FROM FOURIER TRANSFORMS TO WAVELET ANALYSIS: MATHEMATICAL CONCEPTS AND EXAMPLES

LY TRAN
MAY 15, 2006

Abstract. This paper studies two data analytic methods: Fourier transforms and wavelets. Fourier transforms approximate a function by decomposing it into sums of sinusoidal functions, while wavelet analysis makes use of mother wavelets. Both methods are capable of detecting dominant frequencies in the signals; however, wavelets are more efficient in dealing with time-frequency analysis. Due to the limited scope of this paper, only Fast Fourier Transform (FFT) and three families of wavelets are examined: Haar wavelet, Daub$J$, and Coif$I$ wavelets. Examples for both methods work on one dimensional data sets such as sound signals. Some simple wavelet applications include compression of audio signals, denoising problems, and multiresolution analysis. These applications are given for comparison purposes between Fourier and wavelet analysis, as well as among wavelet families.

Although wavelets are a recent discovery, their efficacy has been acknowledged in a host of fields, both theoretical and practical. Their applications can be expanded to two or higher dimensional data sets. Although they are omitted in this paper, more information is available in [7] or many other books on wavelet applications.

1. Introduction

Wavelets are a recent discovery in mathematics; however, their rapid development and wide range of applications make them more powerful than many other long-existing analytical tools. Conventionally, wavelets are often compared to the Fourier transform to promote their advantages. This paper will take a similar approach in attempt to illustrate wavelet transform in various applications.

The Fourier transform makes use of Fourier series, named in honor of Joseph Fourier (1768-1830), who proposed to represent functions as an infinite sum of sinusoidal functions [1]. Joseph Fourier was the first to use such series to study heat equations. After him, many mathematicians such as Euler, d'Alembert, and Daniel Bernoulli continued to investigate and develop Fourier analysis [1]. From the original series, various Fourier transforms were derived: the continuous Fourier transform, discrete Fourier transform, fast Fourier transform, short-time Fourier transform, etc... Fourier analysis is adopted in many scientific applications, especially in dealing with signal processing. As the applications grew more complex over time, the Fourier transform started to reveal its inefficiencies when working with time series or data with certain characteristics. Despite the attempt to tailor the method to different groups of data, the Fourier transform remained inadequate. Consequently, wavelets received more attention as they proved able to overcome the difficulties.

Date: May 15, 2006.

The first block of wavelet theory was started by Alfred Haar in the early 20th century [2]. Other important contributors include Goupillaud, Grossman, Morlet, Daubechies, Mallat and Delprat. Their different focuses helped to enrich the wavelet families and widen the range of wavelet applications. Because of the similarities, wavelet analysis is applicable in all the fields where Fourier transform was initially adopted. It is especially useful in image processing, data compression, heart-rate analysis, climatology, speech recognition, and computer graphics.

This paper focuses on only a few aspects of each analysis. The first section discusses Fourier series in different representations: sinusoidal functions and complex exponential. A discretization method is also introduced so as to provide support for the discussion of Fast Fourier Transform (FFT). After illustrating Fourier analysis with concrete examples, the paper will turn to the Fourier transform's shortcomings, which give rise to wavelets .

The second section discusses three families of wavelets: the Haar wavelets, Daubechies wavelets, and Coiflets. Concepts and general mechanisms will be provided in detail for Haar wavelets and omitted for the others. Finally, we will look at the advantages of wavelets over Fourier transform through a number of examples.

The paper uses three main references: Course notes in Modeling II, *A Primer on Wavelets and their Scientific Applications* by James Walker, and *A First Course in Wavelets with Fourier Analysis* by Boggess and Narcowich. As the paper is aimed at readers at undergraduate level, mathematical background of linear algebra and basic calculus is assumed.

2. Fourier Analysis

Fourier analysis, which is useful in many scientific applications, makes use of Fourier series in dealing with data sets. In this section, a few representations of Fourier series and related concepts will be introduced. Consequently, several examples will implement these defined concepts to illustrate the idea of Fourier analysis.

2.1. **Fourier Series.**

2.1.1. *Sine and Cosine Representation.* A Fourier series is the expression of any function $f(x) : \mathbb{R} \to \mathbb{R}$ as an infinite sum of sine and cosine functions:

$$f(x) = a_0 + \sum_{k=1}^{\infty} a_k \sin(kx) + \sum_{m=1}^{\infty} b_m \cos(mx). \tag{1}$$

In this section, we work with continuous functions on $[0, 2\pi]$, and thus, it is necessary to familiarize ourselves with the *vector space* of such functions. It is an infinite dimensional vector space, denoted $\mathrm{C}[0, 2\pi]$, where each point on the continuous interval $[0, 2\pi]$ represents a dimension. Then, an *orthogonal basis* of $\mathrm{C}[0, 2\pi]$ is:

$$\{1, \sin(kx), \cos(mx) | k, m = 1, 2, 3, ...\}.$$

Fourier series can also be employed to write any continuous function $f(x) : \mathrm{C}[0, 2\pi] \to \mathrm{C}[0, 2\pi]$. Concepts such as inner product, norm, and distance between two functions in an infinite dimensional vector space are defined in a similar manner to that of a finite dimensional vector space. However, the infinite sum gives rise to the use of an integral in the definition:

- *Inner Product:* The inner product of two continuous functions $f(x)$, $g(x) \in$ C$[0, 2\pi]$ is
$$< f, g >= \int_0^{2\pi} f(x)g(x)dx.$$
- *Norm:* Recall that the norm of a vector in finite dimensional vector space is calculated as $\| \vec{x} \|^2 =< \vec{x}, \vec{x} >$. Norm is defined similarly for a function in infinite dimensional space C$[0, 2\pi]$:
$$\| f \|^2 =< f, f >= \int_0^{2\pi} f^2(x)dx.$$
- *Angle between two functions:* The inner product of f and g can also be written as
$$< f, g >=\| f \| \| g \| \cos(\theta).$$
The angle between two functions, θ, can be achieved from the above equation.
- *Distance* Distance between two continuous functions f and g is defined as $\| f - g \|$.

The inner product formula can be applied to prove that the basis mentioned above is indeed orthogonal. Let us consider six possible inner products of the basis vector functions: $\int_0^{2\pi} \sin(kx)dx$, $\int_0^{2\pi} \cos(mx)dx$, $\int_0^{2\pi} \sin(kx)\sin(mx)dx$ $(k \neq m)$, $\int_0^{2\pi} \sin(kx)\cos(mx)dx$ $(k \neq m)$, $\int_0^{2\pi} \cos(kx)\cos(mx)dx$ $(k \neq m)$. The following trignometric identities [3] are helpful in proving that all six inner products are equal to 0, implying an orthogonal basis:

$$\begin{aligned}
\sin(x)\cos(y) &= \frac{1}{2}[\sin(x+y) + \sin(x-y)], \\
\cos(x)\cos(y) &= \frac{1}{2}[\cos(x+y) + \cos(x-y)], \\
\sin(x)\sin(y) &= \frac{1}{2}[\cos(x-y) - \cos(x+y)].
\end{aligned}$$

2.1.2. *Function Coefficients.* Coefficients of f can be obtained by projecting f on the corresponding basis vectors (similar to that of finite dimensional space):

$$\begin{aligned}
a_0 &= \frac{< f, 1 >}{< 1, 1 >} = \frac{1}{2\pi} \int_0^{2\pi} f(x)dx, \\
a_k &= \frac{< f, \sin(kx) >}{< \sin(kx), \sin(kx) >} = \frac{1}{\pi} \int_0^{2\pi} \sin(kx)f(x)dx, \\
b_m &= \frac{< f, \cos(mx) >}{< \cos(mx), \cos(mx) >} = \frac{1}{\pi} \int_0^{2\pi} \cos(mx)f(x)dx.
\end{aligned}$$

2.1.3. *Complex Form.* Instead of taking the integral of individual sine and cosine functions, the complex form representation of Fourier series enables us to compute the various inner product integrals simultaneously. The transformation makes use of *Euler's formula* [3]
$$e^{i\theta} = \cos(\theta) + i\sin(\theta).$$
Simple algebra yields
$$a_k \sin(kx) + b_k \cos(kx) = \frac{b_k - ia_k}{2} e^{ikx} + \frac{b_k + ia_k}{2} e^{-ikx}.$$

Thus, if we let $c_0 = a_0/2$, $c_k = \frac{b_k - ia_k}{2}$, and $c_{-k} = \frac{b_k + ia_k}{2}$, then the complex form representation of the Fourier series is

$$f(x) = \sum_{k=-\infty}^{\infty} c_k e^{ikx}.$$

Then differentiation and integration can be calculated simultaneously [4]:

$$\frac{d}{d\theta} e^{i\theta} = ie^{i\theta} = -\sin(\theta) + i\cos(\theta),$$
$$\int e^{i\theta} d\theta = \frac{1}{i} e^{i\theta} = \sin(\theta) - i\cos(\theta).$$

To illustrate the convenience of Euler's formula, let us compute $< x, \sin(3x) >$ and $< x, \cos(3x) >$ using the complex exponential.

Note that

$$\begin{aligned} < x, e^{3ix} > &= \int_0^{2\pi} xe^{3ix} dx = \int_0^{2\pi} x\cos(3x)dx + i\int_0^{2\pi} x\sin(3x)dx \\ &= < x, \cos(3x) > +i < x, \sin(3x) > . \end{aligned}$$

On the other hand, integration by parts yields

$$\begin{aligned} \int_0^{2\pi} xe^{3ix} dx &= \left(\frac{x}{3i} e^{3ix} - \frac{1}{3i} \int e^{3ix} dx \right) \Big|_0^{2\pi} \\ &= \left(\frac{x}{3i} e^{3ix} + \frac{1}{9} e^{3ix} \right) \Big|_0^{2\pi} \\ &= \frac{2\pi}{3i} e^{6\pi i} + \frac{1}{9} e^{6\pi i} - \frac{1}{9}. \end{aligned}$$

Since $e^{6\pi i} = \cos(6\pi) + i\sin(6\pi) = 1$, it follows that

$$\int_0^{2\pi} xe^{3ix} dx = \frac{2\pi}{3i} = -i\frac{2\pi}{3}.$$

Therefore,

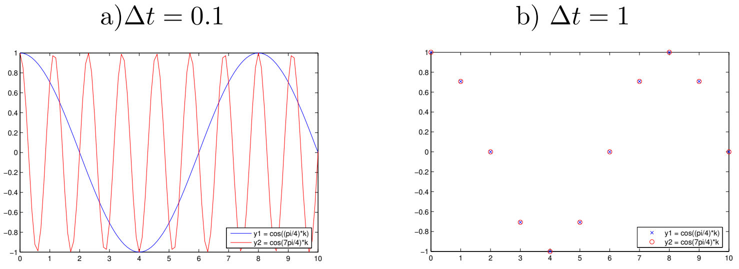
$$\begin{aligned} < x, \cos(3x) > &= 0, \\ < x, \sin(3x) > &= -\frac{2\pi}{3}. \end{aligned}$$

Integrating two functions separately would have doubled the work.

2.1.4. *Fourier Transformation.* Fourier transformation is a method frequently used in signal processing. As the name suggests, it transforms a set of data into a Fourier series. Due to limitation, this paper will only introduce Fast Fourier Transform (FFT).

Discretization. Most data sets are available as a collection of data points. Therefore, the assumption that the function is continuous is no longer required. However, a continuous function can always be represented by N points for some positive integer N. A process in which a continuous function is translated into N representative points is called *discretization.* Discretization translates an infinite dimensional vector space into a finite one [4].

To discretize a function, we choose a finite number of points that represent the function from which the function itself can be sketched. Consider functions $y_1 = \cos(\frac{\pi}{4}t)$ and $y_1 = \ cos(\frac{7\pi}{4}t)$ with $0 \leq t \leq 64$. In Figure 1, these functions



a)$\Delta t = 0.1$ b) $\Delta t = 1$

FIGURE 1. Functions $y_1 = \cos(\frac{\pi}{4}t)$ and $y_2 = \cos(\frac{7\pi}{4}t)$ with different Δt

are plotted with different increments of t (Δt) of 1 and 0.1. When $\Delta t = 0.1$, two functions are different, but when $\Delta t = 1$, the values plotted are essentially the same. This suggests that $\Delta t = 1$ is too large to differentiate these two functions. The same pattern is observed for any pair of sine or cosine functions that are conjugate of each other (for example, $\cos(\alpha t)$ and $\cos(\beta t)$, where $\alpha + \beta = 2\pi$).

