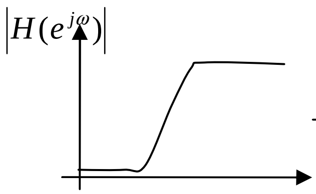
Analysis and Design of Digital Filter

I. What is filter design?



Filter specification in terms of amplitude response and error tolerance

$$H(z) = \frac{\sum_{i=0}^{N} L_i z^{-i}}{1 + \sum_{i=}^{M} K_i z^{-i}}$$

Transfer function

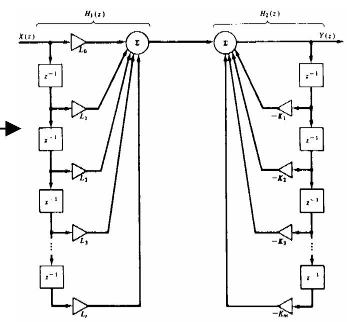


FIGURE 9-1. Direct Form I realization.

II. Two major categories

1. IIR – typically based on transforming a continuous-time analog filter (such as Butterworth) into discrete-time

   Advantages:
   - Decades of experiences in designing analog filters
   - Typical less complex (fewer registers, arithmetic units) than FIR in realizing the design filter spectrum

2. FIR – entirely discrete-time domain method

   Advantages:
   - Advantages of any FIR filter : always stable, linear phase
   - "Optimal" CAD method (not covered in this class)

III. IIR Filter Design

General procedure:

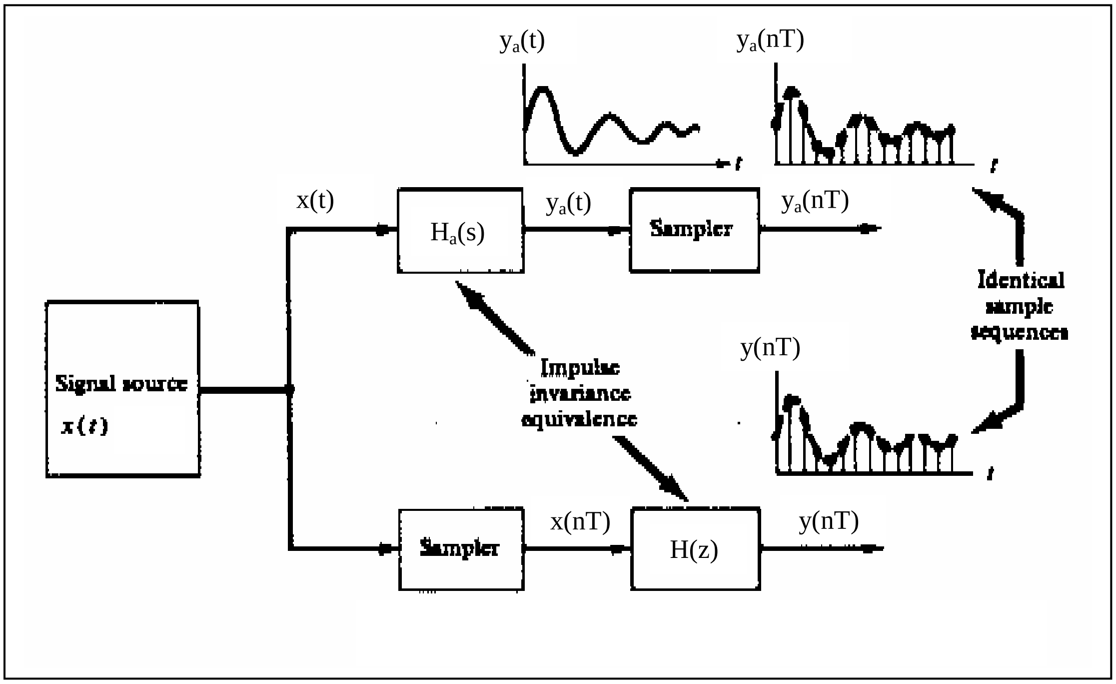
1. Design an analog filter $H_a(s)$ that satisfies the specification.
2. Map the analog filter $H_a(s)$ into a discrete-time filter $H_d(z)$

Focus on step 2. There are two approaches:

1. *Invariance Design*
   - Simple to understand
   - Requires high sampling rate to mitigate aliasing

2. *Bilinear Mapping Design – Frequency warping*
   - No aliasing
   - Frequency distortion

III.1 Invariant Design



Goal: for a GIVEN x(t), we want the output of our digital filter y(nT) to be identical to the sampled version $y_a(nT)$ of the continuous-time filter output.

