CHAPTER 6

Digital Filter Banks
and the Wavelet Transform

In Chapter 5 we investigated the relationship between the wavelet transform and digital filter banks. It turns out that the wavelet transform can be simply achieved by a tree of digital filter banks, with no need of computing mother wavelets. Hence, the filter banks have been playing a central role in the area of wavelet analysis. In this chapter we will introduce the basics of filter banks.

The theory of filter banks was developed a long time ago, before modern wavelet analysis became popular. The reader can find many excellent textbooks in this area, such as Crochiere and Rabiner [12], Strang and Nguyen [47], and Vaidyanathan [48]. What will be introduced in this chapter are the fundamental concepts and designs of some of the most popular filter banks. They cover the majority of commonly used biorthogonal as well as orthogonal wavelets, except for the coiflets.

The Daubechies wavelets achieve the maximum number of wavelet zero moments, whereas the coiflets are a combination of wavelet zero moments and scaling function zero moments. The digital filters used to compute the coiflet wavelet transform cannot be generated by the structures discussed in this chapter. The design of digital filters for the coiflet wavelet transform needs some special skills and a certain level of mathematical preparation, which are beyond the scope of this book. The reader can find related information in the literature: [6], [13], [270], [247], [288], and [400].

Undoubtedly, experiments with new types of filter banks are proceeding, even as this book is being read. Nevertheless, this chapter provides an adequate tutorial for applied engineers and

1. Although we have only shown the proof for the orthogonal wavelet transform, this conclusion, in fact, is also true for the biorthogonal case.
scientists who want to use wavelet analysis. It also can serve as an introduction for students who are new to the topic. While this chapter assumes some knowledge of the $z$-transform and digital filters, such knowledge is not essential. The basic principles of filter banks presented here are well developed and easily understood, so that even those with limited experience in wavelet analysis can quickly master this technique.

### 6.1 Two-Channel Perfect Reconstruction Filter Banks

In Section 5.3 we discussed the relationship between the scaling function, the mother wavelet, and filter banks. Once the lowpass and highpass filters have been determined, we can compute the scaling function and the mother wavelet through the refinement equations (5.40) and (5.65), respectively. Moreover, Section 5.4 further proves that under certain conditions the outputs of the highpass filters are good approximations of the wavelet series. Consequently, the selection of desired scaling functions and mother wavelets reduces to the design of lowpass and highpass filters of two-channel perfect reconstruction (PR) filter banks. The wavelet transform can simply be realized by a tree of two-channel PR filter banks. In this section we will briefly introduce the two-channel PR filter banks.

![Two-channel filter bank](image)

**Figure 6-1** Two-channel filter bank. $H_0(z)$ and $H_1(z)$ form an analysis filter bank, whereas $G_0(z)$ and $G_1(z)$ form a synthesis filter bank. Note that $H(z)$ and $G(z)$ can be interchanged.

Figure 6-1 sketches a typical two-channel filter bank system, where the $z$-transform is defined as

$$H(z) = \sum_{n=0}^{N} h[n]z^{-n} = H(e^{j\omega}) \bigg|_{z = e^{j\omega}} = H(\omega)$$

(6.1)

Clearly, $\omega = 0$ is equivalent to $z = 1$, and $\omega = \pi$ is equivalent to $z = -1$. Hence, $H(0)$ and $H(\pi)$ in the frequency domain correspond to $H(1)$ and $H(-1)$ in the $z$-domain. In this book, we use $N$ for

---

2. While Figures 5-16 and 5-17 depict the orthogonal filter banks, the system in Figure 6-1 represents more general cases. As we shall see shortly, the conditions for the orthogonal filter banks obtained in Section 5.3, such as the paraunitary condition and the power complementarity condition, are special cases of the conclusions derived from the system in Figure 6-1.
the filter order. The length of the filter in Eq. (6.1) is equal to $N + 1$.