Formulas When a continuous function is discretized into N points for FFT, N is usually a power of 2 for computational reasons. Suppose that N points $\{f_0, f_1, ..., f_{N-1}\}$ are spaced out evenly on the interval $[0, 2\pi]$, then the k-th point in Fourier series form is

$$F_k = \sum_{n=1}^{N} f_n e^{-\frac{2\pi i}{N}\cdot(k-1)(n-1)} \tag{2}$$

for $1 \leq k \leq N$. Recall that for two vectors x, y in complex plane, their inner product is defined as

$$< x, y >= x^T \bar{y}.$$

Thus, equation (2) is in fact an inner product between $F = [f_1, f_2, ..., f_N]^T$ and $[e^{-\frac{2\pi i}{N}\cdot 0\cdot(k-1)}, e^{-\frac{2\pi i}{N}\cdot(k-1)}, e^{-\frac{2\pi i}{N}\cdot 2(k-1)}, ..., e^{-\frac{2\pi i}{N}\cdot(k-1)(N-1)}]$ for any integer k in interval $[1, N]$. Hence, we can represent equation (2) by a product of F and an $N \times N$ matrix C, whose entries are determined as follows:

$$C_{kj} = e^{\frac{2\pi i}{N}\cdot(k-1)(j-1)}.$$

This process is called Fourier transformation. C is the *transformation matrix*. Instead of dealing with integrals, we now face easier matrix operations:

$$\begin{bmatrix} F_1 \\ F_2 \\ \vdots \\ F_N \end{bmatrix} = e^{\frac{2\pi i}{N}A} \times \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \end{bmatrix},$$

where

$$A = \begin{bmatrix} 0 & 0 & 0 & \dots & 0 \\ 0 & 1 & 2 & \dots & N-1 \\ \vdots & & & & \vdots \\ 0 & N-1 & 2(N-1) & \dots & (N-1)(N-1) \end{bmatrix}$$

The above cross product suggests that the inverse transformation process is also feasible:

$$f_n = \frac{1}{N}\sum_{k=1}^{N} F_k e^{\frac{2\pi i}{N}\cdot(k-1)(n-1)} \tag{3}$$

for $1 \leq n \leq N$.

Expansion to function f on the interval $[\alpha, \beta]$ The process of Fourier transformation can be extended to any function on the interval $[\alpha, \beta]$ with the appropriate discretization process. Discretize the function f into N equal subintervals, so that each subinterval is

$$\Delta x = \frac{\beta-\alpha}{N} \text{ or } \beta - \alpha = N \cdot \Delta x. \tag{4}$$

and we have N points $\{x_1, x_2, ...x_N\}$ on the interval $[\alpha, \beta]$. To perform FFT, we can find an one-to-one correspondence from the function f to $[0, 2\pi]$:

$$x_j \in [\alpha, \beta] \to t_j = \frac{2\pi}{\beta-\alpha}\cdot(x_j - \alpha) \in [0, 2\pi].$$

The inverse process is:

$$t_j \in [0, 2\pi] \to x_j = \alpha + \frac{\beta-\alpha}{2\pi}\cdot t_j \in [\alpha, \beta].$$

Substitute equation (4) in, we get:

$$t_j = \frac{2\pi}{N\cdot\Delta x}\cdot(x_j-\alpha) \text{ or } x_j = \alpha + \frac{N\cdot\Delta x}{2\pi}\cdot t_j.$$

The above equation suggests that discretizing $f(x)$ over $[\alpha, \beta]$ is the same as $f(\frac{2\pi}{N\cdot\Delta x}(x_j - \alpha))$ over $[0, 2\pi]$. Therefore, we can use the function

$$f(x) = a_0 + \sum_{k=1}^{\infty} a_k \cos(k\frac{2\pi}{N\cdot\Delta x}(x-\alpha)) + \sum_{m=1}^{\infty} b_m \sin(m\frac{2\pi}{N\cdot\Delta x}(x-\alpha))$$

for Fourier transformation.

To summarize, if we have N points with spacing Δx, labeled as $\{f_0, f_1, \ldots, f_N\}$ on the interval $[\alpha, \beta]$, then the relationship between function f and its FFT is expressed in the following equations:

$$f_n = a_0 + \sum_{k=1}^{N/2} a_k \cos(k\frac{2\pi}{N\cdot\Delta x}x_n) + \sum_{m=1}^{N/2} b_m \sin(m\frac{2\pi}{N\cdot\Delta x}x_n), \tag{5}$$

where

$$\begin{aligned} a_0 &= \frac{F_1}{N}, \\ a_k &= \frac{2}{N}\cdot \text{Real}(F_{k+1}), \\ b_m &= -\frac{2}{N}\cdot \text{Imag}(F_{m+1}). \end{aligned}$$

Let Δx is the distance between two consecutive points. The periodic nature of sines and cosines entails the period length to be at least $2\Delta x$ to yield good enough approximations (See Figure 2.1.4. Fix the period to be $2\Delta x$. If the distance between two points is less than half the period, the intersection with the x-axis cannot be

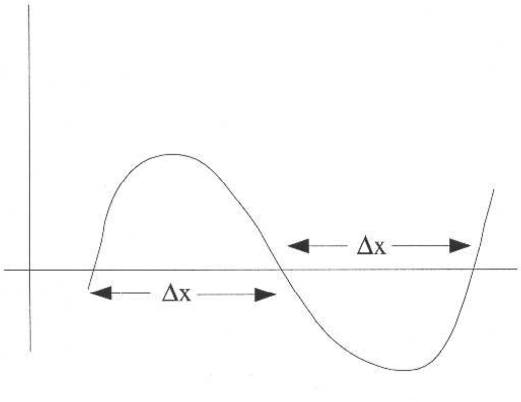


FIGURE 2. Δx and the period length

detected, and hence imprecise estimation is possible. A similar situation occurs when the distance between two points is larger than half the period.) Thus, it is best to have Δx as the distance between two points, or $2\Delta x$ as the period length. For this reason, k and m range from 1 to $N/2$ in equation (5).

From what we know about trignometric functions $\sin(At)$ and $\cos(At)$, we have:

- The functions' periods are

$$\text{Period} = \frac{2\pi}{A} = \frac{N\Delta x}{k}.$$

- The functions' frequencies are

$$\text{Frequency} = \frac{A}{2\pi} = \frac{k}{N\Delta x}.$$

As $k, m = 1, 2, \ldots, \frac{N}{2}$, the periods measured in time unit per cycle are

$$N \cdot \Delta x, \frac{N}{2} \cdot \Delta x, \ldots, 2 \cdot \Delta x.$$

and the frequencies measured in cycles per time unit are

$$\frac{1}{N} \cdot \frac{1}{\Delta x}, \frac{2}{N} \cdot \frac{1}{\Delta x}, \ldots, \frac{1}{2} \cdot \frac{1}{\Delta x}.$$

As k changes from 1 to $\frac{N}{2}$, some functions $\sin(kt_n)$ and $\cos(kt_n)$ might contribute more to the original function f_n than others. To measure each function's "importance", we define a new concept, *frequency content* of k, which is given by

$$\text{freq}(k) = \sqrt{a_k^2 + b_k^2}.$$

A plot of k against the frequency content yields the *power spectrum* of a signal.

2.2. **Applications.** In this section, we will examine how FFT is applied in simple examples.

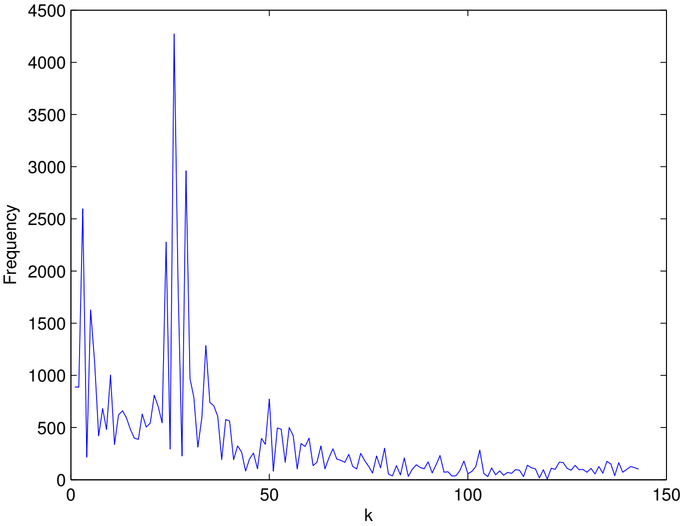


Figure 3. Power spectrum of the sunspot data.

2.2.1. *Sunspot Example.* Data from sunspots have interested scientists for hundreds of years, as they are the indicator for radiation from the sun and therefore have an effect on many scientific fields [5]. MatLab has a collection of data recording the number of sunspots counted on the sun every year between 1700 and 1987. Using Fourier transformation, we will analyze the data to find periodicity in the number of sunspots. Let the domain be 288 years from 1700 to 1987 (thus $N = 288$, $\Delta x = 1$). So k will go from 1 to 143 ($= \frac{N}{2}$).

After the data are loaded, the frequency content of each k is plotted against k (see Figure 3). The plot peaks at the 26th position, which suggests that the component functions with $k = 26$ contribute the most to the FFT. Hence, the period of the sunspot cycle can be estimated using $k = 26$, which yields

$$\frac{N \cdot \Delta x}{k} = \frac{288 \cdot 1}{26} \approx 11.08 \text{ years.}$$

Figure 4 confirms this finding as the frequency content peaks approximately every 11 years.

To see how well the FFT estimates the real function, we will produce a graph of the original function and the approximate function by using the inverse FFT of the ten most significant component functions (with the largest frequency contents) (see Figure 5). Although only ten component functions were used, the approximation shows close estimation to the original function.

Note that the imaginary part resulting from the inverse FFT is negligible since we started with a real function, and the imaginary part therefore should be insignificant.

2.2.2. *Interpolation.* Fourier transformation can be useful for interpolation of data. Given a set of data points, we can fit them with a continuous function using FFT coefficients. The last part of the sunspot example illustrates this method. First,

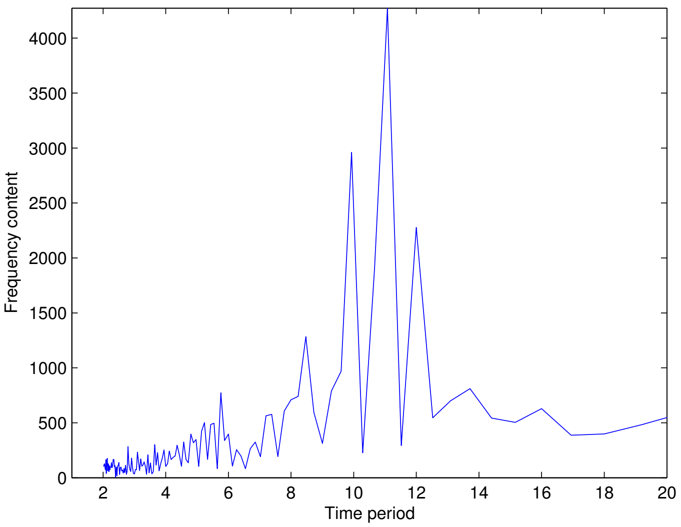


Figure 4. Time cycle and frequency content of sunspot data.

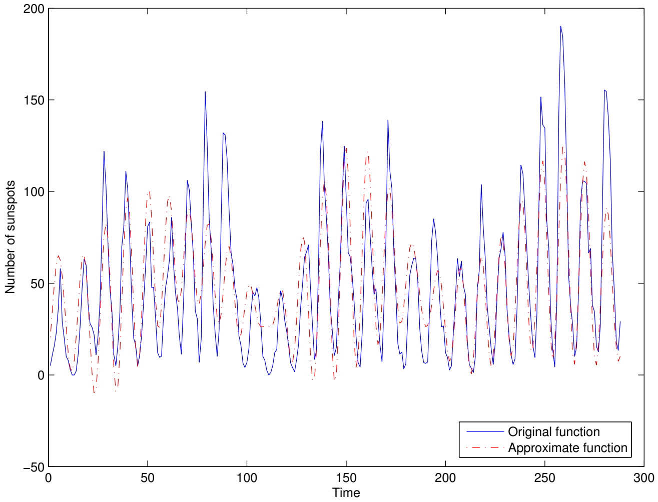


Figure 5. Original function and the approximate function using the ten most significant component functions.

Fourier transform is employed to find FFT coefficients. Then, with an appropriate interval length, Δx, the new continuous function can be constructed. Determining Δx is the key to obtaining an accurate approximation of the original function, given that there are sufficient number of data points.

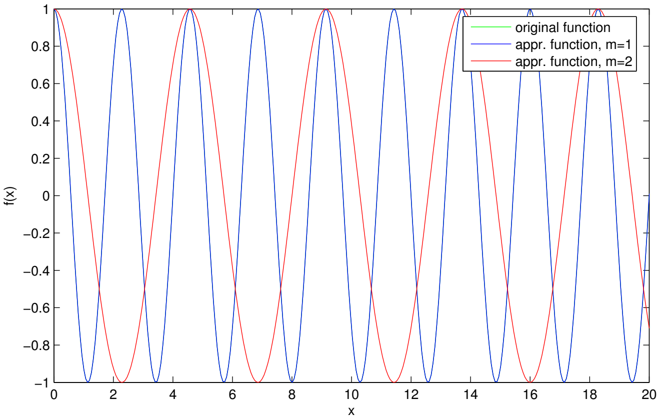


FIGURE 6. Interpolation of 1000 data points from function $f = \cos(\frac{7\pi}{8}k)$. m represents Δx.