If x(t) is the delta function, we effectively have

$$h(nT) = h_a(nT)$$

This is called IMPULSE-INVARIANCE design.

Let's see an example to illustrate all the steps.

Example: Given $H_a(s) = \dfrac{0.5(s+4)}{(s+1)(s+2)}$

Find digital filter H(z) by impulse-invariance.

Step 1: Compute the response of the analog filter $y_a(t)$

Since $x_a(t) = \delta(t) \Rightarrow X_a(s)=1$, thus

$$\begin{aligned} y_a(t) &= L^{-1}[H_a(s)X_a(s)] \\ &= L^{-1}[H_a(s)] \\ &= L^{-1}\left[\frac{0.5(s+4)}{(s+1)(s+2)}\right] \\ &= L^{-1}\left[\frac{1.5}{s+1} - \frac{1}{s+2}\right] = 1.5e^{-t} - e^{-2t} \end{aligned}$$

Step 2: Sample $y_a(t)$ to get $y_a(nT)$

$$y_a(nT) = 1.5e^{-nT} - e^{-2nT}$$

Step 3: Perform Z-transform on $y_a(nT)$

$$Y(z) = Z[y_a(nT)] = \frac{1.5}{1-e^{-T}z^{-1}} - \frac{1}{1-e^{-2T}z^{-1}}$$

Step 4: Perform Z-transform on $x_a(nT)$

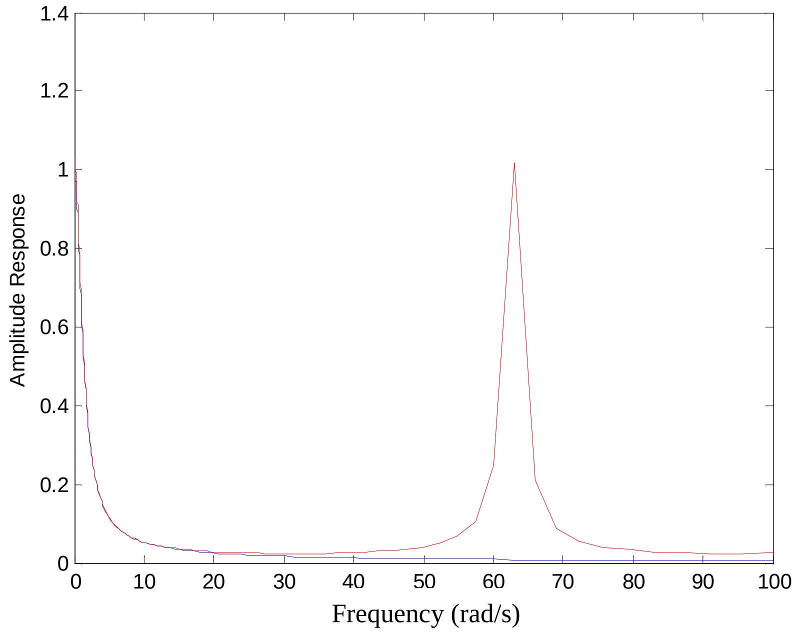
$$X(z) = Z[x_a(nT)] = 1$$

Step 5: Find $H_d(z) = Y(z)/X(z)$

$$H(z) = \frac{1.5}{1-e^{-T}z^{-1}} - \frac{1}{1-e^{-2T}z^{-1}}$$

Step 6: Finally choose a sampling period T and obtain the numerical representation of the digital filter.

```
>> % Impulse-Invariance Design
>> num  = 0.5 * [ 1 4];
>> den  = conv([1 1], [1 2]);
>> Fs   = 10;                        % Sampling freq = 10Hz
>> [bz,az] = impinvar(num,den,Fs);
>>
>> % Compare Analog and Digital Filter
>> [ha,w] = freqs(num,den);
>> f = w/(2*pi);                   % Convert to Normal Freq.
>> hd = freqz(bz,az,f,Fs);
>> plot(w,abs(ha),'b',w,abs(hd),'r');  % Blue:Analog,
Red:Digital
```



