ECE 410 Course Notes

DIGITAL SIGNAL PROCESSING

David C. Munson, Jr.

Electrical Engineering and Computer Science Department
University of Michigan

Spring 2005

UNIVERSITY OF ILLINOIS

ECE
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 1

D. Munson

Overview of Digital Signal Processing

Notation:

$x_a(t)$ ~ analog or continuous signal

$x(n)$ or x_n ~ sequence

1) Fourier transform of $x_a(t)$:

$$X_a(\Omega) \triangleq \int_{-\infty}^{\infty} x_a(t) e^{-j\Omega t}\, dt$$

Inverse:

$$x_a(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X_a(\Omega) e^{j\Omega t}\, d\Omega$$

Say $x_a(t)$ is **bandlimited** to B rad/sec if

$$X_a(\Omega) = 0, \quad |\Omega| > B$$

2) Sampling Theorem:

$$x_a(t) \longrightarrow \underset{T}{/} \longrightarrow \{x_a(nT)\}_{n=-\infty}^{\infty}$$

Suppose $x_a(t)$ is BL to $B = 2\pi F$ rad/sec = F Hz. Twice the highest frequency in $x_a(t)$, i.e., 2F, is referred to as the <u>Nyquist frequency</u>. If we sample above the Nyquist frequency, i.e.,

$$\frac{1}{T} > 2F$$

then it is possible to exactly recover $x_a(t)$ from its samples. Specifically

$$x_a(t) = \sum_{n=-\infty}^{\infty} x_a(nT) \operatorname{sinc} \frac{\pi}{T}(t - nT)$$

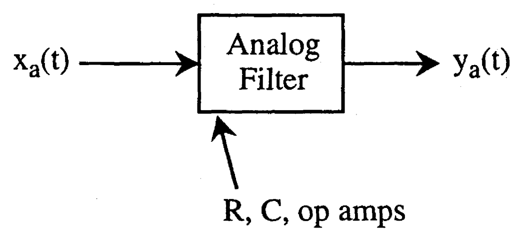
where $\operatorname{sinc}(t) = \dfrac{\sin(t)}{t}$

Selecting $\frac{1}{T} = 2F$ is referred to as sampling at the Nyquist rate. Choosing $\frac{1}{T} > 2F$ is referred to as sampling above the Nyquist rate. In practice, we sample above the Nyquist rate so that the samples $\{x_a(nT)\}_{n=-\infty}^{\infty}$ contain all of the information present in $x_a(t)$.

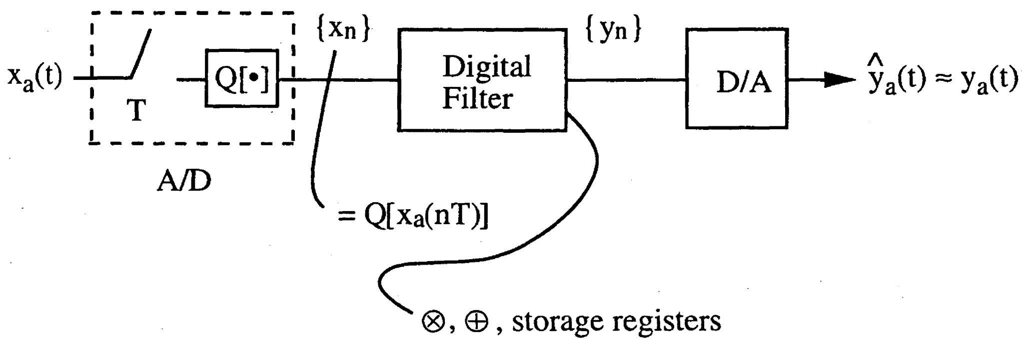
Question: Instead of processing $x_a(t)$ directly, can we process its samples?

3) Analog versus Digital Filtering

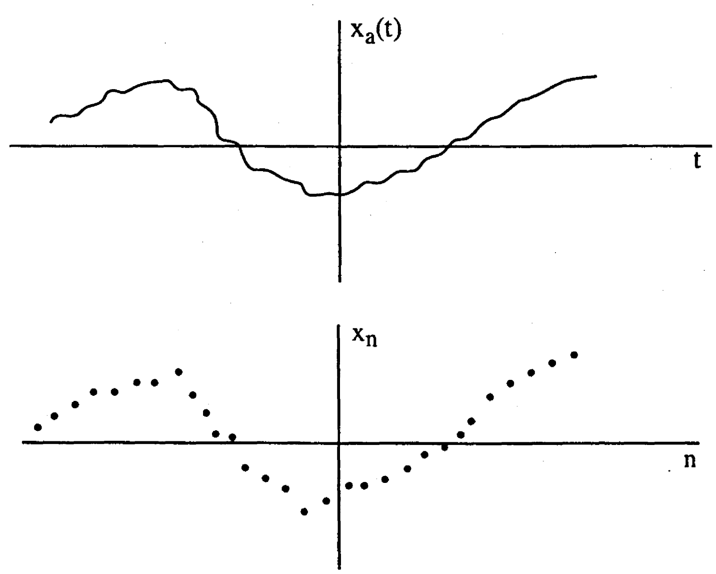
Analog:

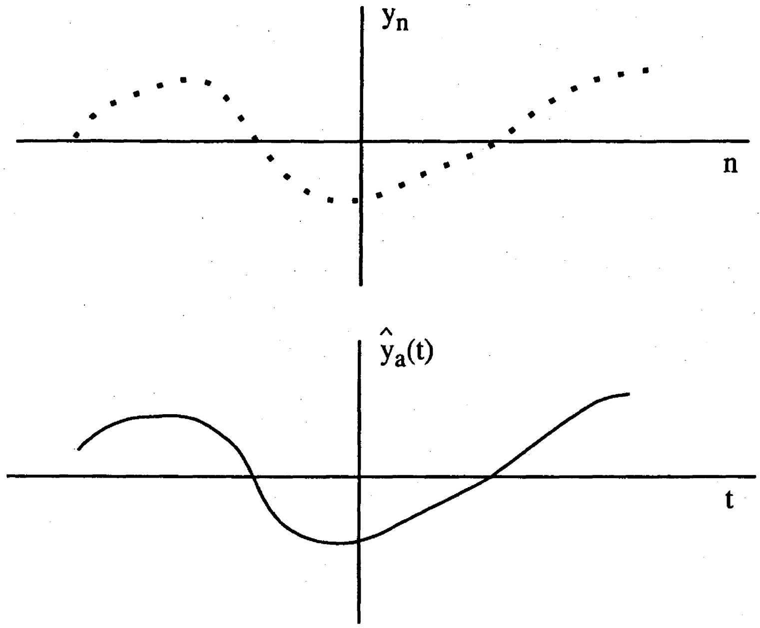


Digital:



Example signals:





Digital filter implements a difference equation. For example, might have

$$y_n = a_0\, x_n + a_1\, x_{n-1} + a_2\, x_{n-2} + \ldots + a_N\, x_{n-N} \qquad \text{(nonrecursive)}$$

or

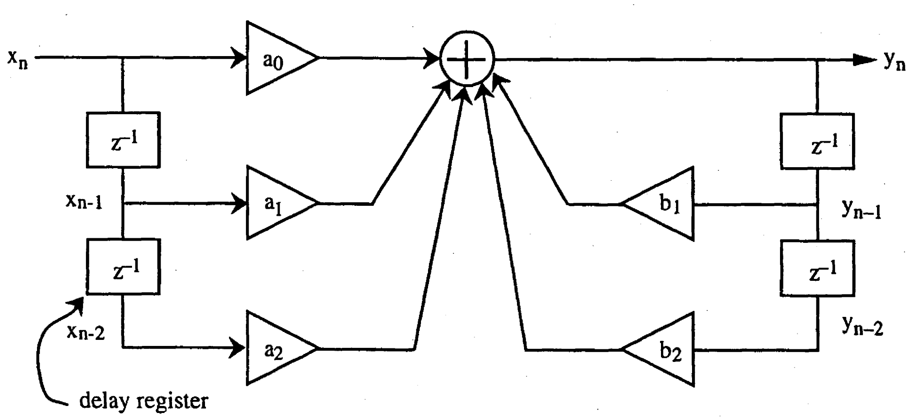
$$y_n = b_1\, y_{n-1} + b_2\, y_{n-2} + \ldots + b_M\, y_{n-M} + a_0\, x_n + \ldots + a_N\, x_{n-N} \qquad \text{(recursive)}$$

Can choose digital filter coefficients so that $\hat{y}_a$ is as close as you like to y_a.

Digital filter can be implemented using a computer, DSP microprocessor, or specialized hardware.

D/A essentially produces an analog signal that passes through the samples $\{y_n\}$.

4) Simple, 2nd-order digital filter:



$$y_n = a_0\, x_n + a_1\, x_{n-1} + a_2\, x_{n-2} + b_1\, y_{n-1} + b_2\, y_{n-2} \qquad \text{(recursive)}$$

5) Example Filtering Problem

Suppose

$$x_a(t) = \underbrace{s_a(t)}_{\text{desired signal}} + \underbrace{A \cos(2\pi \bullet 60 \bullet t)}_{\text{60 Hz noise}}$$

To remove noise could use A/D, digital filter, D/A with coefficients in digital filter chosen to place a notch in the frequency response at 60 Hz.

6) Programmable DSPs:

TI-TMS 320XX
Lucent
Motorola 56000
NEC 7720
AT&T

7) More sophisticated uses of DSP:

a) Telecommunications:
Modulation
Coding
Echo Cancellation
Equalization

b) Speech processing:
Analysis/Synthesis for low bit-rate transmission
Speech Recognition

c) Image processing:
Coding for video conferencing, video phone, fax, HDTV, and image archiving; noise removal; deblurring; object recognition.

d) Consumer electronics:
CD player, Digital Satellite TV

e) Medical imaging:
Computer Tomography (CT)
Magnetic Resonance Imaging (MRI)

8) Reasons Digital Often Preferred

a) More versatile – complicated processing, time varying and adaptive filtering, nonlinear processing, multidimensional (especially image processing)

b) Guaranteed accuracy – determined by register lengths, not nonideal R, L, C, op amps

c) Digital <u>often</u> smaller, cheaper, lower power

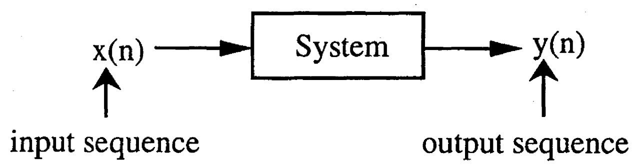
Note: Due to speed limitation of A/D's and computers, digital signal processing was initially employed (during 1960s) in application areas having low-bandwidth signals, such as speech (BW $\approx$ 3 KHz). The practical frequency range for digital processing has increased vastly over the years and continues to climb. As an example, digital signal processing is now commonly employed in radar, where sampling rates may be tens or even hundreds of megahertz.

ECE
University of Illinois

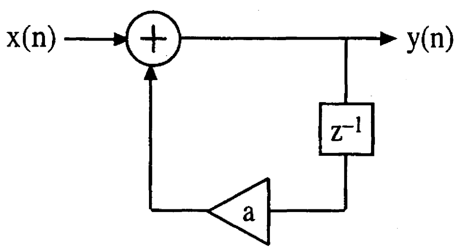
DIGITAL SIGNAL PROCESSING
Lecture 2

D. Munson

Discrete-Time Linear Systems



Ex.



$$y(n) = x(n) + a\,y(n-1)$$

Def. The <u>state</u> of a system at time m is a minimal set of variables which together with x(n), $n \geq m$ uniquely determine y(n), $n \geq m$.

Ex. In above first-order filter can choose state to be the value stored in delay register. State at "time" n is y(n–1).

Linearity

It is very important to be able to distinguish between linear and nonlinear systems, because some of the mathematical tools we will develop apply only to linear systems. The series of definitions below culminates in the definition of linearity.

Def. A system satisfies the <u>decomposition property</u> if its output can be written as

$$y(n) = y_x(n) + y_s(n)$$

$y_s(n)$: response due to initial state or condition (with zero input)

$y_x(n)$: response due to input (with zero IC's)

$y_x(n)$ is called the <u>zero-state response</u>.

$y_s(n)$ is called the <u>zero-input response</u>.

Zero-State Linearity

Suppose IC's = 0 (start at zero state)

a) Property of homogeneity;

If $x(n) \rightarrow y(n)$ then $a \cdot x(n) \rightarrow a \cdot y(n)$

b) Property of additivity:

If $x_1(n) \rightarrow y_1(n)$ and $x_2(n) \rightarrow y_2(n)$

then $x_1(n) + x_2(n) \rightarrow y_1(n) + y_2(n)$

Can combine homogeneity and additivity into single property of superposition.

If $x_1(n) \rightarrow y_1(n)$ and $x_2(n) \rightarrow y_2(n)$

then $a\, x_1(n) + b\, x_2(n) \rightarrow a\, y_1(n) + b\, y_2(n)$

Def. A system is zero-state linear (linear with respect to the input) if for zero IC's, the property of superposition holds.

Zero-Input Linearity

Def. A system is zero-input linear (linear with respect to IC's) if for zero input, the property of superposition holds with respect to the response $y_s(n)$ due to the IC's.

Def. A system is **linear** if it satisfies the decomposition property, is zero-state linear, and is zero-input linear.

In ECE 310 we will concentrate on zero-state linearity (linearity with respect to the input) and will often just call this linearity.

Example 1

Determine whether the system described by $y_n = |x_n|$ is linear or nonlinear.

This system satisfies neither homogeneity nor additivity and is therefore nonlinear. To prove the failure of homogeneity note that $x_1(n) = 1 \; \forall \; n$ and $x_2(n) = -1 \; \forall \; n$ produce the same output and yet $x_2(n) = -x_1(n)$. Similarly additivity fails because $x_1(n) + x_2(n)$ does not produce the sum of the outputs due to $x_1(n)$ and $x_2(n)$ acting individually.

Example 2

Consider a system described by

$$y(n) = \frac{[x(n-4)]^2}{x(n)}$$

Is this system linear or nonlinear?

Check homogeneity:

$$a \bullet x(n) \quad \rightarrow \quad \frac{[a\,x(n-4)]^2}{a\,x(n)} = a\,\frac{[x(n-4)]^2}{x(n)}$$

$$= a\,y(n) \quad ✓$$

So, this system does satisfy the property of homogeneity.

But, it looks like additivity will fail. $\Rightarrow$ Nonlinear. Find an $x_1(n)$ and $x_2(n)$ to demonstrate this:

$$\text{Let } x_1(n) = 1 \quad \forall n$$

$$\Rightarrow y_1(n) = \frac{(1)^2}{1} = 1 \quad \forall n$$

$$\text{Let } x_2(n) = \left(\frac{1}{2}\right)^n \quad \forall n$$

$$\Rightarrow y_2(n) = \frac{\left[\left(\frac{1}{2}\right)^{n-4}\right]^2}{\left(\frac{1}{2}\right)^n} = \frac{\left(\frac{1}{2}\right)^{2n} \bullet 2^8}{\left(\frac{1}{2}\right)^n}$$

$$= 2^8 \left(\frac{1}{2}\right)^n \quad \forall n$$

$$x_1(n) + x_2(n) = 1 + \left(\frac{1}{2}\right)^n \quad \forall n$$

$$\Rightarrow \quad \underbrace{\frac{\left[1+\left(\frac{1}{2}\right)^{n-4}\right]^2}{1+\left(\frac{1}{2}\right)^n}}_{\text{at } n=4 \text{ this } = \frac{4}{1+\frac{1}{16}}} \neq \underbrace{y_1(n) + y_2(n)}_{\text{at } n=4 \text{ this } = 1 + 2^8\left(\frac{1}{2}\right)^4} = 1 + 2^4$$

Example 3 Averaging Filter

$$y(n) = \frac{1}{3}[x(n-1) + x(n) + x(n+1)]$$

This is a simple example of a low-pass digital filter that could be used to smooth signals and reduce noise.

Linear or nonlinear? Let's answer this by checking superposition (we'll check homogeneity and additivity together).

$$x_1(n) \rightarrow y_1(n) = \frac{1}{3}[x_1(n-1) + x_1(n) + x_1(n+1)]$$

$$x_2(n) \rightarrow y_2(n) = \frac{1}{3}[x_2(n-1) + x_2(n) + x_2(n+1)]$$

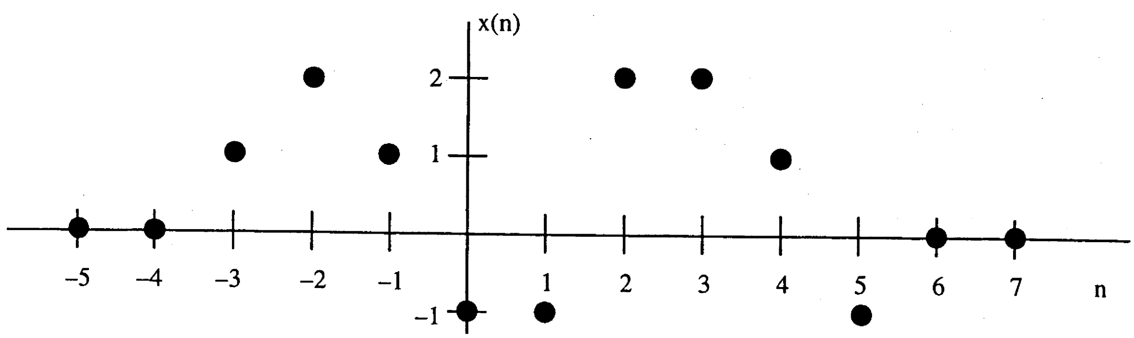
$$\begin{aligned} ax_1(n) + bx_2(n) \rightarrow\ & \frac{1}{3}\Big[ax_1(n-1) + bx_2(n-1) + \big(ax_1(n) + bx_2(n)\big) \\ & + \big(ax_1(n+1) + bx_2(n+1)\big)\Big] \\ =\ & \frac{a}{3}[x_1(n-1) + x_1(n) + x_1(n+1)] + \frac{b}{3}[x_2(n-1) + x_2(n) + x_2(n+1)] \\ =\ & a\,y_1(n) + b\,y_2(n) \quad \checkmark \end{aligned}$$

$\Rightarrow$ Linear

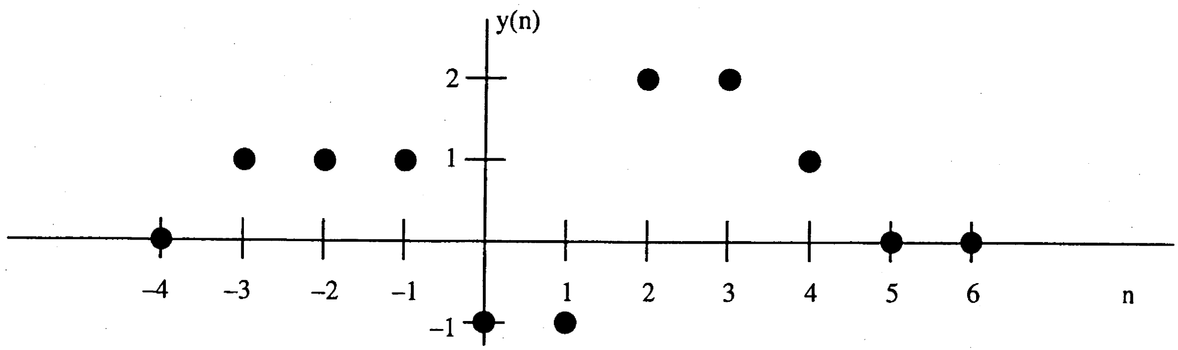
Example 4 Median Filter

$$y(n) = \text{med}\,\{x(n-1), x(n), x(n+1)\}$$

Recall that the median of a sequence of numbers is the middle element in the sequence in terms of algebraic value. Thus, for example, the input sequence



produces the following median filter output



Notice that for any value of n, the median filter output is always equal to one of the elements of the input sequence $\{x(n)\}_{n=-\infty}^{\infty}$. It is easy to visualize the output $\{y(n)\}_{n=-\infty}^{\infty}$ by mentally sliding a length-three window over the input sequence and then simply taking the output to be the middle element (in algebraic value) among those three input elements falling within the window. So, for example,

$$y(-2) = \text{med}\,\{x(-3), x(-2), x(-1)\} = \text{med}\,\{1, 2, 1\} = 1$$

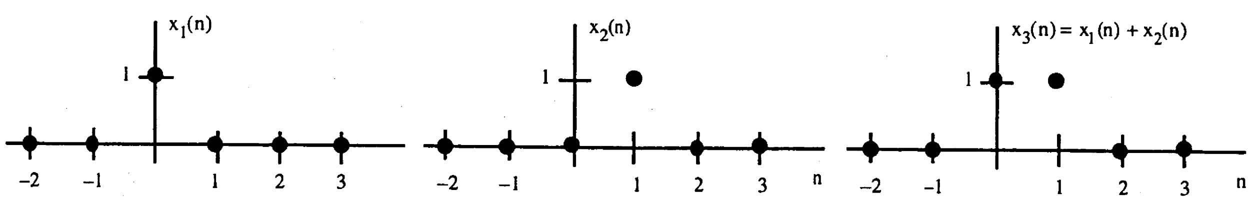
and

$$y(1) = \text{med}\,\{x(0), x(1), x(2)\} = \text{med}\,\{-1, -1, 2\} = -1.$$

Is the median filter linear or nonlinear?

Homogeneity is obviously satisfied, since scaling $\{x(n)\}$ simply scales the median by the same factor.

What about additivity? It appears that the output due to the sum of two inputs may not be the sum of the original two outputs. Choose a particular $\{x_1(n)\}$ and $\{x_2(n)\}$ to demonstrate this. Try



Notice that both $\{y_1(n)\}$ and $\{y_2(n)\}$ are identically zero, whereas

$$y_3(n) = \begin{cases} 1 & n = 0, 1 \\ 0 & \text{otherwise} \end{cases}.$$

Thus, $y_3(n) \neq y_1(n) + y_2(n)$. This shows that the median filter is nonlinear.

Median filters are very useful in image processing, because unlike linear averaging filters, median filters can remove noise while preserving edge structure. Linear filters tend to blur edges, which is very objectionable in image processing. In the 1-D case, it is easy to see that median filters preserve edges. Consider the edge signal

$$x(n) = \begin{cases} 1 & n \geq 0 \\ 0 & n < 0 \end{cases} .$$

For this input, $y(n) = x(n)$, i.e., the median filter preserves the edge. The averaging filter, however, will not preserve the edge. Try it!

Example 5 Modulator

$$y(n) = \cos(7n)\, x(n)$$

This system is linear. The proof is straightforward:

$$x_1(n) \quad \rightarrow \quad y_1(n) = \cos(7n)\, x_1(n)$$

$$x_2(n) \quad \rightarrow \quad y_1(n) = \cos(7n)\, x_2(n)$$

$$a\, x_1(n) + b\, x_2(n) \quad \rightarrow \quad \cos(7n)\, [a\, x_1(n) + b\, x_2(n)]$$

$$= a \cos(7n)\, x_1(n) + b \cos(7n)\, x_2(n)$$

$$= a\, y_1(n) + b\, y_2(n) \qquad \checkmark$$

Example 6

$$y(n) + b_1 y(n-1) + b_2 y(n-2) + \ldots + b_K y(n-K) = a_0 x(n) + a_1 x(n-1) + \ldots + a_N x(n-N) \qquad (\Delta)$$

This is a linear, constant-coefficient difference equation, which describes a linear system. (Linearity of this system is easier to prove after we have studied z-transforms, so we will not give the proof here.) This type of equation is the discrete-time counterpart to the differential equation. Whereas analog circuits are described by differential equations, digital filters are described by difference equations. Difference equations also model numerous situations in the mathematical, physical, and financial worlds. We will see some of these applications as we study, in detail, the solution of difference equations, in the next section.

Def. A system is shift-invariant if a shift in the input always causes a corresponding shift in the output, i.e.,

$$x(n) \rightarrow y(n) \quad \Rightarrow \quad x(n-n_o) \rightarrow y(n-n_o) \qquad \text{(for all } \{x(n)\} \text{ and all } n_o.)$$

The systems in Examples 1, 2, 3 and 5 are shift-invariant. The system in Example 4 is shift-varying. How do we prove this? You'll make the fewest mistakes if you follow the pattern of the proofs below. Let's first revisit Example 3.

Example 3 (revisited)

$$y(n) = \frac{1}{3}[x(n-1) + x(n) + x(n+1)]$$

Is this system shift-invariant? Check:

$$x(n) \quad \rightarrow \quad y(n) = \frac{1}{3}[x(n-1) + x(n) + x(n+1)] \qquad (1)$$

Now, consider a shifted version of $\{x(n)\}$, say $\bar{x}(n) = x(n-n_o)$. Then

$$\begin{aligned} \bar{x}(n) = x(n-n_o) \quad \rightarrow \quad \bar{y}(n) &= \frac{1}{3}[\bar{x}(n-1) + \bar{x}(n) + \bar{x}(n+1)] \\ &= \frac{1}{3}[x(n-1-n_o) + x(n-n_o) + x(n+1-n_o)] \\ &= \frac{1}{3}[x(n-n_o-1) + x(n-n_o) + x(n-n_o+1)] \\ &= y(n-n_o) \quad \checkmark \end{aligned}$$

where the last equality follows from (1). Thus, this system is shift-invariant.

Let's next revisit Example 5.

Example 5 (revisited)

$$y(n) = \cos(7n)\, x(n)$$

Is this system shift-invariant? Check:

$$x(n) \quad \rightarrow \quad y(n) = \cos(7n)\, x(n) \qquad (2)$$

A shifted version of x(n) produces

$$\begin{aligned} \bar{x}(n) = x(n-n_o) \quad \rightarrow \quad \bar{y}(n) &= \cos(7n)\, \bar{x}(n) \\ &= \cos(7n)\, x(n-n_o) \\ &\neq y(n-n_o) \end{aligned}$$

where the inequality follows from (2). Basically, shifting the input $\{x(n)\}$ does not shift the cosine, and so this system (a modulator) is shift-varying.

Def. A system is <u>causal</u> if for every n, $y(n)$ depends only on $x(m)$, $m \leq n$.

Thus, for causal systems, current outputs do not depend on future inputs. Systems that are not causal are called <u>noncausal</u>. Noncausal systems are not physically realizable if the output $y(n)$ must be computed immediately upon acquiring $x(n)$. However, in many DSP systems, data $\{x(n)\}$ is acquired and stored before processing (e.g., store an image $\{x(n, m)\}$ prior to filtering). These systems can be noncausal.

The systems in Examples 1 and 2 are causal. The systems in Examples 3 and 4 are noncausal. The system in Example 6 can be either causal or noncausal, depending on the "direction" in which the equation is iterated. For example, rewriting (Δ) so that $y(n\text{-}K)$ is computed from $x(n)$, $y(n)$, $y(n\text{-}1)$, ... , $y(n-K+1)$ suggests a noncausal realization.

$$y(n-K) = \frac{1}{b_K}\,[a_0x(n) + a_1x(n-1) + \ldots + a_Nx(n-N) - y(n) - b_1y(n-1) - \ldots - b_{K-1}y(n-K+1)].$$

Except when stated otherwise, we will assume that difference equations are iterated in the forward direction, i.e., $y(n)$ is computed via (Δ) as

$$y(n) = a_0x(n)+a_1x(n-1) + \ldots + a_Nx(n-N) - b_1y(n-1) - b_2y(n-2) - \ldots - b_Ky(n-K)$$

Example 7

$$y_n = \frac{x_n}{x_5}$$

This system is nonlinear, shift-varying, and noncausal.

Example 8

$$y(n) = x(-n)$$

This system is linear, shift-varying, and noncausal.

Example 9

$$y(n) = x(|n|)$$

This system is linear, shift-varying, and noncausal.

ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 3

D. Munson

Difference Equations

Homogeneous case (input term equals zero):

$$y_n + a_1 y_{n-1} + a_2 y_{n-2} + \dots + a_K y_{n-K} = 0 \quad n \geq 0 \quad \text{IC's: } y_{-1}, y_{-2}, \dots, y_{-K}$$

Since this D.E. holds for $n \geq 0$, and we assume it iterates in the forward direction, a suitable set of IC's is those shown. This set of IC's would permit us to find $\{y_n\}_{n=0}^{\infty}$ one element at a time, i.e.,

$$\begin{aligned} y_0 &= -a_1 y_{-1} - a_2 y_{-2} - \dots - a_K y_{-K} \\ y_1 &= -a_1 y_0 - a_2 y_{-1} - \dots - a_K y_{1-K} \\ y_2 &= -a_1 y_1 - a_2 y_0 - \dots - a_K y_{2-K} \\ &\vdots \end{aligned}$$

This solution method is generally unsatisfactory, because it does not result in a formula for y_n. Such an expression is needed to discover the form for $\{y_n\}$ and to find values of y_n for arbitrarily large n. Fortunately there is a recipe for finding a closed-form expression for $\{y_n\}$.

Solution Recipe

1) Write characteristic equation:

$$z^K + a_1 z^{K-1} + \dots + a_{K-1} z + a_K = 0$$

Let roots be $\{r_i\}_{i=1}^{K}$.

2) If roots are distinct then solution has form

$$y_n = c_1 r_1^n + c_2 r_2^n + \dots + c_K r_K^n \qquad (*)$$

If a root r_i is repeated m times, then corresponding terms in (*) are replaced by:

$$b_1 r_i^n + b_2 n r_i^n + b_3 n^2 r_i^n + \dots + b_m n^{m-1} r_i^n$$

3) Use IC's to solve for the c_k and b_k.

Note: It is easy to see that (*) is the correct form for the solution. To check that it works, substitute (*) into the D.E. to get

$$\left(c_1 r_1^n + \ldots + c_K r_K^n\right) \quad + a_1 \left(c_1 r_1^{n-1} + \ldots + c_K r_K^{n-1}\right) + \ldots$$

$$+ a_K \left(c_1 r_1^{n-K} + \ldots + c_K r_K^{n-K}\right) \;\stackrel{?}{=}\; 0$$

Rearranging, we have

$$c_1 \left(r_1^n + a_1 r_1^{n-1} + \ldots + a_K r_1^{n-K}\right) + c_2 \left(r_2^n + a_1 r_2^{n-1} + \ldots + a_K r_2^{n-K}\right) + \ldots$$

$$+ c_K \left(r_K^n + a_1 r_K^{n-1} + \ldots + a_K r_K^{n-K}\right) \;\stackrel{?}{=}\; 0$$

Factoring out r_i^{n-K} from each term gives

$$c_1\, r_1^{n-K}\left(r_1^K + a_1 r_1^{K-1} + \ldots + a_K\right) \quad + c_2\, r_K^{n-K}\left(r_2^{n-K} + a_1 r_2^{K-1} + \ldots + a_K\right) + \ldots$$

$$+ c_K\, r_K^{n-K}\left(r_K^K + a_1 r_K^{K-1} + \ldots + a_K\right) \;\stackrel{?}{=}\; 0$$

Now, since each root r_i satisfies the characteristic equation, each parenthesized term is zero, which implies the sought equality.