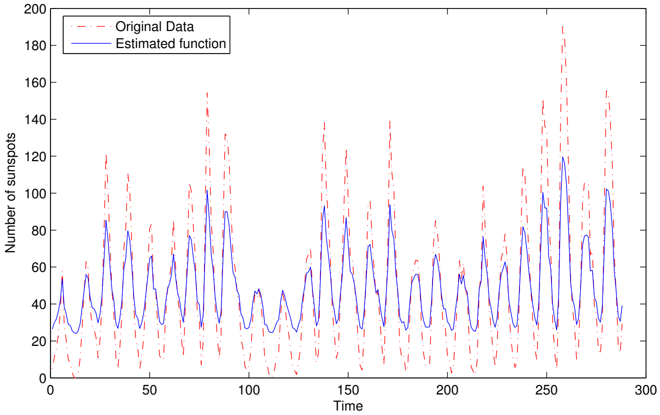


FIGURE 7. Filtering based on 5% percentage method.

Figure 6 is a graph of the original function $f = \cos(\frac{7\pi}{8}k)$ and its approximations with different interval lengths Δx. When $\Delta x = 1$, the approximation matches the original function.

2.2.3. *Filtering.* In the third part of the sunspot example, only the 10 most significant component functions were taken, and number 10 was indeed arbitrarily determined. The process of choosing certain components over others is called *filtering*. Filtering can be extremely useful if data points are suspected to include noise. In this implementation, we take a look at two different methods of filtering.

- *Method 1:* The idea of this method is to choose functions whose coefficients contribute more (or less) than a certain percentage of the total sum. For

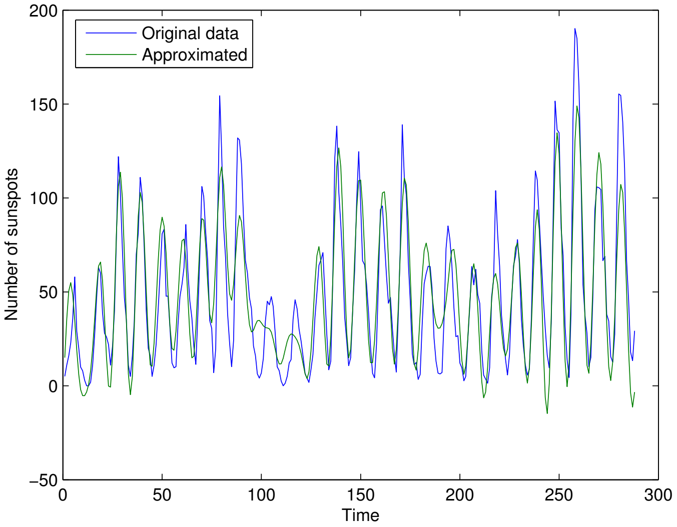


FIGURE 8. The original data and its approximation, resulted from band-pass filtering method.

example, given a function F, we sum up all frequency contents:

$$S = \sum_{k=1}^{N/2} |F(k)|.$$

Then for each k, we determine how much its frequency content contributes to the sum in percentage terms: $c(k) = |F(k)|/S \times 100\%$. Figure 7 displays the original function f and its approximation after filtering all component functions whose contribution is less than 5%. f is the function of sunspot data.

- *Method 2 (band-pass method):* The second method is called band-pass filtering, based on the distribution of frequencies in the signal. The idea is to choose (or to not choose) only the frequencies that lie within one standard deviation from the mean. Using the same set of data, Figure 8 depicts the original function f and its approximation function. Figure 9 displays the original power spectrum, the filtered power spectrum, and the filtered part. Note that the latter two add up to the original power spectrum.

2.3. **The Drawbacks of Fourier Analysis.** Despite its convenience in dealing with different groups of data, Fourier analysis still poses problems in many applications. This section will examine two common issues of FFT: leakage and Gibb's phenomenon.

2.3.1. *Leakage.* To best illustrate the problem of leakage, we will look at the function

$$y = \cos(\frac{7\pi}{8}t).$$

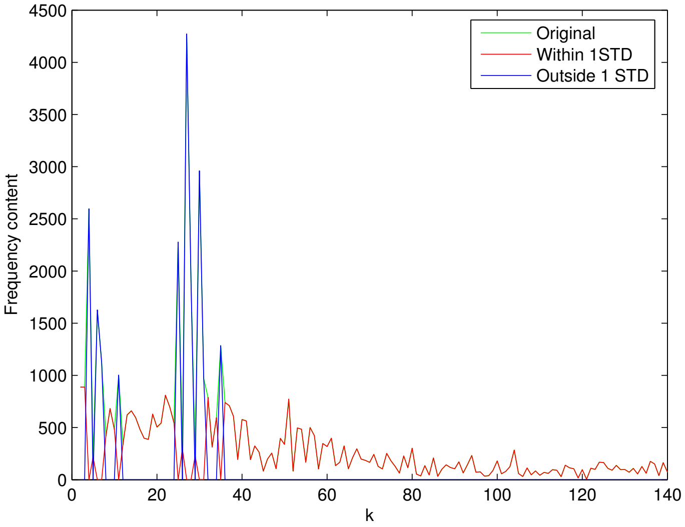


FIGURE 9. Band-pass filtering method. Power spectrum of the original data and the filtered power spectrum.

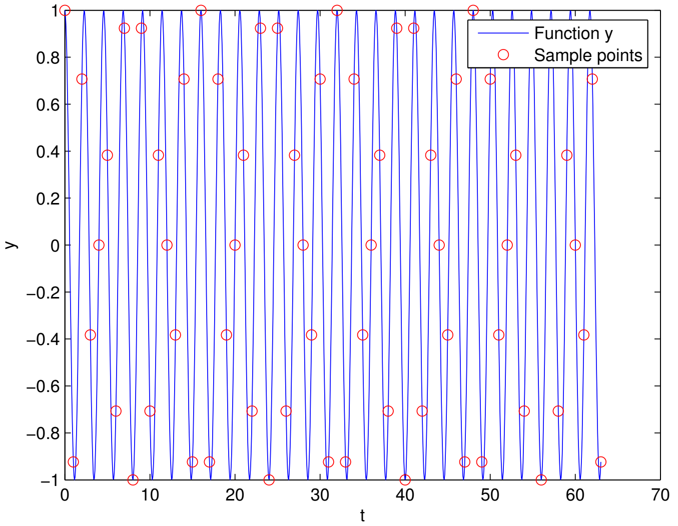
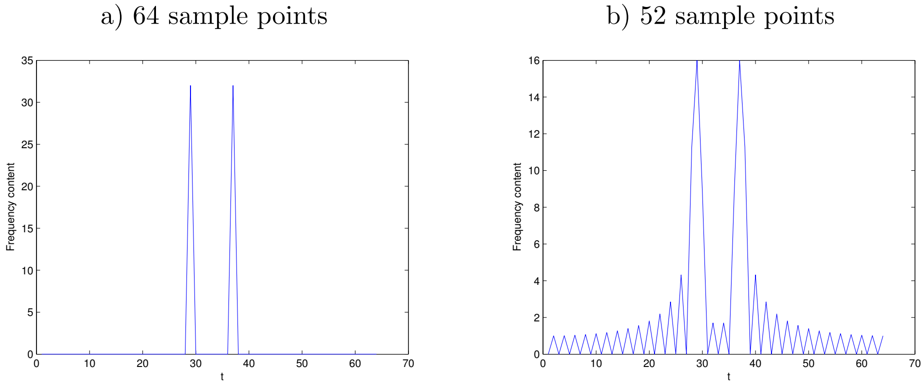


FIGURE 10. Function $y = \cos(\frac{7\pi}{8}t)$ and 64 sample points.

$N = 64$ sample points from 0 to 63 are taken ($\Delta x = 1$). Figure 10 provides us with a graph of y and sample points.

A problem data collectors usually encounter is that there are fewer data points available than they wish. For other data mining methods, a simple method of


a) 64 sample points

b) 52 sample points

FIGURE 11. Power Spectra of the FFT of function y with a) 64 sample points and b) 52 sample points (the rest are zeroed).

zeroing the rest of the sample points can be used. However, it is not the same for Fourier transform: Figure 11(a) plots power spectrum of the FFT of y with 64 sample points, and Figure 11(b) plots the same thing, however, the last 10 sample points are zeroed.

The frequency contents in Figure 11(a) reflect function y better; while the frequency contents in Figure 11(b) appear to include some noise. From these graphs, we can conclude that adding zeros does not help in case of sample points shortage. This effect is known as "leakage": while the frequency content of the signal did not change, the power spectrum did [4].

2.3.2. *Gibb's Phenomenon.* In many cases, an attempt to fit a function with discontinuities or steep slopes using Fourier transform fails. The reason is that FFT only yields smooth functions. This creates a problem known as Gibb's phenomenon.

For example, take a look at a square wave function defined as follow:

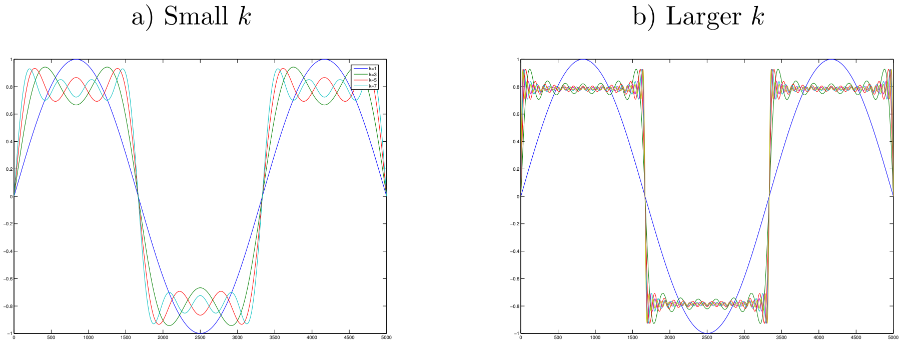
$$f(x) = \begin{cases} 1, & \text{if} \quad 2k\pi \leq x \leq (2k+1)\pi \\ -1, & \text{if} \quad (2k+1)\pi \leq x \leq 2(k+1)\pi \end{cases}$$

for any nonnegative integer k.

Then f can be written as sum of odd harmonics [4]:

$$\begin{aligned} f(x) &= \sin(x) + \frac{\sin(3x)}{3} + \frac{\sin(5x)}{5} + \ldots \\ &= \sum_{k=1}^{\infty} \frac{1}{2k-1} \sin((2k-1)x). \end{aligned}$$

The larger k is, the better the function f is approximated. Figure 12 illustrates this fact: even with large k, the values at the points of discontinuities cannot be approximated. Thus, a perfect square wave can never be obtained.



a) Small k

b) Larger k

FIGURE 12. Gibb's phenomenon. Approximation of function f with a) small k and b) larger k

3. WAVELETS

3.1. **The Transitions from Fourier Analysis.** The traditional Fourier transform is implemented only on the interval $[0, 2\pi]$. This suggests that Fourier transform will encounter difficulty in coping with large data files with complicated patterns. Fourier analysis has been modified in many cases to comply with the available data. For example, windowing - the method of *multiple Fourier transform* - is often adopted to solve the problem of abundant data points [4].

3.1.1. *Lag Matrix.* Multiple transform is the method where we split a large data file into equal time intervals and apply Fourier transform on each of them. Dividing a data file into smaller intervals requires two specifications: (i) the time interval, and (ii) the number of data points between two consecutive time intervals that will be skipped (overlaps are encouraged for the purpose of precision). To make it easier for computation, data can be put into a matrix, whose column vectors act as an individual data set for a Fourier transform. Such a matrix is called a *lag matrix.* Another way to visualize the lag matrix is to slide along the data file a *window* whose length is equal to the time interval. Notice that the window is not slid smoothly but discretely in accordance with the number of skipped data points. The sliding process is called *windowing process.*

For example, consider a time series consisting of 2400 data points. They can be split into 600 individual data sets, each of which contains 16 data points. The skipping step is therefore 4. Hence the data matrix

$$X = \begin{bmatrix} x_1 & x_5 & x_9 & \dots & x_{2385} \\ x_2 & x_6 & x_{10} & \dots & x_{2386} \\ \vdots & & & \vdots & \\ x_{16} & x_{20} & x_{24} & \dots & x_{2400} \end{bmatrix}.$$

3.1.2. *Sunspot Example Revisited.* We will apply the windowing process to the sunspot data set: the data was cut into lengths of 32 time units. Recall that *power spectrum* is a plot of k against its frequency contents. Figure 13 is the spectra of multiple Fourier transforms applied on the sunspot data set, with skipping

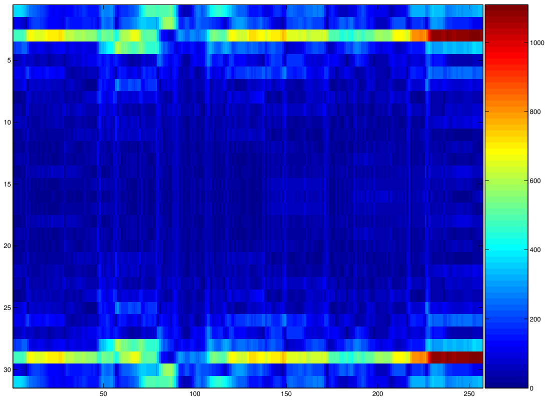


FIGURE 13. Power spectra of the windowed sunspot data, data length = 32, skipping step = 1.

