TIME-VARYING FILTERS

In many applications of digital signal processing it is necessary that different sampling rates coexist within a system. One common example is two systems working at different sampling rates; they have to communicate and the sampling rates have to be made compatible. Another common example is a wideband digital signal that is decomposed into several nonoverlapping narrowband channels in order to be transmitted. In this case, each narrowband channel can have its sampling rate decreased until its Nyquist limit is reached, thereby saving transmission bandwidth.

In this article we will describe such systems. They are generally referred to as *multirate* systems. Most of them have one property in common: They are not shift invariant or they are, at most, periodically shift invariant.

First, we will describe the basic operations of *decimation* and *interpolation* and show how arbitrary rational sampling-rate changes can be implemented using them. Then, we will deal with *filter banks,* showing several ways by which a signal can be decomposed into critically decimated frequency bands, and then be recovered from them with minimum error. Finally, wavelet transforms will be considered. They are a relatively recent development of functional analysis that is arousing great interest in the signal processing community, and their digital implementation can be regarded as a special case of critically decimated filter banks.

DECIMATION, INTERPOLATION, SAMPLING-RATE CHANGES

Intuitively, any sampling-rate change can be effected by recovering the band-limited analog signal $y(t)$ from its samples $x(m)$, and then resampling it with a different sampling rate, thus generating a different digital signal $\bar{x}(n)$. Of course the intermediate analog signal $x(t)$ must be filtered so that it can be resampled without aliasing. As an example, suppose we have a digital signal $x(m)$ that was generated from an analog signal $y(t)$ with sampling period T_1, that is $x(m) = y(mT_1)$, $m = \ldots, 0, 1, 2 \ldots$. It is assumed that $y(t)$ is band limited to $[-\pi/T_1, \pi/T_1]$. Therefore, by replacing each sample of the signal by an impulse proportional to it, we have the equivalent analog signal:

$$y'(t) = \sum_{m=-\infty}^{\infty} x(m)\delta(t - mT_1) \tag{1}$$

Its spectrum is periodic with period $2\pi/T_1$. In order to recover the original analog signal $y(t)$ from $y'(t)$, the repetitions of the spectrum must be discarded. Therefore, $y'(t)$ must be filtered with a filter $h(t)$ whose ideal frequency response $H(j\omega)$ is as follows (1):

$$H(j\omega) = \begin{cases} 1 & \omega \in \left[-\dfrac{\pi}{T_1}, \dfrac{\pi}{T_1}\right] \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

It is then easy to show that (1)

$$y(t) = y'(t) * h(t) = \frac{1}{T_1} \sum_{m=-\infty}^{\infty} x(m) \operatorname{sinc} \frac{\pi}{T_1}(t - mT_1) \tag{3}$$

Then, resampling $y(t)$ with period T_2, generating the digital signal $\bar{x}(n) = y(nT_2)$, $n = \ldots, 0, 1, 2 \ldots$, we have

$$\bar{x}(n) = \frac{1}{T_1} \sum_{m=-\infty}^{\infty} x(m) \operatorname{sinc} \frac{\pi}{T_1}(nT_2 - mT_1) \tag{4}$$

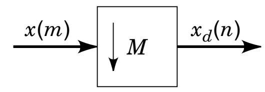

Figure 1. Decimation by a factor of M.

This is the general equation governing sampling-rate changes. Observe that there is no restriction on the values of T_1 and T_2. Of course, if $T_2 > T_1$, and aliasing is to be avoided, the filter in Eq. (2) must have a frequency response equal to zero for $\omega \notin [-\pi/T_2, \pi/T_2]$.

Since this equation consists of infinite summations involving the sinc($\cdot$) function, it is impractical to use it (1). In general, for rational sampling-rate changes, which cover most cases of interest, one can derive expressions working solely on the digital domain. This will be covered in the next three subsections, where three special cases will be considered: decimation by an integer factor M, interpolation by an integer factor L, and sampling-rate change by a rational factor L/M.

Decimation

To *decimate* (or *subsample*) a digital signal $x(m)$ by a factor of M is to reduce its sampling rate M times. It is equivalent to keeping only every Mth sample of the signal. It is represented as in Fig. 1.

The decimated signal is then $x_d(n) = x(nM)$. In the frequency domain, if the spectrum of $x(m)$ is $X(e^{j\omega})$, the spectrum of the subsampled signal, $X_d(e^{j\omega})$, becomes (see Appendix A)

$$X_d(e^{j\omega}) = \frac{1}{M} \sum_{k=0}^{M-1} X(e^{j\frac{\omega - 2\pi k}{M}}) \tag{5}$$

As illustrated in Figs. 2(a) and 2(b) for $M = 2$, Eq. (5) means that the spectrum of $x_d(n)$ is composed of copies of the spectrum of $x(m)$ expanded by M and repeated with period 2π. This implies that, in order to avoid aliasing after subsampling, the bandwidth of the signal $x(m)$ must be limited to the interval $[-\pi/M, \pi/M]$. Therefore, the subsampling operation is generally preceded by a low-pass filter [see Fig. 5(a)], which approximates the following frequency response:

$$H_d(e^{j\omega}) = \begin{cases} 1, & \omega \in \left[-\dfrac{\pi}{M}, \dfrac{\pi}{M}\right] \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

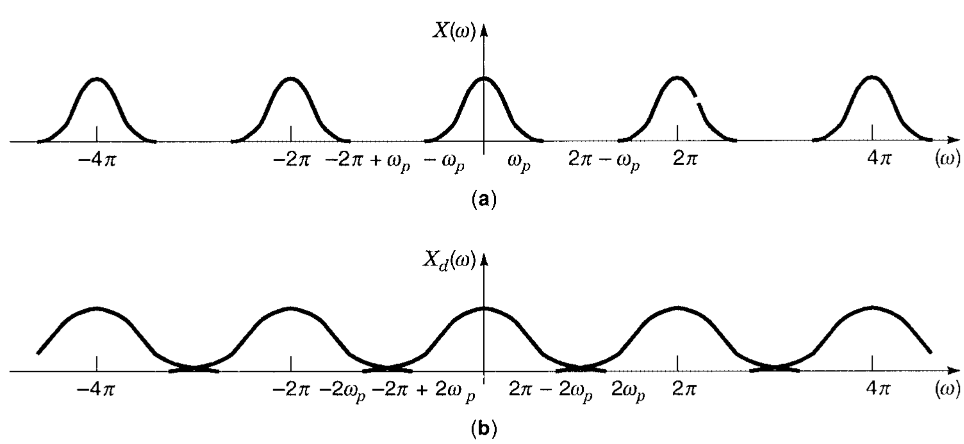

Figure 2. (a) Spectrum of the original signal. (b) Spectrum of the signal decimated by a factor of 2.

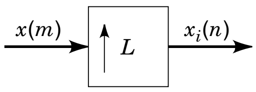

Figure 3. Interpolation by a factor of L.

Some important facts must be noted about the decimation operation:

- It is shift varying, that is, if the input signal $x(m)$ is shifted, the output signal will not generally be a shifted version of the previous output. More precisely, be $\mathscr{D}_M$ the decimation by M operator. If $x_d(n) = \mathscr{D}_M\{x(m)\}$, then in general $\mathscr{D}_M\{x(m-k)\} \neq x_d(n-k)$. However, $\mathscr{D}_M\{x(m-Mk)\} = x_d(n-k)$. Because of this property, the decimation is referred to as a *periodically shift-invariant* operation (2).
- Referring to Fig. 5(a), one can notice that, if the filter $H_d(z)$ is FIR, its outputs need only be computed every M samples, which implies that its implementation complexity is M times smaller than the one of a usual filtering operation. This is not generally valid for IIR filters. One case when this sort of reduction in complexity can be obtained is for IIR filters with transfer function of the type $H(z) = N(z)/D(z^M)$ (2).
- If the frequency range of interest for the signal $x(m)$ is $[-\omega_p, \omega_p]$, with $\omega_p < \pi/M$, one can afford aliasing outside this range. Therefore, the constraints upon the filter can be relaxed, yielding the following specifications for $H_d(z)$ (2):

$$H_d(e^{j\omega}) = \begin{cases} 1, & |\omega| \in [0, \omega_p] \\ 0, & |\omega| \in \left[\dfrac{2\pi k}{M} - \omega_p, \dfrac{2\pi k}{M} + \omega_p\right], k = 1, 2, \ldots, M-1 \end{cases} \tag{7}$$

Interpolation

To *interpolate* (*or upsample*) a digital signal $x(m)$ by a factor of L is to include $L - 1$ zeros between its samples. It is represented as in Fig. (3).

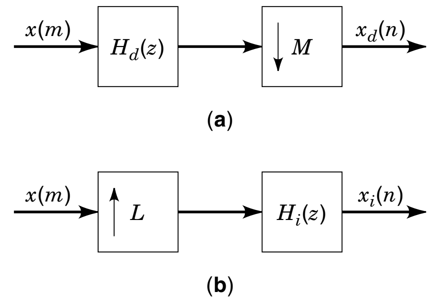

Figure 5. (a) General decimation operation. (b) General interpolation operation.

The interpolated signal is then

$$x_i(n) = \begin{cases} x(n/L), & n = kL, k \in \mathbb{Z} \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

In the frequency domain, if the spectrum of $x(m)$ is $X(e^{j\omega})$, it is straightforward to see that the spectrum of the interpolated signal, $X_i(e^{j\omega})$, becomes (2)

$$X_i(e^{j\omega}) = X(e^{j\omega L}) \tag{9}$$

Figures 4(a) and 4(b) exemplify the spectra of the signals $x(m)$ and $x_i(n)$ for an interpolation factor of L.

Since the digital signal is periodic with period 2π, the interpolated signal will have period $2\pi/L$. Therefore, in order to obtain a smooth interpolated version of $x(m)$, the spectrum of the interpolated signal must have the same shape of the spectrum of $x(m)$. This can be obtained by filtering out the repetitions of the spectra beyond $[-\pi/L, \pi/L]$. Thus, the up-sampling operation is generally followed by a low-pass filter [see Fig. 5(b)] which approximates the following frequency response:

$$H_i(e^{j\omega}) = \begin{cases} L, & \omega \in \left[-\dfrac{\pi}{L}, \dfrac{\pi}{L}\right] \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

Some important facts must be noted about the interpolation operation:

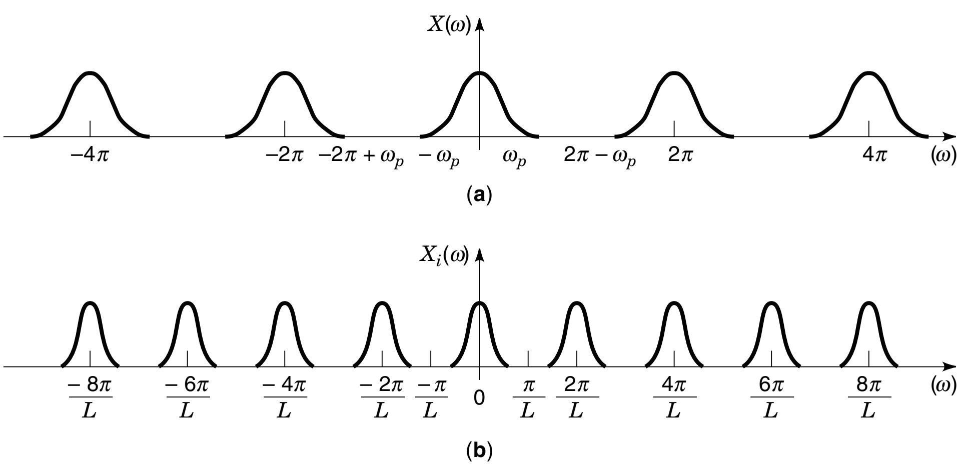

Figure 4. (a) Spectrum of the original signal. (b) Spectrum of the signal after interpolation by L.

- As with the decimation operation, the interpolation is only *periodically* shift invariant. More precisely, if $\mathcal{I}_L$ is the interpolation by L operator, $x_i(n) = \mathcal{I}_L\{x(m)\}$ implies that $\mathcal{I}_L\{x(m-k)\} = x_i(n-kL)$ (2).
- Referring to Fig. 5(b), one can notice that the computation of the output of the filter $H_i(z)$ uses only one out of L samples of the input signal because the remaining samples are zero. This means that its implementation complexity is L times smaller than the one of a usual filtering operation.
- If the signal $x(m)$ is band limited to $[-\omega_p, \omega_p]$, the repetitions of the spectrum will only appear in a neighborhood of radius ω_p/L around the frequencies $2\pi k/L$, $k = 1, 2, \ldots, L-1$. Therefore, the constraints upon the filter can be relaxed as in the decimation case, yielding (2)

$$H_i(e^{j\omega}) = \begin{cases} L, & |\omega| \in \left[0, \dfrac{\omega_p}{L}\right] \\ 0, & |\omega| \in \left[\dfrac{2\pi k - \omega_p}{L}, \dfrac{2\pi k + \omega_p}{L}\right] k = 1, 2, \ldots, L-1 \end{cases} \tag{11}$$

The gain factor L in Eqs. (10) and (11) can be understood by noting that since we are maintaining one out of L samples of the signal, the average energy of the signal decreases by a factor L^2, and therefore the gain of the interpolating filter must be L.

Supposing $L = 2$, two common examples of interpolators are

- $H_i(z) = 1 + z^{-1}$—zero-order hold
- $H_i(z) = \frac{1}{2}(z + 2 + z^{-1})$—linear interpolation

Rational Sampling-Rate Changes

A rational sampling rate change by a factor L/M can be implemented by cascading an interpolator by a factor of L with a decimator by a factor of M. This is exemplified in Fig. 6.

Since $H(z)$ is an interpolation filter, its cutoff frequency must be less than π/L. However, since it is also a decimation filter, its cutoff frequency must be less than π/M. Therefore, it must approximate the following frequency response:

$$H(e^{j\omega}) = \begin{cases} L, & |\omega| \le \min\left\{\dfrac{\pi}{L}, \dfrac{\pi}{M}\right\} \\ 0, & \text{otherwise} \end{cases} \tag{12}$$

Likewise the case of decimation and interpolation, the specifications of $H(z)$ can be relaxed if the bandwidth of the signal is smaller than ω_p. The relaxed specifications are the result of cascading the specifications in Eq. (11), with the specifications in Eq. (7) for ω_p replaced by ω_p/L. Since L and

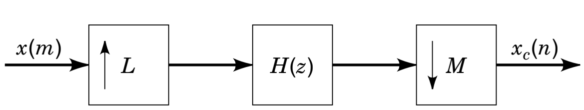

Figure 6. Sampling rate change by a factor of L/M.

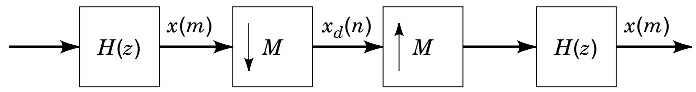

Figure 7. Decimation followed by interpolation.

M can be assumed to be relatively prime, this yields:

$$H(e^{j\omega}) = \begin{cases} L, & |\omega| < \min\left\{\dfrac{\omega_p}{L}, \dfrac{\pi}{M}\right\} \\ 0, & \min\left\{\dfrac{2\pi}{L} - \dfrac{\omega_p}{L}, \dfrac{2\pi}{M} - \dfrac{\omega_p}{L}\right\} \le |\omega| \le \pi \end{cases} \tag{13}$$

Inverse Operations

At this point, a natural question to ask is: are the decimation by M ($\mathcal{D}_M$) and interpolation by M ($\mathcal{I}_M$) operators inverses of each other? In other words, is $\mathcal{D}_M\mathcal{I}_M = \mathcal{I}_M\mathcal{D}_M =$ identity?

It is easy to see that $\mathcal{D}_M\mathcal{I}_M =$ identity, because the $M-1$ zeros between samples inserted by the interpolation operation are removed by the decimation, thereby restoring the original signal. On the other hand, $\mathcal{I}_M\mathcal{D}_M$ is not an identity, since the decimation operation removes $M-1$ out of M samples of the signal and the interpolation operation inserts $M-1$ zeros between samples. Their cascade is equivalent to replacing $M-1$ out of M samples of the signal with zeros, which is obviously not an identity.

However, if the decimation by M operation is preceded by a band-limiting filter from $-\pi/M$ to π/M [Eq. (6)], and the interpolation operation is followed by the same filter, then $\mathcal{I}_M\mathcal{D}_M$ becomes an identity (see Fig. 7). This can be easily seen in the frequency domain. The band-limiting filter prevents aliasing after decimation, and the spectrum of the decimated signal will be in $[-\pi, \pi]$. After interpolation by M, there will be images of the spectrum of the signal in the intervals $[\pi k/M, \pi(k+1)/M]$, $k = -M, -M+1, \ldots M-1$. The band-limiting filter will keep only the image in $[-\pi/M, \pi/M]$, which corresponds to the original signal.

Filter Design Using Interpolation

A very interesting application of interpolation is in filter design. Since the "transition bandwidths" of the interpolated signal are L times smaller than the ones of the original signal, this fact can be used to generate sharp cutoff filters with low computational complexity. A very good example is given by the *frequency masking* approach (3). The process is sketched in Fig. (8), for an interpolation ratio of $L = 4$.

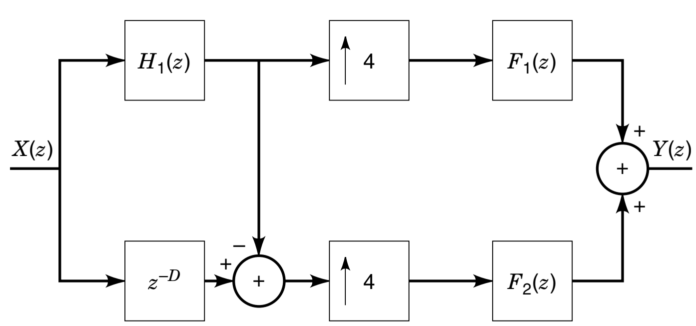

Figure 8. Filter design and implementation using interpolation (frequency masking approach).

