A Tutorial of the Wavelet Transform

Chun-Lin, Liu

February 23, 2010

Chapter 1

Overview

1.1 Introduction

The Fourier transform is an useful tool to analyze the frequency components of the signal. However, if we take the Fourier transform over the whole time axis, we cannot tell at what instant a particular frequency rises. Short-time Fourier transform (STFT) uses a sliding window to find spectrogram, which gives the information of both time and frequency. But still another problem exists: The length of window limits the resolution in frequency. Wavelet transform seems to be a solution to the problem above. Wavelet transforms are based on small wavelets with limited duration. The translated-version wavelets locate where we concern. Whereas the scaled-version wavelets allow us to analyze the signal in different scale.

1.2 History

The first literature that relates to the wavelet transform is Haar wavelet. It was proposed by the mathematician Alfrd Haar in 1909. However, the concept of the wavelet did not exist at that time. Until 1981, the concept was proposed by the geophysicist Jean Morlet. Afterward, Morlet and the physicist Alex Grossman invented the term wavelet in 1984. Before 1985, Haar wavelet was the only orthogonal wavelet people know. A lot of researchers even thought that there was no orthogonal wavelet except Haar wavelet. Fortunately, the mathematician Yves Meyer constructed the second orthogonal wavelet called Meyer wavelet in 1985. As more and more scholars joined in this field, the 1st international conference was held in France in 1987.

In 1988, Stephane Mallat and Meyer proposed the concept of multiresolution. In the same year, Ingrid Daubechies found a systematical method to construct the compact support orthogonal wavelet. In 1989, Mallat proposed the fast wavelet transform. With the appearance of this fast algorithm, the wavelet transform had numerous applications in the signal processing field.

Summarize the history. We have the following table:

- 1910, Haar families.
- 1981, Morlet, wavelet concept.
- 1984, Morlet and Grossman, "wavelet".
- 1985, Meyer, "orthogonal wavelet".
- 1987, International conference in France.
- 1988, Mallat and Meyer, multiresolution.
- 1988, Daubechies, compact support orthogonal wavelet.
- 1989, Mallat, fast wavelet transform.

Chapter 2

Approximation Theory and Multiresolution Analysis

2.1 A Simple Approximation Example

Consider a periodic function $x(t)$ with period T

$$x(t) = 1 - \frac{2|t|}{T}, \quad |t| < T. \tag{2.1}$$

Find its Fourier series coefficient.

Recall the relationship of Fourier series expansion. We can decompose a periodic signal with period T into the superposition of its high order harmonics, which is the synthesis equation

$$x(t) = \sum_{k=-\infty}^{\infty} a_k \exp(j\frac{2\pi kt}{T}). \tag{2.2}$$

According to the orthogonal property of complex exponential function, we have the analysis equation

$$a_k = \frac{1}{T}\int_{-T/2}^{T/2} x(t)\exp(-j\frac{2\pi kt}{T})dt. \tag{2.3}$$

In this example the Fourier series coefficients are

$$a_k = \frac{\sin^2(\pi k/2)}{2(\pi k/2)^2} = \frac{1}{2}\mathrm{sinc}^2(k/2). \tag{2.4}$$

What's the physical meaning of the result? The Fourier series coefficient indicates the amplitude and phase of the high order harmonics, indexing by

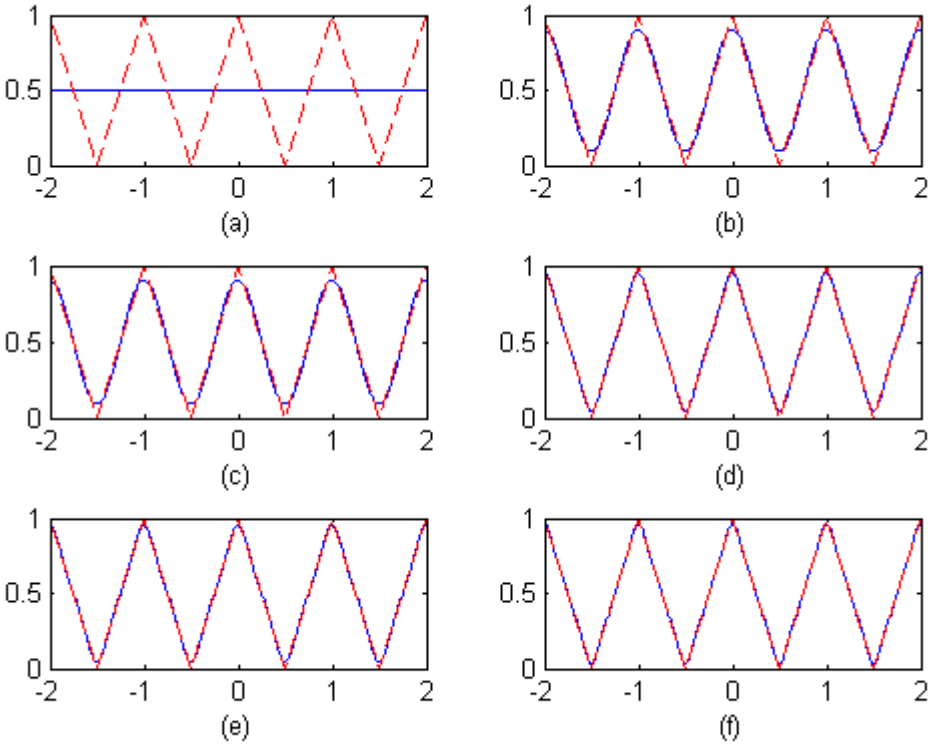


Figure 2.1: An illustration of the approximation example. The red line is the original signal. The blue dash line is the approximated signal with **(a)** $K = 0$ **(b)** $K = 1$ **(c)** $K = 2$ **(d)** $K = 3$ **(e)** $K = 4$ **(f)** $K = 5$

the variable k. The higher k is, the higher frequency refers to. In general, the power of a signal concentrates more likely in the low frequency components. If we truncate the Fourier series coefficient in the range of $[-K, K]$ and set $a_k = 0$ outside this range, we expect the synthesis version of the truncated coefficients be

$$\tilde{x}_K(t) = \sum_{k=-K}^{K} a_k \exp(j\frac{2\pi kt}{T}). \tag{2.5}$$

Intuitively, as K approaches infinity, the reconstructed signal is close to the original one. Refer to Fig. 2.1, we can see the approximation signal is very close to the original one as k grows higher.

2.2 Abstract Idea in the Approximation Example

From the point of linear algebra, we can decompose the signal into linear combination of the basis if the signal is in the the space spanned by the basis. In pp. 364-365 [1], it is,

$$f(t) = \sum_k a_k \phi_k(t). \tag{2.6}$$

where k is an integer index of the finite or infinite sum, the a_k are expansion coefficients, and the $\phi_k(t)$ are expansion functions, or the basis. Compare to 2.2, we know that Fourier series expansion is a special case when $\phi_k(t) = \exp(j2\pi kt/T)$.In general, if we choose the basis appropriately, there exists another set of basis $\{\tilde{\phi}_k(t)\}$ such that $\{\phi_k(t)\}$ and $\{\tilde{\phi}_k(t)\}$ are orthonormal. The inner product [1] is

$$< \phi_i(t), \tilde{\phi}_j(t) >= \int \phi_i(t)\tilde{\phi}_j^*(t)dt = \delta_{ij}. \tag{2.7}$$

where $\{\tilde{\phi}_k(t)\}$ is called the dual function of $\{\phi_k(t)\}$. With this orthonormal property, we can find the coefficients by

$$\begin{aligned} < f(t), \tilde{\phi}_k(t) > &= \int f(t)\tilde{\phi}_k^*(t)dt \\ &= \int (\sum_{k'} a_{k'}\phi_{k'}(t))\tilde{\phi}_k^*(t)dt \\ &= \sum_{k'} a_{k'}(\int \phi_{k'}(t)\tilde{\phi}_k^*(t)dt) \\ &= \sum_{k'} a_{k'}\delta_{k'k} \\ &= a_k. \end{aligned}$$

Rewrite as follows:

$$a_k =< f(t), \tilde{\phi}_k(t) >= \int f(t)\tilde{\phi}_k^*(t)dt. \tag{2.8}$$

The satisfying result comes because of the orthonormal property of the basis. Review 2.3, the Fourier series analysis equation is a special case when $\tilde{\phi}_k(t) =$

[1] If $x(t)$ and $y(t)$ are both complex signals, $< x(t), y(t) >= \int x(t)y^*(t)dt$. If they are both real-valued, the inner product can be reduced to $\int (x(t)y(t)dt$. The latter is more well-known.

$(1/T)\exp(j2\pi kt/T)$. Therefore, it is important to choose an appropriate set of basis and its dual. For the signal we want to deal with, apply a particular basis satisfying the orthonormal property on that signal. It is easy to find the expansion coefficients a_k. Fortunately, the coefficients concentrate on some critical values, while others are close to zero. We can drop the small coefficients and use more bits to record the important values. This process can be applied to data compression while preserving the resemblance to the original signal at the same time. In the example in the previous section, we can approximate the original by only some critical coefficients. Data compression can be achieved.

2.3 Two Examples about Multiresolutions

2.3.1 Example 1: Approximate discrete-time signals using delta function

Consider a discrete-time signal $x[n]$ defined as

$$x[n] = (\frac{1}{2})^{|n|}. \tag{2.9}$$

Now we choose a basis $\{\phi_k[n]\} = \{\delta[n-k]\}$ and its dual $\{\tilde{\phi}_k[n]\} = \{\phi_k[n]\} = \{\delta[n-k]\}$ to find the expansion coefficients of $x[n]$. Note that for the discrete case, the variable is changed to n and use brackets instead of parentheses, and the inner product is revised as summations instead of integrations. But the concept of function expansion and inner product is the same. First, we check whether the basis satisfies the orthonormal property:

$$< \phi_i[n], \tilde{\phi}_j[n] >=< \delta[n-i], \delta[n-j] >= \sum_{n=-\infty}^{\infty} \delta[n-i]\delta[n-j] = \delta_{ij}. \tag{2.10}$$

By 2.8, we can find the expansion coefficients are

$$a_k =< x[n], \delta[n-k] >= \sum_{n=-\infty}^{\infty} (\frac{1}{2})^{|n|}\delta[n-k] = (\frac{1}{2})^{|k|}. \tag{2.11}$$

Likewise, if we reserve the coefficients in the range of index $k \in [K_1, K_2]$, obtained approximations are shown in Fig.2.2.

This example selects a particular basis $\{\delta[n-k]\}$ to find its coefficients. This basis has a different interpretation: It can be viewed as translations of a single delta function $\delta[n]$. $\delta[n-k]$ means the position of the impulse is located at $n = k$. The reconstruction signal using partial coefficients means the position we concern. For example, if we want to analyze the signal when $n \in [-2, 0]$, we can use a_{-2}, a_{-1} and a_0 to find the reconstruction version.

2.3.2 Example 2: Basis Construction by Scaling

In the previous example, we construct our basis by translating a single delta function. Here, we use another common operation, scaling, to see its results.

Consider a continuous function $\phi(t)$ such that

$$\phi(t) = \begin{cases} 1 & 0 \le t < 1, \\ 0 & \text{otherwise.} \end{cases} \tag{2.12}$$

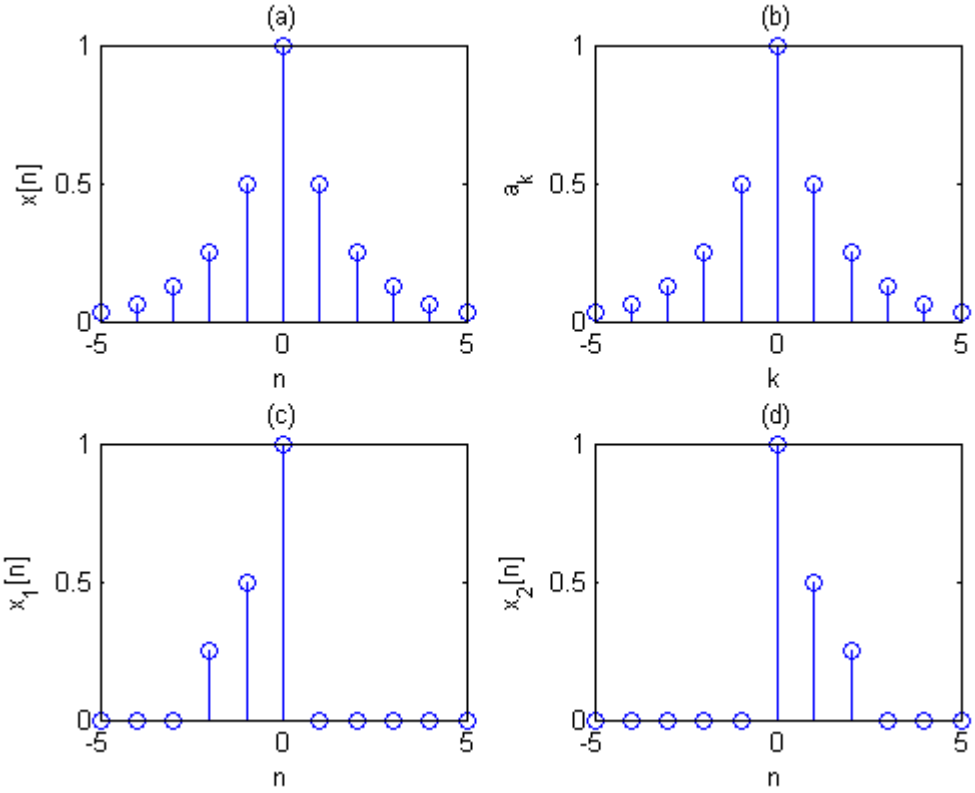


Figure 2.2: The result of approximation using delta functions. **(a)** Original signal $x[n]$ **(b)** Obtained coefficients a_k **(c)** Reconstructed signal $x_1[n] = \sum_{k=-2}^{0} a_k \delta[n-k]$ **(d)** Reconstructed signal $x_2[n] = \sum_{k=0}^{2} a_k \delta[n-k]$

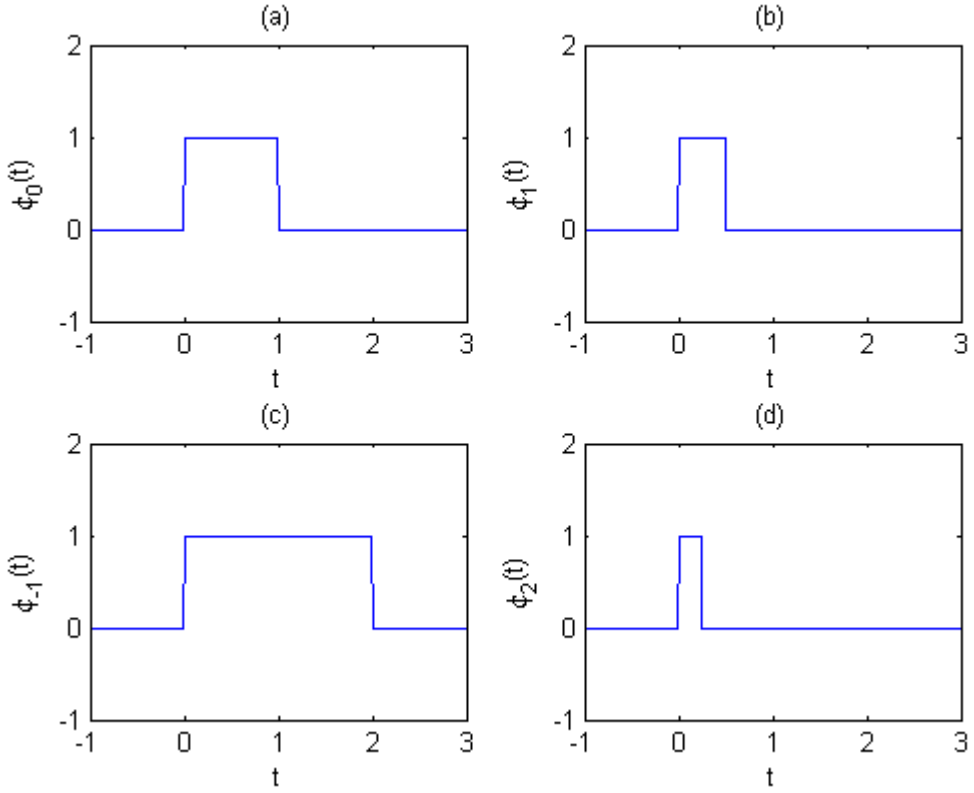


Figure 2.3: Plot of the scaled version of the basis.**(a)** $\phi_n(t) = \phi_0(t)$ **(b)** $\phi_n(t) = \phi_1(t)$ **(c)** $\phi_n(t) = \phi_{-1}(t)$ **(d)** $\phi_n(t) = \phi_2(t)$.

It is a rectangular function centered at $x = 1/2$ with width 1. The scaled version is defined as $\phi_s(t)$ such that

$$\phi_s(t) = \phi(st). \tag{2.13}$$

Note that s is a continuous scaling factor. As s higher, the support of this scaled basis will become narrower. For the discrete basis we discussed above, we let $s = 2^n$ just for simplicity, where n is an integer. Therefore, we rewrite the functions as

$$\phi_n(t) = \phi(2^n t). \tag{2.14}$$

For consistency, this tutorial use index n to indicate the scaling operation, defined as above. Fig. 2.3 shows the scaling operation. Note that once the support of the function becomes narrower, the higher frequency it has. We can approximate the high frequency component more precisely by the higher order scaling.

