FFIE/711/116

WAVELETS AND THEIR USE IN THE DETECTION AND CHARACTERIZATION OF TRANSIENT SIGNALS

URNES Emil

FFI/RAPPORT-97/03066

WAVELETS AND THEIR USE IN THE DETECTION
AND CHARACTERIZATION OF TRANSIENT SIGNALS

URNES Emil

FFIRAPPORT-97/03066

FORSVARETS FORSKNINGSINSTITUTT (FFI)

Norwegian Defence Research Establishment

P O BOX 25
N–2007 KJELLER, NORWAY

REPORT DOCUMENTATION PAGE

1) PUBL/REPORT NUMBER FFI/RAPPORT–97/03066	2) SECURITY CLASSIFICATION UNCLASSIFIED	3) NUMBER OF PAGES
1a) PROJECT REFERENCE FFIE/711/116	2a) DECLASSIFICATION/DOWNGRADING SCHEDULE	161

4) TITLE

WAVELETS AND THEIR USE IN THE DETECTION AND CHARACTERIZATION OF TRANSIENT SIGNALS

(WAVELETS OG DERES ANVENDELSE FOR DETEKSJON OG KARAKTERISERING AV TRANSIENTE SIGNALER)

5) NAMES OF AUTHOR(S) IN FULL (surname first)

URNES Emil

6) DISTRIBUTION STATEMENT

Approved for public release. Distribution unlimited
(Offentlig tilgjengelig)

7) INDEXING TERMS

IN ENGLISH:	IN NORWEGIAN:
a) wavelet transforms	a) wavelet–transformer
b) matched filters	b) matchede filtre
c) transients	c) transienter
d) detectors	d) detektorer
e) strain sensors	e) tøyningssensorer

THESAURUS REFERENCE: INSPEC Thesaurus 1995

8) ABSTRACT

This work was done as a diploma thesis at Forsvarets forskningsinstitutt during the spring of 1997. It represents a self contained mathematical foundation for the detection and characterization of transients using wavelet transforms.

Selected techniques, most of them wavelet based, are demonstrated using experimental data from tests at sea under project 711 CHESS. A matched filter based on orthonormal wavelet transforms is implemented. By basing the matched filter on only a few relevant wavelet transform coefficients, several advantages arise, the most important being a faster algorithm, reduced storage needs and a better separation of events occurring at different scales than in the case of a traditional matched filter.

9) DATE	AUTHORIZED BY This page only	POSITION
25 June 1997	*P Narum* Paul Narum	Director of research

ISBN 82-464-0168-8

FFI–B–22–1982

FORSVARETS FORSKNINGSINSTITUTT (FFI)
Norwegian Defence Research Establishment
P O BOX 25
N-2007 KJELLER, NORWAY

REPORT DOCUMENTATION PAGE

4) TITLE

WAVELETS AND THEIR USE IN THE DETECTION AND CHARACTERIZATION OF TRANSIENT
SIGNALS
(WAVELETS OG DERES ANVENDELSE FOR DETEKSJON OG KARAKTERISERING AV TRANS-
IENTE SIGNALER)

5) NAMES OF AUTHOR(S) IN FULL (surname first)

URNES Emil

6) DISTRIBUTION STATEMENT

Approved for public release. Distribution unlimited
(Offentlig tilgjengelig)

7) INDEXING TERMS

IN ENGLISH:		IN NORWEGIAN:	
a)	wavelet transforms	a)	wavelet-transformer
b)	matched filters	b)	matchede filtre
c)	transients	c)	transienter
d)	detectors	d)	detektorer
e)	atom search	e)	wavelet-søkemotor

THESAURUS REFERENCE: INSPEC Thesaurus 1995

8) ABSTRACT

This work was done as a diploma thesis at Forsvarets forskningsinstitutt during the spring of 1997. It repre-
sents a self contained mathematical foundation for the detection and characterization of transients using wave-
let transforms.

Selected techniques, most of them wavelet based, are demonstrated using experimental data from tests at sea
under project 711 CHESS. A matched filter based on orthonormal wavelet transforms is implemented. By bas-
ing the matched filter on only a few relevant wavelet transform coefficients, several advantages arise, the most
important being a faster algorithm, reduced storage needs and a better separation of events occurring at differ-
ent scales than in the case of a traditional matched filter.

9) DATE	AUTHORIZED BY	POSITION
	This page only	
25 June 1997	Paul Narum	Director of Research

UNCLASSIFIED

3

CONTENTS

Page

CONTENTS

WAVELETS AND THEIR USE IN THE DETECTION AND CHARACTERIZATION OF TRANSIENT SIGNALS

1 INTRODUCTION

This thesis study has been made in connection with the development project CHESS (Composite Hull Embedded Sensor System), which is a collaboration between the Norwegian Defense Research Establishment and Naval Research Laboratory in Washington. The objective of the project is to develop an experimental hull load monitoring system based on fibre optic Bragg grating sensor technology with real time signal processing. The sensor system will be mounted on a SES (Surface Effect Ship) fast patrol boat which is going to be built for the Norwegian Navy by Kværner Mandal. Experimental evaluation of the system will be made during operation at sea.

The sensor system will measure the local strain at selected locations in the hull, which is built from glass fibre reinforced composite materials. At high speeds, wave impacts (slamming) and movement of the ship will induce global and local transient oscillations in the hull and shock waves propagating through the structure. In addition there will be a number of other loads, like pressure oscillations in the air cushion and vibrations generated by the propulsion machinery. The effect of loads and the dynamical behaviour of the hull can in principle be identified and measured by a careful analysis of the strain data.

The signal processing system of the hull monitoring system is planned to perform a number of tasks: digital storage of the signal output from the sensors, real time signal processing of the output for identification and characterization of strain transients, automatic detection of damages to the hull, calculation of dynamical deformation of the hull, and real time presentation of selected results during operation at sea.

In this thesis, we fill focus on wavelet based techniques for the detection and characterization of transient oscillations resulting from wave impacts. A number of techniques have been developed to investigate and characterize oscillatory transient phenomena in mechanical structures. Short-time Fourier transforms and joint time-frequency analysis are frequently used (L. Cohen, 1995; Friedlander & Porat, 1989) and might be useful tools in the present project. However, during the last decade there has been an accelerating research effort to utilize wavelet transforms in signal processing. The concept of wavelets is new. It was introduced by J. Morlet in the late 1970's as a tool for analyzing seismic data. Wavelets are

currently used in data compression, de-noising, detection and characterization of signals, the solution of partial differential equations, fractals, and in many other areas. A review of wavelet applications is given in Kovacevic and Daubechies (1996).

The invention of the multiresolution framework by Mallat (1989) and Meyer (1986), made a link between the elegant mathematics of wavelet theory and the signal processing theory of filter banks and subband coding. Signal processing methods based on wavelets are especially well suited in the study of non-stationary signal and noise processes and transient phenomena covering a large range of frequencies or scales. The strain sensor output contains interesting components scattered over a broad frequency range. Different frequency bands can be analyzed separately, making it easier to separate phenomena occurring at different scales (or frequencies). The (discrete) wavelet transform has a fast implementation through iterated filter banks, making it attractive from a practical viewpoint. In light of the above arguments, it was decided at an early stage of the CHESS project to investigate the application of wavelet transforms, and a preliminary study was made by Eriksen (1996). In this work, Eriksen presents the idea of performing matched filtering in the wavelet domain.

The general purpose of this work is to carry on the study of transient detection by means of wavelets and to develop a self contained theoretical foundation for a practical implementation. The work will cover the following:

- Theory of wavelet transforms for application in signal processing. Special focus will be made on the tools needed in transient detection, the most important of these tools probably being the time invariant transform.

- A short overview of existing literature on methods for transient detection using wavelets.

- A presentation of techniques for transient detection and characterization and how these can be implemented with the help of wavelets.

- Numerical experiments illustrating the above techniques, using experimental data from tests at sea. One purpose of these experiments is to get a background for selecting concrete methods which can later be used in a comprehensive study of a larger experimental data set. Efficient codes for real time processing will be developed at a later stage in the project and are outside the scope of this work.

The structure of this thesis is as follows. In chapter 2, the topic of structure monitoring is addressed, and the particular sensor system used in the monitoring of our vessel is discussed. Plots of typical transients in a data set obtained from real

measurements on a full-size boat operating at various speeds and sea conditions are presented.

Chapter 3 contains the basic theory on wavelets. It begins with the concept of frames, which will be an important link to chapter 4. The main topic of chapter 3 is discrete wavelet transforms in the multiresolution analysis framework. The dyadic orthonormal multiresolution formulation is presented, along with various extensions of the basic scheme.

Chapter 4 treats the problem of detecting and characterizing transients. We will make the assumption of a white Gaussian noise background. Emphasis is put on the matched filter detector. In white noise, this detector is equivalent to a correlator. The matched filter assumes that the form of the transient we are looking for is known in advance. It is usually implemented in the time domain. We show that in tight frames, matched filters can equivalently be implemented in the transformed domain. By basing the matched filter on only a few relevant wavelet transform subbands, several advantages arise, the most important being an increase in the speed of the algorithm and a better separation of events occurring at different scales than in the case of a traditional matched filter. Methods that require less advance knowledge about the transient are also presented in this chapter. One of these methods is using the energy value of the maximum transform coefficient. The methods are demonstrated on experimental data from tests at sea. The numerical experiments were performed using Matlab and the Wavelet Toolbox in Matlab. Some of the results of the experiments were put in chapter 4 to illustrate the theory. Because there are quite many figures, many of them were put in a separate chapter, namely chapter 5. Some were also put in the appendix. In chapter 6 conclusions are presented, and suggestions for further work are pointed out.

The reader is assumed to have some basic knowledge of Hilbert space theory, measure theory, Fourier analysis and the theory of digital sampling. For readers lacking some of that knowledge, a short presentation is given in appendix A, along with references to more literature. Appendix A also explains the notation used in this thesis and gives a list of symbols and abbreviations. The proof of two of the theorems of chapter 3 are given in appendix B, while appendix C contains some of the experimental results that did not fit into chapter 5. Appendix D contains plots of members of the Daubechies, Coiflet and Symlet wavelet families, and appendix E is a printout of the Matlab programs that were developed during the work on this thesis.

2 STRUCTURE MONITORING

Constructions and machinery are subject to mechanical loads and damages. Structure monitoring concerns the monitoring and surveillance of such systems. Important aspects of structure monitoring are:

- Loads. A key issue in structure monitoring is obtaining measurements of the strains imposed by the loads on the system. The measurements can be stored and/or processed real time.

- Verification. By observing the loads, we can verify that the system behaves as planned.

- Damage. An important issue is the detection of a possible machinery defect. Early detection of such a defect can be extremely important, since it may prevent extensive damage later.

- Statistics and logging. The loads and damages on the system can be stored and logged for later analysis. This analysis might reveal important characteristics about our system. After a breakdown, the logged data can be analyzed in order to determine the causes of the breakdown.

In our case, we are concerned with the structure monitoring of ships. The development of fibre optic Bragg grating (FBG) strain sensors has opened the possibility for structure monitoring of hulls (Kersey et al.; Pran, 1995). The sensors can be embedded in the hull and be a part of the structure. The purpose of the sensors is to measure the mechanical strains in the hull. As waves hit the hull, damped oscillations, or transients, in the structure will occur. Both the entire hull and the local panels that the hull consists of will vibrate. The amplitude of these oscillations gives us information about the size of the wave that hit the hull. From the FBG measurements, we can obtain estimates of the amplitudes and arrival times of transients, which is an indication on the stress the hull is exposed to.

The hull of our vessel will mainly consist of composite materials, i.e. glass fibre reinforced polyester. The main damage mechanism of this material is delamination where the fibre reinforced skin is detached from the foam core. Delamination can have serious consequences for the boat, so early detection of delamination is important. Automatic detection of delaminations should be a part of the real time structure monitoring system. When a wave hits the hull, the local panels will start vibrating at a frequency given by their geometry. In addition, several higher harmonics will be excited. Also, the fundamental and higher frequencies are subject to a certain damping. When delamination occurs, the oscillation frequencies will change. Experiments show that the higher harmonics are subject to the greatest change (Jensen, 1997). The estimation of the oscillation frequencies
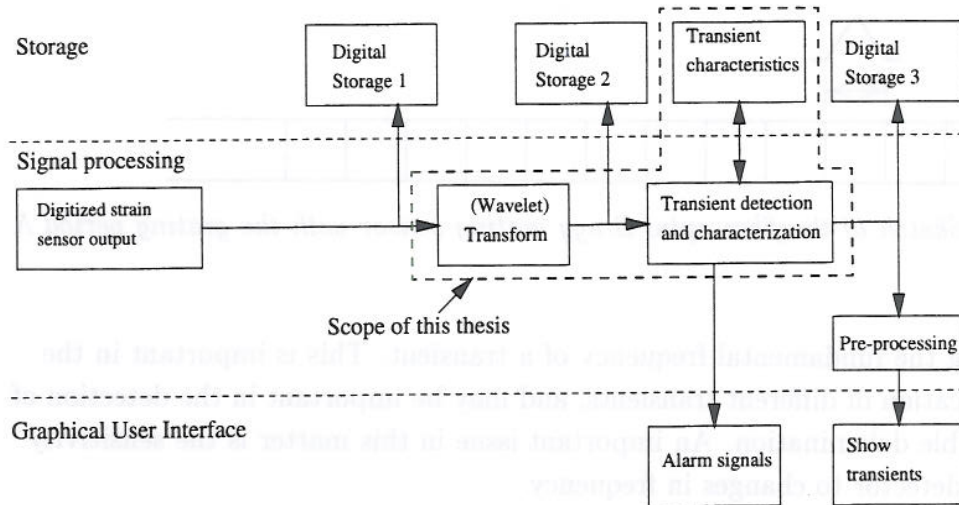
Figure 2.1: Overview of proposed real time structure monitoring system, with an indication of the scope of this thesis

in a transient is therefore an important issue in this type of structure monitoring. The estimation of damping coefficients may also be of importance.

This thesis is a step on the way to a final, operational structure monitoring system. An overview of a wavelet based structure monitoring system is given in figure 2.1. Sensors are mounted on the hull, and the output from these sensors are sampled and fed to a computer. The computer performs a wavelet transform on these data and analyzes the data in order to detect and characterize transients. Some prior knowledge of the transients is required and must be built into the system. The system gives real time feedback to an operator on the current load the vessel is exposed to, i.e. the amplitude of the transients, and it gives instant warning messages if changes in transient characteristics, possibly due to delamination, occur. In addition, both the input signal, the wavelet transform and the detector output can be stored on digital tape for later analysis. Since the wavelet transform is invertible, the digital storage 1 of figure 2.1 might be unnecessary. In this thesis, focus is on the wavelet transform and transient detection and characterization algorithms. Data used in this thesis is processed a posteriori, not real time.

The most important problems that need to be solved by the signal processing system in our particular structure monitoring system are:

- Detection of the transients that give high strains on the hull. These transients are also the easiest to detect, because high-energy transients yields a high signal-to-noise ratio. The sensitivity of the detector is not very important in this case.

- Finding estimates for the amplitudes of the transients. The logging of transients gives an indication on the load the hull has been exposed to during a ride. For low amplitudes, the sensitivity of the detector is important.
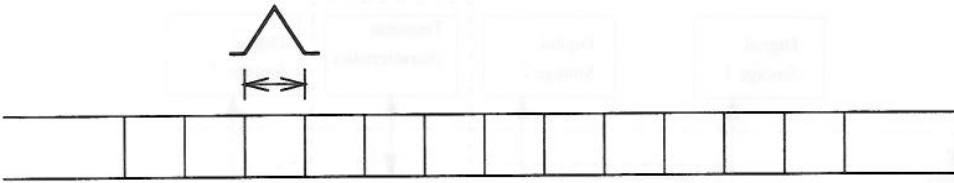
Figure 2.2: Sketch of the fibre optic Bragg grating sensor with the grating period Λ indicated

- Finding the fundamental frequency of a transient. This is important in the classification of different transients, and may be important in the detection of a possible delamination. An important issue in this matter is the sensitivity of the detector to changes in frequency.

- Detecting changes in the higher harmonics of a transient. In the early detection of delamination, these changes are especially important. These higher harmonics will be very weak, and they therefore require high sensitivity in the detectors.

2.1 Fibre optic Bragg grating sensor

This section contains a brief description of the principle of the fibre optic Bragg grating sensors used for detecting mechanical strains. The sensor is an optical fibre with a grating in the core. The grating has a period Λ, as shown in figure 2.2. When broadband illumination is sent into the fibre, the grating structure will reflect light at a wavelength λ according to the Bragg condition (Pran, 1995),

$$\lambda = 2n\Lambda, \tag{2.1}$$

where n is the refractive index of the fibre material. Thus we have a linear relationship between the wavelength λ of the reflected light and the grating period Λ.

The sensor is mounted on a panel. When that panel is strained, the sensor will experience the same strain. There is a linear relationship between the axial strain on the optical fibre and the grating period. The quantity we measure is λ, the wavelength of reflected light. If we denote λ_0 as the reflected wavelength when the fibre does not experience a strain, the quantity $\lambda - \lambda_0$ is proportional to the longitudinal strain in the fibre, and therefore the panel.

The optical fibre can contain more than one fibre optic Bragg grating element. Several sensors can be combined into an array, and the output must be de-multiplexed to determine which wavelength came from which sensor. Also, several fibres can be mounted in the hull, so we can measure the strain at several

locations of the hull, and in several directions along the panels. More material on fibre optic Bragg grating sensors can be found in Pran (1995) and Kersey et al.

2.2 The experimental data

Real data from a full size boat has already been obtained. The boat was equipped with four fibre optic sensors and was operated at various speeds and sea conditions. The data are available on digital tape at NDRE.
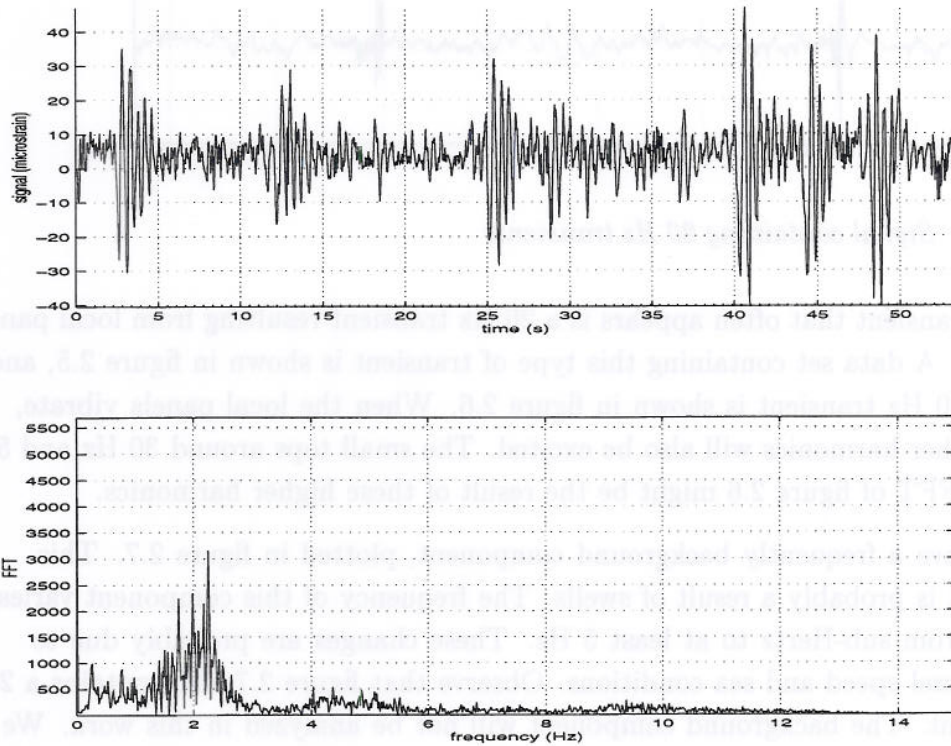

Figure 2.3: Signal containing 2-2.5 Hz transients, along with its FFT
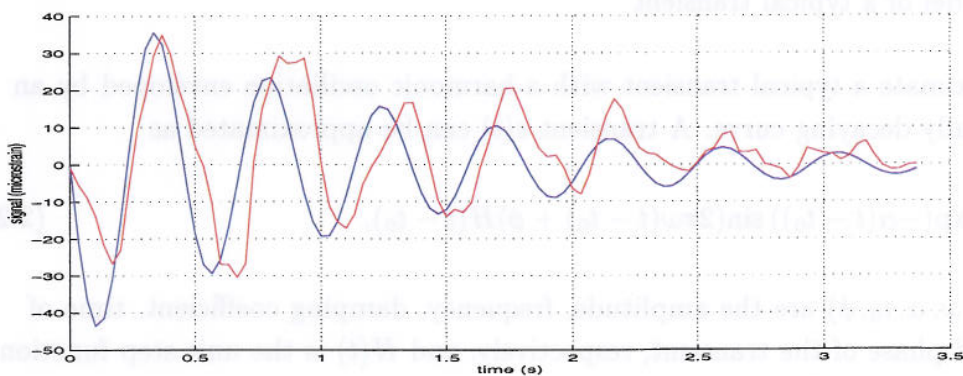

Figure 2.4: Typical 2-2.5 Hz transient (red) and modeled transient (blue)

At least two transient types are present in the data. The most common transient

is a 2-2.5 Hz transient resulting from vibrations in the hull as a whole. A data set containing this type of transient is shown in figure 2.3. This data set will frequently be used in our analysis. One of the transients is shown in figure 2.4. This transient will often be used as a reference transient.
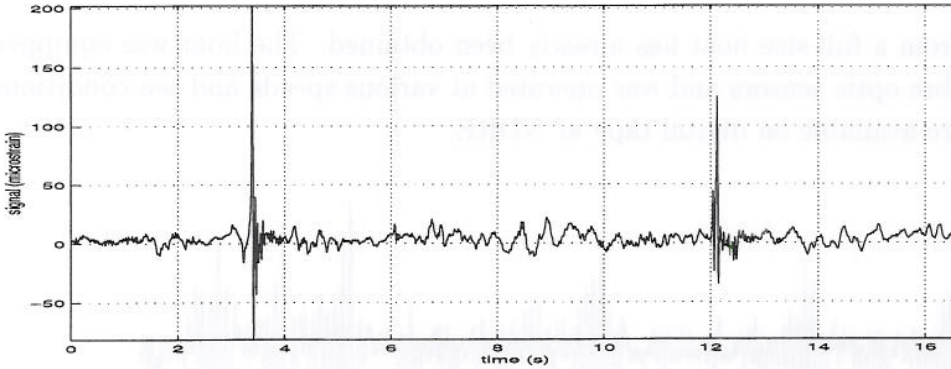

Figure 2.5: Signal containing 20 Hz transients

Another transient that often appears is a 20 Hz transient resulting from local panel vibrations. A data set containing this type of transient is shown in figure 2.5, and a typical 20 Hz transient is shown in figure 2.6. When the local panels vibrate, several higher harmonics will also be excited. The small tops around 30 Hz and 50 Hz in the FFT of figure 2.6 might be the result of these higher harmonics.

We also have a frequently background component, plotted in figure 2.7. This component is probably a result of swells. The frequency of this component varies; it ranges from sub-Hertz to at least 5 Hz. These changes are probably due to varying vessel speed and sea conditions. Observe that figure 2.7 also contains a 20 Hz transient. The background component will not be analyzed in this work. We will focus on the transient phenomena of figure 2.4 and 2.6.

2.2.1 Model of a typical transient

We approximate a typical transient with a harmonic oscillation enveloped by an exponentially decaying curve. A transient $s(t)$ can be approximated as

$$s(t) = A \exp(-\alpha(t - t_0)) \sin(2\pi\omega(t - t_0) + \phi) H(t - t_0), \qquad (2.2)$$

where $(A, \omega, \alpha, t_0, \phi)$ are the amplitude, frequency, damping coefficient, time of arrival and phase of the transient, respectively, and $H(t)$ is the unit step function. We will assume $\phi = 0$.
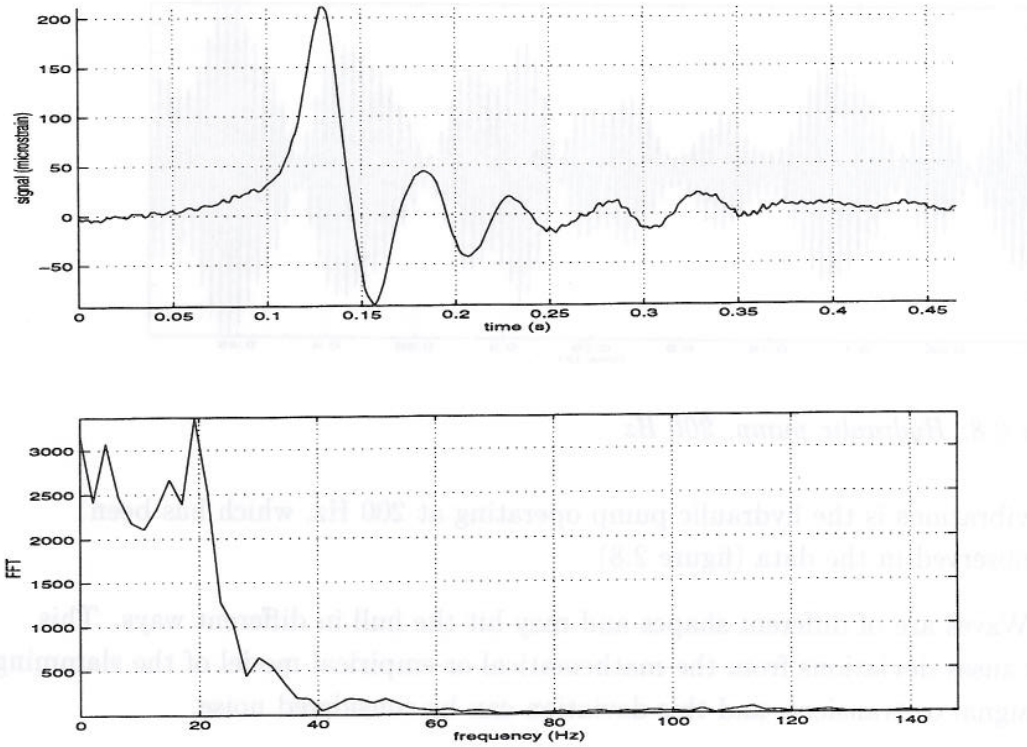
Figure 2.6: Typical 20 Hz transient, along with its FFT
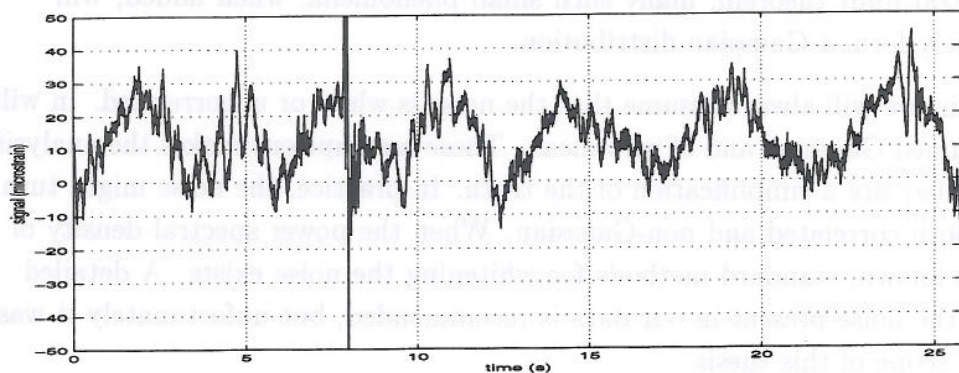
Figure 2.7: Background component at 0.2-0.3 Hz

2.2.2 Noise analysis

The sensor output will be considered a sum of signal and noise. In other words, the noise includes every phenomenon that we do not consider a signal. Possible sources of noise are

- Random errors in sensor output.

- Quantization errors in the digitalization of the analog sensor data.

- Panel vibrations that do not come from wave slammings. One source of such

Figure 2.8: Hydraulic pump, 200 Hz

vibrations is the hydraulic pump operating at 200 Hz, which has been observed in the data (figure 2.8).

- Waves are of different shapes and may hit the hull in different ways. This causes deviations from the mathematical or empirical model of the slamming signal or transient, and this deviation can be considered noise.

- Demodulation of the sensor data.

By the central limit theorem, many such small phenomena, when added, will approximately form a Gaussian distribution.

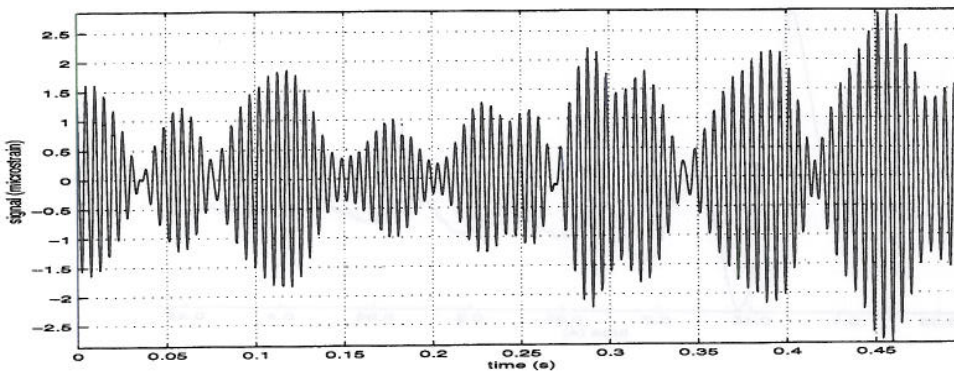In this thesis, we will always assume that the noise is white or uncorrelated. In will also be assumed Gaussian and of zero mean. These assumptions makes the analysis easier, but they are a simplification of the truth. In practice, the noise might turn out to be both correlated and non-Gaussian. When the power spectral density of the noise is known, standard methods for whitening the noise exists. A detailed analysis of the noise present in our data is recommended, but unfortunately it was beyond the scope of this thesis.

3 WAVELET THEORY

This chapter contains the basic theory on wavelets. A *wavelet* ψ is an oscillating function, or a 'small wave', satisfying the property $\int \psi(t)dt = 0$. It is usually local in nature; either it has compact support, or it decays rapidly to zero as $|t| \to \infty$. The wavelet ψ, which we often call the *mother* wavelet, gives rise to a family of wavelets obtained by translating and scaling the mother wavelet. A *wavelet transform* of a function f is a decomposition of f into an infinite sum (or integral) of such wavelets.

The structure of the chapter is as follows. It begins with an overview of *time frequency analysis*. Time frequency analysis is a technique for analyzing signals

both in time and frequency simultaneously. Both the wavelet transform and the *short time Fourier transform* are examples of such techniques. The concept of *frames* are introduced next. Frames is a general concept regarding the decomposition of signals in Hilbert spaces. Both the short time Fourier transform and the wavelet transform are special cases of frames. Frames are particularly interesting when *discretizing* the wavelet transform. For *tight* frames, we have a *Parseval's* theorem. This theorem will be useful in the chapter on detection.

One of the main motivations for discretizing the wavelet transform is that we obtain a *fast algorithm*, as described in the chapter on *multiresolution analysis*. This algorithm is important for practical signal processing applications, where we work on discrete signals. The discrete wavelet transform of a signal of N samples can be calculated in $O(N)$ time. This is faster than the FFT, which takes $O(N \log N)$ time. The fast algorithm is based on the fact that certain types of wavelet transforms can be implemented as *iterated filter banks*. Design of the wavelets is closely related to the design of such filter banks. Wavelet design is briefly discussed in section 3.6, along with a review of the most frequently used wavelets.

The fast algorithm as presented in the section on multiresolution analysis is based on *orthonormal* wavelets using a *dyadic* discretization of scale. Generalizations of the basic scheme are discussed in section 3.7. A well known problem of the standard discrete wavelet transform is that it is not *translation invariant*. Methods for dealing with this problem are reviewed in section 3.8.

In order to make the theory as accessible as possible, I have chosen to merely state the key theorems concerning wavelets. Proofs will either be given in appendix B, or an appropriate reference will be given.

3.1 Time Frequency Analysis

By taking a Fourier transform, one can characterize a signal in the frequency domain. A Fourier transform is a decomposition of a signal into sinusoids, or complex exponentials. These functions are sharply localized in frequency, but poorly localized in time. Thus a Fourier transform can be useful for characterizing the spectral contents of a *stationary* signal. However, if the signal varies with time, it is hard to extract this time dependence from the Fourier transform. For this we need a time-frequency transform; a transform that describes our signal both in time and frequency simultaneously.

Several methods for doing this have been proposed, both linear and non-linear. The most well known technique is probably the short-time Fourier transform, which is a localized form of the Fourier transform. In the past 5-10 years, the wavelet transform has gained popularity as another powerful tool for time-frequency

decompositions. Both the short-time Fourier transform and the wavelet transform are linear. Nonlinear transforms also exist. The Wigner distribution is the most popular among these; see L. Cohen (1995).

3.1.1 Heisenberg's uncertainty principle

The Heisenberg uncertainty principle is best known from quantum mechanics, where it states that the position and the velocity of a particle cannot both be determined with arbitrary accuracy. A similar statement holds for the two variables time and frequency. It is impossible to determine both the time of arrival of a transient and the frequency of the transient with arbitrarily small error simultaneously. From a mathematical viewpoint, the Heisenberg uncertainty principle is a fundamental statement regarding Fourier transform pairs. A narrow waveform yields a wide frequency spectrum and a wide waveform yields a narrow spectrum, and the time waveform and frequency spectrum cannot be made arbitrarily small simultaneously.

More precisely, look at a signal $s(t) \in L^2(\mathbb{R})$. The signal has energy

$$E = \|s\|^2 = \int_{-\infty}^{\infty} |s(t)|^2 dt. \tag{3.1}$$

We can define the average time with respect to the energy density per unit time $|s(t)|^2$ as follows:

$$E[t] = \int_{-\infty}^{\infty} t|s(t)|^2 dt, \tag{3.2}$$

where $E[t]$ denotes the expectation value. Similarly we define the time standard deviation T of the signal as

$$T^2 = E[(t - \bar{t})^2] = \int_{-\infty}^{\infty} (t - \bar{t})^2 |s(t)|^2 dt. \tag{3.3}$$

T is an indication of the duration of the signal.

By Parseval's theorem, the energy of the signal can also be expressed as

$$E = \frac{1}{2\pi} \|\hat{s}\|^2 = \frac{1}{2\pi} \int_{-\infty}^{\infty} |\hat{s}(\omega)|^2 d\omega, \tag{3.4}$$

where $\hat{s}$ denotes the Fourier transform of s. Taking $|\hat{s}(\omega)|^2/2\pi$ to be the energy density per unit frequency, we can define the mean frequency $\bar{\omega}$ and bandwidth B
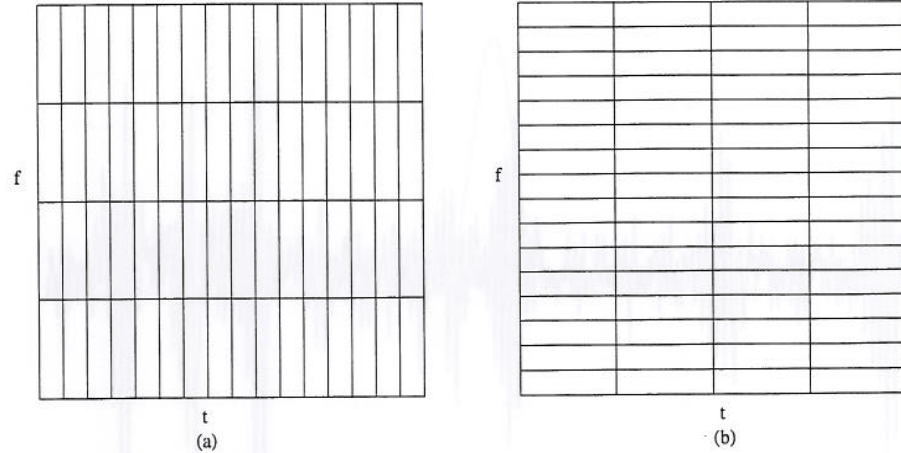
Figure 3.1: Tiling of the time-frequency plane with short-time Fourier transform basis, (a): narrow time window (b): wide time window

as follows

$$E[\omega] = \frac{1}{2\pi} \int_{-\infty}^{\infty} \omega |\hat{s}(\omega)|^2 d\omega, \tag{3.5}$$

$$B^2 = E[(\omega - \overline{\omega})^2] = \frac{1}{2\pi} \int_{-\infty}^{\infty} (\omega - \overline{\omega})^2 |\hat{s}(\omega)|^2 d\omega. \tag{3.6}$$

Theorem 1 *The uncertainty principle. We have the following restriction on the pair (T, B):*

$$TB \geq \frac{1}{2}. \tag{3.7}$$

Therefore one cannot have or construct a signal for which both T and B are arbitrarily small.

For a proof and for more information, see L. Cohen (1995). The proof utilizes the Cauchy-Schwarz inequality and the fact that T and B are connected through the Fourier transform.

3.2 The short-time Fourier transform

The short-time Fourier transform (STFT), also called the windowed Fourier transform, is obtained by first multiplying the signal f by a 'window' function h and then calculating the Fourier transform. The purpose of the window function is to make a time-localized version of f before we take the Fourier transform. The window function is typically localized around 0, so that $h(u - t)$ will be localized

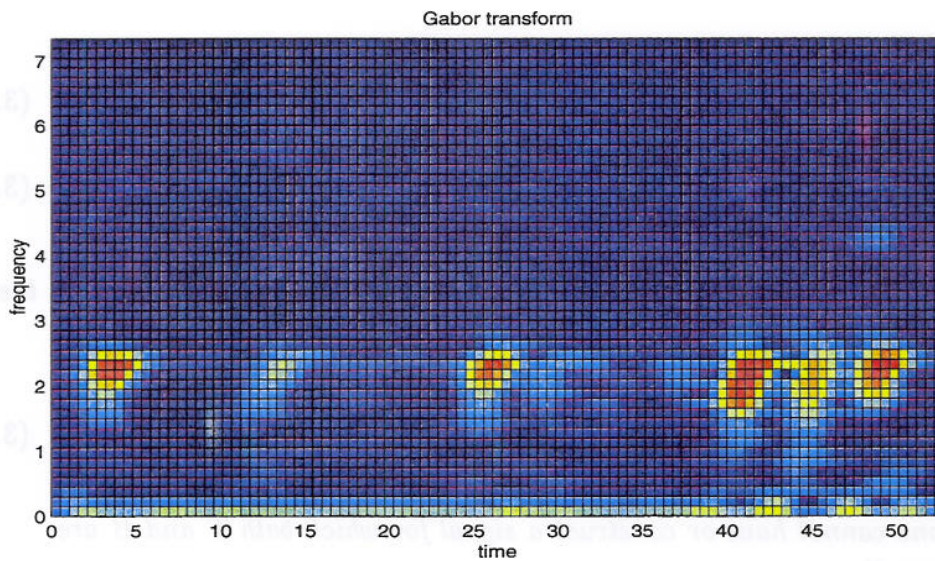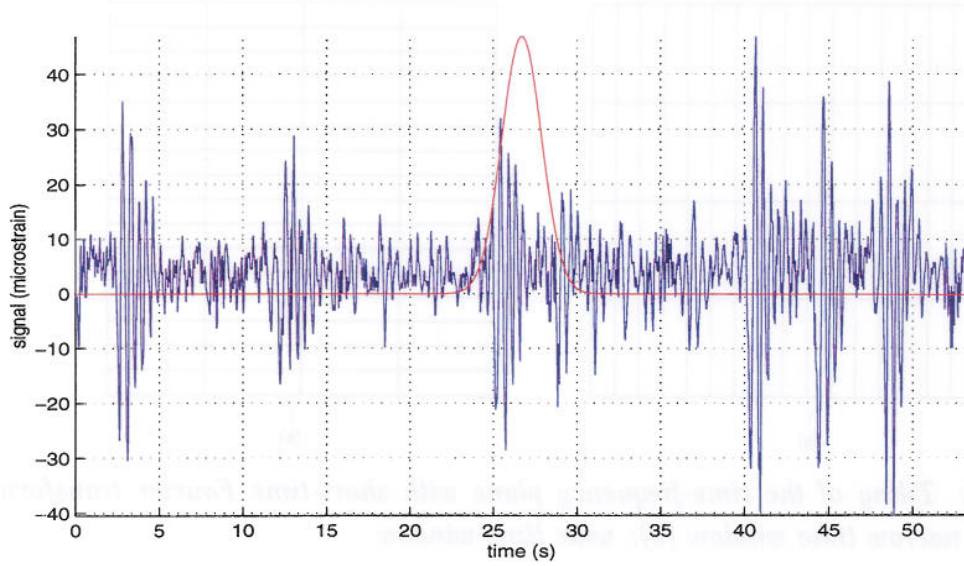*Figure 3.2: The Gabor transform of the signal in figure 2.3. The Gaussian window function is shown in red. The colour scale ranges from dark blue (min) to red (max)*

around $u = t$. In this way, we get resolution both in time and frequency. The STFT is given by

$$(T^{STFT}f)(\omega, t) = \int_{-\infty}^{\infty} f(u)h(u-t)e^{-i\omega u}du. \tag{3.8}$$

The STFT with a Gaussian window,

$$h(t) = norm(0, \sigma^2), \tag{3.9}$$

is called the Gabor transform. With a *discrete* short-time Fourier transform, the resolution in time is the same at all frequencies. Furthermore, the frequency resolution is also independent of frequency. This leads to a uniform tiling of the time-frequency plane, as in figure 3.1. An example of short-Fourier transforms is in figure 3.2. The figure shows the STFT of the signal in figure 2.3, which contains several 2 Hz transients. We see how the transients are present as high coefficient values in the time-frequency plot.

3.3 Wavelet transforms

In chapter 3 we will consider wavelet analyses on $L^2(\mathbb{R})$. Wavelet transforms in other Hilbert spaces are also possible, for example wavelet analyses on an interval.

