T.C.
BAHÇEŞEHİR ÜNİVERSİTESİ

ANALYSIS OF QUADRATURE DOPPLER SIGNALS WITH A MODIFIED DUAL-TREE COMPLEX WAVELET TRANSFORM

Master's Thesis

Görkem SERBES

ISTANBUL, 2009

T.C.
BAHÇEŞEHİR ÜNİVERSİTESİ
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
ELECTRICAL & ELECTRONICS ENGINEERING GRADUATE PROGRAM

ANALYSIS OF QUADRATURE DOPPLER SIGNALS WITH A MODIFIED DUAL-TREE COMPLEX WAVELET TRANSFORM

Master's Thesis

Görkem SERBES

Supervisor: Prof. Dr. Nizamettin AYDIN

ISTANBUL, 2009

T.C.

BAHÇEŞEHİR ÜNİVERSİTESİ

THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

ELECTRICAL & ELECTRONICS ENGINEERING GRADUATE PROGRAM

Name of the thesis: Analysis of Quadrature Doppler Signals with a Modified Dual-Tree Complex Wavelet Transform

Name/Last Name of the Student: Görkem SERBES
Date of Thesis Defense: 24 August 2009

The thesis has been approved by the Institute of Science.

Prof. Dr. Bülent ÖZGÜLER
Director
____

I certify that this thesis meets all the requirements as a thesis for the degree of Master of Science.

Prof. Dr. Bülent ÖZGÜLER
Program Coordinator
____

This is to certify that we have read this thesis and that we find it fully adequate in scope, quality and content, as a thesis for the degree of Master of Science.

Examining Committee Members	Signature
Prof. Dr. Nizamettin AYDIN	____________________
Asst. Prof. Dr. Levent EREN	____________________
Asst. Prof. Dr. Fatih UĞURDAĞ	____________________

ACKNOWLEDGMENTS

This thesis is dedicated to **my mother, my brother** and to **my love, Hasret**.

I would like to express my gratitude to my supervisor **Prof. Dr. Nizamettin AYDIN** for his support, understanding and unlimited tolerance throughout my thesis studies.

I also thank **Asst. Prof. Dr. Levent EREN** for his helps on various topics throughout my academic program and also my life.

Finally, I also would like to express my special thanks to **TUBITAK** for its financial supports during my master program.

ABSTRACT

ANALYSIS OF QUADRATURE DOPPLER SIGNALS WITH A MODIFIED DUAL-TREE COMPLEX WAVELET TRANSFORM

SERBES, Görkem

Electrical & Electronics Engineering

Supervisor: Prof. Dr. Nizamettin AYDIN

August 2009, 83 Pages

Doppler ultrasound systems employ quadrature demodulation techniques at the detection stage. Complex quadrature Doppler signals, which have the information of flow direction, are obtained after demodulation. Flow direction is encoded in the phase relationship between in-phase and quadrature phase channels. A number of methods exist for extracting directional information from the quadrature Doppler signals. The phasing-filtering technique, which is based on Hilbert transform, is most widely used method. After the extraction of directional signals, different signal processing methods can be applied to these directional signals. Discrete wavelet transform, which is becoming a popular tool for analysis of non-stationary biological signals, is one of these signal processing methods. But discrete wavelet transform has some drawbacks. As a solution to these drawbacks, a complex discrete wavelet transform algorithm called dual tree complex wavelet transform was proposed. However, it does not provide directional signal decoding during analysis. In this thesis, a Modified Dual-Tree Complex Wavelet Transform capable of mapping directional signals at the transform output is presented.

Keywords: Complex Wavelets**,** Phase Filtering Technique**,** Discrete Wavelet Transform, Hilbert Transform.

ÖZET

QUADRATURE DOPPLER İŞARETLERİN MODİFİYE EDİLMİŞ ÇİFT-AĞAÇ KOMPLEKS DALGACIK DÖNÜŞÜMÜ İLE İŞLENMESİ

SERBES, Görkem

Elektrik & Elektronik Mühendisliği

Tez Danışmanı: Prof. Dr. Nizamettin AYDIN

Ağustos 2009, 83 Sayfa

Doppler ultrason sistemleri algılama safhasında quadrature demodulasyon teknikleri kullanmaktadırlar. Akış yönü bilgisini içeren kompleks quadrature Doppler işaretleri demodulasyon sonrası elde edilmektedir. Akış yönü bilgisi de bu in-phase ve quadrature-phase bileşenleri arasındaki faz farkında kodlanmıştır. Quadrature Doppler işaretlerinden yön bilgilerinin çıkarılması için kullanılan bir çok yöntem bulunmaktadır. Hilbert dönüşümünü kullanan faz filtreleme yöntemi, bu yöntemlerden en çok kullanılanıdır. Yön bilgisini içeren işaretler elde edildikten sonra, bu işaretlere çeşitli işaret işleme yöntemleri uygulanabilir. Son zamanlarda durağan olmayan biyolojik işaretlerin analizinde kullanılmakta olan ayrık dalgacık dönüşümü, yukarıda belirtilen yöntemlerden biridir. Fakat ayrık dalgacık dönüşümünün birkaç eksikliği bulunmaktadır. Bu eksiklikleri gidermek amacıyla, bir kompleks dalgacık yöntemi olan, Çift-Ağaç Kompleks Dalgacık Dönüşümü tasarlanmıştır. Fakat bu yöntem analiz sürecinde yön işaretlerini vermemektedir. Bu tezde, dönüşüm sonucunda yön bilgisini de veren değiştirilmiş bir Çift-Ağaç Kompleks Dalgacık dönüşümü önerilmiştir.

Anahtar Kelimeler: Kompleks Dalgacıklar, Faz Filtreleme Tekniği, Ayrık Dalgacık Dönüşümü, Hilbert Dönüşümü.

TABLE OF CONTENTS

LIST OF TABLES .... **IX**

LIST OF FIGURES .... **X**

LIST OF ABBREVIATIONS .... **XIII**

LIST OF SYMBOLS .... **XIV**

1. INTRODUCTION **1**

1.1 ORGANIZATION OF THESIS 2

2. THE DIGITAL SIGNAL PROCESSING BASICS **3**

2.1 INTRODUCTION 3
2.2 ANALYSIS AND SYNTHESIS OF SIGNALS 4
2.2.1 Orthogonal Vectors in the Plane 4
2.3 TIME-FREQUENCY AND TIME-SCALE ANALYSIS 6
2.3.1 Fourier Transform 6
2.3.2 The Short Time Fourier Transform 7
2.3.2.1 The spectrogram 9
2.3.3 The Continuous Wavelet Transform 14
2.3.3.1 Comparison with STFT 17
2.3.3.2 The scalogram 20
2.3.4 Comparative Visualization 22
2.3.5 Analysis and Synthesis with Wavelets 23
2.3.5.1 The Haar wavelet 24
2.4 MULTIRESOLUTION ANALYSIS 26
2.4.1 The Two Scale Property of Multiresolution 29
2.4.2 The Scaling Function 29
2.4.2.1 The two scale equation and the filters 31
2.4.3 The Discrete Wavelet Transform 32
2.5 FILTER BANKS AND THE DWT 35
2.5.1 Analysis: From Fine Scale to Coarser Scale 35
2.5.1.1 Filtering and downsampling 38
2.5.1.2 The one-stage analysis filter bank 39
2.5.1.3 The analysis filter bank 40
2.5.2 Synthesis: From Course Scale to Fine Scale 41
2.5.2.1 Upsampling and filtering 42

2.5.2.2 The one-stage synthesis filter bank 43
2.5.2.3 Perfect reconstruction filter bank..44
2.5.2.4 The synthesis filter bank 44
2.5.2.5 Approximations and details 44

2.6 HILBERT TRANSFORM 46

3. THE BASICS OF QUADRATURE DOPPLER SIGNALS 48

3.1 INTRODUCTION.. 48

3.2 PHYSICAL PRINCIPLE OF DOPPLER ULTRASOUND 49

3.2.1 Detection of Doppler Ultrasound Signals 49

3.2.1.1 Demodulation of Doppler frequency shifted signals 50
3.2.1.2 Quadrature phase detection..51

3.3 GENERAL DEFINITION OF COMPLEX QUADRATURE DOPPLER SIGNALS 52

3.4 PHASE FILTERING TECHNIQUE 53

4. PROPOSED METHOD: MODIFIED DUAL-TREE COMPLEX WAVELET TRANSFORM 55

4.1 INTRODUCTION.. 55

4.2 DUAL TREE COMPLEX WAVELET TRANSFORM.. 55

4.2.1 Structure of DTCWT.. 58

4.2.2 Processing of Quadrature Doppler Signals with DTCWT 61

4.3 PROPOSED METHOD: MODIFIED DUAL TREE COMPLEX WAVELET TRANSFORM 61

4.3.1 A Simulation Example using MDTCWT 63

4.3.2 MDTCWT Coefficients.. 64

4.3.3 Success of the Proposed Method.. 65

4.3.4 Complexity of the Proposed Method.. 66

4.3.5 Performance of the Proposed Method: De-Noising with MDTCWT 66

4.3.5.1 Signal and noise model 67
4.3.5.2 Soft thresholding 68
4.3.5.3 Structure of the de-noising algorithms for MDTCWT, DWT and DTCWT..68

5. RESULTS, CONCLUSIONS AND FUTURE SCOPE..70

5.1 RESULTS 70

5.1.1 Success of the Proposed Method.. 70

5.1.2 Complexity of the Proposed Method.. 71

5.1.3 Performance of the Proposed Method 72

5.2 CONCLUSION AND FUTURE SCOPE.. 78

REFERENCES..79

VITAE 83

LIST OF TABLES

Table 5.1 : The PRD values for the forward and reverse flow signals between the PFT and MDTCWT. 70
Table 5.2 : Comparison of the processing times for the PFT with DWT, the PFT with DTCWT and the MDTCWT 72

LIST OF FIGURES

Figure 2.1 : An example vector x, which is built by the orthogonal vectors Ψ_1 and Ψ_2...5
Figure 2.2 : An example of windowing with box function. 7
Figure 2.3 : STFT of a sum of two sinusoids with frequencies f_1 = 1100 Hz and f_2 = 1500 Hz and the window size T = 1/250. 8
Figure 2.4 : STFT of a sum of two sinusoids with frequencies f_1 = 1100 Hz and f_2 = 1500 Hz and the window size T = 1/50. 9
Figure 2.5 : Time domain and frequency domain representation of a signal, x(t),which is the sum of two sinusoids of frequencies f_1 = 500Hz and f_2 = 1500Hz and two impulses at times t_1 = 125 ms and t_2 = 130 ms. 10
Figure 2.6 : Spectrogram of the same signal, x(t) with two dimensional plot. 10
Figure 2.7 : Spectrogram of the same signal, x(t) with three dimensional plot. 11
Figure 2.8 : Spectrogram of a signal, x(t), with two dimensional plot which is the sum of two sinusoids of frequencies f1 = 500Hz and f2 = 1000Hz and two impulses at times t1 = 125 ms and t2 = 130 ms. The window width is T = 2.5 ms. 12
Figure 2.9 : Spectrogram of the same signal, x(t) with three dimensional plot. 12
Figure 2.10 : Spectrogram of a signal, x(t), with two dimensional plot which is the sum of two sinusoids of frequencies f1 = 500Hz and f2 = 1000Hz and two impulses at times t1 = 125 ms and t2 = 130 ms but the window width is T = 8 ms. 13
Figure 2.11 : Spectrogram of the same signal, x(t) with three dimensional plot but the window width is T = 8 ms. 13
Figure 2.12 : The Haar wavelet with two different scales and shifts. 15
Figure 2.13 : The Morlet wavelet with three different scales and shifts. 16
Figure 2.14 : CWT implemented with convolution. 16
Figure 2.15 : STFT implemented with convolution. 17
Figure 2.16 : Time domain and frequency domain representation of Gaussian window. 18
Figure 2.17 : Time domain and frequency domain representation of Morlet wavelet.... 19

Figure 2.18 : Scalogram of a signal, x(t), with two dimensional plot which is the sum of two sinusoids of frequencies f1 = 500Hz and f2 = 1000Hz and two impulses at times t1 = 125 ms and t2 = 130 ms. 20
Figure 2.19 : Scalogram of the same signal but frequency is used instead of scale. 21
Figure 2.20 : Scalogram of the same signal with three dimensional plot. 21
Figure 2.21 : Comparative visualization of various time-frequency representations. 22
Figure 2.22 : Time-scale diagram for the discrete wavelet transform. 24
Figure 2.23 : Orthogonality for Haar wavelet. 25
Figure 2.24 : Nested subspaces in multiresolution analysis. 27
Figure 2.25 : Five level decomposition of an example signal. 29
Figure 2.26 : The two scale equation for the Haar scale. 32
Figure 2.27 : The two scale equation for the Haar wavelet. 32
Figure 2.28 : Analysis and synthesis view of DWT. 36
Figure 2.29 : Downsampler. 38
Figure 2.30 : Filter and downsampler for approximation coefficients. 39
Figure 2.31 : Filter and downsample for detail coefficients. 39
Figure 2.32 : One stage filter bank. 40
Figure 2.33 : Three levels of DWT analysis. 41
Figure 2.34 : Upsampler. 43
Figure 2.35 : Upsample and filter. 43
Figure 2.36 : One stage synthesis filter bank. 43
Figure 2.37 : Perfect reconstruction filter bank. 44
Figure 2.38 : Three stages synthesis filter bank. 44
Figure 2.39 : Frequency response of Hilbert transform. 47
Figure 3.1 : A general Doppler ultrasound signal measurement system. 50
Figure 3.2 : Quadrature phase detection of the Doppler shift signals. 51
Figure 3.3 : Embolic quadrature Doppler signal pair. 52
Figure 3.4 : Block diagram of phase filtering technique. 53
Figure 3.5: Directional signals with PFT. (Red – forward signal and blue – reverse signal) 54
Figure 4.1 : Examples of embolic signals. (Forward signal – red, reverse signal - blue) 57

Figure 4.2 : Shift invariance of DTCWT. 57
Figure 4.3 : Analysis filterbanks for the DTCWT. 58
Figure 4.4 : Synthesis filterbanks for the DTCWT. 59
Figure 4.5 : Level 3 wavelet and scaling functions in time domain. 60
Figure 4.6 : Frequency spectrum of a 4 level DTCWT 60
Figure 4.7 : Processing quadrature Doppler signals with DTCWT. 61
Figure 4.8 : Analysis stage of the MDTCWT algorithm for three levels. 62
Figure 4.9 : Synthesis stage of the MDTCWT algorithm for three levels. 63
Figure 4.10 : Directional signals which are obtained with PFT. 64
Figure 4.11 : Directional signals which are obtained with MDTCWT. 64
Figure 4.12 : Five levels reconstructed detail and approximation coefficients with MDTCWT. 65
Figure 4.13 : De-noising with MDTCW. 68
Figure 4.14 : De-noising with DWT. 68
Figure 4.15 : De-noising with DTCWT. 69
Figure 5.1 : (a) A quadrature embolic Doppler signal, (b) the forward (red line) and the reverse (blue line) outputs using the MDTCWT, (c) the forward (blue line) and the reverse (red line) outputs using the PFT, and corresponding differences of (d) the forward an and (e) the reverse signals obtained by the MDTCWT and the PFT 71
Figure 5.2 : Noised directional signals and normal directional signals. 72
Figure 5.3 : Reconstructed subbands of noised signal with MDTCWT. 73
Figure 5.4 : Reconstructed subbands of noised signal with DWT. 74
Figure 5.5 : Reconstructed subbands of noised signal with DTCWT. 74
Figure 5.6 : Reconstructed subbands of de-noised signal with MDTCWT. 75
Figure 5.7 : Reconstructed subbands of de-noised signal with DWT. 75
Figure 5.8 : Reconstructed subbands of de-noised signal with DTCWT. 76
Figure 5.9 : De-Noised directional signals with three methods. 76
Figure 5.10 : RMS error for MDTCWT, DWT and DTCWT. 77

LIST OF ABBREVIATIONS

Continuous Wavelet Transform : CWT

Discrete Wavelet Transform : DWT

Dual-Tree Complex Wavelet Transform : DTCWT

Fast Fourier Transform : FFT

Fourier Transform : FT

Hilbert Transform : HT

Inverse Discrete Wavelet Transform : IDWT

Modified Dual-Tree Complex Wavelet Transform : MDTCWT

Percent Root Mean Square Difference : PRD

Phase Filtering Technique : PFT

Root Mean Square : RMS

Short-Time Fourier Transform : STFT

LIST OF SYMBOLS

Continuous Wavelet Transform of $x(t)$: $C(a,b)$

Discrete Wavelet Transform of $x(t)$: $C\left(1/2^j, k/2^j\right)$

Fourier Transform of $x(t)$: $X(f)$

Hilbert Transform : $H[\quad]$

Scaling Function : $\phi(\mathrm{t})$

Scaling Space : V_j

Short Time Fourier Transform of $x(t)$: $STFT(f,s)$

Wavelet Function : w(t)

Wavelet Space : W_j

1. INTRODUCTION

Many measurement systems such as magnetic resonance and Doppler ultrasound systems employ quadrature demodulation techniques at the detection stage. After demodulation, complex quadrature Doppler signals, which have the information of flow direction, are obtained. Flow direction is encoded in the phase relationship between in-phase and quadrature phase channels (Evans et al. 1989). A number of methods exist for extracting directional information from the quadrature Doppler signals (Aydin, Fan & Evans 1994), (Aydin, Padayachee & Markus 1999). The phasing-filtering technique, which is based on Hilbert transform, is most widely used method. Fast Fourier transform mapping the directional information in the frequency domain is widely used for the analysis of Doppler signals (Aydin, Fan & Evans 1994). Similarly, a complex continuous wavelet transform algorithm mapping the directional information in the scale domain was introduced in (Aydin & Markus 2000).

In the case of the discrete wavelet transform, which is becoming a popular tool for analysis of non-stationary biological signals, an algorithm mapping directional signals in the scale domain during analysis does not exist. Moreover, an important drawback of the discrete wavelet transform is that the distribution of energy between coefficients at different scales is very sensitive to shifts in the input data (Kingsbury 1999). In the analysis of non-stationary Doppler signals (particularly embolic Doppler ultrasound signals which are similar to transients), any distortion in the phase of the signal cannot be tolerated as the direction of the flow information is encoded in the phase relationship of the in-phase and quadrature-phase components of the quadrature signal.

As a solution to this problem, a complex discrete wavelet algorithm called dual tree complex wavelet transform was proposed in (Kingsbury 2001). However, it does not provide directional signal decoding during analysis. In this thesis, a modified dual tree complex wavelet transform capable of mapping directional signals at the transform output is presented.

1.1 ORGANIZATION OF THESIS

The thesis is organized into five chapters as follows:

Chapter one is an introduction with the comprehensive description of the central theme of this research. A systematic organization of thesis is also presented.

In chapter two, the basics of time-frequency and time-scale methods are given with practical implementations and in addition to wavelet theory basics, Hilbert transform which is a widely used frequency domain transform, is also explained

In chapter three, the basics of Doppler principle and Doppler ultrasound systems are explained. Furthermore, complex quadrature Doppler signals, which are obtained from Doppler ultrasound systems, are explained. Finally, the phasing filtering technique, which is used for the extraction of directional blood flow signals, is discussed.

In chapter four, firstly, the Dual Tree Complex Transform which is a modified discrete wavelet transform with good shift-invariance property is explained. And later the proposed method, Modified Dual Tree Complex Wavelet Transform is introduced. Moreover, the proposed method’s performance, success and computational complexity are compared with other scale domain methods.

In chapter five, the results of the comparisons, which are described in chapter four, are presented. Advantages of the proposed method are explained and future directions for further investigations using proposed method are given.

2. THE DIGITAL SIGNAL PROCESSING BASICS

2.1 INTRODUCTION

In many applications such as acoustic signal processing, observing evolution of the frequency content of a signal over time is important. Signals in real life are non-stationary signals, in which the statistical properties of the signal change over the time. In many applications such as medical diagnosing, frequency (or scale) information is used for the purpose of diagnosing different problems. For example embolic Doppler ultrasound signals are analyzed to diagnose stroke (Aydin, Marvasti & Markus 2004). Fourier transform is widely used to analyze such signals. However for a non-stationary signal, $x(t)$, the standard Fourier transform is not sufficient for analyzing the signal. Information which is localized in time such as spikes and high frequency bursts cannot be easily detected by the Fourier transform. Time-localization can be achieved by first windowing the signal so as to cut off only a well-localized slice of $x(t)$ and then taking its Fourier Transform. This gives rise to the short time Fourier transform, or windowed Fourier transform. But in short time Fourier transform, time resolution and frequency resolution is fixed over the entire time-frequency plane. To overcome this disadvantage, continuous wavelet transform, which provides a time-scale description similar to the short time Fourier transform, was introduced (Rioul & Vetterli 1991). Although, the continuous wavelet transform resolves both time and scale (frequency) events better than the short time Fourier transform, the computational cost for the implementation is very high. Therefore, to reduce the computational cost, a fast implementation of continuous wavelet transform called the discrete wavelet transform was introduced. The practical usefulness of discrete wavelet transform comes from its multiresolution analysis ability. In this chapter, the basics of time-frequency and time-scale methods will be given with practical implementations. In addition to wavelet theory basics, Hilbert transform, which is a widely used frequency domain transform, will be explained.

2.2 ANALYSIS AND SYNTHESIS OF SIGNALS

Signals which are defined on the time interval $a < t < b$ can be added, subtracted and multiplied by constants. If these signals are sampled at times $t = a + nT$ for $0 \leq n \leq (b-a)/T$, then a signal, $x(t)$, turns into a vector $X(n) = x(a+nT)$.

Given two signals, $x(t)$ and $y(t)$, the dot product of the corresponding vectors, X and Y is:

$$\sum_n X(n)\,Y(n) = \sum_n x(a+nT)y(a+nT)$$

$$= \frac{1}{T}\sum_n x(a+nT)y(a+nT)T \approx \frac{1}{T}\int_a^b x(t)y(t)\,dt \qquad (2.1)$$

In this sense the integral of $x(t)$ times $y(t)$ can be thought as a sort of dot product of the two signals. The inner product of two signals are defined as:

$$\text{inner product} = \langle x(t), y(t)\rangle = \int_a^b x(t)y(t)dt \qquad (2.2)$$

Two signals $x(t)$ and $y(t)$, which are orthogonal, can be defined as:

$$\langle x(t), y(t)\rangle = 0 \qquad (2.3)$$

2.2.1 Orthogonal Vectors in the Plane

A vector, x, can be defined with a simple formula in the plane in terms of a pair of orthogonal vectors ψ_1 and ψ_2.

The vector x can be projected onto each of the vectors ψ_1 and ψ_2 obtaining multiplications $c_1\psi_1$ and $c_2\psi_2$, and then can be synthesized as $x = c_1\psi_1 + c_2\psi_2$.

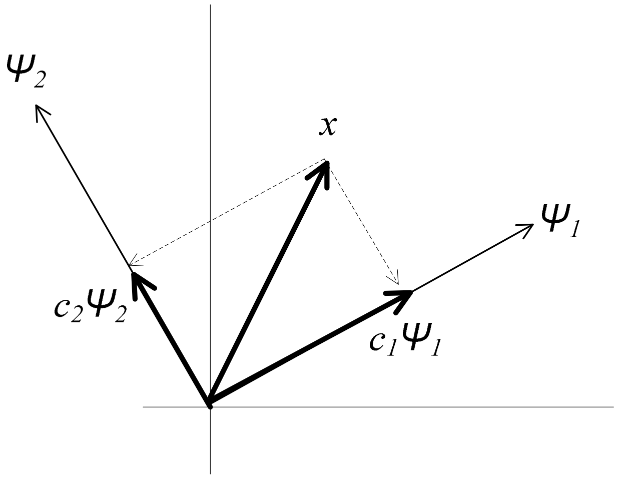


Figure 2.1 : An example vector x, which is built by the orthogonal vectors Ψ_1 and Ψ_2.

This geometric construction can be obtained through dot products. By taking the dot product of x first with ψ_1, the coefficient c_1 can be derived as:

$$\begin{aligned}\langle x, \psi_1\rangle &= \langle c_1\psi_1 + c_2\psi_2, \psi_1\rangle \\ &= c_1\langle \psi_1, \psi_1\rangle + c_2\langle \psi_2, \psi_1\rangle \\ &= c_1\langle \psi_1, \psi_1\rangle \qquad (2.4)\end{aligned}$$

So, c_1 can be solved as:

$$c_1 = \frac{\langle x, \psi_1\rangle}{\langle \psi_1, \psi_1\rangle} \qquad (2.5)$$

Similarily, by taking the inner product of x with ψ_2, $c_2$can be solved as:

$$c_2 = \frac{\langle x, \psi_2\rangle}{\langle \psi_2, \psi_2\rangle} \qquad (2.6)$$

Recall that it can be said that the vectors ψ_1 and ψ_2 are an orthogonal basis for the set of vectors in the plane.

When the coefficients c_1 and c_2 are computed, it can be said that the vector x has been analyzed in terms of the basis ψ_1 and ψ_2. When the vector is expressed as $x = c_1\psi_1 + c_2\psi_2$ it can be said that the vector has been synthesised from the basis.

That is, two processes are going on:

$$Analysis: c_n = \frac{\langle x(t), \psi_n(t) \rangle}{\langle \psi_n(t), \psi_n(t) \rangle} \tag{2.7}$$

$$Synthesis: x(t) = \sum_{n=1}^{2} c_n \psi_n(t) \tag{2.8}$$

Generally, synthesis formula can be written as:

$$Synthesis: x(t) = \sum_{n=-\infty}^{\infty} c_n \psi_n(t) \tag{2.9}$$

The Analysis coefficients c_n are called the Generalized Fourier Coefficients and the Synthesis equation is called the Generalized Fourier Series (Phillips 2009).

2.3 TIME-FREQUENCY AND TIME-SCALE ANALYSIS

2.3.1 Fourier Transform

Fourier transform (FT) is a well-known mathematical tool to transform time-domain signal into frequency-domain for efficient extraction of information (Proakis & Manolakis 2007). For a signal x(t), the FT is given by:

$$X(f) = \int_{-\infty}^{\infty} x(t)\, e^{-j2\pi f t} dt \tag{2.10}$$

The FT has a great ability to capture signal's frequency content as long as $x(t)$ is composed of few stationary components (e.g. sine waves). However, any abrupt change in time for non-stationary signal $x(t)$ is spread out over the whole frequency axis in $X(f)$. The limitation of FT is that it cannot offer both time and frequency localization of a signal at the same time.

2.3.2 The Short Time Fourier Transform

To overcome the limitations of the standard FT, the concept of Short Time Fourier Transform (STFT) is introduced (Cohen 1989). The advantage of STFT is that it uses an arbitrary but fixed-legth window $g(t)$ for analysis, over which the actual nonstationary signal is assumed to be approximately stationary. STFT of a signal $x(t)$ using a window function $g(t)$ can be defined as below:

$$STFT(f,s) = \int_{-\infty}^{\infty} x(t)g(t-s)e^{-j2\pi ft}dt \tag{2.11}$$

The window $g(t)$ can be thought as a sliding along the signal $x(t)$ and for each shift $g(t-s)$, the usual Fourier transform of the product function x$(t)g(t-s)$ is computed . For example, if $g(t)$ is the box of width 1/2 then:

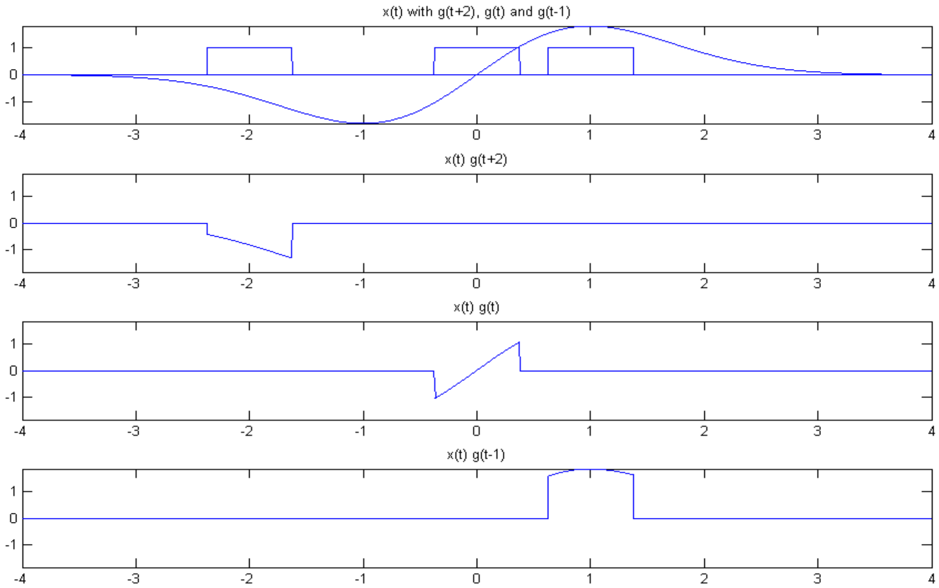


Figure 2.2 : An example of windowing with box function.