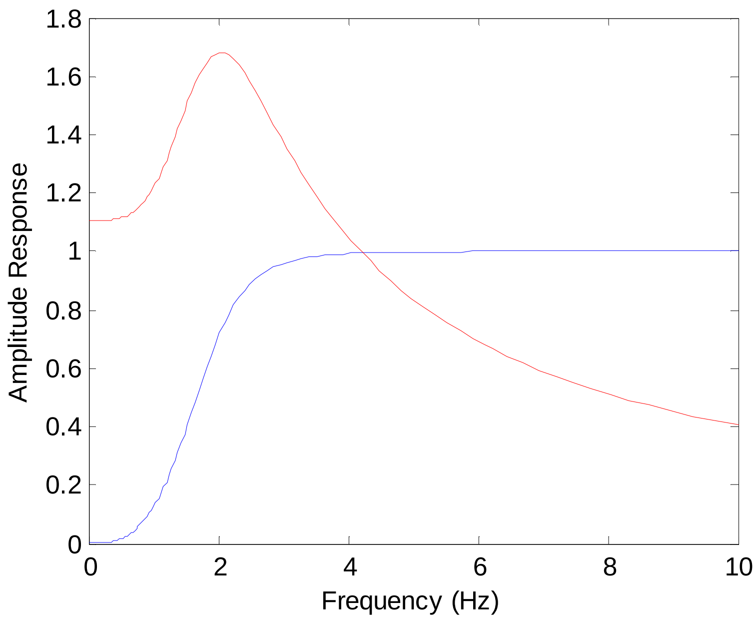
What happen around 63 rad/s?

Ans: The FT of the discrete filter is periodic with period $2\pi f_s = 2\pi(10) \approx 63$ rad/s.

If your client cannot stomach the divergence starting at 30Hz, what should you do?

Ans: Increase the sampling rate.

What about if we want to design a highpass filter $H_{HP}(s) = \dfrac{s^3}{(s+2)(s^2+2s+4)}$?



Important: We cannot use invariance method to design any non band-limited filter such as high-pass or band-stop. To design these filters, we need to understand the concept of "frequency transformation".

III.2 Bilinear Transformation

The basic idea of frequency transformation is straightforward:

1. Start with a low-pass prototype filter $H_p(j\omega)$.
2. Design a frequency transform $\omega' = f(\omega)$ such that …
3. $H_p(j\omega')$ is your desired filter!

Frequency transformation is useful in both analog filter design and analog to discrete conversion.

Example: Convert a low-pass analog filter to high-pass analog filter

Let's say you have a low-pass analog filter with cutoff frequency (-3dB frequency) at 1 rad/s as follows:

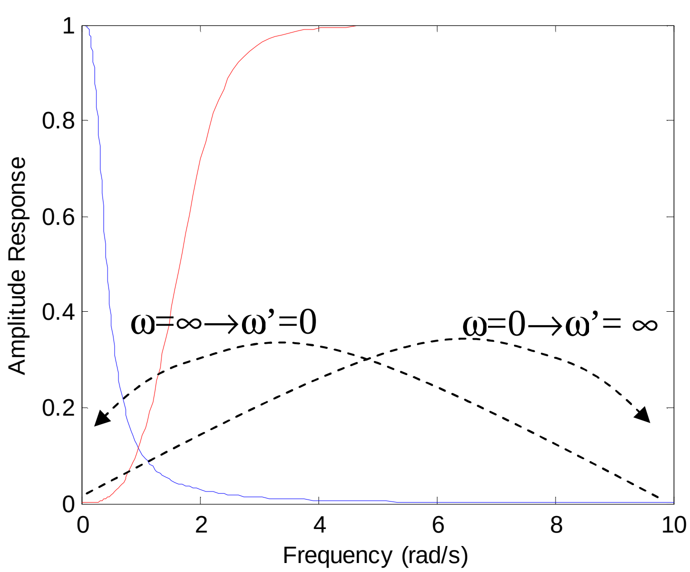
$$H(s) = \frac{1}{(s+1)(s^2+s+1)}$$

To design a HIGH-PASS filter with cutoff frequency at ω_p, we use the following transformation:

$$\omega' = \frac{\omega_p}{\omega} \text{ or equivalently } s' = \frac{\omega_p}{s}$$

Let $\omega_p = 2$,

$$H_{HP}(s) = \frac{1}{\left(\frac{2}{s}+1\right)\left(\left(\frac{2}{s}\right)^2 + \frac{2}{s} + 1\right)} = \frac{s^3}{(s+2)(s^2+2s+4)}$$



Question: How do we use frequency transformation to map analog filter to digital?