As shown in Figure 6-1, the signal $X(z)$ is first filtered by a filter bank consisting of $H_0(z)$ and $H_1(z)$. The outputs of $H_0(z)$ and $H_1(z)$ are downsampled by 2 to obtain $Y_0(z)$ and $Y_1(z)$. After some processing, the modified signals are upsampled and filtered by another filter bank consisting of $G_0(z)$ and $G_1(z)$. The downsampling operators are decimators and the upsampling operators are expanders. If no processing takes place between the two filter banks (in other words, $Y_0(z)$ and $Y_1(z)$ are not altered), the sum of the outputs of $G_0(z)$ and $G_1(z)$ is identical to the original signal $X(z)$, except for a time delay. Such a system is commonly referred to as a two-channel perfect reconstruction filter bank. $H_0(z)$ and $H_1(z)$ form an analysis filter bank, whereas $G_0(z)$ and $G_1(z)$ form a synthesis filter bank. Note that $H(z)$ and $G(z)$ can be interchanged. For instance, we can use $G_0(z)$ and $G_1(z)$ for analysis and $H_0(z)$ and $H_1(z)$ for synthesis. In this book, $H_0(z)$ and $G_0(z)$ denote lowpass filters, while $H_1(z)$ and $G_1(z)$ are highpass filters. The subscripts 0 and 1 represent lowpass and highpass filters, respectively.

Note that for a discrete-time function $x[n]$, the $z$-transform of its interpolated version is $X(z^M)$. This is because if

$$y[n] = \begin{cases} \left\lceil \frac{n}{M} \right\rceil & n = 0, \pm M, \pm 2M \ldots \\ \text{zero} & \text{otherwise} \end{cases}$$ (6.2)

then

$$Y[z] = \sum_n y[n]z^{-n} = \sum_{n = 0, \pm M, \pm 2M \ldots} x \left[ \frac{n}{M} \right] z^{-n} = \sum_k x[k] (z^M)^{-k} = X(z^M)$$ (6.3)

Similarly, the reader can verify that for a discrete-time function $x[n]$, the $z$-transform of its doubly decimated version is

$$Y[z] = \frac{1}{2} [X(z^{1/2}) + X(-z^{1/2})]$$ (6.4)

By applying the relations (6.3) and (6.4), we can derive the output of the lower channel in Figure 6-1 as

$$\hat{Y}_0(z) = \frac{1}{2} G_0(z)[H_0(z)X(z) + H_0(-z)X(-z)]$$ (6.5)

and the output of the upper channel as

$$\hat{Y}_1(z) = \frac{1}{2} G_1(z)[H_1(z)X(z) + H_1(-z)X(-z)]$$ (6.6)

Hence, the output $\hat{X}(z)$ will be

$$\hat{X}(z) = \frac{1}{2} [G_0(z)H_0(z) + G_1(z)H_1(z)]X(z) + \frac{1}{2} [G_0(z)H_0(-z) + G_1(z)H_1(-z)]X(-z)$$ (6.7)

where one term involves $X(z)$ and the other involves $X(-z)$. For perfect reconstruction, the term with $X(-z)$, traditionally called the alias term, must be zero. To achieve this, we need
which reminds us of Eq. (5.62). As a matter of fact, condition (5.62) can be considered as a special case of condition (6.8). While condition (6.8) leads to biorthogonal filter banks, (5.62) is the necessary condition for orthogonal filter banks.

To accomplish Eq. (6.8), we can let

\[ G_0(z) = H_1(-z) \quad \text{and} \quad G_1(z) = -H_0(-z) \] (6.9)

which implies that \( \gamma_0[n] \) can be obtained by alternating the sign of \( \gamma_1[n] \):

\[ \gamma_0[n] = (-1)^n h_1[n] \] (6.10)

Similarly,

\[ \gamma_1[n] = (-1)^n+1 h_0[n] \] (6.11)

Therefore, \( \gamma_1[n] \) and \( h_1[n] \) are the highpass filters if \( \gamma_0[n] \) and \( h_0[n] \) are the lowpass filters. Once \( H_0(z) \) and \( H_1(z) \) [or \( G_0(z) \) and \( G_1(z) \)] are determined, we can find the remaining filters with Eq. (6.9).

