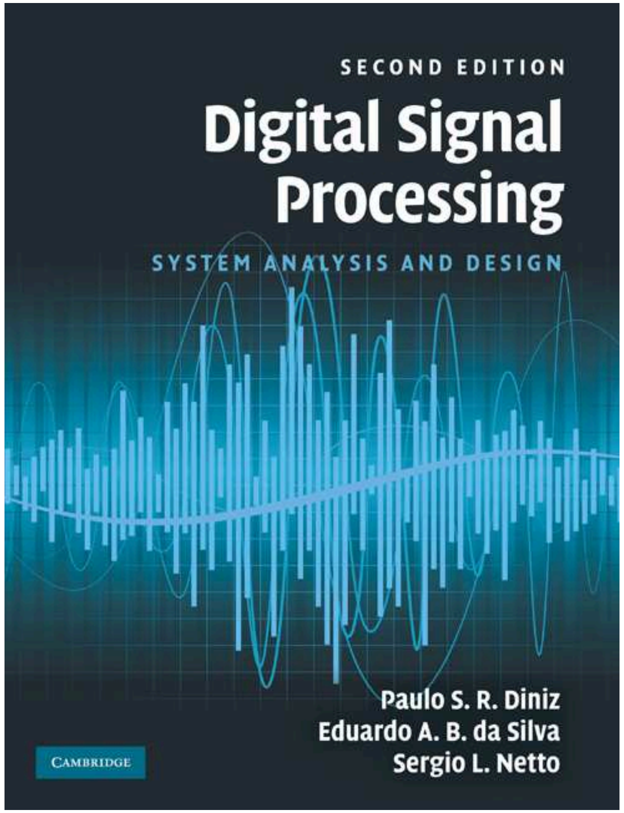
Multirate systems



Paulo S. R. Diniz

Eduardo A. B. da Silva

Sergio L. Netto

diniz,eduardo,sergioln@lps.ufrj.br

September 2010

Contents

Contents

- Decimation and interpolation for efficient filter implementation
- Overlapped block filtering
- Random signals in multirate systems
- Do-It-Yourself - Multirate systems

Introduction

- In many applications of digital signal processing it is necessary that different sampling rates 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 non-overlapping narrow-band channels in order to be transmitted.
  - In such case, each narrow-band 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.

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, $x'(n)$.
- Supposing:

  $x(m)$ generated from an analog signal $x_a(t)$ with sampling period T_1, band-limited to $[-\frac{\pi}{T_1}, \frac{\pi}{T_1}]$, ($x(m) = x_a(mT_1)$)
-
$$x_i(t) = \sum_{m=-\infty}^{\infty} x(m)\delta(t - mT_1) \tag{1}$$

  whose spectrum is periodic with period $\frac{2\pi}{T_1}$.

Basic principles

- To recover the original analog signal $x_a(t)$ from $x_i(t)$, the repetitions of the spectrum must be discarded.

- $x_i(t)$ must be filtered with a filter $h(t)$ whose ideal frequency response $H(j\omega)$ is:

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

and then

$$x_a(t) = x_i(t) * h(t) = \frac{1}{T_1} \sum_{m=-\infty}^{\infty} x(m)\text{sinc}\left[\frac{\pi}{T_1}(t - mT_1)\right] \tag{3}$$

Basic principles

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

$$x'(n) = \frac{1}{T_1} \sum_{m=-\infty}^{\infty} x(m) \text{sinc}\left[\frac{\pi}{T_1}(nT_2 - mT_1)\right] \tag{4}$$

This is the general equation governing sampling-rate changes.

Basic principles

- There is no restriction on the values of T_1 and T_2.
- If $T_2 > T_1$ and aliasing is to be avoided, the filter in equation (2) must have a frequency response equal to zero for $\omega \notin [-\frac{\pi}{T_2}, \frac{\pi}{T_2}]$.
- Since equation (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.
- 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.

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.
- It is represented as in Figure 1.

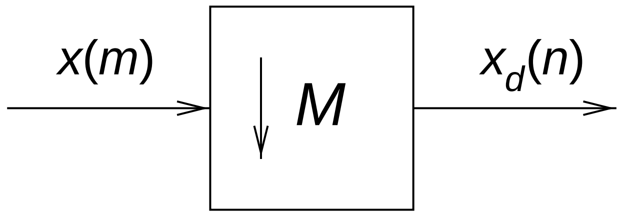


Figure 1: Block diagram representing the decimation by a factor of M.

Decimation

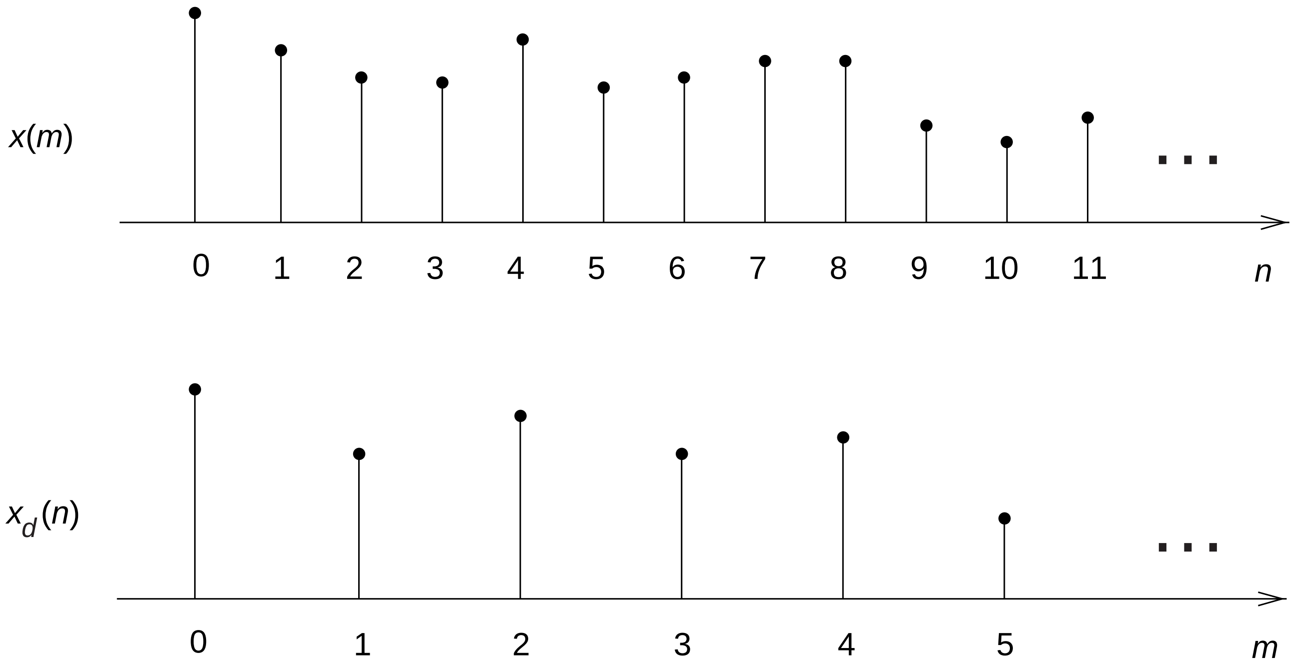


Figure 2: Decimation by 2. $x(m) = \ldots\ x(0)\ x(1)\ x(2)\ x(3)\ x(4)\ \ldots$; $x_d(n) = \ldots\ x(0)\ x(2)\ x(4)\ x(6)\ x(8)\ \ldots$

Decimation

- The relation between the decimated signal and the original one is

$$x_d(n) = x(nM) \tag{5}$$

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

$$X_d(e^{j\omega}) = \frac{1}{M}\sum_{k=0}^{M-1} X\left(e^{j\frac{\omega-2\pi k}{M}}\right) \tag{6}$$

Decimation

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

$$x'(m) = \begin{cases} x(m), & m = nM, n \in \mathbb{Z} \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

which can also be written as

$$x'(m) = x(m) \sum_{n=-\infty}^{\infty} \delta(m - nM) \tag{8}$$

Decimation

- The Fourier transform $X_d(e^{j\omega})$ is then given by

$$
\begin{aligned}
X_d(e^{j\omega}) &= \sum_{n=-\infty}^{\infty} x_d(n)e^{-j\omega n} \\
&= \sum_{n=-\infty}^{\infty} x(nM)e^{-j\omega n} \\
&= \sum_{n=-\infty}^{\infty} x'(nM)e^{-j\omega n} \\
&= \sum_{l=-\infty}^{\infty} x'(l)e^{-j\frac{\omega}{M}l} \\
&=X'(e^{j\frac{\omega}{M}})
\end{aligned}
\tag{9}
$$

Decimation

- But

$$X'(e^{j\omega}) = X(e^{j\omega}) \circledast \mathcal{F}\left\{\sum_{n=-\infty}^{\infty} \delta(m - nM)\right\}$$
$$= X(e^{j\omega}) \circledast \frac{2\pi}{M}\sum_{k=0}^{M-1} \delta\left(\omega - \frac{2\pi k}{M}\right)$$
$$= \frac{1}{M}\sum_{k=0}^{M-1} X\left(e^{j\left(\omega - \frac{2\pi k}{M}\right)}\right) \tag{10}$$

- Then, from equation (9),

$$X_d(e^{j\omega}) = X'\left(e^{j\frac{\omega}{M}}\right) = \frac{1}{M}\sum_{k=0}^{M-1} X\left(e^{j\frac{\omega - 2\pi k}{M}}\right) \tag{11}$$

which is the same as equation (6).

Decimation

- Equation (6) 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 (which is equivalent to copies of the spectrum of $x(m)$ repeated with period $\frac{2\pi}{M}$ and then expanded by M).

- In order to avoid aliasing after decimation, the bandwidth of the signal $x(m)$ must be limited to the interval $[-\frac{\pi}{M}, \frac{\pi}{M}]$.

- The decimation operation is generally preceded by a lowpass filter, which approximates the following frequency response:

$$H_d(e^{j\omega}) = \begin{cases} 1, & \omega \in [-\frac{\pi}{M}, \frac{\pi}{M}] \\ 0, & \text{otherwise} \end{cases} \tag{12}$$

Decimation

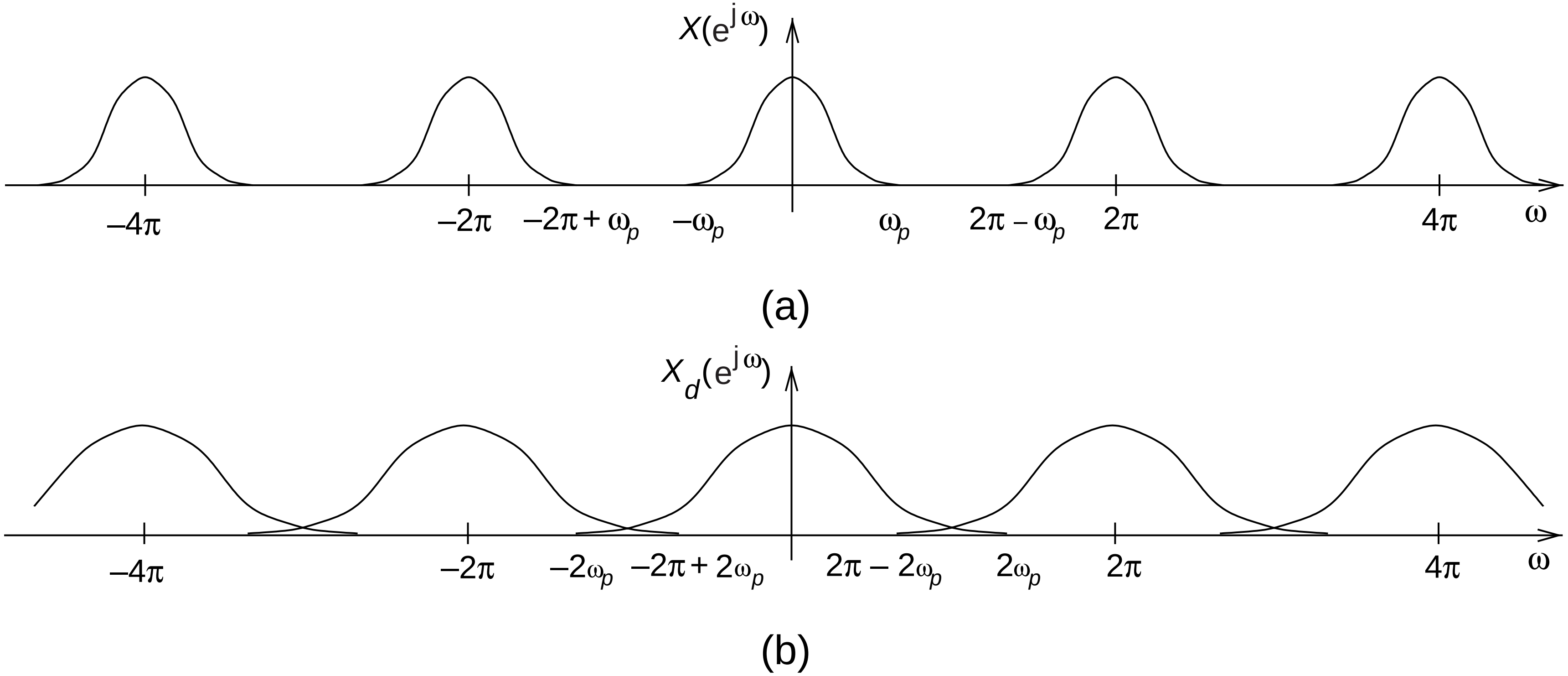


Figure 3: Signal spectra of: (a) original digital signal; (b) decimated signal by a factor of 2.

Decimation

- Including 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 $h_d(m)$

$$x_d(n) = \sum_{m=-\infty}^{\infty} x(m)h_d(nM - m) \tag{13}$$

- Some important facts:
  - 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 (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. 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 $H(z) = \frac{N(z)}{D(z^M)}$

Decimation

- If the frequency range of interest for the signal $x(m)$ is $[-\omega_p, \omega_p]$, with $\omega_p < \frac{\pi}{M}$, one can afford aliasing outside this range. Therefore, the constraints upon the filter can be relaxed, yielding the following specifications for $H_d(z)$

$$H_d(e^{j\omega}) = \begin{cases} 1, & |\omega| \in [0, \omega_p] \\ 0, & |\omega| \in [\frac{2\pi k}{M} - \omega_p, \frac{2\pi k}{M} + \omega_p], \quad k = 1, 2, \ldots, M-1 \end{cases} \tag{14}$$

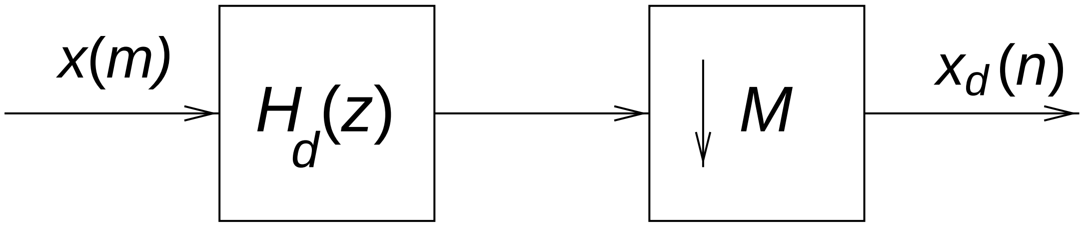


Figure 4: General decimation operation.

Decimation

- The decimation filter can be efficiently designed using the optimum FIR approximation methods.

- In order to do so, one has to define the following parameters:

$$\left.\begin{array}{l} \delta_p \; : \; \text{passband ripple} \\ \delta_r \; : \; \text{stopband attenuation} \\ \omega_p \; : \; \text{passband cutoff frequency} \\ \omega_{r_1} = \left(\frac{2\pi}{M} - \omega_p\right) \; : \; \text{first stopband edge} \end{array}\right\} \qquad (15)$$

- In general, it is more efficient to design a multiband filter according to equation (14), as illustrated below.

Decimation

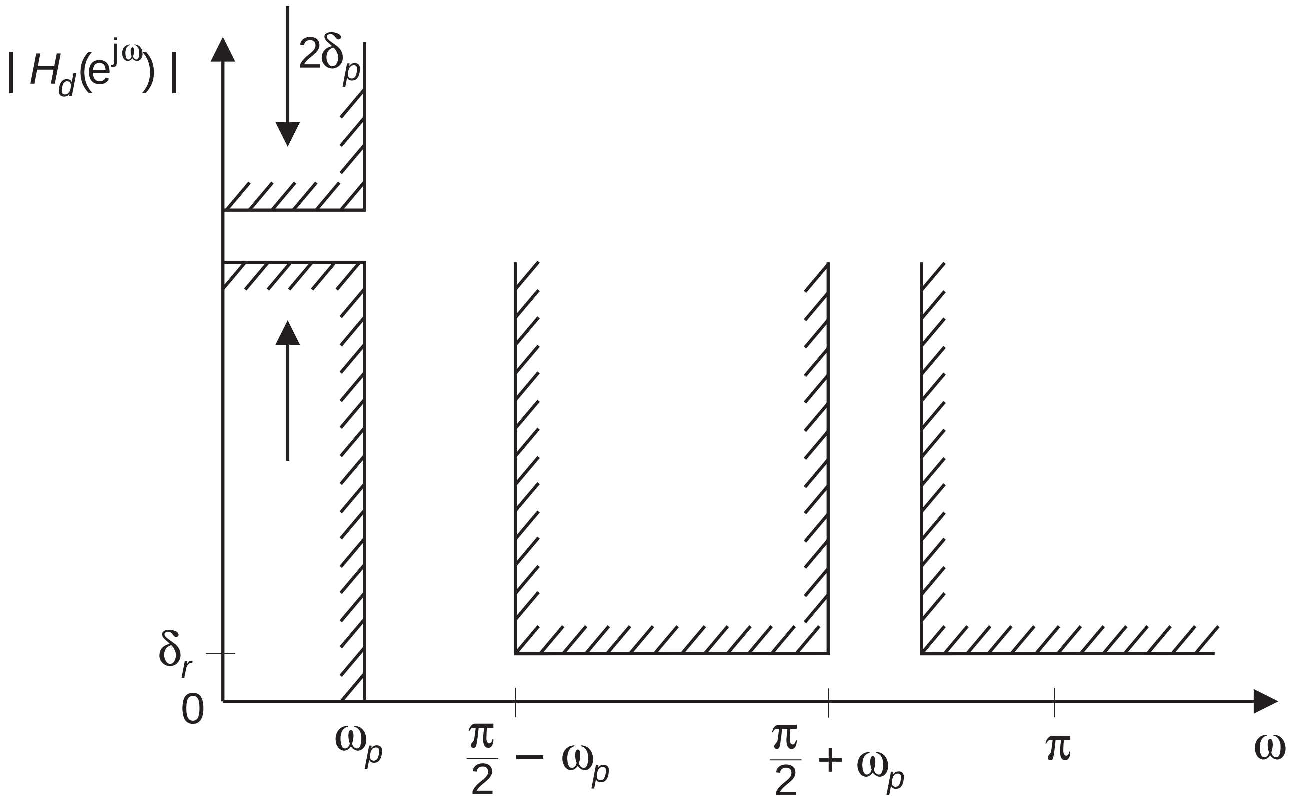


Figure 5: Specifications of a decimation filter for $M = 4$.

