

Chapter 6

Filter Banks

Filter banks are arrangements of low pass, bandpass, and highpass filters used for the spectral decomposition and composition of signals. They play an important role in many modern signal processing applications such as audio and image coding. The reason for their popularity is the fact that they easily allow the extraction of spectral components of a signal while providing very efficient implementations. Since most filter banks involve various sampling rates, they are also referred to as *multirate systems*. To give an example, Figure 6.1 shows an M -channel filter bank. The input signal is decomposed into M so-called *subband signals* by applying M analysis filters with different passbands. Thus, each of the subband signals carries information on the input signal in a particular frequency band. The blocks with arrows pointing downwards in Figure 6.1 indicate downsampling (subsampling) by factor N , and the blocks with arrows pointing upwards indicate upsampling by N . Downsampling by N means that only every N th sample is taken. This operation serves to reduce or eliminate redundancies in the M subband signals. Upsampling by N means the insertion of $N - 1$ consecutive zeros between the samples. This allows us to recover the original sampling rate. The upsamplers are followed by filters which replace the inserted zeros with meaningful values. In the case $M = N$ we speak of *critical subsampling*, because this is the maximum downsampling factor for which *perfect reconstruction* can be achieved. Perfect reconstruction means that the output signal is a copy of the input signal with no further distortion than a time shift and amplitude scaling.

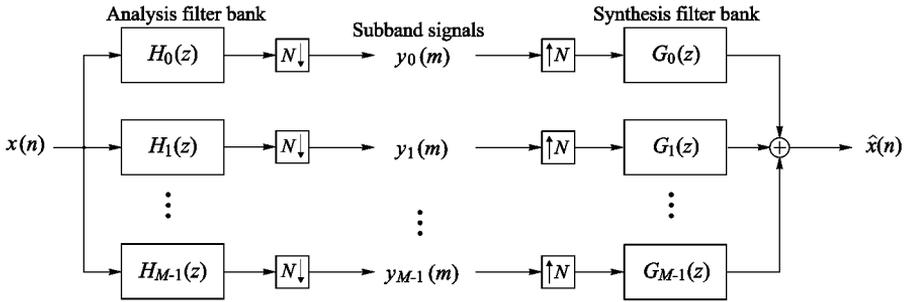


Figure 6.1. M -channel filter bank.

From the mathematical point of view, a filter bank carries out a series expansion, where the subband signals are the coefficients, and the time-shifted variants $g_k(n - iN)$, $i \in \mathbb{Z}$, of the synthesis filter impulse responses $g_k(n)$ form the basis. The main difference to the block transforms is that the lengths of the filter impulse responses are usually larger than N so that the basis sequences overlap.

6.1 Basic Multirate Operations

6.1.1 Decimation and Interpolation

In this section, we derive spectral interpretations for the decimation and interpolation operations that occur in every multirate system. For this, we consider the configuration in Figure 6.2. The sequence $v(n)$ results from inserting zeros into $y(m)$. Because of the different sampling rates we obtain the following relationship between $Y(z)$ and $V(z)$:

$$Y(z^N) = V(z). \tag{6.1}$$

After downsampling and upsampling by N the values $v(nN)$ and $u(nN)$ are still equal, while all other samples of $v(n)$ are zero. Using the correspondence

$$\frac{1}{N} \sum_{i=0}^{N-1} e^{j2\pi in/N} = \begin{cases} 1 & \text{for } n/N \in \mathbb{Z}, \\ 0 & \text{otherwise,} \end{cases} \tag{6.2}$$

the relationship between $v(n)$ and $u(n)$ can be written as

$$v(n) = u(n) \frac{1}{N} \sum_{i=0}^{N-1} W_N^{-in}, \tag{6.3}$$

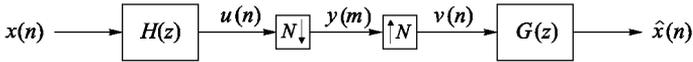


Figure 6.2. Typical components of a filter bank.

where

$$W_N = e^{-j2\pi/N}. \quad (6.4)$$

The z -transform is given by

$$\begin{aligned} V(z) &= \sum_{n=-\infty}^{\infty} v(n)z^{-n} \\ &= \frac{1}{N} \sum_{i=0}^{N-1} \sum_{n=-\infty}^{\infty} u(n) [W_N^i z]^{-n} \\ &= \frac{1}{N} \sum_{i=0}^{N-1} U(W_N^i z). \end{aligned} \quad (6.5)$$

The relationship between $Y(z)$ and $U(z)$ is concluded from (6.1) and (6.5):

$$Y(z) = \frac{1}{N} \sum_{i=0}^{N-1} U(W_N^i z^{\frac{1}{N}}). \quad (6.6)$$

With (6.6) and $U(z) = H(z)X(z)$ we have the following relationship between $Y(z)$ and $X(z)$:

$$Y(z) = \frac{1}{N} \sum_{i=0}^{N-1} H(W_N^i z^{\frac{1}{N}}) X(W_N^i z^{\frac{1}{N}}). \quad (6.7)$$

From (6.1) and (6.7) we finally conclude

$$\begin{aligned} \hat{X}(z) &= G(z) Y(z^N) \\ &= \frac{1}{N} \sum_{i=0}^{N-1} G(z) H(W_N^i z) X(W_N^i z). \end{aligned} \quad (6.8)$$

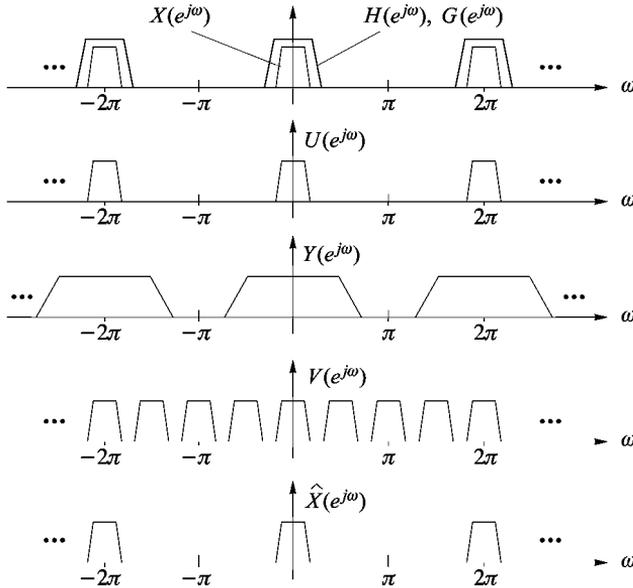


Figure 6.3. Signal spectra for decimation and interpolation according to the structure in Figure 6.2 (non-aliased case).

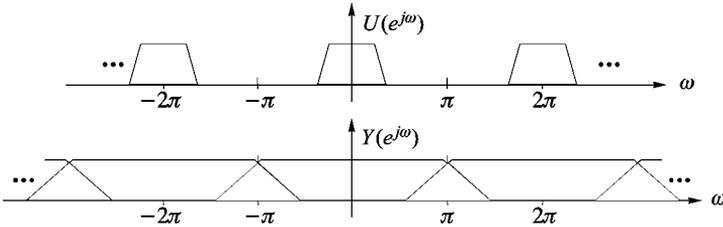


Figure 6.4. Signal spectra in the aliased case.

The spectra of the signals occurring in Figure 6.2 are illustrated in Figure 6.3 for the case of a narrowband lowpass input signal $x(n)$, which does not lead to aliasing effects. This means that the products $G(z)(H(W_N^i z)X(W_N^i z))$ in (6.8) are zero for $i \neq 0$. The general case with aliasing occurs when the spectra become overlapping. This is shown in Figure 6.4, where the shaded areas indicate the aliasing components that occur due to subsampling. It is clear that $x(n)$ can only be recovered from $y(m)$ if no aliasing occurs. However, the aliased case is the normal operation mode in multirate filter banks. The reason why such filter banks allow perfect reconstruction lies in the fact that they can be designed in such a way that the aliasing components from all parallel branches compensate at the output.

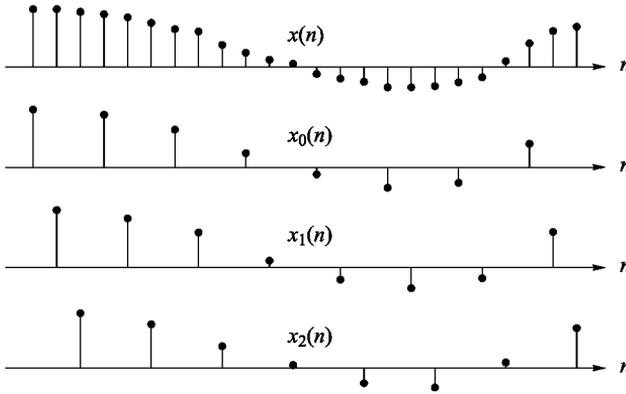


Figure 6.5. Type-1 polyphase decomposition for $M = 3$.

6.1.2 Polyphase Decomposition

Consider the decomposition of a sequence $x(n)$ into sub-sequences $x_i(m)$, as shown in Figure 6.5. Interleaving all $x_i(m)$ again yields the original $x(n)$. This decomposition is called a *polyphase decomposition*, and the $x_i(m)$ are the *polyphase components* of $x(n)$. Several types of polyphase decompositions are known, which are briefly discussed below.

Type-1. A type-1 polyphase decomposition of a sequence $x(n)$ into M components is given by

$$X(z) = \sum_{\ell=0}^{M-1} z^{-\ell} X_{\ell}(z^M), \tag{6.9}$$

where

$$X_{\ell}(z) \longleftrightarrow x_{\ell}(n) = x(nM + \ell). \tag{6.10}$$

Figure 6.5 shows an example of a type-1 decomposition.

Type-2. The decomposition into type-2 polyphase components is given by

$$X(z) = \sum_{\ell=0}^{M-1} z^{-(M-1-\ell)} X'_{\ell}(z^M), \tag{6.11}$$

where

$$X'_{\ell}(z) \longleftrightarrow x'_{\ell}(n) = x(nM + M - 1 - \ell). \tag{6.12}$$

Thus, the only difference between a type-1 and a type-2 decomposition lies in the indexing:

$$X_\ell(z) = X'_{M-1-\ell}(z). \quad (6.13)$$

Type-3. A type-3 decomposition reads

$$X(z) = \sum_{\ell=0}^{M-1} z^\ell \bar{X}_\ell(z^M), \quad (6.14)$$

where

$$\bar{X}_\ell(z) \longleftrightarrow \bar{x}_\ell(n) = x(nM - \ell). \quad (6.15)$$

The relation to the type-1 polyphase components is

$$\begin{aligned} X_0(z) &= \bar{X}_0(z), \\ X_\ell(z) &= z^{-1} \bar{X}_{M-\ell}(z), \quad \ell = 1, \dots, M-1. \end{aligned} \quad (6.16)$$

Polyphase decompositions are frequently used for both signals and filters. In the latter case we use the notation $H_{ik}(z)$ for the k th type-1 polyphase component of filter $H_i(z)$. The definitions for type-2 and type-3 components are analogous.

6.2 Two-Channel Filter Banks

6.2.1 PR Condition

Let us consider the two-channel filter bank in Figure 6.6. The signals are related as

$$\begin{aligned} Y_0(z^2) &= \frac{1}{2} [H_0(z) X(z) + H_0(-z) X(-z)], \\ Y_1(z^2) &= \frac{1}{2} [H_1(z) X(z) + H_1(-z) X(-z)], \\ \hat{X}(z) &= [Y_0(z^2) G_0(z) + Y_1(z^2) G_1(z)]. \end{aligned} \quad (6.17)$$

Combining these equations yields the input-output relation

$$\begin{aligned} \hat{X}(z) &= \frac{1}{2} [H_0(z) G_0(z) + H_1(z) G_1(z)] X(z) \\ &\quad + \frac{1}{2} [H_0(-z) G_0(z) + H_1(-z) G_1(z)] X(-z). \end{aligned} \quad (6.18)$$

The first term describes the transmission of the signal $X(z)$ through the system, while the second term describes the aliasing component at the output

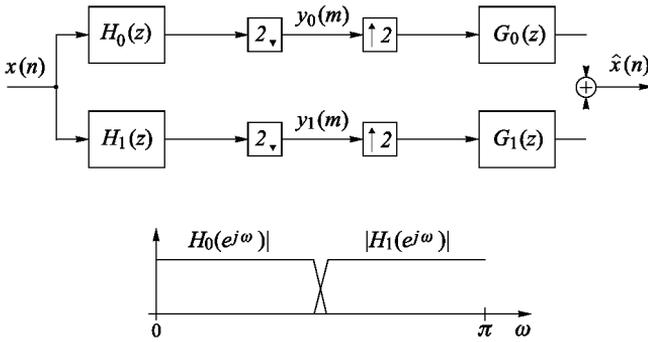


Figure 6.6. Two-channel filter bank.

of the filter bank. Perfect reconstruction is given if the output signal is nothing but a delayed version of the input signal. That is, the transfer function for the signal component, denoted as $S(z)$, must satisfy

$$S(z) = H_0(z) G_0(z) + H_1(z) G_1(z) = 2 z^{-a}, \tag{6.19}$$

and the transfer function $F(z)$ for the aliasing component must be zero:

$$F(z) = H_0(-z) G_0(z) + H_1(-z) G_1(z) = 0. \tag{6.20}$$

If (6.20) is satisfied, the output signal contains no aliasing, but amplitude distortions may be present. If both (6.19) and (6.20) are satisfied, the amplitude distortions also vanish. Critically subsampled filter banks that allow perfect reconstruction are also known as biorthogonal filter banks. Several methods for satisfying these conditions either exactly or approximately can be found in the literature. The following sections give a brief overview.

6.2.2 Quadrature Mirror Filters

Quadrature mirror filter banks (QMF banks) provide complete aliasing cancellation at the output, but condition (6.19) is only approximately satisfied. The principle was introduced by Esteban and Galand in [52]. In QMF banks, $H_0(z)$ is chosen as a linear phase lowpass filter, and the remaining filters are constructed as

$$\begin{aligned} G_0(z) &= H_0(z) \\ H_1(z) &= H_0(-z) \\ G_1(z) &= -H_1(z). \end{aligned} \tag{6.21}$$

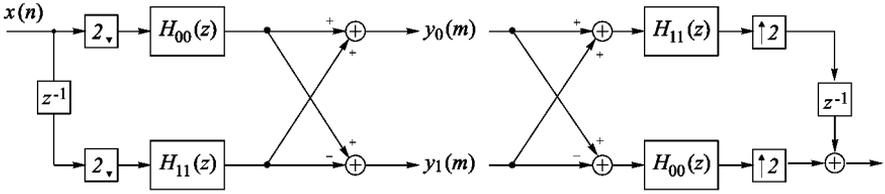


Figure 6.7. QMF bank in polyphase structure.

As can easily be verified, independent of the filter $H_0(z)$, the condition $F(z) = 0$ is structurally satisfied, so that one only has to ensure that $S(z) = H_0^2(z) + H_0^2(-z) \approx 2z^{-q}$. The name QMF is due to the mirror image property

$$|H_1(e^{j\frac{\pi}{2} - \omega})| = |H_0(e^{j\frac{\pi}{2} + \omega})|$$

with symmetry around $\pi/2$.

QMF bank prototypes with good coding properties have for instance been designed by Johnston [78].

One important property of the QMF banks is their efficient implementation due to the modulated structure, where the highpass and lowpass filters are related as $H_1(z) = H_0(-z)$. For the polyphase components this means that $H_{10}(z) = H_{00}(z)$ and $H_{11}(z) = -H_{01}(z)$. The resulting efficient polyphase realization is depicted in Figure 6.7.

