

OPTIMIZABLE MULTIREOLUTION QUADRATIC VARIATION FILTER FOR
HIGH-FREQUENCY FINANCIAL DATA

A THESIS SUBMITTED TO
INSTITUTE OF APPLIED MATHEMATICS
OF
THE MIDDLE EAST TECHNICAL UNIVERSITY

BY

AYKUT ŞEN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
THE DEPARTMENT OF FINANCIAL MATHEMATICS

FEBRUARY 2009

Approval of the thesis:

**OPTIMIZABLE MULTIREOLUTION QUADRATIC VARIATION FILTER FOR
HIGH-FREQUENCY FINANCIAL DATA**

submitted by **AYKUT ŞEN** in partial fulfillment of the requirements for the degree of **Master of Science in Department of Financial Mathematics, Middle East Technical University** by,

Prof. Dr. Ersan Akyıldız _____
Director, Institute of **Applied Mathematics**

Assist.Prof.Dr.İşıl Erol _____
Head of Department, **Financial Mathematics**

Prof. Dr. Ersan Akyıldız _____
Supervisor, **Department of Mathematics, METU**

Assist.Prof.Dr.Kasırğa Yıldırak _____
Co-supervisor, **Institute of Applied Mathematics, METU**

Examining Committee Members:

Assoc.Prof.Dr.Azize Hayfavi _____
Institute of Applied Mathematics, METU

Prof. Dr. Ersan Akyıldız _____
Department of Mathematics, METU

Assist.Prof.Dr.Kasırğa Yıldırak _____
Institute of Applied Mathematics, METU

Assoc.Prof.Dr.Aslıhan Altay Salih _____
Department of Management, Bilkent University

Assist.Prof.Dr.Hakan Öktem _____
Institute of Applied Mathematics, METU

Date: _____

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: AYKUT ŞEN

Signature :

ABSTRACT

OPTIMIZABLE MULTIREOLUTION QUADRATIC VARIATION FILTER FOR HIGH-FREQUENCY FINANCIAL DATA

Şen, Aykut

M.S., Department of Financial Mathematics

Supervisor : Prof. Dr. Ersan Akyıldız

Co-Supervisor : Assist.Prof.Dr.Kasırğa Yıldırak

February 2009, 72 pages

As the tick-by-tick data of financial transactions become easier to reach, processing that much of information in an efficient and correct way to estimate the integrated volatility gains importance. However, empirical findings show that, this much of data may become unusable due to microstructure effects. Most common way to get over this problem is to sample the data in equidistant intervals of calendar, tick or business time scales. The comparative researches on that subject generally assert that, the most successful sampling scheme is a calendar time sampling which samples the data every 5 to 20 minutes. But this generally means throwing out more than 99 percent of the data. So it is obvious that a more efficient sampling method is needed. Although there are some researches on using alternative techniques, none of them is proven to be the best.

Our study is concerned with a sampling scheme that uses the information in different scales of frequency and is less prone to microstructure effects. We introduce a new concept of business intensity, the sampler of which is named Optimizable Multiresolution Quadratic Variation Filter. Our filter uses multiresolution analysis techniques to decompose the data into different scales and quadratic variation to build up the new business time scale. Our empirical findings show that our filter is clearly less prone to microstructure effects than any other

common sampling method.

We use the classified tick-by-tick data for Turkish Interbank FX market. The market is closed for nearly 14 hours of the day, so big jumps occur between closing and opening prices. We also propose a new smoothing algorithm to reduce the effects of those jumps.

Keywords: Microstructure Effects, Integrated Volatility, Realized Variance, Multiresolution Analysis, Wavelets, High-Frequency Finance, Seasonality, Data Smoothing, Turkish Foreign Exchange Market

ÖZ

YÜKSEK FREKANSLI FİNANSAL VERİ İÇİN OPTİMİZE EDİLEBİLİR ÇOK ÇÖZÜNÜRLÜKLÜ KARESEL VARYASYON FİLTRESİ

Şen, Aykut

Yüksek Lisans, Finansal Matematik Bölümü

Tez Yöneticisi : Prof. Dr. Ersan Akyıldız

Ortak Tez Yöneticisi : Yrd.Doç.Dr.Kasırga Yıldırak

Şubat 2009, 72 sayfa

İşlem bazında finansal veriye daha kolay ulaşılabildikçe, anlık oynaklığın tahmin edilmesi önem kazanan bir konu haline gelmiştir. Bunun için yüksek frekanslı verinin doğru ve verimli şekilde işlenmesi gerekmektedir. Ancak gözlemsel bulgular, yüksek frekanslı verinin, mikroyapı etkileri nedeniyle kullanılamaz hale gelebileceğini göstermektedir. Bu sorunun üstesinden gelmek için en sık kullanılan yöntem, takvim, işlem veya işsel yoğunluk ölçütlerinde eşit aralıklı örnekleme yapmaktır. Bu konuda yapılmış karşılaştırmalı araştırmalar, en başarılı örnekleme şemasının 5 ila 20 dakikada bir örnekleme yapan bir takvimsel örnekleme şeması olduğunu ileri sürmektedir. Ancak bu, genelde verinin yüzde 99'unun kullanılamaması anlamına gelmektedir. Bu durumda, daha iyi bir örnekleme yöntemine ihtiyaç duyulduğu çok açıktır. Değişik araştırmalar alternatif örnekleme teknikleri önerse de, bunların hiçbiri en iyi olarak sunulamamaktadır.

Çalışmamız, verinin değişik frekanslarda taşıdığı bilgiyi kullanan ve mikro yapı etkilerine daha az maruz kalan bir örnekleme şeması üzerinde yoğunlaşmaktadır. Örnekleyicisine Optimize Edilebilir Karesel Varyasyon Filtresi adını verdiğimiz yeni bir işsel yoğunluk kavramı sunmaktayız. Filtremiz Çoklu Çözünürlük Tekniklerini kullanarak veriyi değişik frekanslara ayırmakta ve karesel varyasyon kullanarak yeni bir işsel zaman ölçütü yaratmaktadır.

Gözlemsel bulgularımız, filtremizin, diğer sık kullanılan örnekleme yöntemlerine göre mikroyapı etkilerine çok daha az maruz kaldığını göstermektedir.

Kullandığımız veri, Türkiye Bankalararası Döviz Piyasası'nın işlem bazında verileridir. Bu piyasa günde yaklaşık 14 saat kapalı olduğundan, kapanış-açılış fiyatları arasında ciddi oynamalar görülebilmektedir. Bu oynamaların etkilerini azaltmak için ayrıca bir yumuşatma yöntemi önermekteyiz.

Anahtar Kelimeler: Mikroyapı Etkileri, Anlık Oynaklık, Gerçekleşmiş Varyans, Çoklu çözünürlük Analizi, Yüksek Frekanslı Finansal Veri, Mevsimsellik, Veri Yumuşatma, Türkiye Bankalararası Döviz Piyasası

To Sevgi and Alp Tuna

ACKNOWLEDGMENTS

I would like to thank my supervisors in this thesis and also the people of the Graduate School of Applied Mathematics and the Central Bank of Turkey. I also want to thank my family for their understanding and never-ending support.

TABLE OF CONTENTS

ABSTRACT	iv
ÖZ	vi
DEDICATION	viii
ACKNOWLEDGMENTS	ix
TABLE OF CONTENTS	x
LIST OF TABLES	xii
LIST OF FIGURES	xiii
CHAPTERS	
1 INTRODUCTION	1
2 HIGH FREQUENCY DATA DYNAMICS	4
2.1 Doob-Meyer Decomposition, Continuous Square Integrable Martingales and Quadratic Variation	5
2.2 Realized Variance and Integrated Volatility	7
2.3 Microstructure Effects and Time Sampling Schemes	9
2.4 Seasonality Removal and Data Smoothing	11
3 WAVELET THEORY AND MULTIREOLUTION ANALYSIS	14
3.1 Fourier Analysis	15
3.2 Wavelet Theory	21
3.3 Multiresolution Analysis	29
3.4 The Daubechies Wavelet Family	39
4 NEW MODEL FOR HIGH FREQUENCY DATA SAMPLING	48
4.1 Introduction	48
4.2 Rescaling and Smoothing	50
4.3 Multiscale Quadratic Variation Filter	56

4.4	Measuring The Performance	58
5	EMPIRICAL RESULTS AND CONCLUSION	60
5.1	Structure Of The Data And Rescaling	60
5.2	Smoothing	62
5.3	Results	62
5.4	Conclusion And Further Research Topics	68
	REFERENCES	70
	APPENDIX	
A	DECOMPOSITION	72

LIST OF TABLES

TABLES

Table 5.1	Resulting optimized window for rescaling	60
Table 5.2	Comparison of several centered FIR filters with our filter	62
Table 5.3	Relative Bias values for CTS, TTS and our sampling scheme	65
Table 5.4	Relative Bias values for CTS, TTS and the scaling function iterations for best intuitive w_j	67
Table 5.5	Final comparison of the models	67

LIST OF FIGURES

FIGURES

Figure 3.1 Plot of $f(t) = \cos(t) + \frac{1}{2} \sin(2t) + 5 \cos(10t) - \frac{1}{5} \sin(200t)$	16
Figure 3.2 Plot of $f'(t) = \cos(t) + \frac{1}{2} \sin(2t) + 5 \cos(10t)$	16
Figure 3.3 (a) Real portion of the Morlet wavelet (b) Mexican hat wavelet	22
Figure 3.4 (a) Haar wavelet (b) Haar scaling function	26
Figure 3.5 Example of a typical element of V_0	27
Figure 3.6 Diagram of decomposition for multiresolution analyses	39
Figure 3.7 Plot of the iteration steps of Daubechies Scaling Function ($N = 2$)	46
Figure 3.8 Plot of the Daubechies Wavelet Function Extracted from ϕ_8 ($N = 2$)	47
Figure 4.1 Plot of the iteration steps of the smoothing in [Gençay 2001] with USD-TRY transactional data of our research	49
Figure 4.2 Histogram (X-axis in seconds)	51
Figure 4.3 The rescaling algorithm	52
Figure 4.4 A classical FIR filter for fixed intervals	53
Figure 4.5 The smoothing filter in this research	55
Figure 4.6 Plot of the weight of input vector items on output vector	56
Figure 5.1 Scatter diagram of the raw data	61
Figure 5.2 Histograms of the data before and after the scaling algorithm is applied	61
Figure 5.3 Compared scatter diagram of the smoothing process	63
Figure 5.4 Scatter diagram of the data after rescaling and smoothing	64
Figure 5.5 Performances (relative bias) of CTS , TTS and our filter over regarding sampling periods (in seconds)	66

Figure 5.6 Performances (relative bias) of **CTS**, **TTS**, **our filter w/o sf iteration** and **our filter with sf iteration** over regarding sampling periods (in seconds) 68

CHAPTER 1

INTRODUCTION

The most common approach in financial analysis is to assume that most of the parameters that effect the transactions are homogeneous. But in fact, there are a lot of variables that effect prices and trade intensity. For instance, even the composition of the traders, their risk adverseness and the information they have change from minute to minute. There are speculative trades as well as investment trades. Such external effects over the prices that vary continuously on a time scale are called *microstructure effects* ([Dacorogna 2001], [O'Hara 1997]). The impact of those *microstructure effects* may be negligible for macroeconomic analysis or VAR analysis which use a spectrum of time series, varying from daily to annual data where instantaneous or speculative trends are observed sparsely. However when analyzing the high-frequency data, one cannot underestimate those *microstructure effects* since they entangle the whole price information and threat the *stylized facts*.

So, when we enter the world of the high-frequency finance, we know that the markets are not homogeneous and the best financial analysis should take those *microstructure effects* into consideration. They especially count when estimating the integrated volatility. The most common but lazy way in literature to deal with those effects is to sparsely sample the tick-by-tick data and waste much of it. Yet there are some researches in the literature trying to develop ways to make use of all the data available, like [Zhang 2005]. However, most of the high-frequency data analyses still assume that the *microstructure effects* are not variant over time. So, such researches try to find an optimal constant sampling interval dependent on calender, tick or business time scale and assume the samples obtained via those treshold points are homogeneous.

As we stated earlier, *microstructure effects* are negligible when the transactions are done for investment purposes (long term) and are apparent when the trading is done for speculative purposes. So if we can separate the investment trends from speculative trends, we may model those effects better. That's the point, where *wavelets* and *multiresolution analysis* (MRA) enters the scene.

Multiresolution analysis and *wavelets*, originating from *Fourier analysis* are in fact used in signal processing to decompose signals into orthogonal components, then clear distortions due to external effects and finally reconstruct. With an analogy to signal processing, they are mostly used to clear noise from financial data by analyzing the data both in time and frequency resolutions.

What we will present in our work is a new concept of *optimizable business intensity* that is provided via *multiresolution analysis*. We decompose the data into its resolutions of orthogonal components, evaluate new total business intensity and sample for different frequencies. With those samples, we evaluate the realized variance and estimate integrated volatility. Then we iterate this procedure for all levels of resolution. For optimization, we assign a bias measure that we determine due to market practice and evaluate the relative bias for each path. We name the whole procedure as *OPTIMIZABLE MULTIREOLUTION QUADRATIC VARIATION FILTER FOR HIGH-FREQUENCY DATA*.

The research that inspired this work is the 2001 paper of Ramazan Gençay, Faruk Selçuk and Brandon Whitcher, 'Differentiating intraday seasonalities through wavelet multi-scaling' ([Gençay 2001]). In that research, the authors provide a model for removing the seasonalities of high frequency data by using *Discrete Wavelet Transform* and via this model obtain an ACF that is very close to the ACF of a long-memory process that satisfies $\gamma(k) \approx \lambda k^{-\alpha}$ where λ is the scaling parameter and $\alpha \in [0, 1]$. This condition is asserted by [Hosking 1996].

The problem with the approach is that, it oversmooths the data at higher iterations, so that the resulting series do not seem like the original data anymore. We will illustrate this with our data in Chapter 4. So researchers do not tend to use that procedure much, because there is a possibility that the model throws away some useful information. They still look for sampling schemes that sample from the original data with

less bias. So we don't use the oversmoothed data, but the orthogonal complement of smoothed data in *Hilbert Space* in each scale to develop a sampling scheme.

We use the classified *tick-by-tick data* of *USD-TL* trades, obtained from the *Central Bank of Turkey*, which take place in November and December of 2008. The data is provided for internal use only, therefore we cannot publish so much detail in tables. The nice thing about the data is that, it involves some of the trades during the *Global Crisis of 2008 and 2009*. So the data is highly variant by means of frequency and volatility, which is a desirable property for our analysis. To the best of our knowledge, that kind of analysis is done for the very first time with *Turkish Foreign Exchange Market Data*.

The outline of this work is as follows. In Chapter 2, we present the *High Frequency Data Dynamics*. We give the basics about *realized variance* being an estimator for *integrated volatility* and then present the common sampling schemes of interest which are used to compute *realized variance* with minimum bias. We also present the common seasonality removal and data smoothing approaches. In Chapter 3, we present *Wavelets* and *Multiresolution Analysis*. We begin presenting the subject with *Fourier Analysis*, then *wavelet theory* and finally *Daubechies Family of Wavelets*. In Chapter 4, we present our model and obtain our relative bias measure. And finally in Chapter 5, we present the empirical results of our work.

CHAPTER 2

HIGH FREQUENCY DATA DYNAMICS

When analysing the high-frequency data, the risk parameter that researchers desperately seek is the *integrated volatility* which is the instantaneous volatility of an instrument. The value of the parameter is not available in a direct manner, but can be estimated by *realized variance*. However, when estimating the *integrated volatility* they often run into a bunch of *High Frequency Data Dynamics Problems* that stand on the way to a sound analysis. Microstructure effects such as the characteristics of the traders and bid-ask spreads as well as the liquidity of the financial instruments and intraday seasonality (i.e. effects of high trading volume at the beginning and end of a trading session) are some of the issues that must be taken into consideration for they all cause a bias in the analysis.

Empirically, more sparsely sampled data leads to a less biased analysis in terms of *High Frequency Data Dynamics* (but at the same time it invokes some other macro effects from some point on, which is not the concern of this work). However, more sparsely sampled data means more information thrown out to trash, which is not desired at all. So the main concern is to make use of as much data as possible. Therefore, the high frequency data dynamics should be modeled carefully.

In this chapter, the first two sections give the basic concepts about *realized variance* being an estimator for *integrated volatility*. *Doob-Meyer Decomposition Theorem* asserts that every submartingale can be decomposed into a martingale and an increasing process in continuous time. We use this theorem to in another theorem which asserts that the second variation of a square integrable process approaches the increasing process *quadratic variation* in probability. And then, the second section

makes use of this theorem to prove that *realized variance* can be used to estimate *integrated volatility*. Section 2.3 defines the microstructure effects and common practice that is used to minimize the bias caused by them. Then the last section gives some general knowledge about seasonality removal and data smoothing.

2.1 Doob-Meyer Decomposition, Continuous Square Integrable Martingales and Quadratic Variation

In order to give a formal definition for the realized volatility, we must do some preliminary work. To begin with, we must first remark that every submartingale may be decomposed into a martingale and an increasing process in continuous time. Then we will give formal definitions for *continuous square integrable martingales* and their *quadratic variation*, which is the submartingale we are going to decompose by using Theorem 2.1.2. And then, we have to prove quadratic variation theorem for continuous square-integrable martingales, which we will use in order to prove that *realized variance* is an estimator for *integrated volatility*. We mainly follow [Karatzas 1991] and assume that our filtration $\{\mathcal{F}_t\}$, processes $\{A_t\}$ and martingales $\{M_t\}$ on the probability space $(\Omega, \mathcal{F}, \mathbb{P})$ will be defined in continuous time, i.e. $t \in [0, \infty)$, and we assume, our filtration is right-continuous and \mathcal{F}_0 contains all the \mathbb{P} -negligible events in \mathcal{F} .

Although we mainly follow [Karatzas 1991] for the argument, one may find alternative methods in [Rogers 1994.v2] and [Delacherie 1982]. The interested reader may find the basics of the concepts such as *increasing processes*, *natural processes*, *martingale transform*, *uniform integrability* or *Lebesgue Integration* in [Karatzas 1991], [Koralov 2007], [Rogers 1994.v2] and [Delacherie 1982].

We start by making a classification of the right-continuous adapted processes.

Definition 2.1.1 *The right-continuous adapted process $X = \{X_t\}_{t \in \mathbb{R}_+}$ is said to be,*

- *of class D, if the family $\{X_T : T \text{ is a stopping time, where } T \leq \infty\}$ is uniformly integrable*
- *of class DL, if for every $a > 0$, the family $\{X_T : T \text{ is a stopping time, where } T \leq a\}$ is uniformly integrable*

Here, one may refer to [Koralov 2007] chapter 13 and [Rogers 1994.v1] chapter II for the concept of uniform integrability as stated before. The *Doob-Meyer Decomposition Theorem* gives the conditions for the decomposition of those classes of right-continuous adapted processes.

Theorem 2.1.2 (Doob-Meyer Decomposition Theorem) *If $X = X_t$ is a right-continuous submartingale of class DL, then there exists a right-continuous martingale $M = M_t$ and a right-continuous increasing process $A = A_t$, s.t. $X_t = M_t + A_t, \forall t \geq 0$ a.s. If A is natural, then this decomposition is unique. Also, if X is class D, then M and A are uniformly integrable.*

The proof of this theorem may be found in [Karatzas 1991]. Now we may give the definitions for *square integrable martingale* and *quadratic variation*.

Definition 2.1.3 (Square-Integrable Martingale) *A right-continuous martingale $X = \{X_t\}_{t \geq 0}$ is square-integrable if $E(X_t^2) < \infty$ for every $t \geq 0$. It is written as $X \in \mathcal{M}_2$. If X is continuous, it is written as $X \in \mathcal{M}_2^c$.*

The function X^2 of the right-continuous square-integrable martingale $X = \{X_t\}_{t \geq 0}$ is a non-negative submartingale. The proof follows from *Jensen's Inequality*. So X^2 is of class DL (Definiton 2.1.1) and has a unique *Doob-Meyer Decomposition*,

$$X_t^2 = M_t + A_t, \quad \forall t \geq 0 \quad (2.1)$$

where M and A are defined the same way as in Theroem 2.1.2.

Definition 2.1.4 (Quadratic Variation) *The Quadratic Variation of a right-continuous square-integrable martingale process $X = \{X_t\}_{t \geq 0}$ is defined to be the process $\langle X \rangle_t = A_t$ where A is a natural increasing process in the Doob-Meyer Decomposition of X^2 as it is in 2.1.*

Note that, from Theorem 2.1.2, we have $\langle X \rangle$ is a unique, adapted, natural increasing process, for which $\langle X \rangle_0 = 0$ a.s. and $X^2 - \langle X \rangle$ is a martingale.

Definition 2.1.5 (p^{th} Variation) *Let $X = \{X_t\}_{t \geq 0}$ be a process. For a fixed $t > 0$, let $\Pi_m = \{t_0, t_1, \dots, t_m\}$ where $0 = t_0 \leq t_1 \leq \dots \leq t_m = t$ be a partition of $[0, t]$. Then the p -th*

variation of X over the partition Π_m is defined to be,

$$V_t^p(\Pi_m) = \sum_{k=1}^m |X_{t_k} - X_{t_{k-1}}|^p, \quad p > 0$$

Now let us define the mesh, $\|\Pi\|$ of the partition Π_m in Definition 2.1.5 as $\|\Pi\| = \max_{1 \leq k \leq m} |t_k - t_{k-1}|$. Then the limit $V_t^2(\Pi_m)$ converges as $\|\Pi_m\| \rightarrow 0$. Next theorem (Theorem 2.1.6) asserts that this limit is what we call the quadratic variation of X over $[0, t]$ in Definition 2.1.4 and we will use this theorem to prove that *realized variance* is an estimator for *integrated volatility*. What we claim here in fact is that we can neglect the cross-product terms when we square the sums of martingale increments and take expectation.

Theorem 2.1.6 *Let $X = \{X_t\}_{t \geq 0}$ be a continuous square-integrable martingale process. Then considering the partitions Π of $[0, t]$, for every $\epsilon > 0$ and $\eta > 0$ there exists $\delta > 0$ s.t. if $\|\Pi\| < \delta$ then*

$$P\left(\left|V_t^2(\Pi) - \langle X \rangle_t\right| > \epsilon\right) < \eta$$

In other words, $\lim_{\|\Pi\| \rightarrow 0} V_t^2(\Pi) = \langle X \rangle_t$ in probability.

The proof of this theorem can also be found in [Karatzas 1991].

2.2 Realized Variance and Integrated Volatility

In this section we will introduce the terms *realized volatility*, and *integrated volatility* and will use the facts that we have given in previous section to prove that *realized variance* is an estimator for *integrated volatility*.

We need a model for describing continuous time price movements. So let S_t denote the price process of a financial instrument at time t and $X_t = \log S_t$. Let instantaneous returns are generated by a continuous time martingale,

$$dX_t = \sigma_t dW_t \tag{2.2}$$

where W_t is a standart Wiener process. Here, σ_t^2 denotes the *instantaneous (spot) variance*.

What we are completely interested in finding is the *conditional variance* of returns, which is the instantaneous variance of the upcoming instantaneous returns,

given the information at time t . The quadratic variation of the returns in our continuous case is equal to $\int_0^T \sigma_t^2 dt$, namely *integrated volatility* over the period $[0..T]$.

For a fixed T , let Π_m be a equipartition of $[0, T]$ with subintervals of size $\Delta t = \frac{T}{m}$. By definition, *realized volatility* (historical volatility), ν , for exponent p , is

$$\nu(t_i) = \left[\frac{1}{m} \sum_{i=1}^m |X_{t_i} - X_{t_{i-1}}|^p \right]^{1/p} \quad (2.3)$$

where X_{t_i} follows 2.2.

Exponent p is often set to 2, so that ν^2 is the variance of the returns about zero. Changing the ν into scaled form by $\nu_{scaled} = \sqrt{\frac{m\Delta t}{\Delta t}} \nu$ one gets the *realized variance*¹,

$$\nu^2 = \sum_{i=1}^n (X_{t_{i+1}} - X_{t_i})^2 \quad (2.4)$$

For a wider explanation, one may refer to [Hull 1987] or [Dacorogna 2001].

Carefully examining, one may see that 2.4 is analogous to $V_t^2(\Pi_m)$ defined in 2.1.5. So, as we have the quadratic variation $\langle X \rangle_t = \int_0^T \sigma_t^2$ (*integrated volatility*) and $V_t^2(\Pi_m)$, we may make use of our former work. Since the sample path for σ_t is continuous, it follows from Theorem 2.1.6 that, for every $\epsilon > 0$, $\eta > 0$, there exists $\delta > 0$ such that $\Delta t < \delta$ implies

$$P \left[\left| \int_0^T \sigma_t^2 dt - \sum_{i=1}^n (X_{t_{i+1}} - X_{t_i})^2 \right| > \epsilon \right] < \eta \quad (2.5)$$

Therefore we get,

$$p \lim_{n \rightarrow \infty} \left[\int_0^T \sigma_t^2 dt - \sum_{i=1}^n (X_{t_{i+1}} - X_{t_i})^2 \right] = 0 \quad (2.6)$$

So, *realized variance* converges to *integrated volatility* in probability. Then we may conclude that *realized variance* is an estimator for *integrated volatility*. Moreover, using 2.6 we may assert that our estimator is irrespective of the size of Δt and the size of the return intervals can be irregular (See [Dacorogna 2001]). Therefore, we can use different *time sampling schemes*.

