Interpolation Formula:

Signal Restoration: Cont-time operation

$$x_p(t) \longrightarrow \boxed{H_{LP}(j\Omega)} \longrightarrow x_c(t)$$

impulse train obtained from the signal samples.

original

$$x_c(t) = x_p(t) * h_{LP}(t)$$

where $h_{LP}(t) = T_s \frac{\Omega_c}{\pi} \operatorname{sinc}\left(\frac{\Omega_c t}{\pi}\right)$

$$x_p(t) = \sum_{n=-\infty}^{\infty} x_c(nT)\, \delta(t - nT_s)$$

$$\left(\delta(t-t_0) * h(t) = h(t-t_0) \right)$$

$$x_c(t) = \sum_{n=-\infty}^{\infty} x(nT_s) \left(\underbrace{h_{LP}(t) * \delta(t - nT_s)}_{h_{LP}(t-nT_s)} \right)$$

Interpolation formula:

$$x_c(t) = \sum_{n=-\infty}^{\infty} x(nT_s)\, T_s \frac{\omega_c}{\pi} \operatorname{sinc}\left(\frac{\Omega_c (t - nT_s)}{\pi} \right)$$

where T_s: sampling period

Ω_c: cut-off freq. of the lowpass filter.

(usually $\Omega_c = \frac{\Omega_s}{2} = \frac{2\pi}{T_s \cdot 2} = \frac{\pi}{T_s}$)

RHS: $x(0), x(\mp T_s), x(\mp 2T_s), \ldots$

LHS: $x(t)$ cont.-time signal

The signal value at an arbitrary $t = t_0$:

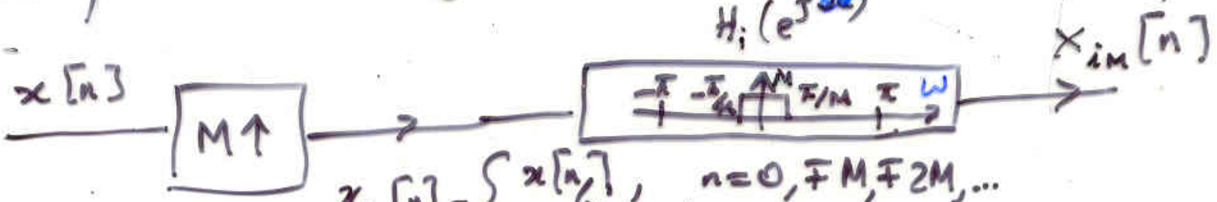
$$x(t_0) = \sum_{n=-\infty}^{\infty} x(nT_s) \operatorname{sinc}\left(\frac{\omega_c (t_0 - nT_s)}{\pi} \right), \quad \omega_c = \frac{\omega_s}{2}$$

Nyquist Frequency:

$$\Omega_s > 2\Omega_M$$

Interpolation formula is not practical to implement because we need to perform infinite sums (sinc is of infinite-extent).

Interpolation by M:

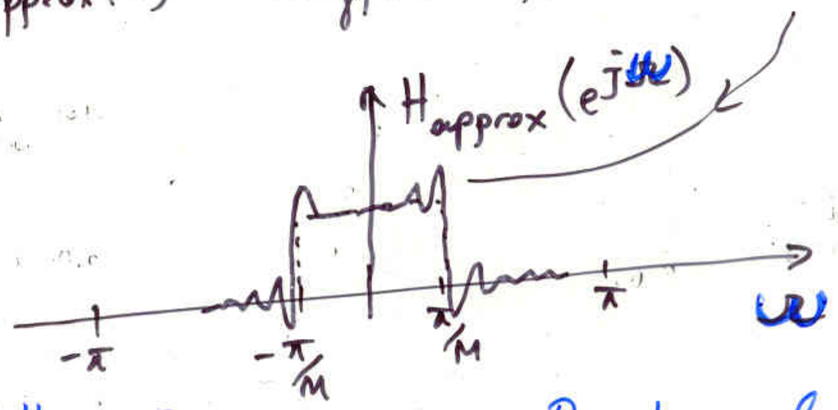


$$x_M[n] = \begin{cases} x[n/M], & n = 0, \mp M, \mp 2M, \dots \\ 0, & \text{otherwise.} \end{cases}$$

* In practice ideal $H_i(e^{j\omega})$ cannot be implemented. $h_i[n] = \frac{1}{2\pi}\int_{-\pi}^{\pi} H_i(e^{j\omega})\, e^{+j\omega n}\, d\omega \rightarrow$ sinc. (It is infinite extent & no recursive form).

$$h_{approx}[n] = \begin{cases} h_i[n], & |n| < K \\ 0, & \text{o.w.} \end{cases}$$

$h_{approx}[n]$ suffers from Gibbs Phenomenon.