Decimation

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:

$$\left.\begin{array}{l} \delta_p = 0.001 \\ \delta_r = 5 \times 10^{-5} \\ \Omega_s = 20\,000 \text{ Hz} \end{array}\right\} \quad (16)$$

Solution

The stopband edges should be located at $\left(\frac{\Omega_s}{4} - \frac{\Omega_s}{10}\right)$ and $\left(\frac{\Omega_s}{4} + \frac{\Omega_s}{10}\right)$ in the first stopband, and $\left(\frac{\Omega_s}{2} - \frac{\Omega_s}{10}\right)$ in the second stopband.

There is a “don’t care” band between $\left(\frac{\Omega_s}{4} + \frac{\Omega_s}{10}\right)$ and $\left(\frac{\Omega_s}{2} - \frac{\Omega_s}{10}\right)$. The order is 85.

In Figure 6, it can be observed that between 7000 Hz and 8000 Hz is located a “don’t care” band as expected.

Decimation

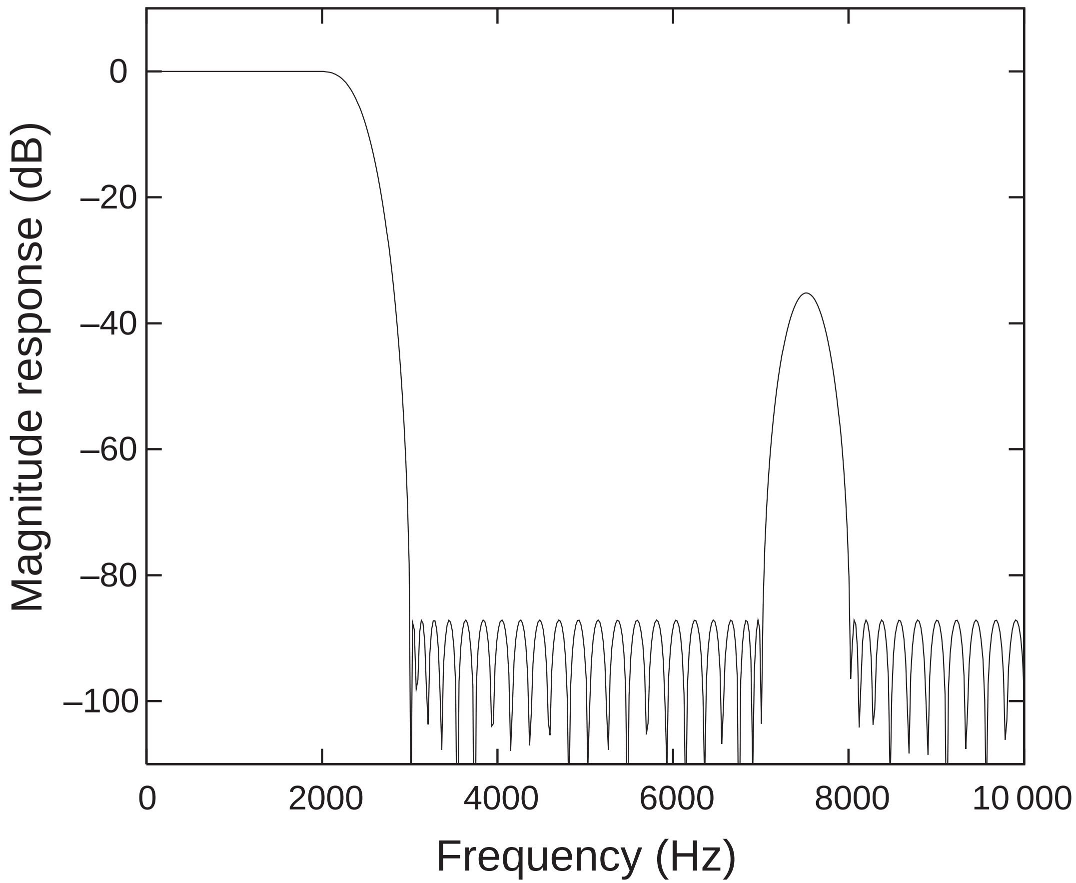


Figure 6: Magnitude response of the decimation filter for $M = 4$.

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 as

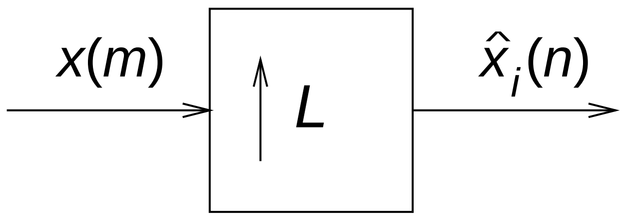


Figure 7: Interpolation by a factor of L.

- The interpolated signal is then given by

$$\hat{x}_i(n) = \begin{cases} x\left(\frac{n}{L}\right), & n = kL,\ k \in \mathbb{Z} \\ 0, & \text{otherwise} \end{cases} \tag{17}$$

Interpolation

- The interpolation operation for the case $L = 2$:
- 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, $\hat{X}_i(e^{j\omega})$, becomes

$$\hat{X}_i(e^{j\omega}) = X(e^{j\omega L}) \tag{18}$$

Interpolation

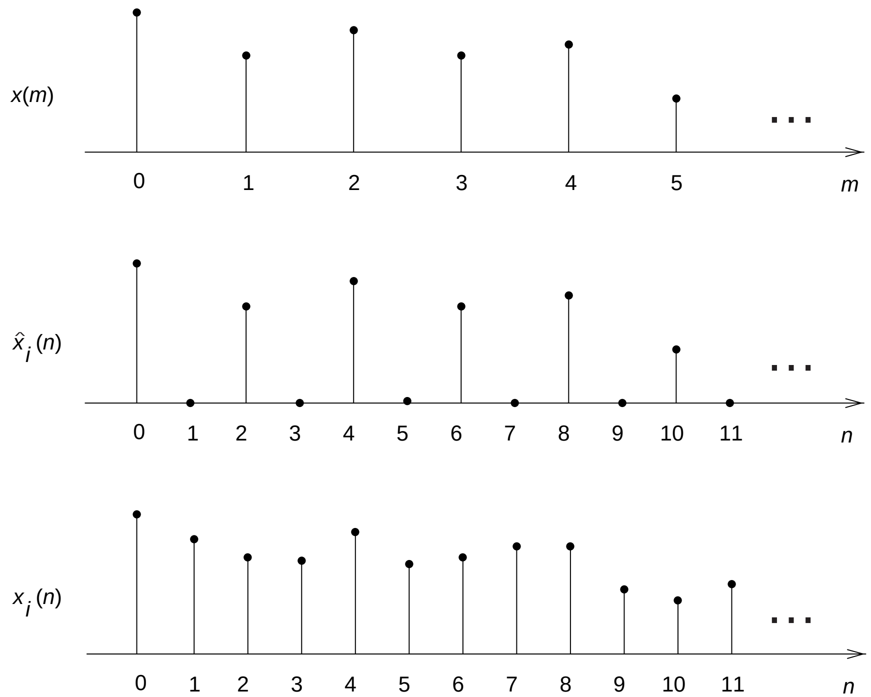


Figure 8: Interpolation by 2. $x(m)$: original signal; $\hat{x}_i(n)$: signal with zeros inserted between samples; $x_i(n)$: interpolated signal after filtering by $H_i(z)$. ($x(m) = \ldots x(0)\, x(1)\, x(2)\, x(3)\, x(4)\, x(5)\, x(6) \ldots$; $\hat{x}_i(n) = \ldots x(0)\, 0\, x(1)\, 0\, x(2)\, 0\, x(3) \ldots$; $x_i(n) = \ldots x_i(0)\, x_i(1)\, x_i(2)\, x_i(3)\, x_i(4)\, x_i(5)\, x_i(6) \ldots$)

Interpolation

Figure below shows the spectra of the signals $x(m)$ and $\hat{x}_i(n)$ for an interpolation factor of L.

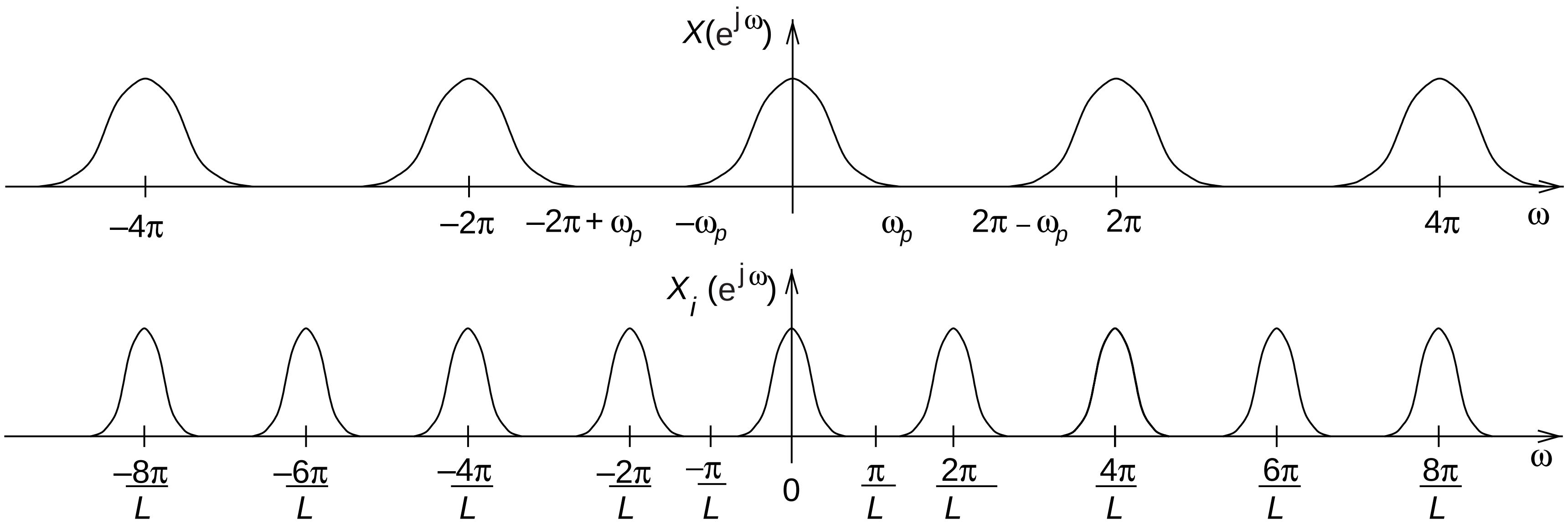


Figure 9: Signal spectra of: (a) original digital signal; (b) interpolated signal by a factor of L.

Interpolation

- Since the spectrum of the original digital signal is periodic with period 2π, the spectrum of the interpolated signal has period $\frac{2\pi}{L}$.
  - 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 $[-\frac{\pi}{L}, \frac{\pi}{L}]$.

    Thus, the interpolation operation is followed by a lowpass filter which approximates the following frequency response:

$$H_i(e^{j\omega}) = \begin{cases} L, & \omega \in [-\frac{\pi}{L}, \frac{\pi}{L}] \\ 0, & \text{otherwise} \end{cases} \tag{19}$$

Interpolation

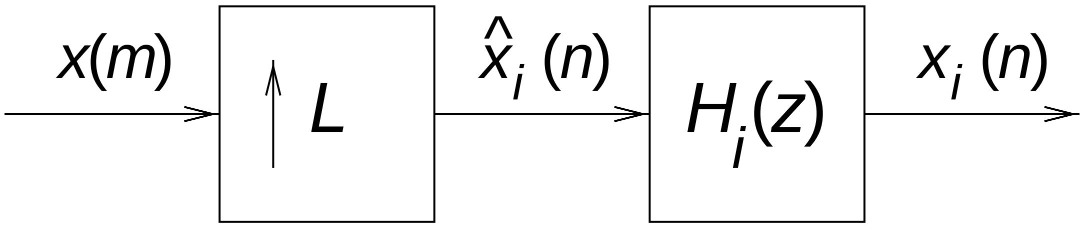


Figure 10: General interpolation operation.

- The interpolation operation is equivalent to the convolution of the interpolation filter impulse response, $h_i(n)$, with the signal $\hat{x}_i(n)$ whose only nonzero samples have a multiple of L index, that is

$$\hat{x}_i(kL) = \begin{cases} x(k), & k \in \mathbb{Z} \\ 0, & \text{otherwise} \end{cases} \tag{20}$$

- In the time domain, the filtered interpolated signal becomes

$$x_i(n) = \sum_{m=-\infty}^{\infty} \hat{x}_i(m)h(n-m) = \sum_{k=-\infty}^{\infty} x(k)h(n-kL) \tag{21}$$

Interpolation

Some important facts must be noted about the interpolation operation

- 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 (5) and (20) imply that $\mathcal{D}_L\{\mathcal{I}_L\{x(m)\}\} = x(m)$, that is, the interpolation operation is invertible. However, $\{\mathcal{I}_L\{x(m-k)\}\} = x_i(n-kL)$, which means that the interpolation is inherently time varying.

- Referring to equation (21), one can see that the computation of the output of the filter $H_i(z)$ 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.

Interpolation

- 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 $\frac{\omega_p}{L}$ around the frequencies $\frac{2\pi k}{L}$, $k = 1, 2, \ldots, L-1$.

  The constraints upon the filter can be relaxed as in the decimation case, yielding

$$H_i(e^{j\omega}) = \begin{cases} L, & |\omega| \in [0, \frac{\omega_p}{L}] \\ 0, & |\omega| \in [\frac{2\pi k - \omega_p}{L}, \frac{2\pi k + \omega_p}{L}], \quad k = 1, 2, \ldots, L-1 \end{cases} \tag{22}$$

The gain factor L in equations (19) and (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.

Examples of interpolators

Supposing $L = 2$, two common examples can be devised as shown in Figure 11:

- Zero-order hold: $x(2n+1) = x(2n)$. From equation (21), this is equivalent to having $h(0) = h(1) = 1$, that is, $H_i(z) = 1 + z^{-1}$.
- Linear interpolator: $x(2n+1) = \frac{1}{2}[x(2n) + x(2n+2)]$. From equation (21), this is equivalent to having $h(-1) = \frac{1}{2}$, $h(0) = 1$, and $h(1) = \frac{1}{2}$, that is, $H_i(z) = \frac{1}{2}(z + 2 + z^{-1})$.

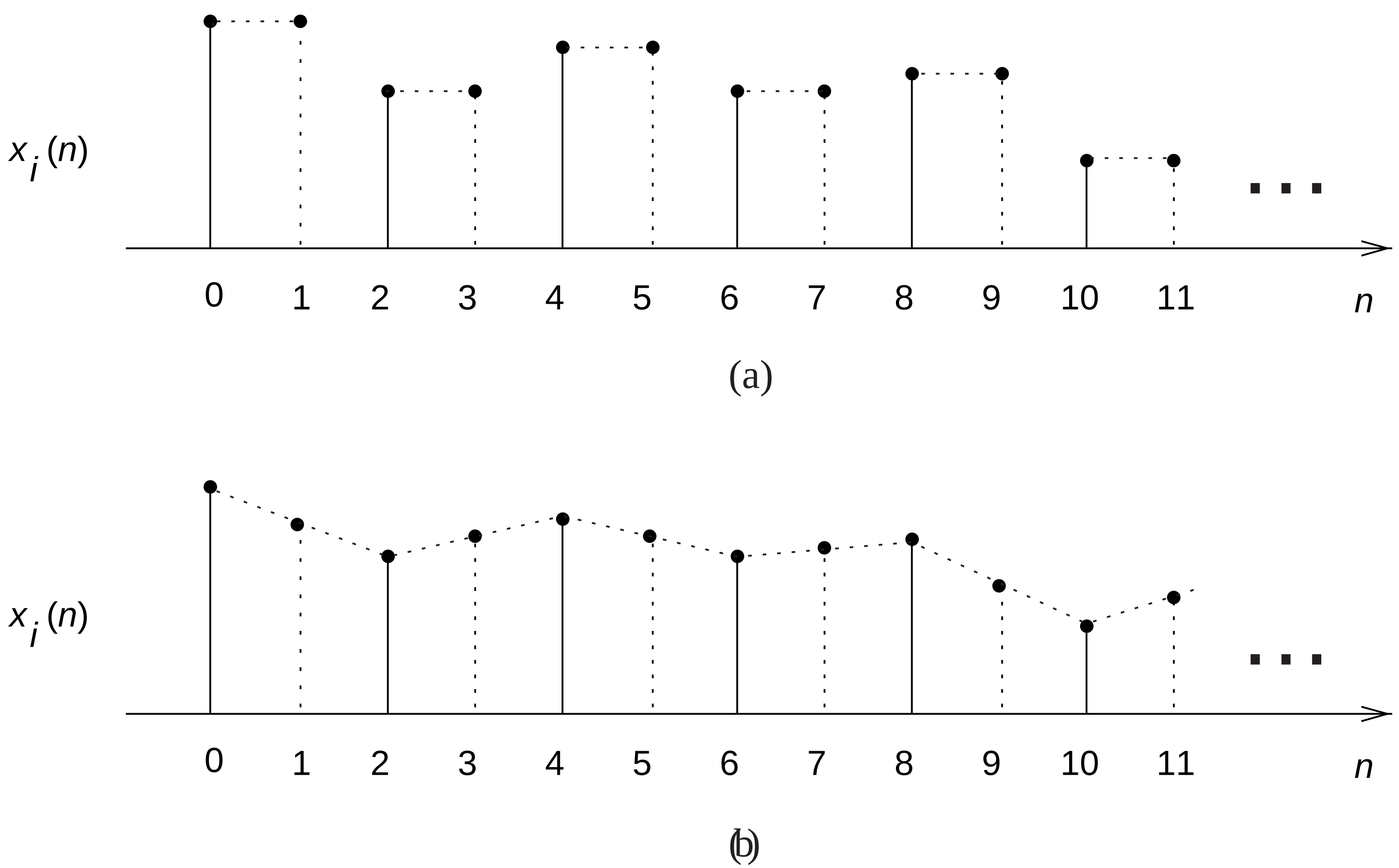


Figure 11: Examples of interpolators: (a) zero-order hold; (b) linear interpolator.