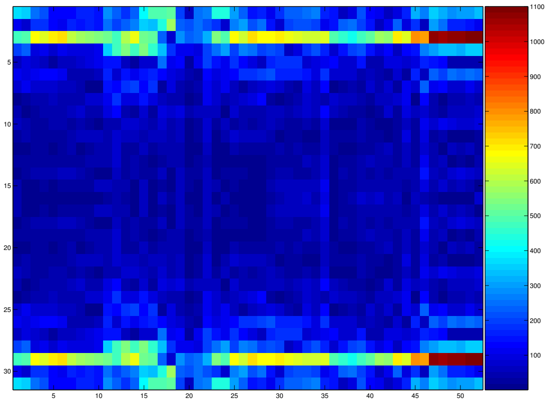


FIGURE 14. Power spectra of the windowed sunspot data, data length = 32, skipping step = 5.

step equal to 1. Figure 14 is the spectra of the same data set; however, the skipping step is now 5.

The color bar next to each graph is the guide to decipher power spectrum in each smaller data set. The symmetric property is retained in both graphs. Figure 13 has better resolution, which results from smaller skipping step. However, both graphs represent the same trend of frequency contents.

3.2. **Haar Wavelets.** Although multiple transforms can be adopted to solve problems of large data files, such a method is not complete. Determining the time

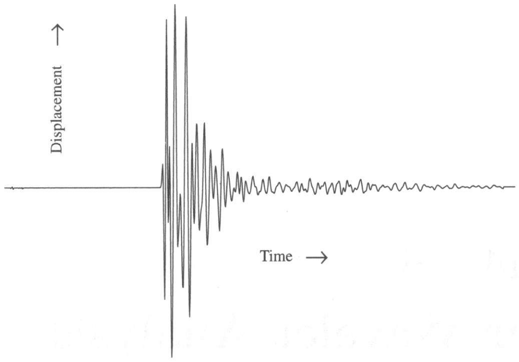


FIGURE 15. Data from seismic survey [6].

interval and the skipping step is not an easy task for a complicated data set. Moreover, for a data set whose patterns vary greatly, a single time frame might not be appropriate. These shortcomings of Fourier transform gave rise to wavelets.

Wavelets were first applied in analyzing data from seismic surveys in geophysics and later on in many other signal processing applications [6]. Figure 15 displays sample data from a seismic survey. Using the multiple transform method is certainly not a good approach in this case: there are short-duration and high-frequency mixed with low-frequency bursts; thus, equal-time intervals will not be able to detect both of them.

In this section, the Haar transform and Haar wavelets will be introduced. They represent the simplest type of wavelet analysis, and can serve as the prototype for all other wavelet operations [7]. Two components that play primary roles in any wavelet analysis are the *scaling function*, known as the *father wavelet*, and the *wavelet function*, also known as the *mother wavelet*. Many theorems or results of this section come directly or after a few simple steps from the definitions with some knowledge of linear algebra; thus, some proofs will be omitted. Keep in mind that all signals are plotted on a two dimensional time-axis against a displacement-axis.

This section only works with discrete signals. Thus, every signal $\mathbf{f}$ is of the form $\mathbf{f} = (f_1, f_2, \ldots, f_N)$, where N is a positive even integer (similar to FFT, N is ideally a power of 2). The Haar transform decomposes a discrete signal $\mathbf{f}$ into two subsignals: one reserves the trend of $\mathbf{f}$, and the other reserves its fluctuation [7].

3.2.1. *Haar transform, 1-level.*

Definition 1. *The first level of the Haar transform is the mapping $\mathbf{H_1}$ defined by*

$$\mathbf{f} \xmapsto{\mathbf{H_1}} (\mathbf{a^1}|\mathbf{d^1}),$$

where $\mathbf{f}$ is a discrete signal, $\mathbf{a^1} = (a_1, a_2, \ldots, a_{N/2})$, and $\mathbf{d^1} = (d_1, d_2, \ldots, d_{N/2})$ such that

$$a_m = \frac{f_{2m-1} + f_{2m}}{\sqrt{2}}, \tag{6}$$

$$d_m = \frac{f_{2m-1} - f_{2m}}{\sqrt{2}}. \tag{7}$$

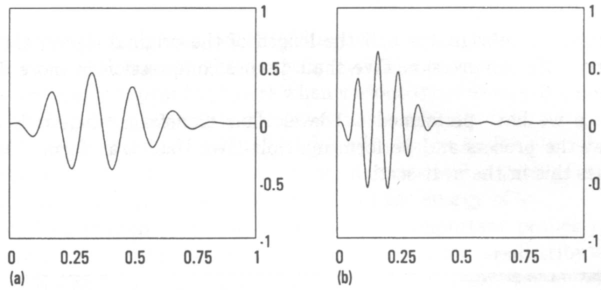


FIGURE 16. The graph of signal $\mathbf{f}$ on the left and its Haar transform, 1-level on the right [7].

For example, let $\mathbf{f} = (4, 6, 10, 12, 8, 6, 5, 5)$, then we have

$$\begin{array}{lcccccccc}
\mathbf{f} & 4 & 6 & 10 & 12 & 8 & 6 & 5 & 5 \\
\mathbf{a^1} & \multicolumn{2}{c}{5\sqrt{2}} & \multicolumn{2}{c}{11\sqrt{2}} & \multicolumn{2}{c}{7\sqrt{2}} & \multicolumn{2}{c}{5\sqrt{2}} \\
\mathbf{d^1} & \multicolumn{2}{c}{-\sqrt{2}} & \multicolumn{2}{c}{-\sqrt{2}} & \multicolumn{2}{c}{\sqrt{2}} & \multicolumn{2}{c}{0}
\end{array}$$

Thus,

$$(4, 6, 10, 12, 8, 6, 5, 5) \overset{\mathbf{H_1}}{\longmapsto} (5\sqrt{2}, 11\sqrt{2}, 7\sqrt{2}, 5\sqrt{2}| - \sqrt{2}, -\sqrt{2}, \sqrt{2}, 0).$$

The mapping $\mathbf{H^1}$ has an inverse; given $(\mathbf{a^1}|\mathbf{d^1})$, $\mathbf{f}$ can easily be obtained:

$$f_{2m-1} = \frac{a_m + d_m}{\sqrt{2}}, f_{2m} = \frac{a_m - d_m}{\sqrt{2}}.$$

If $\mathbf{f}$ is extracted from a continuous signal and the spaced sample values of $\mathbf{f}$ are small, each value of $\mathbf{d^1}$ will be significantly smaller than the magnitude of $\mathbf{f}$. Figure 16 illustrates this idea. While $\mathbf{a^1}$ appears to be a compression of $\mathbf{f}$ ($\mathbf{a^1}$ is the left half of the graph in Figure 16(b)), $\mathbf{d^1}$ (the right half of the same figure) is close to 0 in magnitude. This property is called the *small fluctuations feature*.

Consequently, signals with small fluctuations feature can be compressed using a smaller number of bits. For instance, if small fluctuations in $\mathbf{d^1}$ are disregarded, we obtain a compression that is 50% of the original or a *2:1 compression.*

3.2.2. *Conservation and Compaction of Energy* [7].

Definition 2. *The energy of a signal* $\mathbf{f}$ *is defined by*

$$\mathcal{E}_{\mathbf{f}} = \sum_{i=1}^{N} f_i^2.$$

Theorem 3 (Conservation of Energy). *The 1-level Haar transform conserves energy, i.e.,* $\mathcal{E}_{(\mathbf{a^1}|\mathbf{d^1})} = \mathcal{E}_{\mathbf{f}}$ *for all signal* $\mathbf{f}$.

Proof. We have

$$\begin{aligned}
\mathcal{E}_{\mathbf{a^1}|\mathbf{d^1}} &= \sum_{i=1}^{N/2} a_i^2 + \sum_{i=1}^{N/2} d_i^2 \\
&= \sum_{i=1}^{N/2} \Big[\Big(\frac{f_{2m-1} + f_{2m}}{\sqrt{2}} \Big)^2 + \Big(\frac{f_{2m-1} - f_{2m}}{\sqrt{2}} \Big)^2 \Big] \\
&= \sum_{i=1}^{N/2} \big(f_{2m-1}^2 + f_{2m}^2 \big) \\
&= \mathcal{E}_{\mathbf{f}}
\end{aligned}$$

Hence the proof is complete. □

General Principle - Compaction of Energy: The energy of the trend subsignal $\mathbf{a^1}$ accounts for a large percentage of the energy of the transformed signal $(\mathbf{a^1}|\mathbf{d^1})$. This is consistent with the small fluctuations feature of $\mathbf{d^1}$ mentioned in the previous subsection.

3.2.3. *Haar transform, multiple levels.* The multiple level Haar transform is performed by consecutively applying Formula (1) on the trend subsignal of the previous operation. For instance, $\mathbf{a^1}$ is obtained from the 1-level Haar transform applied on the signal $\mathbf{f}$. Computing the second trend $\mathbf{a^2}$ and the second fluctuation $\mathbf{d^2}$ for $\mathbf{a^1}$, we obtain a 2-level Haar transform of $\mathbf{f}$. We can keep on doing this as long as the resulting trend signal allows. The Conservation of Energy Theorem still holds here:

$$\mathcal{E}_{(\mathbf{a^n}|\mathbf{d^n}|\mathbf{d^{n-1}}|\ldots|\mathbf{d^2}|\mathbf{d^1})} = \mathcal{E}_{\mathbf{f}}.$$

Definition 4. *The cumulative energy profile of a signal* $\mathbf{f}$ *is a signal defined by*

$$\Big(\frac{f_1^2}{\mathcal{E}_{\mathbf{f}}}, \frac{f_1^2 + f_2^2}{\mathcal{E}_{\mathbf{f}}}, \frac{f_1^2 + f_2^2 + f_3^2}{\mathcal{E}_{\mathbf{f}}}, \ldots, 1 \Big).$$

Figure 17 is a graph of $\mathbf{f}$, its 2-level Haar transform, and their cumulative energy profiles. We can see that the elements in the cumulative energy profile of the 2-level Haar transform signal approach 1 much more rapidly than the original signal, suggesting that the compression by Haar transform conserves most of the energy of the original signal.

3.2.4. *Haar Wavelets.* This section introduces two major concepts in wavelet analysis: the wavelet and the scaling signal. They are extremely useful in mathematical operations of wavelet analysis, which will be illustrated in the following sections.