Ex) Filter Design by Rectangular Windowing: (Not great!)

$M = 2$. $K = 2$

Ideal filter:

$$h_i[n] = \frac{1}{2\pi}\int_{-\pi/2}^{\pi/2} 2\, e^{j\omega n}\, d\omega, \quad n = 0, \mp 1, \dots$$

$h_i[0] = 1$

$$h_i[n] = \frac{1}{2\pi}\, 2\, \frac{e^{j\omega n}}{jn}\Bigg|_{-\pi/2}^{\pi/2} = \frac{1}{\pi}\,\frac{e^{j\frac{\pi}{2}n} - e^{-j\frac{\pi}{2}n}}{jn}, \quad n = 1, 2, \dots, -1, -2, \dots$$

$$h_i[1] = \frac{1}{\pi}\,\frac{e^{j\pi/2} - e^{-j\pi/2}}{j1} = \frac{1}{\pi}\,\frac{j + j}{j} = \frac{2}{\pi} = h_i[-1]$$

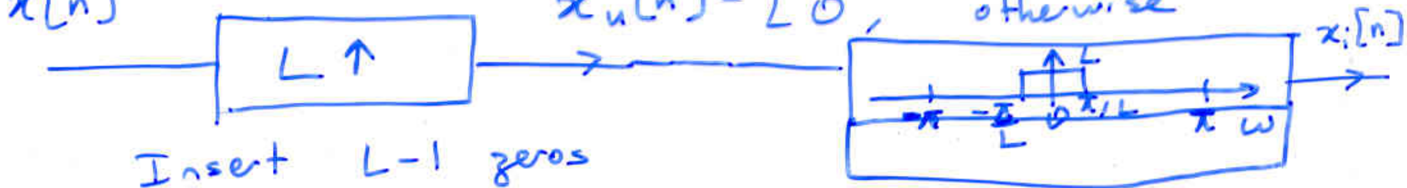
$h_{approx}[n] = \{ h_i[1], h_i[0], h_i[1] \} \longrightarrow$ no good!

Oct. 4/2004

Discrete-time Domain Interpolation:

by a factor of L

$$x_u[n] = \begin{cases} x[n/L], & \text{if } n = 0, \mp L, \mp 2L, \dots \\ 0, & \text{otherwise} \end{cases}$$



Insert L−1 zeros between every other sample.

Discrete-time low-pass filter

Ex: L = 3

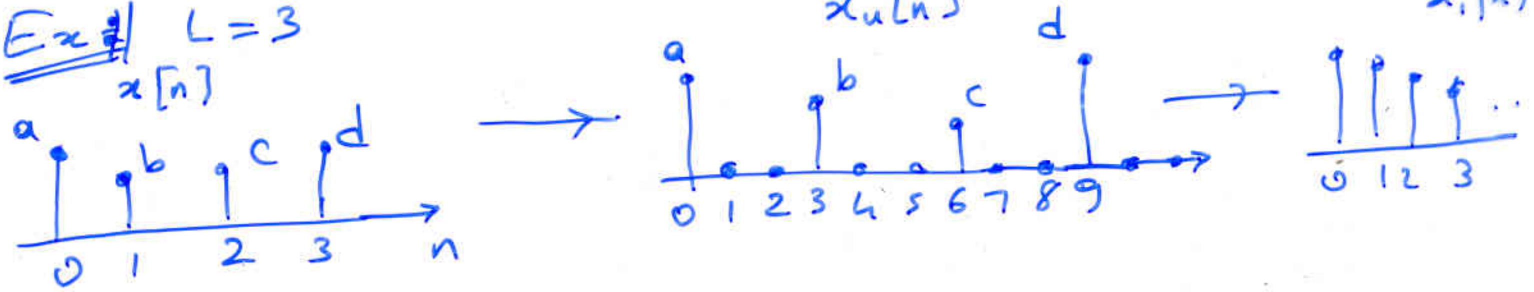


Image Interpolation:

Interp. by L = 2

50x50

rowwise interpolation

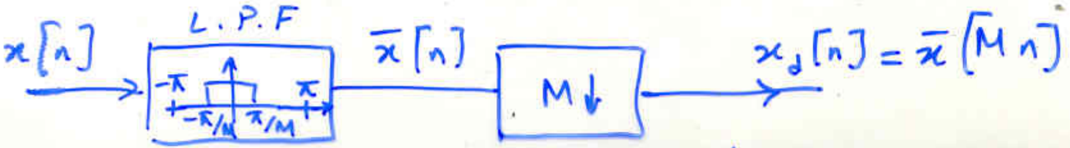
treat each row as a 1-D signal

50 rows x 100 columns

columnwise interp. (treat each column as a 1-D signal

100 X 100

Decimation by M

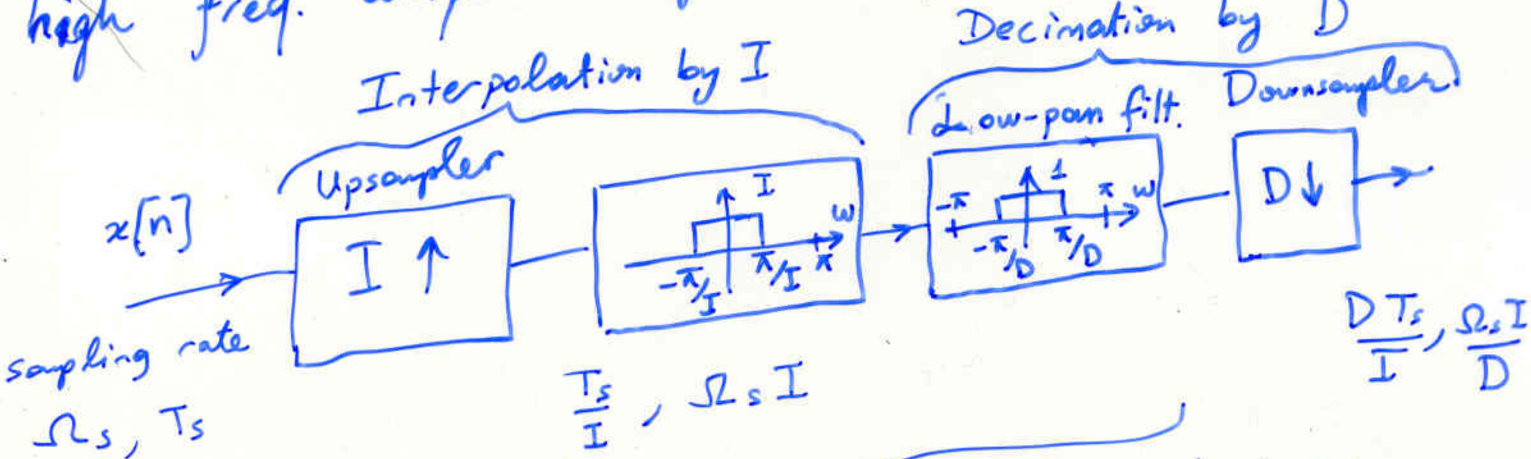


$x_d[n] = \bar{x}[Mn]$

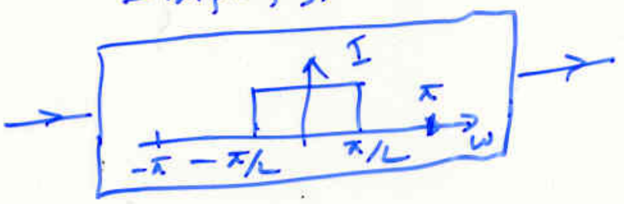
Downsample by M

Sampling rate conversion by a rational factor I/D:

* First interpolate then decimate because in decimation the low-pass filter removes the high freq. components of the signal.



Simplify, use a simple L.P.F.



$$L = \max(I, D).$$