3.3.1 The continuous wavelet transform

Consider a function $\psi \in L^2(\mathbb{R})$. Under certain restrictions, ψ will define a *mother wavelet* for our wavelet analysis. We generate a doubly-indexed family of wavelets from ψ by dilating and translating,

$$\psi_{a,b}(t) = |a|^{-1/2}\psi\left(\frac{t-b}{a}\right),\tag{3.10}$$

where $a \neq 0$ is the scale parameter and b is the translation parameter. The normalization has been chosen so that $\|\psi_{a,b}\|_{L^2(\mathbb{R})} = \|\psi\|_{L^2(\mathbb{R})}$. We define the continuous wavelet transform W with respect to this wavelet family as

$$(Wf)(a,b) = <f, \psi_{a,b}> = \int_{-\infty}^{\infty} f(t)|a|^{-1/2}\overline{\psi\left(\frac{t-b}{a}\right)}dt.\tag{3.11}$$

Daubechies (1992) shows that the continuous wavelet transform has an inverse if the mother wavelet satisfies the admissibility condition,

$$0 < C_\psi = \int_{-\infty}^{\infty} |\omega|^{-1}|\hat{\psi}(\omega)|^2 d\omega < \infty.\tag{3.12}$$

For all practical purposes, this condition is satisfied if $\hat{\psi}(0) = \int \psi(t)dt = 0$. An inverse is then given by

$$f(t) = C_\psi^{-1} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} (Wf)(a,b)\psi_{a,b}(t)\frac{da\,db}{a^2},\tag{3.13}$$

where the integral converges in the weak sense. By this we mean that taking the inner product on both sides of equation (3.13) with any $g \in L^2(\mathbb{R})$ and
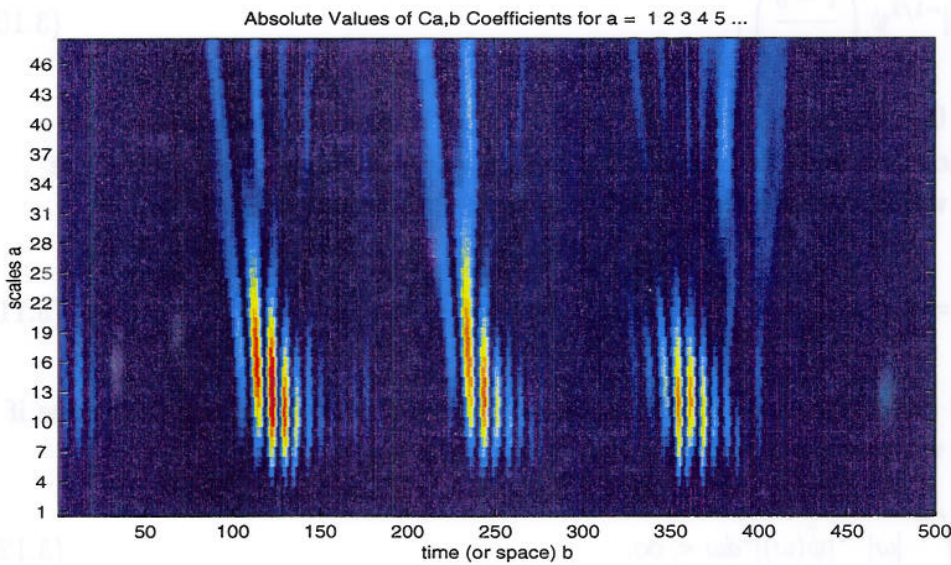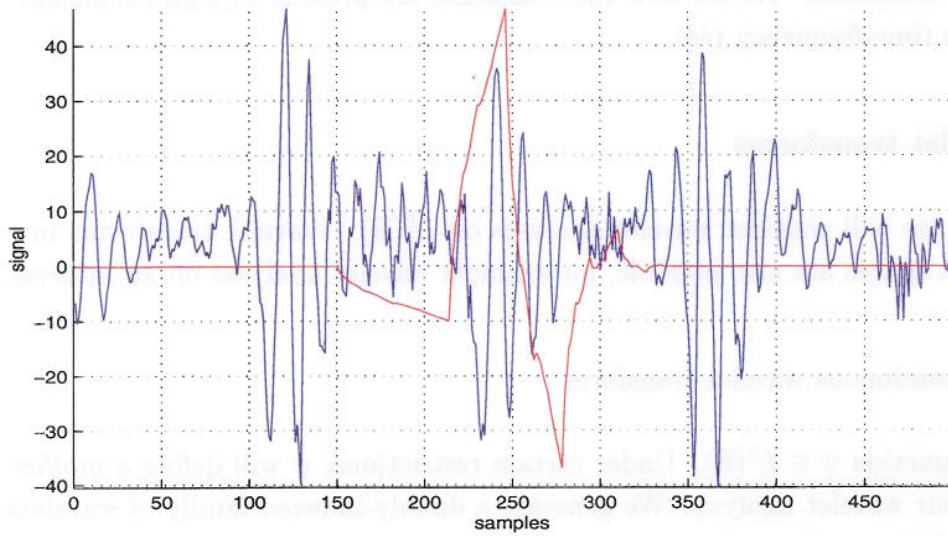
Figure 3.3: Continuous wavelet transform of a part of the signal in figure 2.3, using the Daubechies 2 wavelet (shown in red). The colour scale ranges from dark blue (min) to red (max)
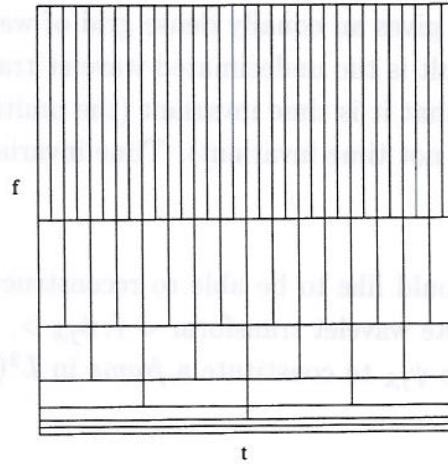
Figure 3.4: Tiling of the time-frequency plane with dyadic wavelet basis

interchanging the orders of integration leads to a true formula. An example of a continuous wavelet transform is in figure 3.2. The figure shows the STFT of a part of the signal in figure 2.3, which contains several 2 Hz transients. We see how the transients are present as high coefficient values in the time-scale plot.

3.3.2 The discrete wavelet transform

In the continuous wavelet transform, we considered a family $\psi_{a,b}(t)$ of functions (given by equation 3.10) indexed by continuous parameters $a, b \in \mathbb{R}$, $a \neq 0$. This is a highly redundant representation. We would like to restrict a, b to discrete values only, while still being able to compute an inverse transform. There are infinitely many ways of doing this, each giving rise to a particular type of discrete wavelet transform.

Fix $a_0 > 1, b_0 > 0$. A usual discretization of scale is choosing $a = a_0^j$, $j \in \mathbb{Z}$. For the discretization of translation, there are two typical choices:

- $b = kb_0 a_0^j$, $j \in \mathbb{Z}$. This option makes a denser grid of wavelets for the small scale wavelets, and a coarse grid for the large scales. We have

$$\psi_{j,k}(t) = a_0^{-j/2}\psi\left(\frac{t - kb_0 a_0^j}{a_0^j}\right) = a_0^{-j/2}\psi(a_0^{-j}t - kb_0). \tag{3.14}$$

This corresponds to the usual discrete wavelet transform in multiresolution analysis (see chapter 3). For the choice $a_0 = 2$ we obtain the dyadic, or 2-band, discrete wavelet transform. This transform gives rise to a logarithmic tiling of the time-frequency plane, as in figure 3.4.

- $b = k b_0$. This option gives an equally dense grid of wavelets no matter what the scale is. The result is the undecimated wavelet transform. The advantage of this transform is that it is time invariant (the multiresolution discrete wavelet transform is not time invariant). Time invariance is discussed in section 3.8.

For any $f \in L^2(\mathbb{R})$, we would like to be able to reconstruct f in a numerically stable way from the discrete wavelet transform $< f, \psi_{j,k} >$. It turns out that this is equivalent to requiring the $\psi_{j,k}$ to constitute a *frame* in $L^2(\mathbb{R})$.

3.4   Frames

Both the short-time Fourier transform, the continuous wavelet transform and the discrete wavelet transform are special cases of a more general concept called *frames*. Frames were introduced by Duffin and Schaeffer (1952) in the context of non-harmonic Fourier series. We will use the definition of Kaiser (1994), which is a more general definition than the classical one. The reason why we use this more general framework, is that also the continuous wavelet transform and the short-time Fourier transform fit into it nicely. This enables us to present the transform-domain matched filtering introduced in the later chapters in a more general setting.

Definition 1 *Let H be a Hilbert space and let M be a measure space with measure μ. A family of vectors $\{e_n \in H : n \in M\}$ is a called a frame in H indexed by M if*
a) For every $f \in H$, $Tf : M \to \mathbb{C}$ defined by

$$(Tf)(n) = < f, e_n >_H \tag{3.15}$$

is measurable.
b) There is a pair of constants $0 < A \le B < \infty$ such that, for all $f \in H$,

$$A\|f\|_H^2 \le \|Tf\|_{L^2(\mu)}^2 \le B\|f\|_H^2. \tag{3.16}$$

The vectors $\{e_n \in H : n \in M\}$ are called the frame vectors, (3.16) is called the frame condition and A, B are the frame bounds. The linear operator $T : H \to L^2(\mu)$ given by (3.15) is called the frame operator or the analyzing operator. The function $(Tf)(n)$ is the transform of f with respect to the frame.

Knowing the transform Tf, it would be nice to be able to reconstruct f. For frames, this is always possible, as we will now see. There exists a *dual frame* $\{\tilde{e}_n \in H : n \in M\}$ which is itself a frame with frame bounds B^{-1}, A^{-1}. The vectors

$\tilde{e}_n$ are the dual frame vectors. An explicit (although complicated!) formula for calculating the dual frame vectors exist. This is shown in Kaiser (1994). Having found the dual frame vectors, we define the synthesizing operator $S : L^2(\mu) \to H$ by

$$Sg = < g, \tilde{e} >_{L^2(\mu)} . \tag{3.17}$$

It can be shown (Kaiser, 1994) that S is a left inverse of T, i.e. $ST = I$. Thus, we can find f from Tf by the formula $f = STf$.

Many interesting results are available for frames (Kaiser, 1994). One of them is

Theorem 2 *Least-Energy Representation. Of of all possible coefficient functions $g \in L^2(\mu)$ for $f \in H$, the function $g = Tf$ is unique in that it minimizes the energy $\|g\|^2_{L^2(\mu)}$.*

3.4.1 Tight frames

In the special case of the two frame bounds being equal, $A = B$, the frame is called *tight*. For a tight frame, we have a simple formula for the dual frame vectors, $\tilde{e}_n = A^{-1}\overline{e_n}$ (Kaiser, 1994). We thus have a simple expression for the synthesizing operator,

$$Sg = A^{-1} < g, \overline{e} >_{L^2(\mu)} . \tag{3.18}$$

For a tight frame, the frame condition reduces to a Plancherel's theorem,

$$\|f\|^2_H = A^{-1}\|Tf\|^2_{L^2(\mu)}. \tag{3.19}$$

This theorem relates the energy in the time domain directly to the energy in the transform domain.

Since we work in Hilbert spaces, the Plancherel theorem implies a corresponding Parseval's theorem,

$$< f, g >_H = A^{-1} < Tf, Tg >_{L^2(\mu)} \tag{3.20}$$

for any $f, g \in H$. This theorem will be very important in the later chapters, where we introduce matched filtering in the transformed domain. We prove Parseval's theorem. The linearity of T implies

$$\|af + bg\|^2_H = A^{-1}\|T(af + bg)\|^2_{L^2(\mu)} = A^{-1}\|aTf - bTg\|^2_{L^2(\mu)} \tag{3.21}$$

for any $a, b \in \mathbb{C}$. Since both H and $L^2(\mu)$ are Hilbert spaces, we can use the polarization identity to obtain

$$
\begin{aligned}
< f, g >_H &= \frac{1}{4} \sum_{n=0}^{3} i^n \|f + i^n g\|_H^2 \\
&= (A^{-1}) \frac{1}{4} \sum_{n=0}^{3} i^n \|Tf + i^n Tg\|_{L^2(\mu)}^2 \\
&= A^{-1} < Tf, Tg >_{L^2(\mu)} .
\end{aligned}
\tag{3.22}
$$

□

A frame in H is also a *basis* if the frame vectors $\{e_n : n \in M\}$ are *linearly independent*, in the sense that

$$
< g, e >_{L^2(\mu)} = 0
\tag{3.23}
$$

implies $g = 0$ as an element of $L^2(\mu)$. Frames, even tight frames, are not necessarily bases. Frames that are not bases, constitute an overcomplete set of spanning functions for our Hilbert space H. We have more functions than we actually need in order to completely span our space. This redundancy has several positive effects:

- The decomposition becomes more robust. This will often mean that the reconstruction is less affected by errors in some of the coefficients than they would be in the orthonormal case.

- It may be easier to interpret the redundant transform, since we have more information available. For instance, the arrival time of a transient is easier to locate in a redundant setting.

- We have more freedom in choosing our frame and dual frame.

Some of the negative effects of redundancy are

- Redundancy is inefficient; more information is computed than is strictly needed.

- While some redundant transforms constitute tight frames, others do not. Parseval's theorem holds only for tight frames.

Both the Fourier transform, the short-time Fourier transform, the continuous wavelet transform and the discrete wavelet transforms are examples of frames.

3.4.2 The continuous wavelet transform

The family $\{\psi_{a,b} : (a,b) \in M\}$ constitutes a frame in $H = L^2(\mathbb{R})$ indexed by the time-scale plane $M = \{a, b \in \mathbb{R}, a \neq 0\}$. M is equipped with the measure $d\mu = da\,db/a^2$. The analyzing operator T is given by

$$(Tf)(a,b) = < f, \psi_{a,b} >_{L^2(\mathbb{R})} . \tag{3.24}$$

The frame is tight, with frame constant $A = C_\psi$, as given by equation 3.12. The synthesizing operator S is given by

$$(Sg)(t) = A^{-1} < g, \overline{\psi_{a,b}} >_{L^2(\mu)} = C_\psi^{-1} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(a,b)\psi_{a,b}(t)\frac{da\,db}{a^2}. \tag{3.25}$$

3.4.3 The short-time Fourier transform

As before, let $h \in L^2(\mathbb{R})$ denote the window function. Define

$$h_{\omega,t}(u) = \overline{h(u-t)}e^{i\omega u}. \tag{3.26}$$

Then $\{h_{\omega,t} : (\omega,t) \in M\}$ constitutes a frame in $H = L^2(\mathbb{R})$ indexed by the time-frequency plane $M = \mathbb{R}^2$. M is equipped with the measure $d\mu = d\omega\,dt$. The analyzing operator T is given by

$$(Tf)(\omega,t) = < f, h_{\omega,t} >_{L^2(\mathbb{R})} . \tag{3.27}$$

The frame is tight, with frame constant $A = \|h\|^2$. The synthesizing operator S is given by

$$(Sg)(u) = A^{-1} < g, \overline{h_{\omega,t}} >_{L^2(\mu)} = \|h\|^{-2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(\omega,t)h_{\omega,t}(u)d\omega\,dt. \tag{3.28}$$

3.4.4 Discrete frames

When M is discrete or countable, we get discrete frames. In the discrete case, we will always assume that M is equipped with the *counting measure* (Kaiser, 1994, p. 91). In that case, we have $L^2(\mu) = l^2(\mathbb{Z})$, the usual space of all square summable sequences of complex numbers. The frame condition (3.16) becomes

$$A\|f\|_H^2 \leq \sum_n |Tf(n)|^2 \leq B\|f\|_H^2. \tag{3.29}$$

This is the usual definition of a frame as used by Daubechies (1992). Note that if the vectors $\{e_n\}$ constitutes a basis for H in addition to being a frame in H, this basis is a *Riesz basis*. This leads to a *biorthogonal* analysis, as in section 3.7.3.

Let $\psi \in L^2(\mathbb{R})$ be a mother wavelet function for the discrete wavelet transform, and let $\{\psi_{j,k} : j, k \in \mathbb{Z}\}$ be a discrete set of wavelet functions created by translating and dilating, as defined in section 3.3.2. As mentioned in that section, in order for the discrete wavelet transform to have a stable inverse, we must require that $\psi_{j,k}$ constitute a discrete frame for $L^2(\mathbb{R})$. This imposes conditions on our wavelet ψ and on our discretization parameters a_0, b_0. The discretization parameters were defined in section 3.3.2. Unfortunately, necessary and sufficient conditions for $\psi_{j,k}$ to constitute a frame are very complicated. We will not say more about this matter here, but refer the reader to Daubechies (1992).

3.5   Multiresolution analysis

We now turn to the concept of multiresolution analysis. This is the construction of discrete wavelet frames for Hilbert spaces $V_j \subset L^2(\mathbb{R})$. One result of this construction is a decomposition of a function into a *finite* number of dilatations of the mother wavelet, $j \in \{0, 1, 2, ..., J\}$. In order to obtain this, we have to introduce an auxiliary function $\phi(t)$, called the *scaling function*, into our decomposition structure. The result is a transform that is fast and efficient, it can be computed in $O(N)$ time, where N is the length of the sequence to be transformed.

Definition 2 *(Multiresolution Analysis). An orthonormal, 2-band multiresolution analysis is an increasing sequence of subspaces V_j of $L^2(\mathbb{R})$ with scaling function $\phi \in V_0$ satisfying the following conditions:*
(i) (successive approximation)[1] $\cdots V_2 \subset V_1 \subset V_0 \subset V_{-1} \subset V_{-2} \subset \cdots$
(ii) (density) $\overline{\cup_{j\in\mathbb{Z}}V_j} = L^2(\mathbb{R})$,
(iii) (separation) $\cap_{j\in\mathbb{Z}}V_j = \{0\}$,
(iv) (scaling) $f(t) \in V_j \Leftrightarrow f(2^j t) \in V_0$,
(v) (orthonormality) $\{\phi(t-k)\}_{k\in\mathbb{Z}}$ *is an orthonormal basis for V_0.*

Remarks. The current definition is a base 2 multiresolution analysis. One can generalize to base M multiresolution analysis. This is described in section 3.7.1. Also, condition (v) is not necessary. It can be replaced by the requirement that the $\phi(t-k)$ constitute a Riesz basis for V_0. Starting from a Riesz basis for V_0, it is possible to construct an orthonormal basis for V_0. The reader is referred to Daubechies (1992, sec. 5.3.1).

---

[1]There are two equivalent ways of ordering the spaces V_j. I use the ordering used by Daubechies (1992) and Misiti, M. & Misiti, Y. & Oppenheim & Poggi (1996) among others, where smaller j means finer subspaces.
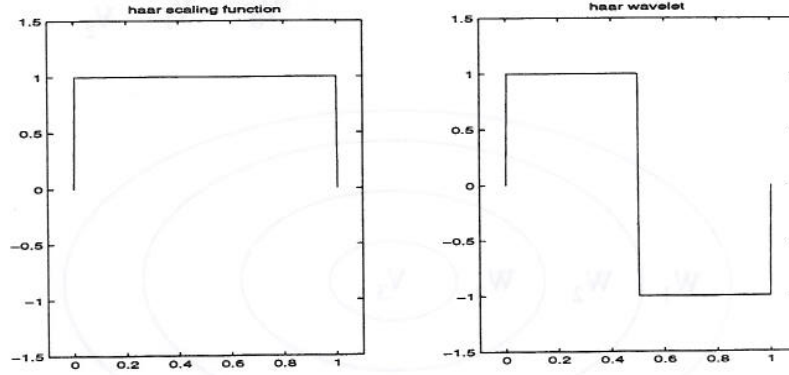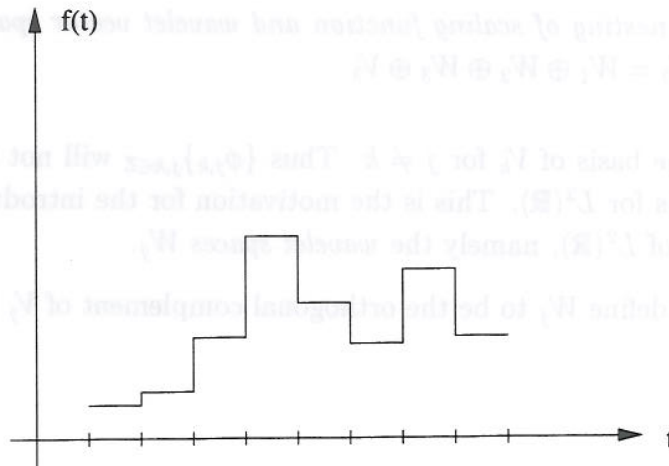
Figure 3.5: The Haar system

Figure 3.6: An example of a function in V_0 for the Haar multiresolution analysis

An example is the *Haar* multiresolution analysis, which is the simplest possible multiresolution analysis. For this analysis, we choose $\phi(t) = \mathbf{1}_{[0,1]}(t)$, where $\mathbf{1}_I$ is the characteristic function on the interval I (see figure 3.5). V_0 then becomes the subspace of $L^2(\mathbb{R})$ of piecewise constant functions with jump discontinuities at the integers $\mathbb{Z}$, as illustrated in figure 3.6. Condition (iv) then implies that V_j is the subspace of $L^2(\mathbb{R})$ of piecewise constant functions with jump discontinuities at $j\mathbb{Z}$.

Now define

$$\phi_{j,k}(t) = 2^{-j/2}\phi(2^{-j}t - k). \tag{3.30}$$

Note that (iv) and (v) implies that $\{\phi_{j,k}\}_{k\in\mathbb{Z}}$ is an orthonormal basis for V_j for each j. V_j is the *scale function space* of scale j.

We would like to construct an orthonormal basis for $L^2(\mathbb{R})$. This will allow a decomposition of a signal into a set of basis functions. But although $\{\phi_{j,k}\}_{k\in\mathbb{Z}}$ is an orthonormal basis for V_j, the elements of this basis will in general not be orthogonal
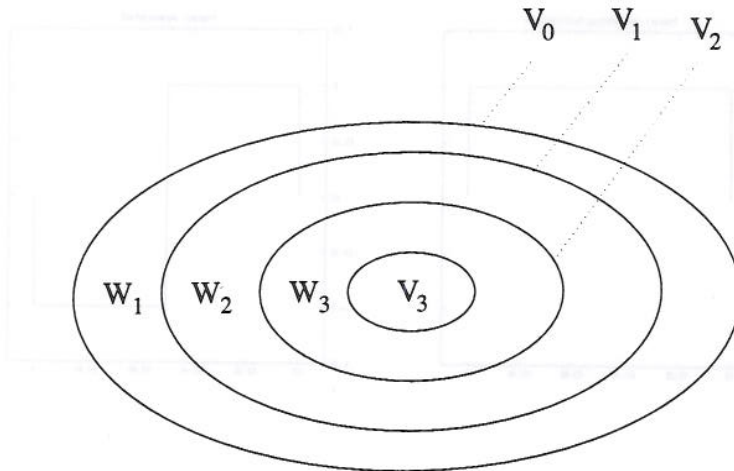
Figure 3.7: The nesting of scaling function and wavelet vector spaces for a level 3 decomposition, $V_0 = W_1 \oplus W_2 \oplus W_3 \oplus V_3$

to elements of the basis of V_k for $j \neq k$. Thus $\{\phi_{j,k}\}_{j,k \in \mathbb{Z}}$ will not constitute an orthonormal basis for $L^2(\mathbb{R})$. This is the motivation for the introduction of a new set of subspaces of $L^2(\mathbb{R})$, namely the *wavelet spaces* W_j.

For every $j \in \mathbb{Z}$, define W_j to be the orthogonal complement of V_j in V_{j-1}. Thus

$$V_{j-1} = V_j \oplus W_j \tag{3.31}$$

and $W_j \perp W_k$ if $j \neq k$.

Let $j_0 < J$ so that V_{j_0} is a fine scale space and V_J is a coarse scale space. By iterating (3.31) we get the following decomposition of V_{j_0} into mutually orthogonal subspaces

$$V_{j_0} = V_J \oplus \left(\bigoplus_{k=j_0+1}^{J} W_k \right). \tag{3.32}$$

Thus a function in V_{j_0} can be expressed as a sum of a function in the coarser space V_J and detail functions in W_k, $k = j_0 + 1, ..., J$. This is illustrated in figure 3.7. As we shall see, the detail functions are the wavelets. Formula (3.32) is the key to multiresolution wavelet decomposition. By (ii) and (iii), we also have the following decompositions of $L^2(\mathbb{R})$ into mutually orthogonal subspaces

$$L^2(\mathbb{R}) = \bigoplus_{k \in \mathbb{Z}} W_k, \tag{3.33}$$

$$L^2(\mathbb{R}) = V_J \oplus \left(\bigoplus_{k=-\infty}^{J} W_k \right). \tag{3.34}$$

Given $\psi \in L^2(\mathbb{R})$, define $\psi_{j,k}(t) = 2^{-j/2}\psi(2^{-j}t - k)$. Assume that $\{\psi_{0,k}\}_{k\in\mathbb{Z}}$ constitutes an orthonormal basis for W_0. In this case, we call ψ the *mother wavelet function*. The integer translations and dilatations of ψ, $\psi_{j,k}$, are the wavelets. Necessary conditions for the above assumption to be true will be given later. Because the W_j space inherit the scaling property (iv) from V_j, we now have that $\{\psi_{j,k}\}_{k\in\mathbb{Z}}$ constitutes an orthonormal basis for W_j. Also, by (3.33) and (3.34), both $\{\psi_{j,k}; k \in \mathbb{Z}\}$ and for fixed J, $\{\phi_{J,k}, \psi_{j,k}; k \in \mathbb{Z}, j \leq J\}$ are orthonormal bases for $L^2(\mathbb{R})$. Given any fixed $J \in Z$ and $f \in L^2(\mathbb{R})$ we thus have

$$f(t) = \sum_{k=-\infty}^{\infty} c_{J,k}\phi_{J,k}(t) + \sum_{j=-\infty}^{J} \sum_{k=-\infty}^{\infty} d_{j,k}\psi_{j,k}(t). \tag{3.35}$$

This is a wavelet *synthesis* or *reconstruction* formula, where f is synthesized from the wavelets $\{\psi_{j,k}; k \in \mathbb{Z}, j \leq J\}$ (J fixed) and the scaling functions $\{\phi_{J,k}; k \in \mathbb{Z}\}$. The corresponding *analysis* or *decomposition* formulas are

$$c_{J,k} = \ <f, \phi_{J,k}>, \tag{3.36}$$
$$d_{j,k} = \ <f, \psi_{j,k}> . \tag{3.37}$$

These formulas follow from the mutual orthogonality of the spaces $\{V_J, W_j; j \leq J\}$ for J fixed. Following the notation of Burrus, Gopinath & Guo (1997), we will often write $c_{j,k}$ as $c_j[k]$ and $d_{j,k}$ as $d_j[k]$, to emphasize the difference between the time translation index k and the scale parameter j.

In a practical implementation, we must have finite sums. A finite sum in j is achieved if we, instead of decomposing $f \in L^2(\mathbb{R})$, decompose the projection of f onto a fine scale space V_{j_0}. By convention, this finest scale is fixed at $j_0 = 0$. Let P_j denote the orthogonal projection onto V_j. We have

$$P_0 f(t) = \sum_{k=-\infty}^{\infty} c_{0,k}\phi_{0,k}(t), \tag{3.38}$$

which expresses $P_0 f \in V_0$ as a linear combination of the basis functions $\phi_{0,k}$ of V_0. This is our starting point before the decomposition. We then choose the number J of decomposition levels, and decompose V_0 by formula (3.32),

$$V_0 = V_J \oplus \left(\bigoplus_{k=1}^{J} W_k \right). \tag{3.39}$$

This means we have the following synthesis formula for $P_0 f$:

$$P_0 f(t) = \sum_{k=-\infty}^{\infty} c_{J,k}\phi_{J,k}(t) + \sum_{j=1}^{J} \sum_{k=-\infty}^{\infty} d_{j,k}\psi_{j,k}(t). \tag{3.40}$$

The calculation of the coefficients $c_{J,k}$ and $d_{j,k}$, $j = 1, 2, ..., J$, is the wavelet decomposition or analysis.

3.5.1 The scaling identity

Since $\phi \in V_0 \subset V_{-1}$, there exists a sequence $h_0[k] \in l^2(\mathbb{Z})$ such that the scaling function satisfies

$$\phi(t) = \sum_{k \in \mathbb{Z}} h_0[k]\phi_{-1,k} = \sqrt{2} \sum_{k \in \mathbb{Z}} h_0[k]\phi(2t - k). \tag{3.41}$$

We will refer to equation (3.41) as the important *scaling identity*. In the frequency domain, the equation takes the form

$$\hat{\phi}(\omega) = H_0\left(\frac{\omega}{2}\right) \hat{\phi}\left(\frac{\omega}{2}\right) \tag{3.42}$$

where $H_0(\omega)$ is the discrete-time Fourier transform of h_0.

Since $\psi \in W_0 \subset V_{-1}$, there exists a sequence $h_1[k] \in l^2(\mathbb{Z})$ such that the wavelet satisfies a corresponding *wavelet identity*,

$$\psi(t) = \sum_{k \in \mathbb{Z}} h_1[k]\phi_{-1,k} = \sqrt{2} \sum_{k \in \mathbb{Z}} h_1[k]\phi(2t - k), \tag{3.43}$$

or equivalently

$$\hat{\psi}(\omega) = H_1\left(\frac{\omega}{2}\right) \hat{\phi}\left(\frac{\omega}{2}\right). \tag{3.44}$$

Note that (3.41) is an implicit equation for $\phi(t)$, whereas (3.43) gives an explicit formula for $\psi(t)$, once $\{h_1[k]\}_{k \in \mathbb{Z}}$ and $\phi(t)$ is given.

As we will see in section 3.5.2, these two identities will play a crucial role in the link between wavelet theory and subband coding. Both h_0 and h_1 will play the role as digital filters; h_0 is a lowpass filter and h_1 is a highpass filter.

3.5.2 Subband coding

We have seen that the scaling and wavelet coefficients $c_j[k], d_j[k]$ can be calculated by inner products, by formulas (3.36) and (3.37). This calculation can be time consuming.

Assume that we have calculated the scaling coefficients $c_{0,k}$ at the finest scale $j = 0$ by some method. (Methods for doing this will be described in section 3.5.3). The
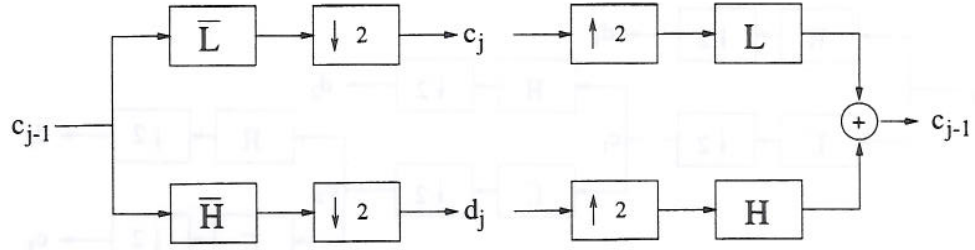
Figure 3.8: Analysis (left) and synthesis (right) of one stage of the subband filtering scheme

Fast Wavelet Transform algorithm is an algorithm for the recursive calculation of the scaling and wavelet coefficients at coarser scales, i.e. the calculation of $c_j[k], d_j[k]$ for $j = 1, 2, .., J$, where J denotes the coarsest scale.

Theorem 3 *(Subband decomposition). Define $c_j[k], d_j[k]$ by (3.36) and (3.37), and assume $c_0[k]$ are given. The coefficients $c_j[k], d_j[k]$, $j = 1, 2, .., J$, are then given recursively by*

$$c_j[k] = \sum_{n \in \mathbb{Z}} \overline{h_0[n - 2k]} c_{j-1}[n], \qquad (3.45)$$

$$d_j[k] = \sum_{n \in \mathbb{Z}} \overline{h_1[n - 2k]} c_{j-1}[n]. \qquad (3.46)$$

In the language of signal processing, the coefficients are obtained by convolution with $\overline{h_0[-n]}$ or $\overline{h_1[-n]}$ followed by downsampling by a factor of 2. Since h_0 play the role as a digital Lowpass filter, we will denote convolution with $\overline{h_0[-n]}$ by the operator $\overline{L}$. Similarly, we denote convolution with the digital Highpass filter $\overline{h_1[-n]}$ by $\overline{H}$.

We have a corresponding reconstruction or synthesis formula:

Theorem 4 *(Subband reconstruction). Assume $\{c_J[k], d_j[k]; j = 1, 2, ..., J, k \in \mathbb{Z}\}$ are given. We have the following formula for the recursive calculation of the finer scale scaling coefficients $c_j[k], 0 < j < J$,*

$$c_{j-1}[k] = \sum_{n \in \mathbb{Z}} h_0[k - 2n] c_j[n] + \sum_{n \in \mathbb{Z}} h_1[k - 2n] d_j[n]. \qquad (3.47)$$

In the language of signal processing, the coefficients c_{j-1} are obtained by upsampling c_j and d_j by 2, convoluting with $h_0[n], h_1[n]$ respectively, and adding. Convolving with $h_0[n], h_1[n]$ will be denoted with L, H, respectively. The proofs of theorems 3 and 4 are given in the appendix.
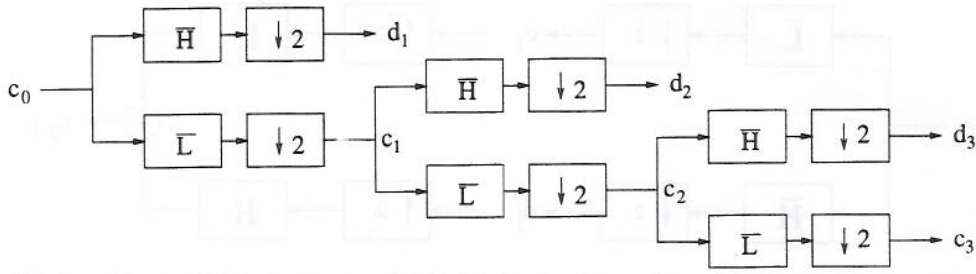
Figure 3.9: Three stage analysis

These two theorems constitute the link between multiresolution analysis and subband coding. One stage of the analysis/synthesis process is illustrated in figure 3.8. A 3-stage analysis is illustrated in figure 3.9.

3.5.3 Practical considerations

When using the wavelet expansion as an abstract mathematical tool, using infinite sums and regarding t as a continuous real variable are appropriate considerations. However, when using the wavelet transform as a practical signal processing tool, we must resort to finite sums, and the function $f(t)$ will be given only at its discrete samples.

First, the function $f(t)$ must be bandlimited, and sampled at a rate no lower than the Nyqvist rate, so that aliasing does not occur (see appendix A.5). We then do the following two approximations:

1. Letting V_0 denote our space of finest scale, we assume $f \in V_0$, or equivalently $\|f - P_0 f\| \approx 0$.

2. Since $f \in V_0$, we have the following expansion for f,

$$f(x) = \sum_{k=-\infty}^{\infty} c_{0,k} \phi_{0,k}(x). \tag{3.48}$$

We need to calculate $c_{0,k} = <f, \phi_{0,k}>$. The usual way of doing this is approximating $c_{0,k}$ by the samples of the function taken at the sample rate (Burrus et al, 1997). We have $\int \phi(t-k)dt = 1$. Also since $j = 0$ corresponds to our finest scale space V_0, the functions $\phi_{0,k}$ will be very narrow and tall. The $\phi_{0,k}$ will have most of its mass in an interval as long as the sampling interval of f. We therefore justify the above approximation by regarding $\phi_{0,k}$ as a Dirac impulse centered at $t = k$. Thus we get

$$c_{0,k} = <f, \phi_{0,k}> \approx <f, \delta(k)> = f(k). \tag{3.49}$$

As soon as we have calculated the coefficients $c_{0,k}$ at the finest scale, the coefficients at other scales $j = 1, 2, ..., J$ can be calculated through the subband filtering scheme (analysis).

Note that the above two approximations do not alter the invertibility of the discrete wavelet transform - by doing subband analysis and then synthesis, we get back to the $c_{0,k}$'s, which still are the samples of the original function. The approximations only mean that what we are doing when computing the discrete wavelet transform is really not computing *exactly* the inner products between the function f and the wavelet/scaling function at different translations and dilatations.

Remark. Instead of making the above two approximations when calculating the discrete wavelet transform, one can create a new formalism - a *discrete multiresolution analysis* - where one starts not with a function $f \in V_0$, but with a sequence in $l^2(\mathbb{Z})$. Both discrete and continous multiresolution analyses are discussed in A. Cohen (1995).

3.6 Wavelet design

Many types of wavelet families have been discovered. The wavelets have varying qualities. The most important properties concerning wavelet design are (Misiti et al, 1996):

- The existence of a scaling function ϕ. Such a function is necessary for a fast filter bank implementation through multiresolution analysis.

- Whether the analysis is orthogonal or biorthogonal.

- The support of ϕ and ψ. This determines whether we are dealing with finite impulse response (FIR) filters, or infinite impulse response (IIR) filters. A digital filter given by the sequence $h[n]$ is said to be of finite impulse response if the number of nonzero elements of $h[n]$ is finite. Otherwise, the filter is of infinite impulse response.

- The symmetry of the wavelets and scaling functions. Symmetry means that the filters have linear phase characteristics, so that dephasing is avoided. This is important in the estimation of arrival time using wavelet based detectors.

- The regularity of the wavelets and scaling functions.

- The number of vanishing moments for ψ or for ϕ (see below).

- The existence of an explicit expression for ψ or for ϕ.

We will discuss how to design FIR filters and discuss different families of wavelets. In chapter 4 and 5, many of these families will be be used in detectors on experimental data from tests at sea. But first, we take a look at some of the conditions that ϕ, ψ must satisfy in order to define a multiresolution analysis. Conditions can equivalently be placed on the filters h_0, h_1. Notice that this discussion applies only to *orthonormal dyadic* multiresolution analyses. However, many of the results can be generalized.

3.6.1 The scaling function and scaling coefficients

Not just any scaling function ϕ will give rise to a multiresolution analysis. It is *not* enough that ϕ is in $L^2(\mathbb{R})$ and that condition (v) of definition 2 on page 27 is satisfied.

A sufficient condition

The following theorem, taken from (A. Cohen, 1995), states *sufficient* conditions for ϕ to define a multiresolution analysis. First we need a new definition.

Definition 3 *A multiresolution is said to be localized if and only if the function ϕ satisfies*

$$\int_{-\infty}^{\infty} (1 + |t|)^m |\phi(t)|^2 dt < \infty \quad for\ all\ m \in \mathbb{N}. \tag{3.50}$$

This is the same as saying that $\hat{\phi}(\omega)$ belongs to all of the Sobolev spaces $H^m(\mathbb{R})$.

In practice, this definition means that ϕ has to have compact support or at least decay rapidly enough to 0 as $t \to \infty$, in the above sense.

Theorem 5 *Let ϕ be a function in $L^2(\mathbb{R})$ that satisfies equation (3.50) and condition (v) of definition 2. Then ϕ defines a localized multiresolution analysis if*
(a) $\hat{\phi}(0) = \int_{-\infty}^{\infty} \phi(t)dt = 1$
(b) $H_0(\omega)$ is C^∞.

Necessary conditions

Various necessary conditions can be placed on the scaling coefficients $h_0[n]$.

Theorem 6 *If $\phi(t)$ is an L^1 solution to (3.41) and $\hat{\phi}(0) \neq 0$ then*

$$\sum_n h_0[n] = \sqrt{2}. \tag{3.51}$$

Furthermore, if $\phi(t) \in L^2(\mathbb{R})$ and condition (v) of definition 2 is satisfied, then

$$\sum_n h_0[n]\overline{h_0[n-2k]} = \delta_{k,0}. \tag{3.52}$$

A corollary to this theorem gives easy-to-check conditions on h_0.

Corollary 1 *If the conditions of theorem 6 are satisfied then*

$$\sum_n |h_0[n]|^2 = 1, \tag{3.53}$$

$$\sum_n h_0[2n] = \sum_n h_0[2n+1] = \frac{1}{\sqrt{2}}. \tag{3.54}$$

The proof of the theorem and corollary is given in Burrus et al. (1997).

In the frequency domain, equations (3.51) and (3.52) take the equivalent forms

$$H_0(0) = \sqrt{2}, \tag{3.55}$$
$$|H_0(\omega)|^2 + |H_0(\omega+\pi)|^2 = 2 \tag{3.56}$$

respectively. In the language of signal processing, equation (3.56) means that the filter h_0 is a so called Quadrature Mirror Filter (QMF). Also, the above two equations imply $H_0(\pi) = 0$, which says that the frequency response at the Nyqvist frequency is zero. This supports the statement that h_0 is a digital lowpass filter.

3.6.2   The wavelet and wavelet coefficients

Theorem 7 *Given a multiresolution analysis with scaling function ϕ. There exists an associated orthonormal wavelet basis $\{\psi_{0,k}\}_{k \in \mathbb{Z}}$ for W_0. The wavelet ψ is determined by*

$$h_1[k] = (-1)^{k+k_1}\overline{h_0[2k_0+1-k]}, \tag{3.57}$$

where k_0 is any integer and $k_1 \in \{0,1\}$ determines the sign of $h_1[k]$. Note that if h_0 is an FIR filter, h_1 is an FIR filter, and k_0 can be chosen so that h_1 is a causal filter.

For a proof of this theorem, see Daubechies (1992, Th. 5.1.1). Since we have the above tight link between the filters h_0 and h_1, it is not surprising that we have

conditions on h_1 similar to conditions (3.51) and (3.52) on h_0:

$$\sum_n h_1[n] = H_1(0) = 0, \tag{3.58}$$

$$\sum_n h_l[n]\overline{h_m[n-2k]} = \delta_{k,0}\delta_{l,m}, \tag{3.59}$$

for $l, m \in \{0, 1\}$. We also have

$$|H_1(\omega)| = |H_0(\omega + \pi)|, \tag{3.60}$$

$$|H_0(\omega)|^2 + |H_1(\omega)|^2 = 2. \tag{3.61}$$

and the familiar condition $\int_{-\infty}^{\infty} \psi(t)dt = 0$. For a proof of these results, see Burrus (1997, ch. 5).

3.6.3 FIR filter design

Finite impulse response filters are the most common when using wavelets for signal processing applications. The following theorem links the length of the scaling function to the length of the filter.

Theorem 8 *If $\phi(t)$ has support on $0 \le t \le K - 1$ then $h_0[k]$ also has support on $0 \le k \le K - 1$. Moreover, the length of the filter h_0 must be an even number.*

For a proof, see (Burrus et al., 1997).

Consider a scaling function with compact support. By the above theorem, this gives rise to a scaling function filter h_0 of even length $2K$. The necessary conditions (3.51) and (3.52) gives us one and K constraints, respectively, on the sequence h_0. Thus we have $2K - K - 1 = K - 1$ degrees of freedom left in choosing h_0. These degrees of freedom can be used to design wavelets with desired properties. One way of using the freedom is forcing wavelet and/or scaling function moments to be zero. We define the kth moments of $\phi(t)$ and $\psi(t)$ as

$$m_0(k) = \int_{-\infty}^{\infty} t^k \phi(t)dt \tag{3.62}$$

and

$$m_1(k) = \int_{-\infty}^{\infty} t^k \psi(t)dt. \tag{3.63}$$

The number of zero wavelet moments are related to the smoothness and differentiability of $\psi(t)$ and $\phi(t)$ (Burrus et al, 1997). Wavelets with many
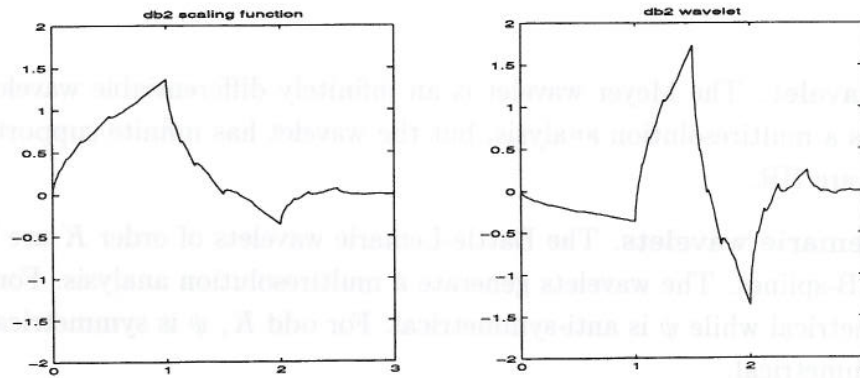
Figure 3.10: The Daubechies 2 system

vanishing wavelet moments are also better for approximating polynomials. A larger number of zero scaling function moments increases the symmetry of the scaling functions and wavelets. Many vanishing scaling function moments also make the approximation of high-resolution scaling coefficients by samples of the signal better.

Some of the most well known wavelet families are explored below. Plots of some of the wavelets and scaling functions are available in appendix D. More information can be found in Daubechies (1992) and Misiti et al (1996).