¹ There is an ambiguity with the name *realized volatility* since [Andersen 2000] has given this name to ν^2 , so one may see the terms *realized volatility* and *realized variance* can be used interchangeably

2.3 Microstructure Effects and Time Sampling Schemes

One theoretical result of 2.6 is, when using *realized variance* as an estimator of *integrated volatility*, sampling in higher frequencies should lead to a better estimation. With our previous assertion that the return intervals can be irregular, one may expect that using tick-by-tick data should lead to the best estimation. That's because an increase in sampling frequency leads to a reduction in the variance of the estimator. In an ideal world, where prices are continuous and have no measurement error, it is the ideal estimator (See [Merton 1980]). Nevertheless, empirical results of different researches such as [Andreou 2002] show that, with tick-by-tick data, realized variance suffers from bias problem.

The main reason for that phenomenon is the amplification caused by the serial correlation of a bulk of noise sources which are called *microstructure effects* such as the bid-ask spread, composition of the traders or transaction costs. Empirical results show that, as the prices are sampled in finer intervals, the bias caused by those *microstructure effects* become more viable. We will not go into the details of the structure of those effects too much, but there is a wide literature, for those, who are interested in the subject, such as [O'Hara 1997], [Dacorogna 2001] and [Hasbrouck 2004]. Obviously the bias caused by those effects change from trading instrument to trading instrument.

It is pretty much clear that we cannot use the tick-by-tick return data directly in the *realized variance* equation. But it is also clear that throwing away a bulk of the most valuable source for an analysis, the data, by sampling in low frequencies is not appreciated. Therefore the aim is to find the optimal sampling frequency by making use of as much data as possible. Of course, the best way is the way that uses all the data with minimum bias. There are a lot of approaches to sampling in that manner but the optimization is generally done by minimizing the *mean squared error* (MSE) of the volatility estimator, which will also be presented in our model in the next chapter. Among different time sampling schemes, most remarkable ones are *Calendar Time Sampling* (CTS), *Tick Time Sampling* (TTS) and *Business Time Sampling* (BTS).

Calendar Time Sampling (CTS) is the most widely used sampling scheme. It is

based on sampling the sequence of prices as,

$$P_{cts} = \{P(t_i^{cts})\}_{i \in \{0,1,\dots,N\}}, \quad t_i^{cts} = i\Delta t = i\frac{T}{N} \quad (2.7)$$

Since *CTS* samples the prices in regularly spaced points in time, an interpolation method usually accompanies this scheme since the tick-by-tick time series are not homogeneous, that means it is hardly the case that a transaction occurs right on $i\Delta t$, and a theoretical price should be found. By utilizing an interpolation method, a homogeneous time series is constructed. One can refer to [Bandi 2003] or [Bandi 2003] for a detailed research on finding the optimal data sampling frequency. Those researches claim that the optimal sampling frequency should be somewhere between 5 minutes and 20 minutes.

The problem with *CTS* is that it throws away too much data. For example, consider a tick-by-tick transactional data with one transaction per second on average. If we sample the prices every 10 minutes, this means that 599 of every 600 data is thrown away.

Tick Time Sampling (TTS) is a sampling scheme which is based on sampling the sequence of prices as,

$$P_{tts} = \{P(t_i^{tts})\}_{i \in \{0,1,\dots,N\}}, \quad t_i^{tts} = i\frac{M}{N} \quad (2.8)$$

where M is the total number of transactions and $\frac{M}{N}$ is integer valued. Interested reader may refer to [Clark 1973] for a detailed coverage of the subject. The scheme is sensitive to the trade intensity of transactions which can be an indicator of movements and may be preferred to (CTS) in some cases. Because, when the transaction volume is rather low, interpolation may itself be a source of bias.

Business Time Sampling (BTS) will be our main interest since we will extend this scheme in our model. This scheme is based on sampling the sequence of prices as,

$$P_{bts} = \{P(t_i^{bts})\}_{i \in \{0,1,\dots,N\}}, \quad t_i^{bts} = \inf_{0 \leq t \leq T} \left\{ \lambda_t \geq i\frac{\lambda_T}{N} \right\} \quad (2.9)$$

where λ_t is the *business intensity* at time t . We first evaluate the total *business intensity*, then get equipartitions of the total *business intensity* and sample the price process every time, the *business intensity* reaches the treshold of the partition. That approach has some advantages since the sampling is done in finer intervals whenever the volatility

is high and in wider intervals whenever the volatility is low. But this also causes some latency problems.

In order to determine the best approach that minimizes the bias and makes use of much of the data, several researches have been made. Among those, [Oomen 2005] and [Zhang 2005] are among the most considerable ones. [Oomen 2005] finds that among above three sampling schemes, (TTS) is generally superior to (CTS) when the trade intensity pattern is volatile, while [Zhang 2005] first compares traditional sampling schemes, then suggests a hybrid sampling scheme.

It can be understood from the literature that it is hard to determine the best approach for a sampling scheme since every sampling scheme has its advantages over others in specific cases. However, in Chapter 4, we will develop a sampling scheme which adapts itself to different conditions caused by the microstructure effects.

2.4 Seasonality Removal and Data Smoothing

In order to give the main concepts about *seasonality removal*, we will follow the guidelines in [Dacorogna 2001]. The main reference for the seasonality is the market activity. Markets are more active during some periods of the calendar time and less active during other times. For instance, take a globally traded currency pair. It is traded actively nearly 23 hours a day, beginning from Far East, then continuing with Europe and ending the day in America. There are hours when the currency pair is traded in two regions² as well as there are hours when the currency pair is traded only by the people on duty during the lunchtime³. *Dacarogna et.al* [Dacorogna 2001] define the *activity variable* as

$$a_{1,2} = \frac{\vartheta_2 - \vartheta_1}{t_2 - t_1} \quad (2.10)$$

where ϑ is the time scale by means of our recognition of activity. This could be the number of ticks, volume of trades, or any other variable that can be considered as activity. This approach of activity is called *volatility based activity*. Although there are many other options, the basic approach is minimizing the weighted square

² for instance late afternoon session for European market coincides with morning session of the USA market

³ for instance in Far East

deviation of the activity, which is

$$\min\left(\frac{[a_{stat,i} - a_i]^2}{\sigma_{error,i}^2}\right) \quad \text{where} \quad \frac{1}{n} \sum_{i=1}^n a_{stat,i} = 1 \quad (2.11)$$

The minimization of the deviation gives us the optimal activity scale, and therefore, we may have our data more homogeneous. But there is still one thing left to fix: The difference between the open and close prices.

There are several ways to deal with the price differences between consecutive days. One is discarding them by only concentrating on the intraday returns. The other way is using a filter in time domain. Discarding the price differences between consecutive days is the simplest and most preferred way to deal with the problem. Statistically, it also smooths the path. However it may not be desirable since it still contains information: All the information between two days, while the markets are closed. One may think that the global markets and the traded instruments nearly never closed during the week. Yet there is a whole weekend of more than 1.5 days still waiting for treatment. For locally traded instruments, the problem is even deeper. For instance USD-TRY market is open only between 5:00AM GMT and 3:30PM GMT. During 13,5 hours of no-trade time, all the information for America and Far East information pass by. So we need to take this information into consideration for a fair analysis. To deal with the statistical effects of this single price information is generally smoothing the data around. Here, we will present the filters as they are presented in [Gençay 2002].

We usually want to stay in the time domain to smooth price jumps since staying in the frequency domain does not make much sense. Filters in the time domain take some part of a series of observations (say x_{t-M} to x_{t+N-1})

$$\{x_t\}_{t=-\infty}^{\infty} = (\dots, x_{-2}, x_{-1}, x_0, x_1, x_2, \dots)$$

and transform this vector of size $M+N$ into an output value y_t by applying *convolution operation* (See Section 3.1) with vector w of size $M+N$. The elements of this vector $(\dots, w_{-2}, w_{-1}, w_0, w_1, w_2, \dots)$ are called *filter coefficients*. If the values of the filter does not change over time, we call those kinds of filters as *time invariant filters*. Given a unit impulse signal, the output sequence of a filter is known as *impulse response*. Practically, it is the sequence of the coefficients of a filter.

We either use *infinite impulse response (IIR) filters* or *noncausal finite impulse response (FIR) filters* in time domain. We will just give a brief explanation of those kind of filters. One may refer to [Gençay 2002] for a wide coverage.

Infinite impulse response (IIR) filters are the filters of the form

$$y_t = \sum_{i=1}^L a_i y_{t-i} + \sum_{i=1}^M w_i x_{t-i}$$

where L lagged values of output y_t and M lagged values of input x_t , as well as the current value of the input are employed to determine the current value of the output.

Noncausal finite impulse response (FIR) filters are preferred more in financial applications. The general form of an FIR filter is

$$y_t = \sum_{i=-N}^M w_i x_{t-i} \tag{2.12}$$

The most common filter in finance is the simple centered moving average, where all the w_i 's are the same ($\frac{1}{M+N+1}$). The centered moving average filters are symmetric with $M = N$ and $w_i = w_{-i}$, but the filter coefficients w_i need not be equal to each other. The centered average filters are preferred because they have a constant impulse response and are easy to implement. We will come to this in section 4.2.

CHAPTER 3

WAVELET THEORY AND MULTIREOLUTION ANALYSIS

Although *wavelets* is a topic of a few decades in the literature, its origins go back to early 1800's when Joseph Fourier asserted that any 2π – *periodic* function may be represented as a sum of sinusoidal functions with appropriate coefficients. Since Ingrid Daubechies' research opened the way to construction of compactly supported orthonormal wavelets, the tool has become popular in signal processing and some other areas. Only in the late 1990's, like many other methods that are inherited from engineering, when the computing abilities showed great development and high-frequency data has become available, researchers started to use wavelets as a tool for financial analysis.

So both *Fourier analysis* and *wavelet analysis* aim to decompose, *signals*¹ into serial components, but some properties make *wavelet analysis* more applicable to financial data.

Fourier series is a linear combination of *sinusoidal functions* whose components (sines and cosines) are also periodic functions. So the bases, constructed in *Fourier analysis* are useful when working with stationary time series. However, most financial data show remarkable changes in pattern due to the effects mentioned in Chapters 1 and 2. Therefore its structural features make *Fourier transform* inefficient in capturing those events. Those structural periodic features of basis functions also cause a lack of information in time. But the *wavelet transform* utilizes a basic function (mother wavelet) that can be stretched and shifted over time to intelligently adapt itself to capture features that are local in time. For a deep coverage of the subject, one can

¹ Throughout this chapter, we use the expression *signal*, literally where needed, as it is in signal processing which is the original area of usage for *Fourier analysis*

refer to [Boggess 2001], [Gençay 2002], [Wojtaszczyk 1997] or [Bachman 2000] which we will also follow throughout this chapter. Moreover, the proofs given in this chapter can also be found with a deep coverage in those references.

The composition of the chapter is as follows. In section 3.1, we present the main features of *Fourier analysis*, then in section 3.2, we make our way through *wavelets*. In section 3.3, we present *Multiresolution Analysis*, which we are going to implement in our model. And then, in section 3.4, we will see how to construct a basis in the *Daubechies Wavelet Family*

3.1 Fourier Analysis

As stated before, *Fourier series* is used to decompose a periodic *signal* into a linear combination of frequency components (*sinusoidal functions*). There are some practical reasons for using the *trigonometric expansion* of a function instead of the function itself. For instance, using this representation gives the description of the component frequencies and makes analyzes in different time scales possible. Let us give an example at this point.

Example 3.1.1 Consider a signal, which is decomposed into the function,

$$f(t) = \cos(t) + \frac{1}{2} \sin(2t) + 5 \cos(10t) - \frac{1}{5} \sin(200t)$$

which is plotted in figure 3.1. We believe that the frequencies greater than 100 denote the noise in the signal. So we eliminate the last component of the function, which has a frequency of 200. So we now have a smoother function,

$$f'(t) = \cos(t) + \frac{1}{2} \sin(2t) + 5 \cos(10t)$$

which is plotted in figure 3.2.

The components of the *Fourier series* are $\sin(kt)$'s and $\cos(kt)$'s which vibrate at a frequency of k times per 2π interval and t is a variable of time. We call this decomposition as *trigonometric expansion*. Let us formalize our statement. Note that we assume that our definitions are on the interval $-\pi \leq x \leq \pi$ for the time being.

Definition 3.1.2 (Trigonometric Expansion) The trigonometric expansion of an integrable function $f(x)$ defined on interval $-\pi \leq x \leq \pi$ is the approximation of $f(x)$

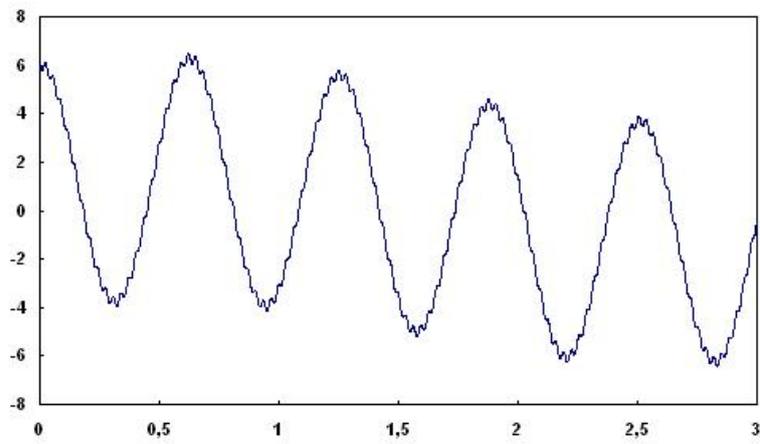


Figure 3.1: Plot of $f(t) = \cos(t) + \frac{1}{2} \sin(2t) + 5 \cos(10t) - \frac{1}{5} \sin(200t)$

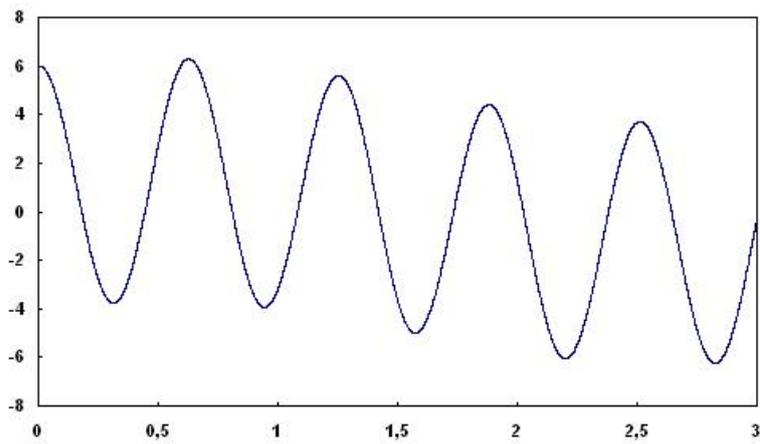


Figure 3.2: Plot of $f'(t) = \cos(t) + \frac{1}{2} \sin(2t) + 5 \cos(10t)$

as a form of the Fourier series,

$$a_0 + \sum_{k \in \mathbb{N}} a_k \cos(kx) + \sum_{k \in \mathbb{N}} b_k \sin(kx) \quad (3.1)$$

where this sum can be finite or infinite and the Fourier Coefficients are

$$a_0 = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x) dx \quad (3.2)$$

$$a_k = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \cos(kx) dx = \left\langle f, \frac{\cos(kx)}{\pi} \right\rangle \quad (3.3)$$

$$b_k = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \sin(kx) dx = \left\langle f, \frac{\sin(kx)}{\pi} \right\rangle \quad (3.4)$$

The inner products in equations 3.3 and 3.4 clearly denote the projection of f on $\frac{\cos(kx)}{\pi}$ and $\frac{\sin(kx)}{\pi}$ s. Since the sum of the projections of f on $\frac{\cos(kx)}{\pi}$ s and $\frac{\sin(kx)}{\pi}$ s give the approximation to our function, the set $\left\{ \dots, \frac{\cos(2x)}{\sqrt{\pi}}, \frac{\cos(x)}{\sqrt{\pi}}, \frac{1}{\sqrt{2\pi}}, \frac{\sin(x)}{\sqrt{\pi}}, \frac{\sin(2x)}{\sqrt{\pi}}, \dots \right\}$ should be an orthonormal set of functions in $L^2([-\pi, \pi])$. Next theorem asserts that property.

Theorem 3.1.3 *On the interval $-\pi \leq x \leq \pi$, the following integral relations hold,*

$$\frac{1}{\pi} \int_{-\pi}^{\pi} \cos(nx) \cos(kx) dx = \begin{cases} 1 & n = k \geq 1 \\ 2 & n = k = 0 \\ 0 & \text{otherwise} \end{cases} \quad (3.5)$$

$$\frac{1}{\pi} \int_{-\pi}^{\pi} \sin(nx) \sin(kx) dx = \begin{cases} 1 & \text{if } n = k \geq 1 \\ 0 & \text{otherwise} \end{cases} \quad (3.6)$$

$$\frac{1}{\pi} \int_{-\pi}^{\pi} \cos(nx) \sin(kx) dx = 0 \quad \text{for all integers } n, k \quad (3.7)$$

Proof. In order to prove the assertions 3.5 and 3.6, we use the following remark from trigonometry,

$$\cos((n+k)x) = \cos(nx) \cos(kx) - \sin(nx) \sin(kx) \quad (3.8)$$

By using this equality we have,

$$\int_{-\pi}^{\pi} \cos(nx) \cos(kx) dx = \frac{1}{2} \int_{-\pi}^{\pi} [\cos((n+k)x) + \cos((n-k)x)] dx$$

So, for $n \neq k$, we have

$$\frac{1}{\pi} \int_{-\pi}^{\pi} \cos(nx) \cos(kx) dx = \frac{1}{2\pi} \left[\frac{\sin((n+k)x)}{n+k} + \frac{\sin((n-k)x)}{n-k} \right] \Big|_{-\pi}^{\pi} = 0$$

for $n = k \geq 1$, we have

$$\frac{1}{\pi} \int_{-\pi}^{\pi} \cos^2(nx) dx = \frac{1}{\pi} \int_{-\pi}^{\pi} \frac{1}{2} (1 + \cos(2nx)) dx = \frac{1}{2\pi} \left[x + \frac{\sin(2nx)}{2n} \right] \Big|_{-\pi}^{\pi} = \frac{2\pi}{2\pi} = 1$$

and for $n = k = 0$, we have

$$\frac{1}{\pi} \int_{-\pi}^{\pi} \cos^2(0) dx = \frac{1}{\pi} \int_{-\pi}^{\pi} 1 dx = \frac{1}{\pi} x \Big|_{-\pi}^{\pi} = \frac{2\pi}{\pi} = 2$$

Next, we prove 3.6. Again, using 3.8, we have

$$\int_{-\pi}^{\pi} \sin(nx) \sin(kx) dx = \frac{1}{2} \int_{-\pi}^{\pi} [\cos((n-k)x) - \cos((n+k)x)] dx$$

So, for $n = k \geq 1$, we have

$$\frac{1}{\pi} \int_{-\pi}^{\pi} \sin^2(nx) dx = \frac{1}{\pi} \int_{-\pi}^{\pi} \frac{1}{2} (1 - \cos(2nx)) dx = \frac{1}{2\pi} \left[x - \frac{\sin(2nx)}{2n} \right] \Big|_{-\pi}^{\pi} = \frac{2\pi}{2\pi} = 1$$

for $n \neq k$, we have

$$\frac{1}{\pi} \int_{-\pi}^{\pi} \sin(nx) \sin(kx) dx = \frac{-1}{2\pi} \left[\frac{\sin((n+k)x)}{n+k} - \frac{\sin((n-k)x)}{n-k} \right] \Big|_{-\pi}^{\pi} = 0$$

and for $n = k = 0$, we have

$$\frac{1}{\pi} \int_{-\pi}^{\pi} \sin^2(0) dx = \frac{1}{\pi} \int_{-\pi}^{\pi} 0 dx = 0$$

3.7 follows from the fact that $\frac{1}{\pi} \int_{-\pi}^{\pi} \cos(nx) \sin(kx) dx$ is an odd function and for an odd function f , we know that $\int_{-a}^a f(x) dx = 0$. ■

So we have proved that the basis for a *Fourier series* is orthonormal. Next, it follows that the Definition 3.1.2 holds for any interval of general length, since any interval can be represented in the form $\left[-\frac{\pi}{a} + c, \frac{\pi}{a} + c\right]$ where $c, a \in \mathbb{R}$. That's because, we set the building blocks as $\cos(c + n\pi x/a)$ and $\sin(c + n\pi x/a)$ which have a period of $2a$ and Theorem 3.1.3 holds for the orthonormal building blocks.

An alternative way of representing the *Fourier series* in 3.1 is the exponential form which is,

$$\sum_{k \in \mathbb{Z}} \alpha_k [f] e^{ikt} \tag{3.9}$$

where the k^{th} *Fourier Coefficient* α_k is defined to be,

$$\alpha_k = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(t) e^{ikt} dt, \quad k \in \mathbb{Z} \tag{3.10}$$

Since the domain of integration is finite, we may consider $\mathcal{F}_k[f]$ as the *finite Fourier transform* of f evaluated at k . If we extend our domain and generalize the basis functions by an index parameter, w , then the *continuous Fourier transform* of f becomes

$$\mathcal{F}_w[f] = \hat{f}(w) = \int_{-\infty}^{+\infty} f(t) e^{-iwt} dt \tag{3.11}$$

and the *inverse continuous Fourier transform* becomes

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \hat{f}(w) e^{iwt} dw \quad (3.12)$$

Note that the factor $\frac{1}{2\pi}$ may be bundled with the *inversion formula* as done in 3.12 or $\frac{1}{\sqrt{2\pi}}$ can appear in both transform and inverse transform equations to provide a symmetric appearance. For a more detailed explanation of the exponential form, one can refer to [Bachman 2000] section 5.4 or [Boggess 2001] section 1.2.

Since the *Fourier transform* takes the whole real line into consideration, as mentioned earlier, it fails to capture some events in non-stationary *signals* such as financial time series. Therefore the concept of filtering is introduced for aid. In order to utilize filters, the concept of *convolution* is used, which is

$$f \star g(t) = \int_{-\infty}^{+\infty} f(t-x) g(x) dx \quad (3.13)$$

where f and g are both *signals* and g is a linear filter component in our case. The *continuous Fourier transform* of a convolution has the nice property,

$$\mathcal{F}_w[f \star g] = \int_{-\infty}^{+\infty} f \star g(t) e^{-iwt} dt = \hat{f}(w)\hat{g}(w) \quad (3.14)$$

so we may consider *convolution* as a linear filter. A good example is the frequency response function (see [Gençay 2002] chapter 2) which is the Fourier transform of the impulse response function, where the lagged values of output and the input are employed to determine the current value of output. One who is interested in the filters may refer to [Gençay 2002] for a wide coverage.

Continuous Fourier transform is a useful technique when analyzing continuous *signals* but the signal is discrete most of the time. Therefore we need to employ a discrete version of the Fourier Transform. We begin with approximating the coefficients of a *Fourier Series* for a function f by using the trapezoidal rule for approximating integrals.

Lemma 3.1.4 *The k^{th} complex Fourier coefficient α_k of the Fourier series of a function f can be approximated by*

$$\alpha_k \approx \frac{1}{n} \sum_{j=0}^{n-1} f\left(\frac{2\pi j}{n}\right) e^{\frac{-2\pi i j k}{n}}$$

where n is the number of the partitions over the interval $[0, 2\pi]$

Proof. We set the step size $h = \frac{2\pi}{n}$ and use the trapezoidal rule to approximate the integral $\frac{1}{2\pi} \int_0^{2\pi} g(t)dt$ for any 2π -periodic function $g(t)$.

$$\frac{1}{2\pi} \int_0^{2\pi} g(t)dt \approx \frac{1}{2\pi} \frac{2\pi}{n} \left[\frac{Y_0}{2} + Y_1 + \dots + Y_{n-1} + \frac{Y_n}{2} \right] = \frac{1}{n} \sum_{j=0}^{n-1} Y_j$$

where $Y_j = g\left(\frac{2\pi j}{n}\right)$ for $j = 0 \dots n$. Note that, since the function $g(t)$ is 2π -periodic, we have $Y(0) = Y_n$. Therefore, we have,

$$\alpha_k = \frac{1}{2\pi} \int_0^{2\pi} f(t) e^{-ikt} \approx \frac{1}{n} \sum_{j=0}^{n-1} f\left(\frac{2\pi j}{n}\right) e^{-\frac{2\pi ijk}{n}}$$

■

So, as we have the coefficients of the *Fourier series*, the *discrete Fourier transform* of an n -periodic sequence of complex numbers, f , becomes,

$$\mathcal{D}_k [f] = \sum_{t=0}^{n-1} f(t) e^{-\frac{i2\pi tk}{n}} \quad (3.15)$$

and the *inverse discrete Fourier transform* becomes

$$f_t = \frac{1}{n} \sum_{k=0}^{n-1} \mathcal{D}_k [f] e^{\frac{i2\pi tk}{n}} \quad (3.16)$$

Again, when it comes to filtering, analogous to continuous case, we introduce *discrete convolution*, which is

$$f \star g_t = \sum_{u=0}^{n-1} f_u g_{t-u} \quad (3.17)$$

where f and g are both discrete signals and g is a linear filter component in our case.

As it is in the continuous case, *convolution* acts as a linear filter since we have,

$$\mathcal{D}_k [f \star g] = \sum_{t=0}^{n-1} f \star g_t e^{-\frac{i2\pi tk}{n}} = \mathcal{D}_k [f] \mathcal{D}_k [g]$$

As it is asserted in [Percival 2000], we have

$$f \star g_t = \sum_{u=0}^{n-1} f_u g_{t-u \bmod n}, \quad t = 0, \dots, n-1$$

since g must be periodic. The *discrete convolution* may also be written in the matrix form as follows.

$$\begin{bmatrix} f \star g_0 \\ f \star g_1 \\ \vdots \\ f \star g_{n-1} \end{bmatrix} = \begin{bmatrix} g_0 & g_{n-1} & g_{n-1} & \cdots & g_1 \\ g_1 & g_0 & g_{n-2} & \cdots & g_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ g_{n-1} & g_{n-2} & g_{n-3} & \cdots & g_0 \end{bmatrix} \begin{bmatrix} f_0 \\ f_1 \\ \vdots \\ f_{n-1} \end{bmatrix}$$

where the $n \times n$ matrix is the circularly shifted version of a single filtering vector. The drawback of such an algorithm is the number of multiplications needed for computation, which is n^2 . However, for $n = 2^k, k \in \mathbb{N}$, the *fast Fourier transform*, which is a modification of the *Fourier transform*, factorizes the square matrix and total cost of the multiplication reduces to $n \log_2 n$.