Examples of interpolators

- Lth-band filters are interesting examples of interpolators.
- When used as interpolators by L they keep the original samples of the signal to be interpolated.
- If one decimates by L the interpolated signal $x_i(n)$ generated with an Lth-band filter, one obtains the original signal $x(m)$.
- This is equivalent to saying that

$$x_i(mL) = x(m) \tag{23}$$

- In this case, equation (21) becomes

$$x_i(mL) = \sum_{k=-\infty}^{\infty} x(k)h(mL - kL) = x(m) \tag{24}$$

- This only happens if

$$h(mL - kL) = \begin{cases} 1, & m = k \\ 0, & m \neq k \end{cases} \tag{25}$$

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 2-band filters, commonly referred as, half-band filters.

Rational sampling-rate changes

- A rational sampling-rate change by a factor $\frac{L}{M}$ can be implemented by cascading an interpolator by a factor of L with a decimator by a factor of M, as represented below:

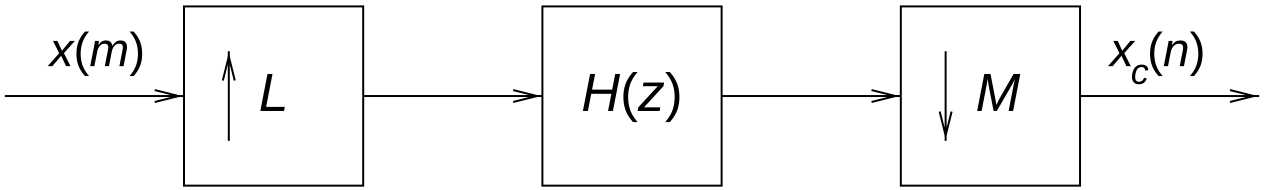


Figure 12: Sampling rate change by a factor of $\frac{L}{M}$.

$$H(z) \text{ is an interpolation filter} \quad \Rightarrow \quad \omega_c < \frac{\pi}{L}$$

$$H(z) \text{ is a decimation filter} \quad \Rightarrow \quad \omega_c < \frac{\pi}{M}$$

$\Rightarrow$ It must approximate the following frequency response:

$$H(e^{j\omega}) = \begin{cases} L, & |\omega| \leq \min\{\frac{\pi}{L}, \frac{\pi}{M}\} \\ 0, & \text{otherwise} \end{cases} \tag{26}$$

- 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 equation (22) and the specifications in equation (14) with ω_p replaced by $\frac{\omega_p}{L}$.

  Since L and M can be assumed, without loss of generality, to be relatively prime, this yields

$$H(e^{j\omega}) = \begin{cases} L, & |\omega| < \min\{\frac{\omega_p}{L}, \frac{\pi}{M}\} \\ 0, & \min\{\frac{2\pi}{L} - \frac{\omega_p}{L}, \frac{2\pi}{M} - \frac{\omega_p}{L}\} \leq |\omega| \leq \pi \end{cases} \quad (27)$$

Inverse operations

- 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.
- $\mathcal{I}_M\mathcal{D}_M$ is not the identity operator since their cascade is equivalent to replacing $(M-1)$ out of M samples of the signal with zeros.

Inverse operations

- However, if the decimation-by-M operation is preceded by a band-limiting filter for the interval $[-\frac{\pi}{M}, \frac{\pi}{M}]$ and the interpolation operation is followed by the same filter then $\mathcal{I}_M \mathcal{D}_M$ becomes the identity operation.

- This can be easily confirmed in the frequency domain
  - The band-limiting filter avoids aliasing after decimation, and which makes the decimation operation remain invertible.
  - After interpolation by M, there are images of the spectrum of the signal in the intervals $[\frac{\pi k}{M}, \frac{\pi(k+1)}{M}]$, $k = -M, -M+1, \ldots, M-1$.
  - The second band-limiting filter keeps only the image inside $[-\frac{\pi}{M}, \frac{\pi}{M}]$, which corresponds to the spectrum of the original signal.

Inverse operations

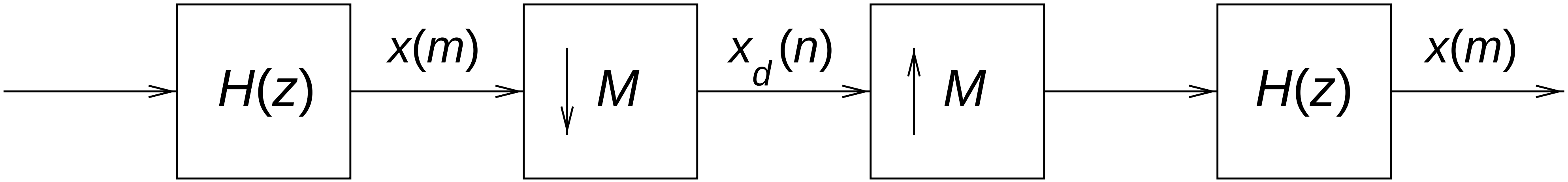


Figure 13: Decimation followed by interpolation.

Inverse operations

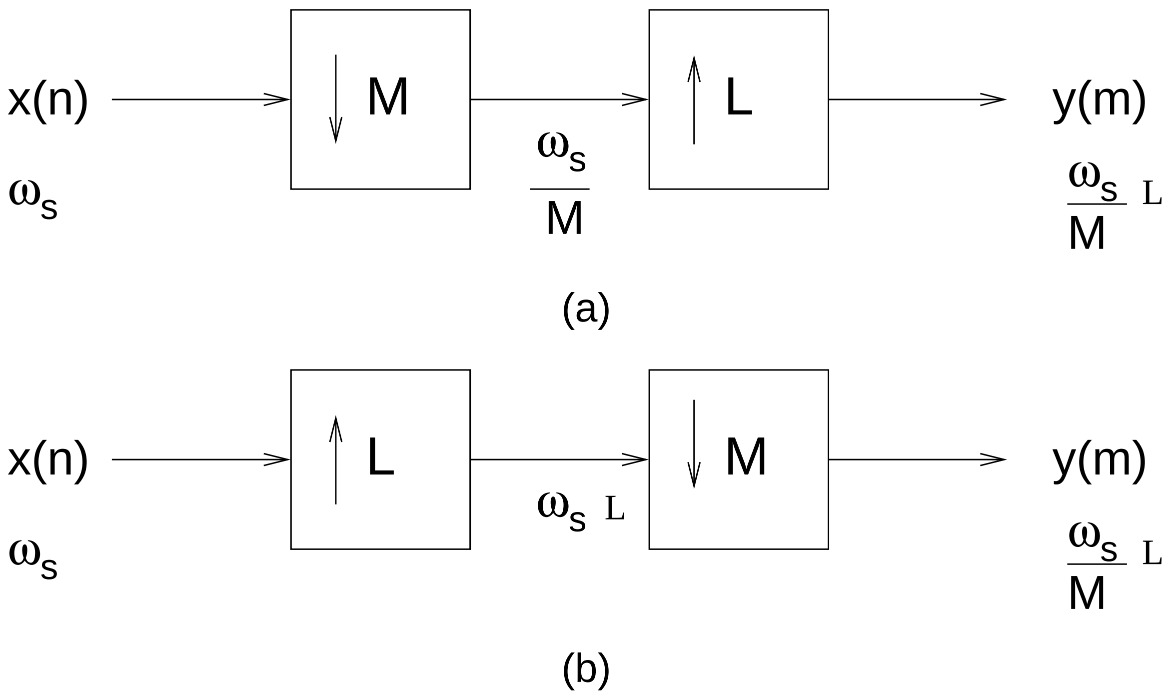


Figure 14: Cascade operations: (a) decimation/interpolation; (b) interpolation/decimation.

Inverse operations

- Under which conditions the decimation and interpolation operations are commutative?

  That is, when the connection $\mathcal{D}_M \mathcal{I}_L$ is equivalent to $\mathcal{I}_L \mathcal{D}_M$.

- We have already seen that when $M = L$ they are not equivalent.

- Usually, these interconnections are not equivalent, unless M and L are relatively prime numbers.

- For compression before expansion the output signal is given by

$$y(m) = \begin{cases} x\left(\frac{mM}{L}\right), & m = kL,\ k \in \mathbb{Z} \\ 0, & \text{otherwise} \end{cases} \tag{28}$$

- For expansion before compression the output signal is given by

$$y(m) = \begin{cases} x\left(\frac{mM}{L}\right), & mM = kL,\ k \in \mathbb{Z} \\ 0, & \text{otherwise} \end{cases} \tag{29}$$

- Note that the condition in equation (28), $m = kL$, $k \in \mathbb{Z}$, implies the condition in equation (29), that is $mM = kML = k'L$, $k' \in \mathbb{Z}$. On the other hand, the condition in equation (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.

Noble identities

- The noble identities are depicted in Figure 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.

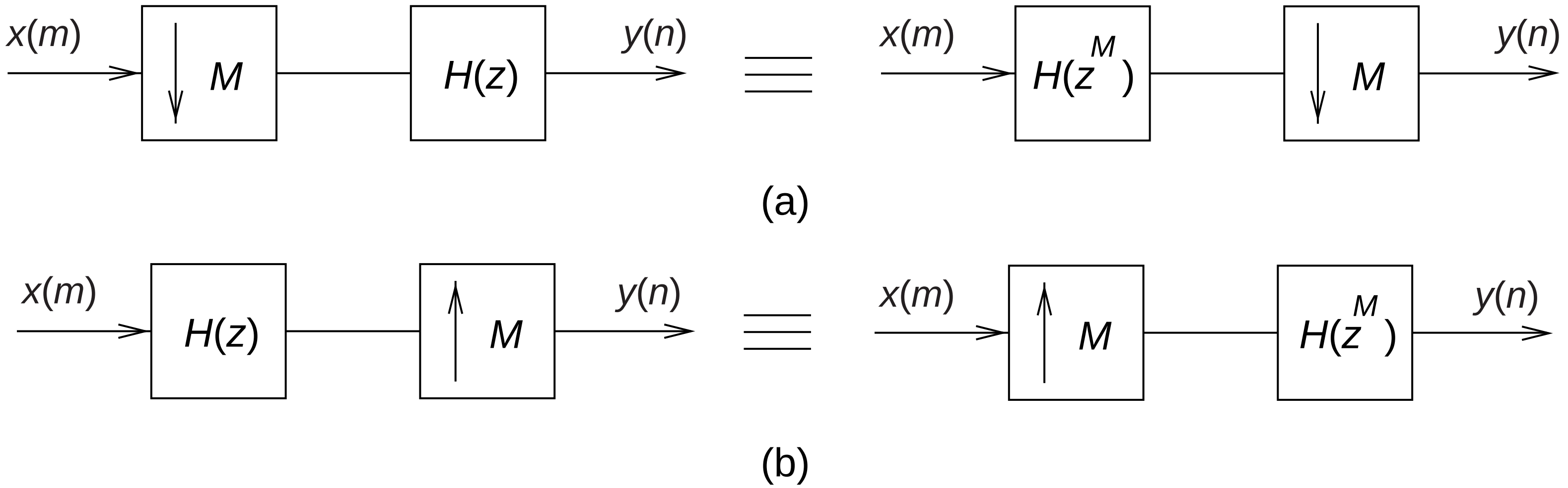


Figure 15: Noble identities: (a) decimation; (b) interpolation.

Noble identities

- The identity in Figure 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

$$\mathcal{D}_M\{X(z)\}H(z) = \mathcal{D}_M\{X(z)H(z^M)\} \tag{30}$$

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

Noble identities

- The identity in Figure 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, it is stated as

$$\mathcal{I}_M\{X(z)H(z)\} = \mathcal{I}_M\{X(z)\}H(z^M) \tag{31}$$

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

Noble identities

- In order to prove the identity in Figure 15a, one begins by rewriting equation (6), which gives the Fourier transform of the decimated signal $x_d(n)$ as a function of the input signal $x(m)$, in the z domain, that is

$$X_d(z) = \frac{1}{M}\sum_{k=0}^{M-1} X\left(z^{\frac{1}{M}} e^{-j\frac{2\pi k}{M}}\right) \tag{32}$$

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

$$Y(z) = H(z)X_d(z) = \frac{1}{M}H(z)\sum_{k=0}^{M-1} X\left(z^{\frac{1}{M}} e^{-j\frac{2\pi k}{M}}\right) \tag{33}$$

- For the filter $H(z^M)$ followed by the decimator, if $U(z) = X(z)H(z^M)$, we have, from equation (32), that

$$
\begin{aligned}
Y(z) =& \frac{1}{M}\sum_{k=0}^{M-1} U\left(z^{\frac{1}{M}}e^{-j\frac{2\pi k}{M}}\right) \\
=& \frac{1}{M}\sum_{k=0}^{M-1} X\left(z^{\frac{1}{M}}e^{-j\frac{2\pi k}{M}}\right) H\left(ze^{-j\frac{2\pi M k}{M}}\right) \\
=& \frac{1}{M}\sum_{k=0}^{M-1} X\left(z^{\frac{1}{M}}e^{-j\frac{2\pi k}{M}}\right) H(z) \qquad (34)
\end{aligned}
$$

which is the same as equation (33), and the identity is proved.

- Proof of the identity in Figure 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 $H(z^M)$.

- The z transform $H(z)$ of a filter $h(n)$ can be written as

$$
\begin{aligned}
H(z) &= \sum_{k=-\infty}^{+\infty} h(k) z^{-k} \\
&= \sum_{l=-\infty}^{+\infty} h(Ml) z^{-Ml} + \sum_{l=-\infty}^{+\infty} h(Ml+1) z^{-(Ml+1)} + \cdots \\
&\quad + \sum_{l=-\infty}^{+\infty} h(Ml+M-1) z^{-(Ml+M-1)} \\
&= \sum_{l=-\infty}^{+\infty} h(Ml) z^{-Ml} + z^{-1} \sum_{l=-\infty}^{+\infty} h(Ml+1) z^{-Ml} + \cdots \\
&\quad + z^{-M+1} \sum_{l=-\infty}^{+\infty} h(Ml+M-1) z^{-Ml} \\
&= \sum_{j=0}^{M-1} z^{-j} E_j(z^M) \qquad (35)
\end{aligned}
$$

Polyphase decompositions

- Equation (35) represents the polyphase decomposition of the filter $H(z)$, and

$$E_j(z) = \sum_{l=-\infty}^{+\infty} h(Ml + j)z^{-l} \tag{36}$$

are called the polyphase components of $H(z)$.

- In such a decomposition, the filter $H(z)$ is split 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 1 plus a multiple of M, and so on.

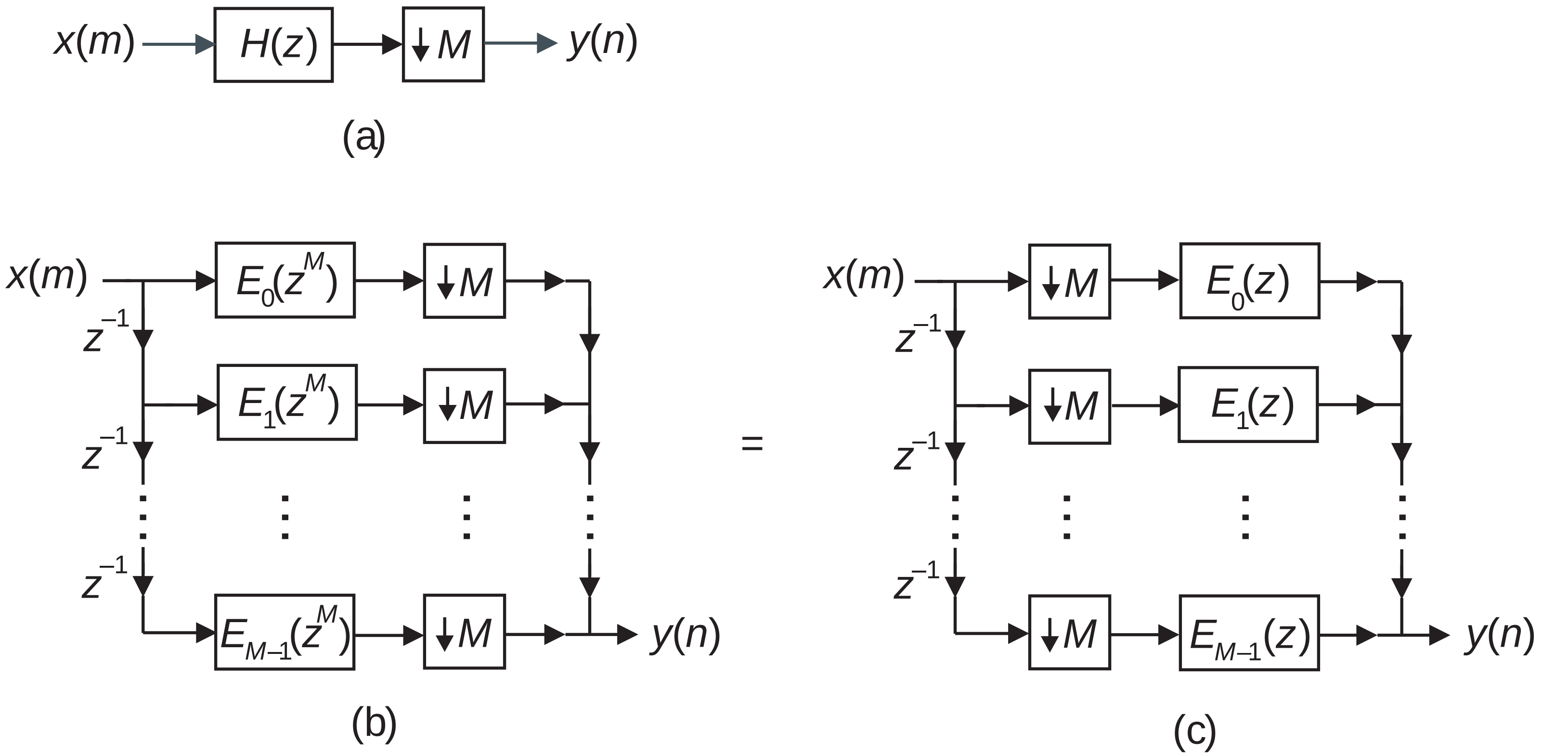
Polyphase decompositions

- Let us now analyze the basic operation of filtering followed by decimation represented in Figure 16a.

