ECE 844 - Unit 2

<u>Changing the Sampling Rate Using Discrete-Time Processing</u> (See Sect. 4.6 in text)

Assume that we have a discrete-time signal **x(n)** which was obtained by sampling an continuous-time signal **x_c(t)** with time spacing T between samples. That is,

$$x(n) = x_c(nT)$$

We now desire to obtain a new discrete-time signal, **x'(n)** which is defined as

$x'(n) = x_c(nT')$, where $T' \neq T$

<u>Approach A</u>:

1. Reconstruct **x_c(t)** from **x(n)** using ideal D/C conversion:

$$x_r(t) = \sum_{n=-\infty}^{\infty} x(n) \frac{\sin[\frac{\pi}{T}(t - nT)]}{\frac{\pi}{T}(t - nT)}$$

Resample $x_r(t)$ at the desired sampling interval T' to obtain $x'(n) = x_c(nT')$

or (usually preferred):

<u>Approach B</u>:

-Approach B1: Use a "sampling rate compressor" to reduce the sampling rate.

or

- Approach B2: Use a "sampling rate expander" to increase the sampling rate.

Reducing the Sampling Rate by an Integer Factor (Sampling Rate Compression)

Let $x_d(n) = x(nM)$ where M is a positive integer.

A sampling rate compressor which implements this re-sampling is shown in Figure 4.20:

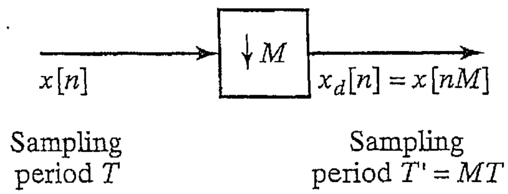


Figure 4.19 Representation of a Compressor or discrete-time samlper.

If **x(n)** was obtained by sampling **x_c(t)** at a sampling rate of $\frac{1}{T}$,
then $x(nM) = x_d(n)$ corresponds to re-sampling $x_c(t)$ at a reduced sampling rate of $\frac{1}{MT}$.
Note that if the original sampling rate was at least M times the Nyquist rate, then the Sampling Theorem will still be satisfied after "downsampling" by a factor of M.

Frequency domain view-point:

From development of the Sampling Theorem, we know that if **x(n)** is obtained by sampling **x_c(t)** at a rate of 1/T, then the relation between $X(e^{j\omega})$ and $X_c(j\Omega)$ is

$$X(e^{j\omega}) = \frac{1}{T}\sum_{k=-\infty}^{\infty} X_c\left[j\left(\frac{\omega}{T} - \frac{2\pi k}{T}\right)\right], \quad \text{where } \omega = \Omega T \qquad \text{(equation 4.71)}$$

Similarly, the relation for $X_d(e^{j\omega})$ can be obtained by replacing T with T_d = MT in the above (and also changing the summation index to r):

$$X_d(e^{j\omega}) = \frac{1}{MT}\sum_{r=-\infty}^{\infty} X_c\left[j\left(\frac{\omega}{MT} - \frac{2\pi r}{MT}\right)\right] \qquad \text{(equation 4.73)}$$

In order to obtain the relation between $X_d(e^{j\omega})$ and $X(e^{j\omega})$, we now express the summation index in the previous equation as

r = i + kM

and perform a double summation with i ranging from 0 to M-1 and with k ranging from $-\infty$ to ∞. Note that the resulting value of r will also range from $-\infty$ to ∞.

Table of values of r

i value	(k=0)	(k=1)	(k=2)	(k=3)	(etc)
0	0	M	2M	3M	etc
1	1	M+1	2M+1	3M+1	etc
2	2	M+2	2M+2	3M+2	etc
etc					
M-1	M-1	2M-1	3M-1	4M-1	etc

Therefore, the expression for $X_d(e^{j\omega})$ becomes

$$X_d(e^{j\omega}) = \frac{1}{M}\sum_{i=0}^{M-1}\left\{\frac{1}{T}\sum_{k=-\infty}^{\infty} X_c\left[j\left(\frac{\omega}{MT} - \frac{2\pi k}{T} - \frac{2\pi i}{MT}\right)\right]\right\} \qquad \text{(equation 4.75)}$$

$$= \frac{1}{M}\sum_{i=0}^{M-1}\left\{\frac{1}{T}\sum_{k=-\infty}^{\infty} X_c\left[j\left(\frac{(\omega-2\pi i)}{MT} - \frac{2\pi k}{T}\right)\right]\right\} = \frac{1}{M}\sum_{i=0}^{M-1} X(e^{j(\omega-2\pi i)/M}) \qquad \text{(equation 4.787}$$

