high frequency signal components. For this reason, it is used in the MPEG Layer 3 (MP3) standard [24], although not with the first order filters described above. The higher band QMF tree structure with first order filters do not increase

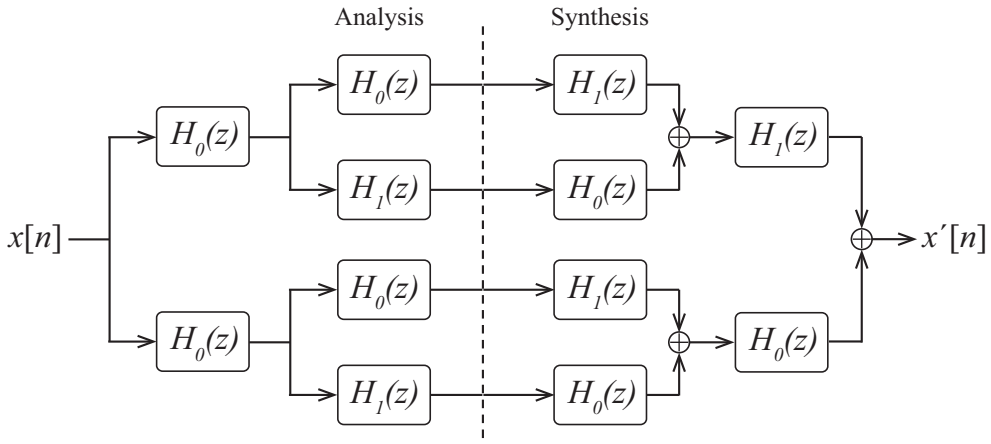

Figure 4.6: Four-band QMF tree structure.

the quality of the TFR, as we show in Figure 4.7. In particular, there are several problems when we try to use this approach to generate TFRs. One problem is that the filters do not divide the frequency space into distinct regions and the resulting TFR assigns frequency components to multiple passbands, resulting in distortion in the time-frequency plane. Indeed this problem is largely a symptom of the poor frequency division achieved by the first order filter, which could be addressed by using a different filter. The frequency response of the tree-structured QMF bank with this filter largely places low and high frequencies in the stopbands, while the midrange components are passed unattenuated by most of the filters.

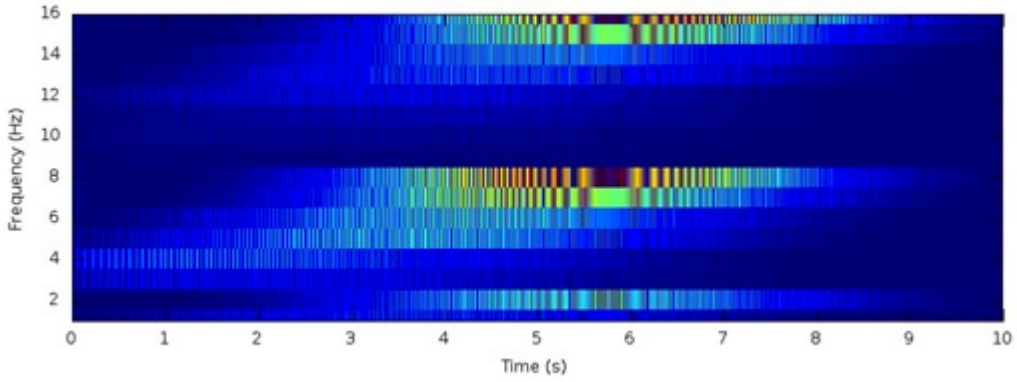

Figure 4.7: Sixteen-band QMF TFR of linear FM chirp.

Another problem is the difficulty in choosing a smoothing filter prior to plotting the resulting TFR. For this reason, we display the 16-band analysis stage of Figure 4.7 in unrectified, unsmoothed form. Furthermore, the determination of where on the TFR plot a given band should be placed requires the additional step of finding the passbands of each terminal filter in the QMF tree, which is in fact a cascade filter constructed from multiple passbands. This problem is not as significant when the frequency division is less complicated.

4.4 Comparative Reconstruction Accuracy

We now compare the reconstruction accuracy of the QMF bank to that of a filterbank with rectangular passbands. As we saw in Figure 3.17, when a filter bank with rectangular passbands is used, a high quality TFR is obtained. We want to investigate the accuracy of the corresponding reconstructed signal. If the synthesis filters are chosen to mirror the analysis filters, the reconstruction can be expected to be quite accurate. For demonstration, we instead consider the pseudo-reconstructed signal in Equation (4.1). The pseudo-reconstructed signal tries to extract the original signal back directly from the TFR. Since we are interested in optimizing the quality of a TFR to start with, it makes sense to consider an optimization criterion that depends on the TFR. We choose this criterion to be the MSE between the input signal and the reconstructed signal, by having the accuracy of the reconstructed signal depend on the TFR. An example of a real application where this optimization would be useful is described next. A rocket motor can be throttled to avoid certain vibrational frequencies at which the vehicle being propelled has destructive resonances. An automated throttling program can review the vibrational data, finding the times during which dangerous resonances were induced, to design a suitable throttle modulation sequence. In such a scenario, signal reconstruction is not of interest, and accuracy of the TFR matrix is paramount. If the TFR matrix indicates the onset of a high amplitude at a dangerous frequency at a particular time, this can be compared to the throttle conditions at

that time, and corrections can be made. In other words, a TFR that is allowed to drift from its apparent meaning in an effort to chase superior signal reconstruction is not necessarily performing its intended function.

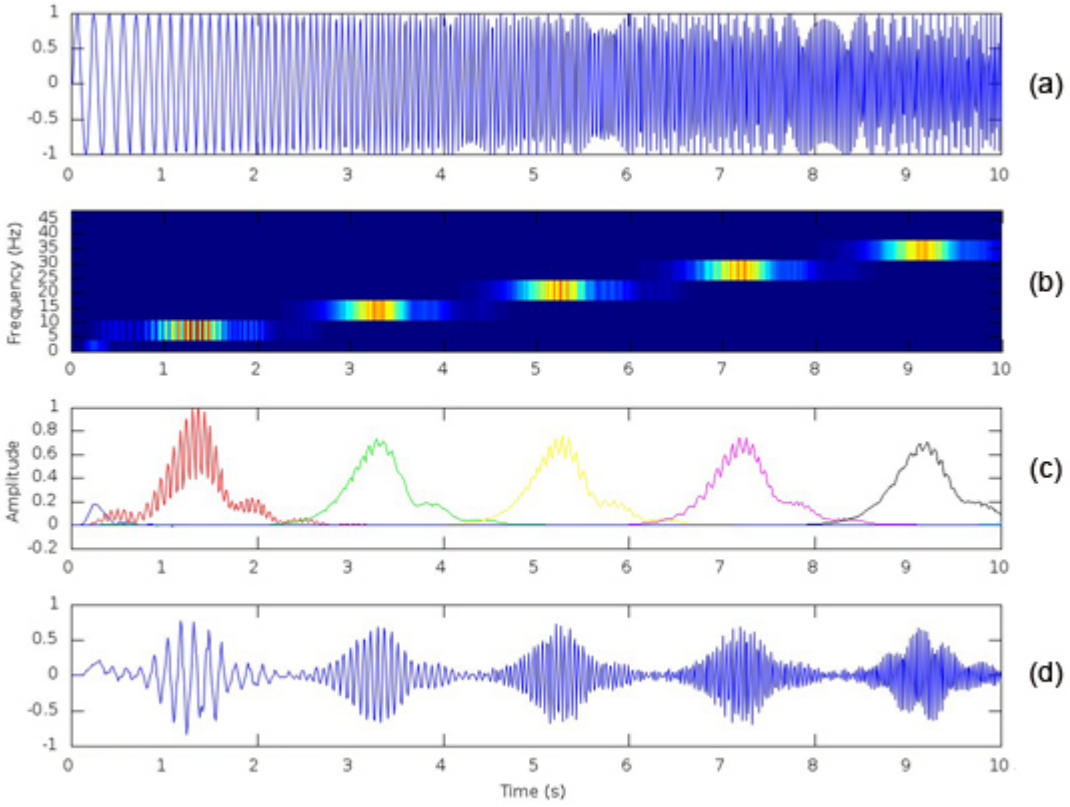

Figure 4.8: Eight-band filterbank. (a) Input linear FM chirp signal to the filter bank with eight rectangular passbands; (b) TFR output of the filter bank; (c) amplitudes of the filter outputs; (d) pseudo-reconstructed signal.

Figure 4.8 (a) shows the linear FM chirp signal with 10 Hz initial and 40 Hz final frequencies that we considered in Figures 4.3, 4.4 and 4.5 with the QMF reconstruction. Figures 4.8 (b)-(d) show the TFR of the signal from an 8-band filter bank spanning the frequency space, the amplitude plots of rectified and smoothed outputs from each of the filters, and the psudo-reconstructed signal. The amplitude peaking of the individual filters in 4.8 (c) causes the reconstructed signal to display similar spurious amplitude oscillations in Figure 4.8 (d). Thus the reconstruction process highlights the importance of minimizing

band overlap. When M filters are used, we can expect no more than M sinusoids contributing to the reconstruction, and each of these sinusoids are expected to have a peaked amplitude at the time index for which the instantaneous frequency of the input chirp is equal to the center frequency of that filter's passband.

Chapter 5

A Time-Frequency Figure of Merit

In this chapter, we propose a method for obtaining time-frequency representations (TFRs) that are highly-localized in the time-frequency plane by designing the parameters of the analysis filter bank to minimize the mean-squared error (MSE) between the input signal and the reconstructed signal at the synthesis stage. We say that the quality of a TFR improves when the TFR becomes more localized in the time-frequency plane in that it can then better resolve all time-varying spectral components in the analysis signal. An extensive discussion on optimization techniques can be found in [22, 23]. The challenge is then to develop a robust and efficient method for minimizing the MSE, which begins by identifying a suitable TFR figure of merit.

5.1 Mean-Squared Error

As discussed in Chapter 4, using an analysis filter bank to generate a high quality TFR has a different motivation than when perfect signal reconstruction is desired. When concentrating on TFR quality, the method of signal reconstruction for calculating the MSE should be modified to better match the TFR, as we discussed briefly in the beginning of Section 4.3. Instead of using mirror filters, for instance, one can use the pseudo reconstruction that is based on extracting signal information by taking the inverse FT of the TFR at different points of interest. Our proposed method for TFR optimization presupposes that the amplitude associated with a given time and frequency grid location should correspond to an existing signal component. Therefore, in reconstructing the signal to determine the accuracy of the plot, we use the pseudo redonstruction in Equation (41). Naturally, the reconstruction errors are much greater than they would otherwise be. We use this approach in the following examples.