Definition 5 (Haar Wavelets)**.** *1-level Haar wavelets* $\mathbf{W^1_1}$, $\mathbf{W^1_2}$, $\ldots$, $\mathbf{W^1_{N/2}}$ *are defined as* $\mathbf{W^1_n} = (w^1_{n1}, w^1_{n2}, \ldots, w^1_{nN})$ *such as*

$$w^1_{ni} = \begin{cases} \frac{1}{\sqrt{2}} & \textit{if} \quad i = 2n-1 \\ \frac{-1}{\sqrt{2}} & \textit{if} \quad i = 2n \\ 0 & \textit{otherwise.} \end{cases}$$

Definition 6 (Haar scaling signals)**.** *1-level Haar scaling signals* $\mathbf{V^1_1}$, $\mathbf{V^1_2}$, $\ldots$, $\mathbf{V^1_{N/2}}$ *are defined as* $\mathbf{V^1_n} = (v^1_{n1}, v^1_{n2}, \ldots, v^1_{nN})$ *such as*

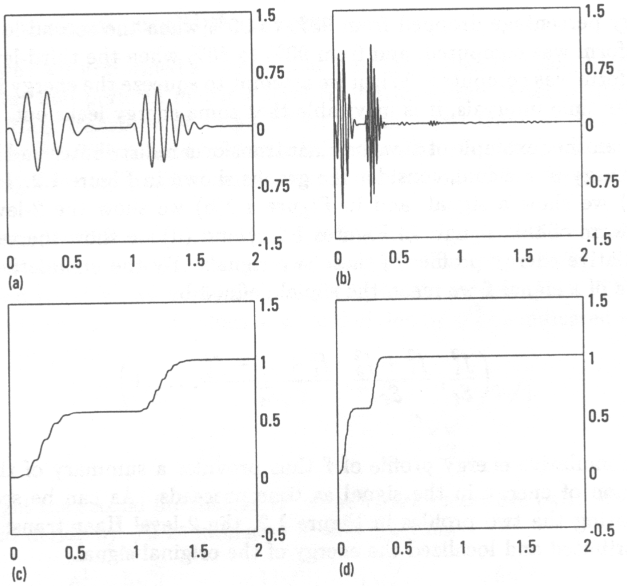


FIGURE 17. The graph of a) signal $\mathbf{f}$, b) 2-level Haar transform of signal $\mathbf{f}$, c) the cumulative energy profile of signal $\mathbf{f}$ and d) the cumulative energy profile of its 2-level Haar transform[7].

$$v^1_{ni} = \begin{cases} \frac{1}{\sqrt{2}} & \textit{if} \quad i = 2n-1 \\ \frac{1}{\sqrt{2}} & \textit{if} \quad i = 2n \\ 0 & \textit{otherwise.} \end{cases}$$

So, the 1-level Haar wavelets are:

$$\begin{aligned} \mathbf{W^1_1} &= (\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, 0, 0, \dots, 0), \\ \mathbf{W^1_2} &= (0, 0, \frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, 0, 0, \dots, 0), \\ &\vdots \\ \mathbf{W^1_{N/2}} &= (0, 0, \dots, 0, \frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}), \end{aligned}$$

and the 1-level Haar scaling signals are:

$$\begin{aligned} \mathbf{V^1_1} &= (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0, 0, \dots, 0), \\ \mathbf{V^1_2} &= (0, 0, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0, 0, \dots, 0), \\ &\vdots \\ \mathbf{V^1_{N/2}} &= (0, 0, \dots, 0, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}). \end{aligned}$$

A relationship can be established between subsignals of $\mathbf{f}$ and the 1-level Haar wavelets and scaling signals, using the familiar scalar product (known as the dot product in Linear Algebra):

$$\begin{aligned} a_m &= \mathbf{f} \cdot \mathbf{V^1_m}, \\ d_m &= \mathbf{f} \cdot \mathbf{W^1_m}. \end{aligned}$$

For any type of wavelets, the mother wavelets act as a window sliding along the signal (*translation*), whereas the scaling signals allow zooming in and out at each point (*dilation*).

We can also define multiple level Haar wavelets $\mathbf{W^m_n} = (w^m_{n1}, w^m_{n2}, \ldots, w^m_{nN})$ and scaling signals $\mathbf{V^m_n} = (v^m_{n1}, v^m_{n2}, \ldots, v^m_{nN})$ in a similar manner:

$$\begin{aligned} v^m_{ni} &= \begin{cases} \frac{1}{\sqrt{2}} & \text{if} \quad 2^{m-1}n + 1 \leq i \leq 2^m n \\ 0 & \text{otherwise,} \end{cases} \\ w^m_{ni} &= \begin{cases} \frac{1}{\sqrt{2}} & \text{if} \quad 2^{m-1}n + 1 \leq i \leq 3 \times 2^{m-2}n \\ \frac{-1}{\sqrt{2}} & \text{if} \quad 3 \times 2^{m-2}n + 1 \leq i \leq 2^m n \\ 0 & \text{otherwise.} \end{cases} \end{aligned}$$

For example,

$$\mathbf{V^2_2} = (0, 0, 0, 0, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0, 0, \ldots, 0),$$

$$\mathbf{W^2_2} = (0, 0, 0, 0, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, \frac{-1}{\sqrt{2}}, \frac{-1}{\sqrt{2}}, 0, 0, \ldots, 0).$$

Consequently, we can obtain multiple level Haar transform from multiple level Haar wavelets and scaling signals:

$$\begin{aligned} \mathbf{a^m} &= \big(\mathbf{f} \cdot \mathbf{V^m_1}, \mathbf{f} \cdot \mathbf{V^m_2}, \ldots, \mathbf{f} \cdot \mathbf{V^m_{N/2^m}}\big), \\ \mathbf{d^m} &= \big(\mathbf{f} \cdot \mathbf{W^m_1}, \mathbf{f} \cdot \mathbf{W^m_2}, \ldots, \mathbf{f} \cdot \mathbf{W^m_{N/2^m}}\big). \end{aligned}$$

The multiple level Haar wavelets and scaling signals are essential in Haar wavelet analysis. Similar to the coefficients in Fourier transform, they provide the means to analyze the signals. More details will be provided in later sections.

3.3. **Daubechies Wavelets.** Haar wavelets are considered the most basic of all. In 1988, Ingrid Daubechies discovered another family of wavelets that were named after her [8]. Unlike Haar wavelets, Daubechies wavelets are continuous. Consequently, they work better with continuous signals. They also have longer supports, i.e. they use more values from the original signals to produce averages and differences. These improvements enable Daubechies wavelets to handle complicated signals more accurately.

We will first examine the simplest of the Daubechies family of wavelets: the Daub4 wavelets. Although the scaling and wavelet numbers are different, the idea of the Daubechies wavelets and Daubechies wavelet transform are very similar to that of the Haar wavelets.

Since all wavelet analyses are similar in definitions and properties, the following sections on different types of wavelets will not go into particulars. The previous section on Haar wavelet analysis might be useful as a reference.

3.3.1. *Definitions.* The Daub4 wavelets use four coefficients for their scaling signals and wavelets, compared to two in that of Haar wavelets.

The scaling coefficients of Daub4 wavelets are

$$\alpha_1 = \frac{1+\sqrt{3}}{4\sqrt{2}}, \quad \alpha_2 = \frac{3+\sqrt{3}}{4\sqrt{2}}, \quad \alpha_3 = \frac{3-\sqrt{3}}{4\sqrt{2}}, \quad \alpha_4 = \frac{1-\sqrt{3}}{4\sqrt{2}},$$

and the wavelet coefficients are:

$$\beta_1 = \alpha_4, \quad \beta_2 = -\alpha_3, \quad \beta_3 = \alpha_2, \quad \beta_4 = -\alpha_1.$$

Then, the first level Daub4 scaling signals are $\mathbf{V}_n^1 = (v_1, v_2, ...v_N)$, for $n = 1, 2, ..., N/2$, in which [7]

$$v_i = \begin{cases} \alpha_1 & \text{if } i = 2n-1 \\ \alpha_2 & \text{if } i = 2n \\ \alpha_3 & \text{if } i = (2n+1) \bmod N \\ \alpha_4 & \text{if } i = (2n+2) \bmod N \\ 0 & \text{otherwise.} \end{cases}$$

The first level Daub4 wavelets $\mathbf{W}_n^1 = (w_1, w_2, ...w_N)$, for $n = 1, 2, ..., N/2$, are defined similarly [7]:

$$w_i = \begin{cases} \beta_1 & \text{if } i = 2n-1 \\ \beta_2 & \text{if } i = 2n \\ \beta_3 & \text{if } i = (2n+1) \bmod N \\ \beta_4 & \text{if } i = (2n+2) \bmod N \\ 0 & \text{otherwise.} \end{cases}$$

According to the definition, we have the 1-level Daub4 wavelets:

$$\begin{aligned} \mathbf{W_1^1} &= (\beta_1, \beta_2, \beta_3, \beta_4, 0, 0, \ldots, 0) \\ \mathbf{W_2^1} &= (0, 0, \beta_1, \beta_2, \beta_3, \beta_4, 0, 0, \ldots, 0) \\ \mathbf{W_3^1} &= (0, 0, 0, 0, \beta_1, \beta_2, \beta_3, \beta_4, 0, 0, \ldots, 0) \\ &\vdots \\ \mathbf{W_{N/2-1}^1} &= (0, 0, \ldots, 0, \beta_1, \beta_2, \beta_3, \beta_4) \\ \mathbf{W_{N/2}^1} &= (\beta_3, \beta_4, 0, 0, \ldots, 0, \beta_1, \beta_2), \end{aligned}$$

and the 1-level Daub4 scaling signals:

$$\begin{aligned} \mathbf{V_1^1} &= (\alpha_1, \alpha_2, \alpha_3, \alpha_4, 0, 0, \ldots, 0) \\ \mathbf{V_2^1} &= (0, 0, \alpha_1, \alpha_2, \alpha_3, \alpha_4, 0, 0, \ldots, 0) \\ \mathbf{V_3^1} &= (0, 0, 0, 0, \alpha_1, \alpha_2, \alpha_3, \alpha_4, 0, 0, \ldots, 0) \\ &\vdots \\ \mathbf{V_{N/2-1}^1} &= (0, 0, \ldots, 0, \alpha_1, \alpha_2, \alpha_3, \alpha_4) \\ \mathbf{V_{N/2}^1} &= (\alpha_3, \alpha_4, 0, 0, \ldots, 0, \alpha_1, \alpha_2). \end{aligned}$$

Figure 18 depicts the two wavelets: Haar and Daub4. Their shapes explain why Haar wavelet works better with discontinuous functions while Daub4 wavelet has an advantage while working with continuous ones.

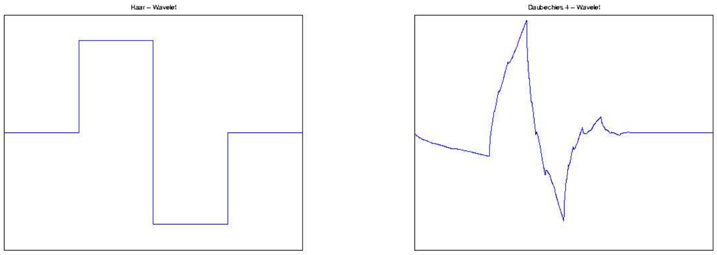


FIGURE 18. (a) Left figure: Haar wavelet, (b) Right figure: Daub4 wavelet

Notice that the scaling signals and wavelets are orthogonal to each other. This can be easily proved using the scaling and wavelet numbers' properties stated in the next section.

3.3.2. *The coefficients' properties.* The wavelet and scaling coefficients satisfy conditions that are essential for the properties of wavelets

$$\begin{aligned}
&\alpha_1^2+\alpha_2^2+\alpha_3^2+\alpha_4^2=1,\\
&\alpha_1+\alpha_2+\alpha_3+\alpha_4=\sqrt{2},\\
&\beta_1^2+\beta_2^2+\beta_3^2+\beta_4^2=1,\\
&\beta_1+\beta_2+\beta_3+\beta_4=0.
\end{aligned}$$

These identities can be used to prove the orthogonality among all scaling signals and wavelets.

3.3.3. *Daub4 First and Multiple-Level Transforms.* Much like 1-level Haar transform, Daub4 transform is defined as the mapping $\mathbf{f} \overset{\mathbf{D_1}}{\longmapsto} (\mathbf{a^1}|\mathbf{d^1})$. Each element in the first trend subsignal $\mathbf{a^1}=(a_1,\dots,a_{N/2})$ is the scalar product $a_i=\mathbf{f}\cdot\mathbf{V}_i^1$. Similarly, the fluctuation subsignal $\mathbf{d^1}=(d_1,\dots,d_{N/2})$ is the scalar product $d_i=\mathbf{f}\cdot\mathbf{W}_i^1$.

Let us define the elementary signals $\mathbf{V_1^0}$, $\mathbf{V_2^0}$, ..., $\mathbf{V_N^0}$ as

$$\begin{aligned}
\mathbf{V_1^0} &= (1,0,0,\dots,0)\\
\mathbf{V_2^0} &= (0,1,0,0\dots,0)\\
&\vdots\\
\mathbf{V_N^0} &= (0,0,\dots,0,1).
\end{aligned}$$

We recognize that

$$\begin{aligned}
\mathbf{V}_m^1 &= \alpha_1\mathbf{V}_{2m-1}^0+\alpha_2\mathbf{V}_{2m}^0+\alpha_3\mathbf{V}_{2m+1}^0+\alpha_4\mathbf{V}_{2m+2}^0,\\
\mathbf{W}_m^1 &= \beta_1\mathbf{V}_{2m-1}^0+\beta_2\mathbf{V}_{2m}^0+\beta_3\mathbf{V}_{2m+1}^0+\beta_4\mathbf{V}_{2m+2}^0,
\end{aligned}$$

where the sub index is $\mod N$.