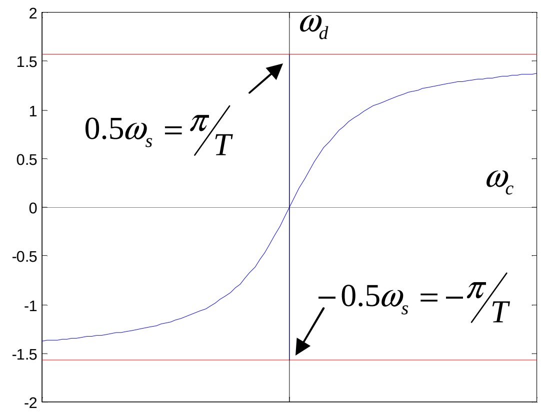
CTFT extends from negative infinity to positive infinity, while DTFT is periodic with period equal to the sampling frequency ($2\pi/T$). To ELIMINATE ALIASING, we need to come up with a frequency transform f such that

f maps analog frequency $\omega_c \in [-\infty,\infty]$ to $\omega_d \in [-\pi/T, \pi/T]$

One such example of f is the arctan:

$$\omega_d = \tfrac{2}{T}\arctan(\omega_c)$$

In the following example T = 2:



When ω_c is small ($-1 \le \omega_c \le 1$), the distortion is relatively small and we have

$$\omega_d \approx \frac{2}{T}\omega_c$$

Beyond that, the transformation compresses ω_c causing a fair amount of distortion.

If we are designing a low-pass filter with passband $-C \le \omega_c \le C$, we can scale ω_c first by C and use the transform:

$$\omega_d = \tfrac{2}{T}\arctan\left(\omega_c/C\right)$$

In most applications, the analog filter is specified in terms of the Laplace variable $s=j\omega_c$ and the target discrete filter needs to be expressed in Z-transform variable $z=e^{j\omega d}$. Thus, it will be nice if we can derive a transform directly from s to z.

Bilinear Transformation: $$s = C\frac{1-z^{-1}}{1+z^{-1}}$$

Proof:

$$
\begin{aligned}
j\omega_c &= C\frac{1-e^{-j\omega_d T}}{1+e^{-j\omega_d T}} \\
&= C\frac{1-e^{-j\omega_d T/2}e^{-j\omega_d T/2}}{1+e^{-j\omega_d T/2}e^{-j\omega_d T/2}} \\
&= C\frac{e^{j\omega_d T/2}-e^{-j\omega_d T/2}}{e^{j\omega_d T/2}+e^{-j\omega_d T/2}} \\
&= C\frac{2j\sin(\omega_d T/2)}{2\cos(\omega_d T/2)} = Cj\tan(\omega_d T/2) \\
\Leftrightarrow \omega_c &= C\tan(\omega_d T/2) \\
\Leftrightarrow \omega_d &= 2/T\arctan(\omega_c/C)
\end{aligned}
$$

To determine C, we typically assume there is a "fixed-point" or "anchor" frequency ω_r which maps to itself:

$$\omega_r = \frac{2}{T}\arctan\left(\frac{\omega_r}{C}\right) \Rightarrow C = \frac{\omega_r}{\tan\left(\frac{\omega_r T}{2}\right)}$$

Example 9-7 $H_a(s) = \dfrac{\omega_c^{\,2}}{s^2+\sqrt{2}\omega_c s+\omega_c^{\,2}}$ (2nd order Butterworth filter) with $\omega_c=1000\pi$ as the anchor frequency and $\omega_s=4000\pi$.

