$$f_2(n) = x(n) + K_1 x(n-1) + K_2[K_1 x(n-1) + x(n-2)]$$
$$= x(n) + K_1(1+K_2)x(n-1) + K_2 x(n-2)$$

The general form of lattice structure for m stage is given by'

$$f_0(n) = g_0(n) = x(n)$$
$$f_m(n) = f_{m-1}(n) + K_m g_{m-1}(n-1) \qquad m = 1, 2, \ldots, M-1$$
$$g_m(n) = K_m f_{m-1}(n) + g_{m-1}(n-1) \qquad m = 1, 2, \ldots, M-1$$

Conversion of lattice coefficients to direct-form filter coefficients. The direct-form FIR filter coefficients $\{\alpha_m(k)\}$ can be obtained from the lattice coefficients $\{K_i\}$ by using the following relations:

$$A_0(z) = B_0(z) = 1$$
$$A_m(z) = A_{m-1}(z) + K_m z^{-1} B_{m-1}(z) \qquad m = 1, 2, \ldots, M-1$$
$$B_m(z) = z^{-m} A_m(z^{-1}) \qquad m = 1, 2, \ldots, M-1$$

Conversion of direct-form FIR filter coefficients to lattice coefficients. Suppose that we are given the FIR coefficients for the direct-form realization or, equivalently, the polynomial $A_m(z)$, and we wish to determine the corresponding lattice filter parameters $\{K_i\}$. For the m-stage lattice we immediately obtain the parameter $K_m = \alpha_m(m)$. To obtain K_{m-1} we need the polynomials $A_{m-1}(z)$ since, in general, K_m is obtained from the polynomial $A_m(z)$ for $m = M-1, M-2, \ldots, 1$. Consequently, we need to compute the polynomials $A_m(z)$ starting from $m = M-1$

$$K_m = \alpha_m(m) \quad \alpha_{m-1}(0) = 1$$
$$\alpha_{m-1}(k) = \frac{\alpha_m(k) - K_m \beta_m(k)}{1 - K_m^2}$$
$$= \frac{\alpha_m(k) - \alpha_m(m)\alpha_m(m-k)}{1 - \alpha_m^2(m)} \qquad 1 \le k \le m-1$$

FIR Filter Design by Windowing

The FIR techniques are based purely in discrete-time, unlike the previous techniques for IIR design that were designed from the transformation of filters specified in continuous time.

Recall ideal LPF

$$H(e^{j\omega}) = \begin{cases} 1 & |\omega| < \omega_c \\ 0 & \text{else} \end{cases}$$

and

$$h_d[n] = \frac{\sin \omega_c n}{\pi n} \qquad \forall n$$

What happens if we just truncate $h_d[n]$ to a finite of samples?

$$h[n] = \begin{cases} h_d[n] & -M \le n \le M \\ 0 & \text{else} \end{cases}$$

It is equivalent to convolution in frequency domain of a sinc function and a rectangle function.

General case:

$$h[n] = \begin{cases} h_d[n] & 0 \le n \le M \\ 0 & \text{else} \end{cases}$$

or could multiply it by some "window" function

$$h[n] = h_d[n]w[n]$$

When the window is rectangle

$$w[n] = \begin{cases} 1 & 0 \le n \le M \\ 0 & \text{else} \end{cases}$$

then

$$H(e^{j\omega}) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H_d(e^{j\theta})W(e^{j(\omega-\theta)})d\theta$$

which is a periodic convolution between $H_d(e^{j\omega})$ and $W(e^{j\omega})$ and can be thought of as smearing.