Note that $X_d(e^{j\omega})$ consists of M copies of a frequency scaled (expanded) version of the original $X(e^{j\omega})$ positioned at $\omega=0$, $\omega=2\pi$, $\omega=4\pi$, . . . , $\omega=(M-1)2\pi$. (The frequency scale factor used is M.) Figure 4.20 shows $X_d(e^{j\omega})$ for the case of M=2. (No aliasing occurs in this case since $\omega_{\max}$ for $X(e^{j\omega})$ is π/2.) Note: $\omega_{\max}$ is defined as the largest value of ω (for $\omega<\pi$) for which $X(e^{j\omega})$ is non-zero.

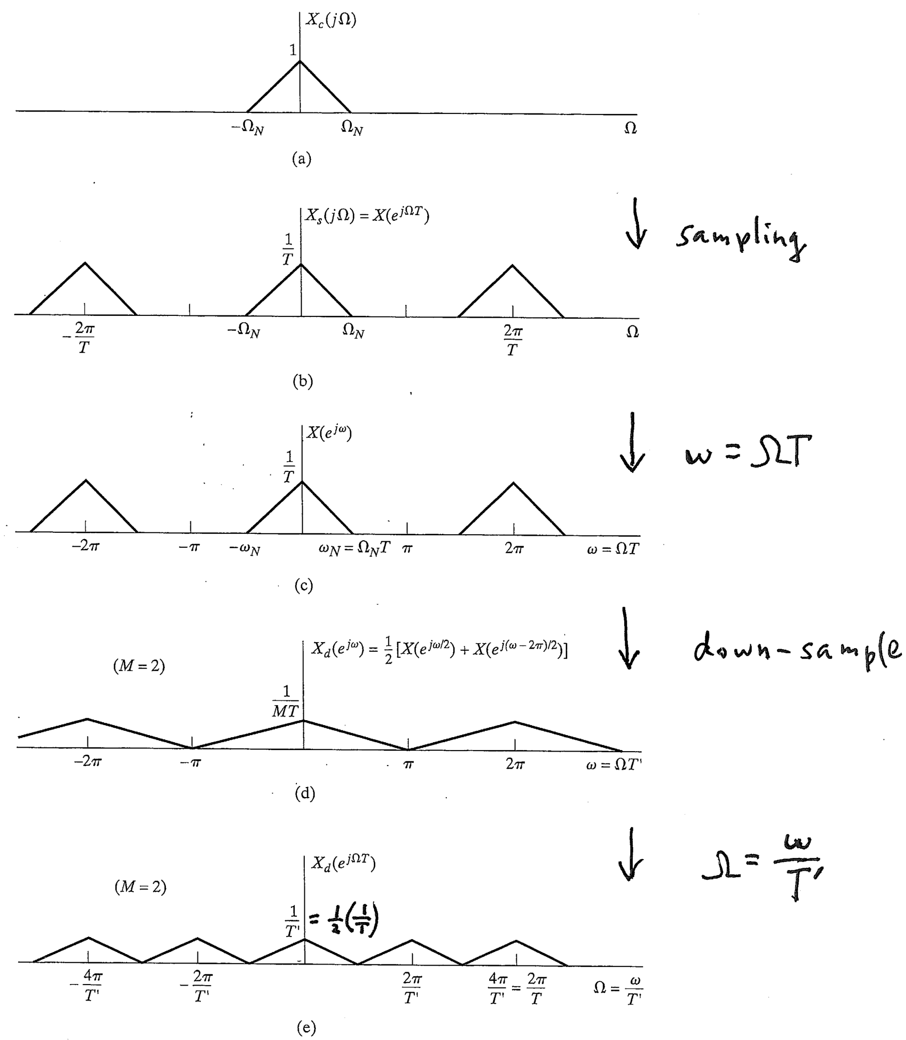


Figure 4.20 Frequency-domain illustration of down-sampling.

Figure 4.21 shows $X_d(e^{j\omega})$ for the case of M=3. This time there is aliasing present in $X_d(e^{j\omega})$.
In this case, aliasing can be avoided, if we are willing to low-pass filter the signal prior to down-sampling, as shown in the figure.