6.2.3 General Perfect Reconstruction Two-Channel Filter Banks

A method for the construction of PR filter banks is to choose

$$\begin{aligned} G_0(z) &= z^{-\ell} H_1(-z) \\ G_1(z) &= -z^{-\ell} H_0(-z). \end{aligned} \tag{6.22}$$

It is easily verified that (6.20) is satisfied. Inserting the above relationships into (6.19) yields

$$H_0(z)G_0(z) + (-1)^{\ell+1} H_0(-z)G_0(-z) = 2z^{-q}. \tag{6.23}$$

Using the abbreviation

$$T(z) = G_0(z) H_0(z), \tag{6.24}$$

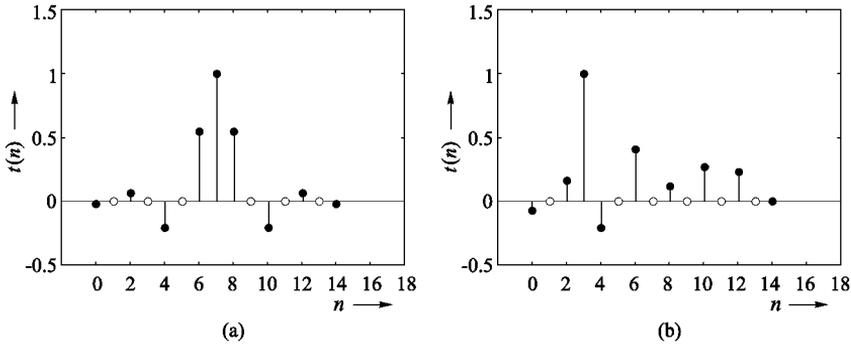


Figure 6.8. Examples of Nyquist filters $T(z)$; (a) linear-phase; (b) short overall delay.

this becomes

$$2z^{-q} = T(z) + (-1)^{\ell+1} T(-z). \quad (6.25)$$

Note that $\frac{1}{2}[T(z) + T(-z)]$ is the z -transform of a sequence that only has non-zero even taps, while $\frac{1}{2}[T(z) - T(-z)]$ is the z -transform of a sequence that only has non-zero odd taps. Altogether we can say that in order to satisfy (6.25), the system $T(z)$ has to satisfy

$$t(n) = \begin{cases} 1 & n = q \\ 0 & n = q + 2\ell, \ell \neq 0 \\ \text{arbitrary} & n = q + 2\ell + 1. \end{cases} \quad \ell \in \mathbb{Z} \quad (6.26)$$

In communications, condition (6.26) is known as the *first Nyquist condition*. Examples of impulse responses $t(n)$ satisfying the first Nyquist condition are depicted in Figure 6.8. The arbitrary taps are the free design parameters, which may be chosen in order to achieve good filter properties. Thus, filters can easily be designed by choosing a filter $T(z)$ and factoring it into $H_0(z)$ and $G_0(z)$. This can be done by computing the roots of $T(z)$ and dividing them into two groups, which form the zeros of $H_0(z)$ and $G_0(z)$. The remaining filters are then chosen according to (6.24) in order to yield a PR filter bank. This design method is known as *spectral factorization*.

6.2.4 Matrix Representations

Matrix representations are a convenient and compact way of describing and characterizing filter banks. In the following we will give a brief overview of the most important matrices and their relation to the analysis and synthesis filters.

Modulation Matrix. The input-output relations of the two-channel filter bank may also be written in matrix form. For this, we introduce the vectors

$$\mathbf{x}_m(z) = \begin{bmatrix} X(z) \\ X(-z) \end{bmatrix}, \quad (6.27)$$

$$\mathbf{y}_p(z) = \begin{bmatrix} Y_0(z) \\ Y_1(z) \end{bmatrix} \quad (6.28)$$

and the so-called *modulation matrix* or *alias component (AC) matrix*

$$\mathbf{H}_m(z) = \begin{bmatrix} H_0(z) & H_1(z) \\ H_0(-z) & H_1(-z) \end{bmatrix}, \quad (6.29)$$

which contains the filters $H_0(z)$ and $H_1(z)$ and their modulated versions $H_0(-z)$ and $H_1(-z)$. We get

$$\mathbf{y}_p(z^2) = \frac{1}{2} \mathbf{H}_m^T(z) \mathbf{x}_m(z), \quad (6.30)$$

$$\begin{aligned} \hat{X}(z) &= [G_0(z), G_1(z)] \mathbf{y}_p(z^2) \\ &= \frac{1}{2} [G_0(z), G_1(z)] \mathbf{H}_m^T(z) \mathbf{x}_m(z). \end{aligned} \quad (6.31)$$

Polyphase Representation of the Analysis Filter Bank. Let us consider the analysis filter bank in Figure 6.9(a). The signals $y_0(m)$ and $y_1(m)$ may be written as

$$\begin{aligned} y_0(m) &= \sum_n h_0(n) x(2m - n) \\ &= \sum_k h_0(2k) x(2m - 2k) + \sum_k h_0(2k + 1) x(2m - 2k - 1) \\ &= \sum_k h_{00}(k) \bar{x}_0(m - k) + \sum_k h_{01}(k) \bar{x}_1(m - k) \end{aligned} \quad (6.32)$$

and

$$\begin{aligned} y_1(m) &= \sum_n h_1(n) x(2m - n) \\ &= \sum_k h_{10}(k) \bar{x}_0(m - k) + \sum_k h_{11}(k) \bar{x}_1(m - k), \end{aligned} \quad (6.33)$$

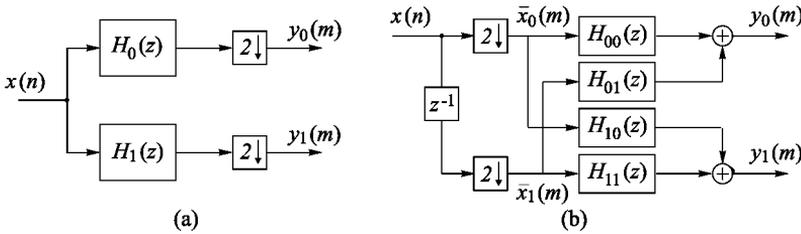


Figure 6.9. Analysis filter bank. (a) direct implementation; (b) polyphase realization.

where we used the following polyphase components:

$$\begin{aligned}
 h_{00}(k) &= h_0(2k), \\
 h_{01}(k) &= h_0(2k + 1), \\
 h_{10}(k) &= h_1(2k), \\
 h_{11}(k) &= h_1(2k + 1), \\
 \bar{x}_0(k) &= x(2k), \\
 \bar{x}_1(k) &= x(2k - 1).
 \end{aligned}$$

The last rows of (6.32), and (6.33) respectively, show that the complete analysis filter bank can be realized by operating solely with the polyphase components, as depicted in Figure 6.9(b). The advantage of the polyphase realization compared to the direct implementation in Figure 6.9(a) is that only the required output values are computed. When looking at the first rows of (6.32) and (6.33) this sounds trivial, because these equations are easily implemented and do not produce unneeded values. Thus, unlike in the QMF bank case, the polyphase realization does not necessarily lead to computational savings compared to a proper direct implementation of the analysis equations. However, it allows simple filter design, gives more insight into the properties of a filter bank, and leads to efficient implementations based on lattice structures; see Sections 6.2.6 and 6.2.7.

It is convenient to describe (6.32) and (6.33) in the z -domain using matrix notation:

$$\mathbf{y}_p(z) = \mathbf{E}(z) \bar{\mathbf{x}}_p(z), \tag{6.34}$$

where

$$\mathbf{E}(z) = \begin{bmatrix} H_{00}(z) & H_{01}(z) \\ H_{10}(z) & H_{11}(z) \end{bmatrix}, \tag{6.35}$$

$$\bar{\mathbf{x}}_p(z) = \begin{bmatrix} \bar{X}_0(z) \\ \bar{X}_1(z) \end{bmatrix}. \tag{6.36}$$

Matrix $\mathbf{E}(z)$ is called the *polyphase matrix* of the analysis filter bank. As can easily be seen by inspection, it is related to the modulation matrix as follows:

$$\mathbf{H}_m(z) = \mathbf{W}^H \mathbf{D}(z) \mathbf{E}^T(z^2), \quad (6.37)$$

$$\mathbf{E}(z^2) = \frac{1}{2} \mathbf{H}_m^T(z) \mathbf{W} \mathbf{D}(z^{-1}),$$

with

$$\mathbf{W} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \quad (6.38)$$

and

$$\mathbf{D}(z) = \begin{bmatrix} 1 & \\ & z^{-1} \end{bmatrix} \quad (6.39)$$

Here, \mathbf{W} is understood as the 2×2 -DFT matrix. In view of the general M -channel case, we use the notation $\mathbf{W}^{-1} = \frac{1}{2} \mathbf{W}^H$ for the inverse.

Polyphase Representation of the Synthesis Filter Bank. We consider the synthesis filter bank in Figure 6.10(a). The filters $G_0(z)$ and $G_1(z)$ can be written in terms of their type-2 polyphase components as

$$G_0(z) = z^{-1} G'_{00}(z^2) + G'_{01}(z^2) \quad (6.40)$$

and

$$G_1(z) = z^{-1} G'_{10}(z^2) + G'_{11}(z^2). \quad (6.41)$$

This gives rise to the following z -domain matrix representation:

$$\hat{\mathbf{X}}(z) = [z^{-1} \ 1] \underbrace{\begin{bmatrix} G'_{00}(z) & G'_{10}(z) \\ G'_{01}(z) & G'_{11}(z) \end{bmatrix}}_{\mathbf{R}(z)} \begin{bmatrix} Y_0(z) \\ Y_1(z) \end{bmatrix}. \quad (6.42)$$

The corresponding polyphase realization is depicted in Figure 6.10. Perfect reconstruction up to an overall delay of $q = 2m_0 + 1$ samples is achieved if

$$\mathbf{R}(z) \mathbf{E}(z) = z^{-m_0} \mathbf{I}. \quad (6.43)$$

The PR condition for an even overall delay of $q = 2m_0$ samples is

$$\mathbf{R}(z) \mathbf{E}(z) = z^{-m_0} \begin{bmatrix} 0 & 1 \\ z^{-1} & 0 \end{bmatrix}. \quad (6.44)$$

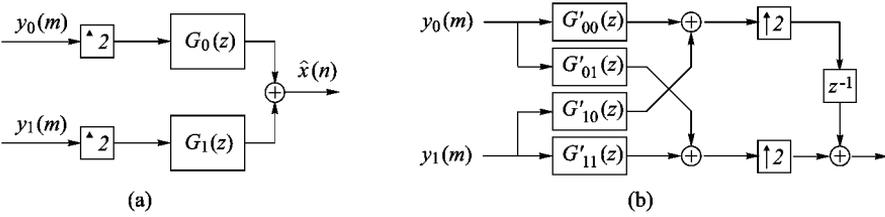


Figure 6.10. Synthesis filter bank. (a) direct implementation; (b) polyphase realization.

6.2.5 Paraunitary Two-Channel Filter Banks

The inverse of a unitary matrix is given by the Hermitian transpose. A similar property can be stated for polyphase matrices as follows:

$$\mathbf{E}^{-1}(z) = \tilde{\mathbf{E}}(z), \tag{6.45}$$

where

$$\tilde{\mathbf{E}}(z) = (\mathbf{E}(z))^H, \quad |z| = 1, \tag{6.46}$$

such that

$$\mathbf{E}(z) \tilde{\mathbf{E}}(z) = \tilde{\mathbf{E}}(z) \mathbf{E}(z) = \mathbf{I}. \tag{6.47}$$

Analogous to ordinary matrices, $(\mathbf{E}(z))^H$ stands for transposing the matrix and simultaneously conjugating the elements:

$$\mathbf{E}(z) = \begin{bmatrix} H_{00}(z) & H_{01}(z) \\ H_{10}(z) & H_{11}(z) \end{bmatrix} \Rightarrow \tilde{\mathbf{E}}(z) = \begin{bmatrix} \tilde{H}_{00}(z) & \tilde{H}_{10}(z) \\ \tilde{H}_{01}(z) & \tilde{H}_{11}(z) \end{bmatrix}.$$

In the case of real-valued filter coefficients we have $\tilde{H}_{ik}(z) = H_{ik}(z^{-1})$, such that $\tilde{\mathbf{E}}(z) = \mathbf{E}^T(z^{-1})$ and

$$\mathbf{E}(z) \mathbf{E}^T(z^{-1}) = \mathbf{E}^T(z^{-1}) \mathbf{E}(z) = \mathbf{I}. \tag{6.48}$$

Since $\mathbf{E}(z)$ is dependent on z , and since the operation (6.46) has to be carried out on the unit circle, and not at some arbitrary point in the z plane, a matrix $\mathbf{E}(z)$ satisfying (6.47) is said to be paraunitary.

Modulation Matrices. As can be seen from (6.37) and (6.47), we have

$$\mathbf{H}_m(z) \tilde{\mathbf{H}}_m(z) = \tilde{\mathbf{H}}_m(z) \mathbf{H}_m(z) = 2 \mathbf{I} \tag{6.49}$$

for the modulation matrices of paraunitary two-channel filter banks.

Matched Filter Condition. From (6.49) we may conclude that the analysis and synthesis filters in a paraunitary two-channel filter bank are related as

$$G_k(z) = \tilde{H}_k(z) \iff g_k(n) = h_k^*(-n), \quad k = 0, 1. \tag{6.50}$$

This means that an analysis filter and its corresponding synthesis filter together yield a Nyquist filter (cf. (6.24)) whose impulse response is equivalent to the autocorrelation sequence of the filters in question:

$$t_k(n) = r_{h_k h_k}^E(n) = r_{g_k g_k}^E(n) = h_k(n) * h_k^*(-n). \quad (6.51)$$

Here we find parallels to data transmission, where the receiver input filter is matched to the output filter of the transmitter such that the overall result is the autocorrelation sequence of the filter. This is known as the *matched-filter condition*. The reason for choosing this special input filter is that it yields a maximum signal-to-noise ratio if additive white noise interferes on the transmission channel.

Power-Complementary Filters. From (6.49) we conclude

$$2 = H_0(z)\tilde{H}_0(z) + H_0(-z)\tilde{H}_0(-z), \quad (6.52)$$

which for $z = e^{j\omega}$ implies the requirement

$$2 = |H_0(e^{j\omega})|^2 + |H_0(e^{j(\omega+\pi)})|^2. \quad (6.53)$$

We observe that the filters $H_0(e^{j\omega})$ and $H_0(e^{j(\omega+\pi)})$ must be power-complementary to one another. For constructing paraunitary filter banks we therefore have to find a Nyquist filter $T(z)$ which can be factored into

$$T(z) = H_0(z)\tilde{H}_0(z). \quad (6.54)$$

Note that a factorization is possible only if $T(e^{j\omega})$ is real and positive. A filter that satisfies this condition is said to be *valid*. Since $T(e^{j\omega})$ has symmetry around $\omega = \pi/2$ such a filter is also called a *valid halfband filter*. This approach was introduced by Smith and Barnwell in [135].

Given Prototype. Given an FIR prototype $H(z)$ that satisfies condition (6.53), the required analysis and synthesis filters can be derived as

$$\begin{aligned} H_0(z) &= H(z) \\ H_1(z) &= z^{-(L-1)}\tilde{H}(-z) \\ G_0(z) &= z^{-(L-1)}\tilde{H}_0(z) \\ G_1(z) &= z^{-(L-1)}\tilde{H}_1(z). \end{aligned} \quad (6.55)$$

Here, L is the number of coefficients of the prototype.

Number of Coefficients. Prototypes for paraunitary two-channel filter banks have even length. This is seen by formulating (6.52) in the time domain and assuming an FIR filter with coefficients $h_0(0), \dots, h_0(2k)$:

$$\delta_{\ell 0} = \sum_{n=0}^{2k} h_0(n) h_0^*(n - 2\ell). \quad (6.56)$$

For $\ell = k$, $n = 2k$, $k \neq 0$, this yields the requirement $0 = h_0(2k) h_0^*(0)$, which for $h_0(0) \neq 0$ can only be satisfied by $h_0(2k) = 0$. This means that the filter has to have even length.