Consider

$$\begin{aligned} W_M(e^{j\omega}) =& FT\{w_M[n]\} \\ =& \sum_{n=0}^{M} e^{-j\omega n} \\ =& \frac{1-e^{-j\omega(M+1)}}{1-e^{j\omega}} \\ =& e^{j\omega M/2}\frac{\sin[\omega(M+1)/2]]}{\sin(\omega/2)} \end{aligned}$$

Note width of mainlobe, frequency $2\pi/(M+1)$, $\omega = 2\pi\Omega$, frequency $1/(M+1)$ or signal period $M+1$, $\frac{M+1}{2}$, $\frac{M+1}{3}$, $\cdots$

Points:

1) to decrease mainlobe width, can increase M.
2) side lobe area doesn't decrease as M increases
$\Longrightarrow W(e^{j(\omega-\theta)})$ slides by ideal filter, we get Gibbs oscillations that doesn't decrease in amplitude as M increases. (if this ?? (manuscript p4) bad, this can cause aliasing in real systems).

Approach: taper off the window generally. (i.e., multiply by $w[n]$), to get other trade offs of mainlobe width, filter length M, and side lobe height.

Common Windows

Rectangular:

$$w[n] = \begin{cases} 1, & 0 \le n \le M \\ 0, & \text{else} \end{cases}$$

Bartlett (triangular):

$$w[n] = \begin{cases} 2n/M, & 0 \le n \le M/2 \\ 2-2n/M, & M/2 < n \le M \\ 0, & \text{else} \end{cases}$$

Hanning:

$$w[n] = \begin{cases} 0.5 - 0.5\cos(2\pi n/M), & 0 \le n \le M \\ 0, & \text{else} \end{cases}$$

Hamming:

$$w[n] = \begin{cases} 0.54 - 0.46\cos(2\pi n/M), & 0 \le n \le M \\ 0, & \text{else} \end{cases}$$

Blackman

$$w[n] = \begin{cases} 0.42 - 0.5\cos(2\pi n/M) + 0.08\cos(4\pi n/M), & 0 \le n \le M \\ 0, & \text{else} \end{cases}$$

Note the above windows are all "linear phase", i.e., shifted version of a symmetric window.

There exist many of these window functions (lots of research papers on this stuff in the 1960's and 1970's).

Properties of Different Windows

- show page 470-471, $M = 50$
- rectangular has narrowest main lobe, but peak side lobe is ?? (manuscript p5) -13dB down $\Longrightarrow$ aliasing
- Bartlett, helps side.
  side lobe width increases, down -25dB
  but main lobe width increases significantly (by about 2)
- Hanning, much lower sidelobe, they taper off to very small.
- Hamming, even lower sidelobe amplitude (-41dB)
- Blackman, even lower sidelobe, but wider mainlobe

Kaiser Window

Kaiser window is very useful in practice.

Zero-th order modified Bessel function of first kind.

$$w[n] = \begin{cases} I_0[\beta(1 - [(n-\alpha)/\alpha]^2)^{1/2}], & 0 \le n \le M \\ 0, & \text{else} \end{cases}$$

where $\alpha = M/2$, $I_0(\cdot)$ is Bessel function, β is a parameter that determines to some extent the "shape" of the filter.

M and β trade off sidelobe amplitude and mainlobe width.

Question: which one obviously controls mainlobe width? M.

β controls both width and tapering off (i.e., as β increases, width gets large but sidelobe amplitudes get smaller.)

Kaiser empirically found formula going from $\delta, \omega_s, \omega_p$ to M and β.

$$\begin{aligned} \Delta\omega &= \omega_s - \omega_p \\ A &= -20\log_{10}\delta \end{aligned}$$

$$\beta = \begin{cases} 0.1102(A - 8.7), & A > 50 \\ 0.5842(A-21)^{0.4} + 0.07886(A-21), & 21 \le A \le 50 \\ 0.0, & A < 21 \end{cases}$$

and

$$M = \frac{A-8}{2.285\Delta\omega}$$

This makes it very easy to do filter design very quickly.

Key: this gives the window $w[n]$, which is to be multiplied by the ideal impulse response $h_d[n]$ to get the actual filter $h[n] = w[n]h_d[n]$.

M does not change ripple error (to achieve δ) this is determined by the sidelobe amplitude (determined by β, or type of window).

Frequency sampling method:

The frequency sampling method allows us to design recursive and nonrecursive FIR filters for both standard frequency selective and filters with arbitrary frequency response.