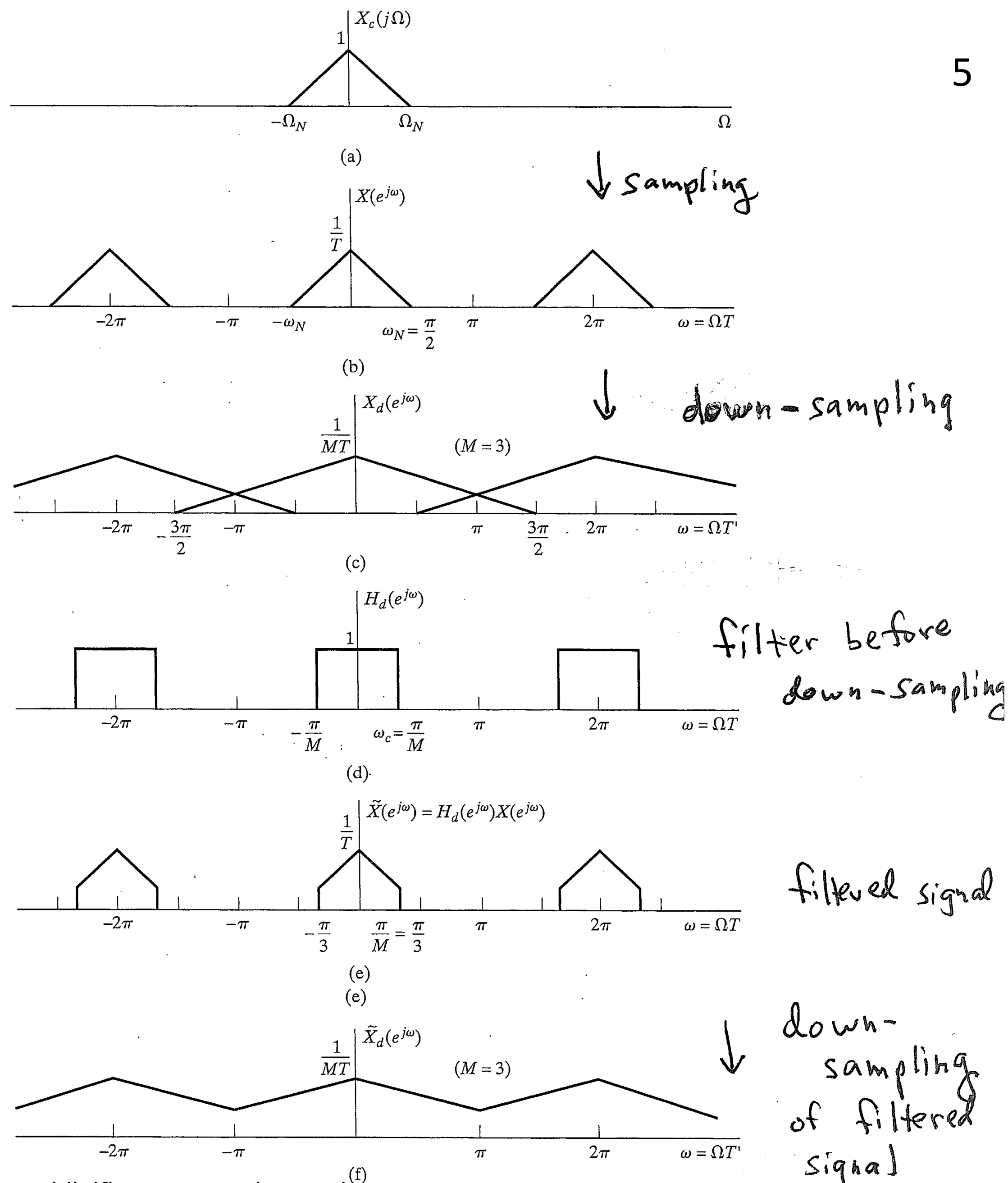


Figure 4.21 (a)-(c) Downsampling with aliasing. (d)-(f) Downsampling with Prefiltering to avoid aliasing.

Increasing the Sampling Rate by an Integer Factor (See Section 4.6.2 in text)

Assume that **x(n)** was obtained by sampling a continuous signal $\mathbf{x_r(t)}$ at a rate of $\frac{1}{T}$. That is, $x(n) = x_c(nT)$

Now we want to obtain new discrete-time signal x'(n) which corresponds to samples of $x_c(t)$ taken at a higher sampling rate of $L\left(\frac{1}{T}\right)$, where L is an integer.

Consider the following approach to obtain x'(n):

First, insert L-1 0's between each of the original samples to increase the total number of samples by a factor of L. Call this signal $\mathbf{x_e(n)}$. This signal can be expressed in terms of the original signal **x(n)** using

$x_e(n) = x(n/L)$, for n an integer multiple of L , and

$x_e(n) = 0$, for all other values of n.