Higher level Daub4 transforms are obtained by applying the 1-level Daub4 transform consecutively on the trend subsignal of the previous level transform. The higher level scaling signals and wavelets are defined accordingly:

$$\begin{aligned}
\mathbf{V}_m^k &= \alpha_1\mathbf{V}_{2m-1}^{k-1}+\alpha_2\mathbf{V}_{2m}^{k-1}+\alpha_3\mathbf{V}_{2m+1}^{k-1}+\alpha_4\mathbf{V}_{2m+2}^{k-1}\\
\mathbf{W}_m^k &= \beta_1\mathbf{V}_{2m-1}^{k-1}+\beta_2\mathbf{V}_{2m}^{k-1}+\beta_3\mathbf{V}_{2m+1}^{k-1}+\beta_4\mathbf{V}_{2m+2}^{k-1}.
\end{aligned}\tag{8}$$

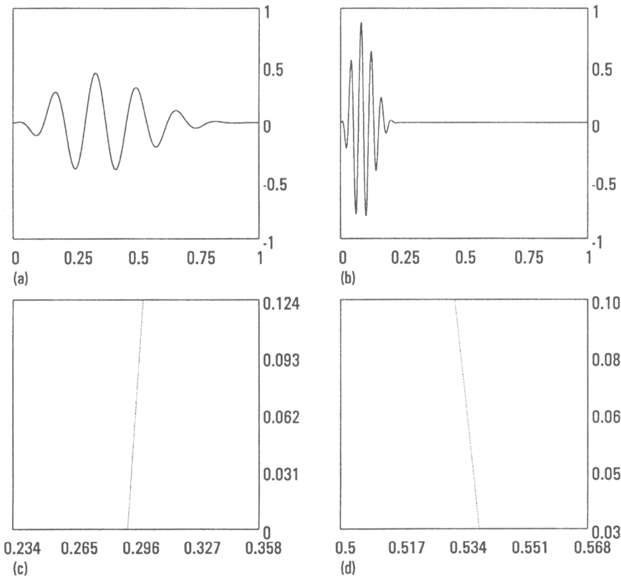


Figure 19. (a) Signal A. (b) 2-level Daub4 transform. (c) and (d) Magnifications of the signal's graph in two small squares; the signal is approximately linear [7].

The above formulas can be modified and applied to any kind of wavelets. To summarize, the first and multiple-level Daub4 transforms are achieved from these wavelets and scaling signals:

$$\begin{aligned} \mathbf{a^m} &= \left(\mathbf{f}\cdot\mathbf{V_1^m}, \mathbf{f}\cdot\mathbf{V_2^m}, \ldots, \mathbf{f}\cdot\mathbf{V_{N/2^m}^m}\right), \\ \mathbf{d^m} &= \left(\mathbf{f}\cdot\mathbf{W_1^m}, \mathbf{f}\cdot\mathbf{W_2^m}, \ldots, \mathbf{f}\cdot\mathbf{W_{N/2^m}^m}\right), \end{aligned} \tag{9}$$

which are similar to those of Haar transforms.

3.3.4. *The Daub4 Transform's Property* [7].

Property 1. *If a signal* $\mathbf{f}$ *is approximately linear over the support of a k-level Daub4 wavelet* $\mathbf{W}_m^k$*, then the k-level fluctuation value* $\mathbf{f}\cdot\mathbf{W}_m^k$ *is approximately zero* [7].

The support of a k-level wavelet depends on the number k. For example, the 1-level Daub4 wavelet has 4 time-unit support (4 non-zero coefficients); the 2-level Daub4 wavelet has 6 time-unit support, etc... Figure 19 illustrates this idea: magnifications of the original signal appear linear; and the fluctuation subsignals seem to be 0. This property is useful in determining whether Daub4 wavelets are adequate for certain applications.

Property 2. *Similar to Haar transform, Daub4 transforms also conserve energy.*

The proof for this can be found in [7]. Figure 20 compares the efficiency of Haar transform and Daub4 transform. The upper graphs show that the detail subsignals from Daub4 transform are significantly less than that from Haar transform. As a result, the cumulative energy of Daub4 transform reaches 1 much faster.

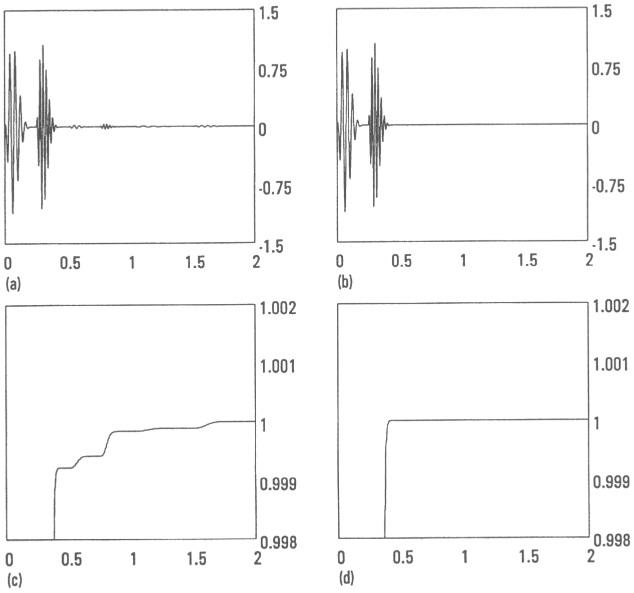


FIGURE 20. (a)2-level Haar transform of signal A. (b) 2-level Daub4 transform on the same signal. (c) Cumulative energy profile for the Haar transform in (a). and (d) Cumulative energy profile for the Daub4 transform in (b)[7].

3.4. **Other Daubechies Wavelets.** Other Daubechies wavelets are very similar to Daub4 wavelets. Ingrid Daubechies developed two families of wavelets: the Daub$J$ wavelets (for $J = 4, 6, 8, \ldots, 20$) and the Coif$I$ wavelets (for $I = 6, 12, 18, 24, 30$) [7].

3.4.1. *DaubJ Wavelets.* We provide a general definition for Daub$J$ wavelets (for $J = 4, 6, 8, \ldots, 20$) as done in [7]. For each J, the scaling numbers α_i (for $i = 1, 2, ..., J$) are computed. The wavelet numbers are then defined accordingly:

$$\beta_i = (-1)^{i+1}\alpha_{J-i}.$$

The 1-level Daub4 scaling signals are: $\mathbf{V}_n^1 = (v_1, v_2, ...v_N)$, for $n = 1, 2, ..., N/2$, in which

$$v_i = \begin{cases} \alpha_1 & \text{if } i = (2n-1) \bmod N \\ \alpha_2 & \text{if } i = (2n) \bmod N \\ \vdots & \\ \alpha_{J-1} & \text{if } i = (2n+J-3) \bmod N \\ \alpha_J & \text{if } i = (2n+J-2) \bmod N \\ 0 & \text{otherwise.} \end{cases}$$

The 1-level Daub4 wavelets are:$\mathbf{W}_n^1 = (w_1, w_2, ...w_N)$, for $n = 1, 2, ..., N/2$, in which

$$w_i = \begin{cases} \beta_1 & \text{if } i = (2n-1) \bmod N \\ \beta_2 & \text{if } i = (2n) \bmod N \\ \vdots & \\ \beta_{J-1} & \text{if } i = (2n+J-3) \bmod N \\ \beta_J & \text{if } i = (2n+J-2) \bmod N \\ 0 & \text{otherwise.} \end{cases}$$

The scaling numbers and wavelet numbers of Daub$J$ transform still satisfy some identities:

$$\begin{aligned} \alpha_1^2 + \alpha_2^2 + \ldots + \alpha_{J-1}^2 + \alpha_J^2 &= 1, \\ \alpha_1 + \alpha_2 + \ldots + \alpha_{J-1} + \alpha_J &= \sqrt{2}, \\ 0^i\beta_1 + 1^i\beta_2 + \ldots + (J-2)^i\beta_{J-1} + (J-1)^i\beta_J &= 0, \\ \text{for } i = 0, 1, ..., J-4. \end{aligned}$$

The multiple level Daub$J$ transforms are defined similarly to Daub4. The 1-level transform is applied consecutively on the previous level's trend signal to achieve a higher transform.

3.4.2. *Coiflets.* Like the Daub$J$ family, the Coif$I$ family, also known as the "coiflets", consists of different wavelets defined in a similar manner [7]. Thus, we will examine the representative Coif6, which should give us good understanding of the family in general.

The six Coif6 scaling numbers are:

$$\begin{array}{ccc} \alpha_1 = \frac{1-\sqrt{7}}{16\sqrt{2}}, & \alpha_2 = \frac{5+\sqrt{7}}{16\sqrt{2}}, & \alpha_3 = \frac{14+2\sqrt{7}}{16\sqrt{2}}, \\ \alpha_4 = \frac{14-2\sqrt{7}}{16\sqrt{2}}, & \alpha_5 = \frac{1-\sqrt{7}}{16\sqrt{2}}, & \alpha_6 = \frac{-3+\sqrt{7}}{16\sqrt{2}}. \end{array}$$

The wavelet numbers are defined based on the scaling numbers:

$$\beta_i = (-1)^{i+1}\alpha_{J-i}.$$

Besides common identities, the Coif6 scaling numbers satisfy additional ones:

$$\begin{aligned} \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 + \alpha_5 + \alpha_6 &= \sqrt{2} \\ (-2)^i\alpha_1 + (-1)^i\alpha_2 + 0^i\alpha_3 + 1^i\alpha_4 + 2^i\alpha_5 + 3^i\alpha_6 &= 0 \\ \text{for } i = 1, 2, \\ \beta_1 + \beta_2 + \beta_3 + \beta_4 + \beta_5 + \beta_6 &= 0 \\ 0\beta_1 + 1\beta_2 + 2\beta_3 + 3\beta_4 + 4\beta_5 + 5\beta_6 &= 0. \end{aligned}$$

The Coif6 first level scaling signals $\mathbf{V}_n^1 = (v_1, v_2, ...v_N)$, and wavelets $\mathbf{W}_n^1 = (w_1, w_2, ...w_N)$, for $n = 1, 2, ..., N/2$, are also determined slightly differently [7]:

$$v_i = \begin{cases} \alpha_1 & \text{if } i = (2n-3) \bmod N \\ \alpha_2 & \text{if } i = (2n-2) \bmod N \\ \alpha_3 & \text{if } i = (2n-1) \bmod N \\ \alpha_4 & \text{if } i = (2n) \bmod N \\ \alpha_5 & \text{if } i = (2n+1) \bmod N \\ \alpha_6 & \text{if } i = (2n+2) \bmod N \\ 0 & \text{otherwise,} \end{cases}$$

$$w_i = \begin{cases} \beta_1 & \text{if } i = (2n-3) \bmod N \\ \beta_2 & \text{if } i = (2n-2) \bmod N \\ \beta_3 & \text{if } i = (2n-1) \bmod N \\ \beta_4 & \text{if } i = (2n) \bmod N \\ \beta_5 & \text{if } i = (2n+1) \bmod N \\ \beta_6 & \text{if } i = (2n+2) \bmod N \\ 0 & \text{otherwise.} \end{cases}$$

Formulas (8) and (9) may be employed to find higher level wavelets, scaling signals, and to perform Coiflet transforms at different levels. Coiflets share all properties with Haar and Daub$J$ wavelets.

3.5. **Wavelet Applications.** In this section, some basic applications of wavelet analysis will be introduced. The Haar wavelet will be examined in all applications as a model for other types of wavelets. Since wavelet analytical mechanisms are similar across different families of wavelets, more complex wavelets will be examined without elaborate explanation. Juxtaposition of two or more different wavelet analyses in one application will help indicate one wavelet's advantages over the others.

3.5.1. *Multiresolution Analysis.* Since discrete signals are subjects of wavelet analysis in this paper, all elementary algebraic operations such as addition, subtraction, and scalar multiplication can be performed on any two or more signals. Multiresolution analysis allows the original signal to be built up from lower resolution signals and necessary details.

Definition 7 (First Signals)**.** *The First Average Signal* $\mathbf{A^1}$ *is defined by*

$$\mathbf{A^1} = \Big(\frac{a_1}{\sqrt{2}}, \frac{a_1}{\sqrt{2}}, \frac{a_2}{\sqrt{2}}, \frac{a_2}{\sqrt{2}}, \dots, \frac{a_{N/2}}{\sqrt{2}}, \frac{a_{N/2}}{\sqrt{2}}\Big).$$

The First Detail Signal $\mathbf{D^1}$ *is defined by*

$$\mathbf{D^1} = \Big(\frac{d_1}{\sqrt{2}}, \frac{-d_1}{\sqrt{2}}, \frac{d_2}{\sqrt{2}}, \frac{-d_2}{\sqrt{2}}, \dots, \frac{d_{N/2}}{\sqrt{2}}, \frac{-d_{N/2}}{\sqrt{2}}\Big).$$

Recalling the elementary signals defined in the previous section, we have:

$$\mathbf{f} = \sum_{i=1}^{N} f_i \mathbf{V_i^0}. \tag{10}$$

The above formula is called the natural expansion of a signal $\mathbf{f}$ in terms of the natural basis of signals $\mathbf{V_1^0}$, $\mathbf{V_2^0}$, …, $\mathbf{V_N^0}$.

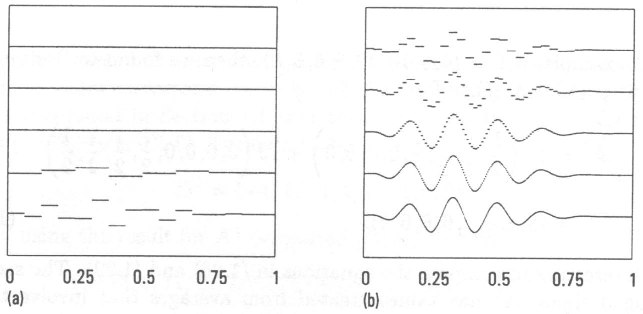


FIGURE 21. The graph of signal A built up from 10-level Haar MRA. Ten averaged signals from $\mathbf{A}^{10}$ to $\mathbf{A^1}$ are displayed from top to bottom, from left to right [7].