In the frequency domain the convolution theorem can be used to recognize $STFT(f,s)$ as the convolution of $X(f)$ with the FT of $g(t-s)$ (which is $e^{-j2\pi fs}G(f)$).

Recall the FT pair for the box function:

$$box(t) = \begin{cases} 1 & for|t| \leq 1/2 \\ 0 & elsewhere \end{cases} \qquad F\{box(t) = sinc(f)\} \tag{2.12}$$

If $g(t)$ is a box of width T , that is, $g(t) = box(t/T)$ then $G(f) = Tsinc(fT)$.

In the case where the signal consists of two sinusoids of frequencies f_1 and f_2 the windowed transform will be the superposition of two shifted sinc functions. The individual frequencies cannot be resolved unless $|f_1 - f_2| > 1/T$. In fact, for adequate separation it should be $|f_1 - f_2| > 2/T$. That is, the ``frequency resolution" of this analysis is $1/T$.

In the following example a signal consisting two sinusoids with frequencies $f_1 = 1100$ Hz and $f_2 = 1500$ Hz is considered. The window size is $T = 1/250$. Two distinct peaks in the frequency response can be seen in figure 2.3:

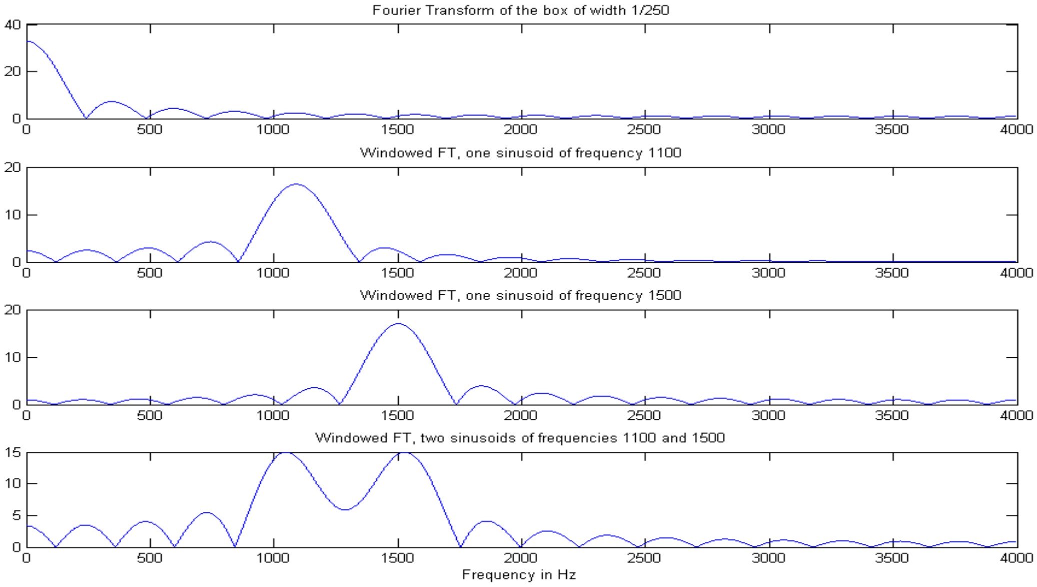


Figure 2.3 : STFT of a sum of two sinusoids with frequencies f_1 = 1100 Hz and f_2 = 1500 Hz and the window size T = 1/250.

In the case where the signal consists of two spikes close together in time, the spikes can be resolved if the window size T is smaller than the time difference between the spikes.

This analysis shows the ``trade-off" between time resolution and frequency resolution: if a window of length $T$ is used then the ``time-resolution" is T, but the frequency resolution is $1/T$.

In the figure 2.4, T is changed to 50 in this case the time-resolution is reduced, and the frequency resolution is increased. As a result the two sinusoids can be discriminated better.

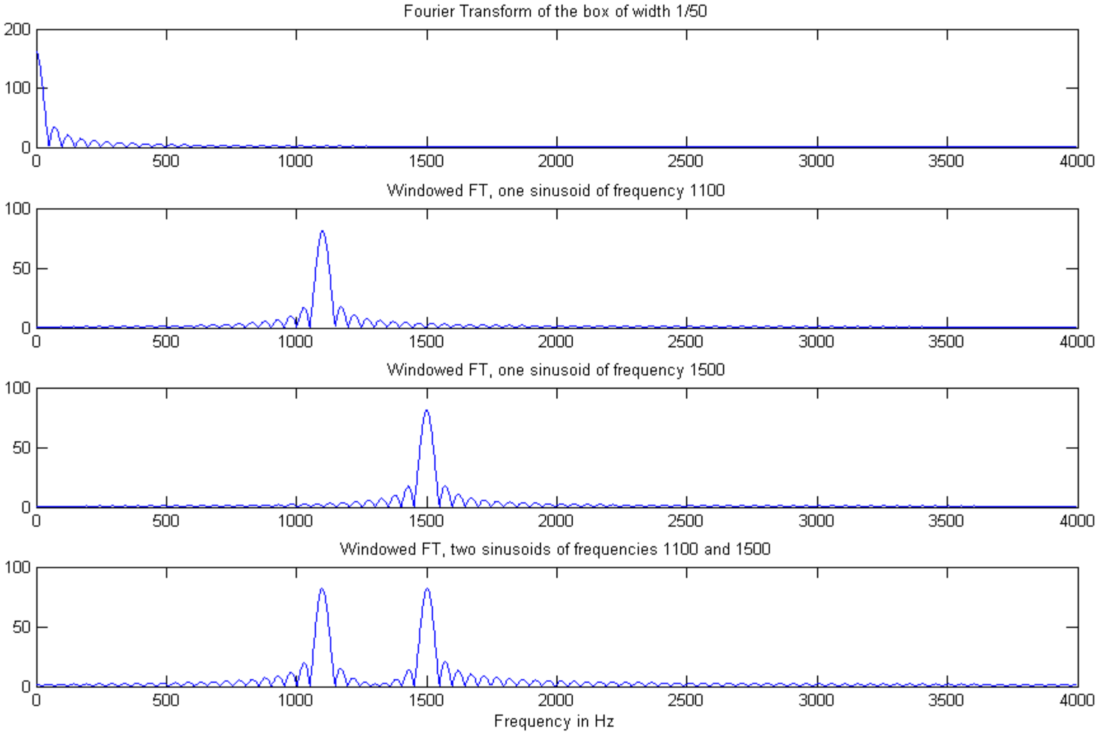


Figure 2.4 : STFT of a sum of two sinusoids with frequencies f_1 = 1100 Hz and f_2 = 1500 Hz and the window size T = 1/50.

2.3.2.1 The spectrogram

The magnitude of the STFT is called the spectrogram. There are two possible ways to show spectrogram; in the first one it can be formed by a 2 dimensional plot with time on the horizontal axis, frequency on the vertical axis and amplitude given by a gray-scale colour. Alternately it can be formed by a 3 dimensional plot where the amplitude is on the third axis. In the following example, a signal $x(t)$ is the sum of two sinusoids of frequencies $f_1 = 500Hz$ and $f_2 = 1500Hz$ and two impulses at times $t_1 = 125$ ms and $t_2 = 130$ms is used, with a window width of $T = 2.5$ ms ($1/T = 400$ Hz).

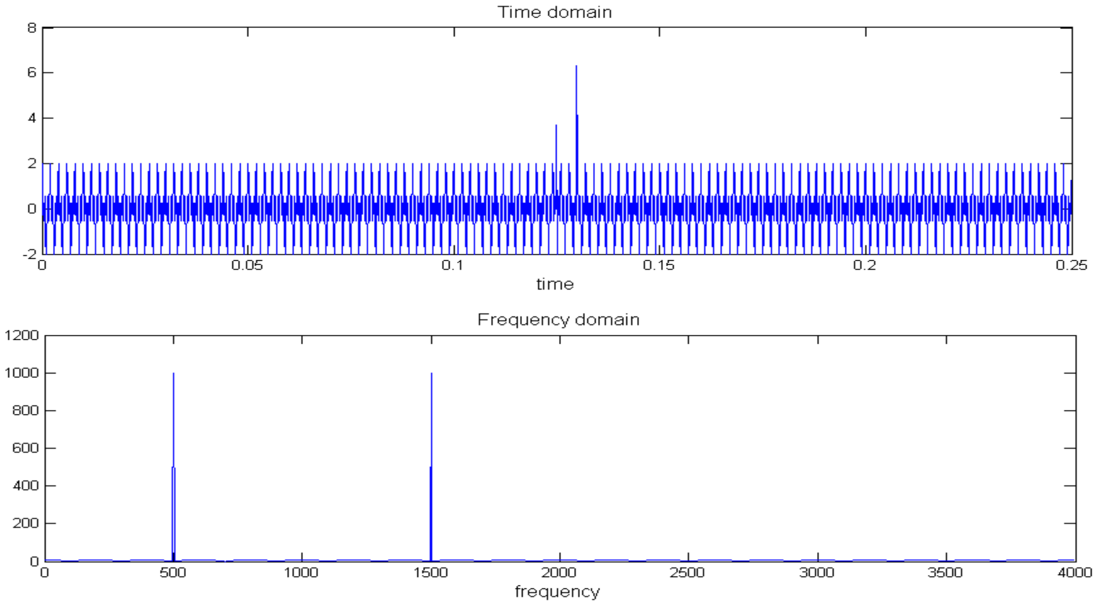


Figure 2.5 : Time domain and frequency domain representation of a signal, x(t),which is the sum of two sinusoids of frequencies f_1 = 500Hz and f_2 = 1500Hz and two impulses at times t_1 = 125 ms and t_2 = 130 ms.

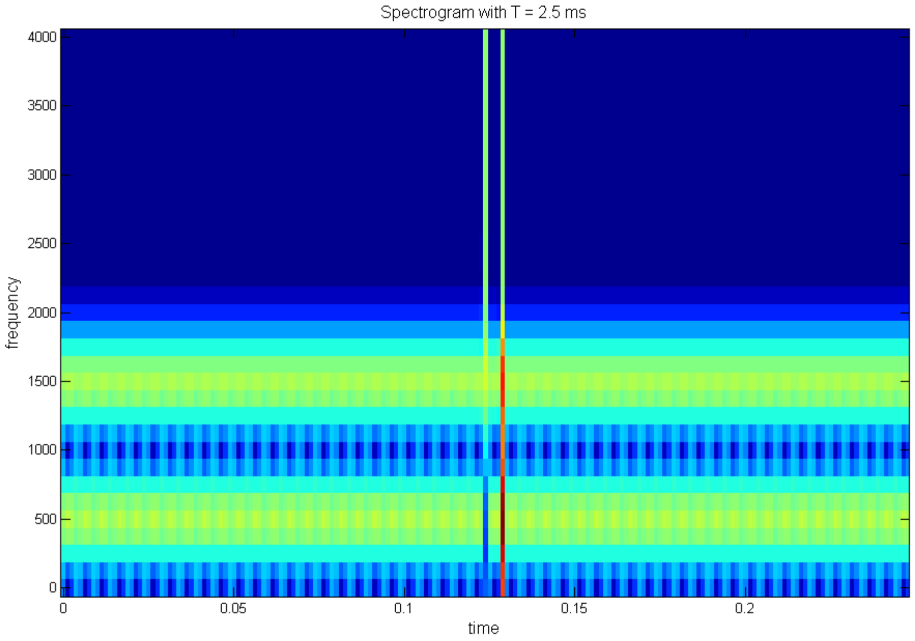


Figure 2.6 : Spectrogram of the same signal, x(t) with two dimensional plot.

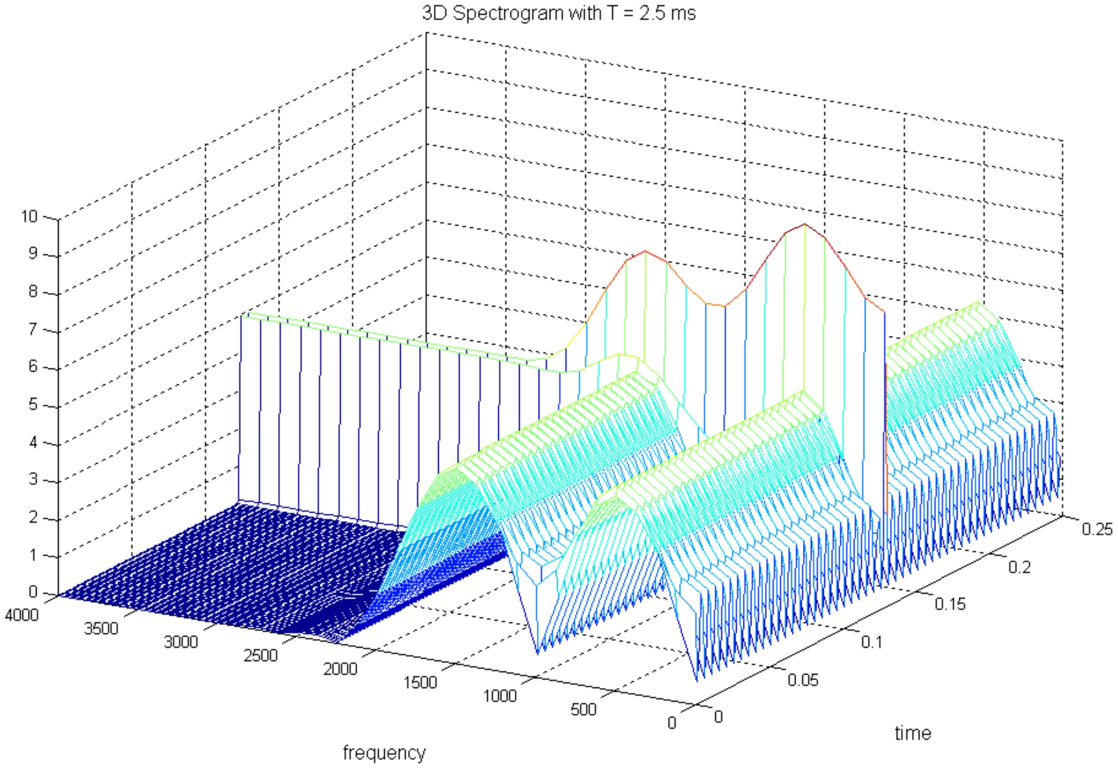


Figure 2.7 : Spectrogram of the same signal, x(t) with three dimensional plot.

The resolution in frequency is $1/T = 400$ Hz. The time resolution is $T = 2.5$ ms. As the figures 2.6 and 2.7 show, the sinusoids and the impulses can be resolved together.

Now suppose that the two frequencies are moved closer together. Let's use a signal $x(t)$ which is the sum of two sinusoids of frequencies $f_1 = 500Hz$ and $f_2 = 1000Hz$ and two impulses at times $t_1 = 125$ ms and $t_2 = 130$ ms with a window width of $T = 2.5$ ms.

As the spectrograms, in figure 2.8 and figure 2.9, now show us the frequencies cannot be resolved but still the spikes can be resolved. The frequency resolution is not good enough to distinguish frequencies.

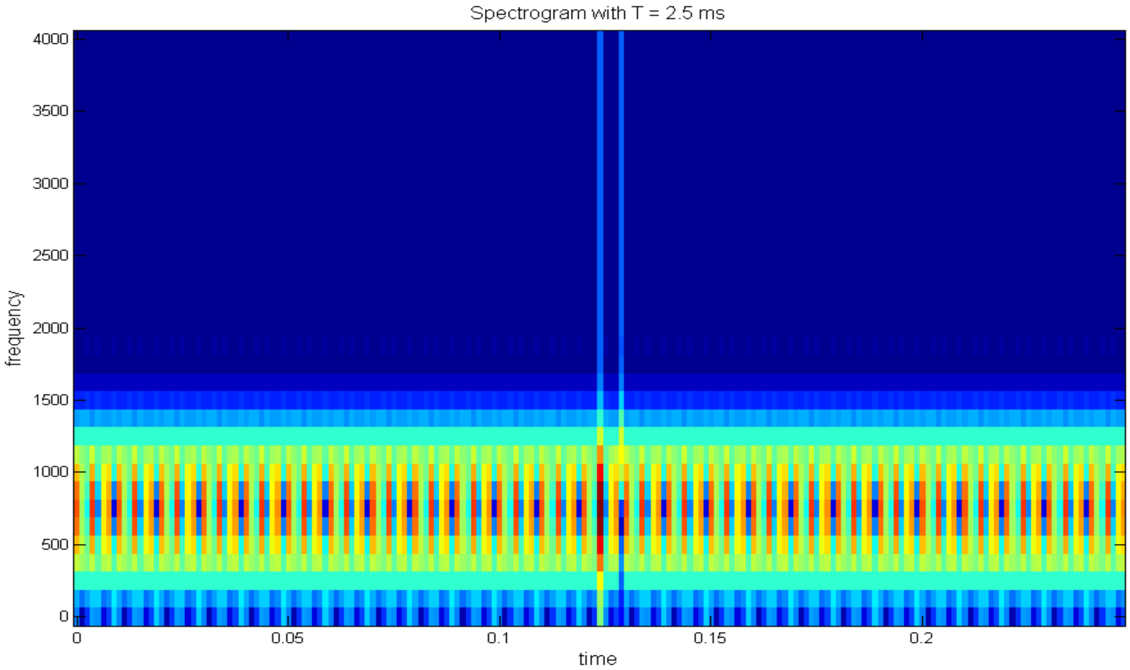


Figure 2.8 : Spectrogram of a signal, x(t), with two dimensional plot which is the sum of two sinusoids of frequencies f1 = 500Hz and f2 = 1000Hz and two impulses at times t1 = 125 ms and t2 = 130 ms. The window width is T = 2.5 ms.

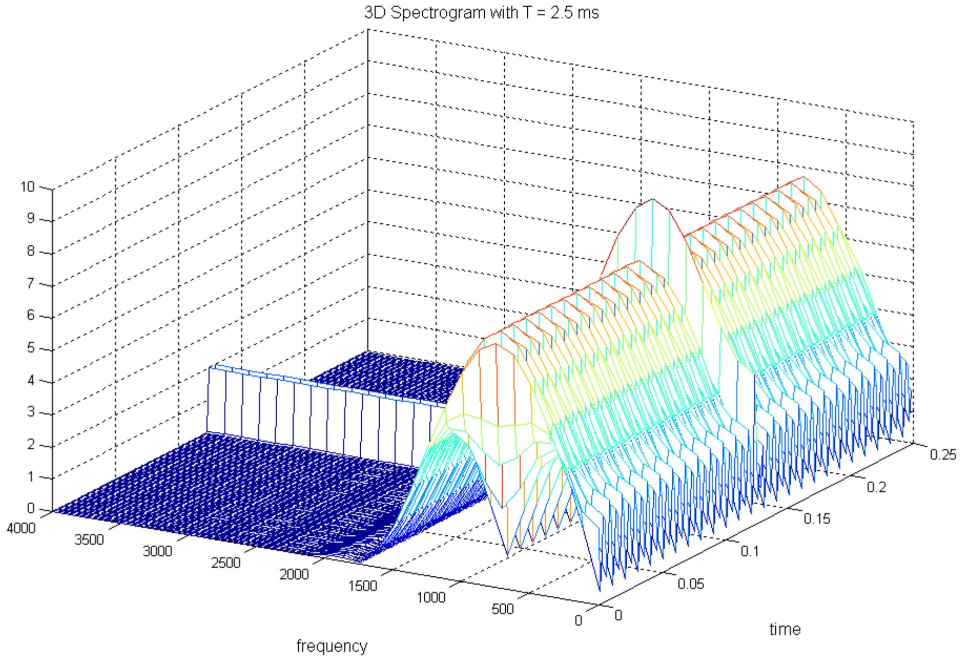


Figure 2.9 : Spectrogram of the same signal, x(t) with three dimensional plot.

Now suppose that the window size is changed to $T = 8$ ms. As the spectrograms in figure 2.10 and figure 2.11 show, the frequencies can be resolved but not the spikes. In that situation the time resolution is not good enough to distinguish spikes.

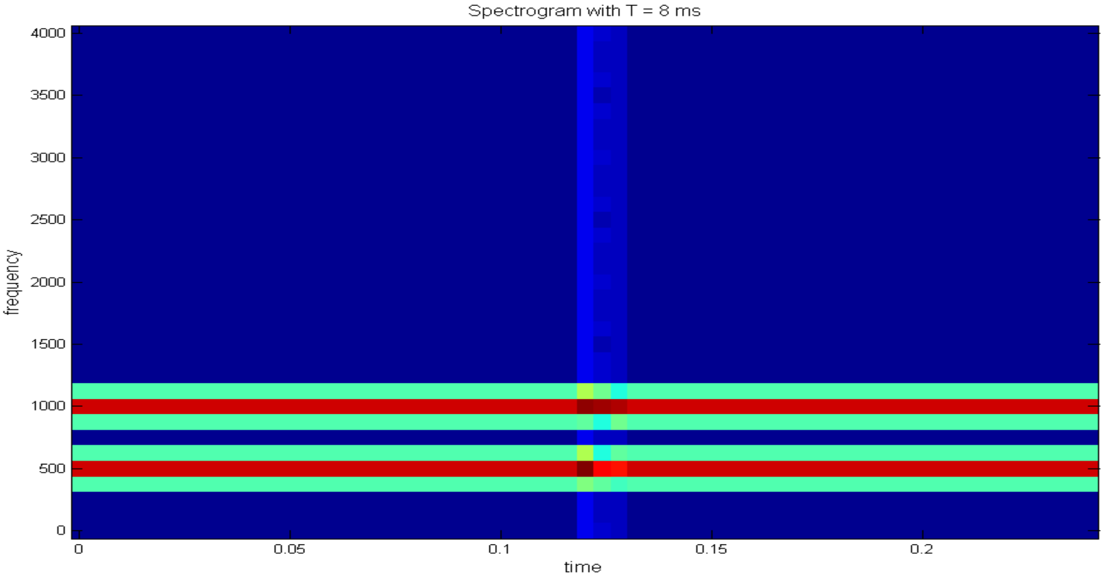


Figure 2.10 : Spectrogram of a signal, x(t), with two dimensional plot which is the sum of two sinusoids of frequencies f1 = 500Hz and f2 = 1000Hz and two impulses at times t1 = 125 ms and t2 = 130 ms but the window width is T = 8 ms.

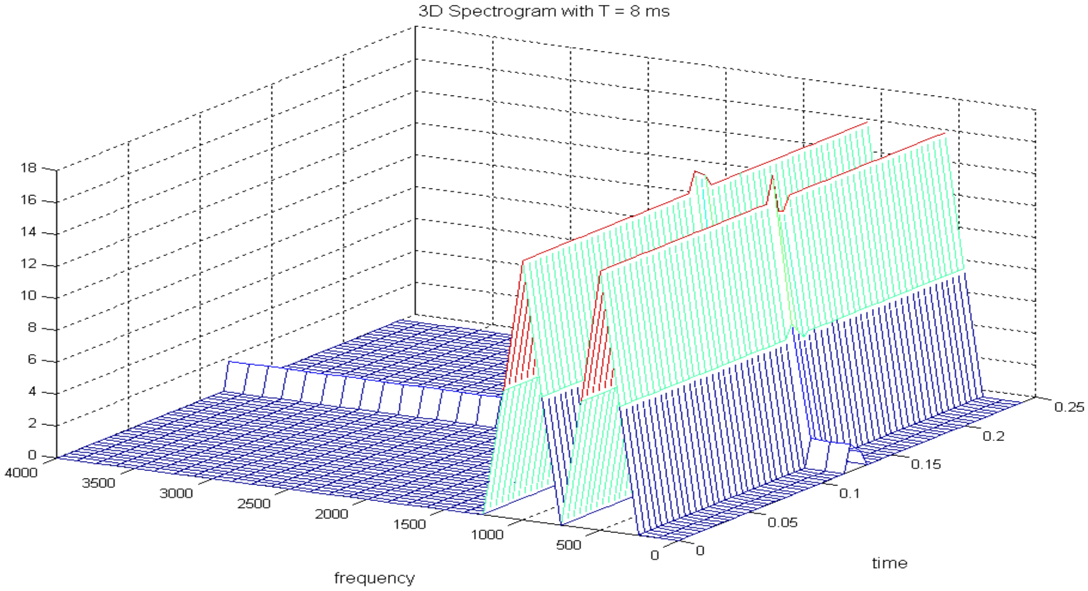


Figure 2.11 : Spectrogram of the same signal, x(t) with three dimensional plot but the window width is T = 8 ms.

There is always a tradeoff between time resolution and frequency resolution in STFT. Once a window has been chosen for STFT, the time-frequency resolution is fixed over the entire time-frequency plane because the same window is used at all frequencies (Rioul & Vetterli 1991).

2.3.3 The Continuous Wavelet Transform

The Continuous Wavelet Transform (CWT) provides a time-scale description similar to the STFT but it has some important differences:

- Scale is used instead of frequency which may have a better relationship to the problem at hand.
- The CWT is able to resolve both time and scale (frequency) events better than the STFT.
- By restricting to a discrete set of parameters, the discrete wavelet transform is obtained which corresponds to an orthogonal basis of functions all derived from a single function called the mother wavelet.
- The basis functions in the discrete wavelet transform are not solutions of differential equations as in the Fourier case.
- The basis functions are ``near optimal" for a wide class of problems. This means that the analysis coefficients drop off rapidly.
- The calculation of the coefficients from the signal can be done efficiently. While the computational complexity of the fast Fourier transform (FFT) is $O(nlog_2 n)$, the complexity of discrete wavelet transform is $O(n)$. This means the number of floating-point multiplications and additions increase linearly with the length of the signal.

The formula for the CWT is:

$$C(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} x(t) w\left(\frac{t-b}{a}\right) dt \qquad (2.13)$$

The function $w(t)$ is called the (mother) wavelet. It is taken to be a ``small wave". For example, the Haar wavelet is a single cycle of the square wave of period 1. Another example, morlet wavelet has the formula:

$$w(t) = e^{-t^2/2} cos(5t) \tag{2.14}$$

It is also a small wave since the gaussian exponential, $e^{-t^2/2}$, is effectively zero outside the interval $-3 < t < 3$.

The graph of $w((t-b)/a)$ is obtained by stretching the graph of $w(t)$ by the factor a, called the scale, and shifting in time by b. The time-shifted and time-scaled wavelet is sometimes called a baby wavelet.

The figure 2.12 shows a signal $x(t)$ along with the Haar wavelet with two different scales and shifts. The subsequent figure shows a signal $x(t)$ along with the Morlet wavelet at three scales and shifts.

