SJÄLVSTÄNDIGA ARBETEN I MATEMATIK

MATEMATISKA INSTITUTIONEN, STOCKHOLMS UNIVERSITET

Wavelets on $\mathbb{Z}_N$

av

Anton Fahlgren

2016 - No 12

Wavelets on $\mathbb{Z}_N$

Anton Fahlgren

Självständigt arbete i matematik 15 högskolepoäng, grundnivå

Handledare: Salvador Rodriguez-Lopez

2016

Contents

Abstract

In this text we lay the foundations of the theory of discrete wavelets in $\mathbb{Z}_N$ and show its usefulness with practical examples. We discuss the problem of simultaneous localization in the time and frequency domains and the uncertainty principle, which motivates the construction of wavelets. We define and prove the validity of multiresolution analysis and give examples of its use in signal compression.

Acknowledgements

I would like to thank Salvador Rodriguez-Lopez for guiding and inspiring my work, and Rikard Bøgvad for refereeing it.

1 Introduction

Using the information we gather from the world around us, mankind has learned to draw the most astonishing conclusions. For example, just analyzing the light coming from stars, we have deduced their chemical properties and that the universe is expanding. Using a prism, Isaac Newton found that white light is composed of all the colors of the rainbow. Building on this idea, we would later be able to analyze the so called frequency spectrum of the light from the stars and identify missing frequencies that are the "fingerprints" of the basic elements [12]. And studying how the wavelengths of light from distant stars is "stretched out" by the doppler effect, we could deduce that the farther away they are, the faster they are moving away from us. [8] [9]

In studying these rays of light, we engage in *signal processing*, the area of applied mathematics where wavelets are used. But what do we mean by the word *signal*? In everyday language, a signal usually means some kind of message, be it a flashing light, a sound, or waving of hands. Here we will define a signal as any kind of event that can be measured and represented by numbers, which certainly applies at least to light and sound, as we can break them down in numbers representing brightness, color, loudness, frequency etc. We gather this information and try to understand it by analyzing the numbers. Is there a rhythm to the sound, or some kind of pattern in the flashing light? A rhythm is, more or less, a simple mathematical relation between time and the strength of the signal. For example, if we measure the ocean waves hitting the shore, there is an obvious pattern to the sequence 1,0,1,0,1,0,..., that is, the waves come in every other second. But some patterns can be more subtle, like relations between the vibrations of strings that produce harmonic sounds.

The sequence we gave above can be associated with what we call a *frequency*, meaning a rate of repetition. We can say that the wave sequence has a frequency of 2; it takes two seconds to return to the initial value. What we have done is represent a possibly infinite sequence by a single number, its rate of repetition, so analyzing frequency can often simplify signals that behave similarly over long periods of time. Your brain can actually do this analysis automatically; think of how you can identify two individual notes that are played simultaneously on a piano. Even though they are not separate in time, in some way we recognize the notes as distinct anyway, just as clearly as this page is separated by the next in space. The mathematical formalization of this process is called the Fourier transform, and is named after Joseph Fourier who discovered it when he was analyzing heat flow in the early 19th century.

If frequencies are also changing over time, it becomes hard to find a "small" set of numbers that describes the signal. On the one hand, a frequency does not exist in an instant of time,

but over a longer period. Likewise, it's hard to describe a sudden event in time as a sum of frequencies. Somehow, these worlds, or domains, are fundamentally separated. In fact, there is a mathematical principle, called the uncertainty principle, which we will prove in the case of discrete signals, that limits how small the set of numbers can be that describes a phenomena in both domains. We say that a signal can't be *localized* in time and frequency simultaneously, roughly meaning that there is no simple representation of any signal in both domains at the same time. If a signal has a simple representation in the time domain, it will have a complicated representation in the frequency domain. Wavelet analysis is partly concerned with maximizing this localization. In other words, we use it to, as efficiently as possible, represent signals that have important properties in both domains, like a sound wave that changes over time, for example.

Another important feature of wavelet analysis is what we call multiresolution analysis, which essentially is a way to divide a signal into different levels of detail. A good analogy to describe this is to compare it with how a sculptor makes a sculpture. To start with, the artist only has a block of wood, rock or clay or whatever material she is using. She starts by carving out the main features, the rough shape of the model object. Finer and finer features are carved out until at last, the smallest scalpel or hammer is brought out to finish the work. With multiresolution analysis, we can do this in reverse by extracting layers of detail, one after one, until nothing is left but the most basic remnant of the original signal. In this way we can analyze a signal at many different scales, depending on our needs. When this decomposition is done, we can easily reconstruct the signal perfectly, or to an extent that leaves out unnecessary details.

Nowadays, wavelets are used extensively in all kinds of fields. One notable application is in image compression. For example, using wavelet analysis, the FBI together with experts developed a new way to compress fingerprint images [2, Prologue]. Since fingerprints all have similar structure, they were able to develop wavelets that were especially efficient in compressing the files without losing essential information. Generally it is also a great tool for detecting certain features in large sets of signals with similar structures, for example it has important application in medical imaging such as X-ray and MRI. Other applications include detecting defects in textiles or integrated circuits in factories, or to analyze DNA sequences as well as satellite images. [5]

In the context of applied mathematics, the major developments of the subject have taken place quite recently. The first reference to it goes back to Alfred Haar (1885-1993) in 1910 and the construction of what would eventually be called the Haar wavelet, the simplest and most intuitive wavelet basis. It is good for finding sharp edges that exist in time or space, for example sudden pops in a sound or sharp edges in an image. In 1946, Dennis Gabor (1900-1979) made an effort to combine frequency analysis with time analysis, where one analyzes the frequency spectrum over intervals of time. In the late 80's the subject exploded, and multiresolution analysis was invented in 1988 by Stephane Mallat and Yves Meyer. Other notable contributors include Ingrid Daubechies, Jan-Olov Strömberg and Jean Morlet, among others [3].

The objective of this text is to lay the foundations of the theory of wavelets in $\mathbb{Z}_N$ and motivate its usefulness using practical examples. The main reference for the theory is Michael Frazier's

book [2], though we try to focus more on intuition and examples, while still respecting rigor, and take a slightly different approach to the construction of the wavelet bases in Chapter 3.

In Chapter 2 we define the discrete Fourier transform (DFT) and discuss its key properties; localization in frequency and connection to what we call linear translation invariant system, like amplifiers and filters. We also describe the fast Fourier transform, an essential property of the DFT which greatly reduces the computer power needed to calculate it. To end the chapter we prove the discrete case of the uncertainty principle, which quantifies the profound duality between the time domain and the frequency domain.

In Chapter 3 we construct a general wavelet basis in the space of functions of size 2^p, the most important special case due to the computational advantages it brings with it. We describe the multiresolution analysis and synthesis algorithm that decomposes any signal with regard to its behavior at different scales, i.e. levels of detail, and give a simple example of this process. Finally, in Chapter 4, we go through some of the numerical computations we made, like implementation of fast Fourier transform.

Had this been a more extensive work, we would have moved on to give some more examples of specific wavelet applications, one particularly illustrative example of multiresolution analysis being compression of image files. Unfortunately, we were not able to contain it in this text.

2 The Discrete Fourier Transform

2.1 Introduction

The Discrete Fourier transform is the conversion from the *time domain*, meaning the standard basis, to the *frequency domain*, meaning the Fourier basis, which we will define in Section 2.3. With it we will be able to express any signal in a basis of periodic functions, the coefficients of which will be called the *frequency components* of the signal. We call the standard basis the time domain since many signals are measured as a function of time. It is also often called the space domain if the values of the signal are associated with positions, like for example in image files.

2.2 Basic Definitions

Definition 2.1. *We define $\ell^2(\mathbb{Z}_N)$ as the set of functions $f : \mathbb{Z}_N \to \mathbb{C}$, and represent a function z by $z = (z(0), z(1), ..., z(N-1))$, where $\mathbb{Z}_N$ denotes the integers modulo N. The functions are defined over all integers, where $N = 0 \mod N$ and so on.*

These functions, often called *signals* in applications, form a vector space, and its structure is equivalent to $\mathbb{C}^N$ in the sense that there exists a bijection between the functions of $\ell^2(\mathbb{Z}_N)$ and the vectors of $\mathbb{C}^N$, in which case z is represented by the column vector $[z(0), z(1), ..., z(N-1)]^t$. This fact is useful when we want to calculate linear transformations by matrix multiplication. Later we may use the words function, signal or vector interchangeably while still using the opposite notation. Next, we remind the reader of some features of these vector spaces.

Definition 2.2. *Let $z, w \in \ell^2(\mathbb{Z}_N)$.*

1. *The* inner product *is defined by $\langle z, w \rangle = \sum_{n=0}^{N-1} z(n)\overline{w(n)}$.*
2. *The* norm *is defined by $||z|| = \sqrt{\sum_n |z(n)|^2}$. The quantity $||z||^2$ is often referred to as the* energy *of z.*
3. *The* complex conjugate *of $z = a + bi$ is written $\overline{z} = a - bi$.*

Remark 2.1. *The term energy comes from an analogy with physics, where the kinetic energy of an object given a velocity vector $v(t)$ is equal to $\frac{m}{2}||v||^2$.*

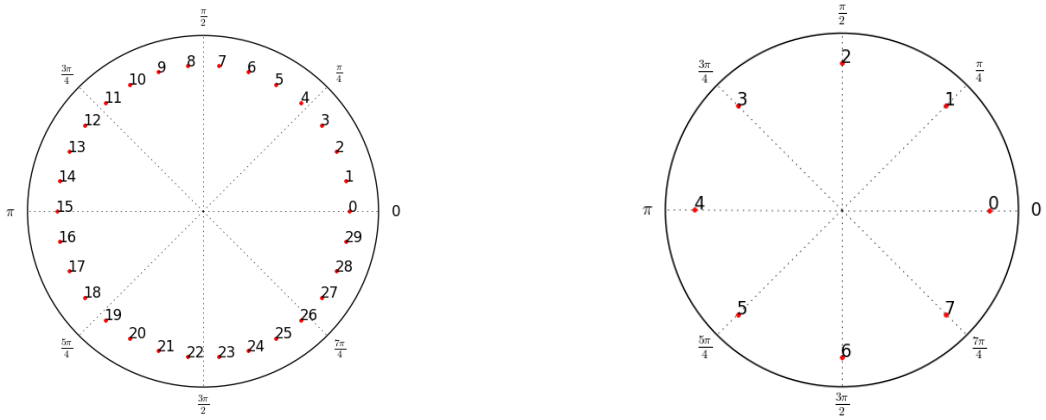
In dealing with the complex numbers, the complex exponential is a very important function.

Definition 2.3. *The complex exponential $z = e^{i\theta}$ is a representation of a point on the unit circle in the complex plane where θ is the positive angle to the real axis. By Euler's formula, $e^{i\theta} = \cos\theta + i\sin\theta$. Since $\cos\theta$ is an even function, the conjugate of $e^{i\theta}$ is $e^{-i\theta}$.*

For reasons we will see soon, the following set of functions play a crucial role in the subject of Fourier transform and wavelets.

Definition 2.4. *We define the set of functions $\{E_0, E_1, ..., E_{N-1}\} \in \ell^2(\mathbb{Z}_N)$ given by*

$$\begin{aligned}
E_0(n) &= \frac{1}{\sqrt{N}}, \\
E_1(n) &= \frac{1}{\sqrt{N}} e^{2\pi i n/N}, \\
E_2(n) &= \frac{1}{\sqrt{N}} e^{2\pi i 2n/N}, \\
&\vdots \\
E_{N-1}(n) &= \frac{1}{\sqrt{N}} e^{2\pi i (N-1)n/N} \quad \text{for } n = 0, 1, 2, ..., N-1.
\end{aligned}$$



(a) E_1 for $N = 30$ (b) E_1 for $N = 8$

Figure 2.1

As exemplified in Figure 2.1 where we have plotted the values of these functions using python, we get N points on the circle of radius $\frac{1}{\sqrt{N}}$ centered at the origin in the complex plane (which may or may not be distinct). For example, as n goes from 0 to $N-1$, the values of E_1 are N distinct and evenly spaced points on this circle.

Remark 2.2. *If we interpret n as a time variable, we can associate the functions above with frequencies. The function E_1 is "slow", or has a low frequency, since the values at consecutive*

indices are close as measured on the unit circle in $\mathbb{C}$. Note also that $e^{2\pi i(N-1)/N} = e^{-2\pi i/N}$, so the function E_{N-1} is also regarded as slow in the same sense. The functions E_k for k near $N/2$ are associated with high frequencies, since the values of consecutive indices are farther apart on the circle. We can think of the function E_0 as the "zero frequency", being a constant function.

The following lemma establishes that the functions E_k form an orthonormal basis in $\ell^2(\mathbb{Z}_N)$.

Lemma 2.1. *The set $\{E_0, E_1, ..., E_{N-1}\}$ forms an orthonormal basis on the space $\ell^2(\mathbb{Z}_N)$.*

Proof. We need to show that each vector E_n is orthogonal to every other vector in the basis, i.e. that the inner product is 0, and that the norm of each vector is 1, or that $\langle E_n, E_n\rangle = ||E_n||^2 = 1$.

$$\begin{aligned}\langle E_j, E_k\rangle = \sum_{n=0}^{N-1} E_j(n)\overline{E_k(n)} = \sum_{n=0}^{N-1} \frac{1}{\sqrt{N}} e^{2\pi i jn/N} \overline{\frac{1}{\sqrt{N}} e^{2\pi ikn/N}} \\ = \frac{1}{N}\sum_{n=0}^{N-1} e^{2\pi i jn/N} e^{-2\pi ikn/N} = \frac{1}{N}\sum_{n=0}^{N-1} (e^{2\pi i(j-k)/N})^n.\end{aligned}$$

We see that if $j = k$, $e^{2\pi i(j-k)/N} = e^0 = 1$, and the inner product is 1. For $j \neq k$, $j-k$ is an integer and $N > 1$, and we have

$$\sum_{n=0}^{N-1} (e^{2\pi i(j-k)/N})^n = \frac{1-(e^{2\pi i(j-k)/N})^N}{1-e^{2\pi i(j-k)/N}} = \frac{1-e^{2\pi i(j-k)}}{1-e^{2\pi i(j-k)/N}} = \frac{1-1}{1-e^{2\pi i(j-k)/N}} = 0,$$

where we use the formula $\sum_{n=0}^{N-1} z^n = \frac{1-z^N}{1-z}$. □

2.3 Defining the Discrete Fourier Transform

Now we are ready to define the discrete Fourier transform (DFT).