For perfect reconstruction, we also need the first term in Eq. (6.7), called the distortion term, to be a constant or a pure time delay. For example,

\[ H_0(z)G_0(z) + H_1(z)G_1(z) = 2z^{-l} \] (6.12)

where \( l \) denotes a time delay.\(^3\) If we satisfy both (6.8) and (6.12), the output of the two-channel filter bank in Figure 6-1 is a delayed version of the input signal

\[ \hat{X}(z) = z^{-l}X(z) \] (6.13)

Let’s rewrite (6.9) as

\[ H_1(z) = G_0(-z) \quad \text{and} \quad G_1(z) = -H_0(-z) \] (6.14)

Substituting Eq. (6.14) into Eq. (6.12) yields

\[ H_0(z)G_0(z) - H_0(-z)G_0(-z) = P_0(z) - P_0(-z) = 2z^{-l} \] (6.15)

where \( P_0(z) \) denotes the product of two lowpass filters, \( H_0(z) \) and \( G_0(z) \)

\[ P_0(z) = H_0(z)G_0(z) \] (6.16)

Eq. (6.15) indicates that all odd terms of the product of two lowpass filters, \( H_0(z) \) and \( G_0(z) \), must be zero, except for order \( l \). But the even order terms are arbitrary. The delay parameter \( l \) must be odd, which is usually the center of the filter \( P_0(z) \). We can summarize these observations by the following formula:

\(^3\) The power complementarity condition (5.63) can be thought of a special case of (6.13). Since the biorthogonal transform generally does not satisfy the power complementarity condition, it usually is not a unitary transform.
Consequently, the design of two-channel PR filter banks reduces to two steps:

1. Design a filter $P_0(z)$ that satisfies Eq. (6.17).
2. Factorize $P_0(z)$ into $H_0(z)$ and $G_0(z)$. Then, use Eq. (6.14) to compute $H_1(z)$ and $G_1(z)$.

Obviously, there are many ways to design $P_0(z)$. And there are many ways to factor it. The choice of four filters given by Eq. (6.9) is also not unique. Experiments are going on even as this book is being written, and undoubtedly they will still be on even as the book is being read. As an introduction to the wavelet transform, however, we will limit our presentation to those cases that are most popular in practical applications. The reader can find a more comprehensive treatment of filter banks in Strang and Nguyen [47] and Vaidyanathan [48].

One of the most well-known selections of the product filter $P_0(z)$ is defined by

$$P_0(z) = (1 + z^{-1})^{2k} Q(z) = (1 + z^{-1})^{2k} \sum_{m=0}^{2k-2} a_m z^{-m}$$  \hspace{1cm} (6.18)

The polynomial $Q(z)$ of degree $2k - 2$ is chosen so that Eq. (6.15) is satisfied. In this case, the order of $P_0(z)$ in (6.18) is always an even number. Moreover, it is a type I filter, i.e.,

$$p_0[n] = p_0[N-n] \quad N \text{ is even}$$  \hspace{1cm} (6.19)

which implies that the number of coefficients $p_0[n]$ is odd, $N + 1$. Based on Eq. (6.17), there are $2k - 1$ odd powers in $P_0(z)$, and $2k - 1$ coefficients to choose in $Q(z)$. Thus, $Q(z)$ is unique.

The special factor $(1 + z^{-1})^{2k}$ is also known as the binomial filter. The binomial itself, without $Q(z)$, represents a spline filter. $Q(z)$ is needed to give perfect reconstruction. It can be proved that $2k$ is the maximum number of zeros that $P_0(z)$ can have while preserving perfect reconstruction. Hence, $P_0(z)$ defined by Eq. (6.18) is traditionally named as the maxflat filter.