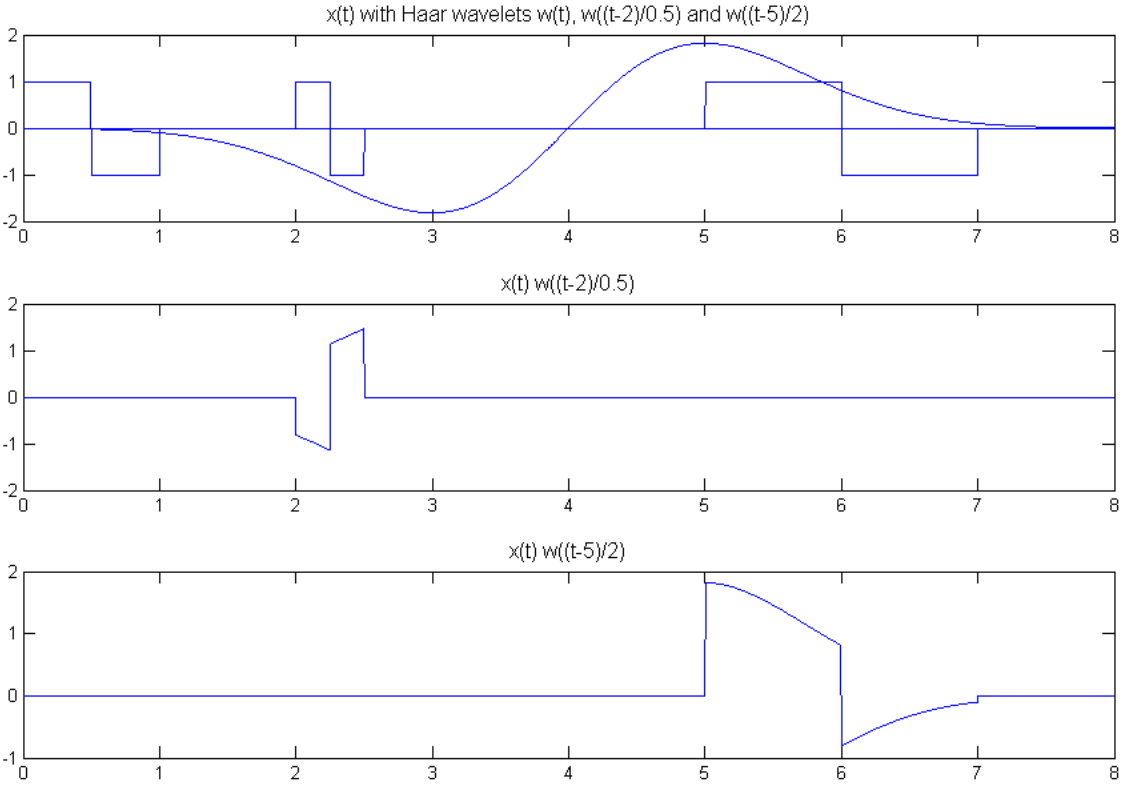


Figure 2.12 : The Haar wavelet with two different scales and shifts.

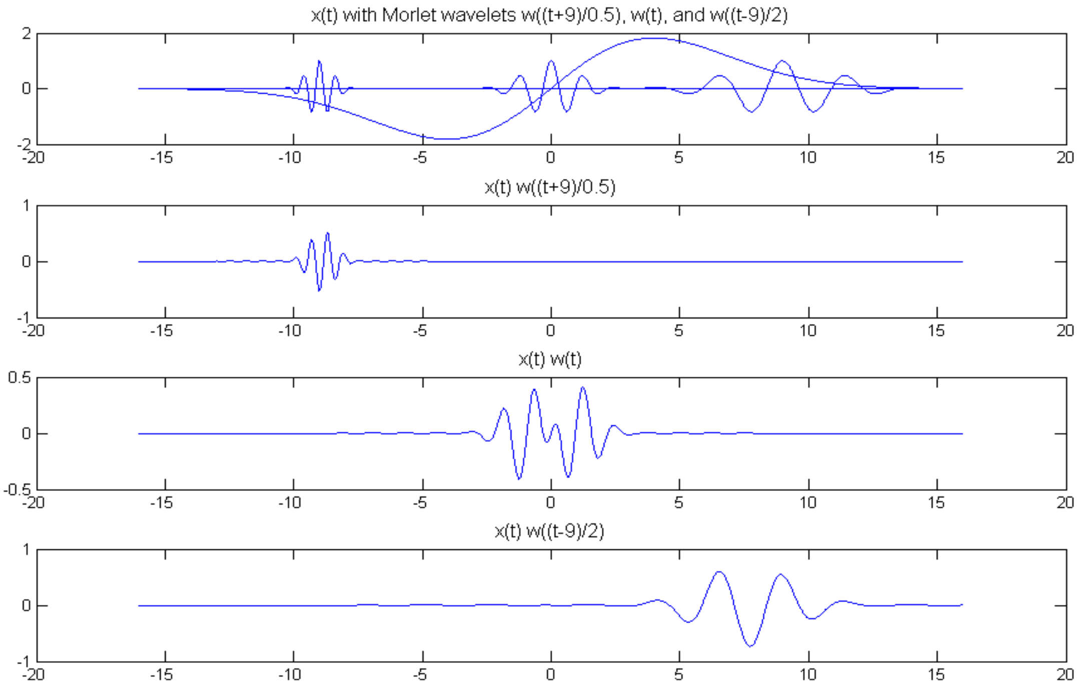


Figure 2.13 : The Morlet wavelet with three different scales and shifts.

CWT can be thought in different ways:

1. The CWT is the inner product or cross correlation of the signal $x(t)$ with the scaled and time shifted wavelet $w((t-b)/a)/\sqrt{a}$. This cross correlation is a measure of the similarity between signal and the scaled and shifted wavelet. It is this point of view that is illustrated in the figures above.
2. For a fixed scale, a, the CWT is the convolution of the signal $x(t)$ with the time reversed wavelet $\frac{1}{\sqrt{a}}w(-t/a)$. That is, the CWT is the output when the signal is fed to the filter with impulse response $\frac{1}{\sqrt{a}}w(-t/a)$.

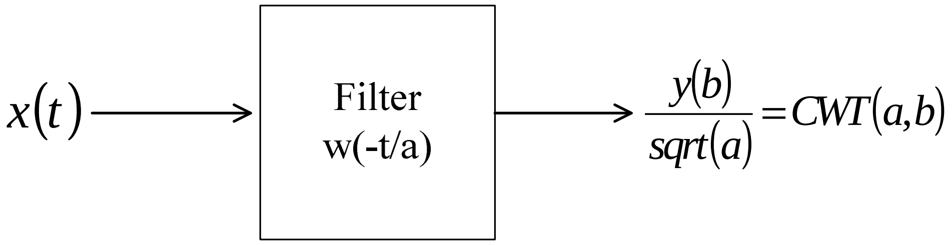


Figure 2.14 : CWT implemented with convolution.

It is this filter point of view which will show the connection to STFT

2.3.3.1 Comparison with STFT

The STFT can be written as:

$$STFT(u,s) = \int_{-\infty}^{\infty} x(t)g(t-s)e^{-j2\pi ut}dt$$

$$= e^{-j2\pi us}\int_{-\infty}^{\infty} x(t)g(t-s)e^{-j2\pi u(t-s)}dt \qquad (2.15)$$

The variable, u, is used for frequency so that later, when the FT is taken, to avoid confusing this frequency variable with the usual one in the transform.

Aside from the initial phase factor, $e^{-j2\pi us}$, this last equation is the convolution of the signal, $x(t)$, with the frequency shifted and time reversed window function, $e^{j2\pi ut}g(-t)$. That is,

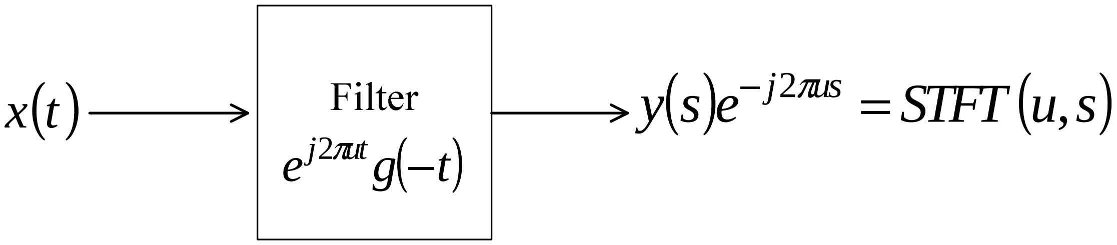


Figure 2.15 : STFT implemented with convolution.

To understand the significance of the filter interpretations of CWT and STFT we can consider the case of the Morlet wavelet, $w(t) = e^{-t^2/2}cos(5t)$, and the STFT with gaussian window function, $g(t) = e^{-t^2/2}$.

The FT of the gaussian window function is: $G(f) = \sqrt{2}e^{-(2\pi f)^2/2}$. Note that this is a window function in the frequency domain. It is a low pass filter which blocks all frquencies above $f = 3/(2\pi) \approx 0.5$ Hz.

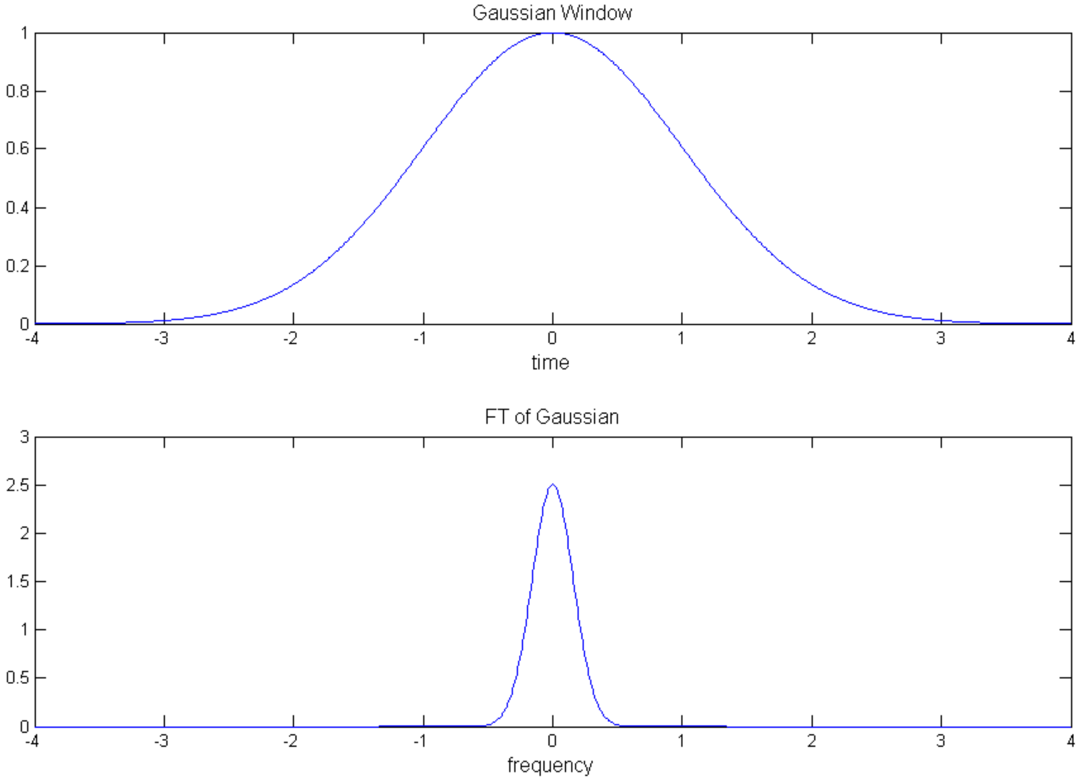


Figure 2.16 : Time domain and frequency domain representation of Gaussian window.

The frequency response of the filter in the STFT is the transform shifted by frequency u. That is, $G(f-u)$. This is a band pass filter centered at frequency u and of approximate width 1 Hz.

That is, computing the spectrogram of a signal using a Gaussian window function is the same as passing the signal through a series of band pass filters of constant bandwidth 1 Hz.

In the case of the CWT the frequency response of the filter when the scale, $a=1$, is: $(1/2)\big(G(f-5/(2\pi))+G(f+5/(2\pi))\big)$. This is a band pass filter centered at frequency $5/(2\pi)\approx 0.8$ Hz with bandwidth 1 Hz.

At scale a the frequency response is $(1/2)\big(G(af-5/(2\pi))+G(af+5/(2\pi))\big)$. This is a band pass filter centered at frequency $5/(2\pi a)$ with a bandwidth of $1/a$ Hz.

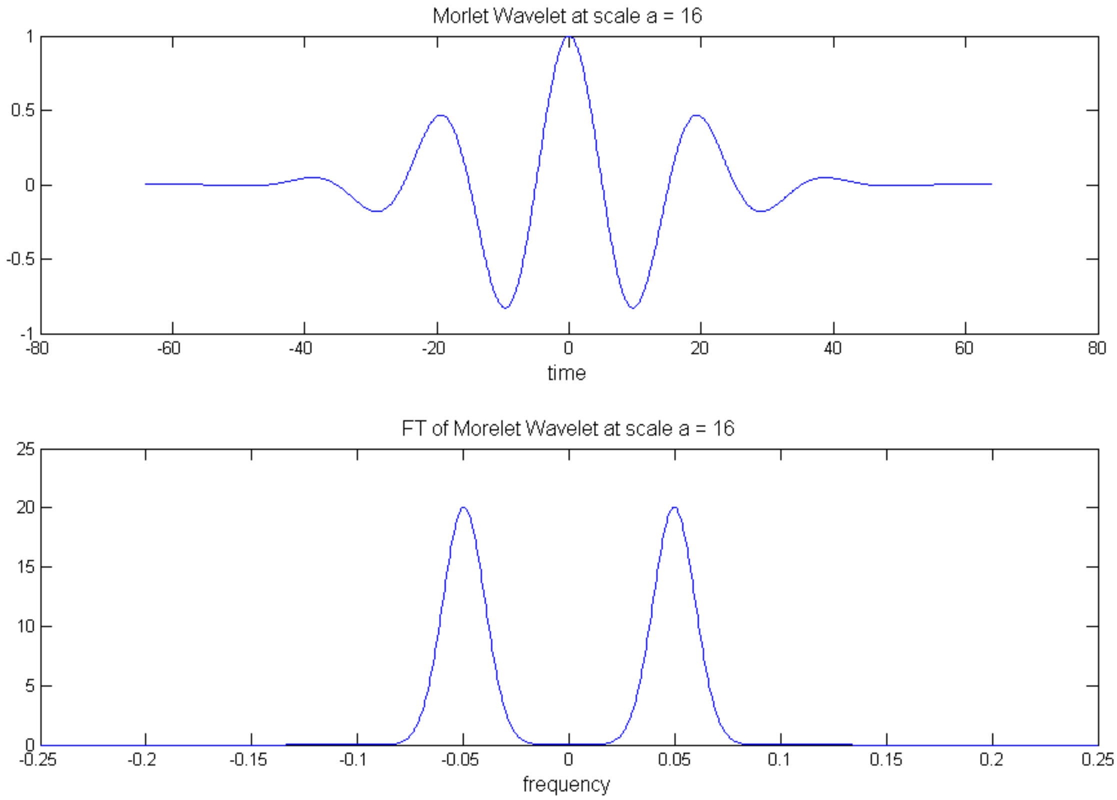


Figure 2.17 : Time domain and frequency domain representation of Morlet wavelet.

The is a constant Q filter

$$Q = \frac{center\ frequency}{bandwidth} = \frac{5/(2\pi a)}{1/a} = \frac{5}{2\pi} \approx 0.8 \qquad (2.16)$$

That is to say, computing the CWT of a signal using the Morlet wavelet is the same as passing the signal through a series of bandpass filters centered at $f = \frac{5/2\pi}{a}$ with constant Q of $5/2\pi$.

This shows the essential difference between the STFT and the CWT. In the STFT the frequency bands have a fixed width (1 Hz for Gaussian). In the CWT the frequency bands grow and shrink with the frequency (scale) being used. This allows good frequency resolution at low frequencies and good time resolution at high frequencies.

2.3.3.2 The scalogram

The magnitude of the CWT is called the scalogram. Scalogram can be shown by 2 dimensional plots with time on the horizontal axis, scale on the vertical axis, and amplitude given by a gray-scale color. Alternately, scalogram can be shown as 3 dimensional plots.

In the following example, $x(t)$ is the sum of two sinusoids of frequencies $f_1 = 500Hz$ and $f_2 = 1000HZ$ and two impulses at times $t_1 = 125$ ms and 130 ms.

Using the Morlet wavelet, the following scalogram is obtained:

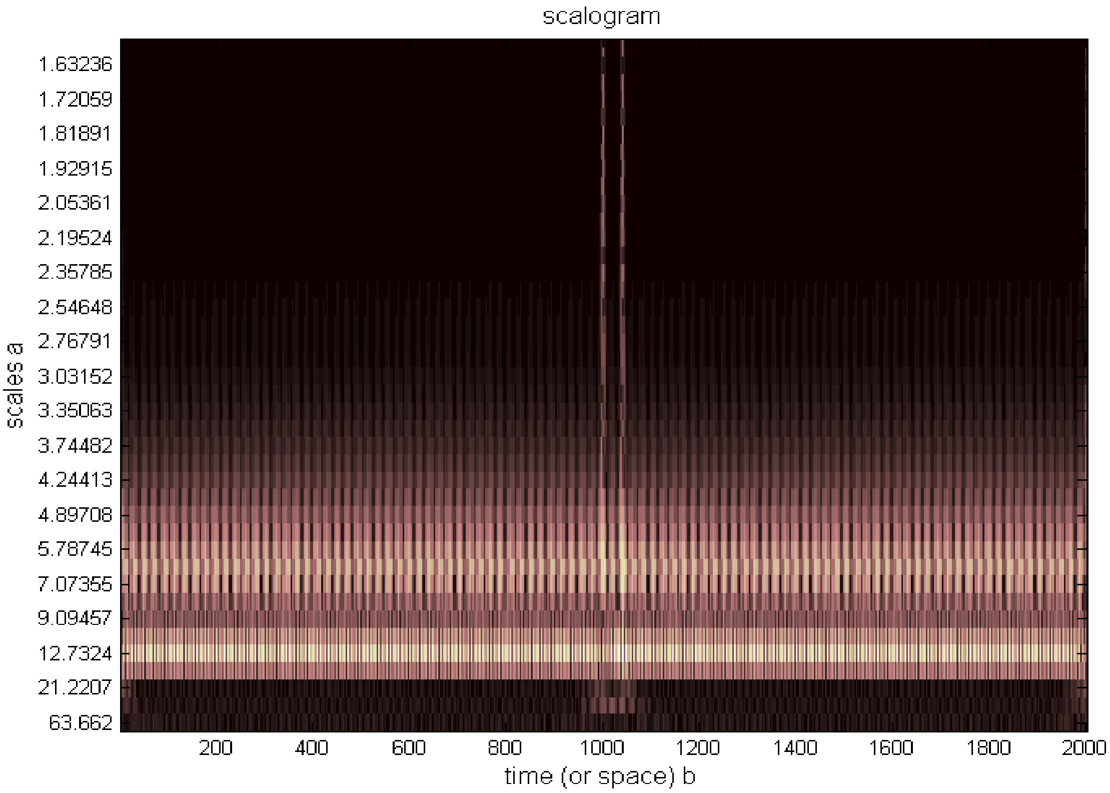


Figure 2.18 : Scalogram of a signal, x(t), with two dimensional plot which is the sum of two sinusoids of frequencies f1 = 500Hz and f2 = 1000Hz and two impulses at times t1 = 125 ms and t2 = 130 ms.

Scale, a, is converted to frequency, f, by using the formula $= (5/2\pi a)$. A new scalogram using frequency instead of scale can be formed shown in figure 2.19:

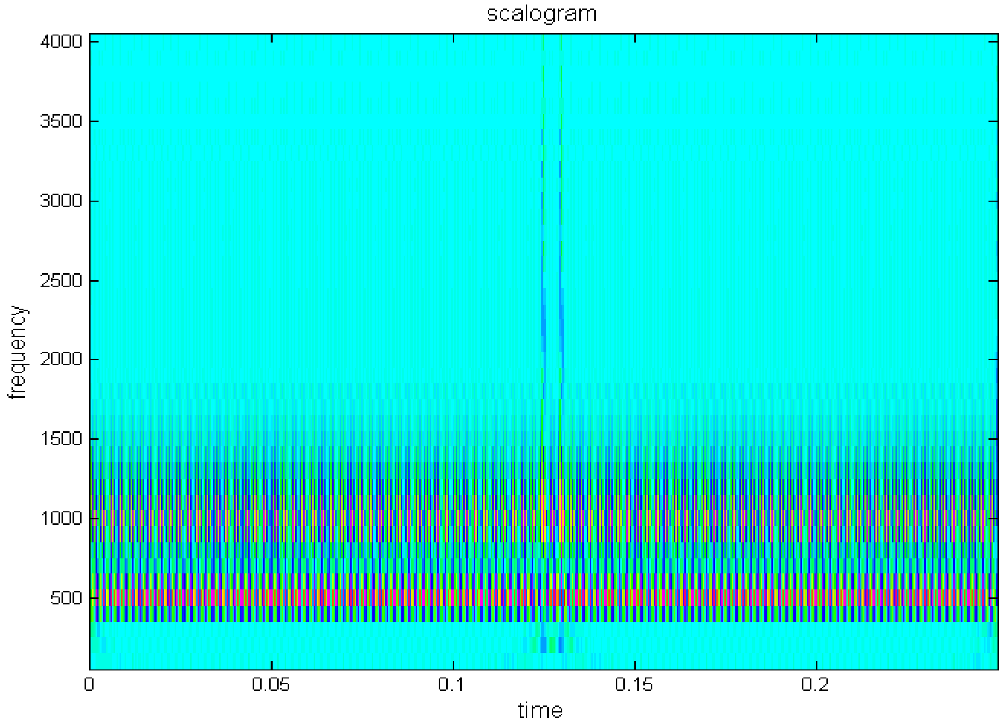


Figure 2.19 : Scalogram of the same signal but frequency is used instead of scale.

To see clearly that the frequencies are resolved by the scalograms, 3 dimensional plot can be used as shown in figure 2.20.

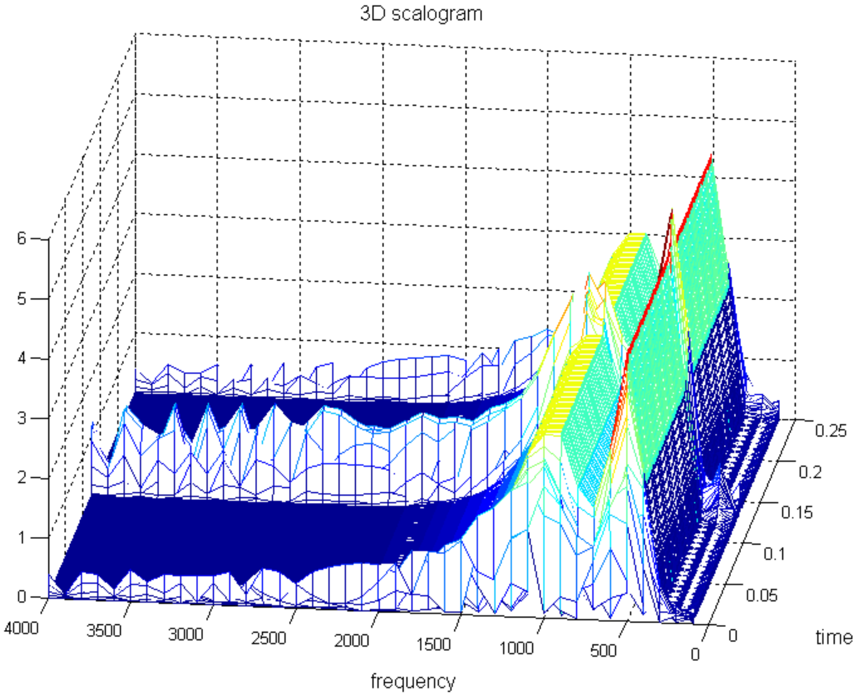


Figure 2.20 : Scalogram of the same signal with three dimensional plot.

As it can be seen from the figures above scalogram gives good frequency-resolution at lower frequencies (high scale) and limited frequency-resolution at high frequencies (low scale) where as spectrogram has fixed resolution.

2.3.4 Comparative Visualization

A comprehensive visualization of various time-frequency representations, shown in figure below, demonstrates the time-frequency resolution for a given signal in various transform domains (Shukla 2003).

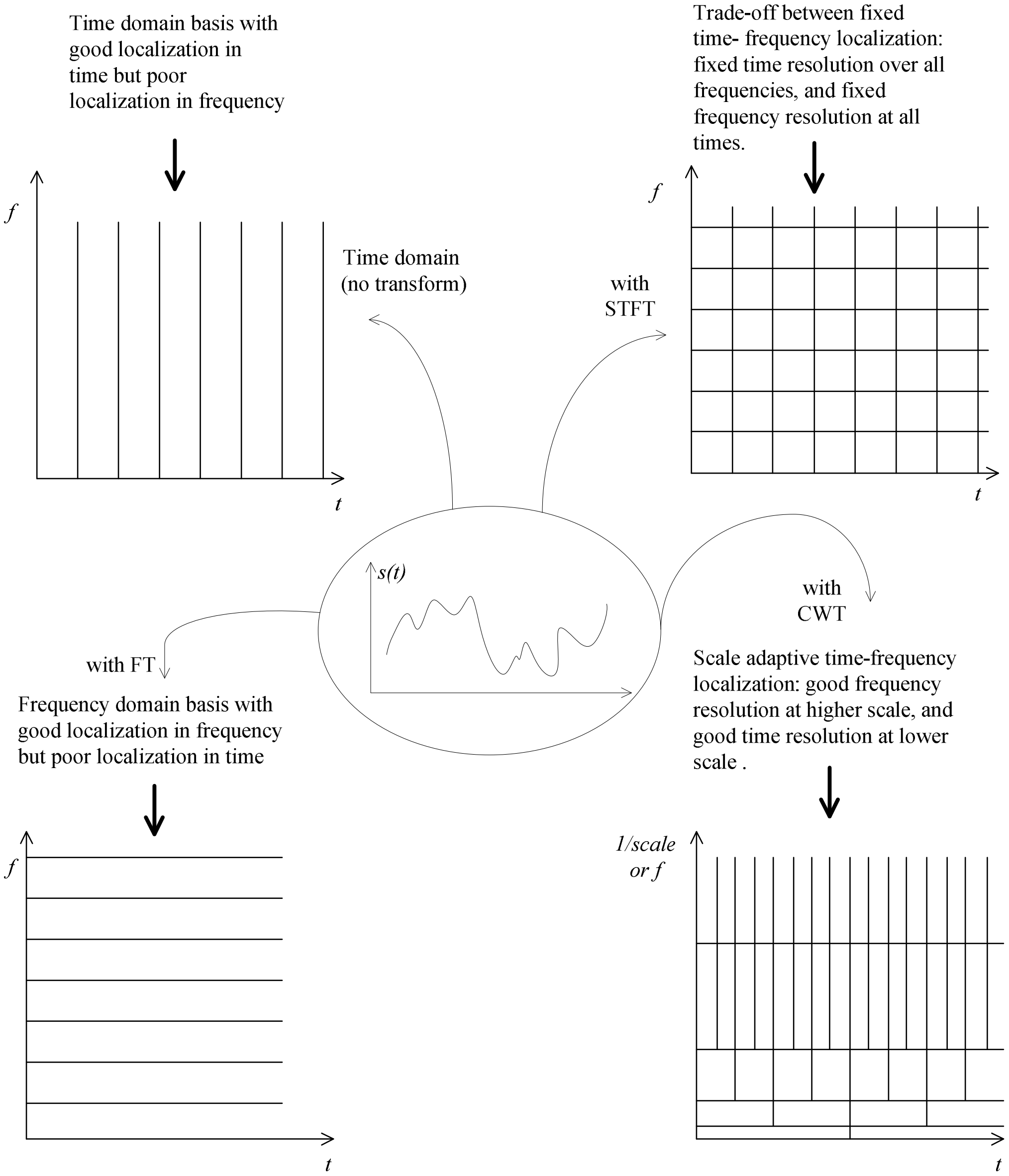

Time domain basis with good localization in time but poor localization in frequency
f
t
Time domain (no transform)
Trade-off between fixed time- frequency localization: fixed time resolution over all frequencies, and fixed frequency resolution at all times.
with STFT
f
t
s(t)
with FT
Frequency domain basis with good localization in frequency but poor localization in time
f
t
with CWT
Scale adaptive time-frequency localization: good frequency resolution at higher scale, and good time resolution at lower scale .
1/scale or f
t

Figure 2.21 : Comparative visualization of various time-frequency representations.

2.3.5 Analysis and Synthesis with Wavelets

Recall that with the STFT, an orthogonal basis of functions can be obtained by choosing equally spaced frequency and time samples $STFT(n/T, mT)$. To ensure the orthogonality the window function must be chosen as a box of width T.