Definition 2.5. *The* discrete Fourier transform *is a linear map $\hat{}: \ell^2(\mathbb{Z}_N) \to \ell^2(\mathbb{Z}_N)$ defined for $m \in \mathbb{Z}_N$ by*

$$\hat{z}(m) = \sqrt{N}\langle z, E_m\rangle = \sum_{n=0}^{N-1} z(n) e^{-2\pi imn/N}.$$

Sometimes we will use the notation $\omega_N = e^{-2\pi i/N}$ for brevity, so the DFT becomes

$$\hat{z}(m) = \sum_{n=0}^{N-1} z(n)\omega_N^{mn}.$$

We say that the coefficients of $\hat{z}$ are the frequency components *of z.*

Since $\hat{}$ is a linear map in a finite dimensional space, it has an associated matrix, which is

$$W_N = \begin{bmatrix} 1 & 1 & 1 & 1 & \dots & 1 \\ 1 & \omega_N^1 & \omega_N^2 & \omega_N^3 & \dots & \omega_N^{(N-1)} \\ 1 & \omega_N^2 & \omega_N^4 & \omega_N^6 & \dots & \omega_N^{2(N-1)} \\ 1 & \omega_N^3 & \omega_N^6 & \omega_N^9 & \dots & \omega_N^{3(N-1)} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega_N^{(N-1)} & \omega_N^{2(N-1)} & \omega_N^{3(N-1)} & \dots & \omega_N^{(N-1)(N-1)} \end{bmatrix},$$

that is, $\hat{z} = W_N z$.

Example 2.1. We calculate the DFT of $z = (1,2,1,2)$. From the formula for W_N we get the transformation matrix

$$W_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -i & -1 & i \\ 1 & -1 & 1 & -1 \\ 1 & i & -1 & -i \end{bmatrix}.$$

Matrix multiplication yields

$$\hat{z} = W_4 z = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -i & -1 & i \\ 1 & -1 & 1 & -1 \\ 1 & i & -1 & -i \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 6 \\ 0 \\ -2 \\ 0 \end{bmatrix},$$

which is equivalent to $\hat{z} = (6,\ 0,\ -2,\ 0)$.

Remark 2.3. *This example helps justify the interpretation of the functions E_k as frequency components. The function z in the example is a simple oscillation between 1 and 2, so we would expect $\hat{z}$ to have a simple representation. Indeed, there are only two frequency components, namely $E_2 = (1,-1,1,-1)$, which we can see has the same rate of oscillation as z, and the constant function E_0 since the oscillation is not around 0, but 1.5.*

The matrix W_N is a special case of a Vandermonde matrix, meaning that the terms in each row form geometric progressions with distinct ratios. A well known result [7] about these matrices is that the determinant of a Vandermonde matrix V with ratios $\alpha_1, \alpha_2, ..., \alpha_N$ (in our case $1, \omega_N, \omega_N^2, ..., \omega_N^{N-1}$) is given by

$$\det(V) = \prod_{1 \le i < j \le n} (\alpha_j - \alpha_i).$$

Since the ratios α_i, α_j are distinct if $i \neq j$, the determinant is nonzero, and thus the matrix is invertible, meaning that the DFT has an inverse operator.

Lemma 2.2. *The* Fourier inversion formula *is given by*

$$z(n) = \frac{1}{N}\sum_{m=0}^{N-1} \hat{z}(m)e^{2\pi imn/N}.$$

Proof. The formula yields the matrix

$$M_N = \frac{1}{N}\begin{bmatrix} 1 & 1 & 1 & 1 & \dots & 1 \\ 1 & \omega_N^{-1} & \omega_N^{-2} & \omega_N^{-3} & \dots & \omega_N^{-(N-1)} \\ 1 & \omega_N^{-2} & \omega_N^{-4} & \omega_N^{-6} & \dots & \omega_N^{-2(N-1)} \\ 1 & \omega_N^{-3} & \omega_N^{-6} & \omega_N^{-9} & \dots & \omega_N^{-3(N-1)} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega_N^{-(N-1)} & \omega_N^{-2(N-1)} & \omega_N^{-3(N-1)} & \dots & \omega_N^{-(N-1)(N-1)} \end{bmatrix}$$

for applying the inversion. When taking the product $M_N W_N$, the elements are on the form $\frac{1}{N}\sum_{n=0}^{N-1} \omega_N^{in}\omega_N^{-jn}$, where $0 \leq i, j \leq N-1$. If $i = j$, the sum is 1. If not, $i - j = k \neq 0$, and the sum is $\frac{1}{N}\sum_{n=0}^{N-1} \omega_N^{kn}$. We have

$$\omega_N^{kN} - 1 = (\omega_N^k - 1)(1 + \omega_N^k + \omega_N^{2k} + ... + \omega_N^{(N-1)k}),$$

and $\omega_N^{kN} = 1$, so the left hand side is 0. Since $k \neq 0$ and $|k| < N$, $\omega_N^k \neq 1$, so $(1 + \omega_N^k + \omega_N^{2k} + ... + \omega_N^{(N-1)k}) = 0$. Therefore the product $M_N W_N = I$ and $M_N = W_N^{-1}$ (it is also the right inverse). □

Definition 2.6. *We define the* inverse discrete Fourier transform *(IDFT) as the function* $\check{}$: $\ell^2(\mathbb{Z}_N) \to \ell^2(\mathbb{Z}_N)$ *given by*

$$\check{w}(n) = \frac{1}{N}\sum_{m=0}^{N-1} w(m)e^{2\pi imn/N}. \tag{2.1}$$

Remark 2.4. *As we see in the discrete Fourier inversion formula, the matrix M_N is just $\frac{1}{N}\overline{W}_N$, and we can see the two operations are closely connected. This means that many of the lemmas and theorems proved for the DFT have analog proofs for the IDFT.*

Note that the set $\{E_k\}$ is not the basis in which $\hat{z}$ is expressed, since $z \neq \sum_{m=0}^{N-1} \hat{z}(m)E_m$. We could have defined the Fourier coefficients as $\langle z, E_m\rangle$, however this would mean that the values of the matrix W_N would be divided by $\sqrt{N}$, and would be very small if N was high. When calculating with computers, this would lead to bigger errors since there would be less significant digits. Of course, we now need to divide by N in the matrix M_N instead of $\sqrt{N}$, but it is better to make only one division rather than two when converting back and forth, since errors accumulate with these divisions. So, even if we won't use it much explicitly, we should clarify the following.

Lemma 2.3. *The set* $\{F_0, F_1, ..., F_{N-1}\}$*, called the* Fourier basis*, defined by*

$$F_m = \frac{1}{\sqrt{N}}E_m$$

is an orthogonal basis in $\ell^2(\mathbb{Z}_N)$, and the coefficients of z in the Fourier basis are $\hat{z}(m)$ for all $m \in \mathbb{Z}_N$.

Proof. By linearity of the scalar product, the set $\{F_k\}$ is an orthogonal basis since $\{E_k\}$ is an orthogonal basis by Lemma 2.1. Also, using equation (2.1),

$$z(m) = \frac{1}{N}\sum_{m=0}^{N-1} \hat{z}(m)e^{2\pi imn/N} = \sum_{m=0}^{N-1} \hat{z}(m)F_m,$$

so $\hat{z}(m)$ are the coefficients of z in the Fourier basis. □

2.4 Properties of The DFT

The two identities below can be interpreted as saying that the DFT preserves proportional angles (in particular it preserves orthogonality) and energies of signals.

Lemma 2.4. *Let $z, w \in \ell^2(\mathbb{Z}_N)$. Then*

1. *(*Parseval's relation*)*
$$\langle z, w\rangle = \frac{1}{N}\sum_{m=0}^{N-1} \hat{z}(m)\overline{\hat{w}(m)} = \frac{1}{N}\langle \hat{z}, \hat{w}\rangle.$$

2. *(*Plancherel's formula*)*
$$||z||^2 = \frac{1}{N}\sum_{m=0}^{N-1} |\hat{z}(m)|^2 = \frac{1}{N}||\hat{z}||^2.$$

Proof. For the first relation, since, by Lemma 2.1, $\{E_k\}_{k=0}^{N-1}$ is an orthonormal basis we have $z = \sum_{m=0}^{N-1}\langle z, E_m\rangle E_m$. Then

$$\langle z, w\rangle = \left\langle \sum_{m=0}^{N-1} \langle z, E_m\rangle E_m, w \right\rangle.$$

By linearity of the scalar product, we have

$$\left\langle \sum_{m=0}^{N-1} \langle z, E_m\rangle E_m, w \right\rangle = \sum_{m=0}^{N-1} \langle z, E_m\rangle\langle E_m, w\rangle = \sum_{m=0}^{N-1} \langle z, E_m\rangle\overline{\langle w, E_m\rangle}$$

using the conjugate symmetry of the inner product. Note that

$$\sum_{m=0}^{N-1} \langle z, E_m\rangle\overline{\langle w, E_m\rangle} = \frac{1}{(\sqrt{N})^2}\sum_{m=0}^{N-1} \hat{z}(m)\overline{\hat{w}(m)} = \frac{1}{N}\langle \hat{z}, \hat{w}\rangle.$$

The second statement is a direct consequence of the first one. □

2.4.1 Localization

As we have hinted, the Fourier basis is very convenient when it comes to dealing with periodic functions, which come up in all kinds of applications. For example, in working with audio signals, a function that looks complicated in the standard basis can have a very simple representation in the Fourier basis, as the following example shows.

Example 2.2. If $z = (0, \frac{1}{\sqrt{2}}, 1, \frac{1}{\sqrt{2}}, 0, -\frac{1}{\sqrt{2}}, -1, -\frac{1}{\sqrt{2}})$, then $\hat{z} = (0, -4i, 0, 0, 0, 0, 0, 4i)$. In fact, z is just the values of $\sin(\frac{\pi n}{4})$. Since by Euler's formula, $\sin(\frac{\pi n}{4}) = \frac{e^{2\pi in/8} - e^{-2\pi in/8}}{2i}$, we can see that E_1 and E_7 should be the only contributing frequencies.

What this suggests is that the Fourier basis is *localized* in frequency. We say that a function is localized if the values of the function are mostly zero (or sufficiently small). In other words, the significant values are concentrated in specific locations. We know for example that the standard basis is localized in time; the base vectors are just the rows of the identity matrix. This means that a specific event in time is localized to certain base vectors. For example, the signal $z = (0,0,0,1,0,0,0,0)$, a single spike in amplitude at $n = 3$, is localized by the standard basis, while the Fourier transform looks like:

$$\hat{z} = (1, -\frac{1}{\sqrt{2}} - \frac{1}{\sqrt{2}}i, i, \frac{1}{\sqrt{2}} - \frac{1}{\sqrt{2}}i, -1, \frac{1}{\sqrt{2}} + \frac{1}{\sqrt{2}}i, -i, -\frac{1}{\sqrt{2}} + \frac{1}{\sqrt{2}}i).$$

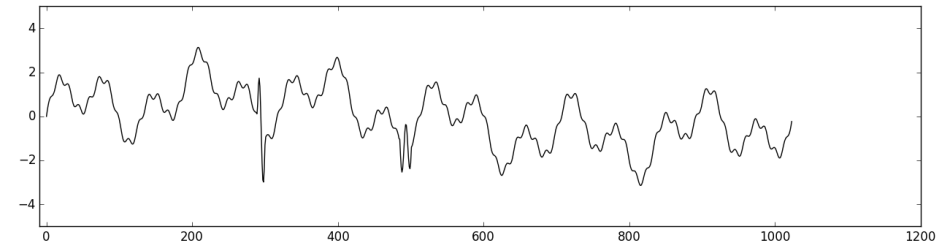
Only looking at $\hat{z}$, it is very hard to see how z behaves as a function of time. However, if we apply the DFT to the Fourier basis vectors, we notice that $\hat{F}_m(n) = \sqrt{N}\langle F_m, E_n\rangle$ is nonzero if and only if $n = m$. Thus, the Fourier basis is perfectly localized in frequency. As Example 2.2 showed, particular frequencies show up at particular indices of $\hat{z}$. This allows us to single out and manipulate different frequency components of a signal, which is very useful.

Example 2.3. In Figure 2.2 (a) we have generated the plot using python code of the function $z \in \ell^2(\mathbb{Z}_{1024})$ such that

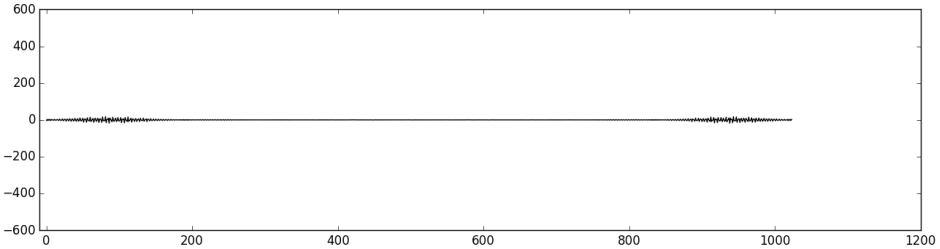
$$z(n) = \sin(\frac{2\pi n}{1024}) + \sin(\frac{12\pi n}{1024}) + \sin(\frac{32\pi n}{1024}) + \frac{1}{5}\sin(\frac{150\pi n}{1024})$$

with added noise around $n = 300$ and $n = 500$ (the exact definition of the function in python code is found in Chapter 4). The discrete values are connected with a line for aesthetic purposes. From the formula we see that there are four dominant frequencies involved, which can clearly be identified in the plot of the imaginary part of $\hat{z}$; yet another example of a signal being localized by the Fourier basis. (Well, we actually see eight spikes in Figure 2.2 (c). We can use the same reasoning as in Example 2.2 to conclude that these functions are E_1, E_6, E_{16} and E_{75} as well as

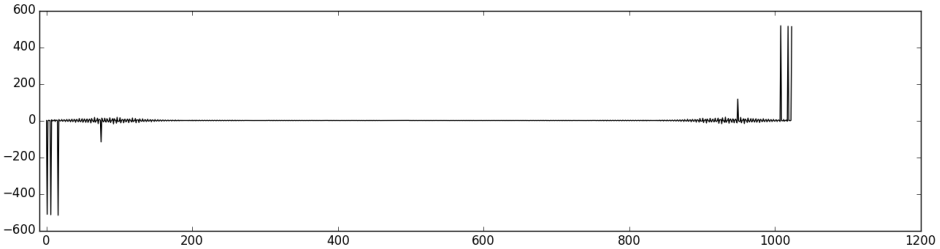
the conjugates E_{1024}, E_{1019}, E_{1009} and E_{950}). After making the DFT of z we could easily filter out the noise if we wanted to by *thresholding*, meaning we only keep the frequencies for which the coefficients are above a certain threshold, in this case a threshold of 50 or so seems to be enough. A plot of the filtered signal can be found in Chapter 4.