Applying bilinear transformation:

$$
\begin{aligned}
H_d(z) &= \frac{\omega_c^{\,2}}{C^2\frac{(1-z^{-1})^2}{(1+z^{-1})^2}+\sqrt{2}\omega_c\frac{1-z^{-1}}{1+z^{-1}}+\omega_c^{\,2}} \\
&= \frac{\omega_c^{\,2}(1+z^{-1})^2}{C^2(1-z^{-1})^2+\sqrt{2}\omega_c(1-z^{-2})+\omega_c^{\,2}(1+z^{-1})^2}
\end{aligned}
$$

Solving C:

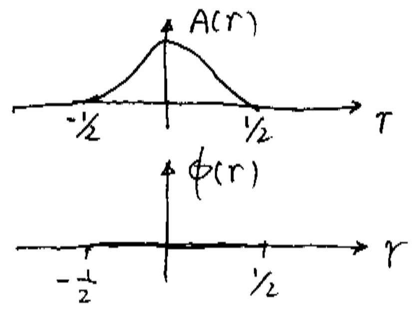
$$C = \frac{\omega_c}{\tan\left(\frac{\omega_c T}{2}\right)} = 229168.76$$

$$\Rightarrow \quad H_d(z) = \frac{0.292893+0.585786z^{-1}+0.292893z^{-2}}{1+0.171573z^{-2}}$$

IV FIR Digital Filter

IV.1 Design Procedure:

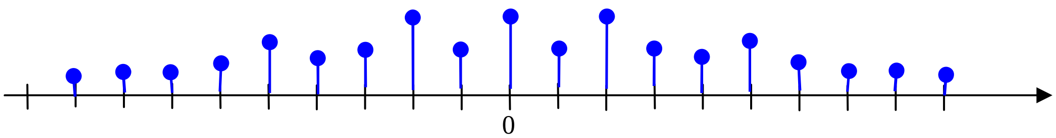
1. Start with desired amplitude response with zero phase.



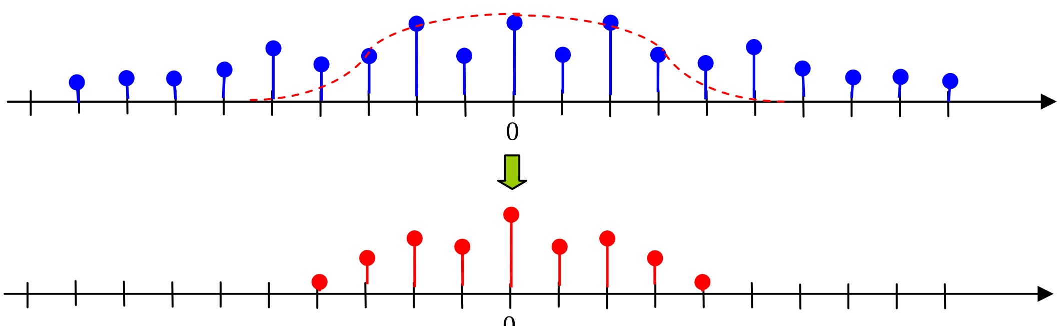
2. Apply inverse DTFT to recover acausal, symmetric and possibility infinitely-long impulse response $h_d(nT)$

   DTFT Formula: $$x(nT) = \frac{T}{2\pi}\int_{-\frac{\pi}{T}}^{\frac{\pi}{T}} X(e^{j\omega T})e^{j\omega nT}d\omega$$

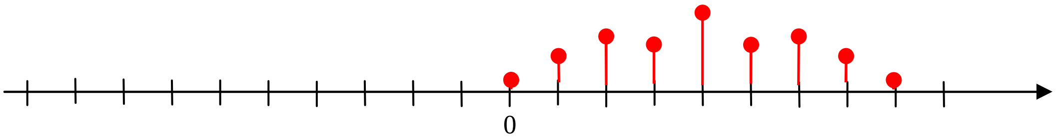
   or in normalized frequency: $$x(nT) = \int_{-0.5}^{0.5} X(e^{j2\pi r})e^{jn2\pi r}dr$$



3. Truncate $h_d(nT)$ to finite extent by multiplying it by an appropriate window function and form $h_{nc}(nT)$.



4. Delay $h_{nc}(nT)$ to make it casual.



Example: Design a FIR filter that approximates $H(r) = \frac{1}{2}(1+\cos 2\pi r)$

Solution:

(1) Inverse DTFT (explicit calculation)

$$H(r) = \frac{1}{2}(1+\cos 2\pi r) = \sum_{n=-\infty}^{\infty} h_d(nT)e^{-jn2\pi r}$$