Filter Energies. It is easily verified that all filters in a paraunitary filter bank have energy one:

$$\|\mathbf{h}_0\|_{\ell_2}^2 = \|\mathbf{h}_1\|_{\ell_2}^2 = \|\mathbf{g}_0\|_{\ell_2}^2 = \|\mathbf{g}_1\|_{\ell_2}^2 = 1. \quad (6.57)$$

Non-Linear Phase Property. We will show that paraunitary two-channel filter banks are non-linear phase with one exception. The following proof is based on Vaidyanathan [145]. We assume that two filters $H(z)$ and $G(z)$ are power-complementary and linear-phase:

$$\left. \begin{aligned} c^2 &= H(z)\tilde{H}(z) + G(z)\tilde{G}(z) \\ \tilde{H}(z) &= e^{j\alpha} z^L H(z), \quad \alpha \in \mathbb{R} \\ \tilde{G}(z) &= e^{j\beta} z^L G(z), \quad \beta \in \mathbb{R} \end{aligned} \right\} \text{(linear-phase property).} \quad (6.58)$$

We conclude

$$(H(z)e^{j\alpha/2} + jG(z)e^{j\beta/2}) (H(z)e^{j\alpha/2} - jG(z)e^{j\beta/2}) = c^2 z^{-L}. \quad (6.59)$$

Both factors on the left are FIR filters, so that

$$\left. \begin{aligned} H(z)e^{j\alpha/2} + jG(z)e^{j\beta/2} &= pz^{-L_1} \\ H(z)e^{j\alpha/2} - jG(z)e^{j\beta/2} &= qz^{-L_2} \end{aligned} \right\} L_1 + L_2 = L, \quad pq = c^2 \quad (6.60)$$

Adding and subtracting both equations shows that $H(z)$ and $G(z)$ must have the form

$$\begin{aligned} H(z) &= az^{-L_1} + bz^{-L_2}, \\ G(z) &= \gamma(az^{-L_1} - bz^{-L_2}), \quad |\gamma| = 1 \end{aligned} \quad (6.61)$$

in order to be both power-complementary and linear-phase. In other words, power-complementary linear-phase filters cannot have more than two coefficients.

6.2.6 Paraunitary Filter Banks in Lattice Structure

Paraunitary filter banks can be efficiently implemented in a lattice structure [53], [147]. For this, we decompose the polyphase matrix $E(z)$ as follows:

$$E(z) = B_{N-1}D(z)B_{N-2} \cdots D(z)B_0. \quad (6.62)$$

Here, the matrices B_k , $k = 0, \dots, N-1$ are rotation matrices:

$$B_k = \begin{bmatrix} \cos \beta_k & \sin \beta_k \\ -\sin \beta_k & \cos \beta_k \end{bmatrix}, \quad k = 0, \dots, N-1, \quad (6.63)$$

and $D(z)$ is the delay matrix

$$D = \begin{bmatrix} 1 & 0 \\ 0 & z^{-1} \end{bmatrix}. \quad (6.64)$$

It can be shown that such a decomposition is always possible [146].

Provided $\cos \beta_k \neq 0$, $k = 0, 1, \dots, N-1$, we can also write

$$E(z) = c A_{N-1}D(z)A_{N-2} \cdots D(z)A_0 \quad (6.65)$$

with

$$A_k = \begin{bmatrix} 1 & \alpha_k \\ -\alpha_k & 1 \end{bmatrix}, \quad c = \prod_{k=0}^{N-1} \frac{1}{\sqrt{1 + \alpha_k^2}}. \quad (6.66)$$

This basically allows us to reduce the total number of multiplications. The realization of the filter bank by means of the decomposed polyphase matrix is pictured in Figure 6.11(a). Given α_k , $k = 0, \dots, N-1$, we obtain filters of length $L = 2N$.

Since this lattice structure leads to a paraunitary filter bank for arbitrary α_k , we can thus achieve perfect reconstruction even if the coefficients must be quantized due to finite precision. In addition, this structure may be used for optimizing the filters. For this, we excite the filter bank with $x_{\text{even}}(n) = \delta_{n0}$ and $x_{\text{odd}}(n) = \delta_{n1}$ and observe the polyphase components of $H_0(z)$ and $H_1(z)$ at the output.

The polyphase matrix of the synthesis filter bank has the following factorization:

$$R(z) = B_0^T D'(z)B_1^T \cdots D'(z)B_{N-1}^T \quad (6.67)$$

with $D'(z) = JD(z)J$, such that $D'(z)D(z) = z^{-1}I$. This means that all rotations are inverted and additional delay is introduced. The implementation is shown in Figure 6.11(b).

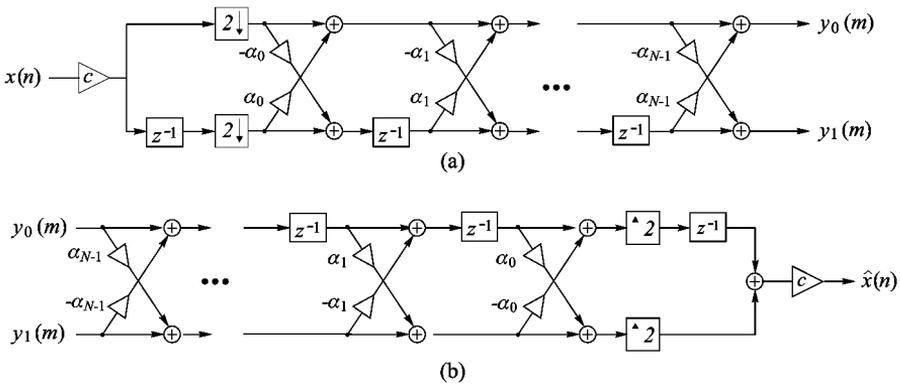


Figure 6.11. Paraunitary filter bank in lattice structure; (a) analysis; (b) synthesis.

6.2.7 Linear-Phase Filter Banks in Lattice Structure

Linear-phase PR two-channel filter banks can be designed and implemented in various ways. Since the filters do not have to be power-complementary, we have much more design freedom than in the paraunitary case. For example, any factorization of a Nyquist filter into two linear-phase filters is possible. A Nyquist filter with $P = 6$ zeros can for instance be factored into two linear-phase filters each of which has three zeros, or into one filter with four and one filter with two zeros. However, realizing the filters in lattice structure, as will be discussed in the following, involves the restriction that the number of coefficients must be even and equal for all filters.

The following factorization of $E(z)$ is used [146]:

$$E(z) = L_{N-1}D(z)L_{N-2} \cdots D(z)L_0 \tag{6.68}$$

with

$$L_k = \begin{bmatrix} 1 & \alpha_k \\ \alpha_k & 1 \end{bmatrix}, \quad L_{N-1} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \quad D = \begin{bmatrix} 1 & 0 \\ 0 & z^{-1} \end{bmatrix}, \quad k = 0, \dots, N-2.$$

It results in a linear-phase PR filter bank. The realization of the filter bank with the decomposed polyphase matrix is depicted in Figure 6.12. As in the case of paraunitary filter banks in Section 6.2.6, we can achieve PR if the coefficients must be quantized because of finite-precision arithmetic. In addition, the structure is suitable for optimizing filter banks with respect to given criteria while conditions such as linear-phase and PR are structurally guaranteed. The number of filter coefficients is $L = 2(N + 1)$ and thus even in any case.

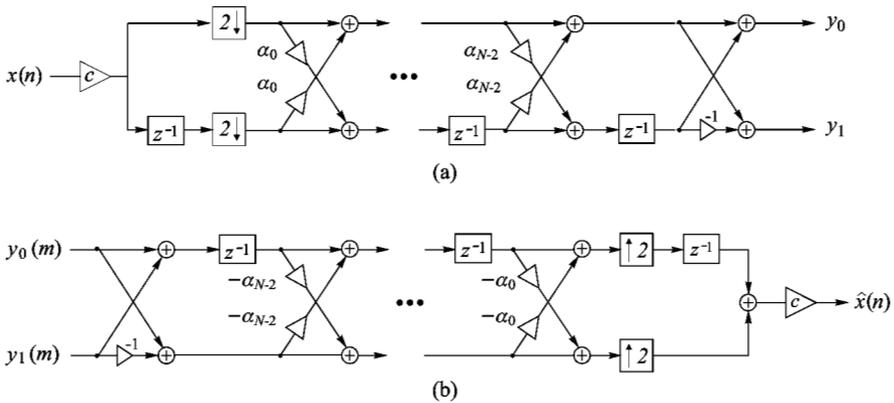


Figure 6.12. Linear-phase filter bank in lattice structure; (a) analysis; (b) synthesis.

6.2.8 Lifting Structures

Lifting structures have been suggested in [71, 141] for the design of biorthogonal wavelets. In order to explain the discrete-time filter bank concept behind lifting, we consider the two-channel filter bank in Figure 6.13(a). The structure obviously yields perfect reconstruction with a delay of one sample. Now we incorporate a system $A(z)$ and a delay z^{-a} , $a \geq 0$ in the polyphase domain as shown in Figure 6.13(b). Clearly, the overall structure still gives PR, while the new subband signal $y_0(m)$ is different from the one in Figure 6.13(a). In fact, the new $y_0(m)$ results from filtering $x(n)$ with the filter

$$H_0(z) = z^{-2a} + z^{-1}A(z^2)$$

and subsampling. The overall delay has increased by $2a$. In the next step in Figure 6.13(c), we use a dual lifting step that allows us to construct a new (longer) filter $H_1(z)$ as

$$H_1(z) = z^{-2b-1} + z^{-2a}B(z^2) + z^{-1}A(z^2)B(z^2).$$

Now the overall delay is $2a + 2b + 1$ with $a, b \geq 0$. Note that, although we may already have relatively long filters $H_0(z)$ and $H_1(z)$, the delay may be unchanged if we have chosen $a = b = 0$. This technique allows us to design PR filter banks with high stopband attenuation and low overall delay. Such filters are for example very attractive for real-time communications systems, where the overall delay has to be kept below a given threshold.

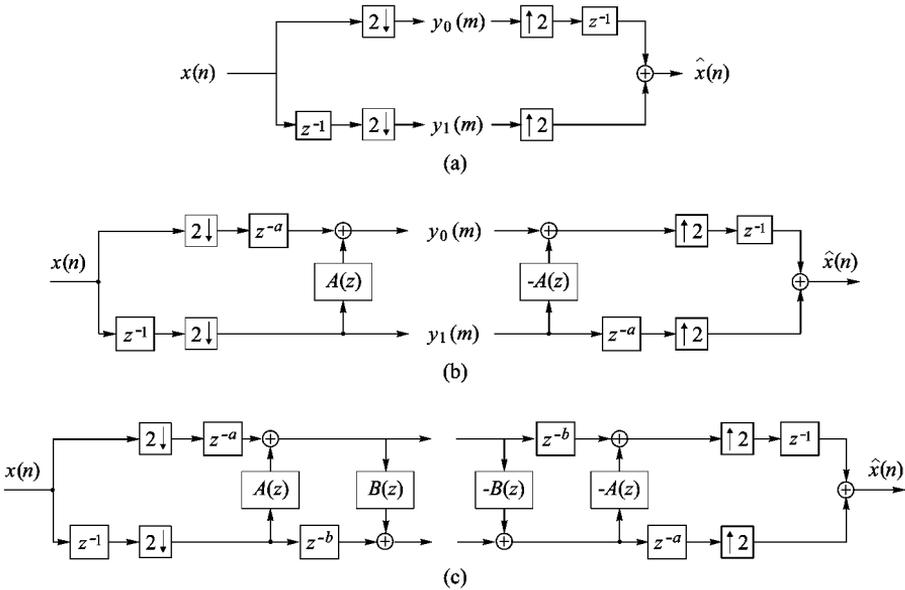


Figure 6.13. Two-channel filter banks in lifting structure.

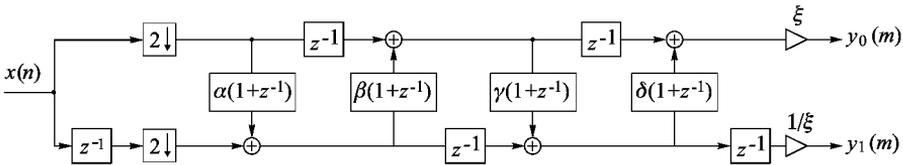


Figure 6.14. Lifting implementation of the 9-7 filters from [5] according to [37]. The parameters are $\alpha = -1.586134342$, $\beta = -0.05298011854$, $\gamma = 0.8829110762$, $\delta = 0.4435068522$, $\xi = 1.149604398$.

In general, the filters constructed via lifting are non-linear phase. However, the lifting steps can easily be chosen to yield linear-phase filters.

Both lattice and lifting structures are very attractive for the implementation of filter banks on digital signal processors, because coefficient quantization does not affect the PR property. Moreover, due to the joint realization of $H_0(z)$ and $H_1(z)$, the total number of operations is lower than for the direct polyphase implementation of the same filters. To give an example, Figure 6.14 shows the lifting implementation of the 9-7 filters from [5], which are very popular in image compression.

An important result is that any two-channel filter bank can be factored into a finite number of lifting steps [37]. The proof is based on the Euclidean algorithm [9]. The decomposition of a given filter bank into lifting steps is not unique, so that many implementations for the same filter bank can be found. Unfortunately, one cannot say *a priori* which implementation will perform best if the coefficients have to be quantized to a given number of bits.

6.3 Tree-Structured Filter Banks

In most applications one needs a signal decomposition into more than two, say M , frequency bands. A simple way of designing the required filters is to build cascades of two-channel filter banks. Figure 6.15 shows two examples, (a) a regular tree structure and (b) an octave-band tree structure. Further structures are easily found, and also signal-adaptive concepts have been developed, where the tree is chosen such that it is best matched to the problem. In all cases, PR is easily obtained if the two-channel filter banks, which are used as the basic building blocks, provide PR.

In order to describe the system functions of cascaded filters with sampling rate changes, we consider the two systems in Figure 6.16. It is easily seen that both systems are equivalent. Their system function is

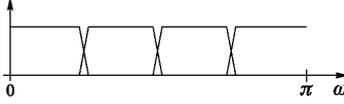
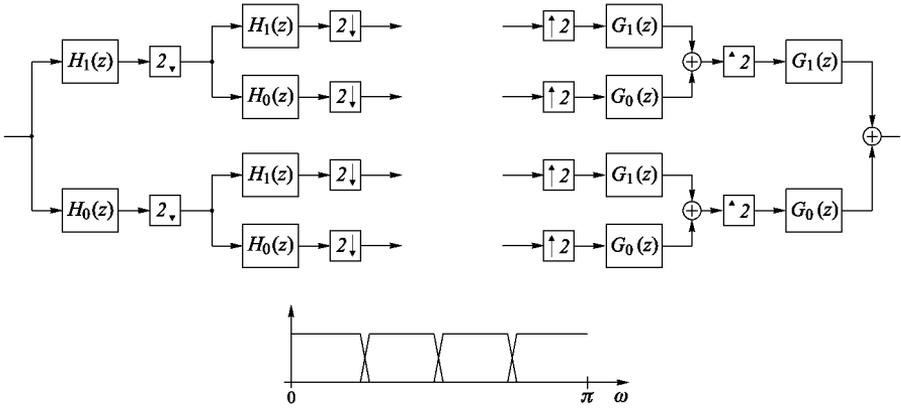
$$\frac{1}{2} H_1(z^2) [H_0(z) + H_0(-z)].$$

For the system $B_2(z^2)$ we have

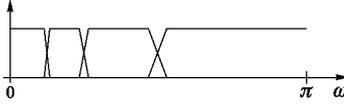
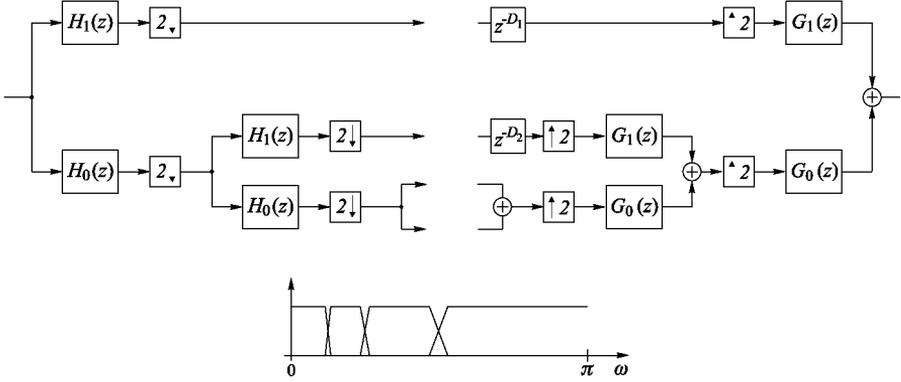
$$B_2(z^2) = H_0(z)H_1(z^2).$$

With this result, the system functions of arbitrary cascades of two-channel filter banks are easily obtained.

