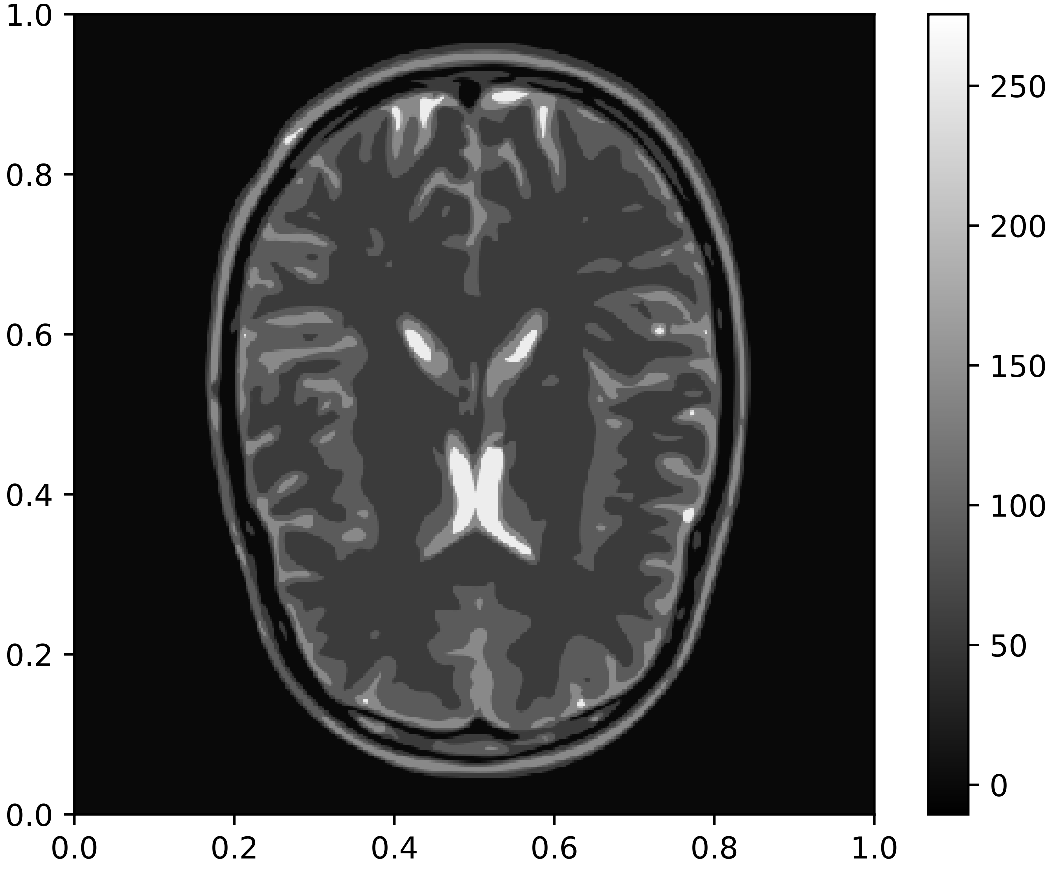
Generalized Sampling

From Fourier to Wavelet

Master Thesis

Josefine Holm and Steffen Lønsmann Nielsen



Aalborg University
Mathematical Engineering

AALBORG UNIVERSITY
STUDENT REPORT

Department of Electronic Systems
& Department of Mathematical Sciences
Aalborg University
http://www.aau.dk

Title:
Generalized Sampling:
From Fourier to Wavelet

Theme:
Master Thesis

Project Period:
Fall semester 2017 and
spring semester 2018

Project Group:
5213a

Participants:
Josefine Holm
Steffen Lønsmann Nielsen

Supervisors:
Morten Nielsen
Thomas Arildsen

Copies: 3

Page Numbers: 93

Date of Completion:
June 6, 2018

Abstract:

In this project we investigate generalized sampling as a tool for signal reconstruction and compression. Generalized sampling is a relatively new method for recovering any element in a finite dimensional space given finitely many samples in an arbitrary frame. The focus is on Fourier frames as sampling space and Daubechies wavelets as reconstruction space. We investigate the subject both in theory and in practise by proving relevant theorems and implementing algorithms in Python. Most of the theory is already published by others. However, to the best of our knowledge, it has not been implemented in Python before. The method is tested on several different signals with overall positive results. Among the test signals are both continuous and discontinuous signals, signals in one and two dimensions, and uniformly and nonuniformly sampled signals. For most of the tested signals compression using generalized sampling results in smaller errors than compression directly in the Fourier frame.

Preface

This master thesis is written in the period from 04-09-2017 to 06-06-2018 by master students in Mathematical Engineering at Aalborg University. The thesis is about generalized sampling. We want to thank our supervisors Morten Nielsen from Department of Mathematical Sciences and Thomas Arildsen from Department of Electronic Systems for advice and guidance throughout the writing of the thesis.

Aalborg University, June 6, 2018

Josefine Holm
<josefineholm94@gmail.com>

Steffen Lønsmann Nielsen
<steffen.l.n@hotmail.com>

Contents

Reading guide

The thesis will use the Havard reference style when citing sources.

The definitions, theorems, etc. are colour coded, i.e. there is a bar to the left to each of these. The colors are: Definition, purple; Theorem, dark blue; Corollary, blue; Lemma, light blue. Proofs and remarks have no colors, but proofs will end with □, while remarks will end with ◂.

In extension of the thesis are two appendixes. Appendix A contains larger versions of specific images while Appendix B contains the Python scripts and modules developed throughout the project. Appendix B can be found in `Appendix_B.rar`, in which there is a Readme file explaining the content.

The thesis has been written and compiled in LaTeX, and all computer based calculations have been made in Python 3.x.

In the thesis, some symbols are commonly used. Some of these are explained in the table below.

Symbol	Explanation
$\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}, \mathbb{C}$	The natural numbers, integers, rational numbers, real numbers and complex numbers respectively.
$\chi_I(x)$	The indicator function, i.e. the function is 1 $\forall x \in I$ and 0 otherwise.
$\overline{A}$	Complex conjugate of $A \in \mathbb{C}^{n \times m}$.
A^*	Complex conjugate and transpose of $A \in \mathbb{C}^{n \times m}$.
$\bullet$	A place holder for the argument in a function.
$\overline{V}$	The closure of a space V.
$\oplus$	The direct sum of linear spaces.
$x^\star$	The minimizer of an optimization.
`Text`	The text style for scripts, modules and functions in Python.

Danish Summary

Specialet har den danske titel Generaliseret Sampling: Fra Fourier til Wavelet. Her undersøges generaliseret sampling - både teoretisk og anvendt i praksis. Den praktiske del består af forsøg med syntetiske testsignaler og vil blive gennemført i programmeringsspoget Python. Målet med specialet er at opnå en dyb og intuitiv forståelse af emnet og lave en implementation i Python.

Specialet startes med en beskrivelse af motivationen for emnet, samt generaliseret sampling i relation til Nyquist-Shannons samplingssætning. Herefter gennemgås grundlæggende funktionsteori for at opnå en bedre forståelse af teorien for generaliseret sampling, som følger umiddelbart efter. Dernæst følger et afsnit om Fourier frames, der bruges som samplingsrum. Herefter beskrives de valgte rekonstruktionsrum: Daubechies wavelets. Heri er blandt andet udledt en eksplicit måde at lave randfunktioner på, både i tids- og frekvensdomænet, som, der efter forfatternes bedste overbevisning, ikke er nogen, der har gjort på denne måde før. Herefter kommer et afsnit med nogle af de vigtigste sætninger i et praktisk perspektiv. Sætningerne angiver, hvor mange Fourier samples man skal have til et givent antal wavelet koefficienter samt, hvor tæt man skal sample frekvenserne, for at være sikker på at generaliseret sampling går godt.

Når den generelle teori for generaliseret sampling er på plads, fokuseres der på det specifikke tilfælde med konstruktion af en basisskiftmatrix mellem Fourier samples og wavelet koefficienter. En af ulemperne er, at disse matricer hurtigt bliver meget store, derfor bliver der beskrevet en måde at lave dette basis skift på, uden at gemme den egentlige matrix. Disse algortimer skal også bruges i optimeringsmetoder for at finde de bedste wavelet koefficienter.

Herefter begynder de praktiske eksperimenter. Først undersøges endimensionale signaler med rekonstruktion i Haar waveletten, efterfulgt af Daubechies 2 waveletten. Det kan betragtes som en kompression, når man laver basisskiftet, og generelt er resultaterne gode. Hvis de sammenlignes med en af de klassiske løsninger på komprimering af Fourier samples, altså at beholde de N største koefficienter og sætte resten til nul, ser man en tydelig forbedring. Hvis signalerne, der bliver brugt, er godt repræsenteret i samplingsrummet, vinder man ikke noget ved basisskiftet.

De gode resultater motiverede forfatterne til at lave nogle simple eksperimenter med todimensionale signaler, altså signaler der kan betragtes som billeder. Her ses at de samme ting gør sig gældende som i én dimension.

I diskussionen bliver emner, som oplagt kan udforskes i forlængelse af dette speciale, diskuteret. Konklusionen beskriver resultatet af specialet: Både forståelsen af emnet og resultaterne af eksperimenterne.

Chapter 1

Introduction

In the thesis we investigate generalized sampling and its application in signal reconstruction and compression. We walk the reader through the theory on the topic and use examples to give a more intuitive understanding of the material. The examples will focus on Fourier samples and wavelet reconstruction, as this is a classic example in the use of generalized sampling, [Adcock et al., 2014b], [Jacobsen et al., 2016b] and [Gataric and Poon, 2016]. The goal of the project is to investigate and understand the theory of generalized sampling and to implement appropriate algorithms in Python 3.x.

The contribution of this project lies in the application of generalized sampling. We will describe and implement algorithms which reconstruct a given signal from its Fourier samples using fewer or the same number of wavelet coefficients. Some important algorithms are based on [Gataric and Poon, 2016]. The Fourier basis is well known, [Folland, 1992] and [Christensen, 2008], but for use in generalized sampling we need a Fourier frame. Such a frame has been described in theory, [Christensen et al., 2016], but it is unconventional in application. In the thesis, we describe how to use a Fourier frame to sample the frequency domain in a subset of $\mathbb{R}$ instead of $\mathbb{Z}$, as with the Fourier basis. The systems of Daubechies wavelets are thoroughly described and widely used, [Daubechies, 1992], [Cohen et al., 1993] and [Nievergelt, 1999]. In this project we have taken a different approach to boundary correction, [Jawerth and Sweldens, 1994], than the classic, [Cohen et al., 1993], and created explicit boundary functions both in the time domain and in the frequency domain.

Parts of the theory are general, but in order to focus on application we need to restrict our work in several ways. First of all, we have chosen to sample in a Fourier frame and reconstruct in a Daubechies wavelet basis. With this choice we also limit the input signals to signals in the Hilbert spaces L^2 and ℓ^2. The examples will mainly include uniformly sampled inputs. All algorithms are able to handle nonuniform inputs. However, research into what constitutes a good sampling scheme is outside the scope of this project. Research into this has been done for instance in [Jacobsen et al., 2016a]. We will limit the work with nonuniform input to testing of the algorithms. In the case of two dimensional signals, we limit our examples to quadratic signals and only attempt to view the problem as a tensor product of one-dimensional problems.

The layout of the thesis is as follows: In Chapter 2 we motivate the use of generalized sampling. In Chapter 3 we state the necessary theoretical foundation and in Chapter 4 the theory specifically related to generalized sampling is introduced. The use of a Fourier frame as sampling space is derived in Section 4.2. Chapter 5 describes the Daubechies wavelet basis and the derivation of the boundary functions in the time domain. Chapter 6 describes the construction of the matrix which maps wavelet coefficients to Fourier coefficients and how to use it to find wavelet coefficients given the Fourier samples. Examples of generalized sampling used on one dimensional signals can be found in Chapter 7. Chapter 8 contains the work done with two dimensional signals and in Chapter 9 we discuss the chosen restrictions and possible extensions of the project and finally concludes on the project. Larger versions of a few pictures can be found in Appendix A. All Python files related to the thesis are collected in Appendix B, which also includes a `ReadMe.txt` file explaining the content. Appendix B is not directly included in the thesis but is a separate file.

Chapter 2

Motivation

In this chapter we motivate the use of generalized sampling. First, we will present a generalization of the well known Nyquist-Shannon sampling theorem. Second, we will give additional reasons for using a Fourier frame for sampling and a wavelet basis for reconstruction. Some of the concepts used in this chapter will first be explained in Chapter 3.

2.1 Generalizing Nyquist-Shannon's Sampling Theorem

The Nyquist-Shannon sampling theorem, as given in Theorem 2.1, is commonly used in modern signal processing and is an important theorem in mathematics of information. It is for instance used in magnetic resonance imaging (MRI). In [Adcock and Hansen, 2012] it is investigated whether the theorem can be improved and, in particular, if the reconstruction can be better given the same information. This depends much on the functions considered, but if a basis for a suitable space is known, in which the function is well represented, improvement might be possible.

Theorem 2.1 (Nyquist-Shannon Sampling Theorem)
Let $x(t)$ be a signal in $L^2(\mathbb{R})$ with bandlimit $K > 0$, i.e. the Fourier transform, $\hat{x}(\omega)$, satisfies

$$\hat{x}(\omega) = 0 \quad \text{for } |\omega| \geq K. \tag{2.1}$$

Then $x(t)$ is uniquely determined by its samples $x_s(n) = x(n \cdot T)$, where $n = 0, \pm 1, \pm 2, \ldots$ and T is the sampling period, if the sampling frequency $\omega_s = \frac{2}{T} > 2K$. We can write $x(t)$ as

$$x(t) = \sum_{n=-\infty}^{\infty} x\left(\frac{n\pi}{K}\right) 2\pi \frac{\sin(\pi n + 2\pi K t)}{\pi n + 2\pi K t}. \tag{2.2}$$

[Oppenheim and Schafer, 2010, p. 170] and [Folland, 1992, p. 230]

Proof: Let us expand $\hat{x}$ in a Fourier series on the interval $[-K, K]$ noting that $\hat{x} \in L^1$. We write $-n$ in place of n for reasons of later convenience,

$$\hat{x}(\omega) = \sum_{n=-\infty}^{\infty} c_{-n} e^{-in\pi\frac{\omega}{K}}, \quad |\omega| \le K. \tag{2.3}$$

The Fourier coefficients c_{-n} are given by

$$c_{-n} = \frac{1}{2K}\int_{-K}^{K} \hat{x}(\omega) e^{in\pi\frac{\omega}{K}} d\omega = \frac{\pi}{K} x\left(\frac{n\pi}{K}\right). \tag{2.4}$$

Using (2.3) and (2.4) we get

$$\begin{aligned} x(t) = \int_{-K}^{K} \hat{x}(\omega) e^{2\pi i\omega t} d\omega &= \int_{-K}^{K} \sum_{n=-\infty}^{\infty} c_{-n} e^{-in\pi\frac{\omega}{K}} e^{2\pi i\omega t} d\omega \\ &= \int_{-K}^{K} \sum_{n=-\infty}^{\infty} \frac{\pi}{K} x\left(\frac{n\pi}{K}\right) e^{-in\pi\frac{\omega}{K}} e^{2\pi i\omega t} d\omega \\ &= \frac{\pi}{K} \sum_{n=-\infty}^{\infty} x\left(\frac{n\pi}{K}\right) \int_{-K}^{K} e^{\pi i\omega(\frac{n}{K}+2t)} d\omega \\ &= \sum_{n=-\infty}^{\infty} x\left(\frac{n\pi}{K}\right) \frac{e^{\pi i\omega(\frac{n}{K}+2t)}}{Ki(\frac{n}{K}+2t)}\bigg|_{-K}^{K} \\ &= \sum_{n=-\infty}^{\infty} x\left(\frac{n\pi}{K}\right) 2\pi \frac{\sin(\pi n + 2\pi Kt)}{\pi n + 2\pi Kt}, \end{aligned} \tag{2.5}$$

which concludes the proof. □

This is a universal theorem which is widely used in signal processing. If the bandlimit is all you know about a signal, then ω_s is the lower boundary for the sampling frequency. When trying to reconstruct a signal based on fewer samples than Theorem 2.1 dictates, we have to exploit additional information we have about the signal.

The problem we are looking at is how to reconstruct a signal, f, based on a finite number of samples in a particular frame, in another frame better suited for the signal. Let $\{s_k\}_{k\in\mathbb{N}}$ denote the sampling frame and $\{\langle f, s_k\rangle\}_{k\in\mathbb{N}}$ be the samples. Let $\{\alpha_k\}_{k\in\mathbb{N}}$ be coefficients such that

$$f = \sum_{k=1}^{\infty} \alpha_k w_k, \tag{2.6}$$

where $\{w_k\}_{k\in\mathbb{N}}$ is a frame well suited for the signal. w_k is referred to as the reconstruction frame. The goal is to use a finite number of samples to compute a finite number of coefficients which reconstruct the signal with a sufficiently small error.

The generalisation of Theorem 2.1 proposed in [Adcock and Hansen, 2012] requires that we introduce infinite matrices and sections thereof. Let U denote the matrix for a change of basis operator from $\ell^2(\mathbb{N})$ to $\ell^2(\mathbb{Z})$ which has the form

$$U = \begin{bmatrix} \vdots & \vdots & \vdots & \iddots \\ u_{-1,1} & u_{-1,2} & u_{-1,3} & \cdots \\ u_{0,1} & u_{0,2} & u_{0,3} & \cdots \\ u_{1,1} & u_{1,2} & u_{1,3} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}, \quad u_{i,j} = \langle s_i, w_j \rangle. \tag{2.7}$$

Let P_n denote the orthogonal projection onto the span of the first n basis vectors for $\ell^2(\mathbb{Z})$.

The paper propose a change of basis:

$$A \begin{bmatrix} \tilde{\alpha}_1 \\ \tilde{\alpha}_2 \\ \vdots \\ \tilde{\alpha}_n \end{bmatrix} = P_n U^* P_m \begin{bmatrix} \langle f, s_1 \rangle \\ \langle f, s_2 \rangle \\ \vdots \\ \langle f, s_m \rangle \end{bmatrix}, \quad A = P_n U^* P_m U P_n|_{P_n \mathcal{H}}. \tag{2.8}$$

Then

$$\tilde{f} = \sum_{k=1}^{n} \tilde{\alpha}_k w_k. \tag{2.9}$$

Theorem 2.2 (Abstract Sampling Theorem)
Let $\mathcal{H}$ be a separable Hilbert space and $\mathcal{S}, \mathcal{W} \subset \mathcal{H}$ be closed subspaces such that $\mathcal{W} \cap \mathcal{S}^\perp = \{0\}$. Assume that $\{s_k\}_{k \in \mathbb{N}}$ and $\{w_k\}_{k \in \mathbb{N}}$ are Riesz bases for $\mathcal{S}$ and $\mathcal{W}$ respectively, with frame bounds $A, B > 0$ for $\{w_k\}_{k \in \mathbb{N}}$ and $C, D > 0$ for $\{s_k\}_{k \in \mathbb{N}}$. Assume that

$$f = \sum_{k \in \mathbb{N}} \alpha_k w_k, \qquad \alpha = \{\alpha_1, \alpha_2, \dots\} \in \ell^2(\mathbb{N}), \tag{2.10}$$

and let $n \in \mathbb{N}$. Then there exists an $M \in \mathbb{N}$ (in particular $M = \min\{\sigma : 0$ is not an eigenvalue for $P_n U^* P_\sigma U P_n|_{P_n \mathcal{H}}\}$) such that for all $m \geq M$, the solution of (2.8) is unique. Furthermore, if $\tilde{f}$ is as in (2.9), then

$$||f - \tilde{f}||_{\mathcal{H}} \leq \sqrt{B}(1 + K_{n,m})||P_n^\perp \alpha||_{\ell^2(\mathbb{N})}, \tag{2.11}$$

where

$$K_{n,m} = ||(P_n U^* P_m U P_n|_{P_n \mathcal{H}})^{-1} P_n U^* P_m U P_n^\perp||_{\ell^2(\mathbb{N})}. \tag{2.12}$$

[Adcock and Hansen, 2012, Theorem 4.1]

This theorem states that the error we make when using generalized sampling is bounded. We have some control of this bound with our choice of sampling and reconstruction space. The factor B in the bound is the upper frame bound of the reconstructions space which will be defined in Definition 3.6. The factor $K_{n,m}$ can be bounded as in Corollary 2.3.

Corollary 2.3
Let the setup be as in Theorem 2.2 and let n be fixed. Then

$$\begin{aligned}\|(P_n U^* P_m U P_n|_{P_n\mathcal{H}})^{-1}\|_{\ell^2(\mathbb{N})} &\xrightarrow{m\to\infty} \|(P_n U^* U P_n|_{P_n\mathcal{H}})^{-1}\|_{\ell^2(\mathbb{N})} \\ &\leq \|(U^*U)^{-1}\|_{\ell^2(\mathbb{N})} \leq \frac{1}{AC}.\end{aligned} \tag{2.13}$$

Furthermore, if U is an isometry, in particular when $\{w_k\}_{k\in\mathbb{N}}$ and $\{s_k\}_{k\in\mathbb{N}}$ are orthonormal, then it follows that

$$K_{n,m} \xrightarrow{m\to\infty} 0. \tag{2.14}$$

[Adcock and Hansen, 2012, Corollary 4.2]

2.2 Magnetic Resonance Imaging

In generalized sampling the most widely used example is that of magnetic resonance imaging, MRI, due to several key aspects of this example. First of all, the way the scanners are made specifies the sampling space and put limitations on possible sampling schemes. When dealing with MRI we are provided with extra information about the input which makes it possible to go below the limit dictated by Theorem 2.1, namely that it is an image. It is also worth mentioning that an improvement in the field of MRI is beneficial in several different ways. Being able to reconstruct images of equivalent quality using less samples would benefit both financially and medically. Less samples mean that each patient needs to be in the scanner for a shorter time. This in turn means that the patients are exposed to less radiation, there is a smaller chance that the patients move and thereby distort the images and it is possible to scan more patients per day.

The sampling space specified for the MRI example is a Fourier frame. Since the signal that needs to be reconstructed is an image, it is known that wavelets would be a good choice for reconstruction space. For these reasons we choose to focus on the case of Fourier samples and wavelet reconstruction. Although MR images are in two or three dimensions we will mainly work on the one dimensional case. This can be considered as a step on the way to multidimensional signals.

Chapter 3

General Theory

In this chapter we present the theoretical foundation for generalized sampling. The chapter will cover Hilbert spaces, both specific Hilbert spaces relevant in future chapters, subspaces of Hilbert spaces and frames for Hilbert spaces.

3.1 Hilbert Spaces

Definition 3.1 (Hilbert Space, $\mathcal{H}$)
Let $w,z \in \mathbb{C}$ and $a,b,c \in \mathcal{H}$. A vector space with inner product, $\langle \cdot, \cdot \rangle$, and associated norm, $|| \cdot || = \sqrt{\langle \cdot, \cdot \rangle}$, is called a Hilbert space, denoted $\mathcal{H}$, if

a) $\langle b, a \rangle = \overline{\langle a, b \rangle}$,

b) $\langle za + wb, c \rangle = z\langle a, c \rangle + w\langle b, c \rangle$,

c) $||za|| = |z| \, ||a||$,

d) $\langle a, a \rangle > 0 \ \ \forall a \neq 0$ and $\langle a, a \rangle = 0$ for $a = 0$

and it is complete w.r.t. convergence in norm.
[Kreyszig, 1978, Definition 3.1-1]

A Hilbert space is separable if it contains a countable dense subspace. Examples of separable Hilbert spaces will now be given.

Definition 3.2 (ℓ^2-space)
Let f and g be complex valued sequences defined on $\mathbb{Z}$. Then the inner product space of square-summable complex-valued sequences with the inner product

$$\langle f, g \rangle = \sum_{n=-\infty}^{\infty} f(n) \cdot \overline{g(n)} < \infty, \tag{3.1}$$

and norm

$$||f|| = \left(\sum_{n=-\infty}^{\infty} |f(n)|^2 \right)^{\frac{1}{2}} < \infty, \tag{3.2}$$

where $n \in \mathbb{Z}$ is an index, is called $\ell^2(\mathbb{Z})$.
[Vetterli et al., 2014, p. 31]

Lemma 3.3
The space ℓ^2 is a separable Hilbert space.
[Kreyszig, 1978, pp. 23, 35]

Proof: We will first show that ℓ^2 is a Hilbert space and subsequently that it is separable. Showing (a)-(d) from Definition 3.1 is trivial. We will show that ℓ^2 is complete, i.e. that all Cauchy sequences converge.

Let $\{f_n\}_{n\in\mathbb{N}}$ be a Cauchy sequence in ℓ^2, where $f_n = \{f_n(1), f_n(2), ...\}$. Then, given $\epsilon > 0$, $\exists N \in \mathbb{N} : \forall m, n \geq N$ we have

$$\|f_n - f_m\| = \left(\sum_{j=1}^{\infty} |f_n(j) - f_m(j)|^2 \right)^{\frac{1}{2}} \leq \epsilon. \tag{3.3}$$

From this we get that, $\forall j \in \mathbb{N}$,

$$|f_n(j) - f_m(j)| \leq \epsilon.$$

So, $\{f_n(j)\}_{n\in\mathbb{N}}$ is a Cauchy sequence in $\mathbb{C}$, which is known to be complete. This means that $\exists f(j) \in \mathbb{C} : \{f_n(j)\} \overset{n\to\infty}{\longrightarrow} f(j)$. We collect all of these limits in a sequence $f = \{f(1), f(2), ...\}$. Let $M \in \mathbb{N}$, then from (3.3) we get

$$\left(\sum_{j=1}^{M} |f_n(j) - f_m(j)|^2 \right)^{\frac{1}{2}} \leq \epsilon,$$

which implies

$$\epsilon \geq \lim_{m\to\infty} \left(\sum_{j=1}^{M} |f_n(j) - f_m(j)|^2 \right)^{\frac{1}{2}} = \left(\sum_{j=1}^{M} |f_n(j) - f(j)|^2 \right)^{\frac{1}{2}}.$$

Let $M \to \infty \Rightarrow \|f_n - f\| \leq \epsilon, \forall n \geq N$, which shows that ℓ^2 is complete.

To show separability, we have to find a subspace, Y, which is countable and dense in ℓ^2. For every $N \in \mathbb{N}$, we set

$$\begin{aligned} Y_N = \{y = \{y(1), y(2), ..., y(N), 0...\} :& \\ & y(n) = a(n) + ib(n) : a(n), b(n) \in \mathbb{Q}, \forall n = 1, 2, \ldots, N\}. \end{aligned}$$

Observe that Y_N can be identified with $\mathbb{Q}^N \times \mathbb{Q}^N$ which is countable because it is a Cartesian product of countable sets. We consider

$$Y = \bigcup_{N=1}^{\infty} Y_N. \tag{3.4}$$

Y is countable since it is a countable union of countable sets.

Now we need to prove that Y is dense in ℓ^2. Let $f = \{f(n)\} \in \ell^2$ and, given $\epsilon > 0$, we will prove that $B(f,\epsilon) \cap Y \neq \varnothing$ ($B(f,\epsilon)$ is the open ball around f with radius ϵ), i.e. $\exists y \in Y : \|f - y\| < \epsilon$. Since $f \in \ell^2$ its norm is finite, i.e.

$$\sum_{n=1}^{\infty} |f(n)|^2 < \infty$$

and

$$\lim_{k\to\infty} \sum_{n=1}^{k} |f(n)|^2 = \sum_{n=1}^{\infty} |f(n)|^2 < \infty.$$

Additionally, $\epsilon > 0 \Rightarrow \frac{\epsilon^2}{2} > 0$. This leads to the following. $\exists N \in \mathbb{N} : \forall k \geq N$

$$\left| \sum_{n=1}^{k} |f(n)|^2 - \sum_{n=1}^{\infty} |f(n)|^2 \right| < \frac{\epsilon^2}{2}.$$

For $k = N$ we get

$$\sum_{n=N+1}^{\infty} |f(n)|^2 < \frac{\epsilon^2}{2}, \tag{3.5}$$

i.e. for large enough N the norm of the tail of the sequence becomes arbitrarily small.

Let $n = 1, 2, ..., N$, $f(n) \in \mathbb{C}$ and $y(n) \in \mathbb{Q} + i\mathbb{Q}$. $\mathbb{Q} + i\mathbb{Q}$ is dense in $\mathbb{C}$. Additionally, $\epsilon > 0 \Rightarrow \frac{\epsilon}{\sqrt{2N}} > 0$. So $\exists y(n) \in \mathbb{Q} + i\mathbb{Q} : |f(n) - y(n)| < \frac{\epsilon}{\sqrt{2N}} \Leftrightarrow |f(n) - y(n)|^2 < \frac{\epsilon^2}{2N}$. From this it follows that

$$\sum_{n=1}^{N} |f(n) - y(n)|^2 < N\frac{\epsilon^2}{2N} = \frac{\epsilon^2}{2}. \tag{3.6}$$

By (3.5) and (3.6) we have that $\exists y \coloneqq \{y(1), y(2), ..., y(N), 0, 0, ...\} \in Y$ such that

$$\begin{aligned} \|f - y\| = \left(\sum_{n=1}^{\infty} |f(n) - y(n)|^2 \right)^{\frac{1}{2}} &= \left(\sum_{n=1}^{N} |f(n) - y(n)|^2 + \sum_{n=N+1}^{\infty} |f(n)|^2 \right)^{\frac{1}{2}} \\ &< \left(\frac{\epsilon^2}{2} + \frac{\epsilon^2}{2} \right)^{\frac{1}{2}} = \epsilon. \end{aligned} \tag{3.7}$$

So for this choice of Y we have a countable dense subspace of ℓ^2, hence, ℓ^2 is separable. □

Definition 3.4 (L^2-space)
Let f and g be complex-valued functions. Then the inner product space of square Lebesque integrable complex-valued functions with the inner product

$$\langle f, g \rangle = \int_{-\infty}^{\infty} f(x) \cdot \overline{g(x)} dx < \infty, \tag{3.8}$$

and the norm

$$||f|| = \left(\int_{-\infty}^{\infty} |f(x)|^2 dx \right)^{\frac{1}{2}} < \infty, \tag{3.9}$$

is called $L^2(\mathbb{R})$, [Vetterli et al., 2014, p. 31]. An equivalent definition of $L^2(\mathbb{R})$ can be found in [Pedersen, 1999, Definition 2.5]. This states that $L^2(\mathbb{R})$ is the completion of $C_0(\mathbb{R})$, the space of continuous functions with compact support. When limiting to the function with compact support on a fixed interval I we get $L^2(I)$.

Remark 3.5: It is trivial to show that $L^2(\mathbb{R})$ is an inner product space and the fact that it is defined to be the completion of $C_0(\mathbb{R})$ implies that it is a Hilbert space. In [Pedersen, 1999, Theorem 2.8] it is shown that the space of polynomials, with coefficients in $\mathbb{Q}$, on the interval, $P(I)$, is dense in $L^2(I)$. Furthermore, this space of polynomials is countable and therefore $L^2(I)$ is separable. ◂

Note that if nothing else is stated the norm and inner product will always be L^2 or ℓ^2. It will be clear from the context which of the two it is.

3.2 Frames

In the project the use of frames will be important therefore some theory is necessary. This section is dedicated to some of this theory. We start by defining frames.

Definition 3.6 (Frame)
A sequence of elements, $\{f_k\}_{k \in \mathbb{N}}$, in a Hilbert space, $\mathcal{H}$, is called a frame for $\mathcal{H}$ if there exists constants $A, B > 0$ such that

$$A||f||^2 \leq \sum_{k=1}^{\infty} |\langle f, f_k \rangle|^2 \leq B||f||^2, \quad \forall f \in \mathcal{H}. \tag{3.10}$$

[Christensen, 2008, Definition 1.1.1]

The constants A and B are called framebounds. A special type of frames are called Riesz bases.

Definition 3.7 (Riesz Basis)
A family of the form $\{Ue_k\}_{k=1}^{\infty}$ is called a Riesz basis for $\mathcal{H}$, where $\{e_k\}_{k=1}^{\infty}$ is an orthonormal basis for $\mathcal{H}$ and $U : \mathcal{H} \to \mathcal{H}$ is a bounded bijective operator.
[Christensen, 2008, Definition 3.3.1]

Remark 3.8: It is easily seen that a Riesz basis is a frame: Any element $f \in \mathcal{H}$ can be written as:

$$\sum_{k=1}^{\infty} |\langle f, Ue_k \rangle|^2 = \sum_{k=1}^{\infty} |\langle U^* f, e_k \rangle|^2 = \|U^* f\|_2^2 \sim \|f\|_2^2. \tag{3.11}$$

This implies that

$$A\|f\|^2 \leq \sum_{k=1}^{\infty} |\langle f, Ue_k \rangle|^2 \leq B\|f\|^2, \quad \forall f \in \mathcal{H}. \tag{3.12}$$

◂

Remark 3.9: If both of the frame bounds A and B, in (3.12), are equal to 1 the sequence $\{Ue_k\}_{k=1}^{\infty}$ is orthonormal, [Christensen, 2008, p. 63]. ◂

Definition 3.10 (Frame Operator)
The frame operator is defined as

$$S : \mathcal{H} \to \mathcal{H}, \quad Sf = \sum_{k=1}^{\infty} \langle f, f_k \rangle f_k. \tag{3.13}$$

[Christensen, 2008, p. 100]

The frame operator has several important properties which will be useful in later sections.

Lemma 3.11
Let $\{f_k\}_{k=1}^{\infty}$ be a frame with frame operator S and frame bounds A and B. Then the following is true:

1. S is bounded, invertible, self-adjoint and positive.
2. $\{S^{-1} f_k\}_{k=1}^{\infty}$ is a frame with frame operator S^{-1} and frame bounds B^{-1} and A^{-1}.
3. If A and B are the optimal frame bounds for $\{f_k\}_{k=1}^{\infty}$ then the bounds B^{-1} and A^{-1} are optimal for $\{S^{-1} f_k\}_{k=1}^{\infty}$.

[Christensen, 2008, Lemma 5.1.6]

An operator, S, is said to be self-adjoint if $\langle Sf, g\rangle = \langle f, Sg\rangle, \forall f, g \in \mathcal{H}$, and it is said to be positive if $\langle Sf, f\rangle \geq 0, \forall f \in \mathcal{H}$.

The next theorem is one of the most important frame results and is called the frame decomposition. It states that if $\{f_k\}_{k=1}^{\infty}$ is a frame for $\mathcal{H}$, then every element in $\mathcal{H}$ has a representation as an infinite linear combination of the frame elements.