We did not go in too much detail with the *Fourier Transform*, but made an overview instead, since, this much detail is sufficient in order to construct a base for the fundamentals of *wavelets* and *multiresolution analysis* but the interested reader may refer to [Boggess 2001], [Gençay 2002] or [Bachman 2000] as stated before.

3.2 Wavelet Theory

As we have mentioned previously, the main idea behind the introduction of *wavelet analysis* is to make use of functions other than periodic sinusoidal functions to approximate a function, which has the drawback of being inefficient in capturing the effects, that are mentioned in Chapters 1 and 2. We define wavelets as,

Definition 3.2.1 (Wavelet) *A wavelet, ψ , is a function which satisfies the two conditions*

$$\int_{-\infty}^{\infty} \psi(t) dt = 0 \quad (3.18)$$

$$\int_{-\infty}^{\infty} |\psi(t)|^2 dt = 1 \quad (3.19)$$

which means that the wavelets have zero average and unit energy.

The above equations (3.18 and 3.19) guarantee that a wavelet has non-zero entries which cancel out, which is a standard by [Morlet 1984] who also introduced a typical example, *Morlet wavelet*. A bunch of wavelets in this manner has been introduced since then. The morlet wavelet is defined as,

$$\psi^{Morlet}(t) = \frac{1}{\sqrt{2\pi}} e^{-iwt} e^{-\frac{t^2}{2}}$$

(See Figure 3.3 (a)). Another example of a typical wavelet is the *Mexican hat wavelet* which is the negative normalized second derivative of a *Gaussian function*. The *Mexican hat wavelet* is defined as,

$$\psi^{MexicanHat}(t) = \frac{1}{\sqrt{2\pi}\sigma^3} \left(1 - \frac{t^2}{\sigma^2}\right) e^{-\frac{t^2}{2\sigma^2}}$$

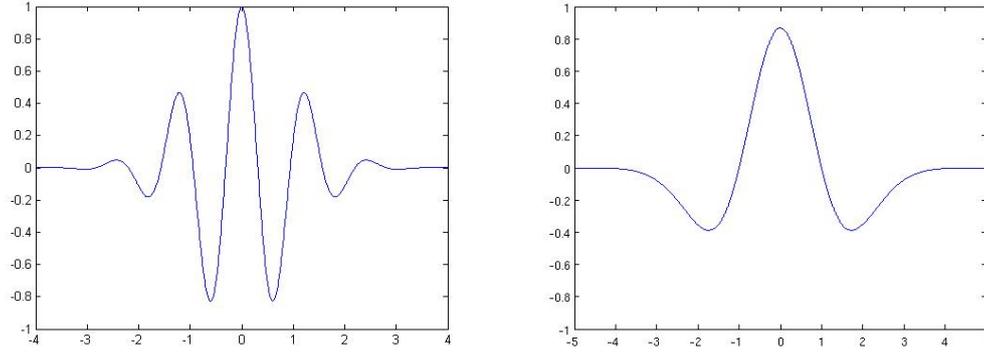


Figure 3.3: (a) Real portion of the Morlet wavelet (b) Mexican hat wavelet

(See Figure 3.3 (b)).

Two very important concepts coming along with the concept of *wavelets* are *dilation* and *translation*. That's the way we stretch and shift a *mother wavelet* and make it suit the approximated function.

Definition 3.2.2 (Translation) Let $f(t)$ be a square-integrable function (See Definition 2.1.3). The translation of $f(t)$ by $k \in \mathbb{R}$

$$T_k(f)(t) = f(t - k) \quad (3.20)$$

Definition 3.2.3 ((Dyadic) Dilation) Let $f(t)$ be a square-integrable function. The (dyadic)² dilation of $f(t)$ by $j \in \mathbb{Z}$ is

$$J_j(f)(t) = f(2^j t) \quad (3.21)$$

The *dilation* and *translation* operators are invertible (i.e. We have $T_k^{-1} = T_{-k}$ and $J_j^{-1} = J_{-j}$). Moreover, for a square integrable function $f(t)$ and integers k and j , $T_k(f)(t)$ and $2^{\frac{j}{2}} J_j(f)(t)$ are isometries since $\|f(t)\|^2 = \|T_k(f)(t)\|^2$ and $\|f(t)\|^2 = \left\| 2^{\frac{j}{2}} J_j(f)(t) \right\|^2$ (i.e. they are energy preserving)(See also [Wojtaszczyk 1997] Chapter 2).

² Apparently, as it is for the *translation* operator, the actual *dilation* operator is also valid for any $r \in \mathbb{R}$. But we will not need this, so for the sake of simplicity in formulation and calculation, we use a special form of *dilation*, namely *dyadic dilation*

So we use $T_k(f)(t)$ and $2^{\frac{j}{2}}J_j(f)(t)$ and denote the j^{th} level dilation and k^{th} level translation of $f(t)$ as

$$f_{j,k}(t) = 2^{-\frac{j}{2}}f(2^{-j}t - k) \quad (3.22)$$

Now that we have defined our operators on wavelets we may define *continuous* and *discrete wavelet transforms*. Being a well-localized tool, *wavelet transform* is especially suited for the analysis of signals that show remarkable changes in pattern on the contrary to the *Fourier transform* which extracts the global properties of a signal.

The *continuous wavelet transform* is the projection of a square-integrable function f onto a *mother wavelet*³, $\psi(t)$. It is given by,

$$(W_\psi f)(j, k) = \int_{-\infty}^{\infty} f(t) \overline{\psi_{j,k}(t)} dt \quad (3.23)$$

where

$$\psi_{j,k}(t) = \frac{1}{\sqrt{j}}\psi\left(\frac{t-k}{j}\right) \quad (3.24)$$

is the *dilated* and *translated* version of the *wavelet* and $\overline{\psi_{j,k}(t)}$ is the complex conjugate of $\psi_{j,k}(t)$ (See [Bachman 2000]).

The *continuous wavelet transform* takes a signal that varies with a single parameter, t and turns that signal into coefficients that depend on two parameters, j and k . The parameter j is important in suiting the frequency domain because, as j increases, the *wavelet* shows a *low frequency* behaviour and as j decreases, the *wavelet* shows a *high frequency* behaviour. On the other hand, the parameter k is important in suiting the time domain. In other words, parameter k , helps the *wavelet* slide over the time scale. Therefore wavelets provide good localization both in time and frequency domains.

The drawback of the transform is the load caused by a huge bunch of coefficients. But if one remembers the *Fourier Transform* which depends only on one parameter, k , and extracts only *the global properties of a signal* in time domain, it is obvious that *continuous wavelet transform* suits the signal in a much better way.

The existence of the *inverse continuous wavelet transform* is guaranteed only if the square-integrable *wavelet*, ψ , satisfies *admissibility condition* which is proposed

³ the term mother wavelet will be defined at the end of the section

by [Mallat 1998]. The condition is,

$$C_\psi = \int_0^\infty \frac{\mathcal{F}_w |\psi(t)|}{w} dw < \infty \quad (3.25)$$

where $\mathcal{F}_w |\psi(t)|$ is the *Fourier transform* of the wavelet. Since there is w in the lower part of the fraction, one may assert that $\mathcal{F}_w |\psi(0)| = 0$ which also means that, $\int_{-\infty}^\infty \psi(t) dt = 0$. From the other direction, if 3.18 holds, and for some $a < \infty$, $\int_{-\infty}^\infty (1 + |t|^a) |\psi(t)| dt < \infty$ holds, then the *admissibility condition* also holds. So, *continuous wavelet transform* becomes energy preserving, by

$$\int_{-\infty}^\infty |f(t)|^2 dt = \frac{1}{C_\psi} \int_{-\infty}^\infty \int_{-\infty}^\infty |(W_\psi f)(j, k)|^2 \frac{dj}{j^2} dk \quad (3.26)$$

Therefore, *inverse continuous wavelet transform* becomes,

$$f(t) = \frac{1}{C_\psi} \int_0^\infty \int_{-\infty}^\infty |(W_\psi f)(j, k)| \psi_{j,k}(t) \frac{dj}{j^2} dk \quad (3.27)$$

which means that, we can reconstruct $f(t)$ from the corresponding wavelet coefficients of $(W_\psi f)(j, k)$.

As we mentioned earlier, although *continuous wavelet transform* is highly redundant, it comes with a heavy burden caused by a huge bulk of coefficients. However, by a clever discretization of the *continuous wavelet transform*, one can reduce the number of coefficients, without loss of any information. Consider the *continuous wavelet transform* (3.23). If we let j and k assume only the discrete values,

$$j = j_0^s, \quad k = uk_0 j_0^s, \quad s, u \in \mathbb{Z} \quad (3.28)$$

we obtain the *discrete wavelet transform*

$$(W_\psi f)(j, k) = (W_\psi f)(j_0^s, uk_0 j_0^s) = j_0^{-\frac{s}{2}} \int_{-\infty}^\infty f(t) \overline{\psi(j_0^{-s}t - uk_0)} dt \quad (3.29)$$

We are therefore considering the countable set of functions,

$$j_0^{-\frac{s}{2}} \psi(j_0^{-s}t - uk_0), \quad s, u \in \mathbb{Z} \quad (3.30)$$

One assertion which is known as the *critical sampling rule* suggests that we choose $j = 2^{-s}$ and $k = u2^{-s}$. Therefore 3.29 becomes,

$$(W_\psi f)(j, k) = 2^{\frac{s}{2}} \int_{-\infty}^\infty f(t) \psi(2^s t - u) dt \quad (3.31)$$

In literature, the transform, j, k 's of which are determined according to the *critical sampling rule* is generally named as the *ordinary discrete wavelet transform*. There

are also other possibilities. For instance, the *maximal overlap discrete wavelet transform* suggests that we choose $j = 2^{-s}$ and $k = u$. Apparently, the choice of j, k 's cause structural changes in the transform. One may find more detail in [Gençay 2002].

The *critical sampling* yields the functions $\psi_{j,k}(t) = 2^{-\frac{j}{2}}\psi(2^{-j}t - k)$ and the set of *discrete wavelet transform coefficients* $(W_\psi f)(j, k)$ for a square-integrable function f . The functions $\psi_{j,k}$ form an orthonormal system and such an orthonormal system is called a *wavelet basis*. Now we can define the constructors of an orthonormal basis.

Definition 3.2.4 (Mother Wavelet) *The function ψ whose dilations and transformations (wavelets) form an orthonormal basis is called the mother wavelet. In other words, considering the critical sampling rule, if the family of the functions*

$$\psi_{j,k}(t) = 2^{\frac{j}{2}}\psi(2^j t - k), \quad j, k \in \mathbb{Z} \quad (3.32)$$

called wavelets, form an orthonormal basis, then the function ψ is called the mother wavelet.

We will define what we mean by *forming an orthonormal basis*, widely at the end of the section (See Theorem 3.2.5). Yet it is enough to continue, if it just makes some sense. One classical example is the *Haar basis*. It is not only the oldest ⁴ *wavelet basis*, but it is also the simplest one. The functions that form the Haar basis are constructed from the *Haar wavelet*, which is given by

$$\psi^{Haar}(t) = \begin{cases} 1 & \text{on } [0, \frac{1}{2}) \\ -1 & \text{on } [\frac{1}{2}, 1) \\ 0 & \text{otherwise} \end{cases} \quad (3.33)$$

(See Figure 3.4 (a)).

If we make a *wavelet analysis* using the *Haar basis* as an example, the concept might be easier to visualize. Sadly, we begin the construction by introducing some more concepts. We will come back to the *mother wavelet* later.

If the *mother* of our *orthonormal basis* is the *Haar Wavelet*, ψ^{Haar} , then the *father* is definitely the *Haar scaling function* ⁵, ϕ^{Haar} . From this point on, we will be using ψ and ϕ instead. These two generate a *family* of functions that can be used to decompose

⁴ The Haar wavelet is discovered by Alfred Haar in the early 1900's

⁵ It is indeed named as the father wavelet in some references

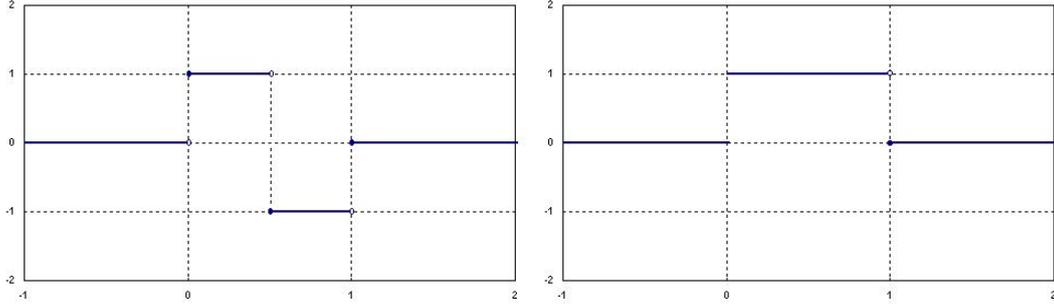


Figure 3.4: (a) Haar wavelet (b) Haar scaling function

or reconstruct a signal. The *Haar scaling function* is given by

$$\phi^{Haar}(t) = \begin{cases} 1 & \text{on } [0, 1) \\ 0 & \text{otherwise} \end{cases} \quad (3.34)$$

(See Figure 3.4 (b)). Apparently, we may translate ϕ as we like. So, let V_0 be a space of all functions of the form

$$\sum_{k \in \mathbb{Z}} a_k \phi(t - k), \quad a_k \in \mathbb{R} \quad (3.35)$$

where k can range over any finite set of positive or negative integers. A typical element of V_0 would be like,

$$f(t) = \phi(t + 1) + 4\phi(t) - 2\phi(t - 1) - 2\phi(t - 2) \in V_0$$

whose graph is shown in Figure 3.5.

The function $f(t)$ has discontinuities at $t = -1, 0, 1$ and 3 . So a frequency of 1 fits this signal. But for analyzing the signals with higher frequency, we need smaller building blocks. So we dilate our scaling function by j and step into V_j , which is the space of all functions of the form

$$\sum_{k \in \mathbb{Z}} a_k \phi(2^j t - k), \quad a_k \in \mathbb{R} \quad (3.36)$$

whose discontinuities are contained in the set $\{\dots, -\frac{2}{2^j}, -\frac{1}{2^j}, 0, \frac{1}{2^j}, \frac{2}{2^j}, \dots\}$. Apparently, any function contained in V_0 is also contained in V_j , which has smaller building blocks. Therefore, we have

$$V_0 \subset V_1 \subset \dots \subset V_{j-1} \subset V_j \subset V_{j+1} \subset \dots \quad (3.37)$$

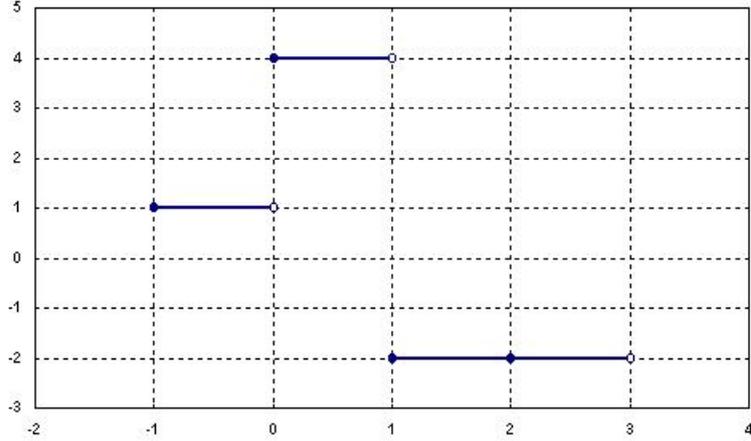


Figure 3.5: Example of a typical element of V_0

Theorem 3.2.5 *The set of functions $\{2^{\frac{j}{2}}\phi(2^j t - k)\}_{k \in \mathbb{Z}}$ is an orthonormal basis of V_j .*

Proof. Let $f_j(k) = 2^{\frac{j}{2}}\phi(2^j t - k)$, $t \in \mathbb{R}$ be a function. For $s, u \in \mathbb{Z}$, we have

$$f_j(s) = 2^{\frac{j}{2}}\phi(2^j t - s) \quad \text{and} \quad f_j(u) = 2^{\frac{j}{2}}\phi(2^j t - u)$$

So we have

$$\int_{-\infty}^{\infty} f_j(s) f_j(u) dt = 2^j \int_{-\infty}^{\infty} \phi(2^j t - s) \phi(2^j t - u) dt = 0$$

Therefore any two elements of the set $\{2^{\frac{j}{2}}\phi(2^j t - k), k \in \mathbb{Z}\}$ is orthogonal. Next, we have

$$\int_{-\infty}^{\infty} |f_j(0)|^2 dt = 2^j \int_{-\infty}^{\infty} |\phi(2^j t)|^2 dt = 1$$

So the orthonormality is proved. ■

At this point, we have building blocks of size $\frac{1}{2^j}$ which are orthogonal to any other building block in their own space V_j . But those building blocks are not necessarily orthogonal to some member of another space V_k . However, in order to obtain a linear decomposition of a signal, we need orthogonal building blocks of different spaces, so that the projection of the signal onto those components mean anything. So we need the *orthogonal complement* of some V_j in V_{j+1} , which we denote by W_j .

Not surprisingly, the *Haar wavelet* is a member of, W_1 , the *orthogonal complement* of V_0 in V_1 . Since it is also a member of the space V_1 , it can be constructed by

$a_k \phi(2t - k)$'s. Actually, the *Haar wavelet* can be redefined as,

$$\psi^{Haar}(t) = \phi^{Haar}(2t) - \phi^{Haar}(2t - 1) \quad (3.38)$$

Moreover, next lemma asserts that, any function that is a member of W_1 can be constructed by using the translations of the *Haar Wavelet*.

Lemma 3.2.6 *Let f_1 be any given function in V_1 .*

$$f_1 = \sum_k a_k \phi(2t - k) \in V_1$$

f_1 is orthogonal to V_0 iff it can be represented as

$$f_1 = \sum_{k \in \mathbb{Z}} a_{2k} \psi(t - k)$$

Proof. f_1 is orthogonal to V_0 means that it is orthogonal to each $\phi(t - l), l \in \mathbb{Z}$. We know that, any function in V_j is contained in V_l , if $l \geq j$. So, from orthogonality condition we have,

$$\begin{aligned} 0 &= \underbrace{\dots}_0 + \underbrace{\phi(t-l) a_{2l-1} \phi(2t-2l+1) + \phi(t-l) a_{2l} \phi(2t-2l)}_0 \\ &\quad + \phi(t-l) a_{2l+1} \phi(2t-2l-1) + \underbrace{\phi(t-l) a_{2l+2} \phi(2t-2l-2)}_0 + \underbrace{\dots}_0 \\ &= \phi(t-l) a_{2l} \phi(2t-2l) + \phi(t-l) a_{2l+1} \phi(2t-2l-1) \end{aligned}$$

which is only possible if $a_{2l} = -a_{2l+1}$. Then,

$$f_1 = \sum_{k \in \mathbb{Z}} a_{2k} (\phi(2t-2k) - \phi(2t-2k-1)) = \sum_{k \in \mathbb{Z}} a_{2k} \psi(t-k)$$

■

Now that we have proved that the *orthogonal complement* of V_0 can be constructed by translations of the *Haar Wavelet*, we may define W_0 as the space of all functions of the form.

$$\sum_{k \in \mathbb{Z}} a_k \psi(t - k), \quad a_k \in \mathbb{R} \quad (3.39)$$

Moreover, since W_0 is the *orthogonal complement* of V_0 in V_1 , we have $V_0 = V_1 \oplus W_0$, which means that any function in V_0 can be constructed as a linear sum of its projections over V_1 and W_1 .

In a similar manner, we may assert that

$$\begin{aligned}
V_j &= W_{j-1} \oplus V_{j-1} \\
&= W_{j-1} \oplus W_{j-2} \oplus V_{j-2} \\
&= W_{j-1} \oplus W_{j-2} \oplus \cdots \oplus W_1 \oplus W_0 \oplus V_0
\end{aligned} \tag{3.40}$$

which also means that any f_j in V_j can be decomposed uniquely as a linear sum

$$\begin{aligned}
f_j &= w_{j-1} + f_{j-1} \\
f_j &= w_{j-1} + w_{j-2} + f_{j-2} \\
f_j &= w_{j-1} + w_{j-2} + \dots + w_1 + w_0 + f_0
\end{aligned} \tag{3.41}$$

where each function w_l belongs to W_l where $0 \leq l \leq j - 1$ (See [Boggess 2001] Theorem 4.8). For *Haar Decomposition* and *Haar Reconstruction* algorithms, one may see [Boggess 2001].

Now that we know what a *mother wavelet*, a *scaling function* and a *wavelet basis* is, we may start explaining how to obtain it by *multiresolution analysis*.

3.3 Multiresolution Analysis

In the previous section, we have seen how the *wavelets* are used to decompose a *finite energy function* (See 3.19) with respect to a resolution (time-scale). We have seen wavelets, acting as linear filters. Throughout this section, we will be introducing the key concepts of a *multiresolution analysis*, and make the fundamentals for going in details with the construction of *Daubechies wavelet family* which are compactly supported, in the next section. We will use this family also in our model.

The drawback with the *Haar decomposition* is that, both the *mother wavelet*, ψ , and the *scaling function*, ϕ , are discontinuous. Therefore, it can only provide crude approximations to a continuously varying signal. So we need a theory which is similar to the one in the previous section, but with a continuous *mother wavelet*, ψ , and a *scaling function*, ϕ . The resulting framework is presented by S.G.Mallat (See [Mallat 1989] and [Mallat 1998]).

Definition 3.3.1 (Multiresolution Analysis) *Let, $V_j, j \in \mathbb{Z}$ be a sequence of subspaces of square integrable functions. The collection of spaces $\{V_j\}_{j \in \mathbb{Z}}$ is called a*

multiresolution analysis with scaling function, ϕ if the following conditions hold:

1. (*nested*) $V_j \subset V_{j+1}$
2. (*density*) $\overline{\cup V_j}$ spans the space of square integrable functions
3. (*seperation*) $\cap V_j = \{0\}$
4. (*scaling*) The function, $f(t)$, belongs to V_j if and only if the function $f(2^{-j}t)$ belongs to V_0
5. (*orthonormal basis*) The scaling function, ϕ , belongs to V_0 and the set $\{\phi(t-k)\}_{k \in \mathbb{Z}}$ is an orthonormal basis for V_j

We know what a V_j is, from the previous section. Here, we name it as an *approximation space*. Our choice of scaling function, ϕ , determines the structure of our *approximation spaces* and therefore the *multiresolution analysis*. Since the decomposition of the signal is unique (see 3.40), every single choice of ϕ leads to a completely different *multiresolution analysis*. Most desirable scaling functions are the ones that have *compact support*, which means that $\phi(t)$ is identically zero outside a finite interval (See [Boggess 2001] Chapter 0). They are desirable because, they lower the computation load pretty much and better fit the analyses.

Haar scaling function is a good example to a compactly supported scaling function (See Figure 3.4 (b)). But the drawback is, it is not continuous. The *Daubechies wavelet family*, as we will see through the next section is not only *compactly supported*, but also continuous.

From Theorem 3.2.5, we know that the (dyadic) dilations and translations of the scaling function, that is $\{\phi_{jk}(t) = 2^{\frac{j}{2}}\phi(2^j t - k)\}_{k \in \mathbb{Z}}$ form an orthonormal basis of V_j . In the theorem, we define the constraints for the coefficients of the *scaling relation* between V_j 's.