We quantify the accuracy of the reconstructed signal by computing its MSE, and we then use the MSE as a measure of the TFR quality. The MSE of the reconstructed signal is given by:

$$\mathscr{E} = \frac{1}{N}\sum(x[n] - x'[n])^2 \tag{5.1}$$

where $x[n]$ is the original signal and $x'[n]$ is the reconstructed signal.

We use the pseudo reconstruction $x_r[n]$ in Equation (4.1) to generate $x'[n]$, attributing all amplitude information in a particular sub-band to the center frequency of the band. The pseudo-reconstructed signal is given by:

$$x_r[n] = \sum_{k=0}^{N_f-1} S_x[n,k;N_f] cos(2\pi\frac{k}{N}n) \tag{5.2}$$

where k represents the discrete center frequencies of the filter passbands, which is also where their values $S_x[n,k;N_f]$ are plotted in the TFR. The synthesis bank effectively consists of ideal bandpass filters. We emphasize that we are not trying to characterize the quality of a synthesis bank, and can therefore replace each real synthesis filter with an ideal one whose magnitude is $|S[k]| = 1$. One consequence of this approach is that wideband filter banks concentrate frequency information into pure sinusoids. We can therefore anticipate that the optimized TFR will tend to use more filters when there are many frequency components close to one-another.

5.1.1 Linear Chirp MSE

The MSE calculation naturally must follow the signal reconstruction stage, which depends on the synthesis bank parameters. The QMF bank, simple synthesis sum, and improved synthesis sum can therefore be compared in terms of their respective MSE, which we call $\mathscr{E}$ in the following discussion.

5.1.1.1 2-Band QMF MSE

In Section 4.1.1, we found that a simple two-band QMF bank was capable of perfect reconstruction when appropriate synthesis filters were used. Taking into consideration the gain of the QMF filters, the calculated MSE was found to be:

$$\mathscr{E} = 0.87288$$

The MSE could be further reduced by taking into consideration the reconstruction delay, given by the transfer function $T(z) = 4z^{-1}$. If we compensate for this delay by shifting the output one sample to the left, then the resulting MSE is:

$$\mathscr{E} = 3.4654 \times 10^{-4}$$

It is less than 0.035% of the amplitude, or nearly zero, as expected.

5.2 Proposed Optimization Procedure

We consider an analysis signal $x[n]$ consisting of N samples, and we assume that we decompose the signal using an analysis filter bank with N_f rectangular non-overlapping passband filters. Using the analysis filter bank, the TFR is given by $S_x[n,k_i;h] = e^{j2\pi(k_i/N)n}$ $(x[n] * h_{BP}[n;k_i])$, where k_i, $i = 0,\ldots,N_f - 1$ are the center frequencies of the passband filters. After pseudo-reconstruction, the MSE can be calculated using the reconstructed signal $x_r[n]$.

For a fair MSE calculation, the reconstructed signal needs to be normalized and time shifted. If the overall transfer function of the analysis stage is known, then it can be used to determine how to adjust the gain and time delay of the reconstructed signal; however, this is normally not the case. In general, we can instead normalize both the input and reconstructed signals to compensate for unknown gain factors, and we can perform a cross correlation between the two normalized signals to compensate for unknown time

shifts [15]. We first obtain the normalized input signal sequence using $\bar{x}[n] = x[n]/x_{\max}$, $n = 0, \ldots, N-1$, where $x_{\max}$ is the maximum value of $x[n]$ over all n. In order to compensate for unknown shifts, cross- correlation is used by first segmenting the reconstructed signal into I frames, with the length of each frame being L_f samples. The optimal value m_i of the time shift in the ith frame is found from

$$m_i = \arg\max_m \sum_{n=n_i}^{n_i+L_f-1} x[n] x_r[n+m] \tag{5.3}$$

We also obtain the normalized and time-shifted reconstructed signal sequence for the ith frame as

$$\bar{x}_{rs}[n] = x_r[n+m_i]/x_{r\ \max} \tag{5.4}$$

for $n = n_i, \ldots, n_i + L_f - 1$, where n_i is the starting time index of the ith frame. The first frame can be chosen to begin at time index $n_0 = 0$, or at some later time to accelerate the process, allowing the optimization to be carried out over a segment of the signal. Subsequent frames begin at $n_i = n_0 + i\,L_f$.

The optimization procedure chooses the optimal number $N_{f,\mathrm{opt}}$ of analysis filters in order to minimize the MSE between the two new signals

$$N_{f,\mathrm{opt}} = \arg\min_{N_f} \sum (\bar{x}[n] - \bar{x}_{rs}[n])^2 \tag{5.5}$$

This results in our proposed optimization procedure: compute the MSE of many frames of the normalized, shifted pseudo-reconstruction, and minimize this by changing the number of filters and their widths. We refer to this computed MSE as our "figure of merit", and we minimize it by changing the number of filters used to analyze the signal. The bandwidths are automatically altered to distribute the frequency space from DC to the Nyquist frequency without overlap. We describe the process and the motivations in detail in the following sections.

5.2.1 *Cross-Correlation Frame Length*

In order to determine the accuracy of the TFR generated by the two-band QMF analysis stage, we cannot use the mirror synthesis filters. Instead, we must use a reconstruction technique that reflects our understanding of the meaning of the TFR matrix. The simple synthesis sum is such a technique, but we hardly need to compute the MSE of the reconstructed signal to immediately recognize that the TFR generated by the two-band QMF analysis stage, shown in Figure 4.4, is poor. Instead, we compute the MSE of the rectangular filter bank's output, applying the preprocessing steps of normalization and time shifting described above. Looking at the reconstructed signal, shown in Figure 5.1, we see that it has an amplitude varying envelope due to the interaction of the filter transition regions and the changing signal frequency. This is not present in the original signal, and will increase the MSE of the output. There is also a phase error introduced by the arbitrary division of the time domain into discrete frequency blocks in the synthesis process. This is a comparatively minor error, and rather than undertaking to correct it, we will use the cross-correlation process to compensate for phase error in the MSE calculation.

The phase error shown in Figure 5.1 is changing over time. Near the beginning of the window, it is a three sample shift, increasing to four samples in the middle, and then decreasing down to 2 samples near the end of the window. A simple cross correlation involving the complete input and output sequences mixes these effects together. The correlation needs to be done on short frames of the two signals in order to avoid this blending effect. We propose the following way of determining the length L_f of the frames, using the ratio of signal length N to filter number N_f:

$$L_f = \lceil (N/N_f) \rceil \tag{5.6}$$

where the operation $\lceil \cdot \rceil$ denotes rounding to the next highest integer.

In the case of Figure 5.1, the filter bank involved $N_f = 32$ filters, and since $N = 1000$, the correlation frame length $L_f = \lceil (\frac{1000}{32}) \rceil = 32$. Examining two different frames

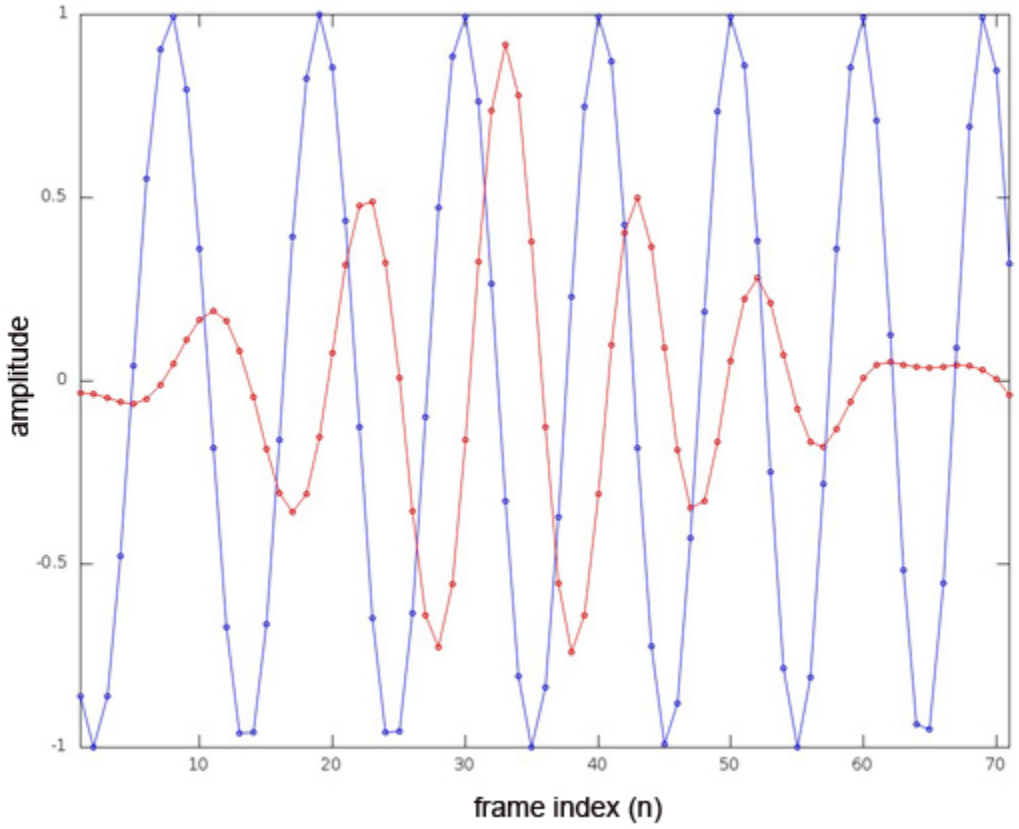

Figure 5.1: Close-Up of input (blue) and reconstructed (red) signals. The reconstructed signal has a varying amplitude due to the narrow passband of the analysis filter, and a phase error due to the arbitrary phase assignment in the pseudo reconstruction process.

in Figure 5.2, each of length $L_f = 32$, we see that the shift m is constant throughout each frame: $m = 3$ in the frame on the left and $m = 5$ in the one on the right. Since the frames constitute regions of the output that share a uniform phase error, it is reasonable to perform a separate cross correlation for each of these. In the case of a 32 band filterbank, we will perform $ceil(N/L_f) = ceil(1000/32) = 32$ separate correlations, each producing its own shift value, and the MSE is computed between the signal and the shifted reconstruction.