In some cases we can get an orthogonal basis of functions in the CWT case by choosing the scales to be powers of 2 and the times to be an integer multiple of the scales. That is to say, for integers j and k we consider:

$$C\left(1/2^j, k/2^j\right) = 2^{j/2} \int_{-\infty}^{\infty} x(t) w\left(2^j t - k\right) dt \tag{2.17}$$

To simplify the notation a doubly indexed set of baby wavelets are defined as follows:

$$w_{j,k} = 2^{j/2} w\left(2^j t - k\right) \tag{2.18}$$

It then follows that the values $C\left(1/2^j, k/2^j\right)$ are the analysis coefficients for these functions. That is,

$$C\left(1/2^j, k/2^j\right) = \langle x(t), w_{j,k}(t) \rangle \tag{2.19}$$

There is a large class of wavelet functions for which the set of baby wavelets is an orthogonal basis. These are the orthogonal wavelets. The simplest of these is the Haar wavelet.

In the case of an orthogonal wavelet the analysis formula is called the discrete wavelet transform (DWT):

$$DWT(Analysis): c_{j,k} = \int_{-\infty}^{\infty} x(t)\, w_{j,k}(t) dt \tag{2.20}$$

The recovery of the signal through the synthesis formula is called the inverse discrete wavelet transform (IDWT).

$$IDWT(Synthesis): x(t) = \sum_{j}\sum_{k} c_{j,k} w_{j,k} \tag{2.21}$$

Note that the Time-Scale Diagram for the DWT is a set of samples of the Time-Scale Diagram for the CWT. The samples are quite ``sparse" for large scale and more ``dense" for small scale.

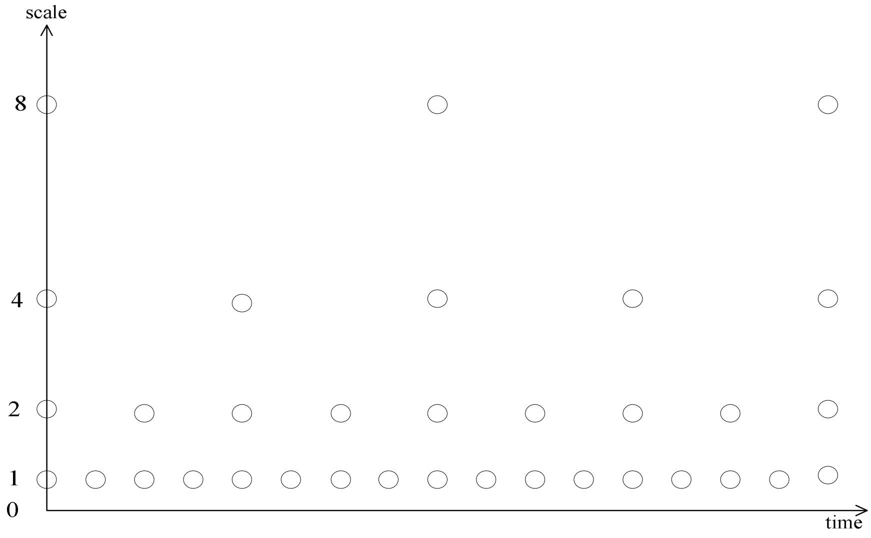


Figure 2.22 : Time-scale diagram for the discrete wavelet transform.

2.3.5.1 The Haar wavelet

Most results about wavelets are simple to see in the case of the Haar wavelet. It is best to keep this case in mind to guide your thinking about wavelets in general. With this in mind we should thoroughly understand the Haar case.

The first point to understand is that the Haar baby wavelets are orthogonal to each other.

The wavelet function is a single cycle of a square wave of period 1.

$$w(t) = \begin{cases} 1 & 0 \le t \le 1/2 \\ -1 & 1/2 < t \le 1 \\ 0 & elsewhere \end{cases} \tag{2.22}$$

Then,

$$w_{j,k}(t) = \sqrt{2^j}\,w(2^j t - k)\ for\ all\ j,k \tag{2.23}$$

The factor of $\sqrt{2^j}$ is to make the energy of the signal 1.

The function $w_{j,k}(t)$ is a single cycle of a square wave extending from time $k/2^j$ to $(k+1)/2^j$.

From this description it is easy to see that baby Haar wavelets, $w_{j,m}$ and $w_{j,n}$ of the same scale, 2^{-j} , but different positions $m/2^j$ and $n/2^j$, are orthogonal because their graphs don't overlap.

It is also true that Haar baby wavelets of different scales are orthogonal. To see this it is best to first consider the case of $w_{0,0}(t)$ and $w_{1,0}(t)$.

Since $w_{1,0}(t) = \sqrt{2}w(2t)$ it follows (see the figure2.23) that $w_{1,0}(t)$ completes its cycle from positive to negative while $w(t)$ is constantly 1 so that the integral of the product is 0.

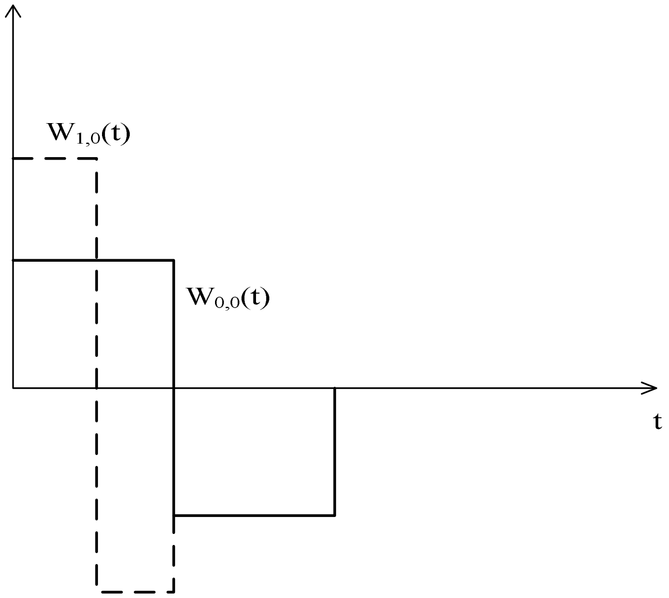


Figure 2.23 : Orthogonality for Haar wavelet.

In the general case of $w_{j,k}$ and $w_{j1,k1}$ if the graphs overlap then one of the functions completes its cycle from 1 to -1 while the other is constant. This shows that these two are orthogonal (Phillips 2009).

2.4 MULTIRESOLUTION ANALYSIS

In multiresolution analysis (Burrus, Gopinath & Guo 1998) we would like to find wavelets, $w(t)$, with the same properties as the Haar case. That is, The baby wavelets, $w_{j,k} = \sqrt{2^j}w\left(2^j t - k\right)$, for all j and k , form an orthogonal basis. This implies that we have the usual Analysis-Synthesis for all signals:

$$Analysis{:}\, c_{j,k} = \int_{-\infty}^{\infty} x(t) w_{j,k}(t) dt \tag{2.24}$$

$$Synthesis{:}\, x(t) = \sum_j \sum_k c_{j,k} w_{j,k}(t) \tag{2.25}$$

Such wavelets give rise to a Multiresolution Analysis derived as follows.

Define W_j to be set of all signals, $x(t)$, which can be synthesized from the baby wavelets $w_{j,k}(t), -\infty < k < \infty$. These spaces are orthogonal to each other and any (energy) signal, $x(t)$ can be synthesized as (note that in the following formula $x_j(t)$ is in the space W_j):

$$x(t) = \sum_{j=-\infty}^{\infty} x_j(t) \; where \; x_j(t) = \sum_{k=-\infty}^{\infty} c_{j,k} w_{j,k}(t) \tag{2.26}$$

There is another way to express this idea. Define V_j to be the set of all signals, $x(t)$, which can be synthesized from the baby wavelets $w_{i,k}$ where $i < j$ and $-\infty < k < \infty$. That is

$$x(t) = \sum_{i=-\infty}^{j-1} \sum_{k} c_{i,k} w_{i,k}(t) \tag{2.27}$$

The spaces V_j are nested inside each other. As follows,

$$\{0\} \subset \cdots \subset V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \subset \cdots \subset L^2 \tag{2.28}$$

As j goes to infinity V_j enlarges to become all energy signals (L^2). And as j goes to negative infinity V_j shrinks down to only the zero signal. It is clear from the definitions that every signal in V_{j+1} is a sum of a signal in V_j and W_j because:

$$x(t) = \sum_{i=-\infty}^{j} \sum_{k} c_{i,k} w_{i,k}(t) = \sum_{i=-\infty}^{j-1} \sum_{k} c_{i,k} w_{i,k}(t) + \sum_{k} c_{j,k} w_{j,k}(t) \tag{2.29}$$

So, it can be written as:

$$V_{j+1} = W_j + V_j \tag{2.30}$$

This shows that the spaces W_j are the differences (in the subspace sense) between adjacent spaces V_j and V_{j+1}. The spaces V_j and W_j can be visualized as in figure 2.24:

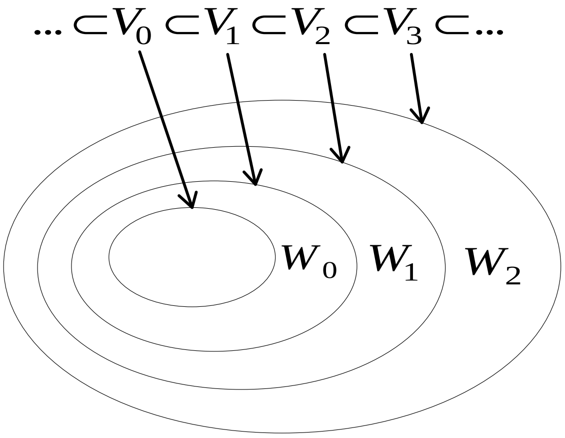

$\ldots \subset V_0 \subset V_1 \subset V_2 \subset V_3 \subset \ldots$
W_0
W_1
W_2

Figure 2.24 : Nested subspaces in multiresolution analysis.

The term Multiresolution Analysis refers to analyzing signals in relation to this nested sequence of subspaces. To get a better idea of multiresolution analysis, let's decompose a signal, $x(t)$, in V_0 a few times:

$$\begin{aligned} V_0 &= V_{-1} + W_0 \\ &= V_{-2} + W_{-2} + W_{-1} \\ &= V_{-3} + W_{-3} + W_{-2} + W_{-1} \\ &= V_{-4} + W_{-4} + W_{-3} + W_{-2} + W_{-1} \qquad (2.31) \end{aligned}$$

This leads to various decompositions:

$$\begin{aligned} x(t) &= A_1(t) + D_1(t) \\ &= A_2(t) + D_2(t) + D_1(t) \\ &= A_3(t) + D_3(t) + D_2(t) + D_1(t) \\ &= A_4(t) + D_4(t) + D_3(t) + D_2(t) + D_1(t) \qquad (2.32) \end{aligned}$$

Where $D_i(t)$, in W_{-i} , is called the detail at level i and $A_i(t)$, in V_{-i} , is called the approximation at level i. Decomposition can be shown in figure 2.25, in the figure a sinusoidal signal with two different frequencies is decomposed into five levels and as you can see the breakdown in frequency can be easily seen in D_1 (First level of detail). In multiresolution analysis different aspects of the signal appear in the details and the approximations.

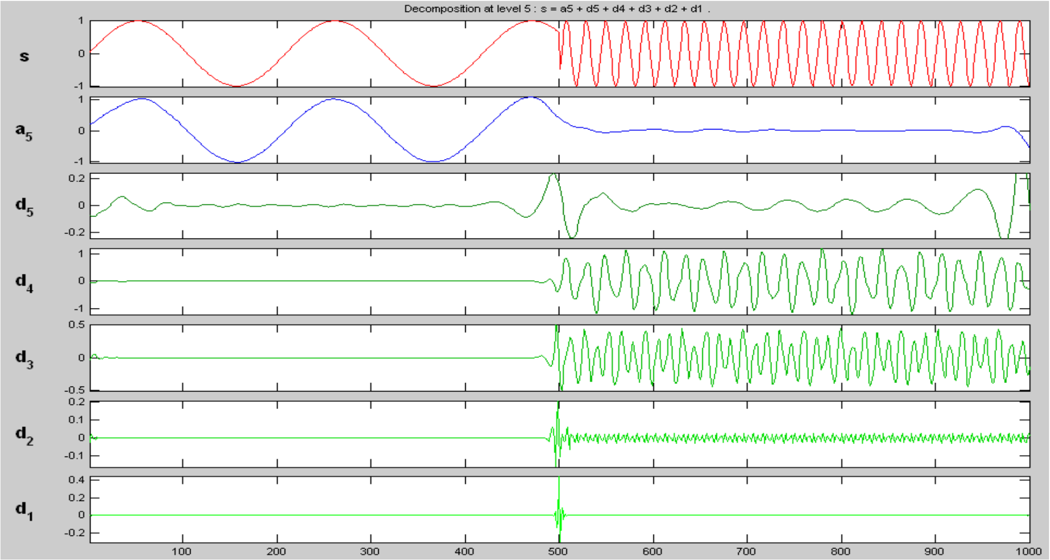


Figure 2.25 : Five level decomposition of an example signal.

2.4.1 The Two Scale Property of Multiresolution

A signal $x(t)$ is in the space V_j if and only if $x(2t)$ is in the next space V_{j+1}.

This follows from the formula:

$$w_{i,k}(2t) = \frac{1}{\sqrt{2}} w_{i+1,k}(t) \tag{2.33}$$

Investigation of the multiresolution analysis leads to a scaling function, a pair of discrete time filters, and a perfect reconstruction filter bank which can be used to calculate the DWT quickly.

2.4.2 The Scaling Function

The useful wavelets, $w(t)$, have a scaling function $\phi(t)$ which can produce the Multiresolution subspaces V_j as follows.

Define the ``baby scaling functions":

$$\phi_{j,k}(t) = \sqrt{2^j}\phi\left(2^j t - k\right) \tag{2.34}$$

Where $-\infty < j < \infty$ and $-\infty < k < \infty$.

Just as for the wavelet, the ``scale" of $\phi_{j,k}(t)$ is $1/2^j$ and the ``position" is $k/2^j$.

Mother scaling function, $\phi(t)$, must be found so that the signals in the space V_j can be synthesized from the baby scale functions $\phi_{j,k}(t)$, $-\infty < k < \infty$.

Since the spaces V_j are obtained from V_0 by time compression or dilation by powers of 2, only the space V_0 is needed to check. That is, the first thing to do is finding a function $\phi(t)$ so that the signals in V_0 can be synthesized from the integer translates $\phi(t-k)$ of the scale function.

For example, in the Haar case the scaling function is the unit box delayed by $1/2$:

$$\phi(t) = \begin{cases} 1 & 0 \leq t \leq 1 \\ 0 & elsewhere \end{cases} \tag{2.35}$$

Then $\phi_{j,k}(t)$ is the box of length $1/2^j$ extending from $k/2^j$ to $(k+1)/2^j$.

To see that the integer translates of $\phi(t)$ form a basis for V_0 note that:

$$\phi(t) = \sum_{J=-1}^{-\infty} 2^{j/2} w_{j,0}(t) \tag{2.36}$$

$$w_{-1,0}(t) = \frac{1}{\sqrt{2}}\big(\phi(t) - \phi(t-1)\big) \tag{2.37}$$

By a similar formula $w_{i,0}(t)$ can be synthesized from $\phi(t)$ and its translates for any negative i.

2.4.2.1 The two scale equation and the filters

There is an important formula connecting the scale function to itself at two different time scales. This fundamental formula is called the Two Scale Equation and it gives rise to one of the filters.

There are discrete time filter coefficients $h_0(n)$ such that:

$$\phi(t) = \sum_n h_0(n)\sqrt{2}\phi(2t-n) \tag{2.38}$$

This follows trivially from the assumption that $V_0 \subset V_1$ but is probably the most important equation involving the scale function.

Since W_0 is also a subset of V_1 there is another two scale equation for the wavelet which gives rise to another filter $h_1(n)$, such that:

$$w(t) = \sum_n h_1(n)\sqrt{2}\,\phi(2t-n) \tag{2.39}$$

For example, in the Haar case, the scale function is the box of width 1 extending from time 0 to time 1.

It follows that $\phi(2t)$ is the box of width 1/2 extending from time 0 to time 1/2.

Similarly, $\phi(2t-1)$ is the box of width 1/2 extending from time 1/2 to time 1.

When these two smaller box functions are added we obtain $\phi(t)$. That is,

$$\phi(t) = \phi(2t) + \phi(2t-1) = \frac{1}{\sqrt{2}}\sqrt{2}\phi(2t) + \frac{1}{\sqrt{2}}\sqrt{2}\phi(2t-1) \tag{2.40}$$

The filter for the scale function is $h_0 = \left[\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right]$

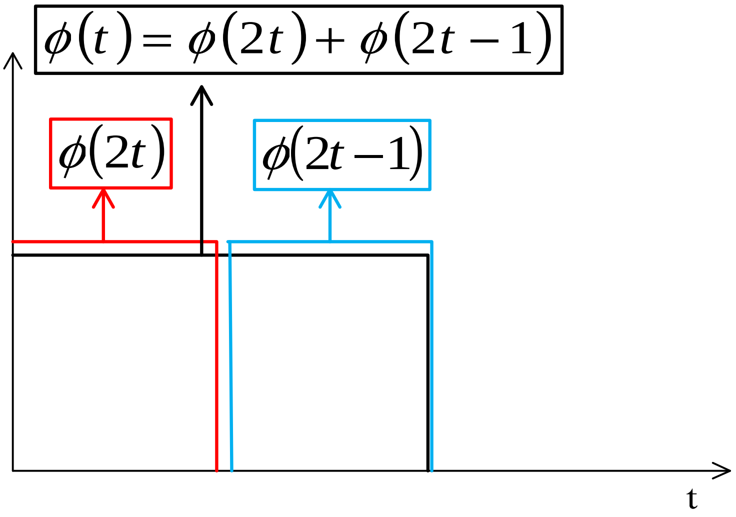


Figure 2.26 : The two scale equation for the Haar scale.

Similarly, the Haar wavelet can be expressed as:

$$w(t) = \phi(2t) - \phi(2t-1) = \frac{1}{\sqrt{2}}\sqrt{2}\phi(2t) - \frac{1}{\sqrt{2}}\sqrt{2}\phi(2t-1) \tag{2.41}$$

The filter for the wavelet is $h_1 = \left[\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}\right]$

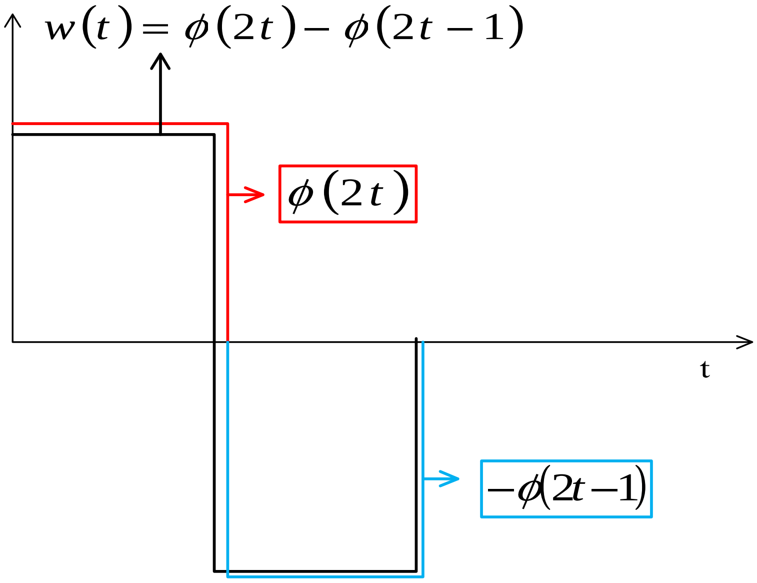


Figure 2.27 : The two scale equation for the Haar wavelet.

2.4.3 The Discrete Wavelet Transform

The Discrete Wavelet Transform (DWT) of a signal, $x(t)$, previously have been defined as a set of analysis coefficients:

$$Analysis: c_{j,k} = \int_{-\infty}^{\infty} x(t) w_{j,k}(t) dt \tag{2.42}$$

From these the signal can be recovered as:

$$Synthesis: x(t) = \sum_{j} \sum_{k} c_{j,k} w_{j,k}(t) \tag{2.43}$$

Assuming the existence of a scaling function, $\phi(t)$ we can now modify this defintion as follows.

Since the spaces V_j are getting larger and larger as j goes to ∞ we can approximate any signal, $x(t)$, closely by choosing a large enough value of $j = J$ and projecting the signal into V_J using the basis $\phi_{J,m}(t)$, (all values of k).

$$cA_0(m) = \int_{-\infty}^{\infty} x(t) \phi_{J,m}(t) dt \tag{2.44}$$

From these we can approximately recover the signal as:

$$x(t) \approx \sum_{m} cA_0(m) \phi_{J,m}(t) \tag{2.45}$$

In effect, we replace the signal, $x(t)$, by the approximate signal given by the projection coefficients, $cA_0(m)$.

After this approximation our signal is now in V_J and we can decompose it using the subspaces V_{J-n} and W_{J-n} with their bases $\phi_{J-n,k}(t)$ and $w_{J-n,k}(t)$. Note that the scale is getting larger and larger as the index $J - n$ gets more negative.

If we take $n = 1$ we get:

$$V_J = W_{J-1} + V_{J-1} \tag{2.46}$$

Using the basis $w_{J-1,k}(t)$ in W_{J-1} and $\phi_{J-1,k}(t)$ in V_{J-1} we have:

$$\begin{aligned} x(t) &= \sum_m cA_0(m)\phi_{J,m}(t) \\ &= \sum_k cA_1(k)\phi_{J-1,k}(t) + \sum_k cD_1(k)w_{J-1,k}(t) \\ &= A_1(t) + D_1(t) \end{aligned} \tag{2.47}$$

The signals $A_1(t)$ and $D_1(t)$ are called the approximation and detail at level 1.

The coefficients $cA_1(k)$ and $cD_1(k)$ are called the approximation coefficients and the detail coeffients at level 1.

$A_1(t)$ can be decomposed further to get:

$$\begin{aligned} x(t) &= A_1(t) + D_1(t) \\ &= \sum_k cA_2(k)\phi_{J-2,k}(t) + \sum_k cD_2(k)w_{J-2,k}(t) + \sum_k cD_1(k)w_{J-1,k}(t) \\ &= A_2(t) + D_2(t) + D_1(t) \end{aligned} \tag{2.48}$$

The signals $A_2(t)$ and $D_2(t)$ are called the approximation and detail at level 2. And the coefficients $cA_2(k)$ and $cD_2(k)$ are called the approximation coefficients and the detail coefficients at level 2.

2.5 FILTER BANKS AND THE DWT

Multiresolution Analysis allows us to decompose a signal into approximations and details. On the theoretical level this is an Analysis-Synthesis situation. That is, the bases $\phi_{j,k}(t)$ and $w_{j,k}(t)$ are used to decompose signals.

On the practical level, we assume that our signal is represented by its approximation coefficients at some scale $1/2^J$ and we decompose it in terms of its coefficients at larger scale. Both points of view are necessary for a real understanding of the subject.

In this section we will show that the approximation and detail coefficients can be computed using the filters previously mentioned. As we must calculate these coefficients at many different scales we will need a filter bank.

2.5.1 Analysis: From Fine Scale to Coarser Scale

In the DWT we have $V_j = V_{j-1} + W_{j-1}$. That is, each signal $x(t)$ in V_j can be expressed in two ways using the basis functions in each of the spaces.

$$\begin{aligned} x(t) &= \sum_k cA_0(k)\phi_{j,k}(t) \\ &= \sum_k cA_1(k)\phi_{j-1,k}(t) + \sum_k cD_1(k)w_{j-1,k}(t) \end{aligned} \tag{2.49}$$

We start with the coefficients $A_0(k)$ at scale index j and produce the two sets of coefficients $A_1(k)$ and $D_1(k)$ at scale index $j-1$ (Analysis). Alternately, we can start with the two sets of coefficients $A_1(k)$ and $D_1(k)$ at scale index $j-1$ and produce the coefficients $A_0(k)$ at scale index j (Synthesis).

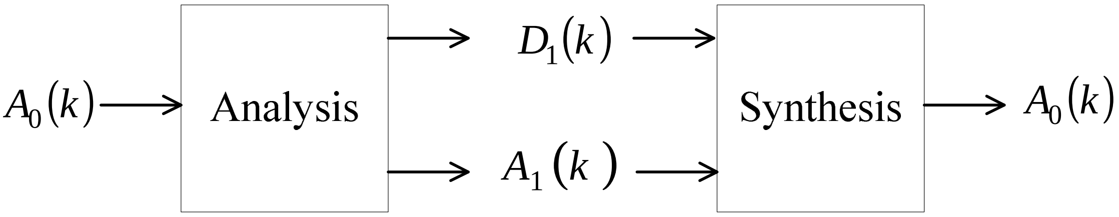


Figure 2.28 : Analysis and synthesis view of DWT.

We can show that the two operations of Analysis and Synthesis are produced by certain filter banks.

As the wavelets and the scales at each index level are orthogonal we can compute the coefficients $cA_1(k)$ and $D_1(k)$ by the usual inner product formula:

$$\begin{aligned} cA_1(k) &= \langle x(t), \phi_{j-1,k}(t)\rangle \\ &= \langle \sum_n cA_0(n)\phi_{j,n}(t), \phi_{j-1,k}(t)\rangle \\ &= \sum_n cA_0(n)\langle \phi_{j,n}(t), \phi_{j-1,k}(t)\rangle \end{aligned} \tag{2.50}$$

To complete this calculation the inner product must be computed:

$$\begin{aligned} \langle \phi_{j,n}(t), \phi_{j-1,k}(t)\rangle &= \int_{-\infty}^{\infty} \sqrt{2^j}\phi(2^j - n)\sqrt{2^{j-1}}\phi(2^{j-1}t - k)dt \\ &= \int_{-\infty}^{\infty} \sqrt{2^{2j-1}}\phi(2^j - n)\phi(2^{j-1}t - k)dt \\ &= \int_{-\infty}^{\infty} \sqrt{2}\phi(2s + 2k - n)\phi(s)ds \\ &= \int_{-\infty}^{\infty} \sqrt{2}\,\phi(2s + 2k - n)\sum_m h_0(m)\sqrt{2}\,\phi(2s - m)ds \end{aligned}$$

$$= \sum_m h_0(m) \int_{-\infty}^{\infty} \phi(2s + 2k - n)\phi(2s - m)2ds$$

$$= h_0(n - 2k) \tag{2.51}$$

The calculation which is previously started can be completed as :

$$cA_1(k) = \sum_n h_0(n - 2k)cA_0(n) \tag{2.52}$$

The detail coefficients can be computed similarly.

$$cD_1(k) = \langle x(t), w_{j-1,k}(t)\rangle$$

$$= \langle \sum_n cA_0(n)\phi_{j,n}(t), w_{j-1,k}(t)\rangle$$

$$= \sum_n cA_0(n)\langle \phi_{j,n}(t), w_{j-1,k}(t)\rangle \tag{2.53}$$

To complete this calculation we have to compute the inner product:

$$\langle \phi_{j,n}(t), w_{j-1,k}(t)\rangle = \int_{-\infty}^{\infty} \sqrt{2^j}\phi\left(2^j - n\right)\sqrt{2^{j-1}}w\left(2^{j-1}t - k\right)dt$$

$$= \int_{-\infty}^{\infty} \sqrt{2^{2j-1}}\phi\left(2^j - n\right)w\left(2^{j-1}t - k\right)dt$$

$$= \int_{-\infty}^{\infty} \sqrt{2}\phi(2s + 2k - n)w(s)ds$$

$$= \int_{-\infty}^{\infty} \sqrt{2}\,\phi(2s + 2k - n) \sum_m h_1(m)\sqrt{2}\,\phi(2s - m)ds$$

$$= \sum_m h_1(m) \int_{-\infty}^{\infty} \phi(2s + 2k - n)\phi(2s - m)2ds$$

$$= h_1(n - 2k) \tag{2.54}$$

Upon substitution of this formula into the previous calculation we get:

$$cD_1(k) = \sum_n h_1(n - 2k)cA_0(n) \tag{2.55}$$

2.5.1.1 Filtering and downsampling

The two formulas for the approximation and detail coefficients look similar to convolution but there is a downsampling involved.

$$cA_1(k) = \sum_n h_0(n - 2k)cA_0(n) \tag{2.56}$$

$$cD_1(k) = \sum_n h_1(n - 2k)cA_0(n) \tag{2.57}$$

Downsampling a discrete time signal $x(n)$ is performed by omitting every other value. We can think of a system whose input is $x(n)$ and whose output is $(n) = x(2n)$.