An example of the frequency responses of non-ideal octave-band filter banks in tree structure is shown in Figure 6.17. An effect, which results from the overlap of the lowpass and highpass frequency responses, is the occurrence of relatively large side lobes.



(a)



(b)

Figure 6.15. Tree-structured filter banks; (a) regular tree structure; (b) octave-band tree structure.

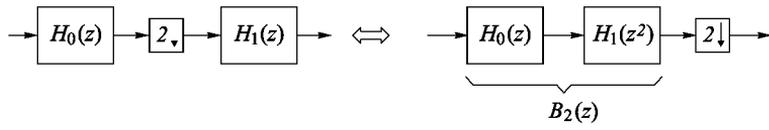


Figure 6.16. Equivalent systems.

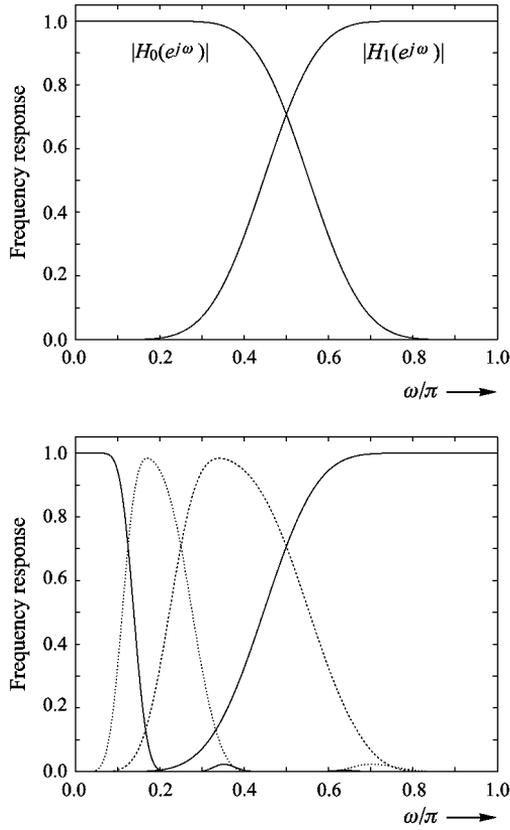


Figure 6.17. Frequency responses in tree-structured filter banks; (a) two-channel filter bank; (b) octave-band filter bank.

6.4 Uniform M -Channel Filter Banks

This section addresses uniform M -channel filter banks for which the sampling rate is reduced by N in all subbands. Figure 6.1 shows such a filter bank, and Figure 6.18 shows some frequency responses. In order to obtain general results for uniform M -channel filter banks, we start by assuming $N \leq M$, where M is the number of subbands.

6.4.1 Input-Output Relations

We consider the multirate filter bank depicted in Figure 6.1. From equations (6.7) and (6.8) we obtain

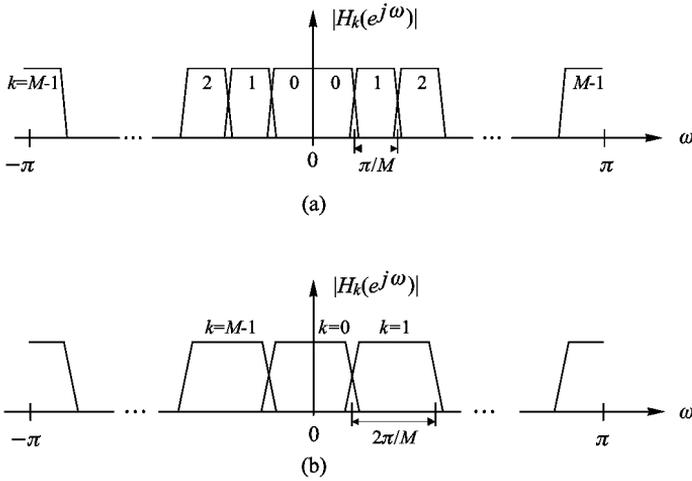


Figure 6.18. Frequency responses of the analysis filters in a uniform M -channel filter bank; (a) cosine-modulated filter bank; (b) DFT filter bank.

$$Y_k(z) = \frac{1}{N} \sum_{i=0}^{N-1} H_k(W_N^i z^{\frac{1}{N}}) X(W_N^i z^{\frac{1}{N}}), \quad k = 0, \dots, M-1, \quad (6.69)$$

and

$$\hat{X}(z) = \frac{1}{N} \sum_{k=0}^{M-1} \sum_{i=0}^{N-1} G_k(z) H_k(W_N^i z) X(W_N^i z). \quad (6.70)$$

In order to achieve perfect reconstruction, suitable filters $H_k(z)$ and $G_k(z)$, $k = 0, \dots, M-1$, and parameters N and M must be chosen. We obtain the PR requirement by first changing the order of the summation in (6.70):

$$\hat{X}(z) = \frac{1}{N} \sum_{i=0}^{N-1} X(W_N^i z) \sum_{k=0}^{M-1} G_k(z) H_k(W_N^i z). \quad (6.71)$$

Equation (6.71) shows that $\hat{X}(z) = z^{-q} X(z)$ holds if the filters satisfy

$$\sum_{k=0}^{M-1} G_k(z) H_k(W_N^i z) = N z^{-q} \delta_{i0}, \quad 0 \leq i \leq N-1. \quad (6.72)$$

Using the notation

$$\mathbf{H}_m(z) = \begin{bmatrix} H_0(z) & H_1(z) & \cdots & H_{M-1}(z) \\ H_0(zW_N) & H_1(zW_N) & \cdots & H_{M-1}(zW_N) \\ \vdots & \vdots & \ddots & \vdots \\ H_0(zW_N^{N-1}) & H_1(zW_N^{N-1}) & \cdots & H_{M-1}(zW_N^{N-1}) \end{bmatrix}, \quad (6.73)$$

and

$$\mathbf{g}(z) = [G_0(z), G_1(z), \dots, G_{M-1}(z)]^T, \quad (6.74)$$

$$\mathbf{x}_m(z) = [X(z), X(zW_N), \dots, X(zW_N^{M-1})]^T, \quad (6.75)$$

the input-output relations may also be written as

$$\hat{X}(z) = \frac{1}{N} \mathbf{g}^T(z) \mathbf{H}_m^T(z) \mathbf{x}_m(z). \quad (6.76)$$

Thus, PR requires that

$$\frac{1}{N} \mathbf{g}^T(z) \mathbf{H}_m^T(z) = z^{-q} [1, 0, \dots, 0]. \quad (6.77)$$

6.4.2 The Polyphase Representation

In Section 6.2 we explained the polyphase representation of two-channel filter banks. The generalization to M channels with subsampling by N is outlined below. The implementation of such a filter bank is depicted in Figure 6.19.

Analysis. The analysis filter bank is described by

$$\mathbf{y}_p(z) = \mathbf{E}(z) \mathbf{x}_p(z), \quad (6.78)$$

where

$$\mathbf{x}_p(z) = [\bar{X}_0(z), \bar{X}_1(z), \dots, \bar{X}_{N-1}(z)]^T \quad (6.79)$$

$$\mathbf{y}_p(z) = [Y_0(z), Y_1(z), \dots, Y_{M-1}(z)]^T \quad (6.80)$$

$$\mathbf{E}(z) = \begin{bmatrix} H_{00}(z) & H_{01}(z) & \cdots & H_{0,N-1}(z) \\ H_{10}(z) & H_{11}(z) & \cdots & H_{1,N-1}(z) \\ \vdots & \vdots & \ddots & \vdots \\ H_{M-1,0}(z) & H_{M-1,1}(z) & \cdots & H_{M-1,N-1}(z) \end{bmatrix} \quad (6.81)$$

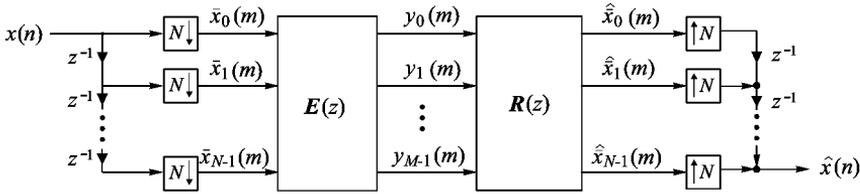


Figure 6.19. Filter bank in polyphase structure.

Synthesis. Synthesis may be described in a similar way:

$$\hat{\mathbf{x}}_p(z) = \mathbf{R}(z) \mathbf{y}_p(z) \tag{6.82}$$

with

$$\mathbf{R}(z) = \begin{bmatrix} G'_{00}(z) & G'_{10}(z) & \cdots & G'_{M-1,0}(z) \\ G'_{01}(z) & G'_{11}(z) & \cdots & G'_{M-1,1}(z) \\ \vdots & \vdots & \ddots & \vdots \\ G'_{0,N-1}(z) & G'_{1,N-1}(z) & \cdots & G'_{M-1,N-1}(z) \end{bmatrix} \tag{6.83}$$

Perfect Reconstruction. From (6.78) and (6.82) we conclude the PR requirement

$$\mathbf{R}(z) \mathbf{E}(z) = z^{-q_0} \mathbf{I}, \tag{6.84}$$

which results in an overall delay of $Mq_0 + M - 1$ samples. The generalization to any arbitrary delay of $Mq_0 + r + M - 1$ samples is

$$\mathbf{R}(z) \mathbf{E}(z) = z^{-q_0} \begin{bmatrix} \mathbf{0} & \mathbf{I}_{M-r} \\ z^{-1} \mathbf{I}_r & \mathbf{0} \end{bmatrix}, \tag{6.85}$$

where $0 \leq r \leq M - 1$ [146].

FIR Filter Banks. Let us write (6.84) as

$$\mathbf{R}(z) = z^{-q_0} \mathbf{E}^{-1}(z) = z^{-q_0} \frac{\text{Adj}\{\mathbf{E}(z)\}}{\det\{\mathbf{E}(z)\}} \tag{6.86}$$

and let us assume that all elements of $\mathbf{E}(z)$ are FIR. We see that the elements of $\mathbf{R}(z)$ are also FIR if $\det\{\mathbf{E}(z)\}$ is a monomial in z . The same arguments hold for the more general PR condition (6.85). Thus, FIR solutions for both the analysis and synthesis filters of a PR filter bank require that the determinants of the polyphase matrices are just delays.

6.4.3 Paraunitary Filter Banks

The paraunitary case is characterized by the fact that the sum of the energies of all subband signals is equal to the energy of the input signal. This may be expressed as $\|\mathbf{y}_p\| = \|\mathbf{x}_p\| \forall \mathbf{x}_p$ with $\|\mathbf{x}_p\| < \infty$, where $\mathbf{x}_p(z)$ is the polyphase vector of a finite-energy input signal and $\mathbf{y}_p(z) = \mathbf{E}(z) \mathbf{x}_p(z)$ is the vector of subband signals. It can easily be verified that filter banks (oversampled and critically sampled) are paraunitary if the following condition holds:

$$\tilde{\mathbf{E}}(z) \mathbf{E}(z) = \mathbf{I}. \quad (6.87)$$

This also implies that

$$h_k(n) = g_k^*(-n) \longleftrightarrow H_k(z) = \tilde{G}_k(z), \quad k = 0, \dots, M-1. \quad (6.88)$$

Especially in the critically subsampled case where $N = M$, the impulse responses $h_k(n-mM)$ and $g_k(n-mM)$, $k = 0, \dots, M-1$, $m \in \mathbb{Z}$, respectively, form orthonormal bases:

$$\sum_n h_j^*(mM+n) h_k(n) = \delta_{m0} \delta_{jk}, \quad (6.89)$$

$$\sum_n g_j^*(mM+n) g_k(n) = \delta_{m0} \delta_{jk}. \quad (6.90)$$

6.4.4 Design of Critically Subsampled M -Channel FIR Filter Banks

Analogous to the lattice structures introduced in Sections 6.2.6 and 6.2.7, we consider the following factorization of $\mathbf{E}(z)$:

$$\mathbf{E}(z) = \mathbf{A}_K \mathbf{D}(z) \mathbf{A}_{K-1} \mathbf{D}(z) \cdots \mathbf{D}(z) \mathbf{A}_0, \quad (6.91)$$

where

$$\mathbf{D}(z) = \begin{bmatrix} \mathbf{I}_{M-1} & \mathbf{0} \\ \mathbf{0} & z^{-1} \end{bmatrix}. \quad (6.92)$$

The matrices \mathbf{A}_k , $k = 0, 1, \dots, K$ are arbitrary non-singular matrices. The elements of these matrices are the free design parameters, which can be chosen in order to obtain some desired filter properties. To achieve this, a useful objective function has to be defined and the free parameters have to be found via non-linear optimization. Typically, one tries to minimize the stopband energy of the filters.

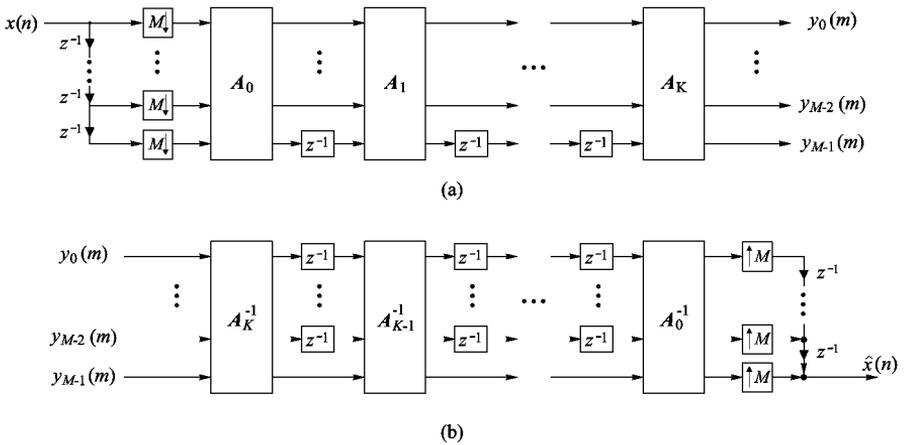


Figure 6.20. M -channel filter bank with FIR filters; (a) analysis; (b) synthesis.

where \mathbf{v}_k is an $M \times 1$ vector with $\|\mathbf{v}_k\| = \mathbf{v}_k^T \mathbf{v}_k = 1$. It is easily proven that $\mathbf{V}_k^T(z^{-1})\mathbf{V}_k(z) = \mathbf{I}$, so that the matrices can indeed be used for parameterization. The matrix \mathbf{U} has to be a general unitary matrix. The parameterization (6.97) directly leads to an efficient implementation, which is similar to the one discussed in Section 3.4.4 for the implementation of Householder reflections: instead of multiplying an input vector $\mathbf{x}(z)$ with an entire matrix $\mathbf{V}_k(z)$, one computes $\mathbf{x}(z) - \mathbf{v}_k[1 - z^{-1}][\mathbf{v}_k^T \mathbf{x}(z)]$ in order to obtain $\mathbf{V}_k(z)\mathbf{x}(z)$.

In addition to the above parameterizations, which generally yield non-linear phase filters, methods for designing linear-phase paraunitary filter banks have also been developed. For this special class the reader is referred to [137].

6.5 DFT Filter Banks

DFT filter banks belong to the class of modulated filter banks, where all filters are derived from prototypes via modulation. Modulated filter banks have the great advantage that only suitable prototypes must be found, not the complete set of analysis and synthesis filters. One prototype is required for the analysis and one for the synthesis side, and in most cases the same prototypes can be used for both sides. Due to the modulated structure very efficient implementations are possible.

