Chapter 2
Filter Banks and DWT

Abstract The study of digital signal processing normally concentrates on the design, realization, and application of single-input, single-output digital filters. There are applications, as in the case of spectrum analyzer, where it is desired to separate a signal into a set of sub-band signals occupying, usually nonoverlapping, portions of the original frequency band. In other applications, it may be desired to combine many such sub-band signals into a single composite signal occupying the whole Nyquist range. To this end, digital filter banks play an important role. Implementation of a filter bank on a processor with finite precision arithmetic necessitates quantization of filter coefficients [95]. This results in loss of perfect reconstruction (PR) property. The theory of filter banks were developed much before modern discrete wavelet transform (DWT) analysis became popular [127, 134]. The study of literature reveals a close relationship between the DWT and digital filter banks. It turns out that a tree of digital filter banks, without computing mother wavelets, can simply achieve the wavelet transform. Hence, the filter banks have been playing a central role in the area of wavelet analysis. It is therefore of interest to study the filter bank theory before addressing the implementation issues of finite precision wavelet transforms. In this chapter, fundamental concept of filter bank theory leading to new implementation issues described in latter chapters is introduced. The material presented in this chapter will be useful in discussing error modeling and parallel computing techniques discussed in the book. In present chapter, the filter bank concept related to DWT is revisited in Sect. 2.1. Section 2.2 presents two-channel PR filter bank. Section 2.3 presents derivation of parallel filter DWT from pyramid DWT structure. Section 2.4 presents frequency response of generated parallel filters followed by conclusion in Sect. 2.5.

Keywords Filter banks · Quantum mirror filter · Aliasing · Computation complexity

K. K. Shukla and A. K. Tiwari, *Efficient Algorithms for Discrete Wavelet Transform*, SpringerBriefs in Computer Science, DOI: 10.1007/978-1-4471-4941-5_2,

2.1 Introduction

In the classical applications of multirate filter banks, a bank of analysis filters is applied to a discrete input signal and then down sampled at fixed rate to produce a set of sub-band signals. If a dual bank of synthesis filters exists, by means of which the original input signal can be recovered by first upsampling each of the above sub-band signals and then applying it to a synthesis filter, then the two filter banks are said to be a perfect reconstruction (PR) pair of filter banks [113]. The term uniform filter bank (UFB) is used to emphasize that all the sub-band signals are downsampled at the same rate [125]. PR pair of wavelet analysis and synthesis filter banks is dual. The discrete wavelet transform (DWT), and multiresolution analysis, can be viewed as the application of a *nonuniform filter bank*, defined by a UFB. In terms of wavelet theory, a *low-pass* filter corresponds *to scaling* function and the subsequent *high-pass* or *band-pass* filter corresponds to *wavelet* function. The DWT computation involves repetitive application of UFB on the *low-pass* channel. In the literatures, wavelet transform have been treated in considerable detail and wavelet decompositions have been related to PR Filter Bank [35, 132, 134, 139].

The concept of PR is meaningful only in the ideal cases. In most real world applications of finite world length, some sort of error is always introduced in the coding process or during the transmission over lossy channels. The advantage of multiresolution scheme is that the redundancy is introduced more in low-frequency channels compared to high-frequency channels. Thus, these representations may be advantageous for certain classes of signals such as natural images.

2.2 Orthogonal Filter Banks

The *digital filter bank* is defined as a set of digital band-pass filters with either a common input or a summed output and is referred as *analysis* and *synthesis* filter bank, respectively. The operation of *analysis* and *synthesis* filter bank is dual to each other. The combined structure of *analysis* and *synthesis* filter bank is quadrature mirror filter (QMF) bank [113].

Process of filtering is usually related with frequency selectivity. For example, an ideal discrete-time *low-pass* filter with cutoff frequency $\omega_c < \pi$ takes any input signal and projects it onto the subspace of signals bandlimited to $[-\omega_c, \omega_c]$. Orthogonal discrete-time filter banks perform a similar projection. Assume a discrete-time filter with finite impulse response $g_g[n] = \{g_g[0], g_g[1], \ldots, g_g[L]\}$, L even, and the property [107]

$$\langle g_g[n], g_g[n - 2k] \rangle = \delta_k \tag{2.1}$$

that is, the impulse response is orthogonal to its even shifts, and $\|g_g\|_2 = 1$ [107]. The z-transform of impulse response $g_g[n]$ is

$$G_g[z] = \sum_{n=0}^{L-1} g_g[n] z^{-n}. \tag{2.2}$$

Further, with an assumption that $g_g[n]$ is a *low-pass* filter, corresponding *high-pass* filter $g_h[n]$ with z-transform, is given as follows:

$$G_h[z] = z^{-L+1} G_g(-z^{-1}). \tag{2.3}$$

Here, three operations have been applied [107] as follows:

1. $z \rightarrow -z$ corresponds to modulation by $(-1)^n$ or transforming the *low pass* into *high pass*.
2. $-z \rightarrow -z^{-1}$ applies time reversal to the impulse response.
3. Multiplication by z^{-L+1} makes the time-reversed impulse response causal.