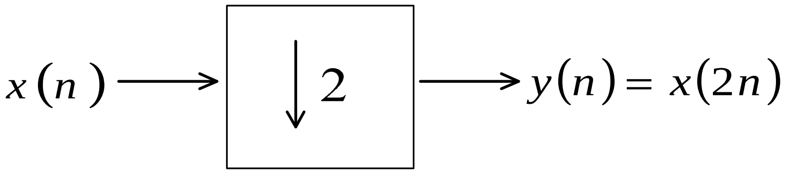


Figure 2.29 : Downsampler.

To understand the approximation and detail formulas it will help to define the time reversed filters $\tilde{h}_0(n) = h_0(-n)$ and $\tilde{h}_1(n) = h_1(-n)$. We temporarily use $m = 2k$ to see the convolution.

$$\begin{aligned} u(m) &= \sum_{n} h_0(n-m)cA_0(n) \\ &= \sum_{n} \tilde{h}_0(m-n)cA_0(n) \\ &= \tilde{h}_0 * cA_0(m) \end{aligned} \tag{2.58}$$

If we follow this filter by the downsampler we get the approximation coefficients at the next level.

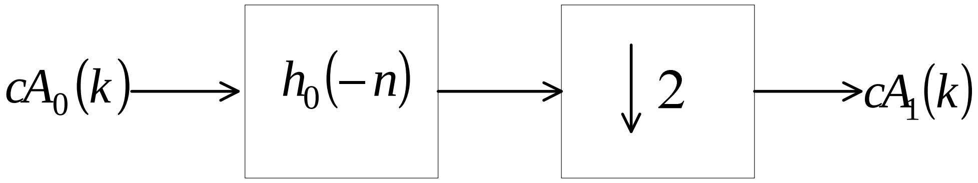


Figure 2.30 : Filter and downsampler for approximation coefficients.

The same calculation holds for the detail coefficients. That is, convolution with the time reversed filter $h_1(-n)$ followed by downsampling produces the detail coefficients at the next level.

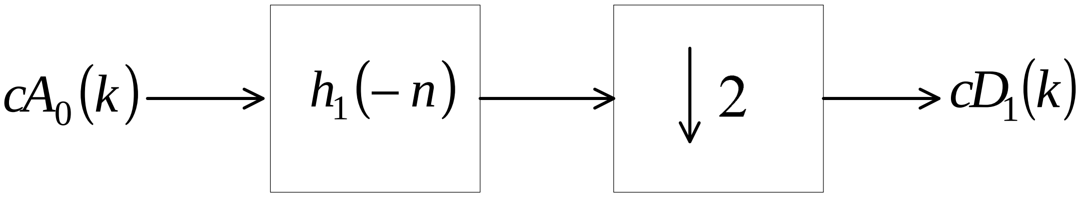


Figure 2.31 : Filter and downsample for detail coefficients.

2.5.1.2 The one-stage analysis filter bank

We actually should think of the two filtering operations followed by downsampling as a filter bank.

We are analyzing a function $x(t)$ in V_j into a detail $D_1(t)$ in $W_{j-1}$and an approximation, $A_1(t)$, in V_{j-1}, using a filter bank to calculate the coefficients $cD_1(k)$ and $cA_1(k)$.

$$x(t) = \sum_k cA_0(k)\phi_{j,k}(t)$$

$$= \sum_k cA_1(k)\,\phi_{j,k}(t)$$

$$= \sum_k cA_1(k)\,\phi_{j-1,k}(t) + \sum_k cD_1(k) w_{j-1,k}(t)$$

$$= \; A_1(t) + D_1(t) \tag{2.59}$$

Note that the number of data values produced by the filter bank is about the same as the number of data values entering the system. To see this let, l_x be the length of the input vector and assume that the filters both have length l_f. The length of the convolution is $l_x{+}l_f - 1$ so that the lengths of cA and cD are $floor\big((l_x{+}l_f - 1)/2\big)$. The overall size of the data emerging from the filterbank is increased by the length of the filter minus 1.

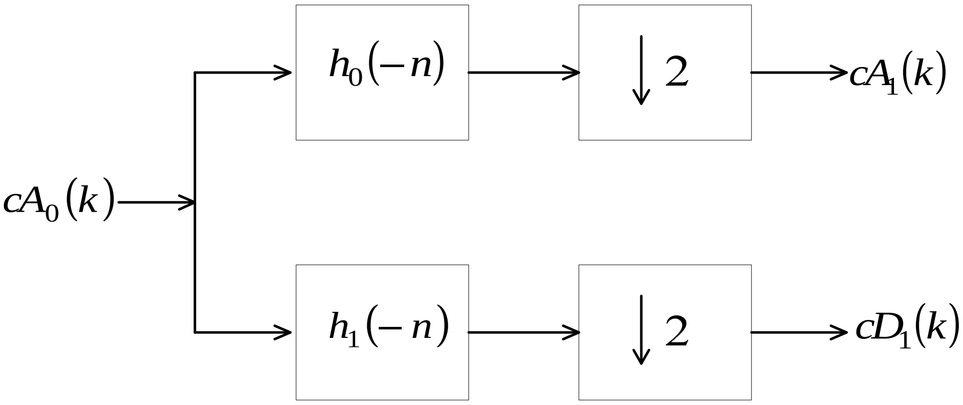


Figure 2.32 : One stage filter bank.

2.5.1.3 The analysis filter bank

We can further decompose $A_1(t)$ to get:

$$x(t) = A_1(t) + D_1(t)$$

$$= \sum_k cA_2(k)\,\phi_{j-2,k}(t) + \sum_k cD_2(k) w_{j-2,k}(t) + \sum_k cD_1(k) w_{j-1,k}(t)$$

$$= \; A_2(t) + D_2(t) + D_1(t) \tag{2.60}$$

We can then decompose $A_2(t)$ to get:

$$x(t) = A_1(t) + D_1(t)$$

$$= \sum_k cA_2(k)\,\phi_{j-2,k}(t) + \sum_k cD_2(k) w_{j-2,k}(t) + \sum_k cD_1(k) w_{j-1,k}(t)$$

$$= A_2(t) + D_2(t) + D_1(t)$$

$$= A_3(t) + D_3(t) + D_2(t) + D_1(t)$$

$$= \sum_k cA_3(k)\,\phi_{j-3,k}(t) + \sum_k cD_3(k) w_{j-3,k}(t) + \sum_k cD_2(k) w_{j-2,k}(t)$$

$$+ \sum_k cD_3(k) w_{j-3,k}(t) \tag{2.61}$$

The coefficients, $cA_m(k)$ and $cD_m(k)$ form $m = 1,2,3$ can be calculated by iterating or cascading the single stage filter bank to obtain a multiple stage filter bank.

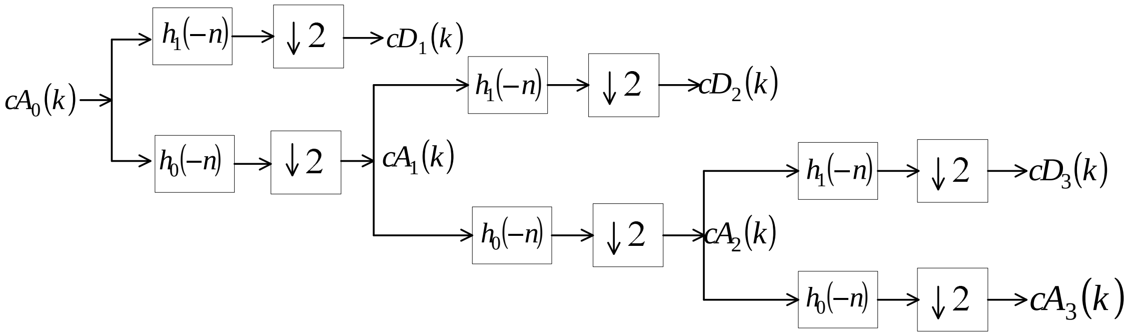


Figure 2.33 : Three levels of DWT analysis.

2.5.2 Synthesis: From Course Scale to Fine Scale

The decomposition of a signal into an approximation and a detail can be reversed. That is, we start with the two sets of coefficients $A_1(k)$ and $D_1(k)$ at scale index $j-1$ and produce the coefficients $A_0(k)$ at scale index j (Synthesis). We have:

$$x(t) = \sum_k cA_0(k)\phi_{j,k}(t)$$

$$= \sum_k cA_1(k)\phi_{j-1,k}(t) + \sum_k cD_1(k)w_{j-1,k}(t)$$

$$= A_1(t) + D_1(t) \tag{2.62}$$

Using the fact that $\phi_{j,n}(t)$ is an orthogonal basis for V_j we have:

$$cA_0(n) = \langle x(t), \phi_{j,n}(t)\rangle$$

$$= \langle \sum_k cA_1(k)\phi_{j-1,k}(t) + \sum_k cD_1(k)w_{j-1,k}(t)\phi_{j,n}(t)\rangle$$

$$= \sum_k cA_1(k)\langle\phi_{j-1,k}(t), \phi_{j,n}(t)\rangle + \sum_k cD_1(k)\langle w_{j-1,k}(t), \phi_{j,n}(t)\rangle$$

$$= \sum_k cA_1 h_0(n-2k) + \sum_k cD_1(k)\, h_1(n-2k) \tag{2.63}$$

This synthesis formula can be understood in terms of upsampling and filtering.

2.5.2.1 Upsampling and filtering

The expressions:

$$u(n) = \sum_k cA_1(k)h_1(n-2k) \tag{2.64}$$

$$v(n) = \sum_k cD_1(k)h_0(n-k) \tag{2.65}$$

look like convolutions but upsampling is involved. Upsampling of a discrete time signal $x(n)$ is performed by inserting zeros between the values. We can think about a system with input $x(n)$ and output $y(n) = x(n/2)$ for even values of n and $y(n)$ for odd values of n.

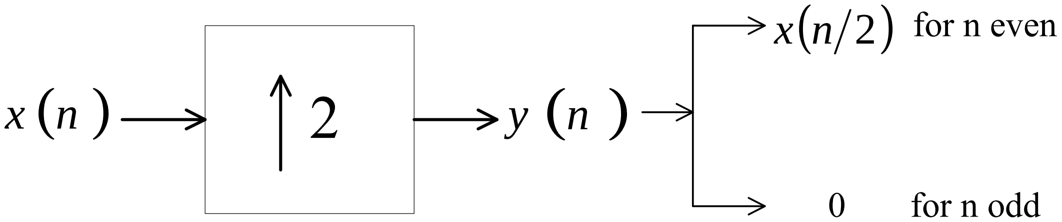


Figure 2.34 : Upsampler.

The expressions for $u(n)$ and $v(n)$ consist of upsampling followed by filtering.

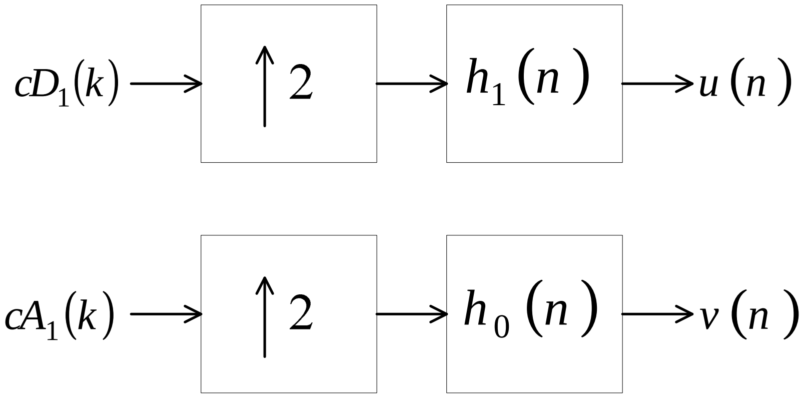


Figure 2.35 : Upsample and filter.

2.5.2.2 The one-stage synthesis filter bank

It follows that the synthesis formula consists of adding the outputs of the upsampled and filtered approximation and detail coefficients.

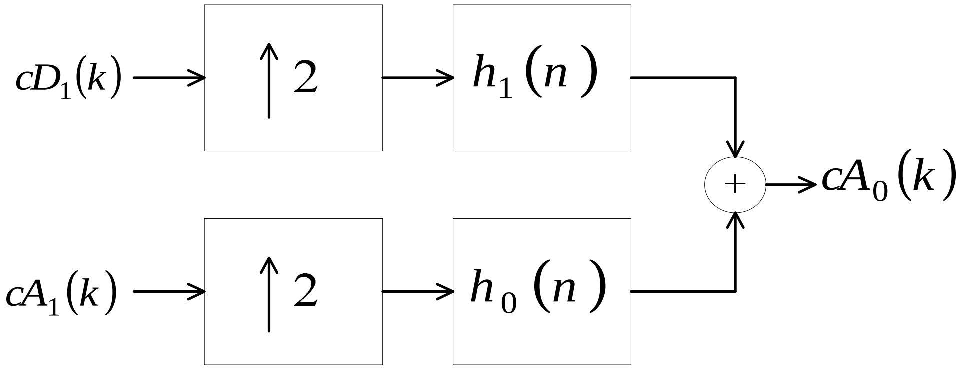


Figure 2.36 : One stage synthesis filter bank.

2.5.2.3 Perfect reconstruction filter bank

If we feed the output of the one-stage analysis filter bank to the input of the one-stage synthesis filter bank then we get the original coefficients back. We say that we have a perfect reconstruction filter bank.

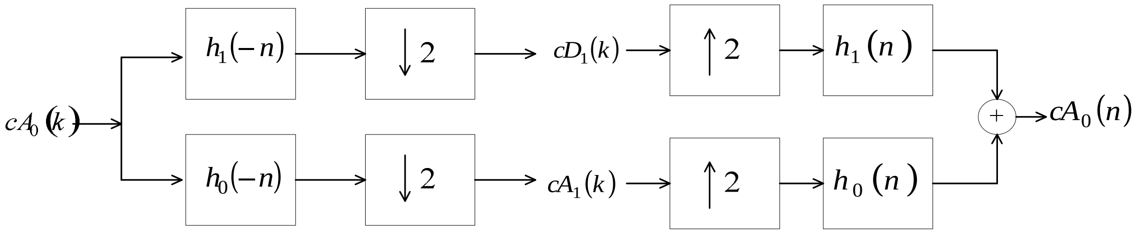


Figure 2.37 : Perfect reconstruction filter bank.

2.5.2.4 The synthesis filter bank

The outputs of the multiple stage analysis filter bank can be fed into a multiple stage synthesis filter bank to reproduce the original coefficients. For example, a 3 level analysis bank produces outputs $D_1(k)$, $cD_2(k)$, $cD_3(k)$, and $cA_3(k)$. These are fed into the 3 level synthesis filter bank as shown:

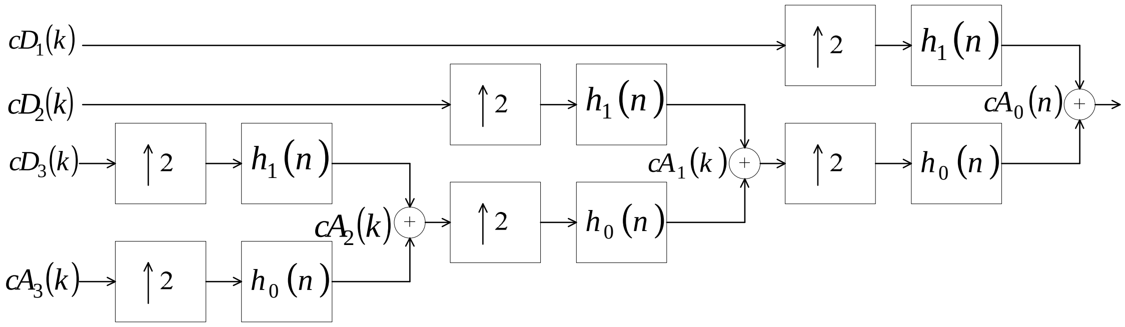


Figure 2.38 : Three stages synthesis filter bank.

2.5.2.5 Approximations and details

We have seen that we can reconstruct the signal $x(t)$ in V_j from the approximation and detail coefficients, $cA_1(k)$ and $cD_1(k)$ at level 1.

$$x(t) = \sum_k cA_0(k)\phi_{j,k}(t)$$

$$= \sum_k cA_1(k)\,\phi_{j-1,k}(t) + \sum_k cD_1(k) w_{j-1,k}(t)$$

$$= A_1(t) + D_1(t) \qquad (2.66)$$

As $A_1(t)$ and $D_1(t)$ are in V_j we can resolve them as:

$$A_1(t) = \sum_k cA_1(k)\,\phi_{j-1,k}(t)$$

$$= \sum_k A_1(k)\,\phi_{j,k}(t) \qquad (2.67)$$

$$D_1(t) = \sum_k cD_1(k) w_{j-1,k}(t)$$

$$= \sum_k D_1(k)\phi_{j,k}(t) \qquad (2.68)$$

Using the same reasoning as before:

$$A_1(n) = \sum_k cA_1(k) h_0(n-2k) \qquad (2.69)$$

$$D_1(n) = \sum_k cD_1(k) h_1(n-2k) \qquad (2.70)$$

That is, we obtain the approximation coefficients at level 0 by upsampling the approximation coefficients, $cA_1(k)$ at level 1 and then filtering with the low pass filter $h_0(k)$.

Similarly, we obtain the detail coefficients at level 0 by upsampling the detail coefficients, $cD_1(k)$ at level 1 and then filtering with the high pass filter $h_1(k)$.

2.6 HILBERT TRANSFORM

The Hilbert transform (HT) is a widely used frequency domain transform. It shifts the phase of positive frequency components by -90^0 and negative frequency components by +90^0. The Hilbert Transform of a given function $x(t)$ can be defined by the convolution between this function and the impulse response of the HT $(1/\pi t)$.

$$H[x(t)] = x(t) * \frac{1}{\pi t} \tag{2.71}$$

Since the Hilbert transformation is a convolution and does not change the domain, both $x(t)$ and $H[x(t)]$ are functions of time.

Specifically, if *X*(*f*) is the FT of *x*(*t*), its HT can be represented by $X_H(f)$ in frequency domain, where

$$X_H(f) = H[X(f)] = H_H(f)X(f) = \big(-jsgn(f)\big)X(f) \tag{2.72}$$

A $\pm 90°$ phase shift is equivalent to multiplying by $e^{\pm j90°} = \pm j$, so the transfer function of the HT $H_H(f)$ can be written as;

$$H_H(f) = -jsgn(f) = \begin{cases} -j, f > 0 \\ j, f < 0 \end{cases} \tag{2.73}$$

The corresponding impulse response is

$$h_H(n) = \begin{cases} 0, & n = 0 \\ \dfrac{2sin^2(\pi n/2)}{\pi n}, & n \neq 0 \end{cases} \tag{2.74}$$

Ideally, a HT with infinite number of coefficients has a flat frequency response. However, in practice numbers of coefficients are limited. Therefore the frequency response of a practical HT is bandlimited.

A HT with 30 coefficients has a frequency response given in figure 2.39. As it can be seen from the figure, transform has bandpass characteristics. Therefore, signals such as unit step function, which contains all the frequencies in it, cannot be analyzed correctly with HT.

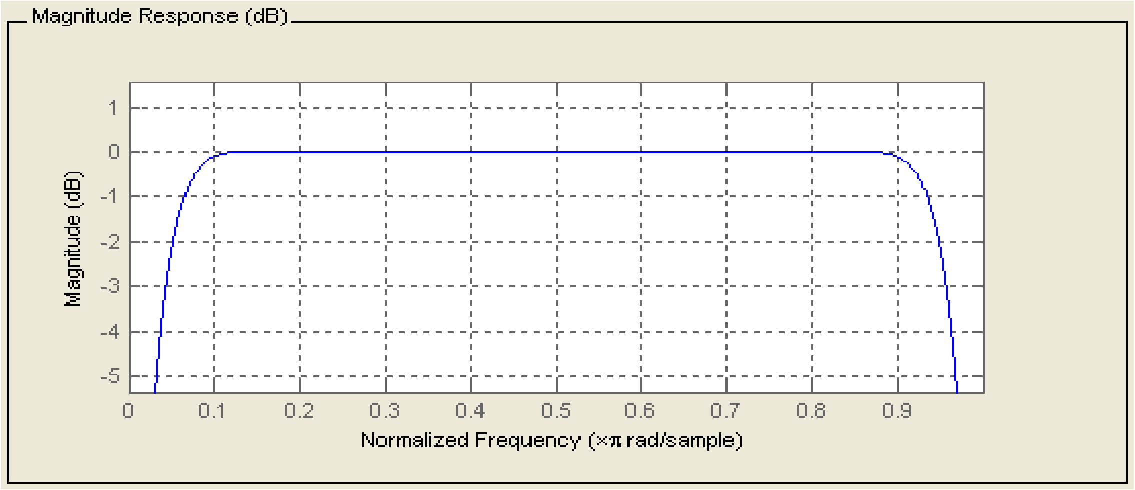


Figure 2.39 : Frequency response of Hilbert transform.

3. THE BASICS OF QUADRATURE DOPPLER SIGNALS

3.1 INTRODUCTION

The Doppler principle, which has many applications in communication and medicine, was first described in the nineteenth century. In medicine, the main usage of Doppler principle is for the study about blood flow. Doppler techniques have been widely used in areas such as cardiology, obstetrics and in general circulation studies. Many different types of commercial equipment are based on the Doppler ultrasound principle and they are widely available.

In Doppler ultrasound systems used in blood flow analysis, the incoming signal from an ultrasonic transducer is multiplied by the transmitted radio frequency signal and 90° phase shifted version of the transmitted signal. After low pass filtering, in-phase and quadrature-phase components of the audio Doppler signal are obtained. Flow direction is encoded in the phase relationship between in-phase and quadrature-phase channels.

Complex quadrature Doppler signals are obtained at the detection stage of the Doppler ultrasound systems employing quadrature demodulation technique. Output of most commercial Doppler ultrasound systems is in quadrature format. Quadrature Doppler signals are dual channel signals.

In this chapter, the basics of Doppler principle and Doppler ultrasound systems will be explained. Furthermore, complex quadrature Doppler signals, which are obtained from Doppler ultrasound systems, will be explained too. And finally, the phasing filtering technique which is used for the extraction of directional blood flow signals from these quadrature signals will be discussed.

3.2 PHYSICAL PRINCIPLE OF DOPPLER ULTRASOUND

Doppler ultrasound is based on the fact that any moving object in the path of a sound beam will shift the frequency of the transmitted signal. It can be shown that the difference between the transmitted frequency f_t and received frequency f_r is given by:

$$f_d = f_t - f_r = \frac{2vf_t cos\theta}{c} \tag{3.1}$$

where v is the velocity of the target, θ the angle between the ultrasound beam and the direction of the target's motion, and c the velocity of sound in the medium. The velocity and the transmitted frequency are known and the angle between the ultrasound beam and the direction of the target's motion can be determined. In this case, the velocity of the target can be found from the expression:

$$v = \frac{f_d c}{2f_t cos\theta} \tag{3.2}$$

Since the reflectors in a moving (flowing) media have different velocities, the Doppler shift signal contains a spectrum of frequencies which are within the audio range (0-20 kHz). The moving media is usually blood flow in clinical applications and Doppler studies are concentrated on interpreting the Doppler shift frequency spectra (Aydin 1994).

Detection of the returned (scattered) Doppler ultrasound signals is only made possible by employing a suitable electronic system. This requires a signal conversion process which is performed by an ultrasonic transducer. The next section introduces the basic principles of processing ultrasound Doppler signals.

3.2.1 Detection of Doppler Ultrasound Signals

Detection of Doppler ultrasound signals is a technical problem rather than a clinical one. It can be taken as a measurement problem and a general ultrasound Doppler signal measurement system can be modeled as in figure 3.1. This system can be divided into the three main parts: transduction, processing, interpretation and display (Aydin 1994).

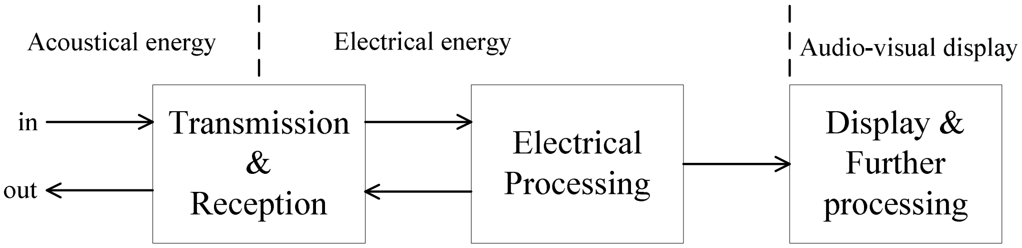


Figure 3.1 : A general Doppler ultrasound signal measurement system.

The transduction stage performs the energy conversion from electrical to acoustic energy and vice-versa. In general terms, a transducer is any device that converts energy in one form to energy in another. However, in its applied usage, the term refers to rather specialized devices. The majority either convert electrical energy to mechanical displacement or convert some nonelectrical physical quantity, such as temperature, sound, or light, to an electrical signal. Electro-acoustical transducers are used in the ultrasound systems (Aydin 1994).

The processing stage prepares the signal for transmission and/or processes the signal already converted to the electrical form by the transducer for display or further analysis. An example of this stage is the Doppler signal demodulator which is an electronic system which extracts the Doppler shifted signals from the returned signal. The last stage is mainly for the presentation and/or further analysis of the processed signals.

3.2.1.1 Demodulation of Doppler frequency shifted signals

One of the most important stages in a Doppler ultrasound system is demodulation of the Doppler frequency shifted signals which are generated in the transducer by the returning ultrasonic signals. Most of the demodulation techniques employed in communication systems are equally applicable to Doppler ultrasound systems. Most Doppler ultrasound systems employ quadrature demodulation techniques to detect signals and to preserve directional information.

3.2.1.2 Quadrature phase detection

A block diagram of quadrature phase detection is shown in figure 3.2. This is the most widely used detection method to preserve direction information.

The returning Doppler signal is demodulated using two reference signals having 90° of phase shift between them. The high frequency components are removed by low-pass filtering and the outputs are within the audio frequency range. This process produces a complex quadrature signal which is composed of the real and imaginary (or in-phase and quadrature-phase) components of the Doppler signal (Aydin 1994).

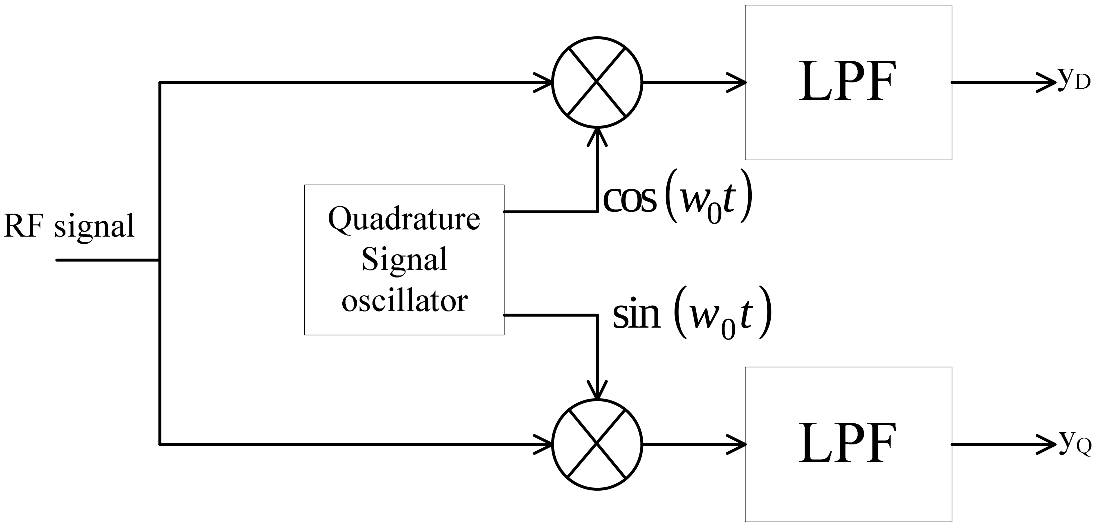


Figure 3.2 : Quadrature phase detection of the Doppler shift signals.