In DFT banks, the analysis and synthesis filters, $H_k(z)$ and $G_k(z)$, are

related to the analysis and synthesis prototypes, $P(z)$ and $Q(z)$, as

$$\begin{aligned} H_k(z) &= P(W_M^k z) \quad \longleftrightarrow \quad h_k(n) = p(n) W_M^{-kn} \\ G_k(z) &= Q(W_M^k z) \quad \longleftrightarrow \quad g_k(n) = q(n) W_M^{-kn}. \end{aligned} \quad (6.99)$$

In order to explain the efficient implementation of DFT banks, let us consider the critically subsampled case. The analysis equation is

$$\begin{aligned} y_k(m) &= \sum_{n=0}^{L-1} h_k(n) x(mM - n) \\ &= \sum_{n=0}^{L-1} p(n) W_M^{-kn} x(mM - n). \end{aligned} \quad (6.100)$$

We now substitute $n = iM + j$, $L = ML_p$ and rewrite (6.100) as

$$\begin{aligned} y_k(m) &= \sum_{j=0}^{M-1} \sum_{i=0}^{L_p-1} p(iM + j) W_M^{-k(iM+j)} x(mM - iM - j) \\ &= \sum_{j=0}^{M-1} W_M^{-kj} \sum_{i=0}^{L_p-1} p(iM + j) x(mM - iM - j) \\ &= \sum_{j=0}^{M-1} W_M^{-kj} \sum_{i=0}^{L_p-1} p_j(i) \bar{x}_j(m - i). \end{aligned} \quad (6.101)$$

Thus, the subband signals can be computed by filtering the polyphase components of the input signal with the polyphase components of the prototype, followed by an IDFT (without pre-factor $1/M$). On the synthesis side, the same principle can be used. The complete analysis/synthesis system, which requires extremely low computation effort, is depicted in Figure 6.21.

For critical subsampling, as shown in Figure 6.21, the PR condition is easily found to be

$$P_k(z) Q_{M-1-k}(z) = \frac{z^{-q_0}}{M}. \quad (6.102)$$

This means that the polyphase components of PR FIR prototypes are restricted to length one, and the filtering degenerates to pointwise scalar multiplication. Thus, critically subsampled DFT filter banks with PR mainly reduce to the DFT.

If oversampling by a factor $\mu = \frac{M}{N} \in \mathbb{Z}$ is considered, the PR condition becomes [33, 86]

$$\sum_{\ell=0}^{\mu-1} P_{k+\ell N}(z) Q_{M-1-k-\ell N}(z) = \frac{z^{-q_0}}{M}. \quad (6.103)$$

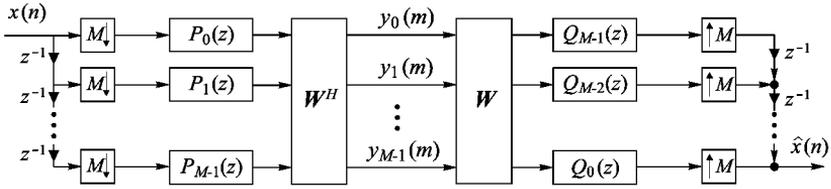


Figure 6.21. DFT polyphase filter bank with critical subsampling.

Clearly, if a filter bank provides PR in the critically subsampled case, it also provides PR in the oversampled case, provided the output signal is downsampled by the oversampling factor. Thus, (6.102) is included in (6.103). This is most easily seen from (6.103) for $\mu = 2$:

$$\underbrace{P_k(z) Q_{M-1-k}(z)}_{z^{-q_0/M}} + \underbrace{P_{k+N}(z) Q_{M-1-k-N}(z)}_{z^{-q_0/M}} = 2 \frac{z^{-q_0}}{M}.$$

In general, (6.103) means an increased design freedom compared to (6.102). This freedom can be exploited in order to design FIR prototypes $P(z)$ and $Q(z)$ with good filter properties.

The prototypes are typically designed to be lowpass filters. A common design criterion is to minimize the stopband energy and the passband ripple:

$$\int_{\text{passband}} \alpha (|P(e^{j\omega})| - 1)^2 d\omega + \int_{\text{stopband}} \beta |P(e^{j\omega})|^2 d\omega \stackrel{\!}{=} \min. \quad (6.104)$$

At this point it should be mentioned that all PR prototypes for M -channel cosine-modulated filter banks, which will be discussed in the next section, also serve as PR prototypes for oversampled $2M$ -channel DFT filter banks. On the other hand, satisfying only (6.103) is not sufficient in the cosine-modulated case. Thus, oversampled DFT filter banks offer more design freedom than cosine-modulated ones.

MDFT Filter Bank. Figure 6.22 shows the MDFT filter bank introduced by Fliege. Compared to the simple DFT filter banks described above, this filter bank is modified in such a way that PR is achieved with FIR filters [55], [82]. The key to PR is subsampling the filter output signals by $M/2$, extracting the real and imaginary parts, and using them to compose the complex subband signals $y_k(m)$, $k = 0, \dots, M - 1$. As can be seen in Figure 6.22, the extraction of the real and imaginary parts takes place in adjoining channels in reverse order.

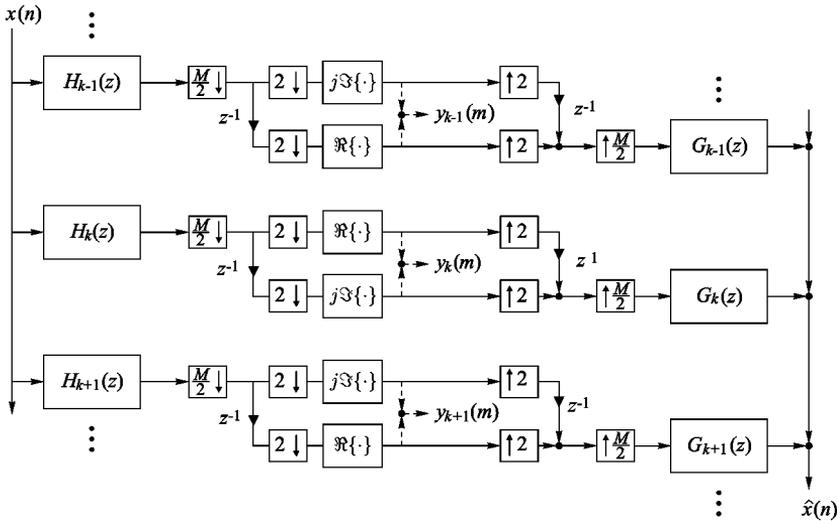


Figure 6.22. Modified complex modulated filter bank with critical subsampling.

DFT Polyphase Filter Bank with IIR Filters and Perfect Reconstruction. We consider the DFT filter bank in Figure 6.21. Husøy and Ramstad proposed to construct the polyphase components of the prototype as first-order IIR allpass filters [74]:

$$P_i(z) = \frac{1}{\sqrt{M}} \frac{a_i + z^{-1}}{1 + a_i z^{-1}}, \quad i = 0, \dots, M - 1. \tag{6.105}$$

Using the synthesis filters

$$Q_{M-1-i}(z) = P_i(z^{-1}) = \frac{1}{\sqrt{M}} \frac{1 + a_i z^{-1}}{a_i + z^{-1}}, \quad i = 0, \dots, M - 1, \tag{6.106}$$

then ensures perfect reconstruction. Unfortunately, this leads to a problem concerning stability: if the analysis filters are stable, the synthesis filters determined according to (6.106) are not. This problem can be avoided by filtering the subband signals “backwards” using the stable analysis filters. Then, the desired output signal is formed by another temporal reversal. This is not a feasible strategy if we work with one-dimensional signals, but in image processing we *a priori* have finite-length signals so that this method can be applied nevertheless.

The quality of a filter bank is not only dependent on whether it reconstructs perfectly or not. The actual purpose of the filter bank is to separate different frequency bands, for example in order to provide a maximal coding

gain. The stopband attenuation of the prototype $P(z)$ composed of IIR allpasses is determined by the parameters a_i , $i = 0, \dots, M - 1$, so that these are the design parameters. Husøy and Ramstad state a stopband attenuation of 36.4 dB for the prototype $P(z)$ of an eight-channel filter bank [74]. In view of the extremely low computational cost this is an astonishing value.

6.6 Cosine-Modulated Filter Banks

Cosine-modulated filter banks are very popular due to their real-valued nature and their efficient implementation via polyphase structure and fast DCT [116, 127, 94, 121, 87, 100, 110, 129, 68]. Cosine-modulated filter banks can be designed as pseudo QMF banks [127], paraunitary filter banks [94, 121, 87, 100, 103], and also as biorthogonal filter banks allowing low reconstruction delay [110, 129, 68, 83, 86]. Perfect reconstruction is easily achieved by choosing an appropriate prototype. For example, the MPEG audio standard [17] is based on cosine-modulated filter banks.

In the following, we will consider biorthogonal cosine-modulated filter banks where the analysis filters $h_k(n)$, $k = 0, \dots, M - 1$, are derived from an FIR prototype $p(n)$ and the synthesis filters $g_k(n)$, $k = 0, \dots, M - 1$, from an FIR prototype $q(n)$ according to

$$h_k(n) = 2p(n) \cos \left[\frac{\pi}{M} \left(k + \frac{1}{2} \right) \left(n - \frac{D}{2} \right) + \phi_k \right], \quad n = 0, \dots, L_p - 1$$

$$g_k(n) = 2q(n) \cos \left[\frac{\pi}{M} \left(k + \frac{1}{2} \right) \left(n - \frac{D}{2} \right) - \phi_k \right], \quad n = 0, \dots, L_q - 1.$$

The length of the analysis prototype is L_p , and the length of the synthesis prototype is L_q . The variable D denotes the overall delay of the analysis-synthesis system. A suitable choice for ϕ_k is given by $\phi_k = (-1)^k \pi/4$ [87, 95].

For the sake of brevity, we confine ourselves to even M , analysis and synthesis prototypes with lengths $L_p = 2mM$ and $L_q = 2m'M$, $m, m' \in \mathbb{N}$, and an overall delay of $D = 2sM + 2M - 1$ samples. Note that the delay can be chosen independently of the filter length, so that the design of low-delay banks is included here. The most common case within this framework is the one where the same prototype is used for analysis and synthesis. However, in order to demonstrate some of the design freedom, we start with a more general approach where different prototypes are used for analysis and synthesis. Generalizations to all filter lengths and delays are given in [68].

In order to derive the conditions that must be met by the prototypes $P(z)$ and $Q(z)$ to yield perfect reconstruction, we first decompose them into $2M$ polyphase components. Note that in the case of DFT filter banks, only M polyphase components were used to describe an M -channel filter bank. We use the type-1 decomposition given by

$$P_j(z) = \sum_{\ell=0}^{m-1} p(2\ell M + j) z^{-\ell}, \quad j = 0, \dots, 2M - 1. \quad (6.107)$$

6.6.1 Critically Subsampled Case

In the critically subsampled case the analysis polyphase matrix can be written as [112, 68]

$$\mathbf{E}(z) = \mathbf{T}_1 \begin{bmatrix} \mathbf{P}_0(z^2) \\ z^{-1} \mathbf{P}_1(z^2) \end{bmatrix}, \quad (6.108)$$

where

$$\begin{aligned} [\mathbf{T}_1]_{k,j} &= 2 \cos \left[\frac{\pi}{M} \left(k + \frac{1}{2} \right) \left(j - \frac{D}{2} \right) + \phi_k \right], \\ &k = 0, \dots, M - 1, \quad j = 0, \dots, 2M - 1, \end{aligned} \quad (6.109)$$

and

$$\begin{aligned} \mathbf{P}_0(z^2) &= \text{diag} [P_0(-z^2), P_1(-z^2), \dots, P_{M-1}(-z^2)], \\ \mathbf{P}_1(z^2) &= \text{diag} [P_M(-z^2), P_{M+1}(-z^2), \dots, P_{2M-1}(-z^2)]. \end{aligned} \quad (6.110)$$

Note that the matrices $\mathbf{P}_0(z^2)$ and $\mathbf{P}_1(z^2)$ contain upsampled and modulated versions of the polyphase filters.

For the synthesis polyphase matrix we get

$$\mathbf{R}(z) = [z^{-1} \mathbf{Q}_1(z^2), \mathbf{Q}_0(z^2)] \mathbf{T}_2^T, \quad (6.111)$$

where

$$\begin{aligned} [\mathbf{T}_2]_{k,j} &= 2 \cos \left[\frac{\pi}{M} \left(k + \frac{1}{2} \right) \left(2M - 1 - j - \frac{D}{2} \right) - \phi_k \right], \\ &k = 0, \dots, M - 1, \quad j = 0, \dots, 2M - 1, \end{aligned} \quad (6.112)$$

and

$$\begin{aligned} \mathbf{Q}_0(z^2) &= \text{diag} [Q_{M-1}(-z^2), \dots, Q_1(-z^2), Q_0(-z^2)], \\ \mathbf{Q}_1(z^2) &= \text{diag} [Q_{2M-1}(-z^2), \dots, Q_{M+1}(-z^2), Q_M(-z^2)]. \end{aligned} \quad (6.113)$$

The perfect reconstruction conditions are obtained by setting

$$\mathbf{R}(z)\mathbf{E}(z) = z^{-q_0} \mathbf{I}_M. \quad (6.114)$$

Using the property [87]

$$\mathbf{T}_2^T \mathbf{T}_1 = (-1)^s 2M \mathbf{I}_{2M} + 2M \begin{bmatrix} \mathbf{J}_M & \mathbf{0} \\ \mathbf{0} & -\mathbf{J}_M \end{bmatrix}, \quad (6.115)$$

this yields the conditions

$$P_k(z) Q_{2M-1-k}(z) + P_{M+k}(z) Q_{M-1-k}(z) \stackrel{!}{=} \frac{z^{-s}}{2M}, \quad (6.116)$$

$$P_k(z) Q_{M+k}(z) - P_{M+k}(z) Q_k(z) \stackrel{!}{=} 0, \quad (6.117)$$

which have to be met for $k = 0, \dots, \frac{M}{2} - 1$. The relationship between q_0 and s is

$$q_0 = 2s + 1. \quad (6.118)$$

The condition (6.117) is satisfied for $Q_k(z) = \alpha z^{-\beta} P_k(z)$ and $Q_{M+k}(z) = \alpha z^{-\beta} P_{M+k}(z)$ with arbitrary α, β , which suggests the use of the same prototype for both analysis and synthesis. Thus, with $Q(z) = P(z)$, the remaining condition is

$$P_{2M-1-k}(z) P_k(z) + P_{M+k}(z) P_{M-1-k}(z) \stackrel{!}{=} \frac{z^{-s}}{2M}, \quad k = 0, \dots, \frac{M}{2} - 1. \quad (6.119)$$

The $M/2$ equations in (6.119) may be understood as PR conditions on $M/2$ non-subsampled two-channel filter banks. The prototype can for instance be designed by using the quadratic-constrained least-squares (QCLS) approach, which was proposed by Nguyen [111]. Here, we write all constraints given by (6.119) in quadratic form and optimize the prototype using constrained numerical optimization. The approach does not inherently guarantee PR, but the PR constraints can be satisfied with arbitrary accuracy.