The idea is to generate a filter with a transition bandwidth one-fourth that of the prototype filter $H_1(z)$. Initially, a prototype half-band filter $H_1(z)$ is designed with a given transition bandwidth, four times larger than the one needed, and therefore, with implementation complexity much smaller than the one of a direct design (1). From this prototype, its complementary filter $H_2(z)$ is generated by a simple delay and subtraction, that is, $H_2(z) = z^{-D} - H_1(z)$. Their frequency responses are illustrated in Fig. 9(a). After interpolation, their responses are as shown in Fig. 9(b). The filters $F_1(z)$ and $F_2(z)$ serve to select the parts of the interpolated spectrum of $H_1(z)$ and $H_2(z)$ that will be used in composing the desired filter response $F(z) = Y(z)/X(z)$ [see Fig. 9(b)]. It is interesting to note that $F_1(z)$ and $F_2(z)$, besides being interpolation filters, are allowed to have large transition bandwidths, and therefore have low implementation complexity (1). As can be seen from Fig. 9(c), one can generate large bandwidth sharp cutoff filters with low implementation complexity.

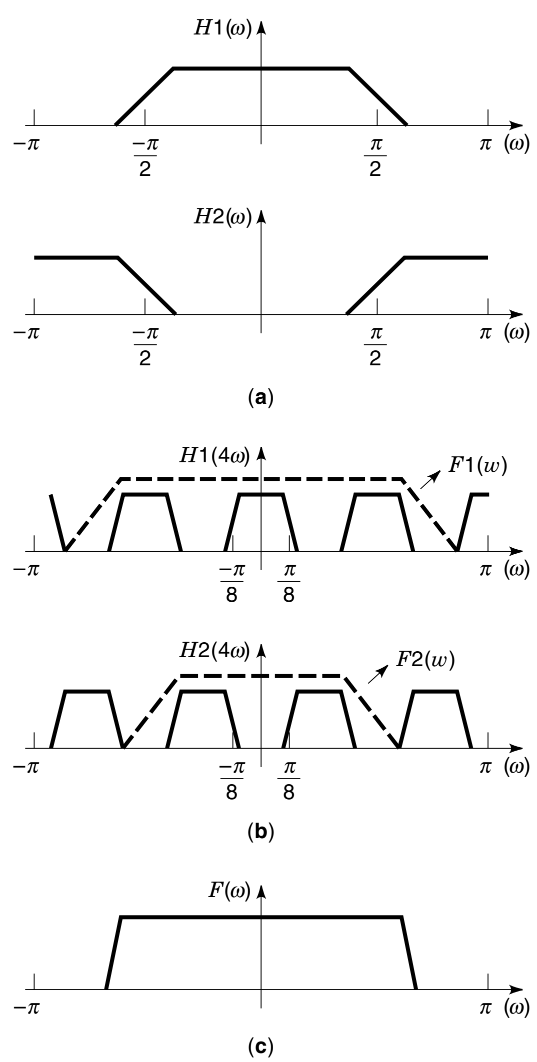

Figure 9. (a) Prototype half-band filter $H_1(z)$ and its complementary $H_2(z)$. (b) Frequency responses of $H_1(z)$ and $H_2(z)$ after interpolation by a factor of 4. Notice the responses of $F_1(z)$ and $F_2(z)$. (c) Frequency response of the equivalent filter $F(z)$.

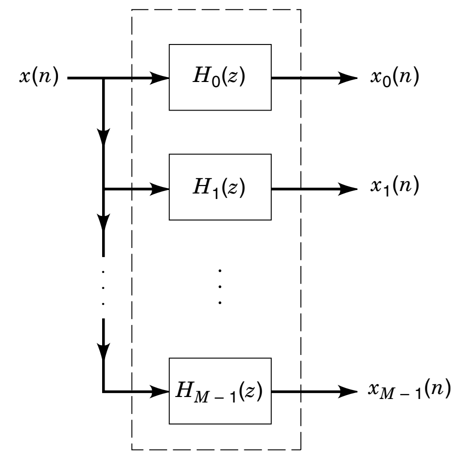

Figure 10. Decomposition of a digital signal into M frequency bands.

FILTER BANKS

In a number of applications, it is necessary to split a digital signal $x(n)$ into several frequency bands, as in Fig. 10.

In this case, each of the bands $x_k(n)$, $k = 0, \ldots, M - 1$, has at least the same number of samples as the original signal $x(n)$. This implies that after the M-band decomposition, the signal is represented with at least M times more samples than the original one. However, there are many cases in which this expansion on the number of samples is highly undesirable. One such case is signal transmission (4), where more samples mean more bandwidth and consequently increased transmission costs.

In the common case where the signal is uniformly split in the frequency domain, that is, each of the frequency bands $x_k(n)$ has the same bandwidth, a natural question to ask is: Since the bandwidth of each band is M times smaller than the one of the original signal, could the bands $x_k(n)$ be decimated by a factor of M without destroying the original information? If this were possible, then one could have a digital signal split into several frequency bands without increasing the overall number of samples. In other words, the question is whether it is possible to recover exactly the original signal from the decimated bands. This section examines several ways to achieve this.

Decimation of a Band-Pass Signal and Its Inverse Operation

Decimation of a Band-Pass Signal. As was seen in the section entitled "Decimation," if the input signal $x(m)$ was low pass and band limited to $[-\pi/M, \pi/M]$, the aliasing after decimation by a factor of M could be avoided [see Eq. (5)]. However, if a signal is split into M uniform frequency bands, at most one band will have its spectrum confined to $[-\pi/M, \pi/M]$. In fact, if a signal is split into M uniform *real* bands, one can say that band $x_k(n)$ will be confined to $[-\pi(k+1)/M, -\pi k/M] \cup [\pi k/M, \pi(k+1)/M]$ [1] (see Fig. 11).

This implies that band k, $k \neq 0$ is not confined to $[-\pi/M, \pi/M]$. However, by examining Eq. (5) one can notice that aliasing is still avoided in this case. The only difference is that, after decimation, the spectrum contained in $[-\pi(k+1)/M, -\pi k/M]$ is mapped to $[0, \pi]$ if k is odd, and to $[-\pi, 0]$ if k is even. Similarly, the spectrum contained in the interval $[\pi k/M, \pi(k+1)/M]$ is mapped to $[-\pi, 0]$ if k is odd and to $[0,$

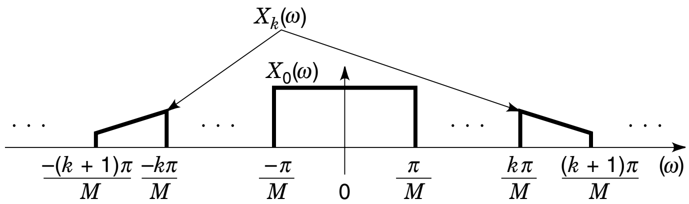

Figure 11. Uniform split of a signal into M bands.

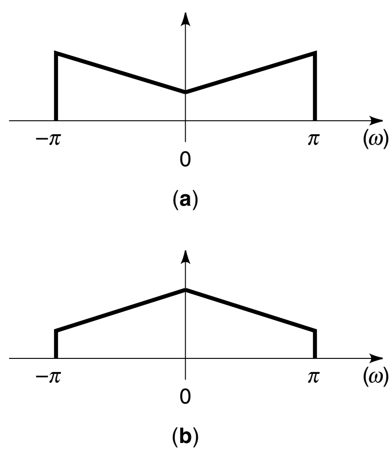

Figure 12. Spectrum of band k decimated by a factor of M: (a) k odd; (b) k even.

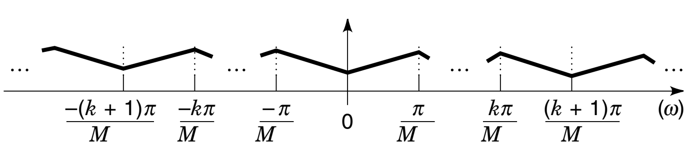

Figure 13. Spectrum of band k after decimation and interpolation by a factor of M for k odd.

π] if k is even (2). Then, the decimated band k will look as in Figs. 12(a) and 12(b) for k odd and even, respectively.

Inverse Operation for Bandpass Signals. We have seen above that a band-pass signal can be decimated by M without aliasing, provided that its spectrum is confined to $[-\pi(k+1)/M, -\pi k/M] \cup [\pi k/M, \pi(k+1)/M]$. The next natural question is: Can the original band-pass signal be recovered from its decimated version by an interpolation operation? The case of low-pass signals was examined in the subsection entitled "Inverse Operations."

The spectrum of a decimated band-pass signal is as in Figs. 12(a) and 12(b) for k odd and even, respectively. After interpolation by M, the spectrum for k odd will be as in Fig. 13.

We want to recover band k as in Fig. 11. From Fig. 13 it is clearly seen that it suffices to keep the region of the spectrum in $[-\pi(k+1)/M, -\pi k/M] \cup [\pi k/M, \pi(k+1)/M]$. The case for k even is entirely analogous. The process of decimating and interpolating a band-pass signal is then very similar to the case of a low-pass signal (see Fig. 7), with the difference that $H(z)$ is a band-pass filter with bandwidth $[-\pi(k+1)/M, -\pi k/M] \cup [\pi k/M, \pi(k+1)/M]$.

Critically Decimated *M*-Band Filter Banks. It is clear that if a signal $x(m)$ is decomposed into M non-overlapping band-pass channels B_k, $k = 0, \ldots,$ and $M - 1$ such that $\bigcup_{k=0}^{M-1} B_k = [-\pi, \pi]$, then it can be recovered from these M channels by just summing them up. However, as conjectured above, exact recovery of the original signal might not be possible if each channel is decimated by M. However, in the section entitled "Decimation of a Band-Pass Signal and Its Inverse Operation," we examined a way to recover the band-pass channel from its subsampled version. All that is needed are interpolation operations followed by filters with passband $[-\pi(k+1)/M, -\pi k/M] \cup [\pi k/M, \pi(k+1)/M]$ (see Fig. 13). This process of decomposing a signal and restoring it from the frequency bands is depicted in Fig. 14. We often refer to it as an *M-band filter bank.* The frequency bands $u_k(n)$ are called *sub-bands.* If the input signal can be recovered exactly from its sub-bands, it is called an M-band *perfect reconstruction* filter bank. Figure 15 details a perfect reconstruction filter bank for the 2-band case.

However, the filters required for the M-band perfect reconstruction filter bank described above are not realizable [see Eqs. (6) and (10)], that is, at best they can be only approximated (1). Therefore, in a first analysis, the original signal

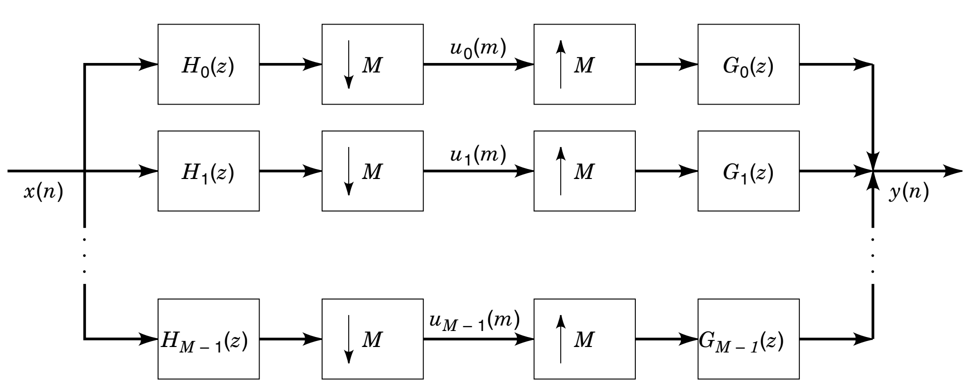

Figure 14. M-band filter bank.

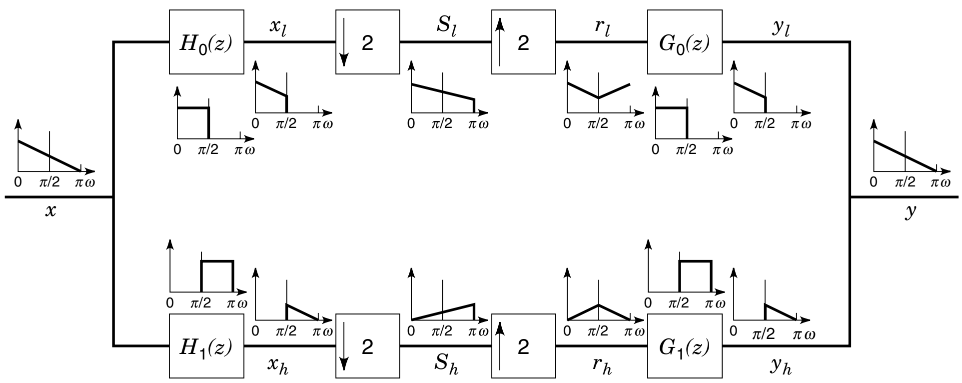

Figure 15. Two-band perfect reconstruction filter bank using ideal filters.

would be only approximately recoverable from its decimated frequency bands. This is well illustrated in Fig. 16, which details a 2-band filter bank using nonideal (or nonrealizable) filters.

In Fig. 16, one can notice that since the filters $H_0(z)$ and $H_1(z)$ are not ideal, the subbands $s_l(m)$ and $s_h(m)$ will have aliasing. In other words, the signals $x_l(n)$ and $x_h(n)$ cannot be respectively recovered from $s_l(m)$ and $s_h(m)$. Nevertheless, by closely examining Fig. 16, one can see that since $y_l(n)$ and $y_h(n)$ are added in order to obtain $y(n)$, the aliased components of $y_l(n)$ will be combined with the ones of $y_h(n)$. Therefore, at least in principle, there exists the possibility that these aliased components cancel out and $y(n)$ is equal to $x(n)$, that is, the original signal can be recovered from its subbands. This is indeed the case, not only for the 2-band but also for the general M-band case (5). In the remainder of this section we will examine methods of designing the *analysis* filters $H_k(z)$ and the *synthesis* filters $G_k(z)$ so that perfect reconstruction can be achieved, or at least arbitrarily approximated.

Perfect Reconstruction

Noble Identities. The noble identities are depicted in Figs. 17(a) and 17(b). They have to do with the commutation of the filtering and decimation or interpolation operations. They are very useful in analyzing multirate systems and filter banks. Their proof can be found in Appendix B.

Polyphase Decompositions. The $\mathcal{Z}$ transform $H(z)$ of a filter $h(n)$ can be written as

$$\begin{aligned} H(z) &= \sum_{k=-\infty}^{+\infty} h(k)z^{-k} = \sum_{l=-\infty}^{+\infty} h(2l)z^{-2l} + z^{-1}\sum_{l=-\infty}^{+\infty} h(2l+1)z^{-2l} \\ &= \sum_{l=-\infty}^{+\infty} h(Ml)z^{-Ml} + z^{-1}\sum_{l=-\infty}^{+\infty} h(Ml+1)z^{-Ml} \\ &\quad + \cdots + z^{-M+1}\sum_{l=-\infty}^{+\infty} h(Ml+M-1)z^{-Ml} \\ &= \sum_{j=0}^{M-1} z^{-j}E_j(z^M) \end{aligned} \tag{14}$$

where $E_j(z) = \sum_{l=-\infty}^{+\infty} h(Ml+j)z^{-l}$ are called *polyphase components* of the filter $H(z)$.

Equation 14 is a *polyphase decomposition* (5) of the filter $H(z)$. In the polyphase decomposition we decompose the filter $H(z)$ into M filters, the first one with every sample of $h(m)$ whose indexes are multiples of M, the second one with every sample of $h(m)$ whose indexes are one plus a multiple of M, and so on. Using a polyphase decomposition, filtering followed by decimation can be represented as in Fig. 18(b). Applying the noble identities to this figure, we arrive at Fig. 18(c) (5).

Figure 18(c) provides an interesting and useful interpretation to the operation of filtering followed by decimation. Figure 18(c) means that this operation is equivalent to filtering the samples of $x(m)$, whose indexes are equal to an integer k plus a multiple of M with a filter composed by only the samples of $h(m)$ whose indexes are equal to the same integer k plus a multiple of M, for $k = 0, \ldots M-1$.

The polyphase decompositions also provide useful insights into the interpolation operation, but in these cases an alterna-

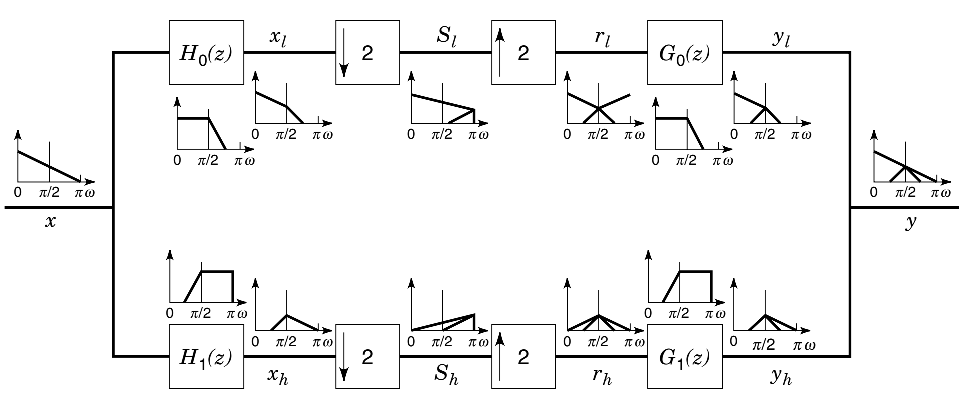

Figure 16. Two-band filter bank using realizable filters.