Theorem 3.12
Let $\{f_k\}_{k=1}^{\infty}$ be a frame with frame operator S. Then

$$f = \sum_{k=1}^{\infty} \langle f, S^{-1} f_k\rangle f_k, \quad \forall f \in \mathcal{H}, \tag{3.14}$$

and

$$f = \sum_{k=1}^{\infty} \langle f, f_k\rangle S^{-1} f_k, \quad \forall f \in \mathcal{H}. \tag{3.15}$$

Both series converge unconditionally for all $f \in \mathcal{H}$.
[Christensen, 2008, Theorem 5.1.7]

Proof: Let $f \in \mathcal{H}$. Using the properties of the frame operator in Lemma 3.11,

$$\begin{aligned} f = SS^{-1}f = \sum_{k=1}^{\infty} \langle S^{-1}f, f_k\rangle f_k = \sum_{k=1}^{\infty} \langle f, S^{-1}f_k\rangle f_k, \\ f = S^{-1}Sf = S^{-1} \sum_{k=1}^{\infty} \langle f, f_k\rangle f_k = \sum_{k=1}^{\infty} \langle f, f_k\rangle S^{-1} f_k. \end{aligned} \tag{3.16}$$

It follows from [Christensen, 2008, Corollary 3.1.5] that the sequences converge unconditionally since they are Bessel sequences. □

That a sequence converges unconditionally means that a sequence converges to the same limit no matter the summation order.

3.3 Subspace Angles

In this section we define subspace angles and the subspace condition which is important for the stability and optimality of generalized sampling.

Definition 3.13 (Subspace Angle)
Let U and V be closed subspaces of a Hilbert space, $\mathcal{H}$, and $P_V : \mathcal{H} \to V$ be the orthogonal projection onto V. The subspace angle, $\theta_{UV} \in [0, \frac{\pi}{2}]$, between U and V is given by

$$\cos(\theta_{UV}) = \inf_{\substack{u \in U \\ ||u||=1}} ||P_V u|| = \inf_{\substack{u \in U \\ ||u||=1}} \sup_{\substack{v \in V \\ ||v||=1}} \langle u, v\rangle. \tag{3.17}$$

> [Adcock et al., 2013, Definition 3.1]

For instance, in the case where U and V are one dimensional subspaces of $\mathbb{R}^2$, the subspace angle is the angle between the lines U and V.

> **Definition 3.14 (Subspace Condition)**
> Let U and V be closed subspaces of a Hilbert space, $\mathcal{H}$. Then U and V is said to satisfy the subspace condition if $U \cap V = \{0\}$ and $U + V$ is closed in $\mathcal{H}$, or equivalently, if $\cos(\theta_{UV^\perp}) > 0$.
> [Adcock et al., 2013, Definition 3.3 and Lemma 3.2]

The subspace condition is not very strict and for most cases the first version is easier to check. The angle between subspaces do play an important role later, so it is relevant to use that version in some cases.

Chapter 4

Theory of Generalized Sampling

Generalized sampling can be used for different purposes depending on the input signal, the sampling space and the reconstruction space. If the samples are fixed we might get a better representation of the signal by moving it to the reconstruction space. Generalized sampling can also be used for compression, i.e. we might be able to represent a signal equally well in the reconstruction space while using less coefficients.

In this chapter we will present the theory of generalized sampling in the one-dimensional case. Measures for optimality and stability will be defined and bounds for these measures will be discussed under different conditions.

We are mainly concerned with functions with compact support in a domain $D \subset \mathbb{R}$, so we define the subspace $H = \{f \in L^2(\mathbb{R}) : \operatorname{supp}(f) \subseteq D\} \subseteq L^2(\mathbb{R})$. A sampling scheme, $\Omega = \{\omega_1, ..., \omega_N\} \subseteq \mathbb{R}$, is a finite set of distinct points in which a given function is sampled. In the one dimensional case it is assumed that $\omega_1 < \omega_2 < \cdots < \omega_N$. Additionally, we need a finite reconstruction space, $\mathcal{W} \subseteq H$. The goal is to compute an approximation, $\tilde{f} \in \mathcal{W}$, to f only using the sampling data. We wish to find a mapping, $F : f \to \tilde{f}$, depending solely on the sampling data which is both quasi-optimal and numerically stable.

Definition 4.1 (The Quasi-optimality Constant)
Let $F : H \to \mathcal{W}, f \mapsto \tilde{f}$ be a mapping, where H is a subspace of a separable Hilbert space. The quasi-optimality constant, $\nu = \nu(F) > 0$, is the smallest number such that

$$\|f - \tilde{f}\| \leq \nu \|f - \mathcal{P}_{\mathcal{W}} f\|, \quad f \in H, \tag{4.1}$$

where $\mathcal{P}_{\mathcal{W}}$ is the orthogonal projection onto $\mathcal{W}$. If no such constant exist then $\nu = \infty$. F is quasi-optimal if $\nu(F)$ is small. If $\nu = 1$ then the mapping F is equivalent to the orthogonal projection which is optimal.
[Adcock et al., 2013, Definition 2.1]

In the case of MRI we are forced to sample in a Fourier frame, but we know that images are generally well represented by wavelets. Therefore we choose $\mathcal{W}$ to be a subspace of $L^2(\mathbb{R})$ spanned by wavelets. We define the Fourier transform as

$$\hat{f}(\omega) = \int_{\mathbb{R}} f(x) e^{-2\pi i \omega x} dx, \quad \omega \in \mathbb{R}, \quad f \in L^2(\mathbb{R}), \tag{4.2}$$

and denote the sampling data

$$\{\hat{f}(\omega) : \omega \in \Omega\}. \tag{4.3}$$

Definition 4.2 (Admissible Sampling Operator)
Let Ω be a sampling scheme, $S : H \to H$ a bounded linear operator and let $\mathcal{W}$ be a finite-dimensional subspace of H. S is said to be an admissible sampling operator for the pair $(\Omega, \mathcal{W})$ if it satisfies

1. for each $f \in H$, Sf depends only on the sampling data $\{\hat{f}(\omega) : \omega \in \Omega\}$,
2. S is self-adjoint with respect to $\langle \cdot, \cdot \rangle$ and satisfies

$$|\langle Sf, g \rangle| \leq \sqrt{\langle Sf, f \rangle \langle Sg, g \rangle}, \quad \forall f, g \in H, \tag{4.4}$$

3. there exists a positive constant $C_1 = C_1(\Omega, \mathcal{W})$ such that

$$\langle Sf, f \rangle \geq C_1 \|f\|^2, \quad \forall f \in \mathcal{W}. \tag{4.5}$$

[Adcock et al., 2014a, Definition 3.1]

C_1 is always assumed to be the largest constant satisfying (4.5). Given an admissible sampling operator, S, the constant $C_2 = C_2(\Omega)$ is defined to be the smallest constant satisfying

$$\langle Sf, f \rangle \leq C_2 \|f\|^2, \quad f \in H. \tag{4.6}$$

The sampling operator can be constructed based on the Fourier transform as:

$$Sf(x) = \sum_{n=1}^{N} \sqrt{\mu_n} \hat{f}(\omega_n) e^{2\pi i \omega_n \cdot x} \chi_D(x), \tag{4.7}$$

where $\mu_n > 0$ are particular weights and D is an interval, [Adcock et al., 2014a, Equation 3.6]. In general the weights are the measure of the Voronoi region for each sample. In the one dimensional case this reduces to

$$\mu_n = \frac{1}{2}(\omega_{n+1} - \omega_{n-1}), \quad n = 1, \ldots, N, \tag{4.8}$$

where we set $\omega_0 = \omega_N - 2K$ and $\omega_{N+1} = \omega_1 + 2K$, where K is the bandwidth of the sampling scheme.

For certain conditions on the sampling scheme, which will be given in Theorem 4.12, the sampling operator, S, is the frame operator for a weighted Fourier frame. In that case C_1 and C_2 are the frame bounds.

Definition 4.3 (Reconstruction)
Given a sampling scheme, Ω, a finite-dimensional subspace, $\mathcal{W}$, and an admissible sampling operator, S, we define the generalized sampling reconstruction as

$$\tilde{f} \in \mathcal{W}, \quad \langle S\tilde{f}, g\rangle = \langle Sf, g\rangle, \quad \forall g \in \mathcal{W}. \tag{4.9}$$

[Adcock et al., 2014a, Equation 3.7]

In our case, with S given by (4.7), the reconstruction is equivalent to the weighted least-squares problem:

$$\tilde{f} = \arg\min_{g \in \mathcal{W}} \sum_{n=1}^{N} \mu_n |\hat{f}(\omega_n) - \hat{g}(\omega_n)|^2, \tag{4.10}$$

[Adcock et al., 2014a, p. 6].

Definition 4.4 (Reconstruction Constant)
Let S be an admissible sampling operator with constants C_1 and C_2 given by (4.5) and (4.6) respectively. The ratio $C(\Omega, \mathcal{W}) = \sqrt{C_2/C_1}$ is referred to as the nonuniform generalized sampling reconstruction constant.
[Adcock et al., 2014a, Definition 3.2]

This constant is a measure of numerical stability and is closely related to relative condition numbers.

Theorem 4.5
Let Ω be a sampling scheme, $\mathcal{W}$ a finite-dimensional subspace of H and assume that S is an admissible sampling operator. Then the reconstruction, $F(f) = \tilde{f}$, exists and is unique for any $f \in H$ and we have the sharp bound

$$\|f - F(f)\| \le \tilde{C}\|f - \mathcal{P}_{\mathcal{W}} f\|, \quad f \in H, \tag{4.11}$$

where $\tilde{C} = \sec(\theta_{\mathcal{W}V^\perp})$ and $V = (S(\mathcal{W}))^\perp$. Moreover $\tilde{C} \le C(\Omega, \mathcal{W})$.
[Adcock et al., 2014a, Theorem 3.3]

Proof: First we prove that $\tilde{C} \leq C(\Omega, \mathcal{W})$. From Definition 3.13 we get

$$\begin{aligned} \frac{1}{\tilde{C}} &= \cos(\theta_{\mathcal{W}V^{\perp}}) \\ &= \inf_{\substack{g \in \mathcal{W} \\ \|g\|=1}} \|\mathcal{P}_{S(\mathcal{W})} g\| \\ &= \inf_{\substack{g \in \mathcal{W} \\ \|g\|=1}} \sup_{\substack{g' \in \mathcal{W} \\ Sg' \neq 0}} \frac{|\langle g, Sg' \rangle|}{\|Sg'\|}. \end{aligned} \tag{4.12}$$

Now set $g' = g$ and use that S is self-adjoint:

$$\frac{1}{\tilde{C}} \geq \inf_{\substack{g \in \mathcal{W} \\ \|g\|=1}} \frac{\langle Sg, g \rangle}{\|Sg\|}. \tag{4.13}$$

Recall that $\|Sg\| = \sup_{\substack{h \in H \\ \|h\|=1}} \langle Sg, h \rangle$, [Christensen, 2008, Lemma 2.3.4]. Hence,

$$\|Sg\| = \sup_{\substack{h \in H \\ \|h\|=1}} \langle Sg, h \rangle \leq \sqrt{C_2} \sqrt{\langle Sg, g \rangle}, \tag{4.14}$$

where the inequality comes from (4.4) and (4.6). Applying (4.5) gives

$$\frac{1}{\tilde{C}} \geq \sqrt{\frac{C_1}{C_2}} \quad \Leftrightarrow \quad \tilde{C} \leq \sqrt{\frac{C_2}{C_1}} = C(\Omega, \mathcal{W}). \tag{4.15}$$

Now we need to show that the inequality (4.11) holds. Since $\mathcal{W}$ and $V^{\perp}$ satisfy the subspace condition, [Adcock et al., 2013, Corollary 3.5] gives

$$\begin{aligned} \|\mathcal{Q}_{\mathcal{W}V} f\| &\leq \tilde{C} \|f\|, \quad \forall f \in H_0, \\ \|f - \mathcal{Q}_{\mathcal{W}V} f\| &\leq \tilde{C} \|f - \mathcal{P}_{\mathcal{W}} f\|, \quad \forall f \in H_0, \end{aligned} \tag{4.16}$$

where $H_0 = \mathcal{W} \oplus V$ and $\mathcal{Q}_{\mathcal{W}V} : H_0 \to \mathcal{W}$ is the projection with range $\mathcal{W}$ and kernel V.

It now remains to show the following: (i) $H_0 = H$ and (ii) $\tilde{f} = \mathcal{Q}_{\mathcal{W}V} f$, $\forall f \in H$.

(i) From [Adcock et al., 2013, Lemma 3.7] we know that $H_0 = H$ if $\dim(S(\mathcal{W})) = \dim(\mathcal{W})$. If the dimensions are not the same there exist a nonzero $g \in \mathcal{W}$ such that $Sg = 0$, but, because S is an admissible sampling operator, this is only true for $g = 0$: a contradiction.

(ii) We now know that $H = \mathcal{W} \oplus V$, this means there is a unique way of splitting $f \in H$, $f = f_{\mathcal{W}} + f_V$, so

$$\langle f, Sg \rangle = \langle f_{\mathcal{W}} + f_V, Sg \rangle = \langle f_{\mathcal{W}}, Sg \rangle \quad \forall g \in \mathcal{W}, \tag{4.17}$$

as f_V is orthogonal to Sg by definition of V. $\mathcal{Q}_{\mathcal{W}V}$ is the projection $H \to \mathcal{W}$, $f \mapsto f_{\mathcal{W}}$ which means that

$$\langle f_{\mathcal{W}}, Sg \rangle = \langle \mathcal{Q}_{\mathcal{W}V} f, Sg \rangle \quad \forall g \in \mathcal{W}. \tag{4.18}$$

Since S is self-adjoint, $\mathcal{Q}_{\mathcal{W}V}$ is a reconstruction by Definition 4.3. The reconstruction is unique due to the unique splitting of $f \in H$, $f = f_{\mathcal{W}} + f_V$. □

Remark 4.6: Note that $\tilde{C}$ is the quasi-optimality constant per Definition 4.1. From this and Theorem 4.5 we deduce that the reconstruction constant is an upper bound for the quasi-optimality constant. ◂

Remark 4.7: In Theorem 4.5 it is implicitly assumed that $\mathcal{W}$ and $V^{\perp}$ satisfy the subspace condition from Definition 3.14. Since S is an admissible sampling operator we have $C < \infty$ and, as we proved, $\tilde{C} \leq C$ which means that $\cos(\theta_{\mathcal{W}V^{\perp}}) > 0$. ◂

4.1 The Stable Sampling Rate

The subspace condition, Definition 3.14, is a condition under which we are guaranteed stable and quasi-optimal reconstruction. Given that the subspace condition is satisfied, it is of great importance to know how large the number of samples, M, must be for a given number of reconstruction coefficients, N. Or reversed, how large N is allowed to be for a given M. For instance, if we want to compress a signal to N coefficients in the reconstruction space, we need to know how many samples to take.

Definition 4.8 (The Stable Sampling Rate)
The stable sampling rate is defined as

$$\Theta(N;\theta) = \min\{M \in \mathbb{N} : C(\Omega, \mathcal{W}) \leq \theta\}, \quad N \in \mathbb{N},\ \theta \in (0, \infty)\,. \tag{4.19}$$

[Adcock et al., 2013, Definition 5.1]

By choosing $M \geq \Theta(N;\theta)$ the definition gives that $C(\Omega, \mathcal{W}) \leq \theta$ so the reconstruction is quasi-optimal and numerically stable up to the magnitude of θ. The condition $M \geq \Theta(N;\theta)$ is both sufficient and necessary to ensure stable quasi-optimal reconstruction.

Conversely, the quantity

$$\Psi(M;\theta) = \max\{N \in \mathbb{N} : C(\Omega, \mathcal{W}) \leq \theta\}, \quad M \in \mathbb{N},\ \theta \in (0, \infty) \tag{4.20}$$

is called the stable reconstruction rate and is used to choose N based on M, [Adcock et al., 2013, Equation (5.1)].

4.2 Sampling in a Fourier Frame

Suppose a sampling scheme, Ω, is chosen such that it gives rise to a weighted Fourier frame, i.e.

$$A\|f\|^2 \leq \sum_{n\in\mathbb{Z}} \mu_n |\hat{f}(\omega_n)|^2 \leq B\|f\|^2, \quad \forall f \in L^2([0,1]), \tag{4.21}$$

for constants $0 < A \leq B < \infty$. The frame operator is well-defined, linear, bounded, invertible and the truncated operators,

$$S_\Lambda : f \mapsto \sum_{n=-\Lambda}^{\Lambda} \sqrt{\mu_n} \hat{f}(\omega_n) e^{2\pi i \omega_n \bullet}, \tag{4.22}$$

converge strongly to S on $L^2([0,1])$ as $\Lambda \to \infty$, i.e. $\|S_\Lambda - S\| \to 0$ as $\Lambda \to \infty$.

Definition 4.9 (Relative Separation)
A sequence of points, $\zeta_k \in \mathbb{R}, k \in I \subseteq \mathbb{N}$, is called separated if $|\zeta_k - \zeta_j| \geq \eta, j \neq k$, for some $\eta > 0$. If $\{\lambda_k\}_{k \in I}$ is a finite union of separated sets, it is called a relatively separated sequence.
[Adcock et al., 2014a, p. 11]

An example of relative separation: Take two sequences in $\mathbb{R}$: $\{n\}_{n \in \mathbb{N}}$ and $\{n + \frac{1}{n}\}_{n \in \mathbb{N}}$. Both of these sequences are separated, but their union is not. However, it is relatively separated.

Definition 4.10 (The Lower Beurling Density)
For a sequence, $\{\omega_n\}_{n \in \mathbb{Z}}$, the lower Beurling density is defined as

$$D^- = \lim_{r \to \infty} \frac{n^-(r)}{r}, \quad n^-(r) = \min_{t \in \mathbb{R}} \operatorname{card}\big(\{n \in \mathbb{Z} : \omega_n \in (t, t+r)\}\big). \tag{4.23}$$

[Adcock et al., 2014a, p. 11]

Remark 4.11: In practise the sampling scheme will only cover a finite interval. When that is the case, the lower Beurling density is calculated as above, but r is limited by the length of the interval instead of infinity. ◂

The lower Beurling density is a measure of the density in the intervals with the least number of samples. A small number means that there are large intervals with few samples. This property can be problematic as stated in the following theorem.

Theorem 4.12
If $\{\omega_n\}_{n \in \mathbb{N}}$ is relatively separated and $D^- > 1$ then $\{\sqrt{\mu_n} e^{2\pi i \omega_n \bullet} \chi_{[0,1]}(\bullet)\}_{n \in \mathbb{Z}}$ forms a weighted frame for $L^2([0,1])$ with weights $\{\mu_n\}_{n \in \mathbb{Z}}$. Conversely, if $\{\sqrt{\mu_n} e^{2\pi i \omega_n \bullet} \chi_{[0,1]}(\bullet)\}_{n \in \mathbb{Z}}$ forms a weighted frame for $L^2([0,1])$ with weights $\{\mu_n\}_{n \in \mathbb{Z}}$, then $D^- \geq 1$ and $\{\omega_n\}_{n \in \mathbb{N}}$ is relatively separated.
[Adcock et al., 2014a, Theorem 4.7]

According to the theorem, if the samples are neither too clustered nor too scattered, we have a weighted Fourier frame for the sampling space. When this is the case, we have en upper bound for the reconstruction constant:

Theorem 4.13
Let $\mathcal{W}$ be a finite-dimensional subspace of $L^2([0,1])$. Suppose that $\Omega_\Lambda = \{\omega_n : |n| \leq \Lambda\}$, where $\{\omega_n : n \in \mathbb{Z}\}$, gives rise to a weighted Fourier frame with weights $\{\mu_n\}_{n\in\mathbb{Z}}$. Then the truncated frame operator,

$$S_\Lambda : f \mapsto \sum_{n=-\Lambda}^{\Lambda} \sqrt{\mu_n} \hat{f}(\omega_n) e^{2\pi i \omega_n \bullet}, \tag{4.24}$$

is admissible for all sufficiently large Λ. Specifically,

$$C(\Omega, \mathcal{W}) \leq \frac{\sqrt{B}}{\sqrt{A - \tilde{E}(\mathcal{W}, \Lambda)^2}}, \tag{4.25}$$

where A and B are the frame bounds for $\mathcal{W}$ and

$$\tilde{E}(\mathcal{W}, \Lambda)^2 = \sup \left\{ \sum_{|n|>\Lambda} \sqrt{\mu_n} |\hat{f}(\omega_n)|^2 : f \in \mathcal{W}, \|f\| = 1 \right\}. \tag{4.26}$$

[Adcock et al., 2014a, Theorem 4.8]

Proof: First we have to show that the sampling operator is admissible, Definition 4.2. Requirement (1) is satisfied because we know the sampling scheme, Ω, and the sampling data, $\hat{f}(\omega)$. μ can easily be calculated from Ω. Requirement (2) is met because

$$\begin{aligned}
\langle S_\Lambda f, g \rangle &= \int_0^1 \sum_{n=-\Lambda}^{\Lambda} \sqrt{\mu_n} \hat{f}(\omega_n) e^{2\pi i \omega_n x} \overline{g(x)} dx \\
&= \sum_{n=-\Lambda}^{\Lambda} \sqrt{\mu_n} \hat{f}(\omega_n) \overline{\int_0^1 g(x) e^{-2\pi i \omega_n x} dx} \\
&= \sum_{n=-\Lambda}^{\Lambda} \sqrt{\mu_n} \hat{f}(\omega_n) \overline{\hat{g}(\omega_n)} \\
&= \sum_{n=-\Lambda}^{\Lambda} \sqrt{\mu_n} \int_0^1 f(x) e^{-2\pi i \omega_n x} dx \overline{\hat{g}(\omega_n)} \\
&= \int_0^1 f(x) \sum_{n=-\Lambda}^{\Lambda} \sqrt{\mu_n} e^{-2\pi i \omega_n x} \overline{\hat{g}(\omega_n)} dx \\
&= \int_0^1 f(x) \overline{S_\Lambda g(x)} dx = \langle f, S_\Lambda g \rangle.
\end{aligned} \tag{4.27}$$

Using the Cauchy-Schwartz inequality,

$$\begin{aligned} |\langle S_\Lambda f, g\rangle| &= |\langle S_\Lambda^{\frac{1}{2}} S_\Lambda^{\frac{1}{2}} f, g\rangle| = |\langle S_\Lambda^{\frac{1}{2}} f, S_\Lambda^{\frac{1}{2}} g\rangle| \leq \|S_\Lambda^{\frac{1}{2}} f\| \|S_\Lambda^{\frac{1}{2}} g\| \\ &= \sqrt{\left|\left\langle S_\Lambda^{\frac{1}{2}} f, S_\Lambda^{\frac{1}{2}} f\right\rangle\right|} \sqrt{\left|\left\langle S_\Lambda^{\frac{1}{2}} g, S_\Lambda^{\frac{1}{2}} g\right\rangle\right|} = \sqrt{|\langle S_\Lambda f, f\rangle| |\langle S_\Lambda g, g\rangle|} \\ &= \sqrt{\langle S_\Lambda f, f\rangle \langle S_\Lambda g, g\rangle}, \end{aligned} \tag{4.28}$$

where we drop the absolute value in the end because S_Λ is a positive operator. Now we want to show Requirement (3): $\langle S_\Lambda f, f\rangle \geq C_1 \|f\|^2$. We have a frame with frame operator S, i.e.

$$\begin{aligned} \langle Sf, f\rangle &= \sum_{n \in \mathbb{Z}} \sqrt{\mu_n} |\hat{f}(\omega_n)|^2, \\ A\|f\|^2 &\leq \langle Sf, f\rangle \leq B\|f\|^2. \end{aligned} \tag{4.29}$$

Using this and the strong convergence of S_Λ, for Λ large enough, we get

$$\begin{aligned} \langle S_\Lambda f, f\rangle &= \langle Sf, f\rangle - \langle (S - S_\Lambda) f, f\rangle \\ &\geq A\|f\|^2 - \sum_{|n| > \Lambda} \sqrt{\mu_n} |\hat{f}(\omega_n)|^2 \\ &\geq (A - \tilde{E}(\mathcal{W}, \Lambda)^2)\|f\|^2, \end{aligned} \tag{4.30}$$

where

$$\tilde{E}(\mathcal{W}, \Lambda)^2 = \sup\left\{\sum_{|n| > \Lambda} \sqrt{\mu_n} |\hat{f}(\omega_n)|^2 : f \in \mathcal{W}, \|f\| = 1\right\}, \tag{4.31}$$

which implies $C_1 \geq A - \tilde{E}(\mathcal{W}, \Lambda)^2$. This concludes the proof that S_Λ is in fact an admissible sampling operator. The only thing left to show is the upper bound for the reconstruction constant. We have already found a lower bound for C_1 so we need to estimate a bound for C_2.

$$\langle S_\Lambda f, f\rangle \leq \langle Sf, f\rangle \leq B\|f\|^2 \Rightarrow C_2 = B. \tag{4.32}$$

By Definition 4.4 the bound for the reconstruction constant becomes

$$C(\Omega, \mathcal{W}) = \sqrt{\frac{C_2}{C_1}} \leq \frac{\sqrt{B}}{\sqrt{A - \tilde{E}(\mathcal{W}, \Lambda)^2}}. \tag{4.33}$$

□

4.2.1 Nonharmonic Discrete Fourier Transform

This section describes how to create samples in a Fourier frame based on a signal in the time domain. We remind the reader that the Fourier transform is defined as

$$\mathcal{F}[f](\omega) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i x \omega} dx, \quad f \in L^2(\mathbb{R}), \tag{4.34}$$

where $\omega \in \mathbb{R}$. We want to implement this in Python. In order to do that, we must approximate it using only a finite number of calculations performed on a finite set of data. We base the derivation on the procedure done in [Folland, 1992, pp. 250-251].

Assume that f vanishes outside $[0,1]$. When this is the case we can replace the integral over $(-\infty,\infty)$ with the interval $[0,1]$. This theory can easily be generalized to arbitrary finite intervals. Next, $\mathcal{F}[f](\omega)$ will be calculated in a finite sequence of points, ω_k, in a bounded interval, $[-\omega_{\max},\omega_{\max}]$. Then the sampling theorem, Theorem 2.1, indicates that $\mathcal{F}[f]$ can be completely determined from values at the points k/T, where $k \in \mathbb{Z}$. However, this will not fulfil the requirements for the later sections, therefore we oversample. The uniform case is described at the bottom of [Folland, 1992, p. 231]. We sample in the points

$$\omega_k = \frac{k\epsilon}{T} = k\epsilon, \quad k \in \mathbb{Z}, \tag{4.35}$$

where $\epsilon > 0$ is called the sampling density and is the distance between two consecutive sample frequencies. In general we will sample in points ω_k such that $(D^-)^{-1} = \epsilon$. We will only consider $\epsilon^{-1} \in \mathbb{N}$. Hence, we are calculating

$$\mathcal{F}[f](\omega_k) = \int_0^1 \sqrt{\mu_k} f(x) e^{-2\pi i x \omega_k} dx, \quad |\omega_k| \leq \omega_{\max}. \tag{4.36}$$

We now replace the integral with a left Riemann sum by dividing the interval $[0,1]$ into N equal subintervals with the endpoints $\frac{nT}{N} = \frac{n}{N}$, where $n = 0,1,\ldots,N$, and we get

$$\begin{aligned} \mathcal{F}[f](\omega_k) &\approx \sum_{n=0}^{N-1} \sqrt{\mu_k} f\left(\frac{n}{N}\right) e^{-2\pi i \omega_k \frac{n}{N}} \frac{1}{\sqrt{N}} \\ &= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} \sqrt{\mu_k} f\left(\frac{n}{N}\right) e^{-2\pi i \omega_k \frac{n}{N}} =: \hat{f}(\omega_k). \end{aligned} \tag{4.37}$$

Since $e^{-2\pi i k} = 1$, $\forall k \in \mathbb{Z}$, the sequence, $\{\hat{f}(\omega_k)\}_k$, is periodic with period $\epsilon^{-1}N$, i.e. $\hat{f}(\omega_{k+\epsilon^{-1}N}) = \hat{f}(\omega_k)$. We have thereby reached a nonharmonic discrete Fourier transform in (4.37). The inverse nonharmonic discrete Fourier transform is

$$f\left(\frac{n}{N}\right) = \frac{1}{\sqrt{N}} \sum_k \sqrt{\mu_k} \hat{f}(\omega_k) e^{2\pi i \omega_k \frac{n}{N}}. \tag{4.38}$$

We will use the shorthand DFT for (4.37) and IDFT for (4.38).

In Figure 4.1 we compare the nonharmonic discrete Fourier transform to the analytical Fourier transform. Here we use the function (7.5). It is our interpretation that the small difference is due to numerical precision.

An implementation of the (inverse) nonharmonic discrete Fourier transform can be found in the Python module called `DFT.py`. The functions `DFT()` and `IDFT()` have time complexities of $\mathcal{O}(MN)$.

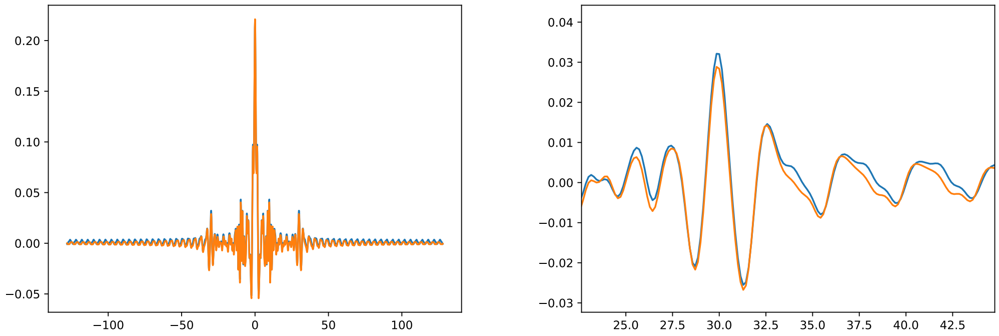


Figure 4.1: The plots show the nonharmonic discrete Fourier transform of a signal (blue) and the analytical Fourier transform of the signal (orange) sampled in the same points. The left is the whole signal and the right is a zoom.

Rank of Transforms

In this section we will investigate inaccuracies which might arise when using the DFT and IDFT transforms. A key factor is the rank of the transforms. Based on Theorem 4.12, it is assumed that $\eta < M$, where M is the number of Fourier coefficients and η is the number of time samples.

Since the transforms are linear, we can write them in matrix-form. The DFT has dimension $M \times \eta$ and the IDFT has dimension $\eta \times M$. By multiplying these matrices together, we get that the dimensions are $\eta \times \eta$ for IDFT·DFT and $M \times M$ for DFT·IDFT. Numerically we have found that IDFT·DFT= I_η, where I_η is the identity matrix of size $\eta \times \eta$. We believe that it can be proved analytically using [Folland, 1992, Lemma 7.1].

DFT·IDFT is a projection:

$$DFT \cdot IDFT \cdot DFT \cdot IDFT = DFT \cdot IDFT. \tag{4.39}$$

The IDFT is not injective, it has range of dimension η, $\eta < M$. Therefore, some error might occur depending on the signal. Signals in the range of IDFT can be transformed without error. Signals that does not lie in the range will have errors of varying degree. However, only signals in the kernel will be transformed to zero.

4.2.2 Fourier Frames

In this section the theory for Fourier frames will be investigated further. A weighted Fourier frame is defined as a system of $\{\sqrt{\mu_k} e^{2\pi i \omega_k x}\}_{k \in \mathbb{Z}}$ satisfying the frame condition (3.10). If $\{\omega_k\} = \mathbb{Z}$ we have an orthogonal basis for $L^2([0,1])$. However, we are more interested in the general frame case for $L^2([0,1])$, where $\omega_k \in \mathbb{R}$. The expansion

$$f(x) = \sum_{k \in \mathbb{Z}} \sqrt{\mu_k} \hat{f}(\omega_k) e^{2\pi i \omega_k x} \tag{4.40}$$

in $L^2(I)$, where $I \subset \mathbb{R}$ and $|I| < \infty$, is called a nonharmonic Fourier series. Notice that (4.40) is very similar to (4.38).

We will now introduce an important frame quantity called the frame radius. The radius states how large an interval a frame is valid for.

Definition 4.14 (Frame Radius)
For a given real sequence, $\Omega = \{\omega_k\}_{k\in\mathbb{Z}}$, the frame radius is defined as

$$R(\Omega) = \sup\{R > 0 | \{\sqrt{\mu_k} e^{2\pi i \omega_k x}\}_{k\in\mathbb{Z}} \text{ is a frame for } L^2(-R,R)\}. \tag{4.41}$$

[Christensen et al., 2016, p. 232]

It follows immediately from the definition that if $\{\sqrt{\mu_k} e^{2\pi i \omega_k x}\}_{k\in\mathbb{Z}}$ is a frame for $L^2(-R,R)$, for some $R > 0$, then it is also a frame for $L^2(-R',R')$, for all $0 < R' \leq R$. In the critical case, where $R = R(\Omega)$, there are cases where $\{\sqrt{\mu_k} e^{2\pi i \omega_k x}\}_{k\in\mathbb{Z}}$ is a frame and cases where it is not, [Christensen et al., 2016, p. 233].

Definition 4.15 (Uniform Density)
A seperated seqence, $\{\omega_k\}_{k\in\mathbb{Z}}$, is said to have uniform density, $\epsilon \in (0,\infty)$, if there exists a number, $L > 0$, such that

$$\left|\omega_k - \frac{\epsilon k}{2\pi}\right| \leq L, \quad \forall k \in \mathbb{Z}. \tag{4.42}$$

[Christensen et al., 2016, p. 233]

Theorem 4.16
If $\{\omega_k\}_{k\in\mathbb{Z}}$ is a seperated sequence with uniform density, $\epsilon > 0$, then $\{\sqrt{\mu_k} e^{2\pi i \omega_k x}\}_{k\in\mathbb{Z}}$ has a frame radius of at least $R \geq \frac{1}{2\epsilon}$.
[Christensen et al., 2016, Theorem 9.8.1]

A proof for Theorem 4.16 can be found in [Duffin and Schaeffer, 1952].

Chapter 5

The Reconstruction Space of Wavelets

Wavelets has been chosen to be the reconstruction basis. This is because wavelet systems are better suited to represent discontinuities than Fourier systems. Discontinuities are often very important in images and therefore it is important to choose a basis, in which they are well represented, for the reconstruction. The focus will be on a specific class of wavelets. This is due to its generally nice properties and the extended research already done on this class.