This special way of obtaining a *high pass* from a *low pass*, introduced as quadrature conjugate filter (QCF) [128], has the following properties:

$$\langle g_h[n],\, g_h[n - 2k] \rangle = \delta_k \tag{2.4}$$

that is, the impulse response is orthogonal to its even shifts and

$$\langle g_g[n],\, g_h[n - 2k] \rangle = 0 \tag{2.5}$$

or the impulse response $\{g_g[n],\, g_h[n]\}$ and their even shifts are mutually orthogonal. Further, $\{g_g[n - 2k],\, g_h[n - 2l]\}_{k,l \in Z}$ is an orthonormal basis for $L_2(Z)$, the space of square summable sequences. Thus, any sequence from $L_2(Z)$ can be written as follows:

$$x[n] = \sum_{k \in Z} \alpha_k g_g[n - 2k] + \sum_{l \in Z} \beta_l g_h[n - 2l] \tag{2.6}$$

where $\alpha_k = \langle g_g[n - 2k],\, x[n] \rangle$ and $\beta_l = \langle g_h[n - 2l],\, x[n] \rangle$, k and $l \in Z$.

2.2.1 Two-Channel Quadrature Mirror Filter Bank

In filter bank applications, a discrete-time signal $x[n]$ is split into sub-band signals by means of an *analysis filter bank*. The sub-band signals are then processed and finally combined by a *synthesis filter bank* resulting in an output signal $y[n]$. If the sub-band signals are bandlimited to frequency ranges much smaller than that of the original input signal, they could be *downsampled* before processing. Due to lower sampling rate, the processing of the downsampled signals can be carried out more

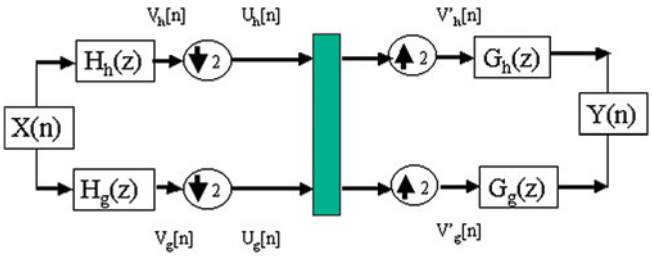

Fig. 2.1 Two-channel filter bank. $H_h(z)$ and $H_g(z)$ form an analysis filter bank, whereas $G_h(z)$ and $G_g(z)$ form a synthesis filter bank

efficiently. After processing, these signals are *upsampled* before being combined by the *synthesis filter bank* into a higher-rate signal. The filter bank theory dealt in detail in literature [80, 100, 127, 134, 139] is discussed in brief in this section.

Once the *low-pass* and *high-pass* filters have been computed, it is possible to compute the *scaling* function and the *mother wavelet*. Moreover, under certain conditions, the outputs of the *high-pass* filters are good approximations of the *wavelet* series. Consequently, the selection of desired *scaling* function and *mother wavelets* reduces to the design of *low-pass* and *high-pass* filters of two-channel PR filter banks. A tree of two-channel PR filter banks can simply realize the wavelet transform. Figure 2.1 sketches a typical two-channel PR filter bank system. It is convenient to analyze the filter bank in z-domain. As shown in Fig. 2.1, the signal $X(z)$ is first filtered by a filter bank consisting of $H_h(z)$ and $H_g(z)$.

The outputs of $H_h(z)$ and $H_g(z)$ are downsampled by 2 to obtain $U(z)$. After some processing, the modified signals are upsampled and filtered by another filter bank consisting of $G_h(z)$ and $G_g(z)$. The downsampling operators are decimators, and the upsampling operators are expanders. If no processing takes place between the two filter banks (in other words, $U(z)$ are not altered), the sum of the outputs of $G_h(z)$ and $G_g(z)$ is identical to the original signal $X(z)$, except for a time delay. Such a system is commonly referred to as a two-channel PR filter bank. $H_h(z)$ and $H_g(z)$ form an analysis filter bank, whereas $G_h(z)$ and $G_g(z)$ form a synthesis filter bank. The z-transform of input–output relations is defined as given by *Upil* in this chapter [26];

$$V_k(z) = H_k(z)X(z) \tag{2.7}$$

$$U_k(z) = \frac{1}{2}\left\{V_k(z^{\frac{1}{2}}) + V_k(-z^{\frac{1}{2}})\right\} \tag{2.8}$$

$$\hat{V}_k(z) = U_k(z^2) \tag{2.9}$$

where k refers to h and g (h and g are outputs of *high-pass* and *low-pass* filters, respectively).

Further, it can be shown that

$$\begin{aligned}\hat{V}_k(z) &= \frac{1}{2}\{V_k(z) + V_k(-z)\} \\ &= \frac{1}{2}\{H_k(z)X(z) + H_k(-z)X(-z)\}\end{aligned} \tag{2.10}$$

and the reconstructed output of the filter bank is given by

$$Y(z) = \frac{1}{2}\left\{G_h(z)\hat{V}_h(z) + G_g(z)\hat{V}_g(z)\right\}. \tag{2.11}$$

Substituting Eq. (2.10) in (2.11), the output of the filter bank is given as follows:

$$\begin{aligned}Y(z) = &\frac{1}{2}\left\{H_g(z)G_g(z) + H_h(z)G_h(z)\right\}X(z) \\ &+ \frac{1}{2}\left\{H_g(-z)G_g(z) + H_h(-z)G_h(z)\right\}X(-z)\end{aligned} \tag{2.12}$$

The second term in the above equation is precisely due to aliasing caused by sampling rate alteration. The above equation is rewritten as follows:

$$Y(z) = T(z)X(z) + A(z)X(-z) \tag{2.13}$$

where

$$T(z) = \frac{1}{2}\left\{H_g(z)G_g(z) + H_h(z)G_h(z)\right\} \tag{2.14}$$

is called distortion transfer function and

$$A(z) = \frac{1}{2}\left\{H_g(-z)G_g(z) + H_h(-z)G_h(z)\right\}, \tag{2.15}$$

the term with $X(-z)$, is traditionally called the *aliasing term* matrix.