Here comes another question: is the family $\{\phi_n(t)\}$ a set of orthogonal family? Check the first two functions, $\phi_0(t)$ and $\phi_1(t)$

$$< \phi_0(t), \phi_1(t) >= \int_{-\infty}^{\infty} \phi_0(t)\phi_1(t)dt = \int_0^{\frac{1}{2}} dt = \frac{1}{2}.$$

Unfortunately, the two functions are not orthogonal to each other, so as the others. But don't be frustrated here, we can apply Gram-Schmidt process to obtain a set of orthonormal basis from existing $\{\phi_n(t)\}$. The procedure is

$$\begin{aligned}
\phi_0'(t) &= \phi_0(t) = \phi(t) \\
\phi_1'(t) &= \phi_1(t) - \frac{< \phi_1(t), \phi_0(t) >}{< \phi_0(t), \phi_0(t) >}\phi_0(t) \\
&= \begin{cases} 1/2 & 0 \le t < 1/2, \\ -1/2 & 1/2 \le t < 1, \\ 0 & \text{otherwise.} \end{cases} \\
&= \psi(t)/2
\end{aligned}$$

We can continuously apply this process to extend the basis. But we look at the first two basis and its Fourier transform. By inspection, $\phi(t)$ has a nonzero mean while $\psi(t)$ is zero in average. The $\phi(t)$ has two jumps at $x = 0, 1$ while $\psi(t)$ jumps at $x = 0, 1/2, 1$. Therefore, the power of $\phi(t)$ is more compact at low frequencies while the power of $\psi(t)$ concentrates at relatively high frequencies. In formal $\phi(t)$ is called scaling function to do approximation and $\psi(t)$ is called wavelet function to find the details.

2.4 First Look at Multiresoltuion Analysis

In the translating example, we found a function which is orthonormal to its translating version. Different scaled version can see different frequency resolutions. Combined with the two properties, we can construct a basis from the scaling function and wavelet function with two parameters: scaling and translating, formally defined in pp. 365-372 [1] as

$$\phi_{j,k}(t) = 2^{j/2}\phi(2^j t - k). \tag{2.15}$$

$$\psi_{j,k}(t) = 2^{j/2}\psi(2^j t - k). \tag{2.16}$$

where j is the parameter about dilation, or the visibility in frequency and k is the parameter about the position. In practice, we may want to see the whole data with "desired" resolution, i.e. for some resolution j. We define the subspace

$$V_j = \text{Span}\{\phi_{j,k}(t)\}. \tag{2.17}$$

$$W_j = \text{Span}\{\psi_{j,k}(t)\}. \tag{2.18}$$

With these formal definition, some requirements for multiresolution analysis are

1. The scaling function is orthogonal to its integer translates: In the example, the scaling function has values only in $[0, 1]$. The integer translates of the scaling function does not overlap with the original scaling function.

2. The subspaces spanned by the scaling function at low scales are nested within those spanned at higher scales: From Fig.2.3, it is clear to show that $\phi_{-1}(t) = \phi_0(t) + \phi_0(t-1)$. In the notations defined above, it's $V_j \subset V_{j+1}$.

3. The only function that is common to all V_j is $f(x) = 0$.

4. Any function can be represented with arbitrary precision.

Requirement 2 is worthy to note. For Haar scaling function (in the example), it is

$$\phi(t) = \phi_{0,0}(t) = \frac{1}{\sqrt{2}}\phi_{1,0}(t) + \frac{1}{\sqrt{2}}\phi_{1,1}(t).$$

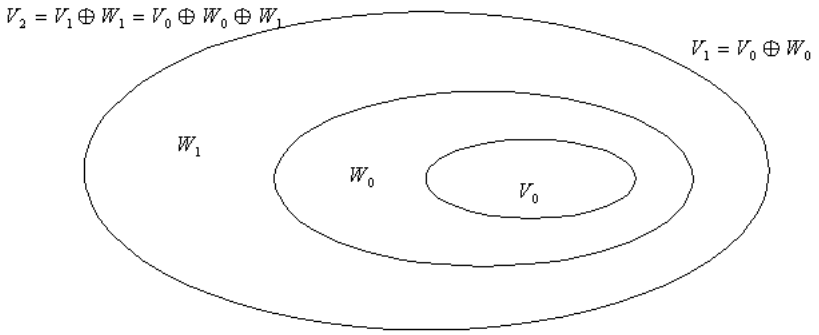


Figure 2.4: The relationship between scaling and wavelet function spaces.

Expand the notation of $\phi_{j,k}(t)$, we have

$$\phi(t) = \frac{1}{\sqrt{2}}(\sqrt{2}\phi(2t)) + \frac{1}{\sqrt{2}}(\sqrt{2}\phi(2t-1)). \tag{2.19}$$

In general, this equation is called refinement equation, multiresolution analysis equation, or dilation equation

$$\phi(t) = \sum_n h_\phi[n]\sqrt{2}\phi(2t-n). \tag{2.20}$$

It is evident to show that $h_\phi[n] = \{1/\sqrt{2}, 1/\sqrt{2}\}$ for Haar scaling functions. A physical meaning of 2.20 is that $\phi(2t)$ is a function with higher frequency components than $\phi(t)$. It is designed properly such that we apply a discrete low pass filter $h_\phi[n]$ to have $\phi(t)$.

Similar relationship exists for the wavelet functions. It is

$$\psi(t) = \sum_n h_\psi[n]\sqrt{2}\phi(2t-n). \tag{2.21}$$

Again, for Haar wavelets, $h_\psi[n] = \{1/\sqrt{2}, -1/\sqrt{2}\}$. These two filters are related by

$$h_\psi[n] = (-1)^n h_\phi[1-n]. \tag{2.22}$$

Finally, a diagram shows the relationship between the sets and the functions. V_0 is the approximation in scale 0. We can union the set spanned by different levels of wavelet functions to have higher order of approximation.

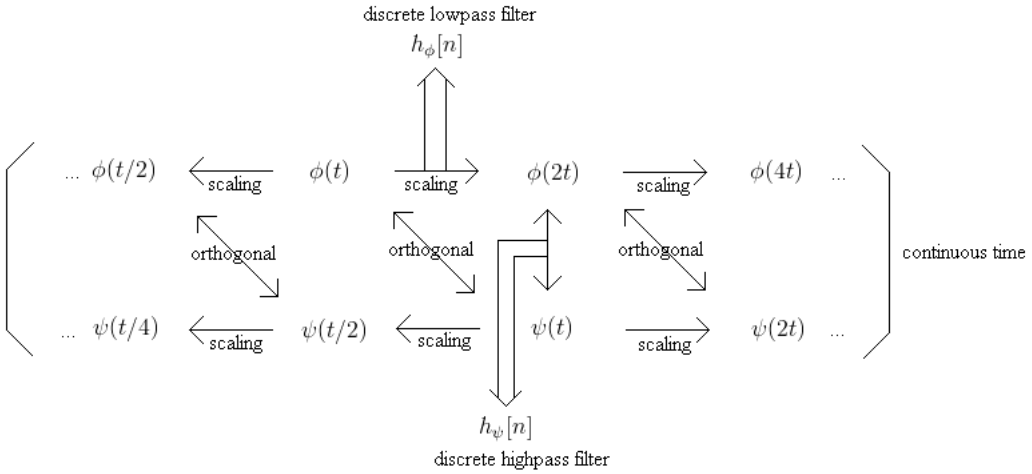


Figure 2.5: The relationship between scaling and wavelet functions.

Once we union infinite wavelet sets, the sets are equal to the $L^2(\mathbf{R})$ set[2]. This is requirement 4. In mathematical words, it is

$$L^2(\mathbf{R}) = V_0 \oplus W_0 \oplus W_1 \oplus .... \tag{2.23}$$

Therefore, we can decompose any function in $L^2(\mathbf{R})$, which is almost in all cases, using the scaling function and wavelet functions. An analogy of this concept is the corss-section of a cabbage. A cabbage is made of a kernel and a lot of leaves. A cabbage on the market can be considered as a scaling function at the finest scale J. As we take a layer of leaves off, the remaining cabbage is like the original cabbage in shape, but different in size. This is the concept of the scaling function at different scales with different supports but with similar shape. The leaves of a layer are the differences between different scales as the wavelet function. It can be used in cookery, or in mathematical analysis. We can peel the leaves until the leaves are all taken off just like the multiresolution analysis.

[2] $L^2(\mathbf{R}) = \{f(x) | \int |f(x)|^2 dx < \infty\}$

Chapter 3

Discrete Wavelet Transform

3.1 Definition

In the previous chapter, we discuss the scaling function, wavelet function, and their properties. Suppose the scaling function and wavelet function are given as Haar, Daubechies,... Say, the basis are known. We can approximate a discrete signal in $l^2(\mathbf{Z})$[1] by

$$f[n] = \frac{1}{\sqrt{M}} \sum_k W_\phi[j_0, k]\phi_{j_0,k}[n] + \frac{1}{\sqrt{M}} \sum_{j=j_0}^{\infty} \sum_k W_\psi[j, k]\psi_{j,k}[n]. \tag{3.1}$$

Here $f[n], \phi_{j_0,k}[n] and \psi_{j,k}[n]$ are discrete functions defined in $[0, M-1]$, totally M points. Because the sets $\{\phi_{j_0,k}[n]\}_{k\in\mathbf{Z}}$ and $\{\psi_{j,k}[n]\}_{(j,k)\in\mathbf{Z}^2, j\geq j_0}$ are orthogonal to each other. We can simply take the inner product to obtain the wavelet coefficients

$$W_\phi[j_0, k] = \frac{1}{\sqrt{M}} \sum_n f[n]\phi_{j_0,k}[n]. \tag{3.2}$$

$$W_\psi[j, k] = \frac{1}{\sqrt{M}} \sum_n f[n]\psi_{j,k}[n] \quad j \geq j_0. \tag{3.3}$$

3.2 are called approximation coefficients while 3.3 are called detailed coefficients.

[1] $l^2(\mathbf{Z}) = \{f[n] | \sum_{n=-\infty}^{\infty} |f[n]|^2 < \infty\}$

3.2 The Fast Wavelet Transform

Start from the definition, if the form of scaling and wavelet function is known, its coefficients is defined in 3.2 and 3.3. If we can find another way to find the coefficients without knowing the scaling and dilation version of scaling and wavelet function. The computation time can be reduced. From 2.20, we have

$$\begin{aligned}\phi_{j,k}[n] &= 2^{j/2}\phi[2^j n - k] \\ &= \sum_{n'} h_\phi[n']\sqrt{2}\phi[2(2^j n - k) - n']. \end{aligned} \tag{3.4}$$

Let $n' = m - 2k$, we have

$$\phi_{j,k}[n] = \sum_m h_\phi[m-2k]\sqrt{2}\phi[2^{j+1}n - m]. \tag{3.5}$$

If we combine the equation above with 3.2, it becomes

$$\begin{aligned}W_\phi[j,k] &= \frac{1}{\sqrt{M}}\sum_n f[n]\phi_{j,k}[n] \\ &= \frac{1}{\sqrt{M}}\sum_n f[n]2^{j/2}\phi[2^j n - k] \\ &= \frac{1}{\sqrt{M}}\sum_n f[n]2^{j/2}\sum_m h_\phi[m-2k]\sqrt{2}\phi[2^{j+1}n - m] \\ &= \sum_m h_\phi[m-2k](\frac{1}{\sqrt{M}}\sum_n f[n]2^{(j+1)/2}\phi[2^{j+1}n - m]) \\ &= \sum_m h_\phi[m-2k]W_\phi[j+1,m] \qquad (3.6) \\ &= h_\phi[-n] * W_\phi[j+1,n]\Big|_{n=2k,k\geq 0}. \qquad (3.7)\end{aligned}$$

Similarly, for the detail coefficients, it is

$$W_\psi[j,k] = h_\psi[-n] * W_\phi[j+1,n]\Big|_{n=2k,k\geq 0}. \tag{3.8}$$

For the commonly used discrete signal, say, a digital image, the original data can be viewed as approximation coefficients with order J. That is, $f[n] = W_\phi[J,n]$ By 3.7 and 3.8, next level of approximation and detail can be obtained. This algorithm is "fast" because one can find the coefficients level by level rather than directly using 3.2 and 3.3 to find the coefficients. This algorithm was first proposed in [2].

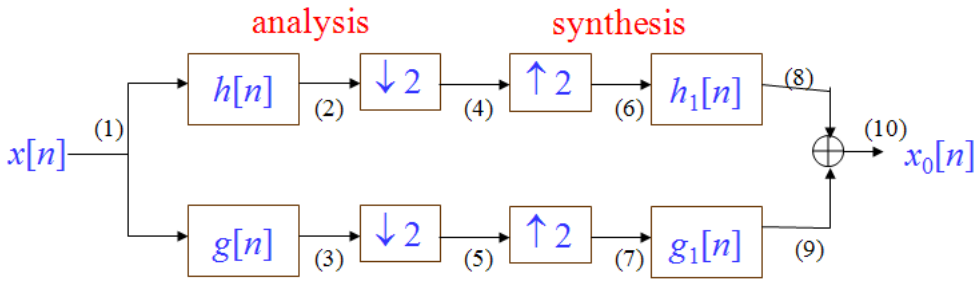


Figure 3.1: The schematic diagram to realize discrete wavelet transform. Here the filter names are changed.

3.3 Subband Coding

The satisfying result in 3.7 and 3.8 is much simpler to implement than 3.2 and 3.3. Convolve $W_\phi[j,n]$ with different filters, $h_\phi[-n]$ and $h_\psi[-n]$. Downsample by the factor of 2. We will find the next-level coefficients. Multiresolution analysis can be achieved by cascading the structure above.
A schematic diagram of this algorithm is shown in Fig. 3.1. The synthesis process does the inverse operation of the analysis process. First upsample by a factor of 2 and then convolve the signal with a known inverse filter. We expect the reconstructed signal, $x_0[n]$, to be exactly the same as the original signal, $x[n]$. Analyze each node in the subband coding schematic diagram in both time domain and z-domain, and we have

1. Time domain: $x_{(1)}[x] = x[n]$,
   z-domain: $X_{(1)}(z) = X(z)$,

2. Through a LTI system with impulse response $h[n]$,
   Time domain: $x_{(2)}[n] = x[n] * h[n]$.
   z-domain: $X_{(2)}(z) = X(z)H(z)$.

