8 Multirate systems

8.1 Introduction

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

Here, we describe such systems which are generally referred to as multirate systems. Multirate systems are used in several applications, ranging from digital filter design to signal coding and compression, and have been increasingly present in modern digital systems.

First, we study the basic operations of decimation and interpolation, and show how arbitrary rational sampling-rate changes can be implemented with them. Then, we describe properties pertaining to the multirate systems, namely their valid inverse operations and the noble identities. With these properties introduced, the next step is to present the polyphase decompositions and the commutator models, which are key tools in multirate systems. The design of decimation and interpolation filters is also addressed. A step further is to deal with filter design techniques which use decimation and interpolation in order to achieve a prescribed set of filter specifications. In addition, some useful forms to represent single-input single-output (SISO) systems in block form are presented. The effects of multirate systems in random signals are also discussed in this chapter. Finally, MATLAB experiments and related functions which aid in the design and implementation of multirate systems are briefly described.

8.2 Basic principles

Intuitively, any sampling-rate change can be effected by recovering the band-limited analog signal $x_a(t)$ from its samples $x(m)$ and then resampling it with a different sampling rate, thus generating a different discrete version of the signal, $x'(n)$. Of course, the intermediate analog signal $x_a(t)$ must be filtered so that it can be resampled without aliasing. One possible way to do so is described here.

Suppose we have a digital signal $x(m)$ that was generated from an analog signal $x_a(t)$ with sampling period T_1; that is, $x(m) = x_a(mT_1)$, for $m \in \mathbb{Z}$. In order to avoid aliasing in the process, it is assumed that $x_a(t)$ is band-limited to $[-\pi/T_1, \pi/T_1]$. Therefore, replacing each sample of the signal by an impulse proportional to it, we have that the equivalent analog signal is

$$x_i(t) = \sum_{m=-\infty}^{\infty} x(m)\delta(t - mT_1), \tag{8.1}$$

whose spectrum is periodic with period $2\pi/T_1$. In order to recover the original analog signal $x_a(t)$ from $x_i(t)$, the repetitions of the spectrum must be discarded. Therefore, as seen in Section 1.6, $x_i(t)$ must be filtered with a filter $h(t)$ whose ideal frequency response $H(j\omega)$ is

$$H(j\omega) = \begin{cases} 1, & \omega \in [-\pi/T_1, \pi/T_1] \\ 0, & \text{otherwise} \end{cases} \tag{8.2}$$

and then

$$x_a(t) = x_i(t) * h(t) = \frac{1}{T_1} \sum_{m=-\infty}^{\infty} x(m)\text{sinc}\left[\frac{\pi}{T_1}(t - mT_1)\right]. \tag{8.3}$$

Then, resampling $x_a(t)$ above with period T_2 to generate the digital signal $x'(n) = x_a(nT_2)$, for $n \in \mathbb{Z}$, we have that

$$x'(n) = \frac{1}{T_1} \sum_{m=-\infty}^{\infty} x(m)\text{sinc}\left[\frac{\pi}{T_1}(nT_2 - mT_1)\right]. \tag{8.4}$$

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, then the filter in Equation (8.2) must have a frequency response equal to zero for $\omega \notin [-\pi/T_2, \pi/T_2]$. As seen in Section 1.6, since Equation (8.4) consists of infinite summations involving the sinc function, it is not of practical use. In general, for rational sampling-rate changes, which covers most cases of interest, one can derive expressions working solely in the discrete-time domain. This is covered in the next sections, where three special cases are considered: decimation by an integer factor M, interpolation by an integer factor L, and sampling-rate change by a rational factor L/M.

8.3 Decimation

To decimate or subsample a digital signal $x(m)$ by a factor of M is to reduce its sampling rate M times. This is equivalent to keeping only every Mth sample of the signal. This operation is represented as in Figure 8.1 and exemplified in Figure 8.2 for the case $M = 2$.

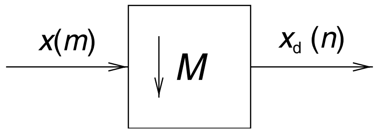

Fig. 8.1. Block diagram representing the decimation by a factor of M.

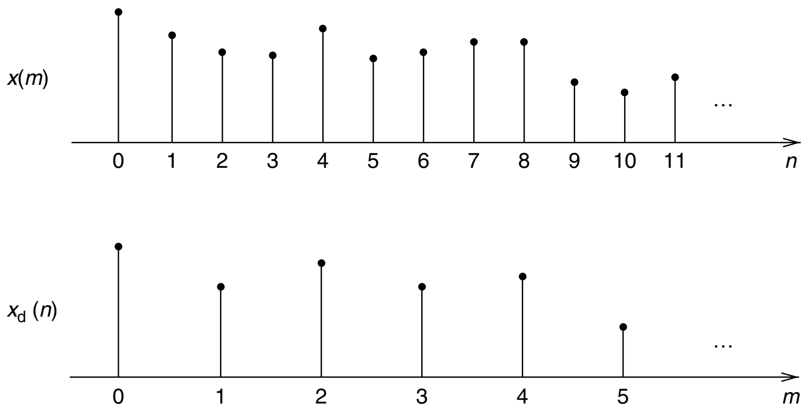

Fig. 8.2. Decimation by 2. $x(m) = \ldots x(0)\ x(1)\ x(2)\ x(3)\ x(4) \ldots$; $x_{\mathrm{d}}(n) = \ldots x(0)\ x(2)\ x(4)\ x(6)\ x(8) \ldots$