- Using the polyphase decomposition, such processing can be visualized as in Figure 16b, and applying the noble identity in equation (30), we arrive at Figure 16c, which provides an interesting and useful interpretation of the operation represented in Figure 16a.

- In fact, Figure 16c shows that the whole 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 of only the samples of $h(m)$ whose indexes are equal to the same integer k plus a multiple of M, for $k = 0, 1, \ldots, M - 1$.

Polyphase decompositions



•

Figure 16: Decimation representations: (a) decimation by a factor of M; (b) decimation using polyphase decompositions; (c) decimation using polyphase decompositions and the noble identities.

Polyphase decompositions

- The polyphase decompositions also provide useful insights into the interpolation operation followed by filtering.
- In this case, a variation of equation (35) is usually employed.
- Defining $R_j(z) = E_{M-1-j}(z)$, the polyphase decomposition becomes

$$H(z) = \sum_{j=0}^{M-1} z^{-(M-1-j)} R_j(z^M) \tag{37}$$

Applying the noble identity.

Polyphase decompositions

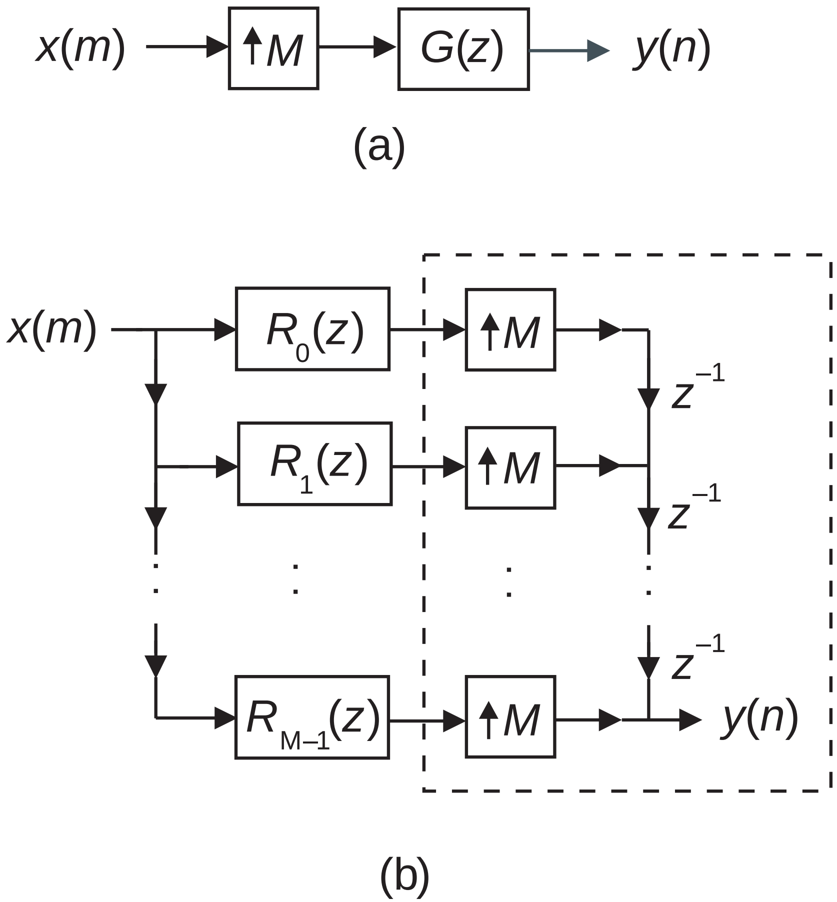


Figure 17: Interpolation representations: (a) interpolation by a factor of M; (b) interpolation using polyphase decompositions and the noble identities.

Commutator models

- The operations at the input and output of Figures 16c and 17b can also be interpreted in terms of rotary switches.
- The decimators and delays are replaced by rotary switches as depicted in Figure 18.

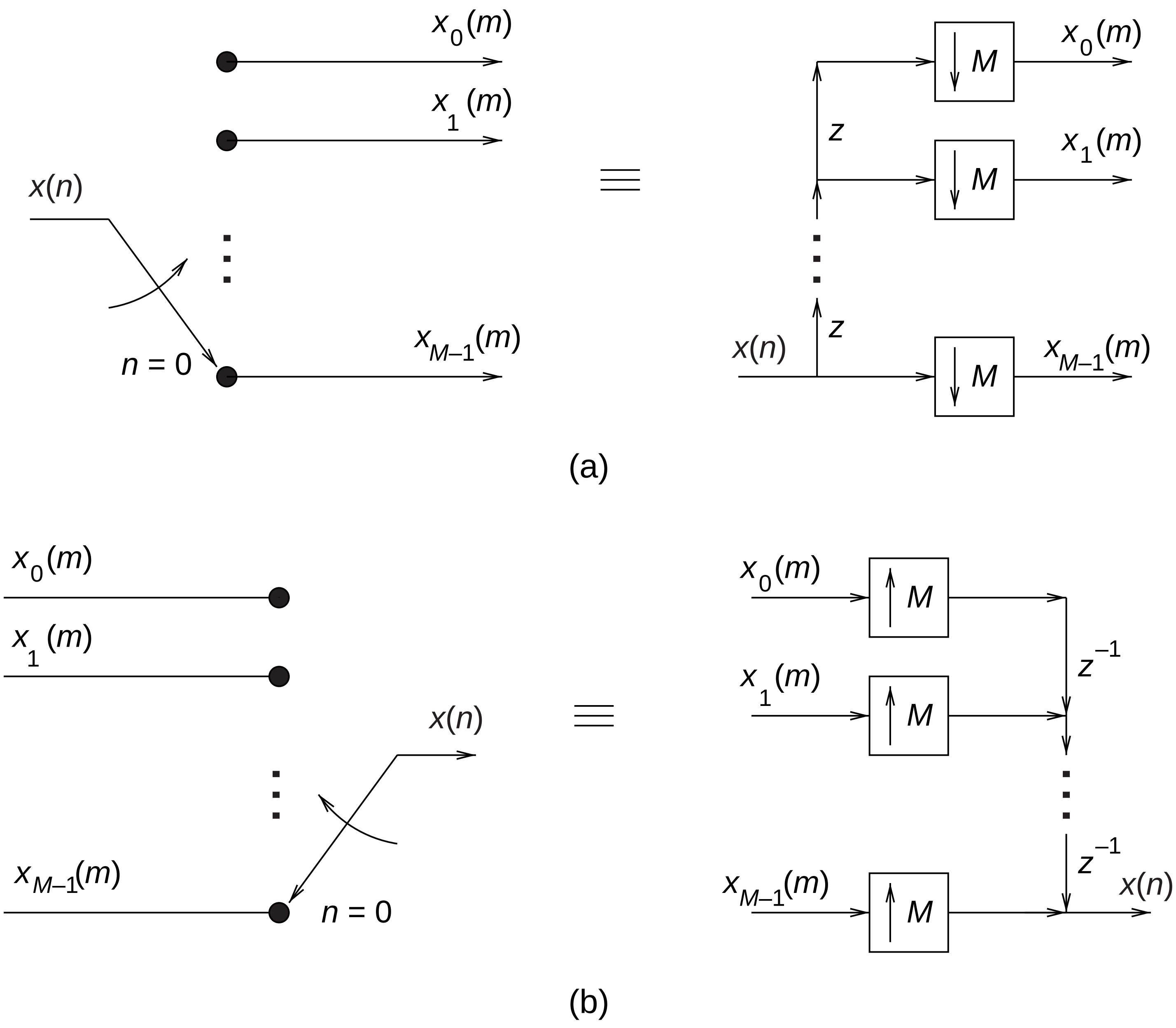


Figure 18: Commutator models for: (a) decimation; (b) interpolation.

- The previous model with decimators and delays is noncausal, having "advances" instead of delays.
- In real-time the causal model of Figure 19 is usually preferred.

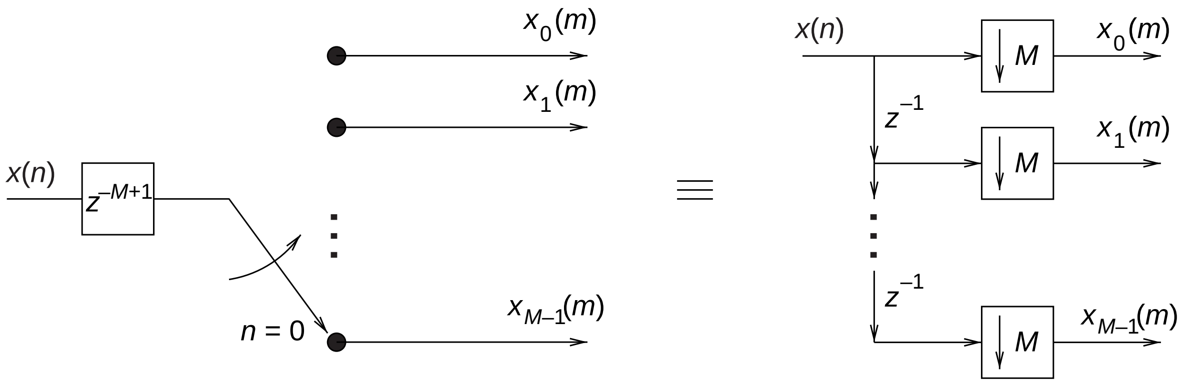


Figure 19: Causal commutator model for decimation.

Commutator models

- The operation depicted in Figure 19 is usually referred to as a serial-to-parallel converter.

$$\mathbf{x}(m) = \begin{bmatrix} x(mM) & x(mM-1) & \ldots & x(mM-M+1) \end{bmatrix}^{\mathsf{T}} \tag{38}$$

- Its inverse operation if the one depicted in Figure 18b. It is usually referred to as a parallel-to-serial converter.

- Note that each $\mathbf{x}(m)$ is a block of M consecutive samples of $x(n)$. According to equation (38), these blocks do not overlap, since there is no common sample between $\mathbf{x}(m)$ and $\mathbf{x}(m-1)$. In addition, the last sample of $\mathbf{x}(m)$ is consecutive to the first sample of $\mathbf{x}(m-1)$. This implies that indeed equation (38) represents the splitting of $x(n)$ in non-overlapping blocks of length M. Likewise, the inverse operation in Figure 18b is equivalent to putting the blocks side-by-side, recovering the signal $x(n)$.

Commutator models

- If we generalize equation (38) to

$$\mathbf{x}_L^M(m) = \begin{bmatrix} x(mM) & x(mM-1) & \ldots & x(mM-L+1) \end{bmatrix}^{\mathrm{T}} \tag{39}$$

where $L > M$, then we have that there is an overlap between the samples of $\mathbf{x}_L^M(m)$ and $\mathbf{x}_L^M(m-1)$. The last $(L-M)$ samples of $\mathbf{x}_L^M(m)$ are the same as the first $(L-M)$ samples of $\mathbf{x}_L^M(m-1)$. That is, we divide the signal $x(n)$ into overlapping blocks. Note that in this overlapping blocks case, the right-hand side of Figure 19a becomes Figure 20a.

Commutator models

- If we define a unit delay operator $\mathcal{D}\{\cdot\}$ applied to a block $\mathbf{x}_L^M(m)$ as

$$\mathcal{D}\{\mathbf{x}_L^M(m)\} = \begin{bmatrix} x(mM-1) & x(mM-2) & \ldots & x(mM-L) \end{bmatrix}^{\mathrm{T}} \tag{40}$$

This delay operation displaces the start of the block by one sample, and thus $\mathcal{D}\{\mathbf{x}_L^M(m)\} \neq \mathbf{x}_L^M(m-1)$. In fact,

$$\mathcal{D}^M\{\mathbf{x}_L^M(m)\} = \mathbf{x}_L^M(m-1) \tag{41}$$

- We can then map a non-overlapping block division to an overlapping block division for $M < L < 2M$ as follows:

$$\mathbf{x}_L^M(m) = \begin{bmatrix} \mathbf{I}_M & \\ \mathcal{D}^M\mathbf{I}_{L-M} & 0 \end{bmatrix} \mathbf{x}_M^M(m) = \begin{bmatrix} \mathbf{I}_{L-M} & 0 \\ 0 & \mathbf{I}_{2M-L} \\ \mathcal{D}^M\mathbf{I}_{L-M} & 0 \end{bmatrix} \mathbf{x}_M^M(m) \tag{42}$$

where $\mathbf{I}_M$ is the $M \times M$ identity matrix.

Commutator models

- If we consider the original signal $x(n)$ in vector form as the concatenation of the non-overlapping blocks $\mathbf{x}_M^M(m)$

$$\mathbf{x} = \left[\ldots \quad \mathbf{x}_M^{M^{\mathsf{T}}}(m+1) \quad \mathbf{x}_M^{M^{\mathsf{T}}}(m) \quad \mathbf{x}_M^{M^{\mathsf{T}}}(m-1) \quad \ldots\right]^{\mathsf{T}} \tag{43}$$

Commutator models

- We can express the serial-to-parallel conversion in the overlapped case as

$$\mathbf{x}_L^M(m) = \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \cdots \\ \cdots & 0 & 0 & 0 & 0 & \cdots \\ \cdots & 0 & \mathbf{I}_{L-M} & 0 & 0 & \cdots \\ \cdots & 0 & 0 & \mathbf{I}_{2M-L} & 0 & \cdots \\ \cdots & 0 & \mathcal{D}^M \mathbf{I}_{L-M} & 0 & 0 & \cdots \\ \cdots & 0 & 0 & 0 & 0 & \cdots \\ \cdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \underbrace{\begin{bmatrix} \vdots \\ \mathbf{x}_M^M(m+1) \\ \\ \mathbf{x}_M^M(m) \\ \\ \mathbf{x}_M^M(m-1) \\ \vdots \end{bmatrix}}_{\mathbf{x}} \tag{44}$$

Commutator models

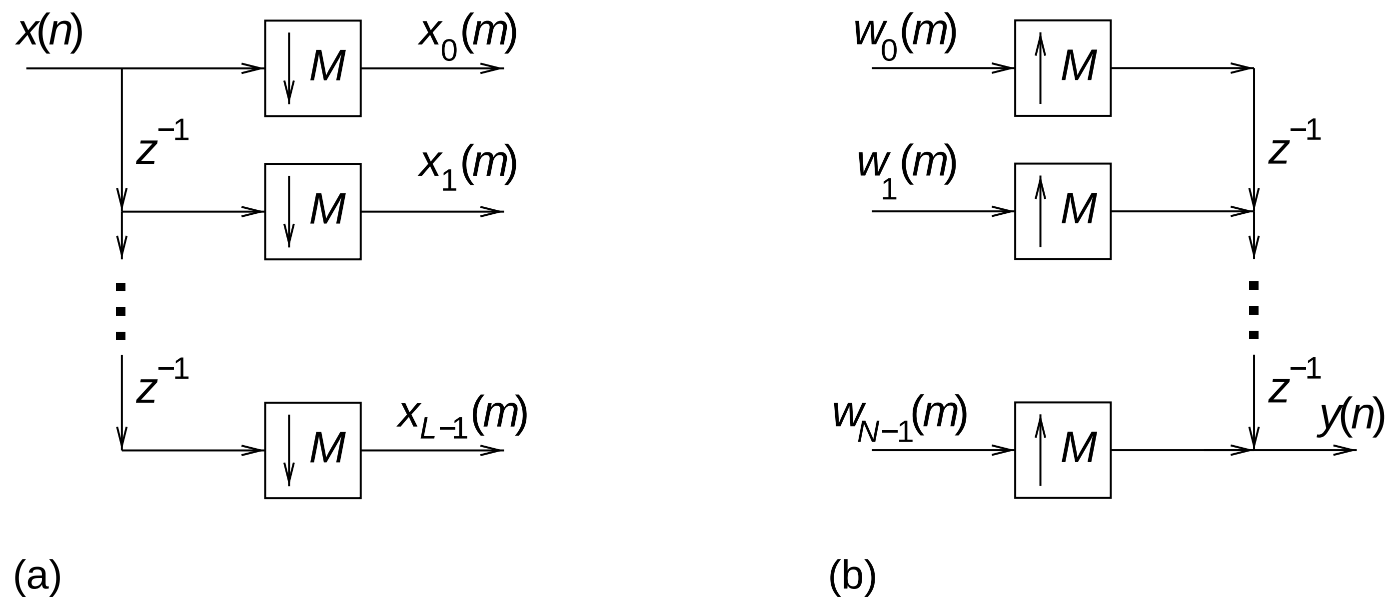


Figure 20: Commutator models for: (a) division into overlapping blocks ($L > M$); (b) generating a signal by summation of overlapping blocks ($N > M$).

Commutator models

- Likewise, if we increase the number of branches in Figure 18b to $N > M$, we get Figure 20b.
- The last $(N - M)$ samples of $\mathbf{x}_N^M(m)$ are added to the first $(N - M)$ samples of $\mathbf{x}_N^M(m-1)$. More precisely, this operation is equivalent to generating a block $\mathbf{y}_M^M(m)$ from $\mathbf{w}_N^M(m)$ as

$$
\begin{aligned}
\mathbf{y}_M^M(m) &= \begin{bmatrix} \mathbf{0} & \\ & \mathbf{I}_M \\ \mathcal{D}^M \mathbf{I}_{N-M} & \end{bmatrix} \mathbf{w}_N^M(m) \\
&= \begin{bmatrix} \mathbf{0} & \mathbf{I}_{2M-N} & \mathbf{0} \\ \mathcal{D}^M \mathbf{I}_{N-M} & \mathbf{0} & \mathbf{I}_{N-M} \end{bmatrix} \mathbf{w}_N^M(m) \qquad (45)
\end{aligned}
$$

Commutator models

- If we consider the output signal $y(n)$ in vector form as the concatenation of the non-overlapping blocks $\mathbf{y}_M^M(m)$,

$$\mathbf{y} = \left[\ldots \quad \mathbf{y}_M^{M^{\mathrm{T}}}(m+1) \quad \mathbf{y}_M^{M^{\mathrm{T}}}(m) \quad \mathbf{y}_M^{M^{\mathrm{T}}}(m-1) \quad \ldots\right]^{\mathrm{T}} \tag{46}$$

- Substituting equation (45) into equation (46) we can express the parallel to serial conversion in the overlapped case as

$$\mathbf{y}=\begin{bmatrix} \vdots \\ \mathbf{y}_M^M(m+1) \\ \mathbf{y}_M^M(m) \\ \mathbf{y}_M^M(m-1) \\ \vdots \end{bmatrix}$$

$$= \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots \\ \cdots & 0 & \mathbf{I}_{2M-N} & 0 & 0 & 0 & 0 & 0 & \cdots \\ \cdots & \mathcal{D}^M \mathbf{I}_{N-M} & 0 & \mathbf{I}_{N-M} & 0 & 0 & 0 & 0 & \cdots \\ \cdots & 0 & 0 & 0 & \mathbf{I}_{2M-N} & 0 & 0 & 0 & \cdots \\ \cdots & 0 & 0 & \mathcal{D}^M \mathbf{I}_{N-M} & 0 & \mathbf{I}_{N-M} & 0 & 0 & \cdots \\ \cdots & 0 & 0 & 0 & 0 & 0 & \mathbf{I}_{2M-N} & 0 & \cdots \\ \cdots & 0 & 0 & 0 & 0 & \mathcal{D}^M \mathbf{I}_{N-M} & 0 & \mathbf{I}_{N-M} & \cdots \\ \cdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} \vdots \\ \mathbf{w}_N^M(m+1) \\ \mathbf{w}_N^M(m) \\ \mathbf{w}_N^M(m-1) \\ \vdots \end{bmatrix}$$

(47

Decimation and interpolation for efficient filter implementation

Narrowband FIR filters

- Consider the system in Figure 21, consisting of the cascade of a decimator and an interpolator by M.