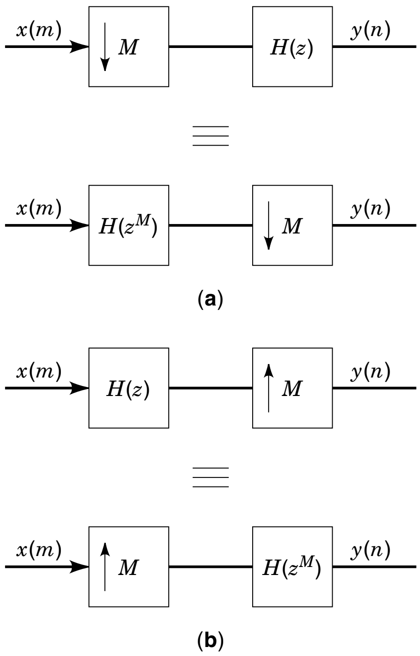

Figure 17. Noble identities. (a) Decimation. (b) Interpolation.

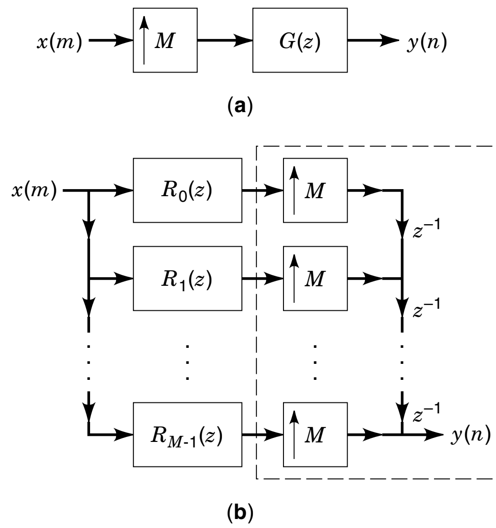

Figure 19. (a) Interpolation by a factor of M. (b) Interpolation using polyphase decompositions and the noble identities.

tive to Eq. (14) is usually employed. In it, we define $R_j(z) = E_{M-1-j}(z)$, and the polyphase decomposition becomes:

$$H(z) = \sum_{j=0}^{M-1} z^{-(M-1-j)} R_j(z^M) \tag{15}$$

Based in Eq. (15), interpolation followed by filtering can be represented in a manner analogous to the one in Fig. 18(c), as depicted in Fig. 19(b) (5).

Commutator Models. The operations described in Figs. 18(c) and 19(b) can also be interpreted in terms of rotary switches. These interpretations are referred to as *commutator models.* In them, the decimators and delays are replaced by rotary switches as depicted in Figs. 20(a) and 20(b) (5).

In Fig. 20(a), the model with decimators and delays is noncausal, having "advances" instead of delays. In causal systems, the causal model of Fig. 21 is preferred.

***M*-Band Filter Banks in Terms of the Filters' Polyphase Components.** By substituting each of the filters $H_k(z)$ and $G_k(z)$ by its polyphase components, the M-band filter bank of Fig. 14 becomes as in Fig. 22(a). The matrices $E(z)$ and $R(z)$ are formed from the polyphase components of $H_k(z)$ and $G_k(z)$. $E_{ij}(z)$ is the jth polyphase component of $H_i(z)$ [see Eq. (14)] and $R_{uv}(z)$ is the uth polyphase component of $G_v(z)$ [see Eq. (15)] (5). In Fig. 22(b), the noble identities were applied.

Perfect Reconstruction *M*-Band Filter Banks. In Fig. 22(b), if $R(z)E(z) = I$, where I is the identity matrix, the M-band filter bank becomes as in Fig. 23.

By substituting the decimators and interpolators in Fig. 23 by the commutator models of Figs. 21 and 20(b), respectively, we arrive at the scheme depicted in Fig. 24, which is clearly equivalent to a pure delay. Therefore, the condition

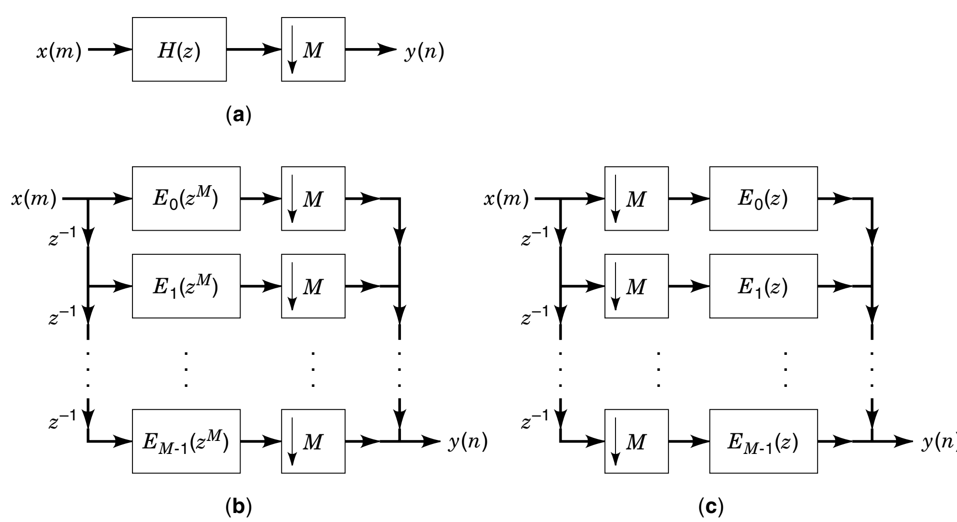

Figure 18. (a) Decimation by a factor of M. (b) Decimation using polyphase decompositions. (c) Decimation using polyphase decompositions and the noble identities.

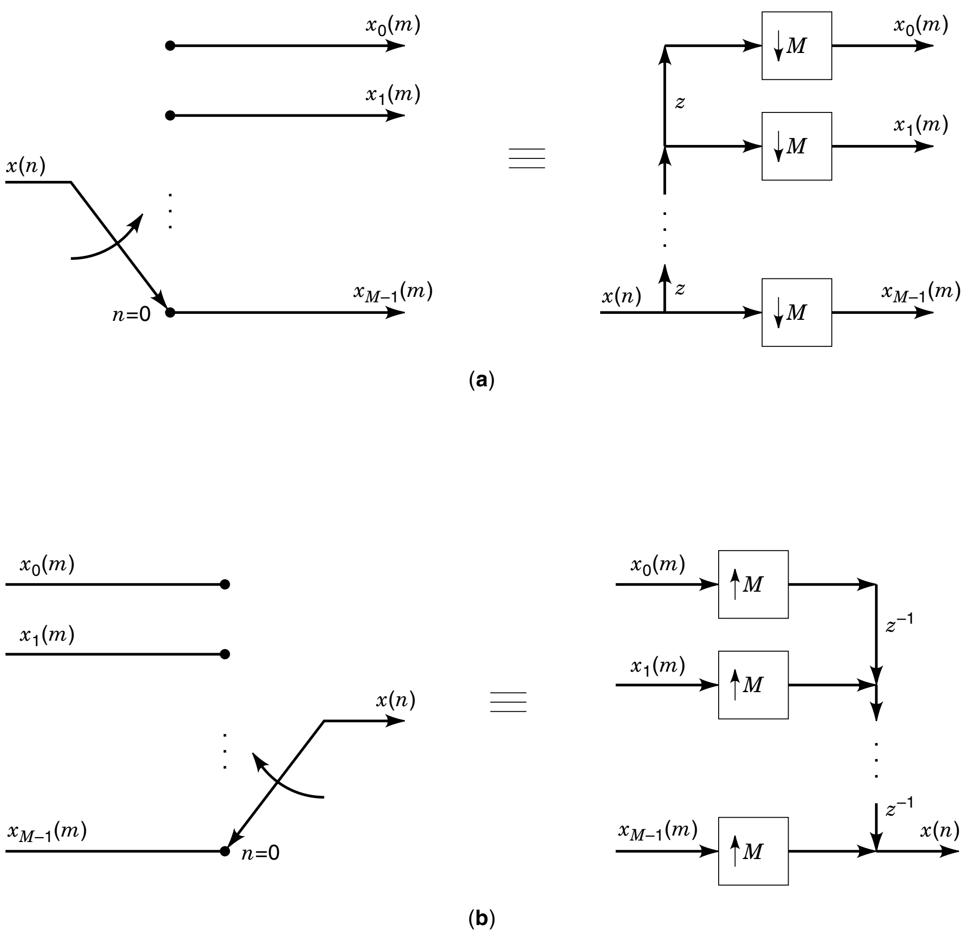

Figure 20. Commutator models for (a) Decimation. (b) Interpolation.

$R(z)E(z) = I$ guarantees perfect reconstruction for the M-band filter bank (5). It should be noted that if $R(z)E(z)$ is equal to the identity times a pure delay, perfect reconstruction still holds. Therefore, the weaker condition $R(z)E(z) = Z^{-\Delta}I$ is sufficient for perfect reconstruction.

Example. Two-Band Perfect Reconstruction Filter Bank Be $M = 2$, and

$$E(z) = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ 1 & -1 \end{pmatrix} \tag{16}$$

$$R(z) = \begin{pmatrix} 1 & \frac{1}{2} \\ 1 & -\frac{1}{2} \end{pmatrix} \tag{17}$$

Clearly $R(z)E(z) = I$. The polyphase components $E_{ij}(z)$ of the analysis filters $H_i(z)$, and $R_{uv}(z)$ of the synthesis filters $G_v(z)$ are then:

$$E_{00}(z) = \frac{1}{2} \quad E_{01}(z) = \frac{1}{2} \quad E_{10}(z) = 1 \quad E_{11}(z) = -1 \tag{18}$$

$$R_{00}(z) = 1 \quad R_{01}(z) = \frac{1}{2} \quad R_{10}(z) = 1 \quad R_{11}(z) = -\frac{1}{2} \tag{19}$$

Then, from Eqs. (19) and (14) we can find the $H_k(z)$, and from Eqs. (18) and (15) we can find the $G_k(z)$. They are

$$H_0(z) = \frac{1}{2}(1 + z^{-1}) \tag{20}$$

$$H_1(z) = 1 - z^{-1} \tag{21}$$

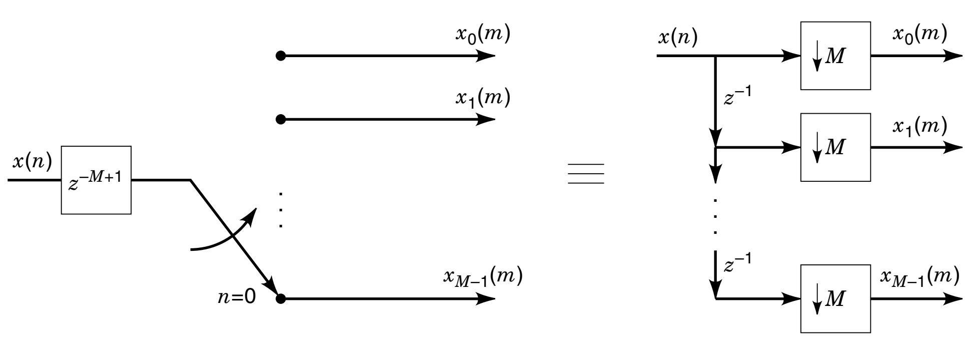

Figure 21. Causal commutator model for decimation.

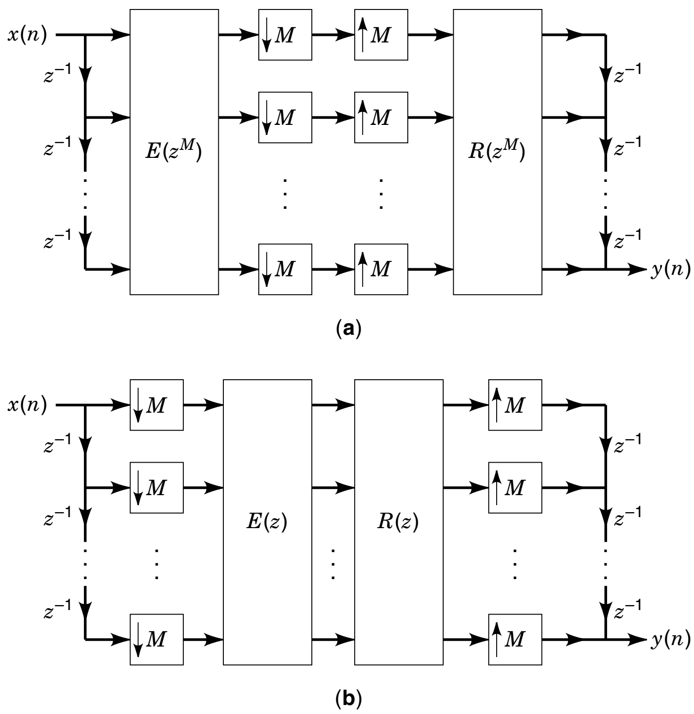

Figure 22. (a) M-band filter bank in terms of the polyphase components of the filters. (b) Same as in (a) but after application of the noble identities.

$$G_0(z) = 1 + z^{-1} \tag{22}$$

$$G_1(z) = -\frac{1}{2}(1 - z^{-1}) \tag{23}$$

This is known as the Haar filter bank. The normalized frequency responses of $H_0(z)$ and $H_1(z)$ are depicted in Fig. 25. One can see that perfect reconstruction could be achieved with filters that are far from being ideal. In other words, even though each subband is highly aliased, one can still recover the original signal exactly at the output.

Transmultiplexers. If two identical M-channel perfect reconstruction filter banks as in Fig. 14 are cascaded, we have that the signal corresponding to $u_k(m)$ in one filter bank is a delayed version of the corresponding signal in the other filter bank, for each $k = 0, \ldots, M - 1$. Therefore, with the same filters as in Fig. 14, one can construct a perfect reconstruction *transmultiplexer* as in Fig. 26, which can combine the M signals $u_k(m)$ into one single signal $y(n)$ and then recover the signals $v_k(m)$, which are just delayed versions of the $u_k(m)$ (5).

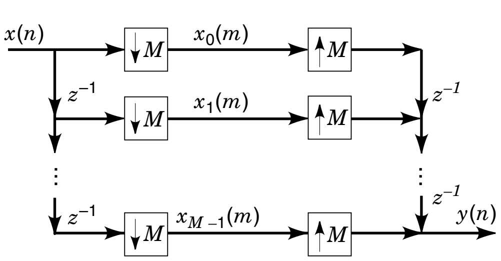

Figure 23. M-band filter bank when $R(z)E(z) = I$.

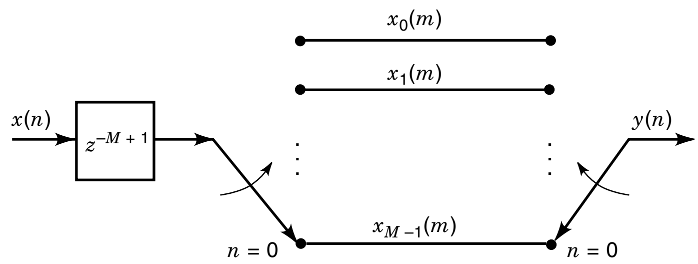

Figure 24. The commutator model of an M-band filter bank when $R(z)E(z) = \mathrm{I}$ is equivalent to a pure delay.

What is very interesting in this case is that the filters need not be selective at all in order for this kind of transmultiplexer to work (see Fig. 26).

Two-Band Perfect-Reconstruction Filter Banks. The two-band case is as seen in Fig. 27.

Representing the filters $H_0(z)$, $H_1(z)$, $G_0(z)$, and $G_1(z)$ in terms of their polyphase components [Eqs. (14) and (15)], we have

$$H_0(z) = H_{00}(z^2) + z^{-1}H_{01}(z^2) \tag{24}$$

$$H_1(z) = H_{10}(z^2) + z^{-1}H_{11}(z^2) \tag{25}$$

$$G_0(z) = z^{-1}G_{00}(z^2) + G_{01}(z^2) \tag{26}$$

$$G_1(z) = z^{-1}G_{10}(z^2) + G_{11}(z^2) \tag{27}$$

The matrices $E(z)$ and $R(z)$ in Fig. 22(b) are then

$$E(z) = \begin{pmatrix} H_{00}(z) & H_{01}(z) \\ H_{10}(z) & H_{11}(z) \end{pmatrix} \quad R(z) = \begin{pmatrix} G_{00}(z) & G_{10}(z) \\ G_{01}(z) & G_{11}(z) \end{pmatrix} \tag{28}$$

If $R(z)E(z) = I$ we have perfect reconstruction (Figs. 23 and 24). In fact, from Fig. 24, we see that the output signal will be delayed by $M - 1 = 1$ sample. In the general case, one can have $R(z)E(z) = Iz^{-\Delta}$, which makes the output signal to be delayed by $\Delta + 1$. Therefore, the 2-band filter bank will be equivalent to a delay of $\Delta + 1$ samples if

$$R(z) = z^{-\Delta}E^{-1}(z) \tag{29}$$

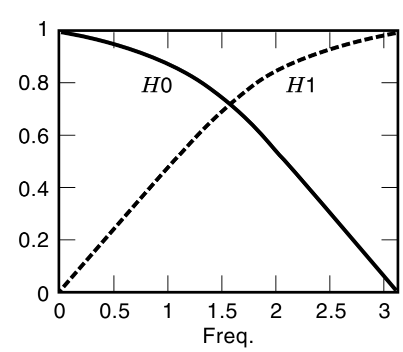

Figure 25. Normalized frequency responses of the filters described by Eqs. (20) and (21).