The relation between the decimated signal and the original one is, therefore, very straightforward; that is:

$$x_{\mathrm{d}}(n) = x(nM). \tag{8.5}$$

In the frequency domain, if the spectrum of $x(m)$ is $X(\mathrm{e}^{\mathrm{j}\omega})$, the spectrum of the decimated signal, $X_{\mathrm{d}}(\mathrm{e}^{\mathrm{j}\omega})$, becomes

$$X_{\mathrm{d}}(\mathrm{e}^{\mathrm{j}\omega}) = \frac{1}{M} \sum_{k=0}^{M-1} X\left[\mathrm{e}^{\mathrm{j}(\omega-2\pi k)/M}\right]. \tag{8.6}$$

Such a result is reached by first defining $x'(m)$ as

$$x'(m) = \begin{cases} x(m), & m = nM, n \in \mathbb{Z} \\ 0, & \text{otherwise} \end{cases}, \tag{8.7}$$

which can also be written as

$$x'(m) = x(m) \sum_{n=-\infty}^{\infty} \delta(m - nM) \tag{8.8}$$

The Fourier transform $X_{\rm d}({\rm e}^{{\rm j}\omega})$ is then given by

$$\begin{aligned}
X_{\rm d}({\rm e}^{{\rm j}\omega}) &= \sum_{n=-\infty}^{\infty} x_{\rm d}(n){\rm e}^{-{\rm j}\omega n} \\
&= \sum_{n=-\infty}^{\infty} x(nM){\rm e}^{-{\rm j}\omega n} \\
&= \sum_{n=-\infty}^{\infty} x'(nM){\rm e}^{-{\rm j}\omega n} \\
&= \sum_{l=-\infty}^{\infty} x'(l){\rm e}^{-{\rm j}(\omega/M)l} \\
&= X'({\rm e}^{{\rm j}\omega/M}).
\end{aligned} \tag{8.9}$$

But, from Equation (8.8) (see also Equation (2.237) and Exercise 2.15):

$$\begin{aligned}
X'({\rm e}^{{\rm j}\omega}) &= X({\rm e}^{{\rm j}\omega}) \circledast \mathcal{F}\left\{\sum_{n=-\infty}^{\infty} \delta(m-nM)\right\} \\
&= X({\rm e}^{{\rm j}\omega}) \circledast \frac{2\pi}{M}\sum_{k=0}^{M-1} \delta\left(\omega - \frac{2\pi k}{M}\right) \\
&= \frac{1}{M}\sum_{k=0}^{M-1} X\left\{{\rm e}^{{\rm j}[\omega-(2\pi k/M)]}\right\}.
\end{aligned} \tag{8.10}$$

Then, from Equation (8.9),

$$X_{\rm d}({\rm e}^{{\rm j}\omega}) = X'\left({\rm e}^{{\rm j}\omega/M}\right) = \frac{1}{M}\sum_{k=0}^{M-1} X\left[{\rm e}^{{\rm j}(\omega-2\pi k)/M}\right], \tag{8.11}$$

which is the same as Equation (8.6).

As illustrated in Figure 8.3 for $M = 2$, Equation (8.6) means that the spectrum of $x_{\rm d}(n)$ is composed of copies of the spectrum of $x(m)$ expanded by M and repeated with period 2π (which is equivalent to copies of the spectrum of $x(m)$ repeated with period $2\pi/M$ and then expanded by M). This implies that, in order to avoid aliasing after decimation, the bandwidth of the signal $x(m)$ must be limited to the interval $[-\pi/M, \pi/M]$. Therefore, the decimation operation is generally preceded by a lowpass filter (see Figure 8.4), which approximates the following frequency response:

$$H_{\rm d}({\rm e}^{{\rm j}\omega}) = \begin{cases} 1, & \omega \in [-\pi/M, \pi/M] \\ 0, & \text{otherwise} \end{cases} \tag{8.12}$$