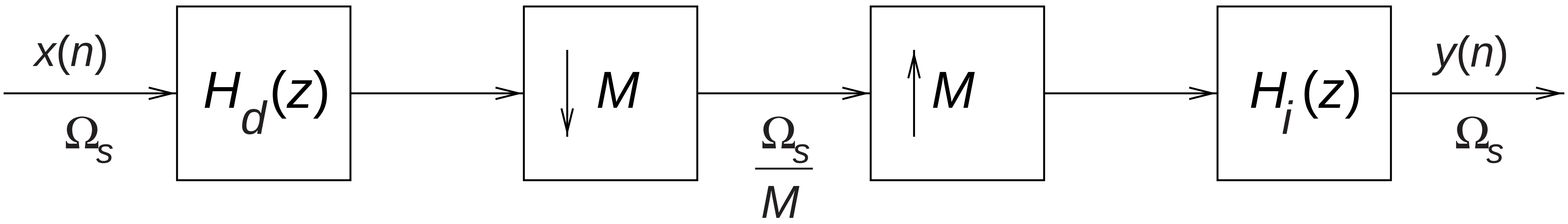


Figure 21: Filter using decimation/interpolation.

- From equations (6) and (18), one can easily infer the relation between the Fourier transforms of $y(n)$ and $x(n)$, which is

$$Y(e^{j\omega}) = \frac{H_i(e^{j\omega})}{M} \left\{ \sum_{k=0}^{M-1} \left[X(e^{j(\omega - 2\pi\frac{k}{M})}) H_d(e^{j(\omega - 2\pi\frac{k}{M})}) \right] \right\} \quad (48)$$

Decimation and interpolation for efficient filter implementation

- Supposing that both the decimation filter, $H_d(z)$, and the interpolation filter, $H_i(z)$, have been properly designed, the spectrum repetitions in the above equation are canceled, yielding the following relation:

$$\frac{Y(e^{j\omega})}{X(e^{j\omega})} = \frac{H_d(e^{j\omega})H_i(e^{j\omega})}{M} = H(e^{j\omega}) \tag{49}$$

- This result shows that the cascading of the decimation and interpolation operations of the same order M is equivalent to just cascading the decimation and interpolation filters, provided that both bandwidths are smaller than $\frac{\pi}{M}$.

Decimation and interpolation for efficient filter implementation

- The advantage is that in the implementation of the decimation operation, there is a reduction by M in the number of multiplications, and the same is true for the interpolation operation.
- The multiplication reduction increases with the value of M such that the bandwidth of the desired filter remains smaller than $\frac{\pi}{M}$.
- If we wish to design a filter with passband ripple δ_p and stopband ripple δ_r, it is enough to design interpolation and decimation filters, each having passband ripple $\frac{\delta_p}{2}$ and stopband ripple δ_r.

Decimation and interpolation for efficient filter implementation

Example 8.2

Using the concepts of decimation and interpolation, design a lowpass filter satisfying the following specifications:

$$\left.\begin{array}{l}\delta_p = 0.001\\ \delta_r = 1\times10^{-3}\\ \Omega_p = 0.025\Omega_s\\ \Omega_r = 0.045\Omega_s\\ \Omega_s = 2\pi\ \text{rad/s}\end{array}\right\} \quad (50)$$

Solution

- With the given set of specifications, the maximum possible value of M is 11. Using the Chebyshev (minimax) method, $H_d(z)$ and $H_i(z)$ can be made identical, and they must be at least of order 177 each.

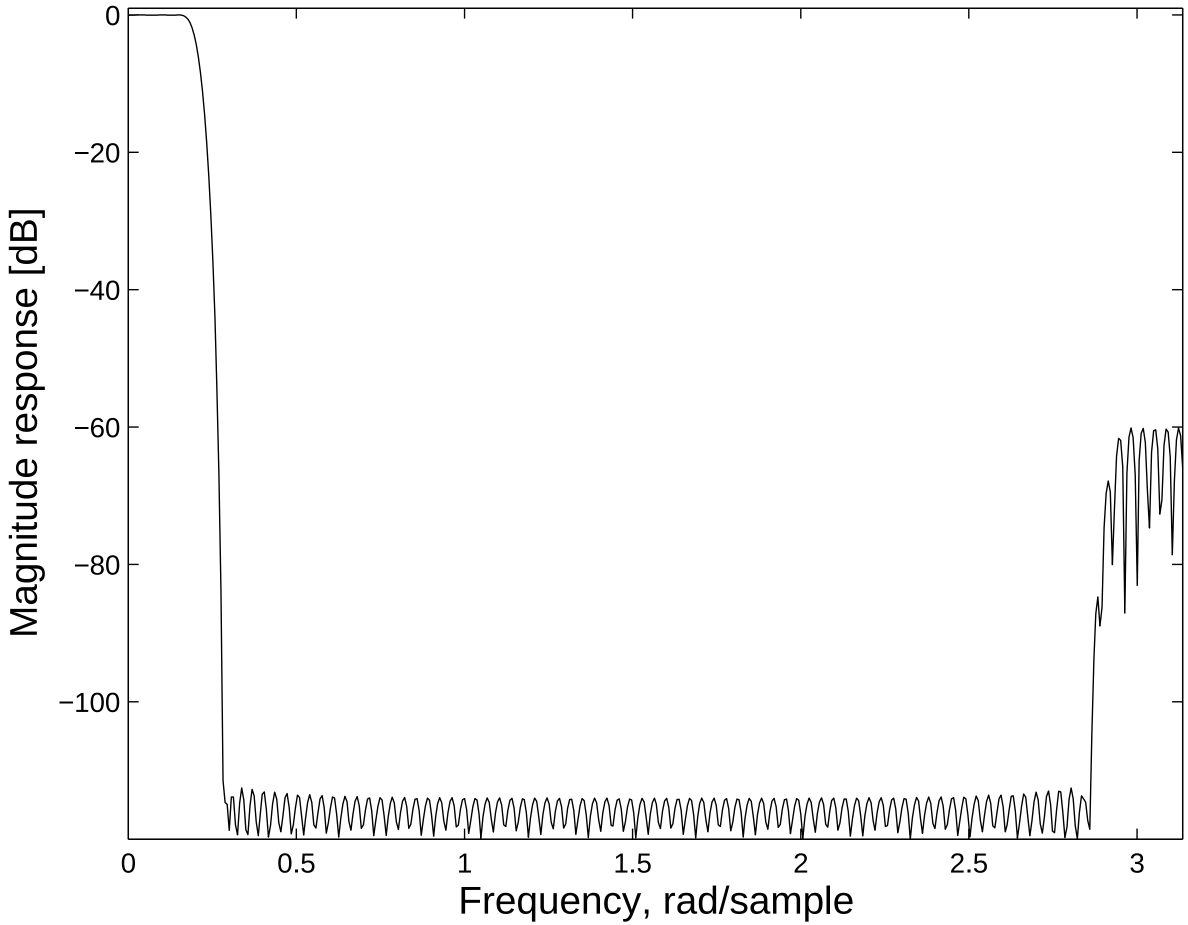


Figure 22: Magnitude response of filter using decimation/interpolation.

Decimation and interpolation for efficient filter implementation

- With the conventional approach, the total number of multiplications per sample would be 87 (as a linear-phase filter of order 173 would be required).
- Using decimation and interpolation, the total number of multiplications per output sample is only $178/11$ ($89/11$ for the decimation and $89/11$ for the interpolation).
- Greater reductions in complexity can be achieved if the decimators and interpolators in Figure 21 are composed of several decimation stages followed by several interpolation stages.
- The procedure can also be used to design narrowband bandpass filters. All we have to do is to choose M such that the desired filter passband and transition bands are contained in an interval of the form $[\frac{i\pi}{M}, \frac{(i+1)\pi}{M}]$, for only one value of i. In such cases, the interpolation and decimation filters are bandpass.
- Highpass and bandstop filters can be implemented based on lowpass and bandpass designs.

Wideband FIR filters with narrow transition bands

- Another interesting application of interpolation is in the design of sharp cutoff filters with low computational complexity using the frequency response masking approach.
- It takes into account that an interpolated filter has a transition band L times smaller than the prototype filter. The complete process is
- An interpolation ratio of $\mathrm{L} = 4$ is exemplified as follows.

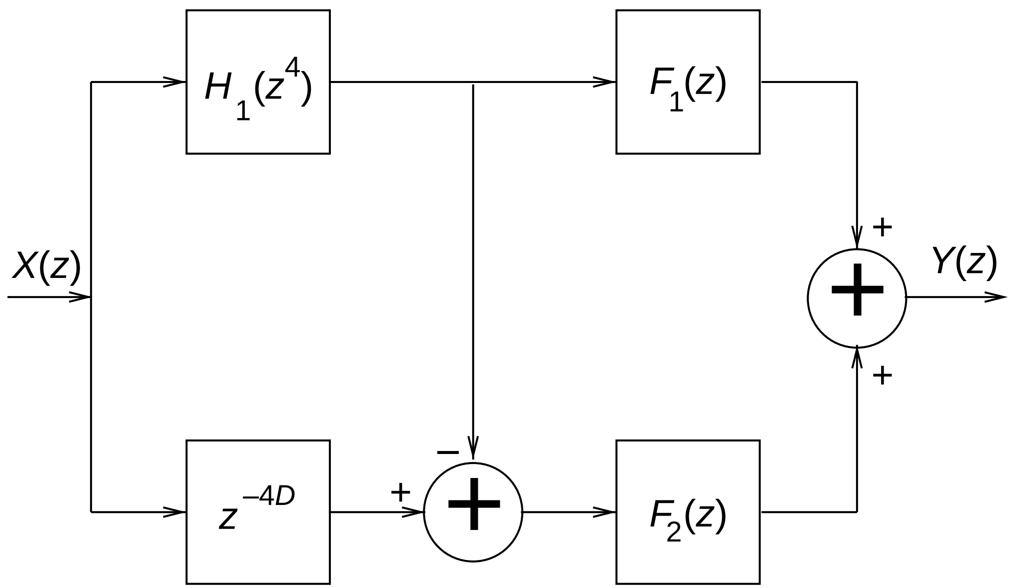


Figure 23: Filter design with the frequency response masking approach using interpolation.

Wideband FIR filters with narrow transition bands

- Suppose, for instance, that we want to design a normalized-lowpass filter having $\omega_p = 5\pi/8$ and $\omega_r = 11\pi/16$, such that the transition width is $\pi/16$.
- Using the frequency response masking approach, we design such a filter starting from a prototype half-band lowpass filter having $\omega_p = \pi/2$ and a transition bandwidth four times larger than the one needed, in this case, $\pi/4$.
- The implementation complexity of this prototype filter is much smaller than the original one.
- From this prototype, the complementary filter $H_2(z)$ is generated by a simple delay and subtraction as

$$H_2(z) = z^{-D} - H_1(z) \tag{51}$$

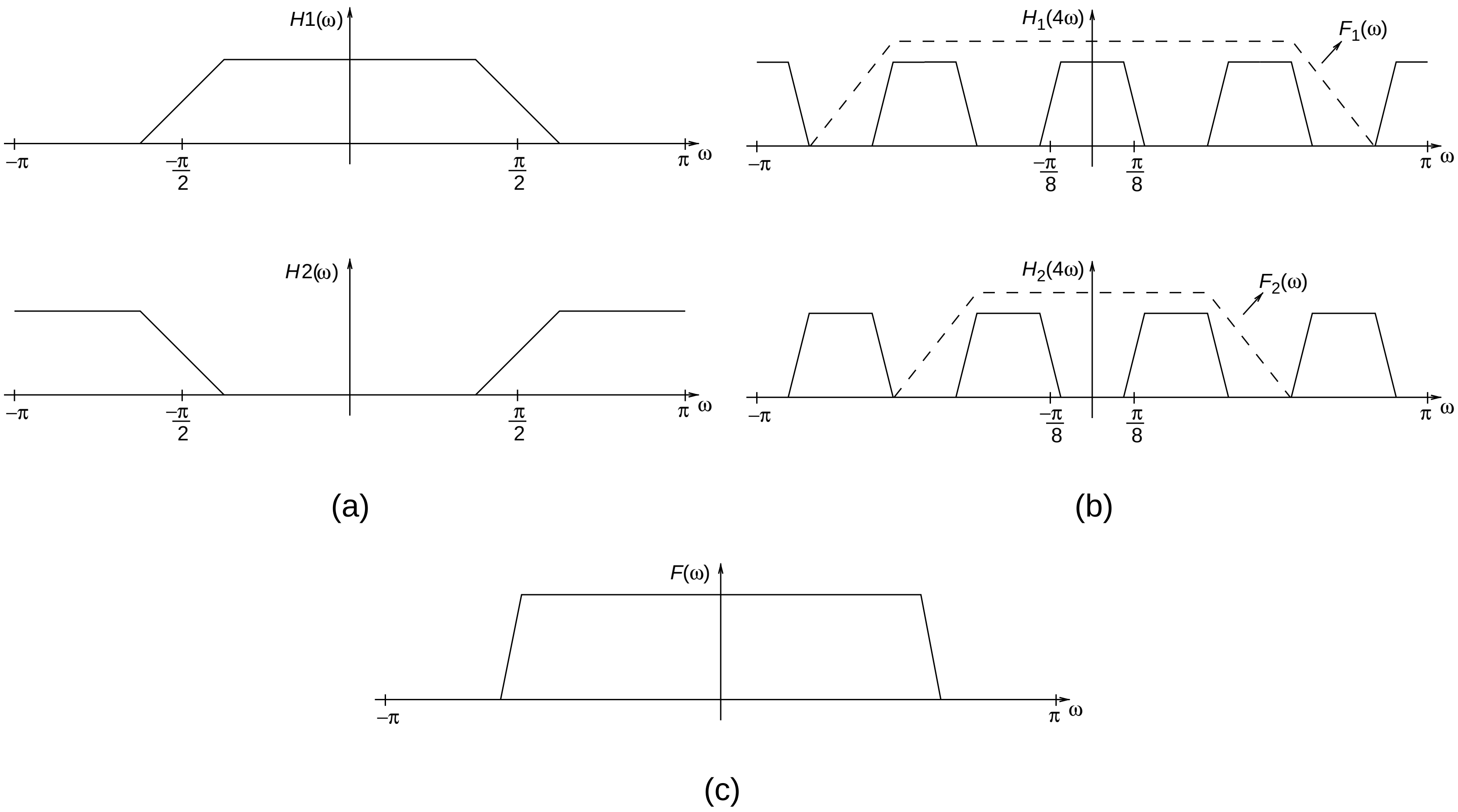


Figure 24: (a) Prototype half-band filter $H_1(z)$ and its complementary filter $H_2(z)$; (b) frequency responses of $H_1(z)$ and $H_2(z)$ after interpolation by a factor of $L = 4$; (c) frequency response of the equivalent filter $F(z)$.

Overlapped block filtering

- Consider the system represented in Figure 25.

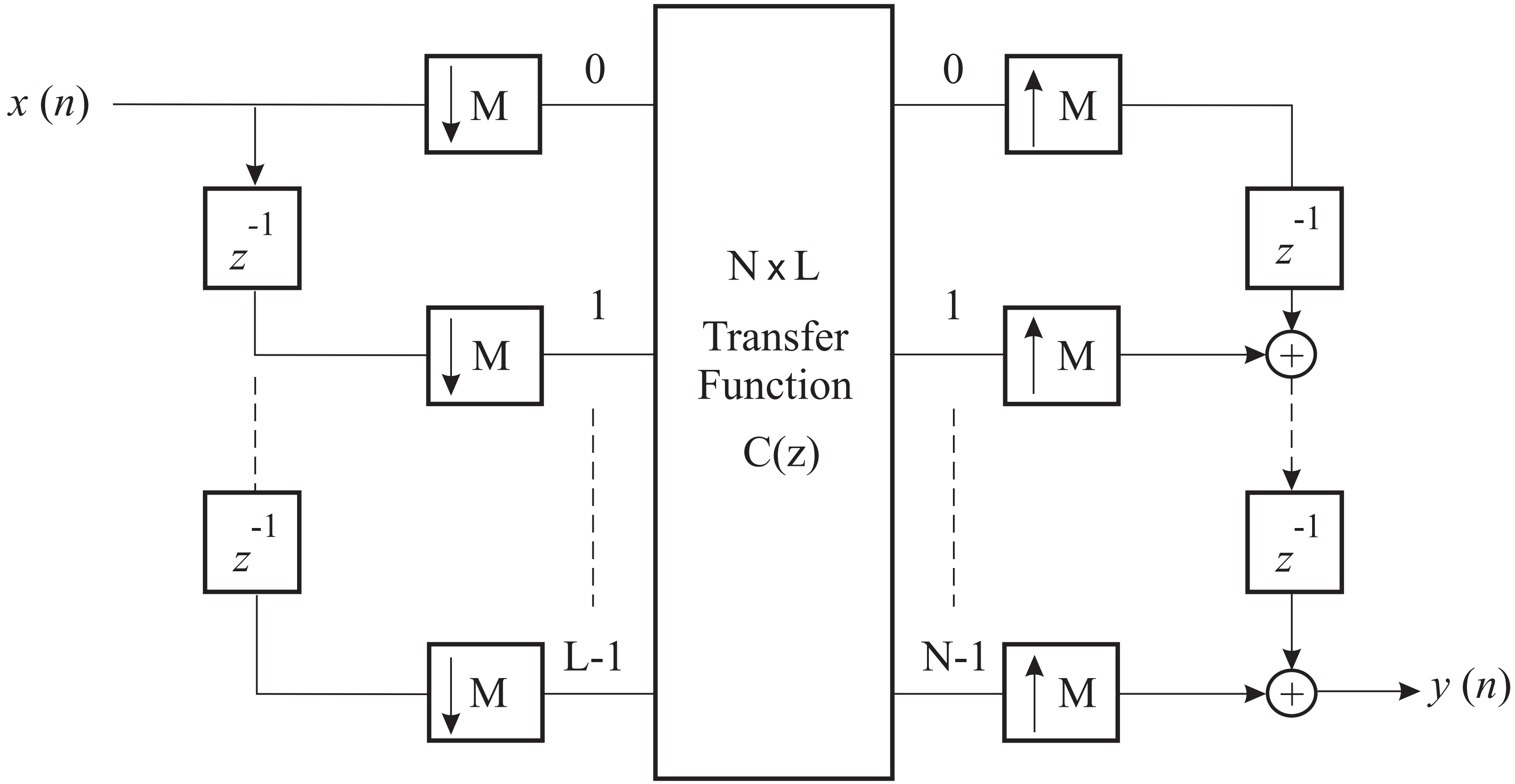


Figure 25: Multirate representation of an overlapped block digital filter.