5.1 Notation

Wavelets are usually associated with a wavelet function, ψ, and a scaling function, ϕ, which are translated and dilated to span the desired space. These operators will now be defined. For completeness we will also define modulation.

Definition 5.1 (Translation, modulation, dilation)
Let $f(\bullet)$, $g(\bullet) \in L^2(\mathbb{R})$, let x, x_0, ω, $\omega_0 \in \mathbb{R}$ and $a \in \mathbb{R}_+$.
A translation is a time shift:

$$g(x) = f(x - x_0) \overset{FT}{\longleftrightarrow} \hat{g}(\omega) = e^{-2\pi i \omega x_0} \hat{f}(\omega). \tag{5.1}$$

A modulation is a frequency shift:

$$g(x) = e^{2\pi i \omega_0 x} f(x) \overset{FT}{\longleftrightarrow} \hat{g}(\omega) = \hat{f}(\omega - \omega_0). \tag{5.2}$$

A dilation is a scaling:

$$g(x) = \sqrt{a} f(ax) \overset{FT}{\longleftrightarrow} \hat{g}(\omega) = \frac{1}{\sqrt{a}} \hat{f}\left(\frac{\omega}{a}\right). \tag{5.3}$$

[Vetterli et al., 2014, p. 620]

From now on, let $\mathfrak{T}_k$ denote the operator which translates with k, let $\mathfrak{M}_l$ denote the operator which modulates with l and let $\mathfrak{D}_a$ denote the operator which dilates with a.

5.2 Daubechies Wavelets

A special class of wavelets is the Daubechies wavelets. These have several nice properties one being that they are dyadic. Dyadic wavelets are special because they are translated only by integers and scaled only by powers of 2. The wavelet functions are defined as

$$\psi_{J,k}(x) = \left(\mathfrak{D}_{2^J}\mathfrak{T}_k\psi\right)(x) = 2^{\frac{J}{2}}\psi\left(2^J x - k\right). \tag{5.4}$$

In the following, scaling and wavelet functions will often be seen with subscripts J and k, this refer to a dilation by 2^J and translation by k, where $J, k \in \mathbb{Z}$.

Daubechies wavelets are generated through a multi resolution analysis.

Definition 5.2 (Multi Resolution Analysis)
A multi resolution analysis for $L^2(\mathbb{R})$ consists of a sequence of closed subspaces, $\{V_J\}_{J\in\mathbb{Z}}$, of $L^2(\mathbb{R})$ and a function, $\phi \in V_0$, such that

1. $\ldots V_{-1} \subset V_0 \subset V_1 \ldots$,
2. $\overline{\bigcup_J V_J} = L^2(\mathbb{R})$ and $\bigcap_J V_J = \{0\}$,
3. $f \in V_J \Leftrightarrow \mathfrak{D}_2 f \in V_{J+1}$,
4. $f \in V_0 \Rightarrow \mathfrak{T}_k f \in V_0, \quad \forall k \in \mathbb{Z}$,
5. $\{\mathfrak{T}_k\phi\}_{k\in\mathbb{Z}}$ is an orthonormal basis for V_0.

[Christensen, 2008, Definition 3.6.2]

Note that ϕ, in Definition 5.2, is called the scaling function.

One particular simple Daubechies wavelet is the Haar wavelet, also known as the Daubechies 1 wavelet.

Definition 5.3 (Haar)
The Haar wavelet function is defined as

$$\psi(x) = \begin{cases} 1 & \text{if } 0 \le x < \frac{1}{2}, \\ -1 & \text{if } \frac{1}{2} \le x < 1, \\ 0 & \text{otherwise.} \end{cases} \tag{5.5}$$

Furthermore, the corresponding scaling function is $\phi = \chi_{[0,1)}$.
[Christensen, 2008, Example 3.6.1 and p. 76]

In 1910 Haar proved that the functions, $\{\psi_{J,k}\}$, constitutes an orthonormal basis for $L^2(\mathbb{R})$ for the choice of ψ as in (5.5). The Haar basis is especially well suited for representing piecewise constant functions; a class of functions which is generally ill represented in Fourier systems.

Another important property of Daubechies wavelets is the number of vanishing moments.

Definition 5.4 (Vanishing Moments)
The number of vanishing moments, a, for a function, $f \in L^2(\mathbb{R})$, is the highest value of a such that

$$\int_{-\infty}^{\infty} x^l f(x) dx = 0, \quad \forall \; l = 0, 1, ..., a, \tag{5.6}$$

given that

$$\int_{-\infty}^{\infty} |x^l||f(x)| dx < \infty, \quad \forall \; l = 0, 1, ..., a. \tag{5.7}$$

[Christensen, 2008, p. 93]

Remark 5.5: It is relevant to note that a Daubechies wavelet basis can be used to reconstruct any polynomial of degree $a - 1$ if the number of vanishing moments for the wavelet is a, [Cohen et al., 1993, p. 55]. ◂

We will denote the Daubechies wavelets "Dba" based on the number of vanishing moments, e.g. the Daubechies wavelet with two vanishing moments is called Db2.

5.2.1 The Cascade Algorithm

For Daubechies wavelets of higher order than the Haar wavelet, i.e. with more vanishing moments, a closed form expression for the scaling function does not exist. However, the scaling function satisfies the two-scale equation,

$$\phi(x) = \sqrt{2} \sum_{k=0}^{2a-1} h_k \phi(2x - k), \tag{5.8}$$

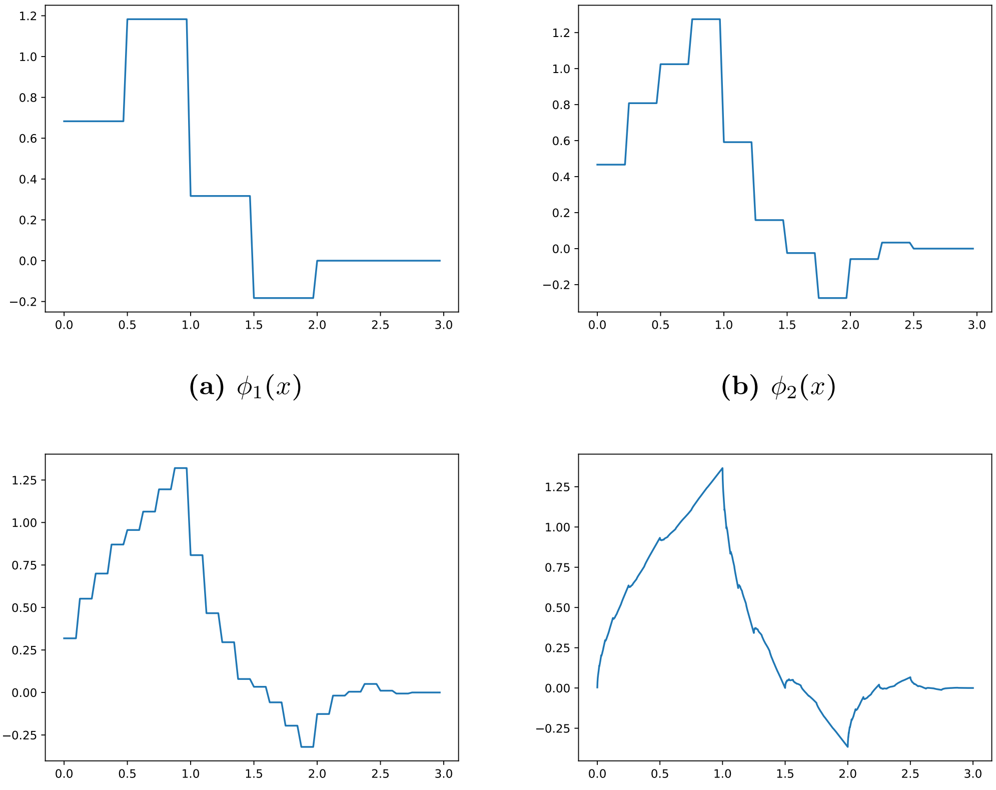
and we can use this to evaluate it in specific points. Here, h_k are filter coefficients which are known for specific choices of Daubechies wavelets, [Daubechies, 1992, Table 6.1]. It is possible to use (5.8) to write a system of equations for integer values of x and solve for the function values. When the function value is known for integer values of x it can be found for $2^{-1}x$ and so on. [Ruch and Van Fleet, 2011, Section 6.2] describes an algorithm to evaluate the scaling function in points $2^{-R}x$ for all integer values of x and all positive integer values of R. The cascade algorithm is iterative, i.e. we start with a guess ϕ_0 and for each iteration we get a better approximation

ϕ_n. The algorithm use the two-scale equation, (5.8), in the iteration as

$$\phi_{n+1}(x) = \sqrt{2}\sum_{k=0}^{2a-1} h_k \phi_n(2x-k). \tag{5.9}$$

A good choice for ϕ_0 is the Haar scaling function, $\chi_{[0,1)}$. In Figure 5.1, the Daubechies wavelet with two vanishing moments is plotted after 1, 2, 3 and 15 iterations.

Theorem 5.6
Suppose $\sum_{k=0}^{2a-1} h_k = \sqrt{2}$. Then the sequence of functions, $\{\phi_n(x)\}$ defined by (5.9), with $\phi_0(x) = \chi_{[0,1)}(x)$, converges to a function, $\phi(x)$, that satisfies the two-scale equation, (5.8). Moreover, $\overline{\mathrm{supp}(\phi_n)} = [0, 2a-1-2^{-n}(2a-2)]$ for $n \geq 1$ and $\overline{\mathrm{supp}(\phi)} = [0, 2a-1]$.
[Ruch and Van Fleet, 2011, Theorem 6.3]



(a) $\phi_1(x)$

(b) $\phi_2(x)$

(c) $\phi_3(x)$

(d) $\phi_{15}(x)$

Figure 5.1: The cascade algorithm used on the Db2 scaling function.

5.3 Daubechies Wavelets as Reconstruction Space

In this section the reconstruction space of wavelets will be investigated. The wavelets considered are limited to Daubechies wavelets on the interval $[0,1]$.

Boundary Wavelets. A higher order wavelet, i.e. with more vanishing moments, is supported on a larger interval than one of lower order. This is problematic on the boundary of $[0,1]$. Therefore, some special functions with smaller support is needed, these are called boundary wavelets. Boundary wavelets will be investigated further in Section 5.4.

Assume the scaling function, ϕ, and the wavelet function, ψ, are supported on the interval $[0,2a-1]$; these functions will be dilated with 2^J such that they are supported well within the interval $[0,1]$. Denote the boundary wavelets at the endpoints 0 and 1 by

$$\begin{aligned} &\phi^L_{J,k}, \phi^R_{J,k}, \quad k = 0,1,\dots,a-1, \\ &\psi^L_{J,k}, \psi^R_{J,k}, \quad k = 0,1,\dots,a-1, \end{aligned} \tag{5.10}$$

where L is the left endpoint, 0, and R is the right endpoint, 1. Denote the boundary corrected scaling and wavelet functions on the interval $[0,1]$ by

$$\begin{aligned} \phi^{[0,1]}_{J,k}(x) &= \begin{cases} \phi^L_{J,k}(x), & 0 \le k < a, \\ 2^{J/2}\phi(2^J x - k), & a \le k < 2^J - a, \\ \phi^R_{J,2^J-k-1}(x), & 2^J - a \le k < 2^J, \end{cases} \\ \psi^{[0,1]}_{J,k}(x) &= \begin{cases} \psi^L_{J,k}(x), & 0 \le k < a, \\ 2^{J/2}\psi(2^J x - k), & a \le k < 2^J - a, \\ \psi^R_{J,2^J-k-1}(x), & 2^J - a \le k < 2^J. \end{cases} \end{aligned} \tag{5.11}$$

In the case of $a = 1$ there is no need for boundary correction. However, if one went through the trouble of creating them, they would be identical to the interior functions.

Now the reconstruction space for one dimension can be defined. Let $\mathbb{N} \ni J \ge \log_2(2a)$. The set

$$\left\{\phi^{[0,1]}_{J,k} : k = 0,1,\dots,2^J-1\right\} \cup \left(\bigcup_{j \ge J} \left\{\psi^{[0,1]}_{j,k} : k = 0,1,\dots,2^j - 1\right\}\right) \tag{5.12}$$

forms an orthonormal basis for $L^2([0,1])$, [Gataric and Poon, 2016, p. A1081]. This basis consists of infinitely many elements, so in practise we will use a finite number. Define

$$\begin{aligned} W^{[0,1]}_J &= \operatorname{span}\left\{\psi^{[0,1]}_{J,k} : k = 0,1,\dots,2^J-1\right\}, \\ V^{[0,1]}_J &= \operatorname{span}\left\{\phi^{[0,1]}_{J,k} : k = 0,1,\dots,2^J-1\right\}, \end{aligned} \tag{5.13}$$

thus, $L^2([0,1]) = V_J^{[0,1]} \oplus \left(\oplus_{j\geq J} W_j^{[0,1]}\right)$. Order the elements in (5.12) in increasing order of scale and denote the space spanned by the first N wavelets by $\mathcal{W}_N$. Then for fixed $r > J \geq \log_2(2a)$ and $N = 2^r$ the reconstruction space is

$$\mathcal{W}_N = V_J^{[0,1]} \oplus W_J^{[0,1]} \oplus \cdots \oplus W_{r-1}^{[0,1]} \tag{5.14}$$

and has dimension N. A special property of Daubechies wavelets is that $V_J^{[0,1]} \oplus W_J^{[0,1]} = V_{J+1}^{[0,1]}$. This can be seen by counting the number of functions in each set and recalling that all the $\psi_{J,k}^{[0,1]}$ and $\phi_{J,k}^{[0,1]}$ functions are orthogonal. Therefore, we have that $\mathcal{W}_N = V_r^{[0,1]}$.

A function, $f(x) \in \mathcal{W}_N$, can then be written as

$$f(x) = \sum_{k=0}^{2^J-1} c_{J,k}\phi_{J,k}^{[0,1]}(x) + \sum_{j=J}^{r-1}\sum_{k=0}^{2^j-1} d_{j,k}\psi_{j,k}^{[0,1]}(x), \tag{5.15}$$

but also

$$f(x) = \sum_{k=0}^{2^r-1} c_{r,k}\phi_{r,k}^{[0,1]}(x), \tag{5.16}$$

for some scaling coefficients, $c_{J,k}$, and some detail coefficients, $d_{J,k}$. We will only be interested in (5.16) since it is equivalent to (2.9). The function in Python, which reconstructs from the wavelet coefficients, can be found in the module called `ReconFunctions.py` and is called `Recon()`. The complexity of `Recon()` is $\mathcal{O}(N)$.

5.4 Boundary Wavelets

In this section the construction of dyadic boundary wavelets will be explored. Dyadic boundary wavelets was first described in [Cohen et al., 1993], we will however use the approach described in [Jawerth and Sweldens, 1994, Section 11.3].

All polynomials of degree less than or equal to $a-1$ can be written as a linear combination of $\{\phi_{J,k}\}_{k\in\mathbb{Z}}$, but when restricted to a closed interval, say $[0,1]$, this is no longer the case. To generate all polynomials up to degree $a-1$ we need to add boundary functions, equal to the number of vanishing moments, at each boundary.

It is desirable to have 2^J scaling functions when working with $[0,1]$. If we use a wavelet with a vanishing moments there are 2^J-2a+2 interior functions for sufficiently large J. This leaves room for $a-1$ extra functions at each boundary which gives a system that can generate polynomials up to degree $a-2$. If we want degree $a-1$, like we have for the corresponding system on $(-\infty,\infty)$, we have to omit the two outermost interior functions to make room for extra boundary functions, [Cohen et al., 1993, p. 70].

As $\{\phi_{J,k}\}_{k\in\mathbb{Z}}$ is an orthonormal Daubechies wavelet basis for V_J, all monomials, x^α, $\alpha \leq a-1$, can be written as $x^\alpha = \sum_k \langle x^\alpha, \phi_{J,k}\rangle \phi_{J,k}(x)$. When restricted to $[0,1]$, we get

$$x^\alpha|_{[0,1]} = \left(\sum_{k=-2a+2}^{0} + \sum_{k=1}^{2^J-2a} + \sum_{2^J-2a+1}^{2^J-1} \right) \langle x^\alpha, \phi_{J,k} \rangle \phi_{J,k}(x)|_{[0,1]}. \tag{5.17}$$

Define

$$\begin{aligned} \phi^L_{J,\alpha} &= \sum_{k=-2a+2}^{0} \langle x^\alpha, \phi_{J,k} \rangle \phi_{J,k}(x)|_{[0,1]}, \\ \phi^R_{J,\alpha} &= \sum_{k=2^J-2a+1}^{2^J-1} \langle x^\alpha, \phi_{J,k} \rangle \phi_{J,k}(x)|_{[0,1]}, \end{aligned} \tag{5.18}$$

then

$$x^\alpha|_{[0,1]} = \phi^L_{J,\alpha} + \sum_{k=1}^{2^J-2a} \langle x^\alpha, \phi_{J,k} \rangle \phi_{J,k}(x)|_{[0,1]} + \phi^R_{J,\alpha} \tag{5.19}$$

and $\{\phi^L_{J,\alpha}\}_{\alpha \leq a-1} \cup \{\phi^R_{J,\alpha}\}_{\alpha \leq a-1} \cup \{\phi_{J,k}|_{[0,1]}\}_{k=1}^{2^J-2a}$ forms a basis for $L^2([0,1])$.

Taking a closer look at $\langle x^\alpha, \phi_{J,k} \rangle$:

$$\langle x^\alpha, \phi_{J,k} \rangle = \int x^\alpha 2^{J/2} \phi(2^J x - k) dx, \tag{5.20}$$

making a change of variable: $u = 2^J x - k \Rightarrow x = 2^{-J}(u+k)$ and $dx = 2^{-J} du$,

$$\begin{aligned} &= 2^{-J} 2^{\frac{J}{2}} \int (2^{-J}(u+k))^\alpha \phi(u) du \\ &= 2^{-J+\frac{J}{2}-J\alpha} \int (u+k)^\alpha \phi(u) du \\ &= 2^{-J+\frac{J}{2}-J\alpha} \int \sum_{l=0}^{\alpha} \binom{\alpha}{l} u^l k^{\alpha-l} \phi(u) du \\ &= 2^{-J+\frac{J}{2}-J\alpha} \sum_{l=0}^{\alpha} \binom{\alpha}{l} k^{\alpha-l} \int u^l \phi(u) du. \end{aligned} \tag{5.21}$$

The quantity $\langle x^l, \phi \rangle$ is known as the moments of the scaling function. In [Kessler et al., 2003, Section 5] a recursion relation for the moments is derived: For $l = 0$ we have $\langle x^0, \phi \rangle = \int \phi(x) dx = 1$. For larger l we have

$$\begin{aligned} \langle x^l, \phi \rangle &= \int x^l \phi(x) dx \\ &= \int \frac{1}{\sqrt{2}} \left(\frac{x}{2}\right)^l \frac{1}{\sqrt{2}} \phi\left(\frac{x}{2}\right) dx \\ &= \frac{1}{2^l \sqrt{2}} \int x^l \sum_k h_k \phi(x-k) dx, \end{aligned} \tag{5.22}$$

were we use the convention for the filter coefficients that $\sum_k h_k = \sqrt{2}$. Now make a change of variable, $v = x - k \Rightarrow x = v + k$,

$$\begin{aligned}
&= \frac{1}{2^l\sqrt{2}}\sum_k h_k \int (v+k)^l \phi(v)dv \\
&= \frac{1}{2^l\sqrt{2}}\sum_k h_k \sum_{m=0}^{l}\binom{l}{m} k^{l-m}\int v^m\phi(v)dv \\
&= \frac{1}{2^l\sqrt{2}}\sum_k h_k \sum_{m=0}^{l-1}\binom{l}{m} k^{l-m}\int v^m\phi(v)dv \\
&\qquad + \frac{1}{2^l\sqrt{2}}\sum_k h_k \int v^l\phi(v)dv.
\end{aligned} \tag{5.23}$$

From the above we obtain

$$\begin{aligned}
\int v^l\phi(v)dv - \frac{1}{2^l}\int v^l\phi(v)dv &= \frac{1}{2^l\sqrt{2}}\sum_k h_k \sum_{m=0}^{l-1}\binom{l}{m} k^{l-m}\int v^m\phi(v)dv \\
\int v^l\phi(v)dv &= \frac{1}{(2^l-1)\sqrt{2}}\sum_k h_k \sum_{m=0}^{l-1}\binom{l}{m} k^{l-m}\int v^m\phi(v)dv.
\end{aligned} \tag{5.24}$$

To obtain an orthogonal basis for V_J, the $\{\phi^L_{J,\alpha}, \phi^R_{J,\alpha}\}$ needs to be orthogonalized, they are already orthogonal to $\phi_{J,m}$ and linearly independent. One way to orthogonalize is the Gram-Schmidt orthogonalization procedure. Orthogonalization will introduce some numerical error and if the functions are far from co-linear it might be better not to orthogonalize. We will investigate this with a series of tests in Sections 5.4.1, 6.1.2 and 7.2.2.

5.4.1 Example with Daubechies 2

In this example we will derive boundary wavelets for the Daubechies wavelet with two vanishing moments. In this case we need two left and two right functions:

$$\begin{aligned}
\phi^L_{J,0} &= \sum_{k=-2}^{0}\langle x^0, \phi_{J,k}\rangle \phi_{J,k}(x)|_{[0,1]}, \\
\phi^L_{J,1} &= \sum_{k=-2}^{0}\langle x^1, \phi_{J,k}\rangle \phi_{J,k}(x)|_{[0,1]}, \\
\phi^R_{J,0} &= \sum_{k=2^J-3}^{2^J-1}\langle x^0, \phi_{J,k}\rangle \phi_{J,k}(x)|_{[0,1]}, \\
\phi^R_{J,1} &= \sum_{k=2^J-3}^{2^J-1}\langle x^1, \phi_{J,k}\rangle \phi_{J,k}(x)|_{[0,1]}.
\end{aligned} \tag{5.25}$$

The calculations are done in Python in the file `BoundaryWavelets.py`. The four functions before and after orthogonalization are shown in Figures 5.2 and 5.3. The boundary functions are far from co-linear before the orthogonalization, so further tests are needed to determine whether or not orthogonalization is a good choice.

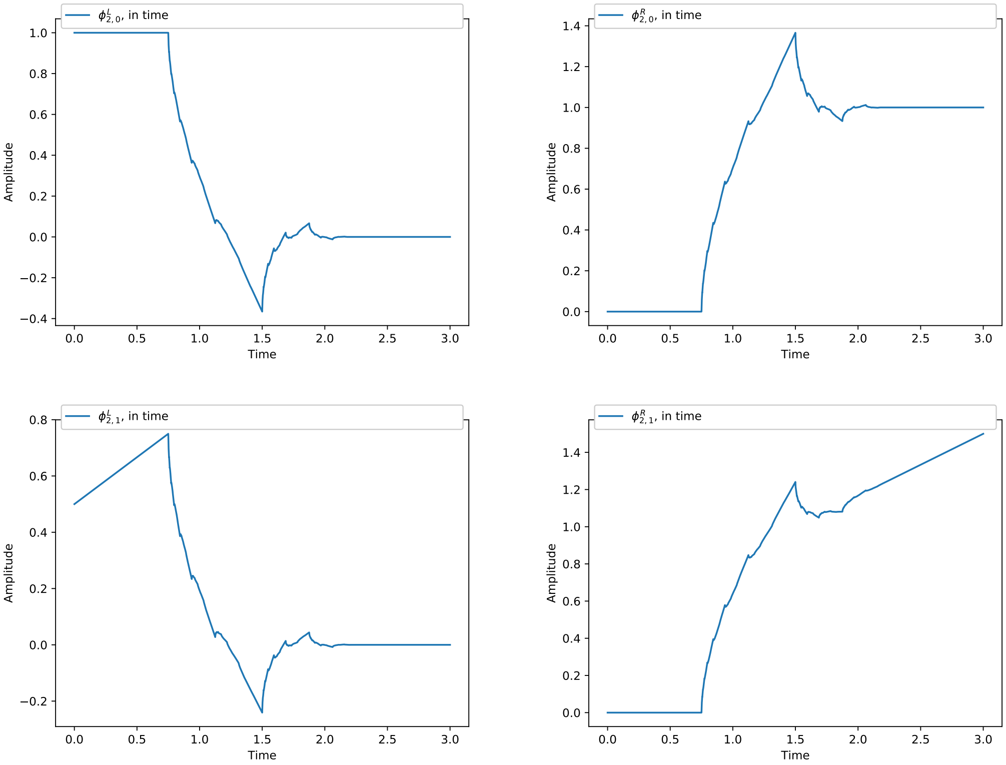


Figure 5.2: Daubechies 2 boundary wavelets, the first column shows the left boundary wavelets and the second column shows the right boundary wavelets.

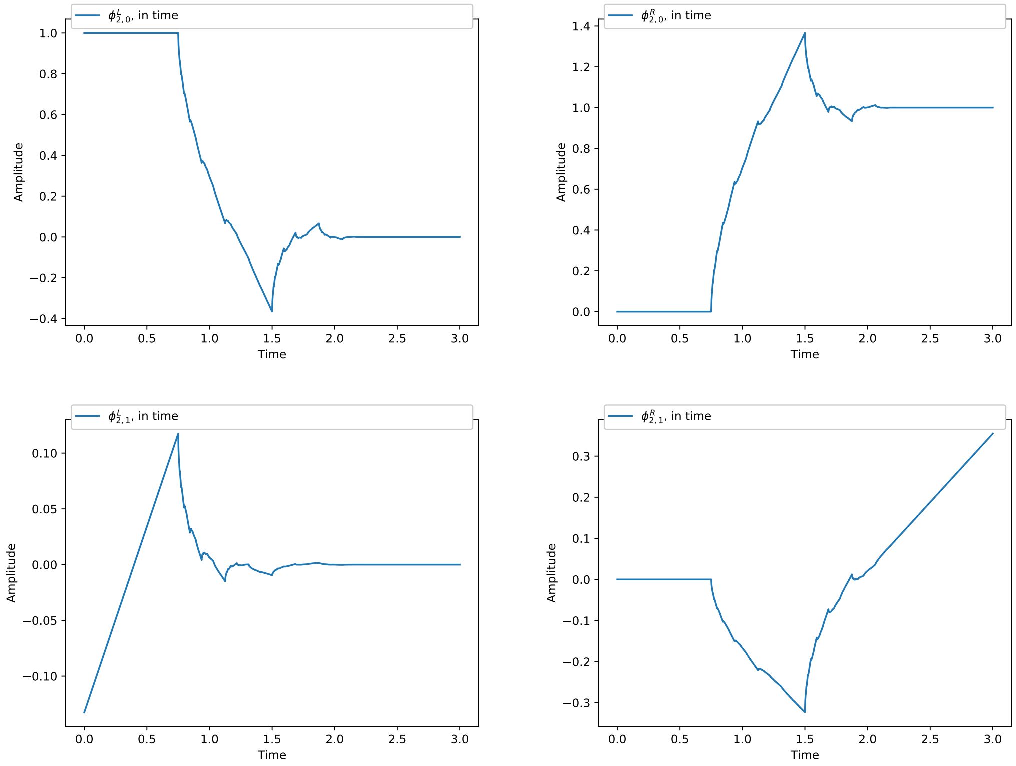


Figure 5.3: Orthogonal Daubechies 2 boundary wavelets, the first column shows the left boundary wavelets and the second column shows the right boundary wavelets.

To check that we are able to reconstruct polynomials up to degree 1, we choose two functions: a constant function and a first degree polynomial. We try to find coefficients which gives perfect reconstruction. For both cases we use a scale of three, i.e. $J = 3$, and use the non-orthogonal boundary functions.

The functions chosen are $f = 1$ and $g = x + 0.366$. All the coefficients have been found and is written in Table 5.1. The reconstructions using these two sets of coefficients are shown in Figure 5.4. It visually evident that the reconstructions are very good. Furthermore, the distance between the true signals and the reconstructions are $||f(x) - \tilde{f}(x)|| = 3.4 \cdot 10^{-13}$ and $||g(x) - \tilde{g}(x)|| = 4.293 \cdot 10^{-11}$.

The Python functions, which calculates the coefficients in Table 5.1, can be found in the module `ReconFunctions.py` in the test section, and are called `TestOfConFunc()` and `TestOfLinFunc()`.

Table 5.1: Table of coefficients for the functions f and g.

	$f(x) = 1$	$g(x) = x + 0.366$
α_0	1	-1.634
α_1	0	8
α_2	1	2
α_3	1	3
α_4	1	4
α_5	1	5
α_6	1	-1.634
α_7	0	8

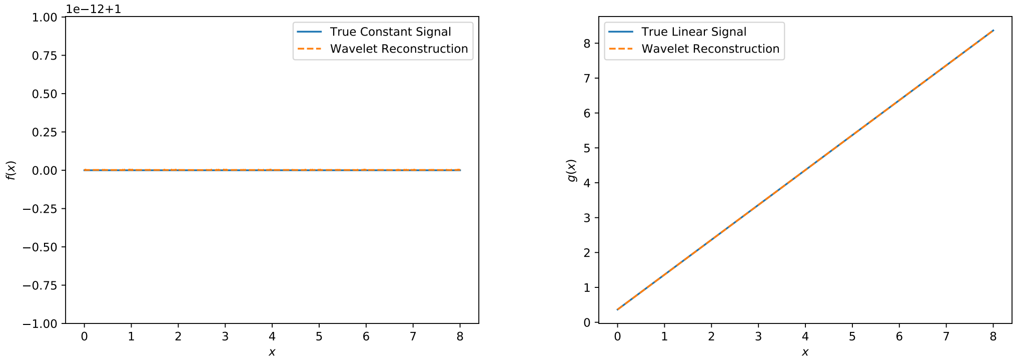


Figure 5.4: The left plot shows the constant function and the right shows the first degree polynomial. Both plots shows the true signal (blue) and the wavelet reconstruction (orange).

5.5 Sampling Densities and Sampling Rates

In this section the sampling density for given wavelets are found. Start by setting the sampling density, ϵ, to be

$$\epsilon \leq \frac{1}{\tau_1 + \tau_2}, \tag{5.26}$$

where $\tau_1 \geq 2a - 1 - 1$ and $\tau_2 \geq 2(2a-1) - 1$ in accordance with [Adcock et al., 2014b, p. 393]. Note that $\frac{1}{\tau_1+\tau_2}$ is the Nyquist criteria for functions supported on $[-\tau_1, \tau_2]$. Then $L^2([0, 2a-1]) \subset \mathcal{W} \subset L^2([-\tau_1, \tau_2])$.

Remark 5.7: For all $\epsilon \leq \frac{1}{\tau_1+\tau_2}$ we have $\mathcal{W} \subset V$. So $\mathcal{W} + V^{\perp}$ is a closed subspace of $\mathcal{H}$ and $\mathcal{W} \cap V^{\perp} = \{0\}$. Thus the subspace condition, Definition 3.14, of generalised sampling is satisfied, [Adcock et al., 2014b, Remark 2.7].

This means that if we choose ϵ according to (5.26) we are guaranteed that the reconstruction, in the wavelet basis, will be stable and quasi optimal. Hence, if we choose ϵ larger than this, we cannot be sure it will produce a stable and quasi optimal reconstruction.

Recall that if $\epsilon > 1$ we no longer have a weighted Fourier frame and the method will fail. ◂

In Definition 4.8 the stable sampling rate was defined. Now that wavelets are known objects some theorems will be presented ending with Theorem 5.10, which is the most important result in practice.

Theorem 5.8
Let $\mathcal{W}$ be the wavelet reconstruction space, as defined in Section 5.3, and let $\mathcal{S}$ be the Fourier sampling space, as defined in Section 4.2. Define $N_R := 2^R\lceil 2a - 1\rceil + (R+1)(\lceil 2a-1\rceil - 1)$, where $R \in \mathbb{N}$. Then $\forall\theta \in (1,\infty)$, there exists $\mathcal{S}_\theta \in \mathbb{N}$, independent of R, such that for $M = \left\lceil \frac{\mathcal{S}_\theta 2^{R+1}}{\epsilon} \right\rceil$ we have $\frac{1}{C(\Omega,\mathcal{W})} \geq \frac{1}{\theta}$. In particular, $\Theta(N;\theta) \leq \left\lceil \frac{2\mathcal{S}_\theta N}{\epsilon\lceil 2a-1\rceil} \right\rceil$. Hence, $\Theta(N;\theta) = \mathcal{O}(N)$ for any $\theta \in (1,\infty)$.
[Adcock et al., 2014b, p. Theorem 4.1]

The proof is very technical and can be found in [Adcock et al., 2014b, Section 5]. The theorem states that the stable sampling rate is linear for wavelet reconstructions from Fourier samples when the wavelets are generated through a multi resolution analysis.

We now introduce the stable sampling ratio, which has meaning since the stable sampling rate is linear for wavelets.

Definition 5.9 (Stable Sampling Ratio)
The stable sampling ratio is defined to be

$$\eta(\theta) := \limsup_{N\to\infty} \frac{\Theta(N;\theta)}{N}, \quad \theta \in (1,\infty). \tag{5.27}$$

[Adcock et al., 2014b, p. 395]

The stable sampling ratio states how many samples needs to be taken as $N \to \infty$, while $\Theta(N;\theta)$ states how many samples at least are needed for a specific choice of N.

Theorem 5.10
Let $\mathcal{W}$ be the wavelet reconstruction space generated by a Daubechies wavelet and let $\mathcal{S}$ be the Fourier sampling space. Define $N_R := 2^R\lceil 2a-1\rceil + (R+1)(\lceil 2a-1\rceil -1)$, where $R \in \mathbb{N}$ and a is the number of vanishing moments. Then there exists

a $\theta \in (1, \infty)$ and $R_0 \in \mathbb{N}$ such that for all $R \geq R_0$, the stable sampling rate is $\Theta(N_R; \theta) = \lceil 2^R / \epsilon \rceil$.

In particular, when $\epsilon^{-1} \in \mathbb{N}$ it suffices to let $\theta > (\inf_{\omega \in [-1/2, 1/2]} |\hat{\phi}(\omega)|)^{-1}$. Moreover, for the special case of the Haar wavelet we have that $\Theta(N_R; \theta) \leq \lceil 2^R / \epsilon \rceil$ for all $R \in \mathbb{N}$.
[Adcock et al., 2014b, Theorem 4.5]

The proof of Theorem 5.10 is very technical and has been omitted. It can be found in [Adcock et al., 2014b, Section 7].