We now test the phase-correction process on a linear chirp increasing from 100 Hz to 800 Hz over a 10 second period, sampled at a rate of 10 kHz and analyzed by a 1000-band Butterworth filter bank. Figure 5.3 shows the input signal, the pseudo-reconstructed signal, and the time-shift compensated reconstructed signal. The phase errors are well

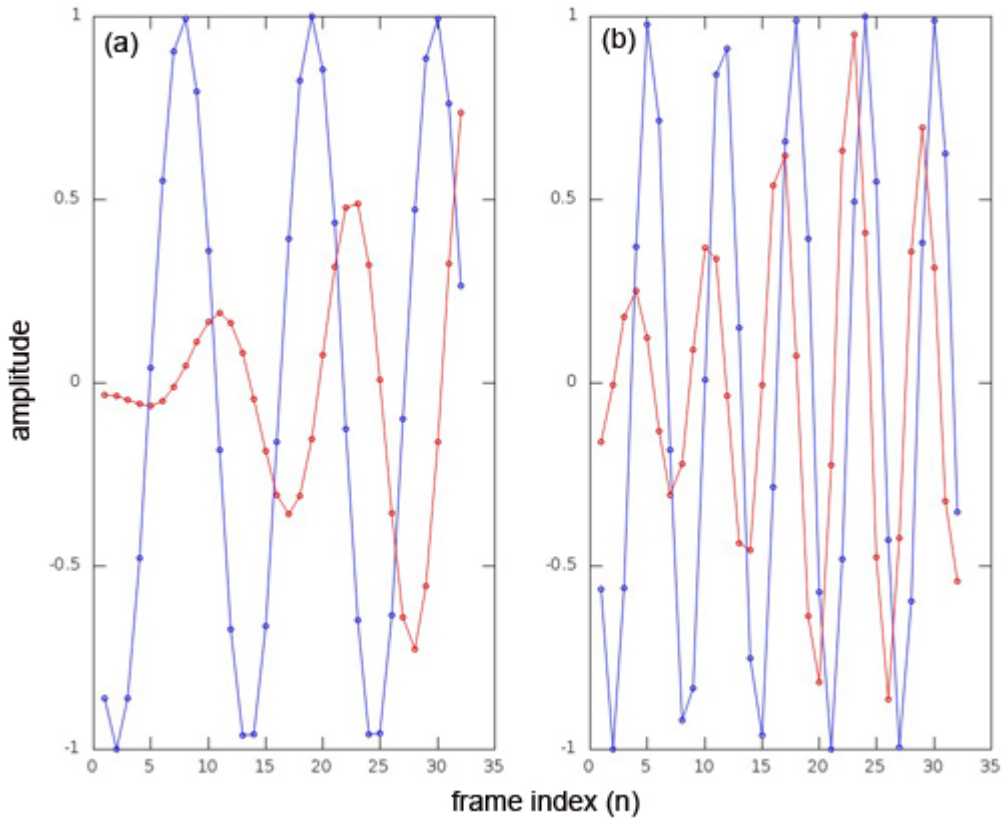

Figure 5.2: Two different frames of an input linear FM chirp and the reconstructed signal. (a) delay = 3 samples; (b) delay = 5 samples.

corrected. The shifted output, shown in green in Figure 5.3, is in phase with the input, shown in blue.

5.3 Filterbank TFR of a Musical Piece

We conclude this chapter by processing a musical piece with a Butterworth filter bank. The results of the program can be seen in the following figures. The program was applied to a .wav recording of "Twinkle Twinkle Little Star" played on a piano with two hands. Figure 5.4 shows the sheet music for the piece. There are 3 staves shown in the piece. The top staff is the melody to be sung, the middle staff is for the right hand to play on the piano, and the bottom staff is for the left hand (these are the low frequency notes). The musical notation expresses frequency on the vertical axis and time on the horizontal

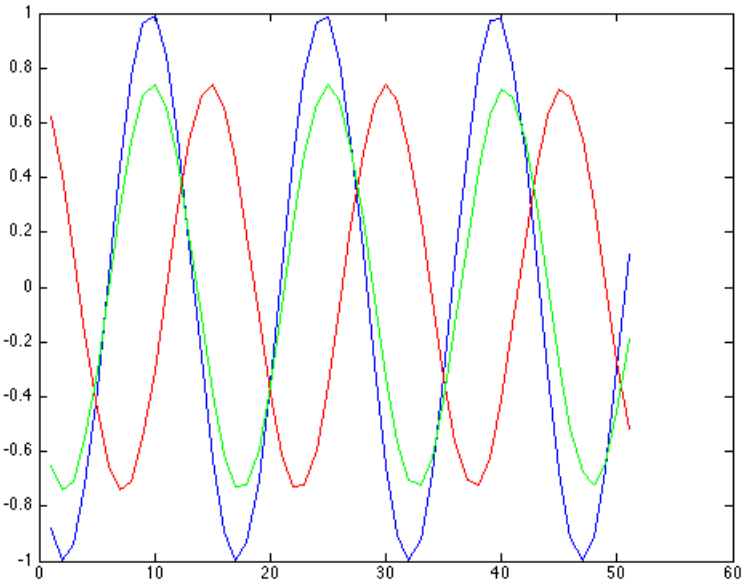

Figure 5.3: Input Signal (blue), Reconstruction (red), Shifted Reconstruction (green).

axis, increasing up and to the right, which is also how we have arranged the TFR plots in this report.

From a preliminary processing operation with 32 frequency bands, the program produces the output shown in Figure 5.5 (a). The synthesized signal resulting from this process is difficult to identify as bearing very much relation to the well-known song. Upon looking at the resulting TFR, one immediately wonders if more sub-bands would improve matters. Indeed, as the number of frequency bins is increased, the melody becomes increasingly recognizable, until a threshold is passed above which further increases in window duration result in a less accurate reproduction of the signal.

The result of a 2048 band analysis bank is shown in Figure 5.5 (c), and although the frequency resolution is high, now the melody is unrecognizable due to the low time resolution. The optimum determined by the procedure proposed in this chapter is obtained with $N_{f\text{opt}} = 512$ frequency bands, shown in Figure 5.5 (b) (only increases of powers of 2 have been attempted). Realistically, this number of bands is overkill, and only needed because the detail resides in several narrow bands, and this analysis bank uses fixed widths

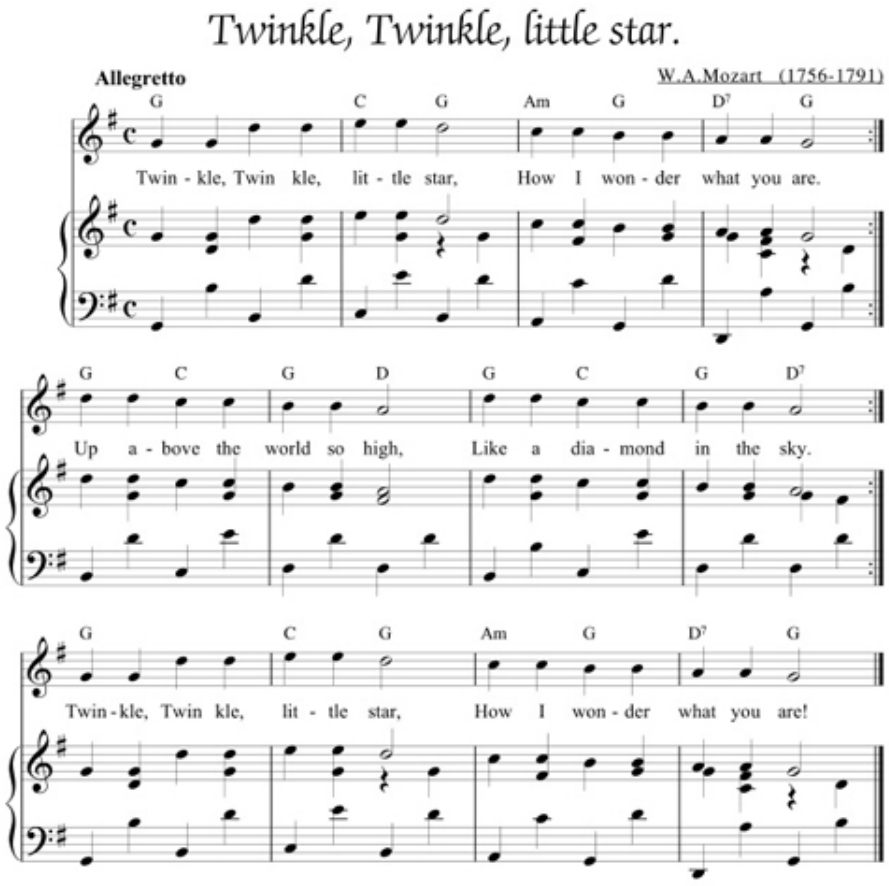

Figure 5.4: Sheet music for song used in filter bank tests, to aid in visual identification of the melody. Figure taken from [25].

for all of the sub-bands. Some of the plots in this report were produced in a grayscale to minimize confusion and for easier comparison with the composition.

The value of the TFR is especially apparent if it is re-plotted showing only the frequency range up to the top note played by the right hand in the piano sheet music. In other words, zooming in on the plot's lower third shows only those notes that are listed in the composition. Higher frequencies that appear in the TFR plots of Figure 5.5 are due to harmonics and are not a part of the composition as written, but are instead a result of the physics of the instrument used to produce the music. Figure 5.6 shows the sheet music alongside the relevant frequency-range portion of the TFR.

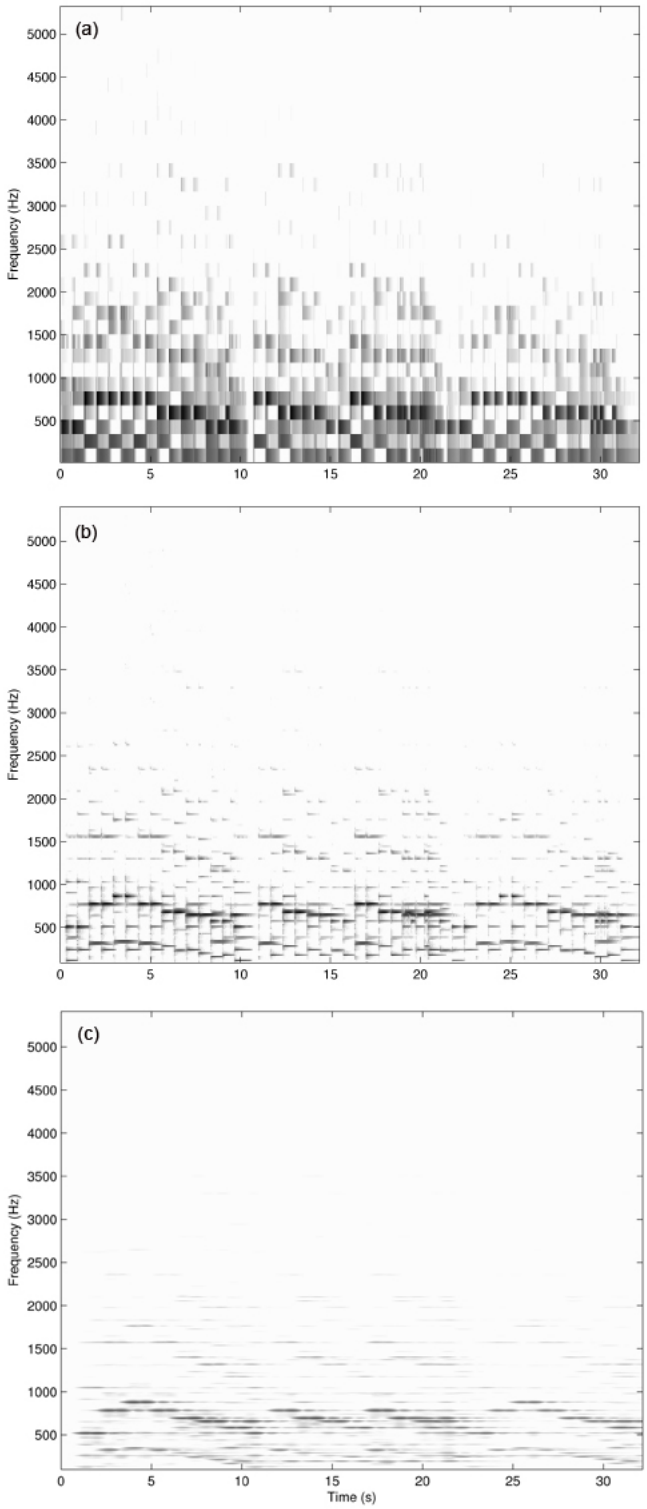

Figure 5.5: Comparison of $N_f = 32$ (a), 512 (b), and 2048 (c) frequency sub-bands used in the analysis stage.