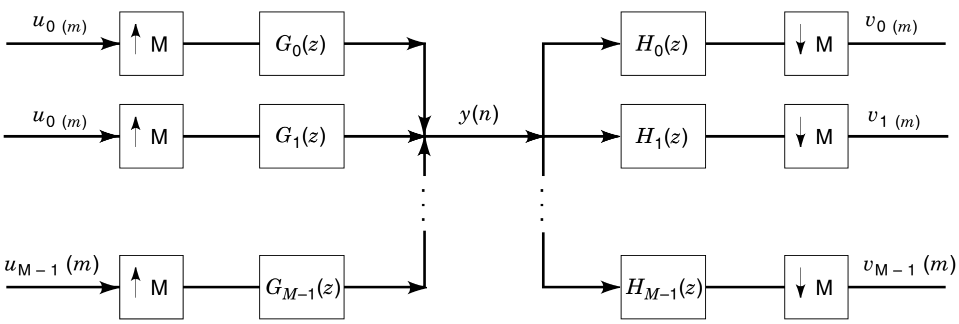

Figure 26. *M*-band transmultiplexer.

From Eq. (28), this implies that

$$\begin{pmatrix} G_{00}(z) & G_{10}(z) \\ G_{01}(z) & G_{11}(z) \end{pmatrix} = \frac{z^{-\Delta}}{H_{00}(z)H_{11}(z) - H_{01}(z)H_{10}(z)} \begin{pmatrix} H_{11}(z) & -H_{01}(z) \\ -H_{10}(z) & H_{00}(z) \end{pmatrix} \tag{30}$$

Equation (30) is enough for IIR filter design, as long as stability constraints are taken into consideration. However, if we want the filters to be FIR, as is often the case, the term in the denominator must be a pure delay. Therefore,

$$H_{00}(z)H_{11}(z) - H_{01}(z)H_{10}(z) = cz^{-l} \tag{31}$$

From Eqs. (24–27) we can express the polyphase components in terms of the filters $H_k(z)$ and $G_k(z)$ as

$$\begin{aligned} H_{00}(z^2) &= \frac{H_0(z)+H_0(-z)}{2} & H_{01}(z^2) &= \frac{H_0(z)-H_0(-z)}{2z^{-1}} \\ H_{10}(z^2) &= \frac{H_1(z)+H_1(-z)}{2} & H_{11}(z^2) &= \frac{H_1(z)-H_1(-z)}{2z^{-1}} \end{aligned} \tag{32}$$

$$\begin{aligned} G_{00}(z^2) &= \frac{G_0(z)-G_0(-z)}{2z^{-1}} & G_{01}(z^2) &= \frac{G_0(z)+G_0(-z)}{2} \\ G_{10}(z^2) &= \frac{G_1(z)-G_1(-z)}{2z^{-1}} & G_{11}(z^2) &= \frac{G_1(z)+G_1(-z)}{2} \end{aligned} \tag{33}$$

Substituting Eq. (32) into Eq. (31), we have that

$$H_0(-z)H_1(z) - H_0(z)H_1(-z) = 2cz^{-2l-1} \tag{34}$$

Now, substituting Eq. (31) into Eq. (30), and computing the $G_k(z)$ from Eqs. (26) and (27), we arrive at

$$G_0(z) = -\frac{z^{2(l-\Delta)}}{c}H_1(-z) \tag{35}$$

$$G_1(z) = \frac{z^{2(l-\Delta)}}{c}H_0(-z) \tag{36}$$

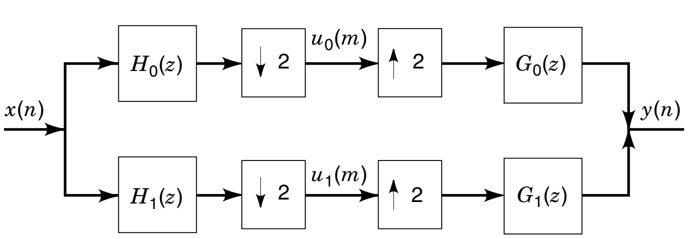

Figure 27. Two-band filter bank.

Equations (34–36) suggest a possible way to design 2-band perfect reconstruction filter banks. The design procedure is as follows (4):

1. Find a polynomial $P(z)$ such that $P(z) - P(-z) = 2z^{-2l-1}$.
2. Factorize $P(z)$ into two factors, $H_0(z)$ and $H_1(-z)$. Care must be taken in order that $H_0(z)$ and $H_1(-z)$ be lowpass.
3. Find $G_0(z)$ and $G_1(z)$ using Eqs. (35) and (36).

Some important points should be noted in this case:

- If one wants the filter bank to be composed of linear phase filters, it suffices to find a linear phase product filter $P(z)$, and make linear phase factorizations of it.
- If the delay Δ is zero, some of the filters will certainly be noncausal: for l negative, either $H_0(z)$ or $H_1(z)$ must be noncausal [see Eq. (34)]; for l positive, either $G_0(z)$ or $G_1(z)$ must be noncausal. Therefore, a causal perfect reconstruction filter bank will always have nonzero delay.
- The magnitudes of the frequency responses, $|G_0(e^{j\omega})|$ and $|H_1(e^{j\omega})|$, are mirror images of each other around $\omega = \pi/2$ [Eq. (35)], the same happening to $|H_0(e^{j\omega})|$ and $|G_1(e^{j\omega})|$ [Eq. (36)].

Design Examples. One product filter $P(z)$ satisfying $P(z) - P(-z) = 2z^{-2l-1}$ is

$$\begin{aligned} P(z) &= \frac{1}{16}(-1 + 9z^{-2} + 16z^{-3} + 9z^{-4} - z^{-5}) \\ &= \frac{1}{16}(1+z^{-1})^4(-1+4z^{-1}-z^{-2}) \end{aligned} \tag{37}$$

We can see from its frequency response in Fig. 28(a) that $P(z)$ is a low-pass filter.

One possible factorization of $P(z)$ results in the following filter bank [Eqs. (35) and (36)], a popular symmetric short length filter (4):

$$H_0(z) = \frac{1}{8}(-1 + 2z^{-1} + 6z^{-2} + 2z^{-3} - z^{-4}) \tag{38}$$

$$G_0(z) = \frac{1}{2}(1 + 2z^{-1} + z^{-2}) \tag{39}$$

$$H_1(z) = \frac{1}{2}(1 - 2z^{-1} + z^{-2}) \tag{40}$$

$$G_1(z) = \frac{1}{8}(1 + 2z^{-1} - 6z^{-2} + 2z^{-3} + z^{-4}) \tag{41}$$

Their frequency responses are depicted in Fig. 28(b).

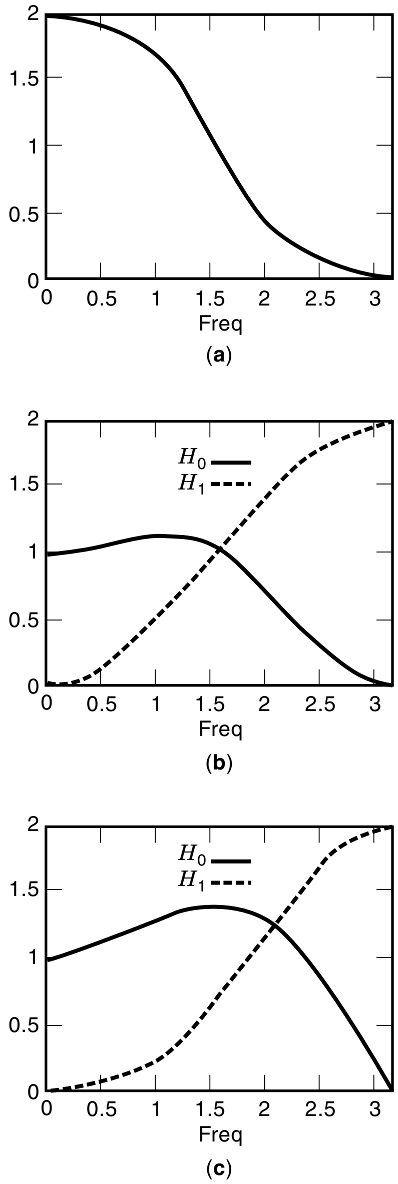

Figure 28. Frequency responses of (a) $P(z)$ from Eq. (37); (b) H_0 and $H_1(z)$ from the factorizations in Eqs. (38) and (40); (c) H_0 and $H_1(z)$ from the factorizations in Eqs. (42) and (44).

Another possible factorization is as follows:

$$H_0(z) = \frac{1}{4}(-1 + 3z^{-1} + 3z^{-2} - z^{-3}) \tag{42}$$

$$G_0(z) = \frac{1}{4}(1 + 3z^{-1} + 3z^{-2} + z^{-3}) \tag{43}$$

$$H_1(z) = \frac{1}{4}(1 - 3z^{-1} + 3z^{-2} - z^{-3}) \tag{44}$$

$$G_1(z) = \frac{1}{4}(1 + 3z^{-1} - 3z^{-2} - z^{-3}) \tag{45}$$

Their frequency responses are depicted in Fig. 28(c).

In what follows we will examine some particular cases of filter bank design that have been widely used in many classes of applications.

Particular Cases of Filter Bank Design

In the section entitled "Perfect Reconstruction," we examine general conditions to design perfect reconstruction filter banks. In the remainder of this section we will analyze some specific filter bank types which have been used a great deal in practice.

Quadrature Mirror Filter Banks. An early, proposed approach to the design of 2-band filter banks is the so-called quadrature mirror filter bank (QMF) (6), where the analysis high-pass filter is designed by alternating the signs of the low-pass filter impulse response samples, that is

$$H_1(z) = H_0(-z) \tag{46}$$

where we are assuming the filters have real coefficients. For this choice of the analysis filter bank, the magnitude response of the high-pass filter ($|H_1(e^{j\omega})|$) is the mirror image of the low-pass filter magnitude response ($|H_0(e^{j\omega})|$) with respect to the quadrature frequency $\pi/2$. This is the origin of the QMF nomenclature.

The analysis of the 2-band filter bank illustrated in Fig. 27 can be alternatively made as follows. The signals after the analysis filter are described by

$$X_k(z) = H_k(z)X(z) \tag{47}$$

for $k = 0, 1$. The decimated signals are

$$U_k(z) = \frac{1}{2}[X_k(z^{1/2}) + X_k(-z^{1/2})] \tag{48}$$

for $k = 0, 1$, whereas the signal after the interpolator are

$$\begin{aligned} U_k(z^2) &= \frac{1}{2}[X_k(z) + X_k(-z)] \\ &= \frac{1}{2}[H_k(z)X(z) + H_k(-z)X(-z)] \end{aligned} \tag{49}$$

Then, the reconstructed signal is represented as

$$\begin{aligned} Y(z) &= G_0(z)U_0(z^2) + G_1(z)U_1(z^2) \\ &= \frac{1}{2}[H_0(z)G_0(z) + H_1(z)G_1(z)]X(z) \\ &\quad + \frac{1}{2}[H_0(-z)G_0(z) + H_1(-z)G_1(z)]X(-z) \\ &= \frac{1}{2}\begin{pmatrix} X(z) & X(-z) \end{pmatrix} \begin{pmatrix} H_0(z) & H_1(z) \\ H_0(-z) & H_1(-z) \end{pmatrix} \begin{pmatrix} G_0(z) \\ G_1(-z) \end{pmatrix} \end{aligned} \tag{50}$$

The last equality represents the so called modulation matrices representation of a two-band filter bank. The aliasing effect is represented by the terms containing $X(-z)$. A possible solution to avoid aliasing is to choose the synthesis filters as follows

$$G_0(z) = H_1(-z) \tag{51}$$

$$G_1(z) = -H_0(-z) \tag{52}$$

Note that this choice keeps the desired features of $G_0(z)$ and $G_1(z)$ being low-pass and high-pass filters, respectively. Also, the alias is now canceled by the synthesis filters instead of being totally avoided by analysis filters, relieving the specifications of the latter filters (see the subsection entitled "Maximally Decimated M-Band Filter Banks").

The overall transfer function of the filter bank after the alias component is eliminated is given by

$$\begin{aligned} H(z) &= \frac{1}{2}\,[H_0(z)G_0(z) + H_1(z)G_1(z)] \\ &= \frac{1}{2}\,[H_0(z)H_1(-z) - H_1(z)H_0(-z)] \end{aligned} \tag{53}$$

where in the last equality we employed the alias elimination constraint in Eq. (52).

In the original QMF design, the alias elimination condition is combined with the alternating-sign choice for the high-pass filter of Eq. (46). In this case the overall transfer function is given by

$$\begin{aligned} H(z) &= \frac{1}{2}\,[H_0^2(z) - H_0^2(-z)] \\ &= 2z^{-1}[E_0(z^2)E_1(z^2)] \end{aligned} \tag{54}$$

Note that the QMF design approach of two-band filter banks consists of designing the low-pass filter $H_0(z)$. The above equation also shows that perfect reconstruction is achievable only if the polyphase components of the low-pass filter ($E_0(z)$ and $E_1(z)$) are simple delays. This limits the selectivity of the generated filters. As an alternative, we can choose $H_0(z)$ to be an FIR linear-phase, low-pass filter, and eliminate any phase distortion of the overall transfer function $H(z)$, which in this case also has linear phase.

In this case, the filter bank transfer function can be written as

$$\begin{aligned} H(e^{j\omega}) &= \frac{e^{-j\omega N}}{2}[|H_0(e^{j\omega})|^2 + |H_0(e^{j(\omega-\pi)})|^2] \\ &= \frac{e^{-j\omega N}}{2}[|H_0(e^{j\omega})|^2 + |H_1(e^{j\omega})|^2] \end{aligned} \tag{55}$$

for N odd. For N even the sum becomes a subtraction, generating an undesirable zero at $\omega = \pi/2$.

The design procedure consists of minimizing the following objective function using an optimization algorithm

$$\begin{aligned} \xi = \xi_1 + \xi_2 &= \delta \int_{\omega_s}^{\pi} |H_0(e^{j\omega})|^2\,d\omega \\ &+ (1-\delta)\int_0^{\pi} \left| H(e^{j\omega}) - \frac{e^{-j\omega N}}{2} \right|^2 d\omega \end{aligned} \tag{56}$$

where ω_s is the stopband edge, usually chosen a bit above 0.5π. The parameter δ provides weighting between the stopband attenuation of the low-pass filter and the amplitude distortion of the filter bank, with $0 < \delta < 1$. Although this objective function has local minima, a good starting point for the coefficients of the low-pass filter and an adequate nonlinear optimization algorithm lead to good results, that is, filter banks with low amplitude distortions and good selectivity of the filters. Usually, a simple window-based design provides a good starting point for the low-pass filter. In any case, the simplicity of the QMF design makes it widely used in practice. Figures 29(a) and 29(b) depict the magnitude responses of the analysis filters of order 15 of a QMF design, along with the amplitude response of the whole filter bank.

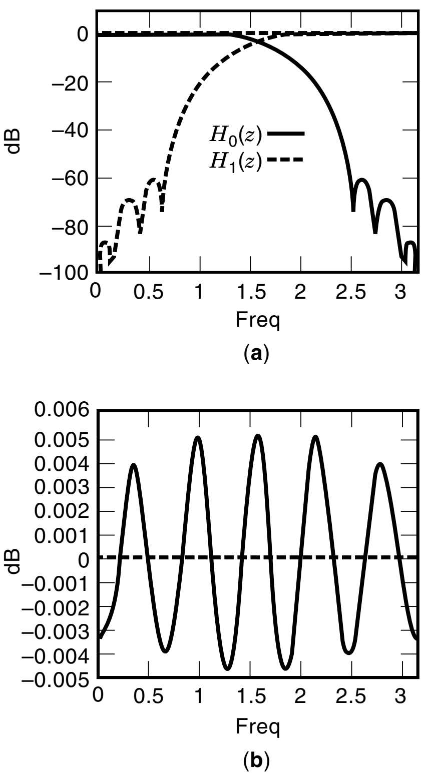

Figure 29. Johnston's QMF design of order 15. (a) Magnitude responses. (b) Overall amplitude response.

The nomenclature QMF filter banks is also used to denote M-channel maximally decimated filter banks. For M-channel QMF filter banks there are two design approaches that are widely used, namely the perfect reconstruction QMF filter banks and the pseudo-QMF filter banks. The perfect reconstruction QMF designs require the use of sophisticated nonlinear optimization programs because the objective function is a nonlinear function of the filter parameters. In particular for a large number of subbands, the number of parameters is usually large. On the other hand, the pseudo-QMF designs consist of designing a prototype filter, with the subfilters of the analysis bank being obtained by the modulation of the prototype. As a consequence, the pseudo-QMF filter has a very efficient design procedure. However, only recently it was discovered that the modulated filter banks could achieve perfect reconstruction. The pseudo-QMF filter banks are also known as cosine-modulated filter banks, since they are designed by applying cosine modulation to a low-pass prototype filter (see the section entitled "Cosine-Modulated Filter Banks").

Conjugate Quadrature Filter Banks. In the QMF design, it was noted that designing the high-pass filter from the low-pass prototype by alternating the signs of its impulse response is rather simple, but the possibility of getting perfect reconstruction is lost except for trivial designs. In a later stage of development (see Ref. 7), it was discovered that by time-reversing the impulse response and alternating the signs of the low-pass filter, we can design perfect reconstruction filter banks with selective subfilters. The resulting filters are called conjugate quadrature (CQF) filter banks.