(a) z



(b) Real part of $\hat{z}$



(c) Imaginary part of $\hat{z}$

Figure 2.2: The signal from Example 2.3 and its discrete Fourier transform.

2.4.2 Translation Invariant Operators in $\ell^2(\mathbb{Z}_N)$

In this section we prove a very important theorem in the subject, which states that certain operators in $\ell^2(\mathbb{Z}_N)$ are significantly simplified when working in the Fourier basis. In order to state the theorem, we must first make some definitions.

Definition 2.7. *Let* $z, w, \mathbf{m} \in \ell^2(\mathbb{Z}_N)$.

1. *We define the operator R_k as having the property:*

$$(R_k z)(n) = z(n-k) \quad \text{for } n,k \in \mathbb{Z},$$

and call it the translation operator.

2. *For $z, w \in \ell^2(\mathbb{Z}_N)$, the* convolution $z * w$ *is the function with values*

$$(z*w)(m) = \sum_{n=0}^{N-1} z(m-n)w(n).$$

If a function b is fixed, the transformation T_b defined by $T_b(z) = b*z$ is called a convolution operator.

3. *We define the* Fourier multiplier operator $T_{\mathbf{m}} : \ell^2(\mathbb{Z}_N) \to \ell^2(\mathbb{Z}_N)$ *by*

$$T_{\mathbf{m}}(z) = (\mathbf{m}\hat{z})^{\check{}}$$

where $\mathbf{m}\hat{z}$ is obtained by component-wise multiplication of $\mathbf{m}$ and $\hat{z}$. The function $\mathbf{m}$ is called a multiplier*, or a* filter.

The translation operator permutes the coordinates of z by rotation (hence the notation), so for example, if $k = 2$ and $z = (1, i, -1, -i)$, then $R_2 z = (-1, -i, 1, i)$. Note that, for example, $(R_2 z)(1) = z(1-2) = z(-1) = z(3)$ since we are working modulo $N = 4$. In audio signal applications, a translate in the standard basis is a shift in time. Regarding the Fourier multiplier operator, this operator scales each component of $\hat{z}$ by a scalar $\mathbf{m}(k)$, and then outputs the signal in the standard basis. This corresponds to amplifying or attenuating (reducing) frequencies in a signal. In Section 2.5 we will give a more intuitive explanation of convolution.

Further, we define translation invariance.

Definition 2.8. *A linear transformation $T : \ell^2(\mathbb{Z}_N) \to \ell^2(\mathbb{Z}_N)$ is* translation invariant *if*

$$T(R_k z) = R_k T(z)$$

for all $z \in \ell^2(\mathbb{Z}_N)$ and all $k \in \mathbb{Z}$.

Now we can state and prove the theorem, and then discuss its applications.

Theorem 2.1. *Let T be a linear transformation $\ell^2(\mathbb{Z}_N) \to \ell^2(\mathbb{Z}_N)$. The following are equivalent.*

i *T is translation invariant.*

ii *T is a convolution operator.*

iii *T is a Fourier multiplier operator.*

Proof. We prove $i \implies ii$ as part of the discussion of linear translation-invariant systems in Section 2.5, so here we prove $ii \implies i$ and $ii \iff iii$.

First, let T_b be a convolution operator.

$$T_b(R_k z)(m) = b * (R_k z)(m) = \sum_{n=0}^{N-1} b(m-n)(R_k z)(n) = \sum_{n=0}^{N-1} b(m-n)z(n-k).$$

Now make the variable substitution $p = n-k$. Then for all m,

$$\begin{aligned} T_b(R_k z)(m) &= \sum_{p=-k}^{N-1-k} b(m-k-p)z(p) = \sum_{p=0}^{N-1} b(m-k-p)z(p) \\ &= (b*z)(m-k) = R_k(b*z)(m) = R_k T_b(z)(m). \end{aligned}$$

Thus T_b is translation invariant.

Next, let $z, w \in \ell^2(\mathbb{Z}_N)$ and consider $(z*w)\hat{}$.

$$\begin{aligned} (z*w)\hat{}(m) &= \sum_{n=0}^{N-1} (z*w)(n)\omega_N^{mn} = \sum_{n=0}^{N-1}\sum_{k=0}^{N-1} z(n-k)w(k)\omega_N^{mn} \\ &= \sum_{n=0}^{N-1}\sum_{k=0}^{N-1} z(n-k)w(k)\omega_N^{m(n-k)}\omega_N^{mk} \\ &= \sum_{k=0}^{N-1} w(k)\omega_N^{mk}\sum_{n=0}^{N-1} z(n-k)\omega_N^{m(n-k)}. \end{aligned}$$

Now, again make the substitution $p = n-k$ in the second sum of the last row.

$$\sum_{n=0}^{N-1} z(n-k)\omega_N^{m(n-k)} = \sum_{p=-k}^{N-1-k} z(p)\omega_N^{mp} = \sum_{p=0}^{N-1} z(p)\omega_N^{mp}.$$

Thus

$$(z*w)\hat{}(m) = \sum_{k=0}^{N-1} w(k)\omega_N^{mk}\sum_{p=0}^{N-1} z(p)\omega_N^{mp} = \hat{z}(m)\hat{w}(m). \tag{2.2}$$

□

Hinging on an important fact about the Fourier Transform, this theorem could prove extremely useful in applications, as it turns the complex and important operation of convolution into Fourier multipliers. Since convolution requires N^2 multiplications (each of the N coefficients of $z*w$ is defined as the sum of N multiplications) the computer power needed to compute them

grows beyond our limits for large enough N. However, the DFT itself is an operation that seems to require N^2 multiplications, but as we will see in the next section, there are shortcuts. These together with Theorem 2.1 are the most important features of the DFT.

2.4.3 The Fast Fourier Transform

For Theorem 2.1 to be effective in applications, we would need a fast algorithm to compute the DFT. Otherwise, to avoid doing the N^2 multiplications of convolution, we would need N^2 multiplications to calculate the DFT. Obviously that would defeat the purpose entirely. Luckily, there is an algorithm, appropriately named the fast Fourier transform (FFT), due in part to Gauss and much later updated by Cooley and Tukey [4]. As we show in Chapter 4, this algorithm is essential if we want to use the DFT on signals of greater size. We show it for the special case when N is even.

Theorem 2.2. *Suppose $M \in \mathbb{N}$, and $N = 2M$. Let $z \in \ell^2(\mathbb{Z}_N)$. Define $u, v \in \ell^2(\mathbb{Z}_M)$ by*

$$u = (z(0), z(2), z(4), ..., z(N-4), z(N-2))$$

and

$$v = (z(1), z(3), z(5), ..., z(N-3), z(N-1)).$$

Let $\hat{z} = W_N z$, $\hat{u} = W_M u$, and $\hat{v} = W_M v$. Then for $m = 0, 1, ..., M-1$

$$\hat{z}(m) = \hat{u}(m) + e^{-2\pi i m/N}\hat{v}(m) \tag{2.3}$$

For $m = M, M+1, M+2, ..., N-1$, let $l = m - M$. Then

$$\hat{z}(m) = \hat{z}(l+M) = \hat{u}(l) - e^{-2\pi i l/N}\hat{v}(l). \tag{2.4}$$

Proof. Remember that by definition

$$\hat{z}(m) = \sum_{n=0}^{N-1} z(n) e^{-2\pi i m n/N}.$$

By the properties of sums

$$\begin{aligned}\hat{z}(m) &= \sum_{k=0}^{M-1} z(2k)e^{-2\pi i 2km/N} + \sum_{k=0}^{M-1} z(2k+1)e^{-2\pi i(2k+1)m/N} \\ &= \sum_{k=0}^{M-1} u(k)e^{-2\pi ikm/(N/2)} + e^{-2\pi im/N}\sum_{k=0}^{M-1} v(k)e^{-2\pi ikm/(N/2)} \\ &= \sum_{k=0}^{M-1} u(k)e^{-2\pi ikm/M} + e^{-2\pi im/N}\sum_{k=0}^{M-1} v(k)e^{-2\pi ikm/M}.\end{aligned}$$

For $m = 0,1,...,M-1$, we now have $\hat{z}(m) = \hat{u}(m) + e^{-2\pi im/N}\hat{v}(m)$. If $m = M, M+1, ..., N-1$, remember $m = l + M$.

$$\begin{aligned}&\sum_{k=0}^{M-1} u(k)e^{-2\pi ik(l+M)/M} + e^{-2\pi i(l+M)/N}\sum_{k=0}^{M-1} v(k)e^{-2\pi ik(l+M)/M} \\ &= \sum_{k=0}^{M-1} u(k)e^{-2\pi ikl/M} + e^{-2\pi il/N}e^{-2\pi iM/N}\sum_{k=0}^{M-1} v(k)e^{-2\pi ikl/M} \\ &= \sum_{k=0}^{M-1} u(k)e^{-2\pi ikl/M} - e^{-2\pi il/N}\sum_{k=0}^{M-1} v(k)e^{-2\pi ikl/M},\end{aligned}$$

since $e^{-2\pi iM/N} = e^{-2\pi iM/(2M)} = e^{-\pi i} = -1$. Thus $\hat{z}(m) = \hat{z}(l+M) = \hat{u}(l) - e^{-2\pi il/N}\hat{v}(l)$. □

Since the DFT requires N^2 multiplications, using this formula once we have reduced the number of multiplications to $(\frac{N}{2})^2 + (\frac{N}{2})^2 = \frac{N^2}{2}$ plus $N/2$ multiplications with the constants in formula (2.3) and (2.4). For example, if $N = 10^4$, $N^2 = 10^8$ while $\frac{N^2}{2} + \frac{N}{2} = 5 \cdot 10^7 + 5 \cdot 10^3 \approx 5 \cdot 10^7$, which is a reduction by about a half. The real power of the method becomes apparent when we feed u and v back into the FFT algorithm. So if $N = 2^p$ for some $p > 1$, we can use Theorem 2.2 again to decompose u and v, which are of size M. Using induction one can show [2, Lemma 2.9] that the FFT can reduce the number of calculations to the order of $N \log_2 N/2$. This number makes intuitive sense, since we can break down $z \in \ell^2(\mathbb{Z}_{2^p})$ $\log_2 N$ times until we can calculate the DFT of N single value vectors. Half of those, corresponding to the v-vectors, are multiplied once in every $\log_2 N$ step until the calculation of $\hat{z}$ is completed.

See Chapter 4 to see an implementation of the FFT algorithm using python code, where we also compare the computation times of the FFT versus the DFT of a large vector.

2.5 Linear Translation Invariant Systems And Convolution

Here we will derive the formula for convolution in the setting of linear translation invariant (LTI), or time invariant, systems and give an example of such a system.

In Figure 2.3 we see a schematic diagram of what is called a system. A translation invariant system S is a system such that $S(R_k z) = R_k S(z)$. If the system is also linear, i.e. $S(z+w) = S(z)+S(w)$, it is called a LTI system. Two audio-related examples of LTI systems are (idealized) guitar amplifiers or concert halls; the sound the listener eventually receives is independent on when the guitar is played or when a singer sings. If S is linear, then the output of a signal z is

$$S(z) = S(z(0)e_0) + S(z(1)e_1) + ... + S(z(N-1)e_{N-1}) = \sum_{k=0}^{N-1} z(k)S(e_k). \tag{2.5}$$

Thus, in a LTI system, we could predict the system output of any signal by inputting a unit impulse, mathematically described by $(1,0,0,...,0) = e_0$, and get $S(e_0) = b$, which is called the *impulse response* of the system. Since e_k is just $R_k e_0$, i.e. e_0 shifted in time, we can calculate the output of any signal z using formula (2.5).

The output of the first impulse of the signal, $z(0)e_0$, will be $S(z(0)e_0) = z(0)S(e_0) = z(0)b$. The next impulse will be $z(1)e_1$, and $S(z(1)e_1) = z(1)R_1 b$. Depending on the impulse response, many terms in formula (2.5) may influence the output at any given time, and we get that the output at $n = 0,1,...,N-1$ is given by

$$(Sz)(n) = \sum_{k=0}^{N-1} z(k)S(e_k)(n) = \sum_{k=0}^{N-1} z(k)R_k b(n) = \sum_{k=0}^{N-1} z(k)b(n-k),$$

which is defined as the n-th coefficient of the convolution $z * b$. This proves $i \implies ii$ in Theorem 2.1.

Example 2.4. In a certain concert hall, the sound of a voice reverberates in the room after a singer has stopped singing. Let's make the big simplification that the time unit is 1 second, $N = 10$, and the singer stands in the same place and either sings the same note or is silent for each second. If she sings for one second at "unit" volume, the output of the signal $e_0 = (1,0,...,0)$, the impulse response, is $b = (1, \frac{4}{5}, \frac{3}{5}, \frac{2}{5}, \frac{1}{5}, 0, 0, 0, 0, 0)$, i.e. the sound echoes for 5 seconds and then stops for the rest of the period.