Theorem 5.10 states that, in the case of Daubechies wavelets, we are able to determine the stable sampling rate exactly. This will come in handy when working on numerical examples in Chapters 7 and 8. Table 5.2 contains the values for ϵ and θ which will be used in the examples. Table 5.3 shows the stable sampling rates.

Note that only values of ϵ such that $\epsilon^{-1} \in \mathbb{N}$ will be used.

Remark 5.11: For values of θ and R as in Theorem 5.10, if $N_{R-1} + 1 \leq N \leq N_R$, then $\Theta(N; \theta) \leq \lceil \frac{2^R}{\epsilon} \rceil$ which implies that

$$\frac{1}{\epsilon \lceil a \rceil} \leq \eta(\theta) \leq \lim_{R \to \infty} \frac{\lceil 2^R / \epsilon \rceil}{N_{R-1} + 1} = \frac{2}{\epsilon \lceil a \rceil}. \tag{5.28}$$

However, in [Adcock et al., 2014b, Section 8] the results suggest that the optimal ratio is $(\epsilon \lceil a \rceil)^{-1}$ and it is only achieved when $N = N_R$.
[Adcock et al., 2014b, Remark 4.6] ◂

Table 5.2: Upper bounds for ϵ based on (5.26) and numerical estimates for θ based on Theorem 5.10. The Python functions, to calculate the values, can be found in the module called `Utilities.py`.

Daubechies wavelet	ϵ	θ^{-1}	θ
1	1	$\frac{2}{\pi}$ (exact)	$\frac{\pi}{2}$ (exact)
2	$\frac{1}{7}$	0.6847	1.460
3	$\frac{1}{13}$	0.6980	1.432
4	$\frac{1}{19}$	0.7031	1.422
5	$\frac{1}{25}$	0.7053	1.417
6	$\frac{1}{31}$	0.7062	1.415
7	$\frac{1}{37}$	0.7067	1.414
8	$\frac{1}{43}$	0.7069	1.414
9	$\frac{1}{49}$	0.7070	1.414

Table 5.3: The stable sampling rate $\Theta\left(N_R;\theta\right)$ for the Daubechies 2 wavelet. The Python function, to calculate the values, can be found in the module called `Utilities.py`.

J	$\Theta\left(N_R;\theta\right)$	J	$\Theta\left(N_R;\theta\right)$	J	$\Theta\left(N_R;\theta\right)$
1	N/A	6	224	11	7 168
2	N/A	7	448	12	14 336
3	14	8	896	13	28 672
4	28	9	1 792	14	57 344
5	56	10	3 584	15	114 688

Chapter 6

The Change of Basis Matrix

In this chapter we describe how to construct the change of basis matrix for the case of sampling in the Fourier frame and reconstruction with Daubechies wavelets.

6.1 Derivation of the Change of Basis Matrix

Let ϕ denote the scaling function of a multi resolution analysis, Definition 5.2. The scaling function, on scale $J \in \mathbb{Z}_+$ with translation $k \in \mathbb{Z}$, is defined as

$$\phi_{J,k}(x) = 2^{J/2}\phi(2^J x - k). \tag{6.1}$$

Let $\mathcal{B}$ be a change of basis matrix and denote the entries by $b_{m,n}$. Then

$$b_{m,n} = \mathcal{F}\left[\phi^{[0,1]}_{J,n-1}\right](\omega_m), \quad \omega_m \in \Omega, \tag{6.2}$$

where Ω is the sampling scheme, [Jacobsen et al., 2016b, p. 5]. For a general scaling function the following is true:

$$\mathcal{F}[\phi_{J,k}](\omega) = \mathcal{F}[\mathfrak{D}_{2^J}\mathfrak{T}_k\phi](\omega) = 2^{-J/2}\exp(-2\pi i k 2^{-J}\omega)\mathcal{F}[\phi](2^{-J}\omega). \tag{6.3}$$

A general Daubechies scaling function, ϕ, is defined by filter coefficients, $\{h_k\}_{k\in\mathbb{Z}}$, where finitely many entries are non-zero. The associated low-pass filter, $m_0(\omega)$, is defined as

$$m_0(\omega) = \sum_{k=0}^{2a-1} h_k \exp(-2\pi i k\omega) \tag{6.4}$$

and the Fourier transform can be computed as

$$\mathcal{F}[\phi](\omega) = \prod_{j=1}^{\infty} m_0(2^{-j}\omega), \tag{6.5}$$

where $m_0(0) = 1$ to ensure convergence, [Cohen et al., 1993, p. 54].

6.1.1 The Haar Wavelet

The scaling function for a Haar wavelet is $\phi_{J,k}^{[0,1]} = \phi_{J,k}$, where ϕ is as in Definition 5.3. The Fourier transform of ϕ is given by (6.5), where the low-pass filter is $m_0(\omega) = \frac{1}{2} + \frac{1}{2}e^{-i2\pi\omega}$, [Cohen et al., 1993, p. 55]. Hence,

$$\mathcal{F}[\phi](\omega) = \prod_{j=1}^{\infty}\left(\frac{1}{2} + \frac{1}{2}e^{-2\pi i 2^{-j}\omega}\right). \tag{6.6}$$

To get a closed form expression we use Euler's formula, $\frac{\sin(x)}{x} = \prod_{j=1}^{\infty}\cos(2^{-j}x)$ and $\sum_{j=1}^{\infty} 2^{-j} = 1$:

$$\begin{aligned} m_0(\omega) =& \frac{1}{2}(1 + e^{-2\pi i\omega}) = e^{-\pi i\omega}\left(\frac{e^{\pi i\omega} + e^{-\pi i\omega}}{2}\right) = e^{-\pi i\omega}\cos(\pi\omega), \\ \prod_{j=1}^{\infty} m_0(2^{-j}\omega) =& \prod_{j=1}^{\infty} e^{-\pi i 2^{-j}\omega}\cos(\pi\omega 2^{-j}) \\ =& e^{-\pi\omega i}\frac{\sin(\pi\omega)}{\pi\omega} = e^{-\pi\omega i}\frac{e^{i\pi\omega} - e^{-i\pi\omega}}{2i\pi\omega} = \frac{1 - e^{-2\pi i\omega}}{2\pi i\omega}, \end{aligned} \tag{6.7}$$

implying

$$\mathcal{F}[\phi](\omega) = \begin{cases} \frac{1-\exp(-2\pi i\omega)}{2\pi i\omega}, & \omega \neq 0, \\ 1, & \omega = 0, \end{cases} \tag{6.8}$$

[Jacobsen et al., 2016b, p. 12].

6.1.2 Boundary Wavelets

All boundary functions can be written as a linear combination of $\phi_{J,k}|_{[0,1]}$ since the Fourier transform is linear. The Fourier transformed boundary wavelets are

$$\begin{aligned} \mathcal{F}\left[\phi_{J,\alpha}^{L}\right] &= \sum_{k=-2a+2}^{0}\langle x^{\alpha}, \phi_{J,k}\rangle\mathcal{F}\left[\phi_{J,k}(x)|_{[0,1]}\right], \\ \mathcal{F}\left[\phi_{J,\alpha}^{R}\right] &= \sum_{k=2^J-2a+1}^{2^J-1}\langle x^{\alpha}, \phi_{J,k}\rangle\mathcal{F}\left[\phi_{J,k}(x)|_{[0,1]}\right], \end{aligned} \tag{6.9}$$

where

$$\begin{aligned} \mathcal{F}\left[\phi_{J,k}\right](\omega) &= 2^{\frac{-J}{2}}\exp(2\pi ik2^{-J}\omega)\mathcal{F}\left[\phi\right](2^{-J}\omega) \\ &= 2^{\frac{-J}{2}}\exp(2\pi ik2^{-J}\omega)\prod_{l=1}^{\infty}m_0(2^{-l}2^{-J}\omega) \\ &= 2^{\frac{-J}{2}}\exp(2\pi ik2^{-J}\omega)\prod_{l=1}^{\infty}\sum_{\nu\in\mathbb{Z}}h_{\nu}\exp(-2\pi i\nu 2^{-l}2^{-J}\omega). \end{aligned} \tag{6.10}$$

Remark 6.1: The sampling scheme is designed to handle functions with support on the interval $[0, 1]$ in the time domain. When such a sampling scheme is used on $\mathcal{F}[\phi_{J,k}](\omega)$ together with the IDFT the function is zero-padded outside $[0, 1]$ in time.

In [Folland, 1992, p. 250] it is explained how the discrete Fourier transform can be used on a function, f, with support in $[0, \Omega]$ to find Fourier coefficients in the interval $[-C, C]$. We exploit this in the opposite direction, i.e. if we sample the analytically found Fourier transform of a function in the interval $[-C, C]$ then the IDFT will have support on $[0, \Omega]$. ◂

Due to Plancherel's theorem, [Christensen, 2008, eq. 2.14], which state that the Fourier transform is norm preserving, it does not matter if the boundary functions are orthogonalized before or after the transformation so we choose to do it after.

Example with Db2

In the frequency domain the boundary wavelets for Daubechies 2 are

$$\begin{aligned}
\phi^L_{J,0} &= \sum_{k=-2}^{0} \langle x^0, \phi_{J,k}\rangle \mathcal{F}\left[\phi_{J,k}\right](\omega),\\
\phi^L_{J,1} &= \sum_{k=-2}^{0} \langle x^1, \phi_{J,k}\rangle \mathcal{F}\left[\phi_{J,k}\right](\omega),\\
\phi^R_{J,0} &= \sum_{k=2^J-3}^{2^J-1} \langle x^0, \phi_{J,k}\rangle \mathcal{F}\left[\phi_{J,k}\right](\omega),\\
\phi^R_{J,1} &= \sum_{k=2^J-3}^{2^J-1} \langle x^1, \phi_{J,k}\rangle \mathcal{F}\left[\phi_{J,k}\right](\omega).
\end{aligned} \tag{6.11}$$

The Python file with the calculations is called `FourierBoundaryWavelets.py`. The boundary wavelets have been constructed in the frequency domain sampled uniformly in the interval $[-128, 128)$ with a density of $\frac{1}{7}$ and scale $J = 2$. When taking the IDFT, as described in Section 4.2.1, they can be compared to the boundary wavelets constructed in Section 5.4. This comparison can be seen in Figure 6.1.

Remark 6.2: Note that, as described in the end of Section 4.2.1, when starting with the IDFT, we introduce some error. However, experimentally it has been observed that the errors are not too severe. ◂

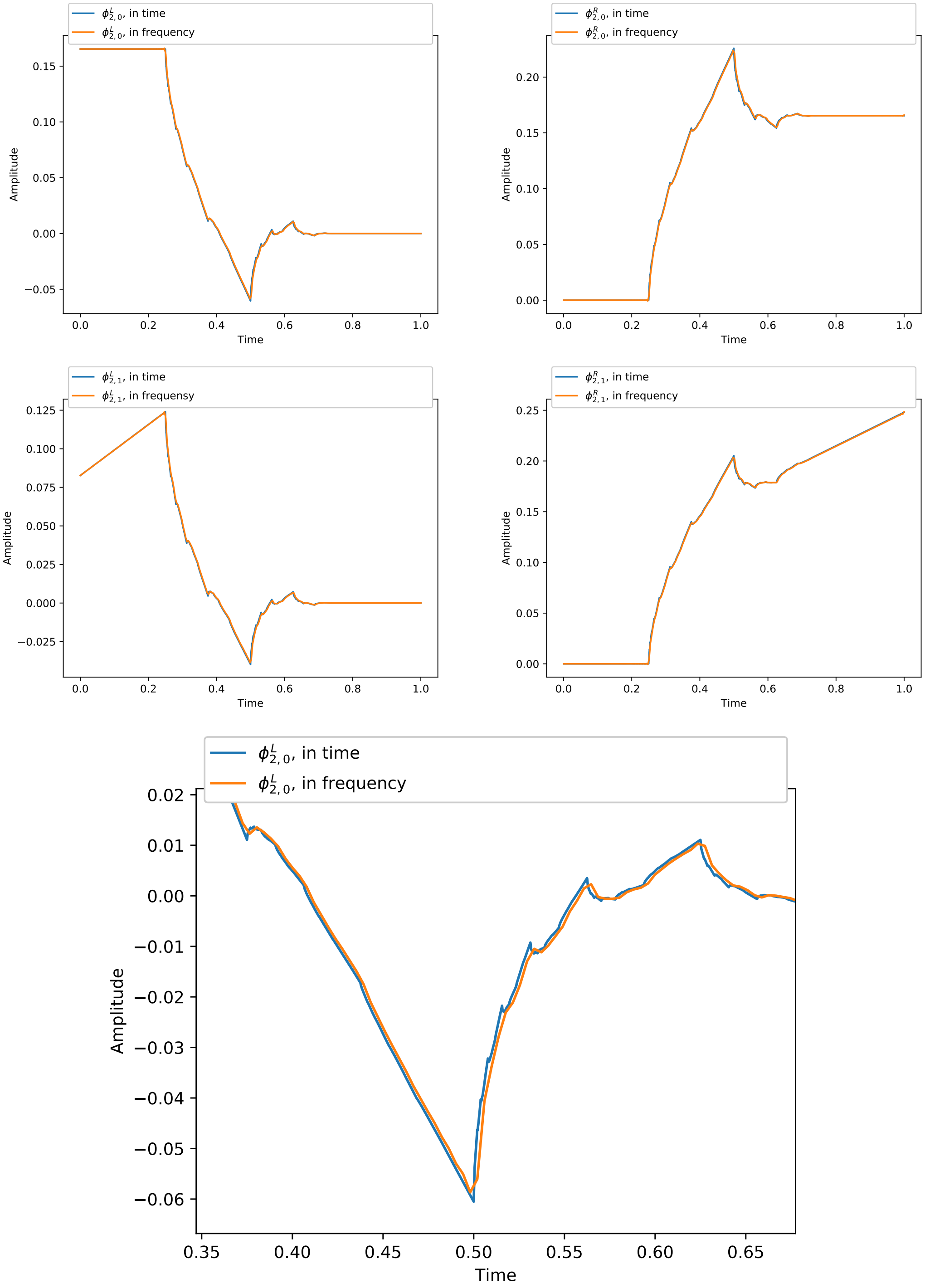


Figure 6.1: Comparison of boundary wavelets created in the time- (blue) and the frequency (orange) domain. The left plots are the left boundary wavelets and the right plots are the right boundary wavelets, the top is 0, the middle is 1 and the bottom is a zoom of left 0.

It is visually evident, in Figure 6.1, that the boundary wavelets from the frequency domain are very close to functions created in the time domain. The zoom shows that the functions are not identical, the boundary wavelets, which are created in the frequency domain, are more smooth. If the boundary wavelets are sampled in a wider interval in the frequency domain, it will become closer to the time boundary functions.

When orthogonalizing these two sets of functions some error occur. Part of the error have been introduced by the IDFT. The orthogonalized functions can be seen in Figure 6.2. The two sets of functions are clearly different when they are orthogonalized. We will, however, do a few more tests.

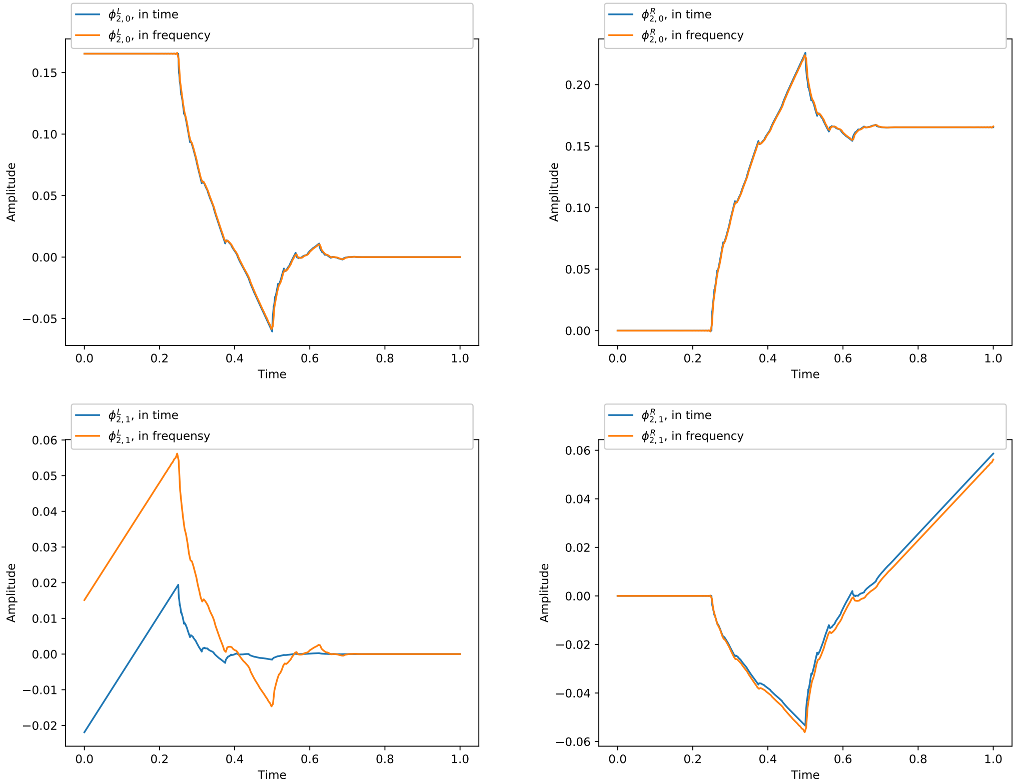


Figure 6.2: Comparison of orthogonalized boundary wavelets created in the time- (blue) and the frequency (orange) domain. The left plots are the left boundary wavelets and the right plots are the right boundary wavelets, the top is 0 and bottom is 1.

6.2 Fast Computations with the Change of Basis Matrix

In [Gataric and Poon, 2016] algorithms, for fast computations with the change of basis matrix and its adjoint, are derived for the case of Fourier sampling space and wavelet reconstruction space. These algorithms are valuable in practice as the change of basis matrix tends to be very large. In these algorithms it is important that N is a power of 2, [Gataric and Poon, 2016, beginning of Section 4].

The product of the change of basis matrix with a vector of scaling coefficients can be written as

$$\beta_m = \sqrt{\mu_{\omega_m}} \sum_{k=0}^{N-1} \alpha_k \langle \phi^{[0,1]}_{J,k}, e_{\omega_m} \rangle, \quad m = 1, ..., M, \tag{6.12}$$

where α_k are the scaling coefficients and $e_\omega(x) = e^{2\pi i \omega x}\chi_{[0,1]}(x)$. Furthermore,

$$\langle \phi^{[0,1]}_{J,k}, e_{\omega_m} \rangle = \widehat{\phi^{[0,1]}_{J,k}}(\omega) = \frac{1}{\sqrt{N}} \hat{\phi}\left(\frac{\omega}{N}\right) \overline{e_\omega\left(\frac{k}{N}\right)}, \quad k = 1, ..., N - 2a. \tag{6.13}$$

The translation variable $k = 1, ..., N-2a$ loops over all inner functions which are not included in the boundary functions. We want α_k to be indexed from 0 to $N-1$, therefore we write the translation, in the following, a bit different such that α_a is associated with the translation 1, or in general α_k is associated with translation $k - a + 1$.

We will split the calculation of β in three operators:

- $F : \mathbb{C}^N \to \mathbb{C}^M$:
$$F(\alpha) = \left(\frac{1}{\sqrt{N}} \sum_{k=a}^{N-a-1} \alpha_k \overline{e_{\omega_m}\left(\frac{k-a+1}{N}\right)} \right)_{m=1}^{M}. \tag{6.14}$$
- $D : \mathbb{C}^M \to \mathbb{C}^M$:
$$D(\tilde{\alpha}) = \left(\hat{\phi}\left(\frac{\omega_m}{N}\right) \tilde{\alpha}_m \right)_{m=1}^{M}. \tag{6.15}$$
- $V : \mathbb{C}^M \to \mathbb{C}^M$:
$$V(\gamma) = (\sqrt{\mu_{\omega_m}} \gamma_m)_{m=1}^{M}. \tag{6.16}$$

The contributions from the boundary functions are calculated as

$$\tilde{\beta}^L = \left(\sum_{k=0}^{a-1} \alpha_k \hat{\phi}^L_{J,k}(\omega_m) \right)_{m=1}^{M}, \quad \tilde{\beta}^R = \left(\sum_{k=N-a}^{N-1} \alpha_k \hat{\phi}^R_{J,N-k-1}(\omega_m) \right)_{m=1}^{M}. \tag{6.17}$$

This leads to Algorithm 1.

Algorithm 1: Fast computation of $\mathcal{B}\alpha$

```
Input: Scaling coefficients $\alpha \in \mathbb{C}^N$.
Output: Fourier coefficients $\beta \in \mathbb{C}^M$.
Compute the coefficients at the boundary: $\tilde{\beta}^L$ and $\tilde{\beta}^R$;
$\tilde{\alpha} \leftarrow F(\alpha)$;
$\tilde{\beta}^{int} \leftarrow D(\tilde{\alpha})$;
$\tilde{\beta} \leftarrow \tilde{\beta}^L + \tilde{\beta}^{int} + \tilde{\beta}^R$;
$\beta \leftarrow V(\tilde{\beta})$;
```

Remark 6.3: This algorithm only works for the case of $a \geq 2$. In the case of $a = 1$ there are no boundary functions so in step 1 we let $\tilde{\beta}^L = \tilde{\beta}^R = 0$ and the sum in $F(\alpha)$ go from 0 to $N - 1$. ◂

We also want an algorithm for fast computations with $\mathcal{B}^*$. The reason for this will be clear in Section 6.3. By transposing and conjugating (6.12) we get

$$\gamma_k = \sum_{m=1}^{M} \zeta_m \sqrt{\mu_{\omega_m}} \overline{\langle \phi_{J,k}^{[0,1]}, e_{\omega_m} \rangle}, \quad k = 0, ..., N-1. \tag{6.18}$$

The interior part can be rewritten as

$$\gamma_k = \frac{1}{\sqrt{N}} \sum_{m=1}^{M} \zeta_m \sqrt{\mu_{\omega_m}} \overline{\hat{\phi}\left(\frac{\omega_m}{N}\right)} e_{\omega_m}\left(\frac{k-a+1}{N}\right), \quad k = a, ..., N-1-a. \tag{6.19}$$

This can be split in three functions similar to the first algorithm:

- $V : \mathbb{C}^M \to \mathbb{C}^M$:
$$V(\gamma) = (\sqrt{\mu_{\omega_m}} \gamma_m)_{m=1}^{M}. \tag{6.20}$$
- $D^* : \mathbb{C}^M \to \mathbb{C}^M$:
$$D(\zeta)^* = \left(\overline{\hat{\phi}\left(\frac{\omega_m}{N}\right)} \zeta_m\right)_{m=1}^{M}. \tag{6.21}$$
- $F^* : \mathbb{C}^M \to \mathbb{C}^N$:
$$F(\zeta)^* = \frac{1}{\sqrt{N}} \sum_{m=1}^{M} \zeta_k e_{\omega_m}\left(\frac{k-a+1}{N}\right), \quad k = a, ..., N-1-a, \tag{6.22}$$

so $\gamma_k = F^*(D^*(V(\zeta)))$, $k = a, ..., N-1-a$. On the boundary we have

$$\gamma_k = \begin{cases} \sum_{m=1}^{M} \zeta_m \sqrt{\mu_{\omega_m}} \overline{\hat{\phi}_{J,k}^{L}(\omega_m)}, & k = 0, \dots, a-1, \\ \sum_{m=1}^{M} \zeta_m \sqrt{\mu_{\omega_m}} \overline{\hat{\phi}_{J,k}^{R}(\omega_m)}, & k = N-a, \dots, N-1. \end{cases} \tag{6.23}$$

Algorithm 2: Fast computation of $\mathcal{B}^*\zeta$

```
Input: $\zeta \in \mathbb{C}^N$.
Output: $\gamma \in \mathbb{C}^M$.
1 $\tilde{\zeta} \leftarrow V(\zeta)$;
2 Compute the coefficients at the boundary using (6.23) for
    $k = 0, \dots, a-1, N-a, \dots, N-1$;
3 $\tilde{\zeta}_\phi \leftarrow D^*(\tilde{\zeta})$;
4 $\gamma_k \leftarrow F^*(\tilde{\zeta}_\phi)$ for $k = a, \dots, N-1-a$;
```

Remark 6.4: In the case of $a = 1$, where there are no boundary functions, simply skip step 2 in the algorithm and compute γ_k as in step 4 for all k. ◂

Algorithm 1 and 2 are implemented in the file `ForwardAndAdjointOperation.py`, and the functions F, F^*, D, D^* and V can be found in the file `SupportModule.py`. Algorithms 1 and 2 both have complexity of $\mathcal{O}\left(aM + M\log\left(\frac{N}{\epsilon}\right)\right)$, [Gataric and Poon, 2016, p. A1088].

Remark 6.5: Note that, if the Fourier samples are sampled uniformly with spacing ϵ, the weights are $\mu_m = \epsilon$ for all m. ◀

6.3 Optimization

It is assumed that the reader is somewhat familiar with optimization methods and therefore the theory will not be described rigorously. Some literature on the subject can be found in e.g. [Nocedal and Wright, 2006] and [Antoniou and Lu, 2007].

The optimization problem, that will be considered here, is

$$\underset{\alpha}{\text{minimize}}\ g(\alpha), \quad g(\alpha) = ||\mathcal{B}\alpha - \beta||_2^2, \tag{6.24}$$

since this objective function is suggested in (4.10). This is a linear least squares problem, hence it is a convex problem. The objective function, g, is easily seen to be continuously differentiable and the gradient of g is

$$\nabla g(\alpha) = 2\mathcal{B}^*(\mathcal{B}\alpha - \beta). \tag{6.25}$$

In order to solve this least squares problem we use an algorithm called the forward-backward proximal splitting algorithm, or FISTA, [Beck and Teboulle, 2009]. An implementation can be found in the Python package pyunlocbox. Within this package, the function used, is `solvers.forward_backward()`. A function which does the whole optimization procedure can be found the Python module called `Optimization.py`. The algorithm has time complexity $\mathcal{O}\left(\frac{1}{k^2}\right)$, or in worst case $\mathcal{O}\left(\frac{1}{k}\right)$, where k is the iteration, [Beck and Teboulle, 2009, p. 186].

In Chapter 8 there will be need for optimization of a matrix instead of vector. The above described algorithm can easily do this. The only changes are that $\alpha \in \mathbb{C}^{N\times N}$ and $\beta \in \mathbb{C}^{M\times M}$.

Note: In the search for a good optimization method a lot of different implementations has been considered. The method used here is the optimization method which was deemed to be the best of those tried. We have not been able to find an implemented method which was made specifically for a setup like this, i.e. a complex matrix $\mathcal{B}$ which were made into a forward algorithm and an adjoint algorithm while optimizing over α which is a real vector. We believe there are something to be gained if such a method were found or made, but further research on this topic is outside the scope of the thesis.

Chapter 7

Examples

This chapter contains examples of different change of basis cases starting with the Haar case and then continuing to Db2. We have done these examples by sampling a function in time and then used the discrete nonharmonic Fourier transform from Section 4.2.1 to get Fourier samples. When Fourier samples are obtained this way they must be scaled, otherwise the reconstructed signal will be different from the original signal by a constant factor. Note that we pretend to sample in the frequency domain and not in the time domain. All test functions used in this chapter can be found in `TestSignals.py`.

7.1 Haar

In the case of the Haar wavelet, the Fourier samples must be scaled by a factor $\mathcal{K} = \sqrt{\epsilon}$; this factor has been determined experimentally. In real world cases where a signal is sampled directly in the frequency domain this constant is without relevance.

7.1.1 Example 1

First we choose the parameters for the example. They are all chosen according to Theorem 5.10, but M is chosen higher than needed. J were chosen first and the other parameters afterwards.

$$\begin{aligned} J &= 9, \\ N &= 2^J, \\ M &= 2^{J+1}, \\ \epsilon &= 1, \\ x &\in [0,1)^{M\epsilon}, \end{aligned} \tag{7.1}$$

where J is the scaling parameter for the scaling function, $\phi_{J,k}(x)$, and x is the free variable. x is sampled uniformly on the interval and sorted such that $0 = x_1 < x_2 <$

$\ldots < x_{M\epsilon}$. We choose the function $f_0(x_i)$ for this example:

$$f_0(x_i) = \begin{cases} 0, & \text{if } x_i < 0 \text{ or } x_i \geq 1, \\ 3, & \text{if } 0 \leq x_i < 0.25, \\ 0.75, & \text{if } 0.25 \leq x_i < 0.5, \\ 1, & \text{if } 0.5 \leq x_i < 0.75, \\ 2, & \text{if } 0.75 \leq x_i < 1. \end{cases} \tag{7.2}$$

We find β, the discrete nonharmonic Fourier transform of the function, as described in Section 4.2.1.

The sampling scheme is $\Omega = \{-512, -511, \ldots, 511\}$ because we chose uniform sampling with $\epsilon = 1$. The weights of the Fourier samples can easily be calculated based on the sampling scheme. Since the sampling is uniform, all weights are equal: $\mu_m = \epsilon = 1$ for $m = 1, 2, \ldots, M$.

Now the change of basis matrix, $\mathcal{B}$, can be calculated explicitly or we can simply use the algorithms for matrix vector products, Algorithm 1 and 2, in the upcoming optimization. Here it is chosen to do the latter.

We wish to optimize the wavelet coefficients, α, as described in Section 6.3. We need an initial guess which is chosen to be $\alpha^0 = \{1, \ldots, 1\}$ of length N. It is important that N is a power of 2 for the algorithms to work. We optimize for α which results in $\alpha^\star$. If we calculate $\mathcal{B}\alpha^\star$, we get the Fourier coefficients corresponding to the wavelet coefficients. These can be seen as the orange graphs in the first column of Figure 7.1. Furthermore, we have that $||\mathcal{B}\alpha^\star - \beta|| = 1.546 \cdot 10^{-3}$.

In Figure 7.1 it is seen that $\mathcal{B}\alpha^\star$ visually lies very close to β, especially in the low frequencies and a little less in the high frequencies.

We investigate how well we can reconstruct the original signal, $f_0(x)$, from the wavelet coefficients. Here we use (2.9),

$$\tilde{f}_0(x) = \sum_{k=0}^{2^J-1} \alpha_k^\star \phi_{J,k}(x), \quad J = 9. \tag{7.3}$$

The result can be seen in Figure 7.1 in the right column. The blue dots are the original signal and the orange graphs are the reconstructions from (7.3).

To compare $f_0(x)$ and $\tilde{f}_0(x)$, we calculate the distance between the two sequences:

$$||\tilde{f}_0 - f_0|| = 0.7796. \tag{7.4}$$

To this end, the conclusion is that the Haar wavelet can represent a piecewise constant function very well when using generalized sampling as described.

The green line, in Figure 7.1, is the inverse discrete nonharmonic Fourier transform of the M Fourier coefficients. It is visually evident that this does not represent the original signal, in between sample points, as well as the Haar coefficients. With this example we show that, for some signals, we can get both a better representation of the signal and compress it at the same time.