Theorem 3.3.2 Let $\{V_j\}_{j \in \mathbb{Z}}$ be a multiresolution analysis with scaling function, ϕ . Then the following scaling relation holds:

$$\phi(t) = \sum_{k \in \mathbb{Z}} p_k \phi(2t - k), \quad \text{where} \quad p_k = 2 \int_{-\infty}^{\infty} \phi(t) \overline{\phi(2t - k)} dt$$

where $\overline{\phi(2t - k)}$ is the complex conjugate of $\phi(2t - k)$. Moreover, we also have,

$$\phi(2^{j-1}t - l) = \sum_{k \in \mathbb{Z}} p_{k-2l} \phi(2^j t - k)$$

or equivalently

$$\phi_{j-1,l} = 2^{-\frac{1}{2}} \sum_k p_{k-2l} \phi_{j,k}, \quad \text{where} \quad \phi_{j,k} = 2^{\frac{j}{2}} \phi(2^j t - k)$$

Proof. From 3.37, we have $\phi(t) \in V_0 \subset V_1$, where V_1 is a linear span of $\{\phi_{1k}\}_{k \in \mathbb{Z}}$. Therefore, for some choice of \tilde{p}_k , the equation $\phi(t) = \sum \tilde{p}_k \phi_{1,k}(t)$ must hold. From the orthonormality condition, in the space of square integrable functions we have,

$$\tilde{p}_k = \langle \phi, \phi_{1,k} \rangle = 2^{\frac{1}{2}} \int_{-\infty}^{\infty} \phi(t) \overline{\phi(2t - k)} dt$$

Therefore,

$$\phi(t) = \sum_{k \in \mathbb{Z}} \tilde{p}_k \phi_{1,k}(t) = \sum_{k \in \mathbb{Z}} \tilde{p}_k 2^{\frac{1}{2}} \phi(2t - k)$$

If we let $p_k = 2^{\frac{1}{2}} \tilde{p}_k$, then we have

$$\phi(t) = \sum_{k \in \mathbb{Z}} p_k \phi(2t - k)$$

So first equation is proved. If we replace t by $2^{j-1}t - l$ in the equation, we get,

$$\phi(2^{j-1}t - l) = \sum_{k \in \mathbb{Z}} p_k \phi(2^j t + 2l - k)$$

and then rescale by setting $k = k - 2l$ we get

$$\phi(2^{j-1}t - l) = \sum_{k \in \mathbb{Z}} p_{k-2l} \phi(2^j t - k)$$

and this proves the second equation simplified version of which (by multiplying by $2^{\frac{j-1}{2}}$) is given in the third equation. ■

So we have given our constraints. As an example, if we recall the orthonormal *Haar system*, values of the p_k 's are $p_0 = p_1 = 1$ and the rest of the p_k 's are zero. The results from this theorem even narrows our constraints, which are:

1. $\sum_{k \in \mathbb{Z}} p_{k-2l} \overline{p_k} = 2\delta_{l0}$
2. $\sum_{k \in \mathbb{Z}} |p_k|^2 = 2$
3. $\sum_{k \in \mathbb{Z}} p_k = 2$
4. $\sum_{k \in \mathbb{Z}} p_{2k} = 1$ and $\sum_{k \in \mathbb{Z}} p_{2k+1} = 1$

Recall that, for orthonormal bases,

$$\delta_{kl} = \begin{cases} 1 & k = l \\ 0 & \text{otherwise} \end{cases}$$

So we have the relation between V_j 's. So far, we have not mentioned about the wavelets. But that's the point, where the wavelets enter the picture. Recall from the previous section that the V_j 's are not orthonormal to each other. Instead, one with bigger j is the *superset* of the one with smaller j . So we extracted the orthogonal complement of V_j in V_{j+1} and represented it with W_j . Therefore we made it possible to extract a signal into a linear combination of functions. Analogously, next theorem extracts, W_j 's for a multiresolution analysis (One may also see [Boggess 2001]).

Theorem 3.3.3 Let $\{V_j\}_{j \in \mathbb{Z}}$ be a multiresolution analysis with scaling function, $\phi(t) = \sum_k p_k \phi(2t - k)$, where the coefficients, p_k , follow Theorem 3.3.2. Also let W_j be the span of $\{\psi(2^{j-t-k})\}_{k \in \mathbb{Z}}$, where

$$\psi(t) = \sum_{k \in \mathbb{Z}} (-1)^k \overline{p_{1-k}} \phi(2t - k) \quad (3.42)$$

Then $W_j \subset V_{j+1}$ is the orthogonal complement of V_j in V_{j+1} . Furthermore, we have, $\{\psi_{j,k}(t) = 2^{\frac{j}{2}} \psi(2^j t - k)\}_{k \in \mathbb{Z}}$ is an orthonormal basis for the W_j .

Proof. We first begin with proving the theorem for $j = 0$. In order to show that the set $\{\psi_{0,k}(t) = \psi(t - k)\}_{k \in \mathbb{Z}}$ is orthonormal, using the fact that the set $\{2^{\frac{1}{2}} \phi(2t - k)\}_{k \in \mathbb{Z}}$ is orthonormal from Theorem 3.2.5, by changing the summation indices, we have

$$\langle \psi_{0,m}, \psi_{0,l} \rangle = \frac{1}{2} \sum_{k \in \mathbb{Z}} \overline{p_{1-k+2m}} p_{1-k+2l}$$

If we make a change of index $\tilde{k} = 1 - k + 2m$, by using the constraint 1 we get

$$\langle \psi_{0,m}, \psi_{0,l} \rangle = \frac{1}{2} \sum_{\tilde{k} \in \mathbb{Z}} \overline{p_{\tilde{k}}} p_{\tilde{k}+2l-2m} = \delta_{m-l,0} = \delta_{m,l}$$

Therefore $\psi_{0,m}$ has to be orthonormal. Next, we show that $\{\psi_{0,m}(t) = \psi(t - m)\}_{m \in \mathbb{Z}}$ is orthogonal to V_0 , which means that $\psi(t - m)$ is orthogonal to $\phi(t - l)$ for all $l \in \mathbb{Z}$ since $\{\phi(t - l)\}_{l \in \mathbb{Z}}$ spans V_0 . Again, by using Theorem 3.2.5 and changing the summation indices, we have

$$\langle \psi_{0,m}, \psi_{0,l} \rangle = \frac{1}{2} \sum_{k \in \mathbb{Z}} (-1)^k p_{1-k+2m} p_{k-2l}$$

If we prove that the sum is equal to zero, we are done with V_0 . When $l = m = 0$ we have,

$$\sum_{k \in \mathbb{Z}} (-1)^k p_{1-k} p_k = \dots - p_2 p_{-1} + p_1 p_0 - p_0 p_1 + p_{-1} p_2 - \dots = 0$$

If $j \geq 0$, we extend this to the general case by noticing that the term in the sum with the index $k = l + m - j$, cancels out with the term with the index $k = l + m + j + 1$ since

$$(-1)^{l+m+j+1} p_{m-l-j} p_{m+j+1-l} = -(-1)^{l+m-j} p_{1-l+m+j} p_{m-l-j}$$

Therefore the sum is zero. So we have the set $\{\psi_{0,m}(t) = \psi(t - m)\}_{m \in \mathbb{Z}}$ is orthonormal.

■

By this theorem (Theorem 3.3.3), the set $\{\psi_{j-1,k}\}_{k \in \mathbb{Z}}$ is an orthonormal basis for the space W_{j-1} , which is the orthogonal complement of V_{j-1} in V_j . Therefore, we have

$$\begin{aligned} V_j &= W_{j-1} \oplus V_{j-1} \\ &= W_{j-1} \oplus W_{j-2} \oplus V_{j-2} \\ &= W_{j-1} \oplus W_{j-2} \oplus \cdots \oplus W_1 \oplus W_0 \oplus V_0 \end{aligned} \quad (3.43)$$

and therefore

$$\begin{aligned} f_j &= w_{j-1} + f_{j-1} \\ f_j &= w_{j-1} + w_{j-2} + f_{j-2} \\ f_j &= w_{j-1} + w_{j-2} + \cdots + w_1 + w_0 + f_0 \end{aligned} \quad (3.44)$$

just like it is in the *Haar system* (See 3.40 and 3.41)

Decomposition and reconstruction of a signal

Now that we have constructed our basis for a *multiresolution analysis*, we can continue with the *decomposition* and *reconstruction* algorithms of a multiresolution analysis. Let f be a *signal* that is already contained in one of our approximation spaces, V_j . Apparently, if the *signal* has higher frequencies, j should be greater as well. The reader would remember from the previous and current sections that, in order to decompose the signal into its different resolution component, there are two orthonormal bases regarding the approximation space V_j , that are V_{j-1} and W_{j-1} , which is, the orthonormal complement of V_{j-1} in V_j . So we have $V_j = W_{j-1} \oplus V_{j-1}$ as orthogonal direct sum of those bases.

So we have from Theorem 3.2.5 that f can be represented as the linear sum of its projections over the orthonormal basis $\{\phi_{j,k}\}_{k \in \mathbb{Z}}$ that constructs the approximation

space, V_j .

$$f = \sum_{k \in \mathbb{Z}} \langle f, \phi_{j,k} \rangle \phi_{j,k} \quad (3.45)$$

Also, as we know from Theorem 3.3.3 that, every item in the basis $\{\phi_{j,k}\}_{k \in \mathbb{Z}}$ can be represented as a linear sum of the bases $\{\phi_{j-1,k}\}_{k \in \mathbb{Z}}$ and $\{\psi_{j-1,k}\}_{k \in \mathbb{Z}}$ since they span V_{j-1} and W_{j-1} respectively. So we also have,

$$f = \underbrace{\sum_{k \in \mathbb{Z}} \langle f, \phi_{j-1,k} \rangle \phi_{j-1,k}}_{f_{j-1}} + \underbrace{\sum_{k \in \mathbb{Z}} \langle f, \psi_{j-1,k} \rangle \psi_{j-1,k}}_{w_{j-1}} \quad (3.46)$$

as it is in 3.44. Our decomposition formula depends on those two processes. But there is also one remark that is to be given before introducing the formula.

Remark 3.3.4 (Parseval's Identity) *Let $\{u_k\}_{k \in \mathbb{N}}$ be an orthonormal basis and V be the complex inner product space that is spanned by $\{u_k\}_{k \in \mathbb{N}}$. Then for any two functions $f, g \in V$ that have the expansions*

$$f = \sum_{k=1}^{\infty} a_k u_k \quad \text{and} \quad g = \sum_{k=1}^{\infty} b_k u_k$$

we have

$$\begin{aligned} \langle f, g \rangle &= \left\langle \sum_{k=1}^{\infty} a_k u_k, \sum_{n=1}^{\infty} b_n u_n \right\rangle \\ &= \sum_{k=1}^{\infty} \sum_{n=1}^{\infty} a_k \bar{b}_n \langle u_k, u_n \rangle \\ &= \sum_{k=1}^{\infty} a_k \bar{b}_k \end{aligned}$$

Apparently, we also have

$$\langle f, f \rangle = \left\langle \sum_{k=1}^{\infty} a_k u_k, \sum_{n=1}^{\infty} a_n u_n \right\rangle = \sum_{k=1}^{\infty} a_k \bar{a}_k = \|f\|^2$$

First we find the coefficients relative to the basis in 3.45 and then use them to evaluate the coefficients relative to the basis in 3.46. By using the scaling relation (Theorem 3.3.2), Theorem 3.3.3 and Remark 3.3.4, we have the *decomposition formula*

$$\text{Decomposition: } \begin{cases} \langle f, \phi_{j-1,l} \rangle = 2^{-\frac{1}{2}} \sum_{k \in \mathbb{Z}} \overline{p_{k-2l}} \langle f, \phi_{j,k} \rangle \\ \langle f, \psi_{j-1,l} \rangle = 2^{-\frac{1}{2}} \sum_{k \in \mathbb{Z}} (-1)^k p_{1-k+2l} \langle f, \phi_{j,k} \rangle \end{cases} \quad (3.47)$$

We made use of the *scaling relation* while constructing the *decomposition formula*. So we need an *inverse scaling relation* for constructing the *reconstruction formula*. From the *decomposition formula*, together with the orthonormality of the basis $\{\phi_{j,k}\}$, we have $\langle \phi_{j,k}, \phi_{j-1,l} \rangle = 2^{-\frac{1}{2}} \overline{p_{k-2l}}$ and $\langle \phi_{j,k}, \psi_{j-1,l} \rangle = 2^{-\frac{1}{2}} (-1)^k \overline{p_{1-k+2l}}$. So our *inverse scaling relation* becomes

$$\phi_{j,k} = \sum_{l \in \mathbb{Z}} 2^{-\frac{1}{2}} \overline{p_{k-2l}} \phi_{j-1,l} + \sum_{l \in \mathbb{Z}} 2^{-\frac{1}{2}} (-1)^k p_{1-k+2l} \psi_{j-1,l} \quad (3.48)$$

So if we apply the Remark 3.3.4 to 3.48 and 3.46, we obtain our *reconstruction formula*.

$$\text{Reconstruction: } \begin{cases} \langle f, \phi_{j,k} \rangle = 2^{-\frac{1}{2}} \sum_{l \in \mathbb{Z}} p_{k-2l} \langle f, \phi_{j-1,l} \rangle \\ \quad + 2^{-\frac{1}{2}} \sum_{l \in \mathbb{Z}} (-1)^k \overline{p_{1-k+2l}} \langle f, \psi_{j-1,l} \rangle \end{cases} \quad (3.49)$$

As the reader should have noticed, all of the formulas above are for the *orthonormal bases*. For various reasons, one may need the *orthogonal versions* of the *decomposition* and *reconstruction* formulas. In this case, our bases for V_j and W_j become $\{\phi(2^j t - k)\}_{k \in \mathbb{Z}}$ and $\{\psi(2^j t - k)\}_{k \in \mathbb{Z}}$ respectively (rather than $\{\phi_{j,k}(t) = 2^{j/2} \phi(2^j t - k)\}_{k \in \mathbb{Z}}$ and $\{\psi_{j,k}(t) = 2^{j/2} \psi(2^j t - k)\}_{k \in \mathbb{Z}}$). So if we let $a_k^j = 2^{\frac{j}{2}} \langle f, \phi_{j,k} \rangle$, then 3.45 becomes

$$\begin{aligned} f &= \sum_{k \in \mathbb{Z}} \underbrace{2^{\frac{j}{2}} \langle f, \phi_{j,k} \rangle}_{a_k^j} \underbrace{2^{-\frac{j}{2}} \phi_{j,k}}_{\phi(2^j t - k)} \\ &= \sum_{k \in \mathbb{Z}} a_k^j \phi(2^j t - k) \end{aligned} \quad (3.50)$$

Moreover, if we let $b_k^j = 2^{\frac{j}{2}} \langle f, \psi_{j,k} \rangle$, this time, we have 3.46 rewritten

$$f = \sum_{k \in \mathbb{Z}} a_k^{j-1} \phi(2^{j-1} t - k) + \sum_{k \in \mathbb{Z}} b_k^{j-1} \psi(2^{j-1} t - k) \quad (3.51)$$

Here, a_k^j is called the *approximation coefficient* and b_k^j is called the *detail coefficient*.

Then our *orthogonal decomposition* and *reconstruction formulas* become

$$\text{Decomposition: } \begin{cases} a_l^{j-1} = 2^{-1} \sum_{k \in \mathbb{Z}} \overline{p_{k-2l}} a_l^j \\ b_l^{j-1} = 2^{-1} \sum_{k \in \mathbb{Z}} p_{1-k+2l} a_k^j \end{cases} \quad (3.52)$$

$$\text{Reconstruction: } a_k^j = \sum_{l \in \mathbb{Z}} p_{k-2l} a_l^{j-1} + \sum_{l \in \mathbb{Z}} (-1)^k \overline{p_{1-k+2l}} b_l^{j-1} \quad (3.53)$$

Implementation

The *decomposition* and *reconstruction formulas* are not as easy as they seem when it comes to implementation. Actually we will need more theoretical help to implement the formulas. Here, since *reconstruction* is not our concern in this work, for minimalization purposes, we will only give the algorithm for *decomposition*. One, who is interested in the reconstruction algorithm may refer to [Boggess 2001], [Gençay 2002], [Wojtaszczyk 1997] or [Bachman 2000]. We will consider the *decomposition algorithm* in three major steps: *initialization*, *iteration* and *termination*.

Initialization is, as it is in many other areas, the most important part of the algorithm that breaks up a signal, f , into its W_j components. First, we have to choose the depth of our analysis, which is j . In other words, we choose the approximation space that best fits the information available on f . Clearly, it depends on the frequency of the signal and also on the choice of multiresolution analysis. Secondly we need to choose $f_j \in V_j$ to best fit the signal itself.

The best approximation to f from V_j , in the sense of energy is the orthogonal projection of the signal onto V_j , $P_j f$. Since, $2^{j/2} \phi(2^j t - k)$ is orthonormal, we have,

$$P_j f = \sum_{k \in \mathbb{Z}} a_k^j \phi(2^j t - k), \text{ where } a_k^j = 2^j \int_{-\infty}^{\infty} f(t) \overline{\phi(2^j t - k)} dt \quad (3.54)$$

In reality, the information from discrete (or sampled) data is generally not sufficient to determine the a_k^j 's. Next theorem, gives a quadrature rule that does a better approximation.

Theorem 3.3.5 *Let $\{V_j\}_{j \in \mathbb{Z}}$ be some multiresolution analysis with a compactly supported scaling function ϕ . If the square integrable function f is continuous, then, for sufficiently large j , we have,*

$$a_k^j = 2^j \int_{-\infty}^{\infty} f(t) \overline{\phi(2^j t - k)} dt \approx m f\left(\frac{k}{2^j}\right)$$

where $m = \int \overline{\phi(t)} dt$

Proof. Recall from the earlier pages of this section that a scaling function that has *compact support* is defined to be non-zero only in a finite interval of the form $\{t : |t| \leq M\}$. Therefore, the interval of integration for a_k^j in 3.54 is $\{t : |2^j t - k| \leq M\}$. So, if we change the variable as $\tilde{t} = 2^j t - k$, we get

$$a_k^j = \int_{-M}^M f(2^{-j} \tilde{t} + 2^{-j} k) \overline{\phi(\tilde{t})} d\tilde{t}$$

When j is sufficiently large, we have $2^{-j}\tilde{t} + 2^{-j}k \approx 2^{-j}k$ for $\tilde{t} \in [-M, M]$. Thus, $f(2^{-j}\tilde{t}) + f(2^{-j}k) \approx f(2^{-j}k)$ in the same interval. So we have the approximation,

$$a_k^j = \int_{-M}^M f(2^{-j}\tilde{t} + 2^{-j}k) \overline{\phi(\tilde{t})} d\tilde{t} \approx f(2^{-j}k) \int_{-M}^M \overline{\phi(\tilde{t})} d\tilde{t}$$

Since ϕ is compactly supported we have

$$\int_{-M}^M \overline{\phi(\tilde{t})} d\tilde{t} = \int_{-\infty}^{\infty} \overline{\phi(\tilde{t})} d\tilde{t} = m$$

Therefore $a_k^j \approx mf\left(\frac{k}{2^j}\right)$ ■

There is no doubt that, if we construct the *multiresolution analysis* so that the *scaling function* has the property $m = \int \phi = 1$, this reduces our computational burden. In that case, we have $a_k^j \approx f\left(\frac{k}{2^j}\right)$. Back to the general case, with $a_k^j \approx mf\left(\frac{k}{2^j}\right)$, the projection of f over V_j becomes

$$P_j f(t) \approx f_j(t) = m \sum_{k \in \mathbb{Z}} f\left(\frac{k}{2^j}\right) \phi(2^j t - k) \quad (3.55)$$

The next step in our algorithm is the *iteration*. Now that we know, we can represent f_j as the sum of some $f_{j-1} \in V_j$ and some $w_{j-1} \in W_j$ using decomposition formula in 3.47. Once we have our components, we recursively do the same thing for $f_{j-1} \in V_j$, then for $f_{j-2} \in V_j$ and so on. The process can be illustrated as,

$$\begin{array}{ccccccccc} f \approx f_j & \rightarrow & f_{j-1} & \rightarrow & f_{j-2} & \cdots & \rightarrow & f_2 & \rightarrow & f_1 & \rightarrow & f_0 \\ & & \searrow w_{j-1} & & \searrow w_{j-2} & & \cdots & \searrow w_2 & & \searrow w_1 & & \searrow w_0 \end{array}$$

Now, let us recall the *convolution* of two series $x = (\cdots, x_{-1}, x_0, x_1, \cdots)$ and $y = (\cdots, y_{-1}, y_0, y_1, \cdots)$ from Section 3.1 (3.17) which is defined as

$$x \star y_t = \sum_{k \in \mathbb{Z}} x_k y_{t-k} \quad (3.56)$$

Now, let h and l be the sequences

$$h_k = \frac{1}{2} (-1)^k p_{k+1} \quad (3.57)$$

$$l_k = \frac{1}{2} \overline{p_{-k}} \quad (3.58)$$

and H and L be two discrete filters which are defined as $H(x) = h \star x$ and $L(x) = l \star x$. If we let $x = a^j$, then $L(x)$ becomes $L(a^j)_l = 2^{-1} \sum_{k \in \mathbb{Z}} \overline{p_{k-l}} a_l^j$. Similarly, $H(x)$ becomes $H(a^j)_l = 2^{-1} \sum_{k \in \mathbb{Z}} p_{1-k+l} a_k^j$. Thus we have $a_l^{j-1} = L(a^j)_{2l}$ and $b_l^{j-1} = H(a^j)_{2l}$ (See 3.52).

There is one more concept that we have to mention about, which is indeed very important. This concept is known as *downsampling*. In cases where we lie in the boundaries of the critical sampling rule (See [Bachman 2000]), we may *downsample* the data. That means, since we can decompose the function $f_j \in V_j$ into $f_{j-1} \in V_{j-1}$ and $w_{j-1} \in W_j$ that are orthogonal to each other, we may reduce the number of samples by a factor of $\frac{1}{2}$ by cleverly selecting the subsamples. Let us give an example from the *Haar System* at this point.

Example 3.3.6 Let $f_j(t) = \sum_k a_k \phi(2^j t - k)$. If we divide the sum into its even and odd terms, we get

$$f_j(t) = \sum_k a_{2k} \phi(2^j t - 2k) + \sum_k a_{2k+1} \phi(2^j t - 2k - 1) \quad (3.59)$$

Recall that, for the Haar System, we have,

$$\begin{aligned} \phi(2^j t - 2k) &= (\phi(2^{j-1} t - k) + \psi(2^{j-1} t - k)) / 2 \\ \phi(2^j t - 2k - 1) &= (\phi(2^{j-1} t - k) - \psi(2^{j-1} t - k)) / 2 \end{aligned}$$

So if we substitute those equations into 3.59, we get

$$\begin{aligned} f_j(t) &= \sum_k a_{2k} (\phi(2^{j-1} t - k) + \psi(2^{j-1} t - k)) / 2 \\ &\quad + \sum_k a_{2k+1} (\phi(2^{j-1} t - k) - \psi(2^{j-1} t - k)) / 2 \\ &= \sum_{k \in \mathbb{Z}} \left(\frac{a_{2k} - a_{2k+1}}{2} \right) \psi(2^{j-1} t - k) + \left(\frac{a_{2k} + a_{2k+1}}{2} \right) \phi(2^{j-1} t - k) \\ &= w_{j-1} + f_{j-1} \end{aligned}$$

As can be seen from the example, we may take only the even terms and discard the odd terms. Such an operation is called *downsampling* and we denote it by,

$$(\mathcal{D}t)_l = t_{2l} \quad l \in \mathbb{Z} \quad (3.60)$$

Sometimes the *downsampling operator* can be represented as $2 \downarrow$ as well.

This completes, our *iteration algorithm*. The iterative step, using the linear filters that we have just defined is as follows.

$$a^{j-1} = \mathcal{D}L(a^j) \quad b^{j-1} = \mathcal{D}H(a^j) \quad (3.61)$$

which is also illustrated in Figure 3.6

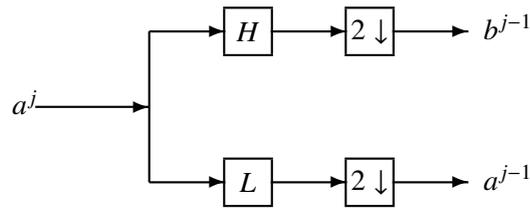


Figure 3.6: Diagram of decomposition for multiresolution analyses

The only step left to complete a successful *multiresolution analysis* is the *termination*. This of course depends on our aim for making a *multiresolution analysis*. For example for data compression purposes, one may decompose all the components of a signal, then filter and transfer the data for reconstruction. But in our case, we will decompose our signal to a level j and then stop, since our signal will be smooth enough, and we will still have a signal $f \in V_j$. And that concludes the section.

3.4 The Daubechies Wavelet Family

The decomposition algorithm discussed in section 3.3 makes multiresolution analysis an easier process, provided that a set of sampling spaces is given. But as one expects the worst case, the hardest thing is to find an appropriate scaling function that spans a fine sampling space. If the given scaling function is discrete (as in *Haar System*), then it does not provide an accurate approximation. If it is not compactly supported (has finite support), it leads to an infinite number of coefficients. Therefore, once again, we cook our own meal and develop our own sampling space that is spanned by a *compactly supported continuous scaling function*. The *recipé* is by *Ingrid Daubechies* who discovered the hierarchy of wavelets that are named after her. We will follow mostly [Boggess 2001] throughtout this section but one may refer to [Gençay 2002] as well.

As we said, our aim is to construct a *multiresolution analysis* that has a *compactly supported continuous scaling function*, but instead of dealing with the sampling spaces, we initially get direct help from *Fourier transform* and redefine the

constraints for a scaling function. Recalling the Theorem 3.3.2 and the latter 4 results derived from it, next theorem provides a formulation of those constraints in terms of the *scaling function*, ϕ .

Theorem 3.4.1 *Let ϕ be a compactly supported continuous function which satisfies the orthonormality condition: $\int \phi(t-k)\phi(t-l)dt = \delta_{kl}$ and let $\{\phi(2^j t - k)\}_{k \in \mathbb{Z}}$ spans V_j . Then the following conditions hold.*

- *The spaces V_j satisfy the separation condition ($\cap V_j = \{0\}$)*
- *If additionally, the conditions for normalization ($\int \phi(t)dt = 1$) and scaling ($\phi(t) = \sum_{k=0}^n p_k \phi(2t-k)$, where $n < \infty$) hold, then some space, V_j satisfies the density condition, that is, any square integrable real function can be approximated by the functions of V_j .*

The proof of this theorem can be found in [Bogges 2001]. Please keep in mind that, Theorem 3.4.1 plays a vital role in our multiresolution analysis construction. In particular, the whole theorem asserts that, if the function ϕ is continuous with compact support and satisfies the *normalization*, *scaling* and *orthonormality* conditions, then the collection of spaces $\{V_j\}_{j \in \mathbb{Z}}$ forms a *multiresolution analysis*.

So let's see how we can represent these constraints in the language of *Fourier Transform*. Recall from 3.11 that the *Fourier Transform* of a function, f is given by

$$\mathcal{F}_\xi[f] = \hat{f}(\xi) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} f(t) e^{-it\xi} dt \quad (3.62)$$

Normalization condition ($\int \phi = 1$) is rather easy to represent. Since we have $\hat{\phi}(0) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} f(t) e^{-it(0)} dt = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} \phi(t) dt$, *Normalization condition* becomes

$$\hat{\phi}(0) = \frac{1}{\sqrt{2\pi}} \quad (3.63)$$

Orthonormality condition needs a little more work and a theorem proof.