Example

$$y_n - 3y_{n-1} = 0 \qquad n \geq 0, \quad y_{-1} = 2$$

<u>Note</u>: Could iterate:

$$y_n = 3y_{n-1} \qquad n \geq 0$$

So,

$$y_0 = 3y_{-1} = 3 \bullet 2 = 6$$

$$y_1 = 3y_0 = 18$$

$$y_2 = 3y_1 = 54$$

$$\vdots$$

However, iteration does not provide a closed-form solution. To get this, follow steps 1) – 3):

Characteristic equation:

$$z - 3 = 0 \qquad \text{Root} = 3$$

$$\Rightarrow \quad y_n = c\, 3^n$$

IC:

$$y_{-1} = 2 = c\,\frac{1}{3} \quad \Rightarrow \quad c = 6$$

$$\Rightarrow \quad \boxed{y_n = 6(3)^n \quad n \geq -1}$$

Example

$$y_n + 4y_{n-1} + 4y_{n-2} = 0 \quad n \geq 0, \;\; y_{-1} = y_{-2} = 1$$

Note that the IC's can be no earlier than at $n = -1, -2$ because the D. E. holds only for $n \geq 0$. Thus, IC's at times earlier than $n = -1, -2$ cannot be used to start the recursion.

Characteristic equation:

$$z^2 + 4z + 4 = 0$$

Roots: $-2, -2$

$$\Rightarrow \quad y_n = c_1(-2)^n + c_2\, n(-2)^n$$

IC's:

$$y_{-1} = 1 = c_1\left(-\frac{1}{2}\right) + c_2\left(\frac{1}{2}\right)$$

$$y_{-2} = 1 = c_1\left(\frac{1}{4}\right) + c_2\,(-2)\left(\frac{1}{4}\right)$$

$$\Rightarrow \quad 1 = -\frac{1}{2}\,c_1 + \frac{1}{2}\,c_2$$

$$1 = \frac{1}{4}\,c_1 - \frac{1}{2}\,c_2$$

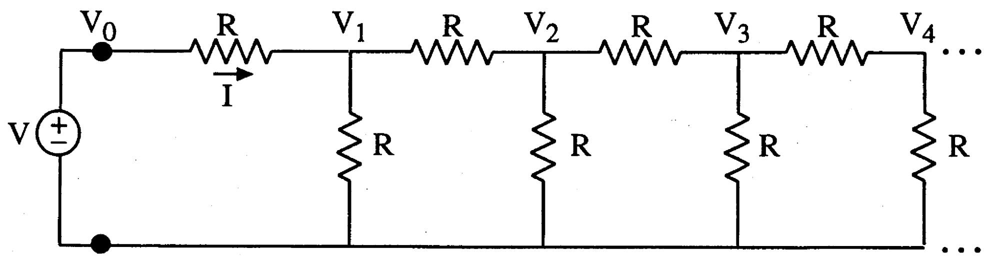
$$\Rightarrow \quad c_1 = -8, \quad c_2 = -6$$

$$\Rightarrow \quad \boxed{y_n = -8\,(-2)^n - 6\,n\,(-2)^n} \qquad n \geq -2$$

Notice that the solution holds for $n \geq -2$ even though the D.E. holds only for $n \geq 0$. Is this correct? Yes, because incorporation of the IC's <u>forced</u> the solution to match at $n = -1, -2$.

Example

A particular transmission line is crudely modeled by the following infinite-length resistive network:



Find the equivalent resistance $\frac{V}{I}$ for this network.

Strategy: Write a D.E. and solve it for V_n. Then

$$I = \frac{V_0 - V_1}{R}.$$

Using KCL at node n gives

$$\frac{V_n - V_{n-1}}{R} + \frac{V_n}{R} + \frac{V_n - V_{n+1}}{R} = 0$$

$\Rightarrow$ $$3\,V_n - V_{n+1} - V_{n-1} = 0$$

$\Rightarrow$ $$V_{n+1} - 3V_n + V_{n-1} = 0 \qquad n \geq 1$$

What do we use for "initial" conditions? Answer: Use $V_o = V$, $V_\infty = 0$.

So, solve

$$V_{n+1} - 3V_n + V_{n-1} = 0\,, \quad n \geq 1, \quad V_0 = V,\ V_\infty = 0.$$

Characteristic equation:

$$z^2 - 3z + 1 = 0$$

Roots:

$$r_1, r_2 = \frac{3 \pm \sqrt{9-4}}{2} = \frac{3}{2} \pm \frac{\sqrt{5}}{2}$$

$\Rightarrow$ $$V_n = c_1 \left(\frac{3}{2} + \frac{\sqrt{5}}{2}\right)^n + c_2 \left(\frac{3}{2} - \frac{\sqrt{5}}{2}\right)^n$$

Now, apply initial conditions.

Since $\frac{3}{2} + \frac{\sqrt{5}}{2} > 1$ and $V_\infty = 0$, we see $c_1 = 0$.

$$\Rightarrow \qquad V_n = c_2 \left(\frac{3}{2} - \frac{\sqrt{5}}{2} \right)^n .$$

Since $V_o = V$, we see $c_2 = V$. Thus,

$$V_n = V \left(\frac{3}{2} - \frac{\sqrt{5}}{2} \right)^n .$$

Now,

$$V_1 = V \left(\frac{3}{2} - \frac{\sqrt{5}}{2} \right).$$

So,

$$I = \frac{V_0 - V_1}{R} = \frac{V - V\left(\frac{3}{2} - \frac{\sqrt{5}}{2} \right)}{R} = \frac{V}{R} \left(\frac{\sqrt{5} - 1}{2} \right)$$

Thus,

$$R_{eq} = \frac{V}{I} = \boxed{R \left(\frac{2}{\sqrt{5} - 1} \right)}$$

Example

$$y_{n+2} + 2\,y_{n+1} + 5\,y_n = 0 \quad n \geq 0; \quad y_0 = 0,\ y_1 = 1$$

Characteristic equation:

$$z^2 + 2z + 5 = 0$$

Roots:

$$\frac{-2 \pm (4-20)^{\frac{1}{2}}}{2} = -1 \pm j2$$

$$\Rightarrow \quad y_n = c_1\,(-1+j2)^n + c_2\,(-1-j2)^n$$

IC's:

$$y_0 = 0 = c_1 + c_2$$

$$y_1 = 1 = c_1\,(-1+j2) + c_2\,(-1-j2)$$

$$\Rightarrow \quad c_1 = \frac{1}{j4}, \quad c_2 = \frac{-1}{j4}$$

$$\Rightarrow \quad y_n = -j\,\frac{1}{4}\,(-1+j2)^n + j\,\frac{1}{4}(-1-j2)^n$$

But know y_n is real because coefficients and IC's of D.E. are real. Write:

$$-j\,\frac{1}{4} = \frac{1}{4}\,e^{-j90^\circ}, \quad j\,\frac{1}{4} = \frac{1}{4}\,e^{j90^\circ}$$

$$-1+j2 = \sqrt{5}\ e^{j\tan^{-1}\left(\frac{2}{-1}\right)} \approx \underset{\substack{\uparrow \\ \text{Let } \theta \,\triangleq\, \tan^{-1}(-2)}}{\sqrt{5}\ e^{j116.6^\circ}}, \quad -1-j2 = \sqrt{5}\ e^{-j\theta}$$

$$\Rightarrow \quad y_n = \frac{1}{4}\,e^{-j90^\circ}\left(\sqrt{5}\,e^{j\theta}\right)^n + \frac{1}{4}\,e^{j90^\circ}\left(\sqrt{5}\,e^{-j\theta}\right)^n$$

$$= \frac{1}{4}\left(\sqrt{5}\right)^n\left[e^{j(\theta n - 90^\circ)} + e^{-j(\theta n - 90^\circ)}\right]$$

$$= \frac{1}{2}\left(\sqrt{5}\right)^n \cos\,(\theta n - 90^\circ) = \frac{1}{2}\left(\sqrt{5}\right)^n \underset{\substack{\uparrow \\ \text{Would usually write } \theta \text{ in radians}}}{\sin\,(\theta n)} \qquad n \geq 0$$

Nonhomogeneous Difference Equations

Consider

$$y(n) + a_1\, y(n-1) + a_2\, y(n-2) + \dots + a_K\, y(n-K) = x(n) \quad n \geq 0, \text{ IC's: } y(-1), \dots, y(-K)$$

Solution procedure:

1) Solve homogeneous equation for homogeneous solution $y_h(n)$.

2) Find any particular solution $y_p(n)$ to nonhomogeneous equation. $y_p(n)$ will not match IC's in general.

3) Total solution is

$$y(n) = y_h(n) + y_p(n)$$

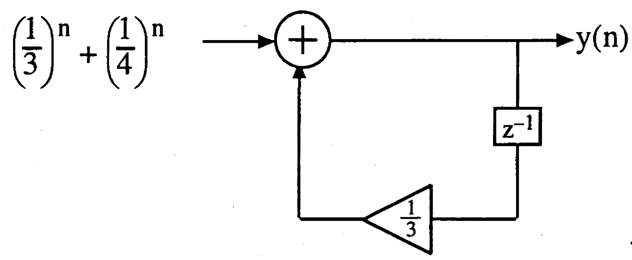
where constants in $y_h(n)$ are chosen to make $y(n)$ match IC's. In Step 2), use method of undetermined coefficients. Basically, just choose $y_p(n)$ to have same form as $x(n)$. Given an input having the form below, select the corresponding form for $y_p(n)$.

Input $x(n)$	$y_p(n)$
a^n	$c\, a^n$
$a_0 + a_1 n + \dots + a_M n^M$	$c_0 + c_1 n + \dots + c_M n^M$
$\cos\theta n$ or $\sin\theta n$	$c_1 \cos\theta n + c_2 \sin\theta n$ or $\bar{c}_1 e^{j\theta n} + \bar{c}_2 e^{-j\theta n}$
$a^n \cos\theta n$ or $a^n \sin\theta n$	$c_1 a^n \cos\theta n + c_2 a^n \sin\theta n$ or $\bar{c}_1 a^n e^{j\theta n} + \bar{c}_2 a^n e^{-j\theta n}$

But, if a term in $x(n)$ is a term in $y_h(n)$, then multiply corresponding part of $y_p(n)$ by n^k with k large enough so that no term in y_p is a term in y_h.

Example

Consider



$y(-1) = 1$

Find $\{y(n)\}_{n=0}^{\infty}$.

The above system is modeled by

$$y(n) = \frac{1}{3}\, y(n-1) + \left(\frac{1}{3}\right)^n + \left(\frac{1}{4}\right)^n \quad n \geq 0, \quad y(-1) = 1$$

$$\Rightarrow \quad y(n) - \frac{1}{3}\, y(n-1) = \left(\frac{1}{3}\right)^n + \left(\frac{1}{4}\right)^n \quad n \geq 0, \quad y(-1) = 1$$

Characteristic equation:

$$z - \frac{1}{3} = 0 \;\Rightarrow\; r = \frac{1}{3}$$

$$\Rightarrow \quad \left[y_h(n) = c\left(\frac{1}{3}\right)^n \right]$$

Try $\quad y_p(n) = A\, n \left(\frac{1}{3}\right)^n + B \left(\frac{1}{4}\right)^n$

$\uparrow$

needed since $\left(\frac{1}{3}\right)^n$ is a term in y_h.

Substitute $y_p(n)$ into D.E. to find A, B:

$$An \left(\frac{1}{3}\right)^n + B \left(\frac{1}{4}\right)^n - \frac{1}{3}\, A\,(n-1) \left(\frac{1}{3}\right)^{n-1} - \frac{1}{3}\, B \left(\frac{1}{4}\right)^{n-1} = \left(\frac{1}{3}\right)^n + \left(\frac{1}{4}\right)^n$$

$$\Rightarrow \quad \left(\frac{1}{3}\right)^n [An - An + A] + \left(\frac{1}{4}\right)^n \left[B - \frac{4}{3}B\right] = \left(\frac{1}{3}\right)^n + \left(\frac{1}{4}\right)^n$$

$$\Rightarrow \quad A = 1, \quad B = -3$$

$$\Rightarrow \qquad \left[y_p(n) = n\left(\frac{1}{3}\right)^n - 3\left(\frac{1}{4}\right)^n \right]$$

Total response:

$$y(n) = c\left(\frac{1}{3}\right)^n + n\left(\frac{1}{3}\right)^n - 3\left(\frac{1}{4}\right)^n$$

IC: $\quad y(-1) = 1 = 3c - 3 - 12 \qquad \Rightarrow \qquad c = \dfrac{16}{3}$

$$\Rightarrow \qquad \boxed{y(n) = \left(\frac{1}{3}\right)^n \left(\frac{16}{3} + n\right) - 3\left(\frac{1}{4}\right)^n} \qquad n \geq -1$$

Example

Let S(n) be the sum of the integers from 1 through n. Find a closed-form expression for S(n).

Solution

$$S(n) = S(n-1) + n \qquad n \geq 2, \quad S(1) = 1$$

or

$$S(n) - S(n-1) = n \qquad n \geq 2, \quad S(1) = 1$$

Characteristic equation:

$$z - 1 = 0$$

$\Rightarrow \qquad S_h(n) = c(1)^n = c$

Choose $\quad S_p(n) = A + Bn$?

No, because A has same form as $S_h(n)$.

Choose $\quad S_p(n) = n(A + Bn) = An + Bn^2$.

Substitute into D. E. to find A, B:

$$An + Bn^2 - A(n-1) - B(n-1)^2 = n$$

$$\Rightarrow \quad n^2\,[B - B] + n\,[A - A + 2B] + A - B = n$$

$$\Rightarrow \quad B = \frac{1}{2}, \; A = B$$

$$\Rightarrow \quad S_p(n) = \frac{1}{2}\,n + \frac{1}{2}\,n^2$$

Total solution:

$$S(n) = c + \frac{1}{2}\,n + \frac{1}{2}\,n^2$$

Apply IC:

$$S(1) = 1 = c + \frac{1}{2} + \frac{1}{2}$$

$$\Rightarrow \quad c = 0$$

$$\Rightarrow \quad \boxed{S_n = \frac{n(n+1)}{2} \quad n \geq 1}$$

Note: This is a rare example where our guess for the particular solution matched the initial condition!

Example

Deposit \$4,000 per year into a retirement account with 8% interest rate compounded annually. Let $B(n)$ = balance at end of nth year. Then the evolution of $B(n)$ is described by

$$B(n) = 1.08\,(B(n-1) + 4000) \quad n \geq 1, \quad B(0) = 0$$

or

$$B(n) - 1.08\,B(n-1) = 4320$$

Homogeneous solution:

$$B_h(n) = c\,(1.08)^n$$

Particular solution:

$$B_p(n) = A \;\Rightarrow\; A - 1.08A = 4320 \Rightarrow A = -54{,}000$$

$$B(n) = B_h(n) + B_p(n) = c\,(1.08)^n - 54{,}000$$

IC: $B_o = 0 = c - 54{,}000 \Rightarrow c = 54{,}000$

$$\Rightarrow \quad B(n) = 54{,}000\left[(1.08)^n - 1\right]$$

Inflation:

Now, if in addition to 8% interest we have 5% inflation, what is the balance in terms of dollars at the beginning of the investment period?

With 5% inflation, something costing \$1 at beginning of first year will cost $\$(1.05)^n$ at end of nth year. So, \$1 at end of nth year is worth only $\$\,\dfrac{1}{(1.05)^n}$ in terms of "real dollars" at beginning of investment period.

$$\Rightarrow \quad \underset{\uparrow}{\overline{B}_n} = \frac{54{,}000\left[(1.08)^n - 1\right]}{(1.05)^n}$$

"real dollars"

For example: $B_{40} = \$1{,}119{,}124.16$

$\overline{B}_{40} = \$158{,}966.75$

Note that $\overline{B} < 40$ (\$4,000)!!

Moral: Increase your yearly contribution with inflation!

ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 5

D. Munson

Definition

The <u>transient response</u> is the part of y(n) that decays to zero as $n \to \infty$.

Definition

The <u>steady-state response</u> is the part of y(n) that is not transient.

Example

Suppose $\quad y(n) = \left(\frac{1}{3}\right)^n + 2^n + n\left(\frac{1}{4}\right)^n + \cos\frac{\pi}{7}n.$

Then $\quad y_{tr}(n) = \left(\frac{1}{3}\right)^n + n\left(\frac{1}{4}\right)^n$

$$y_{ss}(n) = 2^n + \cos\frac{\pi}{7}n$$

Now, earlier we saw that for a linear system (which satisfies the decomposition property) we can also split y(n) into the zero-input response, $y_s(n)$, and the zero-state response, $y_x(n)$. But, when solving difference equations, we found y(n) as the sum of $y_h(n)$ and $y_p(n)$. How are all these components of y(n) related? It's true that

$$y_s(n) + y_x(n) = y_h(n) + y_p(n) = y_{tr}(n) + y_{ss}(n).$$

Sometimes it is falsely assumed that $y_s(n) = y_h(n) = y_{tr}(n)$ and that $y_x(n) = y_p(n) = y_{ss}(n)$. However, this is not true in general. Frequently

$$\begin{array}{ccccccc} & y_s(n) & & & & y_x(n) & \\ \not\swarrow & & \not\searrow & & \not\swarrow & & \not\searrow \\ y_h(n) & \neq & y_{tr}(n) & & y_p(n) & \neq & y_{ss}(n) \end{array}$$

We illustrate these concepts in the following example.

Example

Consider

$$y(n) + 2\,y(n-1) = 2\cos\left(\frac{\pi}{4}n\right), \quad n \geq 0, \quad y(-1) = 1$$

Find

a) homogeneous response
b) particular response
c) zero-input response
d) zero-state response
e) transient response
f) steady-state response

Characteristic equation:

$$z + 2 = 0 \;\Rightarrow\; y_h(n) = c\,(-2)^n$$

$$y_p(n) = A_1\, e^{j\frac{\pi}{4}n} + A_2\, e^{-j\frac{\pi}{4}n}\,,\quad A_1 = A_2^*$$

Substitute $y_p(n)$ into D.E. to find

$$A_1 = \frac{1}{1+2e^{-j\frac{\pi}{4}n}} = 0.3574\, e^{j0.5299}$$

$$\Rightarrow \qquad A_2 = 0.3574\, e^{-j0.5299}$$

$$\Rightarrow \qquad y_p(n) = 0.3574\, e^{j0.5299}\, e^{j\frac{\pi}{4}n} + 0.3574\, e^{-j0.5299}\, e^{-j\frac{\pi}{4}n}$$

$$= 0.3574\left[e^{j(\frac{\pi}{4}n+0.5299)} + e^{-j(\frac{\pi}{4}n+0.5299)}\right]$$

$$= \boxed{0.7148\cos\left(\frac{\pi}{4}n+0.5299\right)}$$

$$y(n) = y_h(n) + y_p(n) = c\,(-2)^n + 0.7148\cos\left(\frac{\pi}{4}n+0.5299\right)$$

IC:

$$y(-1) = 1 = -\frac{1}{2} + 0.7148\cos\left(0.5299-\frac{\pi}{4}\right) = -\frac{1}{2}c + .6916$$

$$\Rightarrow \qquad c = -0.6168 \quad\Rightarrow\quad \boxed{y_h(n) = -0.6168\,(-2)^n}$$

$$\Rightarrow \qquad \boxed{y(n) = -0.6168\,(-2)^n + 0.7148\cos\left(\frac{\pi}{4}n+0.5299\right)} \qquad (*)$$

$y_s(n)$ is response to

$$y(n) + 2y(n-1) = 0 \qquad n \geq 0,\; y(-1) = 1$$

$$\Rightarrow \qquad y_s(n) = a\,(-2)^n$$

IC:

$$y_s(-1) = 1 = -\frac{1}{2}\,a \Rightarrow \quad a = -2$$

$$\Rightarrow \qquad \boxed{y_s(n) = (-2)^{n+1}}$$

$y_x(n)$ is solution to

$$y(n) + 2y(n-1) = 2\cos\left(\frac{\pi}{4}n\right) \qquad n \geq 0\,,\; y(-1) = 0$$

We could solve this for $y_x(n)$, but finding $y_x(n)$ in this way is just as much work as finding the total solution $y(n)$! In fact, this is the reason that ordinarily when solving difference equations, we find $y_h(n)$ and $y_p(n)$ instead of $y_s(n)$ and $y_x(n)$. The former is less work!

Since we have already found $y(n) = y_h(n) + y_p(n)$ and we have also already found $y_s(n)$, let's just use

$$\begin{aligned} y_x(n) &= y(n) - y_s(n) \\ &= -0.6168\,(-2)^n + 0.7148\cos\left(\frac{\pi}{4}n + 0.5299\right) - (-2)^{n+1} \\ &= \boxed{1.3832\,(-2)^n + 0.7148\cos\left(\frac{\pi}{4}n + 0.5299\right)} \end{aligned}$$

Finally, from (*) we see

$$\boxed{y_{tr}(n) = 0}$$

$$\boxed{y_{ss}(n) = y(n) = -0.6168\,(-2)^n + 0.7148\cos\left(\frac{\pi}{4}n + 0.5299\right)}$$

In this example we see that $y_h(n)$, $y_s(n)$, and $y_{tr}(n)$ are all different, which implies that $y_p(n)$, $y_x(n)$, and $y_{ss}(n)$ are also different. It is still true, though, that

$$y(n) = y_h(n) + y_p(n) = y_s(n) + y_x(n) = y_{tr}(n) + y_{ss}(n)$$

as stated earlier.

Comment: Solving difference equations with sinusoidal inputs can be algebraically messy, especially for second and higher order equations. When we were faced with this sort of difficulty in solving differential equations in ECE 210, we applied the methods of phasors and frequency response. For discrete-time systems, we will not apply a phasor-type method (although we could), but we will later develop the idea of discrete-time frequency response and use it extensively.

ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 6

D. Munson

One-Sided z-Transform

Recall that with analog systems, the one-sided Laplace transform could be used to solve differential equations. Later, the Laplace transform was used to develop the notion of transfer function, to compute convolution, and to test for stability. With discrete systems, it is the z-transform that plays this same role. We will use the z-transform to solve difference equations, develop the notion of discrete-time transfer function, compute convolution, and test stability.

The z-transform is a function of a complex variable, just as was the Laplace transform. Before proceeding with the definition of the z-transform, we first review concepts from complex numbers and functions of a complex variable.

Complex Numbers, Variables, and Functions

Item 1

Define a complex number to be an ordered pair of real numbers:

$$c = (a, b)$$

real valued

a – called "real part" of c

b – called "imaginary part" of c

↑

<u>terrible</u> terminology!

Define addition of complex numbers:

$$c_1 + c_2 = (a_1, b_1) + (a_2, b_2)$$

$$\triangleq (a_1 + a_2, b_1 + b_2)$$

Define multiplication of complex numbers:

$$c_1 \bullet c_2 \triangleq (a_1 a_2 - b_1 b_2,\ a_1 b_2 + a_2 b_1) \qquad (*)$$

(*) is cumbersome and hard to remember.

Suppose we use different <u>notation</u> for a complex number:

c = a + ☺ b

Using this notation, the real number sitting next to the smiling face is the imaginary part of c and the real number not sitting next to the smiling face is the real part of c.

Now, pretend that c is a polynomial in the "variable" ☺.

Then write:

$$
\begin{aligned}
c_1 \bullet c_2 &= (a_1 + \text{☺} b_1)\ (a_2 + \text{☺} b_2) \\
&= a_1\, a_2 + \text{☺}^2\, b_1\, b_2 + \text{☺}\, [a_1\, b_2 + a_2\, b_1]
\end{aligned}
$$

If we agree to <u>replace</u> ☺^{2} by -1 this gives

$$
\begin{aligned}
c_1 \bullet c_2 &= a_1\, a_2 - b_1\, b_2 + \text{☺}\, [a_1\, b_2 + a_2\, b_1] \\
&= (a_1\, a_2 - b_1\, b_2,\ a_1\, b_2 + a_2\, b_1)
\end{aligned}
$$

which is the <u>right answer</u>!

We agree to use j instead of ☺, and everywhere j^2 appears we <u>replace it with</u> -1.

So, in this sense "$j^2 = -1$" but <u>j is just a notational aid</u>.

Item 2

Does $f(x) = x^2 + 1$ have "roots" at $x = \pm j$?

(This illustrates how complex numbers are often introduced in high school.)

If $f(x) = x^2 + 1$ is a function of a real variable, then f(x) does <u>not</u> have a root at $x = j$. This doesn't even make sense!

Define function of complex variable to be

$$
f(\underbrace{(x_R, x_I)}_{\text{ordered pair of } \underline{\text{real}} \text{ variables}}) = (x_R, x_I)^2 + (1, 0)
$$

Now, f itself is complex valued, but we can think of plotting Re[f] and Im[f], <u>or</u> $|f|$ and $\angle f$ over the $x_R - x_I$ plane. And, we can ask whether there is any (x_R, x_I) at which

$$
\left[f((x_R, x_I)) = (0, 0) = 0\right]
$$

↑
true if Re[f] = Im[f] = 0
or if $|f| = 0$.

Check:

$$f\left((x_R, x_I)\right) = (x_R, x_I)^2 + (1, 0)$$

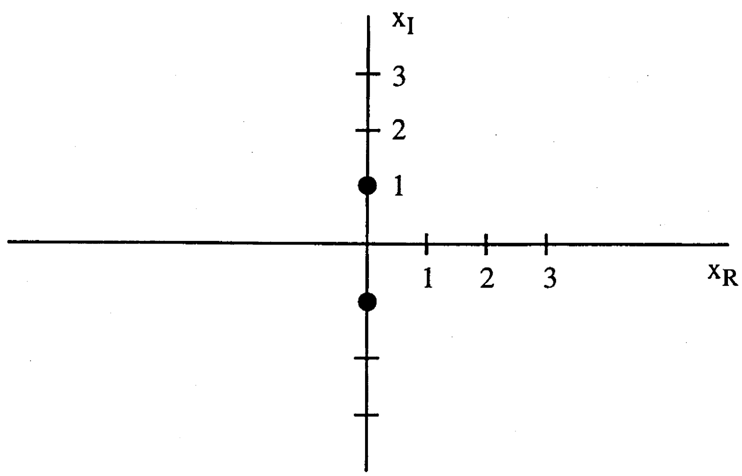
$$= \left(x_R^2 - x_1^2, x_R\, x_I + x_I\, x_R\right) + (1, 0)$$

$$= \left(\underbrace{x_R^2 - x_1^2 + 1}_{\text{Re}[f]},\ \underbrace{2x_R\, x_I}_{\text{Im}[f]}\right)$$

This = (0, 0) iff $x_R = 0$, $x_I = \pm 1$, i.e., if x is a <u>complex</u> variable, then

$f(x) = 0$ at $x = (0, \pm 1) \triangleq \pm j$.

If $x = (x_R, x_I)$ is a complex variable, then can plot $|f(x)|$ as a surface over the $x_R - x_I$ plane. It hits zero at the points shown:

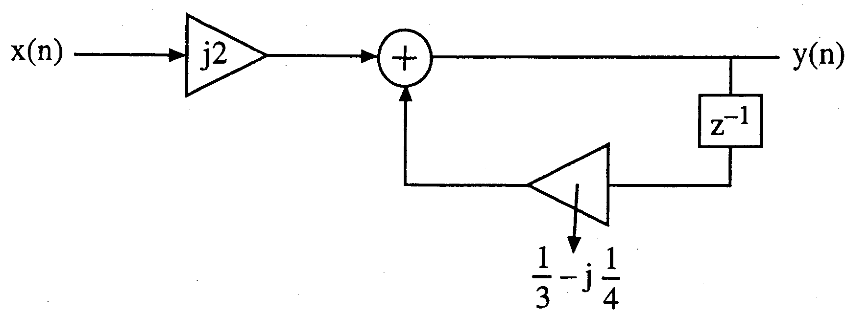


Item 3

We can implement systems that are described by complex numbers! This is commonly done in communications, radar, and many other areas.

Example

Consider



where $x(n) = x_R(n) + j\, x_I(n)$ and $y(n) = y_R(n) + j\, y_I(n)$ are complex-valued sequences. Draw a block diagram that can implement the above system using real signals and real multipliers, adders and delays.

Solution

The first step is to realize that $x(n) = \{x_R(n), x_I(n)\}$ and $y(n) = \{y_R(n), y_I(n)\}$ are pairs of <u>real-valued</u> sequences, i.e., the system above has two inputs $\{x_R(n), x_I(n)\}$ and two outputs $\{y_R(n), y_I(n)\}$. The second step is to recall that complex multiplication and complex addition are defined in terms of <u>real</u> multiplication and <u>real</u> addition, as described on p. 7.1.