Equivalently, we could write

$$x_e(n) = \sum_{k=-\infty}^{\infty} x(k)\delta(n-kL) = x(0)\delta(n) + x(1)\delta(n-L) + x(2)\delta(n-2L) + \cdots$$

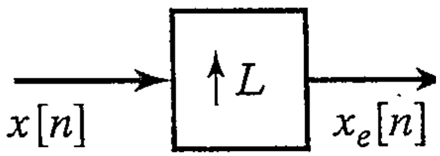
The frequency domain representation of $x_e(n)$ can be found from

$$X_e(e^{j\omega}) = \sum_{n=-\infty}^{\infty}\left(\sum_{k=-\infty}^{\infty} x(k)\delta(n-kL)\right)e^{-j\omega n}$$

$$= \sum_{k=-\infty}^{\infty} x(k)\left(\sum_{n=-\infty}^{\infty} \delta(n-kL)\right)e^{-j\omega n}$$

$$= \sum_{k=-\infty}^{\infty} x(k)e^{-j\omega kL} = X(e^{j\omega L}) \qquad \text{(equation 4.85)}$$

This insertion of zeros as "position holders" between samples of the original signal is represented as:



This system is called a "sampling rate expander."

In the frequency domain, the above operation corresponds to frequency scaling (by a compression factor of L), of the original $X(e^{j\omega})$.

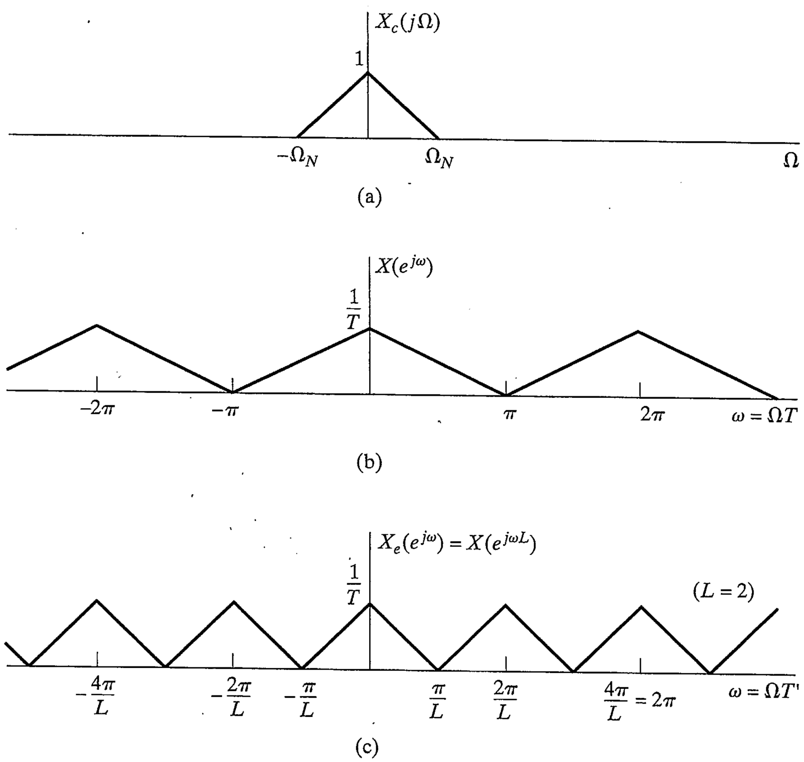


Figure 4.24 Frequency domain illustration of interpolation (parts d and e on next slide)

From a frequency domain perspective, an ideal lowpass filter with cutoff of π/L and gain L can now be used to obtain a frequency domain function equivalent to the DTFT of a discrete-time signal consisting of samples of $x_c(t)$ sampled at a rate L times faster than the rate used to sample x(n).