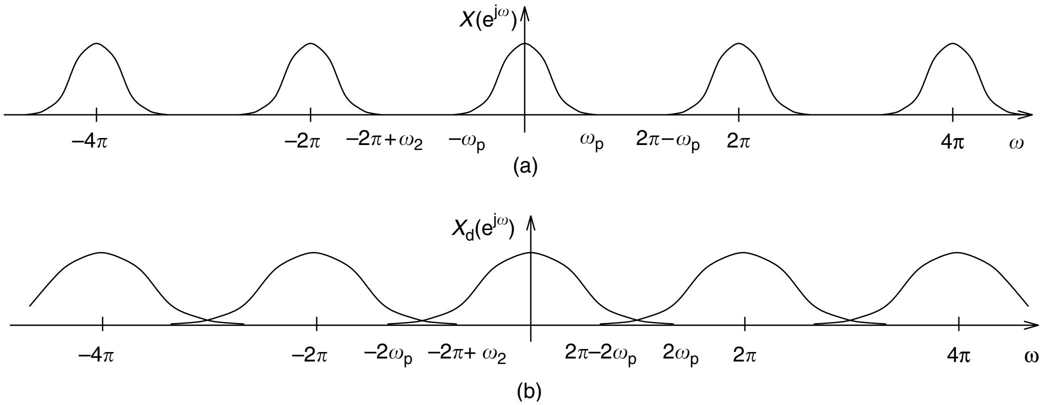

Fig. 8.3. **Signal spectra of: (a) original digital signal; (b) decimated signal by a factor of 2.**

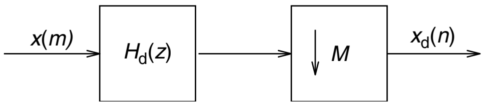

Fig. 8.4. **General decimation operation.**

If we then include the filtering operation, the decimated signal is obtained by retaining every Mth sample of the convolution of the signal $x(m)$ with the filter impulse response $h_d(m)$; that is:

$$x_d(n) = \sum_{m=-\infty}^{\infty} x(m)h_d(nM - m). \tag{8.13}$$

Some important facts must be noted about the decimation operation (Crochiere & Rabiner, 1983):

- It is time varying; that is, if the input signal $x(m)$ is shifted, the output signal will not in general be a shifted version of the previous output. More precisely, let $\mathcal{D}_M$ be the decimation-by-M operator. If $x_d(n) = \mathcal{D}_M\{x(m)\}$, then in general $\mathcal{D}_M\{x(m-k)\} \neq x_d(n-l)$, unless $k = rM$, when $\mathcal{D}_M\{x(m-k)\} = x_d(n-r)$. Because of this property, the decimation is referred to as a periodically time-invariant operation.
- Referring to Equation (8.13), one can see 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 that of a usual filtering operation (Peled & Liu, 1985). This is not valid in general for IIR filters, because in such cases one needs all past outputs to compute the present output, unless the transfer function is of the type $H(z) = N(z)/D(z^M)$ (Martinez & Parks, 1979; Ansari & Liu, 1983).
- 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

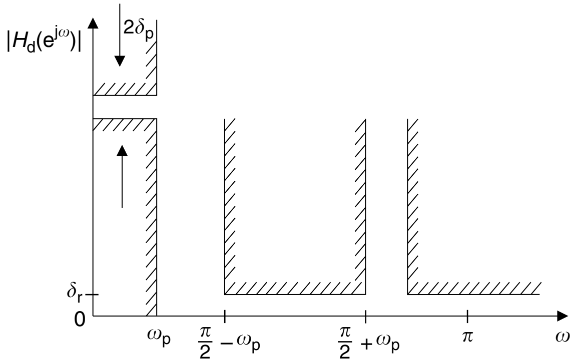

Fig. 8.5. **Specifications of a decimation filter for $M = 4$.**

relaxed, yielding the following specifications for $H_d(z)$:

$$H_d(e^{j\omega}) = \begin{cases} 1, & |\omega| \in [0, \omega_p] \\ 0, & |\omega| \in [(2\pi k/M) - \omega_p, (2\pi k/M) + \omega_p], \quad k = 1, 2, \ldots, M-1. \end{cases} \tag{8.14}$$

The decimation filter can be efficiently designed using the optimum FIR approximation methods described in Chapter 5. In order to do so, one has to define the following parameters:

$$\left.\begin{array}{l} \delta_p: \text{ passband ripple} \\ \delta_r: \text{ stopband attenuation} \\ \omega_p: \text{ passband cutoff frequency} \\ \omega_{r_1} = (2\pi/M) - \omega_p: \text{ first stopband edge} \end{array}\right\}. \tag{8.15}$$

However, in general, it is more efficient to design a multiband filter according to Equation (8.14), as illustrated in Figure 8.5 and exemplified below.

Example 8.1. A signal that carries useful information only in the range $0 \leq \omega \leq 0.1\omega_s$ must be decimated by a factor of $M = 4$. Design a linear-phase decimation filter satisfying the following specifications:

$$\left.\begin{array}{l} \delta_p = 0.001 \\ \delta_r = 5 \times 10^{-5} \\ \Omega_s = 20\,000 \text{ Hz} \end{array}\right\}. \tag{8.16}$$

Solution

According to the specifications, the stopband edges of the decimation filter should be located at $((\Omega_s/4) - (\Omega_s/10))$ and $((\Omega_s/4) + (\Omega_s/10))$ in the first stopband, and $((\Omega_s/2) - (\Omega_s/10))$ in the second stopband. As a result, there is a "don't care" band between $((\Omega_s/4) + (\Omega_s/10))$ and $((\Omega_s/2) - (\Omega_s/10))$. Designing the filter using the Chebyshev

Table 8.1. Coefficient $h(0)$ to $h(42)$ for impulse response of the decimation filter.