Theorem 3.4.2 *A function ϕ satisfies the orthonormality condition iff*

$$F(\xi) = 2\pi \sum_{k \in \mathbb{Z}} |\hat{\phi}(\xi + 2\pi k)|^2 = 1, \quad \forall \xi \in \mathbb{R}$$

⁶ Recall that the factor $\frac{1}{2\pi}$ may be bundled with the *inversion formula* or $\frac{1}{\sqrt{2\pi}}$ can appear in both transform and inverse transform equations to provide a symmetric appearance

In addition, a function $\psi(t)$ is orthogonal to $\{\phi(t - l)\}_{l \in \mathbb{Z}}$ iff

$$\sum_{k \in \mathbb{Z}} \hat{\phi}(\xi + 2\pi k) \overline{\hat{\psi}(\xi + 2\pi k)} = 0, \quad \forall \xi \in \mathbb{R}$$

The proof of this theorem can be found in [Bogges 2001]. In particular, since F is periodic, it has a *Fourier series*, $\sum \alpha_n e^{-int}$, where the *Fourier coefficients* are given by $\alpha_n = \frac{1}{2\pi} \int_0^{2\pi} F(\xi) e^{-in\xi} d\xi$. Thus, by simplification, our *orthonormality condition* becomes

$$-\alpha_n = \delta_{n0} \quad (3.64)$$

Next theorem recasts the *scaling condition* $\phi(t) = \sum_k p_k \phi(2t - k)$ in terms of the *Fourier transform*. But first, we have to give a remark from *Fourier transform*.

Remark 3.4.3 *The Fourier transform of the translated and dilated form of a scaling function, $\phi(bt - a)$ can be evaluated as follows:*

$$\begin{aligned} \mathcal{F}_\xi [\phi(bt - a)] &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} \phi(bt - a) e^{-it\xi} dt = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} \phi(s) e^{-i(\frac{s+a}{b})\xi} \frac{ds}{b} \\ &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} \phi(s) e^{-\frac{ia\xi}{b}} e^{-\frac{is\xi}{b}} \frac{ds}{b} = e^{-\frac{ia\xi}{b}} \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} \phi(s) e^{-\frac{is\xi}{b}} \frac{ds}{b} \\ &= e^{-\frac{ia\xi}{b}} \frac{1}{b} \mathcal{F}_{\xi/2} [\phi] \end{aligned}$$

Theorem 3.4.4 *The scaling condition, $\phi(t) = \sum_k p_k \phi(2t - k)$ is equivalent to*

$$\hat{\phi}(\xi) = \hat{\phi}\left(\frac{\xi}{2}\right) P\left(e^{-\frac{i\xi}{2}}\right)$$

where the polynomial P is given by

$$P(z) = \frac{1}{2} \sum_{k \in \mathbb{Z}} p_k z^k$$

Moreover, we have

$$\hat{\phi}(\xi) = \frac{1}{\sqrt{2\pi}} \prod_{j=1}^{\infty} P\left(e^{-\frac{i\xi}{2^j}}\right)$$

Proof. If we take the *Fourier transform* of both sides of the *scaling condition*, by using Remark 3.4.3, we get,

$$\hat{\phi}(\xi) = \frac{1}{2} \sum_{k \in \mathbb{Z}} \hat{\phi}(\xi/2) p_k e^{ik(\frac{\xi}{2})} = \hat{\phi}\left(\frac{\xi}{2}\right) P\left(e^{-\frac{i\xi}{2}}\right)$$

where $P(z) = \frac{1}{2} \sum_k p_k z^k$, as asserted in the first part. If we continue iteratively, we get

$$\begin{aligned}\hat{\phi}(\xi) &= P(e^{-\frac{i\xi}{2}}) \dots P(e^{-\frac{i\xi}{2^n}}) \hat{\phi}\left(\frac{\xi}{2^n}\right) \\ &= \left(\prod_{j=1}^n P(e^{-\frac{i\xi}{2^j}}) \right) \hat{\phi}\left(\frac{\xi}{2^n}\right)\end{aligned}$$

Therefore, as $n \rightarrow \infty$ we have

$$\begin{aligned}\hat{\phi}(\xi) &= \left(\prod_{j=1}^{\infty} P(e^{-\frac{i\xi}{2^j}}) \right) \underbrace{\hat{\phi}(0)}_{=\frac{1}{\sqrt{2\pi}}} \\ &= \frac{1}{\sqrt{2\pi}} \prod_{j=1}^{\infty} P(e^{-\frac{i\xi}{2^j}})\end{aligned}\tag{3.65}$$

■

Although this theorem has no practical use (because of the infinite product), it will theoretically help us on our way as we will see later. For example, if we recall Theorem 3.3.3, and let $Q(z) = -z\overline{P(-z)}$, for $|z| = 1$, we have, $Q(z) = (\frac{1}{2}) \sum_k (-1)^k \bar{p}_{1-k} z^k$. So we can use the same arguments in Theorem 3.4.4 to show that

$$\hat{\psi}(\xi) = \hat{\phi}\left(\frac{\xi}{2}\right) Q(e^{-\frac{i\xi}{2}})\tag{3.66}$$

Now we combine the latter two theorems (3.4.2 and 3.4.4) to give a necessary condition on $P(z)$ for the existence of a *multiresolution analysis*.

Theorem 3.4.5 *Let ϕ satisfy the orthonormality ($\int \phi(t-k)\phi(t-l)dt = \delta_{kl}$) and the scaling ($\phi(t) = \sum_k p_k \phi(2t-k)$) conditions. Then for $\{|z|=1\}_{z \in \mathbb{C}}$, $P(z) = \sum_k p_k z^k$ satisfies*

$$|P(z)|^2 + |P(-z)|^2 = 1$$

or, equivalently, for $0 \leq t \leq 2\pi$,

$$|P(e^{-it})|^2 + |P(e^{-i(t+\pi)})|^2 = 1$$

Proof. From Theorem 3.4.2, *orthonormality condition* implies

$$\sum_{k \in \mathbb{Z}} |\hat{\phi}(\xi + 2\pi k)|^2 = \frac{1}{2\pi}$$

and from 3.4.4, *scaling condition* implies

$$\hat{\phi}(\xi) = \hat{\phi}\left(\frac{\xi}{2}\right) P(e^{-\frac{i\xi}{2}})$$

Now, if we repeat the procedure of dividing the *orthonormality condition* into even and odd terms, as we did in Example 3.3.6, and then using the *scaling condition* we have,

$$\begin{aligned}
\frac{1}{2\pi} &= \sum_{k \in \mathbb{Z}} |\hat{\phi}(\xi + 2\pi k)|^2 = \sum_{l \in \mathbb{Z}} |\hat{\phi}(\xi + (2l)2\pi)|^2 + \sum_{l \in \mathbb{Z}} |\hat{\phi}(\xi + (2l+1)2\pi)|^2 \\
&= \sum_{l \in \mathbb{Z}} \left(|P(e^{-i(\frac{\xi}{2} + 2l\pi)})|^2 |\hat{\phi}(\frac{\xi}{2} + (2l)\pi)|^2 + |P(e^{-i(\frac{\xi}{2} + (2l+1)\pi)})|^2 |\hat{\phi}(\frac{\xi}{2} + (2l+1)\pi)|^2 \right) \\
&= |P(e^{-i\frac{\xi}{2}})|^2 \underbrace{\sum_{l \in \mathbb{Z}} |\hat{\phi}(\frac{\xi}{2} + 2\pi l)|^2}_{\frac{1}{2\pi}} + |P(-e^{-i\frac{\xi}{2}})|^2 \underbrace{\sum_{l \in \mathbb{Z}} |\hat{\phi}(\frac{\xi}{2} + \pi + 2\pi l)|^2}_{\frac{1}{2\pi}} \\
\Rightarrow 1 &= |P(e^{-i\frac{\xi}{2}})|^2 + |P(-e^{-i\frac{\xi}{2}})|^2
\end{aligned}$$

Since this equation holds for all $\xi \in \mathbb{R}$, we may conclude that $|P(z)|^2 + |P(-z)|^2 = 1$ for all $\{|z| = 1\}_{z \in \mathbb{C}}$ with \blacksquare

Since we have given the necessary condition on $P(Z)$, one thing left is giving the necessary condition on $Q(Z)$, the scaling relation for $\hat{\psi}(\xi)$. Next theorem states those conditions.

Theorem 3.4.6 *Let ϕ satisfy the orthonormality ($\int \phi(t-k)\phi(t-l)dt = \delta_{kl}$) and the scaling ($\phi(t) = \sum_k p_k \phi(2t-k)$) conditions. Now suppose $\hat{\psi}(t) = \sum_k q_k \phi(2t-k)$ and let $Q(z) = \sum_k q_k z^k$. Then we have,*

$$\int \psi(t-k)\phi(t-l)dt = 0 \quad \forall k, l \in \mathbb{Z}$$

or equivalently

$$P(z)\overline{Q(z)} + P(-z)\overline{Q(-z)} = 0$$

Proof. The proof has the same steps as the proof of Theorem 3.4.5. We prefer a more direct proof. We know that $Q(z) = -z\overline{P(-z)}$, for $|z| = 1$. Then we have,

$$P(z)\overline{Q(z)} + P(-z)\overline{Q(-z)} = -zP(z)P(-z) + zP(-z)P(z) = 0$$

\blacksquare

Iterative Procedure

Now that we have our constraints for the polynomials $P(z)$ and $Q(z)$, we may develop a procedure to construct a scaling function. We will prefer constructing a

polynomial that satisfies $|P(e^{-it})|^2 + |P(e^{-i(t+\pi)})|^2 = 1$ and then construct the function $\phi(t)$ that satisfies the scaling condition. Now let's skip the construction of the polynomial and consider it is constructed. How we construct the scaling function by iteration is asserted in the next theorem.

Theorem 3.4.7 Let $P(z) = \frac{1}{2} \sum_k p_k z^k$ satisfies the following conditions,

- $P(1) = 1$
- $|P(z)|^2 + |P(-z)|^2 = 1$ for $|z| = 1$
- $|P(e^{it})| > 0$ for $|t| \leq \frac{\pi}{2}$

Let ϕ_0 be the Haar scaling function and let $\phi_n(t) = \sum_k p_k \phi_{n-1}(2t - k)$ for $n \geq 1$. Then in the space of square integrable functions, the sequence, ϕ_n converges pointwise to a function ϕ which satisfies the orthonormality ($\int_{-\infty}^{\infty} \phi(t - n)\phi(t - m)dt = \delta_{nm}$) and the scaling ($\phi(t) = \sum_k p_k \phi(2t - k)$) conditions.

The proof of this theorem can be found in [Boggess 2001].

In practice, Theorem 3.4.7 means a lot for our purpose. Now we have a sequence inherited from an initial scaling function that converges to a scaling function which satisfies the orthonormality and scaling conditions. However, one must have noticed that the initial scaling function, ϕ_0 is the *Haar scaling function* and we don't employ this scaling function much because of the continuity problems. So we have to generalize this theorem to some extent. So we define $p(\xi) = P(e^{-i\xi})$ and redefine the conditions in Theorem 3.4.7 as

$$p(0) = 1 \tag{3.67}$$

$$|p(\xi)|^2 + |p(\xi + \pi)|^2 = 1 \tag{3.68}$$

$$|p(\xi)| > 0 \quad \text{for} \quad -\frac{\pi}{2} \leq \xi \leq \frac{\pi}{2} \tag{3.69}$$

Now what we have to do is to find a $p(\xi)$ that satisfies all of the above conditions which is, of course, a difficult task. But *Daubechies framework*, makes it easier. Let us consider the trigonometric equation $\cos^2 \frac{\xi}{2} + \sin^2 \frac{\xi}{2} = 1$. If we can decompose some power of left side of the equation into the form $|p(\xi)|^2 + |p(\xi + \pi)|^2$, then we have our $p(\xi)$. Let's consider $n = 3$ for instance. Recalling the trigonometric expansions

$\cos(u) = \sin(u + \frac{\pi}{2})$ and $\sin(u) = -\cos(u + \frac{\pi}{2})$, our trigonometric equation becomes,

$$\begin{aligned} 1^3 = 1 &= \left(\cos^2\left(\frac{\xi}{2}\right) + \sin^2\left(\frac{\xi}{2}\right) \right)^3 \\ &= \cos^6\left(\frac{\xi}{2}\right) + 3 \cos^4\left(\frac{\xi}{2}\right) \sin^2\left(\frac{\xi}{2}\right) + 3 \cos^2\left(\frac{\xi}{2}\right) \sin^4\left(\frac{\xi}{2}\right) + \sin^6\left(\frac{\xi}{2}\right) \\ &= \cos^6\left(\frac{\xi}{2}\right) + 3 \cos^4\left(\frac{\xi}{2}\right) \sin^2\left(\frac{\xi}{2}\right) \\ &\quad + 3 \cos^4\left(\frac{\xi + \pi}{2}\right) \sin^2\left(\frac{\xi + \pi}{2}\right) + \cos^6\left(\frac{\xi + \pi}{2}\right) \end{aligned}$$

With $|p(\xi)|^2 = \cos^6\left(\frac{\xi}{2}\right) + 3 \cos^4\left(\frac{\xi}{2}\right) \sin^2\left(\frac{\xi}{2}\right)$, we have

$$1 = |p(\xi)|^2 + |p(\xi + \pi)|^2$$

Therefore, constraint 3.68 is satisfied. Constraints 3.67 and 3.69 are already satisfied with our choice of $|p(\xi)|^2$ since $|p(0)|^2 = \cos^6\left(\frac{0}{2}\right) + 3 \cos^4\left(\frac{0}{2}\right) \sin^2\left(\frac{0}{2}\right) = 1$ and $\cos\left(\frac{\xi}{2}\right) \geq \frac{1}{\sqrt{2}}$ for $|\xi| \leq \frac{\pi}{2}$. So in order to pull out p from the equation, we rewrite our definition as,

$$\begin{aligned} |p(\xi)|^2 &= \cos^4\left(\frac{\xi}{2}\right) \left(\cos^2\left(\frac{\xi}{2}\right) + 3 \sin^2\left(\frac{\xi}{2}\right) \right) \\ &= \cos^4\left(\frac{\xi}{2}\right) \left| \cos\left(\frac{\xi}{2}\right) + \sqrt{3}i \sin\left(\frac{\xi}{2}\right) \right|^2 \\ \Rightarrow p(\xi) &= \cos^2\left(\frac{\xi}{2}\right) \left| \cos\left(\frac{\xi}{2}\right) + \sqrt{3}i \sin\left(\frac{\xi}{2}\right) \right| \alpha(\xi) \end{aligned}$$

where $\alpha(\xi)$ is a complex-valued function having $|\alpha(\xi)| = 1$ that is to be chosen for our purposes later. If we use the trigonometric decompositions

$$\cos\left(\frac{\xi}{2}\right) = \frac{e^{\frac{i\xi}{2}} + e^{-\frac{i\xi}{2}}}{2} \quad \sin\left(\frac{\xi}{2}\right) = \frac{e^{\frac{i\xi}{2}} - e^{-\frac{i\xi}{2}}}{2i}$$

we get

$$p(\xi) = \frac{1}{8} \left(e^{i\xi} + 2 + e^{-i\xi} \right) \left(e^{\frac{i\xi}{2}} + e^{-\frac{i\xi}{2}} \sqrt{3} e^{\frac{i\xi}{2}} - \sqrt{3} e^{-\frac{i\xi}{2}} \right) \alpha(\xi)$$

Now we choose $\alpha(\xi) = e^{-\frac{3i\xi}{2}}$ that clears all positive and fractional powers of the exponent. Then $p(\xi)$ becomes

$$p(\xi) = \left(\frac{1 + \sqrt{3}}{8} \right) + e^{-i\xi} \left(\frac{3 + \sqrt{3}}{8} \right) + e^{-2i\xi} \left(\frac{3 - \sqrt{3}}{8} \right) + e^{-3i\xi} \left(\frac{1 + \sqrt{3}}{8} \right)$$

Recall that $p(\xi) = P(e^{-i\xi}) = \frac{1}{2} \sum_k p_k e^{-ik\xi}$ which is the form of our equation. So we evaluated our p_k 's.

$$p_0 = \frac{1 + \sqrt{3}}{4}, \quad p_1 = \frac{3 + \sqrt{3}}{4}, \quad p_2 = \frac{3 - \sqrt{3}}{4}, \quad p_3 = \frac{1 - \sqrt{3}}{4} \quad (3.70)$$

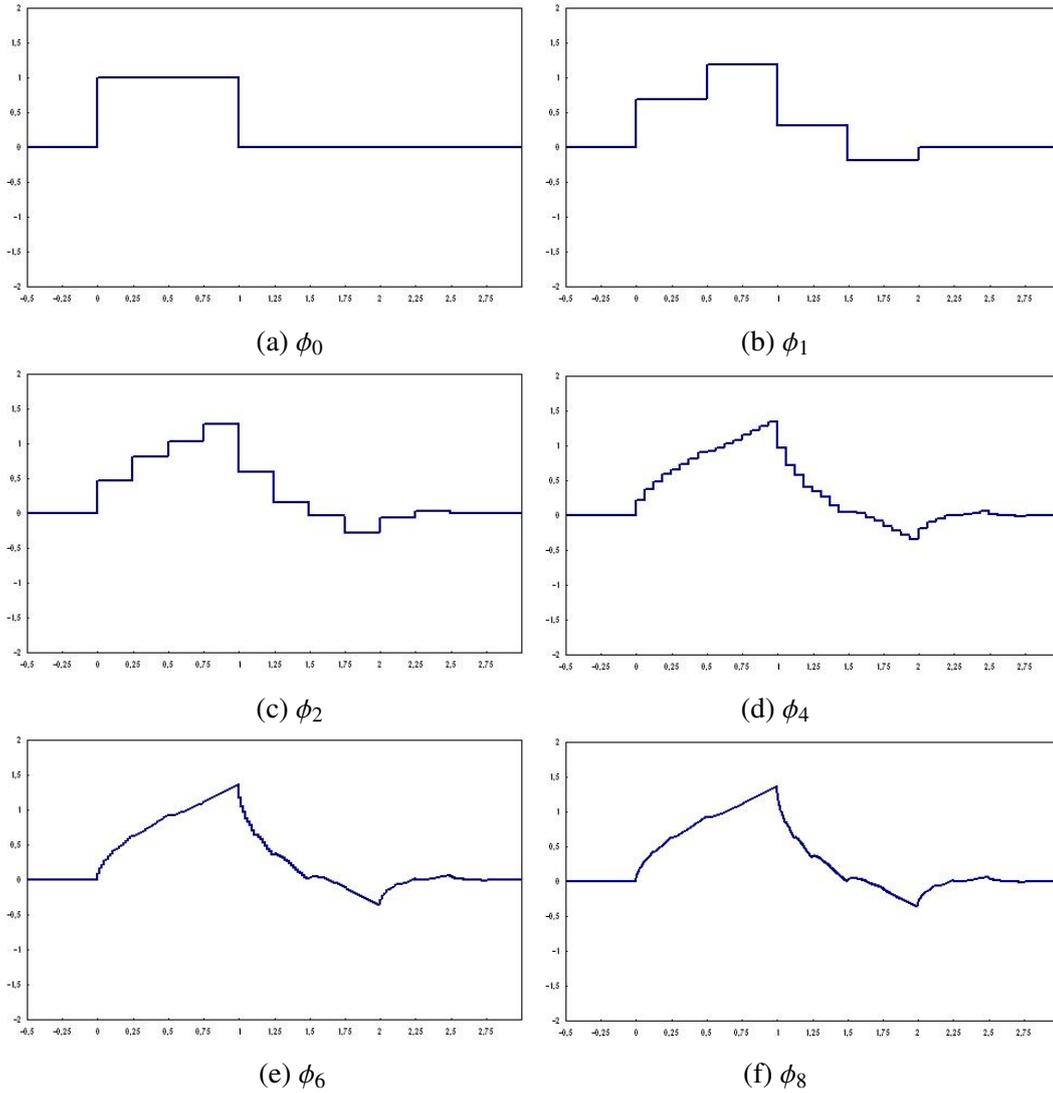


Figure 3.7: Plot of the iteration steps of Daubechies Scaling Function ($N = 2$)

Since we have our p_k 's, at last, we can evaluate the scaling function, using the iteration procedure asserted in Theorem 3.4.7. The plot of the p_k 's after j iterations (ϕ_j 's) are given in Figure 3.7. As can be seen from (f), after 8 iterations, ϕ_8 almost converges its limit.

The rest is even easier now. Once we have identified ϕ and evaluated the p_k 's, we may indentify our wavelet, ψ which is given by the formula

$$\psi(t) = \sum_{k \in \mathbb{Z}} (-1)^k p_{1-k} \phi(2t - k)$$

The plot of the wavelet function extracted from ϕ_8 is shown in Figure 3.8. Since j is

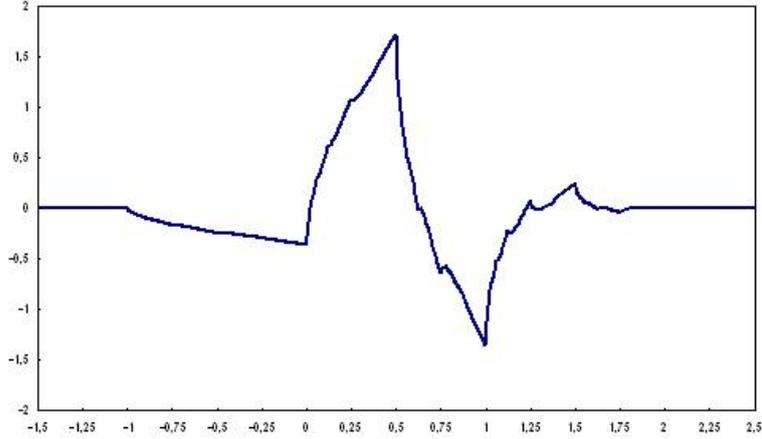


Figure 3.8: Plot of the Daubechies Wavelet Function Extracted from ϕ_8 ($N = 2$)

large enough, the limit of the function (actual wavelet) should be very similar to the plot.

Now that we have our scaling function and wavelet coefficients, we may apply the decomposition algorithm in Section 3.3 and finalize the *multiresolution analysis*.

As one may have noticed, we tripled the trigonometric equation $\cos^2 \frac{\xi}{2} + \sin^2 \frac{\xi}{2} = 1$ and then extracted a $p(\xi) = P(e^{-i\xi})$. However, we have other alternatives. For every odd power $\{n = 2N - 1\}_{N \in \{1,2,3..\}}$, we have the decomposition $P(z) = \frac{1}{2} \sum_k p_k z^k$. So the p_k 's of the order N form our ϕ_0^N of the iterative procedure. There are $2 * N$ p_k 's for ϕ_0^N .

So we conclude the chapter on *Wavelet Theory and Multiresolution Analysis*. In our work, we will use *Daubechies family of scaling functions* of order 2, for simplicity. Nevertheless, one may prefer higher orders of *Daubechies family* ($D(N)$) of scaling functions or different wavelet families such as *Least Asymmetric Family* ($LA(i)$) of scaling functions. The interested reader may refer to [Gençay 2002] for more information about those families.

CHAPTER 4

NEW MODEL FOR HIGH FREQUENCY DATA SAMPLING

4.1 Introduction

As we have stated in the introduction chapter, the starting point of this work is the research of [Gençay 2001] which uses the approximation spaces, V_j 's as the deseasonalized components of a signal. Recall from Chapter 3, there are two orthonormal bases regarding the approximation space V_{j+1} , that are V_j and W_j , which is the orthogonal complement of V_{j+1} . So $f_j \in V_j$ is the residual after the j 'th level wavelet space component, $w_j \in W_j$ of the function f_{j+1} is extracted from the signal.

Although the method of Gençay et al. turns the ACF into a long memory function ACF, there is a problem with this approach. It oversmooths the signal, so that the new plot of the data does not seem like the original one, so there is a good chance of throwing some useful information out as well as the seasonal components. This can be seen from the $f_j \in V_j$ plots of our data (Figure 4.1). One may see the full scatter of the data in Figure 5.1. Although the plot of $f_{15} \in V_{15}$ (frequency is $\frac{1}{2^{15}}$ of the original domain) is very much alike to the plot of original data, the ACF of $f_{15} \in V_{15}$ is not as good as it is expected. However, when the data is smoothed down to $f_4 \in V_4$ (frequency is $\frac{1}{16}$ of the original domain), the ACF is almost perfect, but the smoothed function is not much like our original function. Besides, it is difficult to decompose the marks ¹ on ticks. Because of those reasons, researchers do not tend to use this decomposition much, and sampling is still preferred.

The other side of the picture is not so bright as well. As we have stated in Chapter 2, sampling in higher frequencies ends up with more bias caused by the microstruc-

¹ information on trades such as volume, etc.