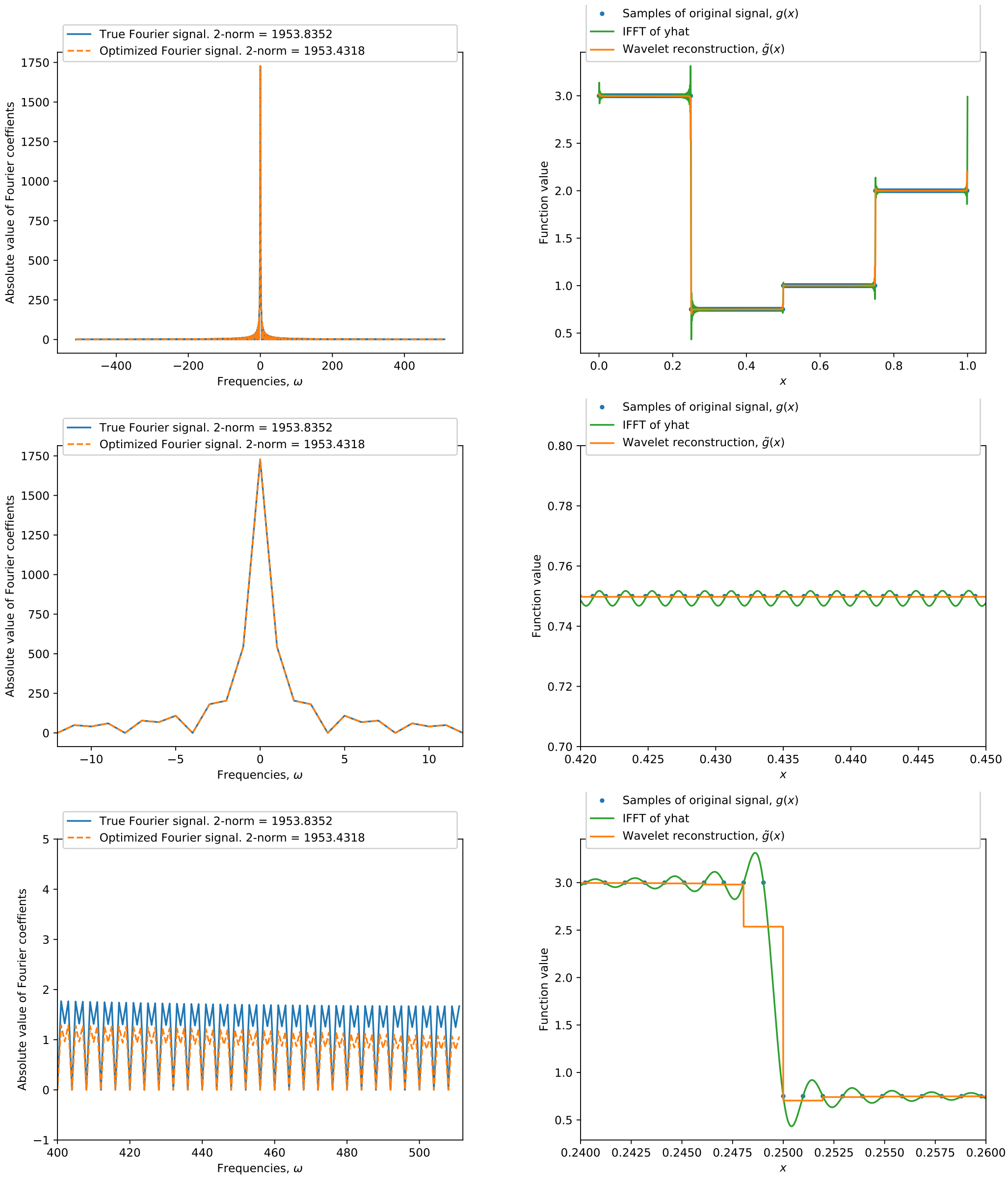


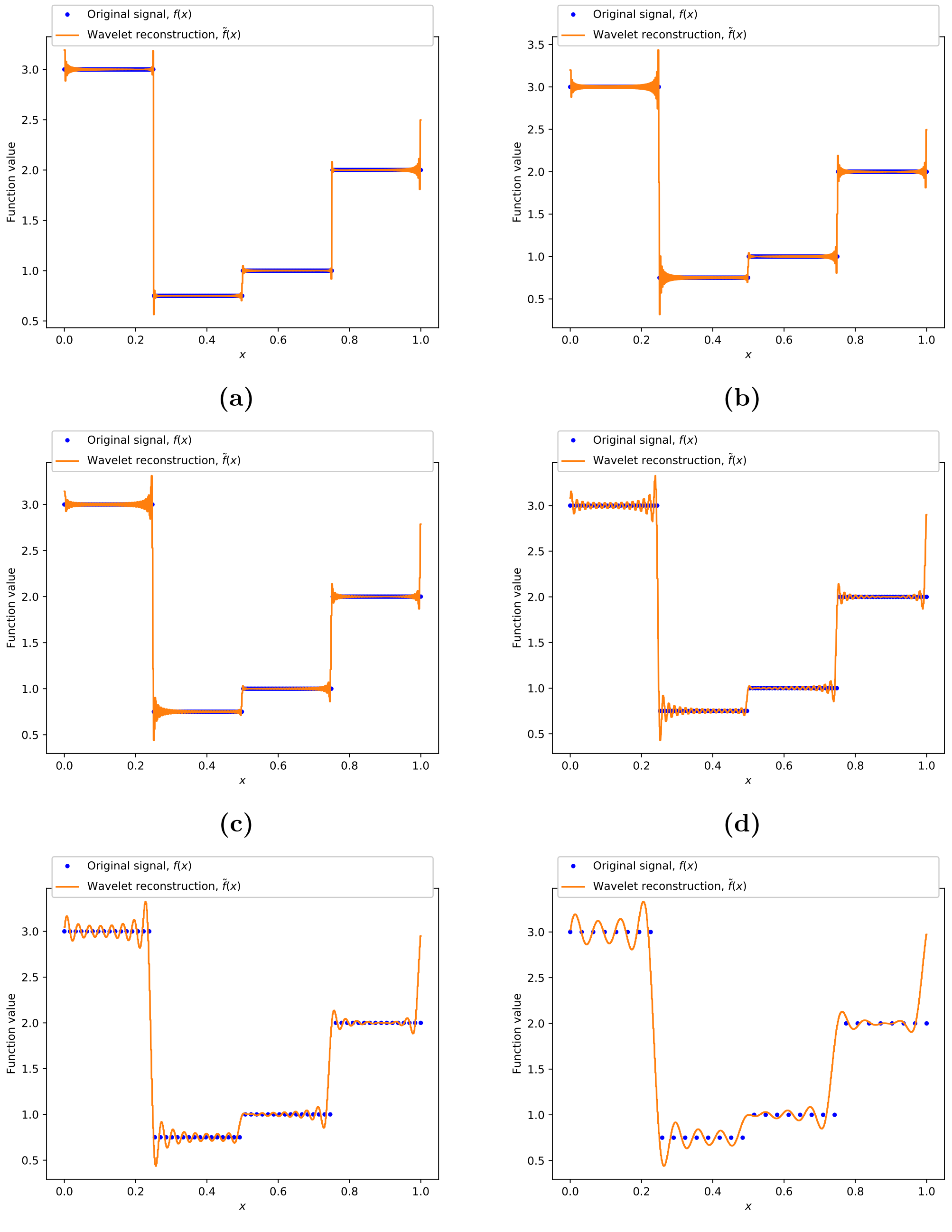
Figure 7.1: The left column shows the absolute values of the Fourier samples. The blue graph is β and the orange graph is $\mathcal{B}\alpha^*$. The first row is the full signal, the second row is a zoom of the low frequencies and the third row is a zoom of the high frequencies.

The right column shows the reconstructed signal. The blue points are the original signal, f_0, the orange graph is the reconstructed signal and the green graph is the inverse discrete nonharmonic Fourier transform of all $M = 1024$ samples. The first row is the full signal, the second row is a zoom of one of the constant pieces and the third row is a zoom of a discontinuity.

Other Ratios between Coefficients

In Figure 7.2 the reconstructions of f_0 for some different ratios between the number of Fourier coefficients and wavelet coefficients have been plotted. The sizes, (N, M), in the figure are: $(512, 514)$, $(512, 512)$, $(512, 256)$, $(512, 128)$, $(512, 64)$ and $(512, 32)$. It is visually evident that choosing M just slightly higher than the stable sampling rate results in a big improvement.

As the number of Fourier coefficients decreases the wavelet reconstruction gets worse. However not all information is lost.



(a)

(b)

(c)

(d)

(e)

(f)

Figure 7.2: The reconstructions of f_0 for some different ratios between the number of Fourier coefficients and wavelet coefficients. The sizes (N, M) are as follows (a):$(512, 514)$, (b):$(512, 512)$, (c):$(512, 256)$, (d):$(512, 128)$, (e):$(512, 64)$ and (f):$(512, 32)$.

7.1.2 Example 2

As a second example we have chosen the function

$$f_3(x_i) = \begin{cases} 0, & \text{if } x_i < 0 \text{ or } x_i \geq 1, \\ 3, & \text{if } 0 \leq x_i < 0.25, \\ \cos(100\pi x_i), & \text{if } 0.25 \leq x_i < 0.5, \\ 1, & \text{if } 0.5 \leq x_i < 0.75, \\ \sin(20\pi x_i), & \text{if } 0.75 \leq x_i < 1. \end{cases} \tag{7.5}$$

We have followed the same procedure as with $f_0(x)$ and used the parameters in (7.1). The function can be seen in the left graph of Figure 7.3. In this example we will focus on how the system handles high frequency signals. To this end, we will focus the interval with the 50 Hz trigonometric polynomial which is shown in the right graph in the figure.

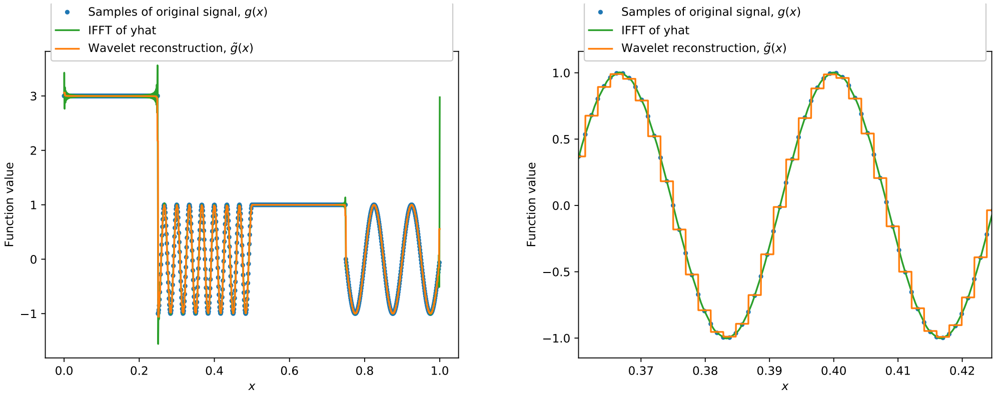


Figure 7.3: The plots show f_3 (blue dots), $\tilde{f}_3$ (orange) and $\mathcal{F}^{-1}(f_3)$ (green).

As we saw in the previous example, the Haar basis is very well suited for representing constant functions and discontinuities, something which the Fourier frame is not. Trigonometic polynomials on the other hand is very well represented in the Fourier frame and less so in the Haar basis. This is visually evident from the right graph of Figure 7.3.

The top graphs of Figure 7.4 show what happens when $M = 128$. The reconstruction is still visually recognizable, but has similar errors as we saw in Figure 7.2. The bottom graphs show what happens when $M = 64$. In this case we can not reconstruct the signal. The reason for this, is that the sampling density is kept constant in the frequency domain and sampled in different frequency intervals. Hence, when taking 64 samples symmetrically around 0 with a density of $\epsilon = 1$ we only get samples in the interval $[-32, 31]$ Hz and therefore we can not reconstruct a signal of 50 Hz.

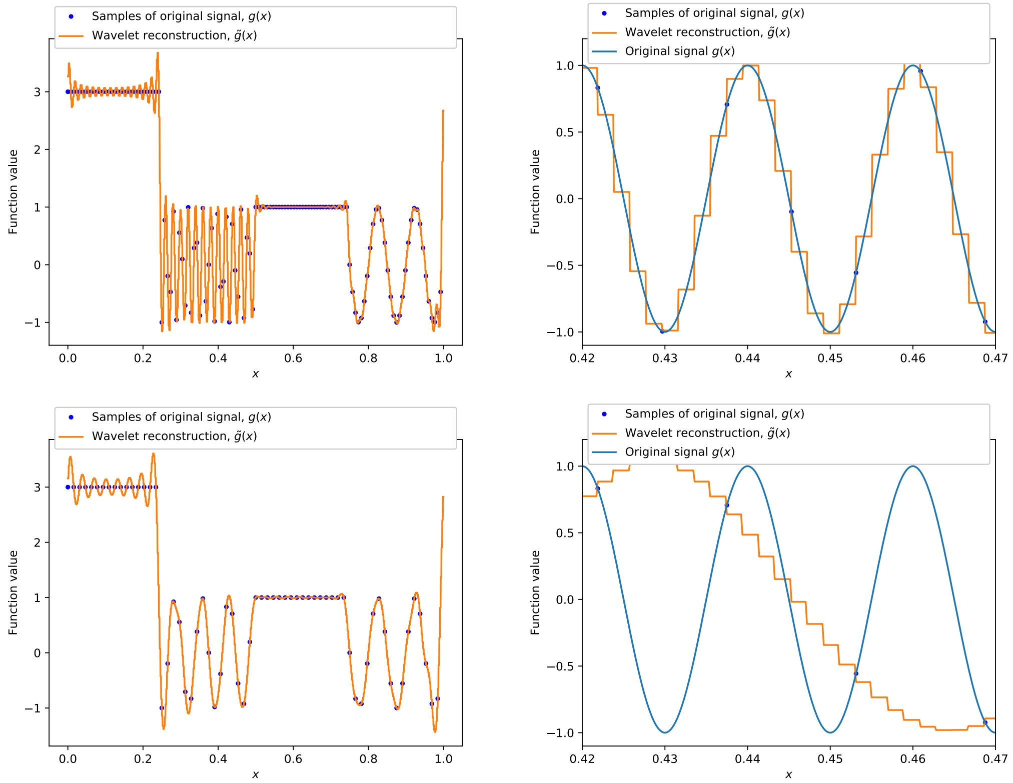


Figure 7.4: The plots show f_3 (blue), the samples (dots) and $\tilde{f}_3$ (orange). The top plots is the wavelet reconstruction using 128 Fourier samples and the bottom is using 64 Fourier samples.

Other Sampling Densities

We will calculate the stable sampling rate for the Haar wavelet in this section. The calculations for Daubechies 2 can be found in Section 7.2.1. Recall that the Fourier transform of the Haar scaling function is given in (6.8):

$$\mathcal{F}[\phi](\omega) = \begin{cases} \frac{1-\exp(-2\pi i\omega)}{2\pi i\omega}, & \omega \neq 0, \\ 1, & \omega = 0. \end{cases}$$

From (5.26) we get that

$$\begin{aligned} \epsilon &\leq \frac{1}{\tau_1 + \tau_2} \\ &\leq \frac{1}{(\lceil 2a-1 \rceil - 1) + (2\lceil 2a-1 \rceil - 1)} \\ &= \frac{1}{0+1} = 1. \end{aligned} \tag{7.6}$$

From Theorem 5.10 we then get

$$\begin{aligned}\theta &> \left(\inf_{\omega\in[-1/2,1/2]} \left|\hat{\phi}(\omega)\right|\right)^{-1}\\ &= \left|\hat{\phi}\left(\frac{1}{2}\right)\right|^{-1}\\ &= \left|\frac{1-\exp\left(-2\pi i\frac{1}{2}\right)}{2\pi i\frac{1}{2}}\right|^{-1}\\ &= \left|\frac{2}{i\pi}\right|^{-1}\\ &= \frac{\pi}{2}.\end{aligned} \tag{7.7}$$

For the special case of the Haar wavelet we have that Theorem 5.10 applies when $\theta \geq \frac{\pi}{2}$. We can now find the stable sampling rate by direct computation of $\Theta(N_R;\theta) = \lceil 2^R/\epsilon \rceil$ for $R \in \mathbb{N}$ large enough. We calculate the stable sampling rate for $\epsilon_1 = 1$ and $\epsilon_2 = \frac{1}{2}$ and get

$$\Theta\left(N_R;\frac{\pi}{2}\right)_{\epsilon_1} = \frac{2^R}{1} = 2^R, \tag{7.8}$$

$$\Theta\left(N_R;\frac{\pi}{2}\right)_{\epsilon_2} = \frac{2^R}{\frac{1}{2}} = 2^{R+1}. \tag{7.9}$$

Thus if we choose N_R as in the example, i.e. 2^9, we get that

$$\Theta\left(2^9;\frac{\pi}{2}\right)_{\epsilon_1} = \frac{2^9}{1} = 2^9, \tag{7.10}$$

$$\Theta\left(2^9;\frac{\pi}{2}\right)_{\epsilon_2} = \frac{2^9}{\frac{1}{2}} = 2^{10}, \tag{7.11}$$

meaning that when we have at least $\Theta\left(2^9;\frac{\pi}{2}\right)$ samples the reconstruction will be stable.

An example of sampling with other densities will now be presented. Two cases are being investigated: One where the sample density $\epsilon = 0.5$, and one where $\epsilon = 1.11$. Figure 7.5 shows the reconstruction of the signals in the first row and the sampled frequencies in the second row. The first column is with every 0.5 Hz and the second is with 1.11 Hz. According to the theory in Section 5.5 and the calculation in (7.6), the reconstruction with Haar wavelets should succeed if the density, ϵ, is less than or equal to 1. Furthermore if $\epsilon > 1$ we do not have a weighted Fourier frame, due to Theorem 4.12, so the method is bound to fail. From the plots in Figure 7.5, it is visually evident that this is the case. With $\epsilon = 0.5$ we have reconstructed the signal and when $\epsilon = 1.11$ we are unable to reconstruct the original signal.

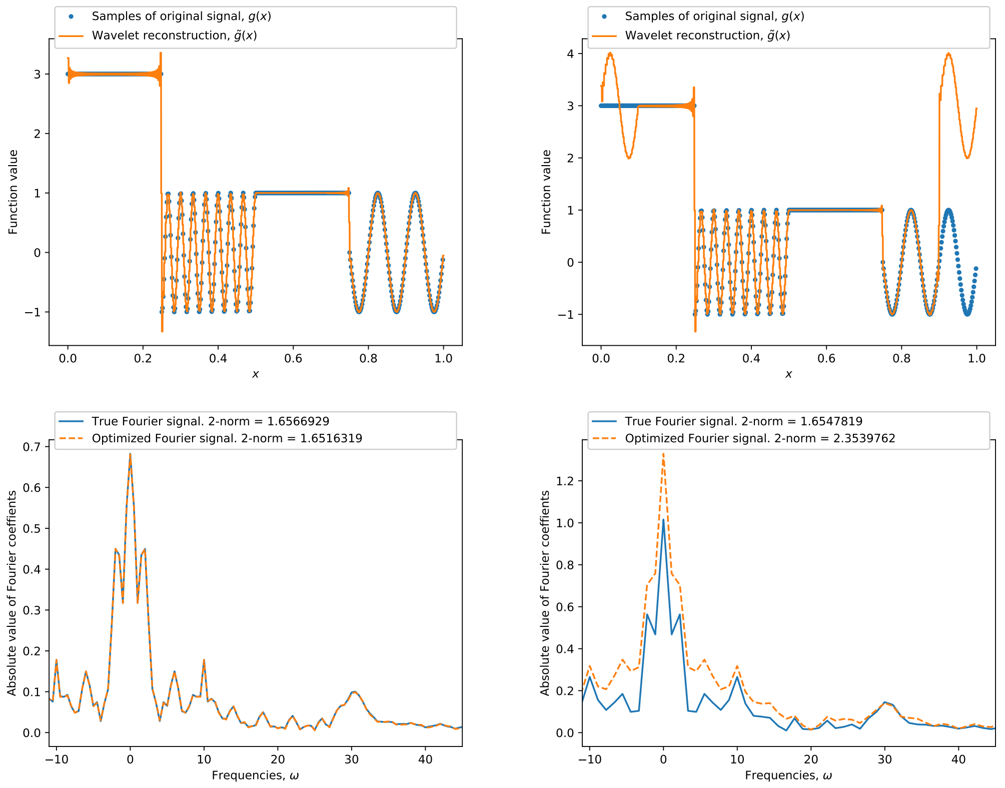


Figure 7.5: In the first column $\epsilon = 0.5$ and in the second column $\epsilon = 1.11$. The first row shows the original signal (blue dots) and the reconstruction (orange). The second row shows the original Fourier coefficients (blue) and the optimized Fourier coefficients (orange).

7.1.3 Summary of Results

We have tested that the algorithms work by choosing M to be two times the stable sampling rate. For this choice, the Haar wavelet reconstruction is better than the Fourier reconstruction for signals with discontinuities.

We then fixed the density, $\epsilon = 1$, and varied the sampling interval. When M equals the stable sampling rate, the reconstruction oscillates around the discontinuities. As M gets smaller the oscillations around the discontinuities gets worse. If the sampling interval is narrow enough, we are unable to reconstruct high frequency trigonometric functions due to the Nyquist-Shannon sampling theorem. Conversely, if M gets larger, even by only a few samples, better reconstructions are obtained.

Furthermore, we fixed the sampling interval and varied the density, ϵ. When $\epsilon \leq 1$ we reconstruct without problems, but when $\epsilon > 1$ aliasing arises and we are not able to reconstruct the original signal.

7.2 Daubechies 2

In this section the Daubechies wavelet with two vanishing moments, Db2, will be tested as reconstruction basis for generalized sampling. The two main differences between the Haar wavelet and the Db2 wavelet is that Db2 can reconstruct polynomials of degree 1 and that boundary wavelets are needed. A few test signals, specifically designed for this wavelet, has been made in order to see how generalized sampling can help reconstruct signals.

Like in the Haar case we multiply a factor, $\mathcal{K}$, on the Fourier samples. The factor is $\mathcal{K} = \frac{\|f_s\|\sqrt{\frac{N}{M}}}{\|\hat{f}\|}$, where f_s is the signal sampled in M points. Recall that this factor is only for the comparison of the original signal and the reconstruction. In cases where a signal is sampled directly in the frequency domain this factor is without relevance. $\mathcal{K}$ has been determined experimentally.

Now we will calculate the sampling density, ϵ, then the constant θ and then the stable sampling rate, $\Theta\left(N_R;\theta\right)$, recall Table 5.2. When this is done, the examples will be given.

7.2.1 The Stable Sampling Rate

First we find ϵ using (5.26): $\tau_1 \geq 2a-1-1$, $\tau_2 \geq 2(2a-1)-1$ and $a=2$, so we get $\tau_1 \geq 2$, $\tau_2 \geq 5$ and

$$\epsilon \leq \frac{1}{\tau_1+\tau_2} \leq \frac{1}{7}. \tag{7.12}$$

Next we find θ using Theorem 5.10. The formula is

$$\theta > \left(\inf_{\omega\in[-1/2,1/2]} \left|\hat{\phi}(\omega)\right|\right)^{-1}, \tag{7.13}$$

so we need to estimate $\hat{\phi}(\omega)$ in the interval $\left[-\frac{1}{2},\frac{1}{2}\right]$. The function is defined by

$$\hat{\phi}(\omega) = \prod_{j=1}^{\infty} \sum_{k=0}^{2a-1} h_k \exp(-2\pi i k 2^{-j}\omega), \tag{7.14}$$

where we choose to take only the first 20 factors of the product. Figure 7.6 shows the absolute value of the function and it is visually evident that the infimum is at the interval boundaries. We find that $\theta^{-1} < 0.684$ which agrees with the result from [Adcock et al., 2014b, bottom of p. 410]. Theorem 5.10 also states that the stable sampling rate is $\Theta(N_R;\theta) = \left\lceil\frac{2^R}{\epsilon}\right\rceil = 7\cdot 2^R$.

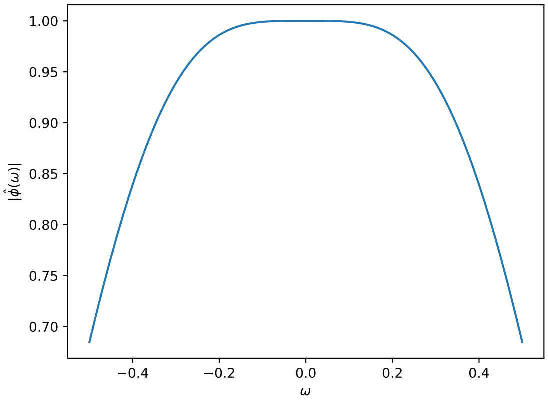


Figure 7.6: $\left|\hat{\phi}(\omega)\right|$ in the interval $\left[-\frac{1}{2}, \frac{1}{2}\right]$.

7.2.2 Orthogonalization of the Boundary Wavelets

As a final test, to see if the boundary wavelets should be orthogonalized, we run the optimization on two test signals both with and without orthogonalizing the boundary wavelets. Other tests have been done in Sections 5.4 and 6.1.2. At the left boundary, it does not matter if the boundary wavelets are orthogonal. Figure 7.7 shows that on the right boundary the reconstruction is slightly worse when they are orthogonal. Therefore we will not orthogonalize the boundary functions in the examples that follow.

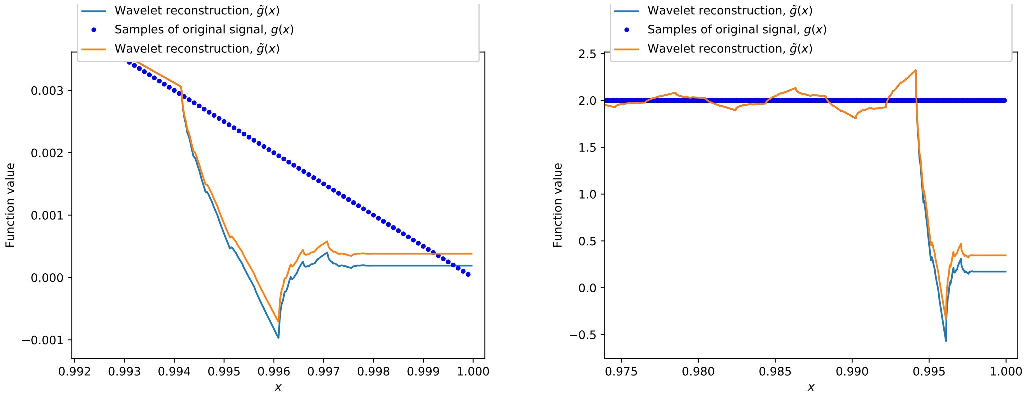


Figure 7.7: The right edge of the signal. Optimized using orthogonal boundary functions (blue) and non-orthogonal boundary functions (orange). On the left is a signal which goes linearly to zero at the edge and the right is a signal which is constant at the edge.

7.2.3 Example 1

The procedure for the examples will be as in Section 7.1.1. Unless otherwise stated, in all the examples with Db2, set

$$\begin{aligned} J &= 9, \\ N &= 2^J, \\ \epsilon &= \frac{1}{7}, \\ M &= 1792, \\ x &\in [0,1)^{M\epsilon}, \end{aligned} \tag{7.15}$$

where M is chosen according to Theorem 5.10. In all the following examples we will have more Fourier coefficients than wavelet coefficients, i.e. a compression is made. Throughout the examples we will focus on two things: How close the reconstruction is to the original and if it is better than reconstructing from Fourier coefficients.

In this first example a continuous, piecewise linear function, which starts and ends in 0, is chosen:

$$f_1(x_i) = \begin{cases} 0, & \text{if } x_i < 0 \text{ or } x_i \geq 1, \\ 3x_i, & \text{if } 0 \leq x_i < 0.25, \\ 0.75, & \text{if } 0.25 \leq x_i < 0.5, \\ -\frac{5}{2}x_i + 2, & \text{if } 0.5 \leq x_i < 0.75, \\ -\frac{1}{2}x_i + \frac{1}{2}, & \text{if } 0.75 \leq x_i < 1. \end{cases} \tag{7.16}$$

Figure 7.8 shows the result of generalized sampling from Fourier to wavelet on f_1. For this simple continuous function the method works well. However, there are small errors around the nondifferential points. The optimization method produces a solution where $\|\mathcal{B}\alpha^\star - \beta\| = 8.765 \cdot 10^{-6}$. We can compare the reconstructed signal to the original by sampling the reconstruction in the same 256 points as the original signal and then calculate the norm:

$$\|f_1 - \tilde{f}_1\| = 2.996 \cdot 10^{-3}. \tag{7.17}$$

In an attempt to compress the signal further, we set $J = 8$ and $N = 2^J$ while the rest of the parameters stay the same. The error is $\|f_1 - \tilde{f}_1\| = 1.282 \cdot 10^{-2}$ which is significantly higher than when we use the stable reconstruction rate for $M = 1792$. It illustrates the trade-off between compression and reconstruction error.

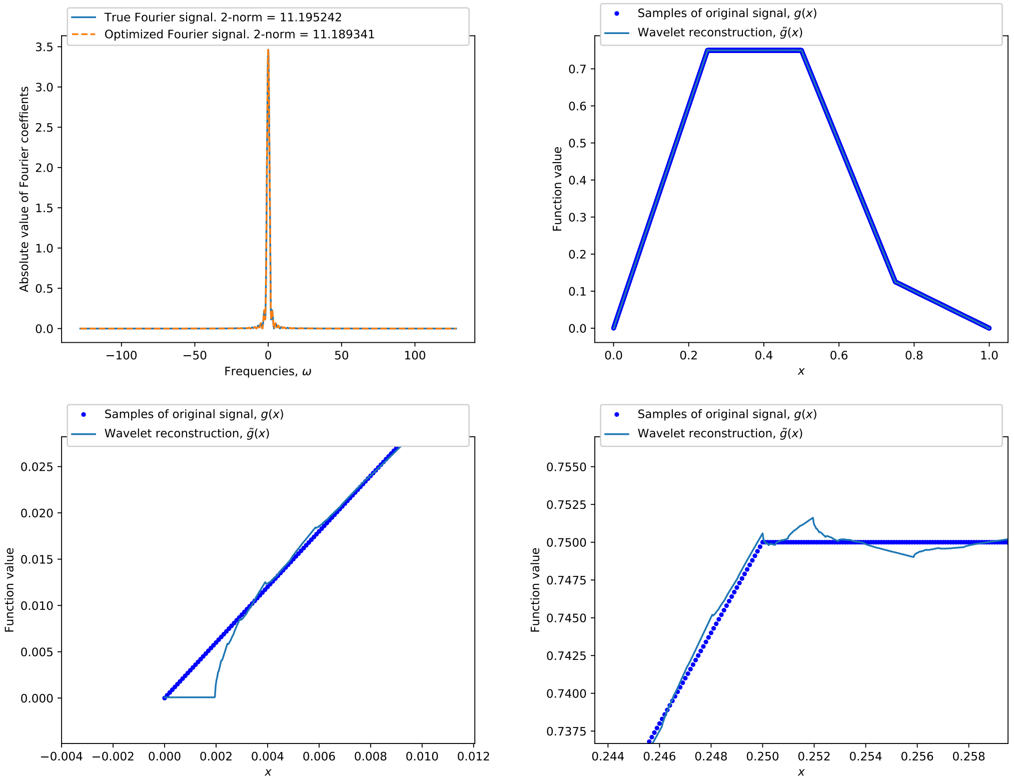


Figure 7.8: The top left plot contains β (blue) and $\mathcal{B}\alpha^*$ (orange). The other three show the original signal in the time domain (dots) and the reconstruction (blue). Bottom left is a zoom at the left edge and the bottom right is a zoom at the first nondifferential point.

7.2.4 Example 2

In this example a piecewise continuous function is chosen:

$$f_2(x_i) = \begin{cases} 0, & \text{if } x_i < 0 \text{ or } x_i \geq 1, \\ x_i, & \text{if } 0 \leq x_i < 0.25, \\ 10(x_i - 0.25)^2 + 1, & \text{if } 0.25 \leq x_i < 0.5, \\ 1.625, & \text{if } 0.5 \leq x_i < 0.75, \\ -2x_i + 2, & \text{if } 0.75 \leq x_i < 1. \end{cases} \tag{7.18}$$

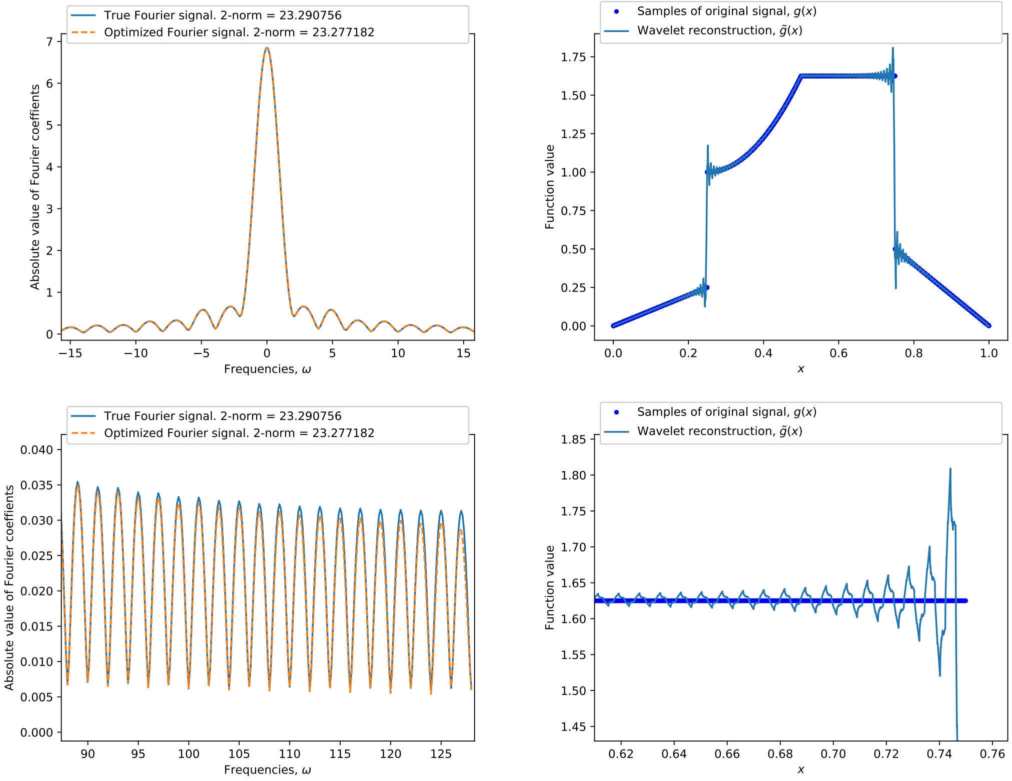


Figure 7.9: The left is β (blue) and $\mathcal{B}\alpha^*$ (orange), the top is a zoom of low frequencies and the bottom is a zoom of high frequencies. The right show the original signal in the time domain (dots) and the reconstruction (line). The top is the whole signal and the bottom is a zoom at the second discontinuity.

Around the discontinuities in the signal, there appears what seems to be Gibbs phenomenon. The oscillations are slightly larger than what can be explained as Gibbs phenomenon, but this might be caused by the poor optimization at the high frequencies. In an attempt to get a better result, we use twice as many Fourier coefficients, i.e. $M = 3584$. It is visually evident in Figure 7.10 that we do get a better result by using twice as many Fourier samples. However, it does not completely remove the oscillations around the discontinuities which support the claim that it is Gibbs phenomenon.

The optimization method, in the case $M = 1792$, produces a solution where $\|\mathcal{B}\alpha^* - \beta\| = 0.7388 \cdot 10^{-3}$. We compare the reconstruction to the original, as in Section 7.2.3, and get

$$\|f_2 - \tilde{f}_2\| = 0.1568. \tag{7.19}$$

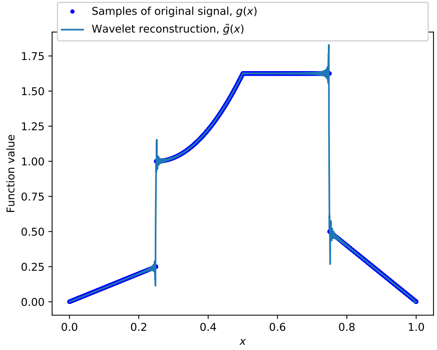


Figure 7.10: The figure shows the original signal in the time domain (dots) and the reconstruction (line) for $M = 3584$.