Notice n ranges from -∞ to ∞

Compute $h_d(nT) = \frac{1}{2}\int_{-1/2}^{1/2}(1+\cos 2\pi r)e^{jn2\pi r}dr$

For $n = 0$, $h_d(0) = \frac{1}{2}\int_{-1/2}^{1/2}(1+\cos 2\pi r)dr = \frac{1}{2}\int_{-1/2}^{1/2}dr + \frac{1}{2}\int_{-1/2}^{1/2}\cos 2\pi r dr = \frac{1}{2}$

For $n \neq 0$,

$$h_d(nT) = \frac{1}{2}\int_{-1/2}^{1/2}e^{jn2\pi r}dr + \frac{1}{2}\int_{-1/2}^{1/2}\frac{e^{j2\pi r}+e^{-j2\pi r}}{2}e^{j2\pi nr}dr$$

$$= \frac{1}{4}\int_{-1/2}^{1/2}e^{j2\pi(n+1)r}dr + \frac{1}{4}\int_{-1/2}^{1/2}e^{j2\pi(n-1)r}dr = \frac{1}{4} \textit{ for } n = \pm 1 \textit{ and } 0 \textit{ otherwise}$$

$$\Rightarrow \begin{cases} H(r) = \sum_{n=-\infty}^{\infty} h_d(nT)e^{-j2\pi nr} = \frac{1}{4}e^{-j2\pi r} + \frac{1}{2} + \frac{1}{4}e^{-j2\pi r} \\ h_d(0) = \frac{1}{2}, \quad h_d(\pm T) = \frac{1}{4}, \quad h_d(nT) = 0, \quad |n| \geq 2 \end{cases}$$

The impulse response is finite duration. Thus there is no need to truncate.

(3) Make it acasual by delaying one sample.

$$H(z) = \frac{1}{4} + \frac{1}{2}z^{-1} + \frac{1}{4}z^{-2}$$

IV.2 Effect of windowing

2M+1 terms

Truncation => $H_{nc}(z) = \sum_{n=-M}^{M} h_d(nT)z^{-n} = \sum_{n=-\infty}^{\infty} w_r(n)h_d(nT)z^{-n}$

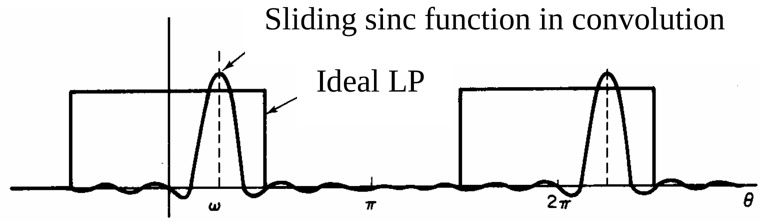
where $w_r(n) = \begin{cases} 1 & |n| \leq M \\ 0 & |n| > M \end{cases}$

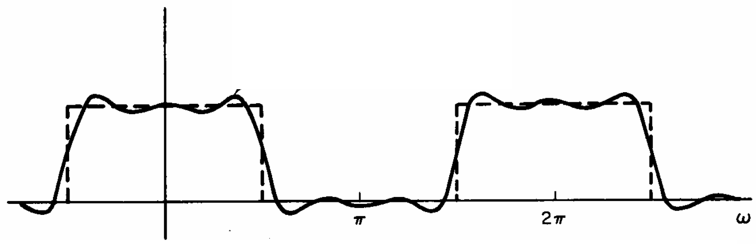
Truncation = time multiplication with rectangular window function

⇔ Convolving in frequency domain with a sinc function

$$w_r(e^{j2\pi r}) = \sum_{n=-M}^{M} e^{-j2\pi n r} = \frac{\sin \pi(2M+1)r}{\sin \pi r}$$ - Sinc function in frequency domain

Suppose the original analog filter H(r) is an ideal low-pass filter, windowing in time domain corresponds to convolution in frequency with a sinc function.