Therefore, in the CQF design, we have that the analysis high-pass filter is given by:

$$H_1(z) = -z^{-N} H_0(-z^{-1}) \tag{57}$$

By verifying again that N must be odd, the filter bank transfer function is given by

$$\begin{aligned} H(e^{j\omega}) &= \frac{e^{-j\omega N}}{2}[H_0(e^{j\omega})H_0(e^{-j\omega}) + H_0(-e^{-j\omega})H_0(-e^{-j\omega})] \\ &= \frac{e^{-j\omega N}}{2}[P(e^{j\omega}) + P(-e^{j\omega})] \end{aligned} \tag{58}$$

From Eqs. (35–36) we have that, in order to guarantee perfect reconstruction, the synthesis filters should be given by

$$G_0(z) = z^{-N} H_0(z^{-1}) \tag{59}$$

$$G_1(z) = -H_0(-z) \tag{60}$$

Perfect reconstruction is equivalent to having the time-domain response of the filter bank equal to a delayed impulse, that is,

$$h(n) = \delta(n - \Delta) \tag{61}$$

Now by examining $H(e^{j\omega})$ in Eq. (58), one can easily infer that the time-domain representation of $P(z)$ satisfies

$$p(n)[1 + (-1)^n] = 2\delta[n - (\Delta - N)] \tag{62}$$

Therefore, the design procedure consists of the following steps:

- By noting that $p(n) = 0$ for odd n except for $n = N$, we can start by designing a half-band filter of order $2N$, specifically a filter whose average value of the passband and stopband edges is equal to $\pi/2$ (that is $\omega_p + \omega_s/2 = \pi/2$) and has the same ripple (δ_{hb}) in the passband and stopband. In this case, the resulting half-band filter will have zero samples on its impulse response for n odd. This half-band filter can be designed by using a standard minimax approach for FIR filters. However, since the product filter $P(e^{j\omega})$ has to be positive, we should add ($\delta_{hb}/2$) to the frequency response of the half-band filter in order to generate $P(e^{j\omega})$. The stopband attenuation of the half-band filter should be at least twice the desired stopband attenuation of the low-pass filter plus 6 decibels (5).
- If one wants the filter bank to be composed of linear phase filters, it suffices to find a linear phase product filter $P(z)$, and make linear phase factorizations of it (see the subsection entitled "Two-Band Perfect-Reconstruction Filter Banks"). For this case, we will obtain the trivial linear phase filters described in Eqs. (20–23), that show very little selectivity, as shown in Fig. 25.
- The usual approach is to decompose $P(z)$ such that $H_0(z)$ has either near linear phase or has minimum phase. In order to obtain near linear phase, one can select the zeros of $H_0(z)$ to be alternatively from inside and outside the unit circle as frequency is increased. Minimum phase is obtained when all zeros are either inside or on the unit circle of the $\mathcal{Z}$ plane.

Cosine-Modulated Filter Banks. The cosine-modulated filter banks are an attractive choice for the design and implementation of filter banks with a large number of sub-bands. Their main features are:

1. ease of design. It consists essentially of generating a low-pass prototype whose impulse response satisfies some constraints required to achieve perfect reconstruction;
2. low cost of implementation, measured in terms of multiplication count, since the resulting analysis and synthesis filter banks rely on the discrete cosine transform (DCT), which is amenable to fast implementation and can share the prototype implementation cost with each subfilter.

In the cosine-modulated filter bank design, we begin by finding a linear phase prototype low-pass filter $H(z)$ whose passband edge is $2\pi/M - \rho$ and the stop-band edge is $2\pi/M + \rho$, 2ρ being the transition band. For convenience, we assume that the length of the prototype filter is an even multiple of the number M of subbands, that is, $N = 2LM$. Although the actual length of the prototype can be arbitrary, this assumption greatly simplifies our analysis.

Given the prototype filter, we generate cosine-modulated versions of it in order to obtain the analysis and synthesis filter banks as follows:

$$h_l(n) = 2h(n)\cos\left[(2l+1)\frac{\pi}{2M}\left(n - \frac{N-1}{2}\right) + (-1)^l\frac{\pi}{4}\right] \tag{63}$$

$$g_l(n) = 2h(n)\cos\left[(2l+1)\frac{\pi}{2M}\left(n - \frac{N-1}{2}\right) - (-1)^l\frac{\pi}{4}\right] \tag{64}$$

for $0 \le n \le N-1$ and $0 \le l \le M-1$. We should notice that in Eq. (63), the term that multiplies $2h(n)$ represents the (l, n) element of the DCT matrix, $c_{l,n}$.

The prototype filter can be represented on its polyphase decomposition as follows

$$H(z) = \sum_{l=0}^{L-1}\sum_{j=0}^{2M-1} h(2lM + j)z^{-(2lM+j)} = \sum_{j=0}^{2M-1} z^{-j}E_j(z^{2M}) \tag{65}$$

where $E_j(z) = \sum_{l=0}^{L-1} h(Ml + j)z^{-l}$ are the polyphase components of the filter $H(z)$. With this formulation, the analysis filter bank can be described as

$$H_l(z) = \sum_{n=0}^{N-1} h_l(n)z^{-n} = \sum_{n=0}^{2LM-1} c_{l,n}h(n)z^{-n} \tag{66}$$

$$= \sum_{l=0}^{L-1}\sum_{j=0}^{2M-1} c_{l,n}h(2lM + j)z - (2lM + j) \tag{67}$$

The expression above can be further simplified if we explore the following property:

$$\begin{aligned} &\cos\left\{(2l+1)\frac{\pi}{2M}\left[(n + 2kM) - \frac{N}{2}\right]\right\} \\ &\quad = (-1)^k\cos\left\{(2l+1)\frac{\pi}{2M}\left[n - \frac{N}{2}\right]\right\} \end{aligned} \tag{68}$$

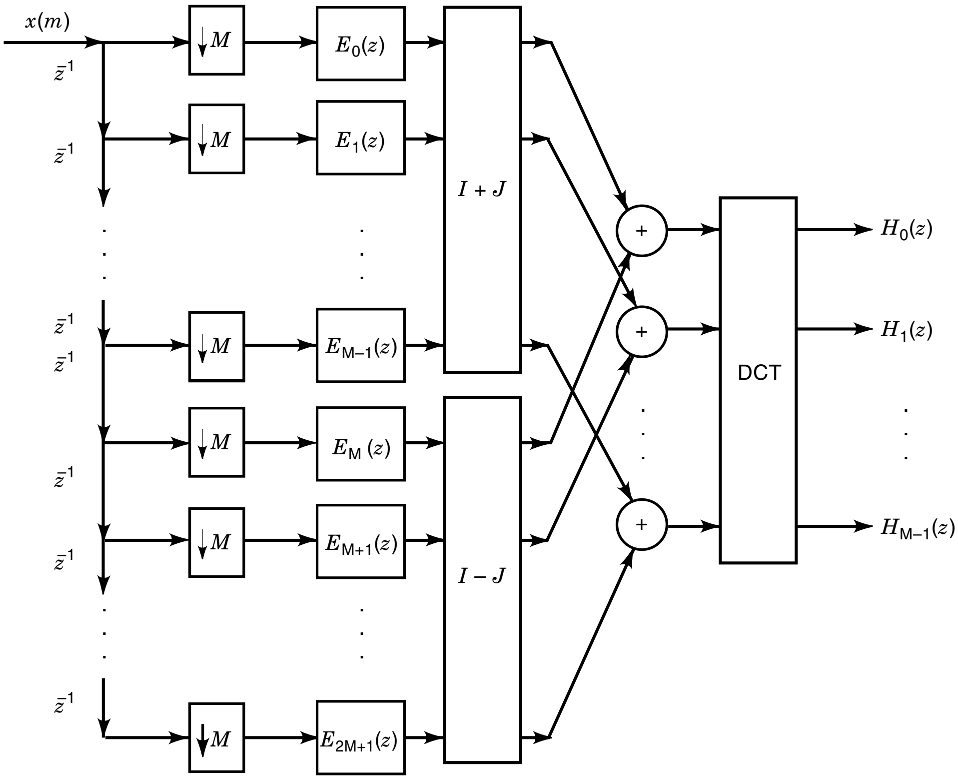

Figure 30. Cosine-modulated filter bank.

which leads to

$$(-1)^k c_{l,n} = c_{l,n+2kM} \tag{69}$$

With this relation and after a few manipulations, we can rewrite Eq. (67) as follows

$$H_l(z) = \sum_{j=0}^{2M-1} c_{l,n} z^{-(j)} E_j(-z^{2M}) \tag{70}$$

The above expression can be rewritten in a compact form as follows

$$e(z) = \begin{pmatrix} H_0(z) \\ H_1(z) \\ \vdots \\ H_{M-1}(z) \end{pmatrix} = (C_1 \quad C_2) \begin{pmatrix} E_0(z^{2M}) \\ z^{-1}E_1(z^{2M}) \\ \vdots \\ z^{-(2M-1)}E_{2M-1}(z^{2M}) \end{pmatrix} \tag{71}$$

where C_1 and C_2 are matrices whose elements are $c_{l,n}$ and $c_{l,n+M}$, respectively, for $0 \leq l$, $n \leq M-1$. The above equation and Eqs. (132–134) suggest the structure of Fig. 30 for the implementation of the cosine modulated filter bank. This structure consists of the implementation of the polyphase components of the prototype filter in cascade with a DCT-based matrix.

Equation (71) can be expressed in a convenient form to deduce the constraint on the prototype impulse-response in order to achieve perfect reconstruction, that is

$$\begin{aligned} e(z) &= \begin{pmatrix} H_0(z) \\ H_1(z) \\ \vdots \\ H_{M-1}(z) \end{pmatrix} \\ &= \begin{pmatrix} C_1 & C_2 \end{pmatrix} \begin{pmatrix} E_0(-z^{-2M}) & & & 0 \\ & E_1(-z^{-2M}) & & \\ & & \ddots & \\ 0 & & & E_{2M-1}(-z^{-2M}) \end{pmatrix} \begin{pmatrix} d(z) \\ z^{-M}d(z) \end{pmatrix} \\ &= \left[C_1 \begin{pmatrix} E_0(-z^{-2M}) & & & 0 \\ & E_1(-z^{-2M}) & & \\ & & \ddots & \\ 0 & & & E_{M-1}(-z^{-2M}) \end{pmatrix} \right. \\ &\quad \left. + z^{-M}C_2 \begin{pmatrix} E_M(-z^{-2M}) & & & 0 \\ & E_{M+1}(-z^{-2M}) & & \\ & & \ddots & \\ 0 & & & E_{2M-1}(-z^{-2M}) \end{pmatrix} \right] d(z) \end{aligned} \tag{72}$$

$$= E(z^M)d(z) \tag{73}$$

where $E(z)$ is the polyphase matrix as defined in Eq. (28).

In order to achieve perfect reconstruction in a filter bank with M channels, we should have $E(z)R(z) = R(z)E(z) = Iz^{-\Delta}$. However, it is well known (see Ref. 5), that the polyphase matrix of the analysis filter bank can be designed to be paraunitary, that is $E^T(z^{-1})E(z) = I$, where I is an identity matrix of dimension M. In this case the synthesis filters can be easily obtained from the analysis filter bank using either Eq. (64), or

$$R(z) = z^{-\Delta}E^{-1}(z) = z^{-\Delta}E^T(z^{-1}) \tag{74}$$

The task remains of showing how the prototype filter can be constrained such that the polyphase matrix of the analysis filter bank becomes paraunitary. The desired result is the following: *The polyphase matrix of the analysis filter bank becomes paraunitary, for a real coefficient prototype filter, if and only if*

$$E_j(z^{-1})E_j(z) + E_{j+M}(z^{-1})E_{j+M}(z) = \frac{1}{2M} \tag{75}$$

for $0 \leq j \leq M - 1$. An outline of the proof of this result is given in Appendix C. These M constraints can be reduced because the prototype filter has linear phase, that is, for M odd $0 \leq j \leq (M - 1)/2$ and for M even $0 \leq j \leq M/2 - 1$.

The necessary and sufficient conditions for perfect reconstruction on cosine-modulated filter banks are equivalent to having pairwise power complementary polyphase components on the prototype filter. This property can be explored to further reduce the computational complexity of these type of filter banks by implementing the power complementary pairs with lattice realizations, which are structures specially suited for this task (see Ref. 5).

Figure 31 depicts the frequency response of the analysis filters each with length 35, for a bank with five subbands.

The filters banks discussed in this section have as main disadvantage the nonlinear phase of the analysis filters, an undesirable feature in applications such as image coding. The lapped orthogonal transforms (LOTs) were originally proposed to reduce the blocking effects caused by discontinuities across block boundaries, specially for images (see Ref. 8). It turns out that LOT-based filter banks are very attractive because they lead to linear-phase analysis filters and have fast implementation. The LOT-based filter banks are members of the family of lossless FIR perfect-reconstruction filter banks with linear phase.

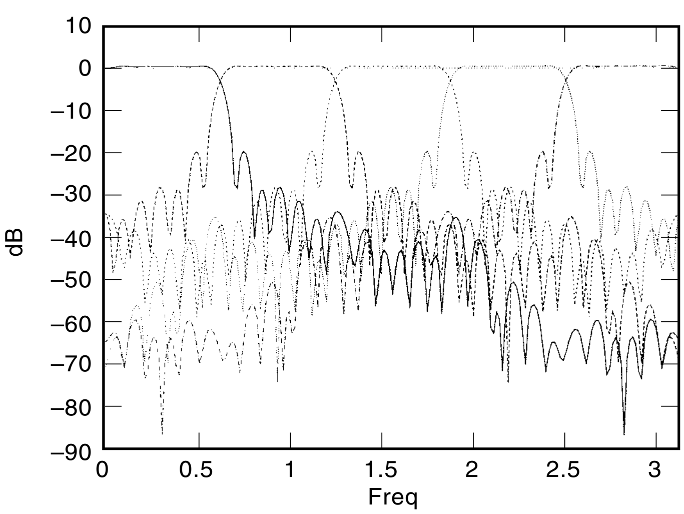

Figure 31. Frequency response of the analysis filters of a cosine-modulated filter bank.

Lapped Transforms. Although there are a number of possible designs for linear-phase filter banks with perfect reconstruction, the LOT-based design is simple to derive and to implement. The term LOT applies to the cases where the analysis filters have length $2M$. Generalizations of the LOT to longer analysis and synthesis filters (length LM) are available. They are known as the extended lapped transforms (ELTs) proposed by Malvar (8) and the generalized LOT (GenLot) proposed in Ref. 9. The ELT is closely related to the cosine-modulated filter banks and does not produce linear-phase analysis filters. The GenLot is a good choice when long analysis filters (with high selectivity) are required together with linear phase.

In this subsection we will briefly discuss the LOT filter bank, where the analysis and synthesis filter banks have lengths $2M$. The analysis filters are given by

$$e(z) = \begin{pmatrix} H_0(z) \\ H_1(z) \\ \vdots \\ H_{M-1}(z) \end{pmatrix} \tag{76}$$

$$= \begin{pmatrix} C_1' & C_2' \end{pmatrix} \begin{pmatrix} 1 \\ z^{-1} \\ \vdots \\ z^{-(2M-1)} \end{pmatrix} \tag{77}$$

where C_1' and C_2' are matrices whose elements are $c_{l,n}'$ and $c_{l,n+M}'$, respectively, for $0 \leq l,\ n \leq M - 1$. The above equation can also be rewritten in a more convenient form as

$$e(z) = \begin{pmatrix} H_0(z) \\ H_1(z) \\ \vdots \\ H_{M-1}(z) \end{pmatrix} \tag{78}$$

$$= \{C_1' + z^{-M}C_2'\}\, d(z) \tag{79}$$

$$= E(z^M)d(z) \tag{80}$$

where $E(z)$ is the polyphase matrix of the analysis filter bank.

The perfect reconstruction condition with paraunitary polyphase matrices is generated if

$$R(z) = z^{-\Delta}E^{-1}(z) = z^{-\Delta}E^T(z^{-1}) \tag{81}$$

The polyphase matrix of the analysis filter bank becomes lossless for a real coefficient prototype filter if the following conditions are satisfied:

$$C_1'^T C_1' + C_2'^T C_2' = I \tag{82}$$

$$C_1'^T C_2' = C_2'^T C_1' = 0 \tag{83}$$

where the last relation guarantees that the overlapping tails of the basis functions are orthogonal. The proof of this result is given elsewhere (8).

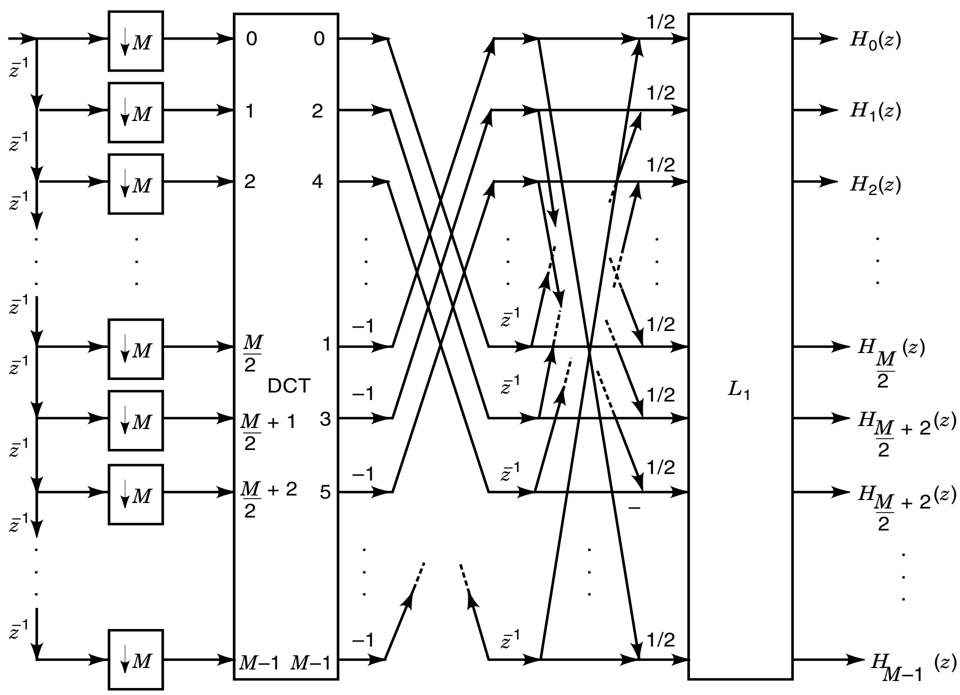

Figure 32. Lapped orthogonal transform.