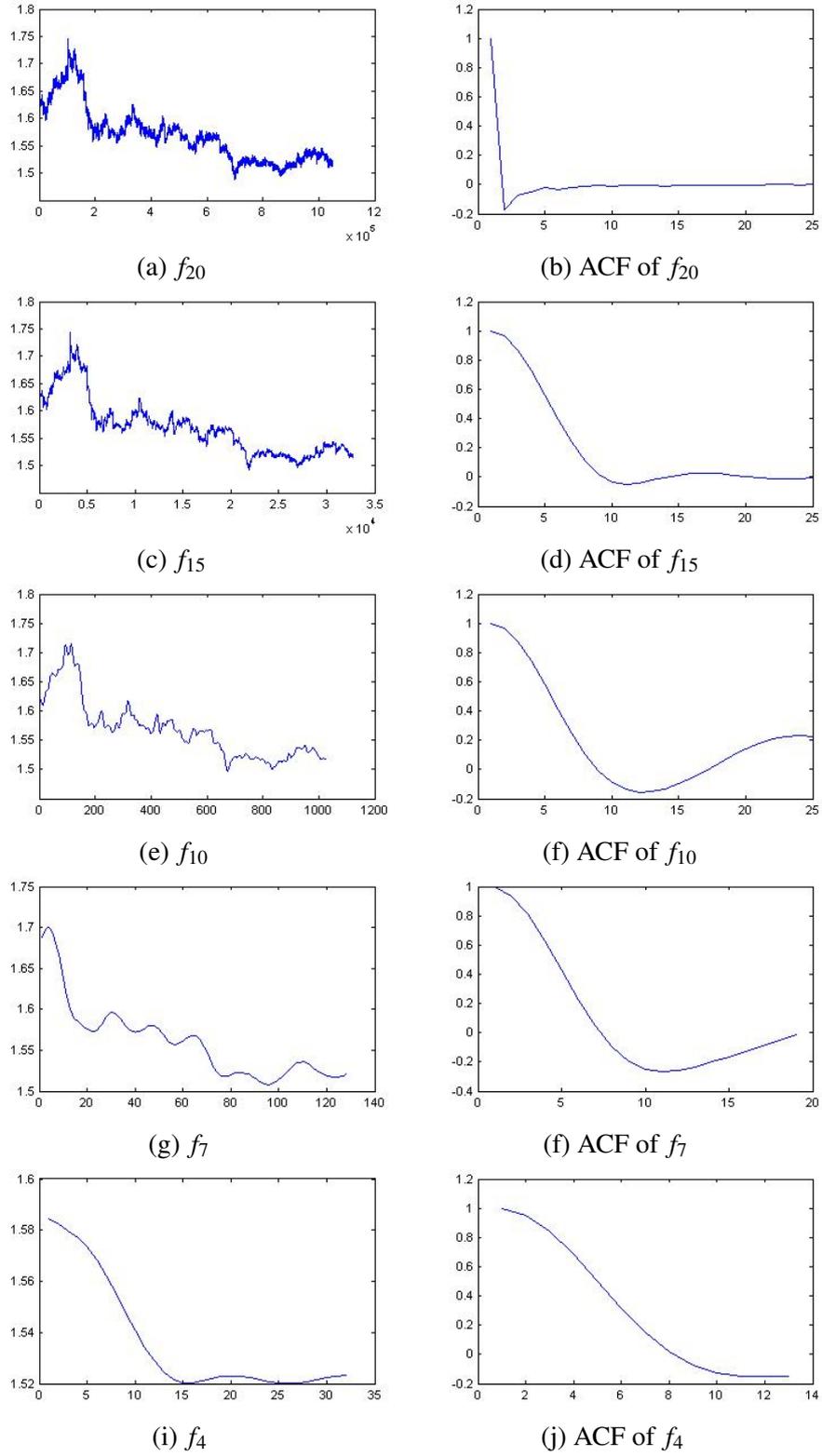


Figure 4.1: Plot of the iteration steps of the smoothing in [Gençay 2001] with USD-TRY transactional data of our research

ture effects and the ACF's are not smooth. So generally a sampling scheme is used to reduce this effect. In the rest of the sections, we will introduce a sampling scheme model that reduces the bias compared to classical sampling schemes that are mentioned in Section 2.3. When we need illustrations, we will generally apply it to our data set, the 37 day tick-by-tick USD-TRY transactions, as we did in Figure 4.1.

As we have mentioned earlier, we will develop a version of the *Business Time Sampling* (BST). Recall that the scheme is based on sampling the sequence of prices as,

$$P_{bst} = \left\{ P \left(t_i^{bst} \right) \right\}_{i \in \{0,1,\dots,N\}}, \quad t_i^{bst} = \inf_{0 \leq t \leq T} \left\{ \lambda_t \geq i \frac{\lambda_T}{N} \right\} \quad (4.1)$$

where λ_t is the *business intensity* at time t . The variable used as the *business intensity* varies since the recognition of the term *business intensity* changes from person to person. Some use the volume of the trades while others use lags between the trades. Our scale of *business intensity* will be the quadratic variation of the projection of the signal f onto each resolution of wavelet space, ($w_j \in W_j$).

4.2 Rescaling and Smoothing

If we look at the histogram of our data (Figure 4.2), we see that there is a huge seasonality with the morning trades. Here, we follow the guidelines in [Dacorogna 2001] roughly. In order to remove this seasonality, as described in Section 2.4, we must choose an activity variable a . We choose it to be the average tick size and minimize the normalized standart deviation. Here, we use the flexibility in choosing of activity variable in our favor, for simplicity reasons. Our method is simple and is not the best for sure, but deseasonalizing the data may be the topic of another paper.

Since we do not use *Maximum Overlap Discrete Wavelet Transform* in order not to give up from orthogonality, we use the ordinary *Discrete Wavelet Transform* which uses *Daubechies Wavelet Family* as one may remember from Chapter 3 (Also see [Gençay 2002]). This means that the size of our domain may be 2^k where k is a positive integer. For this necessity, we do some manual work first and decide on the size of each day. For instance, in our case, we have 37 days of data spread over 10,5 calendar hours a day, which makes a total of 1,398,600 seconds of trading time. Since our data show few seasonality nearly for 7 consecutive intraday calendar hours, and

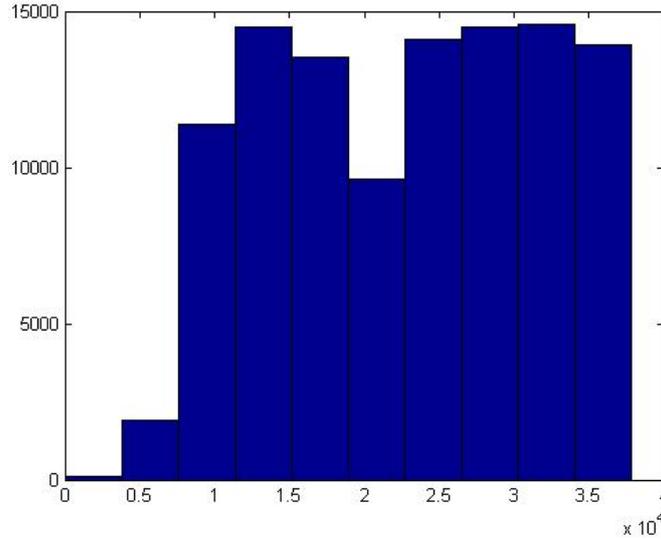
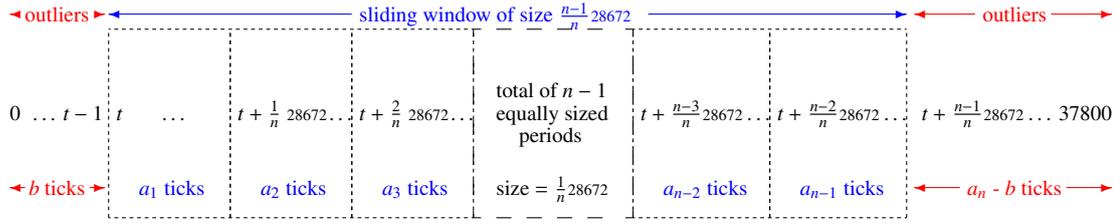


Figure 4.2: Histogram (X-axis in seconds)

this makes a total of 932,400 seconds, we decide on using $2^{20} = 1,048,576$ seconds as the size of our total domain. So we have to squeeze or enlarge the hours that show high seasonality and obtain a domain of size close to 1,048,576.

In order to apply rescaling to our business day, we first determine the optimal partitions. We systematically divide our target day into n equally sized periods where n is sufficiently large and then create a window equal to the size of $n - 1$ periods. The outliers of the window is treated as the n^{th} slice. We calculate the number of trades in each of the n slices and then evaluate the normalized standart deviation for this choice of partition. Next, we slide this window programatically over the day. The new outliers as we slide the $n - 1$ period sized window are still treated as the n^{th} slice. As we slide the window, we keep on evaluating the normalized standart deviation. When we finish sliding the window, we store the partition with minimum normalized standart deviation. Then we decrease the number of partitions n and repeat the same procedure. We repeat this procedure programatically until we end up with n 's. The partition in all of these partitions which has the minimum normalized standart deviation is chosen to be the final partition. Next we rescale the n^{th} partition to have the same size with other slices by using 2.11 and therefore we get our new time scale.

As one may have noticed, in order to find the partition with the mimumum



We minimize the normalized standard deviation of $\{a_i\}_{i=1}^n$ by sliding the window in the domain and then repeating this procedure for each n . When the optimal window is obtained, we rescale the outliers to size $\frac{1}{n}28672$ and obtain new time scale of 28672 seconds

Figure 4.3: The rescaling algorithm

normalized standard deviation, we iterated the same procedure over and over again with different n 's. So when applying this procedure, we must choose the target size that has many divisors as possible. For instance we choose the target size to be 28,672 seconds for our domain. We illustrate the process in Figure 4.3. Also the empirical results of the procedure are shown in Chapter 5.

Now that we have rescaled our data which show high seasonality, our next task becomes smoothing the data. As one can see from Figure 4.1, there are a lot of jumps in our data. That's due to the fact that the market is closed for nearly 13.5 hours and during this time, both American and Far East markets are open. This price jump contains a lot of information. Therefore we have to smooth the data instead of neglecting it.

Although many ways of smoothing, some of which are mentioned in Section 2.4 are available, we will propose a new FIR filter for seasonality removal, which is better for fixed domains. This procedure also can be used for connecting two non-consecutive days with minimum bias from the total intraday realized variance.

Let $X = \{x_{t+i}\}_{i=-N}^M$ be a fixed vector of observations in a larger vector of observations $XX = \{x_{t+i}\}_{i=-NN}^{MM}$ and $\{w_i\}_{i=-n}^m$ be the coefficients of an FIR filter. Recalling 2.12, to smooth X , we apply the filter to XX to give an output vector

$$Y = y_t = \begin{cases} \sum_{k=-n}^m w_k x_{t-k} & \text{if } t \text{ in } [-N, M] \\ x_t & \text{otherwise} \end{cases}$$

This procedure is illustrated in Figure 4.4.

Note that, outside of the domain of X , which is $[-N, M]$ (the dashed box in Figure

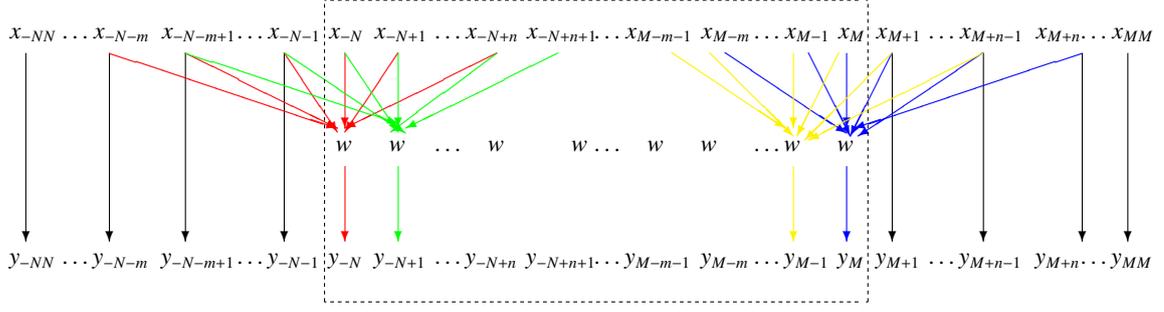


Figure 4.4: A classical FIR filter for fixed intervals

4.4), the filter vector w is defined to be the unit impulse signal, which is

$$w_i = \begin{cases} 1 & i = t \\ 0 & \text{otherwise} \end{cases}$$

That means, outside of this interval, we have $y_t = x_t$. Now let us analyze the overall output vector $Y = \{y_{t+i}\}_{i=-NN}^{MM}$. We assume that the sum of the impulse response ² of w is equal to one, which means $\sum_{i=-n}^m w_i = 1$. Therefore, our total output vector becomes

$$\begin{aligned} Y = & (X_{-NN}, X_{-NN+1}, \dots \\ & \dots, X_{-N-m-1}, (1 + \sum_{i=-m}^{-m} w_{-i})X_{-N-m}, (1 + \sum_{i=-m}^{-m+1} w_{-i})X_{-N-m+1}, (1 + \sum_{i=-m}^{-m+2} w_{-i})X_{-N-m+2}, \dots, \\ & \dots, (1 + \sum_{i=-m}^{-1} w_{-i})X_{-N-1}, (\sum_{i=-m}^{n-1} w_{-i})X_{-N}, (\sum_{i=-m}^1 w_{-i})X_{-N+1}, \dots, \\ & \dots, (\sum_{i=-m}^{n-1} w_{-i})X_{-N+n-1}, X_{-N+n}, X_{-N+n+1}, \dots, \\ & \dots, X_{M-m}, (\sum_{i=-m+1}^n w_{-i})X_{M-m+1}, (\sum_{i=-m+2}^n w_{-i})X_{M-m+2}, \dots, \\ & \dots, (\sum_{i=-1}^n w_{-i})X_{M-1}, (\sum_{i=0}^n w_{-i})X_M, (1 + \sum_{i=1}^n w_{-i})X_{M+1}, \dots \\ & \dots, (\sum_{i=n}^n w_{-i})X_{M+n}, X_{M+n+1}, \dots, X_{MM-1}, X_{MM}) \end{aligned}$$

which means that, if we think of the filter as a whole, impulse response becomes inconsistent. It overweights the input signals near (outside) filter domain while it

² See Section 2.4

underweights the input signals on the two edges of the filter domain. Let us see it in an example:

Example 4.2.1 Suppose our input vector is $XX = \{x_i\}_{i=1}^7$ and the portion of the vector which shows seasonality and must be smoothed be $X = \{x_i\}_{i=3}^5$. Let our filter be a simple centered average filter $w = (0.25, 0.5, 0.25)$. Then our total output becomes,

$$Y = (x_1, x_2, 0.25x_2+0.5x_3+0.25x_4, 0.25x_3+0.5x_4+0.25x_5, 0.25x_4+0.5x_5+0.25x_6, x_6, x_7)$$

which means that the weight of our inputs on the total output becomes

$$(1, 1.25, 0.75, 1, 0.75, 1.25, 1)$$

respectively. For an ideal impulse response filter, this should be

$$(1, 1, 1, 1, 1, 1, 1)$$

If we want to smooth big jumps, we should be using wider filter vectors, which means that the imbalance spreads over a wider domain and this means that the weights of the first and last elements of X may decrease down to 0.5.

We propose a filter that changes continuously and only stays in the domain to be smoothed. The filter coefficients are determined according to the order of filtering. Let S be the size of the domain $X \in XX$ to be smoothed where x_1 is the first and x_S is the last elements of the domain X . We determine the filter coefficients applied to the domain, to yield the a^{th} output element of the output vector to be the coefficients of

$$\left(\frac{a-1}{S-1} + \frac{S-a}{S-1}x\right)^{S-1}$$

which means that for the components of a^{th} filter vector is

$$w_i^a = \binom{S-1}{i-1} \left(\frac{a-1}{S-1}\right)^{S-i} \left(\frac{S-a+1}{S-1}\right)^{i-1}$$

Since $\frac{a-1}{S-1} + \frac{S-a}{S-1} = 1$, our filter satisfies the condition $\sum_i w_i^a = 1$. Moreover, it does not involve the elements of XX that are not the elements of X , into calculations. The procedure is illustrated in 4.5. Let us give an example at this point.

Example 4.2.2 Suppose $X = (x_1, x_2, x_3, x_4)$. Then the filter vectors to yield y_1, y_2, y_3, y_4

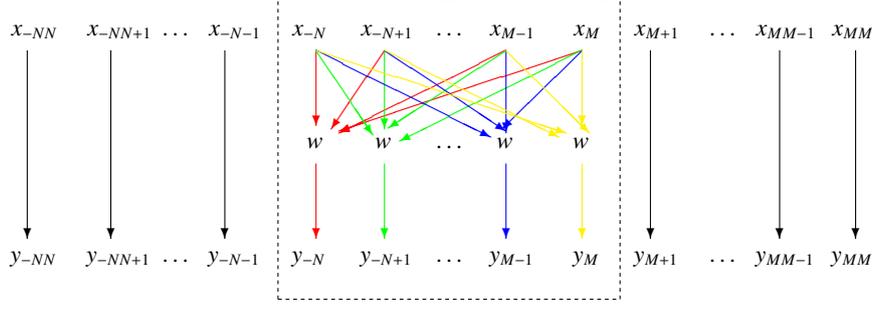


Figure 4.5: The smoothing filter in this research

are respectively

$$w^1 = \binom{3}{0}0^31^0, \binom{3}{1}0^21^1, \binom{3}{2}0^11^2, \binom{3}{3}0^01^3 = (0, 0, 0, 1)$$

$$w^2 = \binom{3}{0}0.33^30.66^0, \binom{3}{1}0.33^20.66^1, \binom{3}{2}0.33^10.66^2, \binom{3}{3}0.33^00.66^3 = (0.04, 0.22, 0.44, 0.3)$$

$$w^3 = \binom{3}{0}0.66^30.33^0, \binom{3}{1}0.66^20.33^1, \binom{3}{2}0.66^10.33^2, \binom{3}{3}0.66^00.33^3 = (0.3, 0.44, 0.22, 0.04)$$

$$w^4 = \binom{3}{0}1^30^0, \binom{3}{1}1^20^1, \binom{3}{2}1^10^2, \binom{3}{3}1^00^3 = (1, 0, 0, 0)$$

so the output vectors become

$$y_1 = X \star w^1 = x_1$$

$$y_2 = X \star w^2 = 0.04x_4 + 0.22x_3 + 0.44x_2 + 0.3x_1$$

$$y_3 = X \star w^3 = 0.3x_4 + 0.44x_3 + 0.22x_2 + 0.04x_1$$

$$y_4 = X \star w^4 = x_4$$

Notice that, in our example, the weight of the inputs on the total output is (1.34, 0.66, 0.66, 1.34) respectively, and this is not ideal. This is because, the weights tend to 1 as the domain size increases. In other words, this approach is suitable when the smoothed domain is large. This is because

$$\lim_{S \rightarrow \infty} \sum_{i=2}^{S-2} \binom{S-1}{a-1} \left(\frac{i-1}{S-1} \right)^{S-a} \left(\frac{S-i+1}{S-1} \right)^{a-1} = 1$$

and can be seen from Figure 4.4. The empirically found advantages of this filter will be presented in detail in Chapter 5.

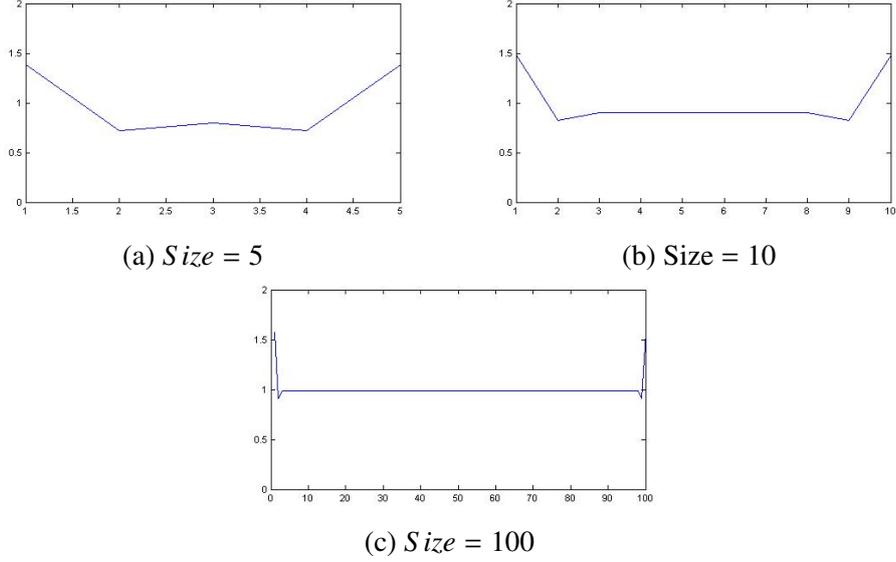


Figure 4.6: Plot of the weight of input vector items on output vector

4.3 Multiscale Quadratic Variation Filter

Once we have our data rescaled and smoothed, we may apply our sampling scheme. As we have considered the 2^n condition when rescaling, we have a domain of size, very close to 2^n . In our case, this is $2^{20} = 1,048,576$. Since we had determined the size of a business day to be $28,672^3$ seconds, we have a total of $1,060,864$ seconds of data. So we have to throw the first $12,288$ seconds of data out, to fit it into our decomposition algorithm⁴.

In order to apply the decomposition described in Chapter 3, we first construct our basis. As we have stated, we will construct Daubechies(2) scaling function and use the decomposition algorithm associated with the multiresolution analysis to decompose our data. Recalling the details of this decomposition from Section 3.3 we have

$$a_k^j \approx mf\left(\frac{k}{2^j}\right) \quad (\text{See Theorem 3.3.5 and 3.55})$$

for the initial case. And also recall from section 3.4 that

$$m = \int \phi = 1$$

³ As one would remember from the prior section, to find the optimal window, we took divisibility into consideration. $28,672$ is divisible by 2^{12} , so we tried as much n 's as possible in order to find the optimal window

⁴ Note that, we have the option to use other enhanced wavelets in order not to throw out any data

for Daubechies Wavelet Family. So we have

$$a_k^j \approx f\left(\frac{k}{2^j}\right)$$

for our initial case. Since j is 20 for our analysis, we have $f_{20} \in V_{20}$ is f itself. Recall from 3.52 that in order to decompose $f_{20} \in V_{20}$ into $f_{19} \in V_{19}$ and $w_{19} \in W_{19}$, we have to evaluate the a_k^{19} 's and b_k^{19} 's. Our p_k 's to evaluate a_k^{j-1} 's and b_k^{j-1} 's are the p_k 's of Daubechies(2) scaling function which are (from 3.70),

$$p_0 = \frac{1 + \sqrt{3}}{4}, \quad p_1 = \frac{3 + \sqrt{3}}{4}, \quad p_2 = \frac{3 - \sqrt{3}}{4}, \quad p_3 = \frac{1 - \sqrt{3}}{4}$$

initially.

Now, recall Figure 3.6 and equation. Since $f_j \in V_j$ can be decomposed into orthogonal components

$$\begin{array}{cccccccc} f \approx f_j & \rightarrow & f_{j-1} & \rightarrow & f_{j-2} & \cdots & \rightarrow & f_2 & \rightarrow & f_1 & \rightarrow & f_0 \\ & & \searrow & & \searrow & & \searrow & & \searrow & & \searrow & & \searrow \\ & & w_{j-1} & & w_{j-2} & & \cdots & & w_2 & & w_1 & & w_0 \end{array}$$

We may treat w_i 's as the increments to the signal in i^{th} resolution. Therefore the changes in w_i 's act like returns. So for each w_i , r_k^i becomes,

$$r_k^i = \log\left(\frac{(f_k^i + w_k^i)}{(f_{k-1}^i + w_{k-1}^i)}\right), \quad k \in \{2, \dots, 2^i\} \quad (4.2)$$

From that point on, we use the aggregate squared returns at the resolution i , up to coefficient k , which is $\sum_{t=2}^k r_t^i$ as the business intensity λ in the equation 4.1. This is equal to the quadratic variation of the returns up to coefficient k and we will use it as the new business intensity scale.

So, as in BTS (see Section 2.3), we calculate the total quadratic variation of w_i in each resolution i , determine equidistant thresholds, and take a sample from the original signal, each time a threshold is exceeded. Since w_i is the best component that represents the activity in i^{th} resolution, we believe that the least biased realized variation of a signal sampled at a frequency λ should be the one yielded by the resolution that is closest to λ . That means, since our initial resolution is $1/1,048,576$, if we are sampling at a frequency of $1/512$, w^9 should yield the best result since $2^9 = 512$. Actually, this is the farthest resolution for this algorithm, since w^8 has only 256 components and we cannot make 512 sampling from this resolution.

There is also a fine tuning we can adapt to the analysis. Let us recall Figure 3.7. The Daubechies(2) scaling function gets smoother as we apply the iteration steps in

Theorem 3.4.7. That means, it fits the data better with each iteration. So we may expect that our w_i fits the i^{th} resolution of the signal better with each iteration as well. But we have a limitation with that approach. This is due to padding we need to make, in order to apply decomposition algorithm.

If we recall equations 3.56, 3.57, 3.58 and 3.61, we apply a *low pass filter*, l to obtain the coefficients a_k^{j-1} and a *high pass filter*, h to obtain the coefficients b_k^{j-1} 's.

$$\begin{aligned} a_k^{j-1} &= D(l \star a^j)_k \\ b_{k+1}^{j-1} &= D(h \star a^j)_k \end{aligned}$$

Therefore if the number of p_k 's are z for the Daubechies scaling function after some iterations, we should have $z - 1$ more elements than the size of f_j , in order to compute a_k^{j-1} 's and b_k^{j-1} 's for last k . But we don't have them. So we apply padding. In other words append extra elements that were not there. There are a lot of padding techniques, such as zero padding (all the added elements are zero), periodic extension (copy elements of the original set beginning from the first element), smooth padding (extrapolate elements) and symmetric extension (copy elements of the original set backwards beginning from the last element). For Daubechies (2) scaling function, we have 4 p_k 's for the first iteration. But the number increases with $(z_1 - 1) * (2^{level} - 1)$ at each iteration. This is like doubling the old number of coefficients for our analysis. It goes like 4, 10, 22, 46, 94, 190, ... and so on.

We will use symmetric extension in our analysis. So if we iterate the scaling 5 times, we will have 94 elements padded to our coefficients. This much padding may not effect quadratic variation of a dataset of size 524, 288 (w_{19}), but it surely effects if the size is 1024 (w_{10}) or 256 (w_8). So padding causes bias, and this totally depends on the structure of data (we may not be foreseeing it). Therefore there is a limit to our fine tuning, bu it still works fine. We will show it empirically in the next chapter.