The sheet music indicates that the first note is middle G, which corresponds to approximately 392 Hz. The recording is played in a different key, thus the TFR of the recording shows a starting note (the first of the note pair circled in blue) that is played somewhat higher (about 523 Hz), which is a C one octave above middle C. Other than the change of pitch, the upper hands are essentially identical, except that the note coincident with the lyric "star" is played twice in the recording (circled in red) whereas Mozart wrote it to be a single half note (also circled in red). In addition, the left hands are slightly different.

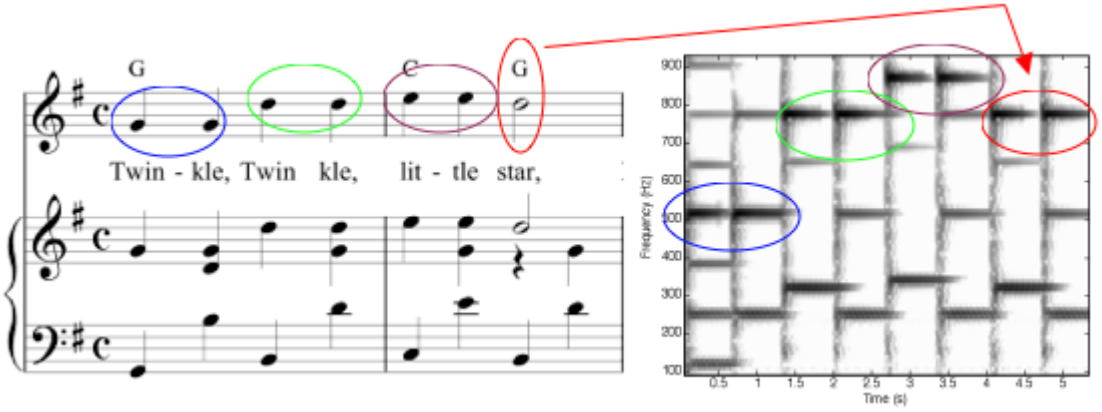

Figure 5.6: Segment of sheet music compared with analysis results. Corresponding notes in the melody are circled in matching colors. The recording plays the last note in the stanza twice, circled in red.

The TFR also highlights the impulsive nature of the piano notes, demonstrating why the piano is categorized as both a string and a percussion instrument. Each note begins with something close to a delayed delta function $\delta(t - t_i)$ where t_i is the time that the ith note is to be played. The sound is produced by a hammer striking a string, which is thereby set into vibration at a frequency given by the tension of the string and its length between fixed nodes. The impulsive nature of the sound's onset is further exacerbated by the processing, so that it produces an audible click in the synthesized .wav file. This can be remedied by post processing techniques to remove broadband short duration components.

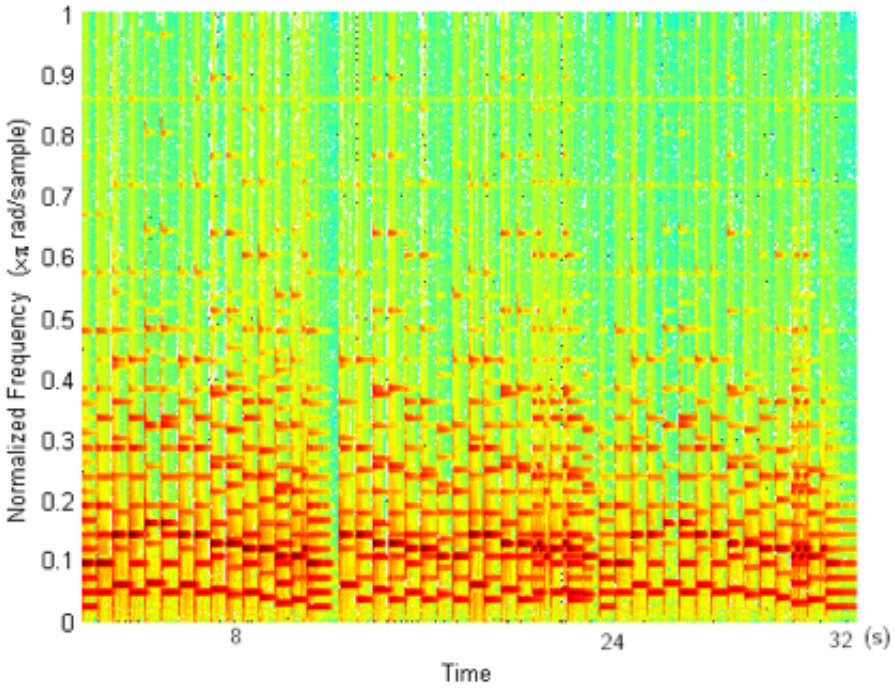

Figure 5.7: Spectrogram of the recording shown in previous figures. The spectrogram is essentially identical to the filter bank analysis output, as expected.

Similar plots to those shown in Figures 5.5 above can be generated by MATLAB's "spectrogram" function. Such a plot is shown in Figure 5.7, and a speed test is performed using MATLAB functions "tic" and "toc" to compare its execution time with that of the filter bank program used in this report. The test is performed for two similar resolution plots. We observed that the elapsed time for the filter bank was 11.9 s and for the spectrogram it was 0.3 s.

The filter bank is considerably slower, primarily because it uses a "for" loop, and each iteration of the loop involves a call of the "butter" function to design the next Butterworth bandpass filter to be used. Vectorization and prior filter construction would decrease execution time.

The MSE criterion for optimization of the filter bank TFR is the phase-corrected sample of the reconstructed signal. In order to determine the accuracy of the plot, it is not

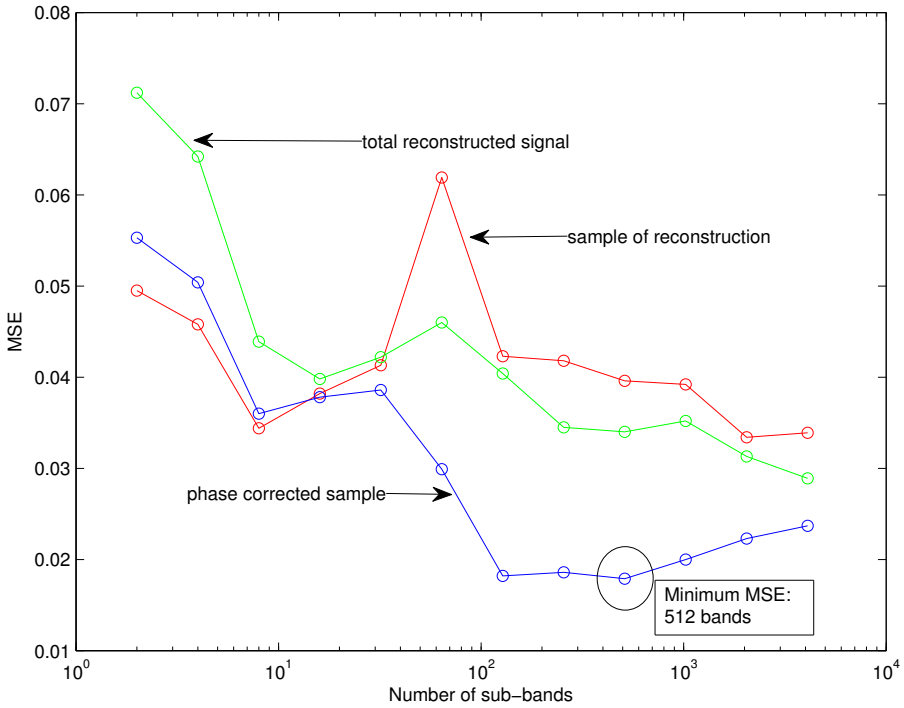

Figure 5.8: MSE as a function of the number of sub-bands. Green: entire reconstruction; Red: segment of reconstruction; Blue: phase-corrected segment of reconstruction.

necessary to phase correct the entire reconstruction, and by choosing to perform the cross-correlation only on a short segment of the output, the process is accelerated. Computing the MSE of the entire signal, a short segment, and the phase-corrected version of the segment, we find that the third approach comports with our intuition from observing the TFRs produced by different numbers of sub-bands. In the case of the audio signal used here, it is also possible to listen to the reconstructed signals and judge the TFR quality by their fidelity to the original on a subjective basis. Both of these measures lead us to the conclusion that 512 sub-bands yields the best representation, and indeed, the MSE of the shifted sample of the reconstructed signal also is minimized when 512 filters are used, as we show in Figure 5.8. Neither the total output's MSE nor that of a short segment agree with this conclusion, and proper filter bank optimization requires the third approach, which is what we have developed in this report.

Chapter 6

Software and Hardware Implementations of the Filter Bank

The advantage of the short time Fourier transform (STFT) performed by MATLAB's "spectrogram.m" function over the filter bank lies in the efficiency of the fast Fourier transform (FFT), which takes advantage of periodicities in the discrete Fourier transform (DFT) to lower the number of multiply-accumulate operations (MAC) needed. Thus the STFT is computationally less expensive.