Daubechies wavelets. The Daubechies family of wavelets are probably the most well-known and most frequently used wavelets. They are multiresolution orthogonal wavelets of compact support. We let $K \in \{1, 2, 3, ...\}$ denote the order of the wavelet. The Daubechies wavelets of order 1 are the Haar wavelets. The length of the subband filters for Daubechies wavelets are $2K$. The wavelets are constructed to have the maximum number of vanishing wavelet moments for a given length $2K$ of the filters. The numbers of vanishing wavelet moments are K. The regularity of the Daubechies wavelets increases with the order; asymptotically as K becomes large, ϕ and ψ belong to $C^{\mu K}$, where $\mu \approx 0.2$. The Daubechies 2 wavelet is plotted in figure 3.10. For more plots, see appendix D. For short, the Daubechies wavelets will often be denoted as 'db' in some of the plots in this thesis.

Symlets. One of the disadvantages of the Daubechies wavelets, is that they are far from symmetric. Motivated by this, Daubechies invented the symlets, which are modification of the classical Daubechies wavelets. The symlets are near symmetrical. For short, the Symlet wavelets will often be denoted as 'sym' in some of the plots in this thesis.

Coiflets. The coiflets are constructed to have both zero wavelet *and* scaling function moments. They are much more symmetrical than the Daubechies wavelets. The coiflets were constructed by Daubechies at the request of R. Coifman. For short, the Coiflet wavelets will often be denoted as 'coif' in some of the plots in

this thesis.

Meyer wavelet. The Meyer wavelet is an infinitely differentiable wavelet. It constitutes a multiresolution analysis, but the wavelet has infinite support, so that the filters are IIR.

Battle-Lemarie wavelets. The Battle-Lemarie wavelets of order K are based on K-degree B-splines. The wavelets generate a multiresolution analysis. For even K, ϕ is symmetrical while ψ is anti-symmetrical. For odd K, ψ is symmetrical while ϕ is anti-symmetrical.

Mexican hat wavelet. The Mexican hat wavelet is proportional to the second derivative of the Gaussian probability density function. It does not generate a multiresolution analysis.

Morlet wavelet. As for the Mexican hat, the Morlet wavelet also has an explicit expression, and it does not generate a multiresolution analysis.

In our practical signal application, we want wavelet transforms that have a fast implementation through iterated filter banks. This means that the wavelet should generate a multiresolution analysis. In addition, the filters should be FIR. By these demands, the Daubechies, Symlet and Coiflet wavelet families are most interresting for our application. The Battle-Lemarie wavelets are also promising, since they both have an associated fast algorithm and are symmetric. In a biorthogonal framework, FIR filters for the Battle-Lemarie wavelets are available. Biorthogonal analysis is discussed in the following section.

3.7 Multiresolution generalizations

Various extensions of the basic multiresolution scheme exist. The most common are

- M-band multiresolution analysis
- Wavelet packets and best basis algorithms
- Biorthogonal wavelets
- Time invariant transforms

The four extensions can be combined. They are explored in the following sections.

3.7.1 M-band multiresolution analysis

In our multiresolution formulation, we have chosen a scale multiplier of $a_0 = 2$. The theory can be generalized to a more general scalar multiplier M, where (3.41)
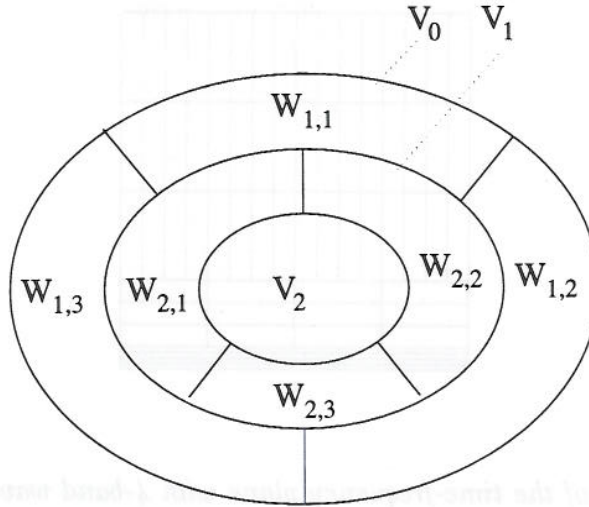
*Figure 3.11: The nesting of scaling function and wavelet vector spaces for a level 2
4-band decomposition,* $V_0 = W_{1,1} \oplus W_{1,2} \oplus W_{1,3} \oplus W_{2,1} \oplus W_{2,2} \oplus W_{2,3} \oplus V_2$

becomes

$$\phi(t) = \sqrt{M} \sum_{k \in \mathbb{Z}} h_0[k]\phi(Mt - k). \tag{3.64}$$

Usually, M will be an integer, although it may be a rational number in some
cases. Many of the results from 2-band wavelet theory can be generalized to
the M-band case. From the subband point of view, we get M bandpass filters
$h_k, k = 0, 1, ..., M - 1$ in an M band filter bank, where the downsampling factor is
equal to M. The multiresolution formulation becomes slightly different. At each
scale j, we get $M - 1$ mutually orthogonal wavelet spaces $W_{j,k}, k = 0, 1, ..., M - 1$.
Equation (3.31) changes to

$$V_{j-1} = V_j \oplus \left(\bigoplus_{k=1}^{M} W_{j,k} \right). \tag{3.65}$$

This is illustrated in figure 3.11. The advantage of M larger than 2 is a more
flexible tiling of the time-scale plane, see figure 3.12. More about M-band wavelet
decompositions can be found in Burrus et al (1997, Ch. 11).

3.7.2 Wavelet packets

The classical dyadic discrete wavelet transform leads to a logarithmic frequency
resolution. At low frequencies we have good frequency resolution but poor
time resolution. At high frequencies, we have good time resolution but poor
frequency resolution. The *wavelet packet* decomposition was proposed by Coifman
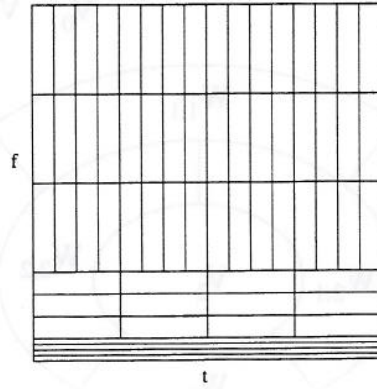
Figure 3.12: Tiling of the time-frequency plane with 4-band wavelet basis
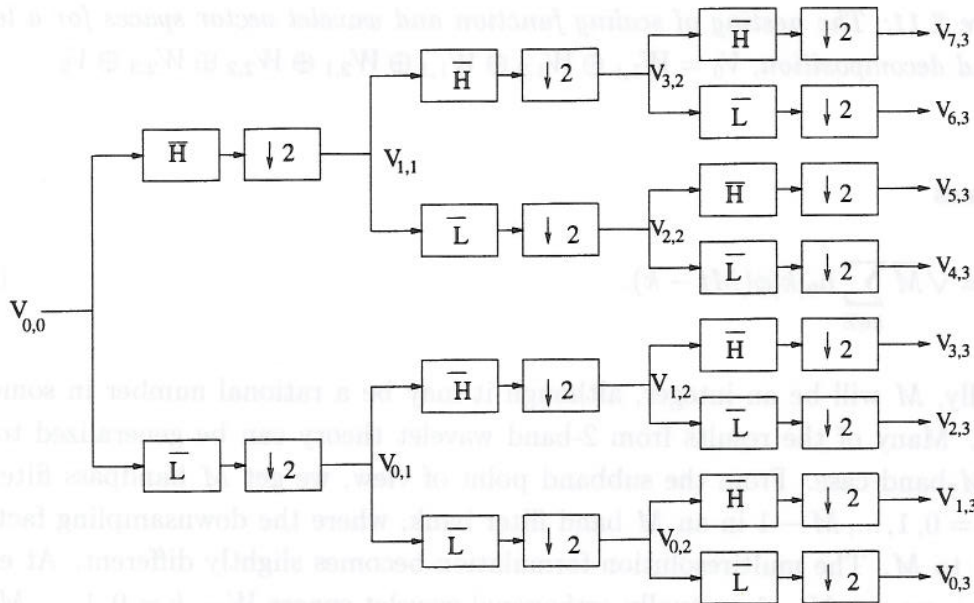

Figure 3.13: The full binary tree for a 3-stage wavelet packet analysis

(1992) to allow a more flexible tiling of the time-scale plane. The wavelet packet decomposition allows for good frequency resolution also at high frequencies, or good time resolution at low frequencies.

The idea of wavelet packets is to iterate (split and down-sample) the highpass branch as well as the lowpass branch in the subband filtering algorithm. The result is a full binary tree as shown in figure 3.13. Assume we start the decomposition at scale $j = 0$. For each scale level $j \geq 0$, we get a decomposition of the original space V_0 into 2^j orthogonal spaces. We label those spaces $V_{n,j}$ with the index $n \in \{0, 1, ..., 2^j - 1\}$. The parameter n will be a measure of the frequency of the basis functions of that space. The decomposition formula (3.31) becomes

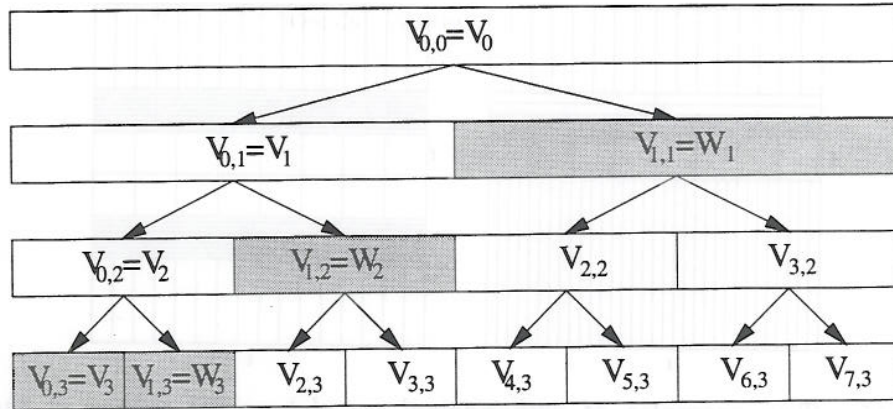$$V_{n,j-1} = V_{2n,j} \oplus V_{2n+1,j}. \tag{3.66}$$

Figure 3.14: Wavelet packet spaces $V_{n,j}$ in a 3 level decomposition organized as a binary tree. The usual wavelet decomposition is shown in shaded colour

The wavelet spaces are special cases in this new, more general structure. We have $V_j = V_{0,j}$ and $W_j = V_{1,j}$. The spaces are illustrated in figure 3.14, with a level 3 wavelet decomposition shown in shaded colour.

Any pruning of this tree will lead to a decomposition of V_0 into orthogonal subspaces. By a pruning we mean the following

- Start with the top node of the tree, which is $V_{0,0}$.

- Either take the current node $V_{n,j}$ or its two children $V_{2n,j+1}, V_{2n+1,j+1}$, given by equation (3.66) as elements of the decomposition.

- Repeat the above step recursively down the tree.

This leads to a very flexible tiling of the time-scale plane. Fine resolution in frequency can be obtained at the frequencies where that is required; see figure 3.15. By combining the wavelet packet transform with M-band multiresolution analysis, an even more flexible tiling of the time-scale plane can be obtained.

The pruning of the tree can be determined before the decomposition is made. If one knows what type of signal is to be analyzed, the pruning can be tailor-made to that kind of signal. An example is in signal detection, when one knows the approximate time-scale structure of the signal one is searching for. In that case, an orthonormal transform can be obtained with the fast algorithm in $O(N)$ time.

The best basis algorithm

Another option is to make the full decomposition tree, and then choose a pruning specific to that particular decomposition. The most typical way of doing that is choosing the orthonormal decomposition with the least cost according to some additive cost function (see below). This algorithm is often called the *best basis algorithm*, and it is due to Coifman (1992).
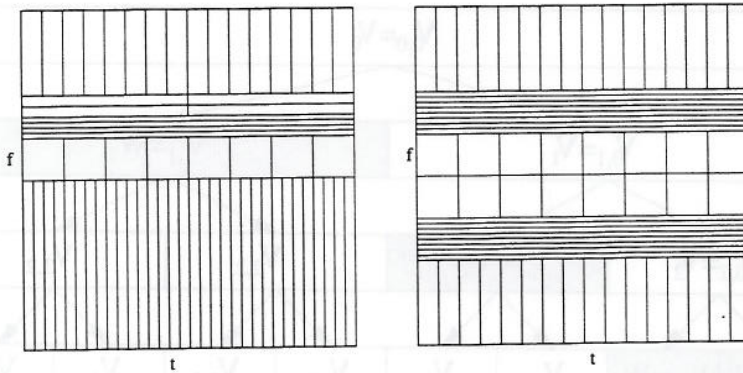
Figure 3.15: Two examples of orthonormal wavelet packet tilings of the time-frequency plane

The cost function should be a measure of the number of coefficients required to accurately describe the signal. The cost should be large when the coefficients are roughly the same size, and small when all but a few coefficients are negligible. Any averaging process should increase the information cost, suggesting that we consider convex cost functions. For practical purposes, our cost function should be *additive*.

Definition 4 *A cost function $C : \mathbb{R}^n \to \mathbb{R}$ is additive if it satisfies the properties*

$$C(0) = 0, \tag{3.67}$$
$$C(x) = \sum_i C(x_i). \tag{3.68}$$

When the cost function is additive, finding the decomposition with the least cost in the binary tree of coefficients can be done in $O(N)$ time (Coifman, 1992).

The most frequently used additive cost function is the *Shannon entropy* function,

$$C(x) = -\sum_i |x_i|^2 \log |x_i|^2. \tag{3.69}$$

Other important additive cost functions are the energy function

$$C(x) = -\sum_i |x_i|^2 \tag{3.70}$$

and the threshold function

$$C(x) = \sum_i H(|x_i| - \epsilon), \tag{3.71}$$

where $H(u)$ is the unit step function and ϵ is a predefined threshold level.

The best basis approach is efficient for denoising and compression tasks. Since the full decomposition takes $O(N \log N)$ time and the search for best basis takes $O(N)$ time, the best basis transform can be done in $O(N \log N)$ time.

3.7.3 Biorthogonal wavelets

Up to now, our analysis has been orthogonal. The same set of wavelets and scaling functions have been used in both analysis and synthesis. By relaxing the orthonormality condition and expanding functions in non-tight frames that still are bases, we obtain biorthogonal wavelets. We get two families of wavelets, one $(\psi_{j,k})$ for analysis and the other $(\tilde{\psi}_{j,k})$ for synthesis. $\psi_{j,k}$ constitute a Riesz basis for $L^2(\mathbb{R})$ and $\tilde{\psi}_{j,k}$ is the dual basis. In the same manner we can introduce two families of scaling functions, $\phi_{j,k}$ and $\tilde{\phi}_{j,k}$, and a multiresolution framework is available (Burrus et al, 1997).

The main advantages of biorthogonal transforms are

- Orthogonal analysis puts many restrictions on the lowpass and highpass filters. In the biorthogonal case, one has much more flexibility in the choice of filters.

- Symmetric wavelets and scaling functions of compact support are available in the biorthogonal framework. This removes the problem of dephasing. As mentioned, symmetry is important for instance in the estimation of arrival time using wavelet based detectors.

- The primary and the dual wavelets can have quite different properties. This can be exploited for example in image compression, where the smoother one in the pair can be used to reconstruct the coded image in order to get better visual appearance.

while the main downsides are

- Parseval's theorem no longer holds in the biorthogonal case.

- Since the transform is no longer orthogonal, white noise will become correlated or coloured after the transformation.

- The analysis gets more complicated.

3.8   Time invariant transform

M-band wavelet filters have the property that shifts by M in the input sequence produce shifts by 1 in the transform coefficients. The reason for this is the

downsampling by a factor of M. However, input sequence shifts by nonmultiples of M produce transform coefficients that are not simply related to shifts of the nominal coefficients.

This is the problem of shift variance of the wavelet transform. Integer shifts of the input sequence f will not necessarily give integer shifts of the output coefficients. This is a problem in for example detection problems. If the wavelet coefficients of a transient is strongly dependent of shifts in the input sequence, it can be more difficult to recognize the transient.
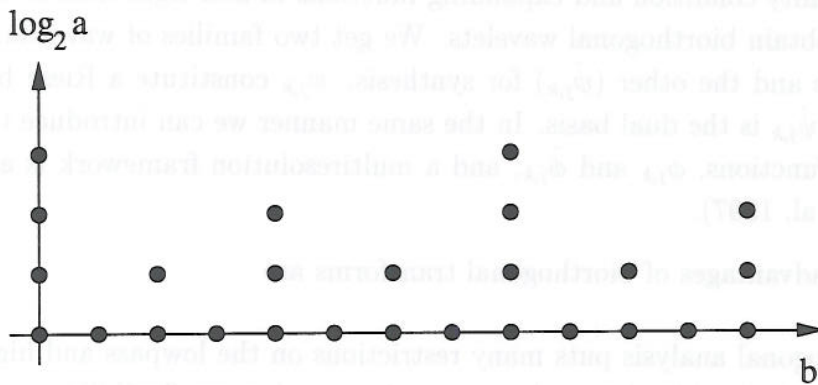

Figure 3.16: Discrete dyadic sampling of the wavelet transform

The problem originates from the way we discretized the wavelet transform. As described in section 3.3.2, the continuous wavelet transform given by equation (3.10) is usually discretized by $a = 2^j, b = k2^j$, where $j, k \in \mathbb{Z}$. This leads to the usual dyadic multiresolution formulation. The shift variance comes from the fact that the translation index b is discretized at different rates depending on the scale j, as shown in figure 3.16.

Assume we compute a J-level discrete wavelet transform. It is shown below that the wavelet coefficients of integer shifts of the input sequence are periodic with a period of 2^j shifts in the input sequence. To account for all possible shifts, we thus need to compute the wavelet transform of 2^J shifts of an input sequence. One might fear that this requires taking 2^J wavelet transforms, thus spending $O(2^J N)$ time and using $2^J N$ storage. Luckily, the situation is not that bad, as we will now see.

We use the following notation for the usual dyadic discrete wavelet transform:

$$(T_\psi f)(j,k) = <f, \psi_{j,k}> = \int_{-\infty}^{\infty} f(t)2^{-j/2}\overline{\psi}(2^{-j}t - k)dt. \qquad (3.72)$$

The transform with respect to the scaling function ϕ is defined similarly. Define

the integer shift operator $S : L^2(\mathbb{R}) \to L^2(\mathbb{R})$ by

$$(Sf)(t) = f(t+1). \tag{3.73}$$

Theorem 9 *For a fixed scale j, shifts in the input sequence f by integer multiples of 2^j will give integer shifts in the transform $(T_\psi f)(j, k)$. Specifically, assume the input sequence is shifted by $q = q_0 + 2^j q_1$, where $q_0 \in \{0, 1, ..., 2^j - 1\}$. We have*

$$(T_\psi S^q f)(j, k) = (T_\psi S^{q_0} f)(j, k + q_1). \tag{3.74}$$

A completely analogous statement holds for T_ϕ.

Proof. We calculate

$$
\begin{aligned}
(T_\psi S^q f)(j, k) &= \int_{-\infty}^{\infty} f(t+q) 2^{-j/2} \overline{\psi}(2^{-j}t - k) dt \\
&= \int_{-\infty}^{\infty} f(u + q_0) 2^{-j/2} \overline{\psi}(2^{-j}(u - 2^j q_1) - k) du \\
&= \int_{-\infty}^{\infty} f(u + q_0) 2^{-j/2} \overline{\psi}(2^{-j}u - (k + q_1)) du \\
&= (T_\psi S^{q_0} f)(j, k + q_1), \tag{3.75}
\end{aligned}
$$

where we have made use of the substitution $u = t + 2^j q_1$. The calculation for T_ϕ is similar.

$\square$

We show that the level J discrete wavelet transform of all shifts of an input signal gives an output of approximately $(J+1)N$ coefficients.

The discrete J-level wavelet transform of a signal consists of detail coefficients at levels $j = 1, 2, ..., J$ plus approximation coefficients at level J. The coefficients are downsampled by a factor of 2 from level to level, starting at full length N at $j = 0$. Thus at level j, the number of detail coefficients are $N2^{-j}$. By the above theorem, we need to calculate the detail coefficients $(T_\psi S^q f)(j, k)$ of 2^j shifts of the input signal f at each each level $j = 1, 2, .., J$. The number of coefficients at scale j for a given translation of f is, as we have seen above, $N2^{-j}$. Thus the total number of detail coefficients at scale j, accounting for all shifts of the input sequence, is $2^j N2^{-j} = N$. So we need JN detail coefficients. In addition we need N approximation coefficients at scale J, giving a total of $(J+1)N$ coefficients. The storage increases by a factor of $J+1$ compared to the ordinary discrete wavelet transform of an unshifted sequence.

By an algorithm originally proposed by Beylkin (1992), the $(J+1)N$ coefficients of all shifts of the input sequence can be calculated in $O(JN)$ time. A full wavelet decomposition involves decomposition down to the scale $N = \log_2 J$. Thus all shifts of a signal can be computed in $O(N \log_2 N)$ time.

Remark. Calculating all these shifts is actually equivalent to computing the undecimated discrete wavelet transform, which was mentioned in section 3.3.2. This transform corresponds to the sampling $a = 2^j, b = k$, which leads to a redundant set of wavelet functions

$$\psi_{j,k}^{und} = 2^{-j/2} \psi \left(\frac{t-k}{2^j} \right). \tag{3.76}$$

Calculating the wavelet transform of all shifts of the input sequence (or equivalently, calculating the undecimated discrete wavelet transform) copes with the translation variance. However, redundancy is introduced, and the storage need increases with a factor $J+1$. One way of achieving a shift invariant discrete wavelet transform of the same length N as the input sequence is to use Coifman's best basis algorithm on all the possible shifts of the input sequence. Coifman's algorithm was described in section 3.7.2. The reason why the algorithm works in this case, is that the wavelet coefficients of all shifts of the input signal can be stored in a binary tree, which can be searched for a minimum cost solution in $O(N)$ time using the same algorithm as in the wavelet packet case. Since the decomposition of all shifts costs $O(JN)$ time, the complexity of this algorithm is $O(JN)$.

An alternative to Coifman's algorithm is the algorithm proposed by Weiss (1994). Weiss proposes a translation invariant orthonormal transform requiring $O(N)$ operations. This algorithm also chooses a shift of the input sequence whose wavelet coefficients minimizes a cost function, but it only finds a local minimum, not necessarily a global minimum. At each stage in the decomposition, it calculates the cost of the two possible shifts of the input approximation coefficients from the last scale and chooses the set with the least cost.

Both the algorithms by Coifman and by Weiss suffer from the fact that they are *nonlinear*, i.e. that the transform of the sum of two signals is not equal to the sum of the transforms in general. The undecimated discrete wavelet transform, however, is both linear and time invariant (but it is redundant).

4 DETECTION AND CHARACTERIZATION

This chapter addresses two topics,

- Algorithms for the detection of transient phenomena in noisy data.

- Methods for characterizing detected transients.

In the detection problem, an important topic is how much advance knowledge is available of the transient. Sometimes, almost no information is available. A well known detector in that case is the *energy detector*, described in section 4.2. A very promising alternative is using the maximum wavelet coefficient as a detection statistic. This detector is described in section 4.5.

When the wave-shape of the transient is completely known, the so-called *matched filter* is optimal. In a white noise background, the matched filter output is simply a *correlator* between the known transient and the input signal. The wave-shape of the transient can be known from previous measurements of the phenomenon. It can be an empirical function, or a function modeled to fit the data. The traditional time-domain matched filtering is presented in section 4.3. In section 4.7 it is shown how the matched filtering can be performed in the transformed domain, which gives certain advantages.

The chapter begins with a presentation of the energy detector and the matched filter, which are classical detectors not based on wavelets. In section 4.4, a review of literature on wavelet based detectors is given. Sections 4.5 and 4.7 concerns the maximum transform coefficient detector and the transform domain matched filter detector. The performance of different detectors are compared using so-called *receiver operating characteristics* curves in section 4.8.

As soon as a transient has been detected, the classification and characterization of the transient is important. Transients are modeled as functions $s(t; \theta)$. The parameters θ can for example be amplitude, frequency, damping and arrival time. I will look for the *maximum likelihood* estimates of the parameters θ. This framework is described in section 4.9 and onwards. Theory on signal detection and on maximum-likelihood estimation of signal parameters can be found in Helstrom (1968).

4.1 The detection of transients

Let $s(t)$ be the waveform of a known transient. We assume that $s(t)$ has compact support and arrival time 0. Thus the support of s is $[0, T]$, where T is the length of the transient.

We want to detect if the transient $s(t)$ is present in our signal $x(t)$ at time $t = t_0$. For this we use the hypothesis test

$$H_0 \quad : \quad x(t) = n(t) \tag{4.1}$$
$$H_1 \quad : \quad x(t) = n(t) + s(t), \tag{4.2}$$

where $n(t)$ is noise. In the following, we will assume that both the signal and noise are real valued. The noise is assumed to be Gaussian and of zero mean, $E[n(t)] = 0$. Furthermore, the noise will be assumed to be white or uncorrelated, with uniform spectral density $N_0/2$.

Remark. If the noise is correlated or coloured, standard methods for whitening of the noise exists. These methods requires knowledge of the power spectral density of the noise.

We will denote the detector output as y. This detection statistic is compared to a fixed threshold level in order to decide between H_0 and H_1. If the output exceeds the threshold, a detection has been made. Setting a high threshold level gives few false-alarm rates but reduces the probability of detection. Setting a low threshold level gives a high probability of detection at the expense of a higher false-alarm rate.

There are two types of detection problems. In the first type, which we will call type 1, we have a fixed observation interval I, and want to test if we have a transient in that interval or not. The detector output is a single variable y. If this variable exceeds a certain threshold, we have a detection. This approach takes no notice as to when the transient arrived, or whether there were several transients present.

In the second type of detection problem, denoted type 2, we look for several transients arriving at unknown arrival times. The detector output y is a function of time. At each time, we compare y to a threshold, and if y exceeds that threshold, we have a detection at that time instant. Since s has support $[0, T]$, we need the values of x in the interval $[t, t+T]$ in order to compare x and s at the time t. Thus the output y is delayed by T. If a transient is inside the signal x, the whole transient has to be received before we can detect it. If the time of arrival of the transient is t, the detector output corresponding to the time of arrival is given by $y(t + T)$.

The second type of detection problem is most relevant in our case, since we are looking for several transients arriving at unknown arrival times. Before we turn to wavelet based detectors, we will describe two very common detectors, namely the energy detector and the matched filter. Both of these detectors are relevant for the second type of detection problem.

4.2 The energy detector

The energy detector assumes no knowledge of the transient but its typical length T. The detector output is simply the signal energy in the interval $[t, t+T]$,

$$y(t + T) = \|x\|^2_{L^2([t, t+T])}. \tag{4.3}$$

The energy detector serves as a lower bound for attainable performance.

4.3 The matched filter

The matched filter assumes full knowledge of the signal s that we are looking for. In this section we define the matched filter, and show that in white noise, it is equal to a correlator between the signal s and the input x.

Assume $x(t)$ consists of the known signal $s(t)$ and an additive white noise component $n(t)$,

$$x(t) = s(t) + n(t). \qquad (4.4)$$

Consider *any* linear filter with impulse response $h(t)$, applied to $x(t)$. Since the filter is linear, the resulting output $y(t + T)$ may be expressed as

$$y(t + T) = s_o(t + T) + n_o(t + T), \qquad (4.5)$$

where $s_o(t + T)$ and $n_o(t + T)$ are produced by the signal and noise components of the input $x(t)$, respectively. We want the output signal component $s_o(t + T)$ to be considerably greater then the output noise component $n_o(t + T)$ at the time $t = 0$ of the arrival of the signal. A way of achieving this is to maximize the output signal-to-noise ratio at $t = 0$, defined as

$$(SNR)_o = \frac{|s_o(T)|^2}{E[|n_o(t)|^2]}. \qquad (4.6)$$

Definition 5 *A matched filter is the linear filter that maximizes the output signal-to-noise ratio, given by equation (4.6), at the time of arrival.*

In the special case of white noise, the impulse response of a matched filter is simply the time reversal of the known signal s.

Theorem 10 *In white noise, the impulse response of a matched filter is given by*

$$h_{mf}(t) = s(T - t), \qquad (4.7)$$

where the filter is matched to the signal $s(t)$, starting at time 0 and with support $[0, T]$.

The output of the matched filter is a *correlator* between the signal s and the input x, as the following calculation show

$$
\begin{aligned}
y(t+T) &= x * h_{mf} \equiv \int_{-\infty}^{\infty} x(\tau) h_{mf}(t+T-\tau) d\tau \\
&= \int_{-\infty}^{\infty} x(\tau) s(T-(t+T-\tau)) d\tau \\
&= \int_{-\infty}^{\infty} x(\tau) s(\tau - t) d\tau \\
&= \int_{0}^{T} x(\tau + t) s(\tau) d\tau.
\end{aligned} \tag{4.8}
$$

When we talk about matched filters in this thesis, we will always implicitly assume white noise, so that the matched filter is equal to a correlator.

The matched filter output using the empirical 2 Hz transient of figure 2.4 as a filter, applied to the signal of figure 2.3, is shown in the second plot of figure 4.1. The shown transients are clearly detected.

Remark. The shown detector output is actually the square of the matched filter output. The reason for this to make it easier to compare the output to other detection techniques. The various matched filter outputs in the plots throughout this thesis will always be based on the square of the quantity $y(t+T)$.

Proof of theorem 10.
The output $y(t+T)$ of the filter on a signal $x(t)$ can be written

$$
\begin{aligned}
y(t+T) &= x * h = \int_{-\infty}^{\infty} x(\tau) h(t+T-\tau) d\tau \\
&= \int_{-\infty}^{\infty} (-x(t+T-u)) h(u) du = \langle \tilde{x}_{t+T}, h \rangle_{L^2(\mathbb{R})},
\end{aligned} \tag{4.9}
$$

where $\tilde{x}_{t+T}(u) = -\overline{x(t+T-u)}$.

Thus, $s_o(T) = \langle \tilde{s}_T, h \rangle$, where $\tilde{s}_T(u) = -\overline{s(T-u)} = -s(T-u)$. Furthermore, since $n_o(t) = \int_{-\infty}^{\infty} n(\tau) h(t-\tau) d\tau$, we have

$$
\begin{aligned}
E[|n_o(t)|^2] &= E\left[\int_{-\infty}^{\infty} n(u) h(t-u) du \int_{-\infty}^{\infty} \overline{n(v) h(t-v)} dv\right] \\
&= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} E[n(u)n(v)] h(t-u) \overline{h(t-v)} du dv.
\end{aligned} \tag{4.10}
$$

Since

$$
E[n(u)n(v)] = \frac{N_0}{2} \delta(u-v), \tag{4.11}
$$

we have

$$E[|n_o(t)|^2] = \frac{N_0}{2} \int_{-\infty}^{\infty} h(t-u)\overline{h(t-u)}du = \frac{N_0}{2}\|h\|^2_{L^2(\mathbb{R})}. \tag{4.12}$$

We now have the following expression for the output signal-to-noise ratio

$$(SNR)_o = \frac{|<\tilde{s}_T, h>|^2}{\frac{N_0}{2}\|h\|^2}. \tag{4.13}$$

The Cauchy- inequality states that

$$|<\tilde{s}_T, h>| \leq \|\tilde{s}_T\|\|h\| = \|s\|\|h\|. \tag{4.14}$$

Thus

$$(SNR)_o \leq \frac{2\|s\|^2}{N_0}. \tag{4.15}$$

The right-hand side of this relation does not depend on the filter $h(t)$. Thus the output signal-to-noise ratio will be a maximum when $h(t)$ is chosen so that equality holds, that is

$$(SNR)_{o,max} = \frac{2\|s\|^2}{N_0}. \tag{4.16}$$

For this case, $h(t)$ assumes its optimum value $h_{opt}(t)$. From the Cauchy-Schwartz inequality, we find that this optimum occurs when h and $\tilde{s}_T$ are linearly dependent, i.e.

$$h_{opt}(t) = A\tilde{s}_T(t) = -As(T-t), \tag{4.17}$$

where $A \neq 0$ is an arbitrary constant. Thus our optimum filter is $h_{opt}(t) = s(T-t)$, which is a time-reversed and delayed version of the input signal $s(t)$.

□

An extensive review of matched filters is given in Turin (1960).

4.4 Wavelet based detectors

This section is a short description of the various wavelet based detectors that the author has encountered in the literature.

1. Mallat (1991) suggests calculating the zero-crossings of a wavelet transform and using them in comparing signals.

2. Frisch and Messer (1991, 1994) use the maximum coefficient a discrete wavelet transform as a detection statistic, and shows improved performance compared to the energy detector.

3. Del Marco and Weiss (1994a, 1994b, 1997) also uses the maximum transform coefficient as detection statistic. They use several different orthonormal bases, some of which performs better than others. Using wavelet packets outperforms the standard wavelet decomposition. Also, time invariant decomposition performs much better than the standard time variant orthonormal transforms.

4. Carter (1994) proposes a detector where coefficient thresholding in the wavelet domain is followed by an inverse transform and then an energy detector.

5. Anant, Dowla and Rodrigue (1994) suggests taking an orthonormal discrete wavelet transform and then using a neural network to recognize individual subbands of transients.

6. Chen (1994) suggests the implementation of a time-varying matched filter through a continuous wavelet transform.

The author has studied the articles of 2 and 3 quite thoroughly. Unfortunately, the detectors presented in these articles are of type 1. They are suited to the problem of testing whether we have a transient present in a fixed observation interval I or not. The problem of detecting several transients of unknown arrival times is not addressed in these articles.

The next section treats the maximum transform coefficient detector, which is the type of detector presented in 2 and 3. A very similar approach, namely using the maximum windowed Fourier transform coefficient as detection statistic, is discussed briefly in section 4.6. A quite different approach, namely using matched filtering in the wavelet domain, is described in section 4.7. All the results of section 4.7 are due to the author, based on an idea by Eriksen (1996).

4.5 Detector based on maximum wavelet coefficient

In this section we will look at the use of the value of the maximum transform coefficient as detector output. This detector requires very little prior knowledge of the transient. The only knowledge we assume of the transient lies in the choice of wavelet. Since this detector is both fast and requires little prior information, it

is useful for an initial 'screening' of the data in order to find the most powerful occurring transient phenomena. The scale corresponding to the maximum coefficient is a rough indication of the frequency of the transient. Matched filters can then be designed according to the found transients.

The maximum transform coefficient detector is of type 1. We have a fixed observation interval I, and want to test if we have a transient in that interval or not. The detector output is a single variable y. If this variable exceeds a certain threshold, we have a detection. This approach takes no notice as to when the transient arrived, or whether there were several transients present.

We first look at the detector defined by the continuous wavelet transform. The detector output y can be defined as

$$y = \max_{a,b}(Wf)(a,b) = \max_{a,b} \int_{-\infty}^{\infty} x(\tau)|a|^{-1/2}\overline{\psi\left(\frac{\tau - b}{a}\right)}d\tau. \tag{4.18}$$

The detector y is the maximum of the outputs of an infinite bank of matched filters. The input signal $x(t)$ is passed through a bank of parallel filters, each matched to the signal $\overline{\psi}$ with a different scaling parameter value a. The output of each filter is passed to a peak detector which identifies the maximum with respect to time. The maximum over all scaling peak values is our detection statistic y. The performance of this detector depends on the degree of similarity between the wavelet and the transient. When the wavelet matches the transient exactly, an exact matched filter results.

For a fast implementation, the above detector is usually discretized, both in scale and time. This yields a bank of matched filters for which only a sample of the possible scalings of ψ are used, and for which the detector output is tested only on a discrete set of time points. The usual dyadic discretization $a = 2^j, b = k2^j$ in a J-level wavelet decomposition, yields the detector

$$y = \max_{k,1\leq j\leq J} \{c_{J,k}, d_{j,k}\}. \tag{4.19}$$

One takes a discrete transform, and picks the highest value coefficient as detection statistic. The disadvantage of this approach, is that the time axis is sampled at a different rate for different scalings, $b = k2^j$. A uniform sampling of the time axis, $b = k$, giving the undecimated time invariant transform, should yield better results.

Frisch and Messer (1991, 1994) use the maximum coefficient of a discrete wavelet transform as a detection statistic, and shows improved performance compared to the energy detector. Del Marco and Weiss (1994a, 1994b, 1997) makes use of the maximum energy coefficient of orthonormal discrete wavelet transforms. They have proposed several variants of the detector using different decompositions:

- 2-band orthonormal discrete wavelet decomposition.

- 2-band best wavelet packet basis decomposition.

- 2-band time invariant best wavelet packet basis decomposition.

- 4-band time invariant best wavelet packet basis decomposition.

- M-band extended time invariant best wavelet packet basis decomposition.

Del Marco et al. evaluate the detector performance by plotting receiver operating characteristic curves generated through Monte Carlo simulation using synthetic transients of the same type as equation (2.2). They find that detector performance improves as you go down the above list. Unlike Frisch and Messer, Del Marco and Weiss does not use the maximum transform coefficient as detector output. Instead, they use the maximum energy transform coefficient, i.e. the maximum of $|c_n|^2$, where c_n denotes the individual transform coefficients. This approach takes no notice of whether the correlation is a large positive or a large negative value. For real signals and wavelets, a large negative transform coefficient value only differs from a large positive value by a sign difference between signal and wavelet.

The maximum transform coefficient represents the function that gives the highest correlation with the input signal. When using a discrete wavelet, a key issue is choosing a wavelet that concentrates the signal energy in as few transform coefficients as possible. This will often give a higher maximum correlation between wavelet and signal, and a better detector results (Del Marco and Weiss, 1994a, 1994b).

In our application, a problem with the above detector is that it is of type 1. The detector takes no notice as to when the transient arrived, or whether several transients were present. We want a detector that gives an output $y(t)$ as a function of time. If y exceeds a threshold at a certain time instant t_0, we have a detection at that time. The problem is: for a given time t, which j and k values should be taken into consideration when determining the detector output at that time? This probably depends on the shape and symmetry of the wavelet.

The author simply chose $k = 0$, and tested the ad hoc detector

$$y_j(q) = \int_{-\infty}^{\infty} x(\tau + q)\overline{\psi_{j,0}(\tau)}d\tau, \tag{4.20}$$

where the detector only makes use of one scale j. We can rewrite this filter output using theorem 9 on page 46:

$$y_j(q) = (T_\psi S^q x)(j, 0) = (T_\psi S^{q_0} f)(j, q_1), \tag{4.21}$$
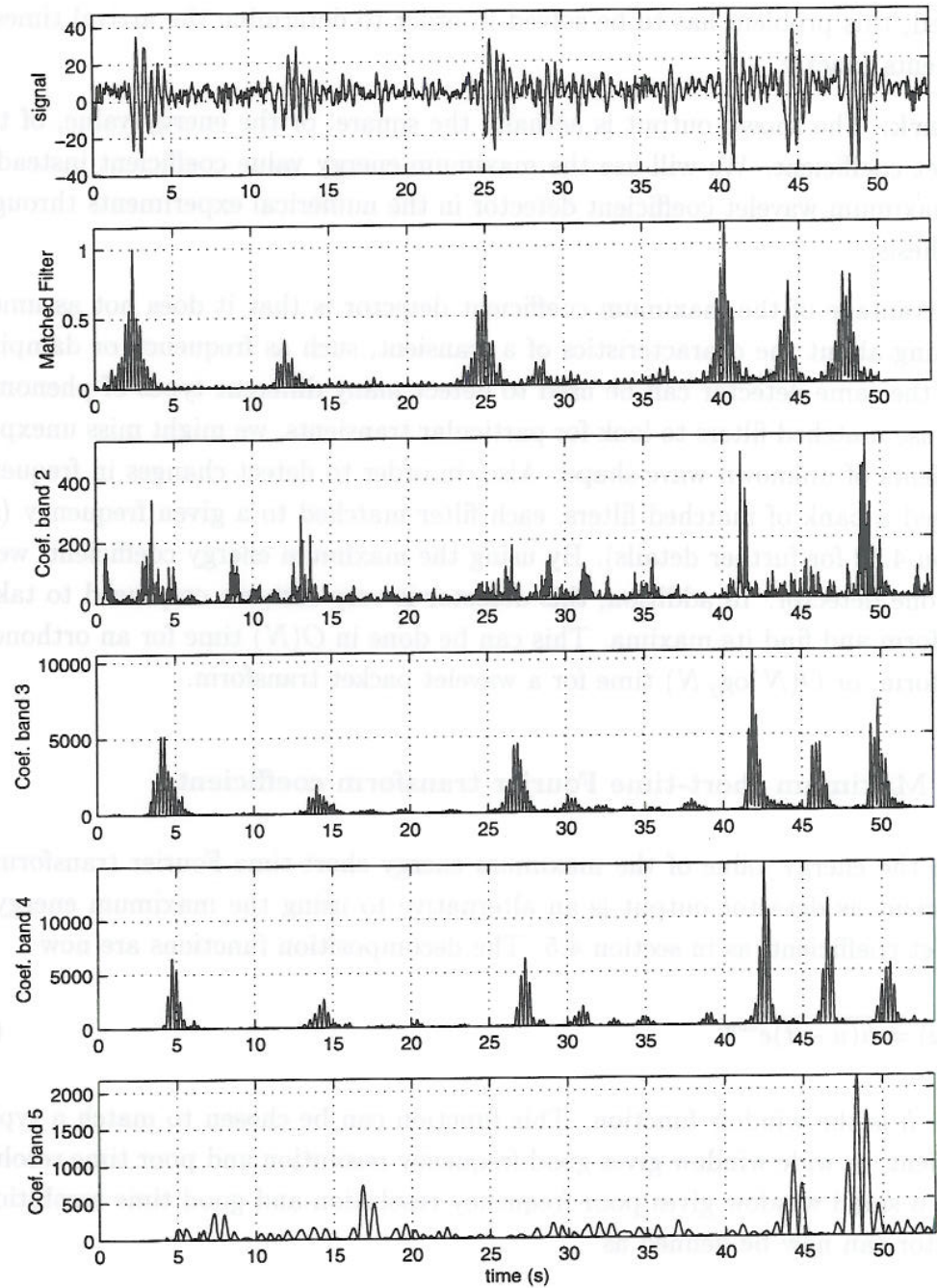
Figure 4.1: Comparison of the matched filter output (second plot) with the type 2 maximum energy wavelet transform coefficient detector of equation (4.21) (plots 3-6), using the Daubechies 6 wavelet. The empirical 2 Hz transient of figure 2.4 was used as the reference signal in the matched filter

where $q = q_0 + 2^j q_1$, $q_0 \in \{0, 1, ..., 2^j - 1\}$. The output of this detector is shown in figure 4.1. We clearly see a time delay between the matched filter output and the above detector. The phase shift increases with scale j. If the above detector is to be used, this problem has to be solved in order to determine the arrival times of transients exactly.

Remark. The shown output is actually the square, or the energy value, of the wavelet coefficient. We will use the maximum energy value coefficient instead of the maximum wavelet coefficient detector in the numerical experiments throughout this thesis.

An advantage of the maximum coefficient detector is that it does not assume anything about the characteristics of a transient, such as frequency or damping. Thus the same detector can be used to detect many different types of phenomena. If we use matched filters to look for particular transients, we might miss unexpected transients of unknown wave-shape. Also, in order to detect changes in frequency, we need a bank of matched filters, each filter matched to a given frequency (see section 4.12 for further details). By using the maximum energy coefficient, we only need one detector. In addition, this detector is very fast, we only need to take a transform and find its maxima. This can be done in $O(N)$ time for an orthonormal transform, or $O(N \log_2 N)$ time for a wavelet packet transform.