The relation for $Y(z)$ may be expressed in the matrix form as follows:

$$Y(z) = \frac{1}{2}[X(z) \quad X(-z)]\begin{bmatrix} H_g(z) & H_h(z) \\ H_g(-z) & H_h(-z) \end{bmatrix}\begin{bmatrix} G_g(z) \\ G_h(z) \end{bmatrix} \tag{2.16}$$

The 2×2 matrix in the above equation is given as follows:

$$H(z) = \begin{bmatrix} H_g(z) & H_h(z) \\ H_g(-z) & H_h(-z) \end{bmatrix} \tag{2.17}$$

In general, the QMF structure discussed above is a linear time-varying system. However, it is possible to select the analysis and synthesis filters such that the aliasing effect is canceled, resulting in a linear time-invariant (LTI) operation. To this end, we need to ensure that

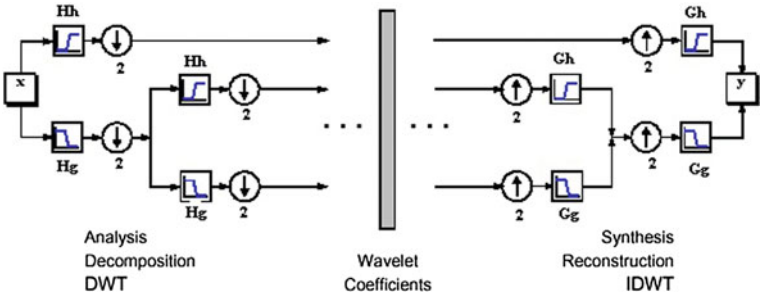

Fig. 2.2 Multilevel wavelet decomposition/reconstruction using multiple PRQMF bank

$$2A(z) = \{H_g(-z)G_g(z) + H_h(-z)G_h(z)\} = 0 \tag{2.18}$$

There are various possible solutions of the above equation. One solution may be given by

$$G_g(z) = H_g(-z), \quad G_h(z) = -H_g(-z). \tag{2.19}$$

If above relation holds, then Eq. (2.13) reduces to

$$Y(z) = T(z)X(z) \tag{2.20}$$

with

$$T(z) = \frac{1}{2}\{H_g(z)H_h(-z) - H_h(z)H_g(-z)\} \tag{2.21}$$

Thus, an orthogonal filter bank splits the input space into *low-pass* approximation space V_g and its *high-pass* orthogonal component V_h. The space V_g corresponds to a *coarse approximation*, while V_h contains additional *details*. This is the first step in the multiresolution analysis that is obtained when iterating the *high-pass/low-pass* division on the *low-pass* branch (Fig. 2.1).

If an alias-free QMF bank has no amplitude and phase distortion, then it is called a perfect reconstruction mirror filter (PRQMF) bank. The time domain equivalent of the output is given by [100]

$$y(n) = \mathrm{d}x(n - n_0) \tag{2.22}$$

for all possible inputs. This indicates that the reconstructed output $y(n)$ is a scaled and delayed replica of the input. Thus, it is evident that the output resembles with the basic properties of the wavelet decomposition/reconstruction. The combination of multiple PRQMF bank results in multilevel wavelet decomposition/reconstruction as shown in Fig. 2.2.

2.2.2 Computational Complexity of Discrete Wavelet Transform

Rioul et al. in their seminal paper [106] have studied the computational complexity of wavelet transforms in detail. In general, the computations are periodic in 2^m for an m-level wavelet. Here, each filtered output is decimated by a factor of 2. This necessitates computation of those signal samples that are not thrown away. Consider an input set of $N = 2^m$ samples. For the first level, each filter computes $N/2$ samples, so the total number of samples generated at the *low-pass* and *high-pass* filters of level-1 wavelet is N. Similarly, each filter in the second-level wavelet computes $N/4$ samples, and the total number of samples computed at level 2 is $N/2$. In an m-level wavelet, the total number of samples computed is

$$N + \frac{N}{2} + \frac{N}{4} + \cdots\cdots + 2 = 2(N - 1). \tag{2.23}$$

Since the wavelet computation is periodic with N samples, the number of samples computed every sample period is $\frac{2(N-1)}{N}$ or $2\left(1 - \frac{1}{N}\right)$, which is upper bounded by 2 [106]. This implies that the maximum number of filters needed for computation in a one-dimensional multilevel forward wavelet transform is two. In other words, one *low-pass* and one *high-pass* filter will always be adequate for computation of one-dimensional DWT. The parallel filter bank structure discussed in next Section will lead to an efficient means for computation of wavelet transform.