$h(0) = 3.8208E{-}06$	$h(15) = 2.5170E{-}03$	$h(30) = -2.9906E{-}03$
$h(1) = -1.5078E{-}04$	$h(16) = 3.7016E{-}03$	$h(31) = 1.2798E{-}02$
$h(2) = -2.4488E{-}04$	$h(17) = 2.0456E{-}03$	$h(32) = 2.5575E{-}02$
$h(3) = -3.4356E{-}04$	$h(18) = -5.8022E{-}04$	$h(33) = 2.3561E{-}02$
$h(4) = -3.7883E{-}04$	$h(19) = -4.0164E{-}03$	$h(34) = 7.6551E{-}03$
$h(5) = 1.4857E{-}06$	$h(20) = -6.3092E{-}03$	$h(35) = -1.9703E{-}02$
$h(6) = 3.5092E{-}04$	$h(21) = -4.1002E{-}03$	$h(36) = -4.4867E{-}02$
$h(7) = 7.4044E{-}04$	$h(22) = 1.8340E{-}04$	$h(37) = -4.7659E{-}02$
$h(8) = 9.4756E{-}04$	$h(23) = 6.0511E{-}03$	$h(38) = -2.0785E{-}02$
$h(9) = 2.5364E{-}04$	$h(24) = 1.0189E{-}02$	$h(39) = 3.9424E{-}02$
$h(10) = -5.2335E{-}04$	$h(25) = 7.5145E{-}03$	$h(40) = 1.1942E{-}01$
$h(11) = -1.4509E{-}03$	$h(26) = 8.3120E{-}04$	$h(41) = 1.9216E{-}01$
$h(12) = -1.9966E{-}03$	$h(27) = -8.8128E{-}03$	$h(42) = 2.3804E{-}01$
$h(13) = -8.6587E{-}04$	$h(28) = -1.6037E{-}02$	
$h(14) = 6.3635E{-}04$	$h(29) = -1.3217E{-}02$	

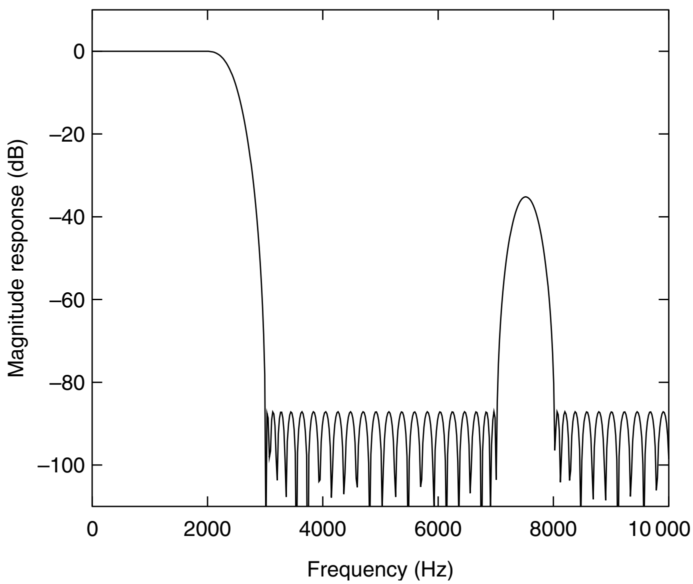

Fig. 8.6. **Magnitude response of the decimation filter for $M = 4$.**

approach from Section 5.6.2, we have that, in order to satisfy the specifications, the minimum required order is 85. The resulting magnitude response of one such filter is shown in Figure 8.6, where it can be observed that between 7000 and 8000 Hz is located a "don't care" band as expected.

Table 8.1 shows the filter coefficients, where, since the filter is linear phase, only half of the coefficients are included. △

8.4 Interpolation

To interpolate or upsample a digital signal $x(m)$ by a factor of L is to include $L-1$ zeros between its samples. This operation is represented in Figure 8.7.

The interpolated signal is then given by

$$\hat{x}_{\mathrm{i}}(n) = \begin{cases} x\,(n/L), & n = kL,\ k \in \mathbb{Z} \\ 0, & \text{otherwise} \end{cases}. \tag{8.17}$$

The interpolation operation is exemplified in Figure 8.8 for the case $L = 2$.

In the frequency domain, if the spectrum of $x(m)$ is $X(\mathrm{e}^{\mathrm{j}\omega})$, it is straightforward to see that the spectrum of the interpolated signal $\hat{X}_{\mathrm{i}}(\mathrm{e}^{\mathrm{j}\omega})$ becomes (Crochiere & Rabiner, 1983)

$$\hat{X}_{\mathrm{i}}(\mathrm{e}^{\mathrm{j}\omega}) = X(\mathrm{e}^{\mathrm{j}\omega L}). \tag{8.18}$$

Figure 8.9 shows the spectra of the signals $x(m)$ and $\hat{x}_{\mathrm{i}}(n)$ for an interpolation factor of L.