A simple construction for the matrices above based on the DCT is choosing

$$C_1' = \frac{1}{2}\begin{pmatrix} C_e - C_o \\ C_e - C_o \end{pmatrix} \tag{84}$$

$$C_2' = \frac{1}{2}\begin{pmatrix} J(C_e - C_o) \\ -J(C_e - C_o) \end{pmatrix} \tag{85}$$

where the matrices C_e and C_o are $M/2$ by M matrices consisting of the even and odd DCT basis of length M, respectively. The reader can easily verify that the above choice satisfies the relations (82) and (83). With this we can build an initial LOT whose polyphase matrix is given by

$$\begin{aligned} E(z) &= \frac{1}{2}\begin{pmatrix} C_e - C_o & z^{-1}J(C_e - C_o) \\ C_e - C_o & -z^{-1}J(C_e - C_o) \end{pmatrix} \\ &= \frac{1}{2}\begin{pmatrix} I & z^{-1}J \\ I & -z^{-1}J \end{pmatrix}\begin{pmatrix} I & -I \\ I & -I \end{pmatrix}\begin{pmatrix} C_e \\ C_o \end{pmatrix} \end{aligned} \tag{86}$$

The last equality above suggests the structure of Fig. 32 for the implementation of the LOT filter bank. This structure consists of the implementation of the polyphase components of the prototype filter using a DCT-based matrix. It is also included in the figure an orthogonal matrix L_1 whose choice is discussed next. The inclusion of this matrix generalizes the choice of the filter bank and keeps the perfect reconstruction conditions. The polyphase matrix is then given by

$$E(z) = \frac{1}{2}L_1\begin{pmatrix} I & z^{-1}J \\ I & -z^{-1}J \end{pmatrix}\begin{pmatrix} I & -I \\ I & -I \end{pmatrix}\begin{pmatrix} C_e \\ C_o \end{pmatrix} \tag{87}$$

The basic construction of the LOT presented above is equivalent to the one proposed by Malvar (see Ref. 8), who utilizes a block transform formulation to generate lapped transforms. This formulation differs somewhat from the one in Eqs. (76–87). He starts with an orthogonal matrix based on the DCT having the following form:

$$L_0 = \frac{1}{2}\begin{pmatrix} C_e - C_o & J(C_e - C_o) \\ C_e - C_o & -J(C_e - C_o) \end{pmatrix} \tag{88}$$

This choice is not at random. First, it satisfies the conditions of Eqs. (82) and (83). Also, the first half of the basis functions are symmetric whereas the second half is antisymmetric, thus keeping the phase linear. The choice of DCT basis is the key to generate a fast implementation algorithm. Starting with L_0 we can generate a family of more selective analysis filters in the following form

$$L_{lot} = L_1 L_0 \tag{89}$$

where the matrix L_1 should be orthogonal and also be amenable to fast implementation. The most widely used form for this matrix is

$$L_1 = \begin{pmatrix} I & 0 \\ 0 & L_2 \end{pmatrix} \tag{90}$$

where L_2 is a square matrix of dimension $M/2$ consisting of a set of plane rotations whose angles are submitted to optimization aiming at maximizing the coding gain when using the filter bank in subband coders, or improving the selectivity of the analysis and synthesis filters (8).

Figure 33 depicts the frequency response of the analysis filters of an LOT with eight subbands.

Fast Algorithms. We now present a general construction of a fast algorithm for the LOT. Start by defining two matrices

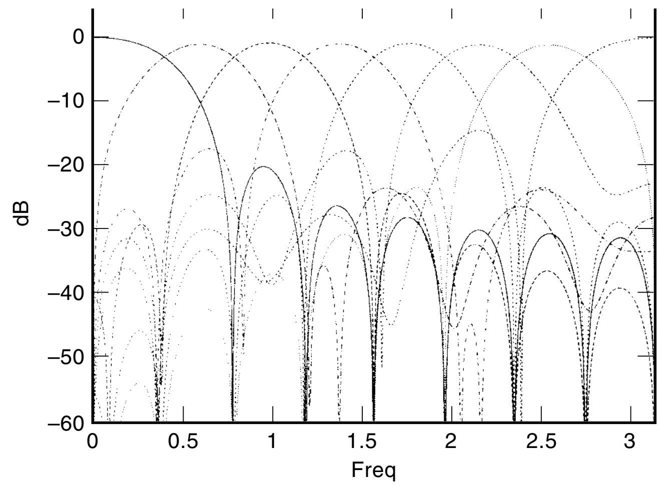

Figure 33. Frequency response of the analysis filters of an LOT-based filter bank with eight sub-bands.

as follows

$$C'_3 = C'_1 C'^T_1 \tag{91}$$

$$C'_4 = C'_1 + C'_2 \tag{92}$$

By premultiplying both terms in Eq. (82) separately by C'_1 and by C'_2, and using the results in Eq. (83), one can show that

$$C'_1 = C'_3 C'_4 \tag{93}$$

$$C'_2 = \{I - C'_3\} C'_4 \tag{94}$$

With the above relations it is straightforward to show that the polyphase components of the analysis filter can be written as

$$E(z) = \{C'_3 + z^{-1}[I - C'_3]\} C'_4 \tag{95}$$

The previously discussed initial solution for the LOT matrix can be analyzed in the light of this general formulation. After a few manipulations the matrices of the polyphase description above corresponding to the LOT matrix of Eq. (88) are given by

$$C'_3 = \frac{1}{2}\begin{pmatrix} C_e C_e^T + C_o C_o^T & C_e C_e^T + C_o C_o^T \\ C_e C_e^T + C_o C_o^T & C_e C_e^T + C_o C_o^T \end{pmatrix} \tag{96}$$

$$C'_4 = \frac{1}{2}\begin{pmatrix} C_e - C_o & J(C_e - C_o) \\ C_e - C_o & -J(C_e - C_o) \end{pmatrix} \tag{97}$$

The substitution of these equations back in Eq. (95) clarifies the relation between the algebraic formulations and the actual structure that implements the algorithm.

WAVELET TRANSFORMS

Wavelet transforms are a relatively recent development from functional analysis and have attracted great attention in the signal processing community (10). The *wavelet transform* of a function belonging to $\mathcal{L}^2\{\mathbb{R}\}$, the space of the square integrable functions, is its decomposition in a base composed by expansions, compressions, and translations of a single mother function $\psi(t)$, called *wavelet.*

Its applications range from quantum physics to signal coding. It can be shown that for digital signals, the wavelet transform is a special case of critically decimated filter banks (11). In fact, its numerical implementation relies heavily on that. In what follows, we will give a brief introduction to wavelet transforms, emphasizing its relation to filter banks. Indeed it is quite easy to find in the literature good material analyzing wavelet transforms from different points of view. For example, Ref. 10 is a very good book on the subject, written by a mathematician. For people with a signal processing background, Ref. 4 is very useful. The text in Ref. 12 is excellent and very clear, at a more introductory level.

Hierarchical Filter Banks

The cascade of 2-band filter banks can produce many different kinds of critically decimated decompositions. For example, one can make a 2^k-band uniform decomposition as depicted in Fig. 34(a) for $k = 3$. Another common type of hierarchical decomposition is the octave-band decomposition, in which only the low-pass band is further decomposed. In Fig. 34(b), one can see a 3-stage octave-band decomposition. The synthesis filter banks are not drawn because they are entirely analogous.

Octave-Band Filter Banks and Wavelet Transforms

Wavelets. Consider the octave-band analysis and synthesis filter bank in Fig. 35, where the low-pass bands are recursively decomposed into low- and high-pass channels. The outputs of the low-pass channels after an $S + 1$ stages decomposition are $x_{S,n}$ and the outputs of the high-pass channels are $c_{S,n}$, $S \geq 1$.

Applying the noble identities to Fig. 35 we arrive at Fig. 36. After $S + 1$ stages and before decimation by a factor of 2^{S+1}, the $\mathcal{Z}$ transforms of the analysis low- and high-pass channels, $H^{(S)}_{\text{low}}(z)$ and $H^{(S)}_{\text{high}}(z)$, respectively, are

$$H^{(S)}_{\text{low}}(z) = \frac{X_S(z)}{X(z)} = \prod_{k=0}^{S} H_0(z^{2^k}) \tag{98}$$

$$H^{(S)}_{\text{high}}(z) = \frac{C_S(x)}{X(z)} = H_1(z^{2^s}) H^{(S-1)}_{\text{low}}(z) \tag{99}$$

The synthesis channels are analogous to the analysis ones, i.e.,

$$G^{(S)}_{\text{low}}(z) = \prod_{k=0}^{S} G_0(z^{2^k}) \tag{100}$$

$$G^{(S)}_{\text{high}}(z) = G_1(z^{2^s}) H^{(S-1)}_{\text{low}}(z) \tag{101}$$

If $H_0(z)$ has enough zeros at $z = -1$, it can be shown (4,12) that the envelope of the impulse response of the filters from Eq. (99) has the same shape for $S = 0, 1, 2, \ldots$ In other words, this envelope can be represented by expansions and contractions of a single function $\psi(t)$ (see Fig. 37 for the analysis filter bank).

In Fig. 37, the envelopes before and after the decimators are the same. However, it must be noted that after decimation we cannot anymore refer to impulse responses in the usual

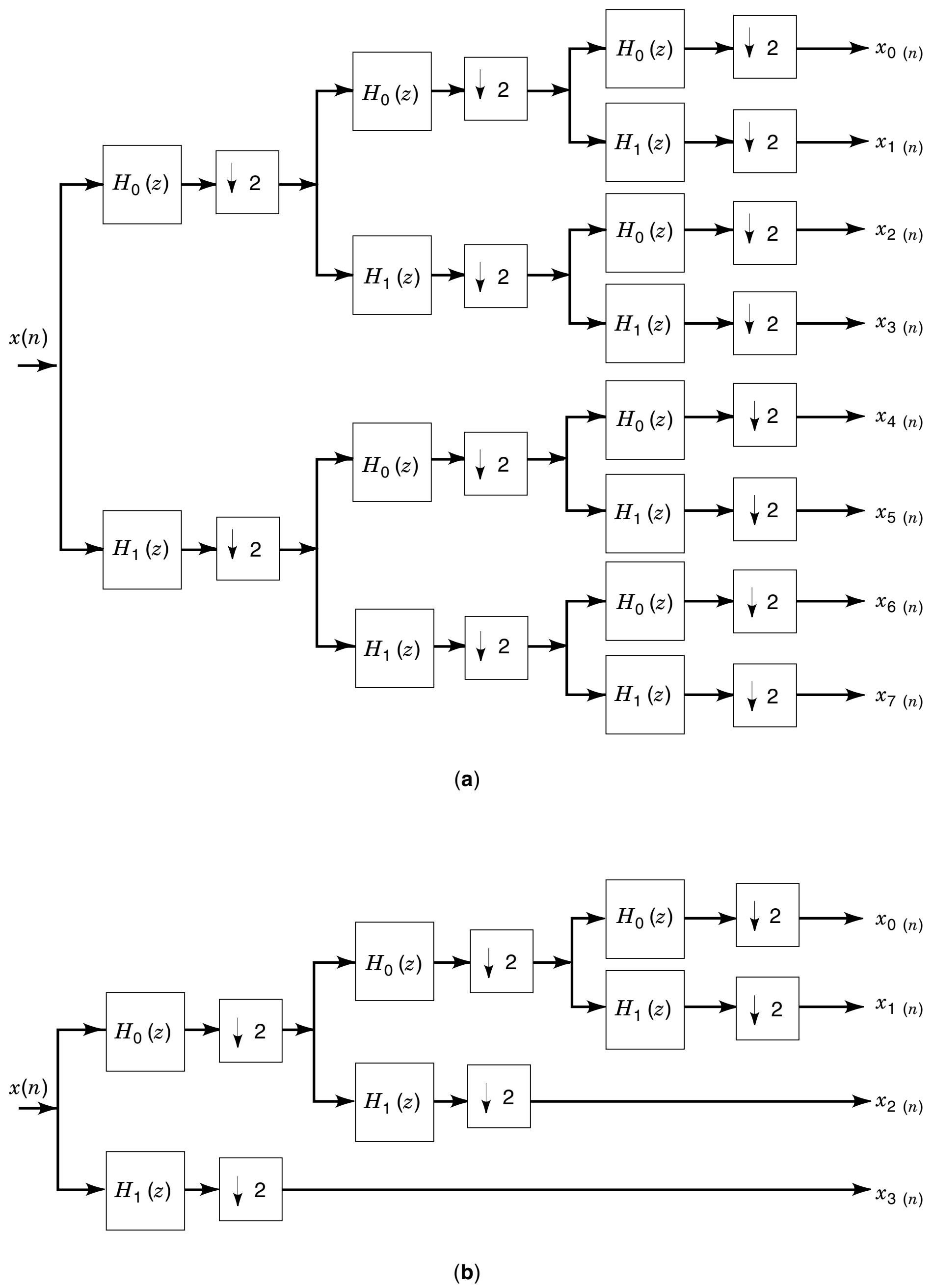

Figure 34. Hierarchical decompositions: (a) 8-band uniform; (b) 3-stage octave-band.

way, because the decimation operation is not shift invariant (see the subsection entitled "Decimation").

If ω_s is the sampling rate at the input of the system in Fig. 37, we have that this system has the same output as the one from Fig. 38, where the boxes mean continuous-time filters with impulse responses equal to the envelopes of Fig. 37. Note that in this case, sampling with frequency ω_s/k is equivalent to subsampling by k.

As stated above, the impulse responses of the continuous-time filters of Fig. 38 are expansions and contractions of a single mother function $\psi(t)$. In Fig. 38, the highest sampling frequency was $\omega_s/2$. Each channel added to the right had an impulse response with double the width and sampling rate half of the previous one. There is no impediment in also adding channels to the left of the channel with sampling frequency $\omega_s/2$. Each new channel added to the left would have an impulse response with half the width and double the sampling rate of the previous one. If we keep adding channels to both the right and the left indefinitely, we arrive at Fig. 39. If a continuous-time signal is input to the system of Fig. 39 its output is referred to as the *wavelet transform* of $x(t)$, and the mother function $\psi(t)$ is called the *wavelet,* or, more specifically, the *analysis wavelet* (13).

Assuming, without loss of generality, that $\omega_s = 2\pi$ ($T_s = 1$), it is straightforward to derive from Fig. 39 that the wavelet transform of a signal $x(t)$ is (actually, in this formula, the impulse response of the filters are expansions and contractions of $\psi(-t)$)

$$c_{m,n} = \int_{-\infty}^{\infty} 2^{-\frac{m}{2}} \psi(2^{-m}t - n)x(t)\,dt \tag{102}$$

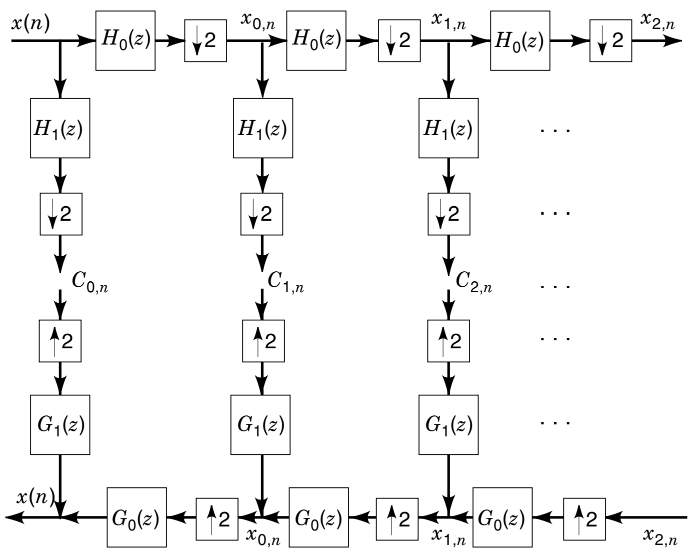

Figure 35. Octave-band analysis and synthesis filter bank.

The constant $2^{-m/2}$ is included because, if $\psi(t)$ has unity energy, $2^{-m/2}\psi(2^{-m}t - n)$ has also unity energy, which can be assumed without loss of generality.

From Figs. 36 and 39 and Eq. (99), one can see that the wavelet $\psi(t)$ is band-pass, because each channel is a cascade of several low-pass filters and a high-pass filter. When the wavelet is expanded by 2, its bandwidth is decreased by 2, as seen in Fig. 40. Therefore, the decomposition in Fig. 39 and Eq. (102) is, in the frequency domain, as in Fig. 41.

In a similar manner, the envelopes of the impulse responses of the equivalent synthesis filters after interpolation [see Fig. 36 and Eq. (101)] are expansions and contractions of a single parent function $\overline{\psi}(t)$. Using a similar reasoning to the one leading to Figs. 37–39, one can obtain the continuous-time signal $x(t)$ from the wavelet coefficients $c_{m,n}$ [Eq. (102)] (4):

$$x(t) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} c_{m,n} 2^{-\frac{m}{2}} \overline{\psi}(2^{-m}t - n) \tag{103}$$

Equations (102) and (103) are the direct and inverse wavelet transforms of a continuous-time signal $x(t)$. The wavelet transform of the corresponding discrete-time signal $x(n)$ is merely the octave-band decomposition in Figs. 35 and 36. A natural question to ask at this point is: How are the continuous-time signal $x(t)$ and the discrete-time signal $x(n)$ related if they generate the same wavelet coefficients? In addition, how can the analysis and synthesis wavelets be derived from the filter bank coefficients and vice and versa? In order to answer these questions we need the concept of a *scaling function*.