Overlapped block filtering

- The input signal is divided into blocks of length L that have an overlap of $(L - M)$ samples. After processing, each length-L block is mapped into a length-N block.
- The output signal is generated by summing these blocks with and overlap of $(N - M)$ samples.
- The $N \times L$ matrix $\mathbf{C}(z)$ represents the linear mapping of an input block of length L to an output block of length N.
- Its element $C_{ij}(z)$ describes a time-invariant linear filtering operation performed on the sequence of elements j of each input block in order to generate the sequence of elements i of each output block.
- A widely used example of overlapped block filtering is the overlap-and-add method for which the decimation factor is N and the lengths of the input and output blocks are both equal to $(N + L - 1)$.

Overlapped block filtering

- The processing carried out by $\mathbf{C}(z)$ is the circular convolution with $h(n)$.
- Due to decimation-by-M operation, aliasing may occur in the process. In this case, one may not be able to describe the relation between the input and output as a linear filtering operation.
- Depending on the relative values of decimation factor M, overlap factor at the input $(L - M)$, and overlap factor at the output $(N - M)$, matrix $\mathbf{C}(z)$ has to satisfy different conditions in order to guarantee that the input-output relation is aliasing-free.

Non-overlapped case

- In the non-overlapped case the blocks do not overlap neither at the input nor at the reconstruction stage at the output. This happens when $L = M = N$.

- Our aim is to use the scheme in to implement a shift-invariant system. The input-output relation of such a system can be expressed as in the z transform domain as

$$Y(z) = \frac{1}{M} \begin{bmatrix} z^{-(N-1)} & \dots & z^{-1} & 1 \end{bmatrix} \mathbf{C}(z^M) \sum_{i=0}^{M-1} \begin{bmatrix} 1 \\ (zW_M)^{-1} \\ \vdots \\ (zW_M)^{-(L-1)} \end{bmatrix} X(zW_M^i) \tag{52}$$

Non-overlapped case

- In the non-overlapped case the blocks do not overlap neither at the input nor at the reconstruction stage at the output. This happens when $\mathrm{L} = \mathrm{M} = \mathrm{N}$.
- If we want the system in equation (52) to be shift-invariant, the aliasing terms in the summation have to be zero. It can be shown that this happens if and only if the matrix **C**(z) is pseudo-circulant, that is

Non-overlapped case

$$
\begin{aligned}
\mathbf{C}(z) &= \begin{bmatrix} C_{0\,0}(z) & C_{0\,1}(z) & \cdots & C_{0\,M-1}(z) \\ C_{1\,0}(z) & C_{1\,1}(z) & \cdots & C_{1\,M-1}(z) \\ \vdots & \vdots & \ddots & \vdots \\ C_{M-1\,0}(z) & C_{M-1\,1}(z) & \cdots & C_{M-1\,M-1}(z) \end{bmatrix} \\
&= \begin{bmatrix} E_0(z) & E_1(z) & \cdots & E_{M-2}(z) & E_{M-1}(z) \\ z^{-1}E_{M-1}(z) & E_0(z) & \cdots & E_{M-3}(z) & E_{M-2}(z) \\ z^{-1}E_{M-2}(z) & z^{-1}E_{M-1}(z) & \cdots & E_{M-4}(z) & E_{M-3}(z) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ z^{-1}E_1(z) & z^{-1}E_2(z) & \cdots & z^{-1}E_{M-1}(z) & E_0(z) \end{bmatrix}
\end{aligned}
\tag{53}
$$

Non-overlapped case

- In the non-overlapped case the blocks do not overlap neither at the input nor at the reconstruction stage at the output. This happens when $L = M = N$. Two important properties of pseudo-circulant matrices are:
  - A product of pseudo-circulant matrices is also pseudo-circulant. This implies that it is possible to exploit the decomposition of a pseudo-circulant matrix as a product of submatrices of the same type.
  - If a matrix $\mathbf{C}(z)$ is pseudo-circulant and has an inverse, its inverse is also pseudo-circulant.

- In this case the overall transfer function becomes

$$H(z) = z^{-M+1}[E_0(z^M) + z^{-1}E_1(z^M) + \cdots + z^{-(M-1)}E_{M-1}(z^M)] \quad (54)$$

that is, in the non-overlapped case, the polyphase components of the overall transfer function correspond to the functions $E_i(z)$ in equation (53).

Non-overlapped case

Example 8.3 Assume that $L = M = N = 2$ and demonstrate that the SISO transfer function is time-invariant when the transfer matrix **C**(z) is pseudo-circulant.

Solution

- The output signal can be described using its polyphase components as

$$Y(z) = [z^{-1} \quad 1] \begin{bmatrix} Y_0(z^2) \\ Y_1(z^2) \end{bmatrix} \tag{55}$$

where, as can be observed the polyphase components of $Y(z)$, denoted by $Y_i(z)$ for $i = 1, 2$, are the outputs of matrix **C**(z) when its inputs are the decimated polyphase components of the input signal.

Non-overlapped case

- Therefore, these polyphase components can be expressed as

$$Y_0(z) = \frac{1}{2}[X_0(z^{\frac{1}{2}}) + X_0(z^{\frac{1}{2}}W_2)]C_{00}(z) + \frac{1}{2}[X_1(z^{\frac{1}{2}}) + X_1(z^{\frac{1}{2}}W_2)]C_{01}(z) \tag{56}$$

and

$$Y_1(z) = \frac{1}{2}[X_0(z^{\frac{1}{2}}) + X_0(z^{\frac{1}{2}}W_2)]C_{10}(z) + \frac{1}{2}[X_1(z^{\frac{1}{2}}) + X_1(z^{\frac{1}{2}}W_2)]C_{11}(z) \tag{57}$$

respectively.

- The filter output can then be described as

$$
\begin{aligned}
Y(z) &= z^{-1}Y_0(z^2) + Y_1(z^2) \\
&= \frac{z^{-1}}{2}\left[(X_0(z) + X_0(zW_2))C_{00}(z^2) + (X_1(z) + X_1(zW_2))C_{01}(z^2)\right] \\
&\quad + \frac{1}{2}\left[(X_0(z) + X_0(zW_2))C_{10}(z^2) + (X_1(z) + X_1(zW_2))C_{11}(z^2)\right] \\
&= \frac{1}{2}\left[z^{-1}C_{00}(z^2)X_0(z) + z^{-1}C_{01}(z^2)X_1(z) + C_{10}(z^2)X_0(z) + C_{11}(z^2)X_1(z)\right] \\
&\quad + \frac{1}{2}\left[z^{-1}C_{00}(z^2)X_0(zW_2) + z^{-1}C_{01}(z^2)X_1(zW_2)\right. \\
&\quad \left. + C_{10}(z^2)X_0(zW_2) + C_{11}(z^2)X_1(zW_2)\right] \\
&= \frac{1}{2}\left[\left(z^{-1}C_{00}(z^2) + C_{10}(z^2)\right)X_0(z) + \left(z^{-1}C_{01}(z^2) + C_{11}(z^2)\right)X_1(z)\right] \\
&\quad + \frac{1}{2}\left[(z^{-1}C_{00}\left(z^2) + C_{10}(z^2)\right)X_0(zW_2) + \left(z^{-1}C_{01}(z^2) + C_{11}(z^2)\right)X_1(zW_2)\right] \\
&= \frac{1}{2}\left[\left(z^{-1}C_{00}(z^2) + C_{10}(z^2)\right)(X_0(z) + X_0(-z))\right. \\
&\quad \left. + \left(z^{-1}C_{01}(z^2) + C_{11}(z^2)\right)(X_1(z) + X_1(-z))\right]
\end{aligned}
\tag{58}
$$

Non-overlapped case

- According to equation (6), the polyphase components of the input signal can be expressed as

$$\begin{aligned} X_0(z) &= \frac{1}{2}\left[X(z) + X(-z)\right] \\ X_1(z) &= \frac{z^{-1}}{2}\left[X(z) - X(-z)\right] \end{aligned} \tag{59}$$

which imply that

$$\begin{aligned} X_0(z) + X_0(-z) &= X(z) + X(-z) \\ X_1(z) + X_1(-z) &= z^{-1}\left[X(z) - X(-z)\right] \end{aligned} \tag{60}$$

As a result, the z transform of the output signal can be expressed as

$$\begin{aligned} Y(z) = & \frac{1}{2}\Big[\left(z^{-1}C_{00}(z^2) + C_{10}(z^2)\right)\left(X(z) + X(-z)\right) \\ & + z^{-1}\left(z^{-1}C_{01}(z^2) + C_{11}(z^2)\right)\left(X(z) - X(-z)\right)\Big] \end{aligned} \tag{61}$$

Non-overlapped case

- If we choose $C_{00}(z) = C_{11}(z) = E_0(z)$ and $C_{01}(z) = zC_{10}(z) = E_1(z)$, then

$$\begin{aligned} Y(z) &= \left[z^{-1}(C_{00}(z^2) + C_{11}(z^2)) + C_{10}(z^2) + z^{-2}C_{01}(z^2)\right] X(z) \\ &= z^{-1}\left[E_0(z^2) + z^{-1}E_1(z^2)\right] X(z) \qquad (62) \end{aligned}$$

which has no aliasing component, meaning that the transfer function between the filter input and output is time invariant. Note that the above equation is in the same form as equation (54). $\triangle$

Overlapped input and output

- We now discuss more general forms of implementing shift-invariant systems using overlapped block filtering.

- We will start from the non-overlapped case analyzed. From such implementations, we use the matrix representations of the serial-to-parallel and parallel-to-serial converters with overlapping in order to, from an $M \times M$ matrix $\mathbf{C}(z)$ (that implements a system without overlapping), generate an implementation corresponding to a factorization

$$\mathbf{C}(z) = \mathbf{P}_{N}^{M}(z)\hat{\mathbf{C}}(z)\mathbf{S}_{L}^{M}(z) \tag{63}$$

In this case, $\hat{\mathbf{C}}(z)$ is an $N \times L$ matrix that implements a system with overlapping, while $\mathbf{S}_{L}^{M}(z)$ and $\mathbf{P}_{N}^{M}(z)$ correspond to the serial-to-parallel and parallel-to-serial converters, respectively.

Overlapped input and output

- In order to derive an expression for the matrices $\mathbf{S}_{L}^{M}(z)$ and $\mathbf{P}_{N}^{M}(z)$ from equations (42) and (45), we must first define the z transform of a signal block as in equation (39), that is

$$\mathbf{X}_{L}^{M}(z) = \sum_{m=-\infty}^{\infty} \mathbf{x}_{L}^{M}(m) z^{-m} \tag{64}$$

- Also, applying the above definition to equation (41), we have that

$$\mathcal{Z}\{\mathcal{D}^{M}\{\mathbf{x}_{L}^{M}(m)\}\} = \mathcal{Z}\{\mathbf{x}_{L}^{M}(m-1)\} = z^{-1}\mathbf{X}_{L}^{M}(z) \tag{65}$$

Overlapped input and output

- Therefore, in the z-transform domain, equations (42) and (45) become, for $L < 2M$ and $N < 2M$,

$$\mathbf{x}_L^M(z) = \begin{bmatrix} \mathbf{I}_{L-M} & 0 \\ 0 & \mathbf{I}_{2M-L} \\ z^{-1}\mathbf{I}_{L-M} & 0 \end{bmatrix} \mathbf{x}_M^M(z) \tag{66}$$

$$\mathbf{Y}_M^M(z) = \begin{bmatrix} 0 & \mathbf{I}_{2M-N} & 0 \\ z^{-1}\mathbf{I}_{N-M} & 0 & \mathbf{I}_{N-M} \end{bmatrix} \mathbf{W}_N^M(z) \tag{67}$$

respectively.

Overlapped input and output

- Since $\mathbf{C}(z)$ represents the non-overlapping block processing and $\widehat{\mathbf{C}}(z)$ the overlapping block processing, we have that

$$\mathbf{Y}_{M}^{M}(z) = \mathbf{C}(z)\mathbf{X}_{M}^{M}(z) \tag{68}$$

$$\mathbf{W}_{N}^{M}(z) = \widehat{\mathbf{C}}(z)\mathbf{X}_{L}^{M}(z) \tag{69}$$

Overlapped input and output

- Therefore, from equations (63), (66), (67), (68), and (69), we conclude that the matrices $\mathbf{S}_L^M(z)$ and $\mathbf{P}_N^M(z)$ have the following general forms:

$$\mathbf{S}_L^M(z) = \begin{bmatrix} \mathbf{I}_{L-M} & 0 \\ 0 & \mathbf{I}_{2M-L} \\ z^{-1}\mathbf{I}_{L-M} & 0 \end{bmatrix} \tag{70}$$

$$\mathbf{P}_N^M(z) = \begin{bmatrix} 0 & \mathbf{I}_{2M-N} & 0 \\ z^{-1}\mathbf{I}_{N-M} & 0 & \mathbf{I}_{N-M} \end{bmatrix} \tag{71}$$

Example 8.4

For the case where $L = M = 4$ and $N = 7$, implement the transfer function

$$H(z) = z^{-3}[E_0(z^4) + z^{-1}E_1(z^4) + z^{-2}E_2(z^4) + z^{-3}E_3(z^4)] \tag{72}$$

in a block form, by noting that overlapping is applied only at the output.

Solution

- We should choose matrix $\hat{\mathbf{C}}(z)$ as

$$\hat{\mathbf{C}}(z) = \begin{bmatrix} E_3(z) & 0 & 0 & 0 \\ E_2(z) & E_3(z) & 0 & 0 \\ E_1(z) & E_2(z) & E_3(z) & 0 \\ E_0(z) & E_1(z) & E_2(z) & E_3(z) \\ 0 & E_0(z) & E_1(z) & E_2(z) \\ 0 & 0 & E_0(z) & E_1(z) \\ 0 & 0 & 0 & E_0(z) \end{bmatrix} \tag{73}$$

Example

- With the above choice, from equations (70) and (71), we respectively have that

$$\mathbf{S}_4^4(z) = \mathbf{I}_4 \tag{74}$$

$$\mathbf{P}_7^4(z) = \begin{bmatrix} 0 & \mathbf{I}_1 & 0 \\ z^{-1}\mathbf{I}_3 & 0 & \mathbf{I}_3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ z^{-1} & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & z^{-1} & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & z^{-1} & 0 & 0 & 0 & 1 \end{bmatrix} \tag{75}$$

Example

- Note that $\mathbf{S}_4^4(z)$ is an identity matrix because there is no overlap between blocks at the input. With these choices, we have $\mathbf{C}(z) = \mathbf{P}_7^4(z)\hat{\mathbf{C}}(z)\mathbf{S}_4^4(z) = \mathbf{P}_7^4(z)\hat{\mathbf{C}}(z)$, so that

$$\mathbf{C}(z) = \begin{bmatrix} E_0(z) & E_1(z) & E_2(z) & E_3(z) \\ z^{-1}E_3(z) & E_0(z) & E_1(z) & E_2(z) \\ z^{-1}E_2(z) & z^{-1}E_3(z) & E_0(z) & E_1(z) \\ z^{-1}E_1(z) & z^{-1}E_2(z) & z^{-1}E_3(z) & E_0(z) \end{bmatrix} \tag{76}$$

which is a pseudo-circulant matrix representing the overall transfer function of the block digital filter, according to equation (52).

$\triangle$

Example

- For the algorithm presented the input blocks are not overlapped, whereas the output blocks are overlapped. We now generalize this algorithm for an input block size $L = M$ and an output block size $N = 2M - 1$. The matrix $\hat{\mathbf{C}}(z)$ of dimensions $(2M - 1) \times M$ should have the form

$$\hat{\mathbf{C}}(z) = \begin{bmatrix} E_{M-1}(z) & 0 & \cdots & 0 \\ E_{M-2}(z) & E_{M-1}(z) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ E_0(z) & E_1(z) & \cdots & E_{M-1}(z) \\ 0 & E_0(z) & \cdots & E_{M-2}(z) \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & E_0(z) \end{bmatrix} \tag{77}$$

in order for the product $\hat{\mathbf{P}}^{M}_{2M-1}(z)\hat{\mathbf{C}}(z)\mathbf{I}_M$ to result in a pseudo-circulant matrix.

Example 8.5

We now consider the implementation of the transfer function of the Example for $L = 7$ and $N = 4 = M$. Note that overlapping is applied only at the input.