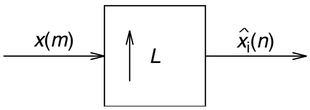

Fig. 8.7. **Interpolation by a factor of *L*.**

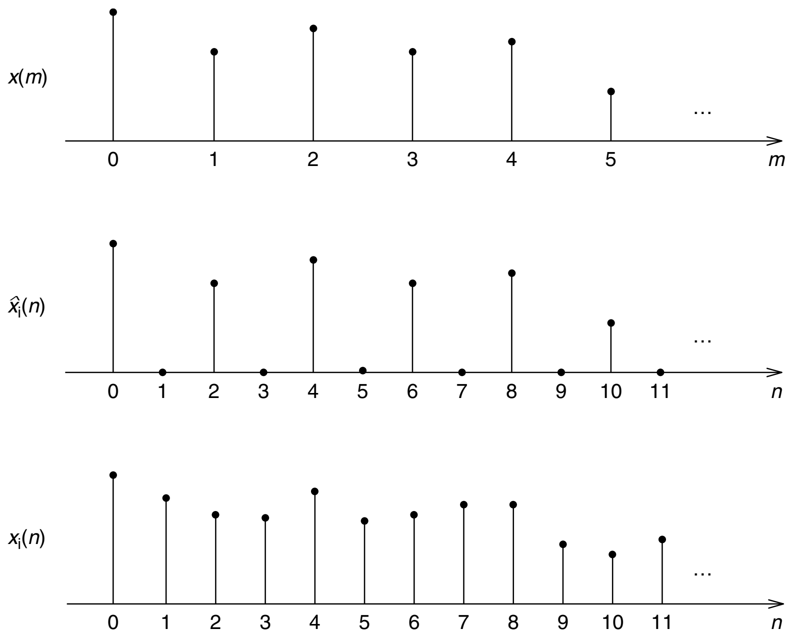

Fig. 8.8. **Interpolation by 2. $x(m)$: original signal; $\hat{x}_{\mathrm{i}}(n)$: signal with zeros inserted between samples; $x_{\mathrm{i}}(n)$: interpolated signal after filtering by $H_{\mathrm{i}}(z)$. ($x(m) = \ldots x(0)\ x(1)\ x(2)\ x(3)\ x(4)\ x(5)\ x(6) \ldots$; $\hat{x}_{\mathrm{i}}(n) = \ldots x(0)\ 0\ x(1)\ 0\ x(2)\ 0\ x(3) \ldots$; $x_{\mathrm{i}}(n) = \ldots x_{\mathrm{i}}(0)\ x_{\mathrm{i}}(1)\ x_{\mathrm{i}}(2)\ x_{\mathrm{i}}(3)\ x_{\mathrm{i}}(4)\ x_{\mathrm{i}}(5)\ x_{\mathrm{i}}(6) \ldots$)**

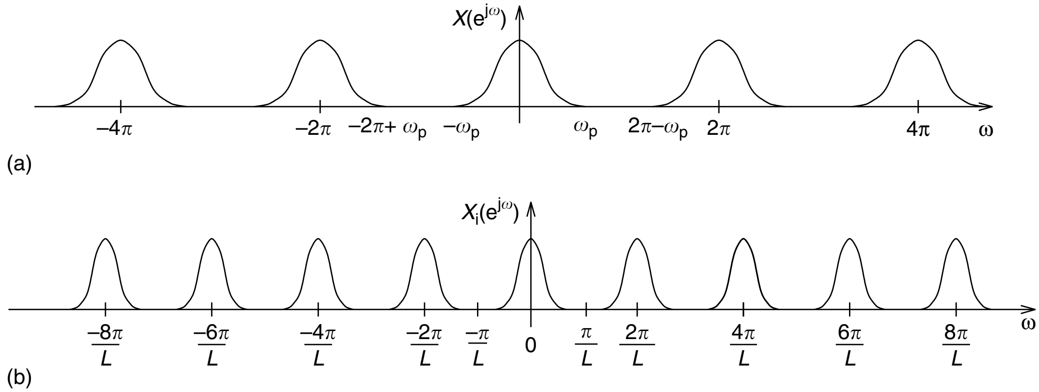

Fig. 8.9. **Signal spectra of: (a) original digital signal; (b) interpolated signal by a factor of *L*.**

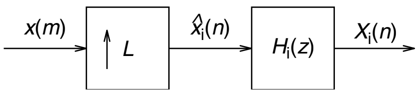

Fig. 8.10. **General interpolation operation.**

Since the spectrum of the original digital signal is periodic with period 2π, the spectrum of the interpolated signal has period $2\pi/L$. In order to obtain a smooth interpolated version of $x(m)$, the spectrum of the interpolated signal must be a compressed version of $X(e^{j\omega})$ in the frequency range $[-\pi, \pi)$, without any spectrum repetitions. This can be obtained by filtering out the repetitions of the spectrum of $\hat{x}_i(n)$ outside $[-\pi/L, \pi/L]$. Thus, the interpolation operation is generally followed by a lowpass filter (see Figure 8.10) which approximates the following frequency response:

$$H_i(e^{j\omega}) = \begin{cases} L, & \omega \in [-\pi/L, \pi/L] \\ 0, & \text{otherwise} \end{cases}. \tag{8.19}$$