2.3 Parallel Filter Bank Realization of Multilevel Discrete Wavelet Transform

As the computation of DWT involves filtering, an efficient filtering process is essential in DWT hardware implementation. A possible solution is based on Mallat algorithm [87] requiring only two filters (one *high-* and one *low*-pass filter). In the multistage DWT, coefficients are calculated recursively, and in addition to the wavelet decomposition stage, extra space is required to store the intermediate coefficients. Hence, the overall performance depends significantly on the precision of the intermediate DWT coefficients [74] as discussed in detail in next chapter. An alternative method for fast and efficient implementation of DWT transform is based on parallel filter implementation. In this, cascaded *high-pass* and *low-pass* filters at different resolution levels will be replaced by their equivalent filter [80], [107]. This necessitates number of filters to be of the order of decomposition level. The main advantage of the parallel filter algorithm is that it does not require storing intermediate coefficients [123]. Another advantage of this architecture is that the word length can be arbitrary and is not restricted to be a multiple of 2^m for m-resolution-level wavelet decomposition.

As discussed, Fig. 2.2 is a multilevel representation of DWT. The DWT evaluation is based on binary tree structured QMF. The output from *high-pass* filter is termed as detailed wavelet coefficients and from *low-pass* filter is termed as approximation coefficients. The approximation coefficients from previous level, after passing through another PRQMF filter bank, generate another set of detailed and approximation coefficients, and the decomposition process is continued until one reaches desired level of decomposition. The limitation here is that if the DWT coefficients of level L are of use, one has to first obtain the DWT coefficients at level $L-1$, thus increasing computational burden. Souani et al. [123] presented an efficient one-dimensional direct DWT computation algorithm. The algorithm enables computation of Lth-level DWT coefficients without prior knowledge of $(L-1)$th-level DWT coefficients. The algorithm is simple and uses a modified filter structure generated out of basic PRQMF filter bank. As discussed in next chapters, the algorithm is suitable from finite precision and parallel implementation viewpoint. Its implementation necessitates, finding equivalent parallel filters generated out of PRQMF filter bank to compute the DWT coefficient at any level from signal itself.

2.3.1 Iterated Filters and Regularity

The DWT filters roughly correspond to octave band filters. In many applications, *low-frequency* content of the signal is an important part. It is what gives the signal its identity. The *high-frequency* content, on the other hand, imparts flavor. For example, in the human voice, removing *high-frequency* components sounds different, but contents can still be inferred. However, removal of the *low-frequency* components sounds gibberish.

It is required to find the equivalent filter corresponding to the lower branch in Fig. 2.2 that is the iterated *low-pass* filter. It can be easily checked that subsampling by two followed by filtering with $G(z)$ is equivalent to filtering with $G(z^2)$ followed by the subsampling [80, 107]. Thus, the first two steps of *low-pass* filtering can be replaced with z-transform $G(z)$. $G(z^2)$, followed by subsampling by 4. In general, representing $G^J(z)$ the equivalent filter to the Jth stages of *low-pass* filtering and subsampling by 2^J [139]:

$$G^J(z) = \prod_{l=0}^{J-1} G(z^{2^l}) \tag{2.24}$$

A necessary condition for the iterated functions to converge to a continuous limit is that the filter $G(z)$ should have sufficient number of zeros at $z=-1$, or half sampling frequency, so as to attenuate repeat spectra [107]. Using this condition, the regular filters, which are both orthogonal and converge to continuous functions with compact support, may be generated. The well-known Daubechies orthonormal filters [36] are deduced from maximally flat *low-pass* filters.

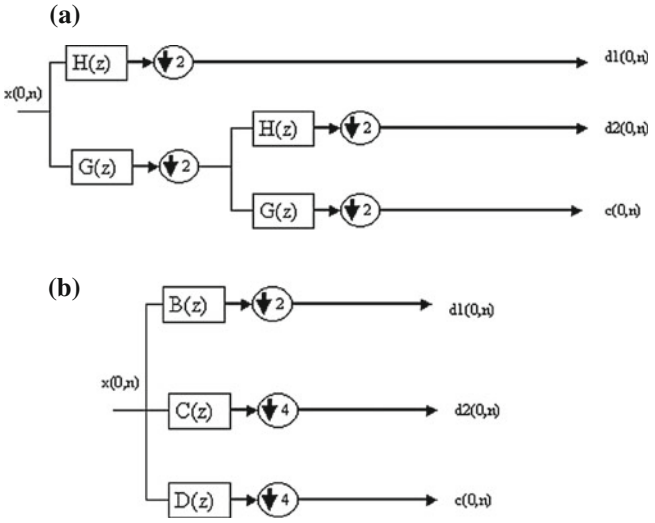

Fig. 2.3 Parallel filter implementation of two-level DWT decomposition. **a** Pyramid structure DWT. **b** Parallel filter DWT

2.3.1.1 Generation of Parallel Filter Banks

In present chapter for the sake of simplicity, the algorithm has been demonstrated only for two levels and three levels of DWT decomposition. For L level, it can be generalized there from. Consider the two-level DWT decomposition Mallat's algorithm [87] and derived parallel filter equivalent as shown in Fig. 2.3.