Call the output of first multiplication, by j2, $v(n) = v_R(n) + j\, v_I(n)$. This multiplication is accomplished as

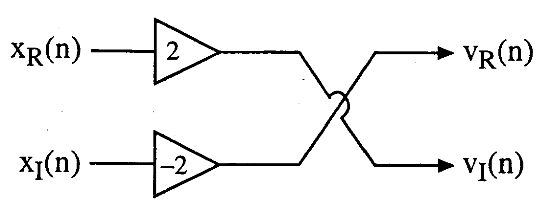
$$v(n) = j2\left(x_R(n) + jx_I(n)\right) = -2\, x_I(n) + j2\, x_R(n)$$

Thus,

$$v_R(n) = -2\, x_I(n)$$

$$v_I(n) = 2\, x_R(n)$$

which is implemented as



Now, let w(n) be the output of the multiplication by $\frac{1}{3} - j\,\frac{1}{4}$. w(n) is computed as

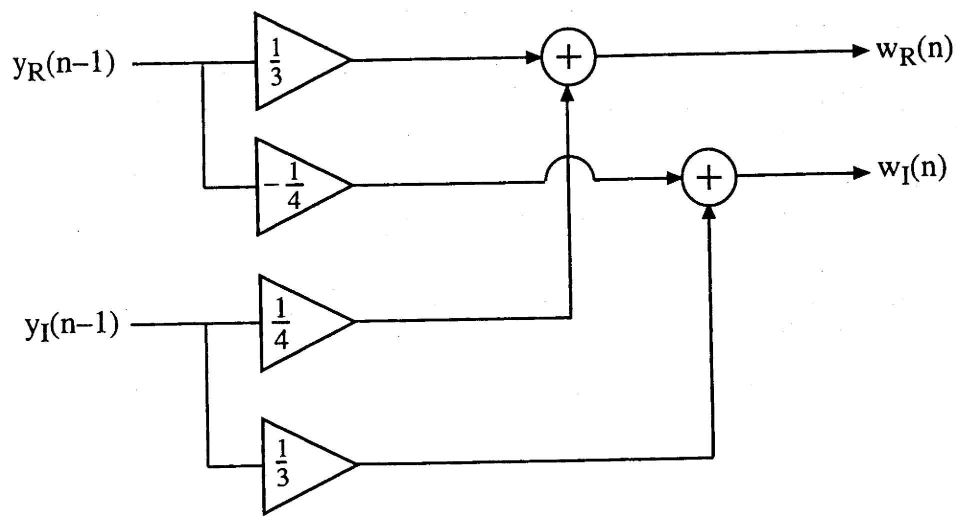
$$\begin{aligned} w(n) &= \left(\frac{1}{3} - j\frac{1}{4}\right)\left(y_R(n-1) + j\, y_I(n-1)\right) \\ &= \frac{1}{3}\, y_R(n-1) + \frac{1}{4}\, y_I(n-1) + j\left(-\frac{1}{4}\, y_R(n-1) + \frac{1}{3}\, y_I(n-1)\right) \end{aligned}$$

Thus,

$$w_R(n) = \frac{1}{3}\, y_R(n-1) + \frac{1}{4}\, y_I(n-1)$$

$$w_I(n) = -\frac{1}{4}\, y_R(n-1) + \frac{1}{3} y_I(n-1)$$

which is implemented as

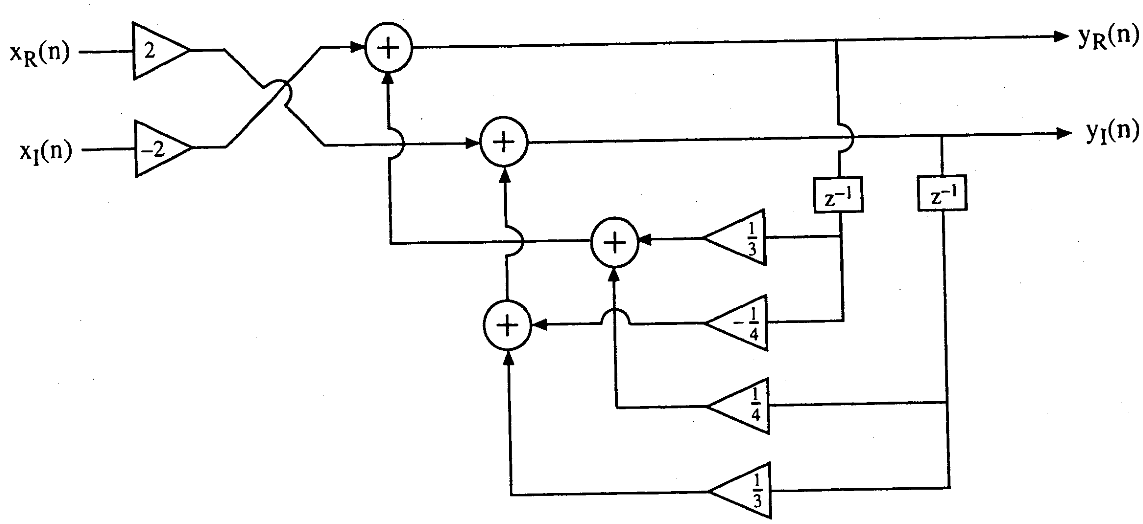


Now, using the fact that $y(n) = v(n) + w(n)$, so that

$$y_R(n) = v_R(n) + w_R(n)$$

$$y_I(n) = v_I(n) + w_I(n)$$

we have the complete block diagram implementation



The original block diagram is simply a concise way of representing this complicated system.

ECE 310 **DIGITAL SIGNAL PROCESSING** **D. Munson**
University of Illinois **Lecture 7**

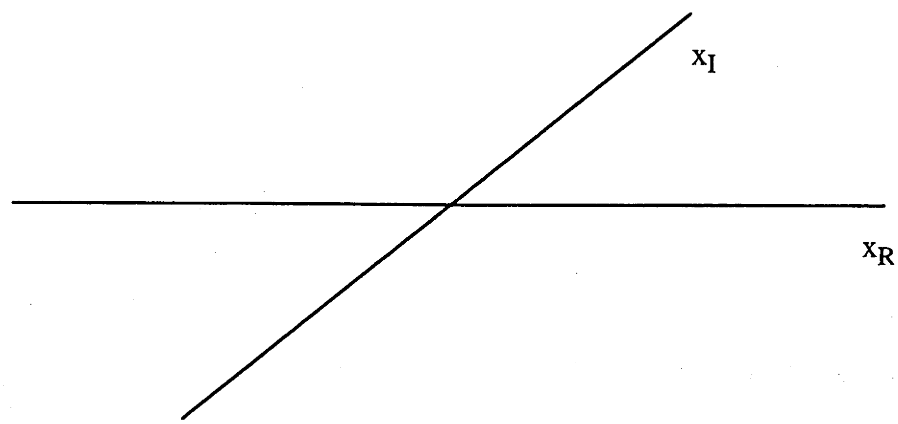
Example

$f(x) = x$
$\uparrow$
Complex variable

Describe the surfaces Re[f(x)], Im[f(x)], |f(x)|, and $\angle f(x)$ plotted over the 2-D complex x-plane.

$$f((x_R, x_I)) = x_R + j\, x_I$$

$\mathrm{Re}\left[f((x_R, x_I))\right] = x_R$ = plane through x_I-axis with slope = 1



$\mathrm{Im}\left[f((x_R, x_I))\right] = x_I$ = plane through x_R-axis with slope = 1.

$$\left|f((x_R, x_I))\right| = \sqrt{x_R^2 + x_I^2}$$

= inverted cone centered at origin

$$\angle f((x_R, x_I)) = \angle(x_R, x_I)$$

= spiral ramp

starting at + x_R-axis (which $\angle(x_R, x_I)$ cuts through), surface ramps up in c. c. direction to the height π along the $-x_R$-axis. In the clockwise direction from $+x_R$-axis the surface ramps down to the level $-\pi$ along the $-x_R$-axis.

$f(x)$ is fully described by the plots of

$$\mathrm{Re}[f(x)], \text{ and } \mathrm{Im}[f(x)]$$

or

$$|f(x)| \text{ and } \angle f(x).$$

Back to One-Sided z-Transform

Now, we will begin our study of the z-transform by first considering the one-sided version of the transform.

Definition

The one-sided z-transform of $\{x_n\}_{n=0}^{\infty}$ is

$$X(z) = \sum_{n=0}^{\infty} x_n z^{-n}, \quad \text{for z such that sum converges.} \qquad (*)$$

Here, z is a complex variable and the set of values of z for which the sum converges are called the region of convergence.

Example

Suppose $x_n = 2^n$. Is $z = 1$ in the ROC of $X(z)$? Is $z = 3$ in the ROC?

First consider $z = 1$:

$$X(1) = \sum_{n=0}^{\infty} 2^n (1)^{-n} = \sum_{n=0}^{\infty} 2^n = \infty$$

Thus, $z = 1$ is not in the ROC of $X(z)$.

Consider $z = 3$:

$$X(3) = \sum_{n=0}^{\infty} 2^n (3)^{-n} = \sum_{n=0}^{\infty} \left(\frac{2}{3}\right)^n$$

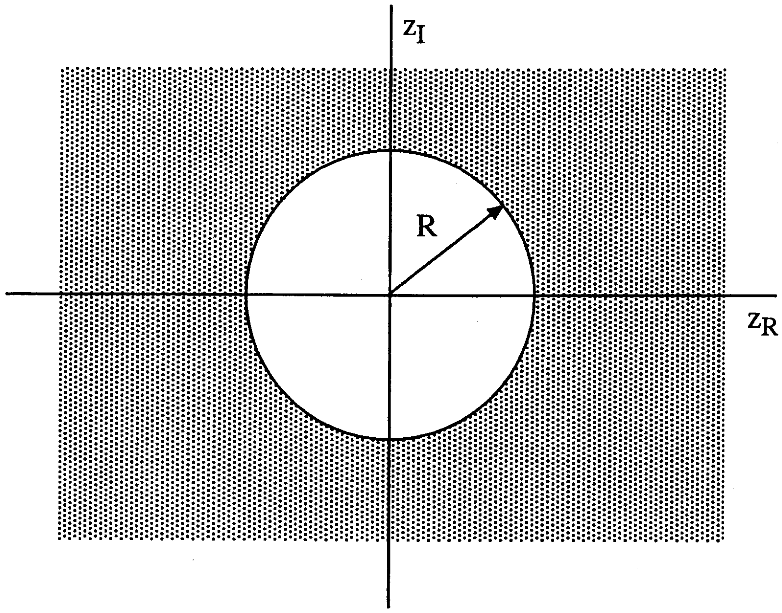
$$= \frac{1}{1-\frac{2}{3}} = 3$$

Thus, $X(z)$ is well defined (finite) at $z = 3$, so that $z = 3$ is a point in the ROC of $X(z)$.

In this example, we saw that the larger value of z is in ROC_X, whereas the smaller value of z is not in ROC_X. It's not a surprise that larger values of z are more likely to be in the ROC. Why so? Because, in the definition of the z-transform, Eq. (*), z is raised to a negative power. Thus, larger values of z offer greater likelihood for convergence of the z-transform sum. In general, X(z) converge for all z that are large enough. Specifically,

X(z) converges for all z such that $|z| > R$ (for some R)

Thus, ROC_X includes all points z lying outside a circle of radius R:



How do we find R? R depends on $\{x_n\}_{n=0}^{\infty}$. We discover the value of R when we try to compute the z-transform sum.

Example

Find the z-transform of

$x_n = a^n \quad n \geq 0$, where 'a' is a fixed constant

$$X(z) = \sum_{n=0}^{\infty} a^n z^{-n} = \sum_{n=0}^{\infty} \left(\frac{a}{z}\right)^n$$

If $\left|\frac{a}{z}\right| < 1$ then this sum converges and has the value

$$X(z) = \frac{1}{1 - \frac{a}{z}} = \frac{z}{z-a}$$

The convergence condition is equivalent to

$$\frac{|a|}{|z|} < 1 \quad \Rightarrow \quad |z| > |a|$$

Thus, $X(z) = \dfrac{z}{z-a}$, $|z| > |a|$

What if $|z| < |a|$? For any such z, X(z) is <u>undefined</u> because the z-transform sum does not converge.

Example

Find the z-transform of

$$x_n = n\, a^n \qquad n \geq 0.$$

Use a trick. We have

$$\begin{aligned}
X(z) &= \sum_{n=0}^{\infty} n\, a^n z^{-n} \\
&= \sum_{n=0}^{\infty} n \left(\frac{a}{z}\right)^n \\
&= a \frac{d}{da} \sum_{n=0}^{\infty} \left(\frac{a}{z}\right)^n \\
&= a \frac{d}{da} \frac{z}{z-a} && |z| > |a| \\
&= a \frac{0-(-z)}{(z-a)^2} && |z| > |a| \\
&= \frac{a z}{(z-a)^2} && |z| > |a|
\end{aligned}$$

In a similar way we can show the more general result

$$n\, x(n) \leftrightarrow -z \frac{d}{dz} X(z)$$

Applying the derivative trick twice gives

$$\frac{1}{2}\, n(n-1)\, a^n \leftrightarrow \frac{a^2 z}{(z-a)^3} \qquad |z| > |a|$$

Applying the trick m times gives the z-transform pair

$$\frac{1}{m!}\, n(n-1)\,(n-2) \ldots (n-m+1)\, a^n \leftrightarrow \frac{a^m z}{(z-a)^{m+1}} \qquad |z| > |a|$$

Example

$y_n = \sin \lambda n \quad n \geq 0$

$$Y(z) = \sum_{n=0}^{\infty} \sin \lambda n \, z^{-n}$$

$$= \sum_{n=0}^{\infty} \frac{1}{j2}\left(e^{j\lambda n} - e^{-j\lambda n}\right) z^{-n}$$

$$= \frac{1}{j2}\left[\sum_{n=0}^{\infty}\left(z^{-1} e^{j\lambda}\right)^n - \sum_{n=0}^{\infty}\left(z^{-1} e^{-j\lambda}\right)^n\right]$$

$$= \frac{1}{j2}\left[\frac{1}{1 - z^{-1} e^{j\lambda}} - \frac{1}{1 - z^{-1} e^{-j\lambda}}\right]$$

$$\text{if} \begin{cases} \left|z^{-1} e^{j\lambda}\right| < 1 \Rightarrow |z| > \left|e^{j\lambda}\right| = 1 \\ \left|z^{-1} e^{-j\lambda}\right| < 1 \Rightarrow |z| > \left|e^{-j\lambda}\right| = 1 \end{cases}$$

So, for $|z| > 1$:

$$Y(z) = \frac{1}{j2}\left[\frac{1 - z^{-1} e^{-j\lambda} - \left(1 - z^{-1} e^{j\lambda}\right)}{1 - z^{-1}\left(e^{j\lambda} + e^{-j\lambda}\right) + z^{-2}}\right]$$

$$= \frac{1}{j2}\left[\frac{z^{-1}\left(e^{j\lambda} - e^{-j\lambda}\right)}{1 - 2z^{-1}\cos\lambda + z^{-2}}\right]$$

$$= \boxed{\frac{z \sin\lambda}{z^2 - 2z\cos\lambda + 1}}$$

Note: We could have shortened our derivation a bit for this example by using the fact that $\sin\lambda n$ is a linear combination of terms of the form a^n, and we already know the z-transform of a^n. Let's try this approach to find the z-transform of $\cos\lambda n$.

Example

Find the one-sided z-transform of $x_n = \cos\lambda n$.

$$x_n = \frac{1}{2} e^{j\lambda n} + \frac{1}{2} e^{-j\lambda n}$$

$$= \frac{1}{2}\left[e^{j\lambda}\right]^n + \frac{1}{2}\left[e^{-j\lambda}\right]^n$$

$$\Rightarrow X(z) = \frac{1}{2}\frac{z}{z-e^{j\lambda}} + \frac{1}{2}\frac{z}{z-e^{-j\lambda}} \qquad |z| > \left|e^{j\lambda}\right| = 1$$

$$\Rightarrow X(z) = \frac{\frac{1}{2}\left(z^2 - z\,e^{-j\lambda} + z^2 - z\,e^{j\lambda}\right)}{\left(z-e^{j\lambda}\right)\left(z-e^{-j\lambda}\right)}$$

$$= \frac{z^2 - z\cos\lambda}{z^2 - 2z\cos\lambda + 1} \qquad |z| > 1$$

Tables of z-transforms are available.

ECE 310
University of Illinois

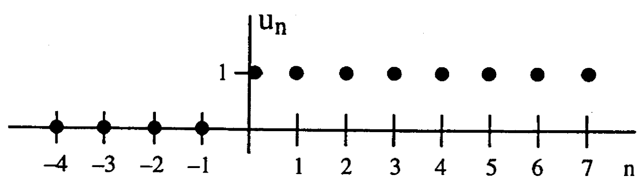
DIGITAL SIGNAL PROCESSING
Lecture 8

D. Munson

Definition: The unit-step sequence u_n is defined as

$$u_n = \begin{cases} 1 & n \geq 0 \\ 0 & n < 0 \end{cases}$$

The unit-step sequence is pictured as



Properties of 1-Sided z-Transform

Notation: $Z\left(\{y_n\}_{n=0}^{\infty}\right) \triangleq Y(z)$

1) Linearity:

$$Z\left(\{a x_n + b y_n\}_{n=0}^{\infty}\right) = a\,X(z) + b\,Y(z)$$

Proof:

$$\sum_{n=0}^{\infty} (a x_n + b y_n)\, z^{-n} = a \sum_{n=0}^{\infty} x_n z^{-n} + b \sum_{n=0}^{\infty} y_n z^{-n}$$

$$= a\,X(z) + b\,Y(z) \quad \checkmark$$

2) Delay Property #1:

Let $k > 0$ be the amount of delay. Then

$$Z(y_{n-k}\, u_{n-k}) = z^{-k}\, Y(z)$$

In words, this property states that truncating a sequence at the origin, and then shifting to the right by k, is equivalent to multiplying the z-transform of the unshifted sequence by z^{-k}.

Proof:

$$Z\left(y_{n-k}\,u_{n-k}\right) = \sum_{n=0}^{\infty} y_{n-k}\,u_{n-k}\,z^{-n}$$

$$= \sum_{n=k}^{\infty} y_{n-k}\,z^{-n}$$

$$= \underset{\substack{\uparrow \\ \ell = n-k}}{\sum_{\ell=0}^{\infty}} y_\ell\,z^{-(\ell+k)}$$

$$= z^{-k} \sum_{\ell=0}^{\infty} y_\ell\,z^{-\ell} = z^{-k}\,Y(z) \qquad \checkmark$$

3) Delay Property #2 ~ can use either this or Advance Prop. to solve D.E.'s.

For cases where $y_{-1}, y_{-2}, \ldots, y_{-k}$ are known or defined ($k > 0$). Here, the y_n sequence is not truncated at the origin, prior to shifting.

Property: $Z\left(\left\{y_{n-k}\right\}_{n=0}^{\infty}\right) = z^{-k}\left[Y(z) + \sum_{m=1}^{k} y_{-m}\,z^{m}\right]$

Proof:

$$Z\left(\left\{y_{n-k}\right\}\right) = \sum_{n=0}^{\infty} y_{n-k}\,z^{-n}$$

$$= \underset{\substack{\uparrow \\ \ell = n-k}}{\sum_{\ell=-k}^{\infty}} y_\ell\,z^{-(\ell+k)}$$

$$= z^{-k} \sum_{\ell=-k}^{\infty} y_\ell\,z^{-\ell}$$

$$= z^{-k}\left[Y(z) + \sum_{\ell=-k}^{-1} y_\ell\,z^{-\ell}\right]$$

$$= \underset{\substack{\uparrow \\ m = -\ell}}{z^{-k}}\left[Y(z) + \sum_{m=1}^{k} y_{-m}\,z^{m}\right] \qquad \checkmark$$

4) Advance: ~ use later to solve D.E.'s

Property: $Z\left(\{y_{n+k}\}_{n=0}^{\infty}\right) = z^k\left(Y(z) - \sum_{\ell=o}^{k-1} y_\ell z^{-\ell}\right)$

Proof:

$$\sum_{n=o}^{\infty} y_{n+k} z^{-n} = z^k \sum_{n=o}^{\infty} y_{n+k} z^{-(n+k)}$$

$$= \underset{\substack{\uparrow \\ \ell = n+k}}{z^k} \sum_{\ell=k}^{\infty} y_\ell z^{-\ell}$$

$$= z^k\left[Y(z) - \sum_{\ell=o}^{k-1} y_\ell z^{-\ell}\right] \qquad \checkmark$$

5) Convolution ~ useful later

Given $\{x_n\}_{n=o}^{\infty}, \{h_n\}_{n=o}^{\infty}$.

$$\left[y_n = \sum_{m=o}^{n} h_m x_{n-m}, \quad n \geq 0 \quad \text{iff} \quad Y(z) = H(z)X(z)\right]$$

convolution sum

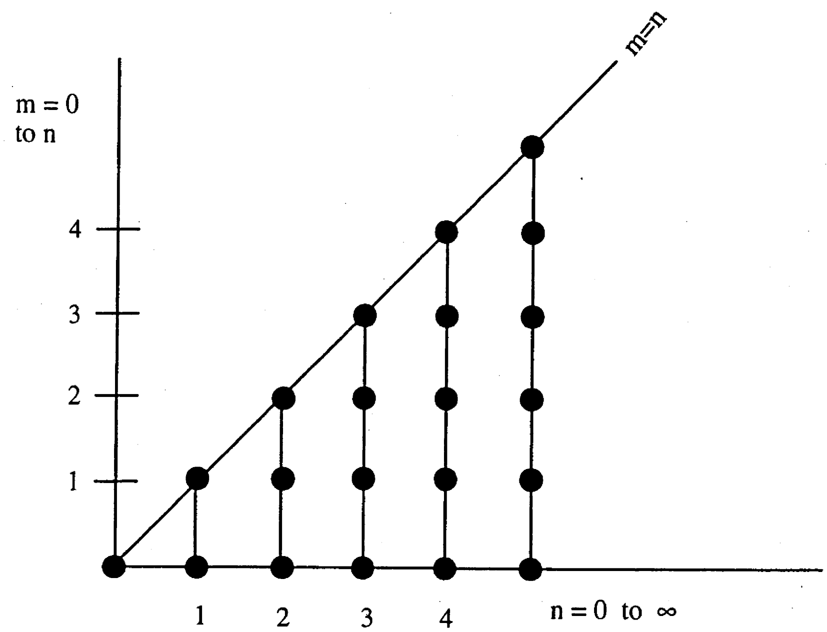
Proof:

Prove only in $\rightarrow$ direction, steps are reversible.

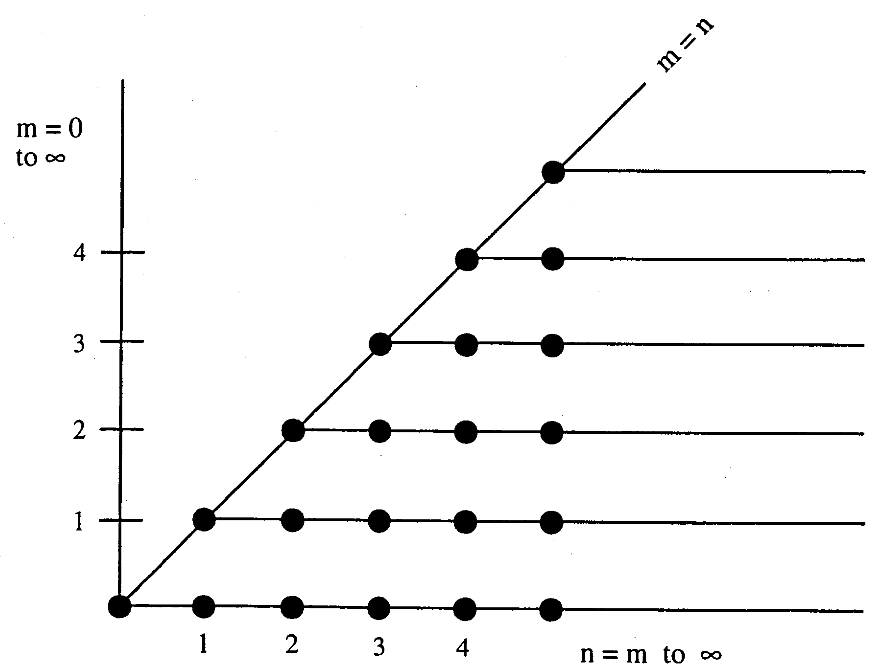
Given $y_n = \sum_{m=o}^{n} h_m x_{n-m}, n \geq o$

$$\Rightarrow Y(z) = \sum_{n=o}^{\infty}\left[\sum_{m=o}^{n} h_m x_{n-m}\right] z^{-n}$$

Look at range of sums:



Instead, compute in this order



$$\Rightarrow Y(z) = \sum_{m=o}^{\infty} \sum_{n=m}^{\infty} h_m x_{n-m} z^{-n}$$

$$= \sum_{m=o}^{\infty} \sum_{\ell=o}^{\infty} h_m x_\ell z^{-(m+\ell)}$$

$\uparrow$

$\ell = n-m$

$$= \sum_{m=o}^{\infty} h_m z^{-m} \sum_{\ell=o}^{\infty} x_\ell z^{-\ell}$$

$$= H(z)\, X(z) \qquad \checkmark$$

Inverse 1-Sided z-Transform

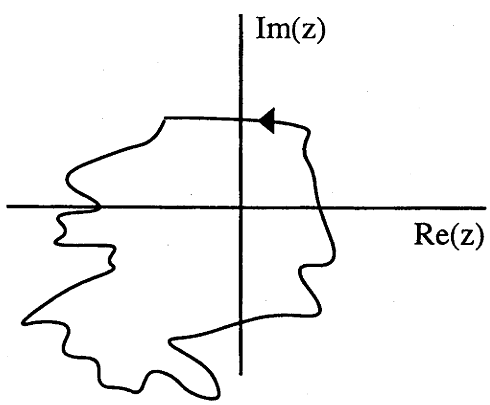
Later: Will take z-transform of both sides of D.E. Solve for Y(z). Take inverse transform to get $\{y_n\}_{n=0}^{\infty}$.

Formula for inverse transform:

$$\left[y_n = \frac{1}{2\pi j} \oint Y(z) z^{n-1} dz \right]$$

$\downarrow$

integral over closed path in cc direction
<u>in region of converge of Y(z)</u>.



Are other inversion methods if Y(z) is a <u>rational function</u> (ratio of polynomials), e.g.,

$$Y(z) = \frac{a_0 + a_1 z + \ldots + a_M z^M}{b_0 + b_1 z + \ldots + b_N z^N}$$

Method 1: Direct division

Usually a poor method – doesn't give closed-form $\{y_n\}$.

Example

$$Y(z) = \frac{z}{z-a}$$

$$\begin{array}{r l}
 & 1+\frac{a}{z}+\frac{a^2}{z^2} \\ \cline{2-2}
z-a \,\big) & z \\
 & \underline{z-a} \\
 & \quad a-\frac{a^2}{z} \\ \cline{2-2}
 & \qquad \frac{a^2}{z} \\
 & \qquad \frac{a^2}{z}-\frac{a^3}{z^2} \\ \cline{2-2}
\end{array}$$

Find:

$$Y(z) = 1 + a\, z^{-1} + a^2\, z^{-2} + a^3\, z^{-3} + \ldots$$

$$Y(z) = \sum_{n=0}^{\infty} y_n\, z^{-n} = y_0 + y_1\, z^{-1} + y_2\, z^{-2} + \ldots$$

$$\Rightarrow\ y_n = \left\{1, a, a^2, \ldots\right\} = a^n$$

Method 2

Partial Fraction Expansion ~ Table Look-Up

Suppose

$$Y(z) = \frac{a_0 + a_1 z + \ldots + a_M z^M}{b_0 + b_1 z + \ldots + b_N z^N} \qquad \text{with } M < N$$

Let roots of denominator be $\{r_i\}_{i=1}^{N}$

If r_i distinct then

$$\left[Y(z) = \sum_{i=1}^{N} \frac{A_i}{z - r_i} \quad \text{with } A_i = (z - r_i)\, Y(z)\Big|_{z = r_i} \right]$$

Example

$$Y(z) = \frac{z-1}{(z-2)(z-3)} \qquad \text{Find } \{y_n\}_{n=0}^{\infty}$$

$$\frac{z-1}{(z-2)(z-3)} = \frac{A_1}{z-2} + \frac{A_2}{z-3}$$

Multiply by $z - 2$ to find A_1:

$$\frac{z-1}{z-3} = A_1 + \frac{A_2(z-2)}{z-3}$$

$$A_1 = \left.\frac{z-1}{z-3}\right|_{z=2} = -1$$

Similarly:

$$A_2 = \left.\frac{z-1}{z-2}\right|_{z=3} = 2$$

$$\Rightarrow \quad Y(z) = \frac{-1}{z-2} + \frac{2}{z-3}$$

$$= -z^{-1}\left(\frac{z}{z-2}\right) + 2z^{-1}\left(\frac{z}{z-3}\right)$$

Previously had

$$a^n \leftrightarrow \frac{z}{z-a}$$

Applying Delay Property #l, we know

$$a^{n-1}\, u_{n-1} \leftrightarrow z^{-1}\, \frac{z}{z-a}$$

Thus

$$y_n = -2^{n-1}\, u_{n-1} + 2\left(3^{n-1}\, u_{n-1}\right)$$

If we prefer, we can rewrite this as

$$y_n = \begin{cases} -\frac{1}{2}2^n + \frac{2}{3}3^n & n \geq 1 \\ 0 & n = 0 \end{cases}$$

Applying Delay Property #1 is inconvenient. We can avoid its use by expanding $\frac{Y(z)}{z}$ in a PFE as

$$\frac{B_1}{z} + \frac{B_2}{z-2} + \frac{B_3}{z-3}$$

Then

$$Y(z) = B_1 + \frac{B_2\, z}{z-2} + \frac{B_3\, z}{z-3}$$

where each individual term is easy to invert. Let's work out the details for this example:

$$\frac{Y(z)}{z} = \frac{z-1}{z(z-2)(z-3)} = \frac{B_1}{z} + \frac{B_2}{z-2} + \frac{B_3}{z-3}$$

$$B_1 = \left.\frac{z-1}{(z-2)(z-3)}\right|_{z=0} = \frac{-1}{6}$$

$$B_2 = \left.\frac{z-1}{z(z-3)}\right|_{z=2} = \frac{-1}{2}$$

$$B_3 = \left.\frac{z-1}{z(z-2)}\right|_{z=3} = \frac{2}{3}$$

So

$$\frac{Y(z)}{z} = \frac{\frac{-1}{6}}{z} - \frac{\frac{1}{2}}{z-2} + \frac{\frac{2}{3}}{z-3}$$

giving

$$Y(z) = -\frac{1}{6} - \frac{\frac{1}{2}z}{z-2} + \frac{\frac{2}{3}z}{z-3}$$

$$\Rightarrow y_n = Z^{-1}\left\{-\frac{1}{6}\right\} - \frac{1}{2}\,2^n + \frac{2}{3}\,3^n \qquad (*)$$

What is the value of $Z^{-1}\left\{-\frac{1}{6}\right\}$?

Note that if a z-transform is constant, say $X(z) = C$, then

$$C = x_0 + x_1 z^{-1} + x_2 z^{-2} + x^3 z^{-3} + \ldots$$

The only way the right side can be constant (i.e., independent of the value of z) is if

$$x_n = \begin{cases} C & n = 0 \\ 0 & n > 0 \end{cases}.$$

Thus,

$$Z^{-1}\left\{-\frac{1}{6}\right\} = \begin{cases} -\frac{1}{6} & n = 0 \\ 0 & n > 0 \end{cases}$$

Substituting into (*) gives

$$y_n = \begin{cases} -\frac{1}{2}2^n + \frac{2}{3}3^n & n \geq 1 \\ -\frac{1}{6} - \frac{1}{2} + \frac{2}{3} & n = 0 \end{cases}$$

$$= \begin{cases} -\frac{1}{2}2^n + \frac{2}{3}3^n & n \geq 1 \\ 0 & n = 0 \end{cases}$$

as before.