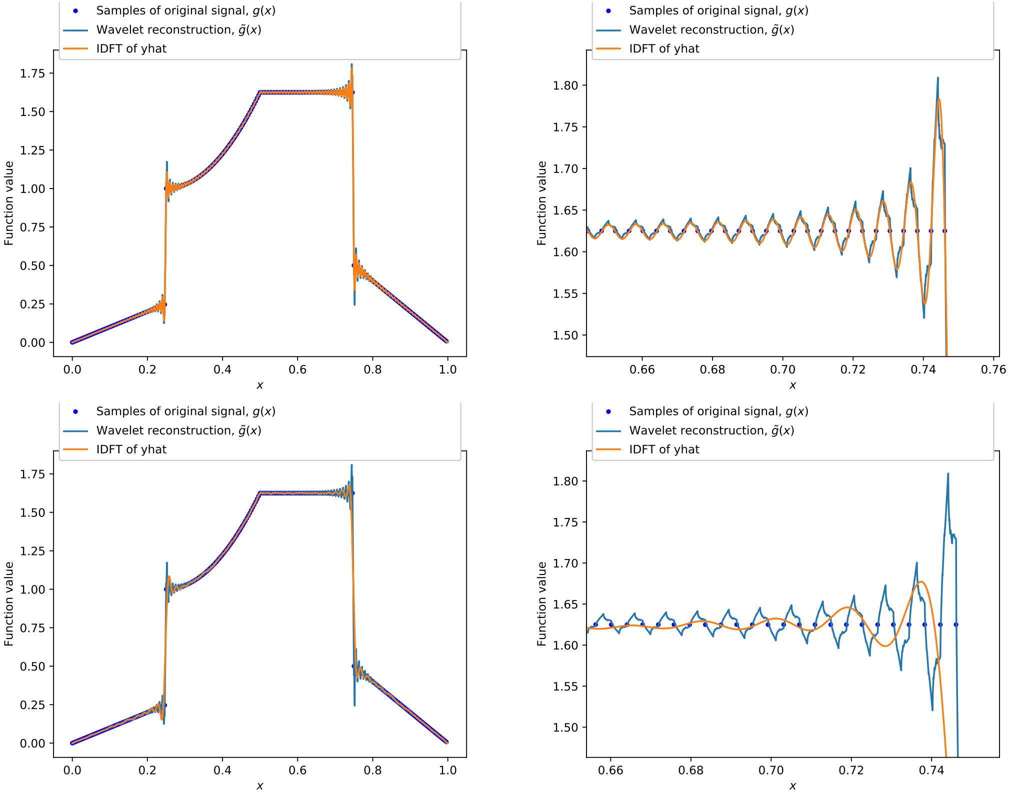


Figure 7.11: The figure show the original signal in the time domain (dots), the wavelet reconstruction (blue line) and the reconstruction from the Fourier coefficients (orange). The left is the whole signal and the right is a zoom at the second discontinuity, the top is with all Fourier coefficients and the bottom is with the N largest Fourier coefficients.

Figure 7.11 shows the samples of the original signal together with the reconstruction from the wavelet coefficients and the reconstruction from the Fourier coefficients. It is visually evident, in the first row of the figure, that both reconstructions goes through all sample points and oscillates in between. Now, set the smallest Fourier coefficients to zero, such that there are only N nonzero coefficients. We will call these truncated Fourier coefficients. When reconstructing from the truncated Fourier coefficients, the reconstruction does not go through all the sample points and the oscillations gets wider. This is illustrated in the bottom row of Figure 7.11.

When we compare the reconstructions to the original signal sampled uniformly in $M\epsilon$ points, we get $\|f_2 - \mathcal{F}^{-1}(\beta)\| = 0.6415$ for the truncated Fourier coefficients and $\|f_2 - \mathcal{F}^{-1}(\beta)\| = 0.001631$ for the complete set of Fourier coefficients. So when we compress to N coefficients directly in the Fourier frame, we get a larger error than in the wavelet basis.

Remark 7.1: The truncated Fourier coefficients are close to the actual Fourier coefficient, therefore the error that arise when using the IDFT is small. ◂

7.2.5 Example 3

In this example the same function as in Section 7.1.2 is used.

We suspect the reason for the very low values at the left edge of the reconstructed signal, in Figure 7.12, is due to the fact that the frame, in which we sample, is wider than the signal. From Theorem 4.16 we get that the radius of the frame is at least $\frac{7}{2}$, but our interval only has a radius of $\frac{1}{2}$. Per construction the signal is zero outside the interval $[0, 1]$. This, combined with the poor optimization at the high frequencies, may explain the low values at the left edge. The optimization method produces a solution where $\|\mathcal{B}\alpha^* - \beta\| = 16.96$. We compare the reconstruction to the original as in Section 7.2.3, and get that

$$\|f_3 - \tilde{f}_3\| = 2.992. \tag{7.20}$$

The orange line in Figure 7.12 is the reconstruction from truncated Fourier coefficients. It is visually evident that it is better both at the left edge, where our method has trouble optimizing, and at the trigonometric parts, where the Fourier frame is better suited. At the constant parts and at the discontinuities we get similar results as in Section 7.2.4, i.e. wavelet reconstruction is better. We compare the reconstruction from truncated Fourier coefficients to the original signal and get $\|f_3 - \mathcal{F}^{-1}(\beta)\| = 2.475$, which is slightly lower than for the wavelet reconstruction. So, for this signal, reconstruction with wavelets are not better, but the errors are of the same order of magnitude.

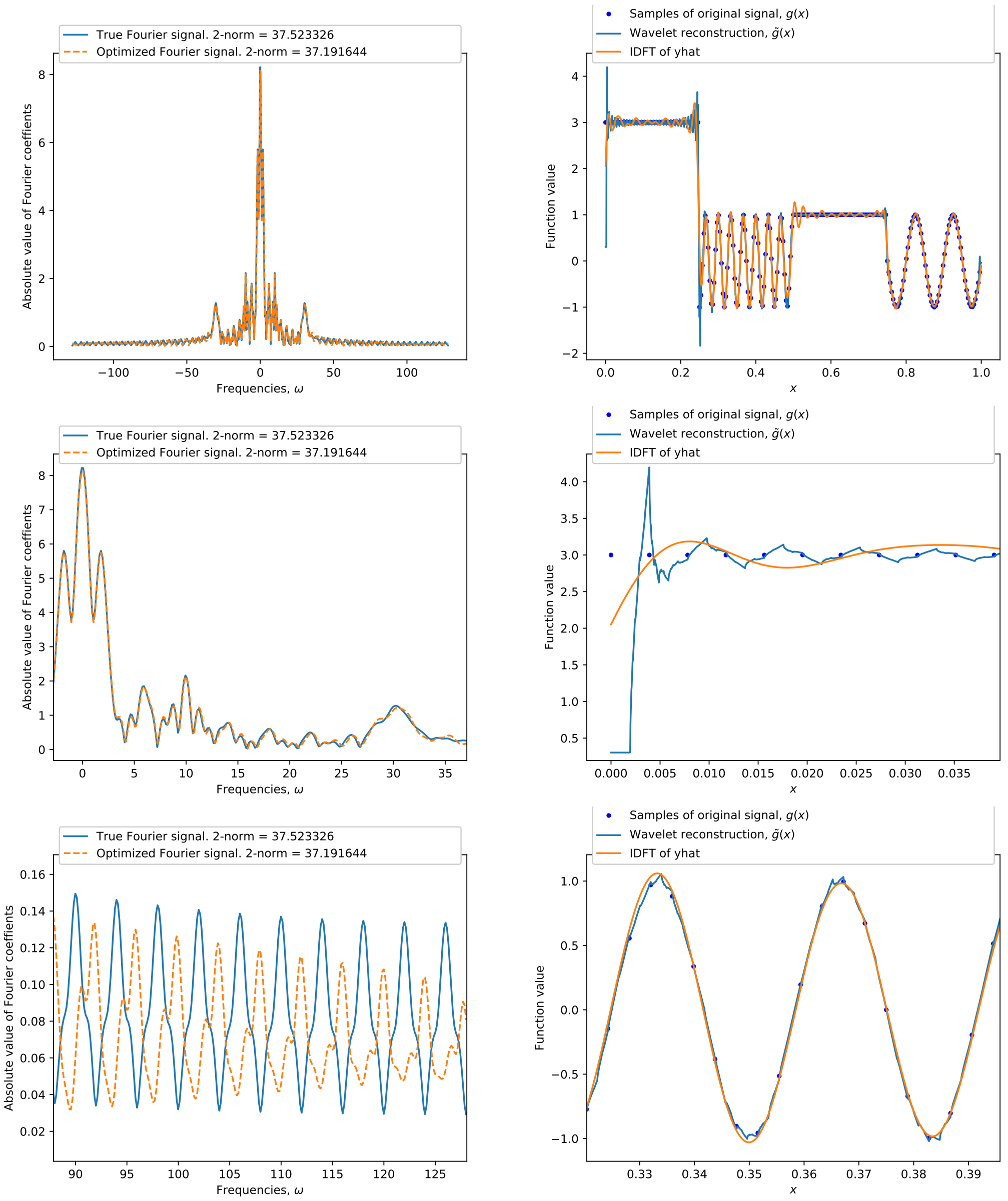


Figure 7.12: The left is β (blue) and $\mathcal{B}\alpha^*$ (orange), the top is the whole signal, the middle is a zoom of low frequencies and the bottom is a zoom of high frequencies. The right show the original signal in the time domain (dots) the wavelet reconstruction (blue line) and the reconstruction from truncated Fourier coefficients (orange). The top is the whole signal, the middle is a zoom of the left edge and the bottom is a zoom of the high frequency part of the signal.

7.2.6 Example 4

In this example, a function consisting of the sum of three sine waves will be used:

$$f_4(x_i) = \sin(2\pi 10 x_i) + \sin(2\pi 30 x_i) + \sin(2\pi 100 x_i). \tag{7.21}$$

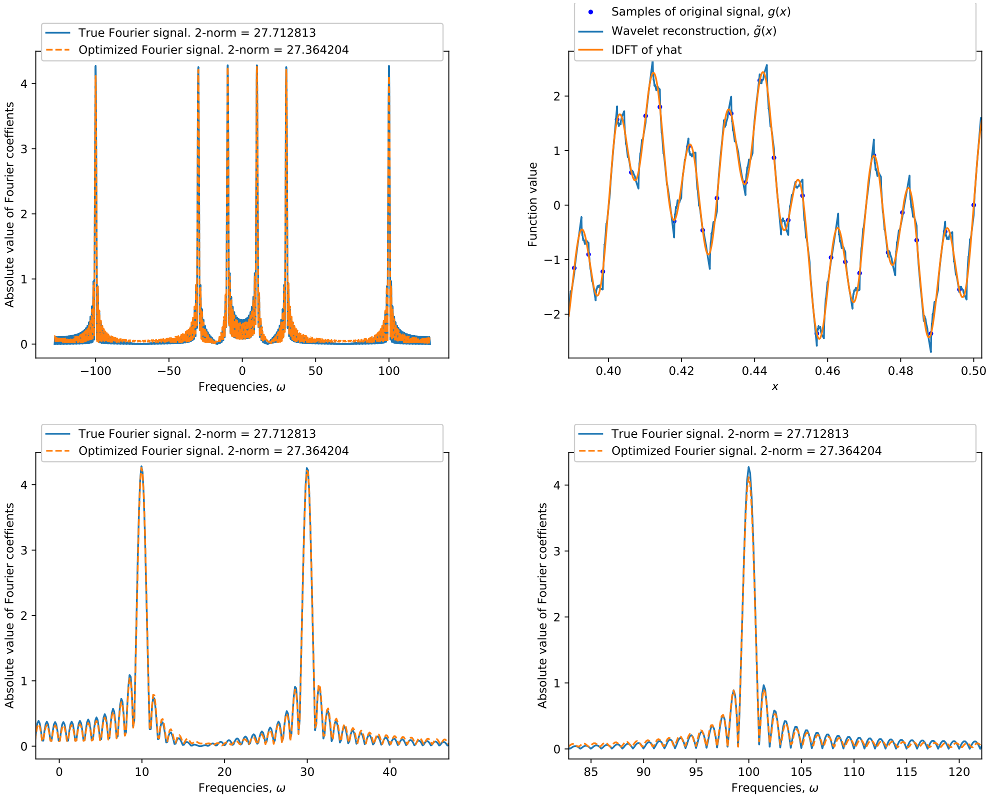


Figure 7.13: The top right show a zoom of the original signal in the time domain (dots) and the reconstruction (line). The other three are β (blue) and $\mathcal{B}\alpha^\star$ (orange). The top left is the whole signal, the bottom left is a zoom of low frequencies and the bottom right is a zoom of high frequencies.

Figure 7.13 shows that not all continuous functions can be reconstructed well with Db2. The rapid changes in the high frequency trigonometric polynomial results in some error at the top and bottom of each oscillation. Functions like f_4 can be reconstructed perfectly from the Fourier coefficients because it has bounded support in the frequency domain. In this case generalized sampling is not of much use. The optimization method produces a solution where $\|\mathcal{B}\alpha^\star - \beta\| = 4.334$. We compare the reconstruction to the original as in Section 7.2.3, and get that

$$\|f_4 - \tilde{f}_4\| = 3.621. \tag{7.22}$$

For comparison the reconstruction from truncated Fourier coefficients gives an error of 0.8952.

7.2.7 Summary of Results

We have shown that the algorithms work for M equal to the stable sampling rate and $\epsilon = \frac{1}{7}$. Generalized sampling is function dependent, i.e. the method does not work equally well for all types of functions. For functions with discontinuities there are some oscillations near the discontinuities. We are confident that this stems from Gibbs phenomenon. However, the reconstruction is very close to the original signal in all the original sampling points.

We see a significant error at the edges of the interval $[0, 1]$ for functions which are not zero on the boundary. This might be because the Fourier frame, in which we sample, is wider than the interval $[0, 1]$. Functions which change rapidly, for instance f_4, are hard to reconstruct in the Db2 basis as this basis is made for reconstructing first order polynomials.

When using generalized sampling, we automatically compress the data. Therefore we have compared the reconstructions with a compression of the data in the Fourier frame. For some of the tested signals the reconstruction error is smaller when using generalized sampling for the compression instead of truncating the Fourier coefficients. For signals which are well represented in the Fourier frame but not in the Db2 basis, such as trigonometric polynomials, the truncated Fourier reconstruction is better than generalized sampling.

7.3 Nonuniform Sampling

In this section we will shortly describe what happens when a signal is sampled nonuniformly in the Fourier frame. It is important that the sampling scheme fulfils the criteria in Theorem 4.12. To make sure of this, we make a jittered sampling scheme by taking a regular grid and adding a small perturbation to each point. The perturbations are pseudo random numbers drawn from a uniform distribution in the interval $[-\epsilon/p, \epsilon/p]$. We have done this for $p = 2, 3, ..., 160$ with different seeds for each p. We started with $p = 2$ to make sure that $\omega_0 < \omega_1 < ... < \omega_M$ which is a requirement for all sampling schemes. Note that this sampling scheme has uniform density equal to ϵ, Definition 4.15. We have used f_2, (7.18), as test function. For each p we have calculated the error, $\|f_2 - \tilde{f}_2\|$. This is illustrated in Figure 7.14, top left. The other plots in the figure show zooms of the reconstruction for all p.

The first five colours are blue, orange, green, red and purple which represent $p = 2, 3, 4, 5, 6$ respectively. Due to the randomness, $p = 6$ results in a larger error than $p = 5$. It is visually evident that, in places where the error of the non-perturbated reconstructions are large, the perturbated reconstructions are close to the non-perturbated reconstruction. However, in places where the error of the non-pertubated reconstruction is small, the pertubated reconstructions are farther from it.

From this experiment we see there is nothing to be gained from making random perturbations to the sampling scheme. however, there is not much lost either. Figure 7.14 shows that the error converge, to the error of the non-perturbated case, when the interval for the perturbations gets smaller.

In some real life cases we might not be able to sample in a completely regular grid due to, for instance, machine limitations. In such cases, it is possible to recover the signal if an almost regular grid is used for sampling.

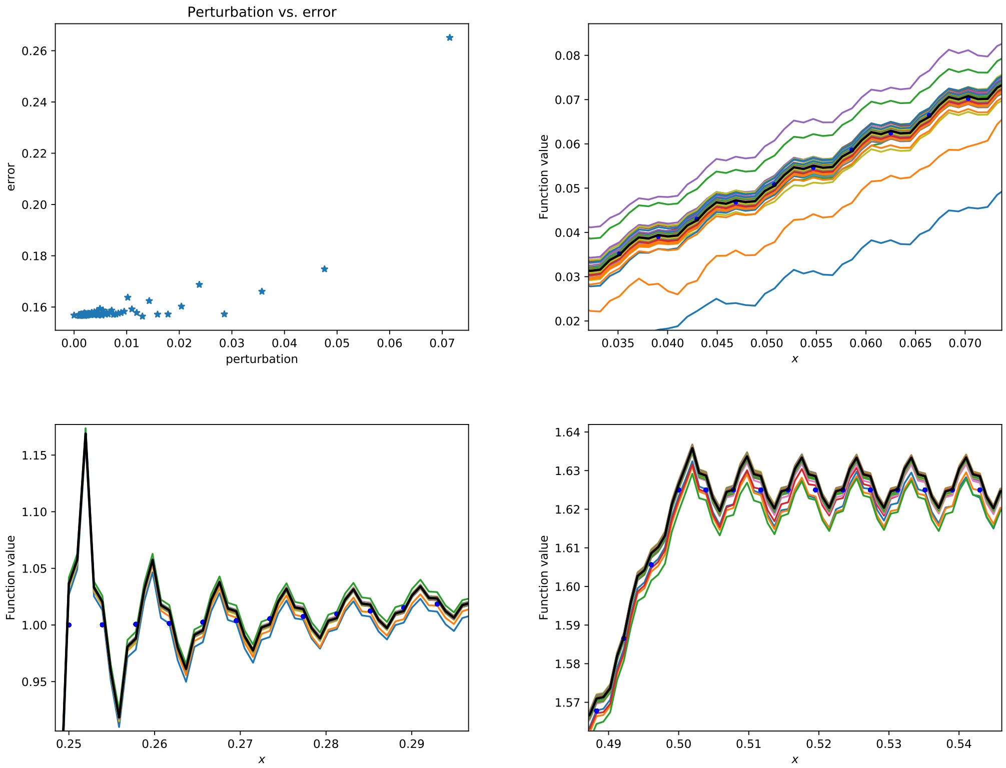


Figure 7.14: The top left plot show the error. The other three show different zooms of the reconstruction for all p, in different colours. The thick black curve in the middle is the reconstruction without perturbations.

7.4 Noisy Samples

We have made a test to see how a signal behaves if bandlimited, Gaussian noise is added to the Fourier samples. We use the signal to noise ratio, SNR, as a measure for the amount of noise added to the signal. The power of the signal is measured in the ℓ^2 norm. The SNR's which are used are 1,10,100 and 1000. Figure 7.15 shows the wavelet reconstructions for f_2, (7.18). The signal is almost unrecognisable when SNR=1, but as the SNR grows, the error decreases, Table 7.1.

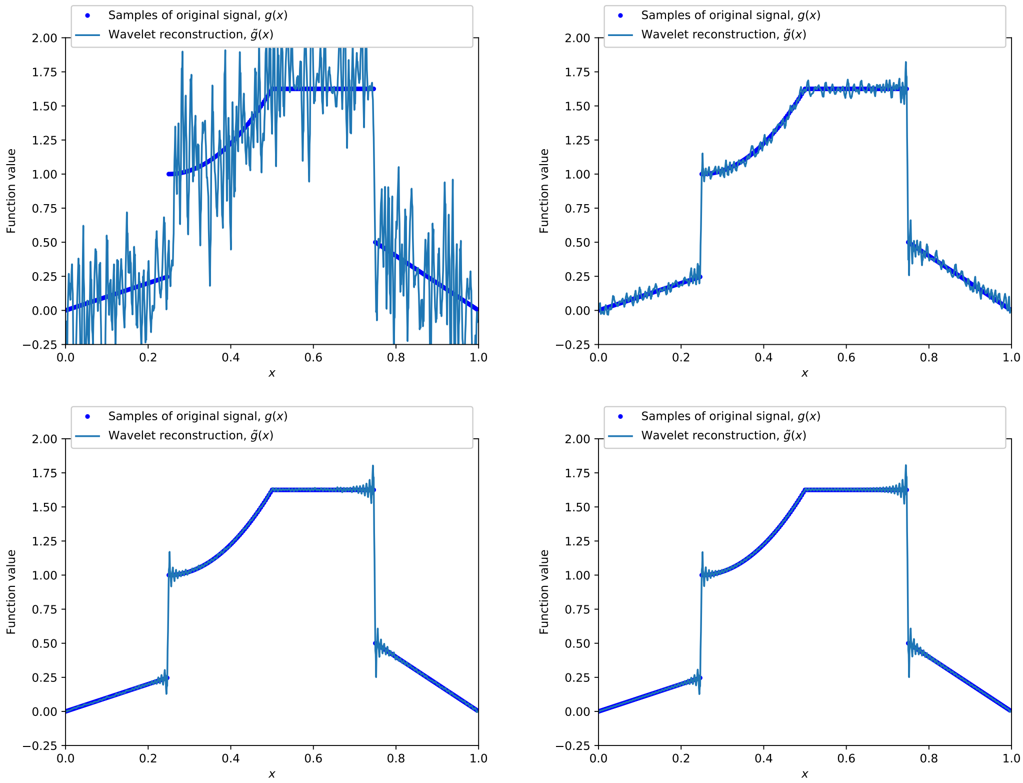


Figure 7.15: Figure of reconstructions of f_2 with white noise added. Top left: Reconstruction where SNR = 1. Top right: Reconstruction where SNR = 10. Bottom left: Reconstruction where SNR = 100. Bottom right: Reconstruction where SNR = 1000.

Table 7.1: This shows the errors of the optimization with noise. The signal to noise ratio is given in the column SNR. SNR = N/A means there is no noise present.

Function	SNR	$\left\|\mathcal{B}\alpha^\star - \ddot{\beta}\right\|$	$\left\|\tilde{f} - f\right\|$
f_2	1	461.2	5.162
	10	4.692	$514.2 \cdot 10^{-3}$
	10^2	$47.49 \cdot 10^{-3}$	$162.9 \cdot 10^{-3}$
	10^3	$1.226 \cdot 10^{-3}$	$156.5 \cdot 10^{-3}$
	N/A	$0.7388 \cdot 10^{-3}$	$156.8 \cdot 10^{-3}$

7.5 Violation of the Subspace Condition

In this section it will be investigated whether the reconstruction fail if the subspace condition is violated. We consider the function f_2, (7.18), and set $\epsilon = 1$. This means that we are not guaranteed stable and quasi-optimal reconstruction, Remark 5.7, but we do have a weighted Fourier frame.

First consider the uniformly sampled signal shown in the left plot of Figure 7.16. Here it is clear that the reconstruction went well and the reconstruction is as in Section 7.2.4.

Now consider a jittered sampling scheme, with the random perturbations in the interval $\left[-\frac{1}{14}, \frac{1}{14}\right]$. This corresponds to $[-\epsilon_0/p, \epsilon_0/p]$, where $p = 2$ and $\epsilon_0 = 1/7$ as in Section 7.3. Five reconstructions with different seeds have been plotted in the right plot of Figure 7.16. It is clear that we are not able to reconstruct the original signal for all seeds. The errors we have in the worst case, i.e. the red graph, are $\|f_2 - \tilde{f}_2\| = 1.844$ and $\|\mathcal{B}\alpha^\star - \beta\| = 7.160$. Both are much larger than the results in Section 7.2.4 and 7.3.

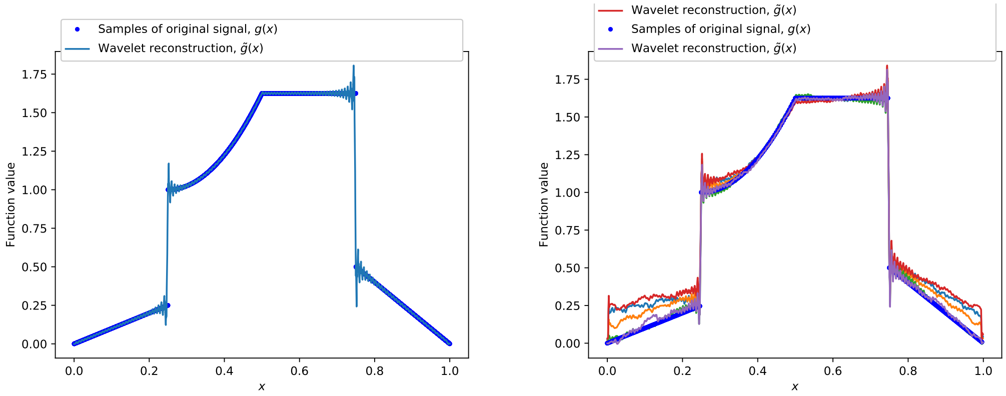


Figure 7.16: The left plot shows f_2 with uniform sampling and $\epsilon = 1$, while the right plot shows f_2 with $\epsilon = 1$ and jittered sampling.

7.6 Higher Order Daubechies Wavelets

In this section we will compare Db2, Db3 and Db4. We choose to let $J = 10$, $\epsilon = \frac{1}{19}$ and $M = 4864$ for all three wavelets and test them on f_2 and f_4. The three wavelets are all able to reconstruct the signals, and the reconstructions are very similar. Figure 7.17 shows zooms of the reconstructions of the two test signals in places where the difference between the reconstructions are visible. The reconstructions for Db3 and Db4 are very similar except at the right edge where the boundary functions are different. Table 7.2 shows the error for the two different signals and the three different reconstruction spaces. We observe that it is slightly harder to optimize for Db3 and Db4 coeffcients. Otherwise the table support what we see in the figure, namely that the reconstructions are very similar.

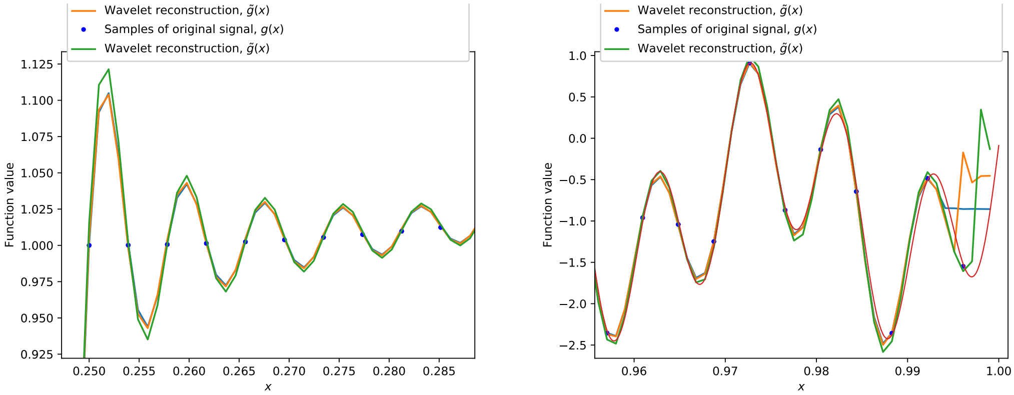


Figure 7.17: The left plot is a zoom of the reconstructions of f_2, the right plot is a zoom of the reconstructions of f_4. Both plots show the samples (dots), the Db4 reconstruction (blue), the Db3 reconstruction (orange), the Db2 reconstruction (green) and in the right the true signal (red).

Table 7.2: The errors for the different experiments in this section. The top half is for f_2 and the bottom half is for f_4.

	$\|\mathcal{B}\alpha^{\star} - \beta\|$	$\|f - \tilde{f}\|$
f_2		
Db4	$0.5742 \cdot 10^{-3}$	0.02090
Db3	$0.4120 \cdot 10^{-3}$	0.02519
Db2	$0.2884 \cdot 10^{-3}$	0.04955
f_4		
Db4	8.655	0.6947
Db3	5.477	1.402
Db2	1.472	0.9395

Chapter 8

Two Dimensions

In this chapter we expand the setup to two dimensions. We will limit this to quadratic signals sampled uniformly on a grid around zero. By quadratic signal we mean a signal with equal number of rows and columns. We will do this as a tensor product, i.e. the procedures described in the previous sections are done on all the rows of the signal and then on all the columns. Recall that the optimization procedure will only be slightly different, Section 6.3.

8.1 Discrete Nonharmonic Fourier Transform

For continuous signals the two dimensional Fourier transform is defined as

$$\hat{f}(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)e^{-2\pi i(xu+yv)}dxdy. \tag{8.1}$$

When discretizing it, Section 4.2.1, we get

$$\hat{f}(\omega_k,\omega_l) = \frac{1}{N}\sum_{n=0}^{N-1}\sum_{\nu=0}^{N-1}\sqrt{\mu_k\mu_l}f\left(\frac{n}{N},\frac{\nu}{N}\right)e^{-2\pi i(\omega_k n+\omega_l \nu)/N}. \tag{8.2}$$

The inverse is

$$f\left(\frac{n}{N},\frac{\nu}{N}\right) = \frac{1}{N}\sum_{\omega_k}\sum_{\omega_l}\sqrt{\mu_k\mu_l}\hat{f}(\omega_k,\omega_l)e^{2\pi i(\omega_k n+\omega_l \nu)/N}. \tag{8.3}$$

Let $\Omega_k := \{\omega_k\}_{k=1}^{M}$ and $\Omega_l := \{\omega_l\}_{l=1}^{M}$. When $\Omega_k = \Omega_l$, (8.2) and (8.3) are the same as using the one dimensional transforms on all rows and then on all columns. This is what we do in the implementation in `DFT.py`.

The complexities for both `DFT_2D` and `IDFT_2D` is $\mathcal{O}(NM^2 + N^2M)$.

8.2 Forward and Adjoint Algorithms

As described in [Gataric and Poon, 2016, Section 6], the two dimensional forward and adjoint operations can be carried out by simply applying the one dimensional algorithm to all rows and then to all columns.

In the two dimensional case, (6.12) becomes

$$\beta_{m_1,m_2} = \sqrt{\mu_{\omega_{m_1}}\mu_{\omega_{m_2}}}\left\langle \sum_{k_1,k_2=0}^{N-1} \alpha_{k_1,k_2}\phi^{[0,1]}_{r,(k_1,k_2)}, e_{m_1}e_{m_2}\right\rangle, \quad m_1,m_2 = 0,\dots,N-1. \tag{8.4}$$

Denote Algorithm 1, applied to the scaling coefficients, by $G : \mathbb{C}^N \to \mathbb{C}^M$.

$$\begin{aligned}\beta_{m_1,m_2} &= \sqrt{\mu_{\omega_{m_1}}}\left\langle \sum_{k_1=0}^{N-1}\left\langle \sqrt{\mu_{\omega_{m_2}}}\sum_{k_2=0}^{N-1}\alpha_{k_1,k_2}\phi^{[0,1]}_{r,k_2}, e_{m_2}\right\rangle \phi^{[0,1]}_{r,k_1}, e_{m_1}\right\rangle \\ &= \sqrt{\mu_{\omega_{m_1}}}\left\langle \sum_{k_1=0}^{N-1}\gamma_{k_1,m_2}\phi^{[0,1]}_{r,k_1}, e_{m_1}\right\rangle = \eta_{m_1,m_2},\end{aligned} \tag{8.5}$$

where, for each $k_1 = 0,\dots,N-1$,

$$(\gamma_{k_1,m_2})_{m_2=1}^{M} = \left(\left\langle \sqrt{\mu_{\omega_{m_2}}}\sum_{k_2=0}^{N-1}\alpha_{k_1,k_2}\phi^{[0,1]}_{r,k_2}, e_{m_2}\right\rangle\right)_{m_2=1}^{M} = G\left((\alpha_{k_1,k_2})_{k_2=0}^{N-1}\right), \tag{8.6}$$

and, for each $m_2 = 1,\dots,M$,

$$(\eta_{m_1,m_2})_{m_1=1}^{M} = G\left((\gamma_{k_1,m_2})_{k_1=0}^{N-1}\right). \tag{8.7}$$

The forward algorithm can be written as

```
Algorithm 3: Fast computation of $\mathcal{B}\alpha$
Input: Scaling coefficients $\alpha \in \mathbb{C}^{N\times N}$.
Output: Fourier coefficients $\beta \in \mathbb{C}^{M\times M}$.
$\tilde{\beta} = \left(G\left((\alpha_{k_1,k_2})_{k_2=0}^{N-1}\right)\right)_{k_1=0}^{N-1} \in \mathbb{C}^{N\times M}$;
$\beta = \left(G\left((\tilde{\beta}_{k_1,m_2})_{k_1=0}^{N-1}\right)\right)_{m_2=1}^{M} \in \mathbb{C}^{M\times M}$;
```

For the two dimensional adjoint algorithm, we first apply Algorithm 2 to all the columns and then to all the rows.

```
Algorithm 4: Fast computation of $\mathcal{B}^*\zeta$
Input: $\zeta \in \mathbb{C}^{M\times M}$.
Output: $\gamma \in \mathbb{C}^{N\times N}$.
$\tilde{\gamma} = \left(G^*\left((\zeta_{m_1,m_2})_{m_1=1}^{M}\right)\right)_{m_2=1}^{M} \in \mathbb{C}^{N\times M}$;
$\gamma = \left(G^*\left((\tilde{\gamma}_{k_1,m_2})_{m_2=1}^{M}\right)\right)_{k_1=0}^{N-1} \in \mathbb{C}^{N\times N}$;
```

The Algorithms 3 and 4 are implemented in the file `FnA2D.py`. Both have the complexity $\mathcal{O}\left(aM^2 + M^2\log\left(\frac{N}{\epsilon}\right)\right)$, [Gataric and Poon, 2016, p. A1092].

8.3 Reconstruction

The idea for the reconstruction is the same as in the previous sections. We will apply the one-dimensional reconstruction to all rows and all columns in turn. There are usually more entries in the reconstruction than there are wavelet coefficients. There are 2^{2J} wavelet coefficients and 2^{2n} reconstruction points, where n is the number of iterations in the cascade algorithm from Section 5.2.1. For each row of wavelet coefficients, we use it to reconstruct 2^{n-J} rows of reconstruction points. Once all rows have been reconstructed, the same is done for the columns.

An implementation of the reconstruction can be found in `ReconFunctions.py`, as the function `Recon2d()`. This function has a complexity of $\mathcal{O}(2^n N)$.

8.4 Examples with Daubechies 2 and Haar

In this section some different two dimensional test signals have been constructed, Fourier transformed, optimized to wavelet coefficients and finally reconstructed in the wavelet basis. We will focus on Daubechies 2 and add examples with Haar for visual comparison. This is the exact same procedure as in Chapter 7, however, other parameters are chosen.

Parameter	Daubechies 2	Haar
J	8	8
N	2^J	2^J
M	896	2^J
ϵ	$\frac{1}{7}$	1
$\mathcal{K}$	$\frac{\|\|f_s\|\|N}{\|\|\hat{f}\|\|2M}$	$2^{J/2}\epsilon$

For both reconstruction bases, we sample uniformly such that $x \in [0,1)^{M\epsilon}$ and $y \in [0,1)^{M\epsilon}$. Recall that J is the scaling parameter for the scaling function, $\phi_{J,k}(\bullet)$; x and y are free variables sampled uniformly and sorted such that $0 = x_1 < x_2 < \ldots < x_{M\epsilon}$ and $0 = y_1 < y_2 < \ldots < y_{M\epsilon}$. M is chosen according to Theorem 5.10. $\mathcal{K}$ is the constant to multiply on $\hat{f}$ and f_s is the signal sampled in $M \times M$ points. Recall that $\mathcal{K}$ is only for comparison of the original signal and the reconstruction, and in cases where a signal is sampled directly in the frequency domain, this factor is without relevance. $\mathcal{K}$ has been determined experimentally. All the test signals can be found in the Python file `TestSignals2D.py`.