Solution

- We should choose:

$$\hat{\mathbf{C}}(z) = \begin{bmatrix} E_0(z) & E_1(z) & E_2(z) & E_3(z) & 0 & 0 & 0 \\ 0 & E_0(z) & E_1(z) & E_2(z) & E_3(z) & 0 & 0 \\ 0 & 0 & E_0(z) & E_1(z) & E_2(z) & E_3(z) & 0 \\ 0 & 0 & 0 & E_0(z) & E_1(z) & E_2(z) & E_3(z) \end{bmatrix} \quad (78)$$

With the above choice, we have from equations (70) and (71) that

$$\hat{\mathbf{S}}_7^4(z) = \begin{bmatrix} \mathbf{I}_3 & 0 \\ 0 & \mathbf{I}_1 \\ z^{-1}\mathbf{I}_3 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ z^{-1} & 0 & 0 & 0 \\ 0 & z^{-1} & 0 & 0 \\ 0 & 0 & z^{-1} & 0 \end{bmatrix} \tag{79}$$

$$\hat{\mathbf{P}}_4^4(z) = \mathbf{I}_4 \tag{80}$$

This case leads to the same $\mathbf{C}(z)$ as the previous example.

$\triangle$

Example

- In the structure of this Example, the input blocks are overlapped, whereas the output blocks are not overlapped. This structure can also be generalized for $M = N$ and $L = 2M - 1$ by employing the block transfer function matrix $\hat{\mathbf{C}}(z)$ of dimensions $M \times (2M - 1)$ given by

$$\hat{\mathbf{C}}(z) = \begin{bmatrix} E_0(z) & E_1(z) & \cdots & E_{M-1}(z) & & \cdots & 0 \\ 0 & E_0(z) & \cdots & E_{M-2}(z) & E_{M-1}(z) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & E_0(z) & E_1(z) & \cdots & E_{M-1}(z) \end{bmatrix} \quad (81)$$

- It is possible to generate several alternative structures for values of L, M, N different from those discussed here. For each case, a distinct form for $\hat{\mathbf{C}}(z)$ is required for proper generation of shift-invariant structure.
- However, determining the right $\hat{\mathbf{C}}(z)$ in order to achieve this is not trivial.

Fast convolution structure I

- Inspired by overlapped block implementation it is possible to derive structure I, corresponding to the case where $L = M = 2$, and $N = 3$. This structure was derived by decomposing matrix $\hat{\mathbf{C}}(z)$ as follows

$$\begin{bmatrix} E_1(z) & 0 \\ E_0(z) & E_1(z) \\ 0 & E_0(z) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} E_1(z) & 0 & 0 \\ 0 & E_0(z)+E_1(z) & 0 \\ 0 & 0 & E_0(z) \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix} \tag{82}$$

Fast convolution structure I

- Each polynomial $E_i(z)$, for $i = 0, 1$, is a polyphase component of $H(z)$, and therefore corresponds to a filter operation with an FIR filter of about half the length of the original filter $H(z)$.
- This structure then shows how to implement an FIR filtering of a given order through three FIR filters of half order and operating at half rate. This can be done recursively, since each half-order FIR filter can be further decomposed into three other subfilters.
- Every time a new decomposition is applied the number of multiplications per sample is reduced whereas the latency (delay) of the response increases.

Fast convolution structure II

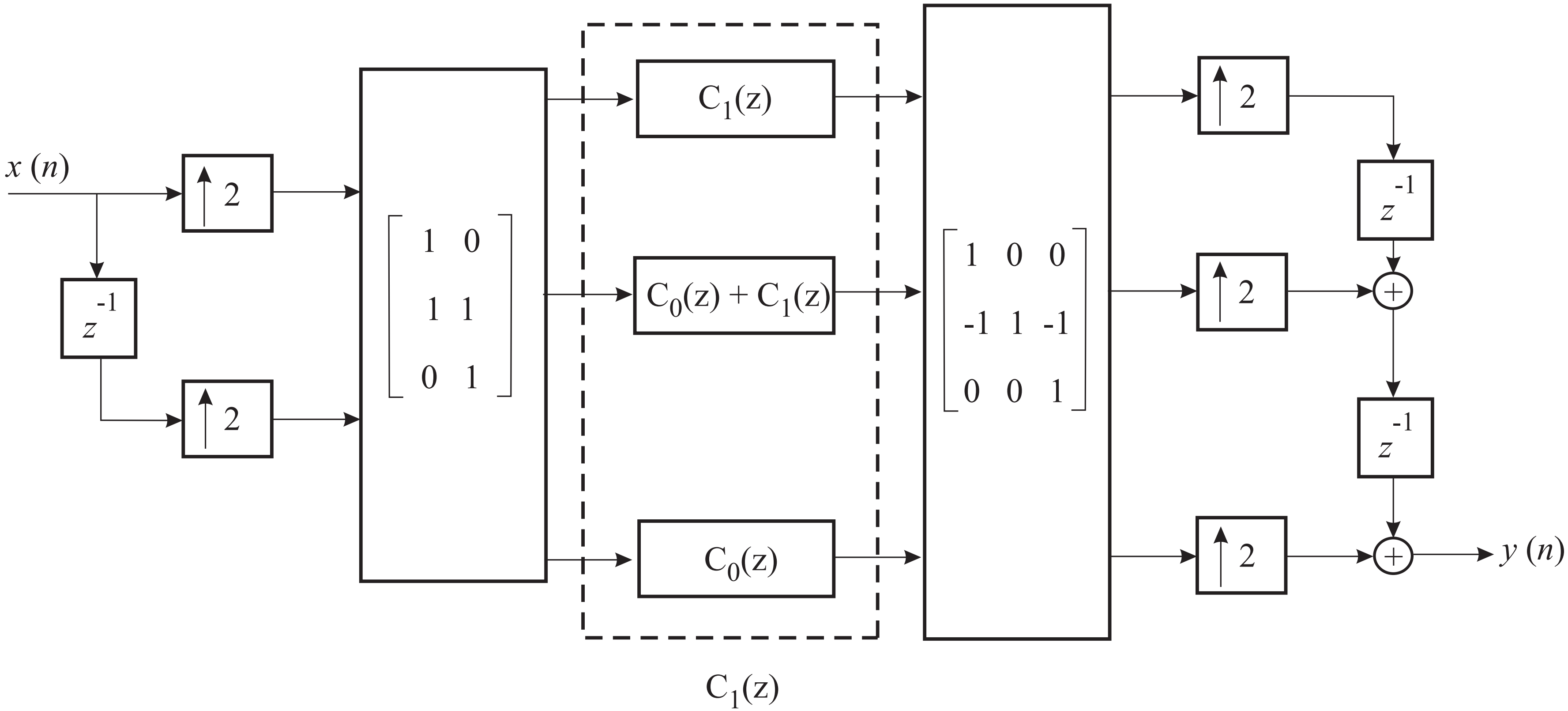


Figure 26: Overlapped structure I for fast convolution. $C_i(z) = E_i(z), i = 0, 1$.

Fast convolution structure II

- Structure II is exactly the transpose of overlapped block structure I. Note that in the transposition of multirate systems, decimators become interpolators and vice versa.

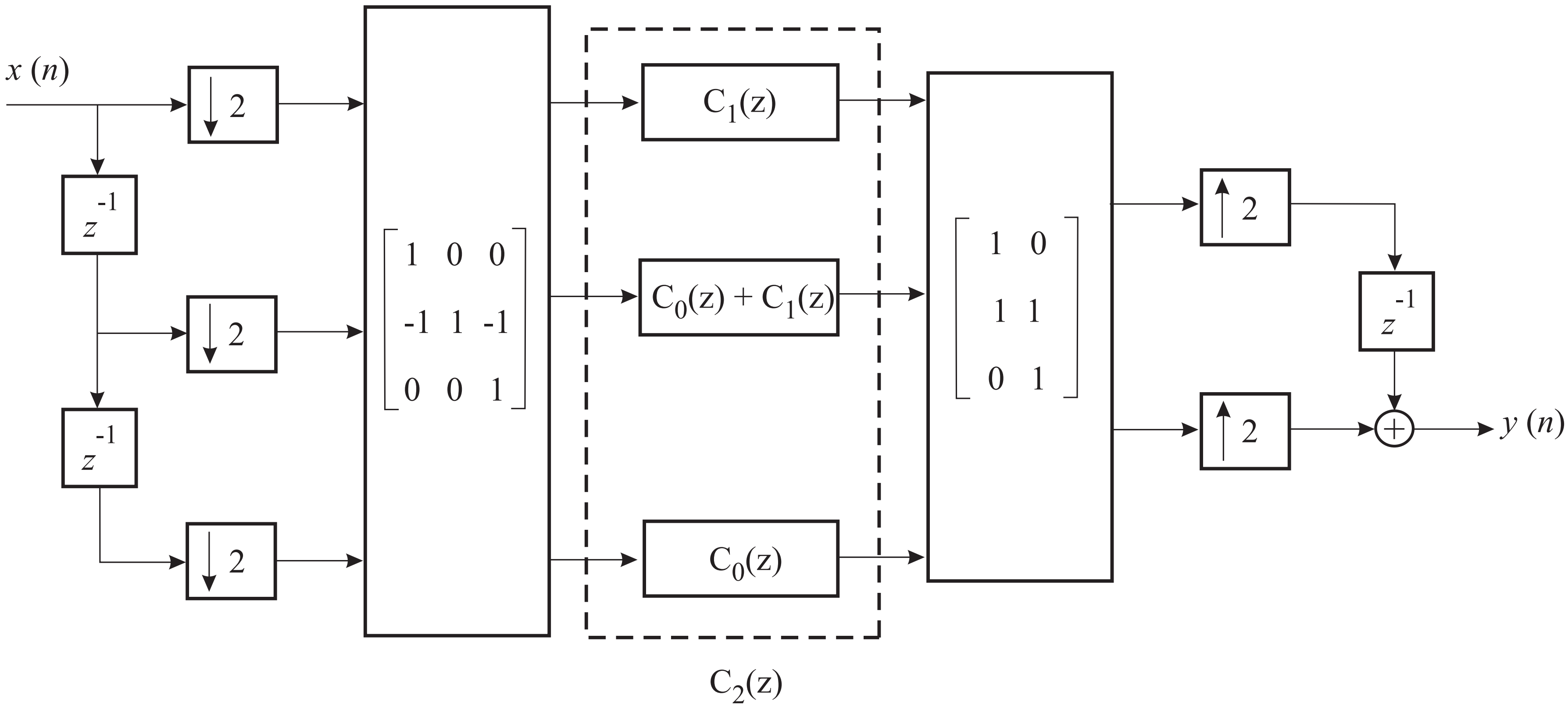


Figure 27: Overlapped structure II for fast convolution. $C_i(z) = E_i(z), i = 0, 1$.

Fast convolution

Example 8.6 Determine if the transfer function matrix below can be the block representation of a linear and time-invariant system.

$$\mathbf{C}(z) = \begin{bmatrix} 1+z^{-1} & -\frac{1}{2a} \\ 2a & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1+2bz^{-1}+z^{-2} \end{bmatrix} \begin{bmatrix} 1 & \frac{(1+z^{-1})}{2a} \\ 0 & 1 \end{bmatrix} \quad (83)$$

Solution

- Let us start by computing $\mathbf{C}(z)$ to access its properties:

$$\begin{aligned}\mathbf{C}(z) &= \begin{bmatrix} 1+z^{-1} & -\frac{1}{2a} - \frac{b}{a}z^{-1} - \frac{1}{2a}z^{-2} \\ 2a & 0 \end{bmatrix} \begin{bmatrix} 1 & \frac{1}{2a}(1+z^{-1}) \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1+z^{-1} & -\frac{1}{2a} - \frac{b}{a}z^{-1} - \frac{1}{2a}z^{-2} + \frac{1}{2a}(1+2z^{-1}+z^{-2}) \\ 2a & 1+z^{-1} \end{bmatrix} \\ &= \begin{bmatrix} 1+z^{-1} & \frac{1-b}{a} \\ 2a & 1+z^{-1} \end{bmatrix} \end{aligned} \tag{84}$$

- The matrix above represents a linear and time-invariant system, if it is pseudo-circulant. Hence, the following condition must be satisfied:

$$\frac{1-b}{a} = 2a \quad \Rightarrow \quad b = 1 - 2a^2 \tag{85}$$

$\triangle$

Fast convolution

Example 8.7

1. Propose a block implementation for a linear time-invariant transfer function using the matrices $\mathbf{S}_L^M(z)$ and $\mathbf{P}_N^M(z)$ in equations (70) and (71), such that the input and output blocks have overlaps given by $L = 4$ and $N = 3$, respectively. The number of subchannels is $M = 2$.

2. Implement the transfer function below with the proposed structure and draw the overall realization

$$H(z) = z^{-4} + z^{-3} + 2z^{-2} + 4z^{-1} \tag{86}$$

Fast convolution

Solution

- (a)

  For this case $N = 3$, $M = 2$, and $L = 4$. Therefore,

$$\mathbf{S}_4^2(z) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ z^{-1} & 0 \\ 0 & z^{-1} \end{bmatrix} \quad (87)$$

$$\mathbf{P}_3^2(z) = \begin{bmatrix} 0 & 1 & 0 \\ z^{-1} & 0 & 1 \end{bmatrix} \quad (88)$$

- A possible and simple solution for $\hat{\mathbf{C}}(z)$, leading to the minimum overall delay, is given by

$$\hat{\mathbf{C}}(z) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ E_0(z) & E_1(z) & 0 & 0 \\ 0 & E_0(z) & E_1(z) & 0 \end{bmatrix} \tag{89}$$

Fast convolution

- By post-multiplying the matrix $\hat{\mathbf{C}}(z)$ by $\mathbf{S}_4^2(z)$, it follows that

$$\hat{\mathbf{C}}(z)\mathbf{S}_4^2(z) = \begin{bmatrix} 0 & 0 \\ E_0(z) & E_1(z) \\ z^{-1}E_1(z) & E_0(z) \end{bmatrix} \quad (90)$$

- By pre-multiplying the resulting matrix by $\mathbf{P}_3^2(z)$, we have

$$\mathbf{C}(z) = \mathbf{P}_3^2(z)\hat{\mathbf{C}}(z)\mathbf{S}_4^2(z) = \begin{bmatrix} E_0(z) & E_1(z) \\ z^{-1}E_1(z) & E_0(z) \end{bmatrix} \quad (91)$$

which is pseudo-circulant.

Fast convolution

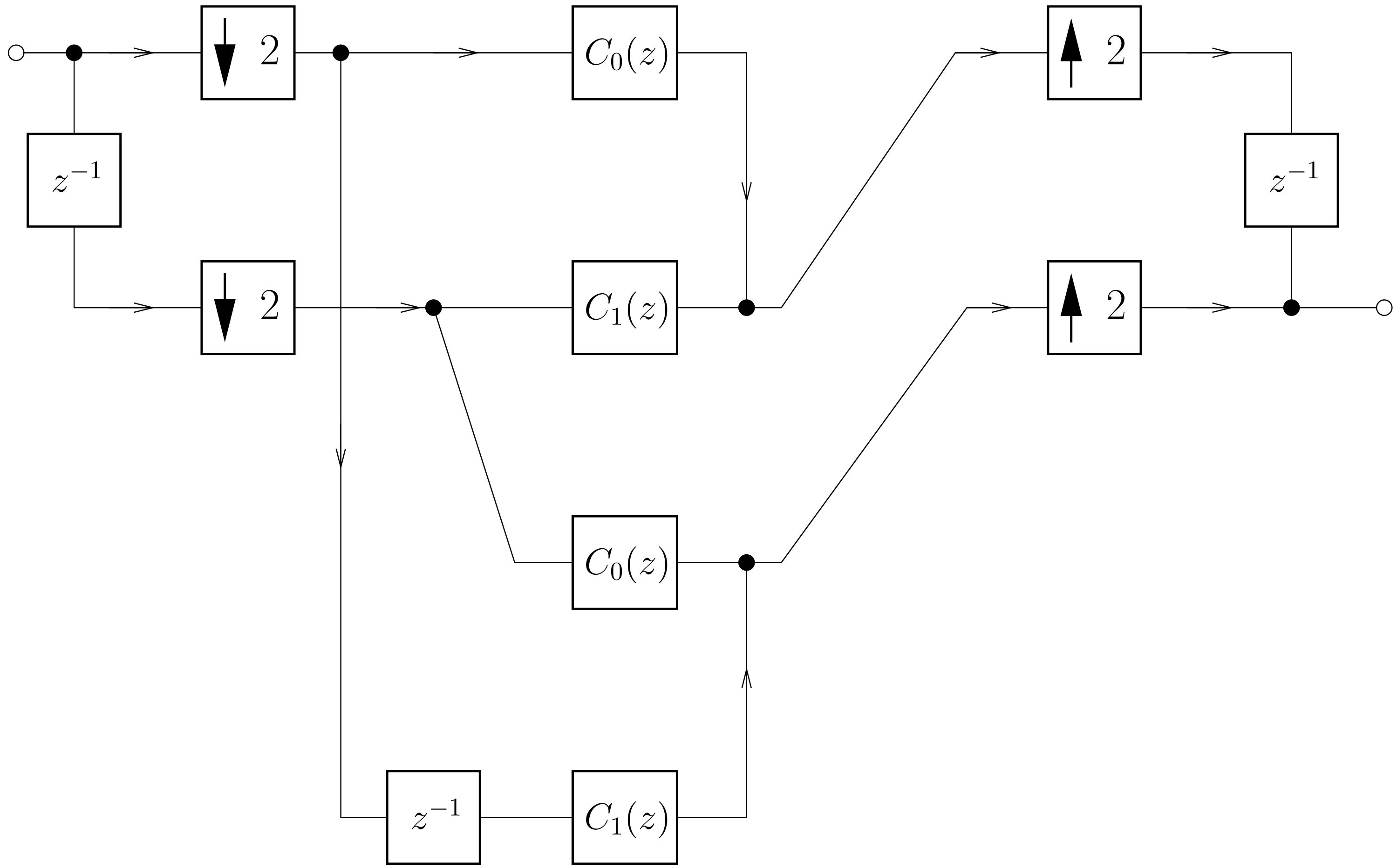


Figure 28: Realization of equation (91). $C_i(z) = E_i(z)$, $i = 0, 1$.

Fast convolution

- Since aliasing is canceled, the transfer function is given by

$$
\begin{aligned}
H(z) &= \begin{bmatrix} z^{-1} & 1 \end{bmatrix} \mathbf{C}(z^2) \begin{bmatrix} 1 \\ z^{-1} \end{bmatrix} \\
&= \begin{bmatrix} z^{-1} & 1 \end{bmatrix} \begin{bmatrix} E_0(z^2) + z^{-1}E_1(z^2) \\ z^{-1}E_0(z^2) + z^{-2}E_1(z^2) \end{bmatrix} \\
&= 2z^{-1}\left[E_0(z^2) + z^{-1}E_1(z^2)\right] \qquad (92)
\end{aligned}
$$

From the above expression and equation (86), $E_0(z)$ and $E_1(z)$ become

$$
\begin{aligned}
E_0(z) &= 2 + \frac{z^{-1}}{2} \qquad (93) \\
E_1(z) &= 1 + \frac{z^{-1}}{2} \qquad (94)
\end{aligned}
$$

$\triangle$

Random signals in multirate systems

- An important concept often present in rate change of stochastic signals is that of cyclostationary processes. A real random process $\{X\}$ is wide-sense cyclostationary (WSCS) with period M if its mean value and autocorrelation function satisfy

$$E\{X(n)\} = E\{X(n + kM)\} \tag{95}$$

and

$$R_X(n, k) = R_X(n + M, k + M) = E\{X(n + M)X(k + M)\} \tag{96}$$

for all n, k.