6.1 Computational Efficiency of the Fast Fourier Transform

The DFT can be expressed as:

$$X[k] = \sum_{n=0}^{N-1} x[n] W_N^{nk} \tag{6.1}$$

where we define $W_N \triangleq e^{-j\frac{2\pi}{N}}$. The sum can be split into even and odd parts:

$$\begin{aligned} X[k] &= \sum_{n=0}^{N/2-1} x[2n] W_{N/2}^{nk} + \sum_{n=0}^{N/2-1} x[2n+1] W_N^{(2n+1)k} \\ &= \sum_{n=0}^{N/2-1} x_{even}[n] W_{N/2}^{nk} + \sum_{n=0}^{N/2-1} x_{odd}[n] W_{N/2}^{nk} W_N^k \\ &= X_{even}[k] + W_N^k X_{odd}[k] \end{aligned} \tag{6.2}$$

where $x_{even}[n] = x[2n]$ and $x_{odd}[n] = x[2n+1]$. The even and odd sums in Equation (6.2) are both periodic, because:

$$\begin{aligned} X_{even}[k+N/2] &= \sum_{n=0}^{N/2-1} x_{even}[n] W_{N/2}^{n(k+N/2)} \\ &= \sum_{n=0}^{N/2-1} x_{even}[n] e^{-j2\pi n\frac{2}{N}(k+N/2)} = \sum_{n=0}^{N/2-1} x_{even}[n] e^{-j2\pi n\frac{2}{N}k} e^{-j2\pi n\frac{2}{N}(N/2)} \\ &= \sum_{n=0}^{N/2-1} x_{even}[n] e^{-j2\pi n\frac{2}{N}k} e^{-j2\pi n} = \sum_{n=0}^{N/2-1} x_{even}[n] e^{-j2\pi n\frac{2}{N}k} = X_{even}[k] \end{aligned} \tag{6.3}$$

Similarly, $X_{odd}[k] = X_{odd}[k+N/2]$. Furthermore, we can express the periodicity of the overall DFT, using Equation (6.2), resulting in:

$$\begin{aligned} X[k+N/2] &= X_{even}[k+N/2] + W_N^{k+N/2} X_{odd}[k+N/2] \\ &= X_{even}[k] + W_N^k e^{-j\frac{2\pi}{N}\frac{N}{2}} X_{odd}[k] \\ &= X_{even}[k] - W_N^k X_{odd}[k], \quad k = 0...N/2-1 \end{aligned} \tag{6.4}$$

Because of the periodicities of Equations (6.4) and (6.2), the N-point DFT can be computed by performing two N/2-point DFTs. A flowchart of the process is shown in Figure 6.1, and is known as a "butterfly" because of its appearance.

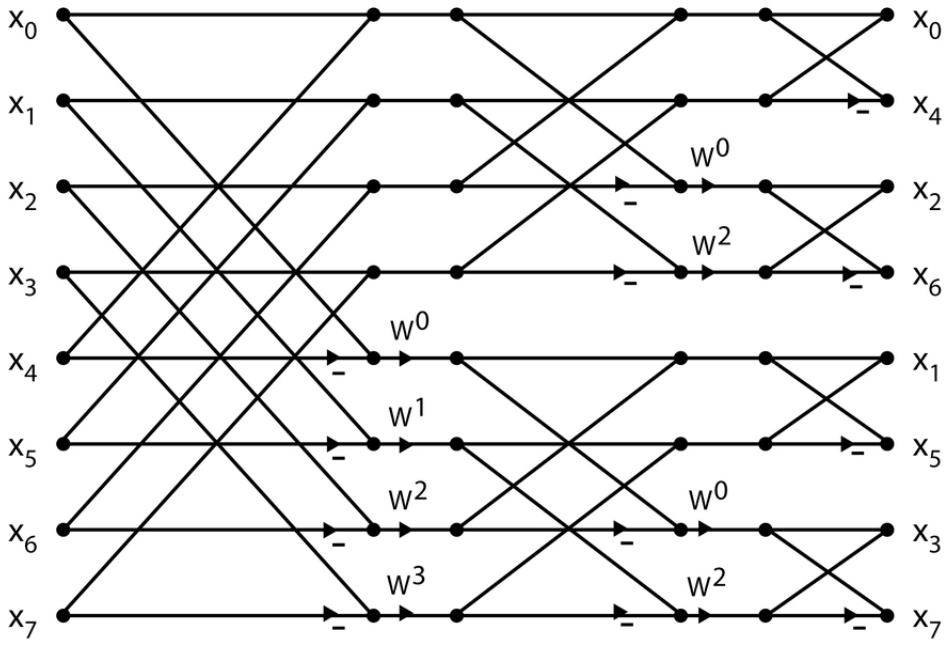

Figure 6.1: 8-point FFT butterfly. Figure taken from [41].

For a sequence of length N, the FFT lowers the number of complex multiplications relative to the DFT from N^2 to $\frac{N}{2} log_2 N$. The computational savings becomes significant when N is large and is a power of 2. It is largely for this reason that digital spectral analysis is often favored, and dedicated digital signal processing (DSP) hardware leverages these algorithmic efficiencies further by implementing parallel MAC stages and fast analog-digital-analog converters.

6.2 Hardware Implementation of Filter Banks

Computer processing is not the only way in which signal analysis can be performed. Physical hardware filter banks can be implemented that compare favorably with the spectrogram generated by digital processing. An example of the difference between the two approaches can be seen in the comparison of the FFT spectral analysis techniques that are often employed in bench-top oscilloscopes, versus swept-tuned spectrum analyzers. The difference between the two instruments is essentially the choice of spectrogram interpretation.

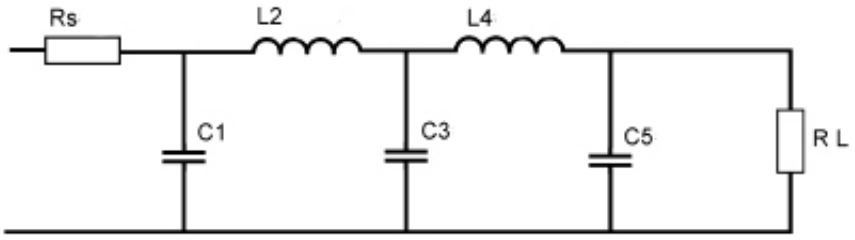

Figure 6.2: Fifth-order passive lowpass Butterworth filter. Figure adapted from [39].

The STFT can be computed with maximum speed by choosing a time window whose length is a power of two. Because the algorithm is performed in software, it is necessary to sample and store the signal. The filter bank, on the other hand, is not inherently a digital process, and can be performed in the analog realm, by constructing bandpass filters as either passive or active filters. A Cauer topology [39] resulting in a fifth order bandpass filter using passive components only is shown in Figure 6.2. Changing the passband is achieved by using different component values. Varactor diodes and electronically variable inductors can be used for this purpose. Alternatively, active filters [36] can be used, as shown in Figure 6.3.

By using either active or passive filters with tunable elements, it is possible to construct the filter bank described in previous chapters in a way that allows fast filtering of the input signal prior to sampling. When a swept-tuned spectrum analyzer inspects a signal,

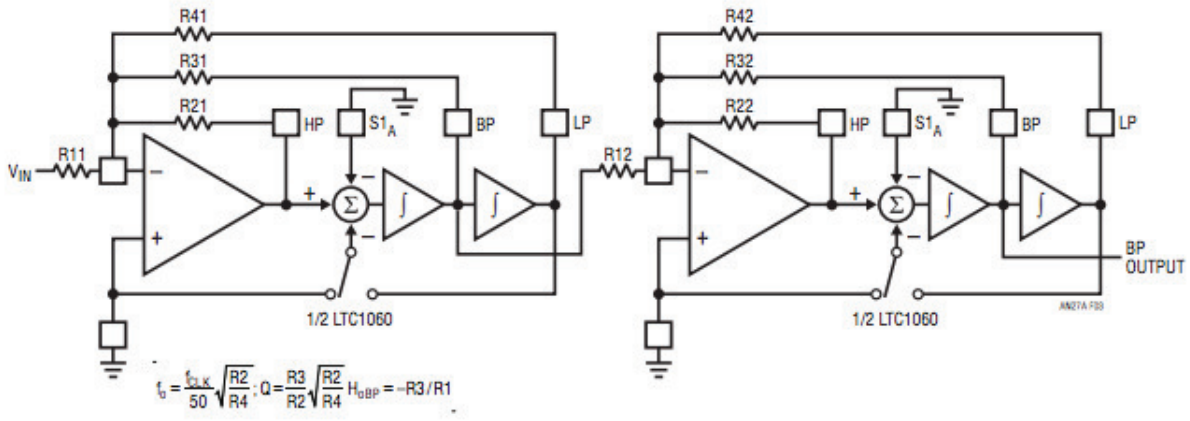

Figure 6.3: Active 4th order Butterworth filter from two cascaded 2nd order sections. Figure taken from [36].

it windows the signal by means of bandpass filters in this way. The windowed signal is then mixed with a swept-frequency complex exponential, and peaks in the resulting output are equivalent to those in the power spectrum computed by the FFT. The process typically results in higher dynamic range within a window, with superior resolution bandwidth, and is also faster than the software FFT [26] . Similarly, radio tuners do not need to sample, digitize, and compute the FFT over the entire spectrum in order to identify the broadcast frequencies.

6.3 Decimation and Interpolation

The hardware filter bank has the potential to render moot the question of when to carry out the signal sampling. In a software implementation of the filter bank, however, this is an important issue. When a signal is distributed into its sub-bands, the filtered outputs contain redundant information. Because the time-frequency representation (TFR) produced by a filter bank records the amplitudes associated with the passband frequencies, all of the information is retained by a critically sampled subset of each sub-band filter output. The TFR can therefore be stored economically by downsampling the filtered data. Multi-rate systems make use of this economy by decimating the filter outputs. The Noble

identity for decimation and interpolation [37], shown schematically in Figure 6.4, adds further efficiency by allowing the downsampling, or decimation, to be performed before filtering.

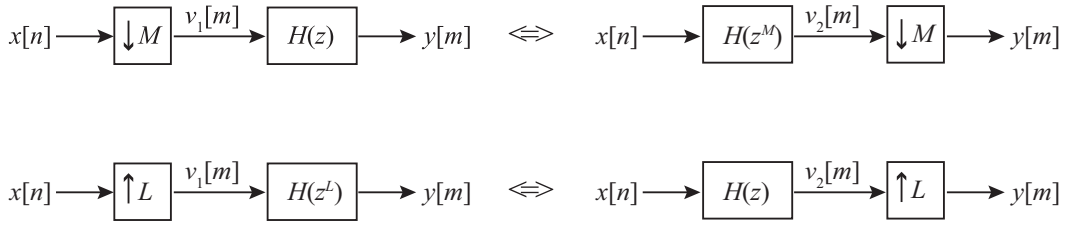

Figure 6.4: Noble identities for decimation (top) and interpolation (bottom). Figure taken from [36].