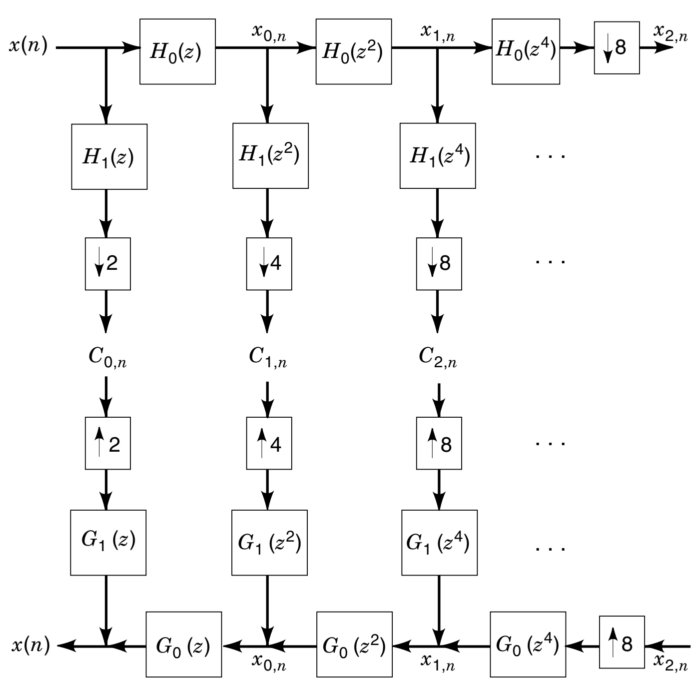

Figure 36. Octave-band analysis and synthesis filter bank after the application of the noble identities.

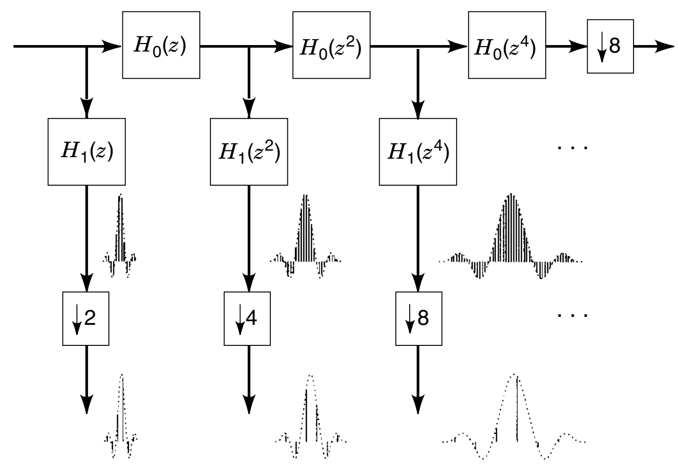

Figure 37. The impulse response of the filters from Eq. (102) has the same shape for every S.

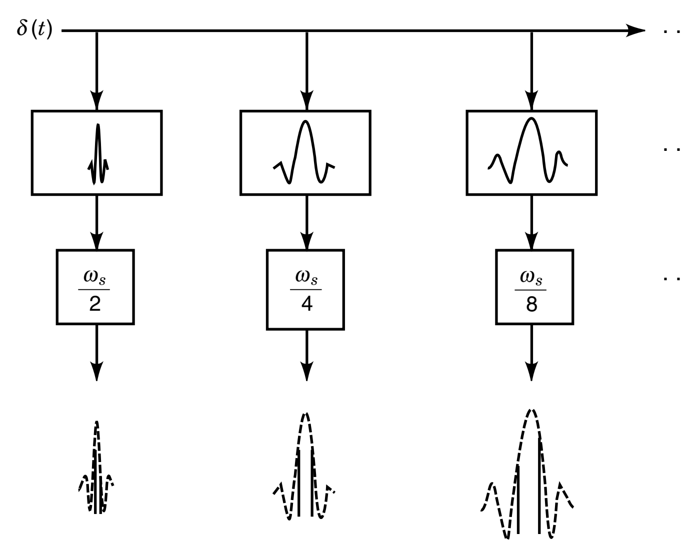

Figure 38. This system has the same outputs as the system from Fig. 37.

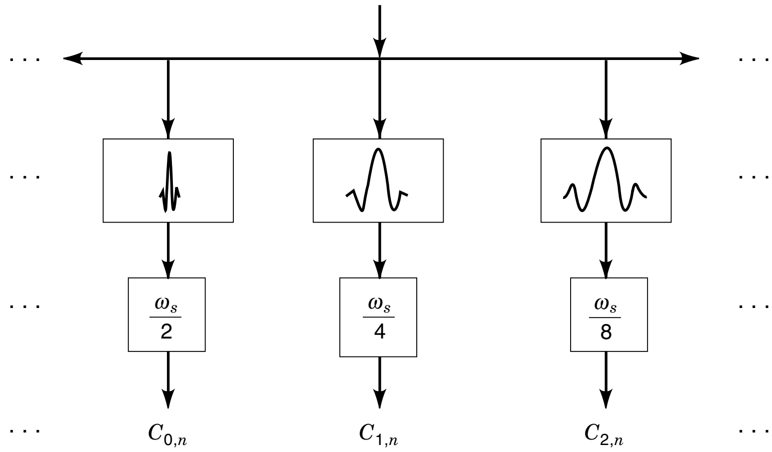

Figure 39. Wavelet transform of a continuous signal $x(t)$.

Scaling Functions. By looking at Eqs. (102) and (103) we see that the values of m which are associated with the "width" of the filters (Fig. 39) range from $-\infty$ to $+\infty$. Since all signals encountered in practice are somehow band limited, one can assume generally that the output of the filters with impulse responses $\psi(2^{-m}t)$ are zero for $m < 0$. Therefore, m now varies from 0 to $+\infty$. Looking at Figs. 35–37, we see that $m \to +\infty$ means the low-pass channels will be indefinitely decomposed. However, in practice, the number of stages of decomposition is fixed, and after S stages we have S band-pass channels and one low-pass channel. Therefore, if we restrict the number of stages of decomposition in Figs. 35–39 and add a low-pass channel, we can modify Eq. (103) such that m assumes only a finite number of values.

In order to do this, we notice that if $H_0(z)$ has enough zeros at $z = -1$, the envelopes of the analysis low-pass channels [Eq. (98)] are also expansions and contractions of a single function $\phi(t)$, which is called *analysis scaling function.* Likewise, the envelopes of the synthesis low-pass channels are expansions and contractions of the *synthesis scaling function* $\overline{\phi}(t)$ (4). Therefore, if we make an $S + 1$ stage decomposition, Eq. (103) becomes:

$$\begin{aligned} x(t) = & \sum_{n=-\infty}^{\infty} x_{S,n} 2^{-\frac{S}{2}} \overline{\phi}(2^{-S}t - n) \\ & + \sum_{m=0}^{S-1} \sum_{n=-\infty}^{\infty} c_{m,n} 2^{-\frac{m}{2}} \overline{\psi}(2^{-m}t - n) \end{aligned} \tag{104}$$

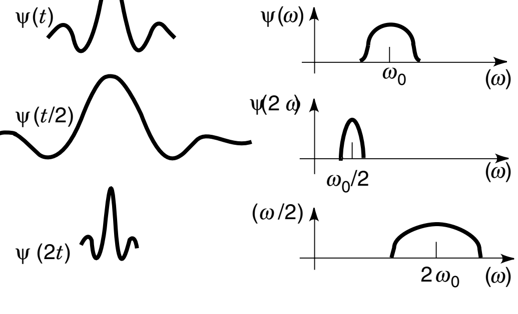

Figure 40. Expansions and contractions of the wavelet in the time and frequency domains.

where

$$x_{S,n} = \int_{-\infty}^{\infty} 2^{-\frac{S}{2}} \phi(2^{-S}t - n) x(t)\, dt \tag{105}$$

Therefore, the wavelet transform is in practice given by Eq. (105). The summations in n will in general depend on the supports (those regions where the functions are nonzero) of the signal, wavelets, and scaling functions (13).

Relation Between x(t) and x(n)<. Equation (105) shows how to compute the coefficients of the low-pass channel after an $S + 1$ stages wavelet transform. In Fig. 35, $x_{S,n}$ are the outputs of a low-pass filter $H_0(z)$ after $S + 1$ stages. Since in Fig. 36 the discrete-time signal $x(n)$ can be regarded as the output of a low-pass filter after "zero" stages, we can say that $x(n)$ would be equal to $x_{-1,n}$. In other words, the equivalence of the outputs of the octave-band filter bank of Fig. 35 and the wavelet transform given by Eqs. (102) and (103) occurs only if the digital signal input to the filter bank of Fig. 35 is equal to $x_{-1,n}$. From Eq. (105) this means:

$$x(n) = \int_{-\infty}^{\infty} \sqrt{2}\phi(2t - n) x(t)\, dt \tag{106}$$

Equation (106) can be interpreted as $x(n)$ being the signal $x(t)$ digitized with the band-limiting filter having as impulse response $\sqrt{2}\phi(-2t)$.

Therefore, a possible way to compute the wavelet transform of a continuous-time signal $x(t)$ is as depicted in Fig. 42. $x(t)$ is passed through a filter having as impulse response the scaling function contracted by 2 in time and sampled with $T_s = 1$ ($\omega_s = 2\pi$), the resulting digital signal being input to the octave-band filter bank in Fig. 35 with the filter coefficients given by Eqs. (107) and (109). At this point, it is important to note that, strictly speaking, the wavelet transform is

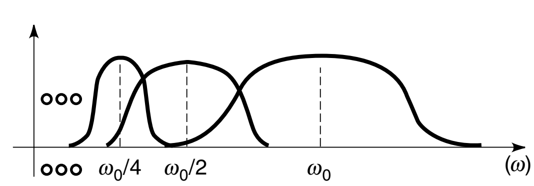

Figure 41. Wavelet transform in the frequency domain.

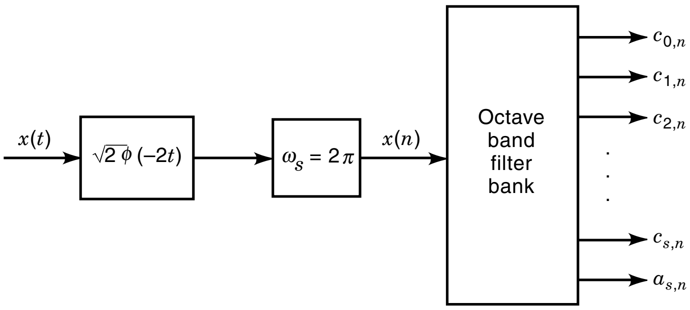

Figure 42. Practical way to compute the wavelet transform of a continuous-time signal.

only defined for continuous-time signals. However, it is common practice to refer to the wavelet transform of a discrete-time signal $x(n)$ as the output of the filter bank in Fig. 35 (4).

Relation between the Wavelets and the Filter Coefficients. If $h_0(n)$, $h_1(n)$, $g_0(n)$ and $g_1(n)$ are the impulse responses of the analysis low- and high-pass filters and synthesis low- and high-pass filters, respectively, and $\phi(t)$, $\overline{\phi}(t)$, $\psi(t)$ and $\overline{\psi}(t)$ are the analysis and synthesis scaling functions and analysis and synthesis wavelets, respectively, we have (11):

$$h_{0n} = \int_{-\infty}^{\infty} \phi(t)\sqrt{2}\overline{\phi}(2t+n)\,dt \tag{107}$$

$$g_{0n} = \int_{-\infty}^{\infty} \overline{\phi}(t)\sqrt{2}\phi(2t-n)\,dt \tag{108}$$

$$h_{1n} = \int_{-\infty}^{\infty} \psi(t)\sqrt{2}\overline{\phi}(2t+n)\,dt \tag{109}$$

$$g_{1n} = \int_{-\infty}^{\infty} \overline{\psi}(t)\sqrt{2}\phi(2t-n)\,dt \tag{110}$$

And, considering their Fourier transforms (4),

$$\Phi(\omega) = \prod_{n=1}^{\infty} \frac{1}{\sqrt{2}} H_0(e^{-j\frac{w}{2^n}}) \tag{111}$$

$$\overline{\Phi}(\omega) = \prod_{n=1}^{\infty} \frac{1}{\sqrt{2}} G_0(e^{j\frac{w}{2^n}}) \tag{112}$$

$$\Psi(\omega) = \frac{1}{\sqrt{2}} H_1(e^{-j\frac{w}{2}}) \prod_{n=2}^{\infty} \frac{1}{\sqrt{2}} H_0(e^{-j\frac{w}{2^n}}) \tag{113}$$

$$\overline{\Psi}(\omega) = \frac{1}{\sqrt{2}} G_1(e^{j\frac{w}{2}}) \prod_{n=2}^{\infty} \frac{1}{\sqrt{2}} G_0(e^{j\frac{w}{2^n}}) \tag{114}$$

When $\phi(t) = \overline{\phi}(t)$ and $\psi(t) = \overline{\psi}(t)$, the wavelet transform is *orthogonal* (13). Otherwise, it is only *biorthogonal* (14). It is important to notice that, for the wavelet transform to be defined, the corresponding filter bank must provide perfect reconstruction.

Regularity

From Eqs. (111)–(114) one can see that the wavelets and scaling functions are derived from the filter bank coefficients by infinite products. Therefore, in order for a wavelet to be defined, these infinite products must converge. In other words, a wavelet transform is not necessarily defined for every two-band perfect reconstruction filter bank. There are cases in which the envelope of the impulse responses of the equivalent filters of Eqs. (98)–(101) is not the same for every S (4).

The *regularity* of a wavelet or scaling function is roughly the number of continuous derivatives that a wavelet has. It is somehow a measure of the extent of convergence of the products in Eqs. (111)–(114). In order to define regularity more formally, we first need the following concept (15):

A function $f(t)$ is *Lipschitz of order* α, $0 < \alpha \le 1$ if, $\forall x, h \in \mathbb{R}$,

$$|f(x+h) - f(x)| \le ch^{\alpha} \tag{115}$$

where c is a constant.

Using this definition, we have that the *Hölder regularity* of a scaling function $\phi(t)$ is $r = N + \alpha$, where N is integer and $0 < \alpha \le 1$, if (15):

$$\frac{d^N\phi(t)}{dt^N} \text{ is Lipschitz of order } \alpha \tag{116}$$

It can be shown that, in order that a scaling function $\phi(t)$ be regular, $H_0(z)$ must have enough zeros at $z = -1$. In addition, supposing that $\phi(t)$ generated by $H_0(z)$ [Eq. (111)] has regularity r, if we take

$$H'_0(z) = \left(\frac{1+z^{-1}}{2}\right) H_0(z)$$

then $\phi'(t)$ generated by $H'_0(z)$ will have regularity $r + 1$ (15).

The regularity of a wavelet is the same as the regularity of the corresponding scaling function (15).

It can be shown that the regularity of a wavelet imposes the following a priori constraints on the filter banks (4):

$$H_0(1) = G_0(1) = \sqrt{2} \tag{117}$$

$$H_0(-1) = G_0(-1) = 0 \tag{118}$$

Equation (117) implies that the filters $H_0(z)$, $H_1(z)$, $G_0(z)$ and $G_1(z)$ have to be normalized in order to generate a wavelet transform.

It is interesting to note that when deriving the wavelet transform from the octave-band filter bank in the subsection "Wavelets," it was supposed that the low-pass filters had enough zeros at $z = -1$. In fact, what was meant there was that the wavelets should be regular.

In Fig. 43 we can see examples of wavelets with different regularities.

Examples

Every two-band perfect reconstruction filter bank with $H_0(z)$ having enough zeros at $z = -1$ has corresponding analysis and synthesis wavelets and scaling functions. For example, the filter bank described by Eqs. (20)–(23), normalized such that Eq. (117) is satisfied, generates the so-called *Haar wavelet*. It is the only orthogonal wavelet that has linear phase (4). The scaling function and wavelets are shown in Fig. 44.

The wavelets and scaling functions corresponding to the filter bank described by Eqs. (38)–(41) are depicted in Fig. 45.