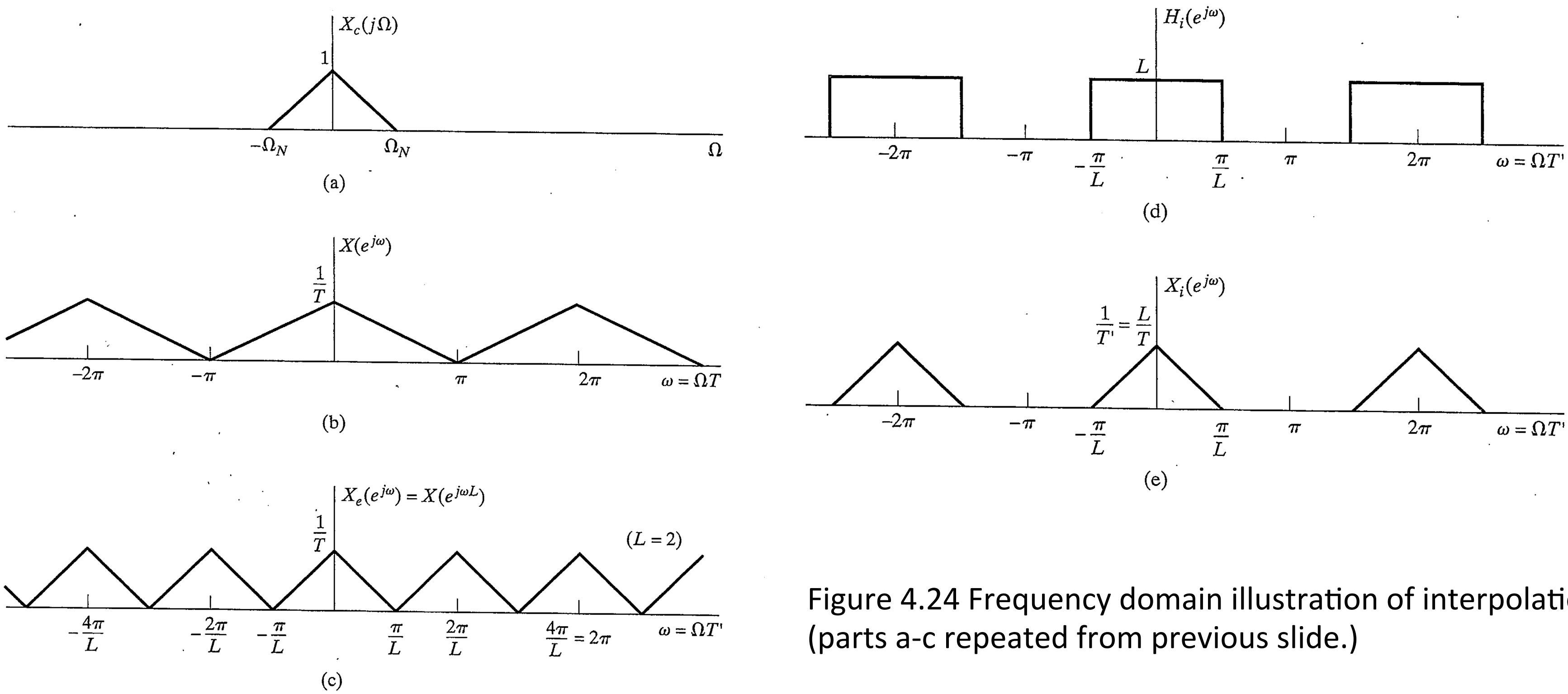


Figure 4.24 Frequency domain illustration of interpolation (parts a-c repeated from previous slide.)

The complete 2-step process for increasing the sampling rate by an an integer factor of L is shown below:

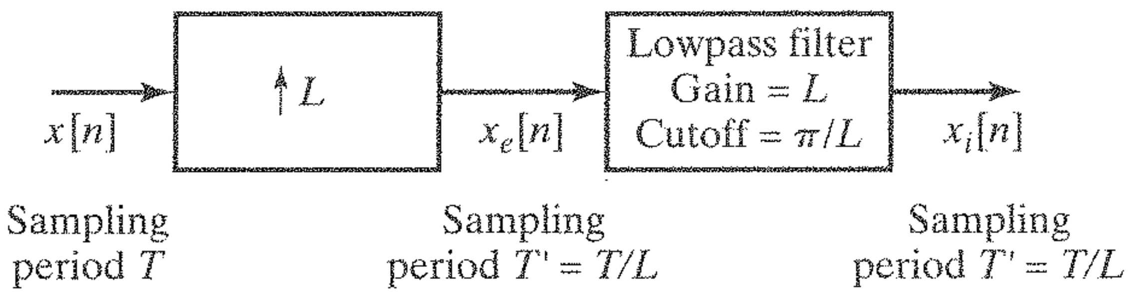


Figure 4.23 General system for sampling rate increase by L.

The overall system, consisting of a sampling rate expander and a lowpass filter, is call an "interpolator."