3. Time domain: $x_{(3)}[n] = x[n] * g[n]$.
   z-domain: $X_{(3)}(z) = X(z)G(z)$.

4. Time domain: $x_{(4)}[n] = x_{(2)}[2n] = \{..., x_{(2)}[-4], x_{(2)}[-2], x_{(2)}[0], x_{(2)}[2], x_{(2)}[4], ...\}$.
   z-domain:
   By definition of $X_{(2)}(z)$, we have

   $$X_{(2)}(z) = ...+x_{(2)}[-2]z^2+x_{(2)}[-1]z^1+x_{(2)}[0]+x_{(2)}[1]z^{-1}+x_{(2)}[2]z^{-2}+....$$

   Let $z = -z$, we have

   $$X_{(2)}(-z) = ...+x_{(2)}[-2]z^2-x_{(2)}[-1]z^1+x_{(2)}[0]-x_{(2)}[1]z^{-1}+x_{(2)}[2]z^{-2}+....$$

Add the two equations above

$X_{(2)}(z)+X_{(2)}(-z) = 2(...+x_{(2)}[-4]z^4+x_{(2)}[-2]z^2+x_{(2)}[0]+x_{(2)}[2]z^{-2}x_{(2)}[4]z^{-4}+...).$

$X_{(2)}(z^{\frac{1}{2}})+X_{(2)}(-z^{\frac{1}{2}}) = 2(...+x_{(2)}[-4]z^2+x_{(2)}[-2]z^1+x_{(2)}[0]+x_{(2)}[2]z^{-1}x_{(2)}[4]z^{-2}+...).$

so

$$X_{(4)}(z) = \frac{X_{(2)}(z^{\frac{1}{2}}) + X_{(2)}(-z^{\frac{1}{2}})}{2}.$$

5. Time domain: $x_{(5)}[n] = x_{(3)}[2n]$
   z-domain:
   $$X_{(5)}(z) = \frac{X_{(3)}(z^{\frac{1}{2}}) + X_{(3)}(-z^{\frac{1}{2}})}{2}.$$

6. Time domain: $x_{(6)}[n] = x_{(4)}[n/2]$ for n=multiple of 2, zero for others.
   z-domain:
   $$X_{(6)}(z) = \sum_{n=-\infty}^{\infty} x_{(6)}[n]z^{-n} = \sum_{n=-\infty}^{\infty} x_{(4)}[n/2]z^{-n} \stackrel{m=n/2}{=} \sum_{m=-\infty}^{\infty} x_{(4)}[m]z^{-2m} = X_{(4)}(z^2).$$

7. Similarly, the z-domain representation at node 7 is
   $$X_{(7)}(z) = X_{(5)}(z^2).$$

8. Convolve $x_{(6)}[n]$ with a LTI filter with impulse response $h_1[n]$. The z-domain representation is
   $$X_{(8)}(z) = H_1(z)X_{(6)}(z).$$

9. z-domain function at node 9 is found to be
   $$X_{(9)}(z) = G_1(z)X_{(7)}(z).$$

10. The reconstructed signal $x_0[n]$ is the sum of node 8 and 9. Finally, the z-domain representation is

$$\begin{aligned} X_0(z) &= X_{(8)}(z) + X_{(9)}(z) \\ &= \frac{1}{2}(H(z)H_1(z) + G(z)G_1(z))X(z) \\ &\quad +\frac{1}{2}(H(-z)H_1(z) + G(-z)G_1(z))X(-z). \end{aligned} \tag{3.9}$$

3.9 gives us some constraint to make the reconstructed signal lossless. For any signal, we let $X(z) = X_0(z)$. The following two conditions can be derived under the perfectly reconstruction assumption,

$$H(z)H_1(z) + G(z)G_1(z) = 2 \tag{3.10}$$

$$H(-z)H_1(z) + G(-z)G_1(z) = 0. \tag{3.11}$$

Rewrite 3.10 and 3.11 into matrix form, we obtain

$$\begin{bmatrix} H(z) & G(z) \\ H(-z) & G(-z) \end{bmatrix} \begin{bmatrix} H_1(z) \\ G_1(z) \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \end{bmatrix}. \tag{3.12}$$

Solve the linear system, the relationship between the decomposition and reconstruction filters is

$$\begin{bmatrix} H_1(z) \\ G_1(z) \end{bmatrix} = \frac{2}{\det(\mathbf{H_m})} \begin{bmatrix} G(-z) \\ -H(-z) \end{bmatrix} \tag{3.13}$$

where

$$\mathbf{H_m} = \begin{bmatrix} H(z) & G(z) \\ H(-z) & G(-z) \end{bmatrix}. \tag{3.14}$$

3.13 shows once the decomposition filters is determined the corresponding reconstruction filters are unique. For most case, $\mathbf{H_m}$ is invertible so $\det(\mathbf{H_m}) \neq 0$. Another sufficient and necessary condition of 3.13 is the biorthogonal relationship. It consists of four equations:

$$\begin{aligned} H(z)H_1(z) + H(-z)H_1(-z) &= 2 && (3.15)\\ G(z)G_1(z) + G(-z)G_1(-z) &= 2 && (3.16)\\ H(z)G_1(z) + H(-z)G_1(-z) &= 0 && (3.17)\\ G(z)H_1(z) + G(-z)H_1(-z) &= 0. && (3.18) \end{aligned}$$

Rewrite in the time domain, we will clearly see the biorthogonality:

$$\begin{aligned} < h[k], h_1[2n-k] > &= \delta[n] && (3.19)\\ < g[k], g_1[2n-k] > &= \delta[n] && (3.20)\\ < g_1[k], h[2n-k] > &= 0 && (3.21)\\ < h_1[k], g[2n-k] > &= 0. && (3.22) \end{aligned}$$

The biorthogonality is an important guideline to design the subband coding algorithm. A regular requirement is to make the four filter FIR filter for simplicity. From 3.13, if we let $\det(\mathbf{H_m}) =$ constant or $\alpha z^{-(2k+1)}$, $H_1(z)$ and $G_1(z)$ are in FIR form. The reconstruction filter can be discussed in three cases:

1. $\det(\mathbf{H_m}) = 2$:

$$h_1[n] = (-1)^n g[n] \tag{3.23}$$
$$g_1[n] = (-1)^{n+1} h[n]. \tag{3.24}$$

2. $\det(\mathbf{H_m}) = -2$:

$$h_1[n] = (-1)^{n+1} g[n] \tag{3.25}$$
$$g_1[n] = (-1)^n h[n]. \tag{3.26}$$

3. $\det(\mathbf{H_m}) = 2z^{-(2k+1)}$:

$$h_1[n] = (-1)^{n+1} g[n+2k+1] \tag{3.27}$$
$$g_1[n] = (-1)^n h[n+2k+1]. \tag{3.28}$$

A wavelet interpretation of this result is that the set of filters, $\{h[n], g[n]\}$, corresponds to an arbitary set of scaling and wavelet functions, $\{\phi(t), \psi(t)\}$, while another set of filters, $\{h_1[n], g_1[n]\}$, maps another set of wavelet functions, $\{\tilde{\phi}(t), \tilde{\psi}(t)\}$. These filters are tied with biorthogonal relations. This result gives us more flexibility to design the filters.

3.4 2D Wavelet Transform

Recall the 2D case in the Fourier transform, the basis are modified into

$$\exp(j(\omega_1 t_1 + \omega_2 t_2)) \tag{3.29}$$

instead of $\exp(j\omega t)$. The transformed coefficient becomes two variable functions so as the 2D wavelet transform. In pp. 386-388 [1], the scaling and wavelet function are two variable functions, denoted $\phi(x, y)$ and $\psi(x, y)$ here. The scaled and translated basis functions are defined as

$$\begin{aligned} \phi_{j,m,n}(x, y) &= 2^{j/2}\phi(2^j x - m, 2^j y - n), & (3.30)\\ \psi^i_{j,m,n}(x, y) &= 2^{j/2}\psi^i(2^j x - m, 2^j y - n), \quad i = \{H, V, D\}. & (3.31) \end{aligned}$$

There are three different wavelet functions, $\psi^H(x, y)$, $\psi^V(x, y)$ and $\psi^D(x, y)$. Conceptually, the scaling function is the low frequency component of the previous scaling function in 2 dimensions. Therefore, there is one 2D scaling function. However, the wavelet function is related to the order to apply the filters. If the wavelet function is separable, i.e. $f(x, y) = f_1(x)f_2(y)$. These functions can be easily rewritten as

$$\begin{aligned} \phi(x, y) &= \phi(x)\phi(y), & (3.32)\\ \psi^H(x, y) &= \psi(x)\phi(y), & (3.33)\\ \psi^V(x, y) &= \phi(x)\psi(y), & (3.34)\\ \psi^D(x, y) &= \psi(x)\psi(y). & (3.35) \end{aligned}$$

If we define the functions as separable functions, it is easier to analyze the 2D function and we can focus on the design of 1D wavelet and scaling functions. The analysis and synthesis equations are modified to

$$\begin{aligned} W_\phi(j_0, m, n) &= \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y)\phi_{j_0,m,n}(x, y), & (3.36)\\ W^i_\psi(j, m, n) &= \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y)\phi^i_{j,m,n}(x, y), i = \{H, V, D\} & (3.37)\\ f(x, y) &= \frac{1}{\sqrt{MN}} \sum_m \sum_n W_\phi(j_0, m, n)\phi_{j_0,m,n}(x, y)\\ &+ \frac{1}{\sqrt{MN}} \sum_{i=H,V,D} \sum_{j=j_0}^{\infty} \sum_m \sum_n W^i_\psi(j, m, n)\phi^i_{j,m,n}(x, y) & (3.38) \end{aligned}$$

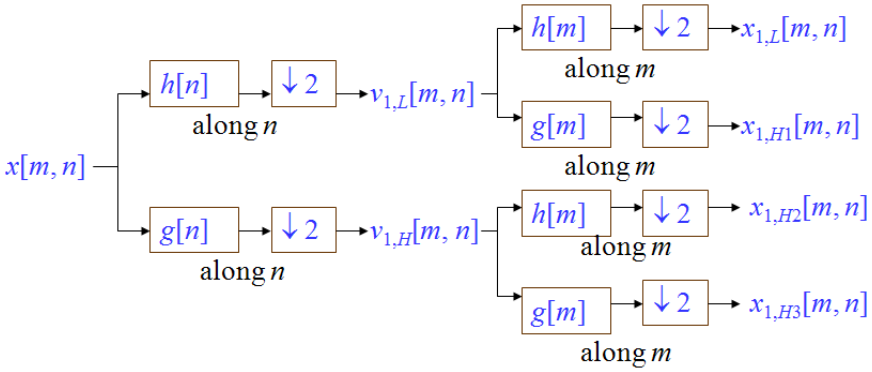


Figure 3.2: Schematic diagram of 2D wavelet transform

Figure 3.3: Original Lena image

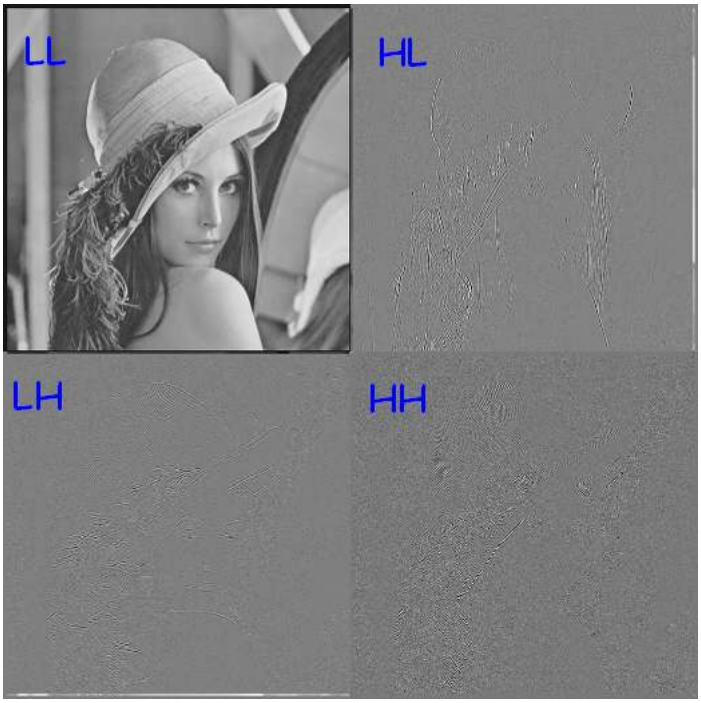


Figure 3.4: Lena image after wavelet decomposition

This is the general form of 2D wavelet transform. If the scaling and wavelet functions are separable, the summation can be decomposed into two stages. First step is along the x-axis and then calculate along the y-axis. For each axis, we can apply fast wavelet transform to accelerate the speed. A schematic diagram is shown in Fig. 3.4. The two dimensional signal (usually image) is divided into four bands: LL(left-top), HL(right-top), LH(left-bottom) and HH(right-bottom). The HL band indicated the variation along the x-axis while the LH band shows the y-axis variation. Fig. 3.3, 3.4, and 3.5 show the decomposition of a image. The power is more compact in the LL band. In the point of coding, we can spend more bits on the low frequency band and less bit on the high frequency band or even set them to zero. A famous algorithm, named Embedded Zerotree Wavelet (EZW) proposed by Shapiro [3] and some modified versions in [4] and [5] are popular.

In addition to decompose the image in two axis, Yu-Si Zhang [6] introduced a method to decompose the image along the natural edge of the image. But in general, the structure of Fig.3.4 is used due to the implementation complexity.

Figure 3.5: Reconstructed Lena image. The border is black due to the convolution.

Chapter 4

Continuous Wavelet Transform

4.1 Introduction

In this chapter, we talk about continuous wavelet transform. This transform works when we use a continuous wavelet function to find the detailed coefficients of a continuous signal. Like the concept in chapter 2, we have to establish a basis to do such analysis. First, we give the definition of continuous wavelet transform and do some comparison between that and the Fourier transform.

4.2 Definition

We define a mother wavelet function $\psi(t) \in L^2(\mathbf{R})$, which is limited in time domain. That is, $\psi(t)$ has values in a certain range and zeros elsewhere. Another property of mother wavelet is zero-mean. The other property is that the mother wavelet is normalized. Mathematically, they are

$$\int_{-\infty}^{\infty} \psi(t)dt = 0 \tag{4.1}$$

$$\|\psi(t)\|^2 = \int_{-\infty}^{\infty} \psi(t)\psi^*(t)dt = 1. \tag{4.2}$$

As the dilation and translation property states, the mother wavelet can form a basis set denoted by

$$\left\{\psi_{s,u}(t) = \frac{1}{\sqrt{s}}\psi(\frac{t-u}{s})\right\}\Big|_{u\in\mathbf{R},s\in\mathbf{R}^+}. \tag{4.3}$$

u is the translating parameter, indicating which region we concern. s is the scaling parameter greater than zero because negative scaling is undefined. The multiresolution property ensures the obtained set $\{\psi_{u,s}(t)\}$ is orthonormal. Conceptually, the continuous wavelet transform is the coefficient of the basis $\psi_{u,s}(t)$. It is

$$\begin{aligned} Wf(s,u) &= < f(t), \psi_{s,u} > \\ &= \int_{-\infty}^{\infty} f(t)\psi^*_{s,u}(t)dt \\ &= \int_{-\infty}^{\infty} f(t)\frac{1}{\sqrt{s}}\psi^*(\frac{t-u}{s})dt. \end{aligned} \tag{4.4}$$

Via this transform, one can map an one-dimensional signal $f(t)$ to a two-dimensional coefficients $Wf(s,u)$. The two variables can perform the time-frequency analysis. We can tell locate a particular frequency (parameter s) at a certain time instant (parameter u).
If the $f(t)$ is a $L^2(\mathbf{R})$ function. The inverse wavelet transform is

$$f(t) = \frac{1}{C_\psi}\int_0^{\infty}\int_{-\infty}^{\infty} Wf(s,u)\frac{1}{\sqrt{s}}\psi(\frac{t-u}{s})du\frac{ds}{s^2}, \tag{4.5}$$

where C_ψ is defined as

$$C_\psi = \int_0^{\infty} \frac{|\Psi(\omega)|^2}{\omega}d\omega < \infty. \tag{4.6}$$

$\Psi(\omega)$ is the Fourier transform of the mother wavelet $\psi(t)$. This equation is also called the admissibility condition.

4.3 An example of wavelet transform

In the previous section, we assume the mother wavelet is given with some satisfactory properties. Here we illustrate a famous mother wavelet function, called Mexican hat wavelet with

$$\psi(t) = \frac{2}{\pi^{1/4}\sqrt{3\sigma}}(\frac{t^2}{\sigma^2} - 1)\exp(-\frac{t^2}{\sigma^2}). \tag{4.7}$$

The complicated form is derived from the second derivative of a Gaussian function, $\exp(-t^2/(2\sigma^2))$. The messy constant multiplied is for the normalization property of mother wavelets. The nickname, "Mexican-hat," is because the shape of the function is like an upside-down Mexican hat. Due to the fast decay of the Gaussian function, this function drop to zero very fast. We can expect it is time-limited in a certain range, say $[-5, 5]$. This meets the property of mother wavelets.
The corresponding Fourier transform of the mother wavelet function is

$$\Psi(\omega) = \frac{-\sqrt{8}\sigma^{5/2}\pi^{1/4}}{\sqrt{3}}\omega^2\exp(-\frac{\sigma^2\omega^2}{2}). \tag{4.8}$$

It is still in a complicated form. The illustration of these two functions is in Fig. 4.1. There is a Gaussian exponential term in its Fourier spectrum. Therefore, we expect its spectrum is roughly band-limited in a certain range, too. Note that the low frequency component is relatively small compared to that at $\omega = \pm 1.5$. This result corresponds to that in the multiresolution theory: the wavelet function is a function with high frequency components. As the perform scaling on the wavelet function, the mainlobe of its spectrum will shift to higher frequency region to have finer frequency resolution.

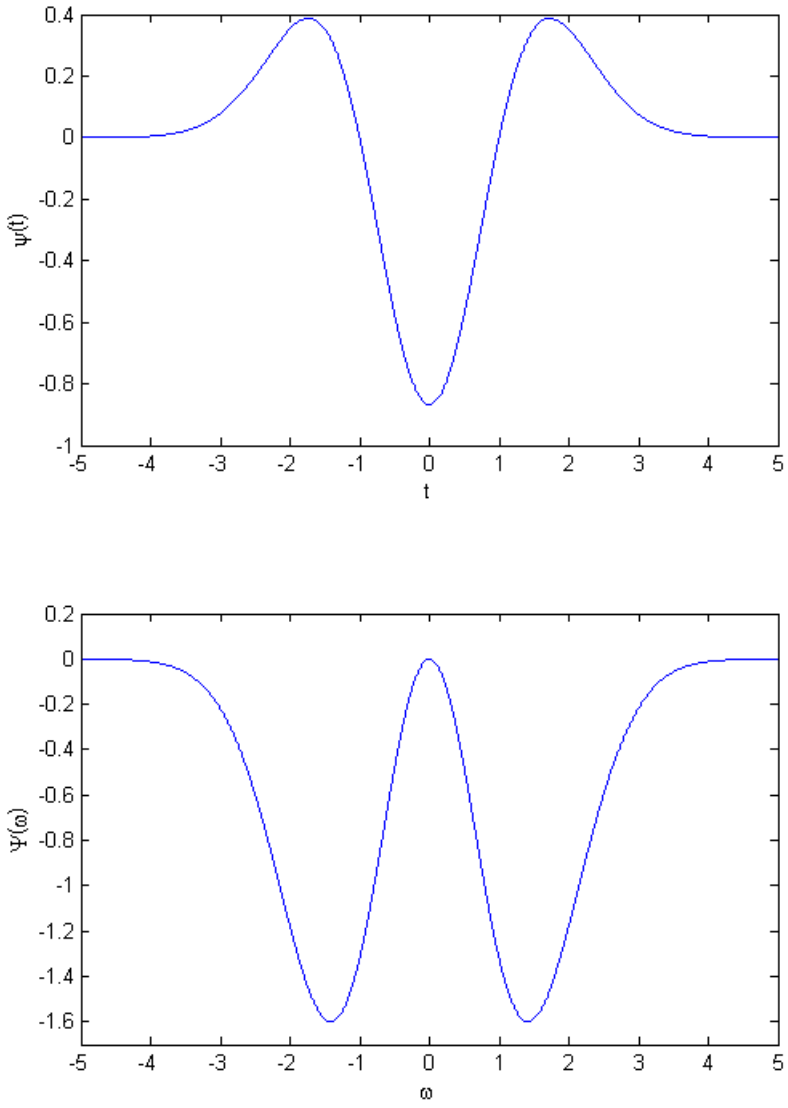


Figure 4.1: Mexican-hat wavelet for $\sigma = 1$ and its Fourier transform.