6.3.1 Polyphase Decomposition

When the decimation is performed prior to filtering, considerable computational savings is realized, but there is a delay as a result. This delay is dependent on the decimation factor M in Figure 6.4. The delay can be accounted for by shifting the samples by an appropriate amount, and then filtering. In other words, the z-domain filter $H(z)$ is replaced with $H(z^{-M})$. When the decimation rate is not an integer, it is possible to achieve the result by a cascade combination of decimation and interpolation. The resulting phase effect is then accounted for with shifts at the appropriate times. The overall process can be distributed among multiple filters for each sub-band. This is known as polyphase decomposition [38].

The polyphase quadrature filter bank (PQF) extends the quadrature mirror filter (QMF) approach described in Chapter 4, by distributing the sub-bands into even and odd-ordered polynomials, each associated with a different delay. As with the QMF bank, perfect reconstruction is possible, and multi-rate synthesis can be achieved with reduced complexity. In the filter bank developed in this report, however, the PQF approach is not

as helpful, because we do not know the decimation factor that will preserve the amplitude information. We developed our TFR in Chapter 5 by finding the passbands that result in the minimum reconstruction MSE when using the inverse discrete cosine transform (IDCT) for the synthesis. These passbands have a bandwidth that minimizes the time-bandwidth product subject to the constraint of no passband overlap permitted. It is possible to produce a lower MSE with different window shapes, or with different TFRs such as the Gabor wavelet transform [5], but we have developed an optimized filter bank TFR that does not allow overlap, with the understanding that we wish to minimize frequency distortion, and are willing to tolerate the temporal spread that this constraint forces.

Because the optimal filters have substantial bandwidth, the frequencies of signals that are passed are unknown. We have developed a way of correcting this issue, so that polyphase methods can be used, allowing multi-rate encoding and perfect reconstruction. The strategy is to identify the indexes associated with peak values, and average these to determine the non integer periods of the signal frequencies. This approach allows the TFR to be optimized by means of the IDCT sum while also permitting near-perfect reconstruction. However, this process is inefficient, and it is not the goal of this report. It is important to distinguish between the different requirements of perfect reconstruction and TFR quality. The latter can be quantified by means of a simple Fourier synthesis sum and the MSE of the resulting reconstruction, as we discuss in Chapter 5.

6.3.2 Signal Compression

Sub-band coding is used in many communication systems, and is also a part of compression schemes such as the MPEG-1 audio layer 3 (MP3) specification. The MP3 algorithm is a lossy data compression method that uses a perceptual model of hearing in which certain information can be discarded without being noticed by the listener. As described in [24], the key to determining what can be left out is in the relative energies of certain sub-bands that mask one another due to the psychoacoustics of the human auditory

system. In order to identify these "inaudible" sub-bands, the processing must preserve the total signal energy prior to compression. The analysis stage is followed by a masking function, based on the properties of the psychoacoustic model. After masking, signal components that fall below an empirically determined hearing threshold are discarded. This provides much of the compression, with the remainder mostly coming from multiple digitization rates.

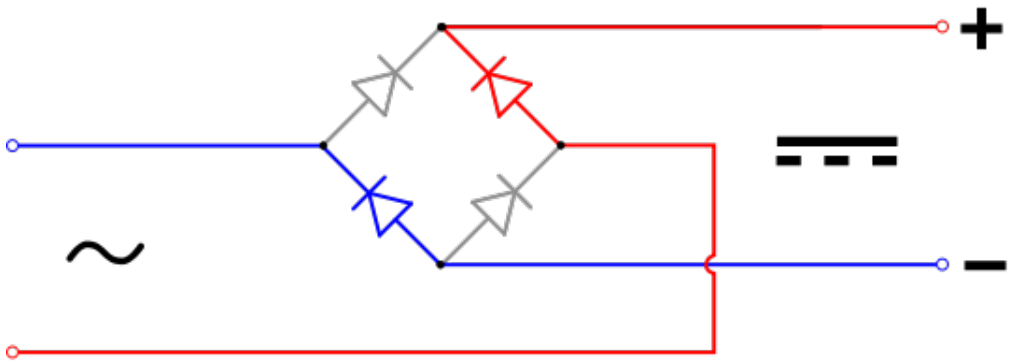

Figure 6.5: Full-wave bridge rectifier. Figure taken from [42].

Decimation and interpolation are not needed to produce the filter bank TFR developed herein. Our distribution can be decimated by any factor, as long as it preserves the group delay information. Instead of decimating at a rate given by the passband, we perform a rectification followed by a smoothing filter, in a manner that is analogous to the output stage of a typical direct current (DC) power supply. In a software implementation of a filter bank, the rectification is achieved simply by taking the absolute value of the filter output, while in a hardware application, a typical approach is an arrangement of diodes to form a full-wave bridge rectifier, shown in Figure 6.5. The smoothing filter is implemented in the same manner as the sub-band filters, but in a lowpass configuration. After reviewing the TFR produced by a filter bank, it is possible to identify opportunities for signal compression. We explore these possibilities for the case of the musical signal of Section 5.4. The signal consists of notes played on a piano, with durations that are on

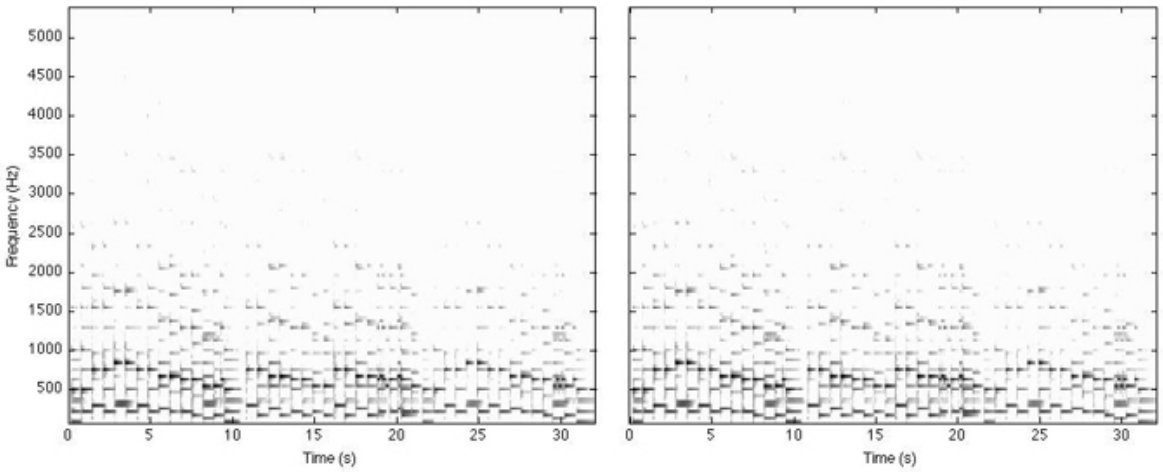

Figure 6.6: Two different output sampling rates. Left: 11.025 kHz, right: 50 Hz. The plots are indistinguishable.

the order of $2/3$ of a second. The broadband frequency component indicating the onset of each note lasts for about $0.1s$. For this reason, very little information is lost if we encode the amplitude data associated with each frequency at $50Hz$, rather than at $11.025kHz$ which is the critical sampling rate for the highest frequency component in the TFR. Since we are dealing strictly with amplitude information, our choice of sampling rate is unrelated to the frequencies that are stored. In Figure 6.6, we show 128 band TFR plots with output frame rates of 11.025 kHz and 50 Hz, and they are indistinguishable. The information that is discarded by lowering the output frame rate is negligible, but the TFR sizes are very different: a 354000x128 matrix for the former compared to a 1609x128 matrix for the latter. Stored as unsigned 8-bit integers, the file sizes are 45.3 Mbytes and 206 kbytes, respectively. This compares well with the original file, which was a 708 kbyte file stored in the "wave" format. Further compression can be achieved by discarding frequency bands that do not exceed a minimum threshold value. A quick glance at the plots in Figure 6.6 is sufficient to surmise that more than half of the sub-bands can be discarded with little loss of information. In other words, it is not unreasonable to expect the file to be compressed to a size of less than 100 kbytes, which would be a compression factor of about 10.

Number of sub-bands: 4096 Filter Bandwidth: 1.3 Hz			Number of sub-bands: 64 Filter Bandwidth: 86.1 Hz		
sub-band	freq. (Hz)	max. amplitude	sub-band	freq. (Hz)	max. amplitude
125	260.8	0.142	3	265.9	0.340
150	293.3	0.112	4	348.9	0.157
177	328.3	0.109			
178	329.6	0.145			
193	349.0	0.124			
326	521.5	0.143	5	431.8	0.215
327	522.8	0.191	6	514.8	0.835
328	524.1	0.300			
329	525.4	0.146			
377	587.7	0.224	7	597.7	0.850
378	589.0	0.202			
431	657.7	0.147	8	680.7	0.985
432	659.0	0.252			
433	660.3	0.321			
434	661.6	0.147			
462	697.9	0.137			
463	699.2	0.307			
464	700.5	0.298			
465	701.8	0.108			
527	782.2	0.106	9	763.7	1.000
528	783.5	0.137			
529	784.8	0.226			
530	786.1	0.356			
531	787.4	0.194			
602	879.5	0.152	10	846.6	0.914
603	880.8	0.251			
604	882.1	0.168			
Threshold (m): 0.1					

Table 6.1: Fusion of narrowband and broadband filter bank sub-bands. Left: High frequency resolution information (27 of the 4096 bands exceed the threshold), Right: High time resolution information (8 of the 64 bands exceed the threshold). All other bands are discarded, with minimal loss of information.

It is also possible to fuse the data from two different filter banks, allowing the high frequency resolution obtained by the narrowband bank to be incorporated into the high time resolution of the broadband bank. Table 6.1 shows the frequencies for which the normalized maximum amplitude exceeds a threshold m, for two different filter banks. The frequencies identified by each bank are grouped according to the similarities of their amplitude time series. The broadband, decimated sub-band signals can be normalized by the maximum amplitude of each of the corresponding narrowband signals. The result can then be interpolated to recover the high sampling rate amplitude time series, and these can be upconverted to the frequencies determined by the sub-bands of the narrowband bank. This process allows the signal to be stored at a low sampling rate, with a minimal number of sub-bands. High frequency resolution is maintained by including a file header containing the amplitude scaling factors and frequencies produced by the narrowband bank.

Chapter 7

Conclusions and Future Work

Analyzing a signal with time-varying frequency content can be accomplished in numerous ways. We showed that the spectrogram or short-time Fourier transform (STFT) can provide adequate time-frequency representations, although they have time-frequency resolution tradeoffs due to the use of windowing. The STFT in discrete form can be viewed in the frequency domain as a filter bank decomposition. The overall structure can be viewed as an analysis bank, followed by a synthesis bank in the event that signal reconstruction is desired. Among the various challenges that accompany the generation of a time-frequency representation (TFR) of a signal is the evaluation of its accuracy. This work proposed a method of evaluating a TFR by comparing the reconstruction that results from a simple synthesis sum with the input signal.