The interpolation operation is thus equivalent to the convolution of the interpolation filter impulse response $h_i(n)$ with the signal $\hat{x}_i(n)$ defined in Equation (8.17). Considering that the only nonzero samples of $\hat{x}_i(n)$ are the ones having a multiple of L index, Equation (8.17) can be rewritten as

$$\hat{x}_i(kL) = \begin{cases} x(k), & k \in \mathbb{Z} \\ 0, & \text{otherwise} \end{cases}. \tag{8.20}$$

With the help of the above equation, it is easy to see that, in the time domain, the filtered interpolated signal becomes

$$x_i(n) = \sum_{m=-\infty}^{\infty} \hat{x}_i(m)h(n-m) = \sum_{k=-\infty}^{\infty} x(k)h(n-kL). \tag{8.21}$$

Some important facts must be noted about the interpolation operation (Crochiere & Rabiner, 1983):

- As opposed to the decimation operation, the interpolation does not entail loss of information. More precisely, if $\mathcal{I}_L$ is the interpolation-by-L operator, Equations (8.5) and (8.20) imply that $\mathcal{D}_L\{\mathcal{I}_L\{x(m)\}\} = x(m)$; that is, the interpolation operation is invertible. However, $\{\mathcal{I}_L\{x(m-k)\}\} = x_{\rm i}(n-kL)$, which means that the interpolation is inherently time varying.
- Referring to Equation (8.21), one can see that the computation of the output of the filter $H_{\rm i}(z)$ uses only one out of every L samples of the input signal, because the remaining samples are zero. This means that its implementation complexity can be made L times simpler than that of a usual filtering operation.
- If the signal $x(m)$ is band-limited to $[-\omega_{\rm p}, \omega_{\rm p}]$, the repetitions of the spectrum will only appear in a neighborhood of radius $\omega_{\rm p}/L$ 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

$$H_{\rm i}(e^{j\omega}) = \begin{cases} L, & |\omega| \in [0, \omega_{\rm p}/L] \\ 0, & |\omega| \in [(2\pi k - \omega_{\rm p})/L, (2\pi k + \omega_{\rm p})/L], \quad k = 1, 2, \ldots, L-1. \end{cases} \tag{8.22}$$

The gain factor L in Equations (8.19) and (8.22) can be understood by noting that, since we are maintaining one out of every L samples of the signal, the average value of the signal decreases by a factor L and therefore, the gain of the interpolating filter must be L to compensate for this.

8.4.1 Examples of interpolators

Supposing $L = 2$, two common examples can be devised, as shown in Figure 8.11:

- Zero-order hold: $x(2n+1) = x(2n)$. From Equation (8.21), this is equivalent to having $h(0) = h(1) = 1$; that is, $H_{\rm i}(z) = 1 + z^{-1}$.
- Linear interpolator: $x(2n+1) = \frac{1}{2}[x(2n) + x(2n+2)]$. From Equation (8.21), this is equivalent to having $h(-1) = \frac{1}{2}$, $h(0) = 1$, and $h(1) = \frac{1}{2}$, that is, $H_{\rm i}(z) = \frac{1}{2}(z+2+z^{-1})$.

Interesting examples of interpolators are the Lth-band filters. They are filters that, when used as interpolators by L, keep the original samples of the signal to be interpolated. This can be stated more precisely by referring to Equation (8.21). There, if one decimates by L the interpolated signal $x_{\rm i}(n)$ generated with an Lth-band filter, one obtains the original signal $x(m)$. This is equivalent to saying that