**Example 6.1** A maxflat filter with $k = 2$ is

$$P_0(z) = (1 + z^{-1})^4 Q(z)$$  \hspace{1cm} (6.20)

Since the order of $Q(z)$ is $2k - 2 = 2$, based on condition (6.17) we can compute

$$P_0(z) = \frac{1}{16} (-1 + 9z^{-2} + 16z^{-3} + 9z^{-4} - z^{-6})$$  \hspace{1cm} (6.21)

and

$$Q(z) = -1 + 4z^{-1} - z^{-2}$$  \hspace{1cm} (6.22)

Hence, the two roots from $Q(z)$ are at

\[\begin{align*}
\rho_n &= 0 & \text{ odd and } n \neq l \\
\rho_n &= 1 & \text{ if } n = l \\
\rho_n &= \text{ arbitrary} & \text{ if } n \text{ even}
\end{align*}\]  \hspace{1cm} (6.17)
While \(|c| < 1\) is inside the circle, its reciprocal \(|1/c| > 1\) is outside the unit circle. There are several possibilities for factoring \(P_0(z)\) in (6.21). For example,

a. \(H_0(z) = (1 + z^{-1})^2\) and \(G_0(z) = (1 + z^{-1})(c - z^{-1})(1/c - z^{-1})\). In this case, \(H_0(z)\) has two zeros at \(z = -1\) (that is, \(\omega = \pi\), which is a typical quadratic spline, as shown in Figure 6-2.

\[
\begin{align*}
\text{Figure 6-2} & \quad \text{Quadratic spline wavelets for case (a)} \\
H(z) & \quad G(z)
\end{align*}
\]

b. \(H_0(z) = (1 + z^{-1})^3\) and \(G_0(z) = (1 + z^{-1})(c - z^{-1})(1/c - z^{-1})\). In this case, \(H_0(z)\) has three zeros at \(z = -1\) (that is, \(\omega = \pi\), which is a typical cubic spline. As shown in Figure 6-3, in this case, the analysis mother wavelet is much smoother than that corresponding to the quadratic spline in Figure 6-2. But the synthesis mother wavelet is less desirable. In both cases (a) and (b), all filters are linear phase. The resulting wavelets are both biorthogonal.

\[
\begin{align*}
\text{Figure 6-3} & \quad \text{Cubic spline wavelets for case (b). Note that the cubic spline analysis mother wavelet is much smoother than that corresponding to the quadratic spline in Figure 6-2. But the corresponding synthesis mother wavelet is less desirable.}
\end{align*}
\]
c. \( H_0(z) = (1 + z^{-1})^2(c - z^{-1}) \) and \( G_0(z) = (1 + z^{-1})^2(1/c - z^{-1}) \). This is the second order Daubechies mother wavelet, as shown in Figure 6-4. Unlike the spline wavelets, filters for the Daubechies mother wavelets are orthogonal. However, they do not have linear phase.

Figure 6-4  Daubechies 2 wavelets for case (c). The Daubechies filters are orthogonal and have the same length. The analysis and synthesis filters have similar behavior.

Example 6.1 demonstrates how to compute the product filter \( P_0(z) \) and how to factor it. The common considerations of designing the lowpass FIR filters \( H_0(z) \) and \( G_0(z) \) include linear phase, minimum phase, and orthogonality. In many applications, such as image processing, linear phase is one of the most fundamental properties. In the time domain, linear phase implies that the coefficients of the causal filters are symmetric or antisymmetric around the central coefficient, i.e.,

\[
\begin{align*}
  h[n] &= h[N - n] & \text{symmetry} \\
  h[n] &= -h[N - n] & \text{antisymmetry}
\end{align*}
\]  

(6.24)

In the \( z \)-transform domain, if \( z_k \) is a zero, then so is \( 1/z_k, z_k^* \), and \( 1/z_k^* \) (assume that \( h[n] \) is real-valued). In other words, the zeros of a linear phase FIR filter \( H(z) \) occur in reciprocal conjugate pairs. Spline filters in cases (a) and (b) of Example 6.1 are all linear phase filters.

For FIR filters, minimum phase implies that all zeros lie inside the unit circle, i.e.,

\[
|z_k| \leq 1
\]

(6.25)

The minimum phase filter possesses minimum phase-lag, but minimum phase has historically been the established terminology. Conversely, we name an FIR filter as a maximum phase filter if all its zeros lie outside the unit circle. For the Daubechies wavelets in Eqs. (6.26) and (6.27), while one has minimum phase, the other must have maximum phase. Obviously, no filter can have linear phase and minimum phase simultaneously.