8.4.1 Square

The first signal which will be tested is a square defined as

$$f_5(x_i, y_j) = \begin{cases} 1 & \text{if } (x_i, y_j) \in [0.25, 0.75]^2, \\ 0 & \text{otherwise.} \end{cases} \tag{8.8}$$

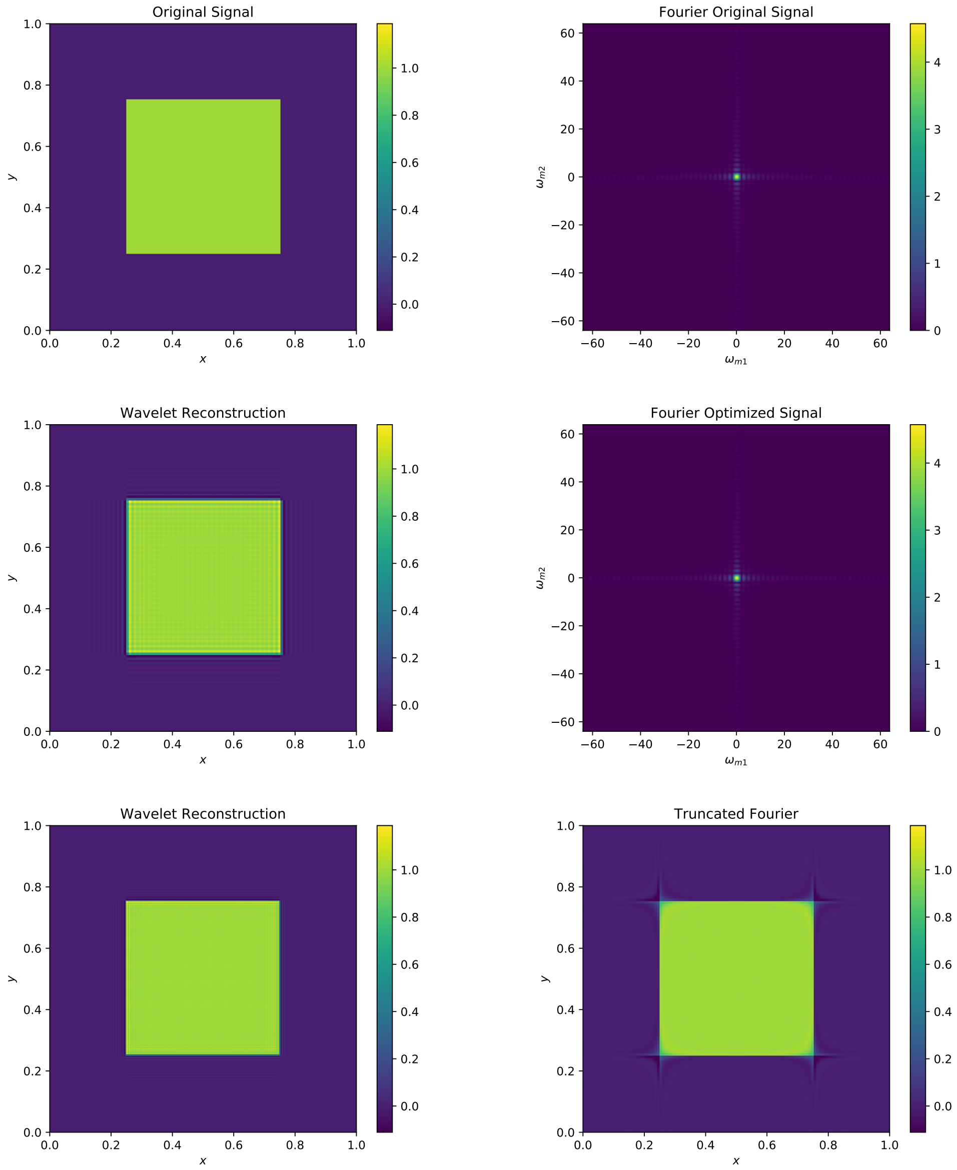


Figure 8.1: Top left: Original time signal. Top right: Original frequency signal. Middle left: Reconstruction with Daubechies 2. Middle right: Optimized frequency signal for Db2. Bottom left: Reconstruction with Haar. Bottom right: Reconstruction from truncated Fourier coefficients.

The function is illustrated in Figure 8.1. It is visually evident from the figure that we are able to reconstruct the signal in the Db2 basis. However, the same artefacts as in the one dimensional case are present, e.g. Gibbs phenomenon, which can especially be seen, at the boundary of the reconstructed square. The optimization method produces a solution where $||\mathcal{B}\alpha^{\star} - \beta|| = 4.636 \cdot 10^{-2}$. We compare the reconstructed signal to the original signal:

$$\|f_5 - \tilde{f}_5\| = 5.818. \tag{8.9}$$

The bottom left plot of Figure 8.1 shows the reconstruction with the Haar basis. Visually it seems to be slightly better, but $||\mathcal{B}\alpha^\star - \beta|| = 5.267 \cdot 10^{-2}$ and $\|f_5 - \tilde{f}_5\| = 11.94$. The reconstruction error is not truly comparable to the one for Db2 as they do not have the same number of sampling points in either time or frequency. The bottom right plot in Figure 8.1 shows the reconstruction from truncated Fourier coefficients made in a similar fashion to the ones from Chapter 7. It is visually evident that the wavelet reconstruction is a lot better, so, for the purpose of compression, generalized sampling works well for this kind of signal.

8.4.2 Wave

The next signal, which is tested, is a wave originating in $(0,0)$:

$$f_6(x_i, y_j) = \begin{cases} \cos\left(2\pi 10\sqrt{x_i^2 + y_j^2}\right) & \text{if } (x_i, y_j) \in [0,1)^2, \\ 0 & \text{otherwise.} \end{cases} \tag{8.10}$$

The function is illustrated in Figure 8.2. It is visually evident that we are able to reconstruct the signal in the Db2 basis. However, similar artefacts as in the one dimensional case are visible, e.g. the boundary wavelets are scaled wrong. The optimization method produces a solution where $||\mathcal{B}\alpha^\star - \beta|| = 2.177 \cdot 10^2$. We compare the reconstructed signal to the original signal:

$$\|f_6 - \tilde{f}_6\| = 23.06. \tag{8.11}$$

The bottom left plot is the reconstruction in the Haar basis, here $||\mathcal{B}\alpha^\star - \beta|| = 6.109 \cdot 10^{-2}$ and $\|f_6 - \tilde{f}_6\| = 15.47$. This look very similar to the other plots in the figure, but has slightly lower values at the peaks of the oscillations. Notice that it does not have the boundary effects which can be seen for Db2. The bottom right plot in Figure 8.2 is the reconstruction from truncated Fourier coefficients. This shows that this signal is well represented in the Fourier frame. The error in this reconstructione is only 4.638.

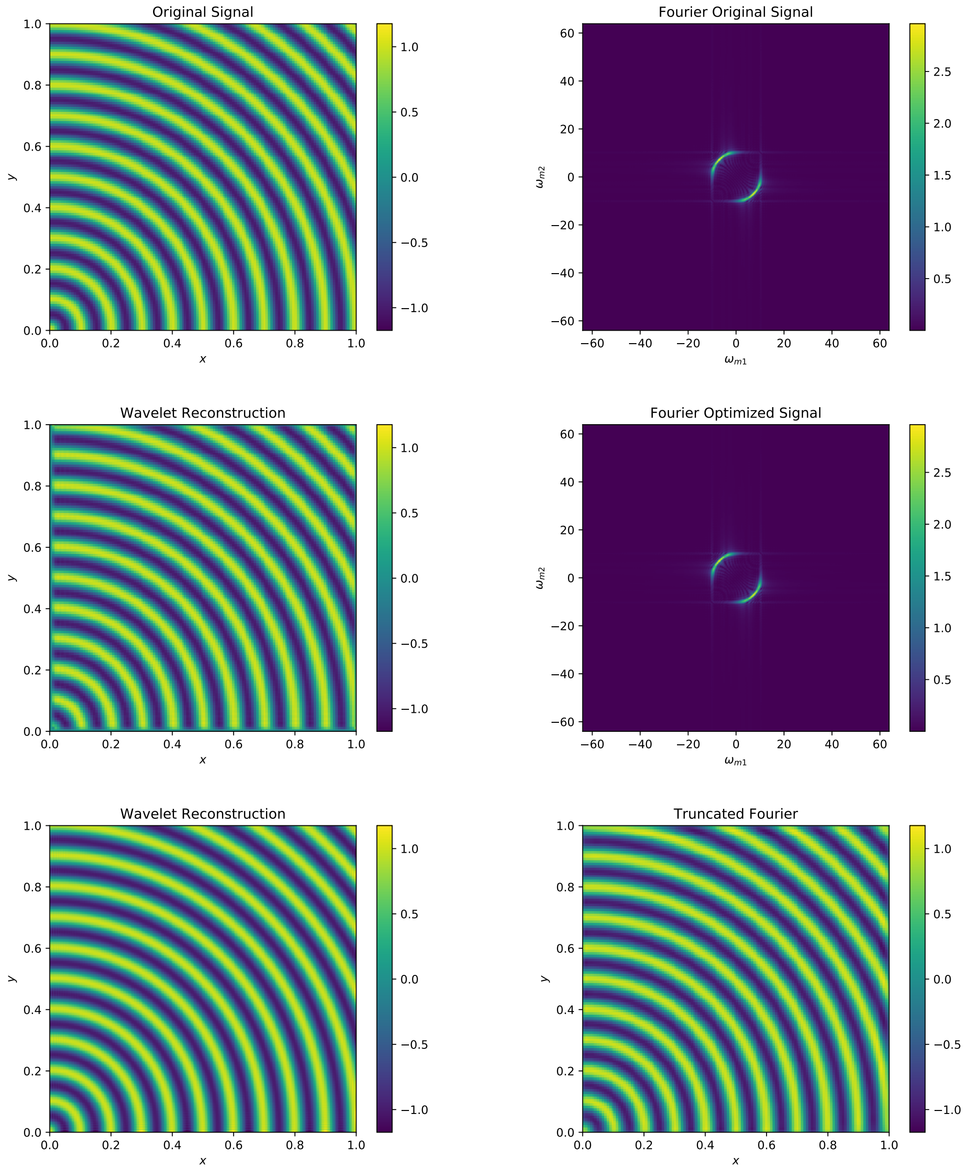


Figure 8.2: Top left: Original time signal. Top right: Original frequency signal. Middle left: Reconstruction with Db2. Middle right: Optimized frequency signal for Db2. Bottom left: Reconstruction with Haar. Bottom right: The reconstruction from truncated Fourier coefficients.

8.4.3 Circle

The next signal, which is tested, is a circle:

$$f_7(x_i, y_j) = \begin{cases} 1 & \text{if } \left(x_i - \frac{1}{2}\right)^2 + \left(y_j - \frac{1}{2}\right)^2 = \left(\frac{1}{4}\right)^2, \\ 0 & \text{otherwise.} \end{cases} \tag{8.12}$$

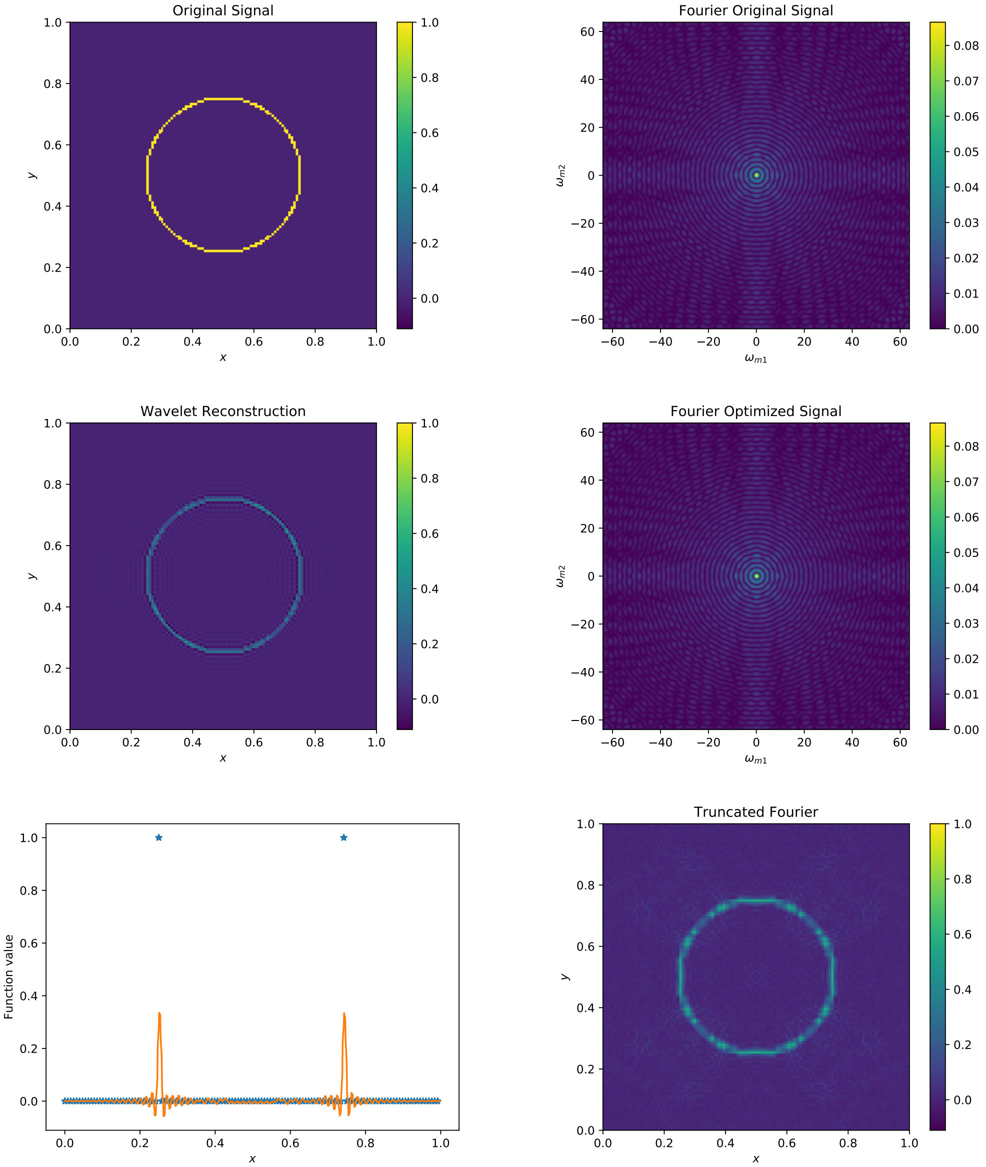


Figure 8.3: Top left: Original time signal. Top right: Original frequency signal. Middle left: Reconstruction in the Db2 basis. Middle right: Optimized frequency signal. Bottom left: The middle row of the original signal (stars) and the reconstruction in the Db2 basis (line). Bottom right: The reconstruction from truncated Fourier coefficients.

The function is illustrated in Figure 8.3. It is visually evident from the figure that the reconstruction in the wavelet basis is not as good as in the other two examples. The reconstructed circle is not close in amplitude to the original circle, this is illustrated in the bottom left plot of the figure. Nevertheless, we are able to see that it is indeed a circle. The optimization method produces a solution where

$||\mathcal{B}\alpha^\star - \beta|| = 2.224 \cdot 10^{-2}$. We compare the reconstructed signal to the original signal:

$$\|f_7 - \tilde{f}_7\| = 11.07. \tag{8.13}$$

The bottom right plot in Figure 8.3 is the reconstruction from truncated Fourier coeffcients. It appears to be blurry which attests to the fact that the Fourier frame is not well suited to handle discontinuities.

8.4.4 Shepp-Logan Phantom

The next signal, we will investigate, is the Shepp-Logan phantom which is a standard test image for image reconstruction algorithms. It was originally designed to serve as a model for the human head. Our method is designed for MRI, so test functions designed to look like body parts are of particular interest. The Shepp-Logan phantom is defined as a sum of 10 ellipses. The specific way to construct the figure can be found in [Shepp and Logan, 1974] and is illustrated in Figure 8.4. It is visually evident that we are able to extract the features, i.e. the ellipses, in the figure. However, we also see a lot of Gibbs ringing which is consistent with earlier examples. The optimization method produces a solution where $||\mathcal{B}\alpha^\star - \beta|| = 1.374$. We want to compare this example with the other examples. In order to do this we divide the norm of the difference by the largest value of the original signal, which is 2:

$$\|\mathrm{SL} - \tilde{\mathrm{SL}}\|/2 = 9.191. \tag{8.14}$$

The bottom left plot shows the reconstruction in the Haar basis, here $||\mathcal{B}\alpha^\star - \beta|| = 4.962 \cdot 10^{-1}$ and $\|\mathrm{SL} - \tilde{\mathrm{SL}}\|/2 = 15.58$. It is visually evident that the artefacts around the large discontinuities are less prominent for the Haar basis than for the Db2 basis.

The bottom right plot in Figure 8.4 shows the reconstruction from truncated Fourier coefficients. It is visually evident that this reconstruction is worse than the wavelet reconstruction due to the amount of red and blue in the plot. Furthermore, this reconstruction have wavelike behaviour in the green region.

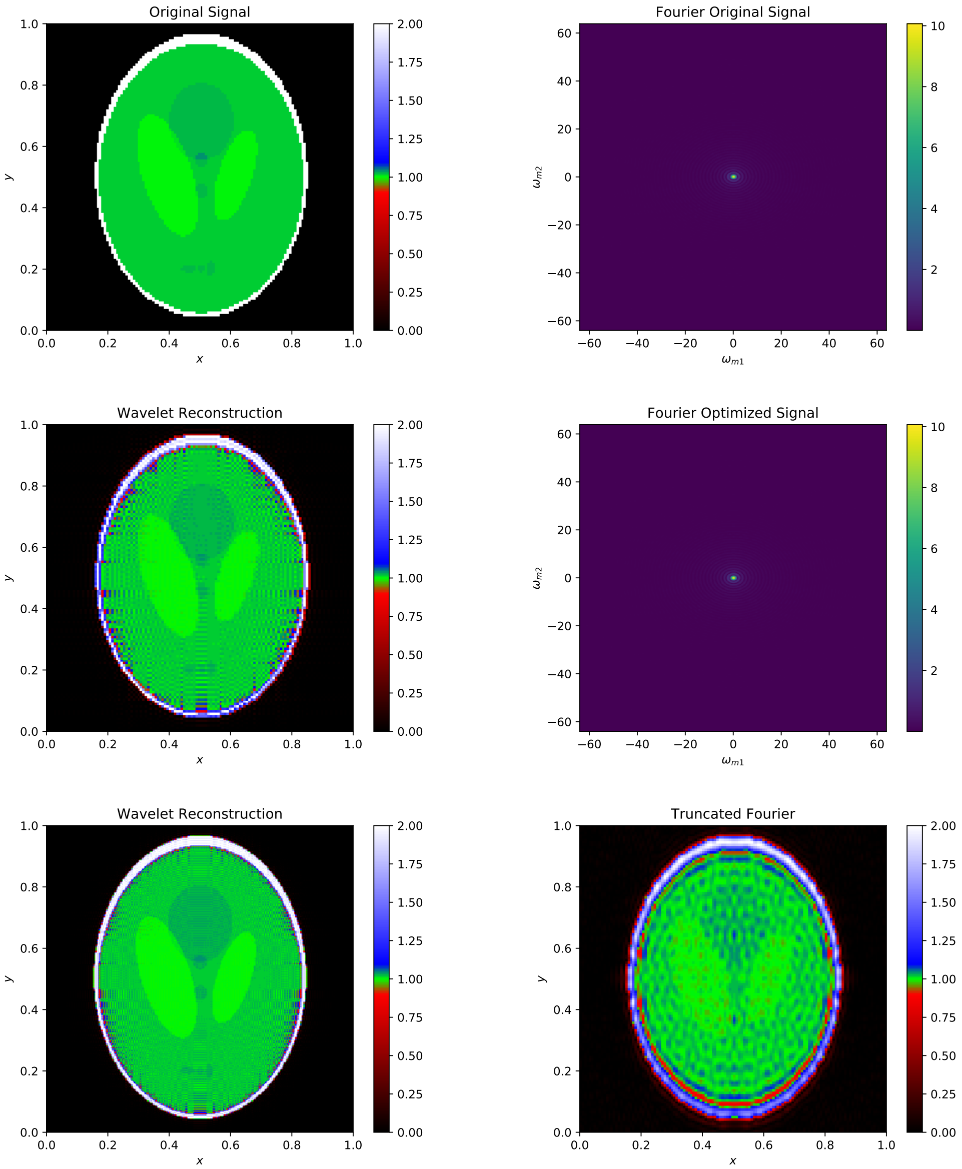


Figure 8.4: Top left: Original time signal. Top right: Original frequency signal. Middle left: Reconstruction with Db2. Middle right: Optimized frequency signal. Bottom left: Reconstruction with Haar. Bottom right: The reconstruction from truncated Fourier coefficients.

8.4.5 Guerquin-Kern Brain Phantom

The second-to-last example, we will work on, is a more realistic brain image than the Shepp-Logan phantom. It can be seen in Figure 8.5, or Figure A.1 for a bigger version. It was introduced in [Guerquin-Kern et al., 2012] and will be denoted Guerquin-Kern phantom. The way we use it is by downloading the .svg picture from http://bigwww.epfl.ch/algorithms/mriphantom/ and convert it to a .png picture in the size needed, here we chose 512 × 512 pixels.

This example has some other parameters than the previous examples:

$$\begin{aligned} \epsilon &= \frac{1}{7}, \\ M &= 3584, \\ J &= 10, \\ N &= 2^J, \\ x &\in [0,1)^{M\epsilon}, \\ y &\in [0,1)^{M\epsilon}. \end{aligned} \tag{8.15}$$

Here we had a fixed number of samples, $M\epsilon = 512$, and found the remaining parameters based on this and Table 5.3. We could have chosen any J between 3 and 10 but chose to take $J = 10$. We do this in order for this example to be as close to the setup of the other examples as possible regarding the stable sampling rate. We remind the reader that we pretend to sample directly in the frequency domain.

The optimization method produces a solution where $||\mathcal{B}\alpha^* - \beta|| = 3.525 \cdot 10^3$. We compare the reconstructed signal to the original signal:

$$\|\mathrm{GK} - \tilde{\mathrm{GK}}\| = 2.331 \cdot 10^3. \tag{8.16}$$

The reconstructed picture can be found in Figure 8.5, or in a larger version in Figure A.2. In order to compare these errors to those in previous sections, we need to take the magnitude of the signal into account. The largest value in the original signal is 255, so the reconstruction error should be divided by this:

$$\|\mathrm{GK} - \tilde{\mathrm{GK}}\|/255 = 9.141. \tag{8.17}$$

It is visually evident that the phantom can be recognized in the wavelet reconstruction. By looking closely, it seems like all the features of the picture are preserved, e.g. we can see the discontinuities between regions. If we compare this to the reconstruction from truncated Fourier coefficients, Figure 8.5 and Figure A.3, we see that the reconstruction from truncated Fourier coefficients is more blurry and the discontinuities are not as sharp as the wavelet reconstruction. Furthermore, the truncated Fourier reconstruction is also more granular in the regions where it is supposed to be flat.

In Figure 8.6 a zoom of the Guerquin-Kern phantom has been plotted. Here it is easily seen that the wavelet reconstruction is very close to the original, while the reconstruction from truncated Fourier coefficients smooths away the edges between regions. When a doctor looks at an image of a brain, details like this can be the difference between life and death for the patient. Therefore, it is very important to have a good reconstruction in this kind of situation.

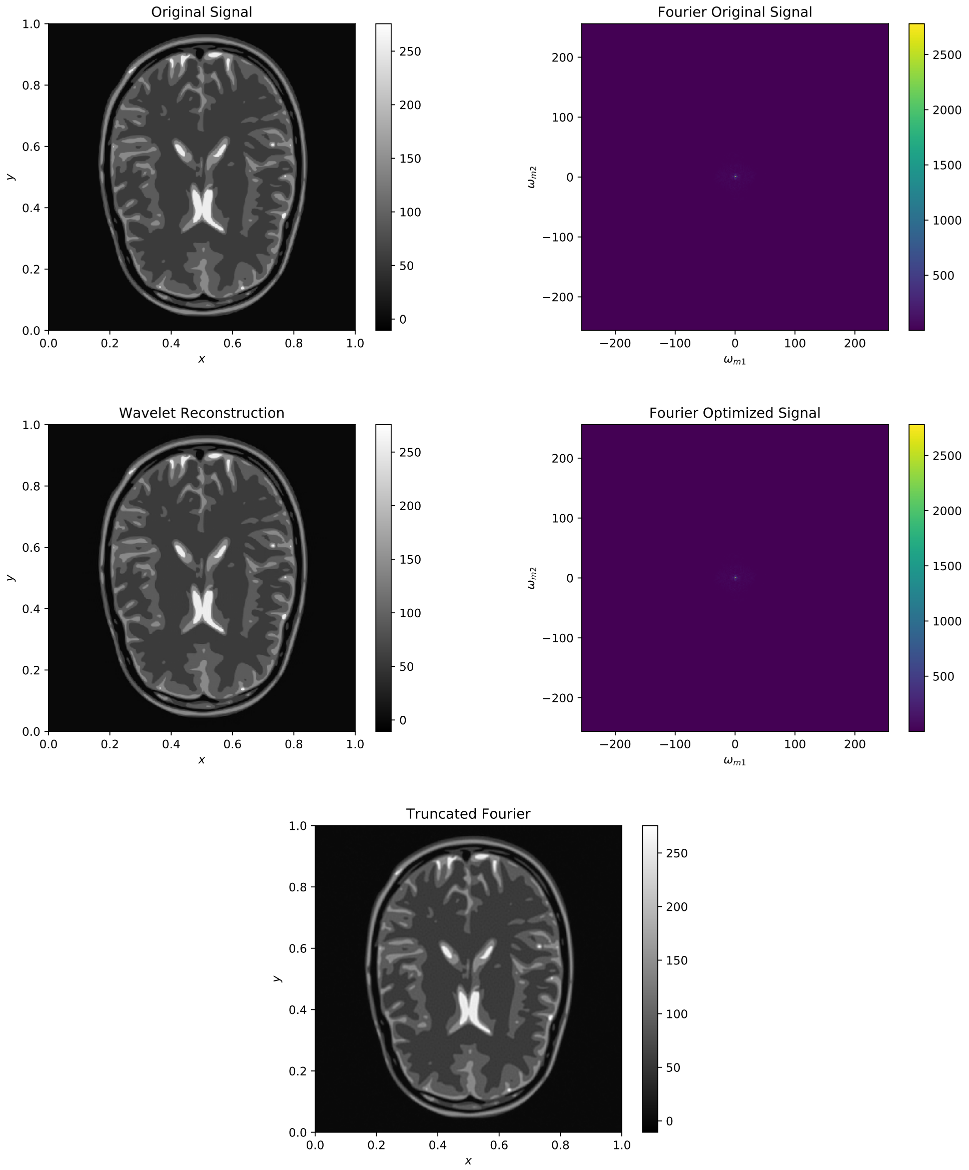


Figure 8.5: Top left: Original time signal. Top right: Original frequency signal. Middle left: Reconstruction in the Db2 basis. Middle right: Optimized frequency signal. Bottom: The reconstruction from truncated Fourier coefficients.

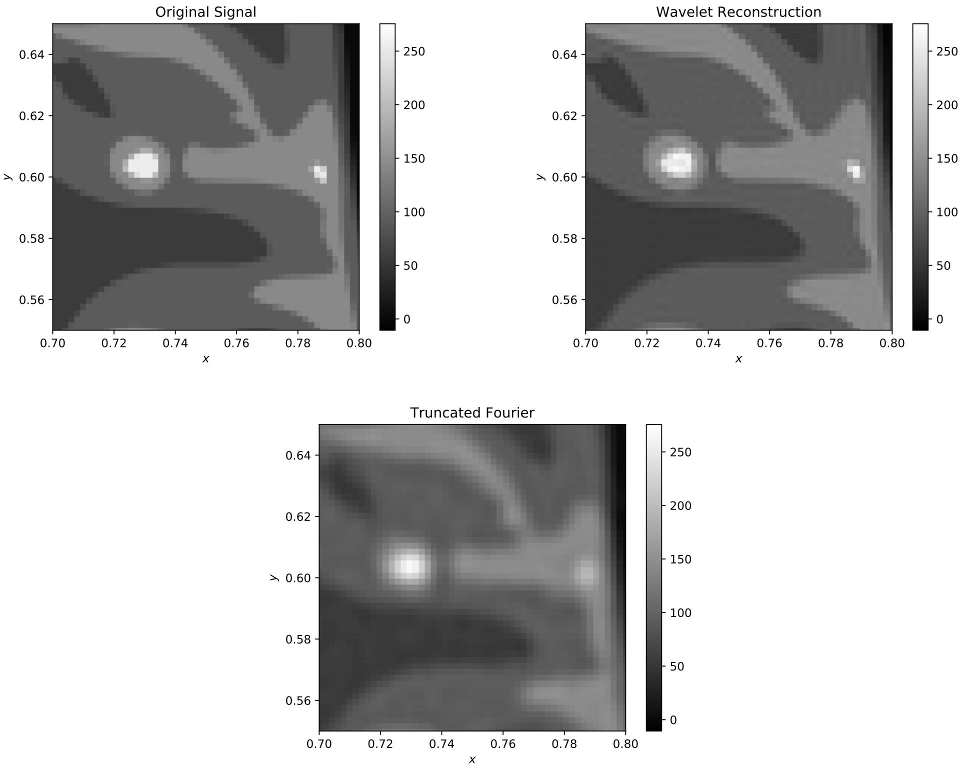


Figure 8.6: A zoom of the Guerquin-Kern phantom. Top left: The original time signal. Top right: Reconstruction in the Db2 basis. Bottom: The reconstruction from truncated Fourier coefficients.

8.4.6 Cameraman

As the final example, we have chosen another standard test image, namely the cameraman, which can be seen in Figure 8.7, or Figure A.4 for a larger image. Unfortunately, we have not been able to find the factor to scale the Fourer coefficients with in order to get the same colorbar in both the original and reconstruction of the picture; it requires the original signal to be sampled arbitrary close which it cannot. This means we have to evaluate the reconstruction visually and without error measures. The parameters of this example are the same as in (8.15).

The most obvious difference, between the two pictures in Figure 8.7, is that the wavelet reconstruction seems to be slightly brighter. This is mainly due to the change in the colorbar's range and helps us see the details in the coat more easily. Furthermore, it seems like the features of the picture are preserved, e.g. we are able to see all the buildings and their details in the reconstruction. The final observation is that the boundary functions still are not scaled correctly, which might be easier to see in the larger versions of the picture in A.5. If we compare this to the reconstruction from truncated Fourier coefficients, it is clear that the wavelet reconstruction has more details as well as sharper edges.

The optimization method produces a solution where $||\mathcal{B}\alpha^{\star} - \beta|| = 2.790$.

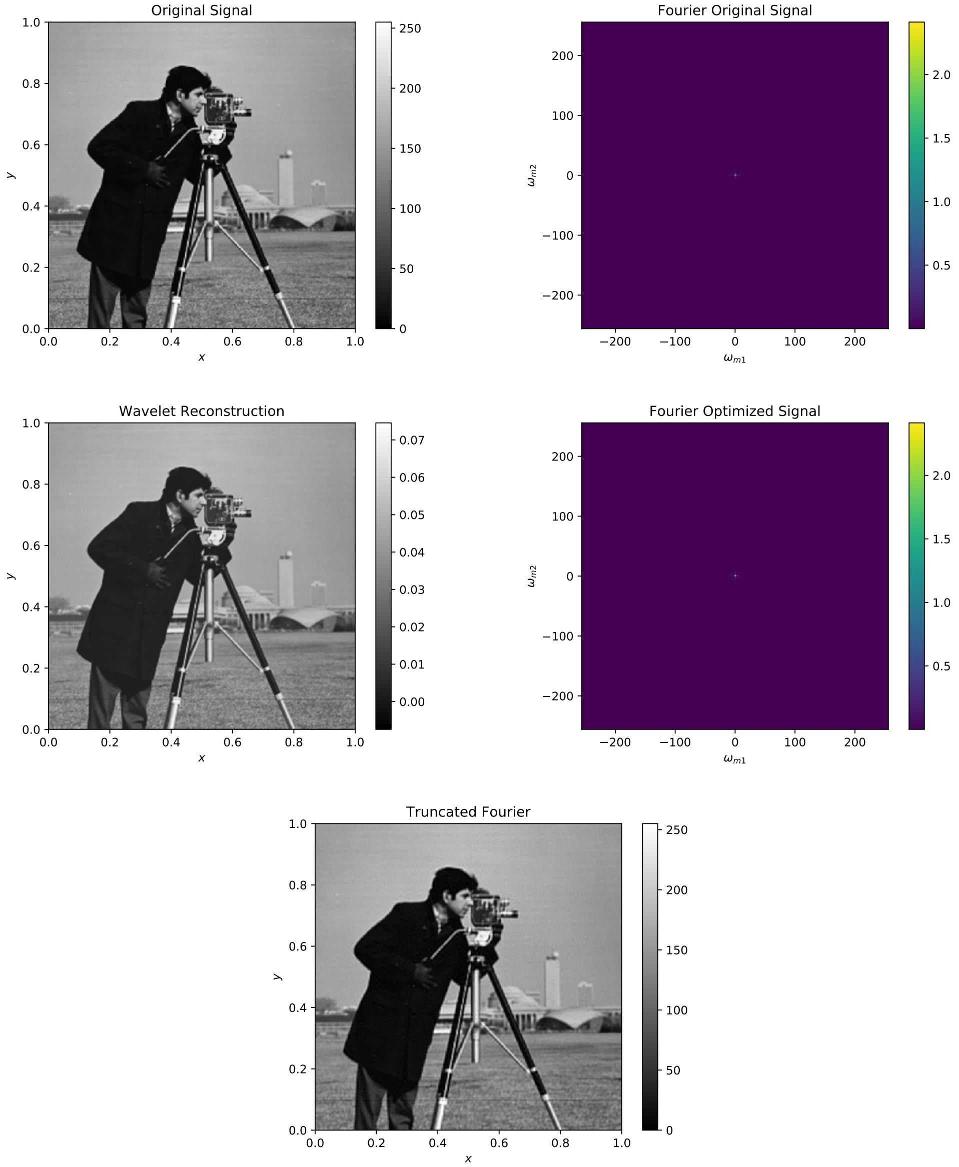


Figure 8.7: Top left: Original time signal. Top right: Original frequency signal. Middle left: Reconstruction in the Db2 basis. Middle right: Optimized frequency signal. Bottom: The reconstruction from truncated Fourier coefficients.