4.4 Comparison Among the Fourier Transform, Short-time Fourier Transform(STFT) and Wavelet Transform

In this section, we talk about features among the three transforms.

4.4.1 Forward transform

Fourier transform

$$F(\omega) = \int_{-\infty}^{\infty} f(t) \exp(-j\omega t) dt. \tag{4.9}$$

As we know, Fourier transform convert signal in time domain to frequency domain by integrating over the whole time axis. However, if the signal is not stationary, that is, the frequency composition is a function of time, we cannot tell when a certain frequency rises.

STFT

$$Sf(u, \xi) = \int_{-\infty}^{\infty} f(t) w(t-u) \exp(-j\xi t) dt. \tag{4.10}$$

The STFT tries to solve the problem in Fourier transform by introducing a sliding window $w(t-u)$. The window is designed to extract a small portion of the signal $f(t)$ and then take Fourier transform. The transformed coefficient has two independent parameters. One is the time parameter τ, indicating the instant we concern. The other is the frequency parameter ξ, just like that in the Fourier transform. However another problem rises. The very low frequency component cannot be detected on the spectrum. It is the reason that we use the window with fixed size. Suppose the window size is 1. If there is a signal with frequency 0.1Hz, the extracted data in 1 second look like flat (DC) in the time domain.

Wavelet transform

$$Wf(s, u) = \int_{-\infty}^{\infty} f(t) \frac{1}{\sqrt{s}} \psi^*(\frac{t-u}{s}) dt. \tag{4.11}$$

Wavelet transform overcomes the previous problem. The wavelet function is designed to strike a balance between time domain (finite length) and frequency domain (finite bandwidth). As we dilate and translate the mother wavelet, we can see very low frequency components at large s while very high frequency component can be located precisely at small s.

4.4.2 Inverse transform

Fourier transform

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) \exp(j\omega t) dt. \tag{4.12}$$

STFT

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} Sf(u, \xi) w(t-u) \exp(j\xi t) d\xi du. \tag{4.13}$$

Wavelet transform

$$f(t) = \frac{1}{C_\psi} \int_{0}^{\infty} \int_{-\infty}^{\infty} Wf(s, u) \frac{1}{\sqrt{s}} \psi(\frac{t-u}{s}) du \frac{ds}{s^2}, \tag{4.14}$$

$$C_\psi = \int_{0}^{\infty} \frac{|\Psi(\omega)|^2}{\omega} d\omega < \infty. \tag{4.15}$$

4.4.3 Basis

Here we discuss the different basis for each transforms. Fig. 4.4.2 shows the results.

Fourier transform

Complex exponential function with different frequencies:

$$\exp(j\omega t). \tag{4.16}$$

STFT

Truncated or windowed complex exponential function:

$$w(t-u) \exp(j\xi t). \tag{4.17}$$

Wavelet transform

Scaled and translated version of mother wavelets:

$$\frac{1}{\sqrt{s}} \psi(\frac{t-u}{s}). \tag{4.18}$$

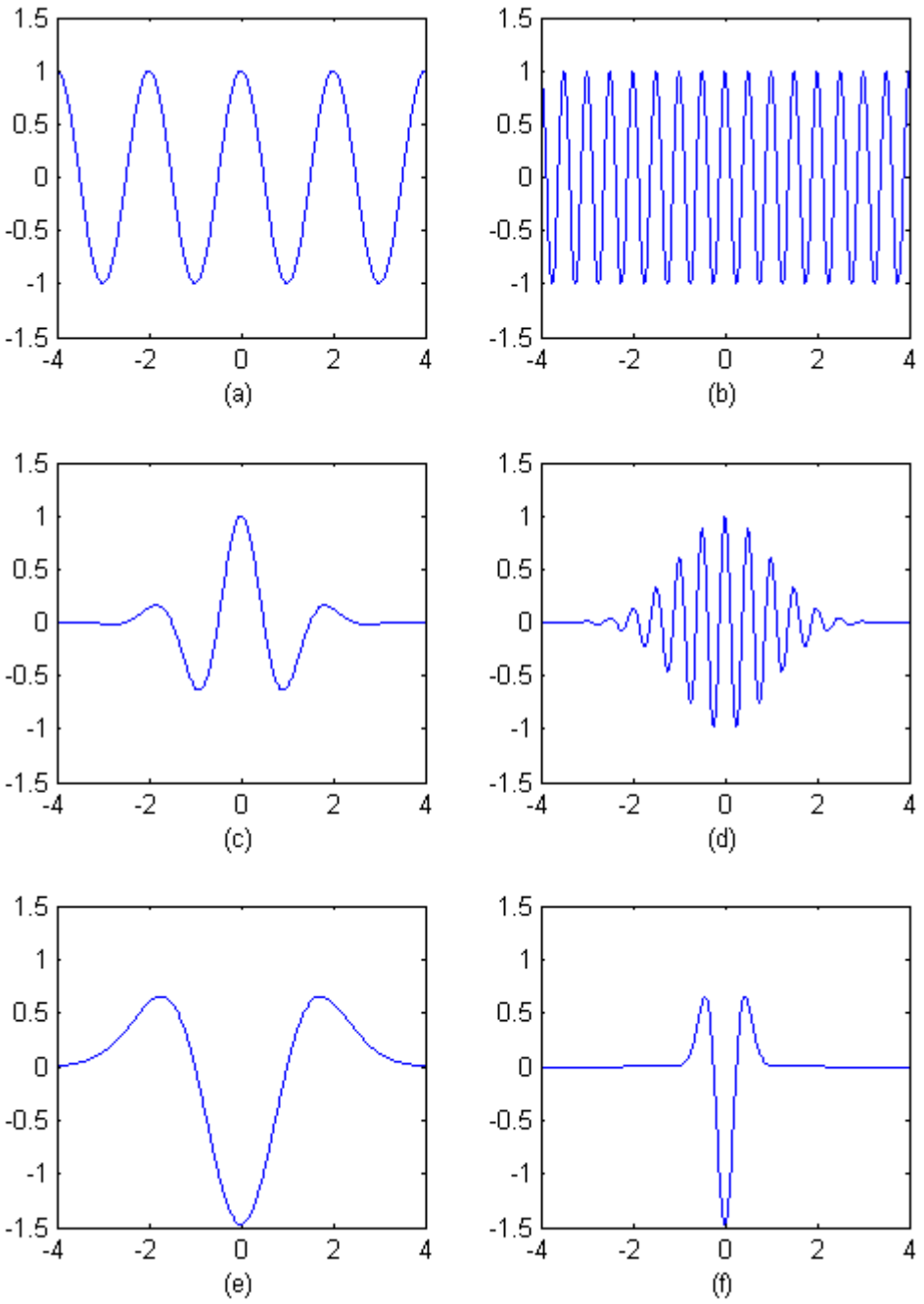


Figure 4.2: Different basis for the transforms. **(a)** Real part of the basis for Fourier transform, $\exp(j\pi t)$. **(b)** Basis for different frequency, $\exp(j4\pi t)$. **(c)** Basis for STFT, using Gaussian window of $\sigma = 1$. It is $\exp(-t^2/2)\exp(j\pi t)$. **(d)** Basis for different frequency, $\exp(-t^2/2)\exp(j4\pi t)$. **(e)** Mexican-hat mother wavelet function and **(f)** $s = 4$.

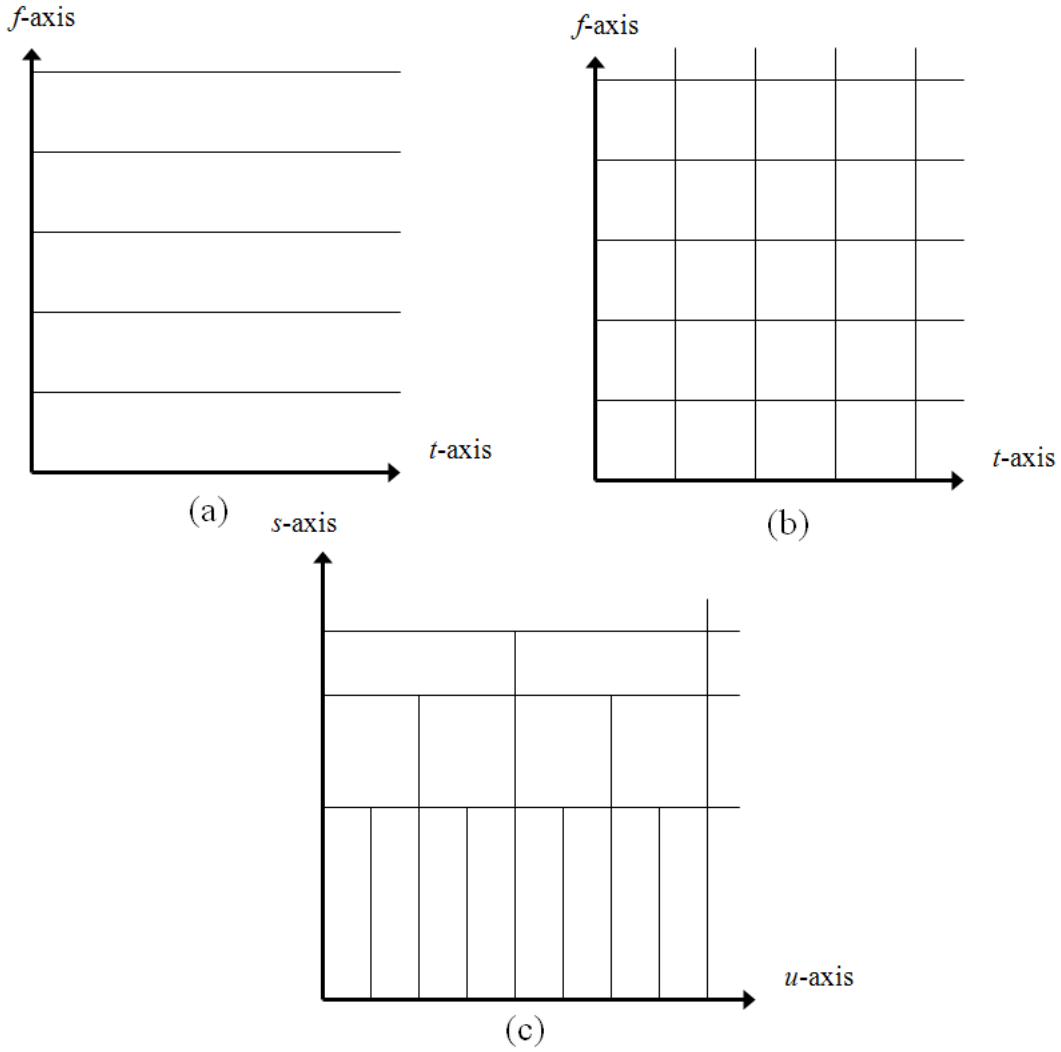


Figure 4.3: Different time-frequency tile allocation of the three transforms: **(a)** Fourier transform, **(a)** STFT and **(a)** wavelet transform.

4.4.4 Time-frequency tiling

In quantum physics, the Heisenberg uncertainty principle states that certain pairs of physical properties, like position and momentum, cannot both be known to arbitrary precision. The same principle holds in signal processing. We cannot locate both time and frequency very precisely. The product of variation in time and variation in frequency is greater than 1/2, i.e. $\sigma_t \sigma_\omega \geq 1/2$. It can be viewed as a rectangle with constant area and different transform adjusts the width and height of the rectangle. The three transforms are illustrated in Fig. 4.4.4.

Fourier transform

The time information is completely lost. Frequency axis is divided uniformly. Frequency resolution can be very precise if we integrate along the whole time axis.

STFT

Add a window to take the time domain information into consideration. The frequency resolution depends on the time resolution, or the size of the window. We cannot zoom in a particular frequency range because the box is uniformly placed.

Wavelet transform

The s parameter is inversely proportional to the frequency. As we see, if we want to focus on low frequencies, larger s is used while higher frequencies uses small s. This flexibility increases the time-frequency analysis.

Chapter 5

Some Common Wavelet Functions

In this chapter, we discuss how to design a mother wavelet. The design equations determine the coefficients of the scaling/wavelet functions and the corresponding lowpass/highpass filters. We will go through some popular wavelet functions and illustrate how they are designed. Original lectures can be found in [7].

5.1 Design Equations

5.1.1 Scaling equation

First, we examine (2.20) and (2.21), which can be considered as the basic definition of scaling and wavelet functions. Setting $t = t/2$ and we have

$$\frac{1}{\sqrt{2}}\phi(\frac{t}{2}) = \sum_n h_\phi[n]\phi(t-n), \tag{5.1}$$

$$\frac{1}{\sqrt{2}}\psi(\frac{t}{2}) = \sum_n h_\psi[n]\phi(t-n). \tag{5.2}$$

Taking the Fourier transform on both equations, we obtain

$$\frac{1}{\sqrt{2}}\Phi(2\omega) = H_\phi(\omega)\Phi(\omega), \tag{5.3}$$

$$\frac{1}{\sqrt{2}}\Psi(2\omega) = H_\psi(\omega)\Phi(\omega), \tag{5.4}$$

where $\Phi(\omega)$ and $\Psi(\omega)$ are the Fourier transform of the scaling and wavelet functions, respectively. $H_\phi(\omega)$ and $H_\psi(\omega)$ are the discrete-time Fourier trans-

form of the discrete filters. If we substitute the recursive equation continuously, we obtain

$$\Phi(\omega) = \prod_{p=1}^{\infty} \frac{H_\phi(\omega)}{\sqrt{2}} \Phi(0). \tag{5.5}$$

By (5.5), once we have the lowpass filter, we can calculate the corresponding scaling function by its Fourier transform. If we know the scaling function in advance, 5.3 helps us to find the lowpass filter.

5.1.2 Conjugate mirror filter

The scaling equation only gives us a connection between the scaling and the corresponding filters. There is still another relationship holds for the filters. It is

$$|H_\phi(\omega)|^2 + |H_\phi(\omega + \pi)|^2 = 2, \tag{5.6}$$

and

$$H_\phi(0) = \sqrt{2}. \tag{5.7}$$

The filters that satisfy this relationship is called conjugate mirror filters. Its proof is technical. One can refer to pp. 271-276 [8] to see the details.

The similar equation for wavelet highpass filter is

$$|H_\psi(\omega)|^2 + |H_\psi(\omega + \pi)|^2 = 2, \tag{5.8}$$

and

$$H_\psi(\omega) H_\phi^*(\omega) + H_\psi(\omega + \pi) H_\phi^*(\omega + \pi) = 0. \tag{5.9}$$

These equations might not strange to you because similar equations appear in (3.15), using subband coding approach. We usually set

$$H_\psi(\omega) = -\exp(-j\omega) H_\phi^*(\omega + \pi). \tag{5.10}$$

It is a trivial solution to the two constraints. Calculating the inverse transform yields

$$h_\psi[n] = (-1)^n h_\phi[1 - n]. \tag{5.11}$$

5.1.3 Real filters

For simplicity, we want the obtained filters to be real-coefficient. This yields

$$H_\phi(\omega) = H_\phi^*(-\omega) \tag{5.12}$$

$$H_\psi(\omega) = H_\psi^*(-\omega) \tag{5.13}$$

5.1.4 Design lowpass filter

Here we have all the design equations and we can focus on the design of the lowpass filter. Note that we only have to design the frequency band $\omega \in [0, \pi/2]$. the value in $[\pi/2, 3\pi/2]$ can be obtained using (5.6) and $[3\pi/2, 2\pi]$ band is the reflection version due to the real filter property.

5.2 Some Design Issues

Besides the above design equations, we want other properties which are useful in the general case. Here, we mention the vanishing moments, the size of support, support versus moments, and regularity. These properties are more strongly related to the practical use of these functions.