Frequency Spectrum of the resulting truncated sequence

Spectrum of the rectangular window that retains 15 samples (M=7):

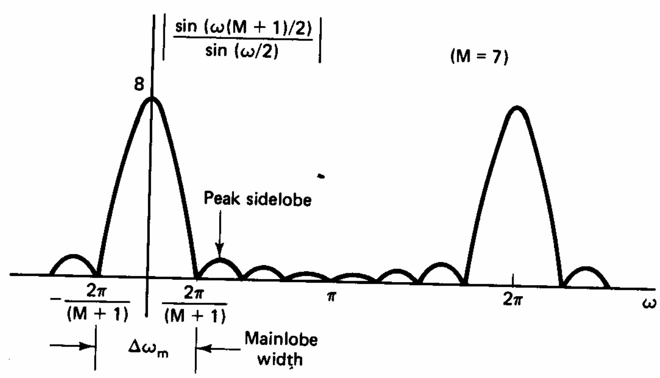


Figure 7.28 Magnitude of the Fourier transform of a rectangular window ($M = 7$).

The frequency spectrum of a good truncation window should resemble as much as possible to a delta function. Typically, it is measured based on two criteria:

1. Narrow Main Lobe
   - for rectangular window : main lobe width = $2\pi/(M+1)$
   - a narrow main lobe produce better frequency transition

2. Small Side Lobes
   - measured by the highest peaks
   - caused by the sharp cutoff of the rectangular window

By tapering the window smoothly to zero at each end, the height of the sidelobes can be diminished; however, this is achieved at the expense of a wider mainlobe and thus a wider transition at the discontinuity.

Some commonly used windows include:

Bartlett (triangular): $$w(n) = \begin{cases} 2n/M & 0 \le n \le M/2 \\ 2 - 2n/M & M/2 \le n \le M \\ 0 & otherwise \end{cases}$$

Hanning: $$w(n) = \begin{cases} 0.5 - 0.5\cos(2\pi n/M) & 0 \le n \le M \\ 0 & otherwise \end{cases}$$

Hamming : $$w_h(n) = \begin{cases} 0.54 - 0.46\cos\dfrac{n\pi}{M} & 0 \le n \le M \\ 0 & otherwise \end{cases}$$

Blackman: $$w(n) = \begin{cases} 0.42 - 0.5\cos(2\pi n/M) + 0.08\cos(4\pi n/M) & 0 \le n \le M \\ 0 & otherwise \end{cases}$$

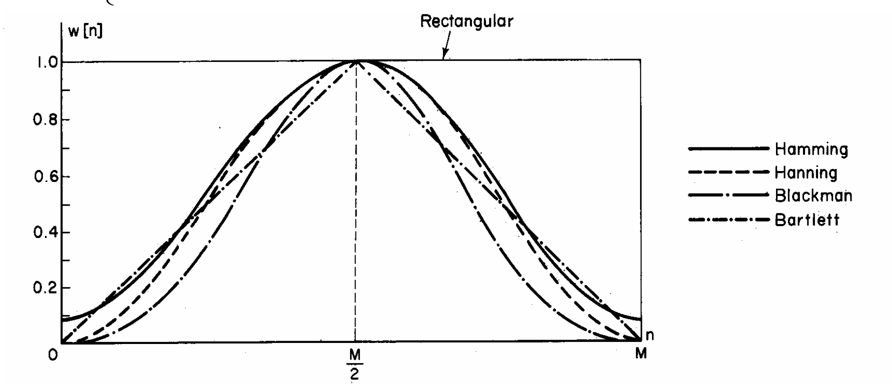
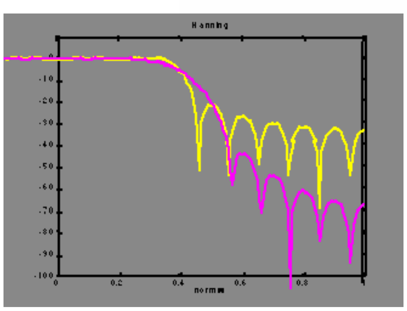
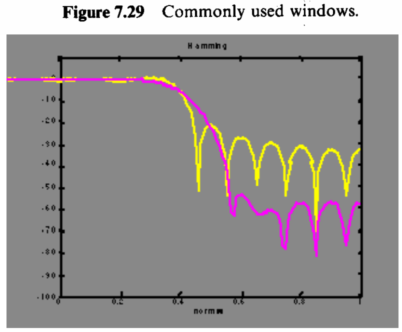
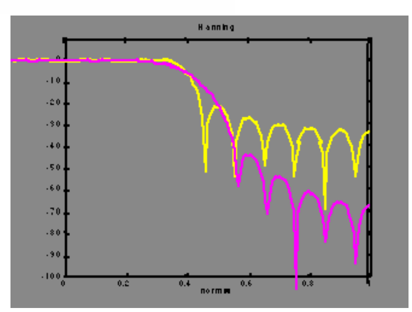


Figure 7.29 Commonly used windows.