4.6 Maximum short-time Fourier transform coefficient

Using the energy value of the maximum energy short-time Fourier transform coefficient as detector output is an alternative to using the maximum energy wavelet coefficient, as in section 4.5. The decomposition functions are now

$$h_{\omega,t}(u) = \overline{h(u - t)e^{i\omega u}}, \tag{4.22}$$

where h is the window function. This function can be chosen to match a typical transient. A wide window gives good frequency resolution and poor time resolution, while a small window gives poor frequency resolution and good time resolution. A detector can now be defined as

$$
\begin{aligned}
y &= \max_{\omega,t} (T^{STFT} f)(\omega, t) \\
&= \max_{\omega,t} \int_{-\infty}^{\infty} f(u) h(u - t) e^{-i\omega u} du.
\end{aligned}
\tag{4.23}
$$

As in the previous section, this corresponds to finding the maximum peak of a bank of parallel matched filters, each filter matched to a given frequency ω.

A matched filter output depending on time can be defined as

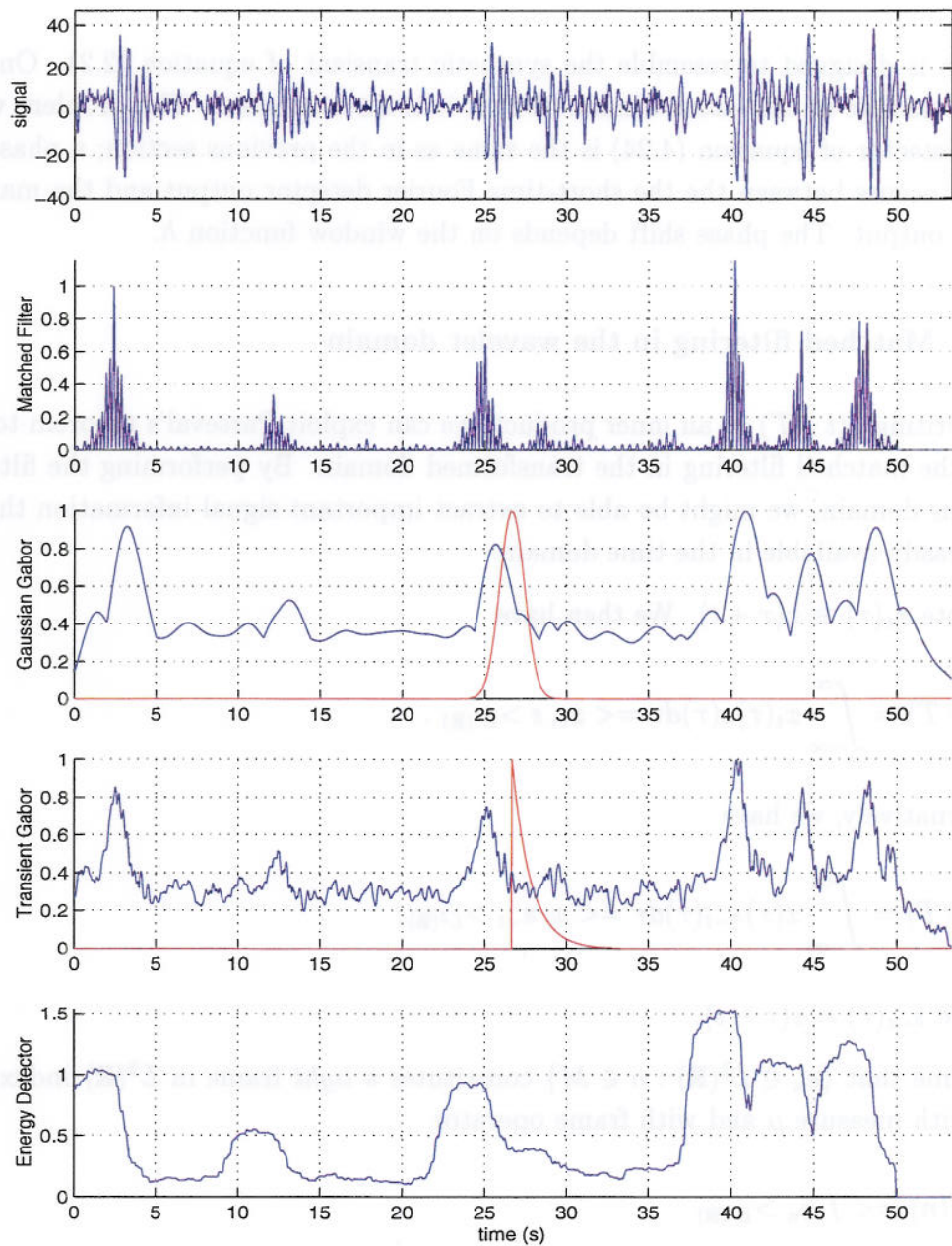$$y(t) = \max_{\omega}(T^{STFT}f)(\omega, 0). \tag{4.24}$$

Figure 4.2: Comparison of the matched filter with the short time Fourier and energy detectors. The windows in the short time Fourier transforms are indicated in red. The empirical 2 Hz transient of figure 2.4 was used as the reference signal in the matched filter

A discretization of this detector was tested on the data of figure 2.3, and the results of that test are in figure 4.2. The window functions used were a Gaussian window and a window of the type

$$h(t) = A \exp(-\alpha t) H(t), \tag{4.25}$$

which is designed to resemble the synthetic transient of equation (2.2). One disadvantage of the latter window is that it is discontinuous. The problem with the detector of equation (4.24) is the same as in the previous section; a phase shift often occurs between the the short-time Fourier detector output and the matched filter output. The phase shift depends on the window function h.

4.7 Matched filtering in the wavelet domain

By writing $y(t+T)$ as an inner product, we can exploit Parseval's theorem to carry out the matched filtering in the transformed domain. By performing the filtering in this domain, we might be able to extract important signal information that is not easily available in the time domain.

Denote $x_t(\tau) \equiv x(\tau + t)$. We then have

$$y(t+T) = \int_{-\infty}^{\infty} x_t(\tau) s(\tau) d\tau = <x_t, s>_{L^2(\mathbb{R})}. \tag{4.26}$$

Alternatively, we have

$$y(t+T) = \int_{-\infty}^{\infty} x(\tau) s_{-t}(\tau) d\tau = <x, s_{-t}>_{L^2(\mathbb{R})}, \tag{4.27}$$

where $s_{-t}(\tau) = s(\tau - t)$.

Assume that $\{e_n \in L^2(\mathbb{R}) : n \in M\}$ constitutes a *tight* frame in $L^2(\mathbb{R})$ indexed by M with measure μ and with frame operator

$$(Tf)(n) = <f, e_n>_{L^2(\mathbb{R})} \tag{4.28}$$

for $f \in L^2(\mathbb{R})$ (see section 3.4). Assume the frame bound is A, so that

$$\|f\|_H^2 = A^{-1} \|Tf\|_{L^2(\mu)}^2. \tag{4.29}$$

Parseval's theorem (equation 3.20) states

$$<f, g>_{L^2(\mathbb{R})} = A^{-1} <Tf, Tg>_{L^2(\mu)} \tag{4.30}$$

for $f, g \in L^2(\mathbb{R})$.

By Parseval's theorem we have

$$
\begin{aligned}
y(t+T) &= <x_t, s>_{L^2(\mathbb{R})} \\
&= A^{-1} <Tx_t, Ts>_{L^2(\mu)} .
\end{aligned}
\tag{4.31}
$$

Thus the matched filtering can be performed as an inner product in the transformed domain. The transformed domain can be *any* tight frame, for example a continuous wavelet transform, a short-time Fourier transform, a discrete wavelet transform or a wavelet packet transform.

4.7.1 Continuous wavelet transform matched filtering

For the continuous wavelet transform, we have $M = \{a, b \in \mathbb{R}, a \neq 0\}$, equipped with the measure $d\mu = dadb/a^2$, as described in section 3.4.2. Denoting the frame operator by

$$
(Tf)(a, b) = <f, \psi_{a,b}>_{L^2(\mathbb{R})},
\tag{4.32}
$$

we have the following matched filtering equation

$$
\begin{aligned}
y(t+T) &= C_\psi^{-1} <Tx_t, Ts>_{L^2(\mu)} \\
&= C_\psi^{-1} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} Tx_t(a, b)\overline{Ts(a, b)}\frac{dadb}{a^2} \\
&= C_\psi^{-1} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} Tx(a, b+t)\overline{Ts(a, b)}\frac{dadb}{a^2} .
\end{aligned}
\tag{4.33}
$$

The matched filtering is performed by matching the pattern of the transform of the transient s to the transform of the signal x.

The continuous wavelet transform does not have a fast implementation, which makes it less attractive for signal processing tasks involving large amounts of data, especially when the calculations are to be performed real time. We will therefore focus on matched filtering using a fast discrete transform with an orthonormal basis.

4.7.2 Discrete wavelet transform matched filtering

Assuming $\{e_n\}$ is an orthonormal basis for $L^2(\mathbb{R})$, Parseval's theorem reduces to

$$
<f, g>_{L^2(\mathbb{R})} = <Tf, Tg>_{l^2(\mathbb{Z})},
\tag{4.34}
$$

where T is the frame operator. The matched filtering in the transform domain takes the form

$$
\begin{aligned}
y(t+T) &= <Tx_t, Ts>_{l^2(\mathbb{Z})} \\
&= \sum_n Tx_t(n)\overline{Ts(n)}.
\end{aligned}
\tag{4.35}
$$

For a J-level multiresolution decomposition with finest scale $j = 0$, this reduces to

$$
y(t+T) = \sum_{k=\infty}^{\infty}(T_\phi x_t)(J,k)\overline{(T_\phi s)(J,k)} + \sum_{j=1}^{J}\sum_{k=\infty}^{\infty}(T_\psi x_t)(j,k)\overline{(T_\psi s)(j,k)}.
\tag{4.36}
$$

In practice, the input signal x is given at discrete times $t_q = q\Delta t$, where $q = 0, 1, 2, ...$ For simplicity, we assume $\Delta t = 1$. This can be obtained by rescaling the time scale of the the input data before sampling. Thus the output from the matched filter will come at discrete times $t = q$. Observing $x_q(\tau) = x(\tau + q) = S^q x(\tau)$ and using theorem 9 we obtain

$$
\begin{aligned}
y(q+T) = \quad & \sum_{k=\infty}^{\infty}(T_\phi S^q x)(J,k)\overline{(T_\phi s)(J,k)} \\
+ & \sum_{j=1}^{J}\sum_{k=\infty}^{\infty}(T_\psi S^q x)(j,k)\overline{(T_\psi s)(j,k)} \\
= \quad & \sum_{k=\infty}^{\infty}(T_\phi S^{q_0(J)}x)(J,k+q_1(j))\overline{(T_\phi s)(J,k)} \\
+ & \sum_{j=1}^{J}\sum_{k=\infty}^{\infty}(T_\psi S^{q_0(j)}x)(j,k+q_1(j))\overline{(T_\psi s)(j,k)},
\end{aligned}
\tag{4.37}
$$

where $q_0(j) \in \{0, 1, ..., 2^j - 1\}$ and $q = q_0(j) + 2^j q_1(j)$.

Matched filtering in this transformed domain is compared to the usual time domain matched filtering in figure 4.3. The matched filtering was performed with the empirical transient of figure 2.4. The wavelet matched filtering was performed using all the coefficients of a level 5 Daubechies 6 decomposition. The figure shows almost identical filter outputs. Performing matched filtering in the transformed domain is equivalent to matched filtering in the time domain. The choice of wavelet is irrelevant when all coefficients are used.

When this matched filtering is performed, we need to calculate the transform of all possible shifts q of the input signal x. By the results of section 3.8, this can be done in $O(JN)$ time, creating $(J + 1)N$ coefficients. A way to reduce computing time and storage need is to shift the transient s *instead* of the signal x, since the
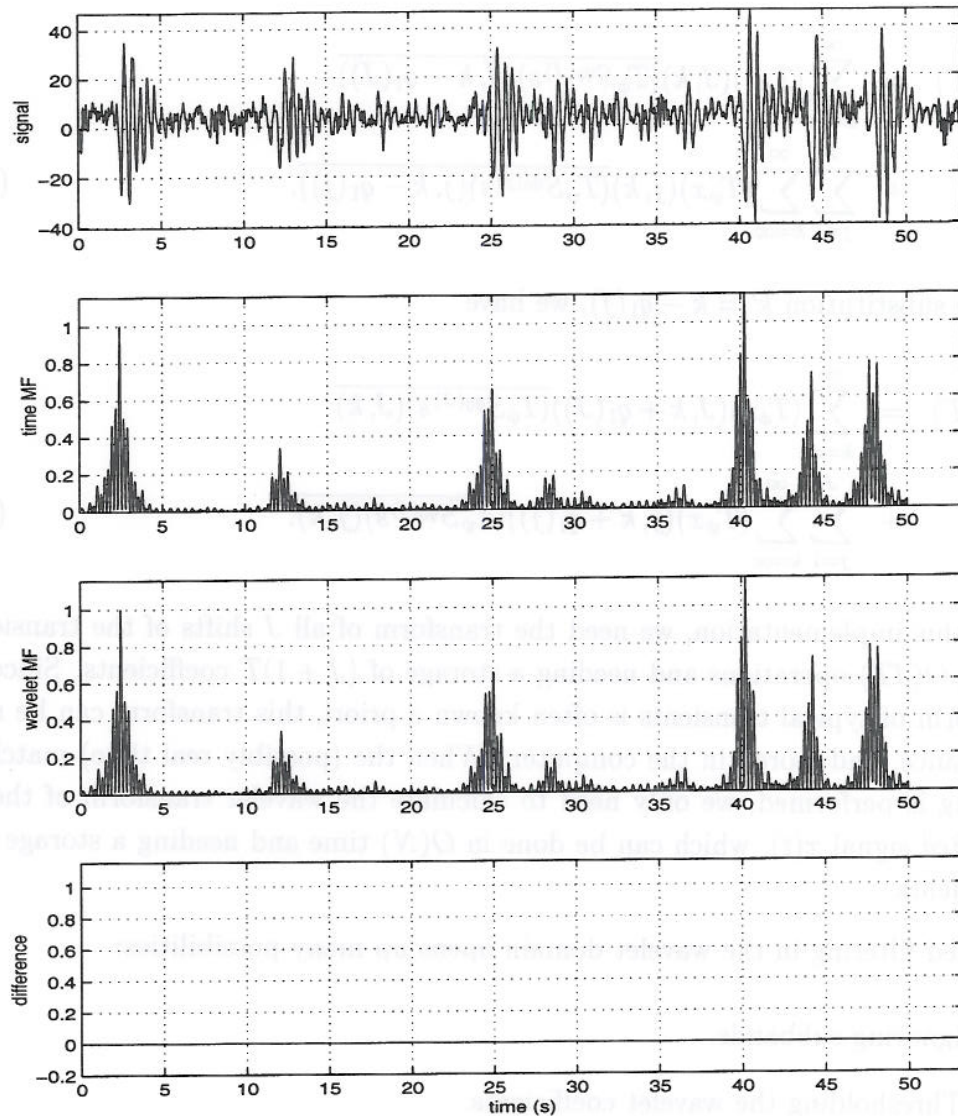
Figure 4.3: *Matched filtering in the time domain compared to matched filtering in the discrete wavelet transform domain using the Daubechies 6 wavelet. The reference signal is the empirical 2 Hz transient of figure 2.4*

transient usually is of much shorter length than the signal itself. We start with formula (4.27), given by

$$y(t + T) = < x, s_{-t} >_{L^2(\mathbb{R})} .$$
$$(4.38)$$

The above analysis is exactly the same, resulting in

$$
\begin{aligned}
y(q + T) &= \sum_{k=\infty}^{\infty} (T_\phi x)(J, k) \overline{(T_\phi S^{q_0(J)} s)(J, k - q_1(J))} \\
&+ \sum_{j=1}^{J} \sum_{k=\infty}^{\infty} (T_\psi x)(j, k) \overline{(T_\psi S^{q_0(j)} s)(j, k - q_1(j))}.
\end{aligned}
$$
$$(4.39)$$

By the substitution $k' = k - q_1(j)$, we have

$$
\begin{aligned}
y(q + T) &= \sum_{k=\infty}^{\infty} (T_\phi x)(J, k + q_1(J)) \overline{(T_\phi S^{q_0(J)} s)(J, k)} \\
&+ \sum_{j=1}^{J} \sum_{k=\infty}^{\infty} (T_\psi x)(j, k + q_1(j)) \overline{(T_\psi S^{q_0(j)} s)(j, k)}.
\end{aligned}
$$
$$(4.40)$$

With this implementation, we need the transform of all J shifts of the transient, taking $O(JT)$ operations and needing a storage of $(J + 1)T$ coefficients. Since the waveform of typical transients is often known a priori, this transform can be made in advance, and stored in the computer. When the (possibly real time) matched filtering is performed, we only need to calculate the wavelet transform of the *unshifted* signal $x(t)$, which can be done in $O(N)$ time and needing a storage of N coefficients.

Matched filtering in the wavelet domain opens up many possibilities:

- Ignoring subbands.

- Thresholding the wavelet coefficients.

- The choice of wavelet basis.

- Wavelet packets.

- M-band wavelets.

We discuss these possibilities in more detail.

Ignoring subbands

By using the information in all subbands, the matched filtering in the wavelet domain is equivalent to matched filtering in the time domain. However, the
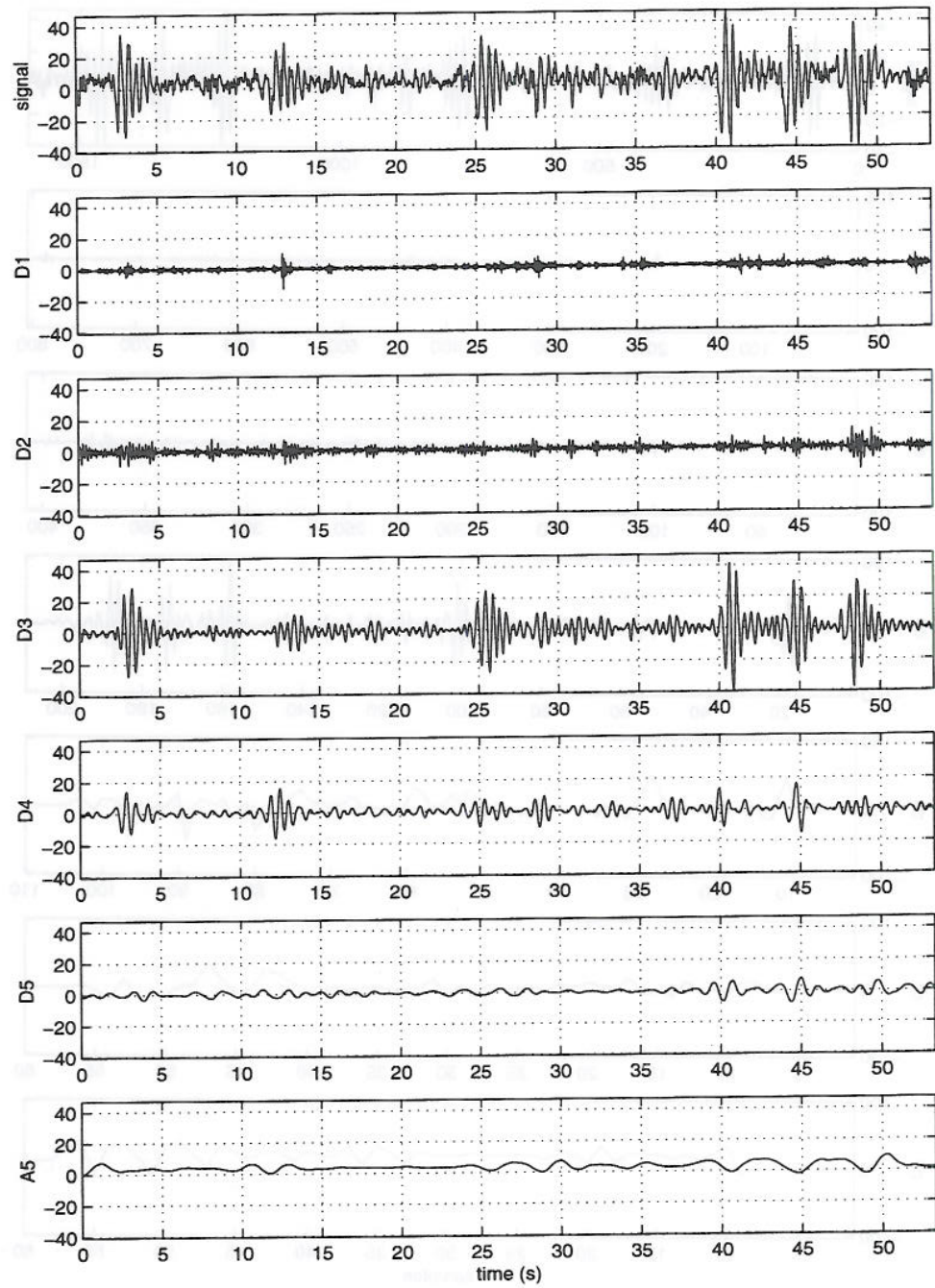
Figure 4.4: The discrete wavelet transform projections of the signal of figure 2.3, using the Daubechies 6 wavelet
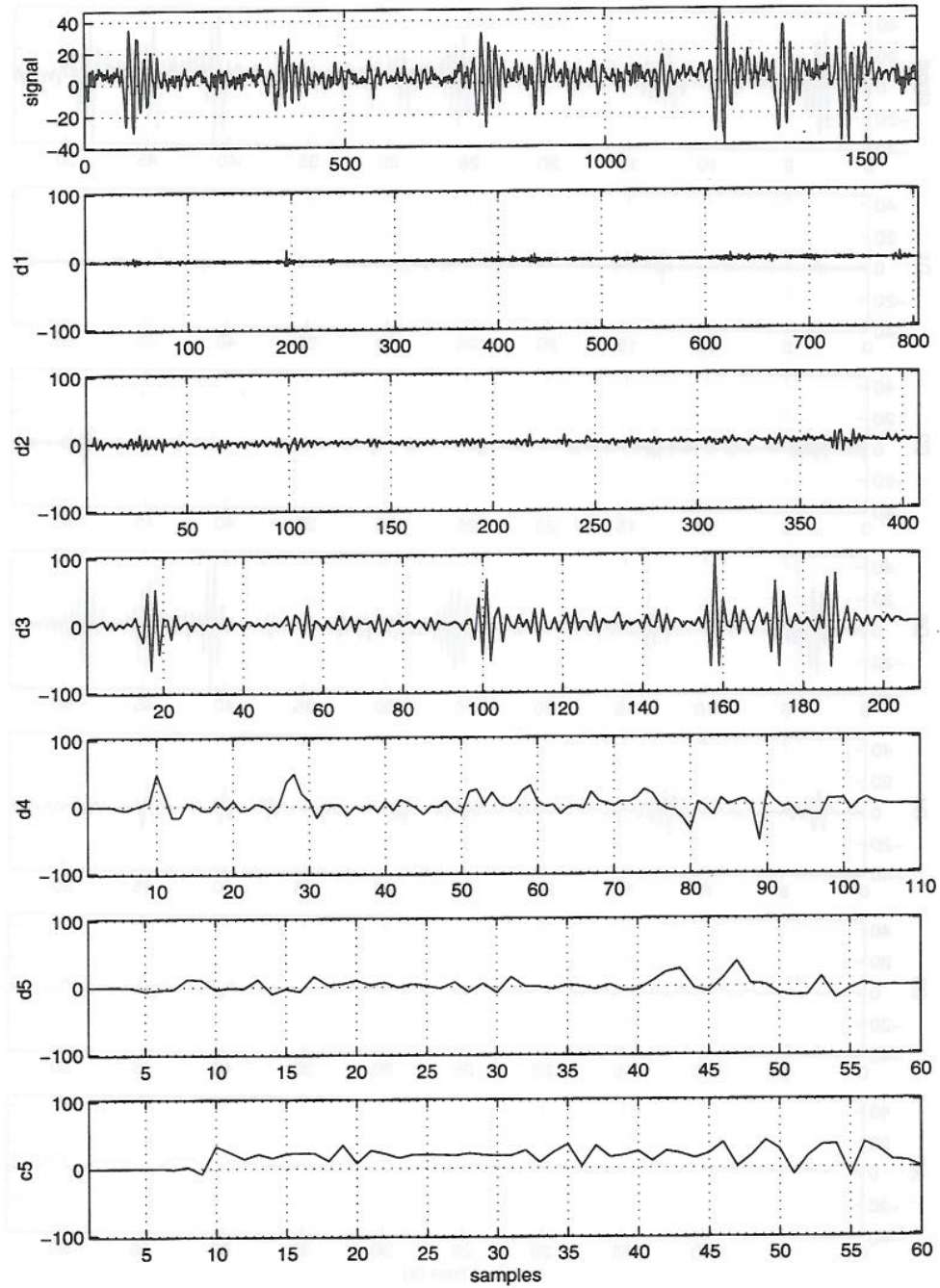
Figure 4.5: The discrete wavelet transform coefficients of the signal of figure 2.3, using the Daubechies 6 wavelet

transient often has a characteristic frequency, and it occurs at a certain scale. This implies that only a small number of the scales j will give a significant correlation between the transient and the wavelet. The discrete wavelet transform of the signal of figure 2.3, containing 2 Hz transients, is shown in figure 4.4 and figure 4.5. Figure 4.4 shows the projection of the signal onto the scale function space V_5 (denoted by A5) and the projections onto the wavelet spaces W_j, $j = 1, 2, 3, 4, 5$ (denoted by Di). In figure 4.5, the transform coefficients c_5 and d_j are plotted. The plots show that most of the transient energy is concentrated in bands 3 and 4.

Matched filtering using only the coefficients of particular subbands was performed on the signal of figure 2.3, using the empirical transient of figure 2.4. The results are in figure 4.6. The type of transform was a level 5 Daubechies 6 decomposition. We see that the detector performs well even when only subband 3 is used. The motivation for picking the Daubechies 6 wavelet is given in section 5.3.

There are several advantages of using only one or a few of the subbands in the matched filtering:

- By using the whole orthonormal wavelet transform in calculating the matched filter output, we must calculate the inner product with approximately T coefficients. By using only a few subbands, only a fraction of those coefficients needs to be processed.

- Separate transient phenomena often occurs at different scales. If these scales are far enough apart, they will not interfere with each other when calculating the matched filter using only some subbands. This idea is illustrated in figure 4.7. The discrete wavelet transform of the 20 Hz transient along with the 2 Hz transient is shown in figure 4.8. We see how the two transients appear at different scales in the wavelet transform. The 20 Hz transient seems to appear mostly at scales 1, 2 and 3. Also notice that two small 20 Hz transients appears inside the 2 Hz transient (at $t \approx 1.7s$ and $t \approx 2.9s$). The 20 Hz transient of figure 2.6 was picked as the reference signal, and a full matched filter was run on the signal in figure 4.8, along with a matched filter using only subbands 1, 2 and 3 and a matched filter using subband 3. Figure 4.9 presents the result of the simulation. We see that the two small 20 Hz transients are clearly detected by the subband matched filters, while the total matched filter gives a more noisy output. The matched filter using only subband 2 seems to have the best performance. One of the reasons why the subband matched filter outperforms the full matched filter in this case, is that the scales that contain most of the 2 Hz transient energy (i.e. scales 5 and 6) are not used. This transient is not interfering with the detection of the 20 Hz transient when we avoid subbands 5 and 6. On the downside, the peak becomes slightly wider when using fewer subbands. However, the *top* of the first peak, not shown in the figure, is quite sharp for all three cases.
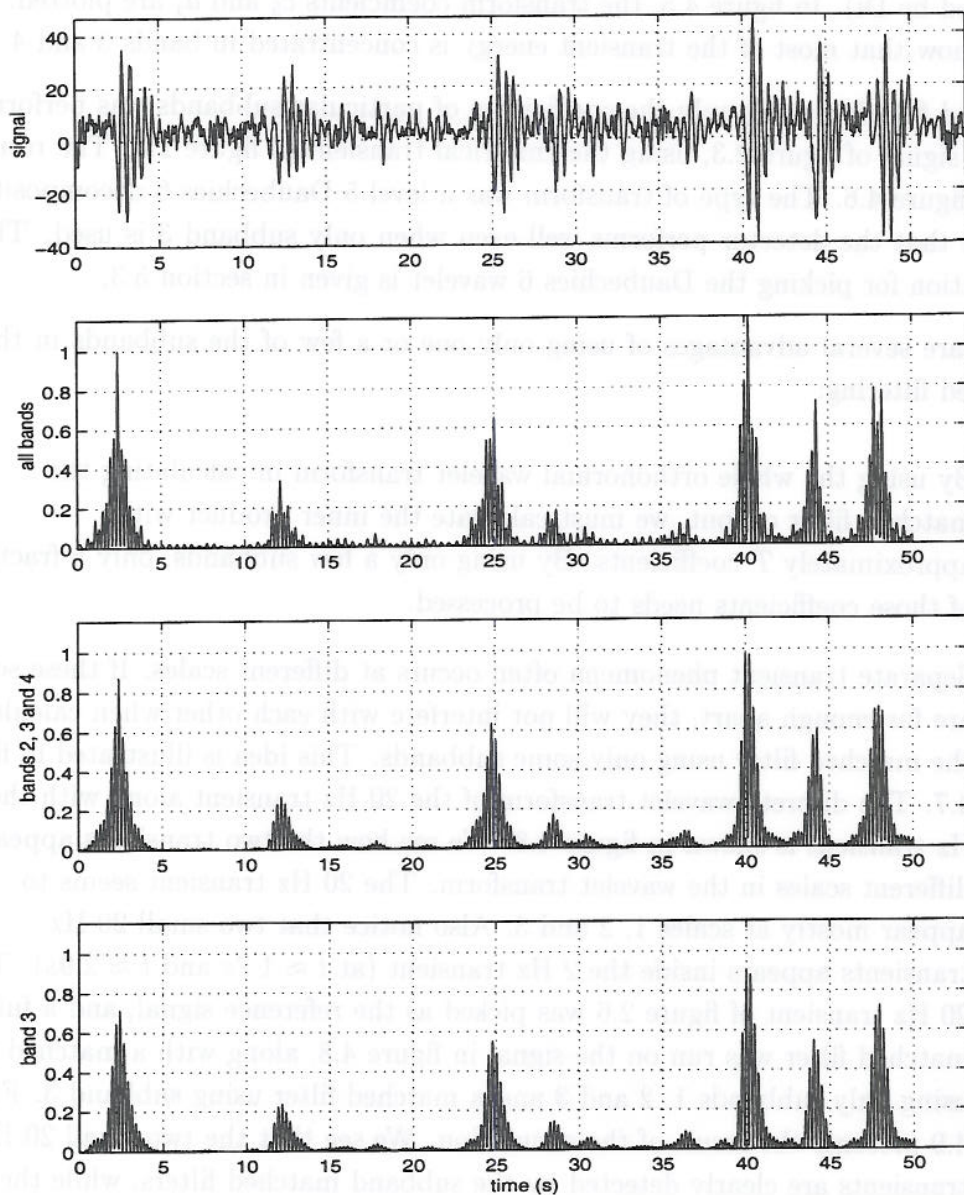
Figure 4.6: Matched filtering using only some of the subbands in the Daubechies 6 wavelet decomposition. The reference signal is the empirical 2 Hz transient of figure 2.4
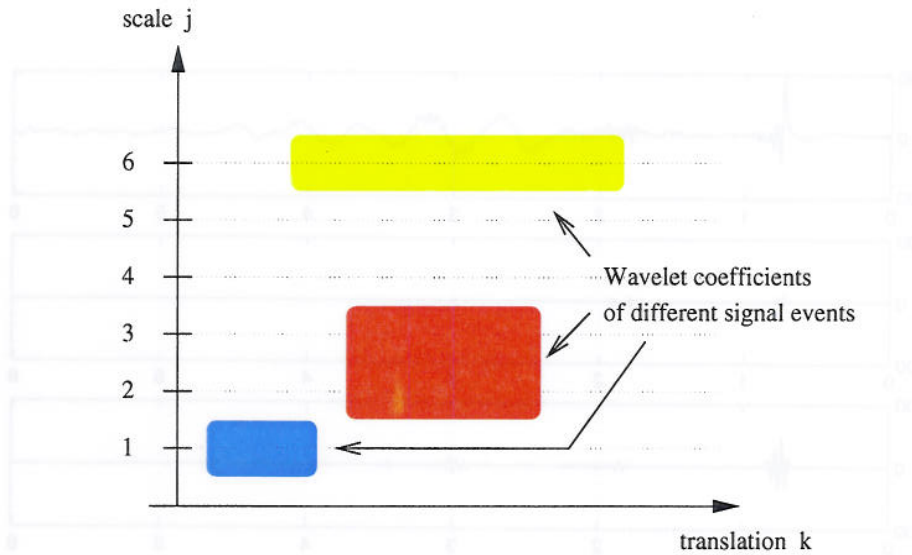
Figure 4.7: Schematic illustration on how different events might occur at different scales. The shaded areas represent the location containing most of the wavelet coefficients of a particular event

- In the ideal case of white noise, a perfect correlator is optimal. In practice, the input noise might not be white, and using only a few subbands might outperform the all-band correlator.

Thresholding

Instead of completely ignoring the coefficients at certain scales, one can apply a thresholding of the coefficients. The threshold level ϵ may differ at each scale. Hard thresholding a coefficient $x \in \mathbb{R}$ consists of setting to zero the coefficient that is below the threshold,

$$h(x) = xH(|x| - \epsilon). \tag{4.41}$$

In soft thresholding one also shrinks the remaining coefficients by a level ϵ, i.e.

$$s(x) = sign(x)\{|x| - \epsilon\}_+. \tag{4.42}$$

Donoho (1995) shows that soft-thresholding of the wavelet coefficients is near optimal in the removal of white Gaussian noise.

In many applications one performs an inverse transform after thresholding the coefficients. This means that the wavelet transform is being used as a *preprocessor* to filter the data, as in figure 4.10. The data can then be analyzed in the time domain. In our case, it is unnecessary to take the inverse transform. The
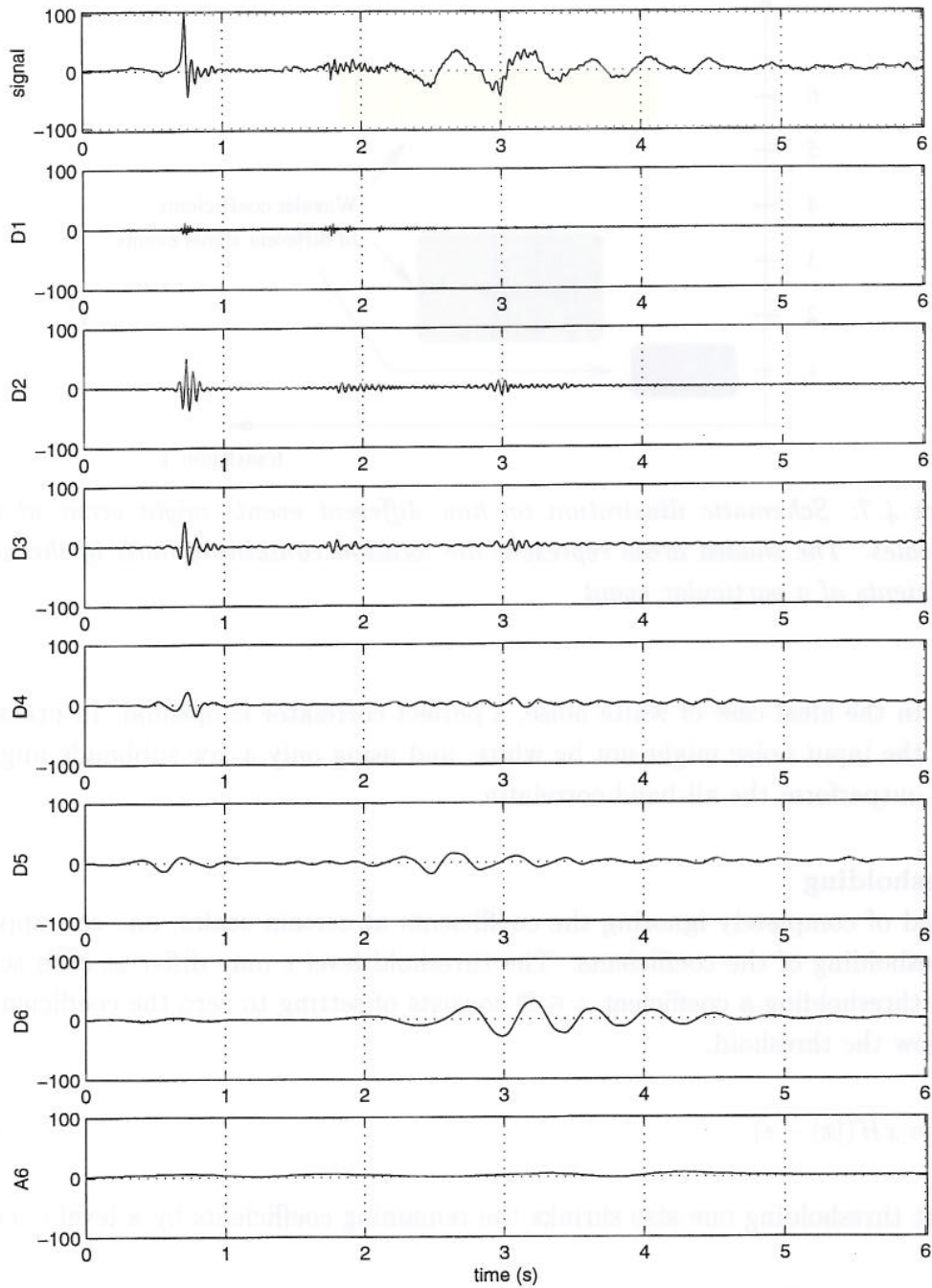
Figure 4.8: The discrete wavelet transform projections of a signal containing both the 20 Hz and the 2 Hz transient, using the Daubechies 6 wavelet. (The sample rate for this signal is 5 times higher than for the other signals used in this thesis. This is the reason why the 2 Hz transient appears in subbands 5 and 6 in this plot, while it appears in bands 2-4 in the other plots in this thesis)

70
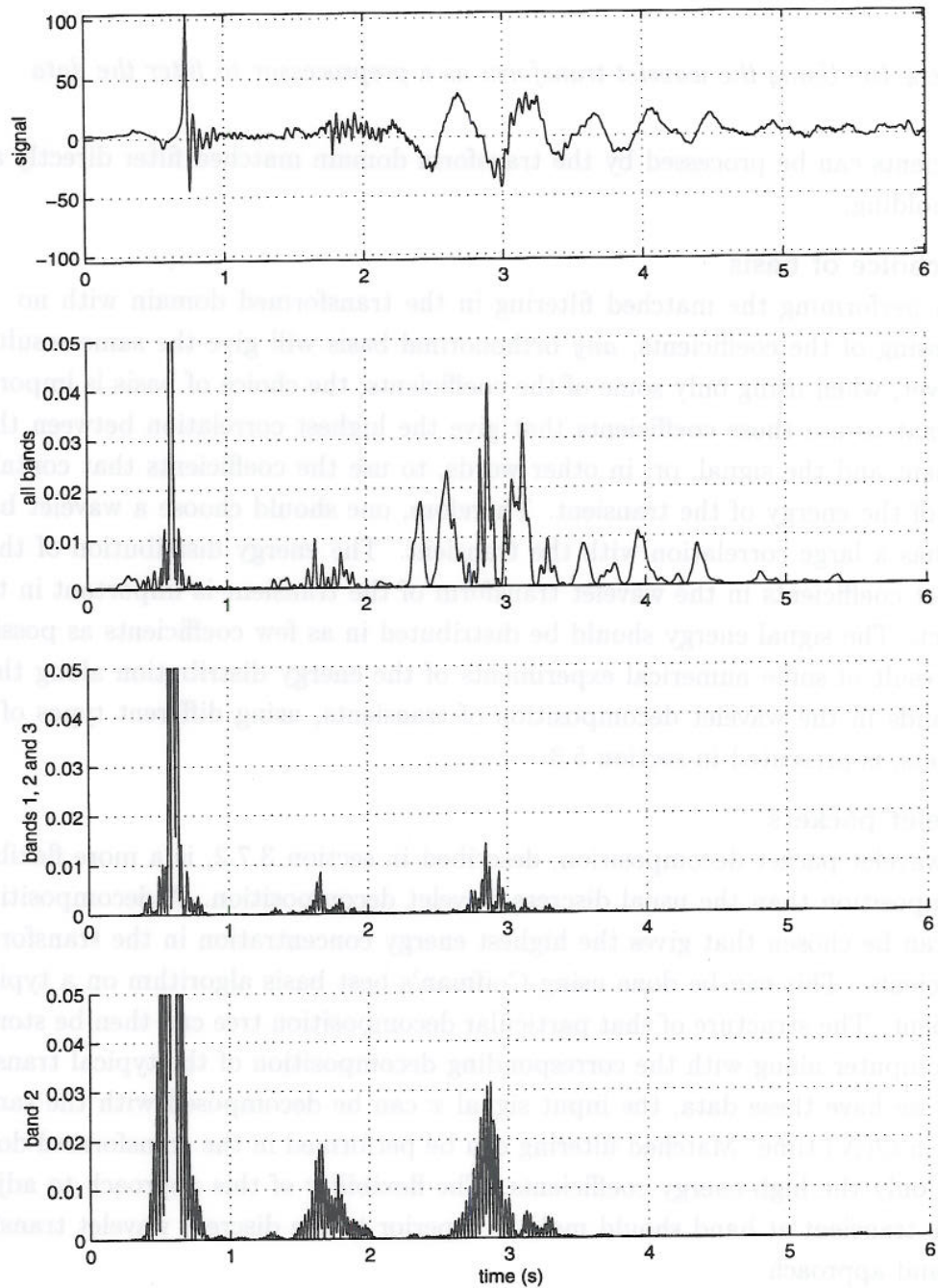
Figure 4.9: Matched filter outputs using only some of the subbands in a Daubechies 6 wavelet decomposition. The reference signal is the 20 Hz transient of figure 2.6. The plots are scaled in order to show the detection at t ≈ 1.7s and t ≈ 2.9s. The first transient has a top of 0.228, 0.228 and 0.377 for the all band, band 1-3 and band 2 matched filters, respectively
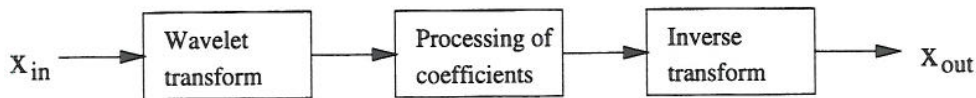
Figure 4.10: Using the wavelet transform as a preprocessor to filter the data

coefficients can be processed by the transform domain matched filter directly after thresholding.

The choice of basis

When performing the matched filtering in the transformed domain with no processing of the coefficients, *any* orthonormal basis will give the same result. However, when using only some of the coefficients, the choice of basis is important. We want to use those coefficients that give the highest correlation between the transient and the signal, or, in other words, to use the coefficients that contain most of the energy of the transient. Therefore, one should choose a wavelet basis that has a large correlation with the transient. The energy distribution of the wavelet coefficients in the wavelet transform of the transient is important in this respect. The signal energy should be distributed in as few coefficients as possible. The result of some numerical experiments of the energy distribution along the subbands in the wavelet decomposition of transients, using different types of wavelets, is presented in section 5.3.

Wavelet packets

The wavelet packet decomposition, described in section 3.7.2, is a more flexible decomposition than the usual discrete wavelet decomposition. A decomposition tree can be chosen that gives the highest energy concentration in the transformed coefficients. This can be done using Coifman's best basis algorithm on a typical transient. The structure of that particular decomposition tree can then be stored in the computer along with the corresponding decomposition of the typical transient. Once we have these data, the input signal x can be decomposed with the same basis in $O(N)$ time. Matched filtering can be performed in the transformed domain using only the high-energy coefficients. The flexibility of this approach to adjust to the transient at hand should make it superior to the discrete wavelet transform subband approach.