The direction information is encoded into the phase relationship between these components. One of the most important considerations in the design of quadrature phase detectors is the amplitude and phase balance of the quadrature carrier signals.

Ideally, the amplitudes of these signals must be identical and the phase difference between them must be 90°. Any gain and/or phase error in the carrier signals will appear at the quadrature outputs. This will lead to crosstalk artifact being generated in the directional outputs (Aydin 1994).

3.3 GENERAL DEFINITION OF COMPLEX QUADRATURE DOPPLER SIGNALS

Complex quadrature Doppler signals are obtained at the detection stage of the Doppler ultrasound systems employing quadrature demodulation technique. Outputs of most commercial Doppler ultrasound systems are in quadrature format. Quadrature Doppler signals are dual channel signals.

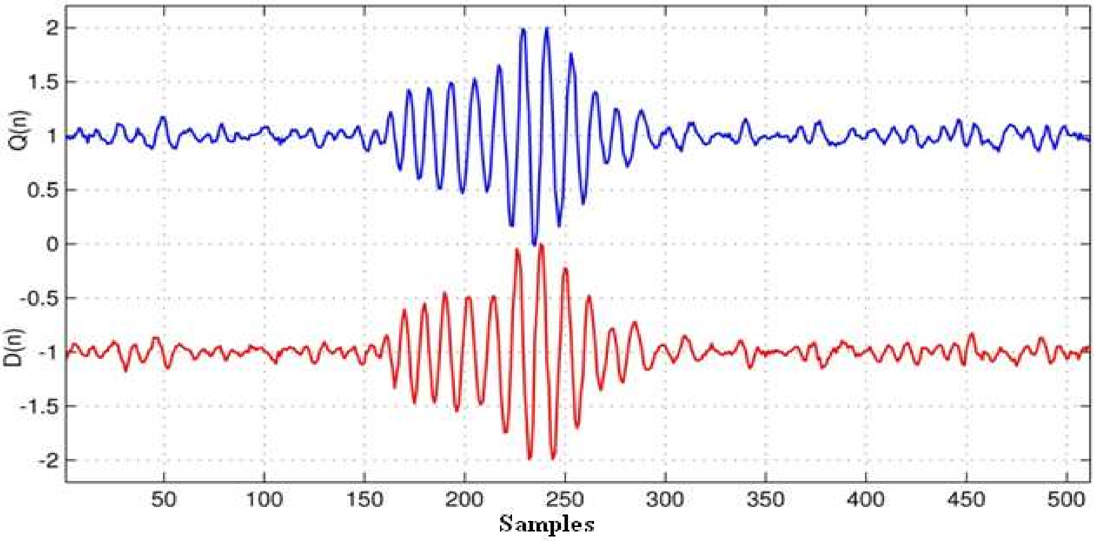


Figure 3.3 : Embolic quadrature Doppler signal pair.

A quadrature Doppler signal can be assumed as a complex signal, in which the real and imaginary parts can be represented as the HT of each other. This means there is a 90° phase difference between real and imaginary parts. An example of embolic quadrature Doppler signals can be seen in figure 3.3.

Mathematically, a discrete quadrature Doppler signal can be modeled as

$$y(n) = D(n) + jQ(n) \tag{3.3}$$

where $D(n)$ is in-phase and $Q(n)$ is quadrature-phase components of the signal. $D(n)$ and $Q(n)$ can also be represented in terms of the directional signals as

$$D(n) = \pm s_f(n) \pm H[s_r(n)] \tag{3.4}$$

$$Q(n) = \pm H\left[s_f(n)\right] \pm s_r(n) \tag{3.5}$$

where $s_f(n)$ and $s_r(n)$ represent forward and reverse signals respectively and H[ ] stands for the HT. The information concerning flow direction is encoded in the phase relationship between $D(n)$ and $Q(n)$. Although there are a number of methods for extracting directional signals from the quadrature signals, the phase filtering technique, which is based on HT, is most widely used method.

3.4 PHASE FILTERING TECHNIQUE

A block diagram of phase filtering technique (PFT) is shown in figure 3.4. The system is based on a wide-band digital HT which produces a 90° phase shift in the input signal. In this technique, HT is applied to the one part of the quadrature signal (real or imaginary), and this produces a 90° phase shift in the applied part of the signal. No application is done to the other part of the quadrature signal. And then, the output of the HT is added and subtracted with other part. After this addition and subtraction directional signals are obtained.

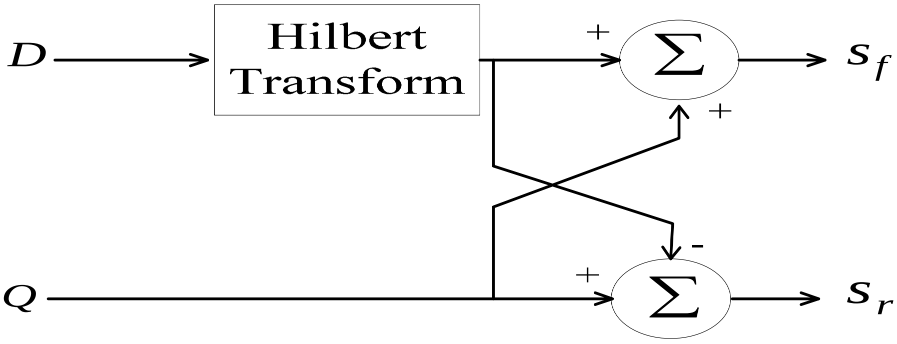


Figure 3.4 : Block diagram of phase filtering technique.

Consider a perfect quadrature detection system having ideal quadrature outputs

$$D = A\cos(w_a t) + B\sin(w_b t) \tag{3.6}$$

$$Q = A\sin(w_a t) + B\cos(w_b t) \tag{3.7}$$

where w_a represents the signal frequency due to flow in one direction and w_b represents the signal frequency due to flow in the other. If these signals are applied to the system shown in figure 3.2, separated outputs are obtained. From the properties of the HT, the output of the HT filter is

$$D_H = Asin(w_a t) - Bcos(w_b t) \tag{3.8}$$

which is the HT of the in-phase component of the input signal.

After addition and subtraction, the obtained results are

$$s_f = 2Asin(w_a t) \tag{3.9}$$

$$s_r = 2Bcos(w_b t) \tag{3.10}$$

In order to confirm these results a simulation can be implemented using Matlab program. Two complex waveforms were created having a sampling frequency of 4 kHz and 512 data points to simulate 100 Hz forward and 200 Hz reverse signals. The resulting signals are showed in figure 3.5, as you can see directional signals are obtained correctly.

The used signals are;

$$D = 10cos(2\pi n\, 100/4000) + 5sin(2\pi n\, 200/4000) \tag{3.11}$$

$$Q = 10sin(2\pi n\, 100/4000) + 5cos(2\pi n\, 200/4000) \tag{3.12}$$

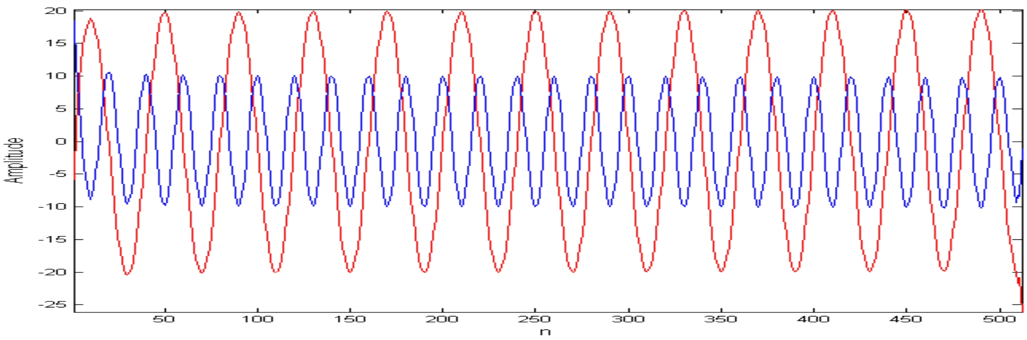


Figure 3.5: Directional signals with PFT. (Red – forward signal and blue – reverse signal)

4. PROPOSED METHOD: MODIFIED DUAL-TREE COMPLEX WAVELET TRANSFORM

4.1 INTRODUCTION

DWT is becoming a popular tool for analysis of non-stationary biological signals such as embolic quadrature Doppler signals. However, DWT does not map directional signals in the scale domain during analysis. Moreover, DWT is not a shift-invariance transform. Embolic directional signals, which are obtained from quadrature Doppler signals, are transient signals and in order to detect these transient parts more precisely, a new modified transform is needed. As a solution to this problem, a complex DWT algorithm called dual tree complex discrete wavelet transform was proposed in (Kingsbury 2001) (Selesnick, Baraniuk & Kingsbury 2005). However, it does not provide directional signal decoding during analysis. In this chapter, a modified dual tree complex wavelet transform capable of mapping directional signals at the transform output is presented.

4.2 Dual Tree Complex Wavelet Transform

DWT is an efficient way for processing quadrature Doppler signals, however quadrature Doppler signals are dual channel signals and for processing them firstly the directional signals must be extracted and then DWT must be performed to these directional signals separately. This situation doubles the computational complexity of whole process.

Additionally, DWT has two main drawbacks which can affect the process of quadrature Doppler signals. These drawbacks are;

- Lack of shift invariance, which means that small shifts in the input signal can cause major variations in the distribution of energy between DWT coefficients at different scales.

- Poor directional selectivity for diagonal features, because the wavelet filters are real.

The lack of shift invariance property of DWT is very important in processing of quadrature Doppler signals. For example the embolic signals, which are extracted from quadrature Doppler signals, are transient signals (short duration). To diagnose emboli in these signals, it is important to catch transient parts. Some examples of embolic signals can be seen in figure 4.1.

To overcome these two drawbacks; a new method, the Dual Tree Complex Wavelet Transform (DTCWT) was introduced with the following properties (Selesnick, Baraniuk & Kingsbury 2005):

- Approximate shift invariance: DTCWT has approximate shift invariance, or in other words, improved time shift sensitivity in comparison with standard DWT (Kingsbury 1999). In figure 4.2, 16 unit step functions with different phases are used as input signals for DWT and DTCWT. As it is seen, the coefficients in DTCWT are less affected.
- Good directional selectivity in M dimensions ($M \geq 2$)
- Perfect reconstruction: DTCWT structure follows perfect reconstruction conditions; hence, the original signal can be reconstructed from the transform domain complex wavelet coefficients.
- Limited redundancy: DTCWT has redundancy of 2:1 for one dimensional signals and 2^m: 1 for *m* dimensional signals.

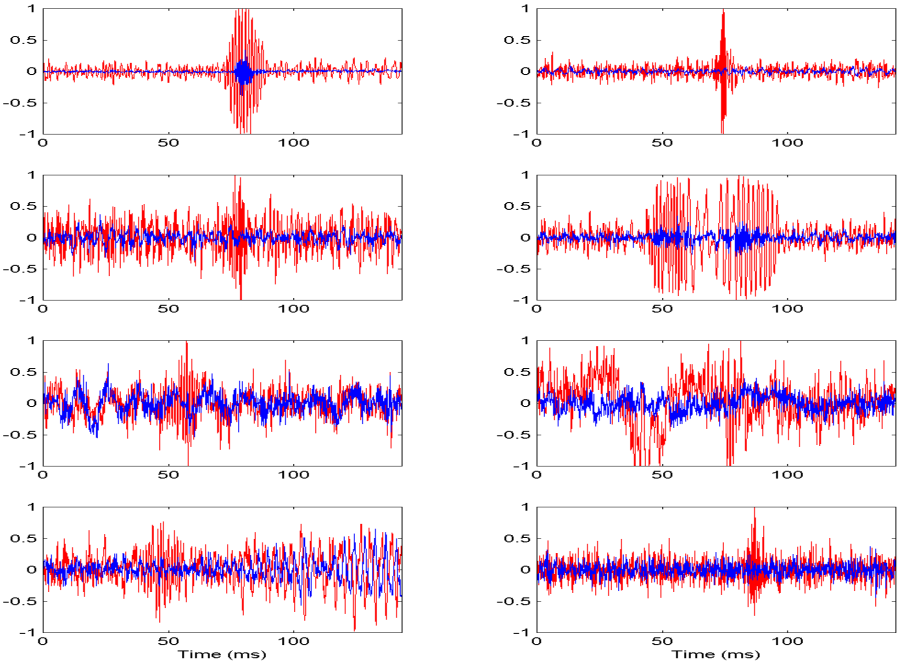
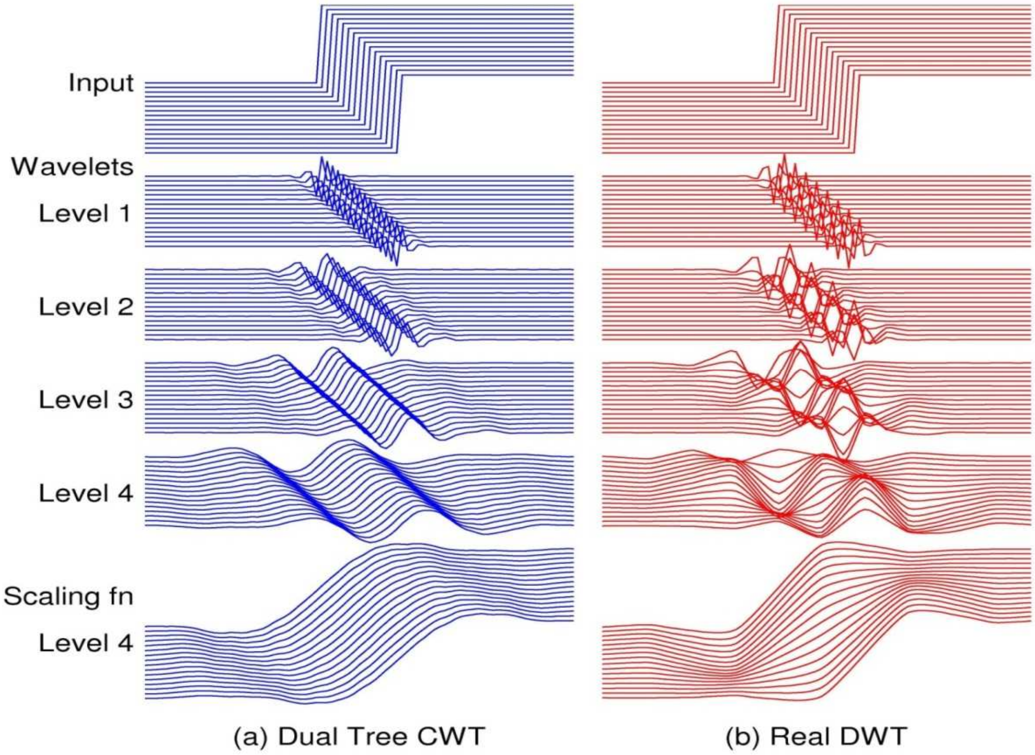


Figure 4.1 : Examples of embolic signals. (Forward signal – red, reverse signal - blue)



(a) Dual Tree CWT

(b) Real DWT

Figure 4.2 : Shift invariance of DTCWT.

4.2.1 Structure of DTCWT

The DTCWT employs two real DWTs; the first DWT can be thought as the real part of the transform while the second DWT can be thought as the imaginary part of the transform. The analysis and synthesis filterbanks used to implement the DTCWT and its inverse are illustrated in figures 4.3 and 4.4.

The two real wavelet transforms use two different sets of filters, with each satisfying the perfect reconstruction conditions. The two sets of filters are jointly designed so that the overall transform is approximately analytic. Let $h_0(n)$, $h_1(n)$ denote the low-pass/high-pass filter pair for the upper filterbank, and let $g_0(n)$, $g_1(n)$ denote the low-pass/high-pass filter pair for the lower filterbank. The two real wavelets associated with each of the two real wavelet transforms will be denoted as $w_h(t)$ and $w_g(t)$.

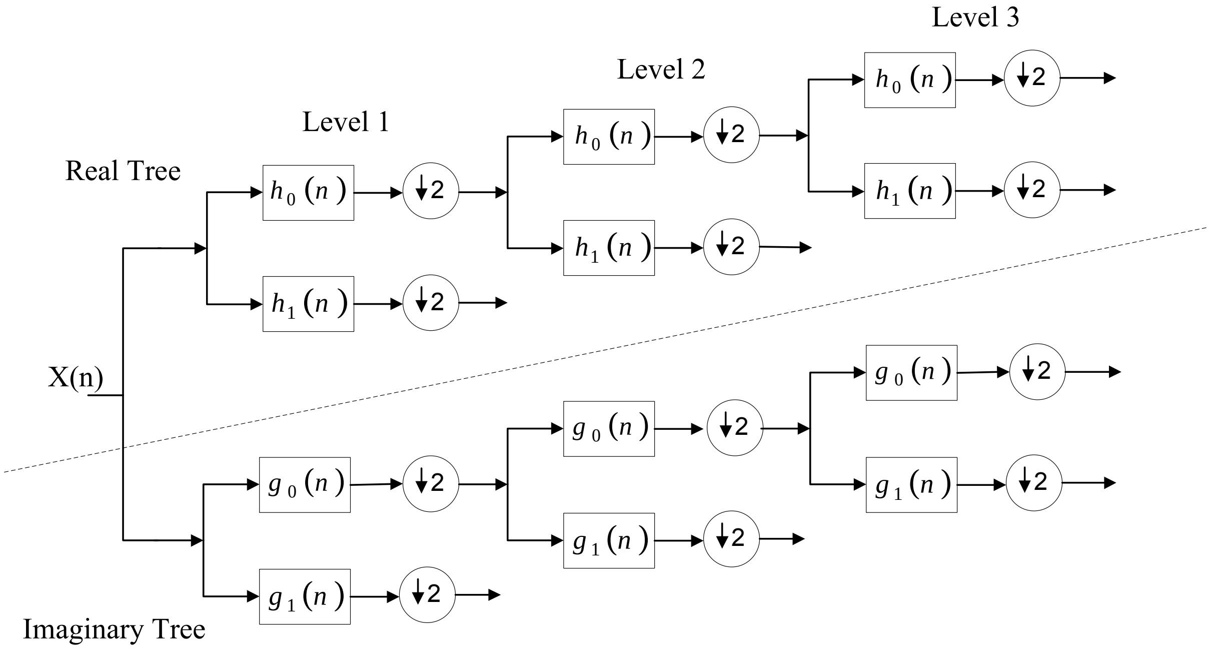


Figure 4.3 : Analysis filterbanks for the DTCWT.

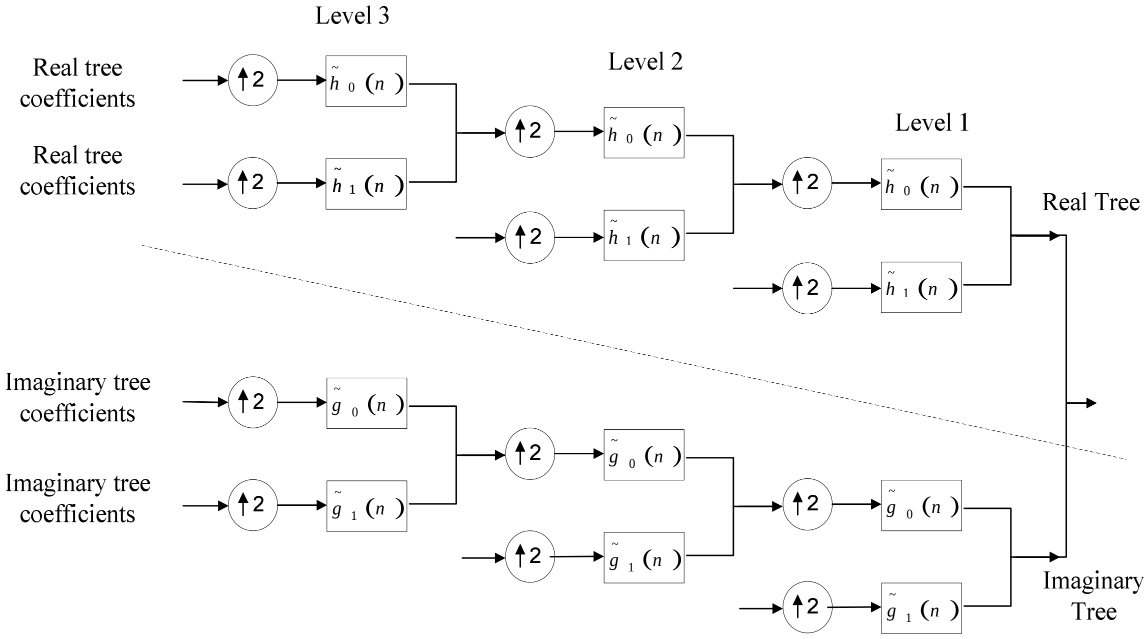


Figure 4.4 : Synthesis filterbanks for the DTCWT.

In addition to satisfying the perfect reconstruction conditions, the filters are designed so that the complex wavelet $w(t) := w_h(t) + jw_g(t)$ is approximately analytic (Selesnick 2002) (Selesnick 2001). Equivalently, they are designed so that $w_g(t)$ is approximately the HT of $w_h(t)$. The impulse response of level three, wavelet and scaling functions in time domain can be seen in figure 4.5. Also the frequency spectrum of complex wavelet, $w(t)$, for different levels can be seen in figure 4.6. As you can see from the figure, frequency spectrum is approximately one sided, which means approximately analytic.

Note that the filters are themselves real; no complex arithmetic is required for the implementation of the DTCWT. The inverse of the DTCWT is as simple as the forward transform. To invert the transform, the real part and the imaginary part are each inverted, the inverse of each of the two real DWTs are used, to obtain two real signals. These two real signals are then averaged to obtain the final output. Note that the original signal $x(n)$ can be recovered from either the real part or the imaginary part alone; however, such inverse DTCWTs do not capture all the advantages an analytic wavelet transform offers.

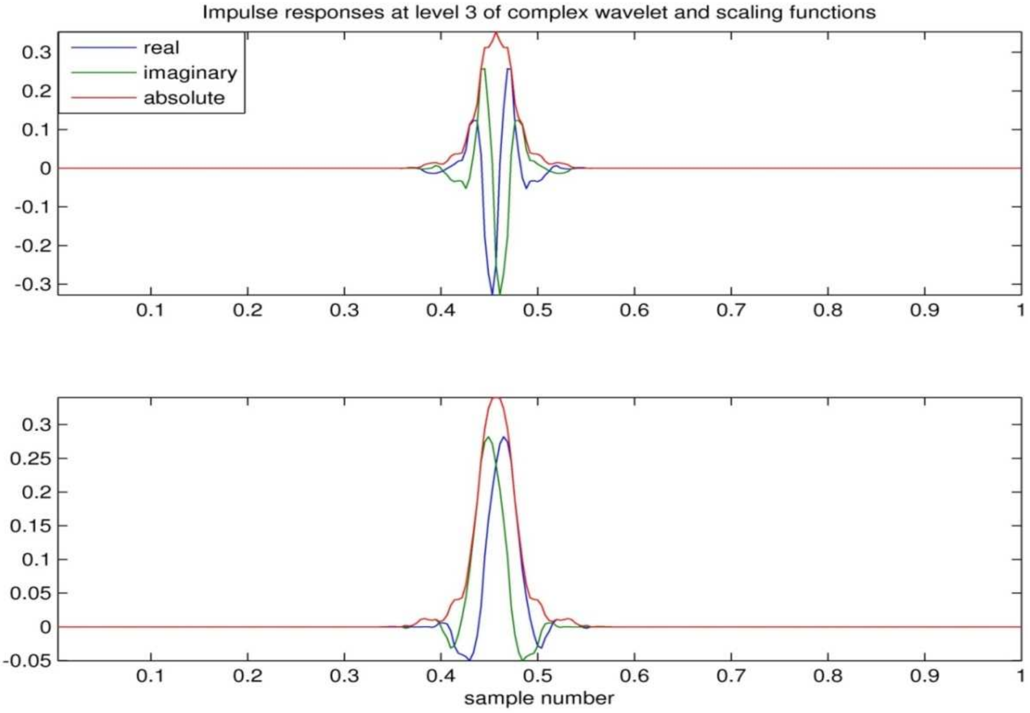


Figure 4.5 : Level 3 wavelet and scaling functions in time domain.

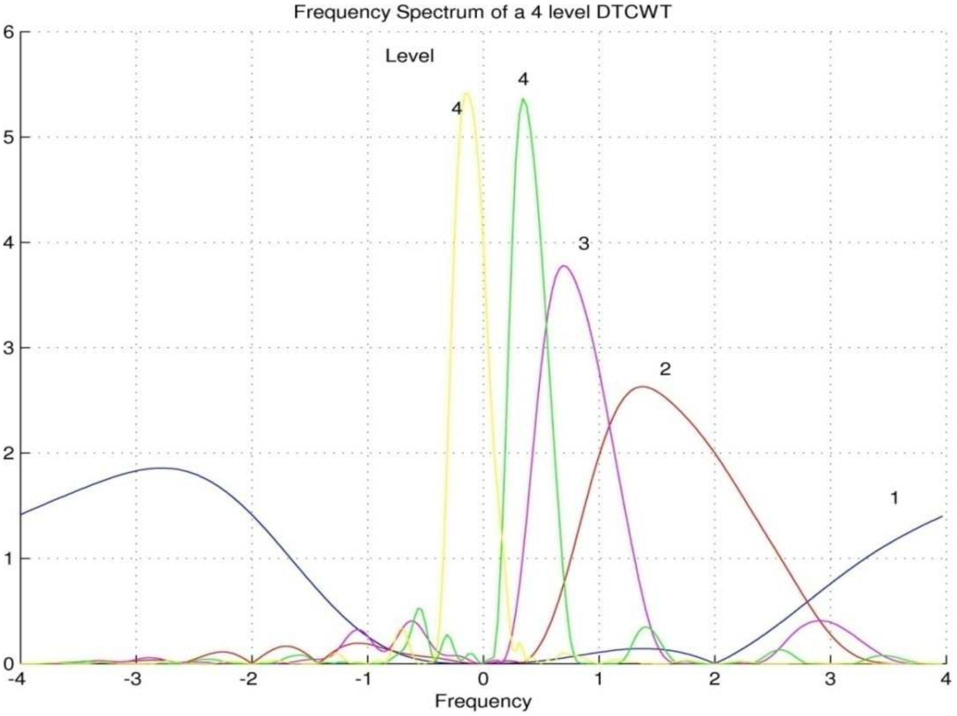


Figure 4.6 : Frequency spectrum of a 4 level DTCWT.

4.2.2 Processing of Quadrature Doppler Signals with DTCWT

As mentioned before, quadrature Doppler signals are dual channel signals and prior processing them, directional signals must be extracted with PFT. After the extraction of directional signals, DTCWT must be applied to these two directional signals. In the DTCWT, different processes can be applied to coefficients, such as de-noising, as illustrated in figure 4.7.

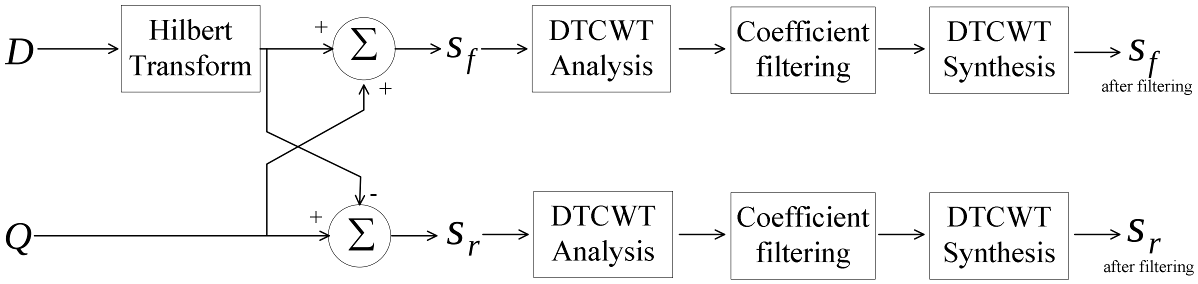


Figure 4.7 : Processing quadrature Doppler signals with DTCWT.