The equivalent analysis filters for two-level DWT (Fig. 2.3) are expressed in terms of PRQMF filter bank as follows:

$$B(z) = H(z) \tag{2.25}$$

$$C(z) = G(z)H\left(z^2\right) \tag{2.26}$$

$$D(z) = G(z)G\left(z^2\right) \tag{2.27}$$

Similarly, the equivalent analysis filters for three-level DWT, as shown in Fig. 2.4, are expressed in terms of PRQMF filter bank as follows:

$$B(z) = H(z) \tag{2.28}$$

$$C(z) = G(z)H\left(z^2\right) \tag{2.29}$$

$$D(z) = G(z)G\left(z^2\right)H\left(z^4\right) \tag{2.30}$$

$$E(z) = G(z)G\left(z^2\right)G\left(z^4\right) \tag{2.31}$$

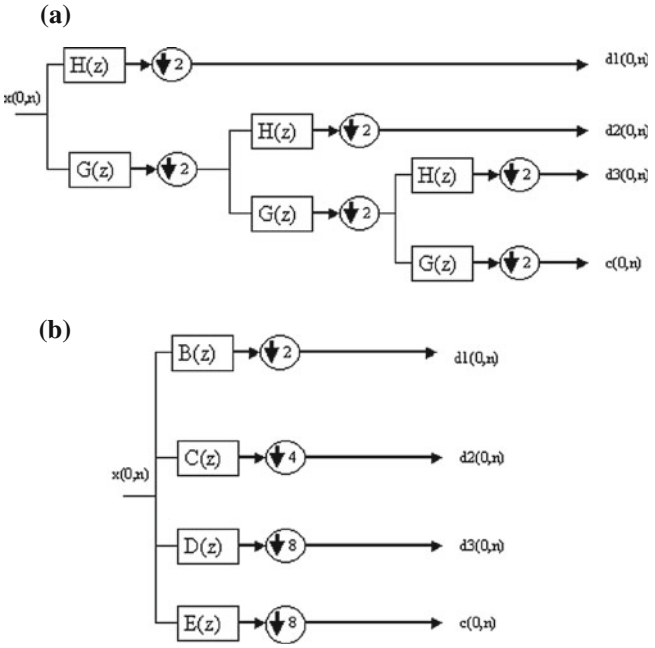

Fig. 2.4 Parallel filter implementation of three-level DWT decomposition. **a** Pyramid structure DWT. **b** Parallel filter DWT

More generally for J-level decomposition, the equivalent filter to J stages of low-pass filtering and subsampling by two (a total subsampling by 2^J) is given by [107]

$$E^J(z) = \prod_{l=0}^{J-1} G(Z^{2^l}) \tag{2.31}$$

This explains that the generated filter length will increase with increase in depth of decomposition. The equivalent synthesis filters can be generated accordingly.

2.3.2 One Set of Forward Discrete Wavelet Transform Computation

The computation of the 3-level wavelet is periodic with period 2^3 (or 8), that is, identical sets of computations are separated by a time index of 2^3 [4]. To explain the generated parallel filter bank structure, it is required to write down the set of computations associated with one period in forward DWT decomposition. This set completely describes the computations. All other computations are generated from

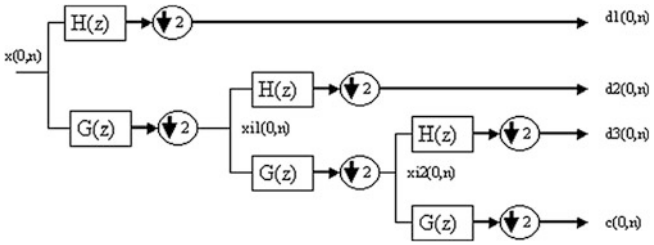

Fig. 2.5 Implementation of three-level DWT decomposition and intermediate coefficients (*xi*1 and *xi*2 are input to PRQMF at level two and three, respectively)

this set by shifting the time by multiples of the period. For simplicity, following transfer function representation of filters used in PRQMF filter bank with *L*–tap filters is assumed as follows:

$$H(z) = \sum_{n=0}^{L-1} h(z)z^{-n} \tag{2.32}$$

$$G(z) = \sum_{n=0}^{L-1} g(z)z^{-n} \tag{2.33}$$

For simplicity, the filter tap is selected to $L = 6$.

For details of multilevel DWT coefficient computation, readers are advised to see [60, 111, 123]. One period of DWT computation, described in Fig. 2.5, is as follows:

The first-level computations are as follows:

$$\begin{aligned}
d1(0) &= h(0)x(0) + h(1)x(-1) + h(2)x(-2) + h(3)x(-3) + h(4)x(-4) + h(5)x(-5)\\
d1(2) &= h(0)x(2) + h(1)x(1) + h(2)x(0) + h(3)x(-1) + h(4)x(-2) + h(5)x(-3)\\
d1(4) &= h(0)x(4) + h(1)x(3) + h(2)x(2) + h(3)x(1) + h(4)x(0) + h(5)x(-1)\\
d1(6) &= h(0)x(6) + h(1)x(5) + h(2)x(4) + h(3)x(3) + h(4)x(2) + h(5)x(1)\\
xi1(0) &= g(0)x(0) + g(1)x(-1) + g(2)x(-2) + g(3)x(-3) + g(4)x(-4) + g(5)x(-5)\\
xi1(2) &= g(0)x(2) + g(1)x(1) + g(2)x(0) + g(3)x(-1) + g(4)x(-2) + g(5)x(-3)\\
xi1(4) &= g(0)x(4) + g(1)x(3) + g(2)x(2) + g(3)x(1) + g(4)x(0) + g(5)x(-1)\\
xi1(6) &= g(0)x(6) + g(1)x(5) + g(2)x(4) + g(3)x(3) + g(4)x(2) + g(5)x(1)
\end{aligned} \tag{2.34}$$