Case (c), in fact, represents one of the most important classes in the wavelet family, the Daubechies wavelets. For the \( k^{th} \) order Daubechies wavelets (which are commonly named Daubechies \( k \) wavelet or simply dbk),
It has been proved that for a given length of orthogonal filters (or given time support), the Daubechies wavelets filters achieve the maximum number of zeros at $z = -1$ (that is, $\omega = \pi$). Since the zeros of the lowpass filter $H_0(z)$ or $G_0(z)$ at $z = -1$ are related to the zero moments of the wavelet $\psi(t)$ for a given time support, the Daubechies filters in Eqs. (6.26) and (6.27) possess the maximum number of vanishing moments.

For $k = 1$, the Daubechies wavelet is identical to the Haar wavelet. As the order $k$ in (6.18) increases, the Daubechies wavelets become more and more smooth and also have more number.
of oscillations. On the other hand, as the order $k$ increases, their time support becomes wider (see Figure 6-5).

Figure 6-6  Scaling functions and wavelets used for the FBI fingerprint compression

Figure 6-7  Zero distribution of the pair of lowpass filters used for the FBI fingerprint compression. While the symbol “x” represents zeros of the analysis lowpass filter, the symbol “o” indicates zeros of the synthesis lowpass filter. The analysis and synthesis filters both have four zeros at $\pi$ (that is, $z = -1$), respectively. Since the reciprocal pairs are assigned to the same filter, the resulting filters have linear phase.

Figure 6-6 illustrates another set of popular scaling functions and wavelets that are employed for the FBI (Federal Bureau of Investigation) fingerprint compression. In this case,

$$P_0(z) = (1 + z^{-1})^8 Q(z)$$

Figure 6-7 depicts the zero distribution of the corresponding lowpass filters, $H_0(z)$ and $G_0(z)$. Both of them have four zeros at $\pi$. Since the pairs of reciprocals, $z_i$ and its reciprocal $1/z_i$ (as well as $z_i^*$ and its reciprocal $1/z_i^*$), are assigned to the same filter, the resulting filters have linear phase.

As mentioned earlier, the derivations in this section are mainly based on the biorthogonal case (see Figure 6-1). The orthogonal transform, in fact, is a special case of its biorthogonal counterpart. In the next section we will investigate conditions, in terms of the product filter $P_0(z)$ and lowpass filters $H_0(z)$ or $G_0(z)$, for orthogonal wavelets.
6.2 Orthogonal Filter Banks

The relationship defined by Eq. (6.9) ensures that the analysis and synthesis filters are orthogonal in the sense of

\[
\sum_n h_i[n-2k] \gamma_i[n] = \delta(k) \quad \text{and} \quad \sum_n h_i[n-2k] \gamma_i[n] = 0 \quad i \neq l
\]

(6.29)

However, the condition (6.29) alone does not ensure that the resulting filter banks \( \{h_i[n]\} \) or \( \{\gamma_i[n]\} \) form orthogonal filter banks and that the resulting wavelets are orthogonal wavelets, such as in the case of the spline wavelets in Example 6.1 a and b. In these cases, there are two sets of scaling functions and mother wavelets. Neither the analysis bank nor the synthesis bank satisfies the power complementarity condition (5.63), i.e.,

\[
H_0(\omega)H_0^*(\omega) + H_1(\omega)H_1^*(\omega) \neq 1
\]

(6.30)

and

\[
G_0(\omega)G_0^*(\omega) + G_1(\omega)G_1^*(\omega) \neq 1
\]

(6.31)

Figure 6-8 plots Eqs.(6.30) and (6.31) for the quadratic spline wavelets in Example 6.1 a. Although the filters in (6.29) meet the condition of perfect reconstruction, the transformation formed by the set of filters \( H_0(\omega) \) or \( G_n(\omega) \) is not energy conserving.

\[
H_0(\omega)H_0^*(\omega) + H_1(\omega)H_1^*(\omega)
\]

\[
G_0(\omega)G_0^*(\omega) + G_1(\omega)G_1^*(\omega)
\]

**Figure 6-8** In the biorthogonal case, neither the analysis filter bank \( H_0(\omega) \) nor the synthesis filter bank \( G_n(\omega) \) satisfy the power complementarity condition. The bases of biorthogonal wavelets do not reproduce the signal energy exactly.