$$x_{\rm i}(mL) = x(m). \tag{8.23}$$

In this case, Equation (8.21) becomes

$$x_{\rm i}(mL) = \sum_{k=-\infty}^{\infty} x(k)h(mL - kL) = x(m). \tag{8.24}$$

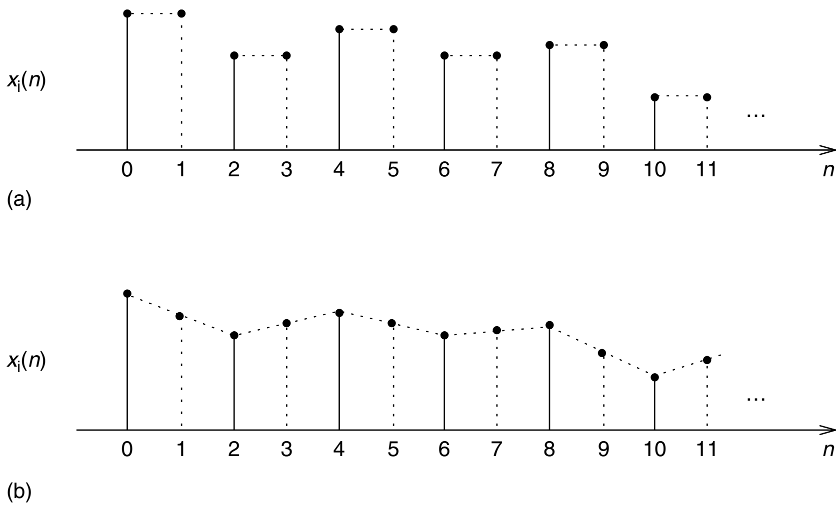

Fig. 8.11. **Examples of interpolators: (a) zero-order hold; (b) linear interpolator.**

This only happens if

$$h(mL - kL) = \begin{cases} 1, & m = k \\ 0, & m \neq k \end{cases}. \tag{8.25}$$

That is, the samples of $h(n)$ that are multiples of L are zero, except the one for $n = 0$, which should be equal to one.

Note that both the zeroth-order hold and the first-order hold are two-band filters, or, as is commonly said, half-band filters.

8.5 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, as represented in Figure 8.12.

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

$$H(e^{j\omega}) = \begin{cases} L, & |\omega| \leq \min\{\pi/L, \pi/M\} \\ 0, & \text{otherwise} \end{cases}. \tag{8.26}$$

Similar to the cases 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 Equation (8.22) and the specifications in

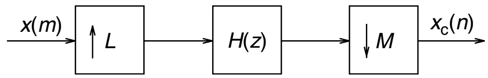

Fig. 8.12. **Sampling rate change by a factor of L/M.**

Equation (8.14) with ω_p replaced by ω_p/L. Since L and M can be assumed, without loss of generality, to be relatively prime, this yields

$$H(e^{j\omega}) = \begin{cases} L, & |\omega| < \min\{\omega_p/L, \pi/M\} \\ 0, & \min\{(2\pi/L) - (\omega_p/L), (2\pi/M) - (\omega_p/L)\} \le |\omega| \le \pi \end{cases}. \tag{8.27}$$

8.6 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, does $\mathcal{D}_M\mathcal{I}_M = \mathcal{I}_M\mathcal{D}_M =$ 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 as long as the two operations are properly aligned, otherwise a null signal will result.

On the other hand, $\mathcal{I}_M\mathcal{D}_M$ is not the identity operator in general. This is so because the decimation operation removes $(M-1)$ out of M samples of the signal and the interpolation operation inserts $(M-1)$ zeros between samples. Then, their cascade is equivalent to replacing $(M-1)$ out of M samples of the signal with zeros. However, if the decimation-by-M operation is preceded by a band-limiting filter for the interval $[-\pi/M, \pi/M]$ (see Equation (8.12)), and the interpolation operation is followed by the same filter (as illustrated in Figure 8.13), then $\mathcal{I}_M\mathcal{D}_M$ becomes the identity operation. This can be easily confirmed in the frequency domain, as the band-limiting filter avoids aliasing after decimation, which makes the decimation operation remain invertible. After interpolation by M, there are images of the spectrum of the signal in the intervals $[\pi k/M, \pi(k+1)/M]$, for $k = -M, (-M+1), \ldots, (M-1)$. However, the second band-limiting filter keeps only the image inside $[-\pi/M, \pi/M]$, which corresponds to the spectrum of the original signal.

We now discuss under which conditions the decimation and interpolation operations are commutative; that is, when the connection $\mathcal{D}_M\mathcal{I}_L$ as depicted in Figure 8.14a is equivalent to $\mathcal{I}_L\mathcal{D}_M$ shown in Figure 8.14b. We have already seen above that when $M = L$ they are not equivalent. Usually, these interconnections are not equivalent, unless M and L are relatively prime numbers. In the connection of Figure 8.14a, the output signal is given by

$$y(m) = \begin{cases} x(mM/L), & m = kL, \ k \in \mathbb{Z} \\ 0, & \text{otherwise} \end{cases}, \tag{8.28}$$

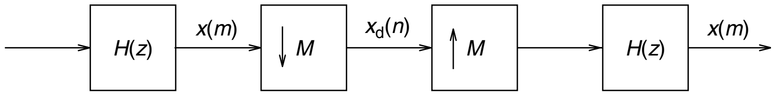

Fig. 8.13. **Decimation followed by interpolation.**

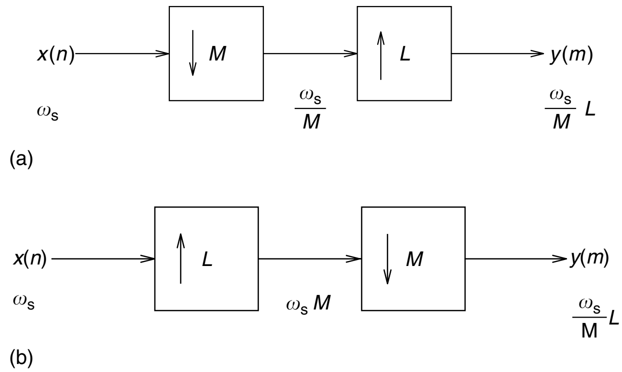

Fig. 8.14. **Cascade operations: (a) decimation/interpolation; (b) interpolation/decimation.**

whereas in the connection of Figure 8.14b the output signal is given by (see Exercise 8.2)

$$y(m) = \begin{cases} x\,(mM/L)\,, & mM = kL,\ k \in \mathbb{Z} \\ 0, & \text{otherwise} \end{cases}. \tag{8.29}$$