The second-level computations are as follows:

$$\begin{aligned}
d2(0) &= h(0)xi1(0) + h(1)xi1(-2) + h(2)xi1(-4) + h(3)xi1(-6) + h(4)xi1(-8) + h(5)xi1(-10)\\
d2(4) &= h(0)xi1(4) + h(1)xi1(2) + h(2)xi1(0) + h(3)xi1(-2) + h(4)xi1(-4) + h(5)xi1(-6)\\
xi2(0) &= g(0)xi1(0) + g(1)xi1(-2) + g(2)xi1(-4) + g(3)xi1(-6) + g(4)xi1(-8) + g(5)xi1(-10)\\
xi2(4) &= g(0)xi1(4) + g(1)xi1(2) + g(2)xi1(0) + g(3)xi1(-2) + g(4)xi1(-4) + g(5)xi1(-6)
\end{aligned} \tag{2.35}$$

The third-level computations are as follows:

$$\begin{aligned} d3(0) &= h(0)xi2(0) + h(1)xi2(-4) + h(2)xi1(-8) + h(3)xi1(-12) + h(4)xi2(-16) + h(5)xi2(-20) \\ c(0) &= g(0)xi2(0) + g(1)xi2(-4) + g(2)xi2(-8) + g(3)xi2(-12) + g(4)xi2(-16) + g(5)xi2(-20) \end{aligned} \tag{2.36}$$

The variables $d1$, $d2$, $d3$, c, x, $xi1$, and $xi2$ are appropriately defined in Fig. 2.5. The negative time indexes in these equations correspond to the reference starting time unit 0. By adding one or multiples of the periods of computation to these equations, the next sets of computations are obtained. The condensed form of Eqs. (2.34–2.36) [123] is as follows:

$$d1(2k) = \sum_{p=0}^{L-1} h(p)x(2k - p) \tag{2.37}$$

$$xi1(2k) = \sum_{p=0}^{L-1} g(p)x(2k - p) \tag{2.38}$$

$$d2(4k) = \sum_{p=0}^{L-1} h(p)xi1(4k - 2p) \tag{2.39}$$

$$xi2(4k) = \sum_{p=0}^{L-1} g(p)xi1(4k - 2p) \tag{2.40}$$

$$d3(8k) = \sum_{p=0}^{L-1} h(p)xi2(8k - 4p) \tag{2.41}$$

$$c(8k) = \sum_{p=0}^{L-1} g(p)xi2(8k - 4p) \tag{2.42}$$

In the above equations, the coefficients obtained by $d1$, $d2$, $d3$, and c are final DWT coefficients and coefficients obtained by $xi1$ and $xi2$ are intermediate. Variable x denotes the input signal. The DWT computation is complex because of the data dependencies at different octaves. Above equations show the relationship among final and intermediate coefficients.

Implementation of the three-level DWT necessitates total of six filters to be used. The filters are a pair of identical PRQMF filter bank used at each stage. The DWT coefficients could be derived in terms of input signal $x(n)$ only, thus eliminating the intermediate-level coefficients. This will lead derivation of new filters, as per Eqs. (2.28–2.31), enabling computation of DWT coefficients independent of intermediate coefficients. The filter B is the same as high-pass filter H with length $L^B = L$. The filter lengths of generated filters C, D, and E are as follows:

Table 2.1 Generated filters length in terms of base PRQMF filter length

PRQMF filter tap length	Generated filter length			
L	B ($L^B = L$)	C ($L^C = 3L - 2$)	D ($L^D = 7L - 6$)	E ($L^E = 7L - 6$)
4	4	10	22	22
6	6	16	36	36
8	8	22	50	50
10	10	28	64	64
12	12	34	78	78