For the filter banks in Example 6.1 a and b the relation corresponding to (5.56) has a form

\[
H_m(z)G_n^*(z) + H_m(-z)G_n^*(-z) = 0 \quad m \neq n
\]

(6.32)

we name the resulting filter banks the biorthogonal filter banks.

In addition to Eq. (6.29), if the filters of the PR filter banks also satisfy the following relationship
then the resulting filter banks are orthogonal filter banks. Obviously, it is a special case of the biorthogonal filter banks. For the orthogonal filter banks, the synthesis and the analysis filters are related by time reversal. The reader can verify that the filter banks for the Daubechies wavelets in Eqs. (6.26) and (6.27) satisfy the condition (6.33).

For orthogonal filter banks, once the product filter $P_0(z)$ is selected, we only need to define $H_0(z)$. With $H_0(z)$, as we will see shortly, we can easily find all the other filters. Many applications demonstrate that the lack of orthogonality complicates quantization and bit allocation between bands, eliminating the principle of conservation of energy. The bases of biorthogonal wavelets do not reproduce the signal energy exactly. Reconstructing a signal from these coefficients may amplify any error introduced in the coefficients. Hence, orthogonal filter banks are often the first choice when we factor the product filter $P_0(z)$.

What constraints does $P_0(z)$ have to meet for orthogonal filter banks?

To achieve Eq. (6.33), we can let

$$H_1(z) = -z^{-N}H_0(-z^{-1})$$

(6.34)

which implies that $h_1[n]$ is the alternating flip of $h_0[n]$, i.e.,

$$(h_1[0], h_1[1], h_1[2], \ldots) = (h_0[N], -h_0[N-1], h_0[N-2], \ldots)$$

(6.35)

From Eqs. (6.9) and (6.34), we can readily compute $G_0(z)$ and $G_1(z)$. For example,

$$G_0(z) = z^{-N}H_0(z^{-1})$$

(6.36)

Therefore, $\gamma_0[n]$ is the flip of $h_0[n]$, i.e.,

$$(\gamma_0[0], \gamma_0[1], \gamma_0[2], \ldots) = (h_0[N], h_0[N-1], h_0[N-2], \ldots)$$

(6.37)

Substituting Eq. (6.36) into Eq. (6.16), we have

$$P_0(z) = z^{-N}H_0(z)H_0(z^{-1})$$

(6.38)

If we define

$$P(z) = H_0(z)H_0(z^{-1})$$

(6.39)

then

$$P_0(z) = z^{-N}P(z)$$

(6.40)

Moreover,

$$P(e^{j\theta}) = \sum_{n=-N}^{N} p[n]e^{-j\theta n} = \left| \sum_{n=0}^{N} h_0[n]e^{-j\theta n} \right|^2$$

(6.41)

which implies that $P(z)$ is non-negative. $P_0(z)$ is the time-shifted non-negative function $P(z)$. It can be shown that the maximally flat filter defined in Eq. (6.18) ensures that this requirement is
met. However, special care must be taken when \( P_0(z) \) has other forms, such as an equiripple half-band filter [47]. Table 6-1 compares different types of filters. Note that the filter banks cannot be orthogonal and linear phase simultaneously.

**Table 6-1 Digital Filters for PR Filter Banks**

<table>
<thead>
<tr>
<th>Filter Type</th>
<th>Location of Zeros</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Real</td>
<td>Complex conjugate symmetrical</td>
<td></td>
</tr>
<tr>
<td>Linear Phase</td>
<td>Each filter must contain both ( z_i ) and its reciprocal ( 1/z_i ). (The pair of reciprocals must be in the same filter.)</td>
<td>Desirable for image processing</td>
</tr>
<tr>
<td>Minimum Phase</td>
<td>All zeros have to be on or inside of the unit circle.</td>
<td>Minimum phase lag</td>
</tr>
<tr>
<td>Orthogonal</td>
<td>Each filter cannot have ( z_i ) and its reciprocal ( 1/z_i ) simultaneously. ( z_i ) and its reciprocal ( 1/z_i ) have to be in separate filters. This condition is contradictory to that required for linear phase filters.</td>
<td>Analysis and synthesis have the same performance Even length ((N \text{ odd})) Convenient for bit allocation and quantization error control Not linear phase</td>
</tr>
</tbody>
</table>