It follows that

$$\begin{aligned}
\mathbf{f} &= \Big(\frac{a_1}{\sqrt{2}}, \frac{a_1}{\sqrt{2}}, \frac{a_2}{\sqrt{2}}, \frac{a_2}{\sqrt{2}}, \ldots, \frac{a_{N/2}}{\sqrt{2}}, \frac{a_{N/2}}{\sqrt{2}}\Big) \\
&\qquad + \Big(\frac{d_1}{\sqrt{2}}, \frac{-d_1}{\sqrt{2}}, \frac{d_2}{\sqrt{2}}, \ldots, \frac{d_{N/2}}{\sqrt{2}}, \frac{-d_{N/2}}{\sqrt{2}}\Big) \\
&= \mathbf{A^1} + \mathbf{D^1} \\
&= \sum_{i=1}^{N/2} a_i \mathbf{V_i^1} + \sum_{i=1}^{N/2} d_i \mathbf{W_i^1} \\
&= \sum_{i=1}^{N/2} \big(\mathbf{f} \cdot \mathbf{V_i^1}\big)\mathbf{V_i^1} + \sum_{i=1}^{N/2} \big(\mathbf{f} \cdot \mathbf{W_i^1}\big)\mathbf{W_i^1}.
\end{aligned}$$

This is the first level of Haar multiresolution analysis (MRA). Since the multiple level Haar transform can be applied consecutively on average subsignals, we can expand further to obtain multiple level Haar MRA:

$$\mathbf{f} = \mathbf{A^k} + \mathbf{D^k} + \ldots + \mathbf{D^2} + \mathbf{D^1},$$

in which

$$\begin{aligned}
\mathbf{A^k} &= \sum_{i=1}^{N/2^k} \big(\mathbf{f} \cdot \mathbf{V_i^k}\big)\mathbf{V_i^k} \\
\mathbf{D^k} &= \sum_{i=1}^{N/2^k} \big(\mathbf{f} \cdot \mathbf{W_i^k}\big)\mathbf{W_i^k}.
\end{aligned}$$

The values $\big(\mathbf{f} \cdot \mathbf{V_i^k}\big)$ and $\big(\mathbf{f} \cdot \mathbf{W_i^k}\big)$ are called *wavelet coefficients.* Each of the component signals has lower resolution than $\mathbf{f}$; however, if a high-enough level of MRA is employed, the original $\mathbf{f}$ can be obtained. In Figure 21, the original signal was achieved after 10 levels of Haar MRA. Since the original signal is continuous, Daubechies wavelets yield better results. Figure 22 and Figure 23 show that more complex wavelets approach the original signal after fewer steps of MRA.

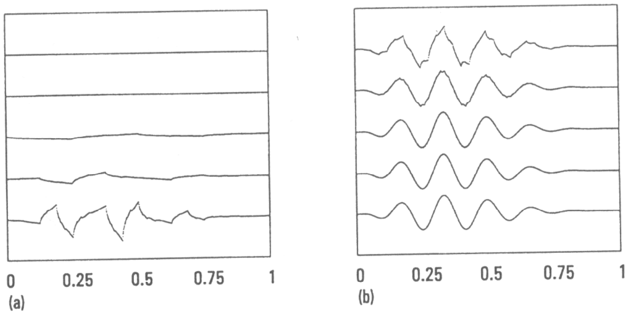


Figure 22. Daub4 MRA of the same signal. The graph are of 10 averaged signals $\mathbf{A}^{10}$ through $\mathbf{A}^1$ [7].

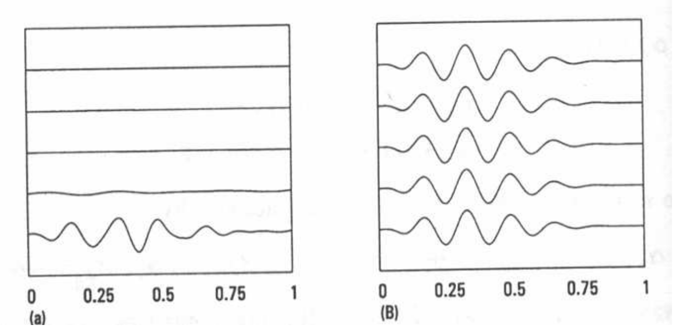


Figure 23. Daub20 MRA of the same signal. The graph are of 10 averaged signals $\mathbf{A}^{10}$ through $\mathbf{A}^1$ [7].

3.5.2. *Compression of Audio Signals.* There are two basic categories of compression techniques: lossless compression and lossy compression [7]. A lossless compression technique yields a decompression free in error from the original signal, while a decompression resulted from lossy compression suffers a degree of inaccuracy. However, the lossy compression usually succeeds more often at reducing the size of the data set. The wavelet transform is a lossy compression technique.

Method of Wavelet Transform Compression [7]

Step 1. Perform a wavelet transform of the signal

Step 2. Set equal to 0 all values of the wavelet transform which are insignificant, i.e., which lie below some threshold value.

Step 3. Transmit only the significant, non-zero values of the transform obtained from Step 2. This should be a much smaller data set than the original signal.

Step 4. At the receiving end, perform the inverse wavelet transform of the data transmitted in Step 3, assigning zero values to the insignificant values which were not transmitted. This decompression step produces an approximation of the original signal.

Figures 24 and 25 are two examples of compression using wavelet transform method. Because of the nature of Haar wavelets, discrete signals like signal 1 in Figure 24 can be more easily compressed with high degree of accuracy, while continuous signals such as signal 2 in Figure 25 are much harder to compress,

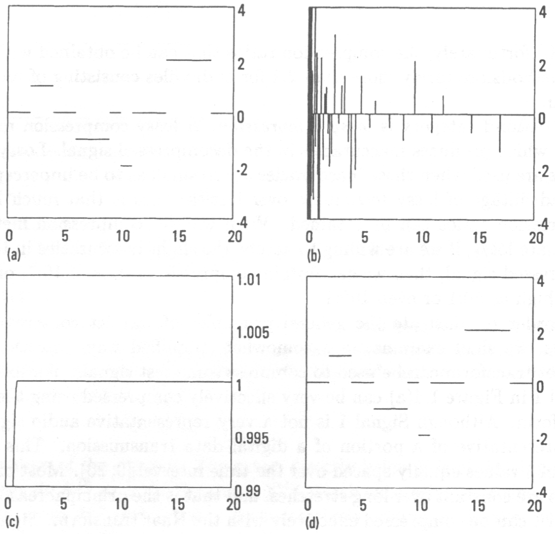


FIGURE 24. a)Original signal 1, b)10-level Haar transform of signal 1, c) energy map of Haar transform, and d) 20:1 compression of signal 1, 100% of energy[7].

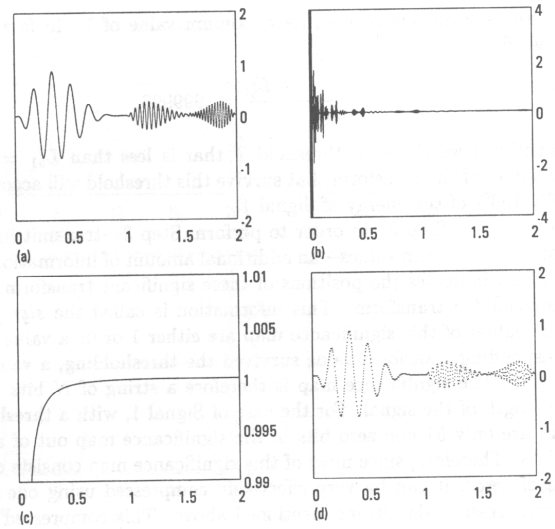


FIGURE 25. a)Signal 2, b)12-level Haar transform of signal 2, c) energy map of Haar transform, and d) 10:1 compression of the original signal, 99.6% of energy of signal 2[7].

and when they are compressed, the decompresion doesn't yield high precision even though a smaller compression size and higher level Haar transform were used.

Although Haar transforms are prominent in compressing piecewise constant signals, Daubechies transforms work much better with continuous signals. Figure 26

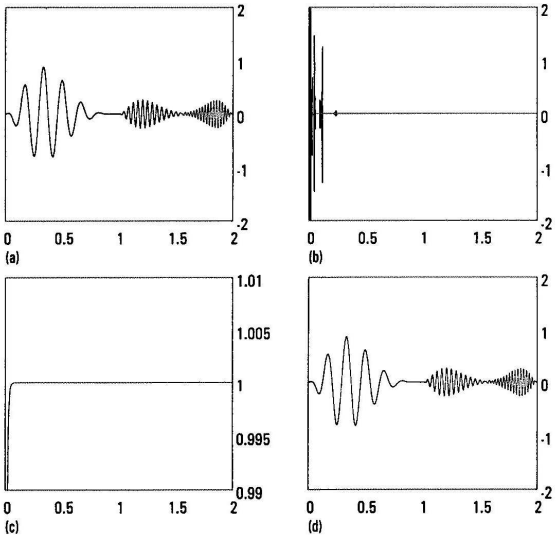


FIGURE 26. (a) The original signal. (b) 12-level Coif30 transform. (c) Energy map of the transform and (d) 32:1 compression of the signal [7].

is the 12-level Coif30 transform of signal 2. We see that a 32:1 compression of this signal by Coif30 transform yields better result when decompressed than the 10:1 compression by Haar transform.

3.5.3. *Removing Noise from Audio Signals.* When transmitted over a distance, signals are usually contaminated with noise, "the undesired change that has altered the values of the original signal" [7]. Noise is often encountered in three types [7]:

(1) *Random noise.* The noise signal is highly oscillatory, its values alternating rapidly between values above and below an average, or mean, value. We will be mostly working with random noise.
(2) *Pop noise.* This type of noise is heard on old analog recordings obtained from phonograph records. The noise is perceived as randomly occuring, isolated "pops."
(3) *Localized random noise.* Sometimes the noise appears as in type 1, but only over a short segment or segments of the signal. This can occur when there is a short-lived disturbance in the environment during transmission of the signal.

The simple model for a contaminated signal is given as follows:

$$\textit{contaminated signal} = \textit{original signal} + \textit{noise}$$

Denoting $\mathbf{f}$ as the contaminated signal, $\mathbf{s}$ as the original signal and $\mathbf{n}$ as noise, we have an equation

$$\mathbf{f} = \mathbf{s} + \mathbf{n}.$$

To filter out random noise, the *threshold method of wavelet denoising* can be implemented. In this method, only the transform values whose magnitudes are greater than a *threshold* $T_{\mathbf{s}} > 0$ will be kept. Equivalently, we can discard all

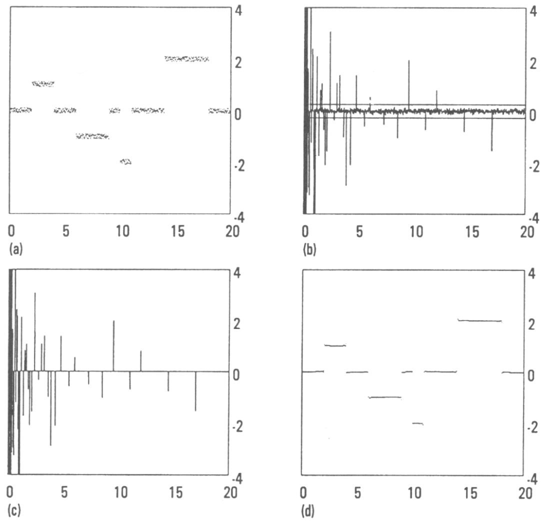


Figure 27. a) Signal B, 2^{10} values. b) 10-level Haar transform of signal B. The two horizontal lines are at values of $\pm.25$ (the denoising threshold). c) Thresholded transform. d) Denoised signal [7].

the transform values whose magnitudes lie below a *noise threshold* $T_{\mathbf{n}}$ satisfying $T_{\mathbf{n}} < T_{\mathbf{s}}$.

The *Root mean Square Error* (RMSE) is used to measure the effectiveness of noise removal method:

$$\begin{aligned} RMSE &= \sqrt{\frac{\sum_{i=1}^{N}(f_i - s_i)^2}{N}} \\ &= \sqrt{\frac{\sum_{i=1}^{N}(n_i)^2}{N}} \\ &= \frac{\sqrt{\mathcal{E}_{\mathbf{n}}}}{\sqrt{N}}. \end{aligned}$$

Smaller RMSE indicates a better denoising result. This is similar to the least square method in determining error for a set of data.