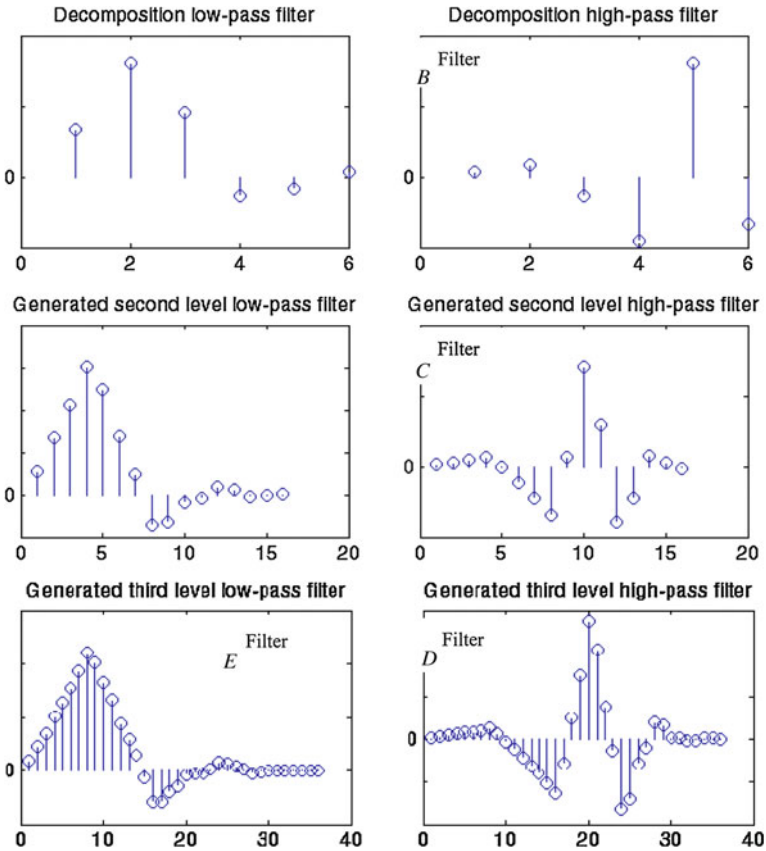

Fig. 2.6 Impulse response plot of generated parallel filters for three-level DWT, (x-axis: filter coefficient number and y-axis: corresponding magnitude)

$$\begin{aligned} L^C &= 3L - 2 \\ L^D &= 7L - 6 \\ L^E &= 7L - 6 \end{aligned} \tag{2.43}$$

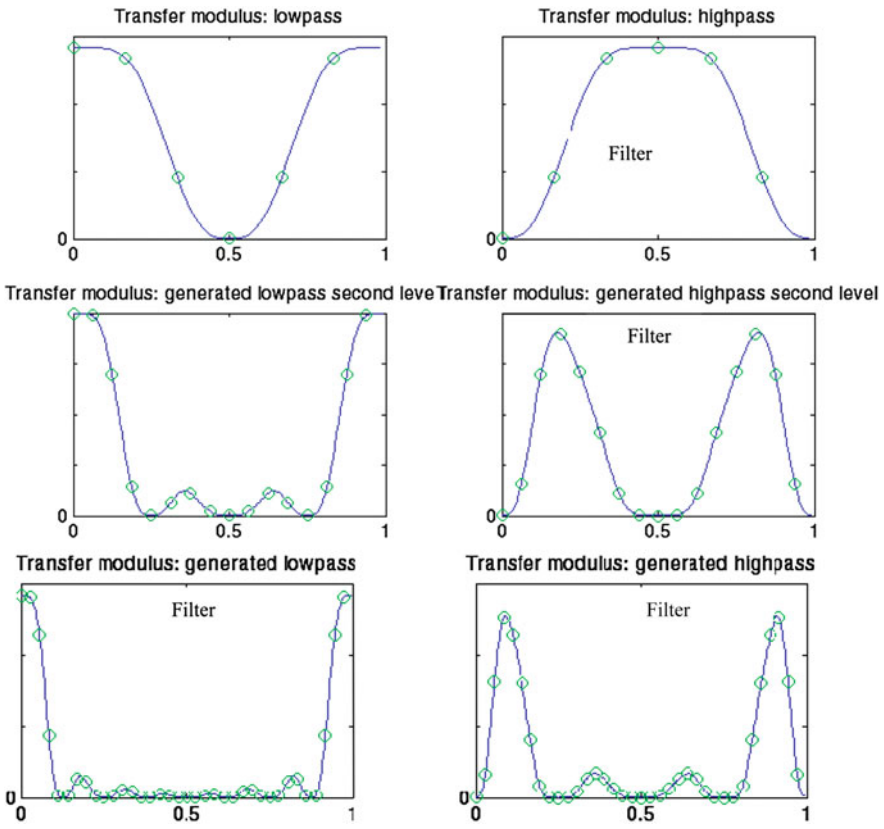

Fig. 2.7 Frequency response of generated parallel filters for three-level DWT

The generated parallel filters lengths for varied PRQMF filter lengths are given in Table 2.1. It is evident that the filter *B* operates every two samples (down-sampling by 2). Filter *C* operates every four samples; filters *D* and *E* operate every eight samples. For an even order of the input data, filters *B*, *C*, *D*, and *E* will operate depending on their decimation rate.

2.4 Frequency Response of Generated Parallel Filter Bank

To validate the parallel filter DWT structure, frequency response plots are generated. The frequency response plots corresponding to three-level DWT decomposition are shown. The selected PRQMF filter bank is a Daubechies filter [37] with six taps, and Symlet filter [14] with eight taps. The experimentation has been carried out on a Pentium III, 733 MHz system using Matlab [93].