The operation of passing $x_e(n)$ though an ideal low-pass filter with cutoff frequency of π/L and gain of L corresponds to convolving $x_e(n)$ with a digital filter whose h(n) is given by

$$h_i(n) = L\frac{\sin(\frac{\pi}{L}n)}{\pi n} \text{ for } n \neq 0, \qquad \text{and} \qquad h_i(n) = 1 \text{ for } n = 0.$$

The output of this convolution can be expressed as

$$\begin{aligned} x_i(n) &= \sum_{k=-\infty}^{\infty} x_e(k)h_i(n-k) \\ &= \sum_{k=-\infty}^{\infty}\left(\sum_{j=-\infty}^{\infty} x(j)\delta(k-jL)\right)h_i(n-k) \\ &= \sum_{j=-\infty}^{\infty} x(j)\sum_{k=-\infty}^{\infty}\delta(k-jL)h_i(n-k) \\ &= \sum_{j=-\infty}^{\infty} x(j)h_i(n-jL) \end{aligned}$$

(now change summation index from j to k)

$$\begin{aligned} x_i(n) &= \sum_{k=-\infty}^{\infty} x(k)h_i(n-kL) \\ &= \sum_{k=-\infty}^{\infty} x(k)\, L\frac{\sin\left(\frac{\pi}{L}(n-kL)\right)}{\pi(n-kL)}, \qquad n \neq \text{integer} \bullet L \\ &= \sum_{k=-\infty}^{\infty} x(k)\frac{\sin\left(\frac{\pi}{L}(n-kL)\right)}{\left(\frac{\pi}{L}\right)(n-kL)} \qquad n \neq \text{integer} \bullet L \end{aligned}$$

(One term in the summation would blow up for some value of k.)

(equation 4.88)

When n = pL,

$$x_i(n) = \sum_{k=-\infty}^{\infty} x(k)h_i(pL - kL)$$

$$= \sum_{k=-\infty}^{\infty} x(k)h_i[L(p-k)]$$

Note that $h_i[L(p-k)] = 1 \text{ when } p = k$

and

$$h_i[L(p-k)] = \frac{\sin\left(\frac{\pi}{L}L(p-k)\right)}{\frac{\pi}{L}L(p-k)} = \frac{\sin\left(\pi(p-k)\right)}{\pi(p-k)} = 0 \qquad \text{when } p \neq k$$

Therefore, (still for the case where n = pL):

$$x_i(n) = \sum_{k=-\infty}^{\infty} x(k)h_i(pL - kL) = x(p)$$

$$= x\left(\frac{n}{L}\right) = x_c(nT_i) \qquad \text{where} \quad T_i = \frac{T}{L}$$

Interesting side note: Linear Interpolation can also be represented by a convolution.

Example: Note that linear interpolation can be used to find x(1), x(2), x(3), and x(4), using known values of x(0) and x(5), as follows:

x(1) = x(0) + .2[x(5) - x(0)] =.8x(0) + .2x(5)

x(2) = x(0) + .4[x(5) - x(0)] =.6x(0) + .4x(5)

x(3) = x(0) + .6[x(5) - x(0)] =.4x(0) + .6x(5)

x(4) = x(0) + .8[x(5) - x(0)] =.2x(0) + .8x(5)

Note this interpolation could be implemented by a convolution with **x_e(n)** and the **h(n)** shown below:

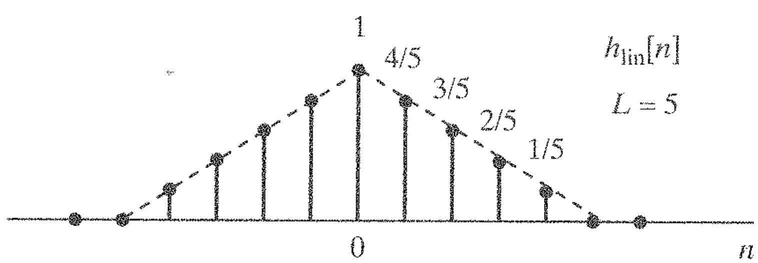


Figure 4.25 Impulse response for linear interpolation.

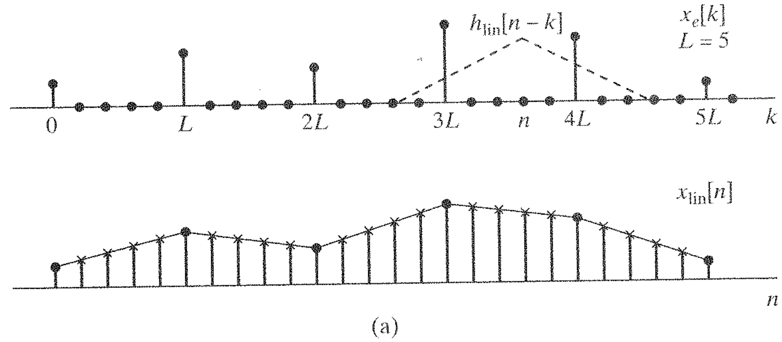


Figure 4.26 (a) Illustration of linear interpolation by filtering. (b) Frequency response of linear interpolator compared with ideal lowpass interpolation filter.