Another approach, which guarantees PR and also leads to a very efficient implementation of the filter bank, is to design the filters via lifting [129, 83]. For this, we write the PR conditions as

$$\mathbf{V}(z)\mathbf{U}(z) = \frac{z^{-1}(-z^{-2})^s}{2M} \mathbf{I}, \quad (6.120)$$

where

$$\begin{aligned} \mathbf{U}(z) &= \begin{bmatrix} P_k(-z^2) & (-1)^s P_{M-1-k}(-z^2) \\ (-1)^{s-1} z^{-1} P_{k+M}(-z^2) & z^{-1} P_{2M-1-k}(-z^2) \end{bmatrix} \\ \mathbf{V}(z) &= \begin{bmatrix} z^{-1} Q_{2M-1-k}(-z^2) & (-1)^{s-1} Q_{2M-1-k-M}(-z^2) \\ (-1)^s z^{-1} Q_{k+M}(-z^2) & Q_k(-z^2) \end{bmatrix}. \end{aligned} \quad (6.121)$$

It is easily verified that (6.120) includes (6.117) and (6.116), but (6.120) can also be derived straightforwardly from (6.114) by using the properties of the cosine functions [83]. The filter design is as follows. We start with

$$\begin{aligned} \mathbf{U}_0(z) &= \begin{bmatrix} p_0 & p_1 \\ -p_2 z^{-1} & p_3 z^{-1} \end{bmatrix} \\ \mathbf{V}_0(z) &= \frac{1}{2M} \frac{1}{p_0 p_3 - p_1 p_2} \begin{bmatrix} p_3 z^{-1} & -p_1 \\ -p_2 z^{-1} & p_0 \end{bmatrix}, \end{aligned} \quad (6.122)$$

where the subscript 0 indicates that this is the 0th iteration. We have

$$\mathbf{V}_0(z) \mathbf{U}_0(z) = \frac{z^{-1}}{2M} \mathbf{I}. \quad (6.123)$$

Longer filters with the same delay are constructed by introducing matrices of the type

$$\mathbf{A}_i^{-1}(z) \mathbf{A}_i(z) = \mathbf{I} \quad (6.124)$$

with

$$\mathbf{A}_i(z) = \begin{bmatrix} a_i z^{-1} & 1 \\ 1 & 0 \end{bmatrix}, \quad \mathbf{A}_i^{-1}(z) = \begin{bmatrix} 0 & 1 \\ 1 & -a_i z^{-1} \end{bmatrix} \quad (6.125)$$

in between the product $\mathbf{V}_i(z) \mathbf{U}_i(z)$:

$$\begin{aligned} \mathbf{U}_{i+1}(z) &= \mathbf{A}_i(z) \mathbf{U}_i(z), \\ \mathbf{V}_{i+1}(z) &= \mathbf{V}_i(z) \mathbf{A}_i^{-1}(z). \end{aligned} \quad (6.126)$$

Note that $\mathbf{U}_{i+1}(z)$ and $\mathbf{V}_{i+1}(z)$ retain the structure given in (6.121). From the new matrices the polyphase components of the prototype are easily extracted. The operation (6.126) can be repeated until the filters contained in $\mathbf{U}_i(z)$ and $\mathbf{V}_i(z)$ have the desired length. Since the overall delay remains constant, this operation is called zero-delay lifting.

A second possibility is to introduce matrices

$$\mathbf{B}_i(z) = \begin{bmatrix} 0 & -z^{-1} \\ -z^{-1} & c_i \end{bmatrix}, \quad \mathbf{C}_i(z) = \begin{bmatrix} c_i & z^{-1} \\ z^{-1} & 0 \end{bmatrix} \quad (6.127)$$

and to construct the new filters as

$$\begin{aligned} \mathbf{U}_{i+1}(z) &= \mathbf{C}_i(z)\mathbf{U}_i(z), \\ \mathbf{V}_{i+1}(z) &= \mathbf{V}_i(z)\mathbf{B}_i(z). \end{aligned} \tag{6.128}$$

This type of lifting is known as maximum-delay lifting. Again, $\mathbf{U}_{i+1}(z)$ and $\mathbf{V}_{i+1}(z)$ have the structure given in (6.121), and since (6.120) is satisfied, PR is structurally guaranteed. Thus, filter optimization can be carried out by optimizing the lifting coefficients in an unrestricted way.

Also other lifting schemes can easily be found. The advantage of the above approach is that only one lifting step with one lifting coefficient a_i or c_i is needed in order to increase the length of two polyphase components of each prototype.

Implementation Issues. The straightforward polyphase implementation of (6.108) is depicted in Figure 6.23. On the analysis side, we see that always those two systems are fed with the same input signal which are connected in (6.116). In the synthesis bank, the output signals of the corresponding synthesis polyphase filters are added. This already suggests the joint implementation of pairs of two filters. However, a more efficient structure can be obtained by exploiting the periodicities in the rectangular matrices \mathbf{T}_1 and \mathbf{T}_2 and by replacing them with $M \times M$ cosine modulation matrices $\tilde{\mathbf{T}}_1$ and $\tilde{\mathbf{T}}_2 = \tilde{\mathbf{T}}_1^T = \tilde{\mathbf{T}}_1^{-1}$ [83]:

$$[\tilde{\mathbf{T}}_1]_{k,j} = \begin{cases} 2 \cos \left[\frac{\pi}{M} \left(k + \frac{1}{2} \right) \left(j - \frac{D}{2} \right) + \phi_k \right], & j = 0, \dots, \frac{M}{2} - 1 \\ 2 \cos \left[\frac{\pi}{M} \left(k + \frac{1}{2} \right) \left(M + j - \frac{D}{2} \right) + \phi_k \right], & j = \frac{M}{2}, \dots, M - 1 \end{cases} \tag{6.129}$$

for $k = 0, \dots, M - 1$. This structure is depicted in Figure 6.24. Note that the following signals are needed as input signals for the cosine transform:

$$\begin{bmatrix} \bar{Z}_k(z) \\ \bar{Z}_{M-1-k}(z) \end{bmatrix} = \begin{bmatrix} P_k(-z^2) & (-1)^s P_{M-1-k}(-z^2) \\ (-1)^{s-1} z^{-1} P_{k+M}(-z^2) & z^{-1} P_{2M-1-k}(-z^2) \end{bmatrix} \begin{bmatrix} \bar{X}_k(z) \\ \bar{X}_{M-1-k}(z) \end{bmatrix} \tag{6.130}$$

Thus, all polyphase filtering operations can be carried out via the lifting scheme described above where four filters are realized jointly.

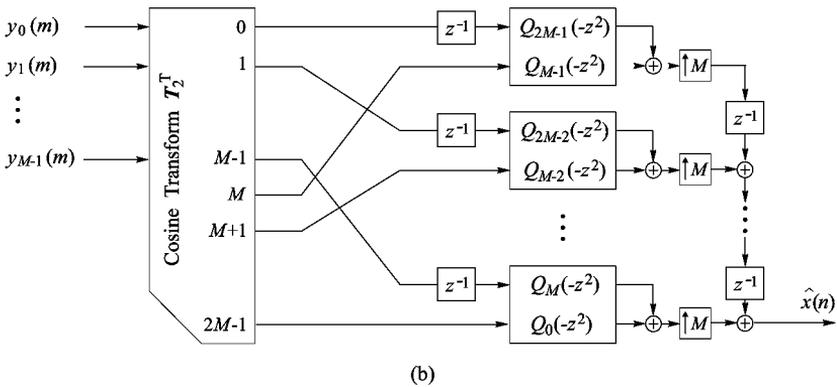
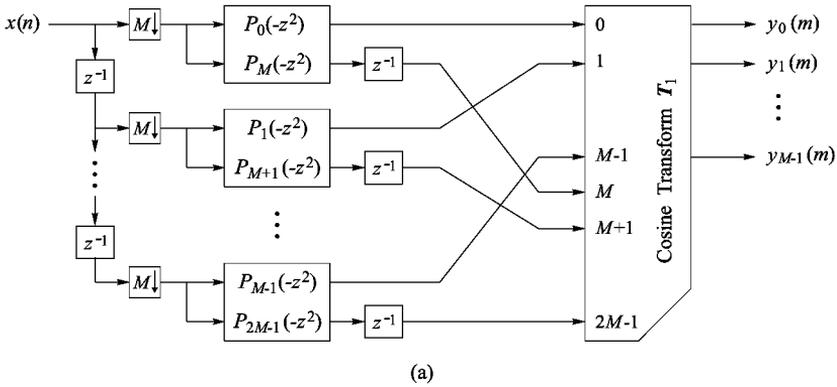


Figure 6.23. Cosine-modulated filter bank with critical subsampling. (a) analysis; (b) synthesis.

6.6.2 Paraunitary Case

In the paraunitary case with critical subsampling we have

$$\tilde{\mathbf{E}}(z) \mathbf{E}(z) = \mathbf{I}_M, \tag{6.131}$$

which leads to the following constraints on the prototype:

1. The prototype has to be linear-phase, that is, $p(L - 1 - n) = p(n)$.
2. The same prototype is required for both analysis and synthesis.
3. The prototype has to satisfy

$$\tilde{P}_k(z)P_k(z) + \tilde{P}_{M+k}(z)P_{M+k}(z) = \frac{1}{2M}. \tag{6.132}$$

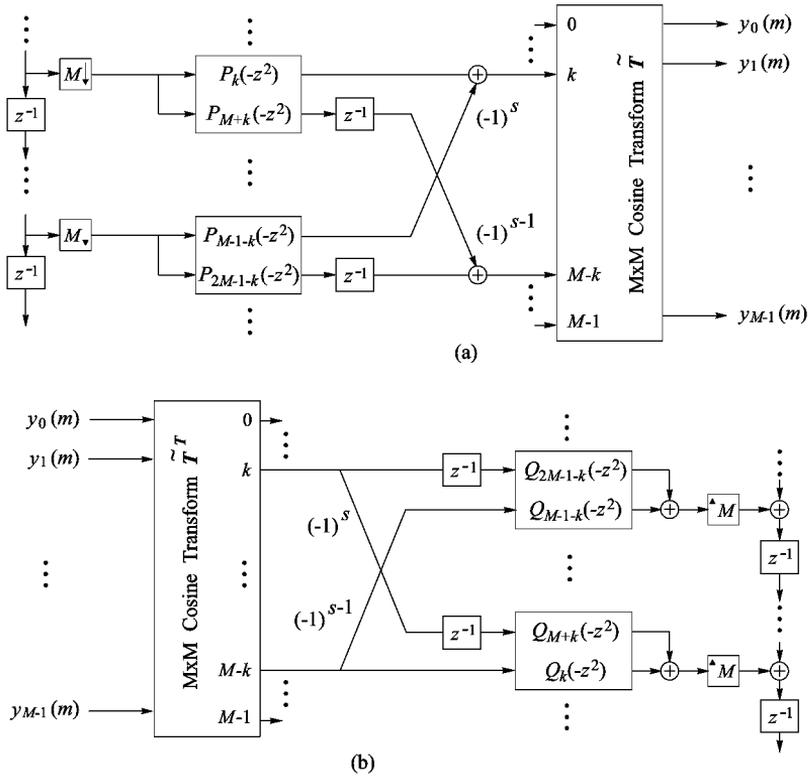


Figure 6.24. Cosine-modulated filter bank with critical subsampling and efficient implementation structure. (a) analysis; (b) synthesis.

The filter design may for instance be carried out by parameterizing the polyphase components using the lattice structure shown in Figure 6.25 and choosing the rotation angles so as to minimize an arbitrary objective function. For this method a good starting point is required, because we have to optimize angles in a cascade of lattices and the relationships between the angles and the impulse response are highly nonlinear. Alternatively, the QCLS approach [111] can be used, which typically is less sensitive to the starting point.

As in the biorthogonal case, the polyphase filters can be realized jointly. One can use the structure in Figure 6.23 and implement two filters at a time via the lattice in Figure 6.25. However, the more efficient structure in Figure 6.24 can also be used, where four filters are realized via a common lattice. This was shown in [95] for special filter lengths. A generalization is given in [62].

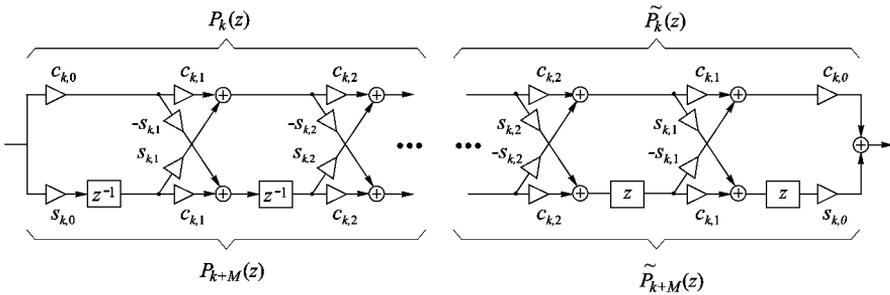


Figure 6.25. Lattice structure for the design and implementation of cosine-modulated filters.

In [103] a method has been proposed that allows the design of discrete-coefficient linear-phase prototypes for the paraunitary case. The design procedure is based on a subspace approach that allows us to perform linear combinations of PR prototype filters in such a way that the resulting filter is also a linear-phase PR prototype. The filter design is carried out iteratively, while the PR property is guaranteed throughout the design process. In order to give some design examples, Table 6.1 shows impulse responses of 8-band prototypes with integer coefficients and filter length $L = 32$. Because of symmetry, only the first 16 coefficients are listed. The frequency responses of the filters #3 and #6 are depicted in Figure 6.26.

Closed Form Solutions. For filter length $L = 2M$ and $L = 4M$ closed form solutions for PR prototypes are known. The special case $L = 2M$ is known as the *modulated lapped transform* (MLT), which was introduced by Princen and Bradley [116]. In this case the PR condition (6.132) reduces to

$$p_k(0)^2 + p_{M+k}(0)^2 = \frac{1}{2M},$$

which means

$$p^2(n) + p^2(M + n) = \frac{1}{2M}. \tag{6.133}$$

An example of an impulse response that satisfies (6.133) is

$$p(n) = \frac{1}{\sqrt{2M}} \sin \left[\left(n + \frac{1}{2} \right) \frac{\pi}{2M} \right]. \tag{6.134}$$

The case $L = 4M$ is known as the *extended lapped transform* (ELT). The ELT was introduced by Malvar, who suggested the following prototype [95]:

$$p(n) = -\frac{1}{4\sqrt{M}} + \frac{1}{2\sqrt{2M}} \cos \left[\left(n + \frac{1}{2} \right) \frac{\pi}{2M} \right]. \tag{6.135}$$

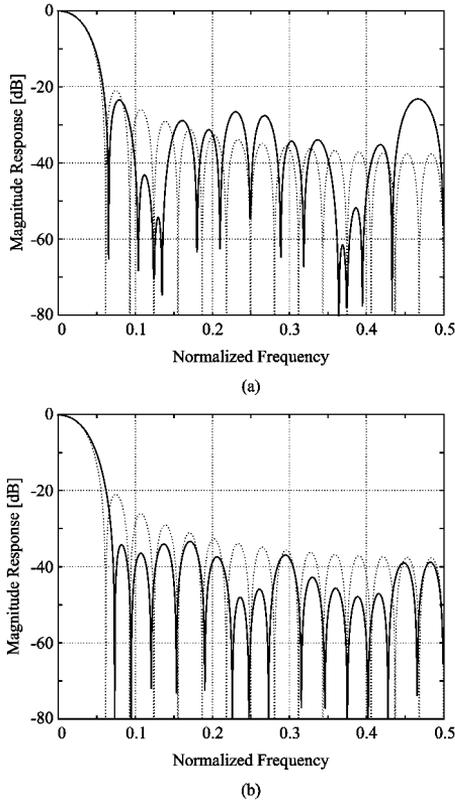


Figure 6.26. Frequency responses of 8-channel prototypes from Table 3.1. (a) filter #3; (b) filter #6. For comparison the frequency response of the ELT prototype is depicted with dotted lines.

Table 6.1.