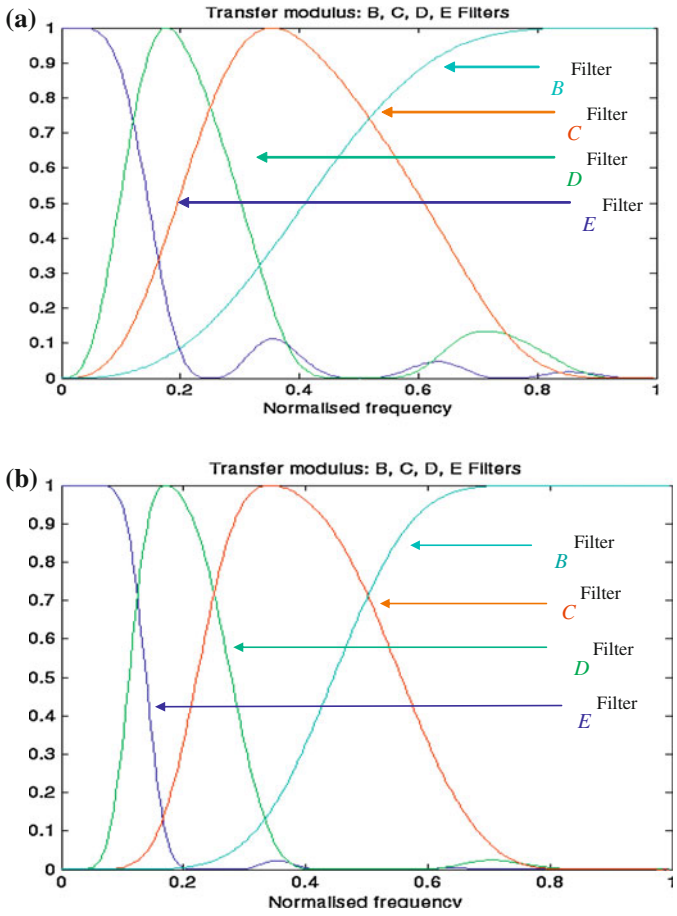

Fig. 2.8 **a** Frequency plot of generated parallel filter structure (Daub 6-tap PRQMF filter bank), **b** Frequency plot of generated parallel filter structure (Symlet 8-tap PRQMF filter bank)

Figure 2.6 is an impulse response plot of generated parallel filter structure. Figure 2.7 plots frequency response of generated parallel filter for two- and three-level DWT decomposition. It is evident from the plots that for one set of PRQMF filter bank, the generated parallel filters do confirm the frequency response desired at various levels. The parallel filter corresponding to approximate DWT coefficients (filter *E*, Figs. 2.3b and 2.4b) resembles *low-pass* filter (*scaling* function). Filters *C* and *D* correspond to *band-pass* (*high-pass*) filters (*wavelet* function) and filter *B* corresponds to *high-pass* filter, which is in turn *high-pass* filter of PRQMF filter bank (Fig. 2.8). Figure 2.9 is gain plot of derived parallel filters. This again confirms suitability of parallel filter structure for DWT decomposition.

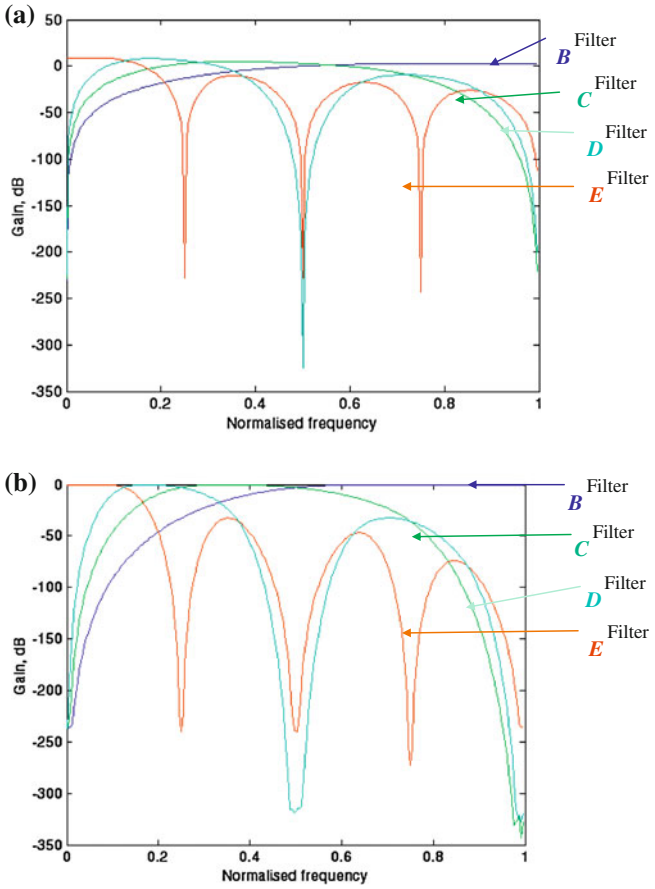

Fig. 2.9 **a** Gain plot of generated parallel filters for three-level DWT (db6), **b** Gain plot of generated parallel filters for three-level DWT (sym8 PRQMF)

2.5 Conclusions

The filter bank structure of DWT is analyzed. The relation for PR is presented. Computational complexity for DWT presented in this chapter will be basis for development of error analysis model in the next chapter. An alternative structure of DWT in terms of parallel filters is also derived. Impulse response and frequency response plots of generated parallel filter structure validate its suitability in terms of dyadic frequency selectivity. Chapter 3 presents comparison of finite precision error of the two models.

http://www.springer.com/978-1-4471-4940-8

Efficient Algorithms for Discrete Wavelet Transform
With Applications to Denoising and Fuzzy Inference
Systems

Shukla, S.K.; Tiwari, A.K.

2013, IX, 91 p. 46 illus., 31 illus. in color., Softcover
ISBN: 978-1-4471-4940-8