In this example, the PFE for $\frac{Y(z)}{z}$ was more complicated (involved one more term) than the PFE for Y(z). In many cases this extra complication does not arise. If the numerator of Y(z) contains a power of z (say z or z^2), then the z in the denominator of $\frac{Y(z)}{z}$ is cancelled, in which case the PFE for $\frac{Y(z)}{z}$ has exactly the same form as the PFE of Y(z).

If r_i not distinct, modify PFE slightly. Suppose r_j repeated q times. Replace:

$\frac{A_i}{z - r_j}$ terms in PFE with:

$$\sum_{i=1}^{q} \frac{B_i}{\left(z - r_j\right)^i}$$

where

$$B_i = \frac{1}{(q-i)!}\left[\frac{d^{q-i}}{dz^{q-i}}\left(z - r_j\right)^q Y(z)\right]\Bigg|_{z = r_j}$$

Don't worry; you don't really need to remember this formula!

Example

$$Y(z) = \frac{z}{(z-1)(z-3)^2}$$

$$\frac{Y(z)}{z} = \frac{1}{(z-1)(z-3)^2} = \frac{A_1}{z-1} + \frac{A_2}{z-3} + \frac{A_3}{(z-3)^2} \qquad (\Delta)$$

$$A_1 = \frac{1}{(z-3)^2}\bigg|_{z=1} = \frac{1}{4}$$

Now, find A_3 <u>before</u> we find A_2. Find the coefficient over the highest power denominator first. Multiplying both sides of (Δ) by $(z\text{–}3)^2$ yields

$$\frac{1}{z-1} = \frac{A_1(z-3)^2}{z-1} + A_2\,(z-3) + A_3 \qquad (\square)$$

Evaluating both sides of ($\square$) at z = 3 gives

$$A_3 = \frac{1}{z-1}\bigg|_{z=3} = \frac{1}{2}$$

Now, a convenient way to find A_2 is to first differentiate ($\square$) with respect to z:

$$\frac{-1}{(z-1)^2} = \frac{2A_1(z-3)(z-1) - A_1(z-3)^2}{(z-1)^2} + A_2$$

Evaluating both sides at z = 3 gives

$$A_2 = \left.\frac{-1}{(z-1)^2}\right|_{z=3} = \frac{-1}{4}$$

Substituting the values of the A_i into (Δ), we have

$$Y(z) = \frac{\frac{1}{4}z}{z-1} - \frac{\frac{1}{4}z}{z-3} + \frac{\frac{1}{2}z}{(z-3)^2}$$

The first two terms are easy to invert. For the third term, recall

$$na^n \leftrightarrow \frac{az}{(z-a)^2}$$

Thus,

$$\begin{aligned} y_n &= \frac{1}{4}(1)^n - \frac{1}{4}(3)^n + \frac{1}{2}\left(\frac{1}{3}\right)n(3)^n \\ &= \frac{1}{4} - \frac{1}{4}\,3^n + \frac{1}{6}\,n\,3^n \end{aligned}$$

Example

$$Y(z) = \frac{2z^3 + z^2 - z + 4}{(z-2)^3} \qquad \text{Find } \{y_n\}_{n=0}^{\infty}$$

Need deg (NUM) < deg (DEN) for PFE.

Write $Y(z) = z\left(\frac{Y(z)}{z}\right)$. Expand $\frac{Y(z)}{z}$ in PFE.

$$\begin{aligned} \frac{Y(z)}{z} &= \frac{2z^3 + z^2 - z + 4}{z(z-2)^3} \\ &= \frac{A_1}{z} + \frac{A_2}{z-2} + \frac{A_3}{(z-2)^2} + \frac{A_4}{(z-2)^3} \end{aligned}$$

$$A_1 = \left.\frac{2z^3 + z^2 - z + 4}{(z-2)^3}\right|_{z=0} = \underline{\frac{-1}{2}}$$

Now, find coefficient of repeated-root term with <u>highest power</u> denominator first.

Mult. $\frac{Y(z)}{z}$ by $(z-2)^3$:

$$\frac{2z^3 + z^2 - z + 4}{z} = \frac{A_1(z-2)^3}{z} + A_2(z-2)^2 + A_3(z-2) + A_4 \quad (*)$$

Eval. at $z = 2$ to find A_4:

$$A_4 = \frac{16 + 4 - 2 + 4}{2} = \underline{11}$$

Take deriv. of (*) to find A_3:

(Rewrite left-hand side of (*) as $2z^2 + z - 1 + 4z^{-1}$)

$$4z + 1 - 4z^{-2} = A_1 \bullet \underbrace{\text{MESS}}_{\text{will be zero when evaluated at } z=2} + 2A_2(z-2) + A_3 \qquad (**)$$

Eval. at $z = 2$ to give A_3:

$$\left[A_3 = 8 + 1 - 4\left(\frac{1}{4}\right) = \underline{8}\right]$$

To find A_2, take deriv. of (**)

$$4 + 8z^{-3} = A_1 \frac{d}{dz}\text{MESS} + 2A_2$$

Eval. at $z = 2$ to give A_2:

$$\left[A_2 = \frac{1}{2}\left[4 + 8\left(\frac{1}{8}\right)\right] = \frac{5}{2}\right]$$

$$\Rightarrow Y(z) = -\frac{1}{2} + \frac{\frac{5}{2}z}{z-2} + \frac{8z}{(z-2)^2} + \frac{11z}{(z-2)^3}$$

Now,

$$Z^{-1}\left\{-\frac{1}{2}\right\} = \begin{cases} -\dfrac{1}{2} & n = 0 \\ 0 & n \geq 1 \end{cases}$$

$$Z^{-1}\left\{\frac{\frac{5}{2}z}{z-2}\right\} = \frac{5}{2}\,2^n \quad n \geq 0$$

$$Z^{-1}\left\{\frac{8z}{(z-2)^2}\right\} = 8\left(\frac{1}{2}\right) n\,2^n \qquad n \geq 0 \qquad \text{(i)}$$

To find the inverse transform of the fourth term, use

$$\frac{1}{2}\,n\,(n-1)\,a^n \leftrightarrow \frac{a^2 z}{(z-a)^3}$$

$$\Rightarrow Z^{-1}\left\{\frac{11z}{(z-2)^3}\right\} = 11 \bullet \frac{1}{2}\,(n-1)\,n\,2^{n-2} \; n \geq 0 \qquad \text{(ii)}$$

Combining gives:

$$y_n = \begin{cases} 2 & n = 0 \\ \dfrac{1}{8}\left(11n^2 + 21n + 20\right)2^n & n \geq 1 \end{cases}$$

ECE 310 **DIGITAL SIGNAL PROCESSING** **D. Munson**
University of Illinois **Lecture 10**

Solution of D.E.'s Using z-Transform

Example

Solve example on p. 4.2, but using 1-sided z-transform.

$$y_n - 3y_{n-1} = 0 \qquad n \geq 0,\ y_{-1} = 2$$

Taking z-transform of both sides, and using Delay Property #2:

$$Y(z) - 3\,z^{-1}\left[Y(z) + z\,y_{-1}\right] = 0$$

$$\Rightarrow \quad Y(z)\left[1 - 3z^{-1}\right] = 6$$

$$\Rightarrow \quad Y(z) = \frac{6z}{z-3}$$

$$\Rightarrow \quad y_n = 6\,(3)^n\,u_n$$ ✔

Example

Solve example on p. 4.3, but using 1-sided z-transform.

$$y_n + 4\,y_{n-1} + 4\,y_{n-2} = 0 \qquad n \geq 0,\ y_{-1} = y_{-2} = 1$$

$$\Rightarrow \quad Y(z) + 4\,z^{-1}\left[Y(z) + z\,y_{-1}\right] + 4\,z^{-2}\left[Y(z) + z\,y_{-1} + z^2\,y_{-2}\right] = 0$$

$$\Rightarrow \quad Y(z)\left[1 + 4z^{-1} + 4z^{-2}\right] = -\,4\,y_{-1} - 4\,z^{-1}\,y_{-1} - 4\,y_{-2}$$

$$\Rightarrow \quad Y(z) = \frac{-8 - 4z^{-1}}{1 + 4z^{-1} + 4z^{-2}}$$

$$= \frac{-8z^2 - 4z}{z^2 + 4z + 4}$$

$$\frac{Y(z)}{z} = \frac{-8z - 4}{(z+2)^2} = \frac{A_1}{z+2} + \frac{A_2}{(z+2)^2}$$

To find A_2, note:

$$-8\,z - 4 = A_1(z+2) + A_2 \qquad (*)$$

$$\Rightarrow \quad A_2 = (-8\,z - 4)\Big|_{z=-2} = 12$$

Differentiate (*):

$$-8 = A_1$$

So

$$Y(z) = \frac{-8z}{z+2} + \frac{12z}{(z+2)^2}$$

Use $n\,a^n \leftrightarrow \dfrac{az}{(z-a)^2}$

$$\Rightarrow \boxed{y_n = -8(-2)^n - 6n(-2)^n \quad n \geq 0}$$ ✔

Example

$$y_{n+2} - \frac{3}{2}\,y_{n+1} + \frac{1}{2}\,y_n = \left(\frac{1}{3}\right)^n \qquad n \geq 0, \quad y_0 = 4, \quad y_1 = 0$$

Take z-transform of both sides and use Advance Property:

$$z^2\left(Y(z) - y_0 - z^{-1}y_1\right) - \frac{3}{2}z\left(Y(z) - y_0\right) + \frac{1}{2}Y(z) = \frac{z}{z - \frac{1}{3}}$$

Note: IC's automatically incorporated.

Have:

$$Y(z)\left[z^2 - \frac{3}{2}z + \frac{1}{2}\right] - 4z^2 + 6z = \frac{z}{z-\frac{1}{3}}$$

$$\Rightarrow Y(z) = \frac{1}{z^2 - \frac{3}{2}z + \frac{1}{2}}\left[\underbrace{\frac{z}{z-\frac{1}{3}}}_{\text{term due to input}} + \underbrace{4z^2 - 6z}_{\text{term due to IC's}}\right]$$

Write:

$$Y(z) = T_1(z) + T_2(z) \text{ with}$$

$$T_1(z) = \frac{z}{\left(z^2 - \frac{3}{2}z + \frac{1}{2}\right)\left(z - \frac{1}{3}\right)}$$

$$T_2(z) = \frac{4z^2 - 6z}{z^2 - \frac{3}{2}z + \frac{1}{2}}$$

$Z^{-1}[T_1(z)]$ is the zero-state response.

$Z^{-1}[T_2(z)]$ is the zero-input response.

Invert T_1 using PFE:

$$\frac{T_1(z)}{z} = \frac{1}{\left(z - \frac{1}{2}\right)(z-1)\left(z - \frac{1}{3}\right)} = \frac{A_1}{z - \frac{1}{2}} + \frac{A_2}{z-1} + \frac{A_3}{z - \frac{1}{3}}$$

Find:

$$A_1 = -12,\ A_2 = 3,\ A_3 = 9$$

$$\Rightarrow \qquad T_1(z) = \frac{-12z}{z - \frac{1}{2}} + \frac{3z}{z-1} + \frac{9z}{z - \frac{1}{3}}$$

So, zero-state response is

$$\left[y_x(n) \;=\; -12\left(\frac{1}{2}\right)^n + 3 + 9\left(\frac{1}{3}\right)^n \qquad n \geq 0 \right]$$

Invert T_2:

$$\frac{T_2(z)}{z} = \frac{4z-6}{\left(z - \frac{1}{2}\right)(z-1)} = \frac{B_1}{z - \frac{1}{2}} + \frac{B_2}{z-1}$$

Find:

$$B_1 = 8,\quad B_2 = -4$$

$$\Rightarrow \qquad T_2(z) = \frac{8z}{z - \frac{1}{2}} - \frac{4z}{z-1}$$

So, zero-input response is

$$\left[y_s(n) = 8\left(\frac{1}{2}\right)^n - 4 \quad n \geq 0 \right]$$

Total response:

$$\boxed{y(n) = y_x(n) + y_s(n) = -4\left(\frac{1}{2}\right)^n - 1 + 9\left(\frac{1}{3}\right)^n \qquad n \geq 0}$$

Comment: To solve D.E.'s of the form

$$y_n + a_1\, y_{n-1} \ldots + a_K\, y_{n-K} = x_n \qquad n \geq 0$$

$$y_i,\ i = -1, \ldots, -K$$

using z-transform, use Delay Property #2.

General Form of Solution of D.E.'s

In this section, we will derive the general form of a solution to a difference equation. We will prove that the zero-state response (response to the input) is given by a convolution. Consider

$$y(n+K) + a_1\, y(n+K-1) + \ldots + a_K\, y(n) = x(n) \qquad n \geq 0, \quad \text{IC's: } y(i)\ i = 0, 1, \ldots, K-1$$

Take z-transform. Use Advance Property:

$$z^K\left[Y(z) - \sum_{m=0}^{K-1} y(m) z^{-m}\right] + a_1\, z^{K-1}\left[Y(z) - \sum_{m=0}^{K-2} y(m) z^{-m}\right]$$

$$+ a_{K-1}\, z[Y(z) - y(0)] + a_K\, Y(z) \;=\; X(z)$$

Let:

$$S(z) \;=\; z^K \sum_{m=0}^{K-1} y(m) z^{-m} + a_1\, z^{K-1} \sum_{m=0}^{K-2} y(m) z^{-m}$$

$$+ \ldots + a_{K-1}\, z\, y(0)$$

$$\Rightarrow \qquad Y(z)\, \underbrace{[z^K + a_1\, z^{K-1} + \ldots + a_K]}_{\text{char. eqn.}} \;=\; X(z) + S(z)$$

$$\text{Let } H(z) \;=\; \frac{1}{z^K + a_1 z^{K-1} + \ldots + a_K}$$

$$\Rightarrow \qquad Y(z) \;=\; H(z)\, [\underset{\substack{\downarrow \\ \text{term due} \\ \text{to input}}}{X(z)} + \underset{\substack{\downarrow \\ \text{term due} \\ \text{to IC's}}}{S(z)}]$$

Notice that the decomposition property holds with

$$\begin{cases} y_s(n) = Z^{-1}\ \{H(z)\, S(z)\} \\ y_x(n) = Z^{-1}\ \{H(z)\, X(z)\} \end{cases}$$

Note: Both homogeneity and superposition hold with respect to y_s and y_x because the z-transform is linear.

$\Rightarrow$ D.E. describes a linear system.
What is the form of $y_s(n)$?

Suppose roots of char. eq. are distinct. Then:

$$\frac{S(z)H(z)}{z} = \frac{B_1}{z-r_1} + \frac{B_2}{z-r_2} + \ldots + \frac{B_K}{z-r_K}$$

z is factor in S(z), so don't need $\frac{B_0}{z}$ term.

$$\Rightarrow \quad S(z)\, H(z) = \frac{B_1 z}{z-r_1} + \ldots + \frac{B_K z}{z-r_K}$$

$$\Rightarrow \quad \left[y_s(n) = \sum_{i=1}^{K} B_i\,(r_i)^n \quad n \geq 0 \right]$$ ~ same form as y_H, already know this from sequence-domain solution.

What is form of $y_x(n)$?

Since $y_x(n) = Z^{-1}\{H(z)\, X(z)\}$, PFE shows that $y_x(n)$ will involve terms <u>both</u> in y_s and in x(n).

Can also rewrite y_x using the convolution property:

$$\left[y_x(n) = \sum_{m=0}^{n} h(m)x(n-m) \right] \qquad (\lozenge)$$

$$\text{where } h(m) = Z^{-1}\{H(z)\} = Z^{-1}\left\{ z\frac{H(z)}{z} \right\}$$

$$= Z^{-1}\left\{ z\left(\frac{D_0}{z} + \frac{D_1}{z-r_1} + \ldots + \frac{D_K}{z-r_K} \right) \right\}$$

$$= \begin{cases} D_0 + \sum_{i=1}^{K} D_i \left(r_i\right)^n & n = 0 \\ \sum_{i=1}^{K} D_i \left(r_i\right)^n & n \geq 1 \end{cases}$$

same <u>form</u> as y_H and y_s for $n \geq 1$.

So, $y_x(n)$ is given by a convolution of the input with $h(n) = Z^{-1}\{H(z)\}$.

What is $\{h(n)\}_{n=0}^{\infty}$? Can be interpreted as the system unit pulse response (u.p.r.) <u>assuming zero IC's</u>.

<u>Definition</u>

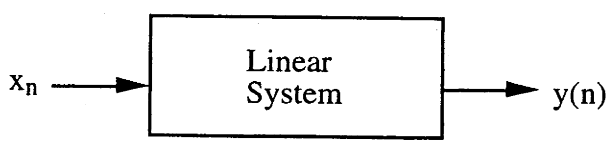
Unit-pulse sequence is

$$\delta_n = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases}.$$

From (◊), the system response to a unit pulse is:

$$\underset{\substack{\downarrow \\ \text{zero IC's}}}{y(n)} = y_x(n) = \sum_{m=0}^{n} h(m)\delta_{n-m} = h(n) \quad \checkmark$$

Example



Suppose if $x(n) = \delta(n)$ with zero IC's then $y(n) = a^n$ for $n \geq 0$. Assuming zero IC's, find y(n) due to $x(n) = b^n$ $n \geq 0$.

Solution:

Given $h(n) = a^n$ $n \geq 0$.

$y(n) = y_x(n)$ since have zero IC's.

$$(\lozenge) \Rightarrow y(n) = \sum_{m=0}^{n} a^m b^{n-m} = b^n \sum_{m=0}^{n} \left(\frac{a}{b}\right)^m$$

$$= \frac{b^{n+1}}{b} \frac{1-\left(\frac{a}{b}\right)^{n+1}}{1-\frac{a}{b}} \quad (a \neq b)$$

$$= \frac{b^{n+1} - a^{n+1}}{b-a}$$

Comments:

1) Can show D.E. sol'n has same or similar form if right-hand side of D.E. is a linear combination of shifted x_n, e.g., $x_n - 2\,x_{n-1}$.

2) Previous D.E. sol'n methods and def. of 1-sided z-transform can be modified for input at $n \neq 0$.

3) Usually interested in only the zero-state response, $y_x(n)$.
Rationale:

Either

a) All IC's are zero $(\Rightarrow y_s(n) = 0)$

or

b) Have stable system $(|r_i| < 1)$ and input applied long ago so that $y_s(n)$ has nearly decayed away for n of interest and $y(n) \approx y_x(n)$.

From now on, we will assume zero IC's $(y(n) = y_x(n))$ unless stated otherwise.

convolution

EE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 11

D. Munson

Response Due to Input Applied at Arbitrary Time (zero IC's)

Constant coeff. D.E.'s describe shift-invariant systems.

Recall that a system is <u>shift-invariant</u> if for any n_o:

$$\{x_n\}_{n=-\infty}^{\infty} \longrightarrow \boxed{\text{System}} \longrightarrow \{y_n\}_{n=-\infty}^{\infty}$$

$$\Rightarrow \left\{x_{n-n_0}\right\}_{n=-\infty}^{\infty} \longrightarrow \boxed{\text{System}} \longrightarrow \left\{y_{n-n_0}\right\}_{n=-\infty}^{\infty}$$

For input applied at n = 0 had

$$y_n = \sum_{m=0}^{n} h_m \, x_{n-m}$$

u.p.r.

Consider more general case of <u>linear</u>, <u>shift-invariant</u> (LSI) system with:

1) Input not necessarily applied at n = 0.

2) Output y_n can depend on future values of input x_n, e.g.,

   $y_n - y_{n-1} = x_n + 3\,x_{n+1}$

   Image processing is an example where current outputs often depend on "future" values of the input.

   Result for general case:

$$\boxed{y_n = \sum_{m=-\infty}^{\infty} h_m \, x_{n-m} = \sum_{m=-\infty}^{\infty} x_m \, h_{n-m}} \qquad (*)$$

   Write $y_n = h_n * x_n$ where h_n is system u.p.r.

   Can show (*) using modified z-transform approach. But, easy to show in sequence domain using the properties:

   1) Linearity
   2) Shift-invariance

Proof involves writing the input as ... $+ x_{-1}\,\delta_{n+1} + x_o\,\delta_n + x_1\,\delta_{n-1} + x_2\,\delta_{n-2} +$ ... and noting that the output is simply the sum of the outputs due to these individual terms.

Proof:

Think of input as a sum of weighted and shifted unit-pulse sequences:

$$\{x_n\}_{n=-\infty}^{\infty} = \cdots + x_{-1}\,\delta_{n+1} + x_0\,\delta_n + x_1\,\delta_{n-1} + x_2\,\delta_{n-2} + \cdots$$

$$= \sum_{m=-\infty}^{\infty} x_m\,\delta_{n-m}$$

But, by shift invariance,

$$\{\delta_{n-m}\}_{n=-\infty}^{\infty} \rightarrow \{h_{n-m}\}_{n=-\infty}^{\infty} \qquad \text{(m fixed)}$$

and by homogeneity,

$$\{x_m\,\delta_{n-m}\}_{n=-\infty}^{\infty} \rightarrow \{x_m\,h_{n-m}\}_{n=-\infty}^{\infty}$$

and by superposition,

$$\left\{\sum_{m=-\infty}^{\infty} x_m\,\delta_{n-m}\right\}_{n=-\infty}^{\infty} \rightarrow \left\{\sum_{m=-\infty}^{\infty} x_m\,h_{n-m}\right\}_{n=-\infty}^{\infty} \qquad \checkmark$$

Recall that a system is <u>causal</u> if for any n_o, $y(n_o)$ depends only on $x(n)$, $n \leq n_o$.

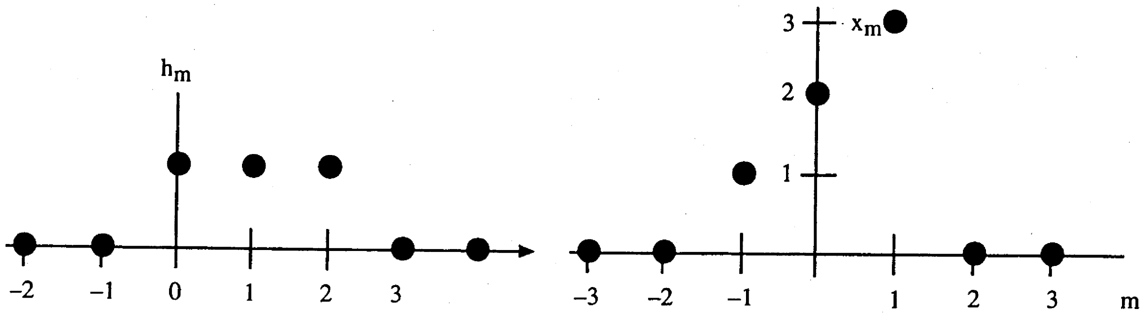
Easy to show LSI system is causal iff h_m in (*) is zero for $m < 0$.

In this case, the convolution formula reduces to

$$y_n = \sum_{m=0}^{\infty} h_m\,x_{n-m} = \sum_{m=-\infty}^{n} x_m\,h_{n-m}$$

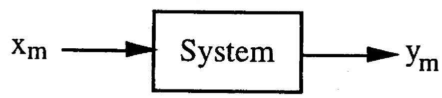
Graphical View of Convolution

Example Given



Find $y_n = \sum_{m=-\infty}^{\infty} h_m x_{n-m}$

To plot x_{n-m} versus m:



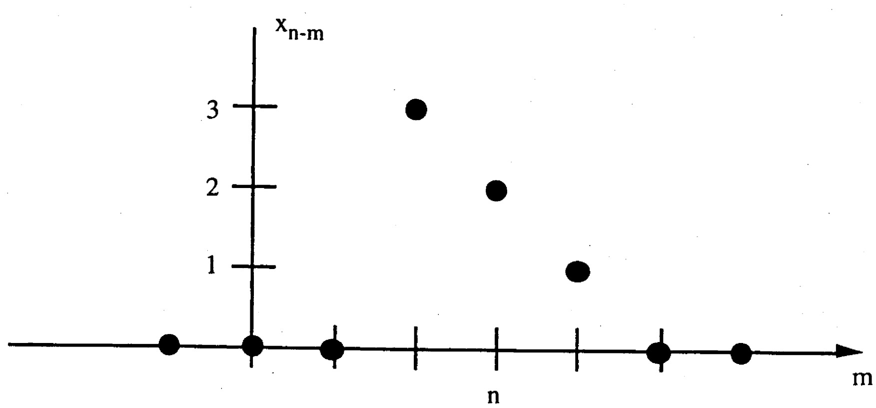
i) Reverse $\{x_m\}_{m=-\infty}^{\infty}$

$\{x_m\} \rightarrow \{x_{-m}\}$

ii) Shift $\{x_{-m}\}$ to right n places

$\{x_{-m}\} \rightarrow \{x_{-(m-n)}\} = \{x_{n-m}\}$ ✓

Result:



From graphs see

$y_n = 0 \qquad n < -1$

$y_{-1} = 1 \cdot 1 = 1$

$y_0 = 2 \cdot 1 + 1 \cdot 1 = 3$

$y_1 = 3 \cdot 1 + 2 \cdot 1 + 1 \cdot 1 = 6$

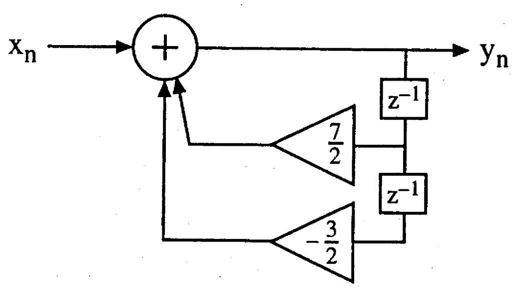
$y_2 = 3 \cdot 1 + 2 \cdot 1 = 5$

$y_3 = 3 \cdot 1 = 3$

$y_n = 0 \qquad n > 3$

Example

Given $x_n = \left(\frac{1}{4}\right)^n u_n$, find the output of the system below:



(zero ICs)

System is described by

$$y_n = \frac{7}{2} y_{n-1} - \frac{3}{2} y_{n-2} + x_n$$

or

$$y_n - \frac{7}{2} y_{n-1} + \frac{3}{2} y_{n-2} = \left(\frac{1}{4}\right)^n \qquad n \geq 0,\ y_{-1} = y_{-2} = 0$$

We could solve this using the classical solution method. Instead, use

$$y_n = \sum_{m=0}^{\infty} h_m x_{n-m}. \qquad (*)$$

How do we find the unit pulse response? One way is to find h_n in the sequence-domain by solving an appropriate D.E. $\{h_n\}_{n=0}^{\infty}$ is <u>defined</u> to be the solution to

$$h_n - \frac{7}{2} h_{n-1} + \frac{3}{2} h_{n-2} = \delta_n \qquad n \geq 0,\ h_{-1} = h_{-2} = 0 \qquad (**)$$

How do we solve this D.E? The input term has a form that changes with n. How do we select $y_p(n)$? We can avoid this question by noting

$$h_n - \frac{7}{2}h_{n-1} + \frac{3}{2}h_{n-2} = 0 \qquad n \geq 1$$

But, now for ICs need h_0, h_{-1}. To find h_0 just use (**):

$$h_0 - \frac{7}{2}h_{-1} + \frac{3}{2}h_{-2} = 1$$

$$\Rightarrow \quad h_0 = 1$$

Now, to find $\{h_n\}_{n=1}^{\infty}$, solve

$$h_n - \frac{7}{2}h_{n-1} + \frac{3}{2}h_{n-2} = 0 \qquad n \geq 1 \qquad h_0 = 1,\ h_{-1} = 0$$

$$z^2 - \frac{7}{2}z + \frac{3}{2} = 0$$

$$\Rightarrow \quad \left(z - \frac{1}{2}\right)(z-3) = 0$$

$$\Rightarrow \quad h_n = c_1\left(\frac{1}{2}\right)^n + c_2(3)^n$$

Apply IC's:

$$h_0 = 1 = c_1 + c_2$$

$$h_{-1} = 0 = 2c_1 + \frac{1}{3}c_2$$

$$\Rightarrow \quad c_1 = \frac{-1}{5} \qquad c_2 = \frac{6}{5}$$

$$\Rightarrow \quad h_n = \frac{-1}{5}\left(\frac{1}{2}\right)^n + \frac{6}{5}(3)^n \qquad n \geq 0$$

We know $h_n = 0$ for $n < 0$ since $h_{-1} = h_{-2} = 0$.

$$\Rightarrow \quad h_n = \left[-\frac{1}{5}\left(\frac{1}{2}\right)^n + \frac{6}{5}(3)^n\right]u_n \qquad (***)$$

Now,

$$y_n = \sum_{m=0}^{\infty} h_m x_{n-m}$$

$$= \sum_{m=0}^{\infty} \left[-\frac{1}{5}\left(\frac{1}{2}\right)^m u_m + \frac{6}{5} 3^m u_m \right] \left(\frac{1}{4}\right)^{n-m} u_{n-m}$$

$$= \sum_{m=0}^{n} \left[-\frac{1}{5}\left(\frac{1}{2}\right)^m + \frac{6}{5}(3)^m \right] \left(\frac{1}{4}\right)^{n-m}$$

$$= -\frac{1}{5} \sum_{m=0}^{n} \left(\frac{1}{2}\right)^m \left(\frac{1}{4}\right)^{n-m} + \frac{6}{5} \sum_{m=0}^{n} (3)^m \left(\frac{1}{4}\right)^{n-m}$$

$$= -\frac{1}{5}\left(\frac{1}{4}\right)^n \sum_{m=0}^{n} 2^m + \frac{6}{5}\left(\frac{1}{4}\right)^n \sum_{m=0}^{n} (12)^m$$

$$= -\frac{1}{5}\left(\frac{1}{4}\right)^n \frac{1-2^{n+1}}{1-2} + \frac{6}{5}\left(\frac{1}{4}\right)^n \frac{1-12^{n+1}}{1-12}$$

$$= \frac{1}{5}\left(\left(\frac{1}{4}\right)^n - 2\left(\frac{1}{2}\right)^n\right) - \frac{6}{55}\left(\left(\frac{1}{4}\right)^n - 12\,(3)^n\right)$$

$$= \boxed{\frac{1}{11}\left(\frac{1}{4}\right)^n - \frac{2}{5}\left(\frac{1}{2}\right)^n + \frac{72}{55}(3)^n}$$

Notes:

1) It was quite a bit of work to find $\{h_n\}_{n=0}^{\infty}$, but once you know $\{h_n\}$, you can find the output due to <u>any</u> input, via the convolution formula.