5.2.1 Vanishing moments

The vanishing moment is a criterion about how a function decays toward infinity. For example, the function $\sin t/t^2$ decays at a rate of $1/t^2$ as t approaches to infinity. We can estimate the rate of decay by the following integration,

$$\int_{-\infty}^{\infty} t^k f(t)dt. \tag{5.14}$$

The parameter k indicates the rate of decay. Take $\sin t/t^2$ for an example again, if $k = 0$, the function $t^k f(t)$ behaves like a function decays at a rate of $1/t^2$ and the $\sin t$ term makes the function oscillating between $1/t^2$ and $-1/t^2$. These properties ensured that the integral converges to zero for $k = 0$. For $k = 1$, the well-known sine integral gives the result to be π. For higher k, the integral diverges. Therefore, we can indicate the decay rate from the parameter k. We say the wavelet function $\psi(t)$ has p vanishing moments if

$$\int_{-\infty}^{\infty} t^k \psi(t)dt = 0 \quad \text{for } 0 \le k < p. \tag{5.15}$$

The definition seems not useful in design of the wavelet functions because it involves a continuous integration. A theorem (pp. 284 in [8]) shows the four statements are equivalent if $|\psi| = O((1+t^2)^{-p/2-1})$ and $|\phi(t)| = O((1+t^2)^{-p/2-1})$,

1. The wavelet ψ has p vanishing moments.
2. The Fourier transform of $\phi(t)$, $\psi(t)$ and its first $p-1$ derivatives are zero at $\omega = 0$.
3. The Fourier transform of $h_\phi[n]$, $H_\phi(e^{j\omega})$ and its first $p-1$ derivatives are zero at $\omega = \pi$.
4. For any $0 \le k < p$,

$$q_k(t) = \sum_{n=-\infty}^{\infty} n^k \phi(t-n) \tag{5.16}$$

is a polynomial of degree k.

If we combine statement 3 with the conjugate mirror filter condition (5.6), we can write the lowpass filter as

$$H_\phi(e^{j\omega}) = \sqrt{2}(\frac{1+e^{j\omega}}{2})^p L(e^{j\omega}) \tag{5.17}$$

Where $L(x)$ is a polynomial. This result simplifies the process of wavelet design.

5.2.2 Size of support

The size of support indicates the filter length. Note that we prefer the discrete filter to be FIR due to the stability and implementation issue. A theorem (pp. 286 in [8]) states that if the support of $\phi(t)$ and $h_\phi[n]$ is $[N_1, N_2]$, the support of $\psi(t)$ is $[(N_1 - N_2 + 1)/2, (N_2 - N_1 + 1)/2]$.

5.2.3 Support versus Moments

Due to (5.17), if we choose a high order vanishing moment, $H_\phi(e^{j\omega})$ is a high order polynomial of $e^{j\omega}$. The corresponding $h_\phi[n]$ must have longer filter size. This is a trade-off between the vanishing moments and the filter length.

5.2.4 Regularity

The regularity of ψ has significant influence on the error introduced by thresholding or quantizing the wavelet coefficients. When reconstructing a signal from wavelet coefficients

$$f = \sum_{j=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} < f, \psi_{j,n} > \psi_{j,n}, \tag{5.18}$$

the reconstruction error is related to the wavelet function we choose.

5.3 Daubechies wavelets

Based on these equation, Daubechies [9], designed a type of wavelet for a given vanishing moment p and find the minimum size discrete filter. The conclusion is that if we want the wavelet function with p vanishing moments, the minimum filter size is $2p$. The derivation first starts from 5.17, rewrite as

$$H_\phi(e^{j\omega}) = \sqrt{2}(\frac{1+e^{-j\omega}}{2})^p R(e^{j\omega}). \tag{5.19}$$

The absolute-square of this function is

$$\begin{aligned} |H_\phi(e^{j\omega})|^2 &= H_\phi(e^{j\omega})H_\phi^*(e^{j\omega}) \\ &= 2(\frac{1+e^{-j\omega}}{2}\frac{1+e^{j\omega}}{2})^p R(e^{j\omega})R^*(e^{j\omega}) \\ &= 2(\frac{2+e^{j\omega}+e^{-j\omega}}{4})^{2p}|R(e^{j\omega})|^2 \\ &= 2(\cos\frac{\omega}{2})^{2p}P(\sin^2\frac{\omega}{2}). \end{aligned} \tag{5.20}$$

The last step makes $P(\sin^2\frac{\omega}{2}) = |R(e^{j\omega})|^2$. Recall (5.6), we can determine the form of $P(x)$. Let $y = \sin^2\frac{\omega}{2}$. We have

$$(1-y)^p P(y) + y^p P(1-y) = 1. \tag{5.21}$$

A theorem in algebra, called Bezout theorem, can solve this equation. The unique solution is

$$P(y) = \sum_{k=0}^{p-1} \binom{p-1+k}{k} y^k. \tag{5.22}$$

The polynomial $P(y)$ is the minimum degree polynomial satisfying (5.21). Once we have $P(y)$, the polynomial $R(e^{j\omega})$ can be derived. First we decompose $R(e^{j\omega})$ according to its roots

$$R(e^{j\omega}) = \sum_{k=0}^{m} r_k e^{-jk\omega} = r_0 \prod_{k=0}^{m}(1 - a_k e^{-j\omega}). \tag{5.23}$$

Let $z = e^{j\omega}$, the relation between P and R is

$$P(\frac{2-z-z^{-1}}{4}) = r_0^2 \prod_{k=0}^{m}(1-a_k z^{-1})(1-a_k z). \tag{5.24}$$

By solving the roots of $P(\frac{2-z-z^{-1}}{4}) = 0$, we have the roots of R, $\{a_k, 1/a_k\}_{k=0,1,\ldots,m}$ and $r_0 = 2^{p-1}$. Usually, we choose a_k lies in the unit circle to have minimum phase filter.

Taking $p = 2$ for a example. The obtained polynomial $P(y)$ is

$$P(y) = \sum_{k=0}^{1} \binom{1+k}{k} y^k = 1 + 2y. \tag{5.25}$$

$$P(\frac{2 - z - z^{-1}}{4}) = 2 - \frac{1}{2}z - \frac{1}{2}z^{-1}. \tag{5.26}$$

The roots are $2 + \sqrt{3}$ and $2 - \sqrt{3}$. After factorization, we have the lowpass filter to be

$$H_\phi(e^{j\omega}) = \frac{\sqrt{2}+\sqrt{6}}{8} + \frac{3\sqrt{2}+\sqrt{6}}{8}e^{-j\omega} + \frac{3\sqrt{2}-\sqrt{6}}{8}e^{-j2\omega} + \frac{\sqrt{2}-\sqrt{6}}{8}e^{-j3\omega}. \tag{5.27}$$

The discrete-time domain representation is

$$h_\phi[n] = \frac{\sqrt{2}+\sqrt{6}}{8}\delta[n] + \frac{3\sqrt{2}+\sqrt{6}}{8}\delta[n-1] + \frac{3\sqrt{2}-\sqrt{6}}{8}\delta[n-2] + \frac{\sqrt{2}-\sqrt{6}}{8}\delta[n-3]. \tag{5.28}$$

The result is the minimum size filter with 2 vanishing moments and the corresponding filter size is 4. Recall the conclusion mentioned above, the filter size is two times the vanishing moment. Higher order Daubechies wavelets are derived at similar way. The coefficient and the plot are shown in the appendix A.

5.4 Symlets

Take a look at the discrete filters and the scaling/wavelet functions of Daubeches wavelets. These functions are far from symmetry. That's because Daubechies wavelets select the minimum phase square root such that the energy concentrates near the starting point of their support. Symmlets select other set of roots to have closer symmetry but with linear complex phase. The coefficients of these filters are list in the Appendix B.

5.5 Coiflets

For an application in numerical analysis, Coifman asked Daubechies [9] to construct a family of wavelets ψ that have p vanishing moments, minimum-size support and

$$\int_{-\infty}^{\infty} \phi(t)dt = 1, \tag{5.29}$$

$$\int_{-\infty}^{\infty} t^k \phi(t) dt = 0 \quad \text{for } 1 \leq k < p. \tag{5.30}$$

The equation above can be taken as some requirement about vanishing moments of the scaling function. The resulting coiflets has a support of size $3p - 1$. These coefficient of the filters are shown in Appendix C.

Chapter 6

Applications of Wavelet Transform

6.1 Introduction

In the previous sections, we mention the fundamental concepts of the wavelet transform. More than a theory, the wavelets are widely used in many applications. Its ability of multiresolution outperforms Fourier-based transform such as discrete cosine transform in terms of compression and quality. Here, we talk more about JPEG200, which is wavelet-based, and some applications of the wavelet transform.

6.2 Image Compression: JPEG 2000

Before JPEG 2000, we mention the JPEG image compression standard. JPEG is based the discrete cosine transform (DCT). However, this standard divides the original image into 8-by-8 blocks and do DCT. Use different quantization tables for luminance and chrominance. We have a good compression ratio and maintain the quality of the original image. If the compression ratio is pretty high, the block effect is perceivable. This effect originates from the artificial division of the blocks. The neighboring blocks might exhibit similar color, but after quantization and inverse quantization, the reconstructed blocks differ in color. The discontinuity between blocks emerges. Fig. 6.2 shows the JPEG coding system.

JPEG 2000 adapts the wavelet transform. The algorithm dynamically selects the high power part and records them. At low compression ratio, the performance of JPEG and JPEG 2000 is almost the same. At high compression ratio, the performance of JPEG 2000 is much better than that

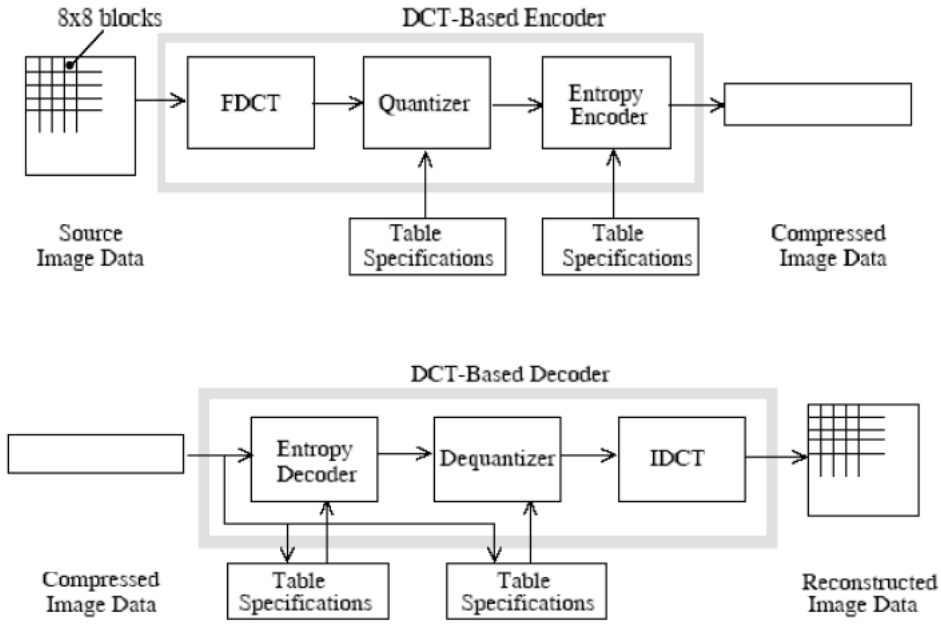


Figure 6.1: The simple system diagram of JPEG

of the JPEG because it has no block effect. See Fig. (6.2).

JPEG 2000 is a wavelet-based image compression standard. It was proposed by the ISO JPEG committee in the year 2000 with the intention of replacing their original DCT-based JPEG standard which was proposed in the year 1991. Different from JPEG, JPEG 2000 regards the whole image as a unit and take 2D wavelet transform many levels. Due to the discussion in the previous chapters, the most important coefficients concentrates in the LL band. We can focus ourselves on how to encode the coefficients by its significance. The coding method [3], [4], [5], [10] are good references.

JPEG 2000 uses two different wavelet transforms, one is biorthogonal Daubechies 5/ 3 for lossless compression and a Daubechies 9 /7 wavelet for lossy compression. The Daub 9/7 wavelet is an irreversible transform because it has quantization noise that depends on the precision of the decoder. The Daub 5/ 3 is reversible because the coefficients are integers, we do not have to round the results. At low compression ratio, the performance of JPEG and JPEG 2000 is almost the same. At high compression ratio, the performance is much better than the JPEG because it has no block effect.

JPEG 2000 requires much computation time than JPEG on the similar compression rates. [11] proposes a new calculation way to compute the convolution and it has much less computation complexity than the usual

Figure 6.2: Comparison between JPEG and JPEG 2000. CR stands for compression ration and RMSE means root mean square error.

convolution.

An overview of the JPEG 2000 system can be found in [12], [13].

6.3 Other Applications

6.3.1 Edge and Corner Detection

An important issue in pattern recognition is to locate the edges. With wavelet transform, we can anaylze a portion of a signal with different scales. We can distinguish the noise and actual corner more precisely.

6.3.2 Pattern recognition

Pattern recognition depends on computing the features of the data. If we first condense the signal by the wavelet transform, the computation effort can be reduced. Once the pattern is probably recognized, we can use wavelet transform to do finer analysis.

6.3.3 Filter design

As the wavelet transform can distinguish the noise and signal edge. We can design an anti-noise filter without distortion of the original signal.

6.3.4 Analysis of the Electrocardiogram (ECG)[14]

An ECG is not smooth with many sudden transitions. If we analyze the spectrum, noises and desired signal transitions cannot be separated using ordinary filters. Here wavelet transform can also remove the noise and preserve the features of ECG.

There are still another applications which are not noted. In short, the wavelet transform is a powerful tool solving the problems beyond the scope of Fourier transform.

Chapter 7

Conclusions

In this tutorial, we explore the world of wavelets. From the abstract idea in approximation, multiresolition theory, we generalize the Fourier transform and start the journey of wavelets. The discrete wavelet is more useful in realization. We often use 2D wavelets to do image compression. The continuous wavelet transform analyze the continuous-time signal in a different perspective. By the advantage of multiresolution, we can locate time and frequency more accurately. The wavelet design is more complicated in mathematics but the design procedure completes the existence of the wavelets. The application chapter mentions the nowadays JPEG and JPEG2000 standard.

[8] and [9] give a thorough approach to the wavelets but in purely mathematically words. [1] and [15] illustrate some examples on the wavelet and then the abstract concepts. Other references mainly focus on some parts of the materials.

The wavelets bring us to a new vision of signal processing. It tactically avoids the problem that Fourier analysis encounters. Its implementation is simple. We need some designed filters to do the task. Although the implementation is quite easy, the filter design includes lots of mathematical originality and this is the research topic. Once we establish wavelets with more ideal properties, lots of difficult problems might be solved.

Appendix A

Daubechies wavelets

A.1 Coefficients

A.1.1 p=2

[1]

n	LoD	HiD	LoR	HiR
0	-0.129409523	-0.482962913	0.482962913	-0.129409523
1	0.224143868	0.836516304	0.836516304	-0.224143868
2	0.836516304	-0.224143868	0.224143868	0.836516304
3	0.482962913	-0.129409523	-0.129409523	-0.482962913

A.1.2 p=3

n	LoD	HiD	LoR	HiR
0	0.035226292	-0.332670553	0.332670553	0.035226292
1	-0.085441274	0.806891509	0.806891509	0.085441274
2	-0.13501102	-0.459877502	0.459877502	-0.13501102
3	0.459877502	-0.13501102	-0.13501102	-0.459877502
4	0.806891509	0.085441274	-0.085441274	0.806891509
5	0.332670553	0.035226292	0.035226292	-0.332670553

[1]The filters are abbreviated in Lo/Hi (lowpass/highpass) and D/R (decomposition/reconstruction) in subband coding.