Now, a certain "song" is 5 seconds long and is given by $z = (5,5,10,10,20,0,0,0,0,0)$, where the singer starts off softly and builds up to a crescendo. The system output, i.e. the sound you receive in a certain seat, at 5 seconds, $(Sz)(4)$, will depend on what the singer is singing at the moment, and the echo of previous seconds performance. The input during the first second will

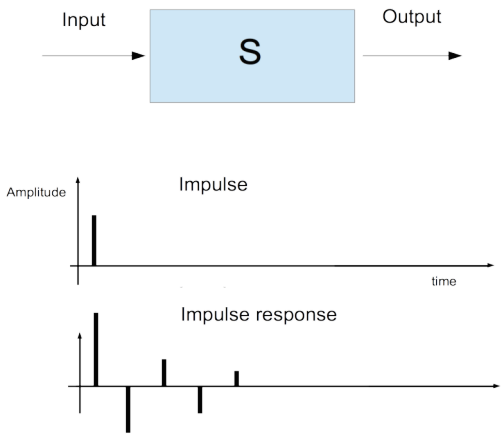


Figure 2.3: In a LTI system, the output of each signal is determined by the impulse response of the system. In this case, the impulse response seems to be some oscillating and diminishing signal.

be echoing with amplitude $5\frac{1}{5} = 1$, for example.

$$\begin{aligned}(Sz)(4) = (b*z)(4) &= \sum_{k=0}^{N-1} b(4-k)z(k) \\ &= 5\cdot\frac{1}{5} + 5\cdot\frac{2}{5} + 10\cdot\frac{3}{5} + 10\cdot\frac{4}{5} + 20\cdot 1 + 0 + 0 + 0 + 0 + 0 \\ &= 1+2+6+8+20 = 37.\end{aligned}$$

Even if it is rather unnecessary in this case, we can calculate the whole output signal $S(z)$ by using the DFT and equation (2.2). With the DFT-program shown in Chapter 4, we get the vectors (in $\mathbb{C}^N$ notation and rounded for readability):

$$\hat{b} = \begin{bmatrix} 3 \\ 1.55-1.54i \\ 0.5-0.69i \\ 0.65-0.36i \\ 0.5-0.16i \\ 0.6 \\ 0.5+0.16i \\ 0.65+0.36i \\ 0.5+0.69i \\ 1.55+1.54i \end{bmatrix} \qquad \hat{z} = \begin{bmatrix} 50 \\ -7.14-33.71i \\ -3.45+14.27i \\ 9.64-12.02i \\ -9.05+8.82i \\ 20 \\ -9.05-8.82i \\ 9.64+12.02i \\ -3.45-14.33i \\ -7.14+33.71i \end{bmatrix},$$

which, using the formula, yields

$$(b*z)\hat{} = \begin{bmatrix} 150 \\ -62.92-41.12i \\ 8.09+9.51i \\ 1.92-11.35i \\ -3.09+5.88i \\ 12 \\ -3.09-5.88i \\ 1.92+11.35i \\ 8.09-9.51i \\ -62.92+41.12i \end{bmatrix}.$$

Taking the inverse DFT of this vector gives us

$$((b*z)\hat{})\check{} = \begin{bmatrix} 5 \\ 9 \\ 17 \\ 23 \\ 37 \\ 27 \\ 18 \\ 10 \\ 4 \\ 0 \end{bmatrix},$$

and we can confirm that our earlier calculation of $(Sz)(4)$ gave the same value. This signal should then be interpreted as the output of the LTI system, where the values represent the sum of the volume of the singer and the echo of previously sung notes. As we can see, the volume reaches its peak in the fifth second, and as we expect the signal dies out five seconds after the singer sings the last note.

If instead $N = 2^p$ had been large, this method, using the FFT, would cut down the number of complex multiplications tremendously.

2.6 The Uncertainty Principle

Note that in Figure 2.2, z has a lot of nonzero values, while $\hat{z}$ seems to have a lot of zero or small values. This is no coincidence, and in fact there is a fundamental principle that says that if a signal is localized in frequency, it cannot also be localized in time and vice versa.

The uncertainty principle is a common name for a great number of statements in different ap-

plications and contexts. The reader may have heard about Heisenberg's uncertainty principle [10] in quantum mechanics, which is also a collection of mathematical statements concerned with the problem of localization. As shown by the famous double-slit experiment, elementary particles exhibit wave-particle duality, which means that they may be described in terms of both waves and particles. When making measurements of some quantum phenomenon, it would be meaningless to speak of the exact position of a wave, or the momentum (related to frequency) of a single particle. Heisenberg's uncertainty principle states that if the location of a quantum entity is known with accuracy, the momentum cannot be accurately determined, and vice versa. Quite analogously, the uncertainty principle that we are concerned with in this text says that a signal cannot be localized in time (or space, or whatever) and frequency simultaneously. For example, if z describes a signal that is concentrated in a short interval of time, and 0 outside that interval, it will have many different nonzero frequency components.

We prove this transcendent piece of mathematics restricted to discrete Fourier analysis, following the ideas of Donoho & Stark [1].

Lemma 2.5. *Let $z \in \ell^2(\mathbb{Z}_N)$. If N_t is the number of nonzero entries in z, then $\hat{z}$ cannot have N_t consecutive zeros. (Because of the cyclical nature of $\ell^2(\mathbb{Z}_N)$, $\hat{z}(N-1)$ and $\hat{z}(0)$ are considered to be consecutive.)*

Proof. In the following we omit the subscript N in ω_N for readability.

Let $\{\tau_j\}$ be the set of indices such that $z(\tau_j) \neq 0$, $j = 1,2,...,N_t$, and let $a_j = z(\tau_j)$ be the corresponding nonzero values. Now consider some interval $m+1, m+2, ..., m+N_t$ of indices of $\hat{z}$. We want to show that $b = (\hat{z}(m+1), \hat{z}(m+2), ..., \hat{z}(m+N_t)) \neq \mathbf{0}$.

When we make the multiplication $W_N z$, since $z(n) = 0$ if $n \neq \tau_j$, we don't need to multiply by the columns in W_N that only multiply with zeros. The formulas for $\hat{z}(m+k)$ are

$$\begin{aligned}
\hat{z}(m+1) &= a_1\omega^{\tau_1(m+1)} + a_2\omega^{\tau_2(m+1)} + ... + a_{N_t}\omega^{\tau_{N_t}(m+1)},\\
\hat{z}(m+2) &= a_1\omega^{\tau_1(m+2)} + a_2\omega^{\tau_2(m+2)} + ... + a_{N_t}\omega^{\tau_{N_t}(m+2)},\\
&\vdots\\
\hat{z}(m+N_t) &= a_1\omega^{\tau_1(m+N_t)} + a_2\omega^{\tau_2(m+N_t)} + ... + a_{N_t}\omega^{\tau_{N_t}(m+N_t)}.
\end{aligned}$$

We can rewrite this as a matrix multiplication:

$$b = \begin{bmatrix}
(\omega^{\tau_1})^{m+1} & (\omega^{\tau_2})^{m+1} & \dots & (\omega^{\tau_{N_t}})^{m+1}\\
(\omega^{\tau_1})^{m+2} & (\omega^{\tau_2})^{m+2} & \dots & (\omega^{\tau_{N_t}})^{m+2}\\
\vdots & \vdots & \dots & \vdots\\
(\omega^{\tau_1})^{m+N_t} & (\omega^{\tau_2})^{m+N_t} & \dots & (\omega^{\tau_{N_t}})^{m+N_t}
\end{bmatrix} a.$$

This matrix is a N_t by N_t square matrix, and to show that $b \neq \mathbf{0}$ is equivalent to showing that the matrix is invertible since a is nonzero in all places. We may divide each column j by $(\omega^{\tau_j})^{m+1} \neq 0$ and get the matrix

$$\begin{bmatrix} 1 & 1 & \dots & 1 \\ \omega^{\tau_1} & \omega^{\tau_2} & \dots & \omega^{\tau_3} \\ \omega^{2\tau_1} & \omega^{2\tau_2} & \dots & \omega^{2\tau_3} \\ \vdots & \vdots & \dots & \vdots \\ \omega^{(N_t-1)\tau_1} & \omega^{(N_t-1)\tau_2} & \dots & \omega^{(N_t-1)\tau_{N_t}} \end{bmatrix},$$

which we recognize as a transposed Vandermonde matrix. Like we observed earlier for the matrix W_N, this means the determinant is nonzero, so $b \neq \mathbf{0}$. □

Theorem 2.3. *(The Discrete Time Uncertainty Principle). For $z \in \ell^2(\mathbb{Z}_N)$, if $N_t, N_f > 0$ are the number of nonzero entries in z and $\hat{z}$, respectively, then*

$$N_t \cdot N_f \geq N,$$

which implies

$$N_t + N_f \geq 2\sqrt{N}.$$

Proof. Assume the negation $N_t N_f < N$. We can write $N = kN_t + r$ for some integers k and $r < N_t$. Because $\hat{z}$ is cyclical, imagine tying together index 0 and $N-1$ to form a circle of indices (see Figure 2.4). By Lemma 2.5, $N_f \geq k$ since for every interval of k indices there is at least one nonzero value of $\hat{z}$. Thus $N_f = k$ by our assumption. If $r = 0$, we have $N_t N_f = N$, a contradiction. If $r > 0$, we must add a zero value, but this is clearly not possible since we already proved the result for $r = 0$.

The second relation follows from

$$0 \leq (N_t - N_f)^2 = N_t^2 - 2N_t N_f + N_f^2 = N_t^2 + 2N_t N_f + N_f^2 - 4N_t N_f \implies (N_t + N_f)^2 \geq 4N_t N_f$$

where $N_t N_f \geq N$ by the first relation. □

The theorem says that if we have a great number of zeros in one of z or $\hat{z}$, the amount of zeros in the other must be small. In the extreme case, if z has only one nonzero entry, $\hat{z}$ has no zeros.

Example 2.5. If $z = (1,0,0,0)$, $\hat{z} = (1,1,1,1)$. In Example 2.2, $N_t = 6$ and $N_f = 2$, so $N_t N_f = 12 \geq 8$.

In more general terms, the uncertainty principle extends to the statement that a signal z with few high values, relative to the rest, will transform to a function $\hat{z}$ with many high values. We don't prove this for the DFT, but an analogy can be made with a result for the continuous Fourier transform that involves probability [11]. Say that a function g describes a signal with a normal

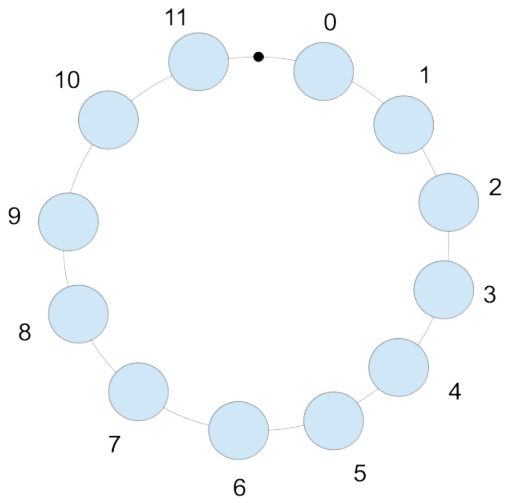


Figure 2.4: The indices of a cyclic function $z \in \ell(\mathbb{Z}_{12})$. 11 and 0 are consecutive indices.

(i.e. Gaussian) probability distribution in time. That is, the probability that the values lie in a certain time interval around the mean μ_t is a function of the standard deviation σ_t. If σ_t is small, then most of the values are concentrated near μ_t so that the exceptional values are few, or in other words, the values are highly localized in a certain range. The uncertainty principle states that if σ_t is small, then the standard deviation of the Fourier transform of g is high, or more precisely,

$$\sigma_t \sigma_f \geq C,$$

for some constant C. This means that the Fourier transform of g is not localized, or that the values are dispersed over a wider range.

3 Wavelets on $\mathbb{Z}_N$

3.1 An Introductory Example

A major reason for developing wavelet theory is data compression. That is, reducing the size of some data set, and preferably as efficiently as possible, meaning that most of the useful information is included in the compressed set of data. A wavelet transformation is a way of expressing a signal in two parts, one that we call the *trend* and one that we call the *fluctuation*. As the names suggest, this split is designed so that the trend represents the basic shape of the signal, while the energy of the fluctuation is small. We begin with an example.

Example 3.1. The Haar transform is one of the simplest examples of a wavelet transform. We define the *first-stage Haar wavelet basis* as the $\ell^2(\mathbb{Z}_N)$ functions

$$\begin{aligned}
X^1 &= \left(\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, 0, 0, ..., 0\right), \\
X^2 &= \left(0, 0, \frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, 0, 0, ..., 0\right), \\
&\vdots \\
X^{N/2} &= \left(0, 0, ..., 0, \frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}\right), \\
Y^1 &= \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0, 0, ..., 0\right), \\
Y^2 &= \left(0, 0, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0, 0, ..., 0\right), \\
&\vdots \\
Y^{N/2} &= \left(0, 0, ..., 0, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right),
\end{aligned}$$

for even N. Theorem 3.1 in Section 3.2.1 will show that this is an orthonormal basis in $\ell^2(\mathbb{Z}_N)$. The first level Haar transform of z can be defined as the operator H_1 that takes a signal z to a the signal (x_1, y_1), where

$$x_1(m) = \langle z, X^m \rangle \quad \text{for } m = 1,2,...,\frac{N}{2}$$
$$y_1(m) = \langle z, Y^m \rangle \quad \text{for } m = 1,2,...,\frac{N}{2}.$$

If we examine the functions x_1 and y_1, we see that the elements of y_1 is a scaled average of two adjacent values of z. The functions Y^k are constructed to produce an approximation of z with half the data points of z. The smaller fluctuations that are lost in the approximation are recorded in x_1. In Figure 3.1 (a) we see the graph of the function from Example 2.3 defined by $z \in \ell^2(\mathbb{Z}_{1024})$ such that

$$z(n) = \sin(\frac{2\pi n}{1024}) + \sin(\frac{12\pi n}{1024}) + \sin(\frac{32\pi n}{1024}) + \frac{1}{5}\sin(\frac{150\pi n}{1024})$$

with added noise around $n = 300$ and $n = 500$ (see Chapter 4 for details). Again, we have connected the discrete values with a line. Figure 3.1 (b) shows the first-stage Haar transform of z, where we see that the basic shape is intact in the trend function y_1, the first 512 values, while some of the fluctuations are stored in x_1, the last 512 values. In fact, roughly 99,5% of the total energy of z ($||z||^2 \approx 1580$) is concentrated in y_1, even though we have compressed the signal z to half its original size. The process is then iterated on the trend function y_1, and after the third level we have a very rough approximation of z in y_3. The energy of the trend of the third level Haar transform is still about 97.1% of the total.

We show below that the Haar transform conserves the energy of any signal:

Claim 3.1. *If H_1 is the first-stage Haar Transform operator in $\ell^2(\mathbb{Z}_N)$ as defined above, then*

$$||H_1(z)||^2 = ||z||^2.$$

Proof. By definition of H_1,

$$\begin{aligned}
||H_1(z)||^2 =& ||(x_1, y_1)||^2 = \frac{(z(0)-z(1))^2}{2} + ... + \frac{(z(N-2)-z(N-1))^2}{2} \\
&+ \frac{(z(0)+z(1))^2}{2} + ... + \frac{(z(N-2)+z(N-1))^2}{2} \\
=& \frac{1}{2}\sum_{k=0}^{N/2-1} z(2k)^2 - 2z(2k)z(2k+1) + z(2k+1)^2 \\
&+ \frac{1}{2}\sum_{k=0}^{N/2-1} z(2k)^2 + 2z(2k)z(2k+1) + z(2k+1)^2 \\
=& \sum_{j=0}^{N-1} z(j)^2 = ||z||^2.
\end{aligned}$$

□

Another key property of the Haar transformation is that it has an inverse transformation, which allows us to completely reconstruct z.