A. No recursive frequency sampling filters :

The problem of FIR filter design is to find a finite–length impulse response h (n) that corresponds to desired frequency response. In this method h (n) can be determined by uniformly sampling, the desired frequency response H_D (ω) at the N points and finding its inverse DFT of the frequency samples as shown in the Figure 4.

$$h(n) = \frac{1}{N}\sum_{k=0}^{N-1} H(k) e^{j(\frac{2\pi}{N})nk}$$

where H (k), k = 0, 1, 2,……., N-1, are samples of the H_D (ω) .
For linear phase filters, with positive symmetrical impulse response, we can write

$$h(n) = \frac{1}{N}\left[\sum_{k=1}^{(N/2)-1} 2\left|H(k)\right|\cos\left[\frac{2\pi k(n-\alpha)}{N}\right] + H(0)\right]$$

where α = (N-1)/2. For N odd, the upper limit in the summation is (N - 1)/2, to obtain a good approximation to the desired frequency

response, we must take a sufficient number of the frequency samples

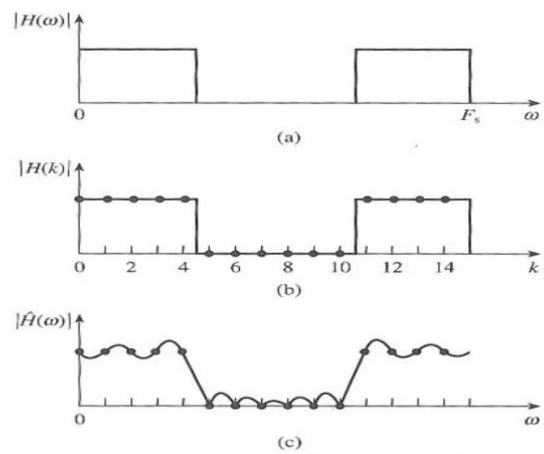

Figure2 (a) Frequency response of an ideal lowpass filter. (b) Samples of the ideal lowpass filter. (c) Frequency response of lowpass filter derived from the frequency samples of (b).

Types 1 and 2 frequency sampling filters :

frequency sampling filters are based on specification of a set of samples of the desired frequency response at N uniformly spaced points around the unit circle. The set of frequencies that have been used until this point is determined by the relation.

$$f_k = \frac{k}{N} F_S, \quad k = 0,1,....,N-1$$

Corresponding to the N frequencies at which an N-point DFT is evaluated. There is a second set of uniformly spaced frequencies for which a frequency sampling structure can conveniently be obtained. This set of frequencies determined by the relation [6].

$$f_k = \frac{(k+1/2)}{N} F_S, \quad k = 0,1,....,N-1$$

Figure (4) shows exactly where the frequency sampling points are located for the two sets of frequencies the first set of frequencies in

Recursive frequency sampling filter :

In recursive frequency sampling method the DFT samples H (k) for an FIR sequence can be regarded as samples of the filters z–transform, evaluated at N points equally spaced around the unit circle.[8]

$$H(k) = H(z)\Big|_{z=e^{j(2\pi/N)k}}$$

thus the z–transform of an FIR filter can easily be expressed in terms of its DFT coefficients,

$$H(z) = \sum_{n=0}^{N-1} h(n) z^{-n} = \sum_{n=0}^{N-1} \left[\frac{1}{N} \sum_{k=0}^{N-1} H(k) e^{j(2\pi/N)nk} \right] z^{-n}$$

$$= \sum_{k=0}^{N-1} \frac{H(k)}{N} \sum_{n=0}^{N-1} \left[e^{j(2\pi/N)k} z^{-1} \right]^n$$

$$= \sum_{k=0}^{N-1} \frac{H(k)}{N} \frac{(1 - e^{j2\pi k} z^{N})}{(1 - e^{j(2\pi/N)k} z^{-1})}$$

by butting $e^{j2\pi k} = 1$, Equation (13) reduces to

$$H(z) = \frac{(1 - z^{-N})}{N} \sum_{k=0}^{N-1} \frac{H(k)}{(1 - z^{-1} e^{j(2\pi/N)k})}$$

This is the desired result