A good example of orthogonal wavelet is the Daubechie's wavelet, whose filters have length 4. They are also an exam-

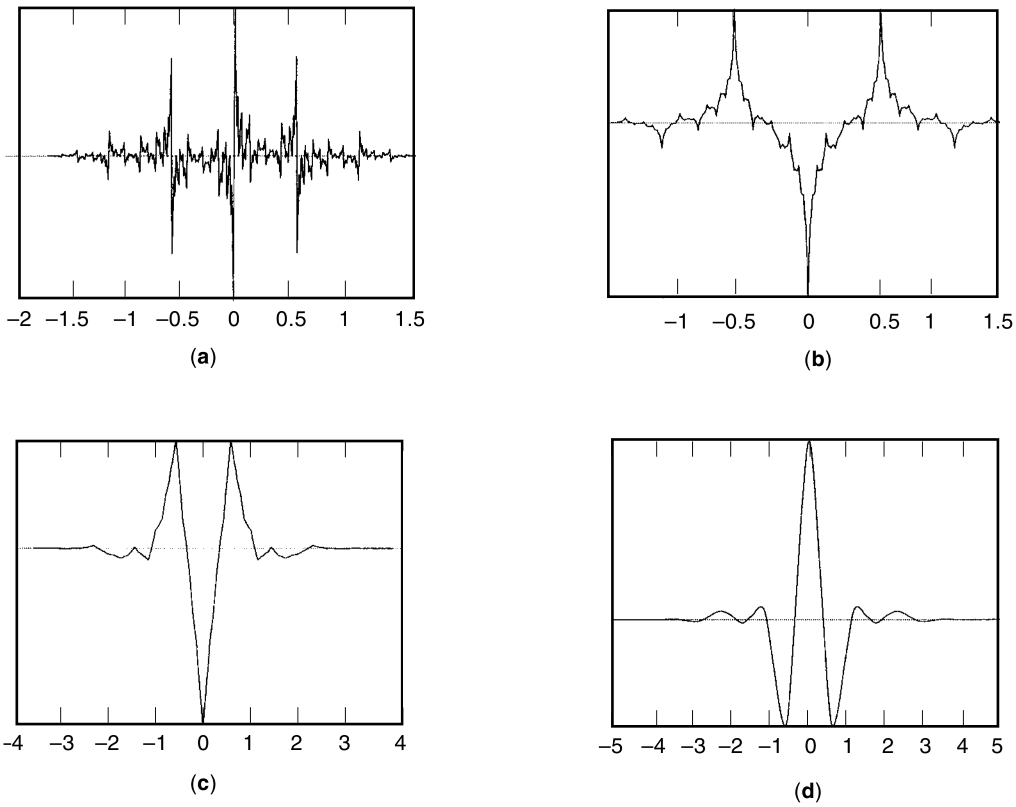

Figure 43. Examples of wavelets with different regularities. (a) Regularity = −1. (b) Regularity = 0. (c) Regularity = 1. (d) Regularity = 2.

ple of CQF filter banks (see the section entitled "CQF filter banks"). The filters are (13)

$$H_0(z) = +0.4829629 + 0.8365163z^{-1} + 0.2241439z^{-2} - 0.1294095z^{-3} \tag{119}$$

$$H_1(z) = -0.1294095 - 0.2241439z^{-1} + 0.8365163z^{-2} - 0.4829629z^{-3} \tag{120}$$

$$G_0(z) = -0.1294095 + 0.2241439z^{-1} + 0.8365163z^{-2} + 0.4829629z^{-3} \tag{121}$$

$$G_1(z) = -0.4829629 + 0.8365163z^{-1} - 0.2241439z^{-2} - 0.1294095z^{-3} \tag{122}$$

Since the wavelet transform is orthogonal, the analysis and synthesis scaling functions and wavelets are the same. The scaling function and wavelet are depicted in Fig. 46.

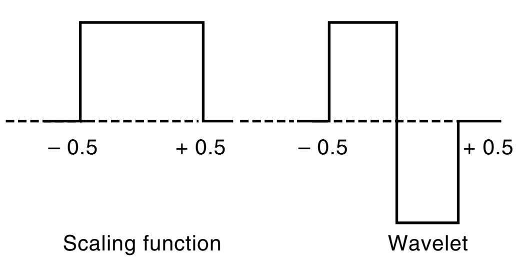

Figure 44. Haar wavelet and scaling function.

It is important to notice that, unlike the biorthogonal wavelets in Fig. 45, these orthogonal wavelets are nonsymmetrical, and, therefore, do not have linear phase.

APPENDIX A

Here we prove Eq. (5), which gives the spectrum of the decimated signal $x_d(n)$ as a function of the spectrum of the original signal $x(m)$. We have that

$$x_d(n) = x(nM) \tag{123}$$

Defining $x'(m)$ as

$$x'(m) = \begin{cases} x(m), & m = nM, n \in \mathbb{Z} \\ 0, & \text{otherwise} \end{cases} \tag{124}$$

$x'(m)$ can also be expressed as:

$$x'(m) = x(m) \sum_{l=-\infty}^{\infty} \delta(m - lM) \tag{125}$$

The Fourier transform of $x_d(n)$, $X_d(e^{j\omega})$ is then:

$$\begin{aligned} X_d(e^{j\omega}) &= \sum_{n=-\infty}^{\infty} x_d(n)e^{-j\omega n} = \sum_{n=-\infty}^{\infty} x(nM)e^{-j\omega n} \\ &= \sum_{n=-\infty}^{\infty} x'(nM)e^{-j\omega n} = \sum_{\theta=-\infty}^{\infty} x'(\theta M)e^{-j\frac{\omega}{M}\theta} = X'(e^{j\frac{\omega}{M}}) \end{aligned} \tag{126}$$

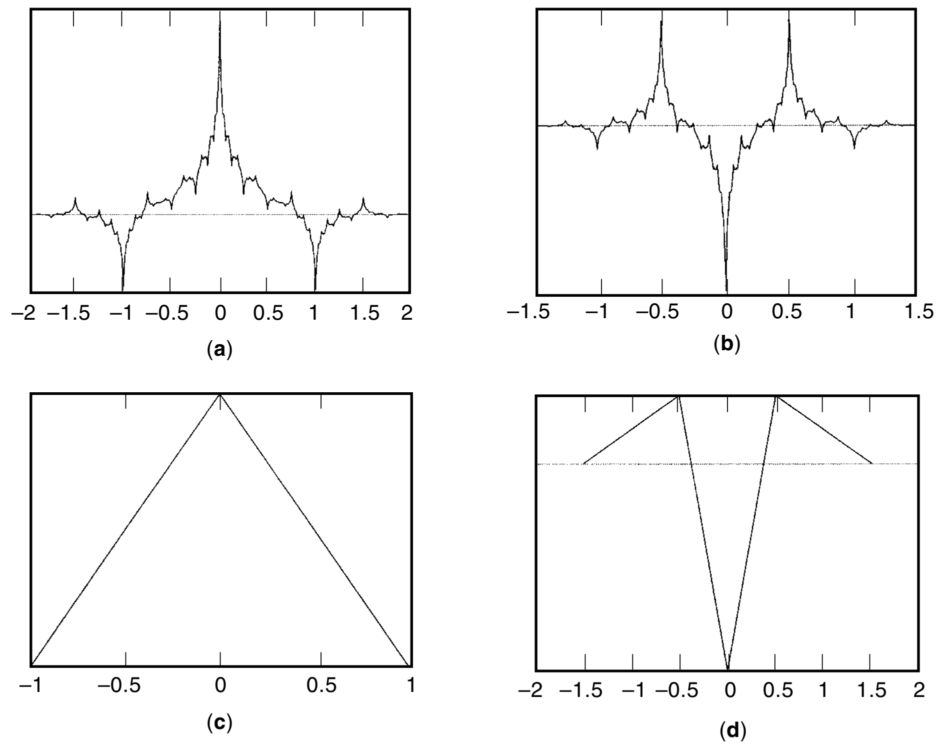

Figure 45. Wavelet transform generated by the filter bank from Eqs. (38)–(40). (a) Analysis scaling function. (b) Analysis wavelet. (c) Synthesis scaling function. (d) Synthesis wavelet.

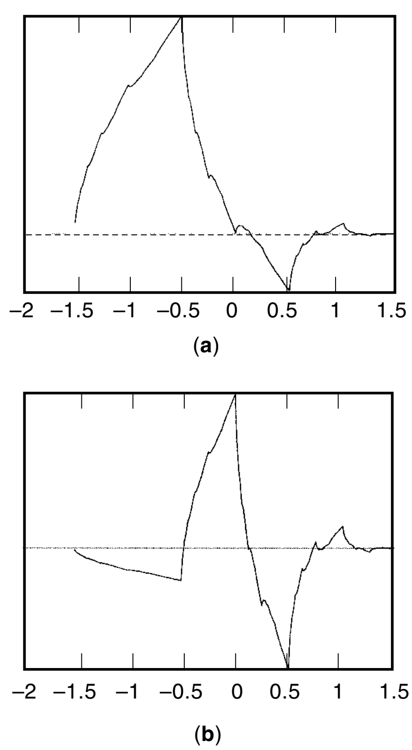

Figure 46. Wavelet and scaling function corresponding to the filter bank from Eqs. (122)–(125). (a) Scaling function. (b) Wavelet.

But from Eq. (125),

$$\begin{aligned} X\,(e^{j\omega}) &= \frac{1}{2\pi} X(e^{j\omega}) * \mathscr{F}\left\{\sum_{l=-\infty}^{\infty} \delta(m - lM)\right\} \\ &= \frac{1}{2\pi} X(e^{j\omega}) * \frac{2\pi}{M} \sum_{k=0}^{M-1} \delta\left(\omega - \frac{2\pi k}{M}\right) \\ &= \frac{1}{M} \sum_{k=0}^{M-1} X\,[e^{\left(\omega - \frac{2\pi k}{M}\right)}] \end{aligned} \tag{127}$$

Then, from Eq. (126),

$$X_d(e^{j\omega}) = X'(e^{j\frac{\omega}{M}}) = \frac{1}{M} \sum_{k=0}^{M-1} X\,[e^{\left(\frac{\omega - 2\pi k}{M}\right)}] \tag{128}$$

which is the same as Eq. (5).

APPENDIX B

In order to prove the identity in Fig. 17(a), one has to first rewrite Eq. (5), which gives the Fourier transform of the decimated signal $x_d(n)$ as a function of the input signal $x(m)$, in terms of $\mathscr{Z}$ transforms:

$$X_d(z) = \frac{1}{M} \sum_{k=0}^{M-1} X\,(z^{\frac{1}{M}} e^{-\frac{2\pi k}{M}}) \tag{129}$$

For the decimator followed by filter $H(z)$, we have that:

$$Y(z) = H(z) X_d(z) = \frac{1}{M} H(z) \sum_{k=0}^{M-1} X(z^{\frac{1}{M}} e^{-\frac{2\pi k}{M}}) \tag{130}$$

For the filter $H(z^M)$ followed by the decimator, if $U(z) = X(z)H(z^M)$ we have, from Eq. (129):

$$Y(z) = \frac{1}{M}\sum_{k=0}^{M-1} U(z^{\frac{1}{M}} e^{-\frac{2\pi k}{M}}) = \frac{1}{M}\sum_{k=0}^{M-1} X(z^{\frac{1}{M}} e^{-\frac{2\pi k}{M}}) H(z e^{-\frac{2\pi M k}{M}})$$
$$= \frac{1}{M}\sum_{k=0}^{M-1} X(z^{\frac{1}{M}} e^{-\frac{2\pi k}{M}}) H(z) \tag{131}$$

This is the same as Eq. (130), and the identity is proved.

The identity in Fig. 17(b) is straightforward. $H(z)$ followed by an interpolator gives $Y(z) = H(z^M)X(z^M)$, which is the expression for an interpolator followed by $H(z^M)$.

APPENDIX C

Before we prove the desired result, we need properties related to the modulation matrix that are required in the proof. These results are widely discussed in the literature, see for example Ref. 5. The results are

$$C_1^T C_1 = 2M[I + (-1)^{L-1} J] \tag{132}$$
$$C_2^T C_2 = 2M[I - (-1)^{L-1} J] \tag{133}$$
$$C_1^T C_2 = C_2^T C_1 = 0 \tag{134}$$

where I is the identity matrix, J is the reverse identity matrix, and 0 is a matrix with all elements equal to zero. All these matrices are square with order M. With this result it straightforward to show that

$$E^T(z^{-1})E(z) = \begin{pmatrix} E_0(-z^{-2}) & & & 0 \\ & E_1(-z^{-2}) & & \\ & & \ddots & \\ 0 & & & E_{M-1}(-z^{-2}) \end{pmatrix} C_1^T . C_1 \begin{pmatrix} E_0(-z^{-2}) & & & 0 \\ & E_1(-z^{-2}) & & \\ & & \ddots & \\ 0 & & & E_{M-1}(-z^{-2}) \end{pmatrix}$$
$$+ \begin{pmatrix} E_M(-z^{-2}) & & & 0 \\ & E_{M+1}(-z^{-2}) & & \\ & & \ddots & \\ 0 & & & E_{2M-1}(-z^{-2}) \end{pmatrix} C_2^T . C_2 \begin{pmatrix} E_M(-z^{-2}) & & & 0 \\ & E_{M+1}(-z^{-2}) & & \\ & & \ddots & \\ 0 & & & E_{2M-1}(-z^{-2}) \end{pmatrix} \tag{135}$$

Since the prototype is a linear phase filter, after a few manipulations it can be shown that

$$\begin{pmatrix} E_0(-z^{-2}) & & & 0 \\ & E_1(-z^{-2}) & & \\ & & \ddots & \\ 0 & & & E_{M-1}(-z^{-2}) \end{pmatrix} .J \begin{pmatrix} E_0(-z^{-2}) & & & 0 \\ & E_1(-z^{-2}) & & \\ & & \ddots & \\ 0 & & & E_{M-1}(-z^{-2}) \end{pmatrix}$$
$$= \begin{pmatrix} E_M(-z^{-2}) & & & 0 \\ & E_{M+1}(-z^{-2}) & & \\ & & \ddots & \\ 0 & & & E_{2M-1}(-z^{-2}) \end{pmatrix} .J \begin{pmatrix} E_M(-z^{-2}) & & & 0 \\ & E_{M+1}(-z^{-2}) & & \\ & & \ddots & \\ 0 & & & E_{2M-1}(-z^{-2}) \end{pmatrix} \tag{136}$$

This result allows some simplification in Eq. (135) after we replace Eqs. (132) and (133). The final result is

$$E^T(z^{-1})E(z) = 2M \left[\begin{pmatrix} E_0(-z^{-2}) & & & 0 \\ & E_1(-z^{-2}) & & \\ & & \ddots & \\ 0 & & & E_{M-1}(-z^{-2}) \end{pmatrix} . \begin{pmatrix} E_0(-z^{-2}) & & & 0 \\ & E_1(-z^{-2}) & & \\ & & \ddots & \\ 0 & & & E_{M-1}(-z^{-2}) \end{pmatrix} \right.$$
$$\left. + \begin{pmatrix} E_M(-z^{-2}) & & & 0 \\ & E_{M+1}(-z^{-2}) & & \\ & & \ddots & \\ 0 & & & E_{2M-1}(-z^{-2}) \end{pmatrix} . \begin{pmatrix} E_M(-z^{-2}) & & & 0 \\ & E_{M+1}(-z^{-2}) & & \\ & & \ddots & \\ 0 & & & E_{2M-1}(-z^{-2}) \end{pmatrix} \right] \tag{137}$$

If the matrix above is equal to the identity matrix, we achieve perfect reconstruction. The result above is equivalent to require that polyphase components of the prototype filter are pairwise power complementary which is exactly the result of Eq. (75).

BIBLIOGRAPHY

1. A. V. Oppenheim and R. W. Schaffer, *Discrete-Time Signal Processing,* Englewood Cliffs, NJ: Prentice-Hall, 1989.

2. R. E. Crochiere and L. R. Rabiner, *Multirate Digital Signal Processing,* Englewood Cliffs, NJ: Prentice-Hall, 1983.
3. Y.-C. Lim, Frequency-response masking approach for the synthesis of sharp linear phase digital filters, *IEEE Trans. Circuits Syst.,* **33**: 357–364, 1986.
4. M. Vetterli and J. Kovacevic, *Wavelets and Subband Coding,* Englewood Cliffs, NJ: Prentice-Hall, 1995.
5. P. P. Vaidyanathan, *Multirate Systems and Filter Banks,* Englewood Cliffs, NJ: Prentice-Hall, 1993.
6. A. Croisier, D. Esteban, and C. Galand, Perfect channel splitting by use of interpolation/decimation/tree decomposition techniques, *Int. Symp. on Info., Circuits and Syst.,* Patras, Greece, 1976.
7. M. J. T. Smith and T. P. Barnwell, Exact reconstruction techniques for tree-structured subband coders, *IEEE Trans. Acoust., Speech Signal Process.,* **34**: 434–441, 1986.
8. H. S. Malvar, *Signal Processing with Lapped Transforms,* Norwood, MA: Artech House, 1992.
9. R. L. de Queiroz, T. Nguyen, and K. R. Rao, The GenLOT: generalized linear-phase lapped orthogonal transform, *IEEE Trans. Signal Process.,* **44**: 497–507, 1996.
10. I. Daubechies, *Ten Lectures on Wavelets,* Philadelphia, PA: Soc. Ind. Appl. Math., 1991.
11. M. Vetterli and C. Herley, Wavelets and filters banks: theory and design, *IEEE Trans. Signal Process.,* **40**: 2207–2232, 1992.
12. G. Strang and T. Nguyen, *Wavelets and Filter Banks,* Wellesley, MA: Wellesley Cambridge Press, 1996.
13. I. Daubechies, Orthonormal bases of compactly supported wavelets, *Commun. on Pure and Appl. Mathematics,* **XLI**: 909–996, 1988.
14. A. Cohen, I. Daubechies, and J.-C. Feauveau, Biorthogonal bases of compactly supported wavelets, *Commun. on Pure and Appl. Math.,* **XLV**: 485–560, 1992.
15. O. Rioul, Simple regularity criteria for subdivision schemes, *SIAM J. Math. Anal.,* **23**: 1544–1576, November 1992.
16. R. David Koilpillai and P. P. Vaidyanathan, Cosine-modulated FIR filter banks satisfying perfect reconstruction, *IEEE Trans. Signal Process.,* **40**: 770–783, 1992.
17. N. J. Fliege, *Multirate Dig. Signal Process.,* Chichester, UK: Wiley, 1994.
18. T. Nguyen and R. David Koilpillai, The theory and design of arbitrary-length cosine-modulated FIR filter banks and wavelets, satisfying perfect reconstruction, *IEEE Trans. Signal Process.,* **44**: 473–483, 1996.

EDUARDO A. B. DA SILVA
PAULO S. R. DINIZ
Program of Electrical Eng. and Electronics Dept.
COPPE/EE/Federal University of Rio de Janeiro

TIMING ANALYSIS. See CRITICAL PATH ANALYSIS.
TOKEN RING. See LOCAL AREA NETWORKS (LANS).