4.3 Proposed Method: Modified Dual Tree Complex Wavelet Transform

Conventionally, prior to applying the DTCWT to the quadrature Doppler signals, first it must be decoded into the directional signals and then two DTCWT algorithms should be applied to each signals. However, there exists an algorithm that results in a reduced computational complexity compared to conventional algorithm. This is attained by combining the part of the PFT with DTCWT as illustrated in the figure 4.8. In the conventional DTCWT transform, a real signal is applied to the both trees for decomposition and the outputs of the both reconstructed trees are added at the end of the reconstruction stage.

In the Modified Dual Tree Complex Wavelet Transform (MDTCWT), two modifications are made to the conventional DTCWT as illustrated in figure 4.8.

- At the analysis stage, instead of applying the complex quadrature signal to the both trees, the in-phase part is applied to the real tree through a Hilbert transformer introducing a 90° degree phase shift into the real part of the signal, and the quadrature-phase part is applied to the imaginary tree directly. The real and imaginary trees in this transform are the same as the conventional DTCWT.

- At the reconstruction stage, in addition to adding the outputs of reconstructed real and imaginary trees, which gives the signals caused by the flow signal in one direction, they are also subtracted resulting in the signals caused by the flow signal in the other direction.

The result is the same as the conventional PFT as described in part 3.4, and the mathematical proof of the MDTCWT would be the same as the PFT (Aydin, Fan & Evans 1994). The described algorithm is the equivalent to first applying the PFT to the quadrature signal and then taking two conventional DTCWTs, but with reduced computational complexity.

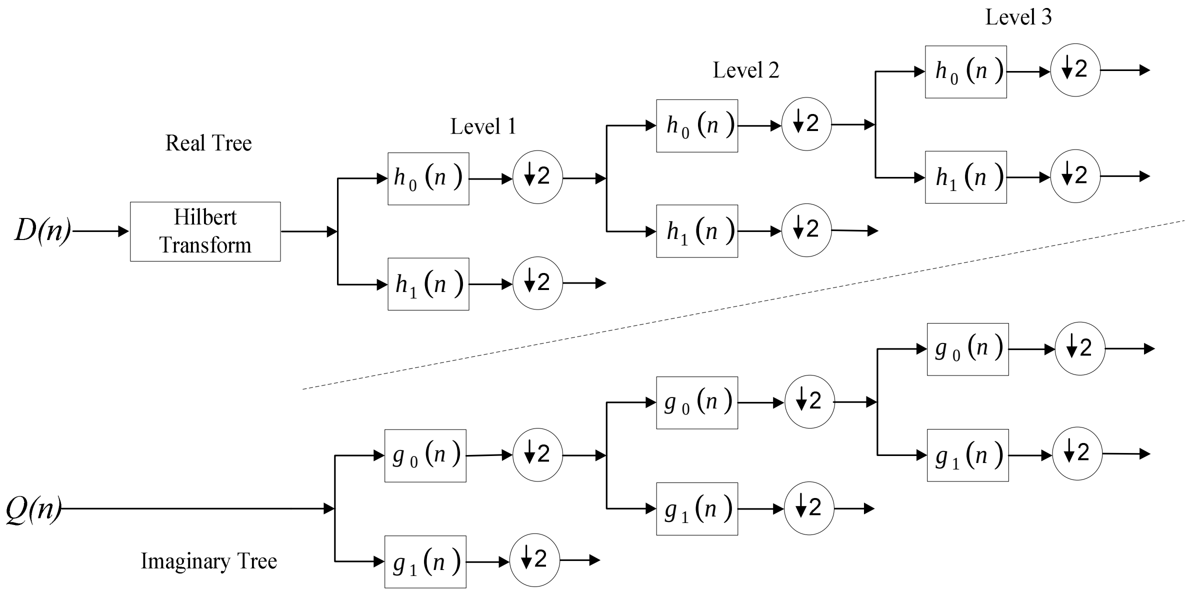


Figure 4.8 : Analysis stage of the MDTCWT algorithm for three levels.

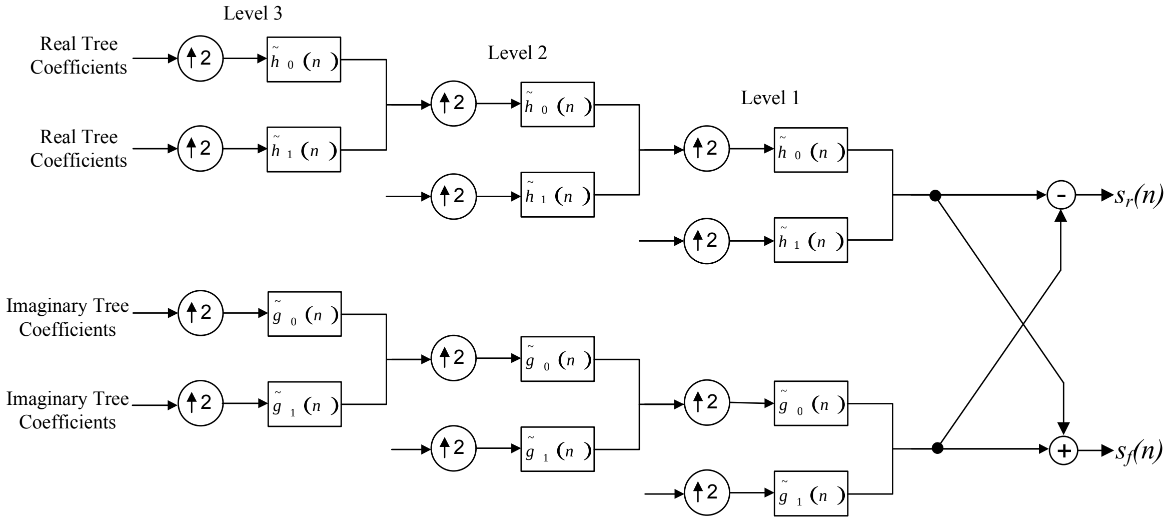


Figure 4.9 : Synthesis stage of the MDTCWT algorithm for three levels.

4.3.1 A Simulation Example using MDTCWT

In order to show that how the proposed algorithm works, a quadrature signal, which is created from sinusoidals, is used. Two complex waveforms are created having a sampling frequency of 40 kHz and 4096 data points to simulate 100 Hz forward and 200 Hz reverse signals. The resulting signals which are obtained with PFT and MDTCWT are showed in figure 4.10 and figure 4.11. The outputs of the both methods are almost the same.

The used signals are;

$$D = cos(2\pi n\, 100/40000) + 0.5sin(2\pi n\, 200/40000) \qquad (4.1)$$

$$Q = sin(2\pi n\, 100/40000) + 0.5cos(2\pi n\, 200/40000) \qquad (4.2)$$

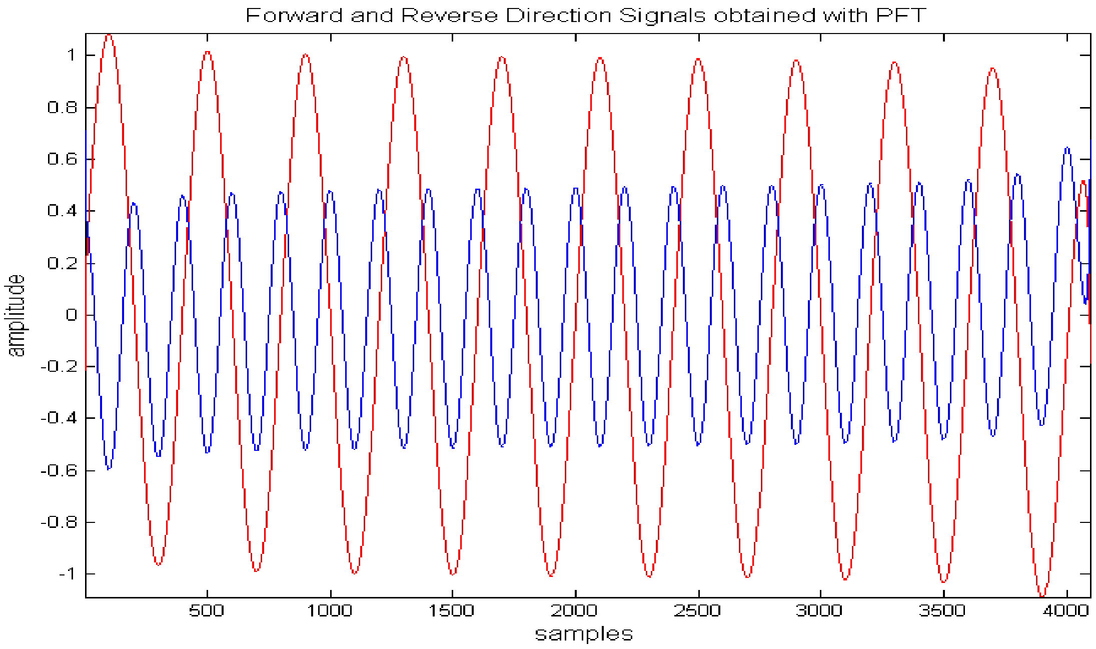


Figure 4.10 : Directional signals which are obtained with PFT.

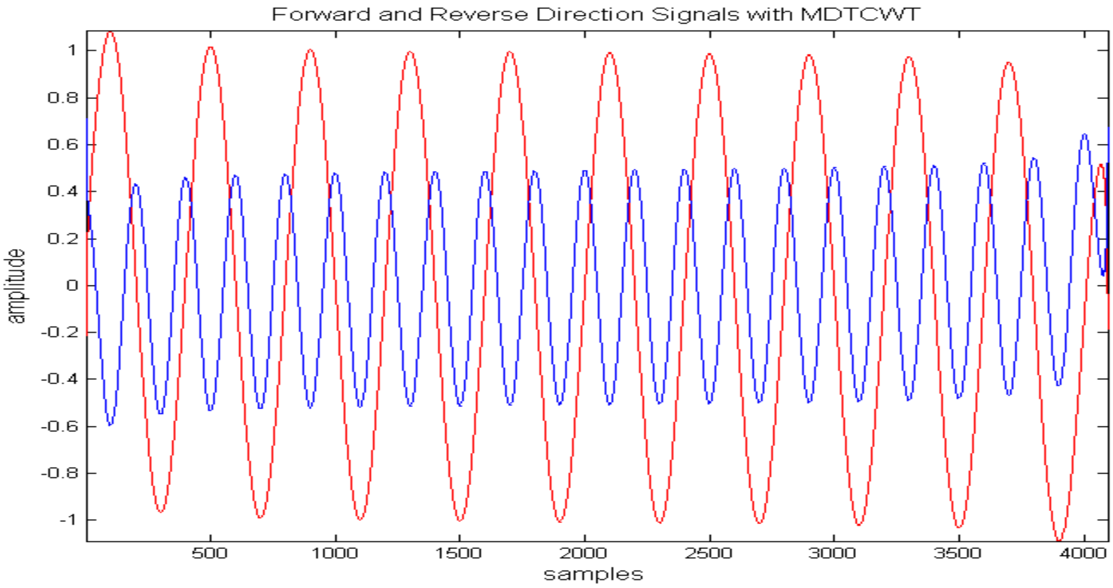


Figure 4.11 : Directional signals which are obtained with MDTCWT.

4.3.2 MDTCWT Coefficients

Different signal processing applications can be done using MDTCWT, such as de-noising. Generally, in order to implement these applications, coefficients of the transform must be used. Five levels reconstructed detail and approximation coefficients with MDTCWT for the simulation signal, which is used in part 4.3.1, can be seen in figure 4.12. As you can see from the figure, the directional signals appear in the fifth level approximation coefficients.

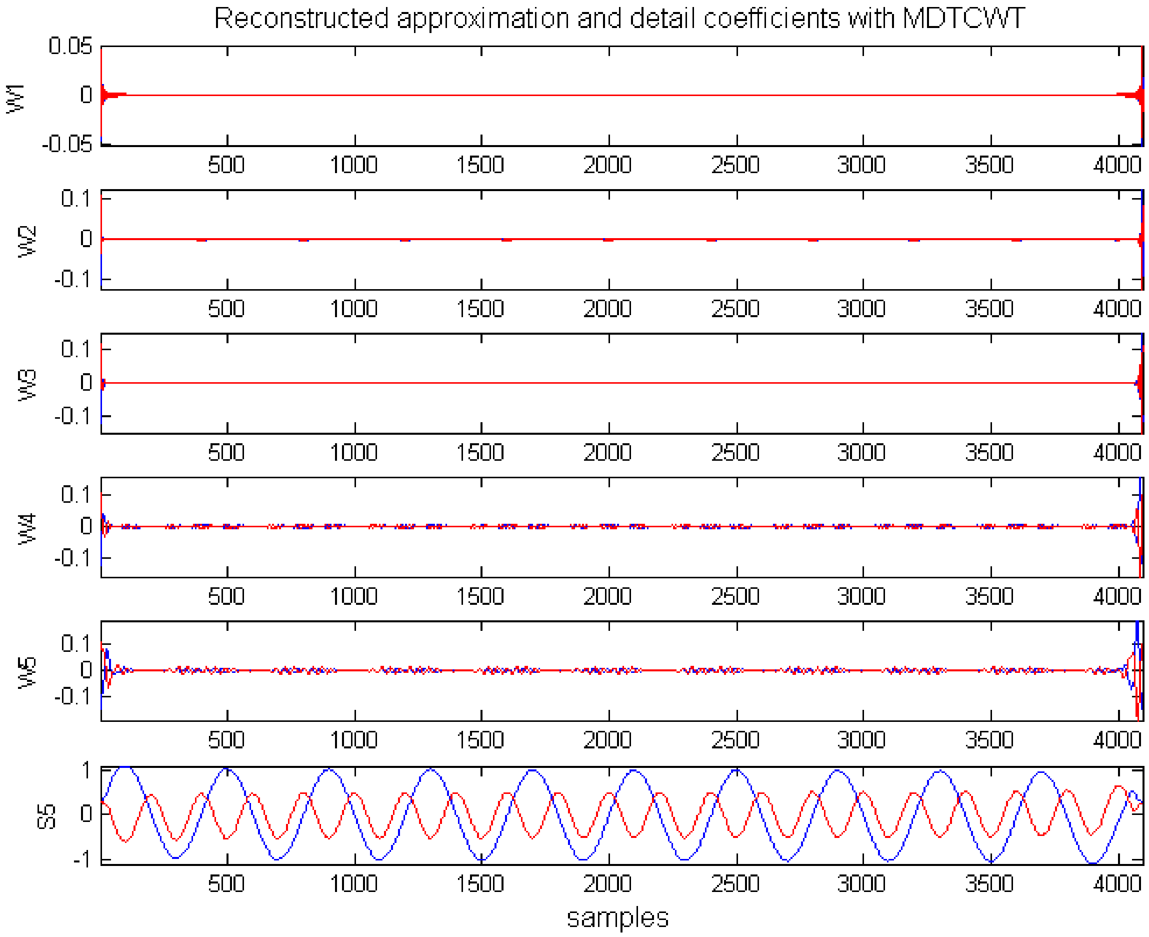


Figure 4.12 : Five levels reconstructed detail and approximation coefficients with MDTCWT.

4.3.3 Success of the Proposed Method

In order to show that the proposed algorithm works as intended, an embolic quadrature Doppler signal recorded from a patient was used. The sampling frequency was 7150 Hz and only 512 points were used. This quadrature signal is illustrated in figure 5.1(a). The signal was normalized to 1 and the in-phase and the quadrature-phase components of the signal were offset by 1 and -1 respectively for clarity. First, the forward and reverse signals were obtained by using the PFT to compare with. Then the same quadrature signal was decomposed to five levels and then reconstructed by using the MDTCWT resulting in the forward and the reverse signals.

Finally, both results were compared statistically by using the percent root mean square difference (PRD) for both forward and reverse signals.

$$PRD = \frac{\sqrt{\sum\left(x_i - x_j\right)^2}}{\sqrt{\sum\left(x_j\right)^2}} \times 100 \tag{4.3}$$

where x_j is the resulting directional signal obtained by the PFT and x_i is the resulting directional signal obtained by the MDTCWT. For an objective evaluation of the PRD, PRD values of 50 embolic Doppler signals with length of 4096 samples each were calculated and the final PRD value was obtained by averaging the 50 PRD values. The results of the PRD values can be seen in chapter 5.

4.3.4 Complexity of the Proposed Method

The computational complexity of the algorithm was also compared with the PFT followed by two real DWTs, and the PFT followed by two DTCWTs on a PC with Pentium M 1.86 GHz processor and 1 GB RAM. The algorithms were implemented in Matlab and tested using a quadrature Doppler signal having 1024 samples. In order to minimize effect of any computational time used by any program, which might be running at the background, each algorithm was run 1000 times and average execution time of the algorithms were calculated. The result of time comparison can be found in chapter 5.

4.3.5 Performance of the Proposed Method: De-Noising with MDTCWT

Many scientific experiments result in signals corrupted with noise, either because of the data acquisition process, or because of environmental effects. A first pre-processing step in analyzing such signals is denoising, that is, estimating the unknown signal of interest from the available noisy data. There are several different approaches to denoise one dimensional signals and images.

Thresholding is a widely used technique for signal and image denoising. The discrete wavelet transform uses two types of filters: (1) averaging filters (Low-pass), and (2) detail filters (High-pass). When a signal is decomposed using the wavelet transform, we are left with a set of wavelet coefficients that correlates to the high frequency subbands. These high frequency subbands consist of the details in the signal. If these details are small enough, they might be omitted without substantially affecting the main features of the signal. Additionally, these small details are often those associated with noise; therefore, by setting these coefficients to zero, we are essentially killing the noise. This becomes the basic concept behind thresholding-set all frequency subband coefficients that are less than a particular threshold to zero and use these coefficients in an inverse wavelet transformation to reconstruct the data set (Wagner 2004).

In this part, the de-noising performance of the proposed algorithm was compared with DWT and DTCWT. In order to measure the performance, a simulation signal, in quadrature format, with noise was constructed in Matlab program and de-noising process was implemented for all three methods, MDTCWT, DWT and DTCWT.

4.3.5.1 Signal and noise model

Signal and noise model for quadrature Doppler signal simulation can be given as:

$$Y_{noised}(n) = D_{noised}(n) + jQ_{noised}(n) \qquad n = 1 \; to \; N \tag{4.4}$$

$$D_{noised}(n) = D(n) + g(n) \tag{4.5}$$

$$Q_{noised}(n) = Q(n) + g(n) \tag{4.6}$$

where, Y_{noised} is an N point noisy quadrature Doppler signal corrupted by Gaussian noise g(n).

4.3.5.2 Soft thresholding

In denoising the noisy simulation signal, a denoising method, knows as soft thresholding, is applied to the wavelet coefficients though all scales and subbands. The soft thresholding method sets coefficients with values less than the threshold T to 0, then subtracts T from the non-zero coefficients. After performing soft thresholding, we take the inverse wavelet transform of the new wavelet coefficients.

4.3.5.3 Structure of the de-noising algorithms for MDTCWT, DWT and DTCWT

To make the comparison, firstly we first took the forward MDTCWT over five scales. Then we applied soft thresholding method to first five wavelet coefficients and to fifth scaling coefficients. Afterwards, we took inverse transform and reconstructed the de-noised signal with MDTCWT. The process can be seen in figure 4.13.

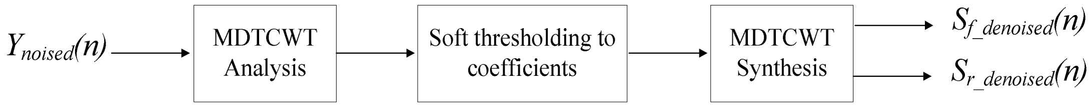


Figure 4.13 : De-noising with MDTCW.

As you can see from the figure above, we obtained de-noised directional signals at the output of MDTCWT.

Secondly, noisy directional signals were obtained with PFT and then de-noising with DWT was applied to these directional signals.

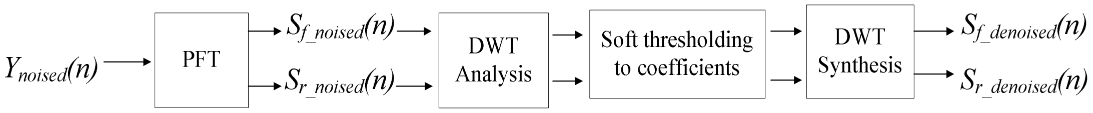


Figure 4.14 : De-noising with DWT.

Thirdly, noisy directional signals were obtained with PFT and then de-noising with DTCWT was applied to these directional signals.

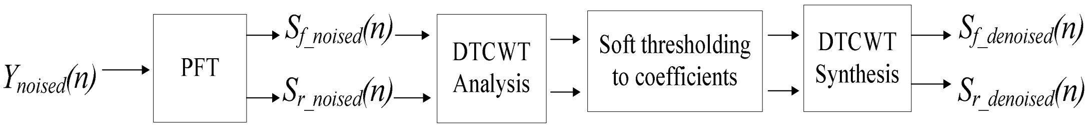


Figure 4.15 : De-noising with DTCWT.

A simulation example and its results can be seen in chapter 5.

5. RESULTS, CONCLUSIONS AND FUTURE SCOPE

5.1 RESULTS

In this part of the thesis, firstly the results of simulation and real-world examples which are presented in chapter 4 will be given. These results will show the success, complexity and performance of the proposed method.

5.1.1 Success of the Proposed Method

In part 4.3.3, it is mentioned that the success of the proposed method is compared with the outputs of PFT. The results of proposed method and PFT can be seen in figure 5.1.

The signals representing forward (red line) and reverse (blue line) flow components of the embolic Doppler signal, which are obtained by using the MDTCWT and the PFT are shown in figures 5.1(b) and 5.1(c) respectively. The error signals obtained by subtracting the signals in figure 5.1(c) from the signals in figure 5.1(b) are illustrated in figures 5.1(d) and 5.1(e) respectively. It is remarkable that the difference signals for both forward and reverse flow signals are around -90 dB, indicating that the algorithm works as exactly intended.

In addition table 5.1 shows the results of the PFT and the MDTCWT comparisons using the PRD. The average PRDs for the reverse flow signals (2.65×10^{-8}) and the forward flow signals (5.69×10^{-8}) are extremely small and negligible. Therefore the outputs of the both algorithms can be assumed the same. It is obvious that these results are in good correlation with the qualitative results shown in the Figure 5.1.

Table 5.1 : The PRD values for the forward and reverse flow signals between the PFT and MDTCWT.

	The Error (PRD)
Forward Output Signal	5.69×10^{-8}
Reverse Output Signal	2.65×10^{-8}

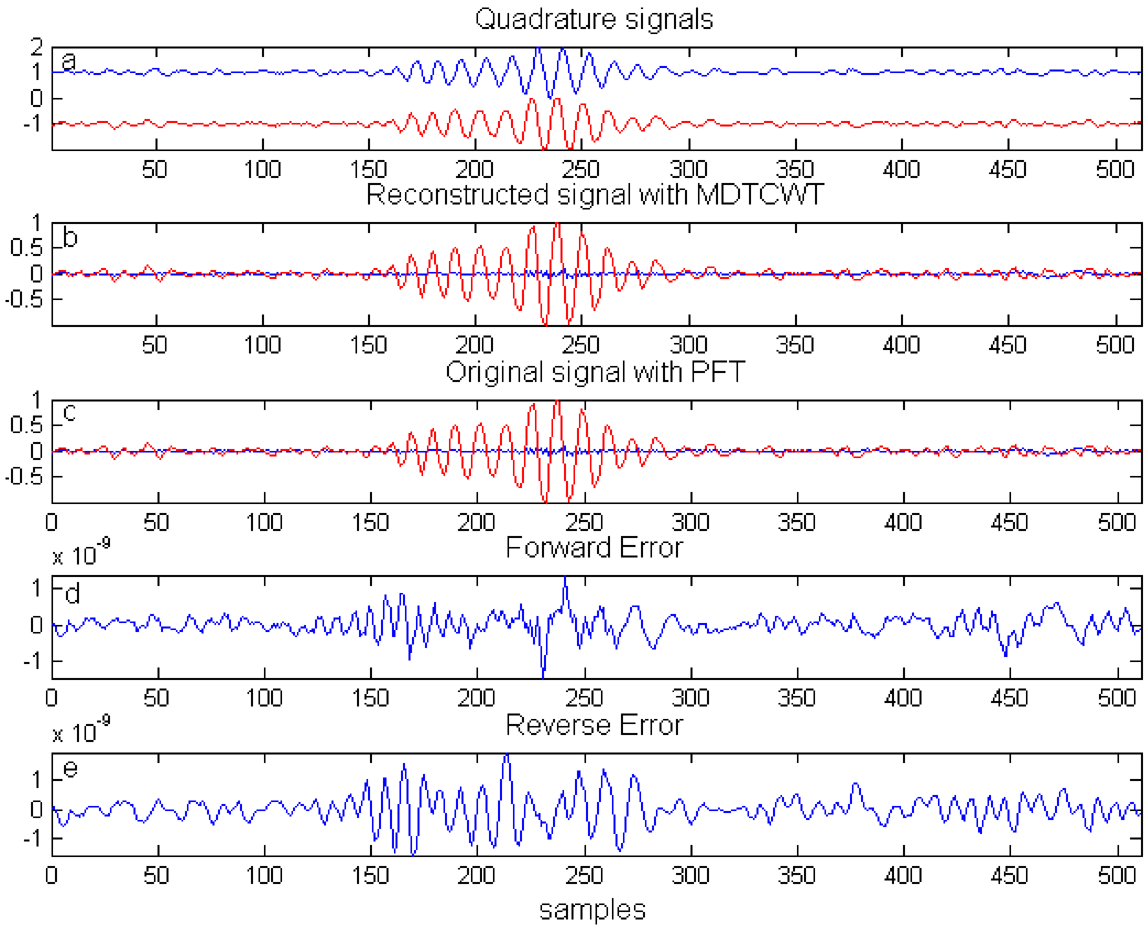


Figure 5.1 : (a) A quadrature embolic Doppler signal, (b) the forward (red line) and the reverse (blue line) outputs using the MDTCWT, (c) the forward (blue line) and the reverse (red line) outputs using the PFT, and corresponding differences of (d) the forward an and (e) the reverse signals obtained by the MDTCWT and the PFT.

5.1.2 Complexity of the Proposed Method

As mentioned in part 4.3.4 the processing times indicating the computational complexities of three methods (the PFT with two DWT, the PFT with two DTCWT, and the MDTCWT) are shown in table 5.2. Computational cost of the proposed algorithm (9.1 ms) is almost same as the PFT algorithm followed by two DWTs (9.0 ms) and half of the PFT algorithm followed by two DTCWTs (18.1 ms).

Table 5.2 : Comparison of the processing times for the PFT with DWT, the PFT with DTCWT and the MDTCWT .

Method:	Processing time (ms)
PFT with DWT	9.0
PFT with DTCWT	18.1
MDTCWT	9.1

5.1.3 Performance of the Proposed Method

As mentioned in part 4.3.5, in order to measure the performance of the proposed method, de-noising application was performed. For the simulation the used signals are;

$$D(n) = cos(2\pi n\, 100/40000) + 0.5sin(2\pi n\, 200/40000) \qquad (5.1)$$

$$Q(n) = sin(2\pi n\, 100/40000) + 0.5cos(2\pi n\, 200/40000) \qquad (5.2)$$

In order to corrupt these signals Gaussian noise was added with the maximum amplitude 0.20, and corrupted signals were de-noised with MDTCWT, DWT and DTCWT respectively. Noised directional signals and original directional signals can be seen in figure 5.2.