Perfect reconstruction prototypes for 8-band filter banks with integer coefficients ($p(L - 1 - n) = p(n)$).

n	$p(n)$					
	#1	#2	#3	#4	#5	#6
0			-1	-6	-72	-2190
1			-1	-4	-97	-1901
2			0	0	-41	-1681
3			0	-6	-48	-426
4			0	7	56	497
5			0	0	62	2542
6			2	8	194	3802
7			2	17	204	6205
8		1	4	24	390	9678
9		1	4	33	524	13197
10		1	6	41	656	16359
11		1	6	48	774	19398
12	1	2	7	56	903	22631
13	1	2	7	62	992	24738
14	1	2	8	66	1048	26394
15	1	2	8	68	1105	27421

6.6.3 Oversampled Cosine-Modulated Filter Banks

In the oversampled case with oversampling by $\mu = \frac{M}{N} \in \mathbb{Z}$, the polyphase matrices may be written as

$$\mathbf{E}^{(\mu)}(z) = \frac{1}{\sqrt{\mu}} \cdot \mathbf{T} \begin{bmatrix} \mathbf{P}_0(z^{2\mu}) \\ z^{-1} \mathbf{P}_1(z^{2\mu}) \\ \vdots \\ z^{-(2\mu-1)} \mathbf{P}_{2\mu-1}(z^{2\mu}) \end{bmatrix} \quad (6.136)$$

and

$$\mathbf{R}^{(\mu)}(z) = \frac{1}{\sqrt{\mu}} \cdot \left[z^{-(2\mu-1)} \mathbf{Q}_{2\mu-1}(z^{2\mu}), \dots, z^{-1} \mathbf{Q}_1(z^{2\mu}), \mathbf{Q}_0(z^{2\mu}) \right] \mathbf{T}^T \quad (6.137)$$

with

$$\mathbf{P}_\ell(z^{2\mu}) = \text{diag} \{ P_{\ell N}(-z^{2\mu}), P_{\ell N+1}(-z^{2\mu}), \dots, P_{\ell N+(N-1)}(-z^{2\mu}) \}, \quad (6.138)$$

$$\mathbf{Q}_\ell(z^{2\mu}) = \text{diag} \{ Q_{\ell N+(N-1)}(-z^{2\mu}), \dots, Q_{\ell N+1}(-z^{2\mu}), Q_{\ell N}(-z^{2\mu}) \}. \quad (6.139)$$

The superscript (μ) indicates the oversampling factor. Requiring

$$\mathbf{R}^{(\mu)}(z) \mathbf{E}^{(\mu)}(z) = z^{-q_0^{(\mu)}} \quad (6.140)$$

for perfect reconstruction yields [86]

$$\sum_{\ell=0}^{2\mu-1} P_{k+\ell N}(z) Q_{2M-1-k-\ell N}(z) \stackrel{!}{=} \frac{z^{-s}}{2M} \quad (6.141)$$

and

$$P_{k+\ell N}(z) Q_{M+k+\ell N}(z) - P_{M+k+\ell N}(z) Q_{k+\ell N}(z) \stackrel{!}{=} 0 \quad (6.142)$$

for $k = 0, \dots, N-1$; $\ell = 0, \dots, \mu-1$. The delay $q_0^{(\mu)}$ is related to s as

$$q_0^{(\mu)} = 2\mu s + 2\mu - 1, \quad (6.143)$$

and the overall delay amounts to

$$q = N - 1 + q_0^{(\mu)} N. \quad (6.144)$$

As we see, these conditions offer increased design freedom for an increasing oversampling rate. This is further discussed in [86], where solutions based on a nullspace approach are presented.

If we restrict an oversampled cosine-modulated filter bank to be paraunitary, that is, $\tilde{\mathbf{E}}^{(\mu)}(z) \mathbf{E}^{(\mu)}(z) = \mathbf{I}_N$, we get the following constraints on the prototype $P(z)$ [85, 86]:

$$\sum_{\ell=0}^{2\mu-1} \tilde{P}_{k+\ell N}(z) P_{k+\ell N}(z) \stackrel{!}{=} \frac{1}{2M} \quad \text{for } k = 0, \dots, \left\lfloor \frac{N}{2} \right\rfloor - 1. \quad (6.145)$$

Interestingly, for $\mu > 1$, we still may choose different prototypes $P(z)$ and $Q(z)$ such that

$$\mathbf{R}^{(\mu)}(z) = z^{-(2mL-1)} \tilde{\mathbf{E}}^{(\mu)}(z) + \mathbf{N}^{(\mu)}(z)$$

with

$$\mathbf{N}^{(\mu)}(z) \mathbf{E}^{(\mu)}(z) = \mathbf{0}.$$

Example. We consider a 16-band filter bank with linear-phase prototype and an overall delay of 255 samples. Figure 6.27 shows a comparison of frequency responses for the critically sampled and the oversampled case. It turns out that the PR prototype for the oversampled filter bank has a much higher stopband attenuation. This demonstrates the increased design freedom in the oversampled case.

6.6.4 Pseudo-QMF Banks

In *pseudo-QMF banks*, one no longer seeks perfect reconstruction, but *nearly perfect reconstruction*. Designing a pseudo-QMF bank is done as follows [127]. One ensures that the aliasing components of adjacent channels compensate exactly. This requires power complementarity of frequency shifted versions of the prototype, as illustrated in Figure 6.28. Furthermore, one tries to suppress the remaining aliasing components by using filters with very high stopband attenuation. Through filter optimization the linear distortions are kept as small as possible. An efficient design method was proposed in [166]. Since the constraints on the prototype are less restrictive than in the PR case, the prototypes typically have a higher stopband attenuation than the PR ones.

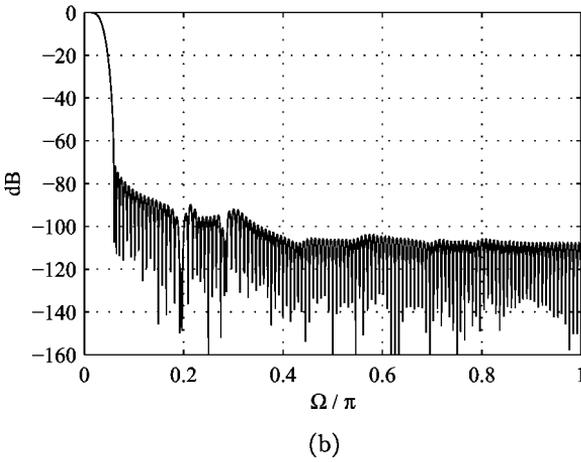
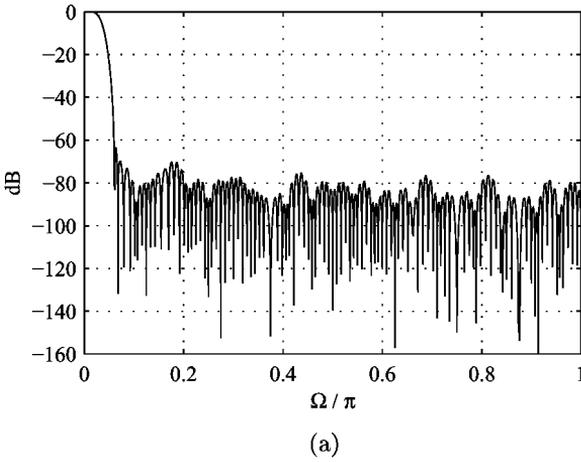


Figure 6.27. Frequency responses of 16-channel prototypes. (a) critical subsampling; (b) oversampling by $\mu = 2$.

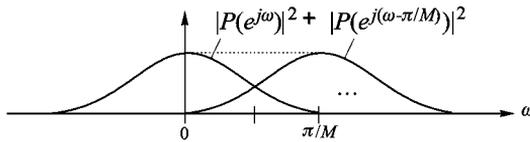


Figure 6.28. Design of pseudo-QMF banks.

The most general way of constructing LOTs is to start with two matrices \mathbf{A} and \mathbf{B} , where \mathbf{A} is a projection and \mathbf{B} is orthogonal. The desired matrices \mathbf{P}_0 and \mathbf{P}_1 are then found from (6.149). This method, however, does not automatically yield linear-phase filters, which are desired in many applications.

In [98], a fast linear-phase LOT based on the DCT was presented, which will be briefly explained in the following. For this, let \mathbf{D}_e and \mathbf{D}_o be matrices that contain the rows of the transposed DCT-II matrix with even and odd symmetry, respectively. Then,

$$\mathbf{Q}^{(0)} = [\mathbf{P}_0^{(0)}, \mathbf{P}_1^{(0)}] = \frac{1}{2} \begin{bmatrix} \mathbf{D}_e - \mathbf{D}_o & (\mathbf{D}_e - \mathbf{D}_o)\mathbf{J} \\ \mathbf{D}_e + \mathbf{D}_o & -(\mathbf{D}_e + \mathbf{D}_o)\mathbf{J} \end{bmatrix} \quad (6.150)$$

is a LOT matrix that already satisfies the above conditions. \mathbf{J} is the counter identity matrix with entries $J_{i,k} = \delta_{i,N-i-1}$, $i = 0, 1, \dots, N - 1$. In an expression of the form $\mathbf{X}\mathbf{J}$, it flips the columns of \mathbf{X} from left to right. Due to the application of \mathbf{J} in (6.150), the first $M/2$ rows of $\mathbf{Q}^{(0)}$ have even and the last $M/2$ rows have odd symmetry. A transform matrix with better properties (e.g. for coding) can be obtained by rotating the columns of $\mathbf{Q}^{(0)}$ such that

$$\mathbf{Q} = \mathbf{Z} \mathbf{Q}^{(0)}, \quad (6.151)$$

where \mathbf{Z} is unitary. For the fast LOT, \mathbf{Z} is chosen to contain only three plane rotations, which help to improve the performance, but do not significantly increase the complexity. The matrix $\mathbf{Q}^{(0)}$ already has a fast implementation based on the fast DCT. See Figure 6.30 for an illustration of the fast LOT. The angles proposed by Malvar are $\theta_1 = 0.13\pi$, $\theta_2 = 0.16\pi$, and $\theta_3 = 0.13\pi$.

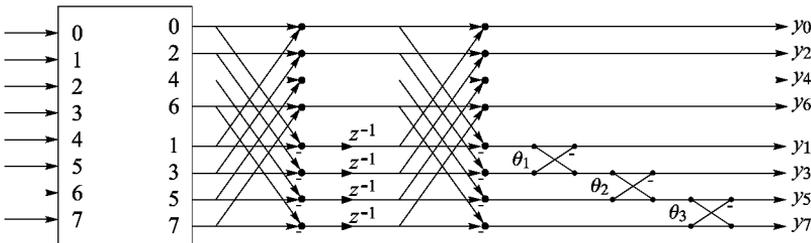


Figure 6.30. The fast lapped orthogonal transform for $M = 8$ based on the DCT and three plane rotations.

6.8 Subband Coding of Images

Two-dimensional filter banks for the decomposition of images can be realized as separable and non-separable filter banks. For the sake of simplicity, we will restrict ourselves to the separable case. Information on non-separable filter banks and the corresponding filter design methods is given in [1, 154].

In separable filter banks, the rows and columns of the input signal (image) are filtered successively. The procedure is illustrated in Figure 6.31 for an octave-band decomposition based on cascades of one-dimensional two-channel filter banks. In Figure 6.32 an example of such an octave-band decomposition is given. Note that this decomposition scheme is also known as the discrete wavelet transform; see Chapter 8. In Figure 6.32(b) we observe that most information is contained in the lower subbands. Moreover, local high-frequency information is kept locally within the subbands. These properties make such filter banks very attractive for image coding applications. In order to achieve high compression ratios, one quantizes the decomposed image, either by scalar quantization, or using a technique known as *embedded zerotree coding* [131, 128]; see also Section 8.9. The codewords describing the quantized values are usually further compressed in a lossless way by *arithmetic* or *Huffman coding* [76, 63]. To demonstrate the characteristics of subband coding with octave-band filter banks, Figures 6.32(c) and (d) show coding results at different bit rates.

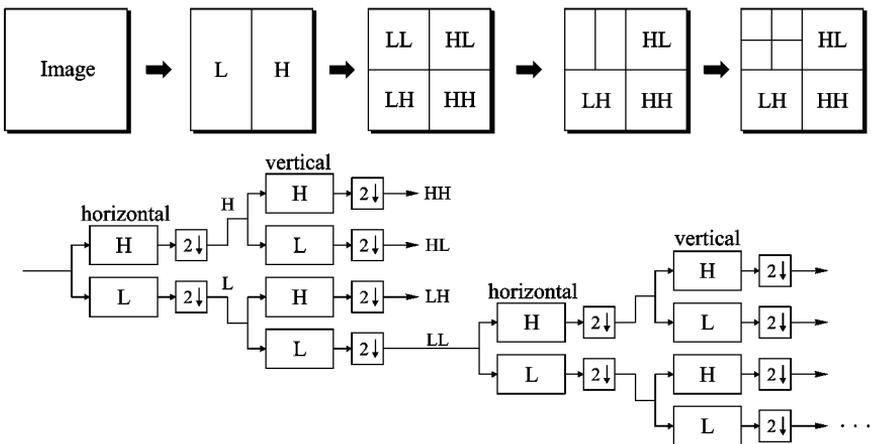


Figure 6.31. Separable two-dimensional octave-band filter bank.

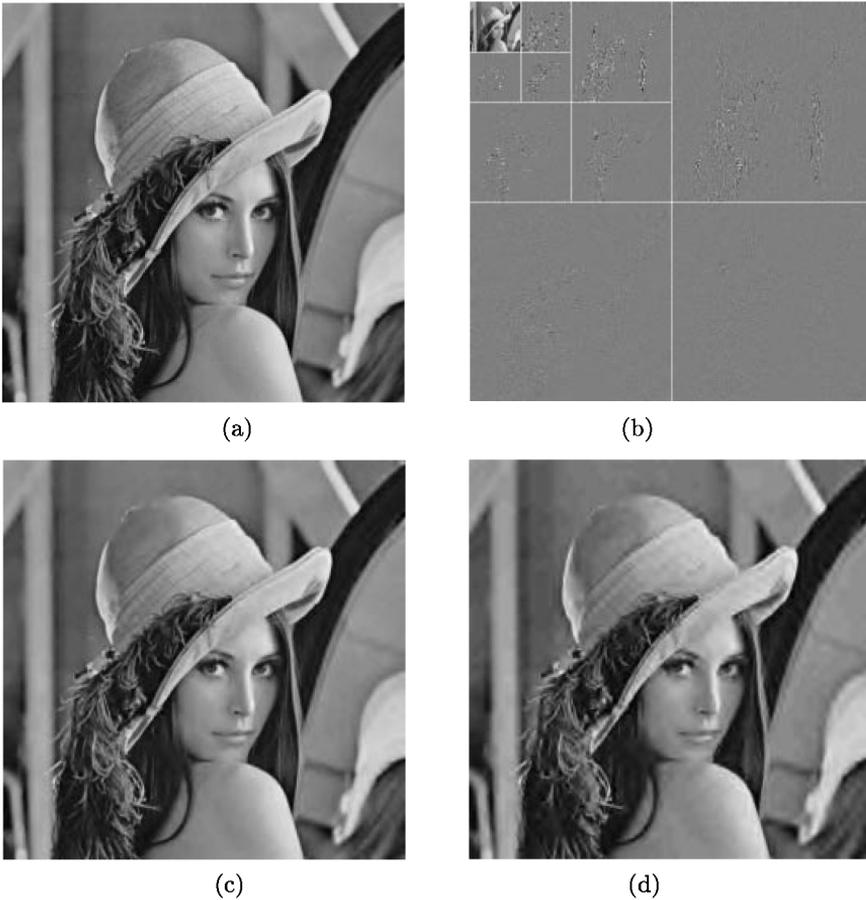


Figure 6.32. Examples of subband coding; (a) original image of size 512×512 ; (b) ten-band octave decomposition; (c) coding at 0.2 bits per pixel; (d) coding at 0.1 bits per pixel.