The mean-squared error (MSE) of the reconstruction is exacerbated by the filter bank gain and phase error. We concentrate on applications in which these signal attributes are not of interest, and consequently, we do not want to confuse TFR accuracy by these sources of error. By normalizing both signals and by performing a cross correlation on short frames of the input and reconstructed signals, we show that these sources of error can be greatly mitigated. The MSE can then be used as a figure of merit from which an optimization routine can determine suitable filter bandwidths.

The importance of the filter bandwidths stems from the relationship between time and bandwidth, which involves an inherent tradeoff. The time window used in the STFT determines the prototype filter's passband shape, and the two functions are Fourier transform pairs. A narrower passband involves a wider time window, and as a result precise frequency identification will blur group delay measurements. An optimized TFR must therefore choose between these two sources of inaccuracy, and the MSE of the gain and

phase-corrected reconstruction is a good candidate for determining the filter bank parameters.

Another source of signal reconstruction error is the passband overlap between adjacent filters. We mention an application in which accurate frequency, time, and amplitude descriptions are important: a rocket engine. The Saturn V rocket that powered the Apollo moon missions involved several different engine designs, eventually including one that could be throttled and one that could not. Both of these rocket engines were determined during their development stages to be capable of vibrating at frequencies that corresponded to destructive resonances in the spacecraft structure. For the fixed-throttle "F-1" engine, the injection plate was designed (among other things) with these frequencies in mind, and for the throttled "J-2" engine, the intake plumbing was modified to avoid these frequencies [27, 28]. In both cases, an accurate TFR would be valuable during the design stage, while one that introduced spurious analysis content would not be as helpful. This is a good example of an application that benefits from the clarity of the filter bank's output in contrast to quadratic TFRs such as the Wigner distribution.

In order to avoid distortion due to stopband leakage or passband overlap, two approaches are useful. In the first, signal reconstruction is emphasized over TFR accuracy, and this is accomplished by using a synthesis bank whose filters "mirror" those of the analysis bank. These "quadrature mirror filter" QMF banks can in principle produce perfect reconstruction after correcting for known gain and delay effects. We demonstrate that a simple two-band QMF structure can produce a TFR of dubious quality while reconstructing the signal with near-perfection. When TFR frequency resolution is prioritized, on the other hand, rectangular passbands are desirable, so that signal amplitudes are not distorted by the passband shape. The resulting reconstruction using a simple synthesis bank can be accomplished in several ways, but for TFR characterization, it is sufficient to use the center frequency of each filter passband in the synthesis. Then the best TFR will be produced

by the bank that distributes the frequency space into non-overlapping rectangular sections, and this is achieved by Butterworth filters.

The Butterworth filter is the maximally flat passband filter for a given filter order. Other filters can be made with steeper transition regions, but at the expense of passband ripple. The Butterworth design process involves a bilinear transformation that maps an analog filter in the s-plane to a digital filter on the unit circle in the z-plane. The process involves a frequency warping to account for the nonlinear frequency mapping that accompanies the transform. The resulting filter has a stopband rolloff that is -6 dB/octave for each order of the filter.

The mathematical equivalence of the filter bank and the STFT results in a prototype lowpass filter that is heterodyned repeatedly to span the frequency space from DC to the Nyquist frequency; but the Butterworth filter design process works by separately designing each filter in the bank. In principle, however, the Butterworth design process is identical to a frequency upconversion, since it involves rotating the poles about the z-plane to achieve the passband relocation. For the best filter bank performance, the filters should be many-ordered, but this can cause stability problems due to rounding errors. Better performance results from the method of "second-order sections," in which filters are described as a series of second-order filters, with quadratic terms on the numerator and denominator. For this reason, they are also described as "biquads" [43]:

$$H_1(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} \tag{7.1}$$

The biquads are then cascaded to produce higher order filters:

$$H(z) = g \prod_{k=1}^{L} \frac{b_{0k} + b_{1k} z^{-1} + b_{2k} z^{-2}}{1 + a_{1k} z^{-1} + a_{2k} z^{-2}} \tag{7.2}$$

where g is a gain term. A 1st order filter can be added for odd-ordered cases. The MATLAB signal processing toolbox includes routines for designing Butterworth filters of various types, and allows them to be specified according to their zeros and poles, or as a matrix

of second-order sections. With the latter approach, high filter orders can be used, resulting in excellent filter bank performance, as we have demonstrated.

We developed the filter bank and its figure of merit using a linear FM chirp as a signal example. We then applied it to a musical piece, which was a piano rendition of Mozart's "Twinkle Twinkle Little Star," played with two hands. By analyzing the piece both with a small number and a large number of filters in the bank, we were able to see the effect on the TFR, and the resulting tradeoffs. The adaptive-window TFR algorithm described in this paper was effective at generating a representation of the signal that is sufficiently detailed and accurate that its musical content is immediately apparent. In addition to allowing the sequence of notes to be seen, this approach also conveyed information concerning the character of the sound. For instance, it can be induced from the TFR that the sound was generated by a piano, due to the broadband impulses that precede each note, which is characteristic of a percussion instrument. If the piece were played on a violin, for instance, we would expect the onset of each note to be a rising amplitude signal at a fixed frequency. The TFR also alerted us to the presence of harmonic content that we otherwise might not expect to find, since we are adapted to home in on the fundamental frequency due to the manner in which the human auditory system processes sound.

7.1 Future Work

Efforts to improve the STFT that have been undertaken in the literature often focus on the addition of a signal-dependent window, which leads in the direction of quadratic TFRs. We have stated that the optimal time window shape (in a linear music-encoding TFR) is that of a sinc function, whose Fourier transform is rectangular, hence, optimum in the frequency domain (neglecting group delay precision). It must be noted that this is only true when the goal is to identify some "fundamental" frequency with precision, at the expense of precise knowledge of other signal details. Theoretically, the minimum time-bandwidth product is achieved with a radially Gaussian filter, but for certain classes

of signals such as musical recordings, frequency precision will generally impose much stricter requirements on filter shape than time location, which can tolerate a fair amount of inaccuracy without affecting the result to a noticeable extent.

The proposed approach can benefit from variable bandwidth filters. Those filters with center frequencies above the frequency space in which the majority of signal detail is found can be broader than the others. This would significantly increase the program's execution speed. The filter bank TFR can be performed with analog signals and filters, with the result from each sub-band rectified and smoothed. The resulting signals can be tested for maximum amplitude, and sub-bands that do not exceed a threshold can be discarded. The remaining signals can be digitized at very low rates, unrelated to the frequencies of the sub-bands, and determined according to their time varying properties. The results from banks with large numbers of narrowband filters can be combined with those from banks with small numbers of broadband filters, allowing higher simultaneous resolution in both frequency and time than is otherwise achievable.

REFERENCES

[1] S. R. S. Ross, *Digital Butterworth Filters for Quadrature Mirror Filtering in Subband Coders*, University of Illinois at Urbana-Champaign, 1984.

[2] T.H. Glisson, *Introduction to Circuit Analysis and Design*, Springer, 2011.

[3] P.P. Vaidyanathan, "Filter banks in digital communications," *IEEE Circuits and Systems Magazine*, vol.1, no.2, pp. 4- 25, 2001.

[4] J. Halamek, M. Kasal, M. Villa, P. Cofrancesco,"Bandpass signal processing," *Radioengineering*, Vol. 6, No. 1, April, 1997.

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

[6] E. Hecht, *Optics*, Addison Wesley, 2002.

[7] W. Heisenberg, *The Physical Principles of the Quantum Theory*, Dover, 1930.

[8] M. Testorf, B. Hennelly, J. Ojeda-Castañeda, "Wigner distribution in optics", *Phase-Space Optics: Fundamentals and Applications*, McGraw-Hill, 2009, pp. 1-44.

[9] M. J. Bastiaans, "The Wigner distribution function and its applications to optics," *AIP conference proceedings*, Vol. 65, No. 1, p. 292, 1980.

[10] A. Luis, L.M. Sanchez-Brea, "Ray picture of diffraction gratings," *Optics Communications*, 2009.

[11] A. W. Lohmann, "Image rotation, Wigner rotation, and the fractional Fourier transform," *Journal of the Optical Society of America A*, vol. 10, no 10, pp. 2181-2186, 1993.

[12] S. Kay, *Intuitive Probability and Random Processes using MATLAB*, Springer, 2006.

[13] L. Stankovic, "A method for time-frequency analysis," *IEEE Transactions on Signal Processing* Vol. 42, No. 1, 1994.

[14] J. H. McClellan, R. W. Schafer, M. A. Yoder, *Signal Processing First*, Prentice Hall, 2003.

[15] J. G. Proakis, M. Salehi, *Communication Systems Engineering*, Prentice Hall, 2001.

[16] A. Spanias, *Digital Signal Processing, An Interactive Approach*, LuLu Press, 2007.

[17] I. W. Selesnick, C. S. Burrus, "Generalized digital Butterworth filter design," *IEEE Transactions on Signal Processing*, Vol. 46, No. 6, pp. 1688 - 1694, 1998.

[18] W. H. Hayt, J. E. Kemmerly, S. M. Durbin, *Engineering Circuit Analysis*, McGraw Hill, 2007.

[19] L. Milic, S. Damjanovic, "Frequency transformations of half-band Butterworth filters with filter bank applications," *Proceedings of the 7th International Conference on Telecommunications in Modern Satellite, Cable and Broadcasting Services*, Vol. 1, pp. 107 - 110, 2005.

[20] S. Chen, M. Kao, "Low-complexity, perfect reconstruction FIR QMF bank," *Electronics Letters*, Vol. 34, No.15, pp. 1477 - 1478, 1998.

[21] T. E. Tuncer, T. Q. Nguyen, "General analysis of two-band QMF banks," *IEEE Transactions on Signal Processing*, Vol. 43, No. 2, pp. 544 - 548, 1995.

[22] S. M. Kay, *Fundamentals of Statistical Signal Processing, Volume I: Estimation Theory*, Prentice Hall, 1993.

[23] S. M. Kay, *Fundamentals of Statistical Signal Processing, Volume II: Detection Theory*, Prentice Hall, 1993.

[24] S. Shlien, "Guide to MPEG-1 audio standard," *IEEE Transactions on Broadcasting*, Vol. 40, No. 4, pp. 206 - 218, 1994.

[25] Virtual Sheet Music. (2009) *Twinkle Twinkle Little Star* [Online]. Available: http://www.virtualsheetmusic.com/downloads/Twinkle.html

[26] Agilent Technologies. (2004). *Agilent Performance Spectrum Analyzer Series Swept and FFT Analysis Application Note* [Online]. Available: http://cp.literature.agilent.com/litweb/pdf/5980-3081EN.pdf

[27] L. R. Ellison, M. D. Moser, "Combustion instability analysis and the effects of drop size on acoustic driving rocket flow," *Proceedings of the JANNAF Propulsion Conference*, Las Vegas, Nevada, May 10-13, 2004.