Figure 27 is a denoising example of signal 1. Part (a) of Figure 27 suggests that some random noise was added to the original signal. Part (b) shows the denoising threshold to be used on 10-level Haar transform of the contaminated signal, part (c) shows the Haar transform after the noise was filtered, and part (d) is the denoised signal. The result is fairly consistent with the original signal given in the previous report. The RMSE between signal B and signal 1 is 0.057. After denoising, the RMSE reduces to 0.011.

Figure 28 is another example of denoising method applied on signal 2. For this signal, 2^{12} values were used, together with a higher level of Haar transform (12 instead of 10), and a smaller denoising threshold (0.2 instead of 0.25). The RMSE between signal C and signal 2 is 0.057. After denoising, the RMSE is 0.035.

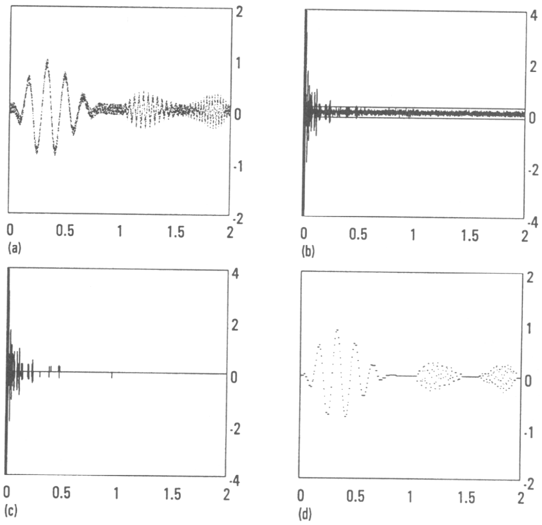


FIGURE 28. a) Signal C, 2^{12} values. b) 12-level Haar transform of signal C. The two horizontal lines are at values of ±.2 (the denoising threshold). c) Thresholded transform. d) Denoised signal [7].

Even though a higher level of Haar transform as well as a smaller denoising threshold were used, denoising of signal C still did not yield good results, which implies that Haar wavelet denoising method does not work particularly well with continuous signals such as signal 2.

Figure 29 consists of signal 2, its 12-level Coif30 transform, the thresholded transform and the denoised signal. In comparison with the denoised signal by Haar transform , this thresholded Coif30 transform yields much better result. This example shows that Daubechies wavelets once again prove to be a better approach in coping with continuous signals.

The thresholded method, however, poses a problem: what should the threshold value be? One method of choosing the threshold is to rely on the mean μ and standard deviation σ of the probability density function [7]. A further explanation can be sought in [7].

3.6. **Other Applications.** All examples introduced in the previous section are simple and more direct applications of wavelet analysis. The same analysis can be applied to more complicated data sets, such as sound signals or two dimensional images, to produce high quality compression images, or to filter the unexpected noise from a recording. Although wavelets' history can be traced back to Haar analysis in 1910, most of the development has been in the last 20 years, starting with Stromberg (1981) and Morlet (1984) [9] while working on seismology. Similar to calculus' development, wavelets quickly evolved beyond mathematics and proved their efficacy in other natural sciences, including both theoretical fields such as physics, chemistry, biology, and appplied fields such as computer science, engineering, and econometrics [9]. The technique is still being developed and put

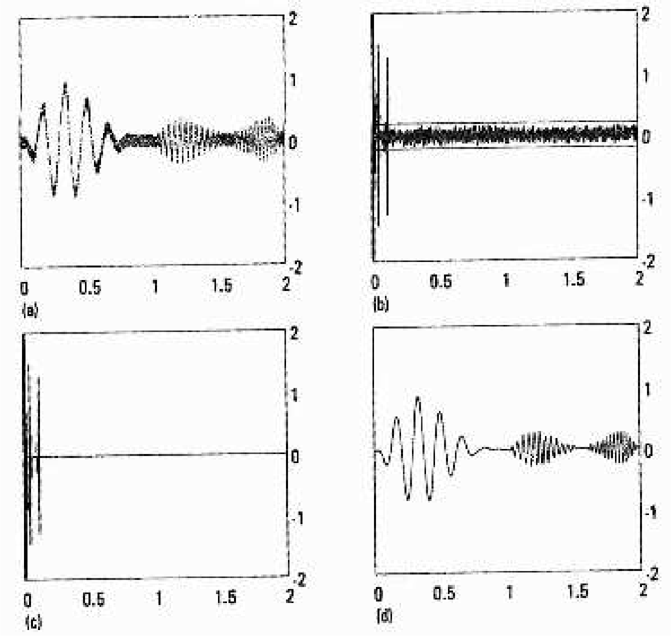


FIGURE 29. (a) The familiar signal. (b) 12-level Coif30 transform, with threshold = ±0.2. (c) Thresholded transform. (d) Denoised signal.[7]

into trial in different disciplines, among them are statisticians and stock market researchers.

4. Wavelets vs. Fourier Analysis

Throughout the previous section, Fourier analysis was occasionally compared to wavelets. This section provides a concrete example inwhich wavelets outperform Fourier analysis in yielding desired results.

The basic Fourier transform gives a global picture of a data set's spectrum, whereas wavelet transforms offer a more flexible way to examine a signal, a function or an image. In addition, wavelet transforms also provide information on where or when each frequency component is occuring. These advantages are especially embraced in studying non-stationary or inhomogeneous objects [9].

Windowed Fourier transform, as mentioned in section 3.1, is capable of obtaining a localized frequency; however, the window width remains the same along the data set, which suggests a limitation if the data set contains jumps with unexpected high or low frequencies. This problem can be resolved by wavelet analysis as it allows the degree of localization to be adjustable.

For example, consider the signal f in Figure 30, defined by:

$$\begin{aligned} f &= \sin(2\pi v_1 x)e^{-\pi(\frac{x-0.2}{0.1})^{10}} \\ &\quad +\big(\sin(2\pi v_1 x)+2\cos(2\pi v_2 x)\big)e^{-\pi(\frac{x-0.5}{0.1})^{10}} \\ &\quad +\big(2\sin(2\pi v_2 x)-\cos(2\pi v_3 x)\big)e^{-\pi(\frac{x-0.5}{0.1})^{10}}. \end{aligned}$$

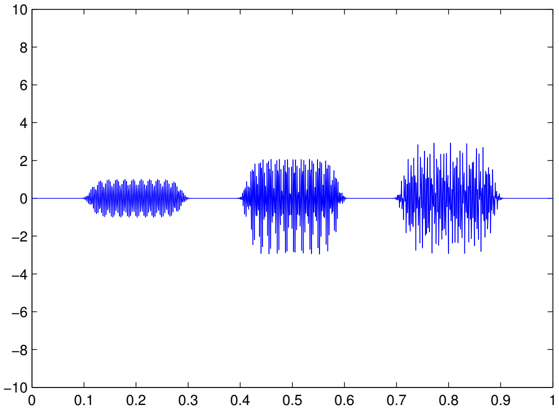


Figure 30. The original signal f.

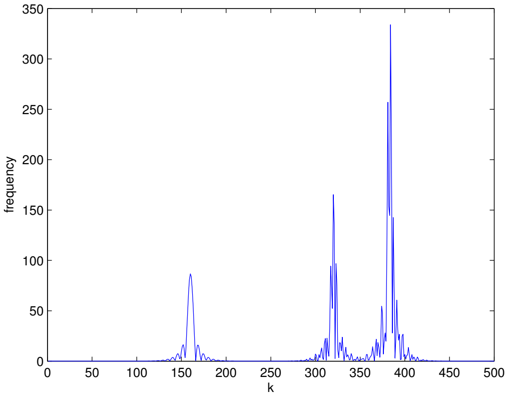


Figure 31. The power spectrum of signal f.

The signal comprises three distinct components; the first one has one dominant frequency v_1, the second one has two dominant frequencies v_1 and v_2, and the third one also has two: v_2 and v_3. Figure 31 is the Fourier transform of f, and Figure 32 displays the power spectrum of each individual component.

Fourier transform only detects the dominant frequencies without telling us where they come in. This problem can be solved by using wavelet analysis. Figure 33 plots the wavelet coefficients of signal f against time and scale (logarithm of frequencies). Both the dominant frequencies and their time of appearance are included in the figure.

5. Conclusion

Compared to other mathematical topics, wavelets are in a rather fledgling stage. However, their applications are widespread in many fields, both theoretical and

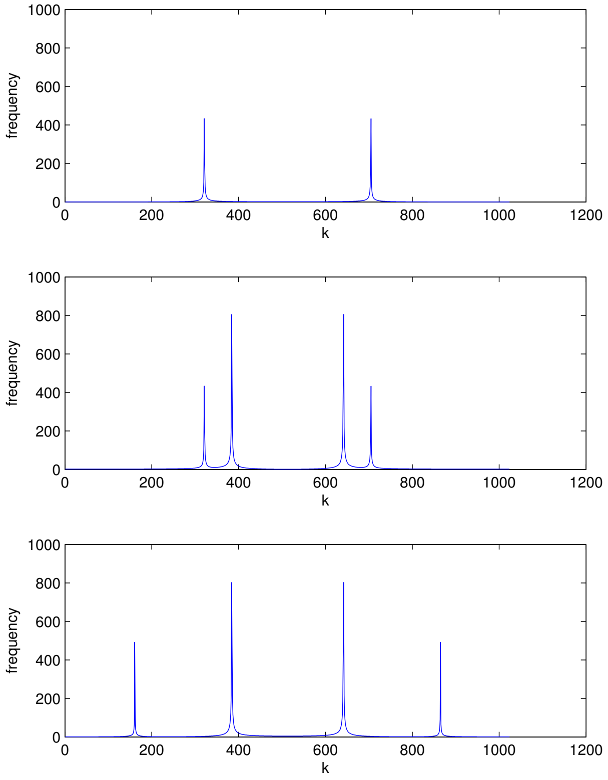


FIGURE 32. The power spectra for each component.

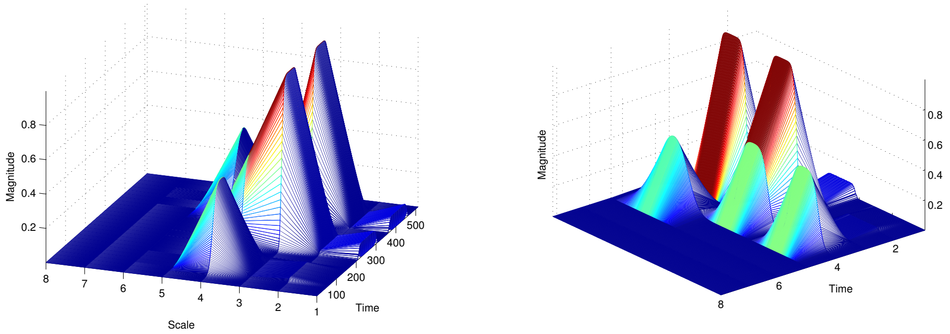


FIGURE 33. Wavelet transform of signal f. The plot is captured at two different angles

practical. Arising from the Fourier analysis' failure to cope with large and complex data files, wavelets rapidly develop to resolve these problems.

This paper briefly introduced Fourier series and chose Fast Fourier Transform (FFT) to be the representative method for Fourier analysis. A modification of FFT - the windowing process or short-term Fourier transform - was also examined as a transition before moving to wavelets. Since all wavelets are constructed similarly, the first and simplest, the Haar wavelet, was studied first and in more detail. Two other families of wavelets were mentioned in the paper: Daub$J$ wavelets and Coiflets. Some simple applications of wavelet analysis mentioned in the paper include multiresolution analysis, the denoising problem and compression of audio signals. These applications also provide a basis to make a comparison between Fourier and wavelet analyses.

All examples in this paper, including discrete data and audio signals, are one-dimensional. However, wavelet analysis is capable of dealing with higher dimensional data sets, such as pictures. Information on wavelet analysis on 2D data sets can be found in [7] or most books on applications of wavelets. More complex applications of wavelets, such as signal detection and applications in statistics, can also be topics for further study.

References

[1] WikiPedia, Fourier Series. `http://en.wikipedia.org/wiki/Fourier_series`

[2] WikiPedia, Fourier Series. `http://en.wikipedia.org/wiki/Wavelets`

[3] Stewart, James. Calculus Early Transcendentals, 4th Edition.

[4] Fourier Analysis, Course Notes, Modeling II.

[5] Sunspots. Retrieved from website `http://www.exploratorium.edu/sunspots/activity.html`.

[6] Boggess, A. and Narcowich, F., A First Course in Wavelets with Fourier Analysis. New Jersey 2001.

[7] Walker, J. , A Primer on Wavelets and their Scientific Applications. CRC Press LLC 1999.

[8] Riddle, L. Ingrid Daubechies. Biographies of Women Mathematicians. Retrieved on Dec 1, 2005 from website `http://www.agnesscott.edu/lriddle/women/daub.htm`

[9] Abramovich, F., Bailey, T. and Sapatinas, T.. Wavelet Analysis and Its Statistical Applications. The Statistician, Vol. 49, No. 1 (2000), pp 1-29.