4.4 Measuring The Performance

The *bias estimation* methods for the integrated volatility is rather heuristic. That's because there is actually no possible way to measure the the real integrated volatility since it's extracted from a theoretical price. However, there are still ways to measure

the performance of a sampling scheme.

As we stated before, empirical results show that bias that is caused by the microstructure effects increase with sampling in higher frequencies. So [Dacorogna 2001] suggests to use bias factor to measure the performance of a sampling scheme at time t_i which is,

$$B(t_i) = \sqrt{\frac{\sum_{j=1}^{mq} r_{\Delta t}(t_{i-mq+j})}{\sum_{j=1}^m r_{\Delta t_{ref}}(t_{i-m+j})}} \quad (4.3)$$

where r is the log return of the price samples in the period, Δt_{ref} is the time interval between the samples of reference sampling scheme and Δt is the time interval between the samples of new sampling scheme. Note that $\Delta t_{ref} = m\Delta t$ here. In practice the bias factor is the ratio of new sampling realized volatility to the reference sampling realized volatility. The reference interval is usually chosen to be sufficiently big to reduce bias of microstructure effects, 2-4 hours or 1 day depending on the structure of data. We choose it to be 1 day, which makes 37 observations. Please note that Δt may not necessarily be the calendar time. In our case, our scale is the quadratic variations of j^{th} resolution component of f .

Next chapter demonstrates our empirical findings.

CHAPTER 5

EMPIRICAL RESULTS AND CONCLUSION

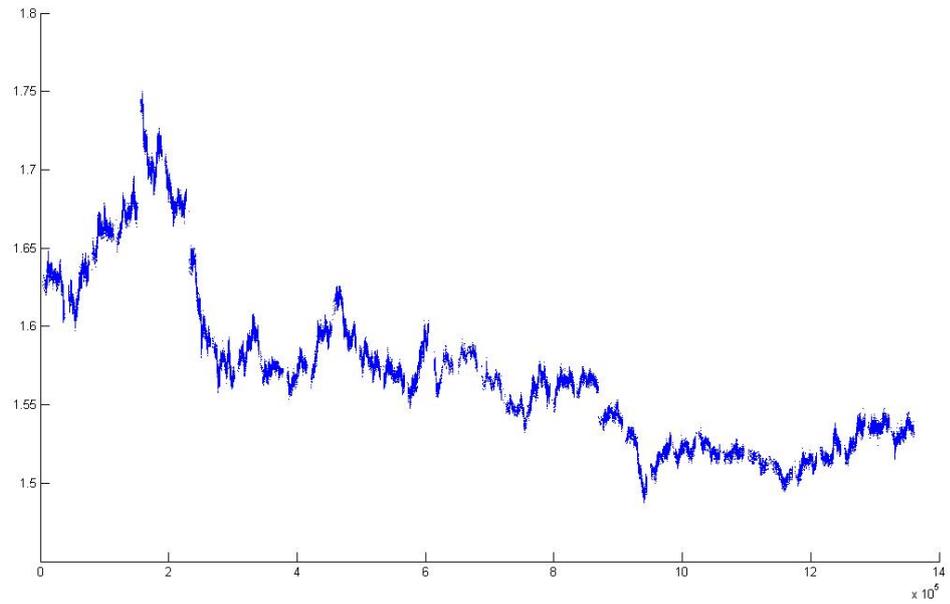
5.1 Structure Of The Data And Rescaling

In this research, we used the classified tick-by-tick data of *USD – TRY* interbank transactions, that had taken place between Nov 14th 2008 and Jan 6th 2009. Our data covers 37 consecutive days and has never been inspected in a research. The trading hours of the market is between 05:00 GMT and 15:30 GMT and that makes 10.5 hours of trading. Our tick size is 104868, which makes 2834 ticks per day and 1 tick per 13 seconds on average. 37 days cover a total of 1,398,600 seconds of trading hours and the scatter diagram of the ticks on a scale of seconds is given in Figure 5.1.

We first applied the rescaling algorithm that is described in Section 4.2. We determined the size of the day to be 28,672 when we consider the described reasons. When we applied the rescaling algorithm, with slide size of 128 (2⁷)seconds, we got the following results:

Table 5.1: Resulting optimized window for rescaling

	START (SECS)	END (SECS)	AVG TICKS
Outlier	0	11943	424
Slice 1	11944	16039	428
Slice 2	16040	20135	361
Slice 3	20136	24231	306
Slice 4	24232	28327	447
Slice 5	28328	32423	422
Slice 6	32424	36519	420
Outlier	36520	37800	115



A total of 104868 trades in 37 consecutive business days are scattered wrt trade timestamps (in seconds). The trading hours is between 05:00 GMT and 15:30 GMT which makes 37800 seconds per day.

Figure 5.1: Scatter diagram of the raw data

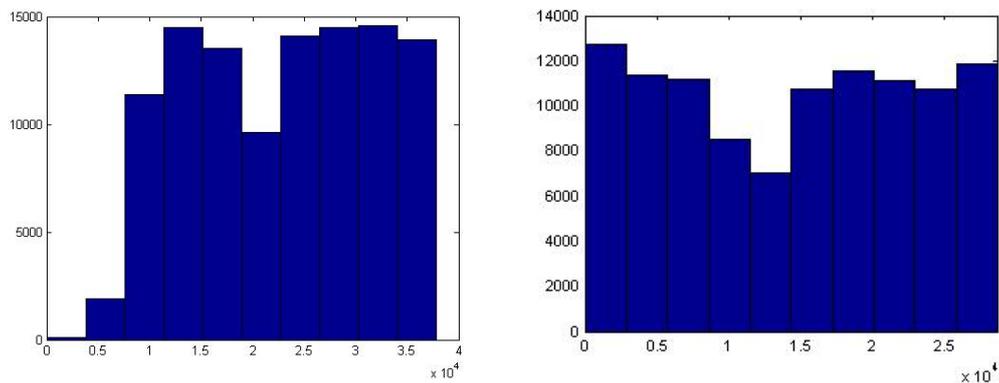


Figure 5.2: Histograms of the data before and after the scaling algorithm is applied

The histograms before and after the scaling algorithm is applied is shown in Figure 5.2. Note that we did not deal with lunch time seasonality in order to keep simplicity but a multi-windowed algorithm could also rescale the period of lunch-time.

5.2 Smoothing

After rescaling, we applied the smoothing algorithm to the outliers of window for full size (539), $\frac{1}{2}$ size (270), $\frac{1}{4}$ size (135) and $\frac{1}{8}$ size (62) to smooth the jumps. We also applied centered average FIR filters with respective size, to our data and included it for comparison. The difference between two series can be seen in Figure 5.3. The total realized variance for the scattered interval are as follows:

Table 5.2: Comparison of several centered FIR filters with our filter

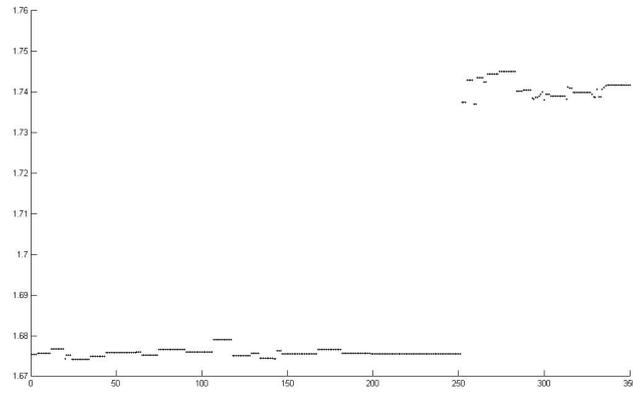
	Realized Variance	Regarding Figure
When jump is ignored	0.01191	Figure 5.3 (a)
When jump is not ignored	0.01323	Figure 5.3 (a)
Centered FIR (n=270)	0.01139	Figure 5.3 (b)
Our Filter (n=270)	0.01141	Figure 5.3 (b)
Centered FIR (n=135)	0.01165	Figure 5.3 (c)
Our Filter (n=135)	0.01167	Figure 5.3 (c)
Centered FIR (n=67)	0.01183	Figure 5.3 (d)
Our Filter (n=67)	0.01187	Figure 5.3 (d)
Centered FIR (n=33)	0.01204	Figure 5.3 (e)
Our Filter (n=33)	0.01226	Figure 5.3 (e)

As can be seen from the table, the quadratic variations of our filter yield better than of centered FIR. Moreover, the scatter of the smoothing for big jumps show that FIR filter oversmooths the data (See Figure 5.3).

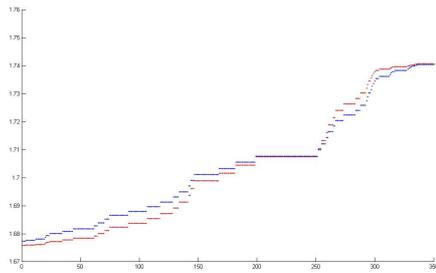
After we have our data rescaled and smoothed, the scatter becomes as it is shown in Figure 5.4.

5.3 Results

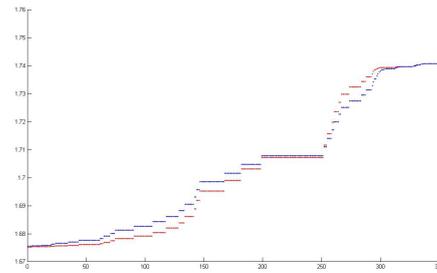
Once we finished our work with rescaling and smoothing, we applied our filter to the data as it is described in Section 4.3. Note that we initially applied the filter without



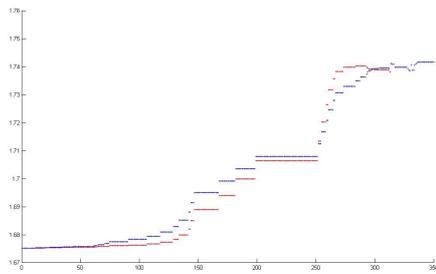
(a) Without smoothing



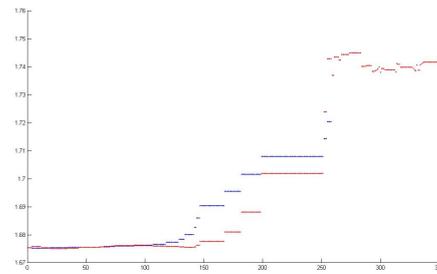
(b) ($n = 270$) Centered FIR and Our Filter



(c) ($n = 135$) Centered FIR and Our Filter



(d) ($n = 67$) Centered FIR and Our Filter



(e) ($n = 33$) Centered FIR and Our Filter

Figure 5.3: Compared scatter diagram of the smoothing process

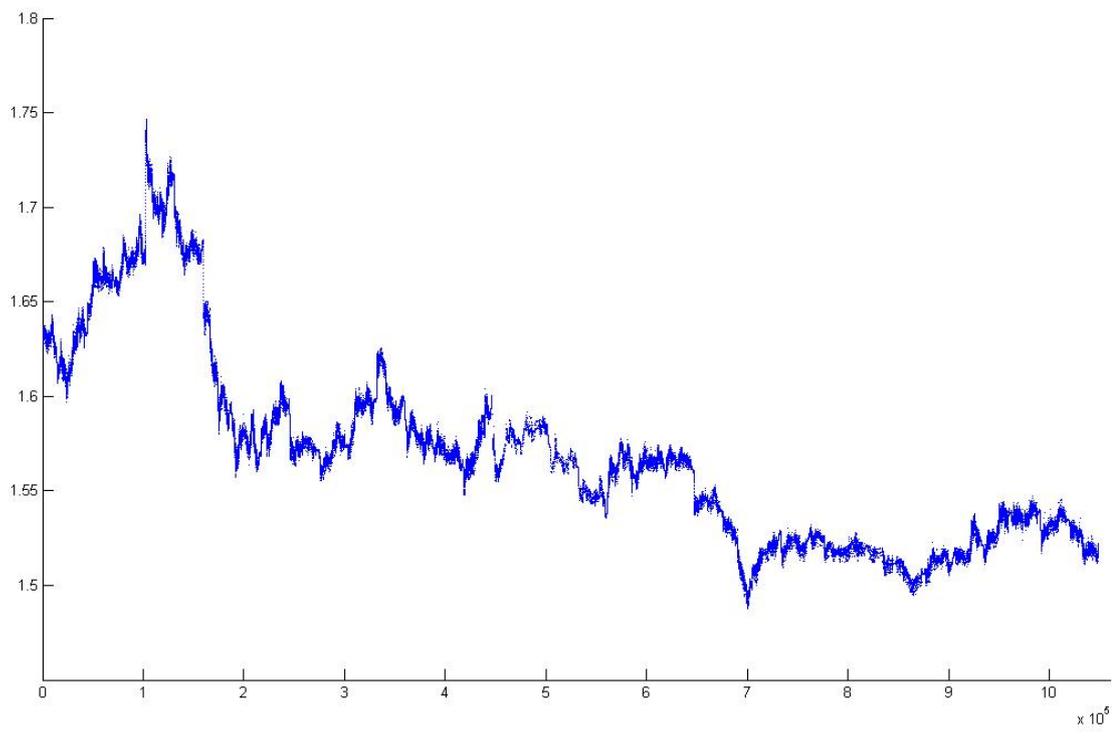


Figure 5.4: Scatter diagram of the data after rescaling and smoothing

fine tuning. In order to compare the results in different sampling frequencies, we chose the sample sizes, which are equivalent to 30 seconds, 1 minute, 2 minutes, 5 minutes, 10 minutes, 15 minutes, 20 minutes, 30 minutes, 40 minutes, 50 minutes, 60 minutes in calendar time. This equivalent to 34953, 17476, 8738, 3495, 1748, 1165, 873, 582, 437, 350 and 291 samples respectively. Here we were expecting to get the best results at the resolution which has a number of coefficients that is closest to the regarding sample size. The relative bias (See 4.3) values of CTS, TTS and our sampling scheme of relative w_j are presented in Table 5.3.

Table 5.3: Relative Bias values for CTS, TTS and our sampling scheme

Sample Size	34953	17476	8738	3495	1748	1165	873	582	437	350	291
CTS	2.82	2.19	1.73	1.40	1.31	1.24	1.25	1.20	1.23	1.14	1.17
TTS	3.46	2.48	1.88	1.43	1.24	1.27	1.23	1.19	1.18	1.15	1.19
w_{19}	3.99	3.32	2.62	1.89	1.50	1.39	1.26	1.25	1.23	1.24	1.23
w_{18}	3.40	2.95	2.48	1.81	1.49	1.34	1.29	1.26	1.18	1.15	1.18
w_{17}	2.97	2.61	2.21	1.73	1.47	1.31	1.20	1.16	1.20	1.18	1.14
w_{16}	2.56	2.30	1.98	1.64	1.37	1.31	1.25	1.16	1.13	1.13	1.13
w_{15}		2.00	1.79	1.51	1.33	1.25	1.19	1.21	1.13	1.10	1.10
w_{14}			1.56	1.41	1.29	1.26	1.17	1.12	1.13	1.14	1.10
w_{13}				1.33	1.28	1.16	1.15	1.14	1.11	1.09	1.09
w_{12}				1.24	1.18	1.20	1.12	1.16	1.21	1.15	1.15
w_{11}					1.12	1.12	1.14	1.11	1.08	1.06	1.08
w_{10}							1.03	1.03	1.02	1.07	0.97
w_{09}									1.01	1.00	1.01

The result of the resolution that gives the best performance is colored with red and the best of CTS-TTS pair is colored with blue in Table 5.3. At higher sampling frequency levels, the relative bias values are generally minimized when the frequency is near regarding resolution, as we have expected. For instance, 34953 (30 seconds in regarding CTS) samples give the best result for w_{16} since the nearest upper resolution is w_{16} (65516). Note that, at w_{15} we only have $2^{15} = 32768$ components in wavelet space and therefore cannot sample for 65516 any further. At lower sampling frequency levels (starting from 873 samples which corresponds to 20 minutes in calendar time sampling), the relative bias results begins fluctuating from some point on and our sampling scheme is not robust anymore. This may be due to the oversmoothing caused by the structure of our scaling function, which is highly discrete at that

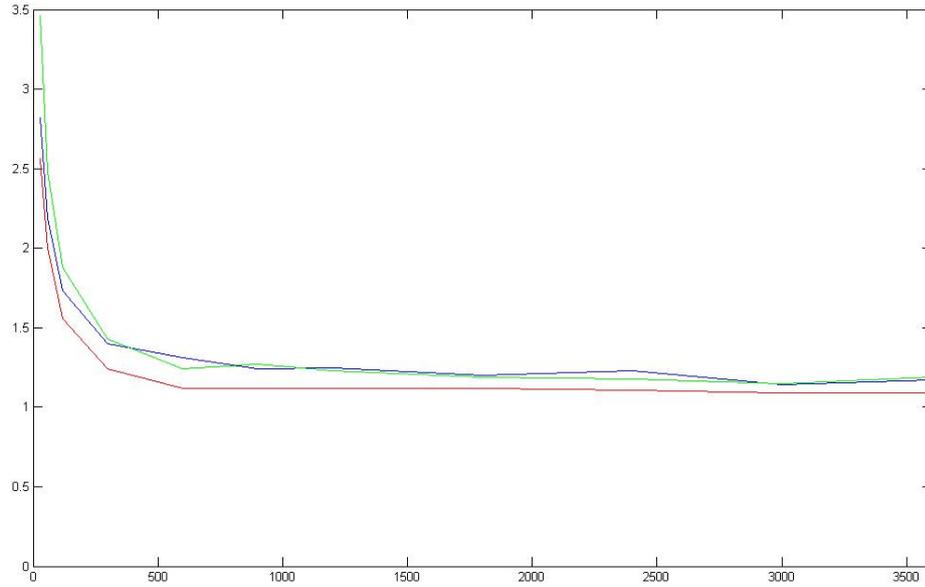


Figure 5.5: Performances (relative bias) of CTS, TTS and our filter over regarding sampling periods (in seconds)

point. But since our design is for high frequencies, we mostly concentrate on higher frequencies and therefore, for lower frequencies, we choose the point where fluctuation begins as the *intuitive best* level. Those *intuitive best* level performances are underlined in Table 5.3. The resulting performances of CTS, TTS and our sampling scheme is shown in Figure 5.5.

Now that we have our first results, we may continue with the fine tuning process that is described in Section 4.3. Note that the orthogonal components of the resolutions were initially decomposed with Figure 3.7 (b) (ϕ_1) which is a highly discrete scaling function. So we kept on with the scaling function iteration procedure. While applying the iteration procedure, our expectation was getting better results with the algorithm, as the scaling function fits the resolution better. The results of the iteration procedure are shown in Table 5.4.

In compliance with our expectations, when we apply the iterative steps to the scaling function (fine tuning), our results even get better. The result of the iteration that gives the best performance is colored with red and the best of CTS-TTS pair is col-

Table 5.4: Relative Bias values for CTS, TTS and the scaling function iterations for best intuitive w_j

Sample Size	34953	17476	8738	3495	1748	1165	873	582	437	350	291
CTS	2.82	2.19	1.73	1.40	1.31	1.24	1.25	1.20	1.23	1.14	1.17
TTS	3.46	2.48	1.88	1.43	1.24	1.27	1.23	1.19	1.18	1.15	1.19
Best w_j	16	15	14	12	11	11	10	10	9	9	10
Best Int. w_j	16	15	14	12	11	11	12	14	13	13	13
Iteration 1	2.56	2.00	1.56	1.24	1.12	1.12	1.11	1.11	1.21	1.15	1.15
Iteration 2	2.48	1.92	1.60	1.21	1.09	1.07	1.12	1.07	1.06	1.06	1.09
Iteration 3	2.42	1.91	1.54	1.20	1.10	1.10	1.16	1.15	1.16	1.11	1.14
Iteration 4	2.39	1.86	1.49	1.26	1.13	1.12	1.10	1.18	1.15	1.07	1.11
Iteration 5	2.32	1.77	1.40	1.23	1.15	1.15	1.12	1.21	1.12	1.15	1.17
Iteration 6	2.20	1.70	1.42	1.22	1.13	1.08	1.14	1.18	1.18	1.19	1.14
Iteration 7	2.06	1.76	1.46	1.22	1.14	1.20	1.12	1.10	1.11	1.11	1.09
Iteration 8	2.14	1.79	1.50	1.24	1.14	1.13	1.27	1.17	1.10	1.13	1.07

ored with blue in Table 5.4. Again we see that, after some iteration level, the results begin to fluctuate. This is due to the increase in number of the coefficients of the scaling function, when iteration is performed and the padding needed for decomposition (See Section 4.3). So again we choose the point where fluctuation begins as the *intuitive best* level. The overall results of the comparison is presented in 5.5 and also plotted in Figure 5.6.

Table 5.5: Final comparison of the models

Sample Size	34953	17476	8738	3495	1748	1165	873	582	437	350	291
CTS	2.82	2.19	1.73	1.40	1.31	1.24	1.25	1.20	1.23	1.14	1.17
TTS	3.46	2.48	1.88	1.43	1.24	1.27	1.23	1.19	1.18	1.15	1.19
Best Int. w_j	16	15	14	12	11	11	12	14	13	13	13
Best Int.SF Iter.	7	6	5	3	2	2	1	2	2	2	2
Value	2.06	1.70	1.40	1.20	1.09	1.07	1.10	1.07	1.06	1.06	1.09

It can be seen clearly from Figure 5.6 that our filter, both with and without scaling function iteration, performs better than CTS and TTS for our data. Some plots of the decompositions and iterative procedures are given in the Appendix.

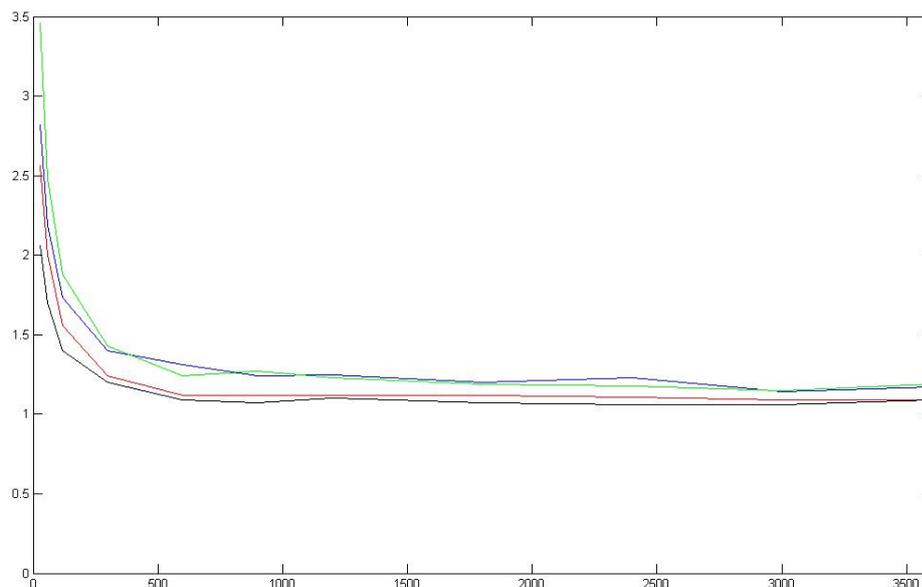


Figure 5.6: Performances (relative bias) of **CTS**, **TTS**, **our filter w/o sf iteration** and **our filter with sf iteration** over regarding sampling periods (in seconds)

5.4 Conclusion And Further Research Topics

In this research, we tried to create a new sampling scheme to estimate the integrated volatility by realized variance with a minimal bias caused by the microstructure effects. We applied the multiscaling properties of wavelets to our high-frequency (tick-by-tick) data and created a new sampling scheme by creating a new business time scale depending on the quadratic variation of the orthogonal wavelet space projections of the data to each resolution. The motivation of this approach was the thought that the best representation of the characteristics of the data sampled at a resolution should be the projection of the data to the orthogonal components at this resolution level.

The common practice in the market was to use Calendar Time Sampling (CST) with a Δt between 5 minutes and 20 minutes. Tick Time Sampling (TTS) was also recommended for some cases. So we used CST and TTS for comparison. We used the realized variance of daily returns as reference and compared the relative bias of our sampling scheme and the others. The result of the comparison was always in

favour of our sampling scheme. After optimization, our sampling scheme yielded more half the bias of the better of CTS and TTS at lower frequencies. It also yielded significantly better in every frequency.

So we conclude that our sampling scheme may be used for sampling, instead of CTS and TTS for it gives the relative bias that CTS gives at 5 minutes at a frequency equal to sampling every 2 minutes in calendar time.

As we have stated in this work, our model is optimizable since we can make fine tuning by a better approximation of the scaling function that spans the basis for decomposing the signal into orthogonal components. But this procedure suffers from another source of bias, after some number of iterations. This is due to padding necessity regarding the decomposition procedure. Padding of 46 data points (4th iteration) may not cause that much of a bias for a total of 2^{19} data points, but it causes a lot of bias for a total of 512 datapoints. So, further researches may focus on this bias that is caused by padding and provide a better fine tuning.