A.1.3 p=4

n	LoD	HiD	LoR	HiR
0	-0.010597402	-0.230377813	0.230377813	-0.010597402
1	0.032883012	0.714846571	0.714846571	-0.032883012
2	0.030841382	-0.630880768	0.630880768	0.030841382
3	-0.187034812	-0.027983769	-0.027983769	0.187034812
4	-0.027983769	0.187034812	-0.187034812	-0.027983769
5	0.630880768	0.030841382	0.030841382	-0.630880768
6	0.714846571	-0.032883012	0.032883012	0.714846571
7	0.230377813	-0.010597402	-0.010597402	-0.230377813

A.1.4 p=5

n	LoD	HiD	LoR	HiR
0	0.003335725	-0.160102398	0.160102398	0.003335725
1	-0.012580752	0.60382927	0.60382927	0.012580752
2	-0.00624149	-0.724308528	0.724308528	-0.00624149
3	0.077571494	0.138428146	0.138428146	-0.077571494
4	-0.03224487	0.242294887	-0.242294887	-0.03224487
5	-0.242294887	-0.03224487	-0.03224487	0.242294887
6	0.138428146	-0.077571494	0.077571494	0.138428146
7	0.724308528	-0.00624149	-0.00624149	-0.724308528
8	0.60382927	0.012580752	-0.012580752	0.60382927
9	0.160102398	0.003335725	0.003335725	-0.160102398

A.1.5 p=6

n	LoD	HiD	LoR	HiR
0	-0.001077301	-0.111540743	0.111540743	-0.001077301
1	0.004777258	0.49462389	0.49462389	-0.004777258
2	0.000553842	-0.751133908	0.751133908	0.000553842
3	-0.031582039	0.315250352	0.315250352	0.031582039
4	0.027522866	0.226264694	-0.226264694	0.027522866
5	0.097501606	-0.129766868	-0.129766868	-0.097501606
6	-0.129766868	-0.097501606	0.097501606	-0.129766868
7	-0.226264694	0.027522866	0.027522866	0.226264694
8	0.315250352	0.031582039	-0.031582039	0.315250352
9	0.751133908	0.000553842	0.000553842	-0.751133908
10	0.49462389	-0.004777258	0.004777258	0.49462389
11	0.111540743	-0.001077301	-0.001077301	-0.111540743

A.1.6 p=7

n	LoD	HiD	LoR	HiR
0	0.000353714	-0.077852054	0.077852054	0.000353714
1	-0.001801641	0.396539319	0.396539319	0.001801641
2	0.000429578	-0.729132091	0.729132091	0.000429578
3	0.012550999	0.469782287	0.469782287	-0.012550999
4	-0.016574542	0.143906004	-0.143906004	-0.016574542
5	-0.038029937	-0.224036185	-0.224036185	0.038029937
6	0.080612609	-0.071309219	0.071309219	0.080612609
7	0.071309219	0.080612609	0.080612609	-0.071309219
8	-0.224036185	0.038029937	-0.038029937	-0.224036185
9	-0.143906004	-0.016574542	-0.016574542	0.143906004
10	0.469782287	-0.012550999	0.012550999	0.469782287
11	0.729132091	0.000429578	0.000429578	-0.729132091
12	0.396539319	0.001801641	-0.001801641	0.396539319
13	0.077852054	0.000353714	0.000353714	-0.077852054

A.1.7 p=8

n	LoD	HiD	LoR	HiR
0	-0.000117477	-0.054415842	0.054415842	-0.000117477
1	0.000675449	0.312871591	0.312871591	-0.000675449
2	-0.00039174	-0.675630736	0.675630736	-0.00039174
3	-0.004870353	0.585354684	0.585354684	0.004870353
4	0.008746094	0.015829105	-0.015829105	0.008746094
5	0.013981028	-0.284015543	-0.284015543	-0.013981028
6	-0.044088254	-0.000472485	0.000472485	-0.044088254
7	-0.017369301	0.128747427	0.128747427	0.017369301
8	0.128747427	0.017369301	-0.017369301	0.128747427
9	0.000472485	-0.044088254	-0.044088254	-0.000472485
10	-0.284015543	-0.013981028	0.013981028	-0.284015543
11	-0.015829105	0.008746094	0.008746094	0.015829105
12	0.585354684	0.004870353	-0.004870353	0.585354684
13	0.675630736	-0.00039174	-0.00039174	-0.675630736
14	0.312871591	-0.000675449	0.000675449	0.312871591
15	0.054415842	-0.000117477	-0.000117477	-0.054415842

A.1.8 p=9

n	LoD	HiD	LoR	HiR
0	3.93E-05	-0.038077947	0.038077947	3.93E-05
1	-0.000251963	0.243834675	0.243834675	0.000251963
2	0.000230386	-0.604823124	0.604823124	0.000230386
3	0.001847647	0.657288078	0.657288078	-0.001847647
4	-0.004281504	-0.133197386	0.133197386	-0.004281504
5	-0.004723205	-0.293273783	-0.293273783	0.004723205
6	0.022361662	0.096840783	-0.096840783	0.022361662
7	0.000250947	0.148540749	0.148540749	-0.000250947
8	-0.067632829	-0.030725681	0.030725681	-0.067632829
9	0.030725681	-0.067632829	-0.067632829	-0.030725681
10	0.148540749	-0.000250947	0.000250947	0.148540749
11	-0.096840783	0.022361662	0.022361662	0.096840783
12	-0.293273783	0.004723205	-0.004723205	-0.293273783
13	0.133197386	-0.004281504	-0.004281504	-0.133197386
14	0.657288078	-0.001847647	0.001847647	0.657288078
15	0.604823124	0.000230386	0.000230386	-0.604823124
16	0.243834675	0.000251963	-0.000251963	0.243834675
17	0.038077947	3.93E-05	3.93E-05	-0.038077947

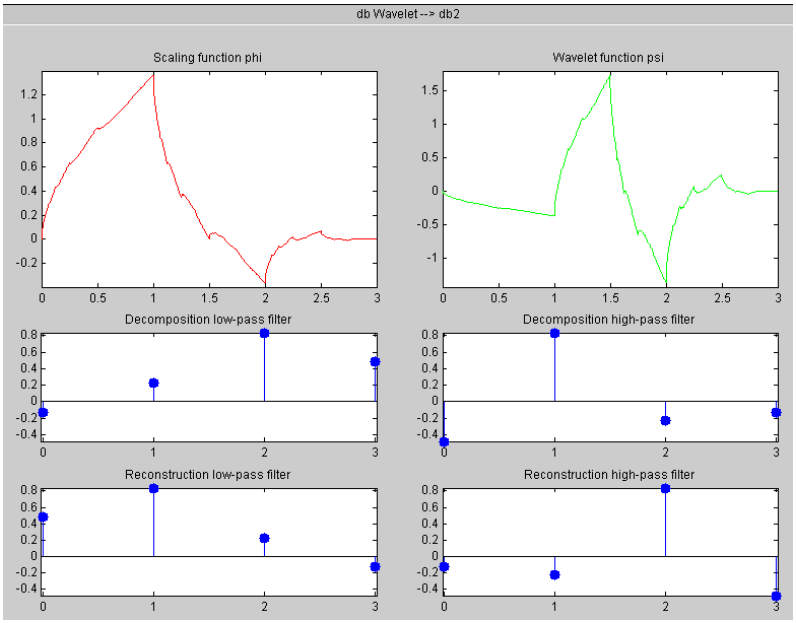
A.1.9 p=10

n	LoD	HiD	LoR	HiR
0	-1.33E-05	-0.026670058	0.026670058	-1.33E-05
1	9.36E-05	0.1881768	0.1881768	-9.36E-05
2	-0.000116467	-0.527201189	0.527201189	-0.000116467
3	-0.000685857	0.688459039	0.688459039	0.000685857
4	0.001992405	-0.281172344	0.281172344	0.001992405
5	0.001395352	-0.249846424	-0.249846424	-0.001395352
6	-0.010733175	0.195946274	-0.195946274	-0.010733175
7	0.003606554	0.12736934	0.12736934	-0.003606554
8	0.033212674	-0.093057365	0.093057365	0.033212674
9	-0.029457537	-0.071394147	-0.071394147	0.029457537
10	-0.071394147	0.029457537	-0.029457537	-0.071394147
11	0.093057365	0.033212674	0.033212674	-0.093057365
12	0.12736934	-0.003606554	0.003606554	0.12736934
13	-0.195946274	-0.010733175	-0.010733175	0.195946274
14	-0.249846424	-0.001395352	0.001395352	-0.249846424
15	0.281172344	0.001992405	0.001992405	-0.281172344
16	0.688459039	0.000685857	-0.000685857	0.688459039
17	0.527201189	-0.000116467	-0.000116467	-0.527201189
18	0.1881768	-9.36E-05	9.36E-05	0.1881768
19	0.026670058	-1.33E-05	-1.33E-05	-0.026670058

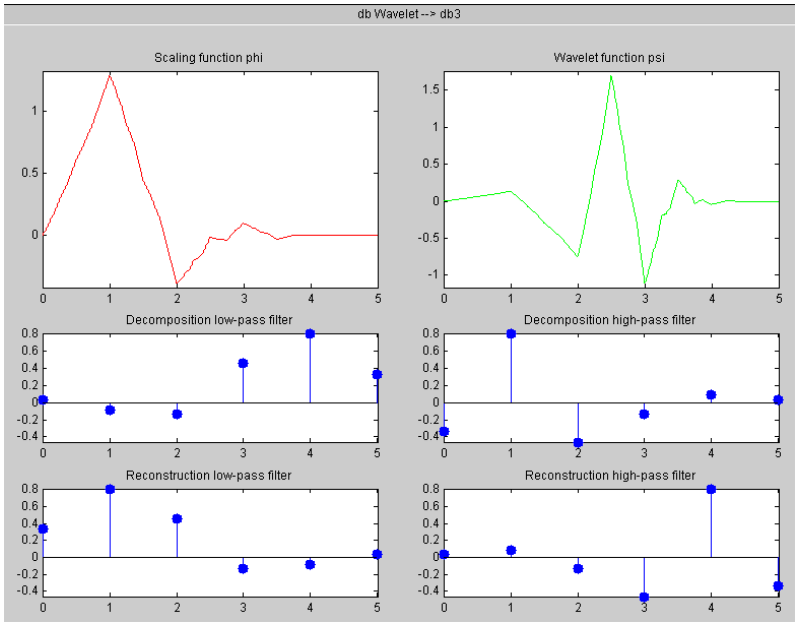
A.2 Function Plot

The function plots are made from MATLAB Wavelet Toolbox. The $\phi(t)$ and $\psi(t)$ are derived from (5.5).

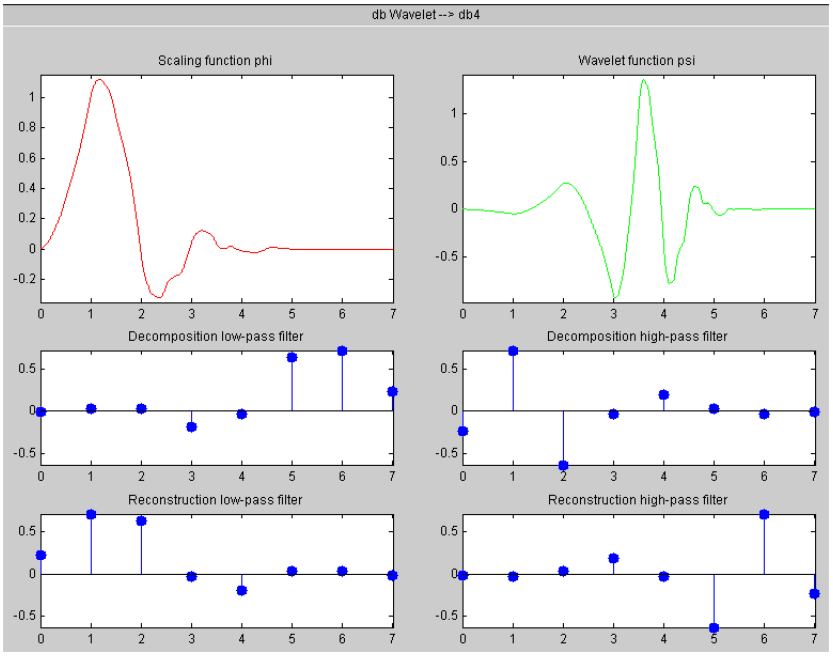
A.2.1 p=2


db Wavelet --> db2
Scaling function phi
Wavelet function psi
Decomposition low-pass filter
Decomposition high-pass filter
Reconstruction low-pass filter
Reconstruction high-pass filter

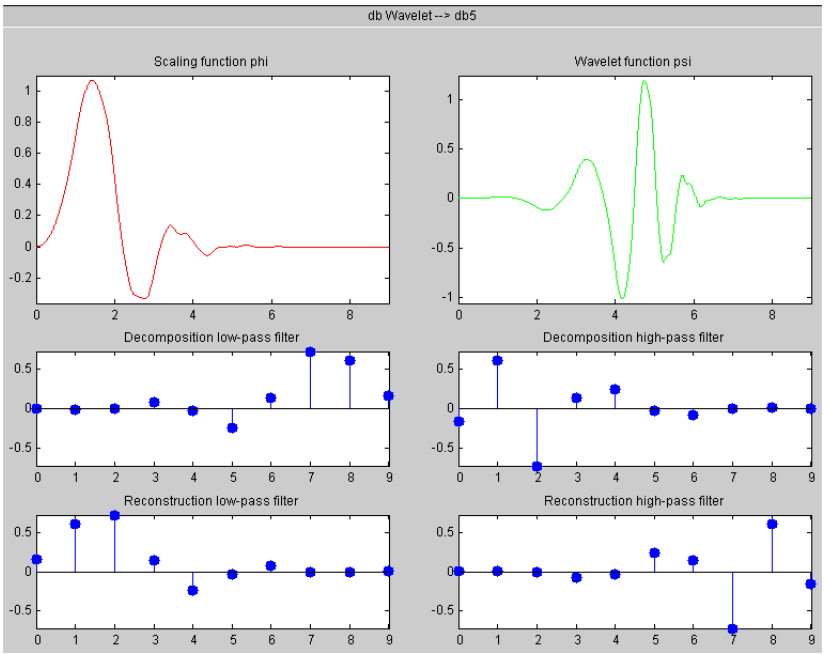
A.2.2 p=3


db Wavelet --> db3
Scaling function phi
Wavelet function psi
Decomposition low-pass filter
Decomposition high-pass filter
Reconstruction low-pass filter
Reconstruction high-pass filter

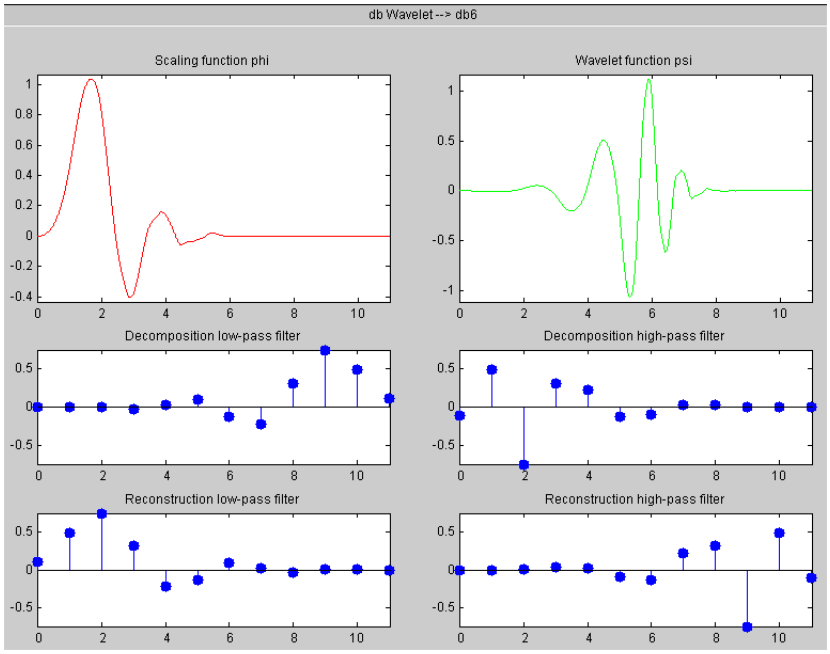
A.2.3 p=4


db Wavelet --> db4
Scaling function phi
Wavelet function psi
Decomposition low-pass filter
Decomposition high-pass filter
Reconstruction low-pass filter
Reconstruction high-pass filter

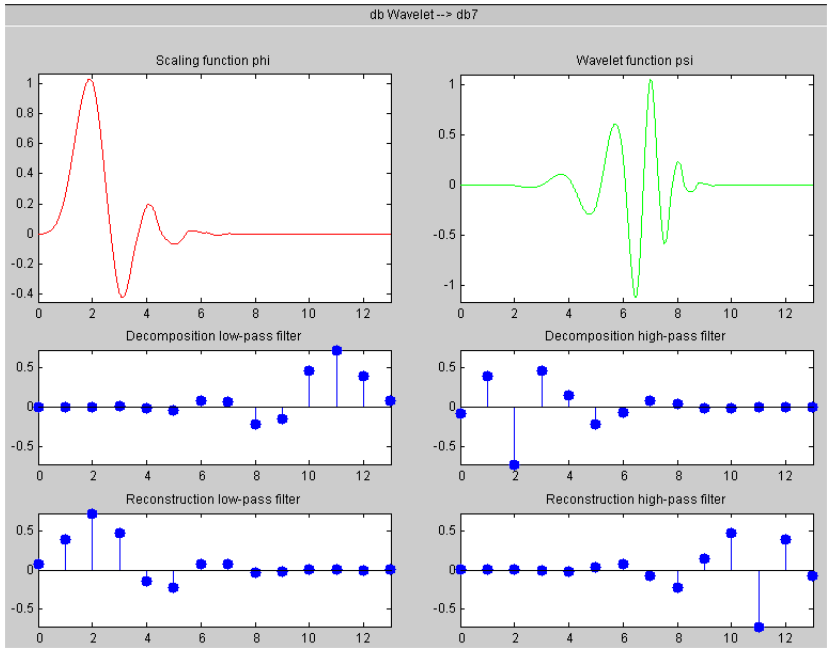
A.2.4 p=5


db Wavelet --> db5
Scaling function phi
Wavelet function psi
Decomposition low-pass filter
Decomposition high-pass filter
Reconstruction low-pass filter
Reconstruction high-pass filter