2) We will later develop an easier way to find $\{h_n\}_{n=0}^{\infty}$, using the z-transform.

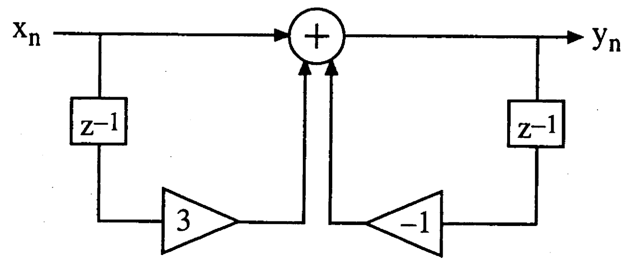
ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 12

D. Munson

Example

Find the unit pulse response $\{h_n\}$ for the system below.



$$y_n = -y_{n-1} + x_n + 3\,x_{n-1}$$

For unit-pulse input with zero IC's have

$$h_n + h_{n-1} = \delta_n + 3\,\delta_{n-1} \qquad n \geq 0,\ \ h_{-1} = 0$$

$$\Rightarrow \qquad h_n + h_{n-1} = 0 \qquad n \geq 2$$

But, need to find h_0, h_1:

$$h_0 = -h_{-1} + \delta_0 + 3\,\delta_{-1} = 1$$

$$h_1 = -h_0 + \delta_1 + 3\,\delta_0 = 2$$

For h_n, $n \geq 2$, solve

$$h_n + h_{n-1} = 0 \qquad n \geq 2,\ h_1 = 2$$

Char. eqn: $z + 1 = 0$ root $= -1$

$$h_n = c(-1)^n$$

IC: $h_1 = 2 = -c \Rightarrow c = -2$

$$\Rightarrow \qquad h_n = -2(-1)^n \quad n \geq 1$$

Note that this solution holds for $n \geq 1$ because the D.E. we were solving holds for $n \geq 2$ and then we forced the solution to hold at $n = 1$ by applying the IC.

Have $h_n = 0$, $n < 0$ since started at zero initial state.

(Can check by iterating D.E. backward for zero IC with input = unit pulse)

$$\Rightarrow h_n = \begin{cases} 0 & n < 0 \\ 1 & n = 0 \\ -2(-1)^n & n \geq 1 \end{cases}$$

The mechanics of discrete-time convolution are the same as for continuous convolution, except have sums instead of integrals. Illustrate by examples.

Example

$$h_n = \left(\frac{1}{4}\right)^n u_n$$

$$x_n = u_{n-7}$$

Find $y_n = h_n * x_n$

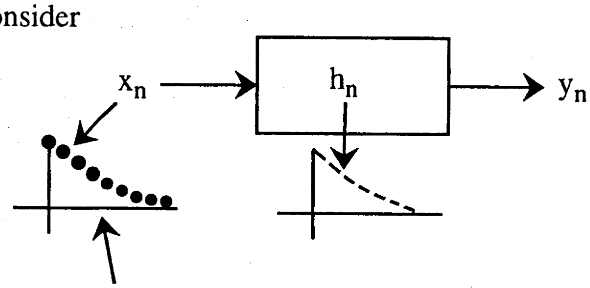
Before solving, have a question.

Know

$$y_n = \sum_m h_m x_{n-m}$$

but why does x_n have to be flipped around?

Consider



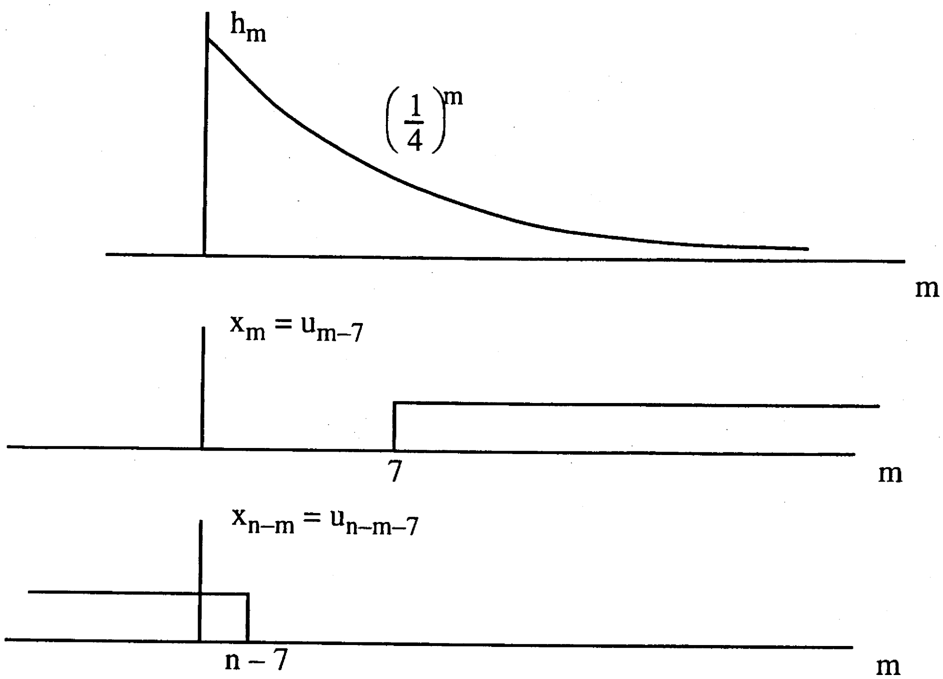
Which sample of $\{x_n\}$ goes into the system first?

Answer: The n = 0 term! x_n enters the system in the reverse order from how it is plotted. This is why $\{x_n\}$ must be flipped around in computing a convolution.

Now, back to the convolution example. Have

$$y_n = \sum_{m=0}^{\infty} h_m x_{n-m}$$

Note: Always draw pictures when working out a convolution. For our example we have:



Case 1 $\quad n-7<0 \Rightarrow n<7$

$y_n = 0$

Case 2 $\quad n-7 \geq 0 \Rightarrow n \geq 7$

$$y_n = \sum_{m=0}^{n-7} \left(\frac{1}{4}\right)^m = \frac{1-\left(\frac{1}{4}\right)^{n-6}}{1-\frac{1}{4}}$$

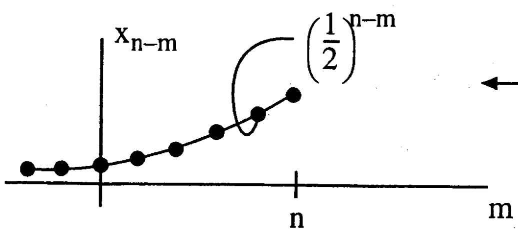
$$= \frac{4}{3}\left[1-\left(\frac{1}{4}\right)^{n-6}\right]$$

Example

$$x_n = \begin{cases} \left(\frac{1}{2}\right)^n & n \geq 0 \\ 0 & n<0 \end{cases} \qquad h_n = \begin{cases} \left(\frac{1}{4}\right)^n & n \geq 0 \\ 0 & n<0 \end{cases}$$

$$y_n = \sum_{m=-\infty}^{\infty} h_m x_{n-m} = \sum_{m=0}^{\infty} \left(\frac{1}{4}\right)^m x_{n-m}$$

$$x_{n-m} = \begin{cases} \left(\frac{1}{2}\right)^{n-m} & n-m \ge 0 \\ 0 & n-m < 0 \end{cases} = \begin{cases} \left(\frac{1}{2}\right)^{n-m} & m \le n \\ 0 & m > n \end{cases}$$



Can always get x_{n-m} versus m by flipping x_m around origin and shifting to right by n.

<u>n < 0</u>

$\Rightarrow \quad y_n = 0$

<u>n ≥ 0</u>

$$y_n = \sum_{m=0}^{n}\left(\frac{1}{4}\right)^m \left(\frac{1}{2}\right)^{n-m} = \left(\frac{1}{2}\right)^n \sum_{m=0}^{n}\left(\frac{1}{4}\right)^m 2^m = \left(\frac{1}{2}\right)^n \sum_{m=0}^{n}\left(\frac{1}{2}\right)^m$$

$$= \left(\frac{1}{2}\right)^n \frac{1-\left(\frac{1}{2}\right)^{n+1}}{1-\frac{1}{2}}$$

$$= 2\left[\left(\frac{1}{2}\right)^n - \left(\frac{1}{2}\right)^{2n+1}\right] = 2\left[\left(\frac{1}{2}\right)^n - \frac{1}{2}\left(\frac{1}{4}\right)^n\right]$$

$$= \boxed{2\left(\frac{1}{2}\right)^n - \left(\frac{1}{4}\right)^n}$$

Example

$$x_n = u_n - u_{n-N}$$

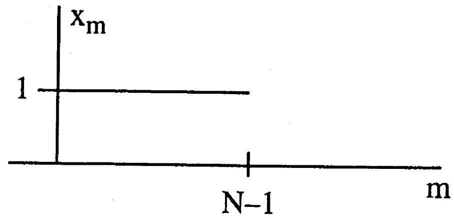
$$h_n = a^n u_n$$

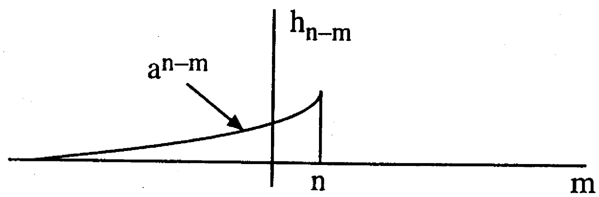
Find $y_n = h_n * x_n$ via

a) Convolution formula

b) 1-sided z-transform method (applicable since both $\{x_n\}$ and $\{h_n\}$ are right-sided).

a) $y_n = \sum_{m=-\infty}^{\infty} x_m h_{n-m}$





<u>Case 1</u> $n < 0$

$y_n = 0$

<u>Case 2</u> $0 \le n \le N-1$

$$y_n = \sum_{m=0}^{n} 1\, a^{n-m} = a^n \sum_{m=0}^{n} \left(\frac{1}{a}\right)^m = a^n \frac{1-\left(\frac{1}{a}\right)^{n+1}}{1-\frac{1}{a}} = \frac{a^{n+1}-1}{a-1}$$

<u>Case 3</u> $n \ge N$

$$y_n = \sum_{m=0}^{N-1} 1\, a^{n-m} = a^n \sum_{m=0}^{N-1} \left(\frac{1}{a}\right)^m = a^n \frac{1-\left(\frac{1}{a}\right)^{N}}{1-\frac{1}{a}}$$

$$= \frac{a^{n+1}-a^{n-N+1}}{a-1}$$

Summarizing:

$$y_n = \begin{cases} 0 & n < 0 \\ \dfrac{a^{n+1}-1}{a-1} & 0 \le n \le N-1 \\ \dfrac{a^{n+1}-a^{n-N+1}}{a-1} & n \ge N \end{cases}$$

b) By Delay Property #1:

$$X(z) = \frac{z}{z-1} - z^{-N}\frac{z}{z-1}$$

$$H(z) = \frac{z}{z-a}$$

$$\Rightarrow Y(z) = \frac{z}{z-a}\left[\frac{z}{z-1} - z^{-N}\frac{z}{z-1}\right]$$

Now, we are faced with a decision. We might consider writing

$$Y(z) = \frac{z}{z-a}\left[\frac{z-z^{1-N}}{z-1}\right] = \frac{z^N - 1}{z^{N-2}(z-a)(z-1)}$$

However, a PFE of Y(z) will then have the form

$$\frac{Y(z)}{z} = \frac{A_1}{z-a} + \frac{A_2}{z-1} + \frac{B_1}{z} + \frac{B_2}{z^2} + \frac{B_3}{z^3} + \ldots + \frac{B_{N-1}}{z^{N-1}}$$

It looks like too much work to find the B_i. Instead, note

$$Y(z) = \overline{Y}(z) - z^{-N}\,\overline{Y}(z)$$

where

$$\overline{Y}(z) = \frac{z^2}{(z-a)(z-1)}$$

Then, by Delay Property #1:

$$y_n = \overline{y}_n - \overline{y}_{n-N}\,u_{n-N}$$

Let's try this approach:

$$\frac{\overline{Y}(z)}{z} = \frac{z}{(z-a)\,(z-1)} = \frac{A}{z-a} + \frac{B}{z-1}$$

$$A = \frac{a}{a-1}\ ,\quad B = \frac{1}{1-a}$$

$$\Rightarrow\ \overline{y}_n = \frac{a}{a-1}\,a^n\,u_n + \frac{1}{1-a}\,u_n$$

Now,

$$y_n = \bar{y}_n - \bar{y}_{n-N} = \begin{cases} 0 & n > 0 \\ \bar{y}_n & 0 \le n \le N-1 \\ \bar{y}_n - \bar{y}_{n-N} & n \ge N \end{cases}$$

$$= \begin{cases} 0 & n < 0 \\ \dfrac{a^{n+1} - 1}{a - 1} & 0 \le n \le N-1 \\ \dfrac{a^{n+1} - a^{n-N+1}}{a - 1} & n \ge N \end{cases}$$

which agrees with our result obtained using the convolution formula.

ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 13

D. Munson

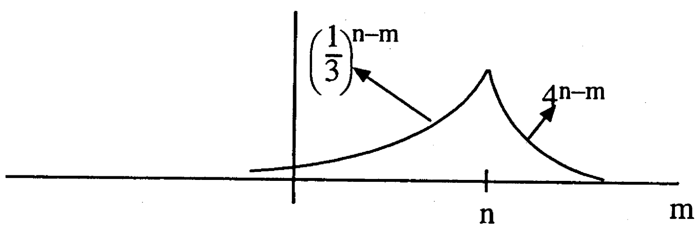
Consider a 2-sided convolution.

Example $h_n = u_{n-1}$ ← step sequence

$$x_n = \begin{cases} \left(\frac{1}{3}\right)^n & n \geq 0 \\ 4^n & n < 0 \end{cases}$$

$$y_n = \sum_{m=-\infty}^{\infty} h_m x_{n-m} = \sum_{m=-\infty}^{\infty} u_{m-1} x_{n-m} = \sum_{m=1}^{\infty} x_{n-m}$$

$$x_{n-m} = \begin{cases} \left(\frac{1}{3}\right)^{n-m} & n-m \geq 0 \\ 4^{n-m} & n-m < 0 \end{cases} = \begin{cases} \left(\frac{1}{3}\right)^{n-m} & m \leq n \\ 4^{n-m} & m > n \end{cases}$$



<u>n < 1</u>

$$y_n = \sum_{m=1}^{\infty} 4^{n-m} = 4^n \sum_{m=1}^{\infty} \left(\frac{1}{4}\right)^m$$

$$= 4^n \left[-1 + \sum_{m=0}^{\infty} \left(\frac{1}{4}\right)^m\right]$$

$$= 4^n \left[-1 + \frac{1}{1-\frac{1}{4}}\right]$$

$$= \frac{1}{3} 4^n$$

<u>$n \geq 1$</u>

$$y_n = \sum_{m=1}^{n} \left(\frac{1}{3}\right)^{n-m} + \sum_{m=n+1}^{\infty} 4^{n-m}$$

$$= \left(\frac{1}{3}\right)^n \left[-1 + \sum_{m=0}^{n} 3^m\right] + 4^n \left[\sum_{m=0}^{\infty} \left(\frac{1}{4}\right)^m - \sum_{m=0}^{n} \left(\frac{1}{4}\right)^m\right]$$

$$= \left(\frac{1}{3}\right)^n \left[-1 + \frac{1-3^{n+1}}{1-3}\right] + 4^n \left[\frac{1}{1-\frac{1}{4}} - \frac{1-\left(\frac{1}{4}\right)^{n+1}}{1-\frac{1}{4}}\right]$$

$$= -\left(\frac{1}{3}\right)^n - \frac{1}{2}\left(\frac{1}{3}\right)^n + \frac{3}{2} + \frac{4}{3}4^n - \frac{4}{3}4^n + \frac{1}{3}$$

$$= -\frac{3}{2}\left(\frac{1}{3}\right)^n + \frac{11}{6}$$

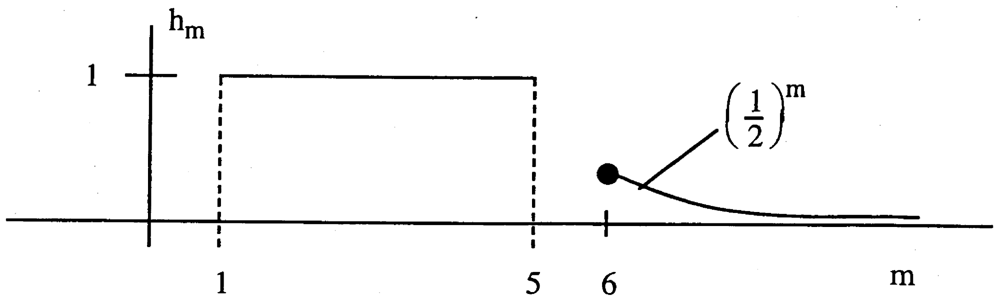
$$\Rightarrow \boxed{y_n = \begin{cases} \frac{1}{3}4^n & n < 1 \\ \frac{11}{6} - \frac{3}{2}\left(\frac{1}{3}\right)^n & n \geq 1 \end{cases}}$$

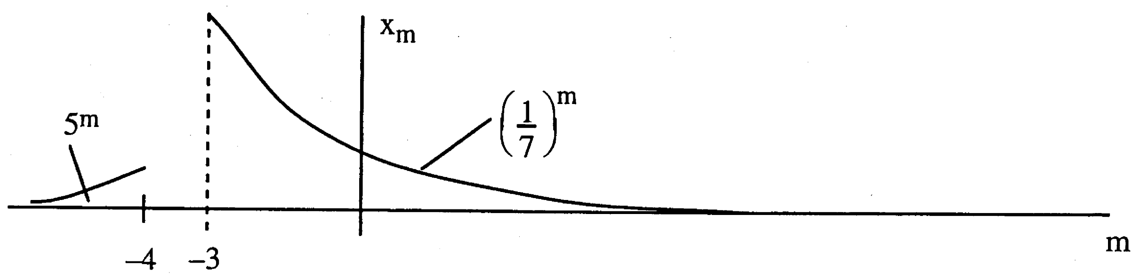
Example

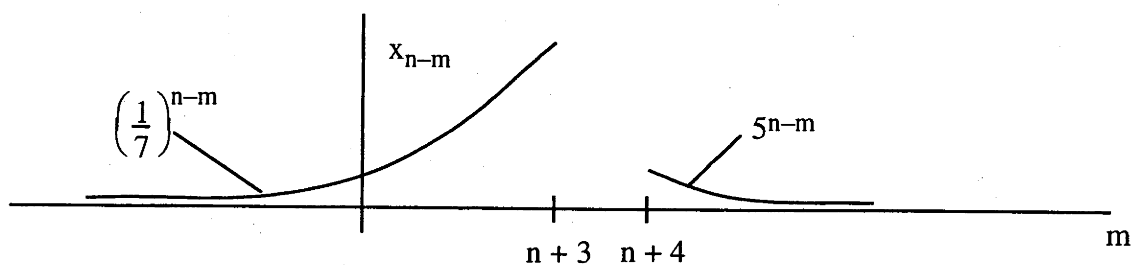
$$h_n = \begin{cases} 0 & n \leq 0 \\ 1 & 1 \leq n \leq 5 \\ \left(\frac{1}{2}\right)^n & n \geq 6 \end{cases}$$

$$x_n = \begin{cases} 5^n & n \leq -4 \\ \left(\frac{1}{7}\right)^n & n > -4 \end{cases}$$

$$y_n = \sum_{m=-\infty}^{\infty} h_m x_{n-m}$$







Case 1 $n + 4 \le 1 \Rightarrow n \le -3$

$$y_n = \sum_{m=1}^{5} 1 \bullet 5^{n-m} + \sum_{m=6}^{\infty} \left(\frac{1}{2}\right)^m 5^{n-m}$$

$$= \ldots$$

Case 2 $1 < n + 4 \le 5 \Rightarrow -3 < n \le 1$

$$y_n = \sum_{m=1}^{n+3} 1 \bullet \left(\frac{1}{7}\right)^{n-m} + \sum_{m=n+4}^{5} 1 \bullet 5^{n-m} + \sum_{m=6}^{\infty} \left(\frac{1}{2}\right)^m 5^{n-m}$$

Case 3 $n + 4 > 5 \Rightarrow n > 1$

$$y_n = \sum_{m=1}^{5} 1 \bullet \left(\frac{1}{7}\right)^{n-m} + \sum_{m=6}^{n+3} \left(\frac{1}{2}\right)^{m} \left(\frac{1}{7}\right)^{n-m} + \sum_{m=n+4}^{\infty} \left(\frac{1}{2}\right)^{m} 5^{n-m}$$

(Treat $n = 2$ as subcase of Case 3, where second sum is defined to be zero.)

ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 14

D. Munson

Two-Sided z-Transform

Definition

$$\underbrace{Y(z)}_{\text{complex variable}} = \sum_{n=-\infty}^{\infty} y_n z^{-n}, \quad z \varepsilon ROC_Y$$

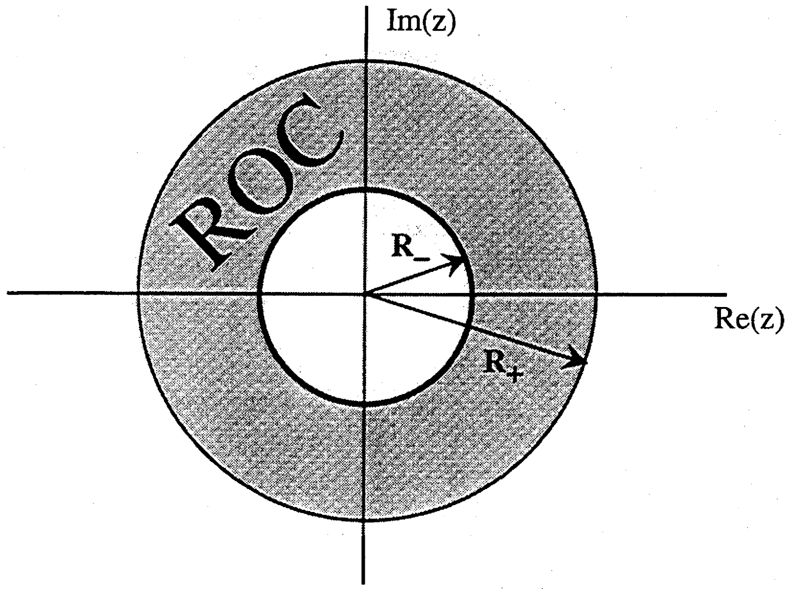
Example

$$y_n = \begin{cases} a^n & n \geq 0 \\ b^n & n < 0 \end{cases}$$

$$\begin{aligned}
Y(z) &= \sum_{n=-\infty}^{-1} b^n z^{-n} + \sum_{n=0}^{\infty} a^n z^{-n} \\
&= \sum_{k=1}^{\infty} b^{-k} z^k + \sum_{n=0}^{\infty} \left(\frac{a}{z}\right)^n \\
&= -1 + \sum_{k=0}^{\infty} \left(\frac{z}{b}\right)^k + \underbrace{\frac{1}{1-\frac{a}{z}}}_{\text{if } \left|\frac{a}{z}\right| < 1 \;\Rightarrow\; |z| > |a|} \\
&= -1 + \underbrace{\frac{1}{1-\frac{z}{b}}}_{\text{if } \left|\frac{z}{b}\right| < 1 \;\Rightarrow\; |z| < |b|} + \frac{z}{z-a} \\
&= -1 + \frac{-b}{z-b} + \frac{z}{z-a}
\end{aligned}$$

$$= \frac{-z}{z-b} + \frac{z}{z-a} \qquad |a| < |z| < |b|$$

In general, $Y(z)$ is defined in a ring:



In our example,

$R_- = |a|, \quad R_+ = |b|$

If $|a| \geq |b|$ then ROC $= \phi$ and z-transform is undefined (i.e. is infinite) for all z.

Why is ROC a ring?

$$Y(z) = \underbrace{\sum_{n=-\infty}^{-1} y_n z^{-n}}_{\substack{\text{converges for} \\ z \text{ small enough} \\ \text{i.e., for } |z| < R_+}} + \underbrace{\sum_{n=0}^{\infty} y_n z^{-n}}_{\substack{\text{converges for} \\ z \text{ large enough, i.e.,} \\ |z| > R_-}}$$

R_- determined by $\{y_n\}_{n=1}^{\infty}$

R_+ determined by $\{y_n\}_{n=-\infty}^{-1}$

If $y_n = 0, \ n < 0$, then

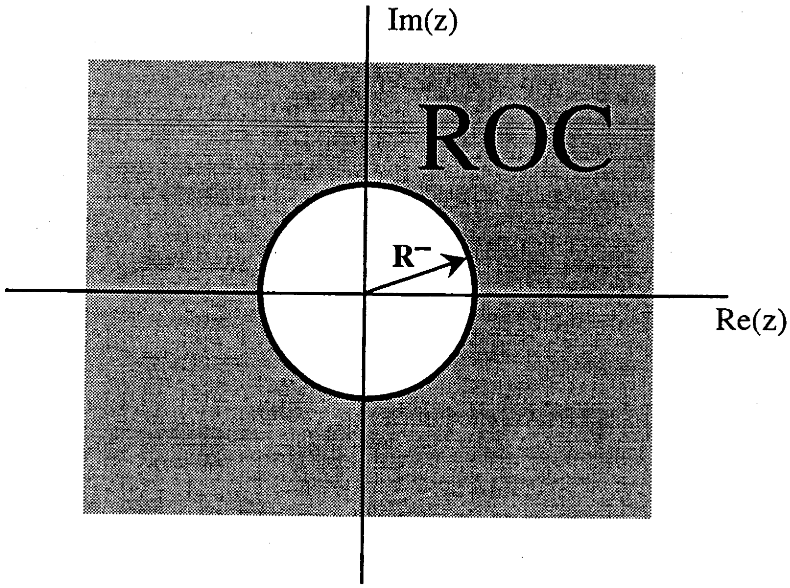
$$Y(z) = \sum_{n=0}^{\infty} y_n z^{-n} \text{ and } R+ = \infty$$

$\uparrow$

1-sided z-transform

so that

ROC:

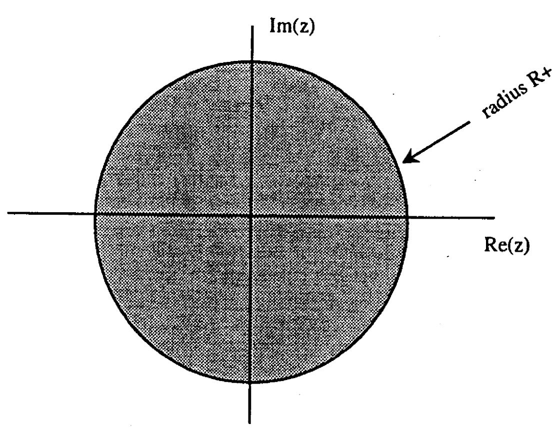


If $y_n = 0$, $n > 0$, then

$$Y(z) = \sum_{n=-\infty}^{0} y_n z^{-n} \text{ and } R_- = 0$$

so that

ROC:



Fact: Must state ROC for $Y(z)$ to correspond to a

unique $\{y_n\}_{n=-\infty}^{\infty}$!

Example

$$x_n = \begin{cases} -(a^n) & n < 0 \\ 0 & n \geq 0 \end{cases}$$

$$y_n = \begin{cases} 0 & n < 0 \\ a^n & n \geq 0 \end{cases}$$

$$\Rightarrow X(z) = -\sum_{n=-\infty}^{-1} a^n z^{-n}$$

$$= -\sum_{k=1}^{\infty} a^{-k} z^k$$

$$= 1 - \sum_{k=0}^{\infty} \left(\frac{z}{a}\right)^k$$

$$= 1 - \frac{1}{1 - \frac{z}{a}}, \quad \left|\frac{z}{a}\right| < 1$$

$$X(z) = \frac{z}{z-a} \qquad |z| < |a|$$

Similarly,

$$Y(z) = \frac{z}{z-a} \qquad |z| > |a|$$

So, algebraic form of $X(z)$ and $Y(z)$ are identical, but they are not the same functions since they are defined on completely different regions.

Fact: $(Y(z), ROC_Y)$ uniquely specify $\{y_n\}_{n=-\infty}^{\infty}$.

Definition

$\{z : Y(z) = 0\}$ are called <u>zeros</u> of $Y(z)$.

$\{z : Y(z) = \infty\}$ are called <u>poles</u> of $Y(z)$.

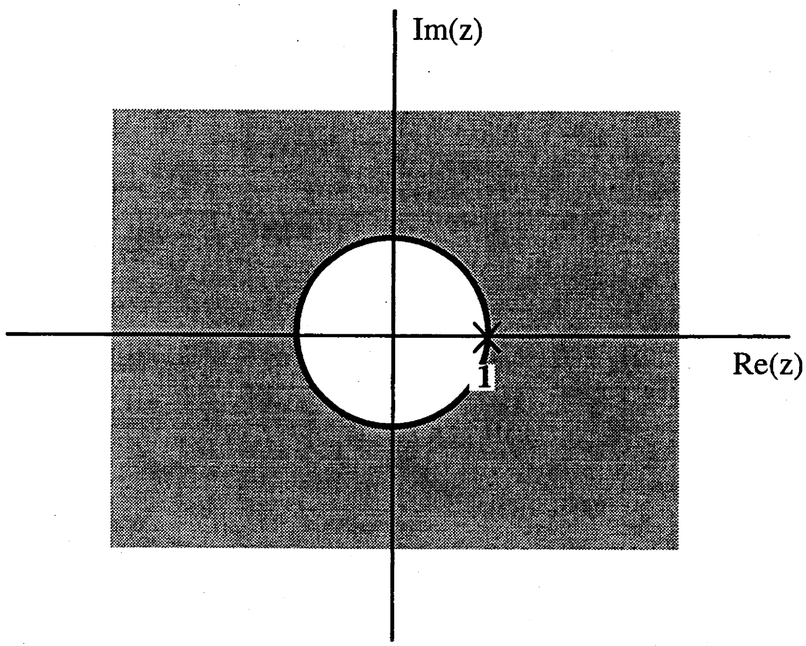
Fact: ROC_Y is bounded by poles (with no poles in the ROC).