M-band wavelets

An M-band wavelet basis (see section 3.7.1) is another possibility for improving the energy concentration in the coefficients. For even better performance, it can be combined with a wavelet packet transform.

4.8 Receiver operating characteristics

A way of measuring the efficiency of a detector algorithm, is plotting the probability of detection,

$$P_D = p(detection \mid H_1) \tag{4.43}$$

as a function of the probability of false alarm,

$$P_{FA} = p(detection \mid H_0). \tag{4.44}$$

This gives rise to the so called receiver operating characteristics plots. Of course, both P_D and P_{FA} depend on the signal-to-noise ratio, and they may also depend of the type of noise and the type of signal.
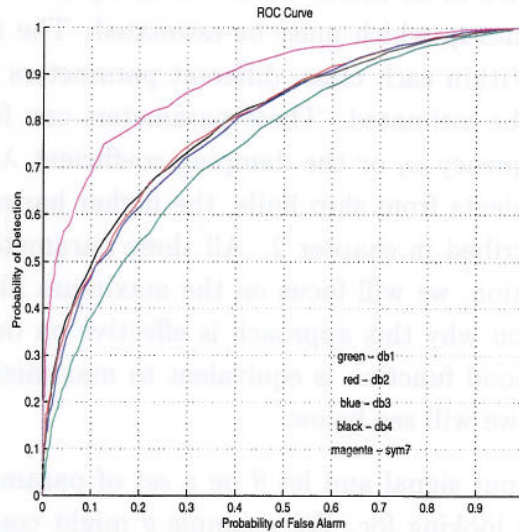

Figure 4.11: ROC curves for the maximum transform coefficient detector in the detection of the synthetic transient of figure 2.4. A level 4 decomposition was performed using the wavelets db1, db2, db3, db4 and sym7

The ROC curves can be simulated numerically by Monte Carlo simulation. In this simulation, one first decides on a signal to be recognized and on a predefined signal-to-noise ratio. One then makes R realizations of noise only and calculates the number of false alarms out of these R realizations at several different threshold levels. R realizations of signal and noise are also made, and the number of detections at different threshold levels are counted.

Since we have as many simulations with transient as we have with noise only, simply picking H_0 or H_1 at random at a 50-50 chance will give a detector performance of $P_{FA} = P_D$. Thus this straight line is the worst possible detector performance.

The best possible detector performance is described by $P_{FA} = 0, P_D = 1$. All our detectors will be positioned between those ideal cases. The position of the curves will depend on the input signal-to-noise ratio. If the energy of noise is large compared to the transient signal energy, the curves will be closer to the line $P_{FA} = 0, P_D = 1$. In the simulations, a noise amplitude that brings the curves into the center region of the plot was chosen. The important aspect is how the individual detectors performs compared to each other, i.e. how the curves are compared to other curves in the same plot. An example of a ROC curve is figure 4.11. The results of the simulations are presented and discussed in the next chapter.

4.9 Transient characterization

Until now we have focused on simply detecting whether a transient is present or not. In practice, we have to do more. There are many transients, and they arrive at unknown arrival times t_i, which must be estimated. The transients can belong to different classes. Within each class, different parameters that characterize the transient need to be estimated. These parameters can for example be the amplitude A_i, the frequency ω_i or the damping coefficient λ_i. In the detection of our particular transients from ship hulls, the higher harmonics $\omega_{i,k}$ are also of importance, as described in chapter 2. All these parameters may need to be estimated. In this section, we will focus on the maximum likelihood estimates of parameters. The reason why this approach is effective on our problem, is that maximizing the likelihood function is equivalent to maximizing the output from the matched filter, as we will see below.

Let $x(t)$ be the the input signal and let θ be a set of parameters characterizing the transient s we are looking for. For example θ might contain some or all of the parameters $(A, \omega, \lambda, t_0)$. Assume we start looking at the data at time t. Our hypothesis test is

$$H_0 \;:\; x(\tau + t) = n(\tau) \tag{4.45}$$
$$H_1 \;:\; x(\tau + t) = n(\tau) + s(\tau; \theta). \tag{4.46}$$

The maximum likelihood estimation approach is to maximize $p(x \mid H_1; \theta)$, called the likelihood function of the data, with respect to the parameters θ. The set of parameters $\hat{\theta}$ for which this function is maximum, are called the maximum likelihood estimators for θ.

Since $p(x \mid H_0)$ is independent of θ, we may maximize

$$\Lambda \equiv \frac{p(x \mid H_1)}{p(x \mid H_0)} \tag{4.47}$$

instead. We call Λ the *likelihood ratio*. Also, since $\ln u$ is an increasing function of u, this is equivalent to the maximization of $\ln \Lambda$.

We will show that this likelihood ratio is a function of the matched filter output. For this, we need the following result from Turin (1960).

Lemma 1 *The probability density of a sample $n(t)$ of white, Gaussian noise of spectral density $N_0/2$, lasting from a to b is given by*

$$p(n) = k \exp\left(-\frac{1}{N_0}\int_a^b n(\tau)^2 d\tau\right), \tag{4.48}$$

where k is independent of $n(t)$ but dependent on the interval $[a, b]$.

We now have

Theorem 11 *The likelihood ratio and the matched filter output are connected as follows,*

$$N_0 \ln \Lambda = 2y(t+T) - E, \tag{4.49}$$

where $y(t+T) = \int_{-\infty}^{\infty} x(\tau+t)s(\tau)d\tau$ is the matched filter output and $E = \|s\|_{L^2(\mathbb{R})}^2$ is the energy of the transient s.

Proof. Since the support of s is in $[0, T]$, our observation interval is $[0, T]$. Lemma 1 gives

$$p(x \mid H_0) = k \exp\left(-\frac{1}{N_0}\int_0^T x(\tau+t)^2 d\tau\right), \tag{4.50}$$

$$\begin{aligned} p(x \mid H_1) &= k \exp\left(-\frac{1}{N_0}\int_0^T (x(\tau+t) - s(\tau))^2 d\tau\right) \\ &= k \exp\left(-\frac{1}{N_0}\int_0^T x(\tau+t)^2 d\tau + \frac{2}{N_0}\int_0^T x(\tau+t)s(\tau)d\tau\right) \\ &\quad \times \exp\left(-\frac{1}{N_0}\int_0^T s(\tau)^2 d\tau\right). \end{aligned} \tag{4.51}$$

We have $\int_0^T s(\tau)^2 d\tau = \|s\|_{L^2(\mathbb{R})}^2 = E$. Thus

$$\ln \Lambda = \ln \frac{p(x \mid H_1)}{p(x \mid H_0)} = \frac{2}{N_0}\int_0^T x(\tau+t)s(\tau)d\tau - \frac{E}{N_0}. \tag{4.52}$$

Since the whole transient is inside $[0, T]$, we can change the interval of integration in (4.52) from $[0, T]$ to $\mathbb{R}$, obtaining the theorem.

$\square$

4.10 Estimation of arrival time

We seek the maximum likelihood estimate of the arrival time t_0 of a transient. That is, we want to maximize the likelihood ratio Λ with respect to arrival time. By theorem 11,

$$N_0 \ln \Lambda = 2y(t + T) - E. \tag{4.53}$$

The energy E of a transient is independent of arrival time. Maximizing Λ is therefore the same as finding the maximum of the matched filter output $y(t + T)$. If that maximum occurs at $t = t_0$, this is our maximum likelihood estimate $\hat{t_0}$ of the arrival time.

Figure 4.12 shows the matched filter output from a sequence of the experimental data. One problem in detecting the maximum output value, is that the matched filter output oscillates near t_0. The matched filter output is given by

$$y(t + T) = \int_{-\infty}^{\infty} x(t + \tau) s(\tau) d\tau. \tag{4.54}$$

When the transient is matched exactly to the signal, a maximum will occur. When the phase difference between the reference transient and the signal is π, we will get a large negative value. At a phase difference of 2π, we will have a large positive value again, and so forth. Since the particular transient we are looking at is oscillatory, the detector output also becomes oscillatory. Because of this, we have to use caution when detecting transients. If we claim a detection every time we have a local maximum in the matched filter output exceeding a certain threshold level, we might get several detections resulting from a single transient. Of course, in general, not all transients are oscillatory. This problem only appears for oscillatory signals, and the problem becomes larger the more oscillatory the signal.

There are many possible ways of dealing with this problem. The author has two proposals. One way is just using caution when the threshold is surpassed. If there is a nearby maximum that is higher, that maximum is chosen instead, and the first maximum is discarded, even if the threshold level was exceeded at that maximum too. For this method to work, we must know the typical width of a matched filter peak. An alternative is to take the wavelet transform of the matched filter output, process the coefficients in order to remove the rapid oscillations, and then take an inverse transform. A Daubechies 6 wavelet transform was performed on the matched filter output, and the first 4 detail coefficient bands were set to zero. An inverse transform was then performed. The result of this process is shown in figure 4.12.

In the case of no noise, the matched filter output is symmetric around the time of arrival. One should be able to incorporate this piece of knowledge when
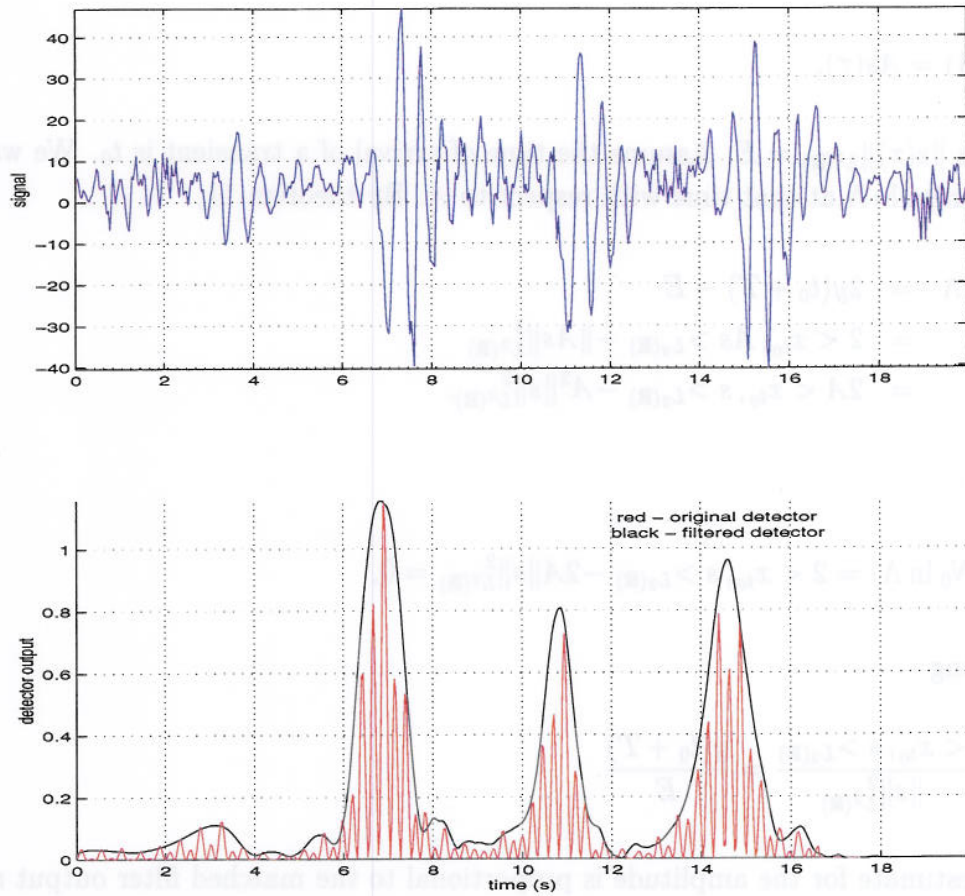
Figure 4.12: Matched filter output (blue) and the same output filtered with a Daubechies 6 wavelet transform (red)

determining the maximum filter output. If for instance the wavelet filtering is chosen, it is probably a good property if the wavelet is as symmetric as possible.

We will not dwell further on the problem of oscillatory filter output, but leave it for future work.

4.11 Estimation of amplitude

For the estimation of amplitude, assume

$$s(\tau; A) = As(\tau), \qquad (4.55)$$

where $\|s(\tau)\|_{L^2(\mathbb{R})} = E$. Assume the time of arrival of a transient is t_0. We want to maximize $\ln \Lambda$ at that time with respect to A. By theorem 11,

$$
\begin{aligned}
N_0 \ln \Lambda &= 2y(t_0 + T) - E \\
&= 2 < x_{t_0}, As >_{L_2(\mathbb{R})} - \|As\|_{L^2(\mathbb{R})}^2 \\
&= 2A < x_{t_0}, s >_{L_2(\mathbb{R})} - A^2 \|s\|_{L^2(\mathbb{R})}^2.
\end{aligned}
\qquad (4.56)
$$

Thus

$$\frac{\partial}{\partial A}(N_0 \ln \Lambda) = 2 < x_{t_0}, s >_{L_2(\mathbb{R})} - 2A\|s\|_{L^2(\mathbb{R})}^2 = 0, \qquad (4.57)$$

yielding

$$\hat{A} = \frac{< x_{t_0}, s >_{L_2(\mathbb{R})}}{\|s\|_{L^2(\mathbb{R})}^2} = \frac{y(t_0 + T)}{E}. \qquad (4.58)$$

The estimate for the amplitude is proportional to the matched filter output at the time of arrival, and the constant of proportion is E^{-1}. Under the hypothesis of zero mean white noise, the estimator has the correct expectation value,

$$E[\hat{A}] = A. \qquad (4.59)$$

We show this. Because s is a deterministic function, our stochastic variable is

$$x_{t_0}(\tau) = x(\tau + t_0) = n(\tau) + As(\tau). \qquad (4.60)$$

Since the noise is zero-mean, we have

$$E[x_{t_0}(\tau)] = As(\tau) \qquad (4.61)$$

so that

$$E(\hat{A}) = E^{-1}E[y(t_0 + T)] = E^{-1}\int_0^T E[x_{t_0}(\tau)]s(\tau)d\tau$$
$$= E^{-1}A\int_0^T s(\tau)^2 d\tau = A. \tag{4.62}$$

$\square$

In practice, the time of arrival is not known, and has to be estimated too. In that case, we use that estimate in our estimate for the amplitude,

$$\hat{A} = \frac{y(\hat{t}_0 + T)}{E}. \tag{4.63}$$

4.11.1 Estimation of amplitude from subbands

We might want to obtain an estimate of the amplitude from the output of an approximate matched filter using only some of the transform coefficients.

The maximum likelihood estimate for the amplitude in the transform domain takes the form

$$\hat{A} = \frac{y(t_0 + T)}{E} = \frac{\sum_{n \in U} Tx_t(n)\overline{Ts(n)}}{\sum_{n \in U} |Ts(n)|^2}, \tag{4.64}$$

where the set U represents all the coefficients needed to completely characterize the signal.

If we use only some of the coefficients, say the coefficients in the set $V \subset U$, in our matched filtering, we must use the same set V in both the matched filter and the calculation of transient energy in order to get an estimate for A with the correct expectation value. Our estimate becomes

$$\hat{A}_V = \frac{\sum_{n \in V} Tx_t(n)\overline{Ts(n)}}{\sum_{n \in V} |Ts(n)|^2}. \tag{4.65}$$

It is easy to show that this estimate satisfies $E[\hat{A}_V] = A$. The estimation of amplitude is demonstrated in section 5.4.

4.12 Estimation of frequency and damping

An optimal filter matched to a signal $s(t)$ is also optimal for signals sharing the same waveform but differing in amplitude and time-delay. However, the matched

filter is not adaptable to signals of different waveforms, including frequency shifted signals or signals with different damping.

Assume we have a transient $s(t; \omega, \lambda)$ of frequency ω and damping λ. The matched filter output is

$$y(t + T; \omega, \lambda) = \int_{-\infty}^{\infty} x(t + \tau) s(\tau; \omega, \lambda) d\tau. \tag{4.66}$$

We want to find the maximum likelihood estimates of frequency and damping. Thus we have to maximize

$$N_0 \ln \Lambda(\omega, \lambda) = 2y(t + T; \omega, \lambda) - E. \tag{4.67}$$

with respect to ω and λ. We will assume that the class of transients $s(t; \omega, \lambda)$ is normalized to have the same energy E independently of ω and λ. We are then left to find

$$\max_{\omega, \lambda} y(t + T; \omega, \lambda), \tag{4.68}$$

where the max is taken over a possibly continuous set of admissible values of ω, λ. In practice, we cannot test for all values of those parameters along the real line, because that involves calculating infinitely many matched filter outputs, one for each pair (ω, λ). The usual way of solving this problem is resorting to a finite bank of matched filters, where each filter in the bank is matched to a transient with characteristics (ω_k, λ_k). The bank has to be spaced so densely that no transients are missed. The spacing depends on how fast the matched filter output of a filter matched to the signal $s(t; \omega, \lambda)$ falls off when the input signal has a transient with frequency $\omega + \Delta\omega$ and damping $\lambda + \Delta\lambda$.

We may have signals that differ in other characteristics than ω and λ. The signals may have totally different waveforms. In that case, a matched filter must be designed for each signal. The estimation of frequency is demonstrated in section 5.5.

5 EXPERIMENTAL RESULTS

In the last chapter, we reviewed techniques for transient detection and characterization, and some of them were demonstrated on experimental data. This chapter contains more practical examples. Section 5.1 presents ROC curves for the maxium wavelet transform coefficient detectors, while section 5.2 presents ROC curves for the matched filter detectors. In section 5.3, the energy distribution of the

wavelet coefficients of the 2 Hz transient using different wavelet decompositions is investigated. This distribution gives an indication on how well a particular wavelet is suited to the analysis of the given transient. In sections 5.4 and 5.5, examples of the estimation of amplitudes and frequencies of transients are given.

5.1 ROC curves for maximum transform coefficient detectors

The maximum transform coefficient detector is a detector of type 1. We have a fixed observation interval I, and want to test if we have a transient in that interval or not.

A wavelet transform was performed on the input x, and the maximum energy value of these coefficients was chosen as the detection statistic. The length of the observation interval was 300 samples. The signal to be detected was the synthetic transient of figure 2.4 using 100 samples, and its arrival time was set to sample number 100.

1000 simulations were performed with noise only, and 1000 simulations were performed using signal plus noise. The added noise was uncorrelated Gaussian, and each noise sample had mean zero and variance 0.18^2. The transient was normalized to have l^2-energy 1. The signal used in the simulation is plotted in figure 5.1, along with a typical noise realization and their sum.

ROC curves for orthonormal wavelet decompositions using members of the Daubechies and Symlet wavelet families are shown in figure 4.11. The Symlet 7 wavelet has the best performance. It also outperformed other members of the Daubechies, Coiflet and Symlet families (shown in figures C.1, C.2 and C.3 in appendix C).

Figure 5.2, indicates that we need to perform a decomposition at least down to the third level in order to get an acceptable detector. Some of the explanation for this lies in the energy distribution over the subbands in a discrete wavelet decomposition of the synthetic transient, as shown in figure 5.3. Most of the energy lies in band 3. Also, the Symlet 7 wavelet has the highest peak energy; more energy is concentrated in fewer subbands. This is one reason for why the Symlet 7 performed so well.

Del Marco and Weiss (1994a, 1994b) have found that a time invariant detector outperforms the usual time variant detector. An undecimated, time invariant wavelet transform was performed using the Daubechies 2 and Symlet 7 wavelets, and the maximum energy transform coefficient was employed as the detection statistic. The results, presented in figure 5.4, support the statement that the time invariant detector has superior performance.

As mentioned earlier, the degree of similarity between the transient and the wavelet
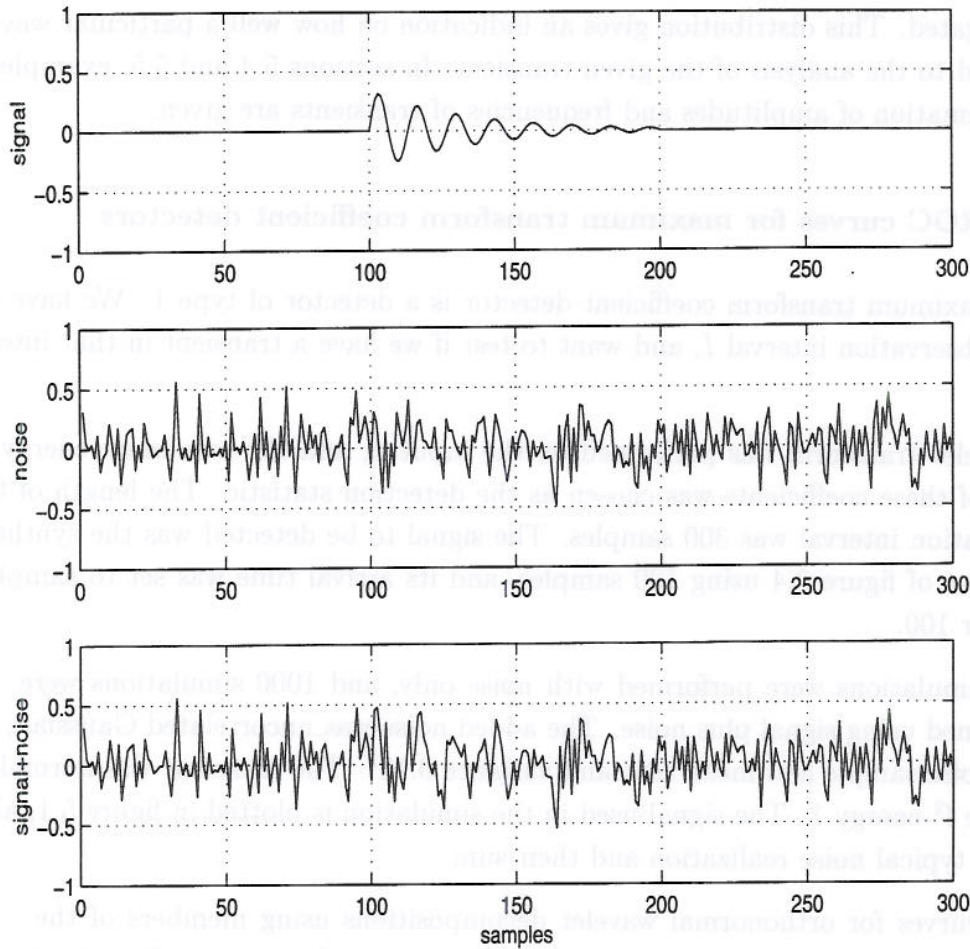
Figure 5.1: The synthetic signal and noise used in the generation the ROC curves for the maximum transform coefficient detectors

is important for the performance of the detector. A simulation similar to the above, but using the 102-sample empirical transient of figure 2.4 instead of the synthetic one, was performed. The results, shown in figure 5.5, show that the Daubechies 6 wavelet performs best in this case. The Symlet 7 is no longer best. Thus, when picking the wavelet for our detector, we have to take into consideration the type of transient we are looking for. One of the reasons why the Daubechies 6 wavelet performs best on the empirical transient, lies in its good energy distribution over the subbands. This is shown in section 5.3.
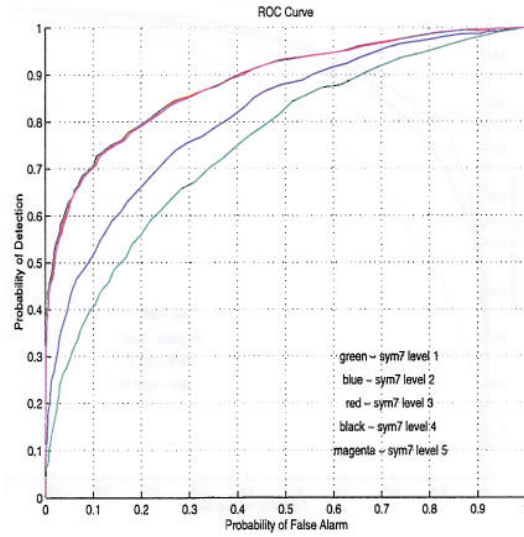
Figure 5.2: ROC curves for the maximum transform coefficient detector in the detection of the synthetic transient of figure 2.4. The decomposition was performed using the wavelet sym7, and the number of decomposition levels was 1, 2, 3, 4 and 5
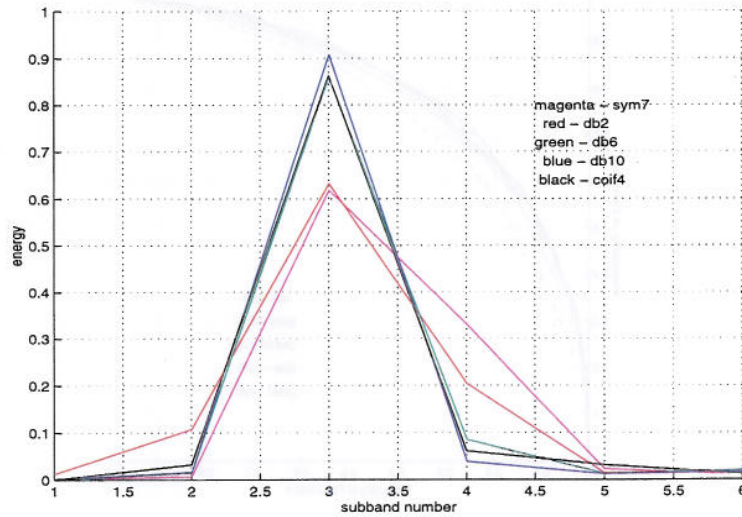
Figure 5.3: The subband energy distribution for the discrete wavelet decomposition of the synthetic transient of figure 2.4, using the db2, db10, coif4 and sym7 wavelets
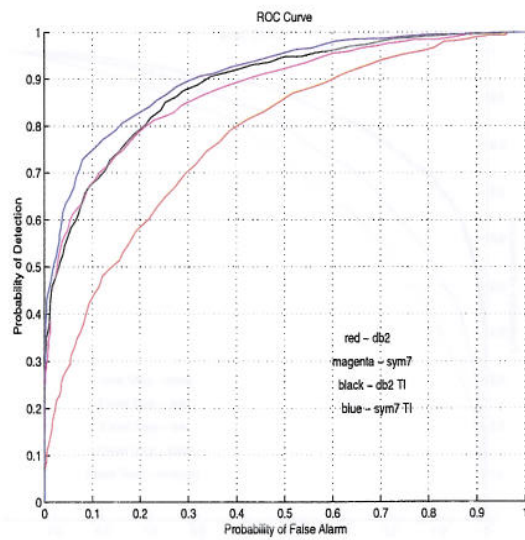
Figure 5.4: ROC curves for the maximum transform coefficient detector in the detection of the synthetic transient of figure 2.4. A detector based on a level 3 decomposition using the wavelets db2 and sym7 is compared to the detector based on an undecimated, time invariant decomposition using the same wavelets
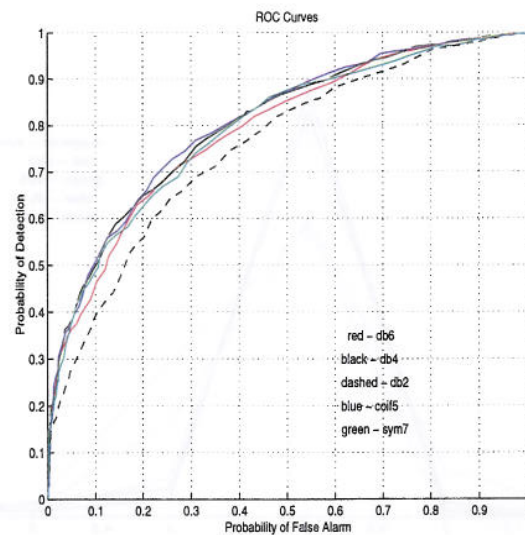
Figure 5.5: ROC curves for the maximum transform coefficient detector in the detection of the empirical transient of figure 2.4. A level 4 decomposition was performed using the wavelets db2, db4, db6, coif5 and sym7

5.2 ROC curves for the matched filters

In this section, we compare the performance of the matched filter with the performance of the matched filter using only some of the subbands. This is done by plotting the detectors in the same ROC plot. For comparison, the energy detector, the time dependent maximum energy wavelet coefficient detector, and the windowed Fourier coefficient detector are also included.
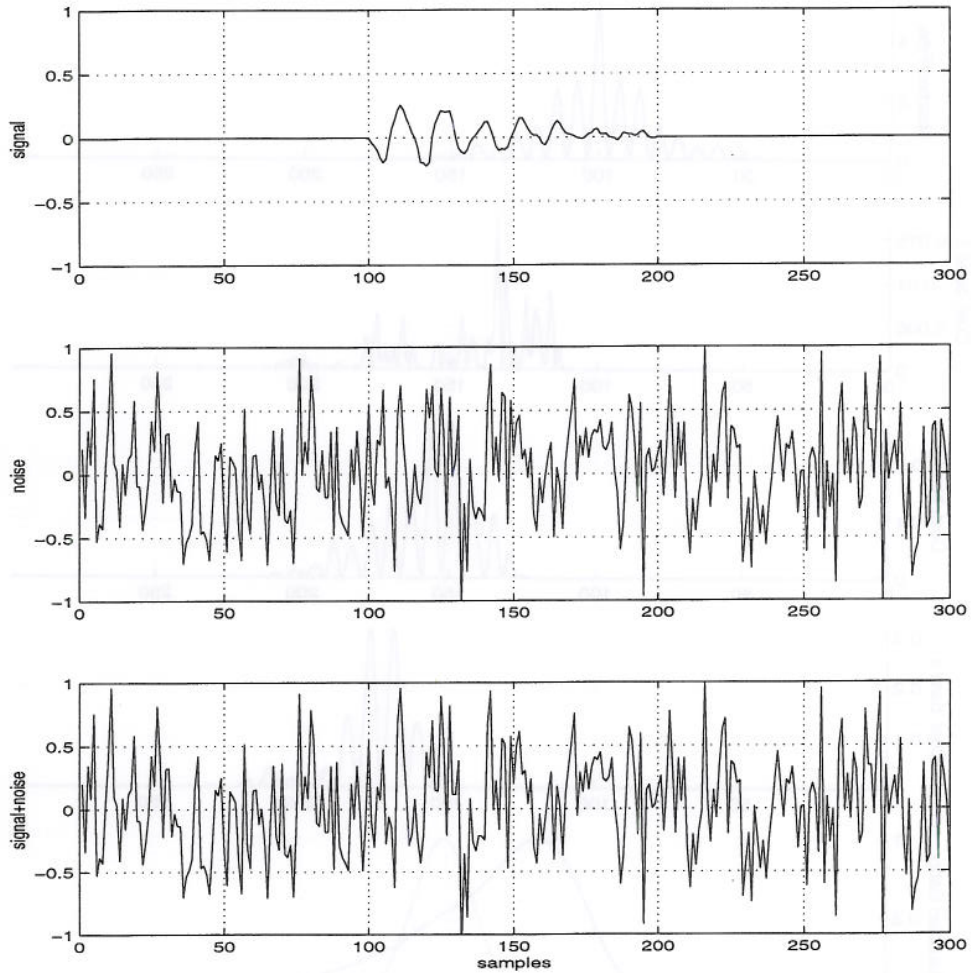
Figure 5.6: The signal and noise used in the generation of the ROC curves of figure 5.8

The length of the observation interval was 300 samples. The transient was the empirical transient of figure 2.4 using 102 samples, and its arrival time was set to sample number 100. All of the above detectors are of type 2. The detector output is compared at one time instant, namely the time of arrival of the transient (i.e. sample number 100). The maximum energy wavelet coefficient detector used was the detector of equation (4.21) in section 4.5. As mentioned in that section, the maximal output of this detector will not come at the true time of arrival. In the
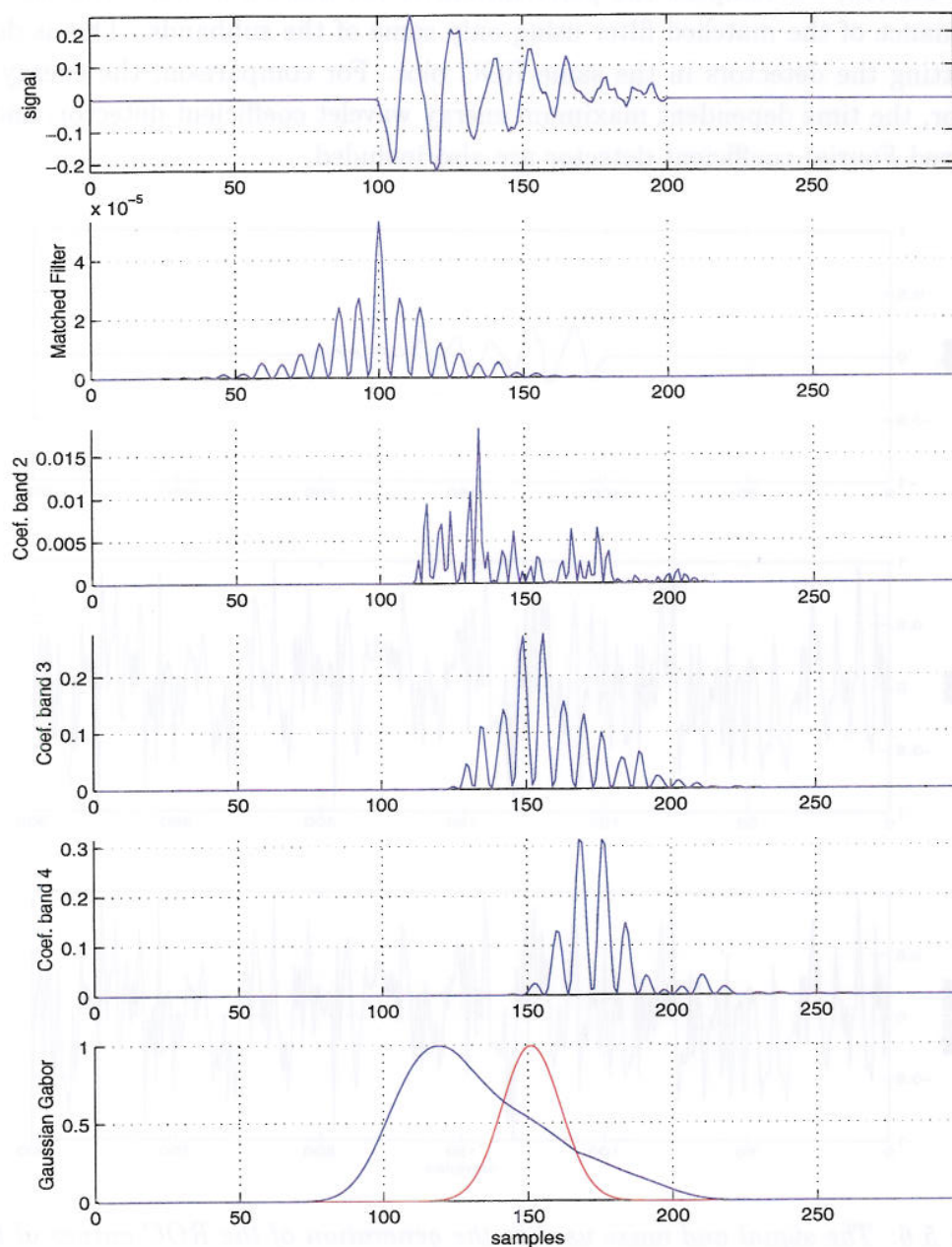
Figure 5.7: Matched filter output (second plot), maximum energy wavelet coefficient detectors (plot 3, 4 and 5) and maximum Gabor coefficient detector (plot 6) applied to the signal used in the ROC simulation of figure 5.8. The window in the short time Fourier transform is indicated in red. The reference signal in the matched filter was the shown transient itself. The figure shows that the estimate for the time of arrival differs from the estimate obtained by the matched filter

case of detecting the present signal, the situation is shown in figure 5.7. The figure shows that the estimate for the time of arrival differs from that of the matched filter. This time offset was 134 for the detector using band 2, 156 for band 3 and 168 for band 4. The same reasoning applies for the Gabor detector, which has an offset of 119. Since the offset is deterministic, we compensated for this offset in the generation of the ROC curve by using the detector value at the time where that detector output was maximum. Note that these time offsets doesn't deteriorate the ability of the detector to detect a transient, but it makes it very hard to estimate the time of arrival. None of the matched filtering techniques have this problem.

1000 simulations were performed with noise only, and 1000 simulations were performed using signal plus noise. The added noise was uncorrelated Gaussian, and each noise sample had mean zero and variance 0.4^2. The transient was normalized to have l^2-energy 1. Observe that the input signal-to-noise ratio is lower in this simulation than in the simulations in section 5.1. This is because the detectors are better, and because they use the prior information that the transient will arrive at time 100. The signal used in the simulation is plotted in figure 5.6, along with a typical noise realization and their sum.

The ROC curves are plotted in figure 5.8. We see that the performance of the matched filter using only bands 2, 3 and 4, is nearly as good as the full matched filter. The performance of the matched filter using only band 3 is also very good. Both the maximum wavelet coefficient detector and the maximum Gabor detector outperform the energy detector, but perform much worse than the matched filters. However, these detectors have the problem with estimation of arrival time. This problem need to be solved in order to use these detectors properly.

ROC Curves

red – MF
blue – MF using subbands 2, 3, 4
dashed – MF using subband 3
magenta – wavelet coef. subband 3
green – Gabor detector
black – energy detector

Probability of Detection

Probability of False Alarm
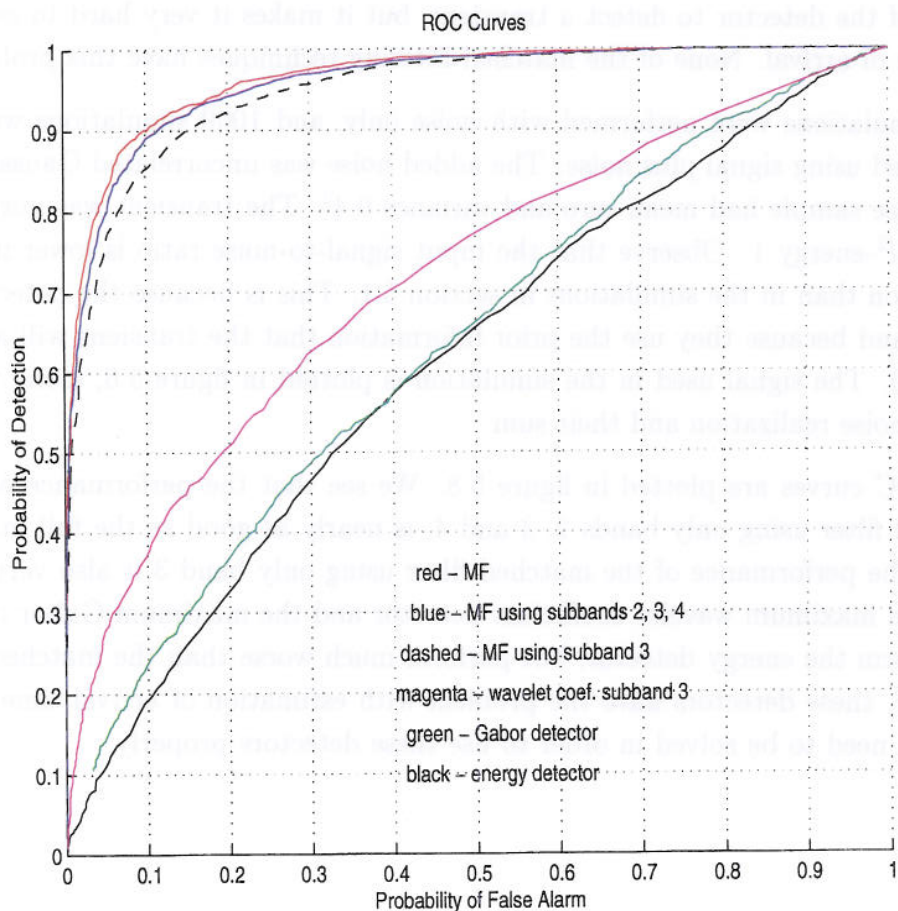
Figure 5.8: ROC curves for the detection of the empirical transient of figure 2.4, where the detector outputs are compared at the time of arrival of the transient. The matched filter is compared to matched filters using only some subbands, a maximum energy wavelet coefficient detector using only subband 3, a Gabor detector and the energy detector

5.3 The energy distribution of the wavelet coefficients
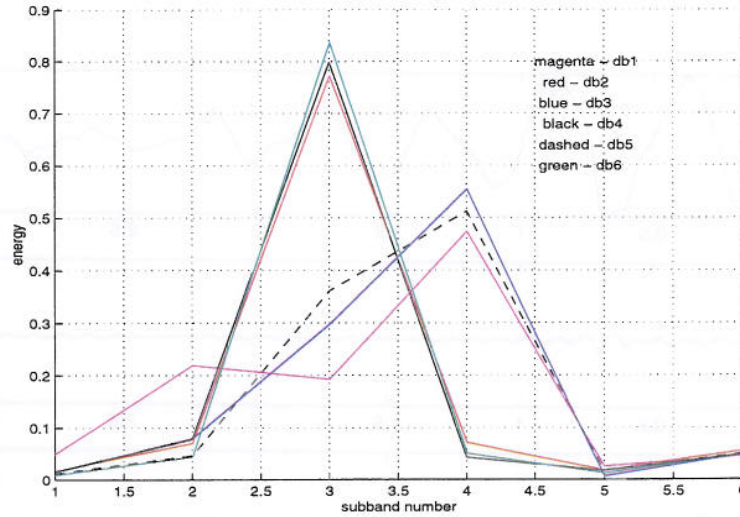
Figure 5.9: The subband energy distribution for the discrete wavelet decomposition of the empirical transient of figure 2.4, using the db1-6 wavelets

An important issue is the choice of wavelet and relevant subbands in our wavelet matched filter or our maximum transform coefficient detector. We have seen that the wavelet transform should concentrate the signal energy in as few transform coefficients as possible. This will often give a higher maximum correlation between wavelet and signal. In this chapter, we measure the energy distribution of the transform coefficients of the transient we are searching for. The energy of the coefficients in the different wavelet subbands are plotted. Discrete wavelet transforms of the empirical transient of figure 2.4 were performed using several different wavelets. The energy distributions over the different subbands using the first members of the Daubechies wavelet family are plotted in figure 5.9. The Daubechies 6 wavelet has the sharpest energy distribution. Most of the energy was in subband 3, with some energy in bands 2 and 4. Some of the wavelet decompositions has most energy in band 4 instead of band 3. The Daubechies 6 also outperformed other members of the Daubechies, Symlet and Coiflet families (plots C.4, C.5 and C.6 in appendix C). In addition, as mentioned in section 5.1, this wavelet performed best in the maximum transform coefficient detector using the empirical transient. This is the motivation behind my frequent use of the Daubechies 6 wavelet in the transform domain matched filtering experiments. The discrete wavelet transform of the empirical transient of figure 2.4 using the Daubechies 6 wavelet is plotted in figure 5.10. We see how scale 3 is dominant.
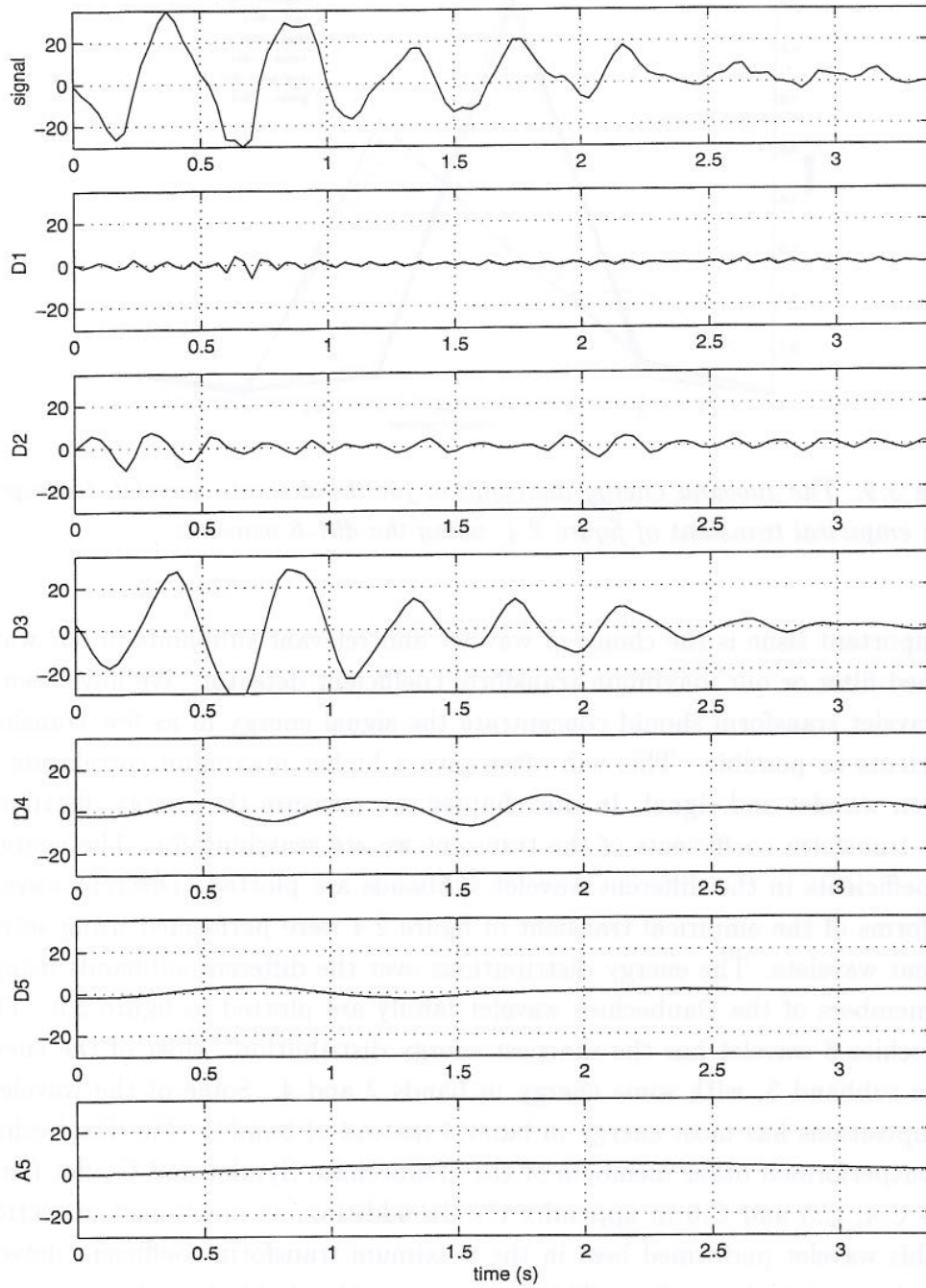
Figure 5.10: *The discrete wavelet transform projections of the empirical transient of figure 2.4 using the Daubechies 6 wavelet*

5.4 Estimation of amplitude

When using only some subbands in the matched filtering, some of the output signal energy is lost. We saw in figure 4.6 how the detector peaks decrease in amplitude when more coefficients are set to zero. In order to get an appropriate estimate for the signal amplitude when using only some of the transform coefficients, formula (4.65) of section 4.11 must be used. An example of estimating the amplitude according to this formula, is shown in figure 5.11. The figure shows matched filter output on the signal of figure 2.3 using the empirical transient of figure 2.4. The type of transform was a level 5 Daubechies 6 decomposition. The effect of losing energy by setting coefficients to zero is compensated by dividing by the reduced transform domain transient energy. The resulting amplitudes are about the same size as the amplitude estimates from the full matched filter.