Claim 3.2. *The inverse of H_1 is given by*

$$H_1^{-1}(x_1,y_1)(n) = \begin{cases} \frac{y_1(n)+x_1(n)}{\sqrt{2}} & \text{if } n = 0,2,4,...,N-2 \\ \frac{y_1(n)-x_1(n)}{\sqrt{2}} & \text{if } n = 1,3,5,...,N-1 \end{cases}.$$

Proof. For even n

$$\frac{y_1(n)+x_1(n)}{\sqrt{2}} = \frac{z(n)+z(n+1)}{2} + \frac{z(n)-z(n+1)}{2} = z(n),$$

and odd n

$$\frac{y_1(n)-x_1(n)}{\sqrt{2}} = \frac{z(n-1)+z(n)}{2} - \frac{z(n-1)-z(n)}{2} = z(n).$$

□

Again we see the usefulness of working with signals of size $N = 2^p$. In this case, we can repeat the process of taking invertible Haar transforms 10 times, since $1024 = 2^{10}$, and then be able to reconstruct z with the amount of detail we would like.

3.2 Construction of the Wavelet Bases

In this section we give the construction of a general wavelet basis for $\ell^2(\mathbb{Z}_{2^p})$, and the ultimate goal is to find bases that can be localized in both time and frequency, at least up to the limitations of the uncertainty principle.

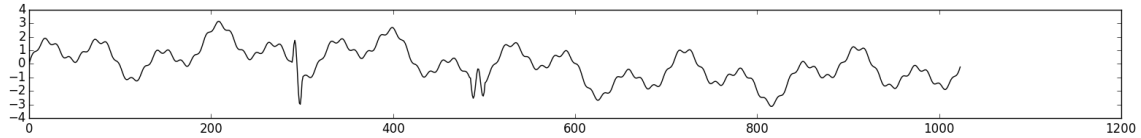
3.2.1 The First-Stage Basis

The conjugate reflection operator will be important going forward.

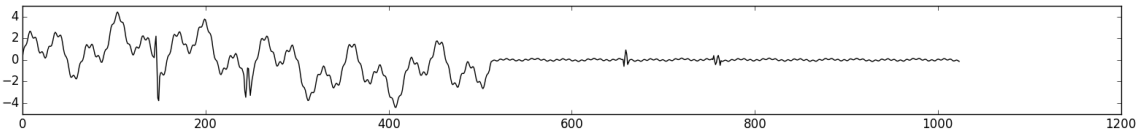
Definition 3.1. *For any $z \in \ell^2(\mathbb{Z}_N)$, define the* conjugate reflection $\tilde{z} \in \ell^2(\mathbb{Z}_N)$ *by*

$$\tilde{z}(n) = \overline{z(-n)} = \overline{z(N-n)}$$

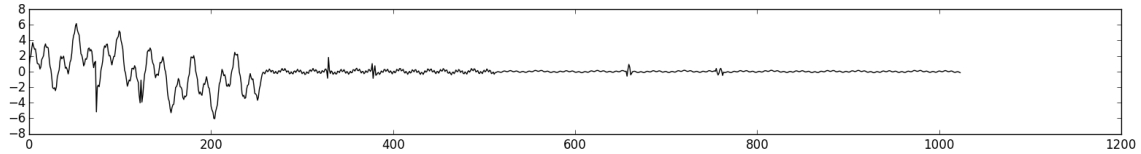
for all n.



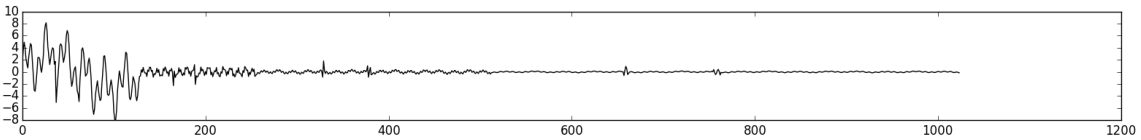
(a) z



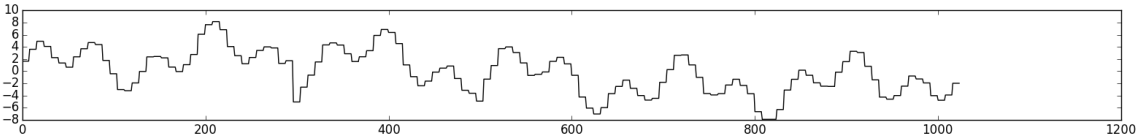
(b) 1st level Haar transform



(c) 2nd level Haar transform



(d) 3rd level Haar transform



(e) Compressed signal

Figure 3.1: A function z and some transformations using the Haar wavelet basis.

Claim 3.3. *For $z \in \ell^2(\mathbb{Z}_N)$,*

$$(\tilde{z})\hat{}(m) = \overline{\hat{z}(m)} \tag{3.1}$$

Proof. Using the definition,

$$(\tilde{z})\hat{}(m) = \sum_{n=0}^{N-1} \tilde{z}(n)\omega_N^{mn} = \sum_{n=0}^{N-1} \overline{z(-n)}\omega_N^{mn} = \sum_{k=0}^{N-1} \overline{z(k)}\omega_N^{-mk} = \sum_{k=0} \overline{z(k)\omega_N^{mk}} = \overline{\hat{z}(m)}.$$

□

The reason for its importance is that it connects convolution with inner products in the following way.

Lemma 3.1. *Suppose $z, w \in \ell^2(\mathbb{Z}_N)$. For any $k \in \mathbb{Z}_N$,*

$$z * \tilde{w}(k) = \langle z, R_k w \rangle \tag{3.2}$$

and

$$z * w(k) = \langle z, R_k \tilde{w} \rangle. \tag{3.3}$$

Proof. By definition,

$$\begin{aligned} \langle z, R_k w \rangle &= \sum_{n=0}^{N-1} z(n)\overline{R_k w(n)} = \sum_{n=0}^{N-1} z(n)\overline{w(n-k)} \\ &= \sum_{n=0}^{N-1} z(n)\tilde{w}(k-n) = \tilde{w} * z(k). \end{aligned}$$

We need to show that convolution is commutative. By definition of convolution,

$$\begin{aligned} x * y(m) &= x(m)y(0) + x(m-1)y(1) + \ldots + x(m-(N-1))y(N-1) \\ y * x(m) &= y(m)x(0) + y(m-1)x(1) + \ldots + y(m-(N-1))x(N-1). \end{aligned}$$

We realize that for any m, $x * y(m) = y * x(m)$, so we conclude equation (3.2). If we replace w with $\tilde{w}$ and note that $\tilde{\tilde{w}} = w$, we get (3.3) and we are done. □

Lemma 3.1 might provide a good starting point in the search of a good basis. Since the coefficients of z in an orthonormal basis are the projections on the base vectors, i.e. scalar products, an orthonormal basis on the form $B = \{R_k w\}_{k=0}^{N-1}$ would make for simple calculations since

$$[z]_B(m) = \langle z, R_m w \rangle = z * \tilde{w}(m),$$

which as we know by now is quickly computed using the FFT. Also, such a basis, like the standard basis, can be localized in time. But as we might suspect, the uncertainty principle will not allow this basis to be localized in frequency whatsoever, as the following lemma proves.

Lemma 3.2. *Let $w \in \ell^2(\mathbb{Z}_N)$. Then $\{R_k w\}_{k=0}^{N-1}$ is an orthonormal basis for $\ell^2(\mathbb{Z}_N)$ if and only if $|\hat{w}(n)| = 1$ for all $n \in \mathbb{Z}_N$.*

Proof. $\{R_k w\}_{k=0}^{N-1}$ is an orthonormal basis if and only if $\langle w, R_k w\rangle = 1$ for $k = 0$ and 0 otherwise. By (3.2), $\langle w, R_k w\rangle = w * \tilde{w}(k)$. Therefore $w * \tilde{w}(k) = e_0$. A simple calculation shows that $\hat{e}_0 = (1,1,...,1)$, so we have

$$1 = \hat{e}_0(n) = (w * \tilde{w})\hat{}(n) = \hat{w}(n)(\tilde{w})\hat{}(n) = \hat{w}(n)\overline{\hat{w}(n)} = |\hat{w}(n)|^2,$$

where we used equation (3.1). □

Now we know that the goal of finding a basis that is localized in time and frequency can't be obtained using such a basis. As we will see, the following type of basis is a much better candidate, which we will discuss later on. This is the definition of a first-stage wavelet basis.

Definition 3.2. *Suppose $N = 2M$, $M \in \mathbb{N}$. An orthonormal basis for $\ell^2(\mathbb{Z}_N)$ of the form*

$$\{R_{2k} v\}_{k=0}^{M-1} \cup \{R_{2k} w\}_{k=0}^{M-1}$$

is called a first-stage wavelet basis *for $\ell^2(\mathbb{Z}_N)$. We call v the* scaling function *and w the* wavelet function*. Sometimes v and w are called the* father *and* mother *wavelet, respectively.*

We need two more lemmas before we can prove a theorem that classifies all wavelet bases.

Lemma 3.3. *Suppose $N = 2M$, $M \in \mathbb{N}$, and $z \in \ell^2(\mathbb{Z}_N)$. Define $z^\star \in \ell^2(\mathbb{Z}_N)$ by*

$$z^\star(n) = (-1)^n z(n) \quad \textit{for all } n.$$

Then

$$(z^\star)\hat{}(n) = \hat{z}(n+M) \quad \textit{for all } n. \tag{3.4}$$

Proof. Using the definitions,

$$\begin{aligned}(z^\star)\hat{}(n) &= \sum_{k=0}^{N-1} z^\star(k) e^{2\pi i k n/N} = \sum_{k=0}^{N-1} (-1)^k z(k) e^{2\pi i k n/N} \\ &= \sum_{k=0}^{N-1} z(k) e^{\pi i k} e^{2\pi i k n/N} = \sum_{k=0}^{N-1} z(k) e^{\pi i k \frac{2M}{2M}} e^{2\pi i k n/N} \\ &= \sum_{k=0}^{N-1} z(k) e^{-2\pi i k(n+M)/N} = \hat{z}(n+M).\end{aligned}$$

□

Remark 3.1. *The operator $\star$ is defined so that $(z+z^\star)$ singles out the even index values of z, since $(z+z^\star)(n) = 2z(n)$ if n is even and 0 otherwise.*

Lemma 3.4. *Suppose $N = 2M$, $M \in \mathbb{N}$, and $w \in \ell^2(\mathbb{Z}_N)$. Then $\{R_{2k}w\}_{k=0}^{M-1}$ is an orthonormal set with M distinct elements if and only if*

$$|\hat{w}(n)|^2 + |\hat{w}(n+M)|^2 = 2 \quad \textit{for } n = 0,1,2,...,M-1.$$

Proof. The condition for orthonormality is $\langle w, R_{2k}w\rangle$=1 if $k = 0$ and 0 otherwise. Using equation (3.2) we need to show when:

$$\langle w, R_{2k}w\rangle = w * \tilde{w}(2k) = \begin{cases} 1 & \text{if } k = 0 \\ 0 & \text{if } k = 1,2,...,M-1. \end{cases} \tag{3.5}$$

By the definition of $\star$,

$$(w*\tilde{w} + (w*\tilde{w})^\star)(n) = \begin{cases} 2w*\tilde{w}(n) & \text{if } n \text{ is even} \\ 0 & \text{if } n \text{ is odd.} \end{cases} \tag{3.6}$$

The cases where n is even give us a new condition for orthonormality:

$$(w*\tilde{w} + (w*\tilde{w})^\star)(2k) = 2w*\tilde{w}(2k) = \begin{cases} 2 & \text{if } k = 0 \\ 0 & \text{if } k = 1,2,...,M-1. \end{cases}$$

By (3.6), we know that $2w*\tilde{w}(n) = 0$ for odd n, so condition (3.5) is equivalent to

$$w*\tilde{w} + (w*\tilde{w})^\star = 2e_0.$$

Since $\hat{e}_0 = (1,1,...,1)$, (3.5) is equivalent to

$$(w*\tilde{w})\hat{}(n) + ((w*\tilde{w})^\star)\hat{}(n) = 2 \quad \text{for all } n, \tag{3.7}$$

by linearity of the DFT. By the equalities (3.1) and (2.2), and the fact that $z\bar{z} = |x|^2$ for $z \in \mathbb{C}$,

$$(w*\tilde{w})\hat{}(n) = \hat{w}(n)(\tilde{w})\hat{}(n) = \hat{w}(n)\overline{\hat{w}(n)} = |\hat{w}(n)|^2.$$

The other term in (3.7) is, using the result directly above and Lemma 3.3,

$$((w*\tilde{w})^\star)\hat{}(n) = (w*\tilde{w})\hat{}(n+M) = |\hat{w}(n+M)|^2.$$

Now we have (3.7) as $|\hat{w}(n)|^2 + |\hat{w}(n+M)|^2 = 2$ for all n. Since $\hat{w}$ has period $N = 2M$ and

$$|\hat{w}(n+M)|^2 + |\hat{w}(n+M+M)|^2 = |\hat{w}(n+M)|^2 + |\hat{w}(n)|,$$

the condition in the lemma is equivalent to (3.5). □

Now we are ready to classify the first-stage wavelet bases.

Theorem 3.1. *Suppose $N = 2M$, $M \in \mathbb{N}$, and let $v, w \in \ell^2(\mathbb{Z}_N)$. Then*

$$\begin{aligned} B &= \{R_{2k}v\}_{k=0}^{M-1} \cup \{R_{2k}w\}_{k=0}^{M-1} \\ &= \{v, R_2v, R_4v, ..., R_{N-2}v, w, R_2w, R_4w, ..., R_{N-2}w\} \end{aligned}$$

is an orthonormal basis if and only if

$$\begin{aligned} |\hat{v}(n)|^2 + |\hat{v}(n+M)|^2 &= 2, \\ |\hat{w}(n)|^2 + |\hat{w}(n+M)|^2 &= 2, \end{aligned}$$

and

$$\hat{v}(n)\overline{\hat{w}(n)} + \hat{v}(n+M)\overline{\hat{w}(n+M)} = 0,$$

for all $n = 0, 1, 2, ..., M-1$.