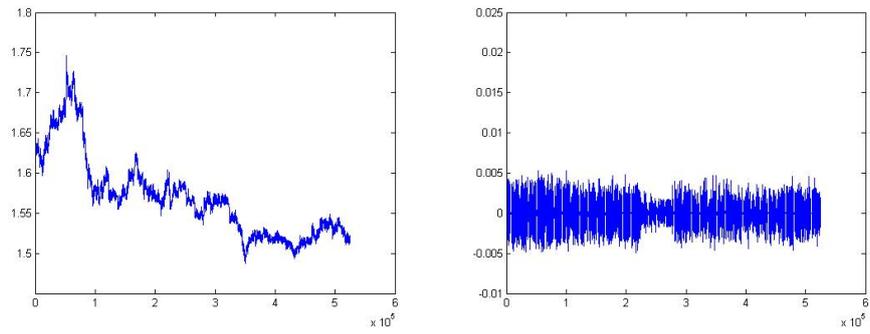
REFERENCES

- [Ait-Sahalia 2005] Y.Ait-Sahalia, P.A.Mykland and L.Zhang *How Often to Sample a Continuous-Time Process in the Presence of Market Microstructure Noise*, Review of Financial Studies, Vol.18 pp.351-416
- [Andersen 2000] T.G.Andersen, T.Bollerslev, F.X.Diebold and P.Labys *Exchange rate returns standardized by realized volatility are (nearly) Gaussian*, NBER Working Paper No:7488
- [Andreou 2002] E.Andreou and E.Ghysels *Rolling-Sample Volatility Estimators: Some New Theoretical, Simulation, and Empirical Results*, Journal of Business and Economic Statistics, Vol.20 pp.363-376
- [Bachman 2000] G.Bachman, L.Narici and E.Beckenstein *Fourier and Wavelet Analysis*, 1st ed., Springer
- [Bandi 2003] F.M.Bandi and J.R.Russell *Microstructure Noise, Realized Volatility and Optimal Sampling*, technical report, University of Chicago
- [Boggess 2001] A.Boggess and F.J.Narcowich *A First Course in Wavelet Analysis with Fourier Series*, 1st ed., Prentice Hall
- [Clark 1973] P.Clark *A Subordinated Stochastic Process Model with Finite Variance for Speculative Prices*, Econometrica, Vol.41 pp.135-156
- [Dacorogna 2001] M.M.Dacorogna, R.Gençay, U.Müller, R.B.Olsen and O.V.Pictet *An Introduction To High-Frequency Finance*, 1st ed., Academic Press
- [Delacherie 1982] C.Delacherie and P.A.Meyer *Probabilities and Potential*, , B.North Holland
- [Gençay 2001] R.Gençay, F.Selçuk and B.Whitcher *Differentiating intraday seasonalities through wavelet multi-scaling*, Physica A, vol. 289, pp.543-556
- [Gençay 2002] R.Gençay, F.Selçuk and B.Whitcher *An Introduction to Wavelets and Other Filtering Methods In Finance*, 1st ed., Academic Press
- [Hasbrouck 2004] J.Hasbrouck *Empirical Market Microstructure: Economic and Statistical Perspectives on the Dynamics of Trade in Securities Markets*, lecture notes, New York University
- [Hosking 1996] J.Hosking *Asymptotic distribution of the sample mean, autocovariances and autocorrelations of long memory time series*, Journal of Econometrics, Vol.73 pp.261-284
- [Hull 1987] J.Hull and A.White *The Pricing of Options on Assets With Stochastic Volatilities*, Journal of Finance, Vol.42 pp.281-300

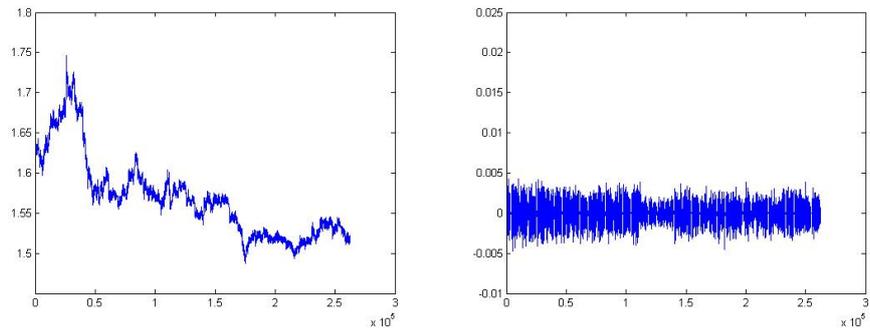
- [Karatzas 1991] I.Karatzas and S.E.Shreve *Brownian Motion and Stochastic Calculus*, 2nd ed., Springer
- [Koralov 2007] L.B.Koralov and Y.G.Sinai *Theory of Probability and Random Processes*, 2nd ed., Springer
- [Mallat 1989] S.G.Mallat *A Theory for Multi-resolution Approximation: The Wavelet Approximation*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.11 pp.674-693
- [Mallat 1998] S.G.Mallat *A Wavelet Tour of Signal Processing*, 2nd ed., Academic Press
- [Merton 1980] R.C.Merton *On Estimating the Expected Return on the Market: An Exploratory Investigation*, Journal of Financial Economics, Vol.8 pp.323-361
- [Morlet 1984] J.Morlet and A.Grossman *Decomposition of Hardy Functions Into Square Integrable Wavelets of Constant Shape*, SIAM Journal of Mathematical Analysis, Vol.15 pp.723-736
- [O'Hara 1997] Maureen O'Hara *Market Microstructure Theory*, 1st ed., Blackwell Business
- [Oomen 2005] R.A.C Oomen *Properties of Bias-Corrected Realized Variance Under Alternative Sampling Schemes*, Journal of Financial Econometrics, Vol 3, pp.555-577
- [Oomen 2006] R.A.C Oomen *Properties of Realized Variance Under Alternative Sampling Schemes*, Journal of Business and Economic Statistics, Vol 24 No 2, pp.219-237
- [Percival 2000] D.B.Percival and A.T.Walden *Wavelet Methods for Time Series Analysis*, 1st ed., Cambridge University Press
- [Rogers 1994.v1] L.C.G.Rogers and D.Williams *Diffusions, Markov Processes and Martingales, Vol 1. Foundations*, 2nd ed., Cambridge Mathematical Library
- [Rogers 1994.v2] L.C.G.Rogers and D.Williams *Diffusions, Markov Processes and Martingales, Vol 2. Ito Calculus*, 2nd ed., Cambridge Mathematical Library
- [Wojtaszczyk 1997] P.Wojtaszczyk *A Mathematical Introduction to Wavelets*, 1st ed., Cambridge University Press (LMSST vol.37)
- [Zhang 2005] L.Zhang, Y.Ait-Sahalia and P.A.Mykland *A Tale of Two Time Scales: Determining Integrated Volatility With Noisy High-Frequency Data*, Journal of the American Statistical Association, Vol.100 No.472 pp.1394-1411

APPENDIX A

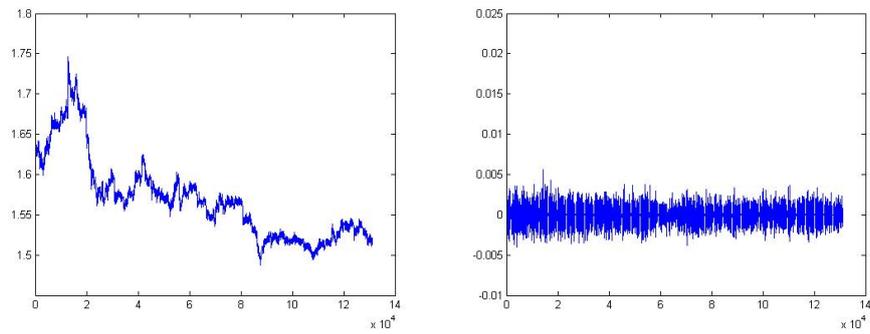
DECOMPOSITION



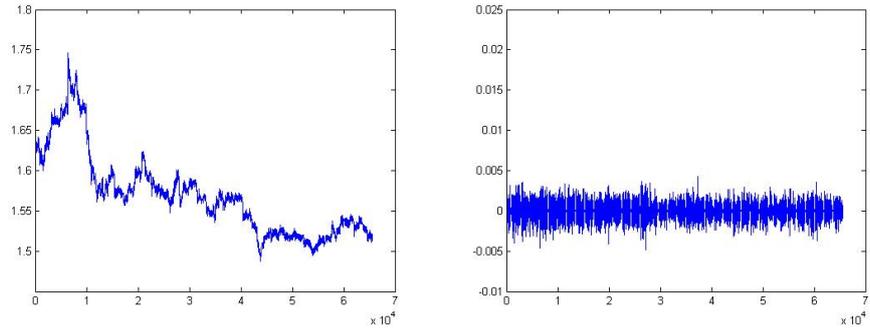
(1-a) V_{19} and w_{19} with no iterations on scaling function



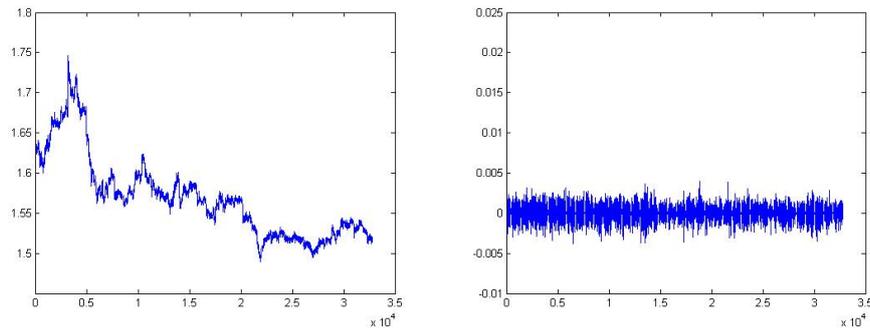
(1-b) V_{18} and w_{18} with no iterations on scaling function



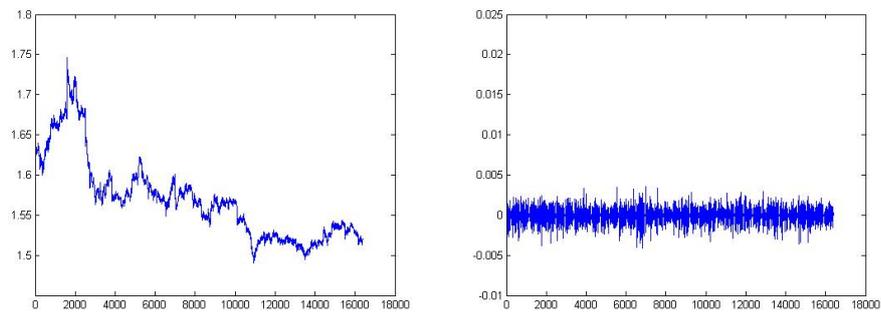
(1-c) V_{17} and w_{17} with no iterations on scaling function



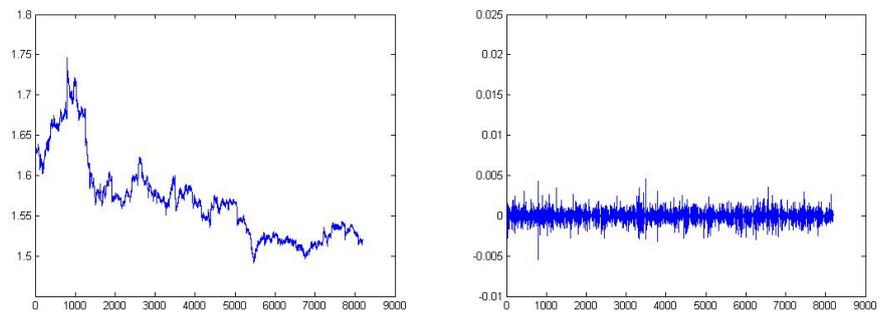
(1-d) V_{16} and w_{16} with no iterations on scaling function



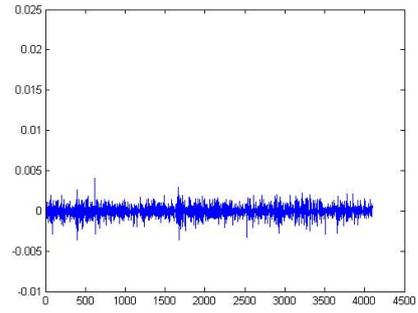
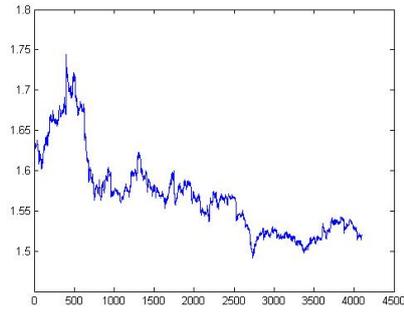
(1-e) V_{15} and w_{15} with no iterations on scaling function



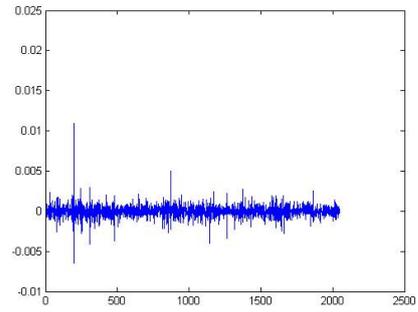
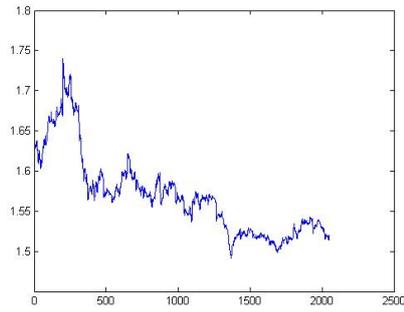
(1-f) V_{14} and w_{14} with no iterations on scaling function



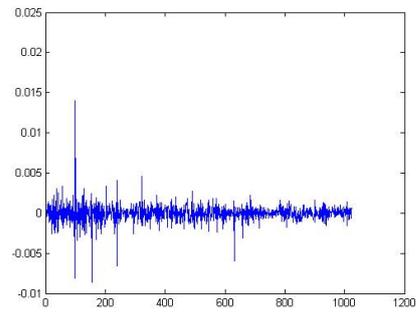
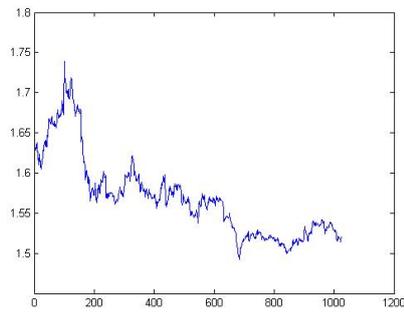
(1-g) V_{13} and w_{13} with no iterations on scaling function



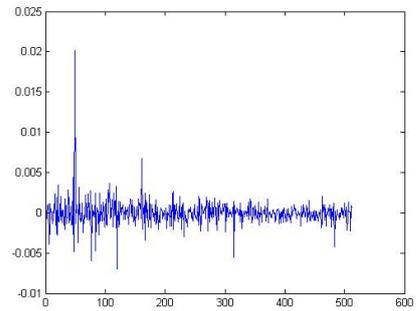
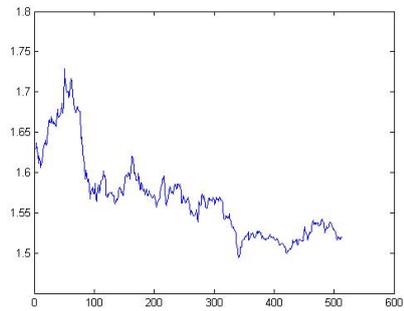
(1-h) V_{12} and w_{12} with no iterations on scaling function



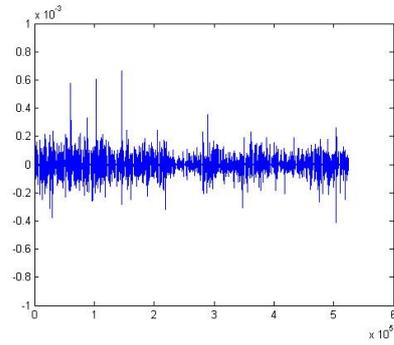
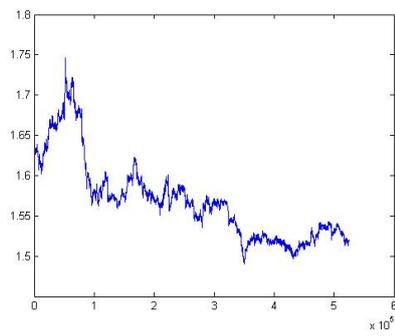
(1-i) V_{11} and w_{11} with no iterations on scaling function



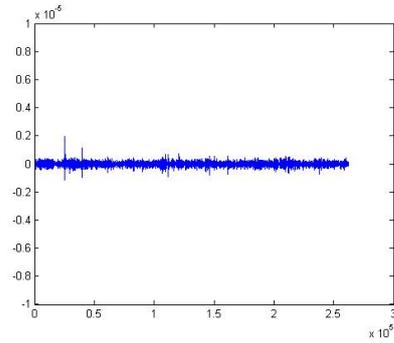
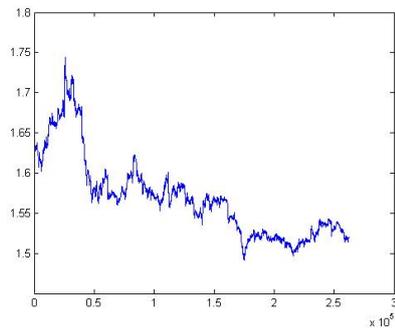
(1-j) V_{10} and w_{10} with no iterations on scaling function



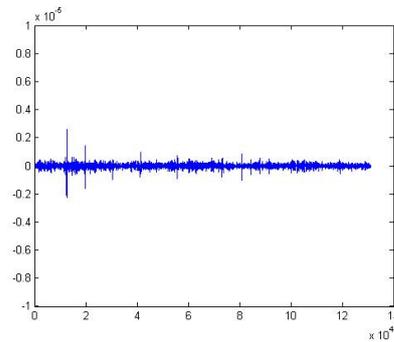
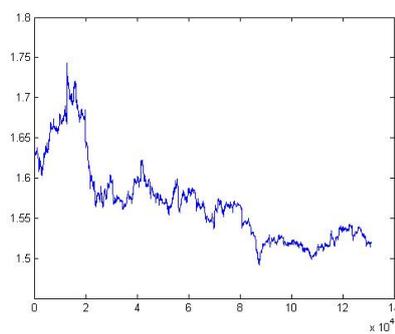
(1-k) V_9 and w_9 with no iterations on scaling function



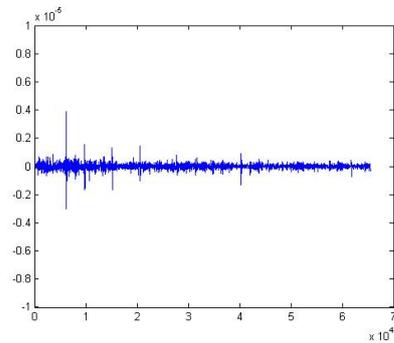
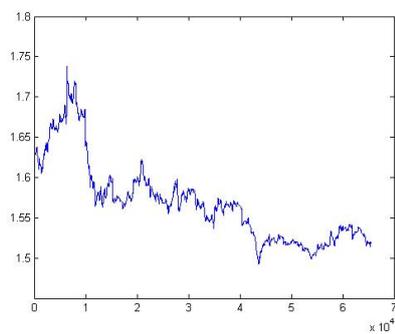
(2-a) V_{19} and w_{19} with 7 iterations on scaling function



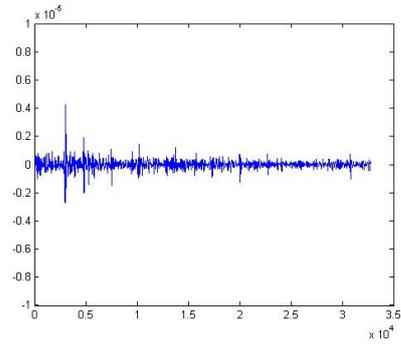
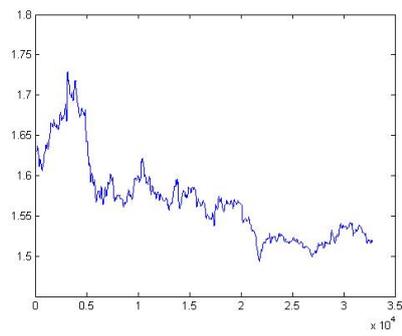
(2-b) V_{18} and w_{18} with 7 iterations on scaling function



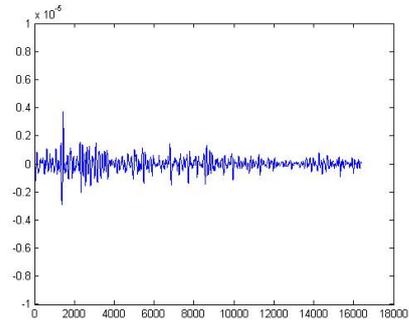
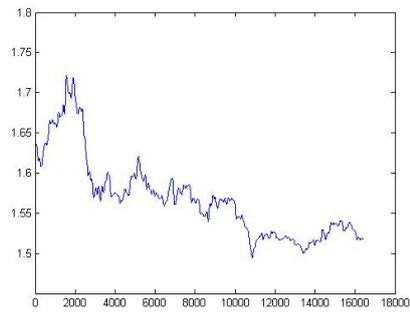
(2-c) V_{17} and w_{17} with 7 iterations on scaling function



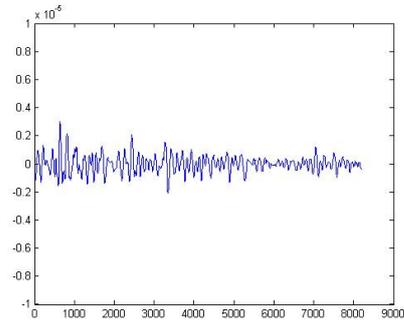
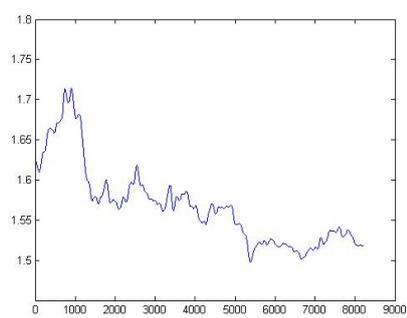
(2-d) V_{16} and w_{16} with 7 iterations on scaling function



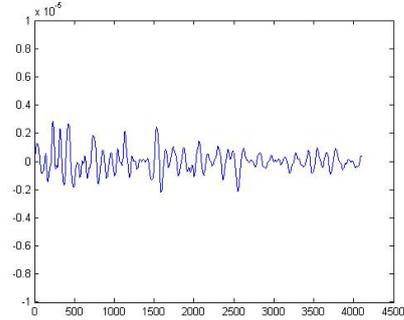
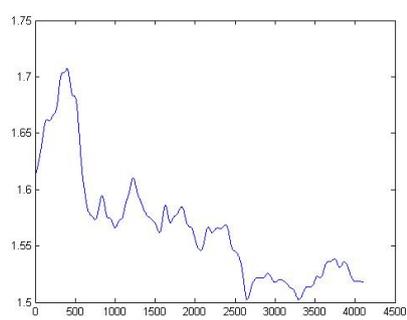
(2-e) V_{15} and w_{15} with 7 iterations on scaling function



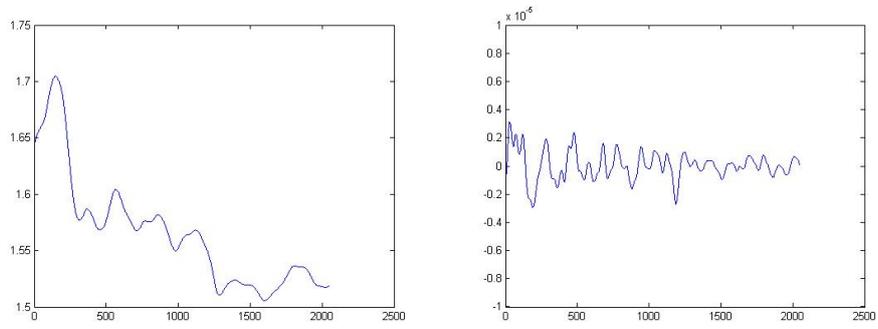
(2-f) V_{14} and w_{14} with 7 iterations on scaling function



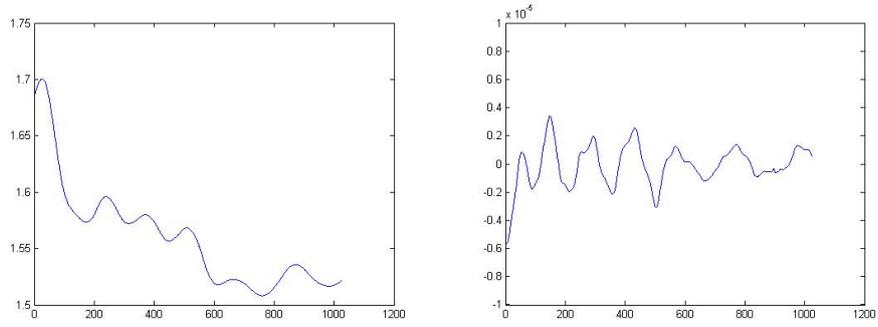
(2-g) V_{13} and w_{13} with 7 iterations on scaling function



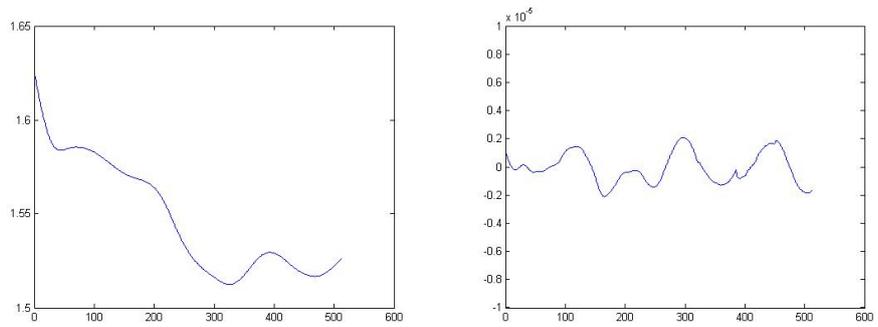
(2-h) V_{12} and w_{12} with 7 iterations on scaling function



(2-i) V_{11} and w_{11} with 7 iterations on scaling function



(2-j) V_{10} and w_{10} with 7 iterations on scaling function



(2-k) V_9 and w_9 with 7 iterations on scaling function