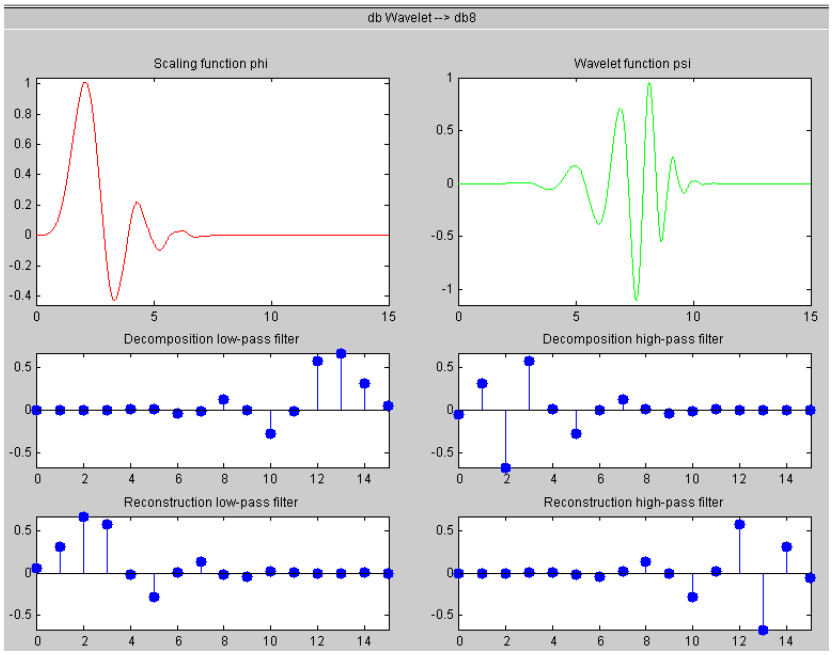
A.2.5 p=6


db Wavelet --> db6
Scaling function phi
Wavelet function psi
Decomposition low-pass filter
Decomposition high-pass filter
Reconstruction low-pass filter
Reconstruction high-pass filter

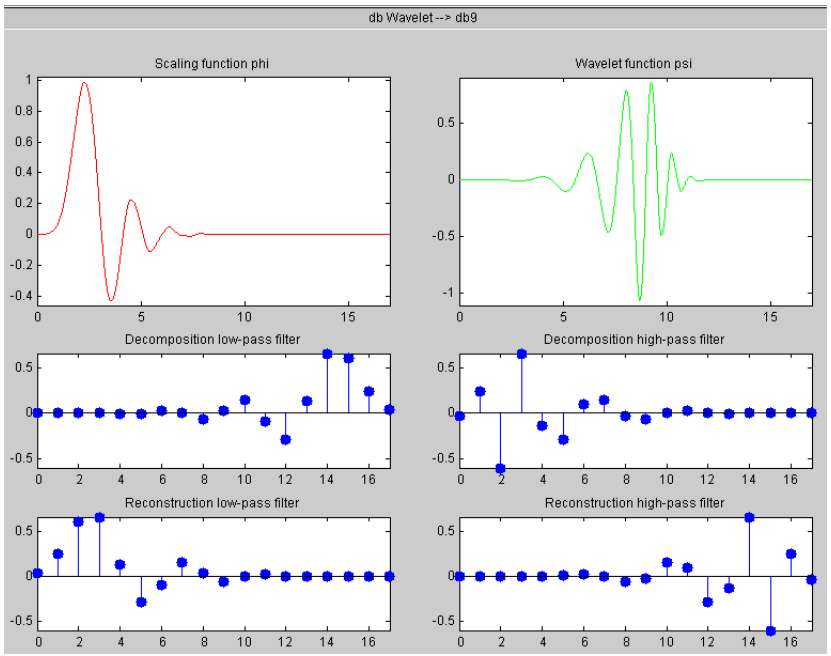
A.2.6 p=7


db Wavelet --> db7
Scaling function phi
Wavelet function psi
Decomposition low-pass filter
Decomposition high-pass filter
Reconstruction low-pass filter
Reconstruction high-pass filter

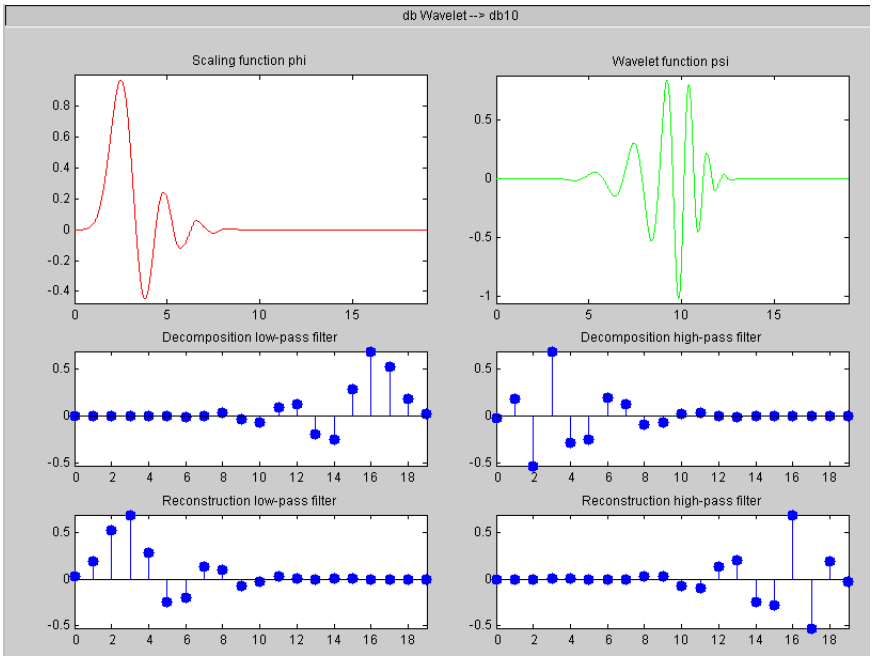
A.2.7 p=8


db Wavelet --> db8
Scaling function phi
Wavelet function psi
Decomposition low-pass filter
Decomposition high-pass filter
Reconstruction low-pass filter
Reconstruction high-pass filter

A.2.8 p=9


db Wavelet --> db9
Scaling function phi
Wavelet function psi
Decomposition low-pass filter
Decomposition high-pass filter
Reconstruction low-pass filter
Reconstruction high-pass filter

A.2.9 p=10


db Wavelet --> db10
Scaling function phi
Wavelet function psi
Decomposition low-pass filter
Decomposition high-pass filter
Reconstruction low-pass filter
Reconstruction high-pass filter

Appendix B

Symlets

B.1 Coefficients

B.1.1 p=2

n	LoD	HiD	LoR	HiR
0	-0.129409523	-0.482962913	0.482962913	-0.129409523
1	0.224143868	0.836516304	0.836516304	-0.224143868
2	0.836516304	-0.224143868	0.224143868	0.836516304
3	0.482962913	-0.129409523	-0.129409523	-0.482962913

B.1.2 p=3

n	LoD	HiD	LoR	HiR
0	0.035226292	-0.332670553	0.332670553	0.035226292
1	-0.085441274	0.806891509	0.806891509	0.085441274
2	-0.13501102	-0.459877502	0.459877502	-0.13501102
3	0.459877502	-0.13501102	-0.13501102	-0.459877502
4	0.806891509	0.085441274	-0.085441274	0.806891509
5	0.332670553	0.035226292	0.035226292	-0.332670553

B.1.3 p=4

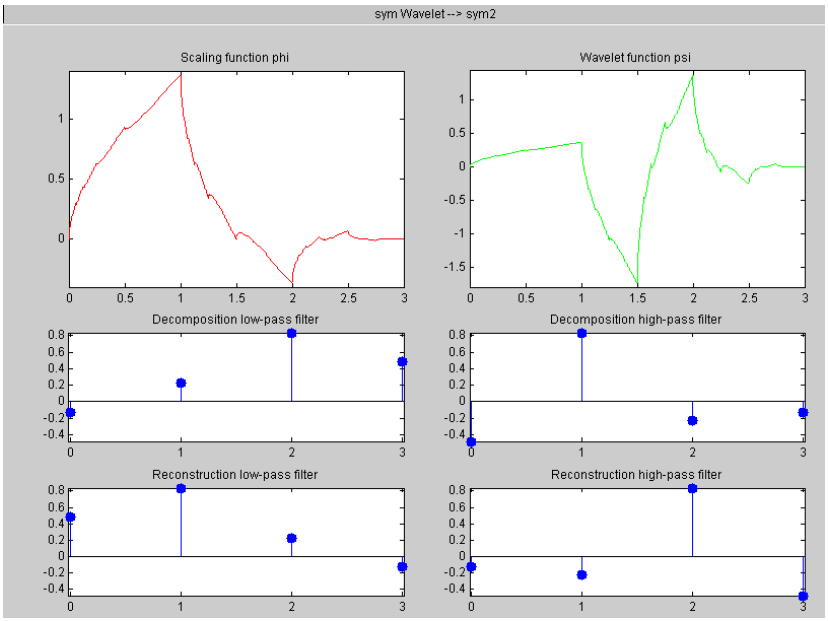
n	LoD	HiD	LoR	HiR
0	-0.075765715	-0.032223101	0.032223101	-0.075765715
1	-0.029635528	-0.012603967	-0.012603967	0.029635528
2	0.497618668	0.099219544	-0.099219544	0.497618668
3	0.803738752	0.297857796	0.297857796	-0.803738752
4	0.297857796	-0.803738752	0.803738752	0.297857796
5	-0.099219544	0.497618668	0.497618668	0.099219544
6	-0.012603967	0.029635528	-0.029635528	-0.012603967
7	0.032223101	-0.075765715	-0.075765715	-0.032223101

B.1.4 p=5

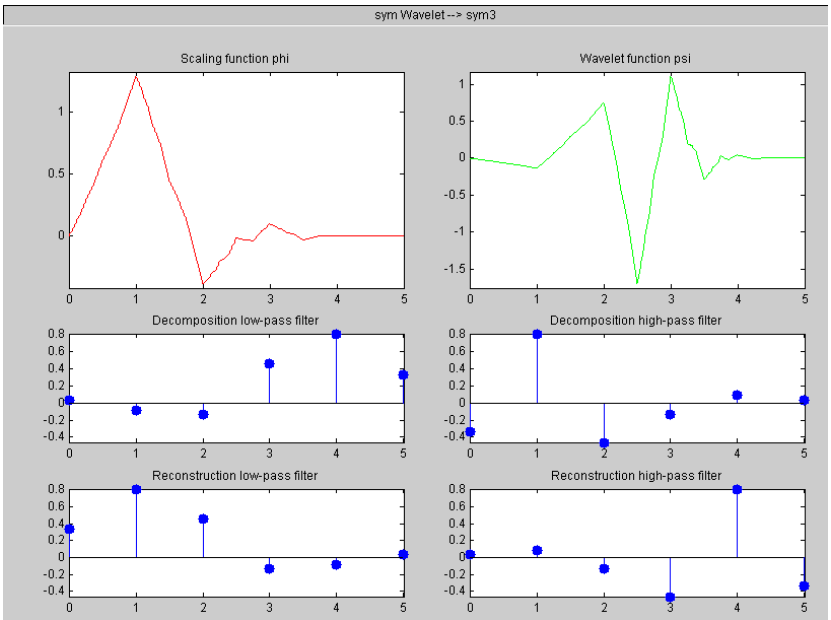
n	LoD	HiD	LoR	HiR
0	0.027333068	-0.019538883	0.019538883	0.027333068
1	0.029519491	-0.021101834	-0.021101834	-0.029519491
2	-0.039134249	0.17532809	-0.17532809	-0.039134249
3	0.199397534	0.016602106	0.016602106	-0.199397534
4	0.72340769	-0.633978963	0.633978963	0.72340769
5	0.633978963	0.72340769	0.72340769	-0.633978963
6	0.016602106	-0.199397534	0.199397534	0.016602106
7	-0.17532809	-0.039134249	-0.039134249	0.17532809
8	-0.021101834	-0.029519491	0.029519491	-0.021101834
9	0.019538883	0.027333068	0.027333068	-0.019538883

B.2 Function Plot

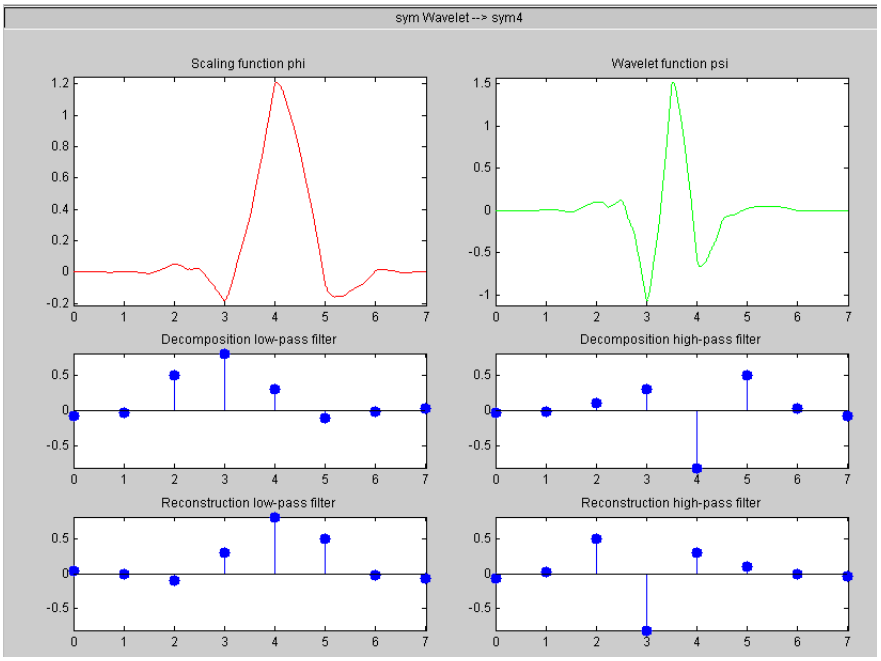
B.2.1 p=2



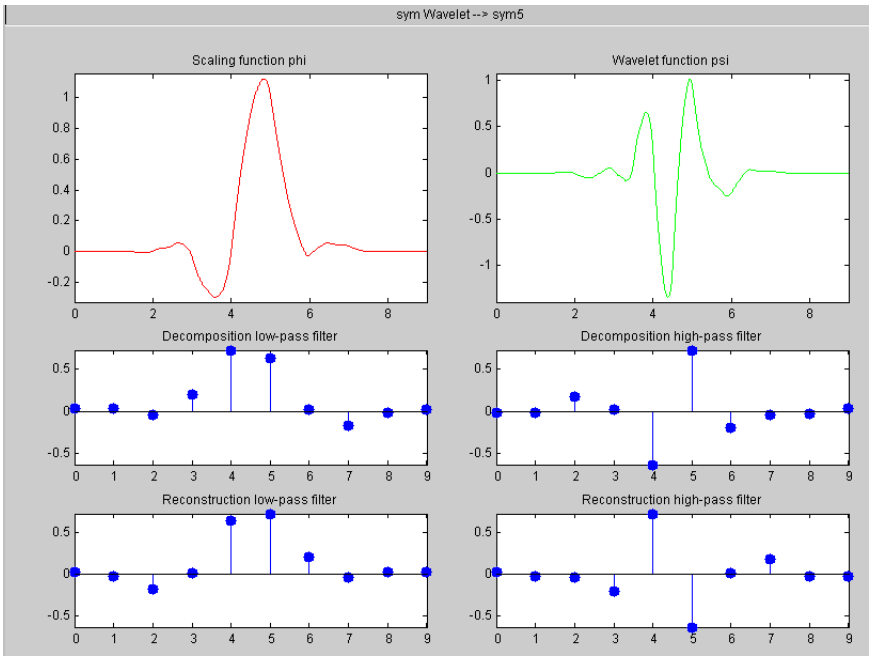
B.2.2 p=3



B.2.3 p=4


sym Wavelet --> sym4
Scaling function phi
Wavelet function psi
Decomposition low-pass filter
Decomposition high-pass filter
Reconstruction low-pass filter
Reconstruction high-pass filter

B.2.4 p=5


sym Wavelet --> sym5
Scaling function phi
Wavelet function psi
Decomposition low-pass filter
Decomposition high-pass filter
Reconstruction low-pass filter
Reconstruction high-pass filter

Appendix C

Coiflets

C.1 Coefficients

C.1.1 p=1

n	LoD	HiD	LoR	HiR
0	-0.015655728	0.07273262	-0.07273262	-0.015655728
1	-0.07273262	0.337897662	0.337897662	0.07273262
2	0.384864847	-0.85257202	0.85257202	0.384864847
3	0.85257202	0.384864847	0.384864847	-0.85257202
4	0.337897662	0.07273262	-0.07273262	0.337897662
5	-0.07273262	-0.015655728	-0.015655728	0.07273262

C.1.2 p=2

n	LoD	HiD	LoR	HiR
0	-0.000720549	-0.016387336	0.016387336	-0.000720549
1	-0.001823209	-0.041464937	-0.041464937	0.001823209
2	0.005611435	0.067372555	-0.067372555	0.005611435
3	0.023680172	0.386110067	0.386110067	-0.023680172
4	-0.059434419	-0.812723635	0.812723635	-0.059434419
5	-0.076488599	0.417005184	0.417005184	0.076488599
6	0.417005184	0.076488599	-0.076488599	0.417005184
7	0.812723635	-0.059434419	-0.059434419	-0.812723635
8	0.386110067	-0.023680172	0.023680172	0.386110067
9	-0.067372555	0.005611435	0.005611435	0.067372555
10	-0.041464937	0.001823209	-0.001823209	-0.041464937
11	0.016387336	-0.000720549	-0.000720549	-0.016387336