Proof. By Lemma 3.4, $\{R_{2k}v\}_{k=0}^{M-1}$ and $\{R_{2k}w\}_{k=0}^{M-1}$ are orthogonal sets if and only if the first two equations hold. The final claim is that the last equation is equivalent to $\langle R_{2k}v, R_{2j}w\rangle$ for all $k, j \in \mathbb{Z}_N$. This expression is equivalent to

$$\langle v, R_{2j}w\rangle = 0 \quad \text{for all } j \in \mathbb{Z}_N$$

since all the possible combinations of terms $R_{2k}v(n)R_{2j}\overline{w(n)}$ in the inner products $\langle R_{2k}v, R_{2j}w\rangle$ are included in the above. This in turn is equivalent to

$$v * \tilde{w}(2k) = 0 \quad \text{for all } k \in \mathbb{Z}_N$$

by equation (3.2). By the definition of $\star$, this is equivalent to

$$v * \tilde{w} + (v * \tilde{w})^\star = 0.$$

Since the DFT of 0 is 0, we have the condition

$$(v * \tilde{w})\hat{} + ((v * \tilde{w})^\star)\hat{} = 0$$

for orthonormality. By formula (2.2)

$$(v * \tilde{w})\hat{}(n) = \hat{v}(n)\overline{w(n)},$$

and by Lemma 3.3

$$((v * \tilde{w})^\star)\hat{}(n) = \hat{v}(n+M)\overline{\hat{w}(n+M)}.$$

Since the expression in the theorem is periodic with period M, it is true if and only if it is true for all n, thus we have shown equivalence. □

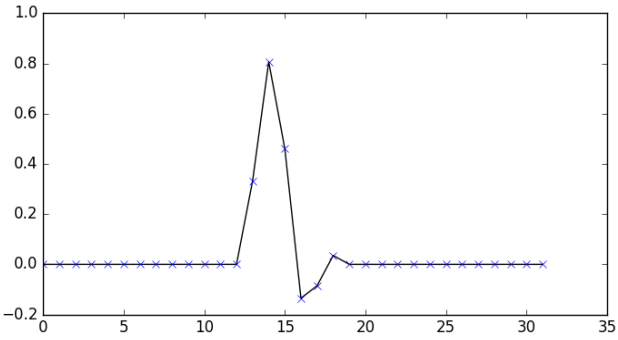
If we inspect the conditions of Theorem 3.1, we see that we might be able to get some localization in frequency, since only the sums $|\hat{v}(n)|^2 + |\hat{v}(n+M)|^2$ and $|\hat{w}(n)|^2 + |\hat{w}(n+M)|^2$ need to be constant, as opposed to the conditions for the basis $\{R_k w\}$ in Lemma 3.2. If for example $|\hat{v}(n)| = 2$ for $n = N/2$, the coefficient associated with the highest frequency, the lowest frequency component $\hat{v}(0)$ will be 0. It turns out that one is able to find many different functions v and w that are also more or less localized in time, so wavelets are a good approach to tackling the "problem" of the uncertainty principle, i.e. optimizing localization in time and frequency simultaneously.

Example 3.2. One example of a well known wavelet basis is due to Ingrid Daubechies, a pioneer in the subject, after whom a whole family of wavelet bases has been named. In Figure 3.2 (a) we see the scaling function v_1 (translated by 13 for centering in the graph) for the basis $D6 \in \ell^2(\mathbb{Z}_{32})$, and in Figure 3.2 (b) we have the wavelet function w_1, also translated by 13. We see that these functions are localized in time quite well. In Figure 3.2 (c) and (d) we show the absolute values of the discrete Fourier transforms of the same functions, which show some localization in frequency also; the scaling function does not contain the highest frequency components, and the wavelet function contains mainly high frequency components. This basis is named $D6$ since there are six nonzero values in v_1 and w_1, respectively. The family of Daubechies wavelet bases also includes $D10$ and $D20$, for example, for which we add more nonzero values in v_1 and w_1 to get more localization in frequency as we lose localization in time. Thus one can tweak the level of localization depending on the problem at hand.

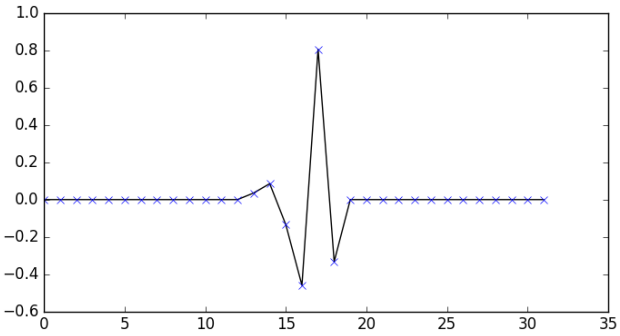
Remark 3.2. *In terms of the uncertainty principle, we can count the number of nonzero values in v_1, w_1 and their respective Fourier transforms. There are 6 nonzero values in both v_1 and w_1, so we should expect at least 6 nonzero values (since $6 \cdot 6 \geq 32$) in $\hat{v}_1$ and $\hat{w}_1$. Obviously, there are more than that, the reason being that the values of v_1 and w_1 are concentrated in a short range of time. In Lemma 2.5 we showed that if z has 6 consecutive nonzero values, $\hat{z}$ cannot have 6 consecutive zero values. In fact we see exactly this, $\hat{v}_1$ and $\hat{w}_1$ both have 5 consecutive zeros. Since the IDFT is connected to the DFT by conjugation of the transformation matrix, one can also show that Lemma 2.5 holds for the reverse direction. I.e. if we want the 6 nonzero values of v_1 and w_1 to be consecutive, and thus have 26 consecutive zeros, we must have at least 27 nonzero values in $\hat{v}_1$ and $\hat{w}_1$, respectively, which exactly matches what we have. This means that, given the structure of v_1 and w_1, this basis is about as localized as possible.*

3.2.2 Multiresolution Analysis

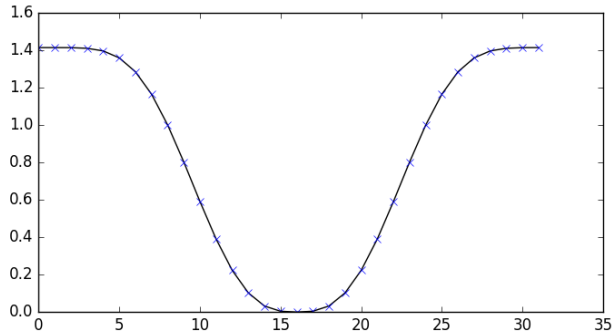
Now we have defined a first-stage wavelet basis, and in Example 3.1 we saw an example of such a basis, the Haar basis, and what it could do. In the same example, we hinted a method of iterating the transform to get multiple levels of fluctuations, or detail, which is the main point



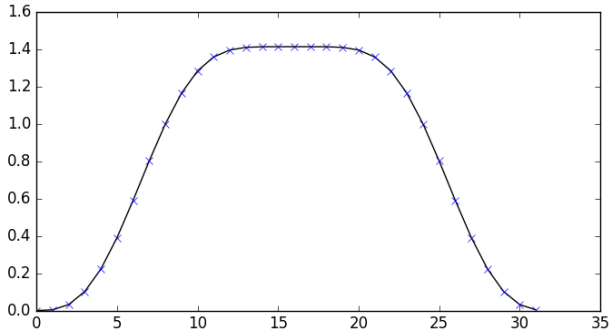
(a) Scaling function $R_{13}v_1$ of $D6 \in \ell^2(\mathbb{Z}_{32})$.



(b) Wavelet function $R_{13}w_1$ of $D6 \in \ell^2(\mathbb{Z}_{32})$.



(c) Values of $|R_{13}\hat{v}_1|$.



(d) Values of $|R_{13}\hat{w}_1|$.

Figure 3.2: Daubechies $D6$ wavelet basis in $\ell^2(\mathbb{Z}_{32})$.

of wavelet transformation. In this section we will give the method of iterating any wavelet transform starting only with a first-stage basis generated by some v_1 and w_1.

To illustrate, we start with a simple example.

Example 3.3. Let's say we are given the signal $z = (5,7,8,6,8,10,11,11)$ in $\ell^2(\mathbb{Z}_8)$. If we start with the first-stage Haar basis of Example 3.1 for $N = 8$, we get the first level Haar transform

$$x_1 = \left(\frac{z(0)-z(1)}{\sqrt{2}}, \frac{z(2)-z(3)}{\sqrt{2}}, \frac{z(4)-z(5)}{\sqrt{2}}, \frac{z(6)-z(7)}{\sqrt{2}}\right) = (-\sqrt{2}, \sqrt{2}, -\sqrt{2}, 0)$$
$$y_1 = \left(\frac{z(0)+z(1)}{\sqrt{2}}, \frac{z(2)+z(3)}{\sqrt{2}}, \frac{z(4)+z(5)}{\sqrt{2}}, \frac{z(6)+z(7)}{\sqrt{2}}\right) = (6\sqrt{2}, 7\sqrt{2}, 9\sqrt{2}, 11\sqrt{2}).$$

To iterate the transform a second time on the trend function x_1, we need a new basis in $\ell^2(\mathbb{Z}_4)$ that does the same thing. The N-the stage Haar basis is easy to obtain using the formulas in Example 3.1; remember, the Haar transform simply approximates a signal with the averages of two neighboring values. Thus we get the next basis using $N = 4$ and the second-stage Haar

transform is

$$x_2 = \left(\frac{-\sqrt{2}-\sqrt{2}}{\sqrt{2}}, \frac{-\sqrt{2}}{\sqrt{2}} \right) = (-2,-1)$$
$$y_2 = \left(\frac{\sqrt{2}(6+7)}{\sqrt{2}}, \frac{\sqrt{2}(9+11)}{\sqrt{2}} \right) = (13,20).$$

In this case we could iterate the transform one last time, which gives us

$$x_3 = \left(-\frac{1}{\sqrt{2}} \right)$$
$$y_3 = \left(\frac{13+20}{\sqrt{2}} \right) = (16.5\sqrt{2}).$$

What we end up with is a complete representation of z as $\{x_1, x_2, x_3, y_3\}$. By Claim 3.2, we can perfectly reconstruct z using this representation. Note, however, that we have not yet shown that this representation is a vector in an $\ell^2(\mathbb{Z}_8)$ basis, which is why we denote it as a set for now.

So what use is this to us? Well, depending on what kind of information we are looking for, we might be able to use only part of the transformed signal. If the interest is only in whether the signal is nonzero or zero, $y_3 = (16.5\sqrt{2})$ will suffice. If we want to know whether the initial value is higher or lower than the last value, y_2 is enough, which is the inverse of (x_3, y_3) by Claim 3.2. If we then invert (x_2, y_2), which gives us y_1, we get a slightly better idea of the basic shape of z, in which case the signal is compressed to half its original size.

In this way, we can choose to look at the signal z on three different scales, or resolutions. The formalization of this method, which we give in Definition 3.4, is therefore called *multiresolution analysis* (MRA). For now, we describe the method, and in Section 3.2.3 we give the justification for using it.

Definition 3.3. *Suppose $N = 2M$, $M \in \mathbb{N}$. Let $z \in \ell^2(\mathbb{Z}_N)$ and $w \in \ell^2(\mathbb{Z}_M)$. The operator $D : \ell^2(\mathbb{Z}_N) \to \ell^2(\mathbb{Z}_M)$ defined by*

$$D(z) = (z(0), z(2), ..., z(N-2))$$

is called the downsampling *operator. The operator $U : \ell^2(\mathbb{Z}_M) \to \ell^2(\mathbb{Z}_N)$ defined by*

$$U(w) = (w(0), 0, w(1), 0, ..., w(N-1), 0)$$

is called the upsampling *operator.*

Definition 3.4. *(Multiresolution Analysis). Suppose $N = 2^p$. A p^{th}-stage* wavelet filter sequence *is a sequence of vectors $v_1, w_1, v_2, w_2, ..., v_p, w_p$ such that for each pair $v_k, w_k \in \ell^2(\mathbb{Z}_{N/2^{k-1}})$*

where $k = 1, 2, ..., p$, the conditions in Theorem 3.1 hold. For an input $z \in \ell^2(\mathbb{Z}_N)$, define

$$x_1 = D(z * \tilde{w}_1) \in \ell^2(\mathbb{Z}_{N/2})$$

and

$$y_1 = D(z * \tilde{v}_1) \in \ell^2(\mathbb{Z}_{N/2}).$$

The following vectors are then defined inductively:

$$x_k = D(y_{k-1} * \tilde{w}_k) \in \ell^2(\mathbb{Z}_{N/2^k})$$

and

$$y_k = D(y_{k-1} * \tilde{v}_k) \in \ell^2(\mathbb{Z}_{N/2^k})$$

for $k = 2, 3, ..., p$. Then the set $\{x_1, x_2, ..., x_p, y_p\}$ is called the output *of the multiresolution analysis of z.*

In words, we get each new stage of the transform by operating on the vector y_k. Note that this method is exactly the one we used in Examples 3.1 and 3.3. By equation (3.2), and since downsampling removes odd index values, this definition of x_1 and y_1 is equivalent to the one we gave in Example 3.1 where we used scalar products instead.

Remark 3.3. *Note that this general definition makes it clear that MRA is an efficient method, since by Theorem 2.1 we can use the FFT to compute convolution quickly. In fact, it can be shown that the number of complex multiplications needed to compute the output is no greater than $4N + N\log_2 N$ [2, Lemma 3.17].*

3.2.3 The p^{th}-Stage Basis

In this section we give the justification for using the algorithm in Definition 3.4. We want to show that the process is reversible and that the output set $\{x_1, x_2, ..., x_p, y_p\}$ is in fact a representation of z in an orthonormal basis in $\ell^2(\mathbb{Z}_{2^p})$. In other words, we want to show that the MRA is a change of basis to what we call the *$p^{th}-$stage wavelet basis*. This will be done by formalizing the MRA steps as operators and examining the properties of these operators.

Definition 3.5. *For a wavelet filter sequence $v_1, w_1, v_2, w_2, ..., v_p, w_p$, define the subspaces of $\ell^2(\mathbb{Z}_{2^p})$*

$$V_1 = \operatorname{span}\{R_{2k} v_1\}_{k=0}^{2^{p-1}}$$

and

$$W_1 = \operatorname{span}\{R_{2k} w_1\}_{k=0}^{2^{p-1}}.$$

If we are given a wavelet filter sequence, then each pair v_1 and w_1 satisfy the conditions in

Theorem 3.1 and generate orthogonal subspaces of $\ell^2(\mathbb{Z}_{2^p})$ so we have

$$\ell^2(\mathbb{Z}_{2^p}) = W_1 \oplus V_1.$$

The following lemma, concerning projections on these subspaces, will be used in proving that MRA is invertible in general.