In order to demonstrate our techniques in practice, matched filter algorithms were performed on a longer portion of the data than shown in figure 2.3. As before, the empirical 2 Hz transient of figure 2.4 was the reference signal, and the wavelet was a Daubechies 6. The threshold level was set to 0.1, and the distribution of the amplitudes was recorded. The result of both a full matched filter and an amplitude normalized matched filter using only subband 3 is shown in figure 5.12. We see that on average, the subband 3 matched filter gives a higher approximation for the amplitudes.

5.4 Estimation of amplitude

When using only some subbands in the matched filtering, some of the output signal energy is lost. We saw in figure 4.6 how the detector peaks decrease in amplitude when more coefficients are set to zero. In order to get an appropriate estimate for the signal amplitude, the ... formula (4.65) of section 4.11 must be used. An example of estimating the amplitude according to ... the matched filter output ... type ... energy has been set to zero is compensated by dividing ... the estimated transform domain transient energy. The resulting amplitudes are about the same size as ... the full matched filter.

In order to demonstrate that estimate in practice, a matched filtering experiment was performed. A larger portion of the data than shown in figure 5.11 ... the amplitude. The transient of figure 2.4 was the reference signal, and the signal was a Daubechies ... wavelet ... had amplitudes from ... to 0.1 ... and the distribution of the amplitude was computed. ... result ... with the amplitude normalized matched filter using only subband 3 is shown in figure 5.12. We see that ... the amplitudes.

Figure 5.11: Matched filtering using only some of the subbands in the Daubechies 6 wavelet decomposition, with the output normalized to give the amplitudes according to formula (4.65). The reference signal is the empirical 2 Hz transient of figure 2.4
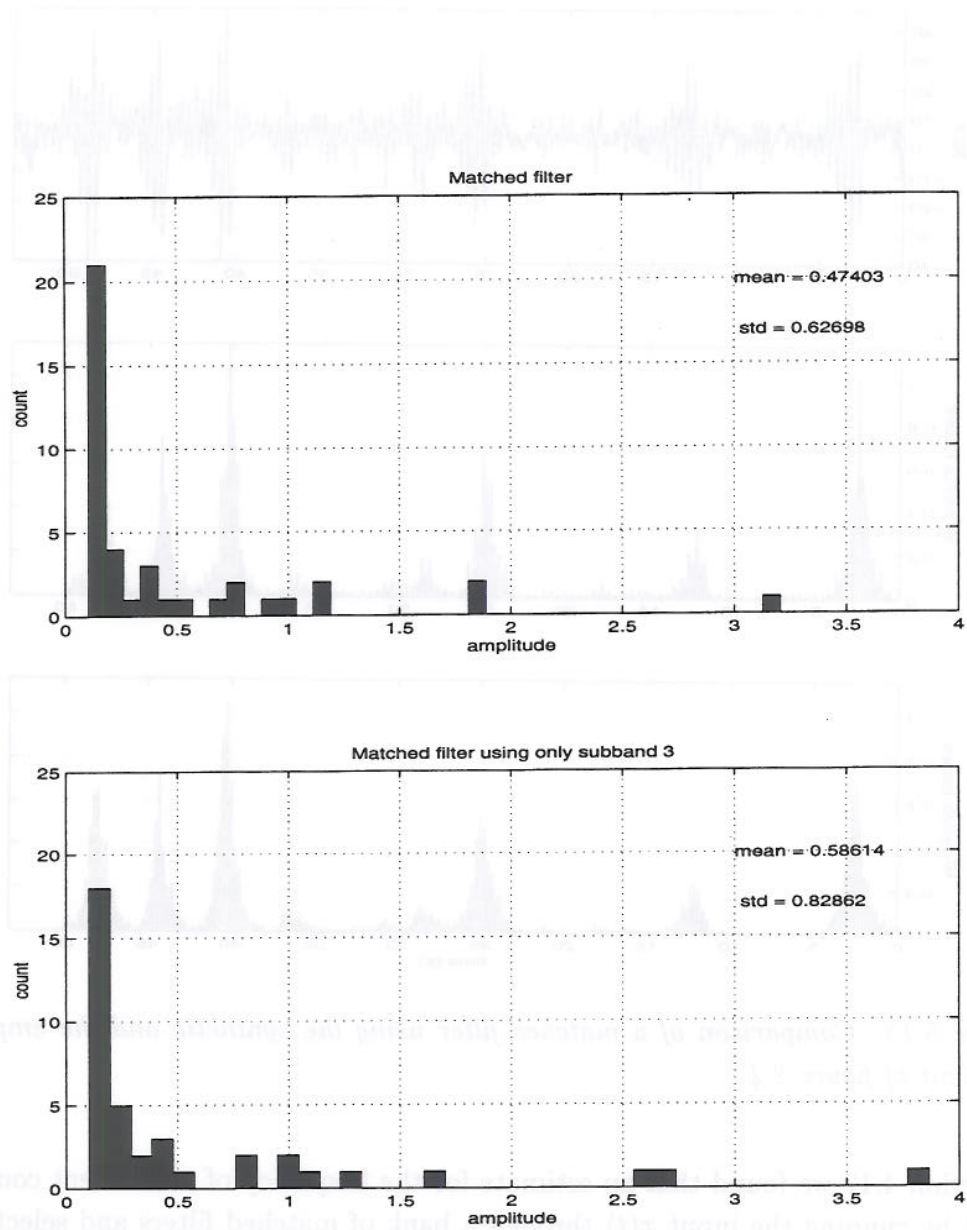
Figure 5.12: *The amplitude distribution of 2 Hz transients in a 5 minutes and 30 seconds long portion of the data, obtained with a matched filter (top) and an amplitude normalized subband 3 matched filter (bottom)*

5.5 Estimation of frequency
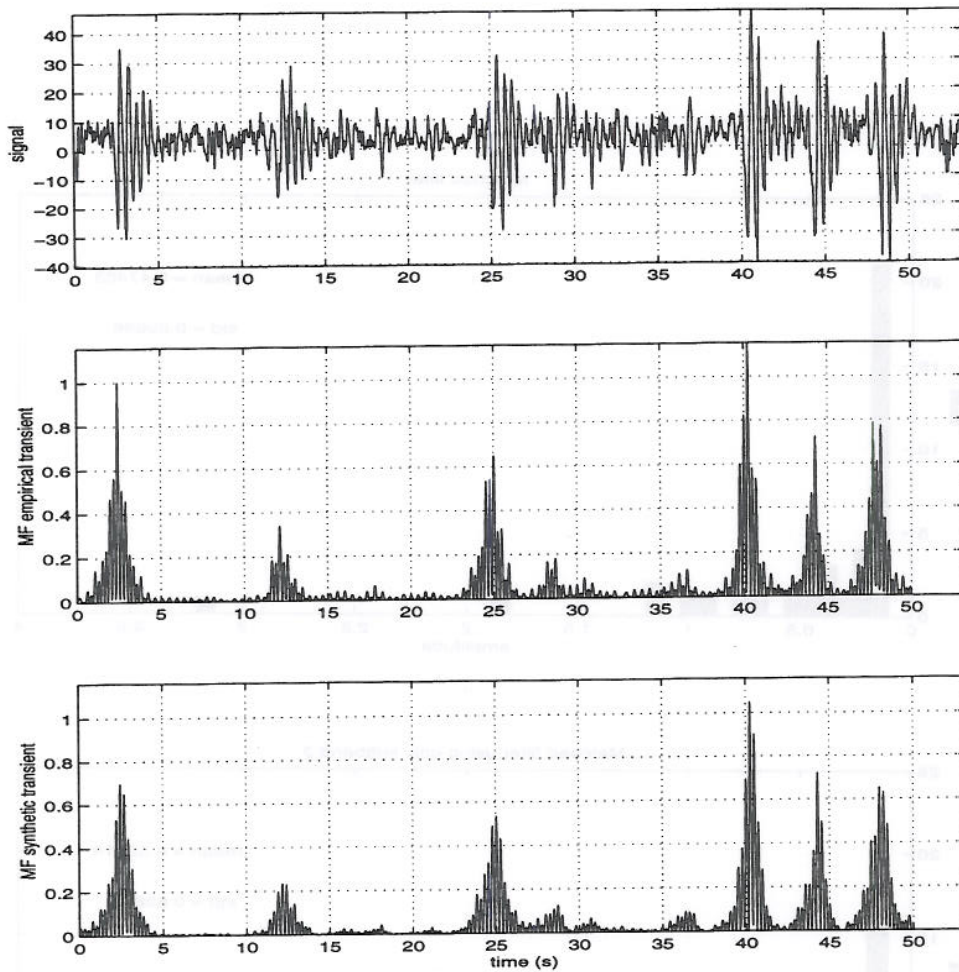
Figure 5.13: Comparison of a matched filter using the synthetic and the empirical transient of figure 2.4

In section 4.12 we found that an estimate for the frequency of a transient could be found by running the input $x(t)$ through a bank of matched filters and selecting the filter with the maximum output.

First, we check how the performance of the matched filter is altered when we use the synthetic transient of figure 2.4 instead of the empirical transient. Figure 5.13 shows this for the matched filter case, and figure 5.14 shows it for the case of a subband 3 matched filter using the Daubechies 6 wavelet. In both cases, the similarity of the filter outputs is reassuring.

The input $x(t)$ was run through a bank of filters, both in the case of the matched filter, and the subband 3 matched filter. The results are in figures 5.15 and 5.16 respectively. In both cases, the highest output occurs when the frequency of the reference transient is chosen to be around 2.1 Hz. In order to establish the
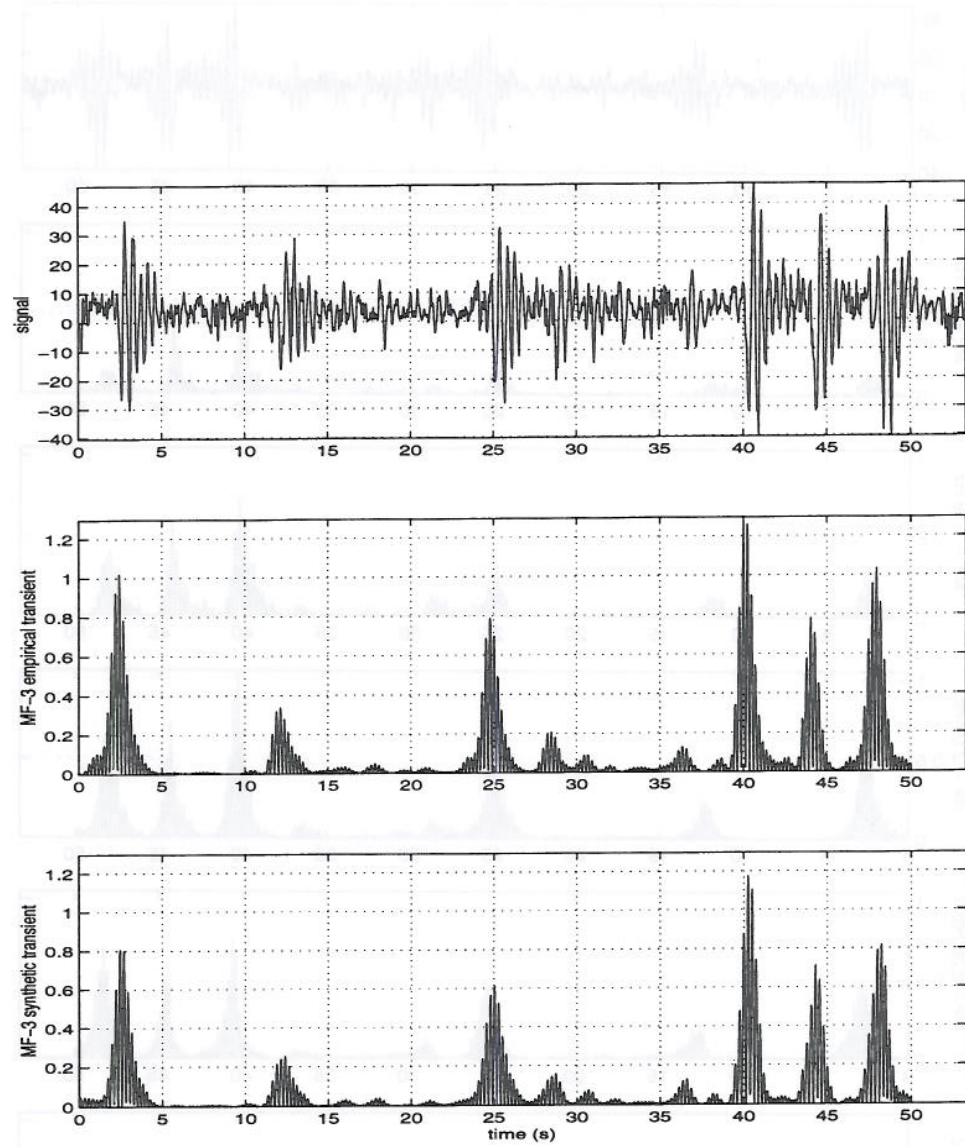
Figure 5.14: Comparison of a subband 3 matched filter using the synthetic and the empirical transient of figure 2.4
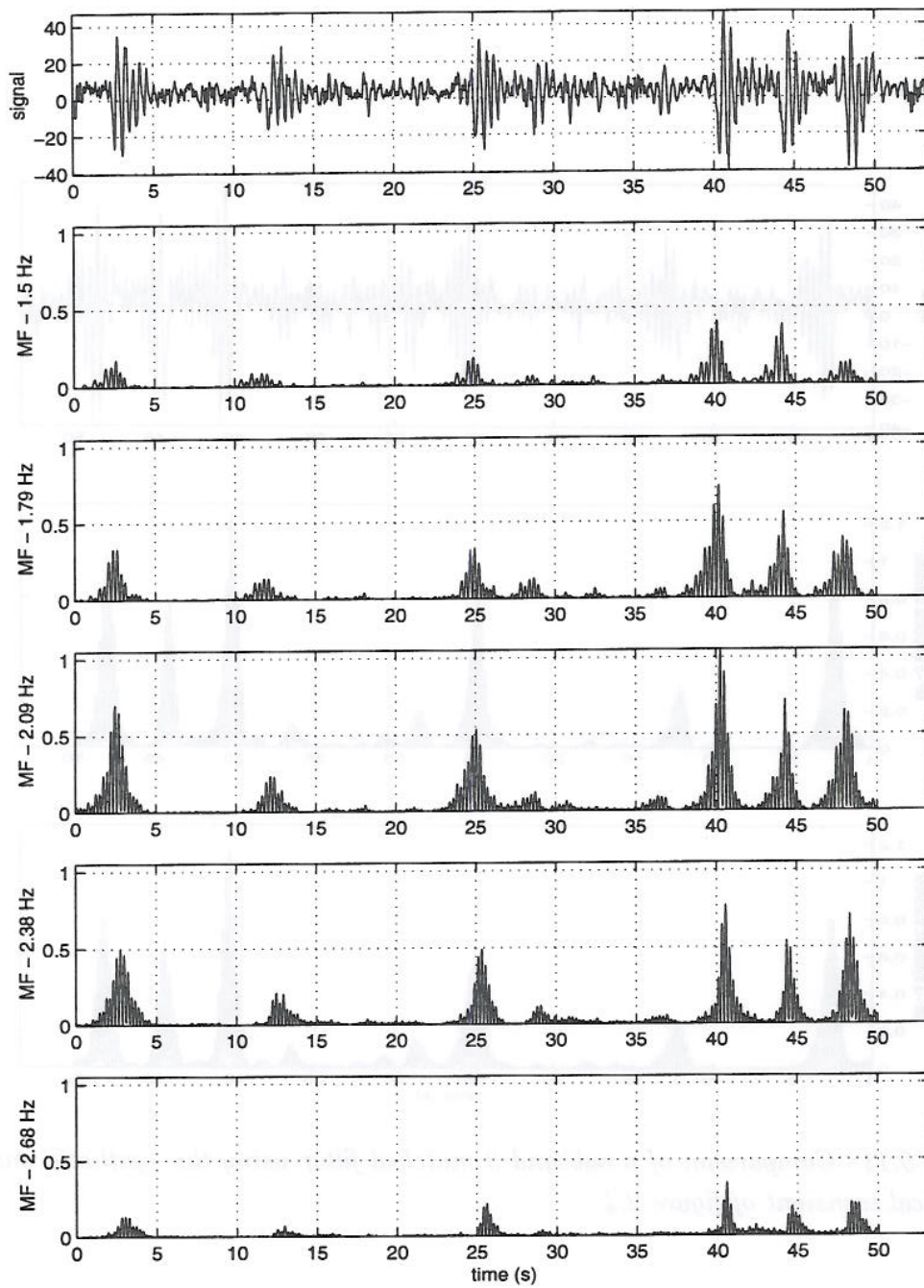
Figure 5.15: Matched filtering using a bank of synthetic filters generated by formula (2.2) with varying frequency and constant damping
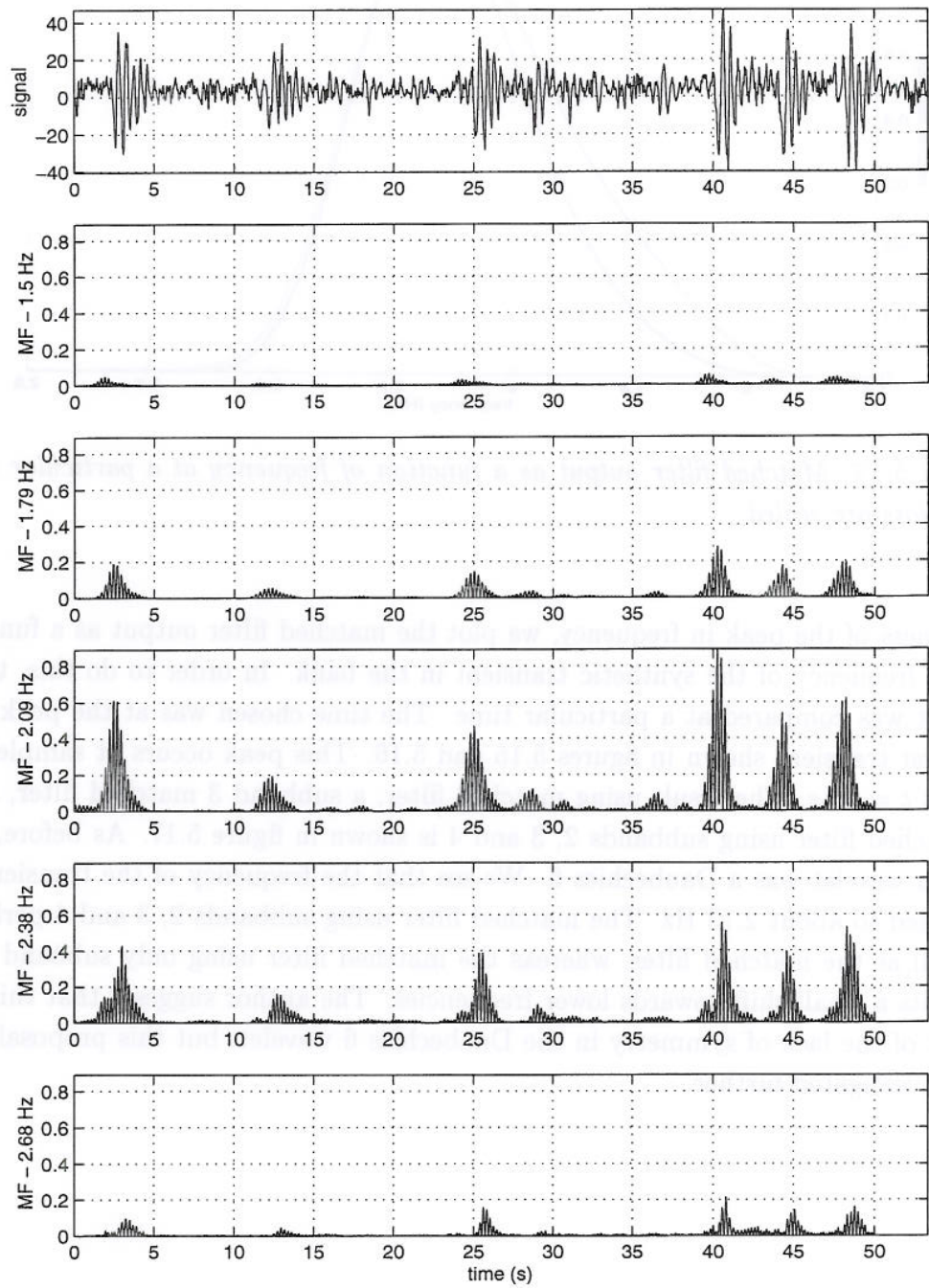
Figure 5.16: Subband 3 matched filtering using a bank of synthetic filters generated by formula (2.2) with varying frequency and constant damping
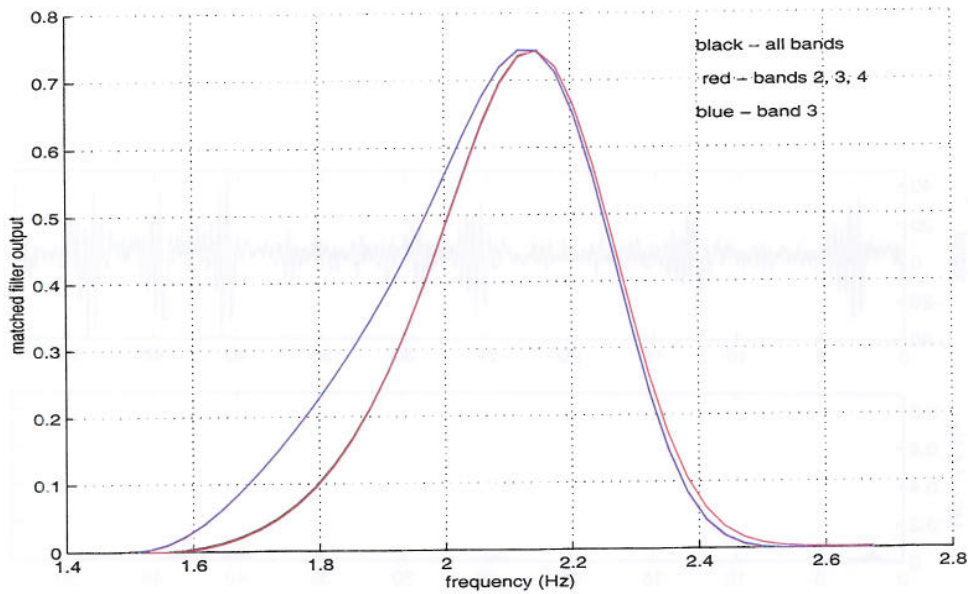
Figure 5.17: Matched filter output as a function of frequency at a particular time. The plots are scaled

sharpness of the peak in frequency, we plot the matched filter output as a function of the frequency of the synthetic transient in the bank. In order to do this, the output was compared at a particular time. The time chosen was at the peak of the first transient shown in figures 5.15 and 5.16. This peak occurs at sample 75, i.e. at $t = 2.5s$. The result using matched filter, a subband 3 matched filter, and a matched filter using subbands 2, 3 and 4 is shown in figure 5.17. As before, the chosen wavelet was a Daubechies 6. We see that the frequency of the transient is localized to about 2.13 Hz. The matched filter using subbands 2, 3 and 4 performs as well as the matched filter, whereas the matched filter using only subband 3 exhibits a small shift towards lower frequencies. The author suggests that this is a result of the lack of symmetry in the Daubechies 6 wavelet, but this proposal was not investigated further.

6 DISCUSSION AND CONCLUSION

This work represents a self contained mathematical foundation for the detection and characterization of transients using wavelet transforms. Of course, this work is by no means complete. Literature on wavelets is huge, and this work is a summary of the parts that are most essential to the problem at hand.

The theory of frames has been reviewed, and it has been shown how a correlator equivalently can be implemented in the transformed domain instead of in the time domain. Wavelet transforms were introduced as special cases of frames. The concept of multiresolution has been emphasized, because it gives a link between wavelet analysis and filter bank theory. Implementation of wavelet transforms through filter banks is very important for a practical signal processing application such as ours, because it offers a fast algorithm which does not increase the storage need. For orthonormal wavelet transforms, the number of wavelet coefficients is approximately the same as the number of input samples. Methods for improving the standard discrete wavelet transform has been discussed. In the problem of detection, time invariant transforms is especially important. Other improvements that might be valuable are the wavelet packet transform, biorthogonal transforms and M-band wavelet transforms.

Our application is the detection and characterization of loads in the hull of high-speed catamaran boats built from glass fibre reinforced composite materials. Three of the most typical phenomena are 2-2.5 Hz transients resulting from long-ship vibrations, 20 Hz transients resulting from local panel vibrations in the hull, and a low frequency signal resulting from swells. In this work, we have mostly made use of the 2 Hz transient to illustrate the detection and characterization algorithms, but some analysis of the 20 Hz transient was also performed. In experimental data from tests at sea, both these transients seems to have quite large signal-to-noise ratios, making them easy to detect.

In white noise, a matched filter is the same as a correlator. A correlator based on orthonormal wavelet transforms has been implemented, and has been shown to be equivalent to the usual time domain matched filter when all the coefficients in the decomposition are used. By basing the correlator only on a few relevant coefficient subbands, several advantages arise. The speed of the algorithm increases, because the computation is performed using a fraction of the wavelet coefficients. In addition, the wavelet approach might decrease the storage needs of our system. In a logging system, input data must be stored. By taking the wavelet transform and storing only the subbands that are essential to our problem, space is saved. Finally, a correlator based on a selected set of subbands has been shown to outperform the full matched filter in some cases. In figure 4.9, 20 Hz transients were detected by a subband matched filter matched to such transients, while the full matched filter

had a more noisy output. The 2 Hz transient was present and was corrupting the output from the time domain matched filter. By using only the high-frequency (small scale) subbands, we were able to focus on the 20 Hz transient, with little corruption from the 2 Hz transient.

In the ideal case of only one transient present in white noise, the performance of the subband matched filter has been shown to be comparable to that of the full matched filter, at least when the signal to be detected was the empirical 2 Hz transient of figure 2.4. This is shown in the ROC curves of figure 5.8. We have demonstrated how ROC curves can be used to compare the performance of different detectors in white noise.

The matched filter based techniques make use of the prior knowledge of the wave-shape of the transient. A detector based on the maximum wavelet coefficient does not need this piece of knowledge. One potential for such a detector, is an initial 'screening' of the data in order to find the most prevailing and dominant phenomena. After these phenomena has been detected and identified, more sophisticated techniques (for example correlators) can be designed. The maximum wavelet coefficient detector has been shown to perform well, at least in the case of a large signal-to-noise ratio. When using this type of detector, using a time invariant transform gives considerably better performance. However, it is hard to find good estimates for the time of arrival using the maximum wavelet coefficient detector.

The choice of wavelet for the decomposition, both for the wavelet based matched filter and for the maximum wavelet coefficient detector, is very dependent on the wave-shape of the sought signal. Even though the empirical 2 Hz transient of figure 2.4 is visually quite similar to the synthetic transient of the same figure, the performance of the maximum wavelet coefficient detector in the detection of the signal was quite different when using the empirical transient instead of the synthetic one, even when using the same wavelet. On the empirical 2 Hz transient of figure 2.4, the Daubechies 6 wavelet performed best, whereas the Symlet 7 wavelet performed best on the synthetic 2 Hz transient. One indication on how well a wavelet performs on detecting a particular transient, is how the energy of the coefficients is distributed over the decomposition subbands. The energy should be concentrated in as few subbands and coefficients as possible in order to separate the signal from noise.

When it comes to characterizing a transient, an estimate for the signal amplitude based on only some of the wavelet coefficients of the signal was suggested in section 4.11. The estimator was an analogy to the maximum likelihood estimate for the amplitude in the case of a full matched filter. The performance of this estimator was briefly tested in section 5.4. The amplitude estimates were generally higher when using fewer coefficients, but to arrive at more specific conclusions, a further study is required. In section 5.5, it was demonstrated how a bank of matched

filters can be used to estimate the frequency of an oscillatory transient. When using only a few subbands in the matched filtering, a small frequency shift was observed (figure 5.17). This is probably due to the fact that the wavelet we used was non-symmetric.

6.1 Further work

The CHESS project is by no means finished, neither is the development of the signal processing system needed in order to detect and characterize the loads in the hull. In this section, we suggest some of the possibilities for further investigation.

A practical problem is deciding on the right threshold level for the detector outputs. Something that could make this choice easier, is a study of the false alarm rate as a function of the threshold level. A more detailed study of the estimators for the amplitude using all bands or only a few subbands could also be interesting. One idea might be an estimation of the estimator variance by a Monte Carlo simulation, where a signal containing white noise and several transients of known amplitudes is analyzed, and the deviations between the real amplitudes and the estimator outputs using different wavelet coefficients are compared at similar signal-to-noise ratios.

In the estimation of frequency, a bank of matched filters where each filter is matched to a given frequency, has been used. A topic to be studied in more detail, is how densely we must specify the filters in the bank. A bank that is too dense wastes cpu time, while too sparse a bank might miss some transients. One idea is to specify the bank in the wavelet domain, using only the relevant subbands. Maybe a Gram-Schmidt procedure can be used in order to extract a basis of matched filters that covers the possible wave-shapes of the sought transients?

It is hard to make the maximum wavelet coefficient detector output a function of time. Making this detector give a good estimate of the time of arrival of transients is a topic for further study. Maybe symmetric wavelets is a solution? In addition, the fact that oscillatory transients yield oscillatory matched filter output, is a problem that needs further investigation. Using biorthogonal wavelets instead of orthogonal is an interesting option. One of the great advantages of these wavelets for our application, is that symmetric wavelets with FIR filter bank implementation are available.

The use of wavelet packets, both in the wavelet based matched filter algorithm and in the maximum wavelet coefficient detector, is a very interesting extension. Del Marco et al. (199) has shown that the wavelet packet decomposition has a performance superior to the usual wavelet transform when used in the maximum energy wavelet coefficient detector. The best basis wavelet packet decomposition allows for the concentration of signal energy in fewer coefficients than possible with

a usual wavelet decomposition. In addition, the wavelet packet decomposition allows for a more flexible tiling of the time-frequency plane. The use of M-band transforms might be combined with the wavelet packet transform for an even more flexible decomposition.

We have used only one type of synthetic transient, namely the transient of equation 2.2. Trying other, more sophisticated models as reference signals in the matched filters might give improved performance. In addition, the 20 Hz transient needs to be studied in more detail. We have focused on detecting the 2 Hz transient. The detection of the higher harmonics which are thought to be present in the 20 Hz transient, is a very interesting field of further development. As mentioned in chapter 2, changes in these frequencies are particularly important in the early detection of delamination. Since these higher harmonics have much lower signal-to-noise ratios than the 20 Hz oscillation itself, we have to be very careful in the design of a detector. A good idea is to first look for the 20 Hz transient, which has a high signal-to-noise ratio, and then look for the higher harmonics in a small neighbourhood around the estimated time of arrival. Since the wavelet packet decomposition allows for a more flexible tiling of the time-frequency plane, it might be useful in the detection of the higher harmonics.

A wavelet analysis of the signal of figure 2.7 could also be interesting. An idea is to use wavelets to extract these phenomena and remove them from the signal, in order to get a clearer picture of other signal components in the time domain. Wavelets can also be used to extract the time interval between each top in the signal of figure 2.7.

We have considered the analysis of the output of only one of the fibre optic sensors. In a final system, many sensors will give simultaneous output from different positions in the hull. A combined analysis of these outputs might give a stronger tool both for the detection of signals, and for the detection of a possible fault or delamination. Another important fact to keep in mind is that the sensor output might be dependent on the location of the sensor. Especially the shape of the 20 Hz transient, which is a result of local panel vibrations, might vary from sensor to sensor depending on the geometry of the panel on which the sensor is placed.

We have made the assumption of white, Gaussian noise. Most likely, our noise is neither white nor Gaussian. Characterizing the noise in our system might reveal more information about the present signals, and in addition, be a basis for the design of detectors tailor-made for that kind of noise. One way of compensating for coloured noise, is to apply a noise whitening filter before the application of a matched filter. However, one have to bear in mind that since sensors have different locations, they might have different noise characteristics. Maybe the wavelet decomposition can be used as a tool in the noise analysis?

More research has to be put into the investigation of the relationship between

delamination and deviation in transient characteristics. This applies both to the signal processing system and to experimental tests of how glass fibre reinforced composite materials behave when they are subject to wave slammings, with and without a present delamination. For instance, we have to know in what direction the ground and higher harmonic frequencies shift, and by how much. In addition, the wave-shape of the transients might change by a delamination. In that case, a modification of the matched filter techniques might be necessary.

The investigation of techniques not based on wavelets for the detection and characterization of transients, and the comparison of these techniques with wavelet based techniques, might also be of interest. Although we have indications of wavelets being a good tool for our problem, we do not know if better techniques exist.

We have investigated various detector techniques using Matlab and the Wavelet Toolbox in Matlab. This makes the implementation easy, and allows for fast prototyping. However, in a final implemented system, Matlab is both much too slow and gives too little flexibility. An implementation in a low level language such as C or C++ is recommended. Maybe it is possible to make use of some of the commercial wavelet software packages available on the market? An additional fact to take into consideration, is that the final implemented system will perform real time, whereas our Matlab routines do the processing a posteriori.

List of Figures

References

Anant, K. S & Dowla, F. U. & Rodrigue, G. H., Detection of the electrocardiogram P-wave using wavelet analysis. *Proceedings of SPIE - The International Society for Optical Engineering 2242* (Apr. 1994), pp. 744-749.

Beylkin, G., On the representation of operators in bases of compactly supported wavelets. *SIAM J. Numer. Anal. 6* (Dec. 1992), pp. 1716-1740.

Burrus, C. S. & Gophinath, R. A & Guo, H, *Introduction to Wavelets and Wavelet Transforms.* Prentice Hall, to be published 1997.

Carter, P. H., Unknown transient detection using wavelets. *Proceedings of SPIE - The International Society for Optical Engineering 2242* (Apr. 1994), pp. 803-814.

Chen, V. C, Radar ambiguity function, time-varying matched filter, and optimum wavelet correlator. *Proceedings of SPIE - The International Society for Optical Engineering 2242* (Apr. 1994), pp. 337-343.

Cohen, A., *Wavelets and Multiscale Signal Processing.* Translated to English by R. D. Ryan, Chapman & Hall, 1995.

Cohen, L., *Time-Frequency Analysis.* Prentice hall P T R, 1995.

Coifman, R. & Wickerhauser, M. V., Entropy-based algorithms for best basis selection. *IEEE Trans. Inf. Th. 38* (Mar. 1992), pp. 713-718.

Cormen, T. H. & Leiserson, C. E. & Rivest, R. L., *An Introduction to Algorithms.* The MIT Press, 1990.

Daubechies, I., *Ten Lectures on Wavelets.* Philadephia: SIAM, 1992.

Dodson, M., Shannon's Sampling Theorem. *Current Science 5* (1992), pp. 253-260.

Donoho, D. L, De-Noising by Soft-Thresholding. *IEEE Trans. Inf. Th. 41* (May 1995), pp. 613-627.

Duffin, R. J. & Schaeffer, A. C., A class of nonharmonic Fourier series. *Trans. Amer. Math. Soc. 72* (1952), pp. 341-366.

Dym, H. & KcKean, H. P., *Fourier series and integrals.* Academic Press, Inc., 1972.

Eriksen, R., Realtime System for Structure Monitoring of Ships Using Wavelets for Transient Characterization and Detection. (FFI/Rapport-96/03149).

Friedlander, B. & Porat, B., Detection of transient signals by Gabor representation. *IEEE Trans. Acoust., Speech Signal Processing 37* (1989), pp. 169-180.

Frisch, M & Messer, H., Detection of a transient signal of unknown scaling and arrival time using the DWT. *IEEE International Conference on Acoustics, Speech and Signal Processing 2* (1991), pp. 1313-1316.

Frisch, M & Messer, H., Detection of a Known Transient Signal of Unknown Scaling and Arrival Time. *IEEE Trans. Signal Proc. 42* (July 1994), pp. 1859-1863.

Helstrom, C. W., *Statistical theory of signal detection, 2nd edition.* Pergamon Press, 1968.

Jensen, A. E. (the Norwegian University of Science and Technology), Personal communication, 1997.

Kaiser, G., *A Friendly Guide to Wavelets.* Birkhauser, 1994.

Kersey, A. D. & Davis, M. A. & Berkoff, T. A & Bellemore, D. G. & Koo, K. P. & Jones, R. T., Progress Towards the Development of Practical Fiber Bragg Grating Instrumentation Systems. Manuscript from the Fiber Optic Smart Structures Section, Naval Research Laboratory, Washington.

Kovacevic, J & Daubechies, I. (editors), Special Issue on Wavelets. *Proc. of the IEEE 84* (April 1996), pp. 505-688.

Mallat, S., Multiresolution approximation and wavelets. *Trans. Am. Math. Soc. 135* (1989), pp. 69-88.

Mallat, S., Zero-Crossings of a Wavelet Transform. *IEEE Trans. Inf. Th. 37* (July 1991), pp. 1019-1033.

Marco, S. Del & Weiss, J. & Jagler, K. B., Wavepacket-based transient signal detector using a translation-invariant wavelet transform. *Proceedings of SPIE Wavelet Applications Conference 2242* (1994a), pp 792-802.

Marco, S. Del & Weiss, J., M-band wavepacket based Transient Signal Detector using a Translation-Invariant Wavelet Transform. *Optical Engineering 33* (1994b), pp. 2175-2182.

Marco, S. Del & Weiss, J., Signal Detection Using a Wavepacket-Based Detector with an Extended Translation-Invariant Wavelet Transform. *IEEE Trans. Sign. Proc. 45* (Apr. 1997), pp. 841-850.

Meyer, Y., Ondelettes, function splines, et analyses graduees. Univ. of Torino, 1986.

Misiti, M. & Misiti, Y. & Oppenheim, G. & Poggi, J.-M, *Wavelet Toolbox for Use with Matlab*. The MathWorks, 1996.

Pran, K., Design of Optical Fibre Bragg Gratings. (Diploma Thesis, the Norwegian University of Science and Technology, Jan. 1995).

Turin, G. L., An Introduction to Matched Filters. *IRE Trans. Inf. Th. IT-6* (June 1960), pp. 311-329.

Vetterli, M. & Kovacevic, J., *Wavelets and Subband Coding*. Prentice Hall P T R, 1995.

Young, N., *An introduction to Hilbert space*. Cambridge University Press, 1988.

A PRELIMINARIES

A.1 Notation

Kronecker's symbol δ is defined as $\delta_{i,j} = 1$ if $i = j$, 0 if $i \neq j$. The Dirac delta function will be denoted as $\delta(t)$. We denote the unit step function as $H(t)$, i.e.

$$H(t) = \begin{cases} 1 & t > 0 \\ 0 & t \leq 0. \end{cases} \tag{A.1}$$

A function $f : \mathbb{R}^n \to \mathbb{R}$ has *compact support* if there exists $A \in \mathbb{R}$ such that $f(x) = 0$ for $|x| > A$. A function f is *convex* if

$$f(\lambda x_1 + (1 - \lambda)x_2) \leq \lambda f(x_1) + (1 - \lambda)f(x_2) \tag{A.2}$$

for any $\lambda \in [0, 1]$ and any $x_1, x_2 \in \mathbb{R}^n$.

For $z \in \mathbb{C}$, we denote complex conjugation by $\overline{z}$. For a set U, we denote the closure by $\overline{U}$.

We will measure the speed of an algorithm with an asymptotic upper bound (Cormen et al, 1990). Let N denote the size of the data to be processed, and let $f(N)$ denote the algorithm speed. We write $f(N) = O(g(N))$ as $N \to \infty$, provided there exists a constant $C > 0$ such that $|f(N)| \leq C|g(N)|$ for all N sufficiently large.