Example

$$Y(z) = \frac{z}{z-1} \qquad |z| > 1$$

Pole is at $z = 1$.

$y_n = u_n$.



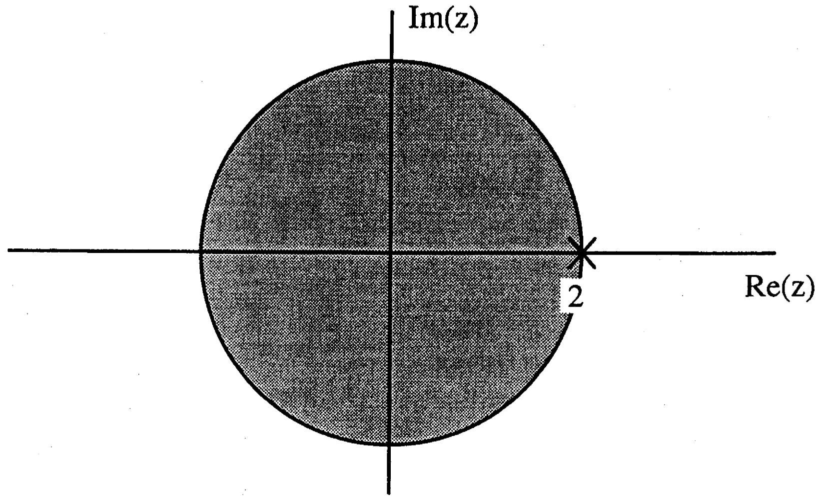
Example

$$Y(z) = \frac{z}{z-2} \qquad |z| < 2$$

Pole is at 2.

$$y_n = \begin{cases} 0 & n \geq 0 \\ -2^n & n < 0 \end{cases}$$

$$= -2^n u_{-n-1}$$



Example

$$Y(z) = \frac{z}{(z-2)^2} \qquad |z| < 2$$

For multiple poles with the left-sided sequence case, we use the same derivative trick as we did for the right-sided case. Similar to p. 7.4 it is easy to show

$$n\, a^n\, u_{-n-1} \quad \leftrightarrow \quad \frac{-a z}{(z-a)^2} \qquad |z| < |a|\,.$$

Thus,

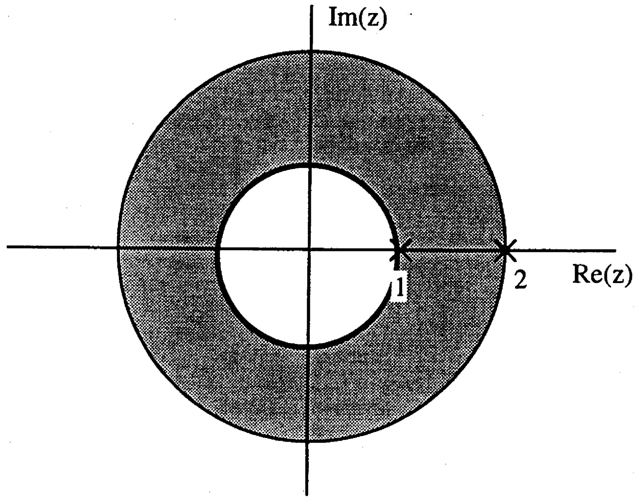
$$y_n = -\frac{1}{2}\, n(2)^n\, u_{-n-1}\ .$$

Example

$$Y(z) = \frac{z}{z-1} + \frac{z}{z-2} \quad 1 < |z| < 2$$

Poles at 1, 2.

$$y_n = \begin{cases} 1 & n \geq 0 \\ -2^n & n < 0 \end{cases}$$



Example

$$x_n = \left(\frac{1}{3}\right)^n \qquad -\infty < n < \infty$$

What is X(z)?

$$X(z) = \sum_{n=0}^{\infty} \left(\frac{1}{3}\right)^n z^{-n} + \sum_{n=-\infty}^{-1} \left(\frac{1}{3}\right)^n z^{-n}.$$

The first sum converges for $|z| > \frac{1}{3}$ and the second sum converges for $|z| < \frac{1}{3}$. There is no z for which both sums converge. Thus, X(z) does not exist for any z. The z-transform of this sequence cannot be defined.

Example

$$x_n = \left(\frac{1}{3}\right)^{|n|} \qquad -\infty < n < \infty$$

What is X(z)?

In this case, we can write

$$x_n = \begin{cases} \left(\frac{1}{3}\right)^n & n \geq 0 \\ \left(\frac{1}{3}\right)^{-n} & n < 0 \end{cases} = \begin{cases} \left(\frac{1}{3}\right)^n & n \geq 0 \\ 3^n & n < 0 \end{cases}$$

Thus,

$$X(z) \;=\; \frac{z}{z-\frac{1}{3}} - \frac{z}{z-3} \qquad \frac{1}{3} < |z| < 3 \;.$$

Note:

If the algebraic form for a z-transform is A(z), e.g.,

$$X(z) = A(z), \qquad z \in ROC_X$$

where

$$A(z) = \frac{N(z)}{(z-p_1)(z-p_2)\ldots(z-p_N)}$$

then ROC_X is generally <u>smaller</u> than the set of z where A(z) is well defined. Indeed, A(z) is well defined at all z except the pole locations $z = p_i$, $1 \le i \le N$, whereas ROC_X must be a ring. It is true that:

1) Poles cannot lie in ROC_X (because even A(z) is undefined at the pole locations).
2) ROC_X is generally smaller than the set of z where A(z) is defined.

Example $\qquad x_n = \left(\frac{1}{2}\right)^n u_n$

$$\Rightarrow X(z) = \frac{z}{z-\frac{1}{2}} \qquad |z| > \frac{1}{2}$$

The algebraic form for X(z) is defined everywhere except at $z = \frac{1}{2}$, and yet the z-transform is <u>not defined</u> for $|z| \le \frac{1}{2}$. For example, consider $z = \frac{1}{4}$. We have

$$\left.\frac{z}{z-\frac{1}{2}}\right|_{z=\frac{1}{4}} = -1$$

However, this does not imply $X\left(\frac{1}{4}\right) = -1$. Indeed, $z = \frac{1}{4}$ is not in ROC_X:

$$\left.\sum_{n=-\infty}^{\infty} \left(\frac{1}{2}\right)^n u_n z^{-n}\right|_{z=\frac{1}{4}} = \sum_{m=0}^{\infty} 2^n \quad \text{which fails to converge.}$$

ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 15

D. Munson

Properties of 2-Sided z-Transform

1. Linearity:

$$Z\left[\{a x_n + b y_n\}\right] = a\,X(z) + b\,Y(z)$$

ROC usually equals $ROC_X \cap ROC_Y$
Sometimes is larger.

Example

Let $w_n = x_n + y_n$ with

$$X(z) = \frac{z}{(z+2)\,(z+3)} \qquad |z| < 2$$

$$Y(z) = \frac{2}{z+2} \qquad |z| < 2$$

$$\Rightarrow W(z) = X(z) + Y(z) = \frac{z + 2(z+3)}{(z+2)(z+3)}$$

$$= \frac{3(z+2)}{(z+2)(z+3)} = \frac{3}{z+3}$$

ROC = ?

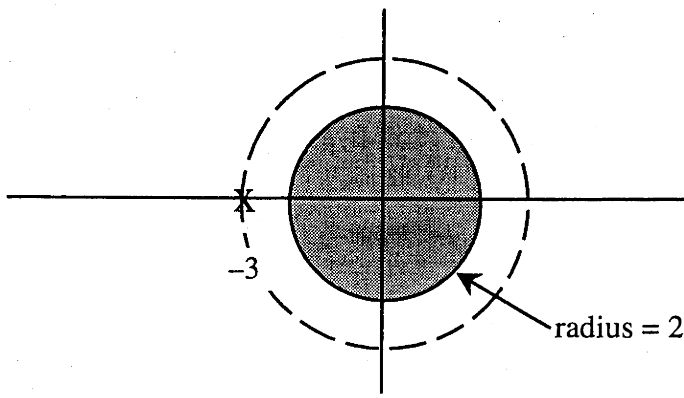
Know two facts:

i) ROC bounded by poles

ii) ROC contains $ROC_X \cap ROC_Y$

For this ex., pole = -3

For this ex., $ROC_X \cap ROC_Y = \{z : |z| < 2\}$



$\Rightarrow$ ROC = $\{z: |z| < 3\}$

So, ROC can be larger than intersection if have pole-zero cancellation on boundary of intersection, in which case the ROC expands outward or inward to be bounded by another pole.

2. Shift:

$$Z\left[\{y_{n\pm k}\}_{n=-\infty}^{\infty}\right] = z^{\pm k}\, Y(z)$$

ROC = ROC_Y except for possible addition or deletion of $z = 0$, or $|z| = \infty$.

Example

$$y_n = \delta_{n-2} \Rightarrow Y(z) = z^{-2} \qquad |z| > 0$$

$$\bar{y}_n = y_{n+3} = \delta_{n+1} \Rightarrow \bar{Y}(z) = z \qquad 0 \le |z| < \infty$$

So, $ROC_{\bar{Y}}$ includes $z = 0$ whereas ROC_Y does not. Similarly, ROC_Y includes $|z| = \infty$ (more precisely, $\lim\limits_{|z|\to\infty} Y(z)$ is finite) whereas $ROC_{\bar{Y}}$ does not.

Proof of Shifting Property

$$Z\left[\{y_{n+k}\}_{n=-\infty}^{\infty}\right] = \sum_{n=-\infty}^{\infty} y_{n+k}\, z^{-n}$$

$$= \sum_{\ell=-\infty}^{\infty} y_l \; z^{-(l-k)}$$

$\downarrow$

$\ell = n + k$

$$= z^k \sum_{\ell=-\infty}^{\infty} y_l z^{-l} = z^k Y(z) \checkmark$$

3. Convolution:

$$y_n = \sum_{m=-\infty}^{\infty} h_m x_{n-m} \quad \text{iff}$$

$$\underset{\substack{\uparrow \\ ROC_Y \text{ contains } ROC_H \cap ROC_X}}{Y(z) = H(z)\,X(z)}$$

ROC_Y can be larger than $\cap$ if have cancellation of pole on boundary of $\cap$. If have cancellation, then ROC expands out to next pole.

Example

$$Y(z) = H(z)\,X(z) \qquad H(z) = \frac{1}{(z+1)(z+2)} \quad 1 < |z| < 2$$

$$X(z) = \frac{z+1}{z+2} \quad |z| < 2$$

$ROC_H \cap ROC_X = \{z: 1 < |z| < 2\}$

<u>But</u> $ROC_Y = \{z: |z| < 2\}$.

<u>Proof of Convolution Formula</u>

Just show $\rightarrow$ direction, steps reversible.

$$y_n = h_n * x_n \Rightarrow$$

$$Y(z) = \sum_{n=-\infty}^{\infty} \left[\sum_{m=-\infty}^{\infty} h_m x_{n-m} \right] z^{-n}$$

$$= \sum_{m=-\infty}^{\infty} h_m \sum_{n=-\infty}^{\infty} x_{n-m} z^{-n}$$

$$\underset{\substack{\uparrow \\ k = n - m}}{=} \sum_{m=-\infty}^{\infty} h_m \sum_{k=-\infty}^{\infty} x_k z^{-(k+m)}$$

$$= \sum_{m=-\infty}^{\infty} h_m z^{-m} \sum_{k=-\infty}^{\infty} x_k z^{-k}$$

$$= H(z)\, X(z) \checkmark$$

Note: Soon we will use z-transform to compute convolution of 2-sided sequences. Then we will need ROC_Y to invert Y(z).

Definition

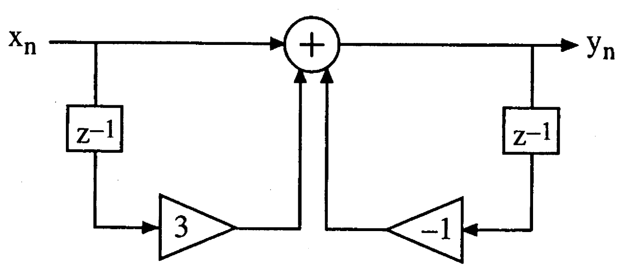
The transfer function of an LSI system is

$$H(z) = \left.\frac{Y(z)}{X(z)}\right|_{\text{zero IC's}}$$

(For LSI systems $\frac{Y(z)}{X(z)}$ is independent of x_n!)

Can get H(z) directly from block diagram and compute Z^{-1} to find $\{h_n\}$.

Example



Find H(z) and $\{h_n\}$.

$$y_n = -y_{n-1} + x_n + 3\,x_{n-1}$$

$$\Rightarrow Y(z) = -z^{-1}\,Y(z) + X(z) + 3\,z^{-1}\,X(z)$$

$$\Rightarrow Y(z)\left[1 + z^{-1}\right] = X(z)\left[1 + 3z^{-1}\right]$$

$$\Rightarrow H(z) = \frac{Y(z)}{X(z)} = \frac{1 + 3z^{-1}}{1 + z^{-1}} = \frac{z+3}{z+1}$$

$$\Rightarrow H(z) = \frac{z}{z+1} + \frac{3}{z+1}$$

Assume causal system (difference equation recurses in forward direction, as written).

$\Rightarrow$ h_n right-sided

$\Rightarrow$ $h_n = (-1)^n u_n + 3(-1)^{n-1} u_{n-1}$

$$= \begin{cases} 0 & n < 0 \\ 1 & n = 0 \\ -2(-1)^n & n > 0 \end{cases}$$

This is the same answer we found in Lecture 14 by solving the D.E.

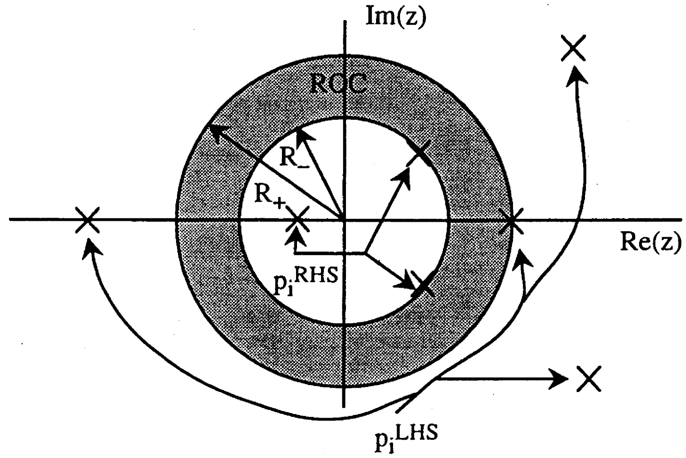
ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 16

D. Munson

Inverting 2-Sided Z-Transform

Main thing to remember:



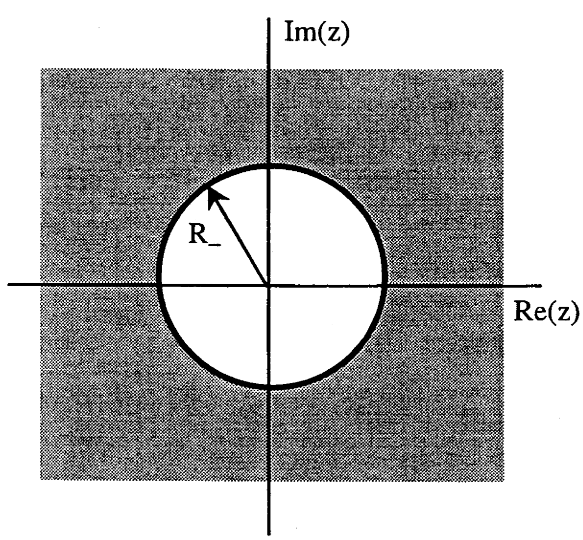
Here p_i^{RHS} are poles in PFE in terms that correspond to RHS $[\{y_n\}]$.

p_i^{LHS} are poles in PFE in terms that correspond to LHS $[\{y_n\}]$.

Why are p_i^{RHS} inside the ROC doughnut?

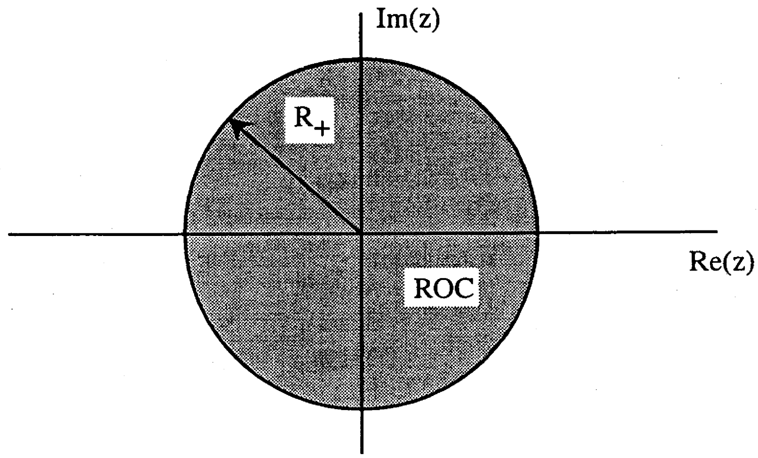
Why are p_i^{LHS} outside the ROC doughnut?

Answer: Know ROC associated with $Z[\text{RHS}\{y_n\}]$ must have looked like:



i.e., ROC = $\{z : |z| > R_-\}$.

Since poles p_i^{RHS} associated with $Z[RHS\{y_n\}]$ can't be in ROC, they must satisfy $0 \le |p_i^{RHS}| \le R_-$. Similarly, ROC associated with $Z[LHS\{y_n\}]$ must look like:



i.e., ROC = $\{z : |z| < R_+\}$. Since poles p_i^{LHS} associated with $Z[LHS\{y_n\}]$ can't be in ROC, they must satisfy $|p_i^{LHS}| \ge R_+$

<u>Bottom line</u>:

a) $0 \le |p_i^{RHS}| \le R_-$

b) $|p_i^{LHS}| \ge R_+$

Example

$$Y(z) = \frac{z}{(z-1)(z-2)}$$

Cannot invert Y(z) unless someone <u>tells us</u> ROC_Y.

Are three possibilities:

1) $ROC_Y = \{z : |z| < 1\}$

   $\Rightarrow \{y_n\}$ is left-sided

2) $ROC_Y = \{z : 1 < |z| < 2\}$

   $\Rightarrow \{y_n\}$ is 2-sided

3) $ROC_Y = \{z : |z| > 2\}$

$\Rightarrow \{y_n\}$ is right-sided

Find PFE:

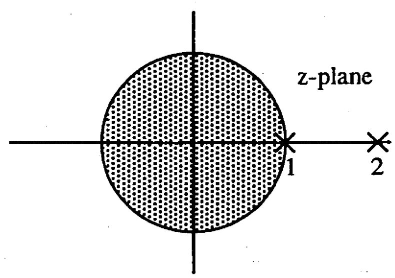
$$\frac{Y(z)}{z} = \frac{1}{(z-1)(z-2)} = \frac{A}{z-1} + \frac{B}{z-2}$$

$$A = -1\ , B = 1$$

$$\Rightarrow Y(z) = \frac{-z}{z-1} + \frac{z}{z-2}$$

Now, to invert terms in PFE, need to know ROC_Y. Three possibilities.

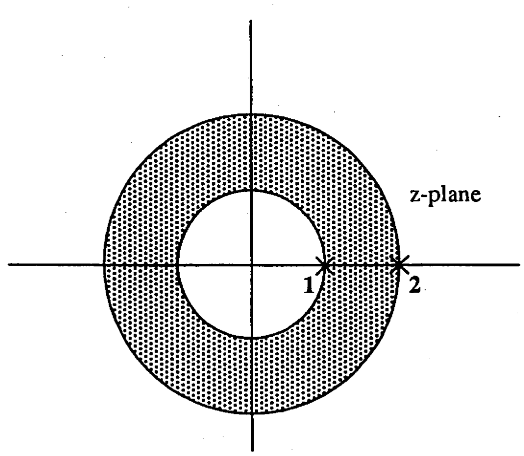
1) $ROC_Y = \{z : |z| < 1\}$



$\Rightarrow \{y_n\}$ left-sided

$$\Rightarrow \boxed{y_n = \begin{cases} 1-2^n & n<0 \\ 0 & n \geq 0 \end{cases}}$$

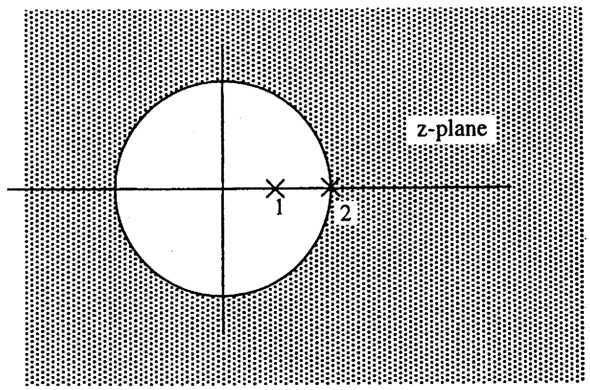
2) $ROC_Y = \{z : 1 < |z| < 2\}$



$\Rightarrow$ $\{y_n\}$ is 2-sided where pole at z = 1 caused by RHS $\{y_n\}$ and pole at z = 2 caused by LHS $\{y_n\}$.

$$\Rightarrow \quad \boxed{y_n = \begin{cases} -2^n & n < 0 \\ -1 & n \geq 0 \end{cases}}$$

3) $ROC_Y = \{z : |z| > 2\}$



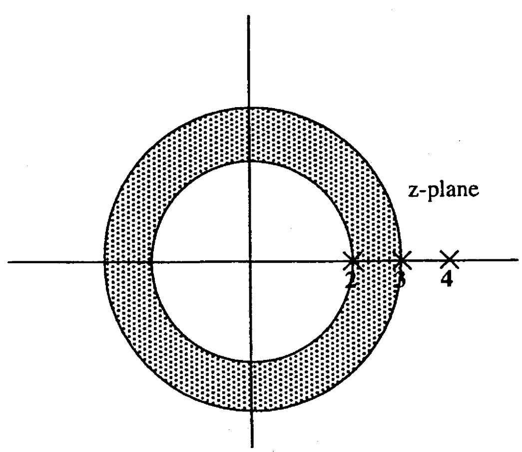
$\Rightarrow$ $\{y_n\}$ right-sided

$$\Rightarrow \quad \boxed{y_n = \begin{cases} 0 & n < 0 \\ -1 + 2^n & n \geq 0 \end{cases}}$$

Again, someone must tell us which of the ROC's is the correct one if we hope to find $\{y_n\}$.

Example

$$Y(z) = \frac{z}{(z-2)(z-3)(z-4)} \qquad 2 < |z| < 3$$



RHS: 2 , LHS: 3, 4

PFE:

$$Y(z) = \underbrace{\frac{\frac{1}{2}z}{z-2}}_{\text{RHS}} - \underbrace{\frac{z}{z-3}}_{\text{LHS}} + \underbrace{\frac{\frac{1}{2}z}{z-4}}_{\text{LHS}}$$

$$\Rightarrow \quad \boxed{y_n = \begin{cases} \frac{1}{2}2^n & n \geq 0 \\ 3^n - \frac{1}{2}4^n & n < 0 \end{cases}}$$

Example

Given X(z), find $\{x_n\}_{n=-\infty}^{\infty}$.

$$X(z) = \frac{1}{(z^2+1)} \qquad |z| < 1$$

$$= \frac{1}{(z+j)(z-j)}$$

$$\frac{X(z)}{z} = \frac{1}{z(z+j)(z-j)}$$

$$= \frac{A}{z} + \frac{B}{z+j} + \frac{C}{z-j}$$

Know $C = B^*$

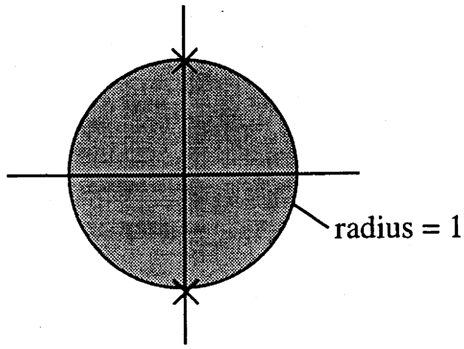
$$A = \left.\frac{1}{(z+j)(z-j)}\right|_{z=0} = 1$$

$$B = \left.\frac{1}{z(z-j)}\right|_{z=-j} = \frac{-1}{2}$$

$$C = B^* = \frac{-1}{2}$$

16.6

$$\Rightarrow X(z) = 1 - \underbrace{\frac{\frac{1}{2}z}{z+j} - \frac{\frac{1}{2}z}{z-j}}_{\text{LHS since } |\text{pole}| \geq R_+}$$



$$\Rightarrow \qquad x_n = \begin{cases} 0 & n > 0 \\ 1 & n = 0 \\ \frac{1}{2}(-j)^n + \frac{1}{2}(j)^n & n < 0 \end{cases}$$

Now, since x_n is a sum of two complex-conjugate terms, we can express x_n in real form. Write

$$(j)^n + (-j)^n = \left(e^{j\frac{\pi}{2}}\right)^n + \left(e^{-j\frac{\pi}{2}}\right)^n$$

$$= e^{j\frac{\pi}{2}n} + e^{-j\frac{\pi}{2}n}$$

$$= 2\cos\frac{\pi}{2}n$$

$$\Rightarrow x_n = \begin{cases} 0 & n > 0 \\ 1 & n = 0 \\ \cos\frac{\pi}{2}n & n < 0 \end{cases} = \boxed{\begin{cases} 0 & n > 0 \\ \cos\frac{\pi}{2}n & n \leq 0 \end{cases}}$$

ECE 310
University of Illinois

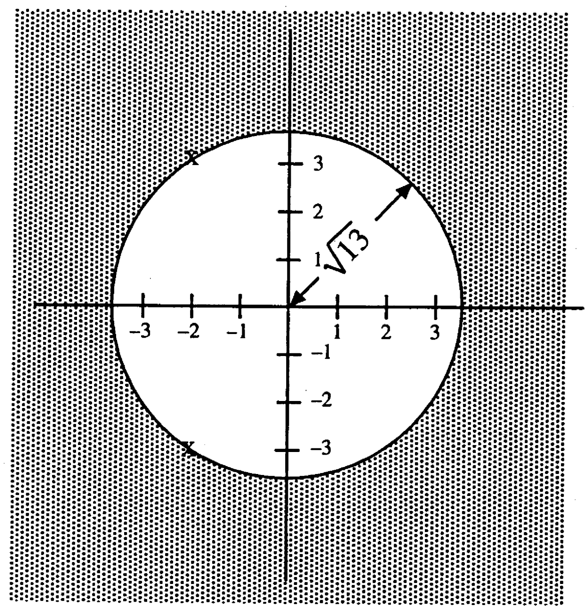
DIGITAL SIGNAL PROCESSING
Lecture 17

D. Munson

Example

$$Y(z) = \frac{z}{z^2+4z+13} \qquad |z| > \sqrt{13}$$

Poles: $$\frac{-4 \pm \sqrt{16-52}}{2} = -2 \pm j3$$



$\Rightarrow \{y_n\}$ is right-sided.