Lemma 3.5. *Suppose $N = 2^p$. Let P_{V_1} and P_{W_1} be the orthogonal projections on the spaces V_1 and W_1, respectively. Then for any $z \in \ell^2(\mathbb{Z}_N)$,*

$$v_1 * U(D(z * \tilde{v}_1)) = P_{V_1} z$$

and

$$w_1 * U(D(z * \tilde{w}_1)) = P_{W_1} z.$$

Proof. By the linearity of the convolution operator, U and D, respectively, we can write

$$\begin{aligned} v_1 * U(D(z * \tilde{v}_1)) &= v_1 * U(D(\sum_{k=0}^{N-1} \langle z, R_k v_1 \rangle e_k)) = \sum_{k=0}^{N-1} \langle z, R_k v_1 \rangle v_1 * U(D(e_k)) \\ &= \sum_{j=0}^{N/2-1} \langle z, R_{2j} v_1 \rangle (v_1 * e_{2j}) = \sum_{j=0}^{N/2-1} \langle z, R_{2j} v_1 \rangle R_{2j} v_1 \end{aligned}$$

using equation (3.2). We recognize this as the orthogonal projection on V_1. The proof for w_1 is similar. □

Now we define the first level MRA operator and its inverse.

Definition 3.6. *Define the operators $\mathscr{V}_1 : \ell^2(\mathbb{Z}_{2^p}) \to \ell^2(\mathbb{Z}_{2^{p-1}})$ and $\mathscr{W}_1 : \ell^2(\mathbb{Z}_{2^p}) \to \ell^2(\mathbb{Z}_{2^{p-1}})$ by*

$$\mathscr{V}_1(z) = D(z * \tilde{v}_1), \quad \mathscr{W}_1(z) = D(z * \tilde{w}_1)$$

Definition 3.7. *Define the operators $\mathscr{V}_1^{-1} : \ell^2(\mathbb{Z}_{2^{p-1}}) \to V_1$ and $\mathscr{W}_1^{-1} : \ell^2(\mathbb{Z}_{2^{p-1}}) \to W_1$ by*

$$\mathscr{V}_1^{-1}(z) = v_1 * U(z), \quad \mathscr{W}_1^{-1}(z) = w_1 * U(z)$$

Lemma 3.6. *The operator $T_1 : W_1 \oplus V_1 \to \ell^2(\mathbb{Z}_{2^{p-1}}) \oplus \ell^2(\mathbb{Z}_{2^{p-1}})$ defined by*

$$T_1(z) = (\mathscr{W}_1(z), \mathscr{V}_1(z)) = (D(z * \tilde{w}_1) | D(z * \tilde{v}_1)) \tag{3.8}$$

is a linear isomorphism. That is, $W_1 \oplus V_1 \simeq \ell^2(\mathbb{Z}_{2^{p-1}}) \oplus \ell^2(\mathbb{Z}_{2^{p-1}})$.

Proof. By linearity of the operators involved, $\mathscr{V}_1$ is a linear operator. If we apply $\mathscr{V}_1$ to $R_k w_1$ for any k, we get $D(R_k w_1 * \tilde{v}_1) = 0$ since $R_k w_1$ and v_1 are orthogonal by definition of the wavelet

filter sequence. Then by linearity, $\mathscr{V}_1(x) = 0$ for $x \in W_1$. Hence

$$\mathscr{V}_1(z) = \mathscr{V}_1(P_{V_1}z) + \mathscr{V}_1(P_{W_1}z) = \mathscr{V}_1(P_{V_1}z).$$

By Lemma 3.5, $\mathscr{V}_1^{-1} \circ \mathscr{V}_1(z) = P_{V_1}z$. By the equality above,

$$\mathscr{V}_1^{-1} \circ \mathscr{V}_1(z) = \mathscr{V}_1^{-1} \circ \mathscr{V}_1(P_{V_1}z) = P_{V_1}z,$$

so $\mathscr{V}_1^{-1} \circ \mathscr{V}_1$ restricted to V_1 is the identity operator. Now we want to show that $\mathscr{V}_1 \circ \mathscr{V}_1^{-1}$ is also the identity. By commutativity and associativity of convolution,

$$\begin{aligned}\mathscr{V}_1 \circ \mathscr{V}_1^{-1}(z) &= \mathscr{V}_1(v_1 * U(z)) = D((v_1 * U(z)) * \tilde{v}_1) \\ &= D(v_1 * \tilde{v}_1 * U(z)) = D(v_1 * \tilde{v}_1) * z.\end{aligned}$$

Using equation (3.2), $v_1 * \tilde{v}_1(k) = \langle v_1, R_k v_1 \rangle$, so $v_1 * \tilde{v}_1$ is equal to e_0 by construction of v_1. Therefore $\mathscr{V}_1 \circ \mathscr{V}_1^{-1}(z) = e_0 * z = z$, so we have proved that $\mathscr{V}_1$ is a linear isomorphism. Similarly, one can show that the analogous results hold for $\mathscr{W}_1$ and $\mathscr{W}_1^{-1}$. □

This lemma shows that we can reverse any first-stage wavelet transform. After completing the first transform, v_2 and w_2 generate a new orthonormal basis of $\ell^2(\mathbb{Z}_{2^{p-1}})$ which allows us to iterate the process on the signal y_1 in particular and apply the same results. Thus, we can get a reversible representation

$$\{x_1, x_2, ..., x_p, y_p\}$$

for any $z \in \ell^2(\mathbb{Z}_{2^p})$.

However, the reason why we show this as a set instead of a vector is that we still haven't shown that it is given in an orthonormal basis in $\ell^2(\mathbb{Z}_{2^p})$. For example, the vectors x_p and y_p are one-dimensional, since they are downsampled p times, so we can't compute with these sets at the moment. We would very much like to do so without going back and forth to the standard basis vector z. The rest of the section will show that the the set $\{x_1, x_2, ..., x_p, y_p\}$ is in fact a vector with coefficients in an orthonormal basis in $\ell^2(\mathbb{Z}_{2^p})$.

Lemma 3.7. *The operator T_1 preserves scalar products. In particular, it preserves angles and distance.*

Proof. First we show that $\mathscr{V}_1$ preserves scalar products. Since $\{R_{2k}v_1\}$ is an orthonormal basis for V_1 and scalar product is distributive, i.e. $\langle z, x+y \rangle = \langle z, x \rangle + \langle z, y \rangle$, we only need to prove the property for the basis vectors. So we want to show that

$$\langle R_{2k}v_1, R_{2j}v_1 \rangle_{\ell^2(\mathbb{Z}_{2^p})} = \langle D(R_{2k}v_1 * \tilde{v}_1), D(R_{2j}v_1 * \tilde{v}_1) \rangle_{\ell^2(\mathbb{Z}_{2^{p-1}})}.$$

The left hand side is clearly 1 if $k = j$ and 0 otherwise. By equation (3.2), $R_{2k}v_1 * \tilde{v}_1 = e_{2k}$. The downsampling of this vector is e_k. So the right hand side is equal to $\langle e_k, e_j \rangle$ which is 1 if $k = j$

and 0 otherwise which makes the equality true. Similarly we can prove that T_1'' preserves scalar product.

Since we have the orthogonal spaces V_1 and W_1 in $\ell^2(\mathbb{Z}_{2^p})$, we can calculate $\langle x,y\rangle$ by $\langle P_{V_1}x, P_{V_1}y\rangle + \langle P_{W_1}x, P_{W_1}y\rangle$. The result then follows from the argument above since T_1 restricted to V_1 is $\mathscr{V}_1$ and T_1 restricted to W_1 is $\mathscr{W}_1$. □

So, if we have two wavelet filters v_k and w_k that are orthogonal by construction, then we can get orthogonal vectors in $\ell^2(\mathbb{Z}_{2^p})$ by taking the pullback of these vectors with the inverse MRA operations $\mathscr{V}_k^{-1}$ that are defined similarly to $\mathscr{V}_1^{-1}$ from Definition 3.7. We will now show that this approach generates an orthonormal basis in $\ell^2(\mathbb{Z}_{2^p})$, and moreover that the coefficients of z in this basis are exactly the output values of the MRA of z.

Now, say we have a wavelet filter sequence $v_1, w_1, v_2, w_2, ..., v_p, w_p$. The vector w_j exists in the space $\ell^2(\mathbb{Z}_{2^{p-j+1}})$, for example w_3 is in $\ell^2(\mathbb{Z}_{2^{p-2}})$. The pull-back of w_3 (see Figure 3.3 for schematic) into $\ell^2(\mathbb{Z}_{2^{p-1}})$ with $\mathscr{V}_2^{-1}$ is

$$\mathscr{V}_2^{-1}(w_3) = v_2 * U(w_3).$$

Then we can apply $\mathscr{V}_1^{-1}$ to get

$$\mathscr{V}_1^{-1}(\mathscr{V}_2^{-1}(w_3)) = v_1 * U(v_2 * U(w_3)) = v_1 * Uv_2 * U^2 w_3 \in \ell^2(\mathbb{Z}_{2^p}).$$

If we do this for "deeper" w_k's until we get all the way back to $\ell^2(\mathbb{Z}_{2^p})$, we will eventually be able to make the following definition.

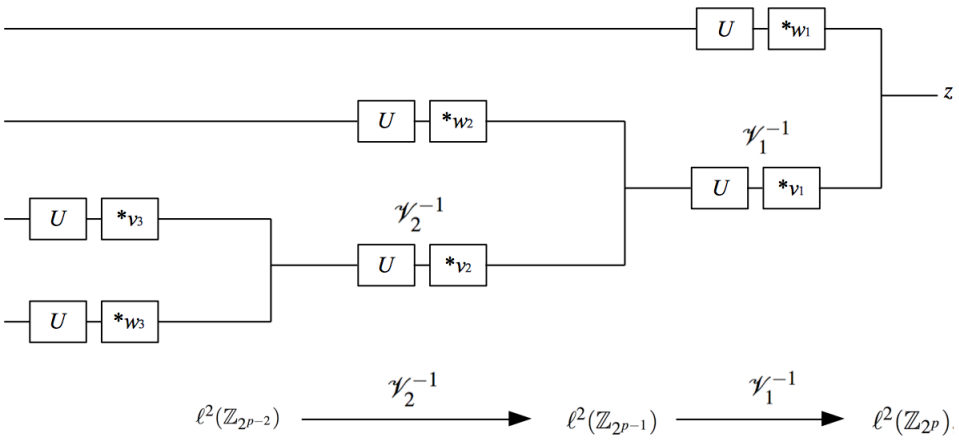


Figure 3.3: Pull-back of vectors into higher dimension spaces using the inverse MRA operators.

Definition 3.8. *Given a wavelet filter sequence (as in Definition 3.4), define*

$$f_k = v_1 * U(v_2) * U^2(v_3) * ... * U^{k-2}v_{k-1} * U^{k-1}w_k$$

and

$$g_k = v_1 * U(v_2) * U^2(v_3) * ... * U^{k-2}v_{k-1} * U^{k-1}v_k$$

for $k = 1, 2, ..., p$.

Or, equivalently, inductively define

$$f_k = g_{k-1} * U^{k-1}(w_k)$$

and

$$g_k = g_{k-1} * U^{k-1}(v_k)$$

for $k = 1, 2, ..., p$.

The pull-back of a translation $R_{2k}w_j$ yields

$$\mathscr{V}_{j-1}^{-1}(R_{2k}w_j) = v_{j-1} * U(R_{2k}w_j) = R_{4k}v_{j-1} * U(w_j)$$

since, quite clearly, $U(R_n x) = R_{2n}U(x)$. Continuing, for each orthonormal set $\{R_{2k}w_j\}$ we have an orthonormal set $\{R_{2^j k}f_j\}$ in $\ell^2(\mathbb{Z}_{2^p})$ by Lemma 3.7 (since $\mathscr{V}_k^{-1}$ preserves angles).

Now we define a general p^{th}-stage wavelet basis.

Definition 3.9. *Let $N = 2^p$. An orthonormal set B on the form*

$$\{R_{2k}f_1\}_{k=0}^{2^{p-1}-1} \cup \{R_{4k}f_2\}_{k=0}^{2^{p-2}-1} \cup ... \cup \{R_{2^{p-1}k}f_{p-1}\}_{k=0}^{1} \cup \{f_p\} \cup \{g_p\}$$

*is called a p^{th}-*stage wavelet basis for $\ell^2(\mathbb{Z}_N)$. *We say that $f_1, f_2, ..., f_p, g_p$* generate *B.*

Lemma 3.8 and its corollary below confirms that the output of the MRA of z can be given as a vector in the basis generated by the vectors defined in Definition 3.8.

Lemma 3.8. *Suppose $N = 2^p$. Given a wavelet filter sequence $\{v_k, w_k\}_{k=1}^p$; if $x_1, x_2, ..., x_p$, $y_1, y_2, ..., y_p$ are defined as in Definition 3.4 and $f_1, f_2, ..., f_p, g_1, g_2, ..., g_p$ as in Definition 3.8, then*

$$x_k = D^k(z * \tilde{f}_k),$$

and

$$y_k = D^k(z * \tilde{g}_k).$$

Proof. First, observe that

$$\tilde{f}_k = (g_{k-1} * U^{k-1}(w_k))\tilde{} = \tilde{g}_{k-1} * (U^{k-1}(w_k))\tilde{} = \tilde{g}_{k-1} * U^{k-1}(\tilde{w}_k) \tag{3.9}$$

since

$$(z * w)\tilde{}(m) = (\overline{z * w})(-m) = (\overline{z} * \overline{w})(-m) = \overline{z}\hat{}(-m)\overline{w}\hat{}(-m) = \tilde{z}\hat{}(m)\tilde{w}\hat{}(m) = (\tilde{z} * \tilde{w})(m).$$

The proof is then done on both x_k and y_k by induction on k. If $k = 1$, then $f_1 = w_1$ and $g_1 = v_1$ and the equalities are true by definition. Now suppose the result holds for $k-1$. Then, by the induction hypothesis, we get

$$\begin{aligned} x_k &= D(y_{k-1} * \tilde{w}_k) = D(D^{k-1}(z * \tilde{g}_{k-1}) * \tilde{w}_k) \\ &= D^k(z * \tilde{g}_{k-1} * U^{k-1}(\tilde{w}_k)) = D^k(z * \tilde{f}_k), \end{aligned}$$

by equality (3.9). The proof for y_k is similar. □

Corollary 3.1. *The above are the coefficients of z in the basis given in Definition 3.9.*

Proof. By equation (3.2),

$$z * \tilde{f}_k(m) = \langle z, R_m f_k \rangle.$$

The k-th level downsampling of this vector has coefficients $\langle z, R_{2^k n} f_k \rangle$ for $n = 0, 1, ..., 2^{p-k} - 1$, which is exactly the projection on the vectors of Definition 3.9. □

This last theorem finally shows that the basis in question indeed is an orthonormal basis in $\ell^2(\mathbb{Z}_{2^p})$.