C.1.3 p=3

n	LoD	HiD	LoR	HiR
0	-3.46E-05	0.003793513	-0.003793513	-3.46E-05
1	-7.10E-05	0.007782596	0.007782596	7.10E-05
2	0.000466217	-0.023452696	0.023452696	0.000466217
3	0.001117519	-0.065771911	-0.065771911	-0.001117519
4	-0.002574518	0.06112339	-0.06112339	-0.002574518
5	-0.009007976	0.405176902	0.405176902	0.009007976
6	0.015880545	-0.793777223	0.793777223	0.015880545
7	0.034555028	0.428483476	0.428483476	-0.034555028
8	-0.082301927	0.071799822	-0.071799822	-0.082301927
9	-0.071799822	-0.082301927	-0.082301927	0.071799822
10	0.428483476	-0.034555028	0.034555028	0.428483476
11	0.793777223	0.015880545	0.015880545	-0.793777223
12	0.405176902	0.009007976	-0.009007976	0.405176902
13	-0.06112339	-0.002574518	-0.002574518	0.06112339
14	-0.065771911	-0.001117519	0.001117519	-0.065771911
15	0.023452696	0.000466217	0.000466217	-0.023452696
16	0.007782596	7.10E-05	-7.10E-05	0.007782596
17	-0.003793513	-3.46E-05	-3.46E-05	0.003793513

C.1.4 p=4

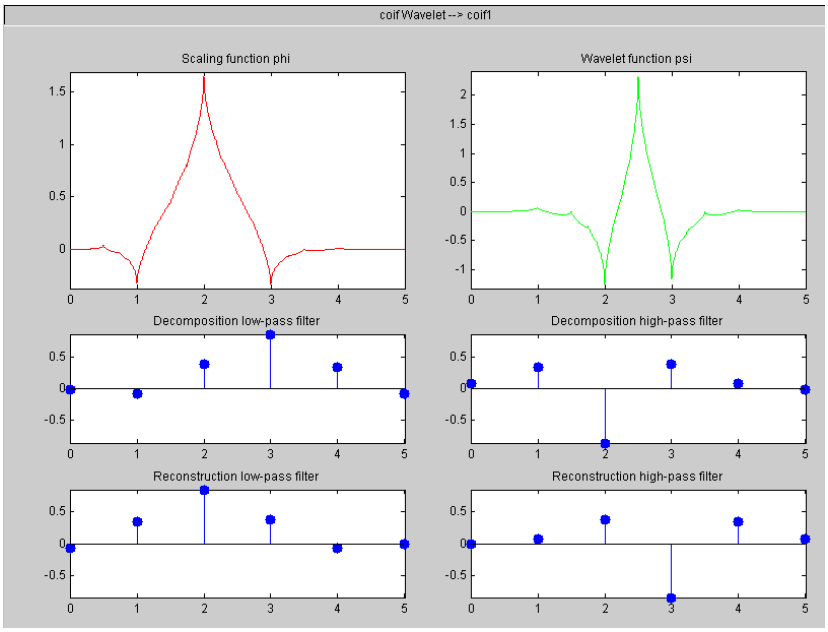
n	LoD	HiD	LoR	HiR
0	-1.78E-06	-0.000892314	0.000892314	-1.78E-06
1	-3.26E-06	-0.001629492	-0.001629492	3.26E-06
2	3.12E-05	0.007346166	-0.007346166	3.12E-05
3	6.23E-05	0.016068944	0.016068944	-6.23E-05
4	-0.000259975	-0.0266823	0.0266823	-0.000259975
5	-0.000589021	-0.0812667	-0.0812667	0.000589021
6	0.001266562	0.056077313	-0.056077313	0.001266562
7	0.003751436	0.415308407	0.415308407	-0.003751436
8	-0.005658287	-0.782238931	0.782238931	-0.005658287
9	-0.015211732	0.434386056	0.434386056	0.015211732
10	0.025082262	0.066627474	-0.066627474	0.025082262
11	0.039334427	-0.096220442	-0.096220442	-0.039334427
12	-0.096220442	-0.039334427	0.039334427	-0.096220442
13	-0.066627474	0.025082262	0.025082262	0.066627474
14	0.434386056	0.015211732	-0.015211732	0.434386056
15	0.782238931	-0.005658287	-0.005658287	-0.782238931
16	0.415308407	-3.75E-03	3.75E-03	0.415308407
17	-0.056077313	1.27E-03	1.27E-03	0.056077313
18	-0.0812667	0.000589021	-0.000589021	-0.0812667
19	0.0266823	-0.000259975	-0.000259975	-0.0266823
20	0.016068944	-6.23E-05	6.23E-05	0.016068944
21	-0.007346166	3.12E-05	3.12E-05	0.007346166
22	-0.001629492	3.26E-06	-3.26E-06	-0.001629492
23	0.000892314	-1.78E-06	-1.78E-06	-0.000892314

C.1.5 p=5

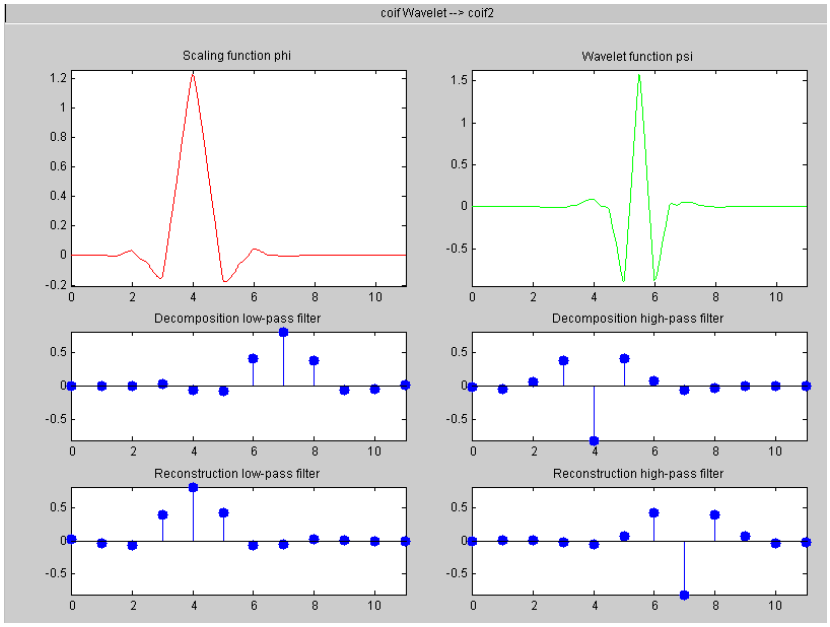
n	LoD	HiD	LoR	HiR
0	-9.52E-08	0.000212081	-0.000212081	-9.52E-08
1	-1.67E-07	0.00035859	0.00035859	1.67E-07
2	2.06E-06	-0.002178236	0.002178236	2.06E-06
3	3.73E-06	-0.004159359	-0.004159359	-3.73E-06
4	-2.13E-05	0.010131118	-0.010131118	-2.13E-05
5	-4.13E-05	0.023408157	0.023408157	4.13E-05
6	0.000140541	-0.028168029	0.028168029	0.000140541
7	0.00030226	-0.091920011	-0.091920011	-0.00030226
8	-0.000638131	0.052043163	-0.052043163	-0.000638131
9	-0.001662864	0.421566207	0.421566207	0.001662864
10	0.002433373	-0.774289604	0.774289604	0.002433373
11	0.006764185	0.437991626	0.437991626	-0.006764185
12	-0.009164231	0.062035964	-0.062035964	-0.009164231
13	-0.019761779	-0.105574209	-0.105574209	0.019761779
14	0.032683574	-0.041289209	0.041289209	0.032683574
15	0.041289209	0.032683574	0.032683574	-0.041289209
16	-0.105574209	1.98E-02	-1.98E-02	-0.105574209
17	-0.062035964	-9.16E-03	-9.16E-03	0.062035964
18	0.437991626	-0.006764185	0.006764185	0.437991626
19	0.774289604	0.002433373	0.002433373	-0.774289604
20	0.421566207	1.66E-03	-1.66E-03	0.421566207
21	-0.052043163	-6.38E-04	-6.38E-04	0.052043163
22	-0.091920011	-3.02E-04	3.02E-04	-0.091920011
23	0.028168029	1.41E-04	1.41E-04	-0.028168029
24	0.023408157	4.13E-05	-4.13E-05	0.023408157
25	-0.010131118	-2.13E-05	-2.13E-05	0.010131118
26	-0.004159359	-3.73E-06	3.73E-06	-0.004159359
27	0.002178236	2.06E-06	2.06E-06	-0.002178236
28	0.00035859	1.67E-07	-1.67E-07	0.00035859
29	-0.000212081	-9.52E-08	-9.52E-08	0.000212081

C.2 Function Plot

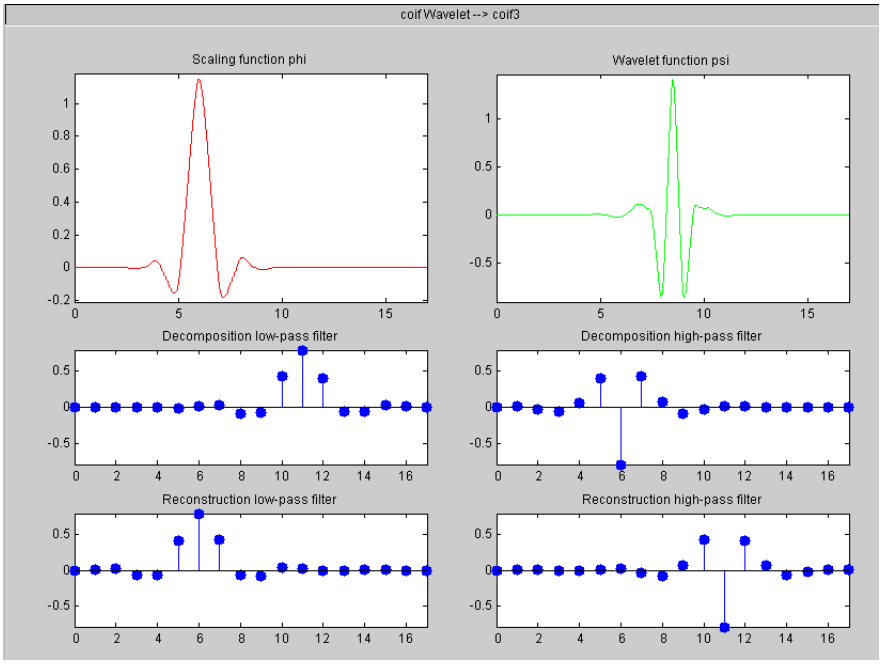
C.2.1 p=1



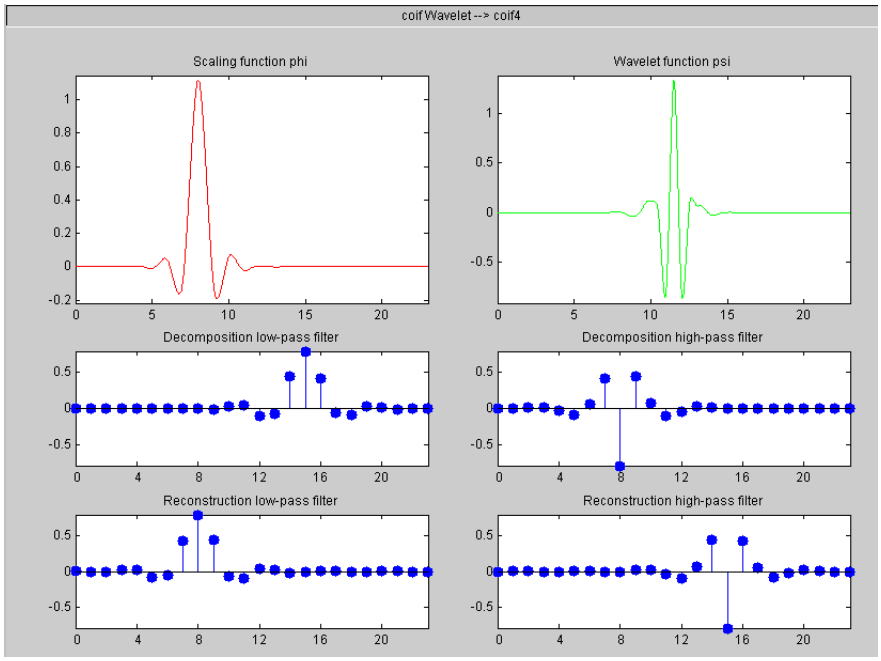
C.2.2 p=2



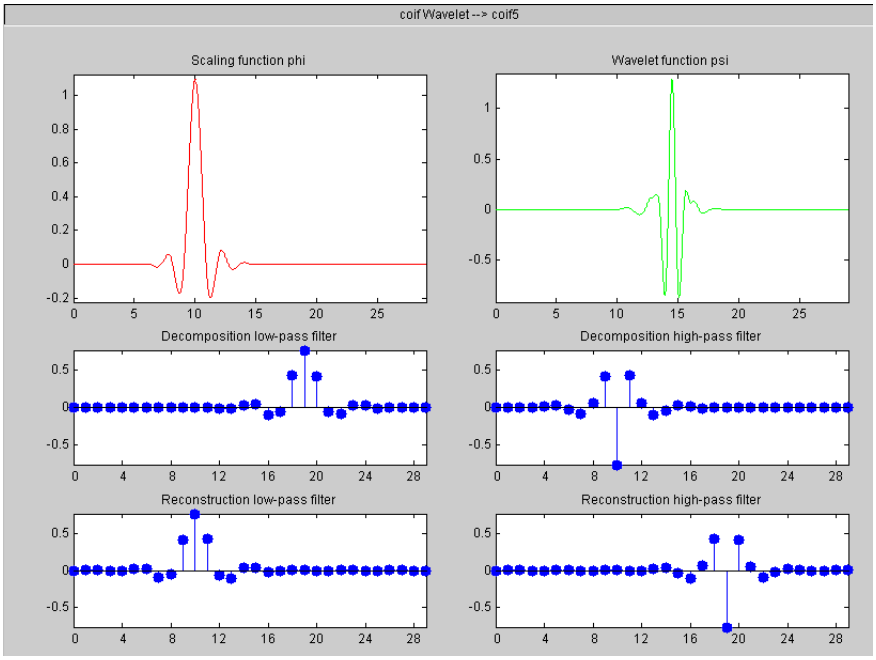
C.2.3 p=3


coif Wavelet --> coif3
Scaling function phi
Wavelet function psi
Decomposition low-pass filter
Decomposition high-pass filter
Reconstruction low-pass filter
Reconstruction high-pass filter

C.2.4 p=4


coif Wavelet --> coif4
Scaling function phi
Wavelet function psi
Decomposition low-pass filter
Decomposition high-pass filter
Reconstruction low-pass filter
Reconstruction high-pass filter

C.2.5 p=5


coif Wavelet --> coif5
Scaling function phi
Wavelet function psi
Decomposition low-pass filter
Decomposition high-pass filter
Reconstruction low-pass filter
Reconstruction high-pass filter

References

[1] R. C. Gonzalez and R. E. Woods, *Digital Image Processing (2nd Edition)*. Prentice Hall, January 2002.

[2] S. G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," vol. 11, pp. 674–693, July 1989.

[3] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," vol. 41, pp. 3445–3462, Dec. 1993.

[4] A. Said and W. A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," vol. 6, pp. 243–250, June 1996.

[5] D. Taubman, "High performance scalable image compression with ebcot," in *Proc. International Conference on Image Processing ICIP 99*, vol. 3, pp. 344–348, Oct. 24–28, 1999.

[6] Y.-S. Zhang, "Multiresolution analysis for image by generalized 2-d wavelets," Master's thesis, 2008.

[7] I. Daubechies, *Ten lectures on wavelets*, vol. 61 of *CBMS-NSF Regional Conference Series in Applied Mathematics*. Philadelphia, PA: Society for Industrial and Applied Mathematics (SIAM), 1992.

[8] S. Mallat, *A Wavelet Tour of Signal Processing, 3rd ed., Third Edition: The Sparse Way*. Academic Press, 3 ed., December 2008.

[9] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Communications on Pure and Applied Mathematics*, vol. 41, no. 7, pp. 909–996, 1988.

[10] P.-S. T. Tinku Acharya, *JPEG2000 Standard for Image Compression*, ch. Coding Algorithms in JPEG2000, pp. 163–196. 2005.

[11] N.-C. Shen, "Sectioned convolution for discrete wavelet transform," Master's thesis, 2008.

[12] M. Rabbani and R. Joshi, "An overview of the jpeg 2000 still image compression standard," pp. 3–48, Elsevier, January 2002.

[13] C. Christopoulos, A. Skodras, and T. Ebrahimi, "The jpeg2000 still image coding system: an overview," vol. 46, pp. 1103–1127, Nov. 2000.

[14] P. M. A. K. Louis and A. Rieder, *Wavelets Theory and Applications*. John Wiley & Sons, 1997.

[15] Z.-N. Li and M. S. Drew, *Fundamentals of Multimedia*. Prentice Hall, October 2003.