In general, the h(n) that can be used to perform linear interpolation between x(n) values that are spaced by n = L is

$$h_{lin}(n) = 1 - |n|/L, \qquad \text{for } |n| \le L$$
$$= 0, \qquad \text{for all other n.}$$

A frequency domain view-point of the linear interpolator is shown below in figure 4.26 (b). The frequency domain version of the ideal interpolator, $H_i(e^{j\omega})$, is also shown to permit comparison.

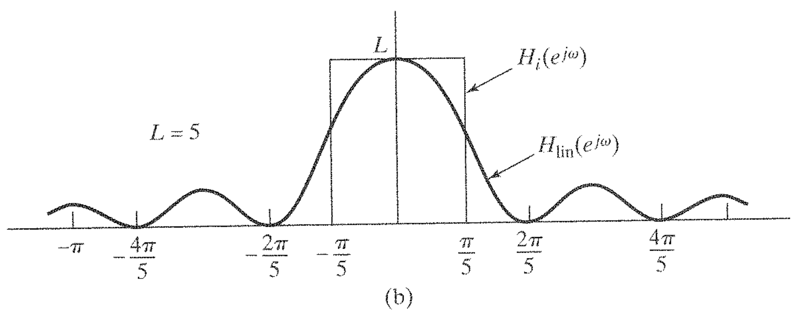


Figure 4.26 (a) Illustration of linear interpolation by filtering. (b) Frequency response of linear interpolator compared with ideal lowpass interpolation filter.

Changing the sampling rate by a non-integer factor (See Section 4.6.4)

Now it is desired to change the sampling period T to $T\frac{M}{L}$, where M and L are integers. This is the same as changing the sampling rate to $\left(\frac{1}{T}\right)\left(\frac{L}{M}\right)$. This change in T and $\frac{1}{T}$ can be accomplished by combining two steps:

1. Upsample by a factor of L.
2. Downsample by a factor of M .

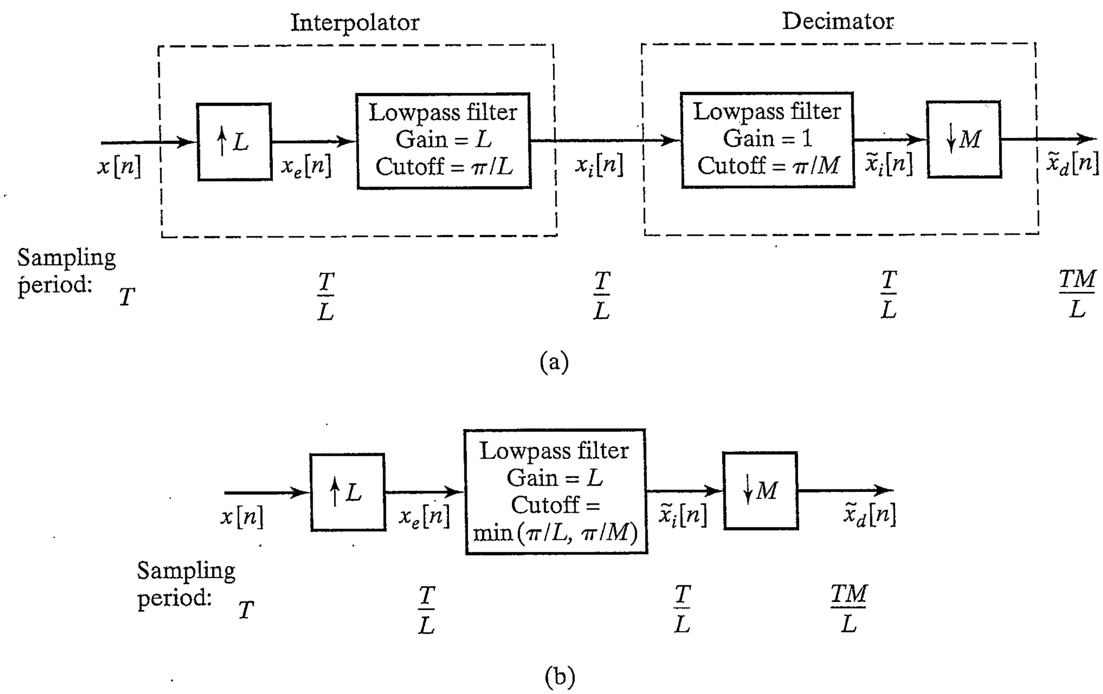


Figure 4.29 (a) System for changing the sampling rate by a noninteger factor. (b) Simplified system in which the decimation and interpolation filters are combined.