List of symbols and abbreviations.

db	Daubechies wavelet family.
sym	Symlet wavelet family.
$coif$	Coiflet wavelet family.
ϕ	The scaling function.
ψ	The wavelet mother function.
h_0	Digital lowpass filter corresponding to ϕ.
h_1	Digital highpass filter corresponding to ψ.
H_0	Discrete-time Fourier transform of h_0.
H_1	Discrete-time Fourier transform of h_1.
FIR	Finite Impulse Response digital filter.
IIR	Infinite Impulse Response digital filter.
K	Length of FIR digital filter.
N	Number of samples in a discrete input signal.
J	Highest decomposition level, when decomposition starts at scale $j = 0$.
T	Length of transient.
$O(g(N))$	Upper bound for speed of algorithm measured as a function of N.
a	Continuous scale parameter.
b	Continuous translation parameter.
j	Discrete scale parameter.
k	Discrete translation parameter.
V_j	Scale function space of scale j.
W_j	Wavelet space of scale j.
$V_{n,j}$	Wavelet packet space of scale j and parameter n.
$c_{j,k}$	Scaling function decomposition coefficients, $c_{j,k} = <f, \phi_{j,k}>$.
$d_{j,k}$	Wavelet decomposition coefficients, $d_{j,k} = <f, \psi_{j,k}>$.
T_ϕ	Discrete scaling function operator, $(T_\phi f)(j, k) = <f, \phi_{j,k}>$.
T_ψ	Discrete wavelet operator, $(T_\psi f)(j, k) = <f, \psi_{j,k}>$.
S	Shift operator, $(Sf)(t) = f(t + 1)$.
$n(t)$	Noise.
$s(t)$	Known signal or transient.
$x(t)$	Input signal to be analyzed.
$E[\cdot]$	Expectation value.

A.2 Hilbert spaces

This section contains a very brief summary on the theory of Hilbert spaces. More theory on Hilbert spaces is given in Young (1988).

Definition 6 *A Hilbert space H is an inner product space with inner product $<u, v>$. H is a complete metric space with respect to the metric induced by its inner product.*

The Hilbert space norm of u is given by $\|u\| = < u, u >^{\frac{1}{2}}$. We will follow the mathematician's convention and use inner products linear in the *first* argument,

$$< \lambda_1 u_1 + \lambda_2 u_2, v > = \lambda_1 < u_1, v > + \lambda_2 < u_2, v > . \tag{A.3}$$

Theorem 12 *The Cauchy-Schwarz inequality. For u, v in an inner product space,*

$$| < u, v > | \leq \|u\| \|v\|, \tag{A.4}$$

with equality if and only if u and v are linearly dependent.

Theorem 13 *The polarization identity. For any u, v in an inner product space,*

$$< u, v > = \frac{1}{4} \sum_{n=0}^{3} i^n \|u + i^n v\|^2. \tag{A.5}$$

For a proof of the Cauchy-Schwarz inequality and the polarization identity, see Young (1988).

Our most frequently used Hilbert space is $L^2(\mathbb{R})$, the space of all square summable Lebesgue measurable functions, with inner product

$$< f, g > = \int_{-\infty}^{\infty} f(t) \overline{g(t)} dt. \tag{A.6}$$

Another important Hilbert space is $l^2(\mathbb{Z})$, the set of all square summable sequences of complex numbers indexed by integers, with inner product

$$< x, y > = \sum_{n=-\infty}^{\infty} x_n \overline{y_n}. \tag{A.7}$$

Yet another important Hilbert space is the Sobolev space $H^m(\mathbb{R})$ for $0 \leq m < +\infty$. It is defined as all functions f for which

$$\int_{-\infty}^{\infty} (1 + |t|)^m |\hat{f}(t)|^2 dt < \infty. \tag{A.8}$$

Hilbert spaces always have orthonormal bases. For separable Hilbert spaces[2] this is equivalent to the existence of families of vectors $\{e_n\}$ in H such that

$$< e_i, e_j > = \delta_{i,j} \tag{A.9}$$

---

[2]A separable Hilbert space is a Hilbert space in which orthonormal bases are countable.

and

$$u = \sum_n < u, e_n > e_n \tag{A.10}$$

for all $u \in H$.

A Riesz basis $\{e_n\}$ for a Hilbert space H is a basis for which there exist $\alpha > 0, \beta < \infty$ so that

$$\alpha \|u\|^2 \le \sum_n | < u, e_n > |^2 \le \beta \|u\|^2, \tag{A.11}$$

for all $u \in H$.

For a Riesz basis $\{e_n\}$, there exists a unique *dual basis* $\{\tilde{e}_n\}$ such that

$$< e_i, \tilde{e}_j >= \delta_{i,j}. \tag{A.12}$$

Each vector u can then be decomposed as

$$u = \sum_n < u, e_n > \tilde{e}_n \tag{A.13}$$

or equivalently

$$u = \sum_n < u, \tilde{e}_n > e_n. \tag{A.14}$$

One of the bases is used to analyze the element u (by taking inner products) and the other basis is used for the synthesis (by taking linear combinations). Orthogonality corresponds to the particular case where the two families $\{e_n\}$ and $\{\tilde{e}_n\}$ are identical. More information on Riesz bases is in (Daubechies, 1992, pp. xvii-xix; Cohen, 1995, pp. 97-98).

A.3 Measure theory

A measure μ on a set M is a mapping that assigns values $\mu(A)$ to subsets $A \subset M$. For example if $M = R^2$ is the Cartesian plane, a measure μ on M could be defined as the area $\mu(A) = \int \int_A dx dy$. Often the measure will be given in differential form, in this case $d\mu = dx dy$.

We denote $L^2(\mu)$ as the Hilbert space of square summable functions with respect to the measure μ. The inner product is

$$< f, g >= \int_{-\infty}^{\infty} f(t) \overline{g(t)} d\mu. \tag{A.15}$$

If μ is the discrete counting measure on $\mathbb{Z}$, we get the inner product

$$< f, g >= \sum_n f(n)\overline{g(n)}, \tag{A.16}$$

which is the usual inner product in $l^2(\mathbb{Z})$.

For more information on measure theory, the reader is referred to Kaiser (1994, section 1.5).

A.4 Fourier analysis

This section contains a short summary of central aspects of Fourier analysis. For more information, see (Vetterli & Kovacevic, 1995, sec. 2.4; Dym & McKean, 1972).

A.4.1 The continuous Fourier transform

We use the following convention for the Fourier transform $\hat{f}(\omega)$ of $f(x)$:

$$\hat{f}(\omega) = \int_{-\infty}^{\infty} f(t)e^{-i\omega t}dt =< f(t), e^{i\omega t} >_{L^2(\mathbb{R})} . \tag{A.17}$$

The inverse Fourier transform is then given by

$$f(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{f}(\omega)e^{i\omega t}d\omega. \tag{A.18}$$

With this normalization, the Plancherel formula takes the form

$$\|f\|_{L^2(\mathbb{R})}^2 = \frac{1}{2\pi}\|\hat{f}\|_{L^2(\mathbb{R})}^2 \tag{A.19}$$

and Parseval's formula is

$$< f, g >_{L^2(\mathbb{R})}= \frac{1}{2\pi} < \hat{f}, \hat{g} >_{L^2(\mathbb{R})} . \tag{A.20}$$

A.4.2 The discrete-time Fourier transform

Given a sequence $\{f(n)\}_{n\in\mathbb{Z}}$, we define its discrete time Fourier transform (DTFT) by

$$F(\omega) = \sum_{n=-\infty}^{\infty} f[n]e^{-i\omega n}, \tag{A.21}$$

which is a 2π-periodic function in $L^2([0, 2\pi])$. Its inverse is

$$f[n] = \frac{1}{2\pi} \int_0^{2\pi} F(\omega) e^{i\omega n} d\omega. \tag{A.22}$$

A.4.3 The discrete Fourier transform

The discrete Fourier transform (DFT) acts on a finite-length sequence (which can be one period of a periodic sequence). The DFT $\{F(k)\}_{k=0}^{N-1}$ of an N-length sequence $\{f(n)\}_{n=0}^{N-1}$ is

$$F[k] = \sum_{n=0}^{N-1} f[n] e^{-i2\pi nk/N}, \tag{A.23}$$

and its inverse is

$$f[n] = \frac{1}{N} \sum_{k=0}^{N-1} F[k] e^{i2\pi nk/N}. \tag{A.24}$$

The DFT is very important for computational reasons, since it can be implemented using the Fast Fourier Transform (FFT) in $O(N \log N)$ time, where N is the sequence length.

A.5 Digital sampling

In practice, our signal is not given as a continuous function of time, but is given at discrete times. The sampling rate is usually uniform in time, so that the samples are given at times $t_n = 2\pi n/\omega$, where ω is the *sampling frequency*. When sampling a continuous function, we don't want to loose information. The famous sampling theorem of Shannon states that this is possible for *bandlimited* functions.

A.5.1 Shannon's sampling theorem

Shannon's sampling theorem states that a bandlimited signal f with maximum frequency Ω is determined by its values (samples) taken at twice Ω or $f(n\pi/\Omega)$. More precisely, define H_Ω as the Hilbert space of bandlimited functions of maximum bandwidth Ω, i.e.

$$H_\Omega = \{f \in L^2(\mathbb{R}); \hat{f}(\omega) = 0 \ for \ |\omega| > \Omega\}. \tag{A.25}$$

We define $T = \pi/\Omega$ and

$$e_n = \frac{1}{\sqrt{T}} \left(\frac{\sin(\frac{\pi t}{T} - n\pi)}{\frac{\pi t}{T} - n\pi} \right) \qquad (A.26)$$

Theorem 14 *(The Shannon sampling theorem). The $\{e_n\}_{n \in \mathbb{Z}}$, as defined above, constitute an orthonormal basis for H_Ω. Also, $< f, e_n >= \sqrt{T}f(nT)$. Thus*

$$f(t) = \sum_{n=-\infty}^{\infty} < f, e_n > e_n(t) = \sum_{n=-\infty}^{\infty} \sqrt{T}f(nT)e_n(t). \qquad (A.27)$$

A.5.2 The Nyquist frequency

When f contains frequency components up to the frequency $\Omega = \pi/T$, T is the maximum sampling period of f and $\Omega_s = 2\Omega$ is the minimum sampling frequency, often called the *Nyquist* frequency. This can be proven from the relation $\Omega < \Omega_B \Leftrightarrow H_\Omega \subset H_{\Omega_B}$. We have to sample with at least this frequency to avoid *aliasing*. Aliasing is a spectrum overlap where high frequencies can appear as low frequencies. For more details on this topic and a proof of the above theorem see (Dodson, 1992; Vetterli et al., 1995).

A.6 Noise theory

Let $n(t)$ be a random process in continous time ($t \in \mathbb{R}$). For each time $t \in \mathbb{R}$, we denote the probability density function of $n(t)$ as $p(n(t))$. The expectation value is given by

$$E[n(t)] = \int_{-\infty}^{\infty} n(t)p(n(t))d\{n(t)\}. \qquad (A.28)$$

We define the *autocorrelation function* of n as

$$R_n(t_1, t_2) = E[n(t_1)n(t_2)], \qquad (A.29)$$

where t_1 and t_2 denote two values of the time at which the process is observed.

A stochastic process $n(t)$ is *stationary* if its probability density function is independent of t, ie $p(n(t_1)) = p(n(t_2))$. For a stationary stochastic process the autocorrelation function depends only on the time difference $\tau = t_1 - t_2$,

$$R_n(\tau) = E[n(t)n(t + \tau)]. \qquad (A.30)$$

For a stationary random process we define the *power spectral density* $S_n(\omega)$ as the Fourier transform of the autocorrelation function $R_n(t)$.

Definition 7 *The zero mean stationary stochastic process $n(t)$ is defined as **white** if it is uncorrelated, i.e.*

$$R_n(t) = \frac{N_0}{2}\delta(t). \tag{A.31}$$

Alternatively we have the formulation $S_n(\omega) = \frac{N_0}{2}$, the power spectral density is constant. The power spectral density of a white process is customarily denoted as $\frac{N_0}{2}$.

B ADDITIONAL PROOFS

B.1 Proof of theorem 3

We start with the scaling identity, $\phi(t) = \sqrt{2}\sum_{n\in\mathbb{Z}} h_0[n]\phi(2t-n)$ (see section 3.5.1). Now,

$$
\begin{aligned}
\phi_{j,k}(t) &= 2^{-\frac{j}{2}}\phi(2^{-j}t-k) = 2^{-\frac{j}{2}+\frac{1}{2}}\sum_{n\in\mathbb{Z}} h_0[n]\phi(2(2^{-j}t-k)-n) \\
&= 2^{-\frac{(j-1)}{2}}\sum_{n\in\mathbb{Z}} h_0[n-2k]\phi(2^{-(j-1)}t-n) \\
&= \sum_{n\in\mathbb{Z}} h_0[n-2k]\phi_{j-1,n}(t).
\end{aligned} \tag{B.1}
$$

Thus

$$
\begin{aligned}
c_j[k] &= <f,\phi_{j,k}> \\
&= \sum_{n\in\mathbb{Z}} \overline{h_0[n-2k]} <f,\phi_{j-1,n}> = \sum_{n\in\mathbb{Z}} \overline{h_0[n-2k]}c_{j-1}[n].
\end{aligned} \tag{B.2}
$$

By the wavelet identity (section 3.5.1), we similarly obtain

$$\psi_{j,k}(t) = \sum_{n\in\mathbb{Z}} h_1[n-2k]\phi_{j-1,n}(t). \tag{B.3}$$

Thus

$$d_j[k] = <f,\phi_{j,k}> = \sum_{n\in\mathbb{Z}} \overline{h_1[n-2k]}c_{j-1}[n]. \tag{B.4}$$

This concludes the proof.

B.2 Proof of theorem 4

We want to calculate

$$c_{j-1}[k] = <f, \phi_{j-1,k}>. \tag{B.5}$$

Formula 3.34 states

$$L^2(\mathbb{R}) = V_{j-1} \oplus \left(\bigoplus_{k=-\infty}^{j-1} W_k \right). \tag{B.6}$$

By decomposing $f \in L^2(\mathbb{R})$ according to this formula, all components coming from the W_k spaces are orthogonal to $\phi_{j-1,k}$. Thus the only components contributing to the inner product in equation (B.5) are the components of f in the space V_{j-1}, in other words, the projection $P_{j-1}f$. Thus

$$c_{j-1}[k] = <f, \phi_{j-1,k}> = <P_{j-1}f, \phi_{j-1,k}>. \tag{B.7}$$

Furthermore, since $V_{j-1} = V_j \oplus W_j$ and $\{\psi_{j,k}\}_{k \in \mathbb{Z}}$ is an orthonormal basis for W_j, we have

$$
\begin{aligned}
P_{j-1}f &= P_j f + \sum_{n \in \mathbb{Z}} <f, \psi_{j,n}> \psi_{j,n} \\
&= \sum_{n \in \mathbb{Z}} c_j[n] \phi_{j,n} + \sum_{n \in \mathbb{Z}} d_j[n] \psi_{j,n}.
\end{aligned} \tag{B.8}
$$

Thus

$$
\begin{aligned}
c_{j-1}[k] &= <P_{j-1}f, \phi_{j-1,k}> \\
&= \sum_{n \in \mathbb{Z}} c_j[n] <\phi_{j,n}, \phi_{j-1,k}> + \sum_{n \in \mathbb{Z}} d_j[n] <\psi_{j,n}, \phi_{j-1,k}>.
\end{aligned} \tag{B.9}
$$

Furthermore, by (B.1) and (B.3),

$$
\begin{aligned}
c_{j-1}[k] &= \sum_{n \in \mathbb{Z}} c_j[n] \sum_{m \in \mathbb{Z}} h_0[m-2n] <\phi_{j-1,m}, \phi_{j-1,k}> \\
&+ \sum_{n \in \mathbb{Z}} d_j[n] \sum_{m \in \mathbb{Z}} h_1[m-2n] <\phi_{j-1,m}, \phi_{j-1,k}> \\
&= \sum_{n \in \mathbb{Z}} c_j[n] \sum_{m \in \mathbb{Z}} h_0[m-2n] \delta_{m,k} \\
&+ \sum_{n \in \mathbb{Z}} d_j[n] \sum_{m \in \mathbb{Z}} h_1[m-2n] \delta_{m,k} \\
&= \sum_{n \in \mathbb{Z}} h_0[k-2n] c_j[n] + \sum_{n \in \mathbb{Z}} h_1[k-2n] d_j[n].
\end{aligned} \tag{B.10}
$$

$\square$

C MORE EXPERIMENTAL RESULTS

This appendix contains more plots from the numerical experiments.

C.1 ROC curves for maximum transform coefficient detectors
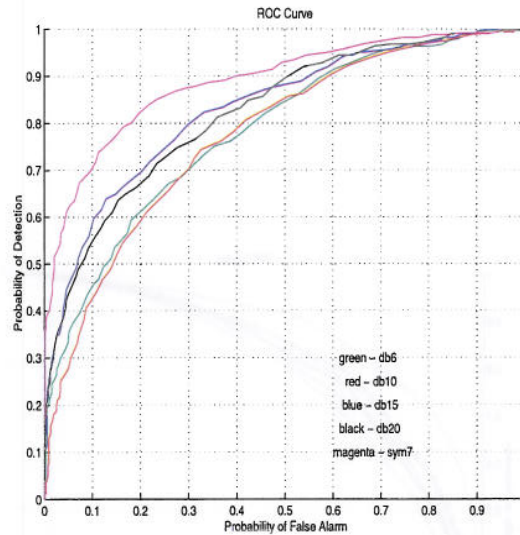
Figure C.1: ROC curves for the maximum transform coefficient detector in the detection of the synthetic transient of figure 2.4. A level 4 decomposition was performed using the wavelets db6, db10, db15, db20 and sym7
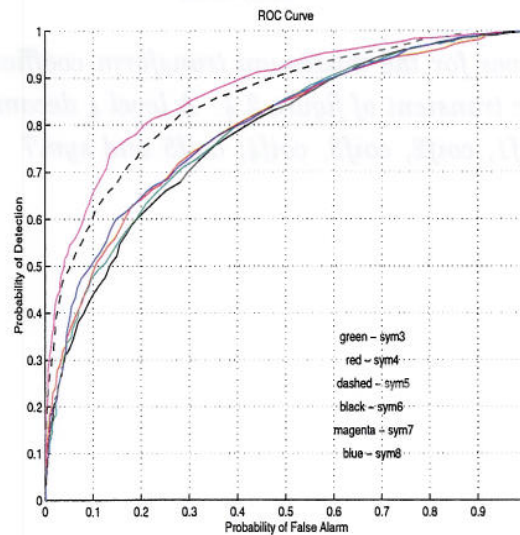
Figure C.2: ROC curves for the maximum transform coefficient detector in the detection of the synthetic transient of figure 2.4. A level 4 decomposition was performed using the wavelets sym3, sym4, sym5, sym6, sym7 and sym8
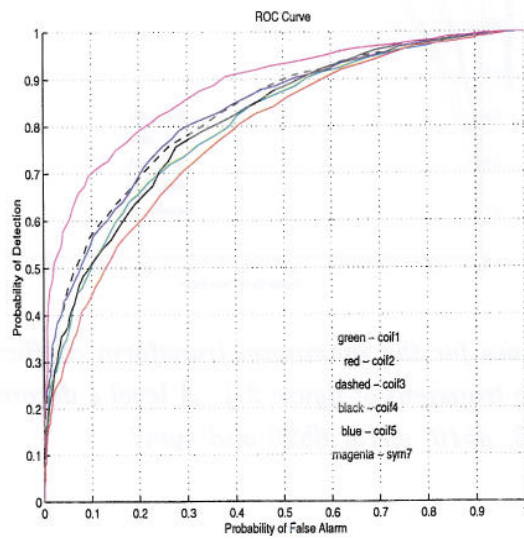
Figure C.3: ROC curves for the maximum transform coefficient detector in the detection of the synthetic transient of figure 2.4. A level 4 decomposition was performed using the wavelets coif1, coif2, coif3, coif4, coif5 and sym7

C.2 The energy distribution of the coefficients
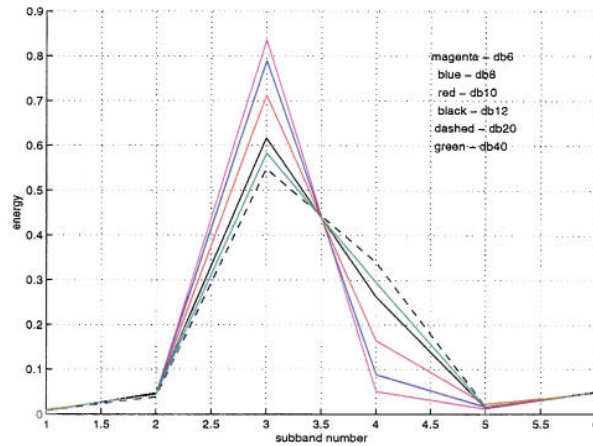
Figure C.4: The subband energy distribution for the discrete wavelet decomposition of the empirical transient of figure 2.4, using the db6, db8, db10, db12, db20 and db40 wavelets
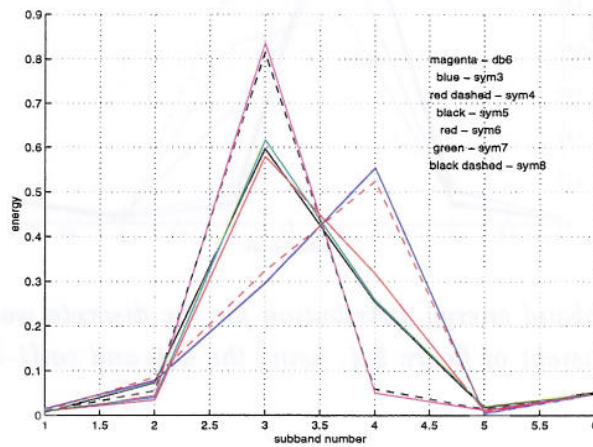
Figure C.5: The subband energy distribution for the discrete wavelet decomposition of the empirical transient of figure 2.4, using the db6 and sym3-8 wavelets
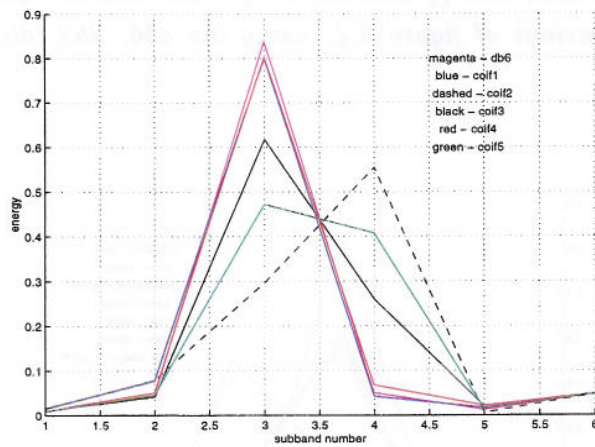
Figure C.6: The subband energy distribution for the discrete wavelet decomposition of the empirical transient of figure 2.4, using the db6 and coif1-5 wavelets

D   PLOTS OF WAVELETS

This appendix contains plots of some of the members of the wavelet families described in section 3.6. The plots were generated using the 'wavefun' command in the MatLab Wavelet Toolbox. The number of iterations in this routine was set to 8.
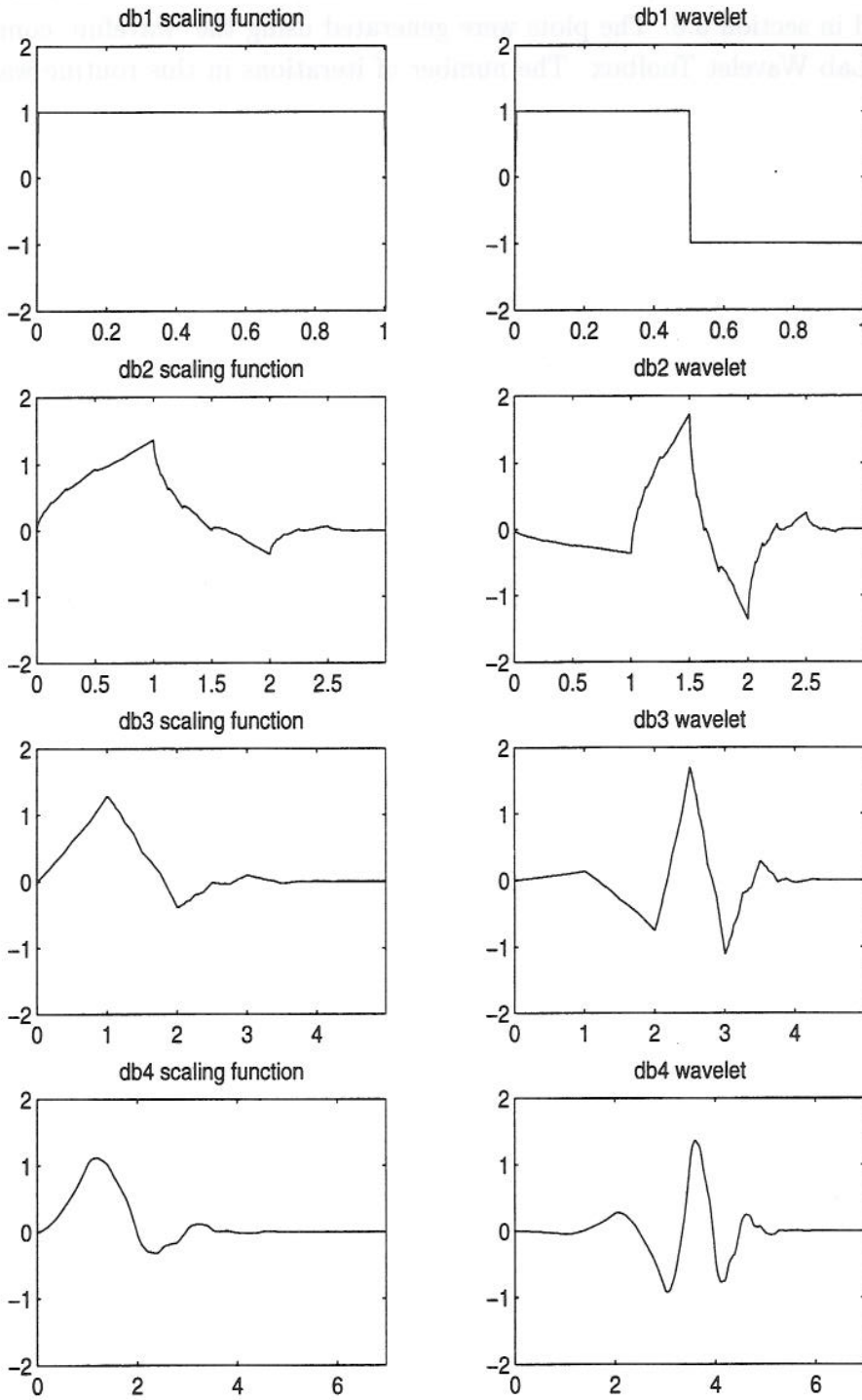
Figure D.1: Daubechies wavelets and scaling functions, $K = 1, 2, 3, 4$
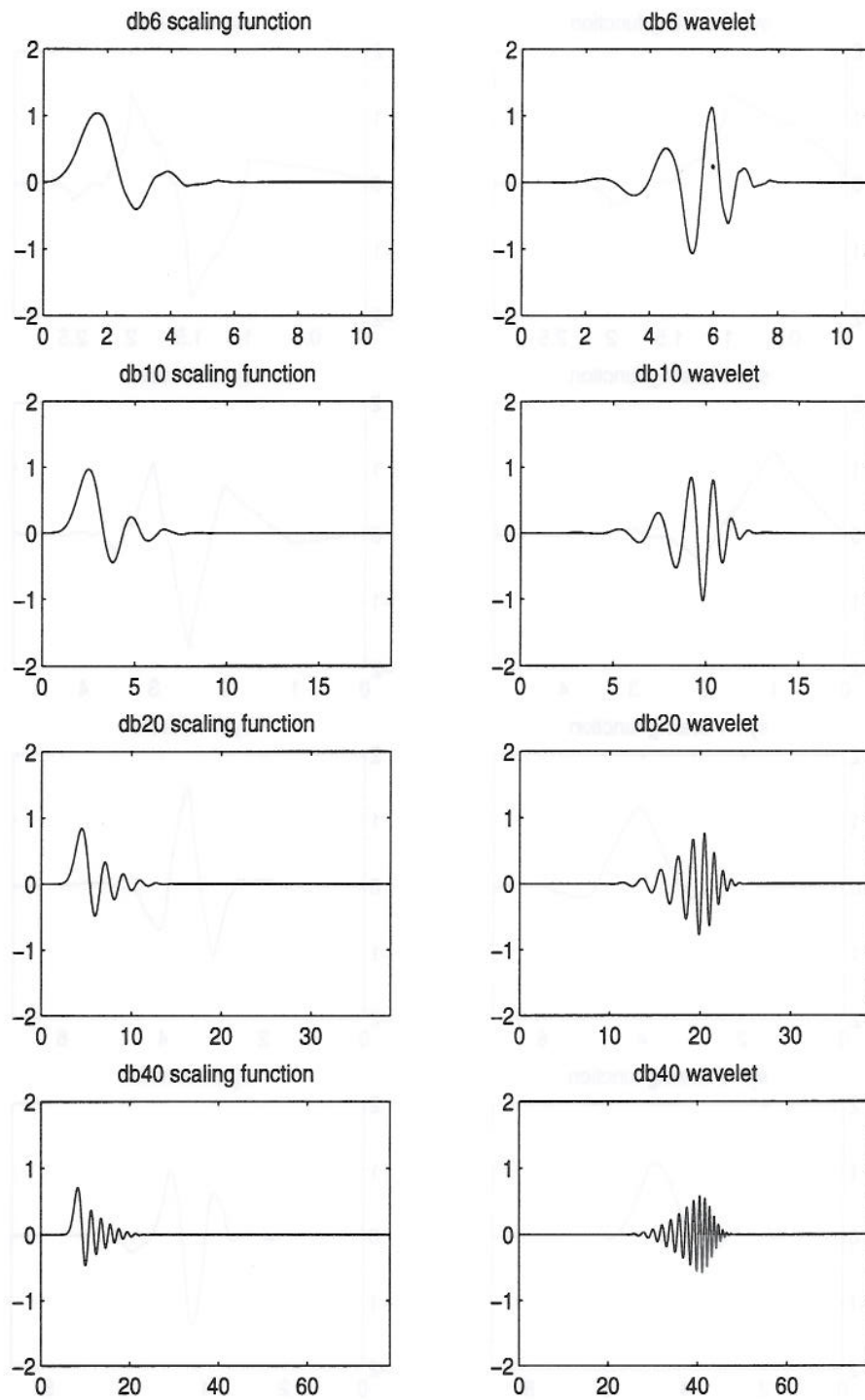
Figure D.2: Daubechies wavelets and scaling functions, $K = 6, 10, 20, 40$
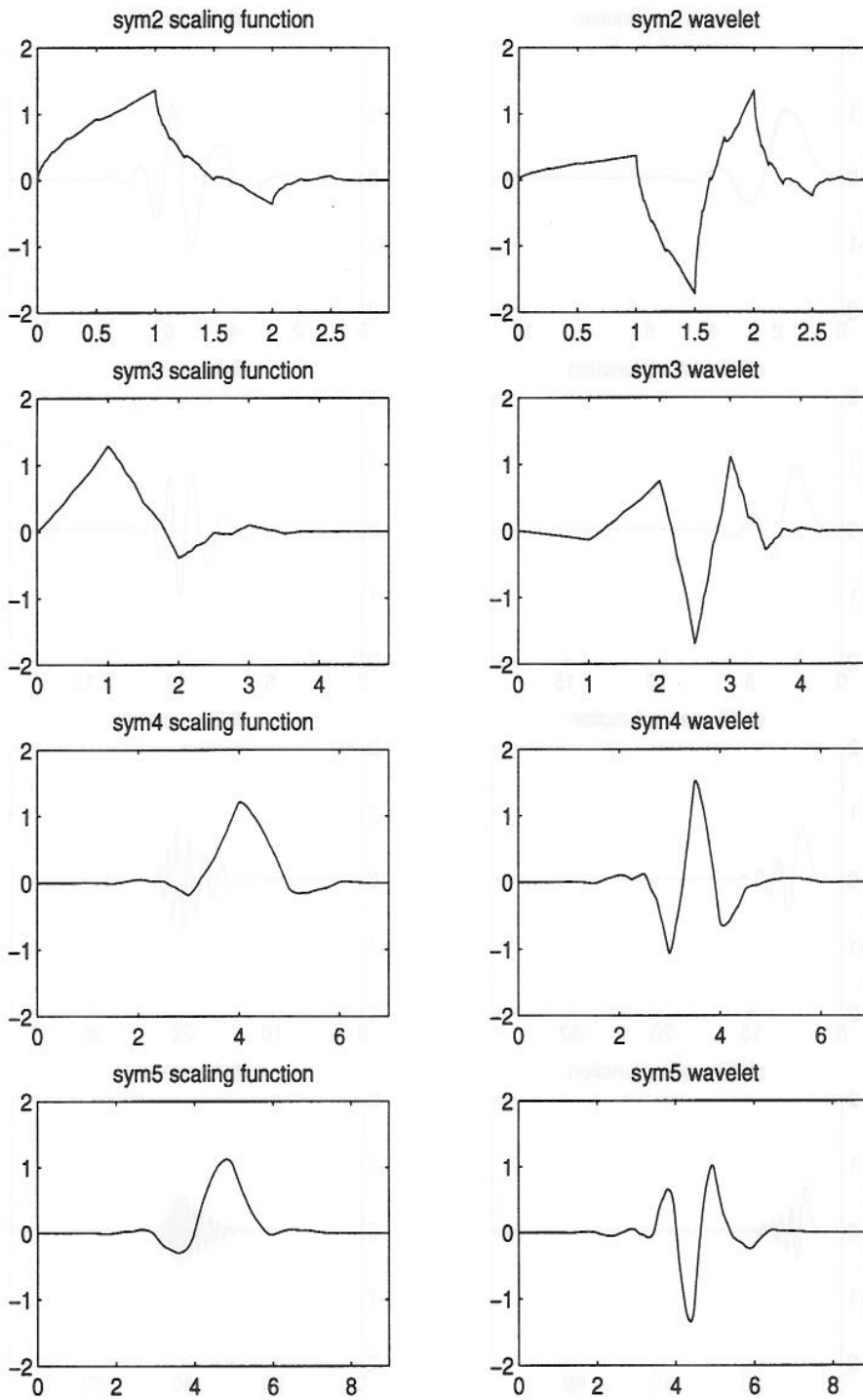
Figure D.3: Symlet wavelets and scaling functions, $K = 2, 3, 4, 5$
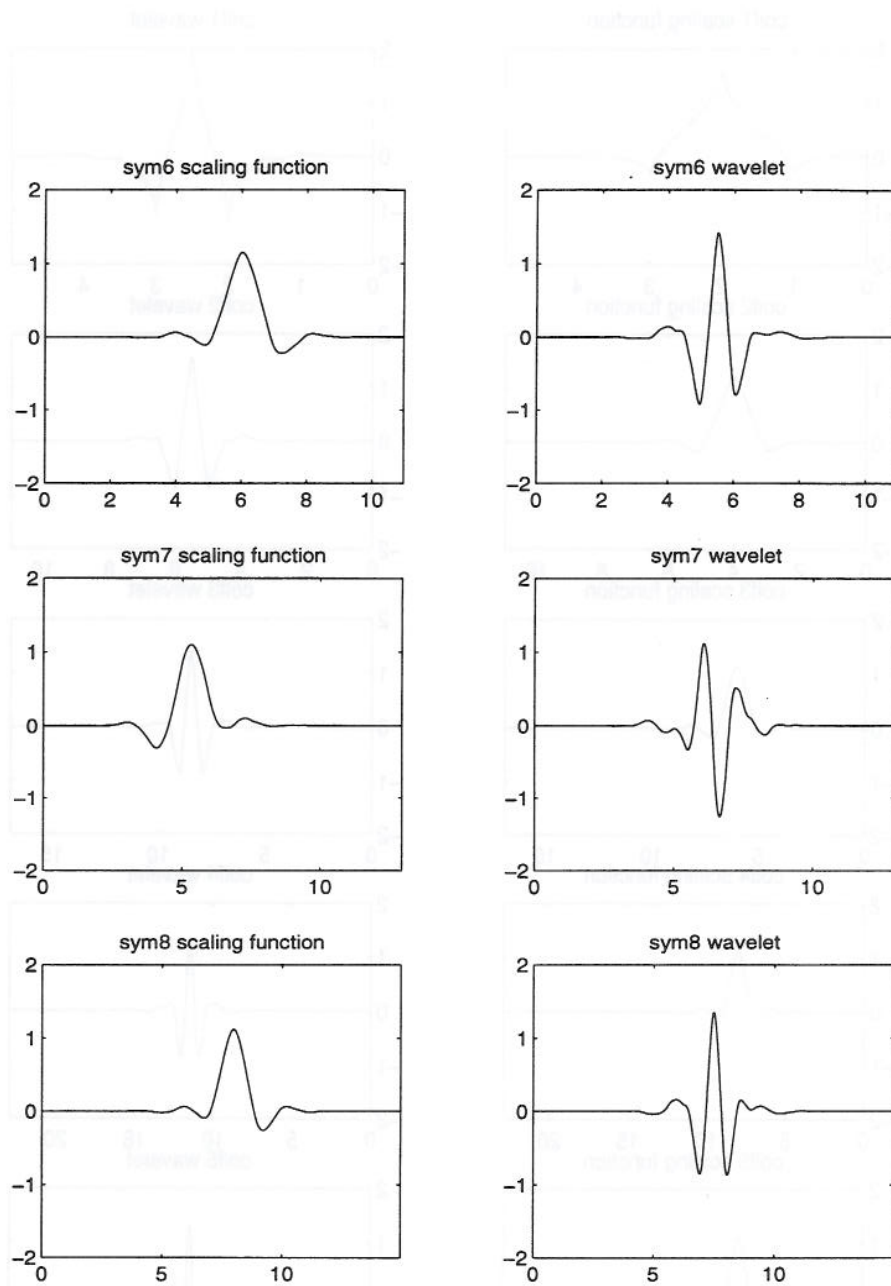
Figure D.4: Symlet wavelets and scaling functions, $K = 6, 7, 8$
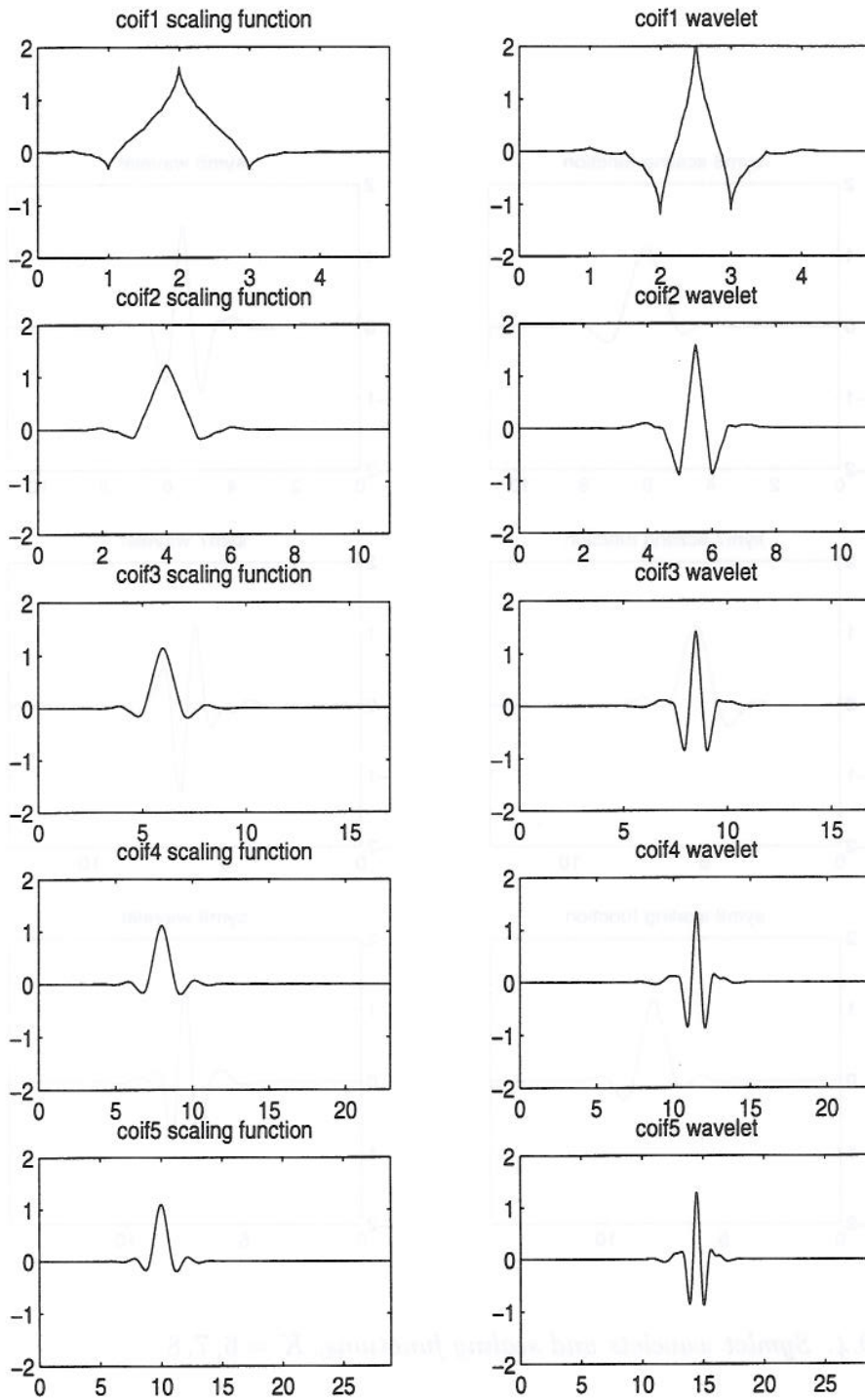
Figure D.5: Coiflet wavelets and scaling functions, $K = 1, 2, 3, 4, 5$
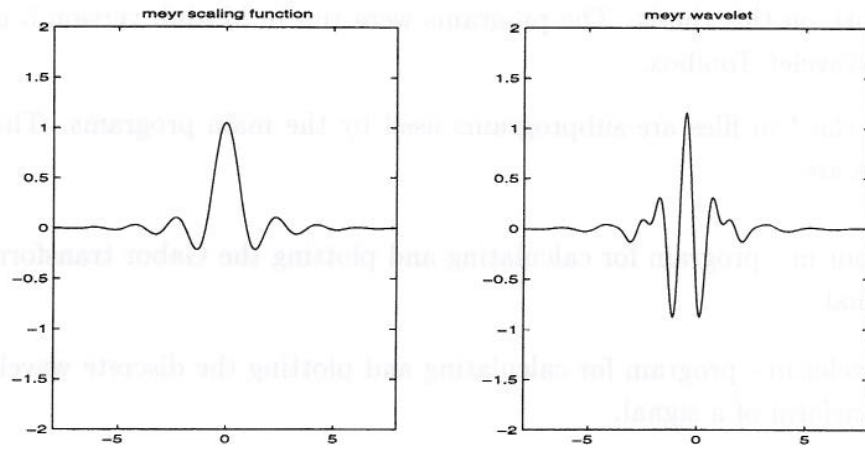
Figure D.6: Meyer wavelet and scaling function
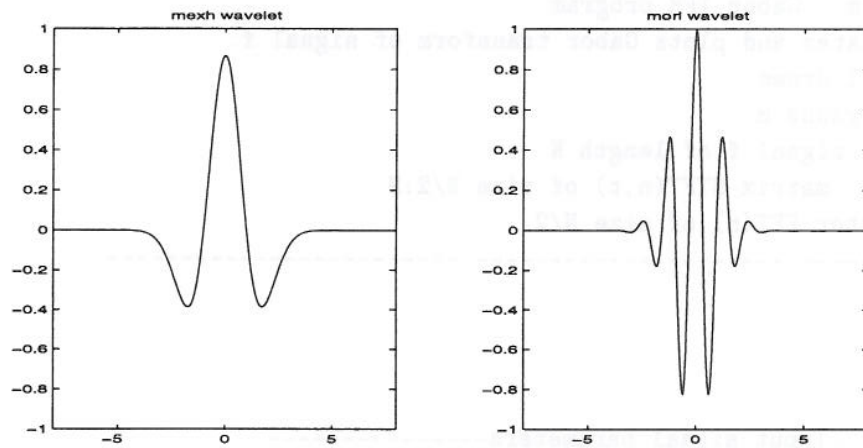
Figure D.7: Mexican hat and Morlet wavelets (no scaling function exists for these wavelets)

E MATLAB PROGRAMS

This appendix contains the most important Matlab programs that I have written in my work on this thesis. The programs were run in Matlab version 5 using the Matlab Wavelet Toolbox.