The discussion in this section has been focused on the two-channel perfect reconstruction filter banks. The relationship of the Fourier transform and the scaling function to the frequency response of the FIR filter is given by the infinite products (5.47). From these connections, we reason that since \( H_0(z) \) or \( G_0(z) \) is lowpass and, if it has a high order zero at \( z = -1 \) (i.e., \( \omega = \pi \)), the Fourier transform of the analysis/synthesis scaling function \( \phi(t) \) should drop off rapidly and, therefore, \( \phi(t) \) should be smooth. It turns out that this is indeed true. This is related to the fact that the differentiability of a function is tied to how fast the magnitude of its Fourier transform drops off as the frequency goes to infinity.

It can be shown [6] that the number of zeros at \( z = -1 \) of the lowpass filter \( H_0(z) \) or \( G_0(z) \) determines the number of zero moments of the wavelets. Table 6-2 lists the discrete and continuous moments of the second (db2, see Example 6.1) and third order (db3) Daubechies scaling function and wavelets. While the continuous moments are defined in (5.79) and (5.80), the discrete moments are defined as

\[
\mu_0[k] = \sum_n n^k h_0[n] \quad (6.42)
\]

and

\[
\mu_1[k] = \sum_n n^k h_1[n] \quad (6.43)
\]

The Daubechies filter coefficients ensure the maximum number of zero moments of the wavelets (or maximum vanishing moments), which is weakly related to the number of oscilla-
tions. Researchers have also recognized that in some applications, the zero moments of the scaling function are also useful. However, the filters yielding a combination of zero wavelet and zero scaling function moments cannot be directly generated by the filter \( P_0(z) \) or \( P(z) \) that were introduced earlier. The design of these kinds of filters need to employ other techniques that are beyond the scope of this book. The resulting wavelets are traditionally named \textit{coiflets}. The reader can find related materials in [13], [270], [247], [288], and [400].

Table 6-2  Moments of Daubechies Scaling and Wavelet Functions [6]

<table>
<thead>
<tr>
<th>( k )</th>
<th>( \mu_0[k] )</th>
<th>( m_0[k] )</th>
<th>( \mu_1[k] )</th>
<th>( m_1[k] )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>db2</td>
<td>db3</td>
<td>db2</td>
<td>db3</td>
</tr>
<tr>
<td>0</td>
<td>1.41421</td>
<td>1.41421</td>
<td>1.00000</td>
<td>1.00000</td>
</tr>
<tr>
<td>1</td>
<td>0.89657</td>
<td>1.15597</td>
<td>0.63439</td>
<td>0.81740</td>
</tr>
<tr>
<td>2</td>
<td>0.56840</td>
<td>0.94489</td>
<td>0.40192</td>
<td>0.66814</td>
</tr>
<tr>
<td>3</td>
<td>-0.8643</td>
<td>-0.2243</td>
<td>0.13109</td>
<td>0.44546</td>
</tr>
<tr>
<td>4</td>
<td>-6.0593</td>
<td>-2.6274</td>
<td>-0.3021</td>
<td>0.11722</td>
</tr>
<tr>
<td>5</td>
<td>-23.437</td>
<td>5.30559</td>
<td>-1.0658</td>
<td>-0.0466</td>
</tr>
</tbody>
</table>

The theory of perfect reconstruction filter banks was developed a long time before wavelet analysis became popular, but the original filter banks had no vanishing moments and thus did not always generate finite energy wavelets. "The connection between the number of vanishing moments of a filter and the corresponding wavelet having finite energy is not immediately apparent. But having a conjugate mirror filter \( h[n] \) such that its Fourier transform \( H(\omega) \) vanishes at \( \omega = \pi \) \( (\zeta = -1) \) is a necessary condition so that the cascade of such filters defines a finite energy scaling function and hence a finite energy wavelet. In addition, the number of vanishing moments of a wavelet is equal to the number of zeros of the Fourier transform of its filter at \( \omega = \pi \); saying that a wavelet has one vanishing moment is equivalent to saying that \( H(\pi) = 0 \). More generally, if a wavelet has \( k \) vanishing moments, then \( H(\omega) \) and its first \( k - 1 \) derivatives vanish at \( \omega = 0 \)" [22].