<u>Example</u>: L = 2 and M = 3

- sampling period increased from T to $\frac{3}{2}T$.
- sampling rate decreased from $\frac{1}{T}$ to $\frac{2}{3}\frac{1}{T}$

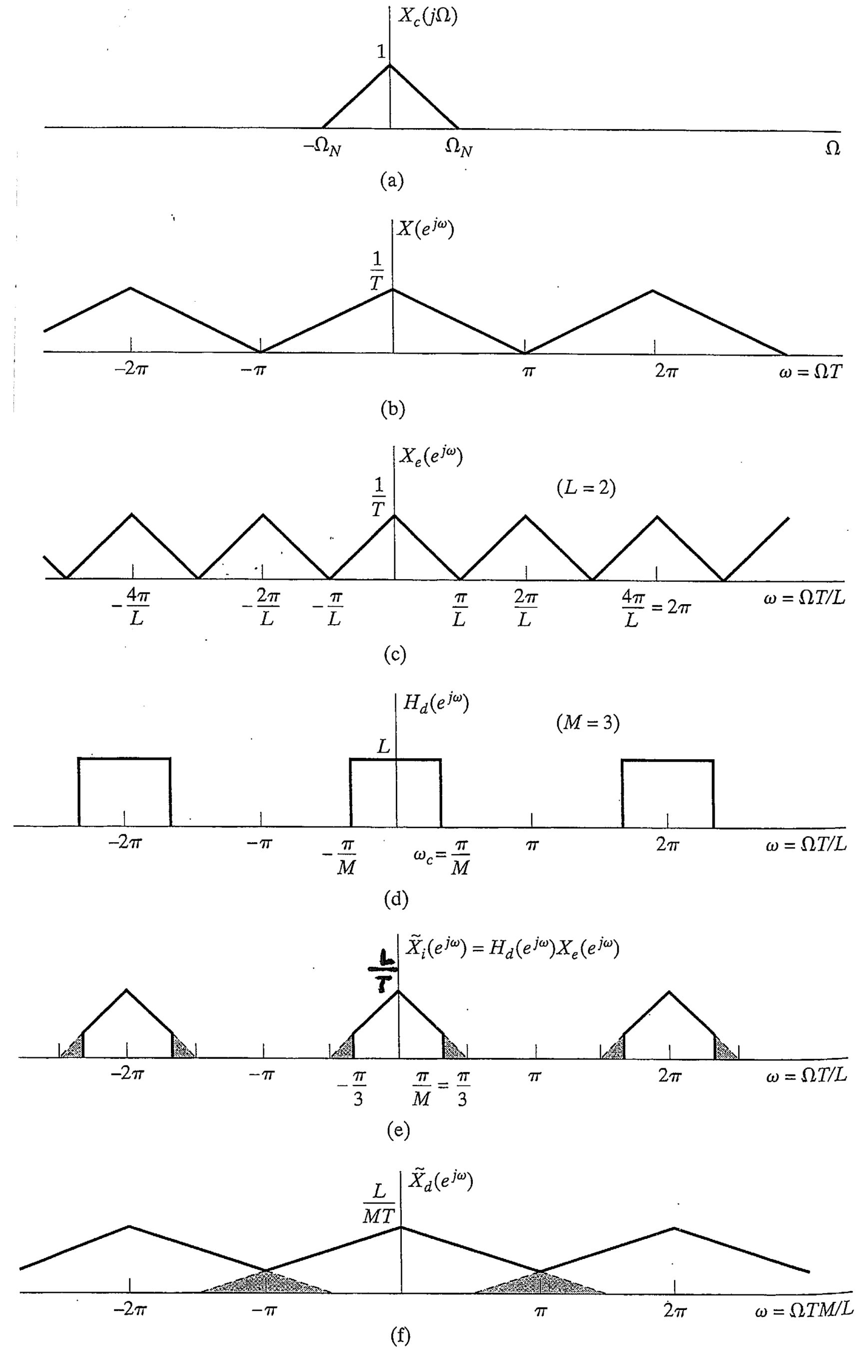


Figure 4.30 Illustration of changing
The smapling rate by a noninteger factor