Theorem 3.2. *Suppose $N = 2^p$. The set*

$$\{R_{2k} f_1\}_{k=0}^{2^{p-1}-1} \cup \{R_{4k} f_2\}_{k=0}^{2^{p-2}-1} \cup ... \cup \{R_{2^{p-1}k} f_{p-1}\}_{k=0}^{1} \cup \{f_p\} \cup \{g_p\}$$

generated by the vectors $f_1, f_2, ..., f_p, g_p$ defined as in Definition 3.8 forms a p^{th}-stage wavelet basis in $\ell^2(\mathbb{Z}_N)$.

Proof. Since the functions f_k and g_k are defined in terms of pull-back of the wavelet filter sequence vectors by the functions $\mathscr{V}_j$ and $\mathscr{W}_j$ and their inverses, which we proved preserve scalar products (and thus lengths of vectors and orthogonality), the sets $\{R_{2^j k} f_j\}_{j=1}^{p}$ and the set $\{R_{2^p k} g_p\}$ are orthonormal sets by assumption of the wavelet filter sequence. Now, $\ell^2(\mathbb{Z}_N)$ is divided into the orthogonal spaces V_1 and W_1. The set $\{R_{2k} f_1\}$ spans W_1 and is orthogonal to all subspaces of V_1 from which all the sets $\{R_{2k} f_{1+j}\}$ and $\{R_{2k} g_{1+j}\}$ are generated. The same reasoning goes for $\{R_{4k} f_2\}_{k=0}^{(N/4)-1}$ that spans W_2 which is orthogonal to V_2, where all the other remaining sets live, and so on. Finally, $\{f_p\} \cup \{g_p\}$ is also orthonormal by Lemma 3.7; it is the pull-back of $w_p \cup v_p$. □

Now we could redefine the MRA (Definition 3.4) to output a vector instead of a set, and we can make vector computations in this new basis since there is a one-to-one correspondence with the standard basis. If we extend Definition 3.5 to define W_k and V_k as the subspaces of $\ell^2(\mathbb{Z}_{2^p})$ spanned by the functions generated by f_k and g_k, we have the following representation:

$$\ell^2(\mathbb{Z}_{2^p}) = W_1 \oplus W_2 \oplus ... \oplus W_p \oplus V_p. \tag{3.10}$$

We get this result by at the k-th stage operating on V_k with T_k, defined similarly to T_1 in (3.8), and on the other spaces by the identity operator.

4 Numerical work

4.1 Examples 2.3 and 3.1

The functions plotted in Figures 2.2 and 3.1 are generated using the code for z in Figure 4.1.

```
z=[]
t=[]
N=1024
for n in range(N):
    z.append(np.sin(2*n*np.pi/N)+np.sin(12*n*np.pi/N)
    +np.sin(32*n*np.pi/N)+0.2*np.sin(150*n*np.pi/N))
    t.append(i)
    if n<=300 and n>=290:
        z[n]+=2*np.sin(600*n/N)
    elif n<=500 and n>=485:
        z[n]+=np.cos(600*n/N)
    else:
        pass
```

Figure 4.1

That is,

$$z(n) = \sin(\frac{2\pi n}{1024}) + \sin(\frac{12\pi n}{1024}) + \sin(\frac{32\pi n}{1024}) + \frac{1}{5}\sin(\frac{150\pi n}{1024})$$

plus $2\sin(600n/1024)$ if n is between 290 and 300, and plus $\cos(600n/1024)$ if n is between 485 and 500. In Figure 4.3 we see the effectiveness of filtering high frequencies using thresholding on the DFT of z. The process is straightforward; we transform the signal using the FFT, and if any value of $\hat{z}$ is below a certain threshold, here we chose $|\hat{z}(m)| \leq 50$, we replace it with zero. The top image shows the imaginary part of $\hat{z}$ (remember the real part contained no spikes in frequency), and the plot below is the filtered signal using the threshold $|\hat{z}| = 50$. This filter method perfectly removed what we labeled as "noise", which essentially could not have been done without knowing the DFT of z. The code for the filter is found if Figure 4.2.

```
filtered_signal=[]
for n in range(1024):
    if np.linalg.norm(np.fft.fft(z)[n])>=50:
        filtered_signal.append(np.fft.fft(z)[n])
    else:
        filtered_signal.append(0)
```

Figure 4.2

In Figure 4.4 we find the implementation of the Haar transform in python code.

4.2 Implementing the FFT

The most famous algorithm using the FFT is the Cooley-Tukey FFT algorithm (named after J.W. Cooley and John Tukey who rediscovered it it in 1965, but in fact its essence was conceived by Gauss 160 years earlier). In Figure 4.5 we see an implementation in python code [6]. As we can see, if the size of the input signal z in the FFT program is greater than 1, then it will be split into two signals U and V that are fed back into the program.

When we run these programs on the 2^{14}-vector $z = [1, 1, ..., 1]$, we see in Figure 4.6 that the FFT algorithm computes $\hat{z}$ in about 0.517 seconds, while the brute force DFT program takes 36.58 seconds to get (essentially) the same result. Even worse, the errors in the results of the DFT algorithm are actually orders of magnitude bigger than the errors of the FFT algorithm. We can conclude this since we would expect only one nonzero component; the input z is a constant function, so the only frequency contributing should be the "zero frequency", i.e. the function E_0, the coefficient of which is $16,384$. Since the FFT requires less computations, and the natural errors of computer calculations accumulate along the way, the FFT is faster *and* better.

Since the IDFT matrix $M_N = \overline{W}_N$ (where we define the line to mean taking the complex conjugate of each entry in W_N) is so closely related to related to W_N, one can follow the proof of Theorem 2.2 and construct a similar algorithm to calculate the inverse FFT.

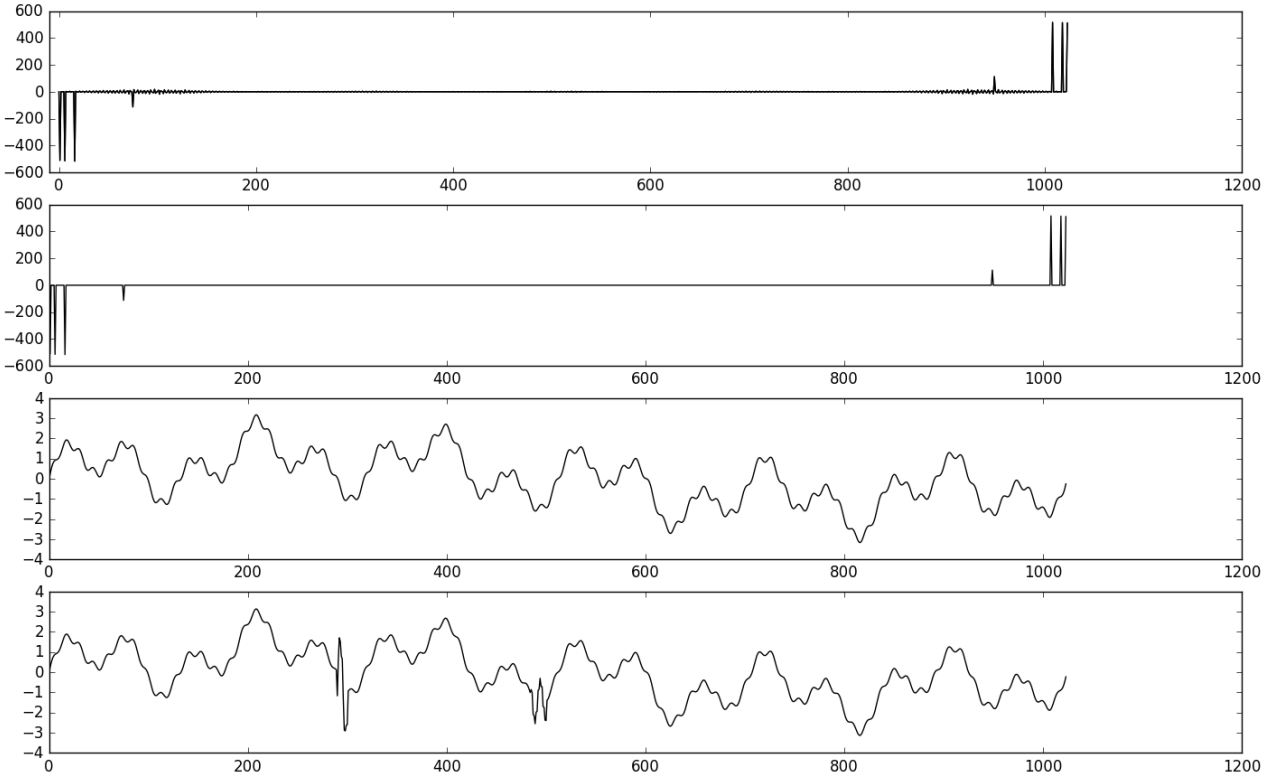


Figure 4.3: From top to bottom: The imaginary part of $\hat{z}$, the imaginary part of the filtered signal, the filtered signal inverted to the standard basis, the original signal.

```
def HaarN(z):
    N=int(z.shape[0])
    U=[]
    V=[]
    s=np.sqrt(2)
    for i in range(int(N/2)):
        u=[0]*N
        v=[0]*N
        u[2*i]=1/s
        u[2*i+1]=1/s
        v[2*i]=1/s
        v[2*i+1]=-1/s
        U.append(u)
        V.append(v)
    H=[]
    for i in range(int(N/2)):
        H.append(np.dot(U[i],z))
    for i in range(int(N/2)):
        H.append(np.dot(V[i],z))
    return(np.asarray(H))
```

Figure 4.4

```
import numpy as np
import time

def DFT(list):
    z=np.asarray(list)
    N=z.shape[0]
    n=np.arange(N)
    k=n.reshape(N,1)
    W=np.exp(-2j*np.pi*k*n/N)
    return np.dot(W,z)

def FFT(list):
    z = np.asarray(list)
    N = z.shape[0]
    if N == 1:
        return DFT(z)
    else:
        U = FFT(z[::2])
        V = FFT(z[1::2])
        c = np.exp(-2j * np.pi * np.arange(N) / N)
        return np.concatenate([U + c[:N / 2] * V,
                               U + c[N / 2:] * V])
```

Figure 4.5: Implementation of the FFT in python code.

```
def timeFFT(list):
    start=time.clock()
    print(FFT(list))
    print('It took ',time.clock()-start,' seconds')

def timeDFT(list):
    start=time.clock()
    print(DFT(list))
    print('It took ',time.clock()-start,' seconds')

>>>timeFFT([1]*2**14)
[  1.63840000e+04 +0.00000000e+00j,
-6.38676465e-13 -1.22464680e-16j
  -6.38676442e-13 -2.44929360e-16j ...,
  2.12892134e-13 -1.22464680e-16j
   6.38676442e-13 -2.44929360e-16j
   6.38676465e-13 -1.22464680e-16j]
It took  0.51654  seconds

timeDFT([1]*2**14)
[  1.63840000e+04 +0.00000000e+00j,
-5.70210545e-13 -3.44056381e-13j
   3.82138765e-12 -1.37239092e-12j ...,
   2.04739825e-09 +2.75143276e-11j
  -2.57208876e-09 -1.81357195e-11j
  2.16681726e-08 -3.42337799e-11j]
It took  36.581999  seconds
```

Figure 4.6: Running the FFT program on a large vector.

5 Conclusions

The main result we have shown is that any properly defined wavelet filter sequence (as in Definition 3.4) also uniquely determines an orthonormal basis in $\ell^2(\mathbb{Z}_{2^p})$. The equivalence (3.10) shows that there is a one-to-one correspondence between the functions of $\ell^2(\mathbb{Z}_{2^p})$ and the outputs of the MRA algorithm, which proves we have perfect reconstruction in the synthesis phase of the multiresolution analysis. Thus we can operate on the output of the MRA as in $\ell^2(\mathbb{Z}_{2^p})$, for example by scalar products, matrix multiplication etc. In other words, if T is the complete MRA operator, meaning $T(z)$ is the output of the MRA, we have for any linear operator $A : \ell^2(\mathbb{Z}_{2^p}) \to \ell^2(\mathbb{Z}_{2^p})$:

$$T(A(z)) = A(T(z)).$$

We have also motivated the construction of wavelets by discussing the problem of localization in time and frequency and the uncertainty principle. We have given examples that illustrate how wavelets tackle this problem (Example 3.2) and the use of wavelets in compression of signals (Examples 3.1 and 3.3). We showed the power of the fast Fourier transform together with Theorem 2.1 that facilitate fast calculations of convolution, the main operator involved in the wavelet transform.

The next step would have been to study two dimensional wavelet analysis and particularly give examples of image compression using wavelets, and give more examples of specific wavelet bases constructed to solve different kinds of problems.

Bibliography

[1] David L. Donoho, Philip B. Stark. *Uncertainty Principles And Signal Recovery*. Siam Journal on Applied Mathematics, Vol. 49, No. 3, pp. 906-931, June 1989.

[2] Michael W. Frazier, F.W. Gehring, S. Axler. *Introduction to Wavelets Through Linear Algebra*. Springer, 1999.

[3] R.X. Gao, R. Yan. *Wavelets: Theory and Applications for Manufacturing* 17 DOI 10.1007/978-1-4419-1545-0_2,#Springer Science+Business Media, LLC 2011

[4] Michael T. Heideman, Don H. Johnson, C. Sidney Burrus. *Gauss and the History of the Fast Fourier Transform* IEEE ASSP Magazine, October 1984.

[5] F. Truchetet, O. Laligant. *Wavelets in industrial applications: a review* `http://le2i.cnrs.fr/IMG/publications/wavelet-spie.pdf`

[6] Jake Vanderplas. `https://jakevdp.github.io/blog/2013/08/28/understanding-the-fft/`

[7] `https://proofwiki.org/wiki/Vandermonde_Determinant`. 20 May, 2016.

[8] `https://en.wikipedia.org/wiki/Relativistic_Doppler_effect` 23 May, 2016.

[9] `https://en.wikipedia.org/wiki/Redshift` 23 May, 2016.

[10] `https://en.wikipedia.org/wiki/Uncertainty_principle`. 20 May, 2016.

[11] `https://en.wikipedia.org/wiki/Fourier_transform#Uncertainty_principle` 20 May, 2016.

[12] `https://en.wikipedia.org/wiki/Astronomical_spectroscopy`