PFE:

$$\frac{Y(z)}{z} = \frac{1}{[z-(-2+j3)][z-(-2-j3)]} = \frac{A}{z-(-2+j3)} + \frac{B}{z-(-2-j3)}$$

$$A = \frac{1}{j6}, \quad B = A^* = \frac{-1}{j6}$$

$$\Rightarrow Y(z) = \underbrace{\frac{\frac{1}{j6}z}{z-(-2+j3)}}_{\text{RHS}} - \underbrace{\frac{\frac{1}{j6}z}{z-(-2-j3)}}_{\text{RHS}}$$

$$\Rightarrow \quad y_n = \begin{cases} \frac{1}{j6}(-2+j3)^n - \frac{1}{j6}(-2-j3)^n & n \geq 0 \\ 0 & n < 0 \end{cases}$$

To express y_n in real form, write

$$-2 + j3 = \sqrt{13}\, e^{j\theta}$$

where $\theta = \operatorname{atan}\left(\frac{3}{-2}\right)$.

$$\Rightarrow \quad y_n = \begin{cases} \frac{1}{j6}\left[\left(\sqrt{13}\right)^n e^{j\theta n} - \left(\sqrt{13}\right)^n e^{-j\theta n}\right] & n \geq 0 \\ 0 & n < 0 \end{cases}$$

$$= \begin{cases} \frac{1}{j6}\left(\sqrt{13}\right)^n \left[e^{j\theta n} - e^{-j\theta n}\right] & n \geq 0 \\ 0 & n < 0 \end{cases}$$

$$= \boxed{\begin{cases} \frac{\left(\sqrt{13}\right)^n}{3} \sin\theta n & n \geq 0 \\ 0 & n < 0 \end{cases}}$$

Example

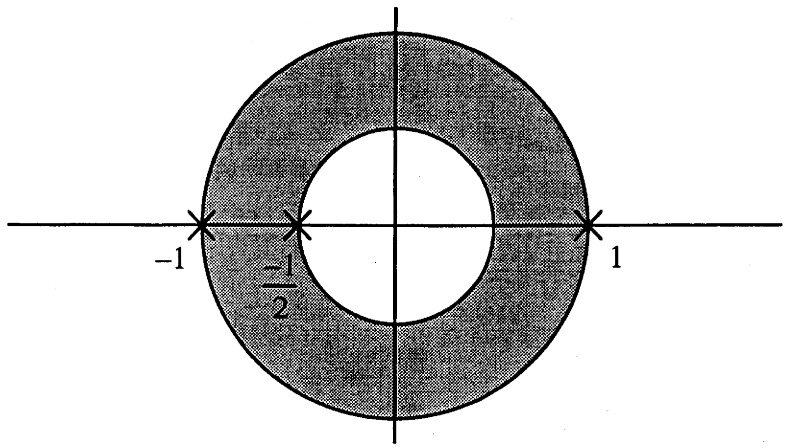
$$X(z) = \frac{z}{\left(z+\frac{1}{2}\right)(z-1)(z+1)} \qquad \frac{1}{2} < |z| < 1$$

Find $\{x_n\}_{n=-\infty}^{\infty}$.

$$\frac{X(z)}{z} = \frac{A}{z+\frac{1}{2}} + \frac{B}{z-1} + \frac{C}{z+1}$$

$$\underline{A = -\frac{4}{3}}, \ \underline{B = \frac{1}{3}}, \ \underline{C = 1}$$

$$X(z) = \underbrace{\frac{\frac{-4}{3}z}{z+\frac{1}{2}}}_{\text{RHS}} + \underbrace{\frac{\frac{1}{3}z}{z-1}}_{\text{LHS}} + \underbrace{\frac{z}{z+1}}_{\text{LHS}} \qquad \frac{1}{2} < |z| < 1$$



$$\Rightarrow \qquad \boxed{x_n = \begin{cases} -\frac{4}{3}\left(-\frac{1}{2}\right)^n & n \geq 0 \\ -\frac{1}{3} - (-1)^n & n < 0 \end{cases}}$$

Example

$$x_n = \left(\frac{1}{2}\right)^n u_n$$

$$h_n = 3^n u_{-n-1}$$

Find $y_n = h_n * x_n$ via the two-sided z-transform method. (Note: The one-sided z-transform cannot be used, since h_n is not right-sided.)

$$X(z) = \frac{z}{z-\frac{1}{2}} \quad , \quad ROC_X = \left\{z : |z| > \frac{1}{2}\right\}$$

$$H(z) = \frac{-z}{z-3} \quad , \quad ROC_H = \{z : |z| < 3\}$$

$$Y(z) = H(z)\,X(z) = \frac{-z^2}{\left(z-\frac{1}{2}\right)(z-3)} \qquad ROC_Y = ROC_X \cap ROC_H = \left\{z : \frac{1}{2} < |z| < 3\right\}$$

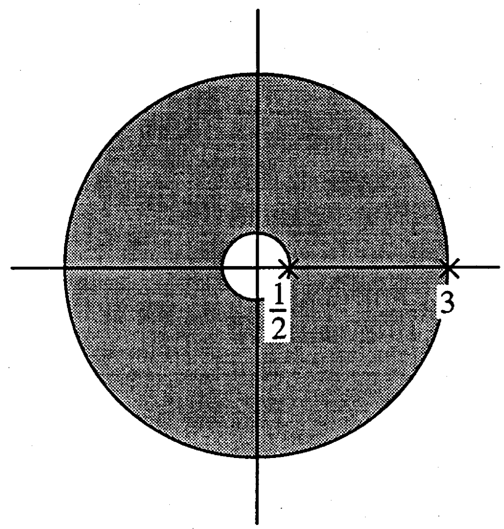
Here, ROC_Y is simply the intersection of ROC_X and ROC_Y because neither of the poles is cancelled when multiplying H(z) and X(z). Now,

$$\frac{Y(z)}{z} = \frac{-z}{\left(z-\frac{1}{2}\right)(z-3)} = \frac{A}{z-\frac{1}{2}} + \frac{B}{z-3}$$

$$A = \frac{1}{5}\ ,\ B = \frac{-6}{5}$$

$$\Rightarrow Y(z) = \underbrace{\frac{\frac{1}{5}z}{z-\frac{1}{2}}}_{\text{RHS}} - \underbrace{\frac{\frac{6}{5}z}{z-3}}_{\text{LHS}}$$

$$\Rightarrow y_n = \begin{cases} \frac{1}{5}\left(\frac{1}{2}\right)^n & n \geq 0 \\ \frac{6}{5}(3)^n & n < 0 \end{cases}$$



Now, you might ask what will happen if we interchange the forms of x_n and h_n so that both grow, instead of decay. Consider

$$\bar{x}_n = 3^n u_n$$

$$\bar{h}_n = \left(\frac{1}{2}\right)^n u_{-n-1}$$

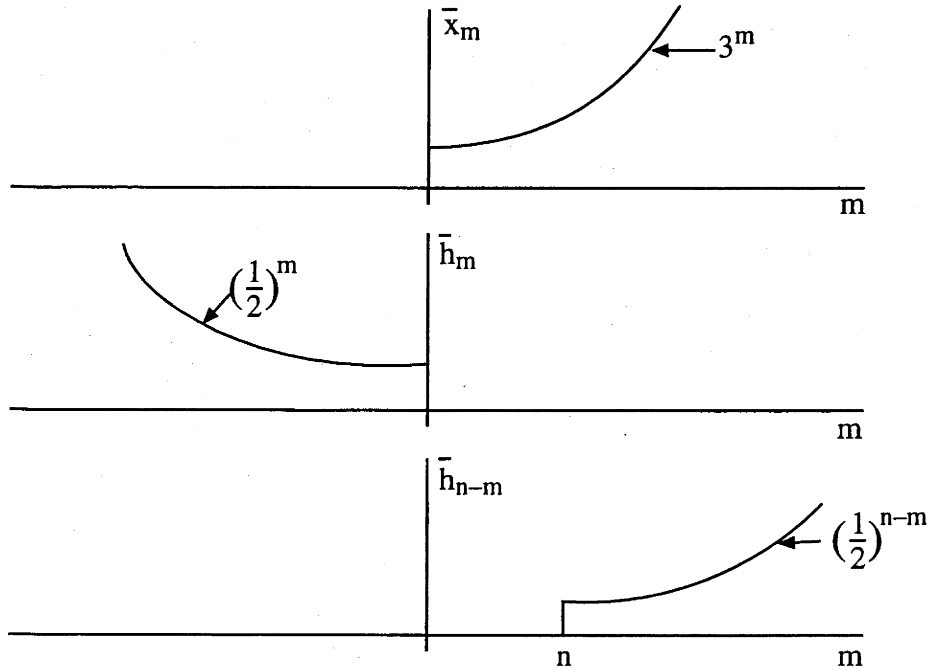
Then

$$\bar{X}(z) = \frac{z}{z-3}\ ,\ ROC_{\bar{X}} = \{z : |z| > 3\}$$

$$\bar{H}(z) = \frac{-z}{z-\frac{1}{2}}\ ,\ ROC_{\bar{H}} = \{z : |z| < \frac{1}{2}\}$$

and $ROC_{\bar{Y}} = ROC_{\bar{X}} \cap ROC_{\bar{H}} = \phi$, That is, $\bar{Y}(z) = \bar{H}(z)\,\bar{X}(z)$ does not exist (i.e., is not finite) for any z. In fact, in this example, y_n is undefined (infinite) for every n. This is easy to see by visualizing the convolution in the sequence domain.

We have:



We see that $\bar{y}_n = \sum_{m=-\infty}^{\infty} \bar{x}_m \, \bar{h}_{n-m}$ is infinite for every value of n.

Example Find $y_n = h_n * x_n$ where:

$$x_n = \begin{cases} 2^n & n < 0 \\ 0 & n \geq 0 \end{cases} \qquad h_n = \begin{cases} \left(\frac{1}{3}\right)^n & n \geq 0 \\ 0 & n < 0 \end{cases}$$

Use z-transform method of solution, but note that in sequence domain can see result will be two-sided.

$$X(z) = \frac{-z}{z-2} \qquad |z| < 2$$

$$H(z) = \frac{z}{z - \frac{1}{3}} \qquad |z| > \frac{1}{3}$$

$$Y(z) = \frac{-z^2}{\left(z - \frac{1}{3}\right)(z-2)} \qquad \underbrace{\frac{1}{3} < |z| < 2}_{= ROC_X \cap ROC_H}$$

Note: There are no pole-zero cancellations in multiplying X and H so ROC_Y is just the intersection of ROC_X with ROC_H.

$$\frac{Y(z)}{z} = \frac{-z}{\left(z-\frac{1}{3}\right)(z-2)} = \frac{A}{z-\frac{1}{3}} + \frac{B}{z-2}$$

$$\underline{A = \frac{1}{5}} \qquad\qquad \underline{B = \frac{-6}{5}}$$

$$\Rightarrow \quad Y(z) = \underbrace{\frac{\frac{1}{5}z}{z-\frac{1}{3}}}_{\text{RHS}} - \underbrace{\frac{\frac{6}{5}z}{z-2}}_{\text{LHS}} \qquad \frac{1}{3} < |z| < 2$$

$$\Rightarrow \quad \boxed{y_n = \begin{cases} \frac{1}{5}\left(\frac{1}{3}\right)^n & n \geq 0 \\ \frac{6}{5}2^n & n < 0 \end{cases}}$$

Example Find $y_n = h_n * x_n$ where:

$$x_n = \begin{cases} n & n \geq 0 \\ 0 & n < 0 \end{cases} \qquad\qquad h_n = \begin{cases} (-1)^n & n \geq 0 \\ 0 & n < 0 \end{cases}$$

$$X(z) = \sum_{n=0}^{\infty} n z^{-n} = -z \frac{d}{dz} \sum_{n=0}^{\infty} z^{-n}$$

$$= -z \frac{d}{dz} \frac{z}{z-1}$$

$\uparrow$

$\text{ROC} = \{z : |z| > 1\}$

$$= -z \frac{1 \bullet (z-1) - 1 \bullet z}{(z-1)^2} = -z \frac{-1}{(z-1)^2}$$

$$= \frac{z}{(z-1)^2} \qquad |z| > 1$$

$$H(z) = \frac{z}{z+1} \qquad |z| > 1$$

$$Y(z) = \frac{z^2}{(z+1)(z-1)^2} \qquad |z| > 1$$

Again, there is no pole-zero cancellation in finding Y(z), so $ROC_Y = ROC_X \cap ROC_H$.

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

$$= \frac{A}{z+1} + \frac{B}{z-1} + \frac{C}{(z-1)^2}$$

$$A = \left.\frac{z}{(z-1)^2}\right|_{z=-1} = \frac{-1}{4}$$

Next, find C:

$$\frac{z}{z+1} = \frac{A(z-1)^2}{z+1} + B\,(z-1) + C \qquad (*)$$

$$\Rightarrow C = \left.\frac{z}{z+1}\right|_{z=1} = \frac{1}{2}$$

Differentiate (*) and evaluate at z = 1 to find B:

$$\frac{1 \bullet (z+1) - 1 \bullet z}{(z+1)^2} = \text{MESS} + B + 0$$

Note: MESS = 0 when evaluated at z = 1, so

$$B = \left.\frac{1}{(z+1)^2}\right|_{z=1} = \frac{1}{4}$$

$$\Rightarrow Y(z) = \underbrace{\frac{-\frac{1}{4}z}{z+1} + \frac{\frac{1}{4}z}{z-1} + \frac{\frac{1}{2}z}{(z-1)^2}}_{\text{RHS}} \qquad |z| > 1$$

$$\Rightarrow \boxed{y_n = \frac{-1}{4}(-1)^n u_n + \frac{1}{4}u_n + \frac{1}{2}n\,u_n}$$

Example Find $y_n = h_n * x_n$ where:

$$x_n = \begin{cases} 5^n & n < 0 \\ 2^n & n \geq 0 \end{cases} \qquad h_n = \delta_n - 5\,\delta_{n-1}$$

Could solve this quickly using $y_n = [\delta_n - 5\delta_{n-1}] * x_n = x_n - 5x_{n-1}$. Instead, use z-transform method:

$$X(z) = \frac{-z}{z-5} + \frac{z}{z-2} \qquad 2 < |z| < 5$$

$$= \frac{-3z}{(z-5)(z-2)} \qquad 2 < |z| < 5$$

$$H(z) = 1 - 5\,z^{-1} \qquad |z| > 0$$

$$Y(z) = \frac{-3z\left(1-5z^{-1}\right)}{(z-5)(z-2)} = \frac{-3z \bullet z^{-1} \bullet (z-5)}{(z-5)(z-2)}$$

$$= \frac{-3}{z-2}$$

$ROC_X \cap ROC_H = ROC_Y = \{z : 2 < |z| < 5\}$

But, pole at $z = 5$ was canceled. $Y(z)$ has only a single pole at $z = 2$.

Since ROC_Y :

a) Is bounded by the pole at $z = 2$

b) Contains $ROC_X \cap ROC_H$

the only possibility is $ROC_Y = \{z : |z| > 2\}$

$$\Rightarrow \qquad Y(z) = z^{-1}\,\frac{-3z}{z-2} \qquad |z| > 2$$

$$\Rightarrow \qquad \boxed{y_n = -3(2)^{n-1}\,u_{n-1}}$$

Example Find $y_n = h_n * x_n$ where

$$\{x_n\} = \{\underset{\uparrow}{1}, 1, 3, 0, 0, 0, \ldots\}$$

$$\{h_n\} = \{\underset{\uparrow}{2}, 1, 0, 0, 0, \ldots\}$$

$$X(z) = 1 + z^{-1} + 3\,z^{-2} \qquad |z| > 0$$

$$H(z) = 2 + z^{-1} \qquad |z| > 0$$

$$Y(z) = \left(2+z^{-1}\right)\left(1+z^{-1}+3z^{-2}\right) \qquad |z| > 0$$

$$= 2 + 2\,z^{-1} + 6\,z^{-2} + z^{-1} + z^{-2} + 3\,z^{-3}$$

$$= 2 + 3\,z^{-1} + 7\,z^{-2} + 3\,z^{-3}$$

$$= y_0 + y_1\,z^{-1} + y_2\,z^{-2} + y_3\,z^{-3} + \dots$$

$$\Rightarrow \quad \boxed{\{y_n\} = \{\underset{\uparrow}{2},\ 3,\ 7,\ 3,\ 0,\ 0,\ \dots\ \}}$$

Easy to check answer via convolution.

From this example, we see that convolution is equivalent to multiplication of polynomials (variable of our polynomials is z^{-1}).

Coefficients of a product of polynomials are <u>convolution</u> of original coefficient sets.

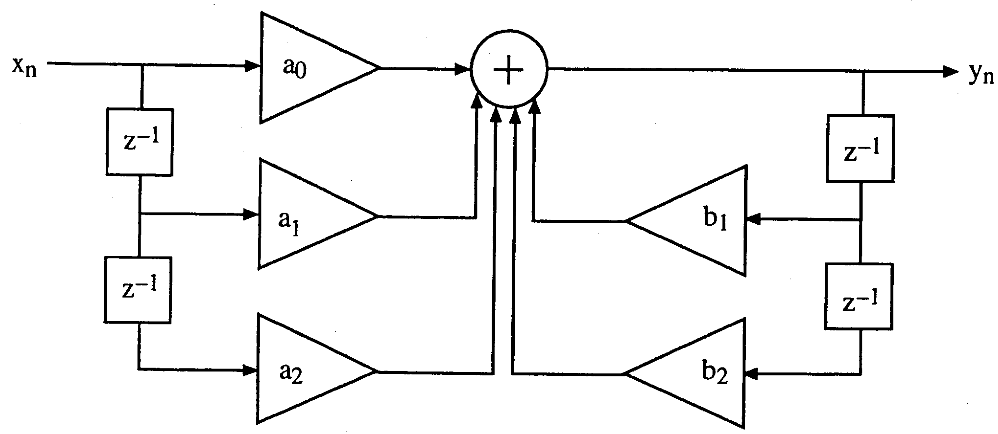
ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 18

D. Munson

Common 2nd-Order Digital Filter Structures

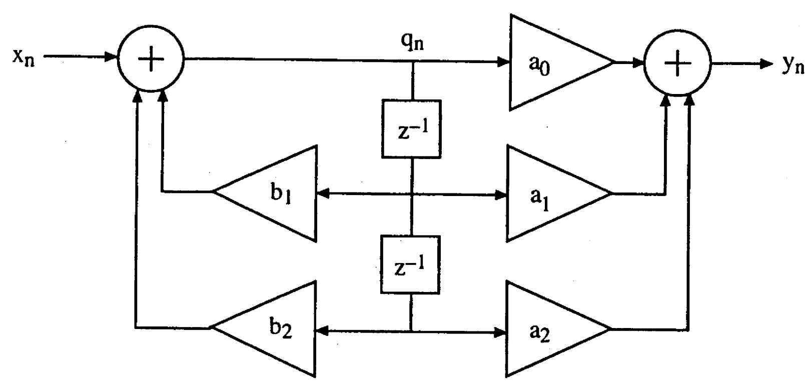
Direct Form 1:



Students: Write D. E. and take z-transform of both sides to show

$$H(z) = \frac{a_0 + a_1 z^{-1} + a_2 z^{-2}}{1 - b_1 z^{-1} - b_2 z^{-2}}$$

Direct Form 2:



Has same transfer function as Direct Form 1. Let's show this.
Hard to write y_n in terms of x_n. Introduce "dummy variable" q_n. Write two D.E.'s.

1) $y_n = a_0\, q_n + a_1\, q_{n-1} + a_2\, q_{n-2}$

$$\Rightarrow \; Y(z) = a_0\, Q(z) + a_1\, z^{-1}\, Q(z) + a_2\, z^{-2}\, Q(z)$$

2) $q_n = x_n + b_1 q_{n-1} + b_2 q_{n-2}$

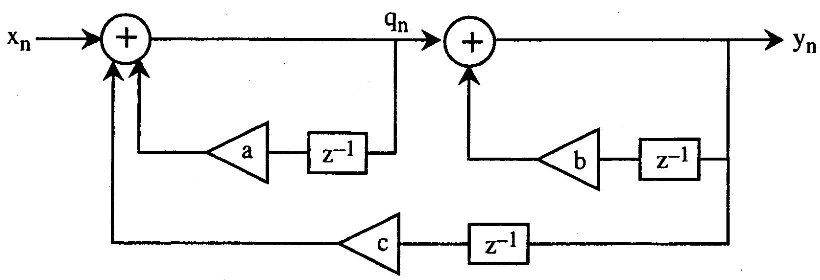
$$\Rightarrow Q(z)\left[1 - b_1 z^{-1} - b_2 z^{-2}\right] = X(z)$$

$$\Rightarrow Y(z) = \left[a_0 + a_1 z^{-1} + a_2 z^{-2}\right]\frac{X(z)}{1 - b_1 z^{-1} - b_2 z^{-2}}$$

$$\Rightarrow H(z) = \frac{a_0 + a_1 z^{-1} + a_2 z^{-2}}{1 - b_1 z^{-1} - b_2 z^{-2}} \quad ✓$$

Example

Find H(z) for



Can write D.E.'s and then z-transform them, or else just write z-transforms directly:

$$Y(z) = Q(z) + b\,z^{-1}\,Y(z) \qquad (1)$$

$$Q(z) = X(z) + a\,z^{-1}\,Q(z) + c\,z^{-1}\,Y(z) \qquad (2)$$

$$(1) \Rightarrow Q(z) = Y(z)\left[1 - b z^{-1}\right]$$

Substitute into (2):

$$Y(z)\left[1 - bz^{-1}\right] = X(z) + a\,z^{-1}\,Y(z)\left[1 - bz^{-1}\right] + c\,z^{-1}\,Y(z)$$

$$\Rightarrow Y(z)\left[1 - bz^{-1} - az^{-1} + abz^{-2} - cz^{-1}\right] = X(z)$$

$$\Rightarrow \boxed{H(z) = \frac{1}{1 - (a+b+c)z^{-1} + abz^{-2}}}$$

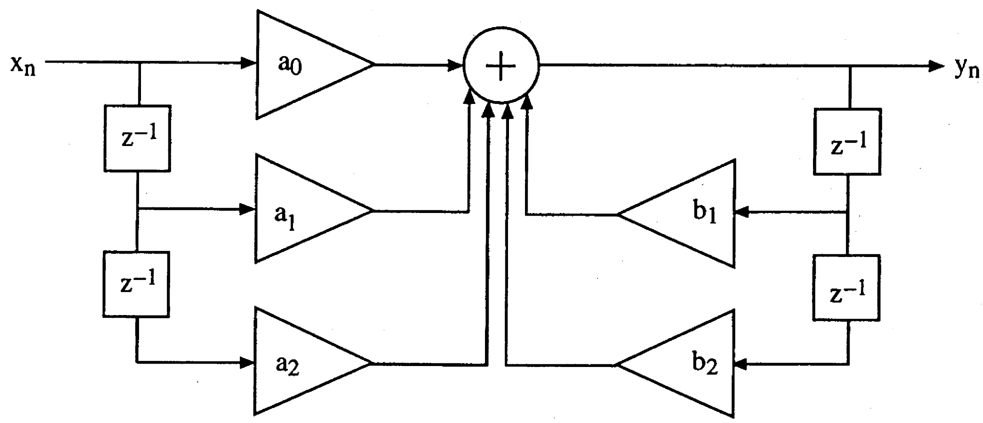
Notes on digital filter implementation

$\{h_n\}$ and H(z) are <u>input-output</u> descriptions of digital filters. Given an input $\{x_n\}$, we can use either $\{h_n\}$ or H(z) to determine the output $\{y_n\}$. In this sense, both $\{h_n\}$ and H(z) <u>summarize</u> the behavior of the system. However, neither $\{h_n\}$ nor H(z) tell us what the internal structure of the digital filter looks like. Indeed, for any given H(z), there are an infinite number of filter structures that will all have this same transfer function. For a second-order transfer function

$$H(z) = \frac{a_0 + a_1 z^{-1} + a_2 z^{-2}}{1 - b_1 z^{-1} - b_2 z^{-2}}$$

the Direct Form 1 and Direct Form 2 structures are just the two most obvious possibilities.

At this point, you may wonder how the filter structure or diagram relates to the actual filter implementation. The answer is multifaceted. Let's consider the Direct Form I structure as an example.



Suppose we implement this filter in a DSP microprocessor. Then, the first thing we must realize is that the system is clocked. The clock is not shown in our digital filter diagram. Ordinarily it takes many clock cycles, corresponding to many microprocessor instructions, to compute a single value of the output sequence $\{y_n\}$. For example, if our DSP has a single multiplier/accumulator, then the clock might trigger the following instructions:

1) multiply x_n by a_0
2) multiply x_{n-1} by a_1 and add to 1)
3) multiply x_{n-2} by a_2 and add to 2)
4) multiply y_{n-1} by b_1 and add to 3)
5) multiply y_{n-2} by b_2 and add to 4) to give y_n.

The values of x_n, x_{n-1}, x_{n-2}, y_{n-1}, y_{n-2} are stored in memory locations. You might expect that after y_n is computed, then in preparation to compute y_{n+1}, we should use a sequence of instructions to move x_{n+1} to the old x_n location, x_n to the old x_{n-1} location, x_{n-1} to the old x_{n-2} location, y_n to the old y_{n-1} location, and y_{n-2} to the old y_{n-1} location. However, especially in higher order filters, this would be a huge waste of clock cycles (instructions). Instead, a pointer is used to address the proper memory location at each clock cycle. Thus, it is not necessary to move data from memory location to memory location after computing each y_n.

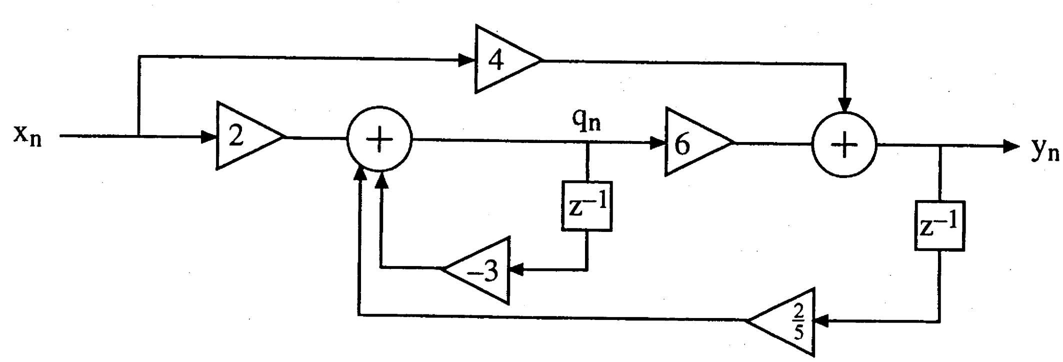
Just as there are a large number of filter structures that implement the same transfer function, there are many algorithms (for a specific DSP) that can implement a given filter structure. What are the considerations in choosing a structure/algorithm? There are generally two:

1) Speed (number of clock cycles per output)
2) Error due to finite register length.

We have not yet addressed 2). The fact that the DSP has finite-length registers, both for memory locations and the arithmetic unit, means that the digital filtering algorithm is not implemented in an exact way. There will be error at the filter output due to coefficient quantization and arithmetic roundoff. Of course, the longer the register lengths, the lower the error at the filter output. Generally, there is a tradeoff between 1) and 2). For a fixed register length, error usually can be reduced by using a more complicated (than Direct Form) filter structure, requiring more multiplications, additions, and memory locations. This in turn reduces the speed of the filter. The filter structure used in practice depends on H(z) (some transfer functions are more difficult to implement with low error), on the available register length, and on the number of clock cycles available per output.

Example

Find the transfer function of the system below and sketch a Direct Form 2 filter structure that implements the same transfer function.



We establish the intermediate quantity q_n and then write:

(i) $Y(z) = 6\,Q(z) + 4\,X(z)$

(ii) $Q(z) = 2\,X(z) - 3\,z^{-1}\,Q(z) + \dfrac{2}{5}\,z^{-1}\,Y(z)$

Solve (i) for Q(z) and substitute into (ii):

$$\text{(i)} \Rightarrow Q(z) = \frac{1}{6} Y(z) - \frac{2}{3} X(z)$$

$$\text{(ii)} \Rightarrow \frac{1}{6} Y(z) - \frac{2}{3} X(z) = 2\,X(z) - 3\,z^{-1}\left[\frac{1}{6} Y(z) - \frac{2}{3} X(z)\right] + \frac{2}{5} z^{-1} Y(z)$$

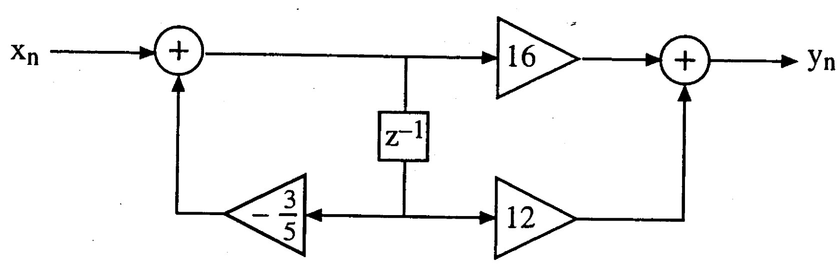
$$\Rightarrow Y(z)\left[\frac{1}{6} + \frac{1}{2} z^{-1} - \frac{2}{5} z^{-1}\right] = X(z)\left[\frac{2}{3} + 2 + 2\,z^{-1}\right]$$

$$\Rightarrow H(z) = \frac{\frac{8}{3} + 2\,z^{-1}}{\frac{1}{6} + \frac{1}{10} z^{-1}}$$

Now, the quickest way to map H(z) into a Direct Form filter structure is to first normalize the denominator of H(z) to have a leading term equal to one. Thus, multiply both the numerator and denominator of H(z) by 6 to give

$$H(z) = \frac{16 + 12\,z^{-1}}{1 + \frac{3}{5} z^{-1}}$$

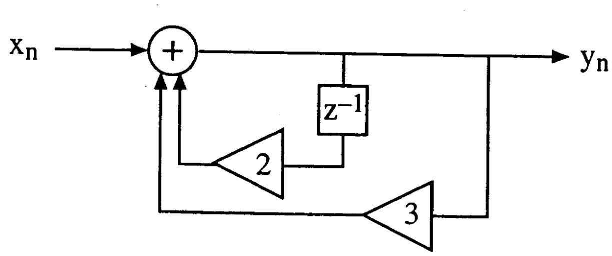
The Direct Form 2 structure having this transfer function is then



This structure is far simpler than the previous one and it computes exactly the same output $\{y_n\}$.

Important Note: Digital filter structures cannot have delay-free loops.

Example Consider the filter structure

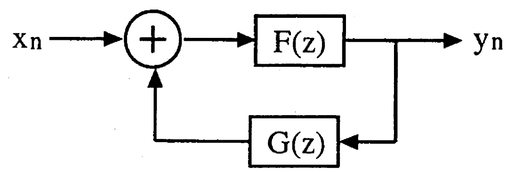


This system is unrealizable because

$$y_n = x_n + 2\,y_{n-1} + 3\,y_n$$

Since the system is clocked and the elements of $\{y_n\}$ are computed one at a time, we cannot have element y_n depend on itself as in the above equation.

A handy fact:



$$\Rightarrow H(z) = \frac{F(z)}{1 - F(z)G(z)}$$

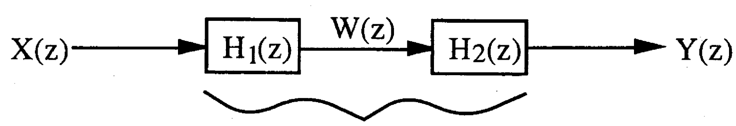
Proof:

$$Y(z) = F(z)\,[X(z) + G(z)\,Y(z)]$$

$$\Rightarrow\ Y(z)\,[1 - F(z)\,G(z)] = F(z)\,X(z)$$

$$\Rightarrow\ H(z) = \frac{Y(z)}{X(z)} = \frac{F(z)}{1 - F(z)G(z)} \qquad \checkmark$$

Cascade (series) and **parallel** connections:



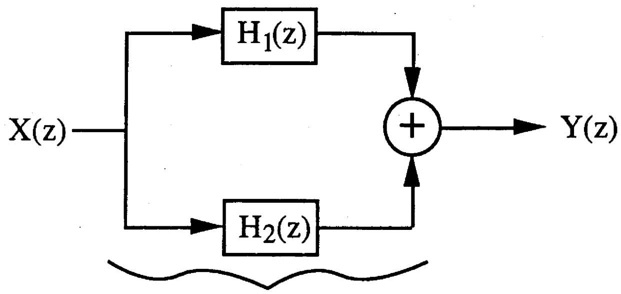
$TF = H_1(z)\,H_2(z)$

Proof:

$$Y(z) = H_2(z)\,W(z)$$

$$= H_2(z)\,[H_1(z)X(z)]$$

$$\Rightarrow\ H(z) = \frac{Y(z)}{X(z)} = H_1(z)\,H_2(z) \qquad \checkmark$$



Proof:

$$Y(z) = H_1(z)\,X(z) + H_2(z)\,X(z)$$

$$= [H_1(z) + H_2(z)]\,X(z)$$

$$\Rightarrow H(z) = \frac{Y(z)}{X(z)} = H_1(z) + H_2(z) \quad ✓$$

Complex Systems

In Lecture 6, we pointed out that systems with complex-valued inputs, outputs, adders, and multipliers are realizable. That is, they are implemented using real adders, real multipliers, and real delays. The following example gives further insight into how this can be done.