Unlike the Fourier transform in which there is only one set of basis functions, for the wavelet transform one can choose from an infinite number of mother wavelets. The success of the application of wavelet analysis largely hinges on the selection of the mother wavelet. In most applications, such as denoising, the ideal wavelet is one that will encode a signal using the greatest possible number of zero coefficients, or the majority of coefficients closest to zero. Unfortu-

---

4. Except for the 0th moment. For a valid scaling function, its 0th moment has to be equal to one in order for the scaling function to be of lowpass.
nately, such requirements cannot be described mathematically in most cases. The most effective procedure for selecting a proper mother wavelet may be though trial and error. With the help of computer software, such as the Signal Processing Toolset provided by National Instruments, engineers and scientists now can immediately see the effect on their data samples of selecting between different product filters $P_0(z)$ and factorization schemes.

6.3 General Tree-Structure Filter Banks and Wavelet Packets

Once the two-channel perfect reconstruction filter banks have been determined, based on the result obtained in Section 5.4, we can readily compute the discrete wavelet transform through the tree of filter banks, as shown in Figure 6-9. While the subscript “1” indicates a highpass, the subscript “0” corresponds to a lowpass filter. Hence, the symbol $y_{001}$ represents a sequence after two consecutive stages of lowpass filtering and one stage of highpass filtering. Accordingly, the symbol $y_{000}$ represents a sequence after three consecutive stages of lowpass filtering. To compute the discrete wavelet transform, we repeatedly split, filter, and decimate the lowpass bands. The outputs of the highpass filters, such as $y_1$, $y_{01}$, and $y_{001}$, are the discrete wavelet transform that we need. The computational complexity of such a wavelet transform is $O(N)$.

![Figure 6-9](image)

**Figure 6-9** The tree of filter banks for computing the discrete wavelet transform

Intuitively, the tree of filter banks illustrated in Figure 6-9 is not the only possible decomposition scheme. For instance, the signal can also be decomposed by the system depicted in Figure 6-10. Particularly, if we split both the lowpass and highpass bands at all stages, the resulting filter bank structure becomes a full binary tree, as shown in Figure 6-11. This type of tree takes $O(N \log N)$ calculations and results in a completely evenly spaced frequency resolution. The result of the full binary tree is similar to that calculated by the STFT algorithm.

At this point, the question arises as to whether one can generate alternative decomposition trees that allow branches on the highpass bands, and whether such a decomposition has an interpretation in terms of basis functions for linear vector spaces of functions. The answer is yes. The reader can find a corresponding mathematical proof in [27] or [40]. The resulting decomposition is called *wavelet packets* by Coifman and Wickerhauser [261]. The main advantage of the wavelet packets is the flexibility it offers, which allows adaptation to particular signals. The potential
of wavelet packets lies in the capacity to offer a rich menu of orthonormal bases from which the “best” one can be chosen (“best” according to a particular criterion).

![Figure 6-10](image)

**Figure 6-10** An alternative three-stage decomposition

![Figure 6-11](image)

**Figure 6-11** The full binary tree results in a completely evenly spaced frequency resolution, which is similar to that calculated by the STFT.

It can be easily verified that for a given number of stages \( k \), the number of different decompositions is

\[
P(k) = P(k - 1)^2 + 1 \quad P(1) = 1
\]  

(6.44)

The number of possible choices grows dramatically as the stage \( k \) (or scale) increases. A successful application of the wavelet packets includes the FBI standard for fingerprint image compression (see [248] and [250]).