Random signals in multirate systems

- As the definitions state, the mean and the autocorrelation function are periodic with period M. Very often this property appears in several practical applications. Examples include sampling in communication systems, modulation, multiplexing, and the interaction of WSS processes with multirate systems.

- Let us assume now that the serial-to-parallel converter of consecutive samples of a realization of a WSCS process without overlap as follows

$$\mathbf{x}_M^M(m) = [X(mM) \;\; X(mM-1) \;\; \ldots \; X(mM-M+1)]^{\mathsf{T}} \tag{97}$$

For a given random input vector, the autocorrelation matrix is defined as

$$\mathbf{R}_{\mathbf{X}_M^M}(m) = E\{\mathbf{X}_M^M(m)\mathbf{X}_M^{M^{\mathsf{T}}}(m)\} \tag{98}$$

Random signals in multirate systems

- The characteristics of the autocorrelation matrix play a key role in understanding the effects of multirate processing in random signals. Note that if the input process is WSCS with period M, the block vector $\mathbf{X}_M^M(m)$ is WSS, that is, the matrix $\mathbf{R}_{\mathbf{X}_M^M}(m)$ does not depend on m.

Random signals in multirate systems

- Assume that an $M \times 1$ WSS vector $\mathbf{x}_M^M(m)$ is input to a transfer function matrix $\mathbf{C}(z)$ of dimensions $N \times M$, then the power spectral density (PSD) of the output vector will be given by

$$\boldsymbol{\Gamma}_{\mathrm{u}}(z) = \mathbf{C}(z)\boldsymbol{\Gamma}_{\mathrm{x}_M^M}(z)\mathbf{C}^{\mathsf{T}}(z^{-1}) \tag{99}$$

where

$$\boldsymbol{\Gamma}_{\mathrm{x}_M^M}(z) = \sum_{\nu=-\infty}^{\infty} \mathbf{R}_{\mathrm{x}_M^M}(\nu) z^{-\nu} \tag{100}$$

is the PSD of the input signal vector. The expressions in (99) and (100) are M-dimensional generalizations.

Random signals in multirate systems

- A very important property of the PSD matrix formulation is that if the input process to a serial-to-parallel converter is WSS, then the PSD matrix $\mathbf{\Gamma}_{X_M^M}(z)$ is pseudo-circulant Conversely, in case the vector $\mathbf{x}_M^M(m)$ output by a serial-to-parallel converter is WSS and its PSD matrix is pseudo-circulant, then the input process to the serial-to-parallel converter is WSS.

Interpolated random signals

- If a WSS random signal is applied to the input of an interpolator, the signal random $\hat{X}(n)$ at the output is WSCS with period L.
- Its blocked output vector is given by

$$\hat{\mathbf{x}}_L^L(m) = [\hat{X}(mL)\ \hat{X}(mL-1)\ \ldots\ \hat{X}(mL-L+1)]^T$$
$$= [\hat{X}(mL)\ 0\ \ldots\ 0]^T \quad (101)$$
$$= [X(m)\ 0\ \ldots\ 0]^T \quad (102)$$

Interpolated random signals

- So that its autocorrelation matrix is

$$
\begin{aligned}
\mathbf{R}_{\hat{\mathbf{X}}_L^L}(m) &= E\{\hat{\mathbf{X}}_L^L(m)\hat{\mathbf{X}}_L^{L^T}(m)\} \\
&= \begin{bmatrix} E\{\hat{X}^2(mL)\} & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix} \qquad (103) \\
&= \begin{bmatrix} E\{X^2(m)\} & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix} \qquad (104)
\end{aligned}
$$

Interpolated random signals

- Since $X(m)$ is WSS, then $E\{X^2(m)\}$ is constant for all m, and therefore the correlation matrix is not a function of m, that is, the vector $\hat{\mathbf{x}}_L^L(m)$ is WSS. This implies that its unblocked version $\hat{X}(n)$ is WSCS with period L.

Decimated random signals

- Consider now the case where a random signal is applied to a decimator as in Figure 1.
- The decimated random signal $X_d(n)$ is the result of retaining every Mth sample of the input random signal denoted as $X(nM)$.
- If we assume the general case where the input signal is WSCS with period N, the decimated process will also be WSCS but with a period P.

Decimated random signals

- In order to determine the value of P, we analyze the properties of the autocorrelation function of the decimated signal, that is

$$\begin{aligned} R_{X_d}(n, l) &= E\{X_d(n)X_d(l)\} \\ &= E\{X(nM)X(lM)\} \end{aligned} \tag{105}$$

If the output signal is WSCS with period P then

$$\begin{aligned} R_{X_d}(n+P, l+P) &= E\{X_d(n+P)X_d(l+P)\} \\ &= E\{X\left((n+P)M\right)X\left((l+P)M\right)\} \end{aligned} \tag{106}$$

Decimated random signals

- Considering that we assumed the input process WSCS with period N, the equality of (106) holds if $PM = iN$ for some integer i. Therefore, the period P should be

$$P = \frac{N}{\text{gcd}(M, N)} \tag{107}$$

where $\text{gcd}(\cdot)$ stands for the greatest common divisor between two integer numbers.

Some special choices for M and N are worth mentioning:

  – If $N = 1$, then $P = 1$, meaning that if the input to the decimator is WSS, then its output is also WSS.

  – If N and M are prime numbers, then $P = N$.

  – If $N = M$, then $P = 1$, indicating that a cyclostationary signal, when decimated by its cyclostationarity period becomes WSS.

Do-it-yourself: Multirate systems

Experiment .1: Consider a sinusoidal signal s of frequency $f_1 = 0.01$ Hz corrupted by noise and sampled at $F_s = 1$ samples/s for a time interval of ten minutes.

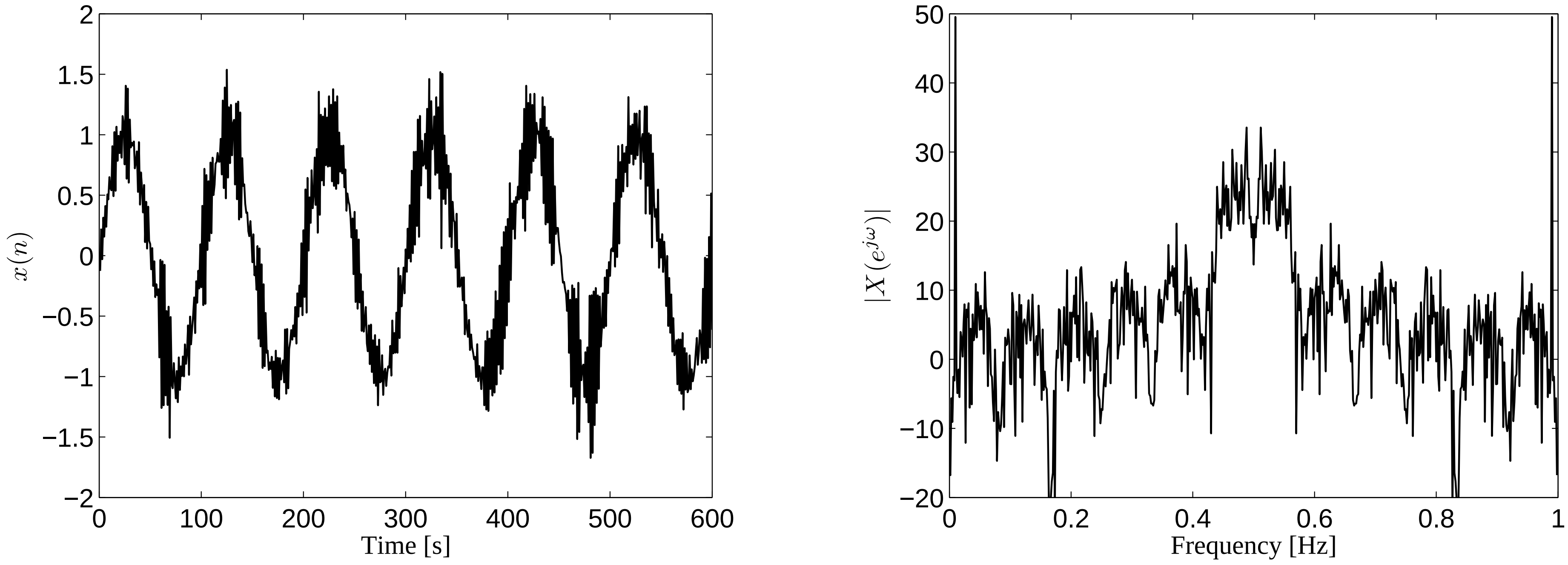
- Assume that the noise component is the output of the filter

$$H_1(z) = \frac{1}{12}\sum_{i=0}^{11}(-1)^i z^{-i} \tag{108}$$

to a zero-mean and unit-variance Gaussian noise, such that

```
Fs = 1; Ts = 1/Fs; duration = 600;
time = 0:Ts:(duration-Ts); Ntime = length(time);
s = sin(2*pi*f1*time);
w = randn(1,Ntime); w = w-mean(w); w =
w./sqrt(w*w');
h1 = [1 -1 1 -1 1 -1 1 -1 1 -1 1 -1]./12;
wh1 = filter(h1,1,w);
x = s+wh1;
```

Do-it-yourself: Multirate systems



(a) (b)

Figure 29: Signal s corrupted by noise: (a) time domain; (b) frequency domain.

Do-it-yourself: Multirate systems

- In order to simplify the sinusoidal storage or transmission, we may decimate signal `x` by `M=10`, after performing a proper lowpass filtering to minimize aliasing distortion, as given by

```
ordh2 = 20; h2 = ones(1,ordh2+1)./(ordh2+1);
xh2 = filter(h2,1,x);
M = 10; xdec = xh2(1:M:Ntime);
```

  resulting in the `xdec` signal characterized in Figure 30.

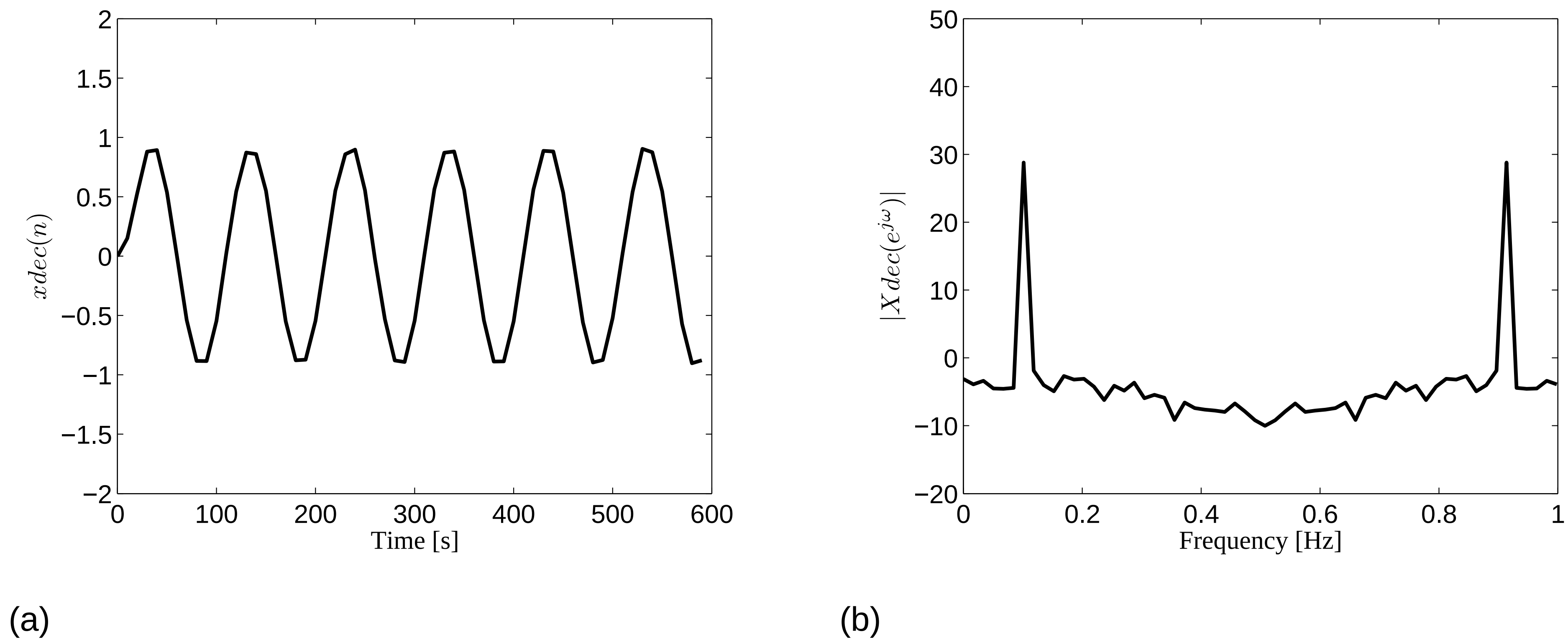


Figure 30: Signal s corrupted by noise filtered and decimated by 10: (a) time domain; (b) frequency domain.

Do-it-yourself: Multirate systems

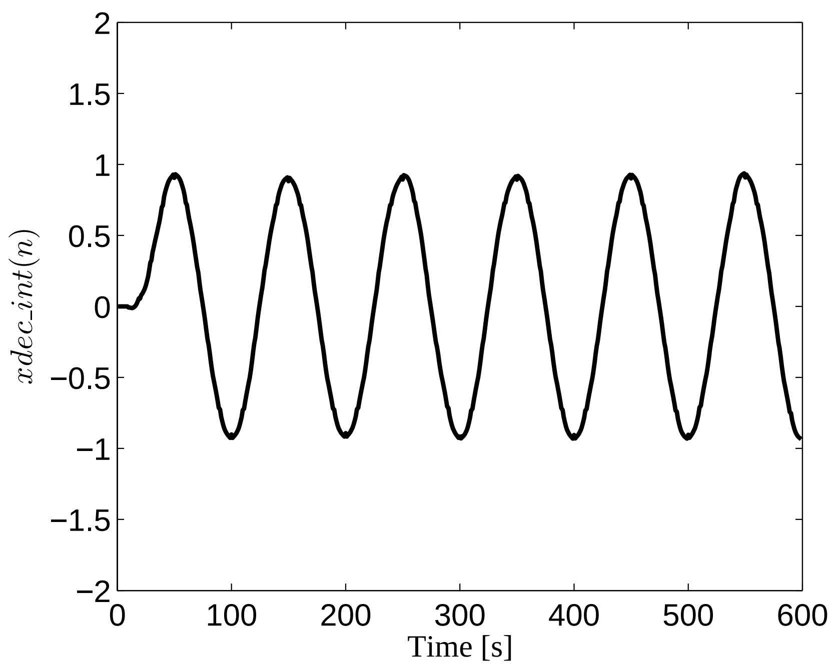
- The sampling rate can be expanded back to its original value by introducing `(M-1)` zeros between each two consecutive samples of `xdec`, which in MATLAB can be performed as

```
xaux = [xdec; zeros(M-1,Ntime/M)];
xaux2 = reshape(xaux,1,Ntime);
```

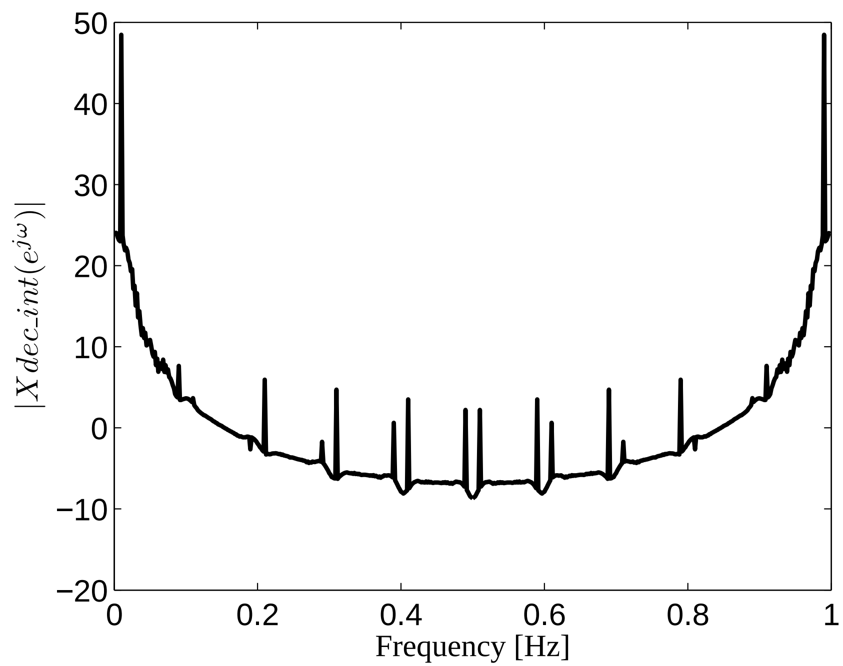
  This procedure causes spectral repetitions that must be removed by a proper lowpass filtering such as

```
h3 = firpm(30,[0 0.01 0.09 0.5]*2,[1 1 0 0]);
xdec_int = filter(M*h3,1,xaux2);
```

  Figure 31 depicts the filtered signal and its corresponding spectral representation, which can be readily compared to signal `xh2`, before the decimation operation.



(a)



(b)

Figure 31: Decimated signal from Figure 30 interpolated by 10 and filtered: (a) time domain; (b) frequency domain.

Do-it-yourself: Multirate systems

- In the last step of this experiment, one can use a bandpass, instead of a lowpass, filter `h3` to generate a modulated version of the original signal. This type of processing is employed in Experiment 11.1, which the student is motivated to read, to modulate a signal without an explicit multiplication by a high-frequency sinusoidal carrier.

  All rate-changing operations employed in the present experiment can be performed automatically with the MATLAB commands `decimate` and `interp`, which already include the corresponding lowpass filtering stage.

$\triangle$