8.4.7 Summary of Results

We have shown that generalized sampling works in two dimensions. We see some of the same effects as in the one dimensional case, e.g. problems with rapid changes in the signal and Gibbs phenomenon. For most inputs, generalized sampling is better for compression than reconstructing from truncated Fourier coefficients. From the last examples it is visually evident that some details, which are lost when the Fourier coefficients are truncated, are preserved when using generalized sampling.

Chapter 9

Discussion and Conclusion

9.1 Discussion

We have made several restrictions to the problem considered in order to focus our work on what we find essential. In this section we will discuss these restrictions and possible extensions of the project. The possible extensions includes finding better optimization algorithms, alternative sampling schemes, other reconstruction spaces and improvement of the algorithms in terms of complexity.

9.1.1 Optimiziation Methods

In Section 6.3 we briefly discussed our choice of optimization method and the problems it poses. The optimization method used is based on [Beck and Teboulle, 2009] which were made for real-valued α, $\mathcal{B}$ and β. The implementation does seem to work with complex values, but we have not been able to confirm its efficiency. We chose this implementation since it accepts Python functions as input and it was designed for linear least squares problems. Furthermore, it converged to a solution in contrast to some of the other methods tried.

We believe that there is a better optimization method for this specific setup. However, we have not been able to find or implement such a method within the time constraints of the project.

9.1.2 Nonuniform Sampling

A minor experiment with nonuniform sampling was performed in Section 7.3. It showed that the setup described, in the thesis and the code supporting it, does work for nonuniformly sampled inputs. We have chosen to limit our work in this area to this simple test for several reasons. A jittered sampling scheme is simple and often used as an example of random sampling, [Adcock et al., 2017],[Adcock et al., 2015] and [Adcock et al., 2014a]. It is also common in practise where physics dictates where it is possible to sample or we do not know the exact sampling scheme beforehand but are able to correct it. Additionally, a jittered sampling scheme is easy to control;

even though it is random, we know the sampling density and the order of the samples beforehand. This is an advantage as the sampling scheme must fulfil the theoretical criteria from Theorem 4.16. Other sampling schemes are also possible as long as they fulfil the theoretical criteria. However, research into what constitute a good sampling scheme for different inputs is outside the scope of this project. Some work has been done in [Jacobsen et al., 2016a].

9.1.3 Other Frames

Generalized sampling is not limited to the Fourier to wavelet case. It is theoretically possible to use any frame as a sampling space, although no examples has been found using any other frames than Fourier. However, for reconstruction spaces several different types have been used. For an example of piecewise polynomial reconstructions from Fourier coefficients, see [Adcock and Hansen, 2011] and [Adcock and Hansen, 2015]. For the more general approach of an arbitrary basis for reconstruction, see [Adcock and Hansen, 2012].

An obvious extension of the work on this project could be to compose a reconstruction frame of two different bases which both spans $L^2([0,1])$. Specifically, it could be a composition of the Haar and another, higher order, Daubechies wavelet. The Haar wavelet is well adapted for reconstructing discontinuities, while the higher order Daubechies wavelets are better at reconstructing polynomials. We therefore conjecture that this frame may be able to reconstruct signals better, or at least as good, than either of them would separately.

9.1.4 Two Dimensions

Our investigations into two dimensional generalized sampling has been limited to uniformly sampled, square inputs. We will discuss these two restrictions separately.

We have chosen to restrict our work to square inputs, i.e. inputs where the sampling scheme is a tensor product of a one dimensional sampling scheme with itself. This results in a square matrix of reconstruction coefficients. The two dimensional discrete nonharmonic Fourier transform, that we derived in Section 8.1, can only handle these kind of sampling schemes. We trust that it is possible to expand the derived transform such that two dimensional schemes, which are tensor products of two different one dimensional schemes, can be used, but this has not been tested. Furthermore, the two dimensional forward and adjoint algorithms presented in [Gataric and Poon, 2016, Section 6] are only made for square inputs.

The restriction to uniform sampling is mainly due to the additional complexity of the forward and adjoint algorithms in the general case. The algorithms have been derived in the general case in [Gataric and Poon, 2016, Section 5] and then simplified to the version that we use in Algorithm 3 and 4. The general algorithms have higher computational complexity and require additional theoretical work.

9.1.5 Reconstruction Quality and Compression

In generalized sampling, two factors determine how well we can reconstruct and how much we can compress. When we have chosen a sampling space and a reconstruction space the stable sampling/reconstruction rates are fixed. When keeping the spaces fixed, the type of signal and the ratio between the number coefficients determines the reconstruction error and the compression ratio.

For Daubechies wavelets of higher order than Haar, there will always be some level of compression due to the stable reconstruction rate, or equivalently the stable sampling rate, which is shown in Table 5.3. We have shown that, for some types of signals, this way of compression is more efficient than truncating the Fourier coefficients. So the usefulness of generalized sampling depends on the input signal. As mentioned in Chapter 2, the signals, for which generalized sampling is useful, are the ones which are ill represented in the sampling space and well represented in the reconstruction space. An additional advantage of generalized sampling for compression is that it is not always necessary to decompress in order to analyse the signal and, if it is, decompression is simple to do.

The stable rates are the limits for how close N and M can be to each other. If we fix N and choose M to be larger than the stable sampling rate, we get a better reconstruction, but we also have to sample more. On the other hand, if we fix M and choose N to be smaller than the stable reconstruction rate, we compress the signal more, but we also make a larger error.

9.1.6 Time Complexity of Algorithms

Throughout the chapters we have observed certain time complexities for different algorithms, they are summarized in Table 9.1.

Table 9.1: Table of time complexities in $\mathcal{O}$ notation for different algorithms.

	1D complexity	2D complexity
DFT	MN	$NM^2 + N^2M$
IDFT	MN	$NM^2 + N^2M$
Forward operation	$aM + M\log\left(\frac{N}{\epsilon}\right)$	$aM^2 + M^2\log\left(\frac{N}{\epsilon}\right)$
Adjoint operation	$aM + M\log\left(\frac{N}{\epsilon}\right)$	$aM^2 + M^2\log\left(\frac{N}{\epsilon}\right)$
Wavelet reconstruction	N	2^nN

The discrete Fourier transform and reconstruction algorithm only has to run once for every signal tested. Therefore, it is clear that it is the forward and adjoint operations, Algorithm 1-4, which are the bottlenecks; they have to run several times per iteration in the optimization method. However, in the general case, of solving a linear system in one dimension, it requires $\mathcal{O}(MN)$ operations, but here we are able to do it in $\mathcal{O}(M\log N)$ operations [Gataric and Poon, 2016, p. A1077]. Furthermore, with the forward and adjoint algorithms at hand, we do not have to store the change of basis matrix explicitly.

Returning to the DFT, we have implemented our own version in order to get non-integer frequencies, which is not standard in the fast Fourier transform (FFT) algorithms known in Python, e.g. `numpy.fft.fft()` and `scipy.fft()`. Furthermore, our implementation is adapted to nonuniform sampling schemes which again is not a standard thing in Python. We are aware there are implementations of the so-called nonuniform fast Fourier transform, NFFT. NFFT is a different approach to the nonuniform discrete Fourier transform than the one in Section 4.2.1. NFFT is based on regridding or interpolating, where samples on a uniform grid is found, based on the nonuniform samples with as small an error as possible. Once the samples are on an uniform grid the regular FFT algorithm can be used, [Dutt and Rokhlin, 1995] and [Gelb and Song, 2014]. This method does not allow for sampling in a frame which is essential for our purpose.

9.2 Conclusion

We have investigated generalized sampling and its application in image reconstruction and compression with focus on a Fourier frame as sampling space and a wavelet basis as reconstruction space. We have described theory primarily based on work by Ben Adcock and Anders C. Hansen. Their results include an upper bound for the reconstruction constant and an easily calculated stable sampling rate.

The theory on Fourier frames is extensive. However, certain areas are not covered by standard theory. So, in Section 4.2.1, we extended existing theory to the discrete nonharmonic Fourier transform, which can also be used for nonuniform sampling schemes. This theory is necessary when the sampling scheme has a density lower that one, which is needed in the case for Daubechies wavelets of order two or higher. Additionally, we have chosen an unconventional approach to boundary correction of wavelet bases. We have pieced together theory from multiple sources in creating explicit boundary functions both in the time domain and the frequency domain.

One of the key aspects of this project is the application of generalized sampling. Therefore, a major part of the work has gone towards the implementation of relevant algorithms. We have chosen to use Python as it is a widely used, open source programming language which has not previously been used for generalized sampling. The central algorithms are based on work by Milana Gataric and Clarice Poon. Other implemented algorithms include the discrete nonharmonic Fourier transform, the boundary wavelets in the time- and frequency domains and a transform from Daubechies wavelet coefficients to time.

Through a series of examples, we have shown that our implementation of generalized sampling is beneficial for many different types of input. It is of great importance to have a sufficiently low sampling density and a sufficiently high number of samples, otherwise we can not guarantee that the method works. These bounds have been calculated for several different reconstruction bases, all in the class of Daubechies wavelets. Additionally, we have found that the method works for square two dimensionally inputs, nonuniformly sampled inputs and noisy inputs. It is an advantage to have the input signal sampled uniformly and with as little noise as possible.

For most of the tested signals, it is advantageous to use generalized sampling for compression and reconstruction rather than compressing and reconstructing in the Fourier frame. There is nothing to gain from using generalized sampling if a signal is well represented in the sampling space. However, the method can still be used.

Everything considered, we succeeded in gaining a deeper understanding of generalized sampling, implement some of the methods in Python and had success in making examples which gives a more intuitive understanding of the subject. Furthermore, we showed that using generalized sampling, to transform samples in the Fourier frame to coefficients in a wavelet basis, is a very efficient way to compress a variety of different signals.

Bibliography

Ben Adcock and Anders C. Hansen. Stable reconstructions in hilbert spaces and the resolution of the gibbs phenomenon. *Applied and Computational Harmonic Analysis*, 32(3):357–388, 2011.

Ben Adcock and Anders C. Hansen. A generalized sampling theorem for stable reconstruction in arbitrary bases. *Journal of Fourier Analysis and Applications*, 2012.

Ben Adcock and Anders C. Hansen. Generalized sampling and the stable and accurate reconstruction of piecewise analytic functions from their fourier coefficients. *Mathematics of Computation*, 84(291):237–270, 2015.

Ben Adcock, Anders C. Hansen, and Clarice Poon. Beyond consistent reconstructions: optimality and sharp bounds for generalized sampling, and application to the uniform resampling problem. *SIAM Journal on Mathematical Analysis*, 45(5): 3132–3167, 2013.

Ben Adcock, Milana Gataric, and Anders C. Hansen. On stable reconstructions from nonuniform fourier measurements. *SIAM Journal on Imaging Sciences*, 7(3): 1690–1723, 2014a. doi: 10.1137/130943431.

Ben Adcock, Anders C. Hansen, and Clarice Poon. On optimal wavelet reconstructions from fourier samples: linearity and universality of the stable sampling rate. *Applied and Computational Harmonic Analysis*, 36(3):387–415, 2014b.

Ben Adcock, Milana Gataric, and Anders C. Hansen. *Recovering Piecewise Smooth Functions from Nonuniform Fourier Measurements*, pages 117–125. Springer International Publishing, Cham, 2015. ISBN 978-3-319-19800-2. doi: 10.1007/978-3-319-19800-2_8. URL https://doi.org/10.1007/978-3-319-19800-2_8.

Ben Adcock, Milana Gataric, and Anders C. Hansen. Weighted frames of exponentials and stable recovery of multidimensional functions from nonuniform fourier samples. *Applied and Computational Harmonic Analysis*, 42(3):508 – 535, 2017. ISSN 1063-5203. doi: https://doi.org/10.1016/j.acha.2015.09.006. URL http://www.sciencedirect.com/science/article/pii/S1063520315001384.

Andreas Antoniou and Wu-Sheng Lu. *Practical Optimization: Algorithms and Engineering Applications.* Springer, 1. edition, 2007.

Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

Ole Christensen. *Frames and Bases, an introductory course.* Birkhäuser, 1. st. edition, 2008.

Ole Christensen et al. *An introduction to frames and Riesz bases.* Springer, 2016.

Albert Cohen, Ingrid Daubechies, and Pierre Vial. Wavelets on the interval and fast wavelet transforms. *Applied and computational harmonic analysis*, 1(1):54–81, 1993.

Ingrid Daubechies. *Ten lectures on wavelets*, volume 61. Siam, 1992.

Richard J. Duffin and Albert C. Schaeffer. A class of nonharmonic fourier series. *Transactions of the American Mathematical Society*, 72(2):341–366, 1952.

Alok Dutt and Vladimir Rokhlin. Fast fourier transforms for nonequispaced data, ii. *Applied and Computational Harmonic Analysis*, 2(1):85–100, 1995.

Gerald B. Folland. *Fourier Analysis and Its Applications.* American Mathematical Society, 1. edition, 1992.

Milana Gataric and Clarice Poon. A practical guide to the recovery of wavelet coefficients from fourier measurements. *SIAM Journal on Scientific Computing*, 38(2):A1075–A1099, 2016.

Anne Gelb and Guohui Song. A frame theoretic approach to the nonuniform fast fourier transform. *SIAM Journal on Numerical Analysis*, 52(3):1222–1242, 2014.

Matthieu Guerquin-Kern, L Lejeune, Klaas Paul Pruessmann, and Michael Unser. Realistic analytical phantoms for parallel magnetic resonance imaging. *IEEE Transactions on Medical Imaging*, 31(3):626–636, 2012.

Robert Dahl Jacobsen, Jesper Møller, Morten Nielsen, and Morten Grud Rasmussen. Investigations of the effects of random sampling patterns on the stability of generalized sampling. *arXiv preprint arXiv:1607.04424*, 2016a.

Robert Dahl Jacobsen, Morten Nielsen, and Morten Grud Rasmussen. Generalized sampling in julia. *arXiv preprint arXiv:1607.04091*, 2016b.

Björn Jawerth and Wim Sweldens. An overview of wavelet based multiresolution analyses. *SIAM review*, 36(3):377–412, 1994.

BM Kessler, GL Payne, and WN Polyzou. Wavelet notes. *arXiv preprint nucl-th/0305025*, 2003.

Erwin Kreyszig. *Introductory functional analysis with applications*, volume 1. wiley New York, 1978.

Yves Nievergelt. *Wavelets made easy*, volume 174. Springer, 1999.

Jorge Nocedal and Stephen J. Wright. *Numerical Optimization.* Springer, 1. edition, 2006.

Alan V. Oppenheim and Ronald W. Schafer. *Discrete-Time Signal Processing.* Pearson, 3rd edition, 2010.

Michael Pedersen. *Functional analysis in applied mathematics and engineering*, volume 31. CRC press, 1999.

David K Ruch and Patrick J Van Fleet. *Wavelet theory: an elementary approach with applications.* John Wiley & Sons, 2011.

Lawrence A. Shepp and Benjamin F. Logan. The fourier reconstruction of a head section. *IEEE Transactions on nuclear science*, 21(3):21–43, 1974.

Martin Vetterli, Jelena Kovacevic, and Vivek K. Goyal. *Foundations of signal processing.* Cambridge University Press, 1. st. edition, 2014.

Appendix A

Appendix

In this appendix, are some larger versions of pictures from Chapter 8.

A.1 Guerquin-Kern Brain Phantom

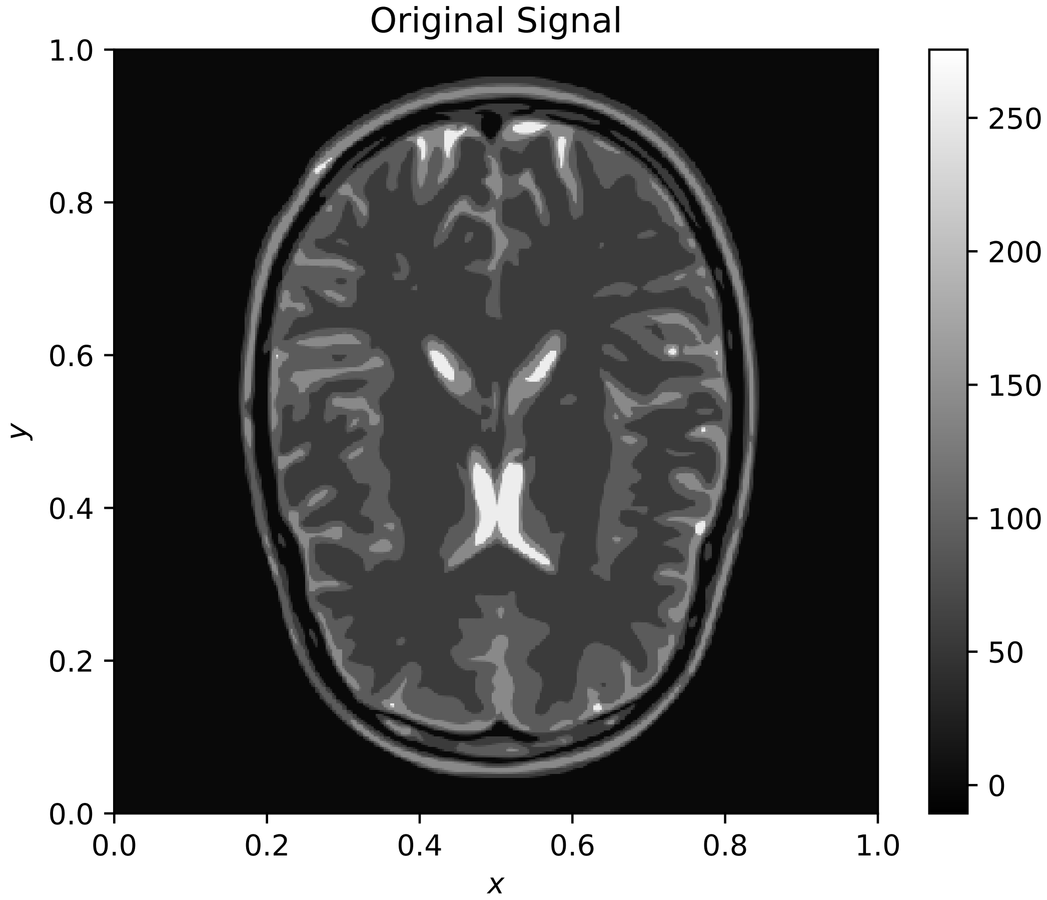


Figure A.1: The original Guerquin-Kern phantom.

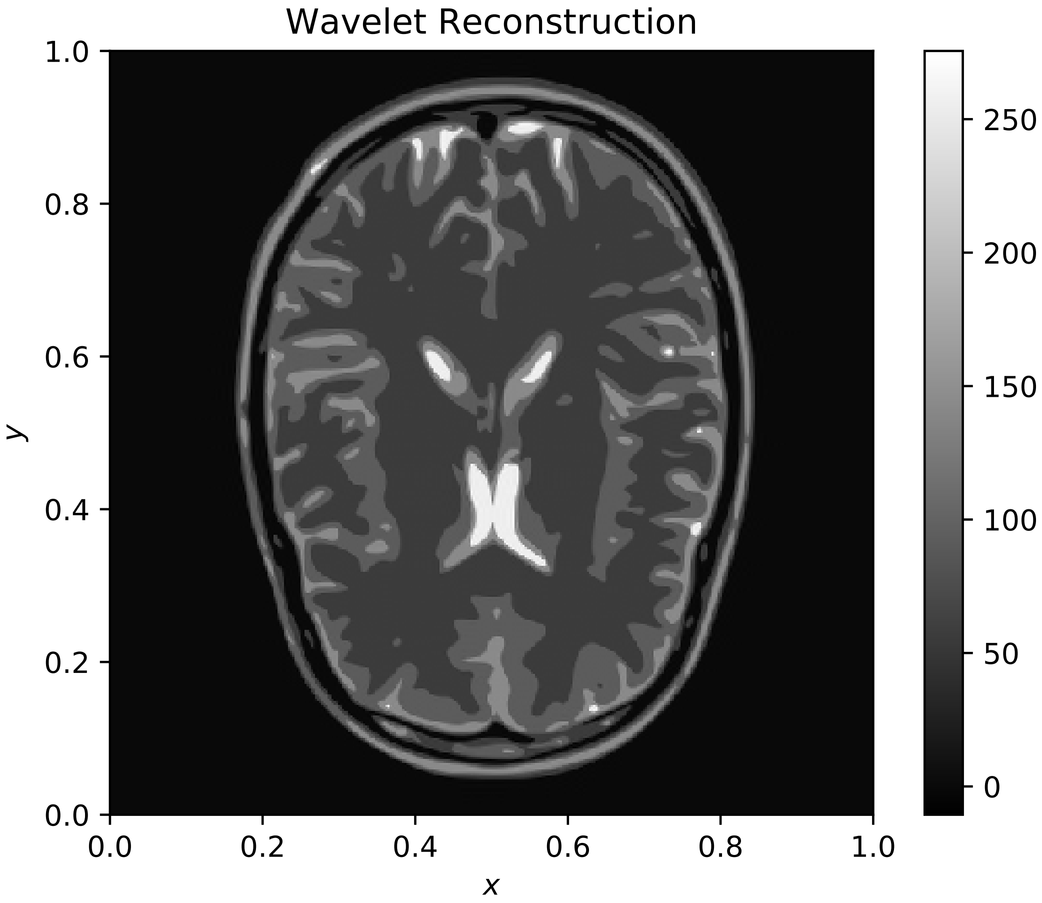


Figure A.2: The wavelet reconstruction of Guerquin-Kern phantom.

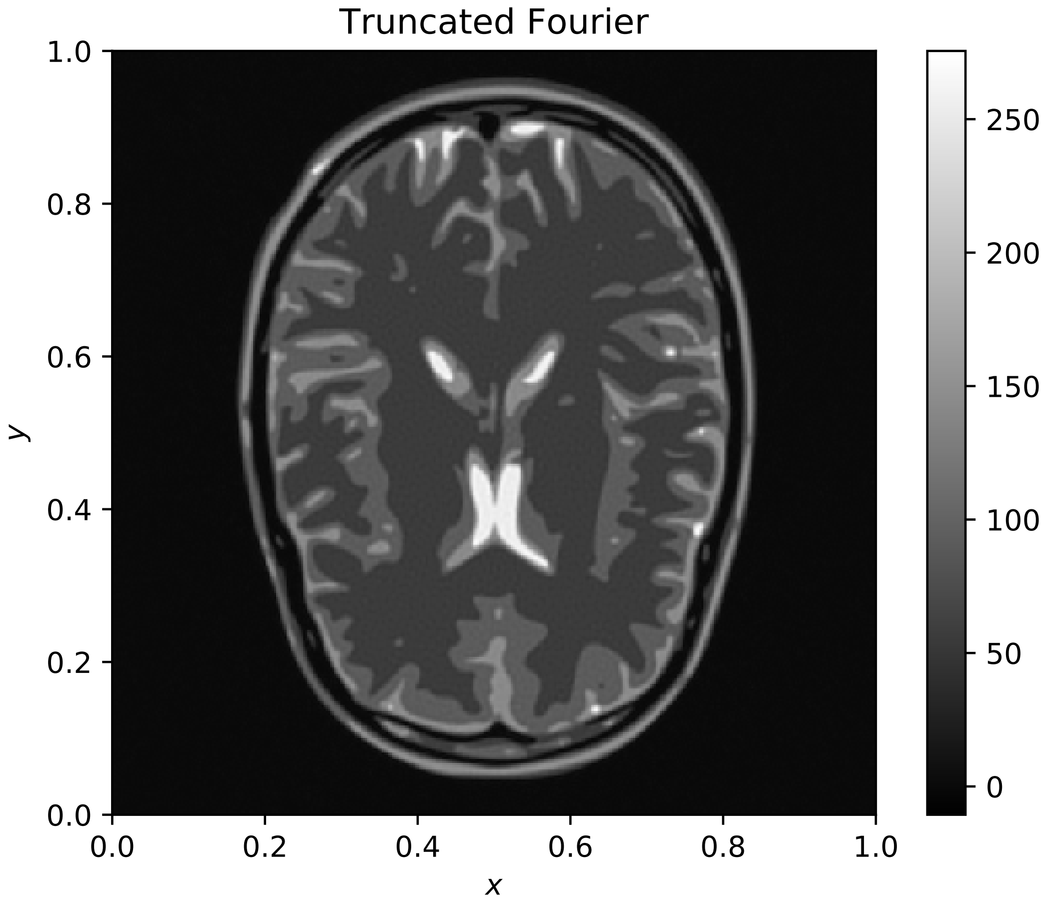


Figure A.3: The reconstruction from truncated Fourier coefficients of the Guerquin-Kern phantom.

A.2 Cameraman

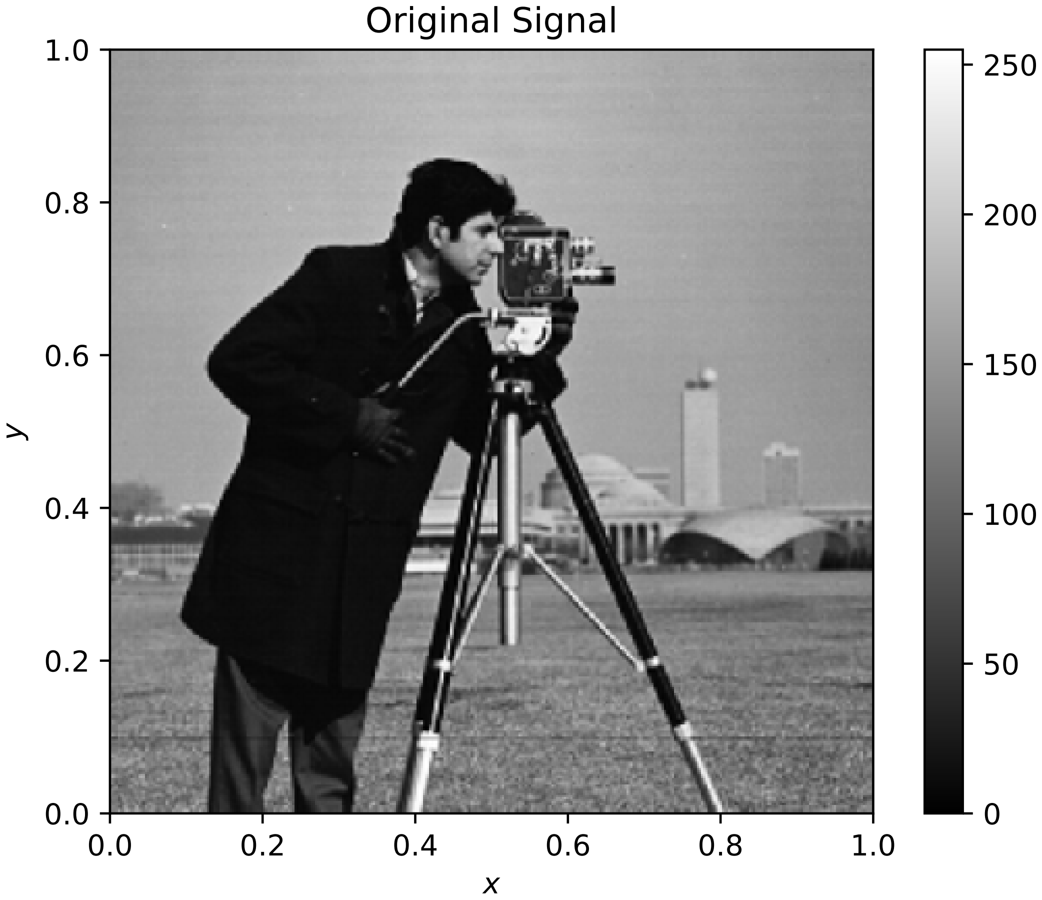


Figure A.4: The original image of the cameraman.

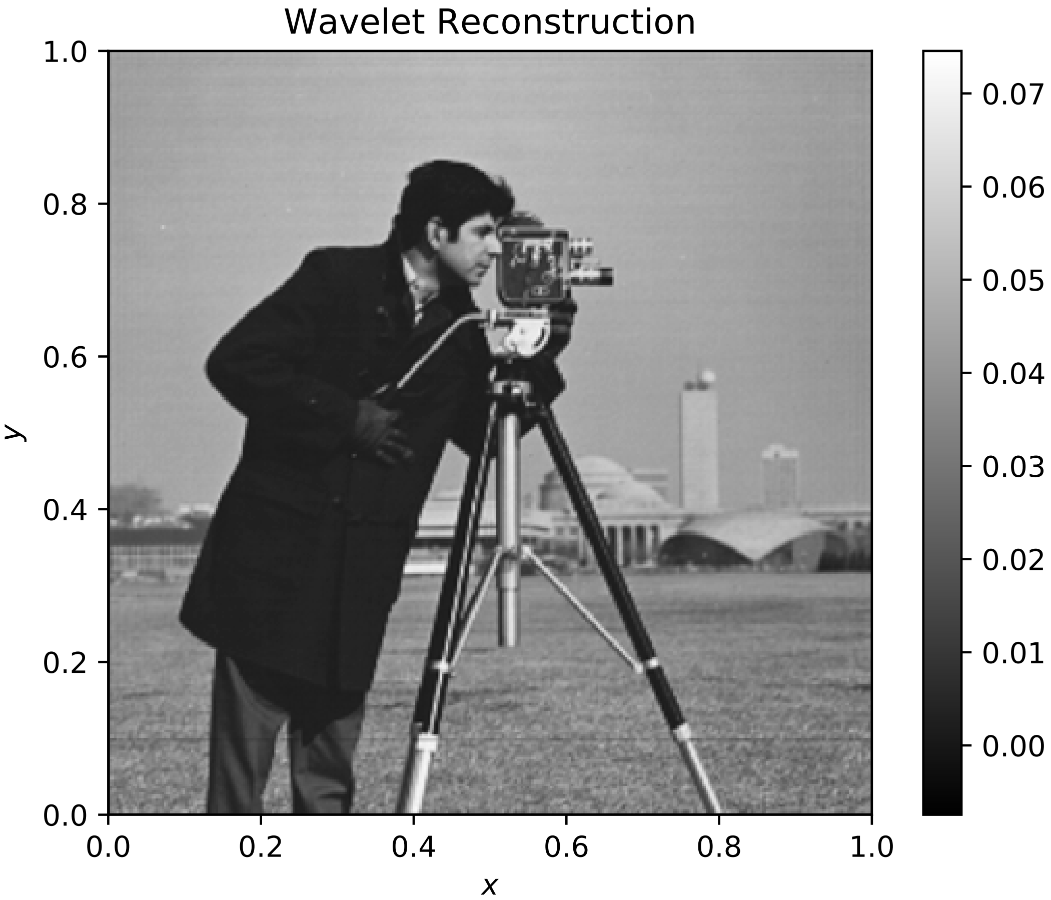


Figure A.5: The wavelet reconstruction of the cameraman.

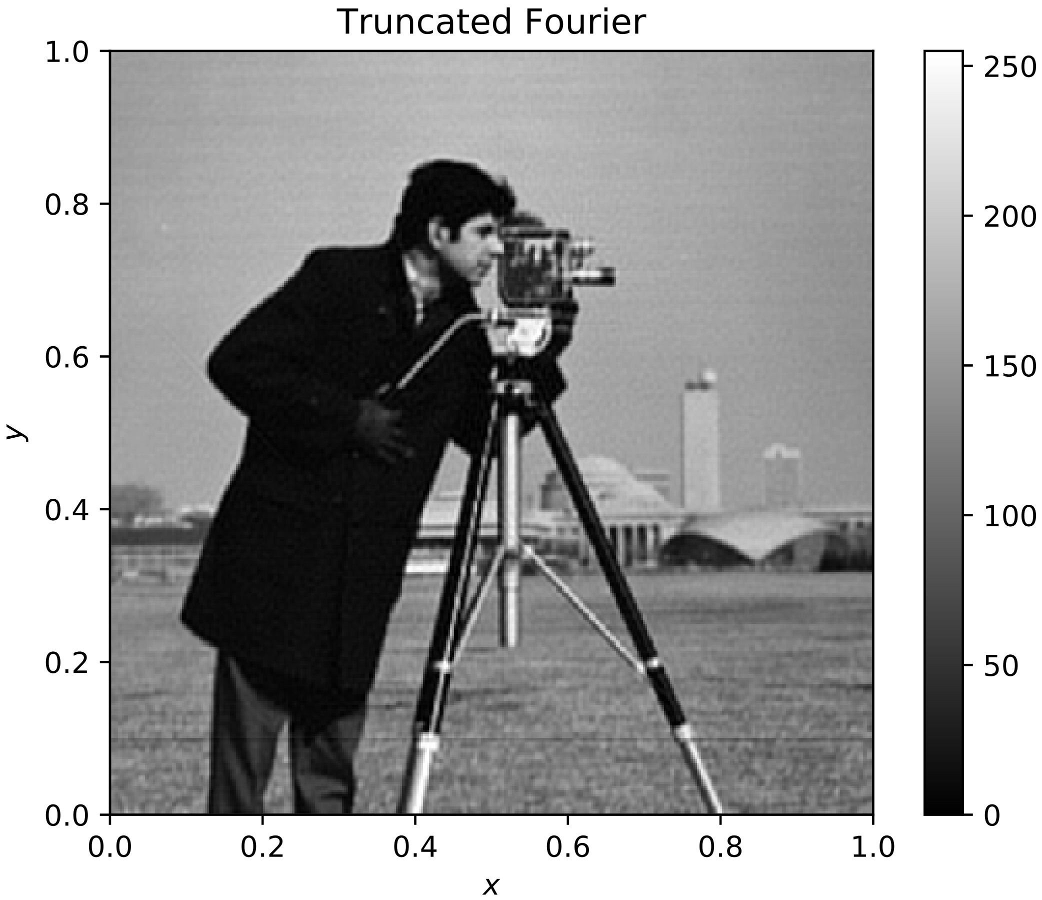


Figure A.6: The reconstruction from truncated Fourier coefficients of the cameraman.