Many of the *.m files are subprograms used by the main programs. The main programs are

- gabor.m - program for calculating and plotting the Gabor transform of a signal.

- wavelet.m - program for calculating and plotting the discrete wavelet transform of a signal.

- plotenergy.m - program for plotting the energy distribution in the different subbands in the discrete wavelet transform of a signal.

- detect.m - performs transient detection using various detectors, both matched filters and others.

- ROC.m - calculates and plots ROC curves for various detectors.

- proto.m - gives statistics on the amplitudes of arriving transients.

```
% ------------------------------------------------------------
% gabor.m - Gabor TFR program
% Calculates and plots Gabor transform of signal f.
% By Emil Urnes
% Uses: gauss.m
% Input: signal f of length N
% Output: matrix STFT(n,t) of size N/2:N.
%     vector FFT(n) of size N/2.
% ------------------------------------------------------------

clear;
clf;

% --------Input signal parameters--------------
% winsize is the size of the gaussian window
winsize=0.5
% timesteps is the number of steps in time direction
% freqsteps is the number of steps in freq direction
timesteps=100
freqsteps=50
% [freq1, freq2] defines the frequency window
% where the STFT is shown. Maximum is [0,1]
freq1=0.0
```

```
freq2=0.5

% The input signal is given in f, of length N.
% N is the signal length. Must be even.

% PLOTTYPE=0 yields signal+FFT, 1 yields signal+STFT, 2 yields
% signal+FFT+STFT.
PLOTTYPE=0;

% fsamp: Sampling frequency
fsamp=600;

% You have two options for the input signal.
% LOADFROMDISK=1 means load signal from disk
LOADFROMDISK=1
% Option 1. Load from disk.
if LOADFROMDISK==1
  load ch4;
  f=ch4(203461:203740);
%  load signal2;
%  f=signal;
  N=length(f)
% Option 2. Create a test signal
else
  N=400
  for i=0:N-1,
    if i<N/2 f(i+1)=sawtooth(2*pi*i*20/N);
    else f(i+1)=sin(2*pi*i*60/N)
      +random('Normal',0,0.5)+sin(2*pi*i*40/N);
    end
  end
end

% ------- Main program ---------------------------

% First calculate FFT, results in FFTplot
F=fft(f,N);
i1=freq1*(N/2-1)+1;
i2=freq2*(N/2-1)+1;
FFTplot=abs(F([i1:i2]))';

% Make STFT
% Scale winsize to make it independent of signal length
winsize=0.001*winsize*N^2;
% Build Gaussian curve
n=1:2*N;
g(n)=gauss(n-N-1,winsize);
% for gauss2, 0<t<1 is equivalent to the length of transient.
% lTrans=110;
% g(n)=gauss2((n-N-1)/lTrans,2);
```

```
% Create the STFT matrix
% t is the time variable in the STFT
tinc=N/timesteps;
for j=0:timesteps-1;
%  j
  t=floor(j*tinc);
  % h is the signal f enveloped in a gauss curve g.
  h=f.*g([N+1-t:2*N-t]);
  H=fft(h,N);
  STFT(:,j+1)=abs(H([1:N/2]))';
end

% Extract results to plot from STFT, results in STFTplot
% Extract from i1 to i2 in STFT
i1=freq1*(N/2-1)+1;
i2=freq2*(N/2-1)+1;
istep=(i2-i1)/freqsteps;
if istep<1
  istep=1
  freqsteps=i2-i1+1;
end
for i=0:freqsteps-1
  freq=floor(i1+istep*i);
  STFTplot(i+1,:)=STFT(freq,:);
end;


% Generate time scale
k=1:N;
t(k)=(k-1)/fsamp;

% Generate freq scale
k=1:N*freq2/2;
fr(k)=freq1*fsamp/2+(k-1)*fsamp/(N);

% Plot the results
if PLOTTYPE==2 nplots=3;
else nplots=2;
end

subplot(nplots,1,1);
hold on
plot(t,f,'b');
gscale=max(f)/max(g);
if PLOTTYPE>0 plot(t,gscale*g([N/2+1:3*N/2]),'r');
end
xlabel('time (s)');
ylabel('signal (microstrain)');
axis([t(1) t(N) min(f) max(f)]);
```

```
grid
hold off

if (PLOTTYPE==0) ! (PLOTTYPE==2)
subplot(nplots,1,2);
plot(fr,FFTplot,'b');
xlabel('frequency (Hz)');
ylabel('FFT');
axis([fr(1) fr(N*freq2/2) min(FFTplot) max(FFTplot)]);
grid
end

if PLOTTYPE>0
  k=1:timesteps;
  tSTFT(k)=(k-1)*N/(fsamp*timesteps);
  k=1:freqsteps;
  fSTFT(k)=freq1*fsamp/2+(k-1)*freq2*fsamp/(2*freqsteps);
  subplot(nplots,1,PLOTTYPE+1);
  pcolor(tSTFT,fSTFT,STFTplot);
  xlabel('time');
  ylabel('frequency');
  title('Gabor transform');
  colormap(jet)
end

% ---------------------------------------------------------------
% wavelet.m - Discrete Wavelet Transform TFR program
% By Emil Urnes, FFI
% Based on the Wavelet Toolbox from the Mathworks
% Input: signal f of length N
% ---------------------------------------------------------------

clear;
clf;

% --------Input signal parameters---------------

% Type of wavelet used in decomposition:
type='db4';
% Number of detail levels:
level=5;
% cleararray: which levels to clear; first detail 1:J, then approx.
% 0 means clear the coefficents at that level.
% A reconstruction is plotted if this array is not identically zero.
cleararray=[1 1 1 1 1 1];

% plotarray: which levels to plot; first detail 1:J, then approx.
plotarray=[1 1 1 1 1 1];

% plotcoefs: 1 if coefficients are to be plot instead of projections
```

```
plotcoefs=0;
% if you plot coeffs, samescale=1 ensures same scale on each plot in
% x and y direction
% Warning: samescale=(1,0) is NOT supported
samescalex=0;
samescaley=1;

% fsamp: Sampling frequency
fsamp=600/6;

% The input signal is given in f, of length N.
% N is the signal length. Must be even.

load signal2;
f=signal;
N=length(f)

% ------- Main program --------------------------

% Calculate DWT
[C,L]=wavedec(f,level,type);


% Reconstruct the projections Pj and Qj's
% Approximation in A(i,t), detail in D(i,t).
% Coefficients in cAi(k) and cDi(k).
for i=1:level
  eval(['A(',int2str(i),',:)=wrcoef(''a'',C,L,type,i);']);
  eval(['D(',int2str(i),',:)=wrcoef(''d'',C,L,type,i);']);
  eval(['cA',int2str(i),'=appcoef(C,L,type,i);']);
  eval(['cD',int2str(i),'=detcoef(C,L,i);']);
end

% Clear some coefficients
index=1;
for i=1:level+1
  if cleararray(level+2-i)==0
    C(index:index+L(i)-1)=zeros(size(C(index:index+L(i)-1)));
  end
  index=index+L(i);
end

% Reconstruct signal
f0=waverec(C,L,type);

% Generate time scale
k=1:N;
t(k)=(k-1)/fsamp;

% *** Plot the results ***
```

```
% Make plot room for signal, approx/details and (evt) reconstruct.
nplots=1+norm(plotarray,1);
plotconstruct=~(norm(cleararray,1)==length(cleararray));
if plotconstruct
  nplots=nplots+1;
end

subplot(nplots,1,1);
if plotcoefs
  plot(f,'b');
  axis([0 N-1 min(f) max(f)]);
else
  plot(t,f,'b');
  axis([t(1) t(N) min(f) max(f)]);
end
ylabel('signal');
grid

% index keeps track of the current plot window
index=2;
if plotconstruct
 subplot(nplots,1,index);
 plot(t,f0,'b');
 ylabel('Reconstruct');
 axis([t(1) t(N) min(f) max(f)]);
 index=index+1;
 grid
end

% Plot highest approximation and all details.
Cmax=norm(C,inf);
for i=1:level+1
  if plotarray(level+2-i)==1
    subplot(nplots,1,index);
    index=index+1;
    if plotcoefs==0
      if i==level+1 % approx
        plot(t,A(level,:));
        eval(['ylabel(''A',int2str(level),''')']);
        xlabel('time (s)');
      else
        plot(t,D(i,:));
        eval(['ylabel(''D',int2str(i),''')']);
      end
      axis([t(1) t(N) min(f) max(f)]);
    else
      if i==level+1 % approx
        eval(['plot(cA',int2str(level),');']);
        eval(['ylabel(''c',int2str(level),''')']);
```

```
        xlabel('samples');
      else
        eval(['plot(cD',int2str(i),');']);
        eval(['ylabel(''d',int2str(i),''')']);
      end
      if samescalex==1
        axis([1 N -Cmax Cmax]);
      else
        if samescaley==1
          if i==level+1 eval(['axis([1 length(cA',
              int2str(level),') -Cmax Cmax]);']);
          else eval(['axis([1 length(cD',int2str(i),')
              -Cmax Cmax]);']);
          end
        else
          if i==level+1 eval(['axis([1 length(cA',
              int2str(level),') min(cA',int2str(level),')
              max(cA',int2str(level),')]);']);
          else eval(['axis([1 length(cD',int2str(i),')
              min(cD',int2str(i),') max(cD',int2str(i),')]);']);
          end
        end
      end
    end
    grid
  end
end

% Calculate Energy
% The variable energy contains energy in details 1..level and
% entry level+1 is the energy in the remaining V-space.

for i=1:level
  energy(i)=eval(['norm(cD',int2str(i),')^2']);
end
energy(level+1)=eval(['norm(cA',int2str(level),')^2']);
totEn=sum(energy);
norm(f)^2

% ------------------------------------------------------------
% plotenergy.m - Plots the distribution of energy in wavelet bands
% for different types of wavelets.
% By Emil Urnes, FFI
% Based on the Wavelet Toolbox from the Mathworks
% Uses: calcEn.m
% ------------------------------------------------------------

clear;
clf;
```

```
% Load signal to be analyzed and normalize energy.
load transient2;
signal=transient;
signal=signal/norm(signal,2);

% ---------------- Main prog. ------------------

level=5;

type='db6';
calcEn;
en(1,:)=energy;

type='db4';
calcEn;
en(2,:)=energy;

type='db3';
calcEn;
en(3,:)=energy;

type='sym7';
calcEn;
en(4,:)=energy;

type='coif4';
calcEn;
en(5,:)=energy;

type='coif5';
calcEn;
en(6,:)=energy;

type='sym8';
calcEn;
en(7,:)=energy;


% Plot results
clf
subplot(1,1,1)
hold on
plot(en(1,:),'m');
plot(en(2,:),'b');
plot(en(3,:),'k--');
plot(en(4,:),'k');
plot(en(5,:),'r');
plot(en(6,:),'g');
%plot(en(7,:),'k--');
text(4.5,0.8,'magenta - db6');
```

```
text(4.5,0.775,' blue - coif1');
text(4.5,0.75,' dashed - coif2');
text(4.5,0.725,' black - coif3');
text(4.5,0.70,' red - coif4');
text(4.5,0.675,' green - coif5');
%text(4.5,0.65,'black dashed - sym8');
grid
hold off



% ----------------------------------------------------------
% calcEn.m - calculates energy in wavelet bands
% By Emil Urnes, FFI
% Based on the Wavelet Toolbox from the Mathworks
% Uses: calcEn.m
% Input: level, type
% Output: energy
% ----------------------------------------------------------


% Calculate DWT
[C,L]=wavedec(signal,level,type);


% Coefficients in cAi(k) and cDi(k).
for i=1:level
  eval(['cA',int2str(i),'=appcoef(C,L,type,i);']);
  eval(['cD',int2str(i),'=detcoef(C,L,i);']);
end


% Calculate Energy
% The variable energy contains energy in details 1..level and
% entry level+1 is the energy in the remaining V-space.

for i=1:level
  energy(i)=eval(['norm(cD',int2str(i),')^2']);
end
energy(level+1)=eval(['norm(cA',int2str(level),')^2']);



% --------------------------------------------------------------
% detect.m - Detection of transients
% Made by Emil Urnes, FFI
% Input: x(n): data to be analyzed, si(n): typical transients
% Output: detector outputs
% Uses: detect_MF_time.m, detect_MF_wav.m, detect_wavcof.m,
%   detect_ED.m, detect_gabor.m, detect_WTx.m, detect_WTs.m,
%   genbank.m
% --------------------------------------------------------------


clear;
```

```
% BANK is >0 if a bank of filters is to be used, 0 otherwise.
% BANK==1 gives the usual plot
% BANK==2 gives alternative plot
BANK=0;


% --------Input signal parameters-------------------

% fsamp: Sampling frequency
fsamp=600/20;

% ** Wavelet parameters **
% type: Type of wavelet used in decomposition:
% J: Number of levels in wavelet dec.
% same type and J has to be used in one simulation.
type='db6';
J=5;


% Tree more parameters, detail, approx and AMPNORM
% have to be set for detectors detect_MF_wav
% and detect_wavcof, individually in the main program.
% Here is an explanation:
% detail: Use detail of level i in MF; 1=yes, 0=no
% first number is lowest detail, ie 1 2 3 .. J.
% approx: Use highest level approx in MF; 1=yes, 0=no
% AMPNORM=1 for amplitude normalization, 0 otherwise.


% ** Wavelet thresholding parameters **
% thrDetail: detail threshold level. Use 0 for no thresholding
% thrApprox: approximation threshold level. Use 0 for no thresholding
% thrType: 'h'=hard, 's'=soft
% thrTransient: =1 if the transient should also be thresholded

thrDetail=[0. 0. 0. 0. 0.];
thrApprox=0.;
thrType='s';
TPTR='heursure';
SCAL='mln';
thrTransient=0;


% -----------Load input data and transients------------

% x(n) is our input to be analyzed

%load ch4dec20;
%x=ch4dec20;
load signal1;
x=signal;

% sig(i,n) are the signals to look for in our input,
```

```
% of lengths Nsig(i).

% sig(1) is the first appearing transient in the data
load transient1;
sig(1,:)=transient;
Nsig(1)=length(sig(1,:));

% sig(2)..sig(Nbank+1) are synthetic ~2.5Hz transients
% fetch them from bank.mat, generated by genbank.m
load bank
Nbank=length(bank(:,1));
for i=1:Nbank,
  sig(i+1,:)=bank(i,:);
  Nsig(i+1)=length(sig(i+1,:));
end

% --------- Main program --------------------------

Nx=length(x);

% Calculate DWT of all shifts of x.
detect_WTx

if BANK>0
  Ndetectors=Nbank; % no of detectors.
  for det=1:Ndetectors,
    % Detector det: ordinary time domain MF with transient det
    eval(['title',int2str(det),'=['' MF - ''
      num2str(fsamp/Nsig(det)*freq(det),3) '' Hz''];']);
    s=sig(det+1,:);
    N=Nsig(det+1);
    AMPNORM=0;
    detail=[0 1 1 1 0];
    approx=0;
    detect_WTs;
    detect_MF_wav;
    %detect_MF_time;
    yOut(det,:)=abs(y).^2;
  end
else % BANK==0
  % See possibilities in detect.tmp.m!!
  Ndetectors=3;

  % Detector 1: Matched filter
  title1='Matched Filter';
  s=sig(1,:);
  N=Nsig(1);
  detect_MF_time;
  yOut(1,:)=abs(y).^2;
```

```
% Detector 2: Matched filter
title2='MF band 2';
s=sig(1,:);
N=Nsig(1);
AMPNORM=1;
detail=[0 1 0 0 0];
approx=0;
detect_WTs;
detect_MF_wav;
yOut(2,:)=abs(y).^2;


% Detector 3: Wavelet maxcof detector
title3='Coef. band 2';
N=Nsig(1);
type='db4';
detail=[0 1 0 0 0];
approx=0;
detect_wavcof;
yOut(3,:)=y;
end


% save data for later use
%save results/wang2 x s yOut;


% Generate time scale
k=1:Nx;
time(k)=(k-1)/fsamp;


clf
if BANK==2
  % ** Alternative: evaluate bank of filters **
  for det=1:Ndetectors,
    %ySum(det)=sum(yOut(det,:));
      ySum(det)=yOut(det,75);
  end
  subplot(1,1,1)
  plot(fsamp/Nsig(2)*freq,ySum)
  xlabel('frequency (Hz)');
  ylabel('detector output');
  grid
  %save ySum4 ySum freq
else
  % **** Plot signal+detector results ****
  for i=0:Ndetectors,
    subplot(Ndetectors+1,1,i+1)
    if i==0
      plot(time,x)
      ylabel('signal');
      axis([time(1) time(Nx) min(x) max(x)]);
    else
```

```
        hold on
        if i==5 plot(time,g([Nx/2+1:3*Nx/2])/max(g),'r');
        end
        plot(time,yOut(i,:));
        eval(['ylabel(title',int2str(i),');']);
        axis([time(1) time(Nx) 0 max(yOut(i,:))]);
        if i==Ndetectors
          xlabel('time (s)');
        end
      end
    end
    grid
  end
end


% ------------------------------------------------------------
% detect_MF_time.m - Time domain Matched Filter detection
% Made by Emil Urnes, FFI
% Output: y(n): Normed MF time domain
% ------------------------------------------------------------

% These input parameters must be pre-defined in global
% workspace (see MF.m)

% x(n) is our input to be analyzed
% s(n) is the signal to look for in our input, of length N
% N=length(s)

% ------- Calculate time-domain MF
Es=norm(s)^2;
lengthY=length(x)-N+1;
for i=1:length(x),
  if (i>lengthY) y(i)=0;
    else y(i)=sum(x(i:i+N-1).*s)/Es;
  end
end


% ------------------------------------------------------------
% detect_MF_wav.m - Wavelet Matched Filter detection of transients
% Made by Emil Urnes, FFI
% Output:
%         y(n): Normed Parseval wavelet domain MF output
% ------------------------------------------------------------

%This routine calculates the MF output.

if AMPNORM
  % Find the wavelet norm of the transient, for
  % amplitude normalization.
  % Es contains the transient energy.
```

```matlab
% We start with highest J approximation.
  if (approx==1)
   eval(['Es=sum(cs',int2str(J),'.*cs',int2str(J),');']);
  else Es=0;
  end
  % now do the details from j=1:J
  for j=1:J,
    if (detail(j)==1)
      eval(['Es=Es+sum(ds',int2str(j),'.*ds',int2str(j),');']);
    end
  end
else
  Es=norm(s)^2;
end

% Do the wavelet domain matched filtering, output in y.
lengthY=length(x)-N+1;
for t=0:lengthY-1
% tot contains the inner product.
% We start with highest J approximation.
  if (approx==1)
    [a, b] = factor2j(t,J);
    eval(['ls=length(cs',int2str(J),');']);
    eval(['tot=sum(cs',int2str(J),'.*c',
      int2str(b),'x',int2str(J),'(a+1:a+ls));']);
  else tot=0;
  end
  % now do the details from j=1:J
  for j=1:J,
    if (detail(j)==1)
      [a, b] = factor2j(t,j);
      eval(['ls=length(ds',int2str(j),');']);
      eval(['tot=tot+sum(ds',int2str(j),'.*d',int2str(b),'x',
        int2str(j),'(a+1:a+ls));']);
    end
  end
  y(t+1)=tot/Es;
end
for t=lengthY+1:length(x)
 y(t)=0;
end

% ------------------------------------------------------------
% detect_wavcof.m - Wavelet Max coefficient detection
% Made by Emil Urnes, FFI
% Output: y(n): max wav. coefficient MF output
%         f(n): index of maxcoeff; 0=approx, 1..J=detail
% ------------------------------------------------------------

% These input parameters must be pre-defined in global
```

```
% workspace (see MF.m)

% x(n) is our input to be analyzed
% s(n) is the signal to look for in our input, of length N
% N=length(s)

% type: Type of wavelet used in decomposition:
% detail: Use detail of level i in MF; 1=yes, 0=no
% first number is lowest detail, ie 1 2 3 .. J.,
% f. ex. detail=[0 0 1 0 0];
% approx: Use highest level approx in MF; 1=yes, 0=no
% thrDetail: detail threshold level. Use 0 for no thresholding
% thrApprox: approximation threshold level. Use 0 for no thresholding
% thrType: 'h'=hard, 's'=soft
% thrTransient: =1 if the transient should also be thresholded
% J: Number of levels in wavelet dec., J=length(detail)


% ------- Main program --------------------------


% Do maximum coefficient detector
lengthY=length(x)-N+1;
for t=0:length(x)-1
% compensate for wavelet offset
woff=0;
  % Find max coefficient for k=2^{J-j}a, all j, first approx:
  if (approx==1)
    [a, b] = factor2j(t,J);
    eval(['maxcof=abs(c',int2str(b),'x',int2str(J),
      '(1+a+woff))^2;']);
  else maxcof=0;
  end
  index_maxcof=0;
  % Now detail:
  for j=1:J,
  if (detail(j)==1)
    [a, b] = factor2j(t,j);
    eval(['maxcof_candidate=abs(d',int2str(b),'x',int2str(j),
      '(1+a+woff))^2;']);
    if maxcof_candidate>maxcof
      maxcof=maxcof_candidate;
      index_maxcof=j;
    end
  end
  end
  f(t+1)=index_maxcof;
  y(t+1)=maxcof;
end
```

```
% ------------------------------------------------------------
% detect_ED.m - Energy detection of transients
% Made by Emil Urnes, FFI
% Output: y(n): energy detector
% ------------------------------------------------------------

% These input parameters must be pre-defined in global
% workspace (see MF.m)

% x(n) is our input to be analyzed
% N=length(s), where s is the signal to look for in our input

% ------- Calculate energy output
Es=norm(s)^2;
lengthY=length(x)-N+1;
for i=1:length(x),
  if (i>lengthY) y(i)=0;
    else y(i)=norm(x(i:i+N-1))^2/Es;
  end
end

% ------------------------------------------------------------
% detect_gabor.m - Gabor max coefficient detection of transients
% Made by Emil Urnes, FFI
% Output: y(n): maxcoeff detector
%         f(n): frequency of maxcoeff
% Uses: gauss.m, gauss2.m
% ------------------------------------------------------------

% These input parameters must be pre-defined in global
% workspace (see MF.m)

% x(n) is our input to be analyzed
% N=length(s) is the length of the signal to look for in our input
% window: type of window: 0==gabor, 1==transient type.
% damp: damping factor for window.

% ------- Calculate gabor detector

Nx=length(x);
% Build Gaussian curve
n=1:2*Nx;
% for gauss2, 0<t<1 is equivalent to the length of transient.
% The curve starts at N+1
if window==0
  g(n)=gauss((n-Nx-1)/N,damp);
else
  g(n)=gauss2((n-Nx-1)/N,damp);
end
```

```
lengthY=length(x)-N+1;
for i=1:length(x),
    h=x.*g([Nx+2-i:2*Nx+1-i]);
    H=fft(h,Nx);
    Hstft=abs(H([1:floor(Nx/2)]));
    [y(i),freq]=max(Hstft);
    f(i)=freq(1); % In case of multiple max, choose first.
end
% Normalize output
y=y/max(y);

% ------------------------------------------------------------------
% detect_WTx.m -
% Calculates the wavelet coefficients of all shifts
% of the input sequence x.
% The number of coefficients is optimally (J+1)N, but
% this prototype routine calculates N2^J coefficients!
% (ie serveral coefficients are calculated twice)
% input:
%   x - input sequence
%   J - no. of levels
%   type - wavelet type
% output:
%   Cix,Lix - DWT of x with shift i=0:2^J-1
%   cixJ,dixj - coefficients of x at scale j with a shift of i.
% By Emil Urnes, FFI.
% Uses: lshift.m
% ------------------------------------------------------------------

% Calculate DWT of 2^J left shifts of x, Cix/Lix
for i=0:2^J-1
  eval(['[C',int2str(i),'x,L',int2str(i),'x]=
      wavedec(lshift(x,i),J,type);']);
end

% Automatic denoising by wden, see 8-170 in Wavelet Toolbox guide
%dummy=x;
%for i=0:2^J-1
%  eval(['[dummy,C',int2str(i),'x,L',int2str(i),'x]=
  wden(C',int2str(i),'x,L',int2str(i),'x,TPTR,thrType,
      SCAL,J,type);']);
%end

% Put approx/detail coefficients in cixJ(k)/dixj(k)
% for all shifts i.
for i=0:2^J-1,
  eval(['c',int2str(i),'x',int2str(J),'=
    appcoef(C',int2str(i),'x,L',int2str(i),'x,type,J);']);
  for j=1:J,
    eval(['d',int2str(i),'x',int2str(j),'=
```

```
        detcoef(C',int2str(i),'x,L',int2str(i),'x,j);']);
  end
end


% -----------------------------------------------------------------
% detect_WTs.m
% Calculates the wavelet transform of the transient s
% input:
%   s - transient
%   J - no. of levels
%   type - wavelet type
% output:
%   Cs,Ls  - DWT of s
%   csJ,dsj - coefficients of s at scale j
% By Emil Urnes, FFI.
% -----------------------------------------------------------------

% Calculate DWT of s
[Cs,Ls]=wavedec(s,J,type);

% Automatic denoising by wden, see 8-170 in Wavelet Toolbox guide
%dummy=x;
%if thrTransient==1
%   [dummy,Cs,Ls]=wden(Cs,Ls,TPTR,thrType,SCAL,J,type);
%end

% Put approx/detail coefficients in csJ(k)/dsj(k).
eval(['cs',int2str(J),'=appcoef(Cs,Ls,type,J);']);
for j=1:J,
  eval(['ds',int2str(j),'=detcoef(Cs,Ls,j);']);
end


% -----------------------------------------------------------
% lshift.m - left shift of the vector x by n
% Made by Emil Urnes, FFI.
% -----------------------------------------------------------

function X = lshift(x,n);
if (n>=length(x)) ERROR_lshift_vectortoolong
else
  X(1:length(x)-n)=x(n+1:length(x));
end


% -----------------------------------------------------------
% gauss.m - returns centered Gauss curve
% Made by Emil Urnes, FFI.
% -----------------------------------------------------------

function [g] = gauss(x,a)
```

```
g = sqrt(1/(2*pi*a))*exp(-x.^2/(2*a));


% ----------------------------------------------------------
% gauss2.m - returns one sided Gauss curve
% Made by Emil Urnes, FFI.
% ----------------------------------------------------------


function [g] = gauss2(x,a)
g=(x>=0).*exp(-a*x);


% ----------------------------------------------------------
% genbank.m - Generates bank of synthetic transient
% signals of length N
% The bank is generated with a spectrum of frequencies
% All transients have L2 norm 1
% By Emil Urnes, FFI
% ----------------------------------------------------------


clear bank;

% ----- Input parameters --------
load transient1;
N=length(transient);
damp=3;
freq=5.1:1:9.1;
t0=0.0;
fsamp=600/20;

% ------ Main Program -----------
sizebank=length(freq);
for k=1:sizebank,
  for i=0:N-1,
    t=i/N;      % t in [0,1]
    bank(k,i+1)=-trans(t-t0,damp,freq(k));
  end
  bank(k,:)=norm(transient,2)*bank(k,:)/norm(bank(k,:),2);
end

save bank bank freq

% ----- Plot data --------
% Generate time scale
k=1:N;
t(k)=(k-1)/fsamp;

clf
hold on
subplot(1,1,1)
plot(t,bank(3,:),'b');
```

```
plot(t,transient,'r')
grid
xlabel('time (s)');
ylabel('signal (microstrain)');


% -----------------------------------------------------------
% processMF.m -
% Removes oscillations from MF output by wavelet technique
% By Emil Urnes, FFI.
% wavtype: Type of wavelet used in decomposition:
% cleararray: which levels to clear; first detail 1:J, then approx.
% 0 means clear the coefficents at that level.
% mf0: input matched filter
% mf1: output matched filter
% -----------------------------------------------------------

function [mf1] = processMF(mf0,wavtype,cleararray)

% level: number of decomposition levels
level=length(cleararray)-1;

% Calculate DWT
[C,L]=wavedec(mf0,level,wavtype);

% Clear some coefficients
index=1;
for i=1:level+1
  if cleararray(level+2-i)==0
    C(index:index+L(i)-1)=zeros(size(C(index:index+L(i)-1)));
  end
  index=index+L(i);
end

% Reconstruct signal
mf1=waverec(C,L,wavtype);

% Keep the inf-norm!
mf1=mf1*norm(mf0,inf)/norm(mf1,inf);


% -----------------------------------------------------------
% ROC.m - simulate Receiver Operating Characteristics
% Makes NMC noise only realizations and NMC signal+noise realizations
% Plots prob. of detection vs. prob. of false alarm
% By Emil Urnes, FFI
% Uses: ROC_0detect.m, ROC_1detect.m
% -----------------------------------------------------------

clear
```

```
% Set the LOADRESULTS to 0 if ROC curves are to be calculated,
% and 1 if results are on disk.
LOADRESULTS=1;

% Set TRANS_SYNTH to 1 if a synthetic transient is to be computed,
% 0 if an empirical one should be used.
TRANS_SYNTH=0;

if ~LOADRESULTS
% ------------- Input Parameters -----------------

% ROCTYPE=0 means using ROC_0detect
% ROCTYPE=1 means using ROC_1detect
% NMC is the number of monte carlo simulations
% an is the amplitude of noise
% Nx is number of signal samples
% The transient starts at sample t0

ROCTYPE=1;
if ROCTYPE==0
  an=0.4
else
  an=0.18
end
NMC=1000;
Nx=300;
t0=100;

if TRANS_SYNTH
  % Transient placed in data parameters
  % Nt is number of tranisent samples
  % damp, freq are transient parameters
  damp=3;
  freq=7.5;
  Ns=30;
  % Generate transient s
  for i=0:Ns-1,
    t=i/Ns;
    s(i+1)=trans(t,damp,freq);
  end
else % empirical transient
  load transient1
  s=transient;
  Ns=length(s);
end
% Transient should have norm 1.
s=s/norm(s,2);


% thr is an array of tresholds for which Pd and Pfa
```

```
% will be calculated.
if ROCTYPE==0
  Ndetectors=6;
  thr(1,:)=-1:0.01:2;
  thr(2,:)=2.5:0.01:5.5;
  thr(3,:)=6.5:0.04:18.5;
  thr(4,:)=-1:0.01:2;
  thr(5,:)=-1:0.01:2;
  thr(6,:)=-1:0.01:2;
else
  Ndetectors=5;
  thr(1,:)=0.0:0.005:1.5;
  thr(2,:)=0.0:0.005:1.5;
  thr(3,:)=0.0:0.005:1.5;
  thr(4,:)=0.0:0.005:1.5;
  thr(5,:)=0.0:0.005:1.5;
  thr(6,:)=0.0:0.005:1.5;
end


% ------------- Main Program ----------------------

n_thr=length(thr);

% Build Gaussian curve for Gabor detector.
gabordamp=0.01;
n=1:2*Nx;
% for gauss2, 0<t<1 is equivalent to the length of transient.
% The curve starts at Nx+1
g(n)=gauss((n-Nx-1)/Ns,gabordamp);

% First simulate NMC noise only realizations
Pfa=zeros([Ndetectors n_thr]);
for sim=1:NMC
  x=an*randn([1 Nx]);
  if ROCTYPE==0
    ROC_0detect
  else % ROCTYPE==1
    ROC_1detect
  end
  for ilev=1:Ndetectors,
    Pfa(ilev,:)=Pfa(ilev,:)+(lev(ilev)>thr(ilev,:));
  end
end
Pfa=Pfa/NMC;

% Now simulate NMC noise+transient realizations
Pd=zeros([Ndetectors n_thr]);
for sim=1:NMC
  x=an*randn([1 Nx]);
```

```
    x(t0:t0+Ns-1)=x(t0:t0+Ns-1)+s;
    if ROCTYPE==0
      ROC_0detect
    else
      ROC_1detect
    end
    for ilev=1:Ndetectors,
      Pd(ilev,:)=Pd(ilev,:)+(lev(ilev)>thr(ilev,:));
    end
  end
end
Pd=Pd/NMC;

% Save results for later use!
save results/roc Pfa Pd ROCTYPE

% Save signal, noise and signal+noise for plotting.
rocnoise=an*randn([1 Nx]);
rocsignal=zeros(1,Nx);
rocsignal(t0:t0+Ns-1)=s;
rocsum=rocsignal+rocnoise;
save rocdata rocnoise rocsignal rocsum

else %LOADRESULTS
 load results/rocx
end


% ********** Plot ROC curves *************

clf
hold on
plot(Pfa(1,:),Pd(1,:),'r');
plot(Pfa(2,:),Pd(2,:),'k');
plot(Pfa(3,:),Pd(3,:),'k--');
plot(Pfa(4,:),Pd(4,:),'b');
plot(Pfa(5,:),Pd(5,:),'g');
%plot(Pfa(6,:),Pd(6,:),'g');
%plot([0 1],[0 1],'k');
axis([0 1 0 1]);
ylabel('Probability of Detection');
xlabel('Probability of False Alarm');
title('ROC Curves');
text(0.6,0.35,'   red - db6');
text(0.6,0.30,'   black - db4');
text(0.6,0.25,'   dashed - db2');
text(0.6,0.20,'   blue - coif5');
text(0.6,0.15,'   green - sym7');
%text(0.6,0.10,'   green - xx');
grid
hold off
```

```
% ----------------------------------------------------------------
% ROC_0detect.m - Detectors for ROC.m
% The detector output is based on a single observation at t=t0.
% Output in lev(:)
% By Emil Urnes, FFI
% Uses: ROC_wavMF.m, ROC_maxwav.m, ROC_MFsmooth.m
% ----------------------------------------------------------------

% Time domain MF
Es=norm(s,2)^2;
lev(1)=sum(x(t0:t0+Ns-1).*s)/Es;
% Energy Detector
lev(2)=norm(x(t0:t0+Ns-1));
% Gabor Detector
gaboroff=19; % From detect.m
h=x.*g([Nx+2-t0-gaboroff:2*Nx+1-t0-gaboroff]);
H=fft(h,Nx);
lev(3)=max(abs(H([1:floor(Nx/2)])));

% Wavelet domain MF bands 234
type='db6';
detail=[0 1 1 1 0];
J=length(detail);
approx=0;
ROC_wavMF
lev(4)=y;

% Wavelet domain MF band 3
type='db6';
detail=[0 0 1 0 0];
J=length(detail);
approx=0;
ROC_wavMF
lev(5)=y;

% Time dependent Maxcof
type='db6';
detail=[0 0 1 0 0];
J=length(detail);
approx=0;
% compensate for wavelet offset
woff=56; % Obtained from detect.m
ROC_maxwav
lev(6)=y;

% Time MF + smoothing
%type='db2';
%cleararray=[0 0 0 1];
%ROC_MFsmooth
%lev(5)=y;
```

```
% -------------------------------------------------------------------
% ROC_wavMF.m - Wavelet domain matched filter for use with ROC.m
% By Emil Urnes, FFI
% Uses: detect_WTs.m
% -------------------------------------------------------------------

% First calculate WT of transient
detect_WTs

% Calculates only the one shift 'b' we need.
t=t0-1;
[a, b] = factor2j(t,J);

% Calculate DWT of the b left shift of x, Cbx/Lbx
eval(['[C',int2str(b),'x,L',int2str(b),'x]=
  wavedec(lshift(x,b),J,type);']);

% Automatic denoising by wden, see 8-170 in Wavelet Toolbox guide
%dummy=x;
%eval(['[dummy,C',int2str(b),'x,L',int2str(b),'x]=
%   wden(C',int2str(b),'x,L',int2str(b),'x,TPTR,
%      thrType,SCAL,J,type);']);
%end

% Put approx/detail coefficients in cixJ(k)/dixj(k) for all shifts i.
eval(['c',int2str(b),'x',int2str(J),'=appcoef(C',int2str(b),'x,L',
  int2str(b),'x,type,J);']);
for j=1:J,
  eval(['d',int2str(b),'x',int2str(j),'=detcoef(C',int2str(b),'x,L',
  int2str(b),'x,j);']);
end

% Do the wavelet domain matched filtering, output in y.
% tot contains the inner product.
% We start with highest J approximation.
if (approx==1)
  [a, b] = factor2j(t,J);
  eval(['ls=length(cs',int2str(J),');']);
  eval(['tot=sum(cs',int2str(J),'.*c',int2str(b),'x',
    int2str(J),'(a+1:a+ls));']);
else tot=0;
end
% now do the details from j=1:J
for j=1:J,
  if (detail(j)==1)
    [a, b] = factor2j(t,J);
    eval(['ls=length(ds',int2str(j),');']);
    eval(['tot=tot+sum(ds',int2str(j),'.*d',int2str(b),'x',
      int2str(j),'(2^(J-j)*a+1:2^(J-j)*a+ls));']);
  end
```

```
end
y=tot;


% -----------------------------------------------------------------
% ROC_maxwav.m - Time dependent Maxcof for use with ROC.m
% By Emil Urnes, FFI
% -----------------------------------------------------------------

% Calculates only the one shift 'b' we need.
t=t0-1+woff;
[a, b] = factor2j(t,J);

% Calculate DWT of the b left shift of x, Cbx/Lbx
eval(['[C',int2str(b),'x,L',int2str(b),'x]=
  wavedec(lshift(x,b),J,type);']);

% Automatic denoising by wden, see 8-170 in Wavelet Toolbox guide
%dummy=x;
%eval(['[dummy,C',int2str(b),'x,L',int2str(b),'x]=
% wden(C',int2str(b),'x,L',int2str(b),'x,TPTR,thrType,
% SCAL,J,type);']);
%end

% Put approx/detail coefficients in cixJ(k)/dixj(k) for all shifts i.
eval(['c',int2str(b),'x',int2str(J),'=appcoef(C',int2str(b),'x,L',
  int2str(b),'x,type,J);']);
for j=1:J,
  eval(['d',int2str(b),'x',int2str(j),'=detcoef(C',int2str(b),'x,L',
    int2str(b),'x,j);']);
end

% Find max coefficient for k=2^{J-j}a, all j, first approx:
if (approx==1)
  [a, b] = factor2j(t,J);
  eval(['maxcof=abs(c',int2str(b),'x',int2str(J),'(1+a))^2;']);
else maxcof=0;
end
% Now detail:
for j=1:J,
 if (detail(j)==1)
  [a, b] = factor2j(t,J);
  eval(['maxcof_candidate=abs(d',int2str(b),'x',int2str(j),
    '(1+2^(J-j)*a))^2;']);
  if maxcof_candidate>maxcof
    maxcof=maxcof_candidate;
  end
 end
end
y=maxcof;
```

```
% ------------------------------------------------------------------
% ROC_MFsmooth.m - Smoothed matched filter for use with ROC.m
% By Emil Urnes, FFI
% ------------------------------------------------------------------

% ------- Calculate time-domain MF
Es=norm(s)^2;
lengthY=length(x)-Ns+1;
for j=1:length(x),
  if (j>lengthY) y(j)=0;
    else y(j)=sum(x(j:j+Ns-1).*s)/Es;
  end
end

test=y;
out=processMF(y,type,cleararray);
y=out(t0);


% ------------------------------------------------------------------
% ROC_1detect.m - Detectors for ROC.m
% The detector output is based on the whole observation iterval
% by taking the maximum transform coefficient.
% Output in lev(k)
% By Emil Urnes, FFI
% Uses: ROC_1TI.m
% ------------------------------------------------------------------

[Cx,Lx]=wavedec(x,4,'db6');
lev(1)=max(abs(Cx));
[Cx,Lx]=wavedec(x,4,'db4');
lev(2)=max(abs(Cx));
[Cx,Lx]=wavedec(x,4,'db2');
lev(3)=max(abs(Cx));
[Cx,Lx]=wavedec(x,4,'coif5');
lev(4)=max(abs(Cx));
[Cx,Lx]=wavedec(x,4,'sym7');
lev(5)=max(abs(Cx));

%J=3;
%type='db2';
%ROC_1TI
%lev(3)=y;

%type='sym7';
%ROC_1TI
%lev(4)=y;


% ------------------------------------------------------------------
% ROC_1TI.m - TI detector for ROC
```

```
% Calculate DWT of 2^J left shifts of x, and find the global max
% Result in y
% By Emil Urnes, FFI
% ------------------------------------------------------------------

maximum=0;
for ind=0:2^J-1,
  [Cx,Lx]=wavedec(lshift(x,ind),J,type);
  newmax=max(abs(Cx));
  if newmax>maximum
    maximum=newmax;
  end
end
y=maximum;


% ------------------------------------------------------------------
% proto.m - Program for extracting amplitude information for a larger
% dataset. The purpose is to show the techniques work.
% Made by Emil Urnes, FFI
% Uses: proto_find.m
% ------------------------------------------------------------------

clear
% wang contains signal x, transient s and output yOut.
% yOut(1,:) is MF detector
% yOut(2,:) is db6 wavelet MF using only band 3.
load results/wang

% A detection is made if the threshold level is surpassed.
threshold=0.1;

% res is the number of samples in the resolution between transients.
res=20;

Ny=length(yOut(1,:));

% First detector
y=yOut(1,:);
proto_find
amp1=amp;
arrival1=arrival;

% Second detector
y=yOut(2,:);
proto_find
amp2=amp;
arrival2=arrival;

clf
maxX=4;
```

```
maxY=25;
% Plot results
subplot(2,1,1)
hist(amp1,40)
ylabel('count');
xlabel('amplitude');
axis([0 maxX 0 maxY]);
title('Matched filter');
number=mean(amp1);
text(3,20,['mean = ' num2str(number)])
number=std(amp1);
text(3,17,[' std = ' num2str(number)])
grid

subplot(2,1,2)
hist(amp2,40)
ylabel('count');
xlabel('amplitude');
axis([0 maxX 0 maxY]);
title('Matched filter using only subband 3');
number=mean(amp2);
text(3,20,['mean = ' num2str(number)])
number=std(amp2);
text(3,17,[' std = ' num2str(number)])
grid

% ----------------------------------------------------------------
% proto_find.m - Routine for extracting amplitude information for
% a particular detector output in y.
% The results are put in arrival and amp.
% Made by Emil Urnes, FFI
% ----------------------------------------------------------------

clear amp
clear arrival
i=1;
k=1;
while i<=(Ny-res),
  if y(i)<threshold
    i=i+1;
  else % Detection!
    % Check for better tops further back.
    backindex=i-res;
    if backindex<1
      backindex=1;
    end
    if y(i)>=max(y(backindex:i)) % no fake!
      top=y(i);
      % Check for better tops further ahead.
      while (i<=(Ny-res)) & (max(y(i:i+res))>top),
```

```
            [top, index]=max(y(i:i+res));
             i=i+index(1)-1;
          end
          % store the arrival time and amplitude
          arrival(k)=i;
          amp(k)=y(i);
          k=k+1;
       end
       i=i+1;
    end
end
```