[28] V. Yang, W. Anderson, "Liquid rocket engine combustion instability," *Progress in Aeronautics and Astronautics* Vol. 169, P. Zarchan, editor, 1995.

[29] F. Sattar, G. Salomonsson, "The use of a filter bank and the Wigner-Ville distribution for time-frequency representation," *IEEE Transactions on Signal Processing*, Vol 47, No. 6, pp. 1776 - 1783, 1999.

[30] B. Chen, Y. Chung, D. Huang, "Optimal time-frequency deconvolution filter design for nonstationary signal transmission through a fading channel: AF filter bank approach," *IEEE Transactions on Signal Processing*, Vol. 46, No. 12, pp. 3220 - 3234, 1998.

[31] B. Chen, C. Lin, Y. Chen, "Optimal signal reconstruction in noisy filter bank systems: multirate Kalman synthesis filtering approach," *IEEE Transactions on Signal Processing*, Vol. 43, No. 11, pp. 2496 - 2504, 1995.

[32] J. C. Anderson, "A spectral magnitude analysis theorem and applications," *Proceedings of the IEEE-SP International Symposium on Time-Frequency and Time-Scale Analysis*, pp. 277 - 280, 1992.

[33] F. Hlawatsch, G. F. Boudreaux-Bartels, "Linear and quadratic time-frequency representations," *IEEE Signal Processing Magazine*, Vol. 9, No. 2, pp. 21 - 67, 1992.

[34] D. L. Jones, R. G. Baraniuk, "An adaptive optimal-kernel time-frequency representation," *IEEE Transactions on Signal Processing*, Vol. 43, No. 10, pp. 2361 - 2371, 1995.

[35] R. N. Czerwinski, D. L. Jones, "Adaptive short-time Fourier analysis," *IEEE Signal Processing Letters*, Vol. 4, No. 2, pp. 42 - 45, 1997.

[36] N. Sevastopoulos, R. Markell, (1988). *A Simple Method of Designing Multiple Order All Pole Bandpass Filters by Cascading 2nd Order Sections*, Linear Technology, Application note 27A, June 1988, [Online] Available: http://cds.linear.com/docs/Application_Note/an27af.pdf

[37] P.Schniter, *Noble identities*, [Online]. Available: http://cnx.org/content/m10432/latest/#nobleidentiesfig2

[38] J.-H. Lee, W.J. Kang, "Designing filters for polyphase filter banks," *Circuits, Devices and Systems, IEE Proceedings G* , vol.139, no.3, pp. 363 - 369, June 1992.

[39] S. Winder, *Analog and Digital Filter Design*, Newnes, 2002.

[40] http://h2physics.org/?cat=49

[41] C. Burrus, *The Cooley-Tukey fast Fourier transform algorithm*, [Online]. Available: http://cnx.org/content/m16334/latest/

[42] "Diode Bridge", [Online]. Available: http://en.wikipedia.org/wiki/Diode_bridge

[43] "Biquad Filter," *Mathworks Documentation Center*, DSP system toolbox, [Online]. Available: http://www.mathworks.com/help/dsp/ref/biquadfilter.html

[44] M. Portnoff, "Time-frequency representation of digital signals and systems based on short-time Fourier analysis," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 28, pp. 55-69, 1980.

[45] P. P. Vaidyanathan, "Multirate digital filters, filter banks, polyphase networks, and applications: a tutorial," *Proceedings of the IEEE*, vol. 78, pp. 56-93, 1990.

APPENDIX A

MATLAB CODE:

DISCRETE_WIGNER.M, DFT.M, HETERODYNING.M, STFT.M, SUBBANDS.M, CROSSCORR.M

This appendix contains the code for six MATLAB functions that were specifically written as part of this work. These functions are:

(a) Discrete_Wigner.m: computes the discrete Wigner distribution of an analytic signal. The function includes information to define a signal consisting of two linear frequency-modulated chirps with chirp rates given by assigning start and end frequencies "k0" and "k1". The function is also used to produce a signal consisting of 4 cosinusoids with amplitudes of 1, $\sqrt{2}$, $\sqrt{3}$, and 2; and with frequencies given by k, $1.5k$, $2k$, and $2.5k$, and with a data record length of $N = 800$.

(b) DFT.m: computes the unscaled discrete Fourier transform of a signal.

(c) Heterodyning.m: computes the DFT of a prototype time window and 4 modulated versions of the window, to observe the resulting effect on the filter passbands in a filter bank implementation of the STFT.

(d) STFT.m: computes the short-time Fourier transform of a signal consisting of a stationary sinusoid, using a rectangular time window of length 100. By commenting the signal definition, the function can be applied to any signal previously defined as "x".

(e) Subbands.m: generates a TFR using the filter bank approach described in the thesis. The process operates on a signal previously defined as "x", with sampling rate "fs". The number of filters used in the analysis can be altered by changing the definition of "N", and the bandwidth of the filters is automatically altered to distribute the frequency space from DC to half the sampling rate into N non-overlapping filters. Butterworth filters of order 16 are designed within a filtering loop, using the "butter" command from the signal processing toolbox. The filters are converted from [z,p,k] syntax to cascaded second-order-sections, and then to digital filter objects, whereupon they are used to filter the analysis signal. The filtered outputs are rectified and smoothed prior to being placed into a matrix at a position given by the filter number. Finally, a pseudo-reconstruction is generated, called "song" for use in the error minimization process.

(f) crosscorr.m: computes the cross correlation of a signal and its reconstruction from "Subbands.m", and finds the peak index for use in the phase-correction portion of the error minimization process.

```
%Discrete_Wigner.m
close all; clear all;
N=800;n=[0:N-1]; % signal discrete time index vector
k0=50;k1=50; % discrete frequencies for chirps
slope=(k1-k0)/(2*N);k=slope.*n+k0; % unconventional way of ...
    defining a chirp
% four analytic cosines with different frequencies and amplitudes
x=exp(j*k/N*2*pi.*n)+1.414*exp(j*k/N*3*pi.*n)+...
sqrt(3)*exp(j*k/N*4*pi.*n)+2*exp(j*k/N*5*pi.*n);
W=zeros(N/2,N);ACS=zeros(N/2,N/2); % initialize matrices
m=[-N/4:N/4-1]; % discrete delay vector
kk=[0:N/2]; % discrete frequency vector
pad=zeros(1,N/2);x=[pad x pad];  % pad for circshift
% compute complex non-stationary ACS
for i=1:N/2
        temp=circshift(conj(x),[0,m(i)]).*circshift(x,[0,-m(i)]);
    ACS(i,:)=temp(N/2+1:2:3*N/2)/((N/2)^2)*4/3; % a factor of ...
        4/3 is needed for proper normalization, why?
end
eee=exp(-j*4*pi*kk'*m/N);  % matrix of complex exponentials for DFT
W=eee*ACS; % compute WD
imagesc([0 N],[0 N/2],abs(W));axis xy; % plot WD
stationary_ACS=sum(ACS'); %stationary ACS is line integrated ...
    non-stationary ACS
psd=zeros(1,N/2); % initialize psd vector
psd=stationary_ACS*eee'; % compute PSD
figure(2); plot(abs(psd)); %plot PSD
```

```
% DFTm
h=[1 1 1 1 1];
m=[0:length(h)-1];
N=500;
l=[-N/2:N/2];
e=exp(-j*2*pi/N*m'*l);
H=h*e;
plot(real(H))
```

```
% Heterodyning.m
% This code assumes that DFT.m has previously been executed.
% It then heterodynes the DFT 4 times, plotting each new DFT
% in a different color.
hold on
color=['r','g','c','m'];
for k=1:4
    h_BP=h.*exp(j*2*pi/N*100*k.*m); % up convert the window
    H_BP=h_BP*e;  % compute its DFT
    plot(abs(H_BP),color(k));
end
```

```
% STFT.m
% Compute STFT of a stationary sinusoid and plot it.
N=1000;n=[0:N-1]; % duration, time index vector
l=[0:N/2]; % discrete frequency vector
k=50;f0=k/N; % sinusoidal frequency
f1=1*f0; % change multiplier for linear chirp
f=(f1-f0)/N*n+f0; % frequency vector
x=sin(2*pi()*f.*n); % input sinusoidal signal
h=ones(1,100); % rectangular time window
pad=zeros(1,N-length(h)); h=[h pad]; % zero padding for window
e=exp(-j*2*pi/N*n'*l); % matrix of complex exponentials for DFT
%initialize STFT matrices:
x1=zeros(N,N); xe=zeros(N,N/2+1);h1=zeros(N,N);
for o=1:N % start STFT loop
   h1(o,:)=circshift(h,[1,o-1]);  % move time window along signal
   x1(o,:)=x.*h1(o,:); % window the signal
   xe(o,:)=x1(o,:)*e; % compute DFT of windowed signal
end
imagesc(abs(xe')) %plot
```

```
%Subbands.m
N=512; W=fs/(2*N);  % number and bandwidth of sub-bands...
% ... fs is the sampling frequency of the input signal x
% define the center frequencies of the sub-bands:
freq_bands=linspace(0,fs/2-W,N);
L=length(x);
Ts=1/fs; t=(0:Ts:L*Ts-Ts); %sampling period and time vector
% initialise outputs
 output=zeros(L,N); song=zeros(1,L);
% Sub-band filtering loop
for kk=1:N
  % generate and apply band-pass filter
  % use zero-pole design:
  [z,p,k]=butter(16,[2*freq_bands(kk)/fs ...
     2*(freq_bands(kk)+W)/fs],'bandpass');
  % convert to second-order sections (see chapter 6):
  [sos,g]=zp2sos(z,p,k);
  h1=dfilt.df2sos(sos,g); % make digital filter object
  y=filter(h1,x); %filter the signal with it
  % rectify and low-pass with smoothing filter in the same fashion
  [smooth_z,smooth_p,smooth_k]=butter(8,2*freq_bands(kk)/fs,'low');
  [smooth_sos,smooth_g]=zp2sos(smooth_z,smooth_p,smooth_k);
  Hsmoothing=dfilt.df2sos(smoothing_sos,smoothing_g);
  %smooth the rectified signal:
  smoothed_y=filter(Hsmoothing,abs(y));
  % store it according to sub-band center frequency :
  output(:,kk)=smoothed_y;
  %resynthesize:
  song=song+smoothed_y'.*real(exp(j*2*pi*freq_bands(kk).*t));
end
```

```
% crosscorr.m
cross = zeros(1,length(x)); % initialize cross correlation vector
for gg = 1:length(x)}
  % matrix multiply signal with sliding reconstruction:
  cross(gg) = x*(circshift(xreconstructed,[0,-gg])');
end
% find index of max value in the cross correlation:
[max_val,shift]=max(cross)
```