Hanning vs. Rectangular


Hamming vs. Rectangular


Blackman vs. Rectangular

Example: Design a 17-tap low-pass FIR digital filter with Hamming window to approximate

$$H(r) = \begin{cases} 1 & |r| \le 0.15 \\ 0 & 0.15 < |r| \le 0.5 \end{cases}$$

Step 1: Inverse Fourier Transform

$$h_d(nT) = \int_{-0.15}^{0.15} e^{j2\pi n r} dr = \frac{1}{j2\pi n}(e^{j0.3\pi} - e^{-j0.3\pi}) = \frac{1}{\pi n}\sin 0.3\pi n$$

$$h_d(0T) = \lim_{\pi n \to 0} \frac{d(\sin 0.3\pi n)/d(\pi n)}{d(\pi n)/d(\pi n)} = \lim_{\pi n \to 0} \frac{0.3\cos 0.3\pi n}{1} = 0.3$$

$$\Rightarrow \begin{cases} h_d(0) = 0.3 & \\ h_d(nT) = \dfrac{\sin 0.3\pi n}{\pi n} & n \ne 0 \end{cases}$$

Step 2 & 3: Multiply by 17-tap Hamming window (M=8) and make it acausal.

$$H_{NC}(z) = \sum_{n=-8}^{8} h_d(nT) w_h(n) z^{-n}, \quad H_C = z^{-8} H_{NC}(z)$$

TABLE 9-2
Filter Weights for FIR Low-Pass Filter ($f_c = 0.15 f_s$)

n	Unit Pulse Response with Rectangular Window, $h(nT)$	Hamming Window Function, $w_h(nT)$	Unit Pulse Response with Hamming Window, $w_h(nT)h(nT)$
−8	0.037841	0.08	0.003027
−7	0.014052	0.115015	0.001616
−6	−0.031183	0.214731	0.006696
−5	−0.063662	0.363966	−0.023171
−4	−0.046774	0.54	−0.025258
−3	0.032788	0.716034	0.023477
−2	0.151365	0.865269	0.130972
1	0.257518	0.964985	0.248501
0	0.3	1.0	0.3
1	0.257518	0.964985	0.248501
2	0.151365	0.865269	0.130972
3	0.032788	0.716034	0.023477
4	−0.046774	0.54	−0.025258
5	−0.063662	0.363966	−0.023171
6	−0.031183	0.214731	−0.006696
7	0.014052	0.115015	0.001616
8	0.037841	0.08	0.003027

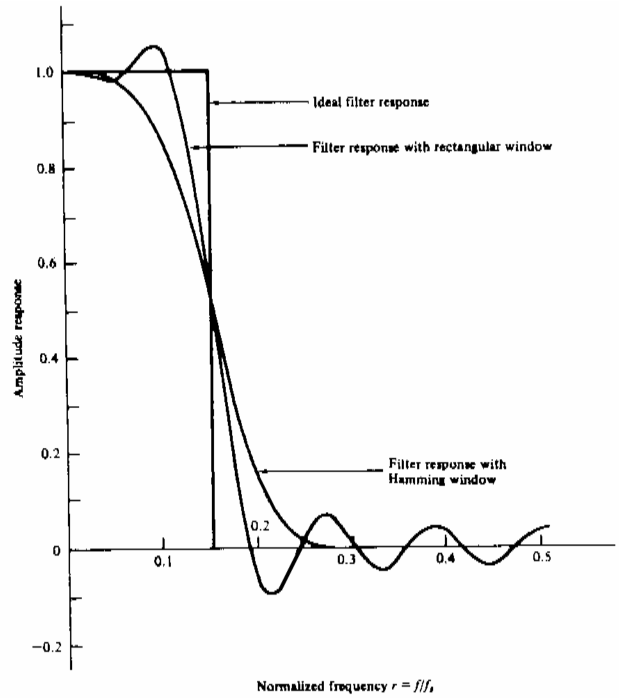


FIGURE 9-31. Amplitude response of digital low-pass filter. (The negative portions are shown negative for convenience.)