6.9 Processing of Finite-Length Signals

The term “critical sampling”, used in the previous sections, was used under the assumption of infinitely long signals. This assumption is justified with sufficient accuracy for audio and speech coding. However, if we want to decompose an image by means of a critically subsampled filter bank, we see that the number of subband samples is larger than the number of input values. Figure 6.33 gives an example. If we simply truncate the number of subband samples to the number of input values – which would be desirable for coding – then PR is not possible any longer. Solutions to this problem that yield PR

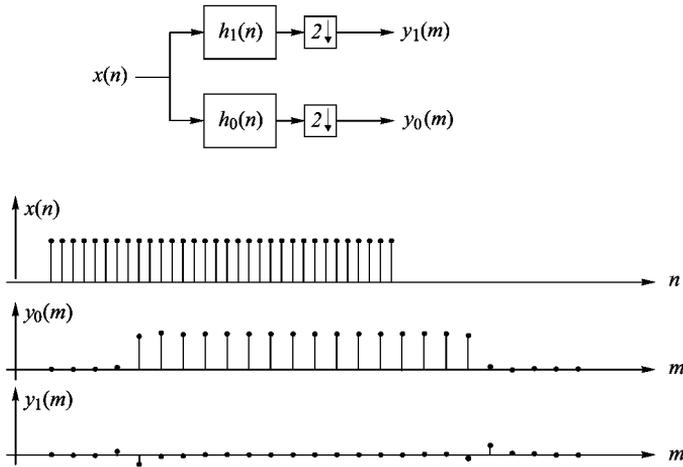


Figure 6.33. Two-channel decomposition of a finite-length signal.

with a minimum number of subband samples are discussed in the following.

Circular Convolution. Assuming that the length of the signal to be processed is a multiple of the number of channels, the problem mentioned above can be solved by circular convolution. In this method, the input signal is extended periodically prior to decomposition [165], which yields periodic subband signals of which only one period has to be stored or transmitted. Figures 6.34(a) and 6.34(c) give an illustration. Synthesis is performed by extending the subband signals according to their symmetry, filtering the extended signals, and extracting the required part of the output signal. A drawback of circular convolution is the occurrence of discontinuities at the signal boundaries, which may lead to annoying artifacts after reconstruction from quantized subband signals.

Symmetric Reflection. In this method, the input signal is extended periodically by reflection at the boundaries as indicated in Figures 6.34(b) and 6.34(d), [136, 16, 23, 6]. Again, we get periodic subband signals, but the period is twice as long as with circular convolution. However, only half a period of the subband signals is required if linear-phase filters are used, because they lead to symmetry in the subbands. By comparing Figures 6.34(a) and (b) (or 6.34(c) and (d)) we see that symmetric reflection leads to smoother transitions at the boundaries than circular convolution does. Thus, when quantizing the subband signals, this has the effect of less severe boundary distortions.

The exact procedure depends on the filter bank in use and on the signal

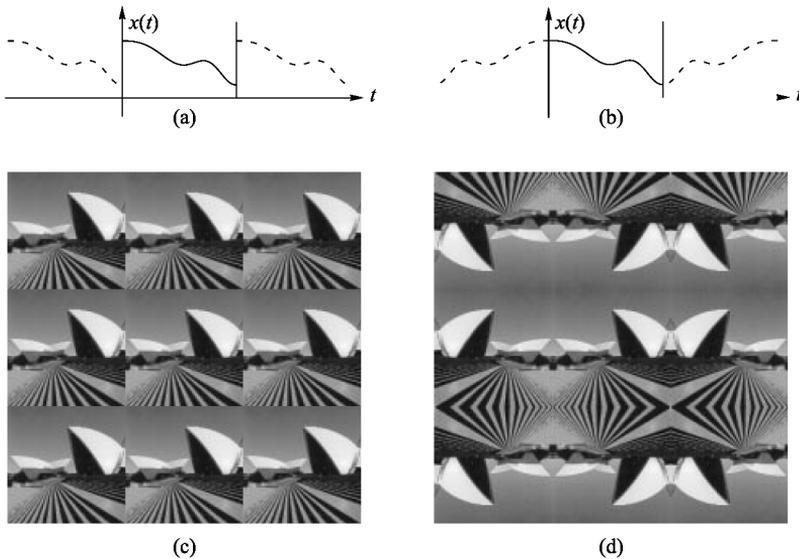


Figure 6.34. Periodic extension of the input signal; (a) one-dimensional circular convolution; (b) one-dimensional symmetric reflection; (c) two-dimensional circular convolution; (d) two-dimensional symmetric reflection.

length. Figure 6.35(a) shows a scheme suitable for the two-band decomposition of an even-length signal with linear-phase odd-length biorthogonal filters. The input signal is denoted as x_0, x_1, \dots, x_7 , and the filter impulse responses are $\{A, B, C, B, A\}$ for the lowpass and $\{-a, b, -a\}$ for the highpass. The upper row shows the extended input signal, where the given input samples are shown in solid boxes. The lowpass and highpass subband samples, c_n and d_n , respectively, are computed by taking the inner products of the impulse responses in the displayed positions with the corresponding part of the extended input signal. We see that only four different lowpass and highpass coefficients occur and have to be transmitted. A second scheme for the same filters which also allows the decomposition of even-length signals into lowpass and highpass components of half the length is depicted in Figure 6.35(b). In order to distinguish between both methods we say that the starting position in Figure 6.35(a) is even and the one in Figure 6.35(b) is odd, as indicated by the indices of the samples. Combinations of both schemes can be used to decompose odd-length signals. Moreover, these schemes can be used for the decomposition of 2-D objects with arbitrary shape. We will return to this topic at the end of this section.

Schemes for the decomposition of even-length signals with even-length

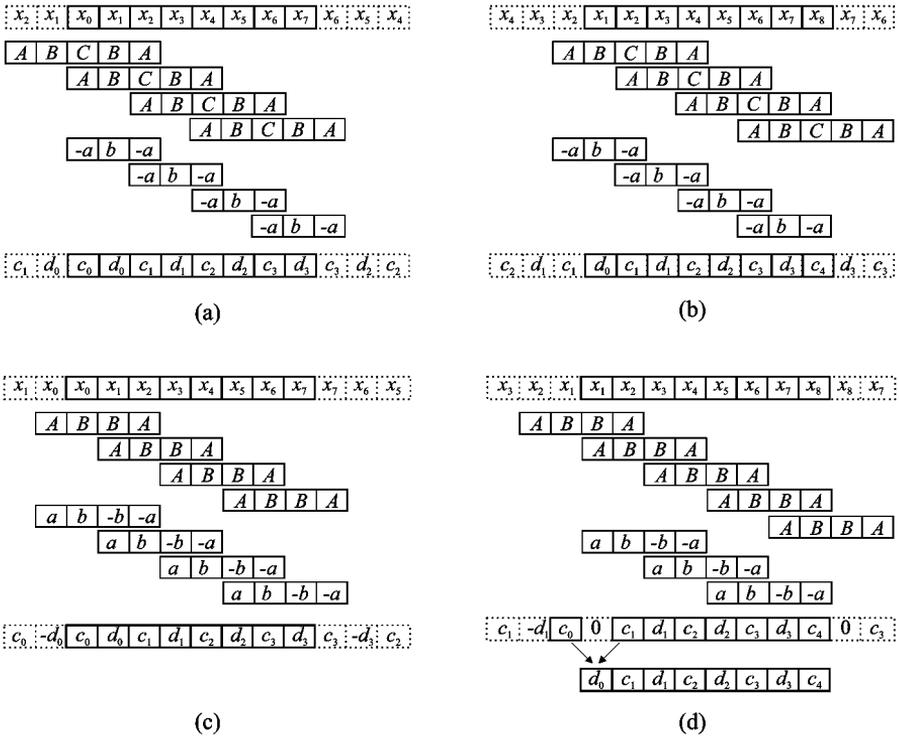


Figure 6.35. Symmetric reflection for even-length signals. (a) odd-length filters, segment starting at an even position; (b) odd-length filters, segment starting at an odd position; (c) even-length filters, segment starting at an even position; (d) even-length filters, segment starting at an odd position.

linear-phase filters are depicted in Figures 6.35(c) and (d). The filter impulse responses are $\{A, B, B, A\}$ for the lowpass and $\{-a, -b, b, a\}$ for the highpass. Note that a different type of reflection is used and that we have other symmetries in the subbands. While the scheme in Figure 6.35(c) results in the same number of lowpass and highpass samples, the one in Figure 6.35(d) yields an extra lowpass value, while the corresponding highpass value is zero. However, the additional lowpass samples can be turned into highpass values by subtracting them from the following lowpass value and storing the differences in the highpass band.

In object based image coding, for instance MPEG-4 [109], it is required to carry out subband decompositions of arbitrarily shaped objects. Figure 6.36

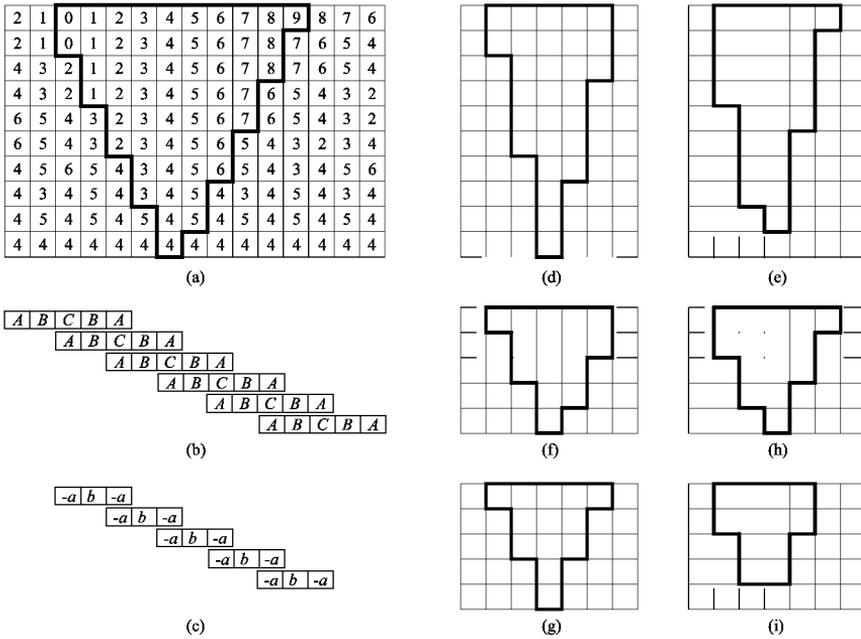


Figure 6.36. Shape adaptive image decomposition using symmetric reflection for odd-length two-channel filter banks; (a) arbitrarily shaped object and horizontal extension with pixel values as indicated; (b) lowpass filter; (c) highpass filter; (d) and (e) lowpass and highpass subbands after horizontal filtering; (f) and (g) lowpass and highpass decompositions of the signal in (d); (h) and (i) lowpass and highpass decompositions of the signal in (e).

shows a scheme which is suitable for this task using odd-length filters. The arbitrarily shaped input signal is shown in the marked region, and the extension for the first horizontal decomposition is found outside this region. Figures 6.36(d) and (e) show the shape of the lowpass and highpass band, respectively. Figures 6.36(f)–(i) finally show the object shapes after the vertical decomposition of the signals in Figures 6.36(d) and (e) based on the same reflection scheme. Such schemes are often called shape adaptive wavelet transforms. Note that the overall number of subband samples is equal to the number of input pixels. Moreover, the scheme yields a decomposition where the interior region of an object is processed as if the object was of infinite size. Thus, the actual object shape only influences the subband samples close to the boundaries. The 2-D decomposition is carried out in such a way that the horizontal decomposition introduces minimal distortion for the next vertical one and vice versa.

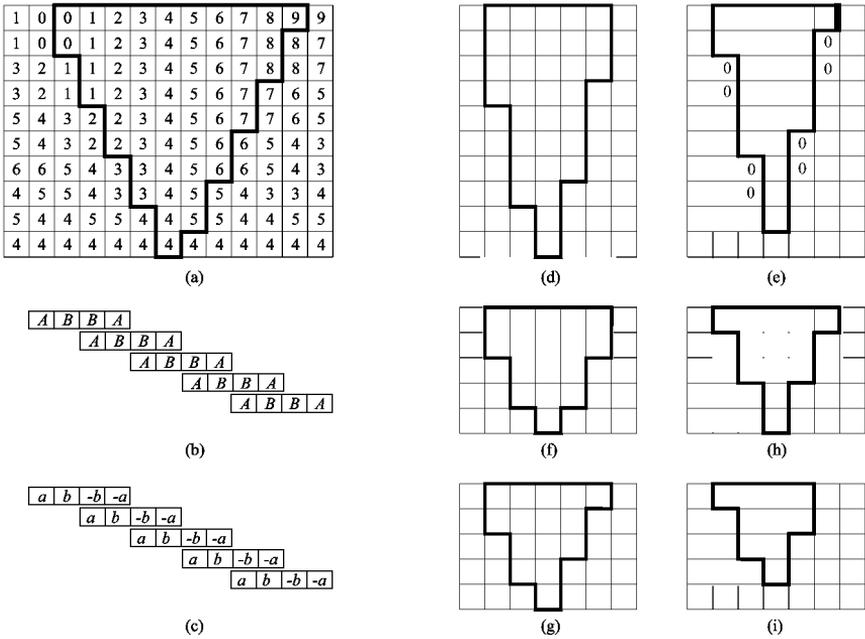


Figure 6.37. Shape adaptive image decomposition using symmetric reflection for even-length two-channel filter banks; see the comments to Figure 6.36 for further explanation.

A scheme for the decomposition of arbitrarily shaped 2-D objects with even-length filters is depicted in Figure 6.37. Note that in this case, the lowpass band grows faster than the highpass band. The shaded regions in Figures 6.37(d) and (e) show the shape of the lowpass and highpass band after horizontal filtering. The brighter regions within the object in Figure 6.37(d) indicate the extra lowpass samples. The zero-marked fields in Figure 6.37(e) are positions where the highpass samples are exactly zero.

If the faster growing of the lowpass band is unwanted the manipulation indicated in Figure 6.35(d) can be applied. Then the subbands obtained with even-length filters will have the same shape as the ones in Figure 6.36.

In addition to the direct use of symmetric reflection, one can optimize the boundary processing schemes in order to achieve better coding properties. Methods for this task have been proposed in [70, 69, 101, 27, 102, 39, 40]. These include the two-band, the more general M -band, and the paraunitary case with non-linear phase filters.

6.10 Transmultiplexers

Transmultiplexers are systems that convert time-division multiplexed (TDM) signals into frequency-division multiplexed (FDM) signals and vice versa [151]. Essentially, these systems are filter banks as shown in Figure 6.38. Contrary to the subband coding filter banks considered so far, the synthesis filter bank is applied first and the analysis filter bank is then used to recover the subband samples $y_k(m)$, which may be understood as components of a TDM signal. At the output of the synthesis filter bank we have an FDM signal where each data stream $y_k(m)$ covers a different frequency band.

The transmission from input i to output k is described by the impulse responses

$$t_{i,k}(m) = q_{i,k}(mM), \tag{6.152}$$

where

$$q_{i,k}(n) = g_i(n) * h_k(n). \tag{6.153}$$

In the noise-free case, perfect reconstruction of the input data with a delay of m_0 samples can be obtained when the following condition holds:

$$t_{i,k}(m) = \delta_{ik} \delta_{mm_0}, \quad i, k = 0, 1, \dots, M - 1. \tag{6.154}$$

Using the notation of modulation matrices these PR conditions may be written as

$$\mathbf{T}(z^M) = \mathbf{H}_m^T(z) \mathbf{G}_m(z) = M z^{-m_0M} \mathbf{I}, \tag{6.155}$$

where the overall transfer matrix depends on z^M . This essentially means that any PR subband coding filter bank yields a PR transmultiplexer if the overall delay is a multiple of M .

Practical problems with transmultiplexers mainly occur due to non-ideal transmission channels. This means that intersymbol interference, crosstalk between different channels, and additive noise need to be considered in the transmultiplexer design. An elaborate discussion of this topic is beyond the scope of this section.

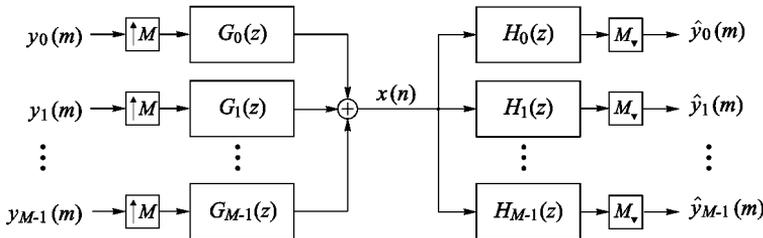


Figure 6.38. Transmultiplexer filter bank.