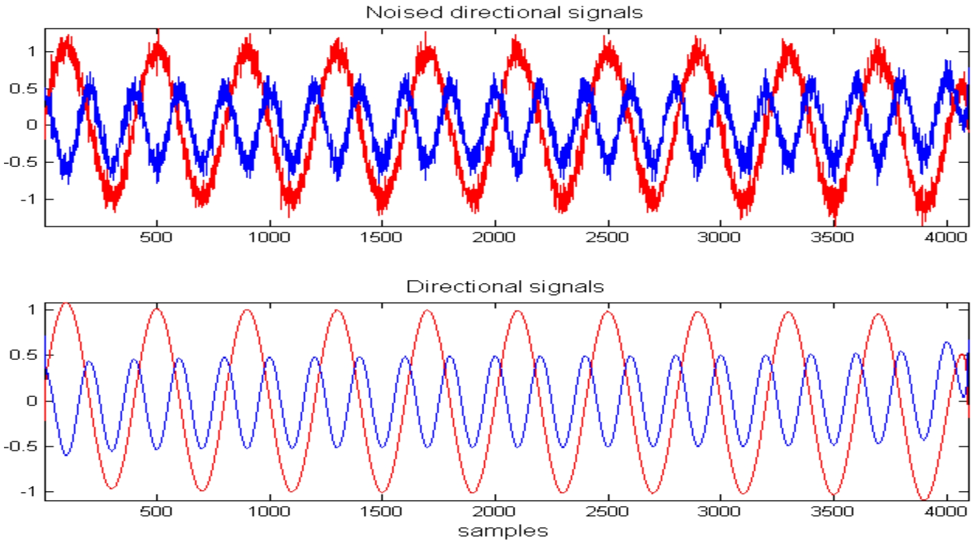


Figure 5.2 : Noised directional signals and normal directional signals.

For the de-noising process firstly noised signals were decomposed to five levels with three methods and soft thresholding was applied to these subbands. Then the de-noised subbands were reconstructed with three methods. The reconstructed subbands for the noised and de-noised signals can be seen in figure 5.3, 5.4, 5.5, 5.6, 5.7, 5.8.

In figures 5.6, 5.7 and 5.8, we use 0.2 for the threshold value. And in figure 5.9, the reconstructed signals for three methods can be seen together.

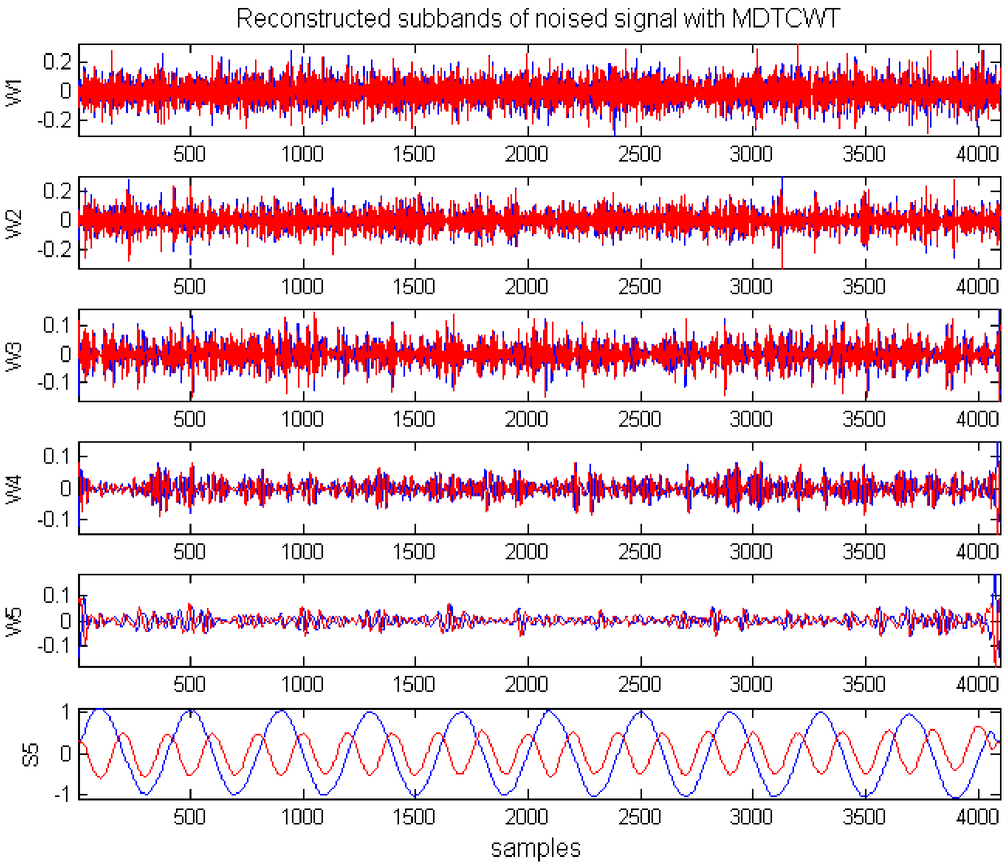


Figure 5.3 : Reconstructed subbands of noised signal with MDTCWT.

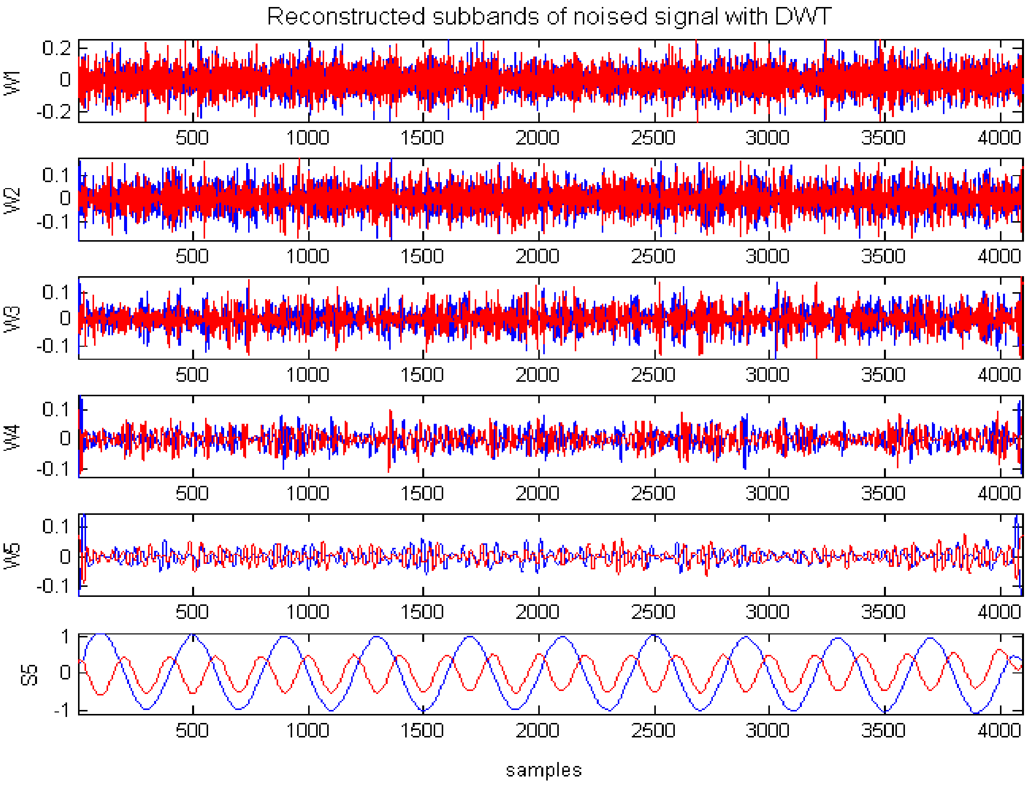


Figure 5.4 : Reconstructed subbands of noised signal with DWT.

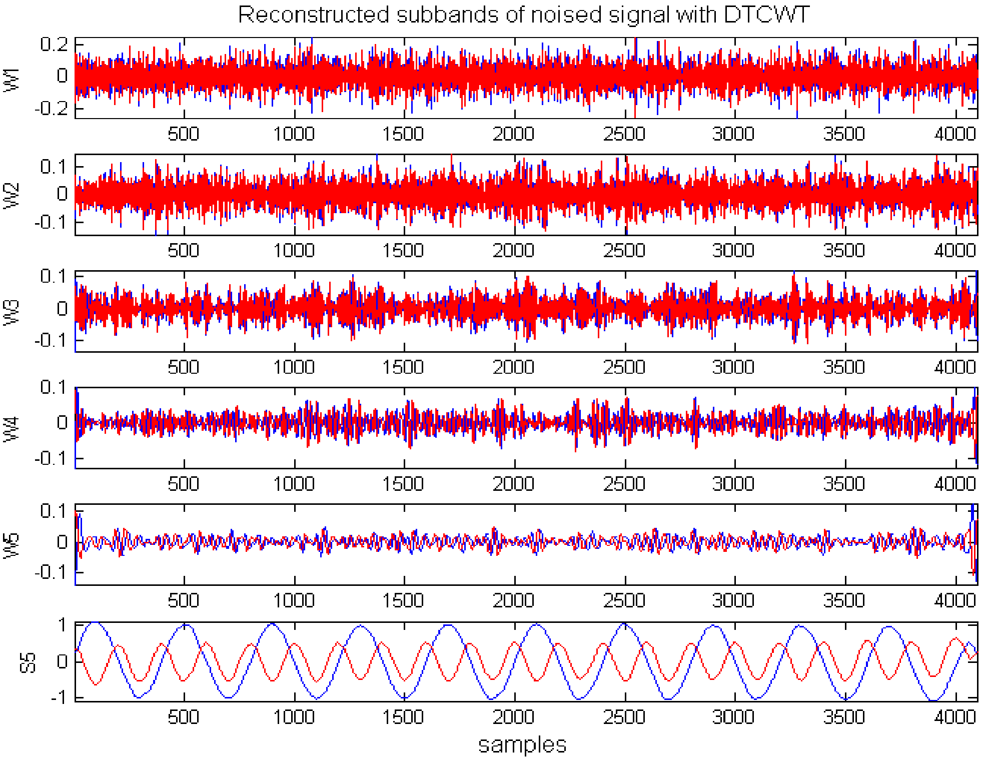


Figure 5.5 : Reconstructed subbands of noised signal with DTCWT.

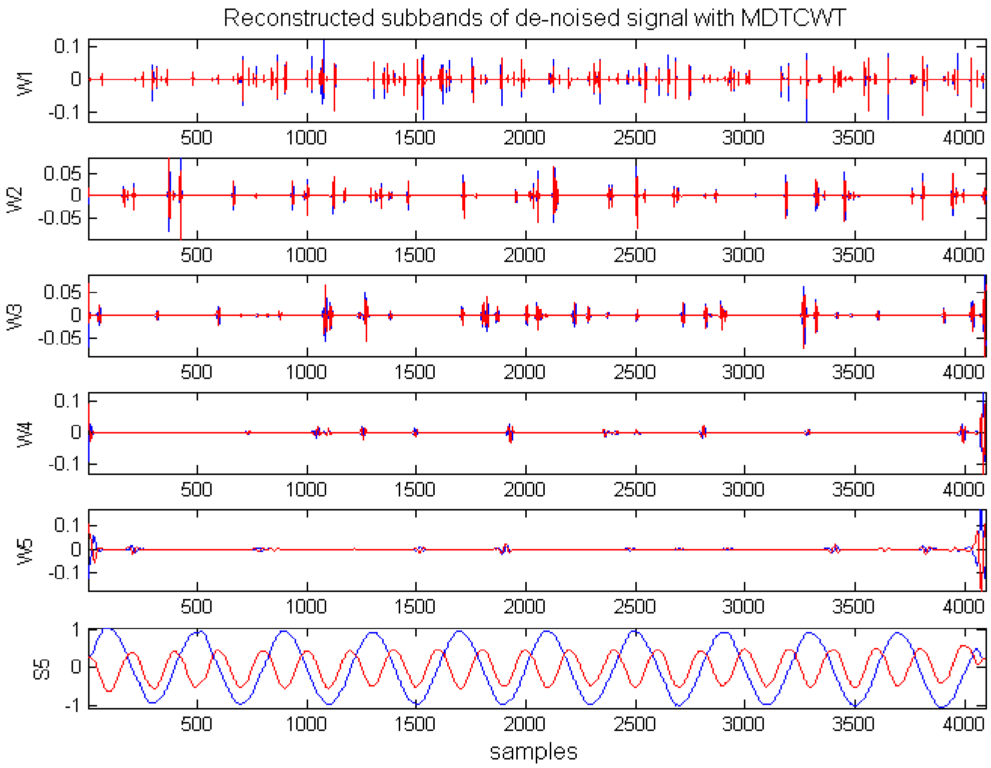


Figure 5.6 : Reconstructed subbands of de-noised signal with MDTCWT.

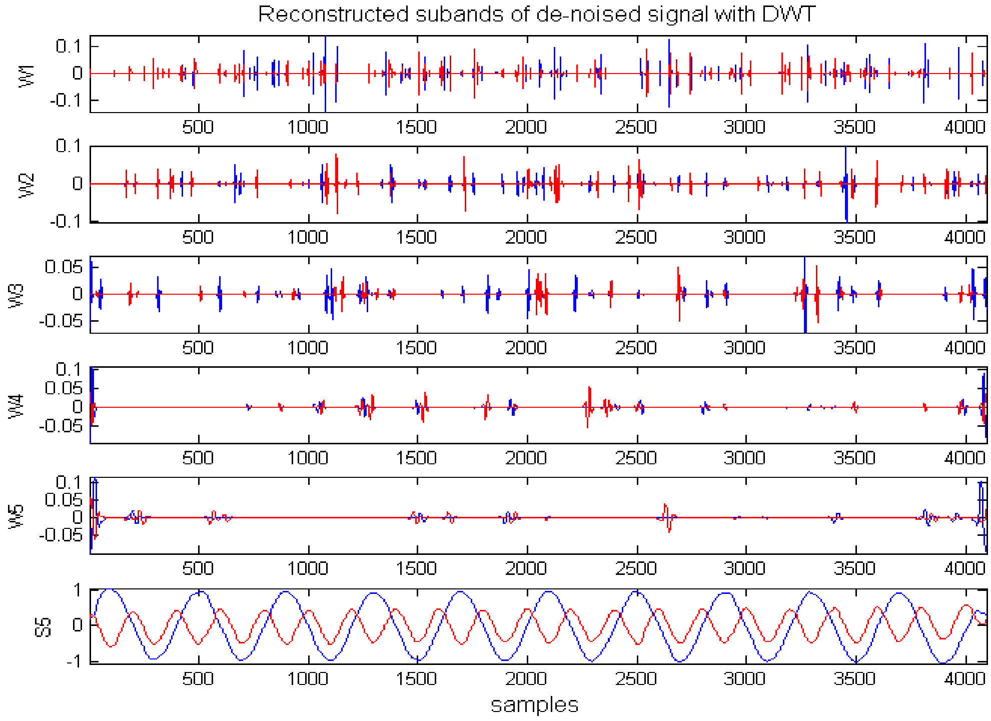


Figure 5.7 : Reconstructed subbands of de-noised signal with DWT.

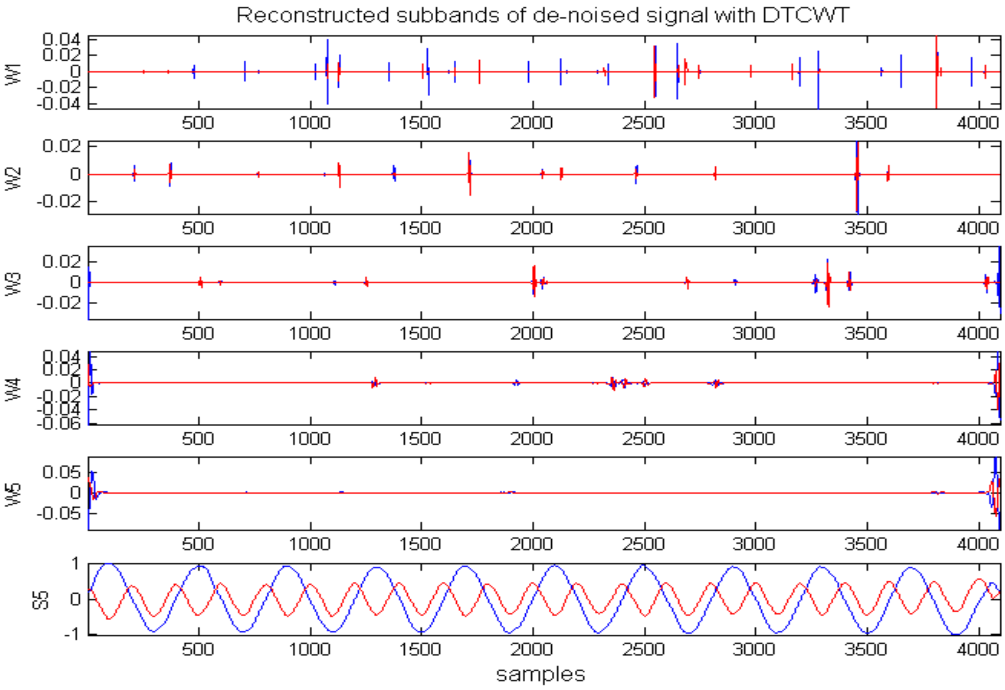


Figure 5.8 : Reconstructed subbands of de-noised signal with DTCWT.

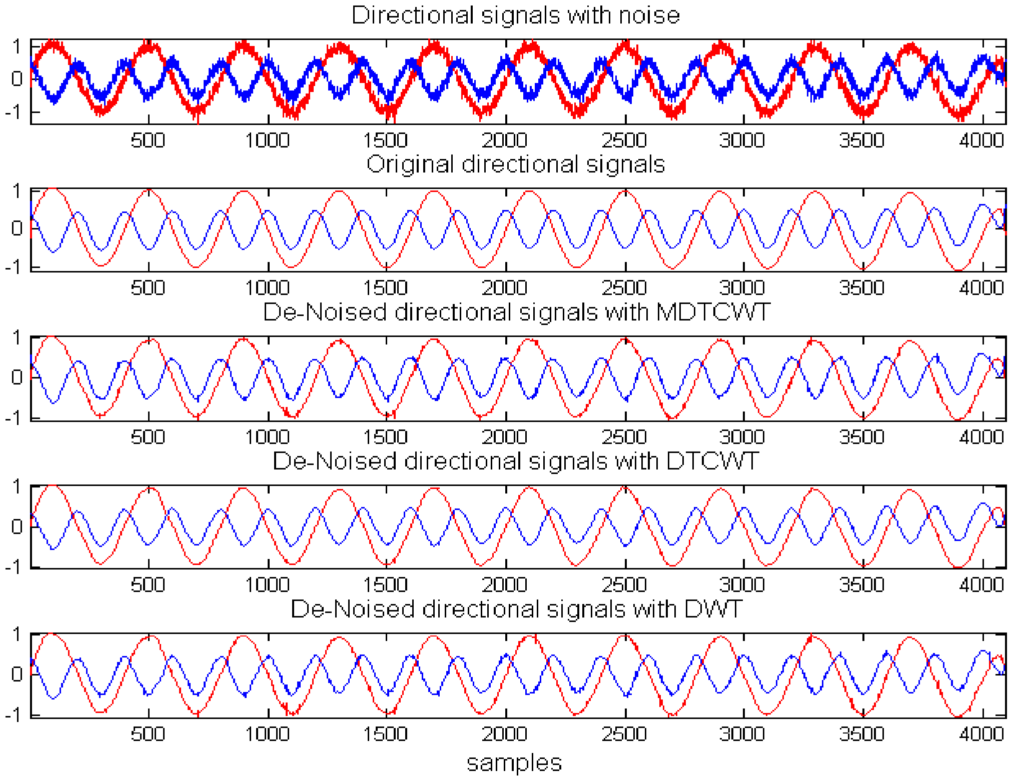


Figure 5.9 : De-Noised directional signals with three methods.

In order to compare the performance of the proposed method numerically, the difference between the original PFT output and reconstructed outputs of the MDTCWT, DWT, and DTCWT are computed as Root Mean Square Error (RMS). For the comparison, a simulation signal is used in the below format;

$$D(n) = A\cos\left(2\pi n\, f_{forward}/40000\right) + A\sin(2\pi n\, f_{reverse}/40000) \tag{5.3}$$

$$Q(n) = A\sin\left(2\pi n\, f_{forward}/40000\right) + A\cos(2\pi n\, f_{reverse}/40000) \tag{5.4}$$

In the simulation example in order to measure performance of the proposed method fairly, we choose the amplitudes and frequencies of direction signals equal.

From figure 5.10 you can see the RMS error for all three methods. We use 1 for the amplitudes and 200 Hz for the frequencies.

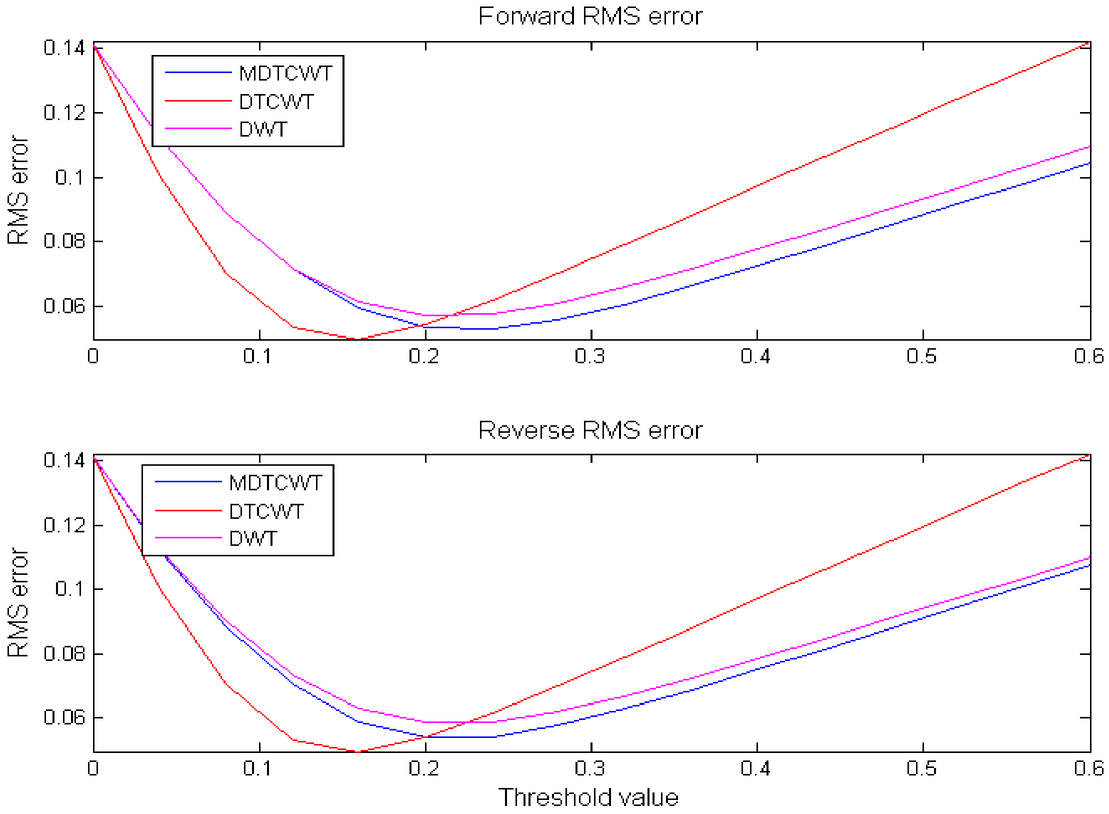


Figure 5.10 : RMS error for MDTCWT, DWT and DTCWT.

As you can see from the above figure the proposed method, MDTCWT, has better de-noising performance in both directions than DWT. The MDTCWT method can reduce the noise of a signal from 0.2 to about 0.053, whereas the DWT method can reduce the noise of a signal from 0.2 to about 0.058. This is a notable improvement. And this shows us that proposed method has same computational complexity with DWT and better de-noising performance than DWT.

5.2 CONCLUSION AND FUTURE SCOPE

In conclusion, it can be said that, the MDTCWT algorithm is computationally efficient, inherently offers advantages provided by the conventional DTCWT, and additionally maps directional signals at the end of the reconstruction stage. In the future, it may be possible to design new complex wavelet filters that will have properties similar to that of a Hilbert transformer for further reducing the computational complexity.

REFERENCES

Books

Burrus, CS, Gopinath, AR & Guo, H 1998, *Introduction to Wavelets and Wavelet Transforms : a primer*, Prentice Hall.

Proakis, JG & Manolakis, DG 2007, *Digital Signal Processing*, 4th edn, Pearson Prentice Hall, New Jersey.

Periodical Publications

Aydin, N, Fan, L & Evans, DH 1994, 'Quadrature-to-directional format conversion of Doppler signals using digital methods', *Physiol. Meas.*, vol 18, pp. 181-199.

Aydin, N & Markus, HS 2000, 'Directional wavelet transform in the context of complex quadrature Doppler signals', *IEEE Signal Processing Letters*, vol 10, no. 7, pp. 278-280.

Aydin, N, Marvasti, F & Markus, HS 2004, 'Embolic Doppler Ultrasound Signal Detection Using Discrete Wavelet Transform', *IEEE Transactions on Information Technology in Biomedicine*, vol 8, no. 2, pp. 182-190.

Aydin, N, Padayachee, S & Markus, HS 1999, 'The use of wavelet transform to describe embolic signals', *Ultrasound Med Biol.*, vol 25, no. 6, pp. 953-958.

Cohen, L 1989, 'Time-frequency distributions-a review', *Proceedings of the IEEE*, vol 77, no. 7, pp. 941 - 981.

Evans, DH, McDicken, WN, Skidmore, R & Woodcock, JP 1989, *Doppler Ultrasound: Physics, Instrumentation and Clinical Applications*, John Wiley, Chichester.

Kingsbury, NG 1999, 'Shift invariant properties of the Dual-Tree Complex Wavelet Transform', *IEEE International Conference On ACOUSTIC, SPEECH AND SIGNAL PROCESSING* , Phoenix.

Kingsbury, NG 2001, 'Complex wavelets for shift invariant analysis and filtering of signals', *Journal of Applied and Computational Harmonic Analysis*, vol 10, no. 3, pp. 234-253.

Rioul, O & Vetterli, M 1991, 'Wavelets and Signal Processing', *IEEE Signal Processing Magazine*, vol 8, no. 4, pp. 14-38.

Selesnick, IW 2001, 'Hilbert Transform Pairs of Wavelet Bases', *IEEE Signal Processing Letters*, vol 8, no. 6, pp. 170-173.

Selesnick, IW 2002, 'The Design of Approximate Hilbert Transform Pairs of Wavelet Bases', *IEEE Transactions on Signal Processing*, vol 50, no. 5, pp. 1144-1152.

Selesnick, IW, Baraniuk, NG & Kingsbury, NG 2005, 'The dual-tree complex wavelet transform', *IEEE Signal Processing Magazine*, vol 22, no. 6, pp. 123-151.

Other Publications

Aydin, N 1994, 'Computerised Graft Monitoring', Medical Physics, Faculty of Medicine.

Phillips, WJ 2009, *Dalhouse University - Faculty of Engineering*, <http://www.engmath.dal.ca/courses/engm6610/notes/node2.html>.

Phillips, WJ 2009, *Dalhouse University - Faculty of Engineering*, <http://www.engmath.dal.ca/courses/engm6610/notes/node4.html>.

Shukla, PD 2003, 'Complex Wavelet Transforms and Their Applications', Electronic and Electrical Engineering, Glasgow.

Wagner, C 2004, *Framelet Software at Broklyn Poly*, <http://taco.poly.edu/selesi/DoubleSoftware/index.html>.

VITAE

Name Surname : Görkem SERBES

Address : Bahçeşehir Üniversitesi Mühendislik Fakültesi
Çırağan Cd. Osmanpaşa Mektebi Sk. No: 4 – 6
34349 Beşiktaş / İstanbul / Türkiye

Birth Place / Year : ISTANBUL - 1984

Languages : Turkish (native) - English

Elementary School : Halil Bedii Yönetken Primary School – 1995

High School : Adnan Menderes Anatolian High School - 2002

BSc : Bahçeşehir University - 2006

MSc : Bahçeşehir University – 2009

Name of Institute : Institute of Science

Name of Program : Electrical & Electronics Engineering

Publications : SERBES G., AYDIN N., Doppler İşaretlerinin İşlenmesinde Yeni Bir Dalgacık Dönüşümü Yöntemi, BIYOMUT 2009, (2009)

SERBES G., AYDIN N., A Complex Discrete Wavelet Transform for Processing Quadrature Doppler Ultrasound Signals, ITAB 2009, submitted, (2009)

Work Experience : Bahçeşehir University Mechatronics Engineering Department
Teaching and Research Assistant
(August 2006 – to date)