Note that the condition in Equation (8.28), $m = kL$, $k \in \mathbb{Z}$, implies the condition in Equation (8.29); that is, $mM = kML = k'L$, $k' \in \mathbb{Z}$. On the other hand, the condition in Equation (8.29), $mM = kL$, $k \in \mathbb{Z}$, only implies that $m = k'L$, $k' \in \mathbb{Z}$ if M and L have no common multiple; that is, if they are relatively prime.

8.7 Noble identities

The noble identities are depicted in Figure 8.15. They have to do with the commutation of the filtering and decimation or interpolation operations, and are very useful in analyzing multirate systems and filter banks.

The identity in Figure 8.15a means that to decimate a signal by M and then filter it with $H(z)$ is equivalent to filtering the signal with $H(z^M)$ and then decimating the result by M. A filter $H(z^M)$ is one whose impulse response is equal to the impulse response of $H(z)$ with $(M-1)$ zeros inserted between adjacent samples. Mathematically, it can be stated as

$$\mathcal{D}_M\{X(z)\}H(z) = \mathcal{D}_M\{X(z)H(z^M)\}, \tag{8.30}$$

where $\mathcal{D}_M$ is the decimation-by-M operator.

The identity in Figure 8.15b means that to filter a signal with $H(z)$ and then interpolate it by M is equivalent to interpolating it by M and then filtering it with $H(z^M)$. Mathematically,

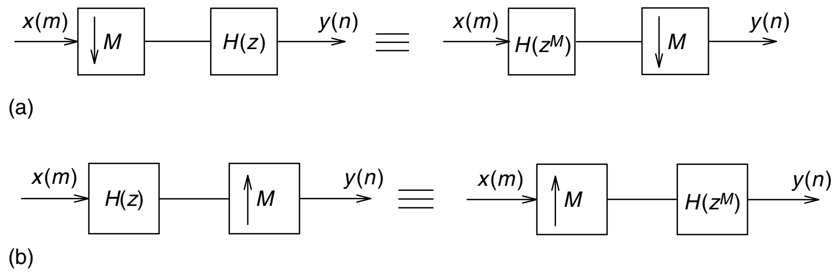

Fig. 8.15. Noble identities: (a) decimation; (b) interpolation.

it is stated as

$$\mathcal{I}_M\{X(z)H(z)\} = \mathcal{I}_M\{X(z)\}H(z^M), \tag{8.31}$$

where $\mathcal{I}_M$ is the interpolation-by-M operator.

In order to prove the identity in Figure 8.15a, one begins by rewriting Equation (8.6), which gives the Fourier transform of the decimated signal $x_{\mathrm{d}}(n)$ as a function of the input signal $x(m)$, in the z domain; that is:

$$X_{\mathrm{d}}(z) = \frac{1}{M}\sum_{k=0}^{M-1} X\left(z^{1/M}\mathrm{e}^{-\mathrm{j}2\pi k/M}\right). \tag{8.32}$$

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

$$Y(z) = H(z)X_{\mathrm{d}}(z) = \frac{1}{M}H(z)\sum_{k=0}^{M-1} X\left(z^{1/M}\mathrm{e}^{-\mathrm{j}2\pi k/M}\right). \tag{8.33}$$

For the filter $H(z^M)$ followed by the decimator, if $U(z) = X(z)H(z^M)$, then we have, from Equation (8.32), that

$$\begin{aligned} Y(z) &= \frac{1}{M}\sum_{k=0}^{M-1} U\left(z^{1/M}\mathrm{e}^{-\mathrm{j}2\pi k/M}\right) \\ &= \frac{1}{M}\sum_{k=0}^{M-1} X\left(z^{1/M}\mathrm{e}^{-\mathrm{j}2\pi k/M}\right)H\left(z\mathrm{e}^{-\mathrm{j}2\pi Mk/M}\right) \\ &= \frac{1}{M}\sum_{k=0}^{M-1} X\left(z^{1/M}\mathrm{e}^{-\mathrm{j}2\pi k/M}\right)H(z), \end{aligned} \tag{8.34}$$

which is the same as Equation (8.33), and the identity is proved.

Proof of the identity in Figure 8.15b is straightforward, as $H(z)$ followed by an interpolator gives $Y(z) = H(z^M)X(z^M)$, which is the same as the expression for an interpolator followed by $H(z^M)$.