Example

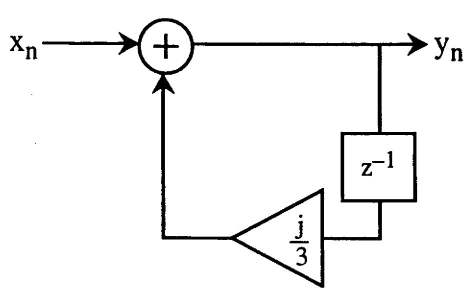
Draw a block diagram of a system that implements $y_n = h_n * x_n$ where $\{x_n\}$ and $\{y_n\}$ are complex-valued and $h_n = \left(\frac{j}{3}\right)^n u_n$. All adders, multipliers, and delays should be real.

<u>Solution</u>

We have

$$H(z) = \frac{z}{z - \frac{j}{3}} = \frac{1}{1 - \frac{j}{3}z^{-1}}$$

So, we might consider



Here, though, $\{x_n\}$ and $\{y_n\}$ are each pairs of real-valued sequences. Write $x_n = (x_R(n), x_I(n))$ and $y_n = (y_R(n), y_I(n))$. Then, recalling the definitions of complex multiplication and addition, we have

$$\frac{j}{3}y_{n-1} = \left(0, \frac{1}{3}\right) \bullet (y_R(n-1), y_I(n-1)) = \left(-\frac{1}{3}y_I(n-1), \frac{1}{3}y_R(n-1)\right).$$

Then

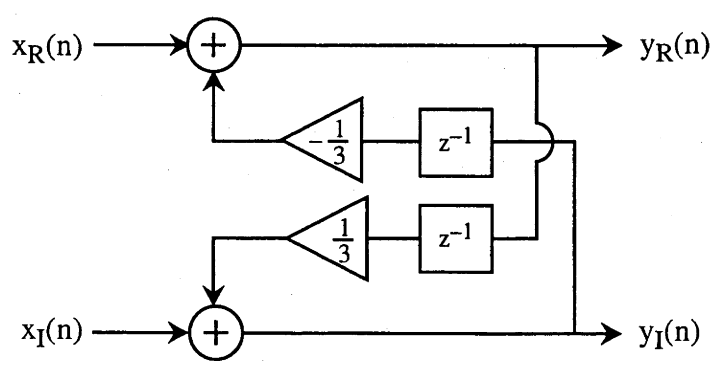
$$\begin{aligned} y_n &= (y_R(n), y_I(n)) = x_n + \frac{j}{3}y_{n-1} \\ &= (x_R(n), x_I(n)) + \left(-\frac{1}{3}y_I(n-1), \frac{1}{3}y_R(n-1)\right) \\ &= \left(x_R(n) - \frac{1}{3}y_I(n-1), x_I(n) + \frac{1}{3}y_R(n-1)\right) \end{aligned}$$

Thus,

$$y_R(n) = x_R(n) - \frac{1}{3}\, y_I(n-1)$$

$$y_I(n) = x_I(n) + \frac{1}{3}\, y_R(n-1)$$

These last two equations tell us exactly how to implement the system:



This is a physical implementation of the previous block diagram.

Approaching the implementation problem in an alternate way, we can find a more complicated, but equivalent, physical realization. Write

$$\begin{aligned} y_R(n) + j\, y_I(n) &= (h_R(n) + jh_I(n)) * (x_R(n) + jx_I(n)) \\ &= (h_R(n) * x_R(n) - h_I(n) * x_I(n)) + j(h_R(n) * x_I(n) + h_I(n) * x_R(n)) \qquad (\Delta) \end{aligned}$$

Furthermore, we can write

$$H(z) = H_R(z) + j\,H_I(z)$$

where $H_R(z)$ is the z-transform of $h_R(n)$ and $H_I(z)$ is the z-transform of $h_I(n)$. Since both $h_R(n)$ and $h_I(n)$ are real-valued, the coefficients of both $H_R(z)$ and $H_I(z)$ must be real-valued. How do we find $H_R(z)$ and $H_I(z)$? There are two ways. The easiest is to write

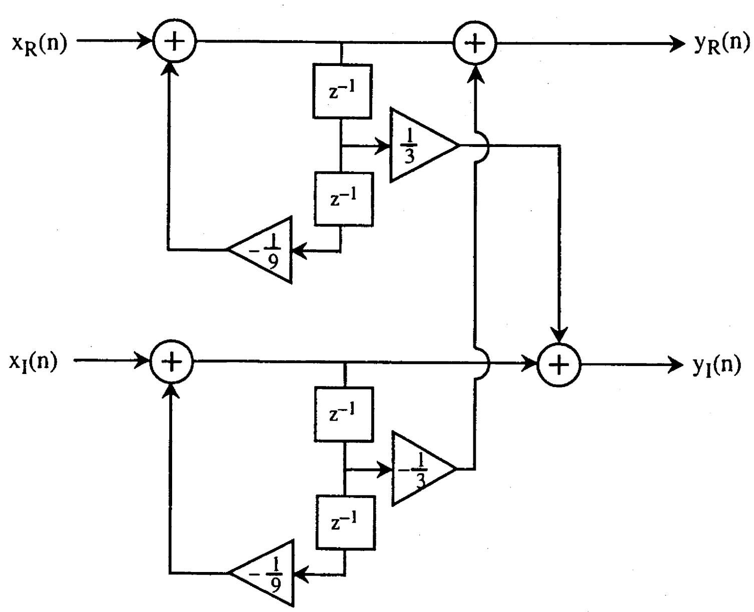
$$H(z) = \frac{z}{z-\frac{j}{3}} = \frac{z}{z-\frac{j}{3}}\,\frac{z+\frac{j}{3}}{z+\frac{j}{3}} = \frac{z^2+j\frac{1}{3}z}{z^2+\frac{1}{9}}$$

$$= \frac{1}{1+\frac{1}{9}z^{-2}} + j\,\frac{\frac{1}{3}z^{-1}}{1+\frac{1}{9}z^{-2}}$$

Thus,

$$H_R(z) = \frac{1}{1+\frac{1}{9}z^{-2}}, \qquad H_I(z) = \frac{\frac{1}{3}z^{-1}}{1+\frac{1}{9}z^{-2}} \qquad (\Delta\Delta)$$

Using this, with Eq. (Δ) above, our implementation of $H(z)$ has two copies of $H_R(z)$ and two copies of $H_I(z)$, with inputs $x_R(n)$ and $x_I(n)$. The outputs of the copies of $H_R(z)$ and $H_I(z)$ are then interconnected to produce $y_R(n)$ and $y_I(n)$. Since $H_R(z)$ and $H_I(z)$ are nearly the same in this example, however, the diagram can be simplified to



Although this diagram is quite different from our earlier implementation, it is equivalent in the sense that it computes the same $y_R(n)$ and $y_I(n)$.

Note: ($\Delta\Delta$) can be derived in an alternate, but lengthier way. Since $h_n = \left(\frac{j}{3}\right)^n u_n$, we have

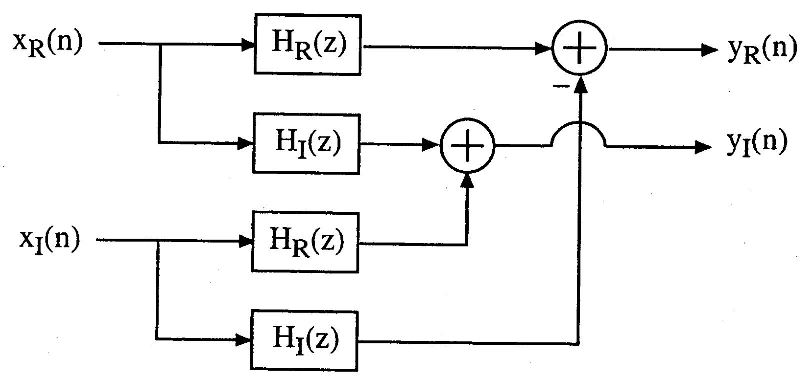
$$
\begin{aligned}
h_R(n) &= 1 \quad 0 \quad \frac{-1}{9} \quad 0 \quad \frac{1}{81} \quad 0 \quad \ldots \\
h_I(n) &= 0 \quad \frac{1}{3} \quad 0 \quad \frac{-1}{27} \quad 0 \quad \frac{1}{243} \quad \ldots
\end{aligned}
$$

Taking z-transforms of these sequences gives ($\Delta\Delta$).

In general

$$
\begin{aligned}
Y(z) &= \left(H_R(z)+jH_I(z)\right)\left(X_R(z)+jX_I(z)\right) \\
&= \left(H_R(z)X_R(z)-H_I(z)X_I(z)\right)+j\left(H_R(z)X_I(z)+H_I(z)X_R(z)\right)
\end{aligned}
$$

so that a possible implementation is always



ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 19

D. Munson

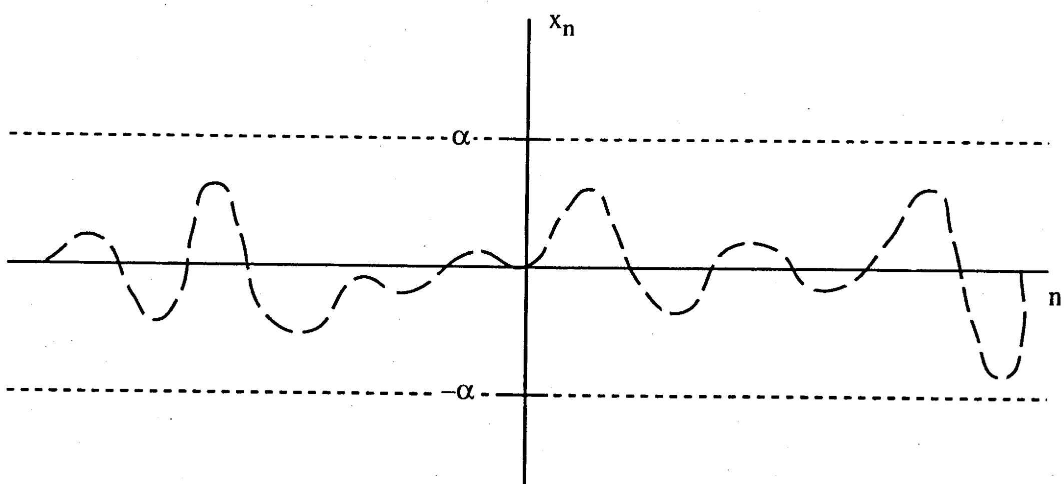
Stability

Def. System is bounded-input, bounded-output (BIBO) stable if for every bounded input the resulting output is bounded, i.e.,

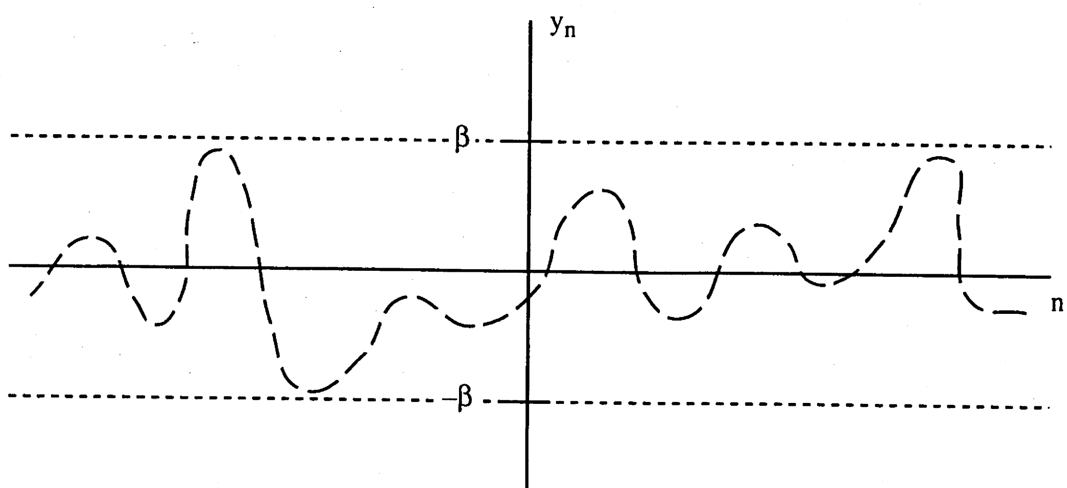
$$|x_n| < \alpha < \infty \;\forall\, n \Rightarrow |y_n| < \beta < \infty \;\forall\, n$$

indep. of n

Pictorially, if every bounded x_n:



causes a bounded y_n:



then system is BIBO stable.

Why do we care about stability? If a digital system is unstable, so that a bounded input may cause an unbounded output, might the system "blow up" or explode? **Absolutely not!** In a digital system, if the output continues to grow, at some point it will exceed the dynamic range of the binary register in which it is stored, i.e., we will have saturation or overflow. If this occurs, the implementation of the system is no longer linear and the outputs y_n will be **incorrect**.

How do we check if a system is BIBO stable? We can't possibly try every bounded input and check that the resulting outputs are bounded! Instead, the following theorems provide simple tests.

Theorem 1 An LSI system is BIBO stable iff

$$\sum_{n=-\infty}^{\infty} |h_n| < \infty .$$

Proof: (Sufficiency)

Given $\sum_n |h_n| < \gamma < \infty$ and choose any $0 < \alpha < \infty$ and any $\{x_n\}$ satisfying $|x_n| < \alpha$ for all n.

Then:

$$\begin{aligned} |y_n| &= \left| \sum_{m=-\infty}^{\infty} h_m x_{n-m} \right| \\ &\leq \sum_{m=-\infty}^{\infty} |h_m x_{n-m}| \\ &= \sum_{m=-\infty}^{\infty} |h_m|\,|x_{n-m}| \\ &< \alpha \sum_{m=-\infty}^{\infty} |h_m| \\ &< \alpha \bullet \gamma < \infty \end{aligned}$$

$\Rightarrow$ output bdd, i.e., $|y_n| < \beta$ for all n, where $\beta = \alpha \bullet \gamma$.

(Necessity)

Given $\sum_m |h_m| = \infty$. Show there exists bdd $\{x_n\}$ with $y_{n_o} = \infty$ for some fixed n_o.

$\uparrow$

$\{y_n\}$ not bdd.

$$y_{n_o} = \sum_{m=-\infty}^{\infty} x_m h_{n_o-m}$$

Assuming $\{h_n\}$ is real-valued, choose $x_m = \text{sign}\left[h_{n_o-m}\right] = \pm 1$ (bdd) (define sign [0] to be + 1)

Then:

$$y_{n_o} = \sum_{m=-\infty}^{\infty} \text{sign}\left[h_{n_o-m}\right] h_{n_o-m}$$

$$= \sum_{m=-\infty}^{\infty} \left|h_{n_o-m}\right|$$

$$= \underset{\substack{\uparrow \\ k = n_o - m}}{\sum_{k=-\infty}^{\infty}} \left|h_k\right| = \infty \Rightarrow \{y_n\} \text{ not bdd}$$

Can also determine stability directly from the transfer function H(z).

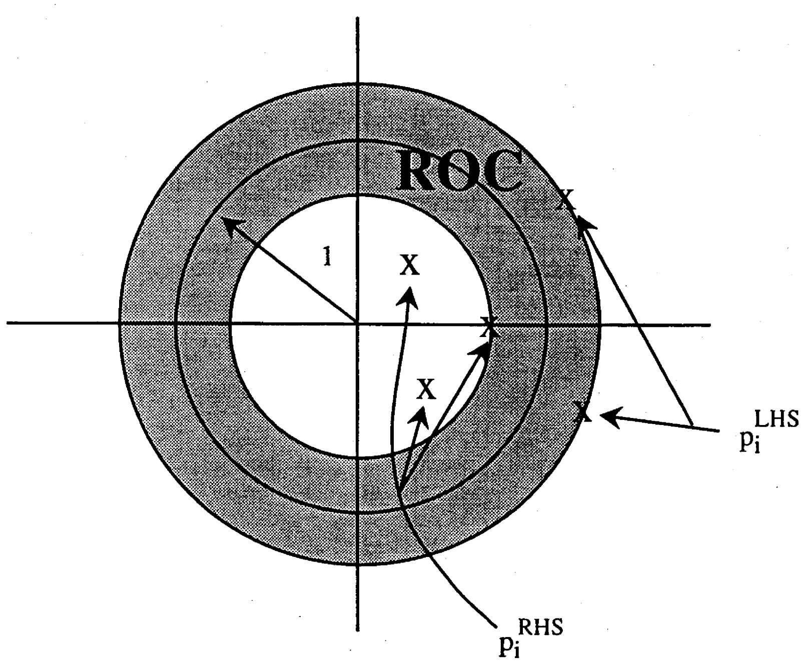
Theorem 2 An LSI system with a rational transfer function (in minimal form) is BIBO stable iff ROC_H includes the unit circle.

Proof: (Sufficiency)

Assume ROC_H includes u.c. Show $\sum_n |h_n| < \infty$.

If ROC_H includes u.c., then $|p_i^{RHS}| < 1$ and $|p_i^{LHS}| > 1$

since:



$\{h_n\}_{n=-\infty}^{\infty}$ has the form

$$h_n = \begin{cases} \sum_{i=1}^{K} a_i \left(p_i^{RHS}\right)^n & n \geq 0 \\ \sum_{i=1}^{L} b_i \left(p_i^{LHS}\right)^n & n < 0 \end{cases}$$

(can modify for repeated pole case)

Since $\left|p_i^{RHS}\right| < 1$ and

$\left|p_i^{LHS}\right| > 1$

we see

$$\begin{aligned} \sum_{n=-\infty}^{\infty} |h_n| &= \sum_{n=0}^{\infty} |h_n| + \sum_{n=-\infty}^{-1} |h_n| \\ &= \sum_{n=0}^{\infty} \left| \sum_{i=1}^{K} a_i \left(p_i^{RHS}\right)^n \right| + \sum_{n=-\infty}^{-1} \left| \sum_{i=1}^{L} b_i \left(p_i^{LHS}\right)^n \right| \\ &\leq \sum_{n=0}^{\infty} \sum_{i=1}^{K} |a_i| \left|p_i^{RHS}\right|^n + \sum_{n=-\infty}^{-1} \sum_{i=1}^{L} |b_i| \left|p_i^{LHS}\right|^n \\ &= \sum_{i=1}^{K} |a_i| \underbrace{\sum_{n=0}^{\infty} \left|p_i^{RHS}\right|^n}_{<\infty} + \sum_{i=1}^{L} |b_i| \underbrace{\sum_{n=-\infty}^{-1} \left|p_i^{LHS}\right|^n}_{<\infty} \quad < \infty \ \checkmark \end{aligned}$$

(Necessity)

Assume $\sum_{n=-\infty}^{\infty} |h_n| < \infty$ (i.e., assume BIBO stable)

Show ROC_H includes u.c., i.e., show H(z) is finite for any point z on the unit circle.

$$\left.|H(z)|\right|_{|z|=1} = \left.\left| \sum_{n=-\infty}^{\infty} h_n z^{-n} \right|\right|_{|z|=1}$$

$$\leq \sum_{n=-\infty}^{\infty} |h_n||z^{-n}| \Bigg|_{|z|=1}$$

$$= \sum_{n=-\infty}^{\infty} |h_n| \left(\frac{1}{|z|}\right)^n \Bigg|_{|z|=1}$$

$$= \sum_{n=-\infty}^{\infty} |h_n| < \infty$$

$\Rightarrow$ ROC_H includes u.c.

Corollary of Theorem 2: A causal LSI system with a rational TF (in minimal form) is BIBO stable iff all poles are inside the u.c.

Proof: Causal system has all poles between origin and ROC_H with at least one pole on the inner radius of ROC_H. So, poles are inside u.c. iff ROC_H includes the u.c. But, by Theorem 2, ROC_H includes the u.c. iff system is BIBO stable.

Example

$$h_n = (\cos\theta n) u_n$$

$$\sum_{n=-\infty}^{\infty} |h_n| = \sum_{n=0}^{\infty} |\cos\theta n| = \infty$$

Alternatively, can check poles are on u.c.

$\Rightarrow$ Not BIBO stable.

Example

$$H(z) = \frac{z^2 - 3z + 2}{z^3 - 2z^2 + \frac{z}{2} - 1}, \text{ causal}$$

Factor denominator:

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

$$\text{poles} = \pm j \frac{1}{\sqrt{2}}$$

Causal, and poles inside u.c.

$\Rightarrow$ BIBO stable

Note: As we did in this example, we must cancel factors that are common to the numerator and denominator before applying the stability text.

Example

$$H(z) = \frac{z}{z+100} \qquad |z| < 100$$

ROC_H includes u.c. $\Rightarrow$ BIBO stable. (In this example, h_n is left-sided.)

Example

$$h_n = \begin{cases} 4^n & 0 \le n \le 10^6 \\ n\left(\frac{1}{2}\right)^n & 10^6 + 1 \le n < \infty \\ 0 & n < 0 \end{cases}$$

$$\sum_{n=-\infty}^{\infty} |h_n| = \sum_{n=0}^{10^6} 4^n + \sum_{n=10^6+1}^{\infty} n\left(\frac{1}{2}\right)^n \quad < \quad \infty$$

$\Rightarrow$ BIBO stable

ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 20

D. Munson

Example

$$H(z) = \frac{z}{\left(z - \frac{1}{4}\right)(z-2)}$$

is known to be stable. Can system be causal?

Answer: No, h_n must be two-sided. Why? Because stability of system implies ROC_H includes the unit circle so that $ROC_H = \left\{z: \frac{1}{4} < |z| < 2\right\}$.

Example

$$y_n = (x_n)^2$$

This system is not linear. Therefore, we cannot apply a stability test involving either the unit-pulse response or transfer function. (For this system, $y_n \neq h_n * x_n$ and $Y(z) \neq H(z)\,X(z)$.) Instead, we appeal to the definition of BIBO stability. Suppose x_n is bounded:

$$|x_n| < \alpha < \infty$$

Then

$$|y_n| = |x_n|^2 < \alpha^2 < \infty .$$

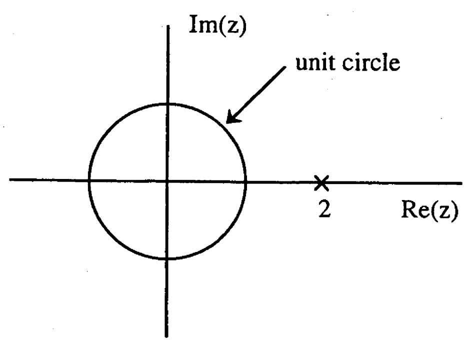
Thus, any bounded input produces a bounded output $\Rightarrow$ BIBO stable.

Unbounded Outputs

Given an unstable LSI system, how do we find a bounded input that will cause an unbounded output? Illustrate by example for some causal systems:

Example

$$H(z) = \frac{z}{z-2} \qquad |z| > 2$$

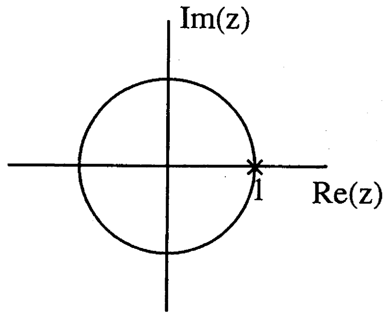


$\Rightarrow h_n \;=\; 2^n u_n$

Here, since h_n grows without bound, almost any bounded input x_n will cause the output y_n to be unbounded. For example, take $x_n = \delta_n$ so that $y_n = h_n$.

Example

$$H(z) \;=\; \frac{z}{z-1} \qquad |z| > 1$$



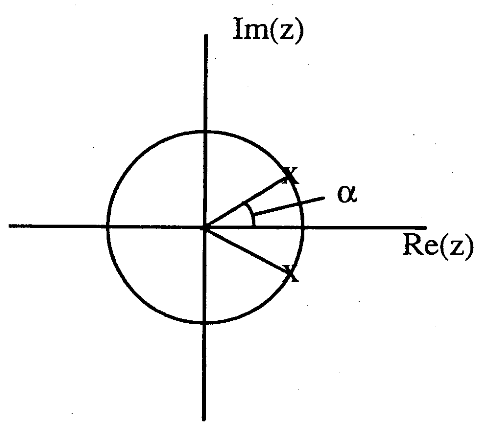
$\Rightarrow \quad h_n \;=\; u_n$

Here we could choose $x_n = u_n$ (which is bounded) so that y_n will be a ramp. In the z-transform domain, this corresponds to forcing Y(z) to have a double pole at z = 1:

$$Y(z) \;=\; H(z)\,X(z) = \frac{z^2}{(z-1)^2}$$

Example

$$H(z) \;=\; \frac{z^2 - z\cos\alpha}{(z-e^{j\alpha})(z-e^{-j\alpha})} \qquad |z| > 1$$



Here, we have a complex conjugate pair of poles on the unit circle corresponding to the sinusoid

$$h_n \;=\; \cos(\alpha n)u_n$$

Thinking in the transform domain, we note that choosing $x_n = h_n$ will cause Y(z) to have double poles at $z = e^{\pm j\alpha}$, which will in turn cause y_n to have the form n times $\cos \alpha\, n$, which is unbounded.

<u>Comment</u>: From the above examples, we see (for causal systems) that for poles outside the unit circle, we would have to work hard to find a bounded input that will <u>not</u> cause an unbounded output. For poles on the unit circle, we must work hard to find a bounded input that <u>will</u> cause the output to be unbounded.

Impulse Distribution ~ Digression

Def. A distribution maps a function to a number.

Def. The impulse δ is the distribution:

$$\delta[f(t)] \triangleq f(0) \qquad (1)$$

where f may be any continuous function.

Often write δ as $\delta(t)$ and use the notation

$$\boxed{\int_{-\infty}^{\infty} \delta(t) f(t) dt \triangleq f(0)} \qquad (2)$$

Special case:

$$\int_{-\infty}^{\infty} \delta(t) dt = 1$$

But:

a) $\delta(t)$ is not a function.

b) The integral sign in (2) is not an integral.

c) (2) is just alternate notation for (1).

Can generalize to a broader family of impulse distributions:

$$\delta_{t_o} [f(t)] \triangleq f(t_o) \qquad (3)$$

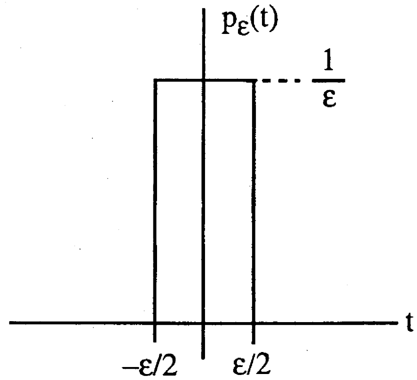
Notation usually used is

$$\int_{-\infty}^{\infty} \delta(t - t_o) f(t)\, dt \triangleq f(t_o) \qquad (4)$$

but (4) is just alternate notation for (3).

$\delta(t)$ is <u>thought of</u> as a limit of tall, narrow pulses, each having area = 1, e.g.,

$$\delta(t) = \lim_{\varepsilon \to 0} p_\varepsilon(t) \text{ with}$$



But, in fact, this <u>limit does not exist</u> at t = 0; δ is <u>not</u> a function. Here is what's true:

$$f(0) = \lim_{\varepsilon \to 0} \int_{-\infty}^{\infty} p_\varepsilon(t)\, f(t)\, dt = \underbrace{\int_{-\infty}^{\infty} \delta(t) f(t)\, dt}_{\uparrow \text{ defined by (2)}} \tag{5}$$

and

$$f(t_o) = \lim_{\varepsilon \to 0} \int_{-\infty}^{\infty} p_\varepsilon(t - t_o) f(t)\, dt = \underbrace{\int_{-\infty}^{\infty} \delta(t - t_o) f(t)\, dt}_{\uparrow \text{ defined by (4)}} \tag{6}$$

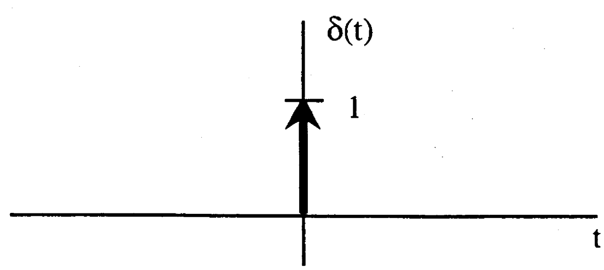
Saying $\delta(t) = \lim_{\varepsilon \to 0} p_\varepsilon(t)$ (7)

or

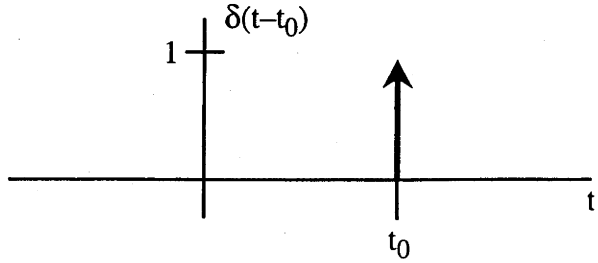
$$\delta(t-t_o) = \lim_{\varepsilon \to 0} p_\varepsilon(t-t_o) \tag{8}$$

brings the limit inside the integral, which is mathematically incorrect and is an abuse of notation. Equations (7) and (8) actually mean (5) and (6), respectively.

$\delta(t)$ is usually pictured as:



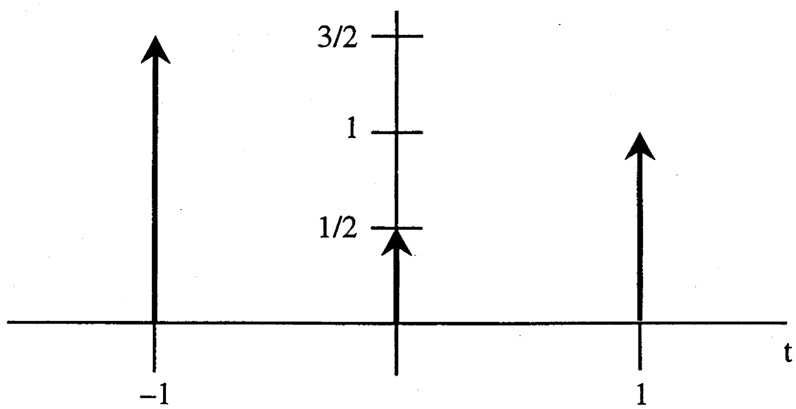
$\delta(t-t_o)$ is pictured as



Example

$$\frac{1}{2}\,\delta(t) + \delta(t-1) + \frac{3}{2}\,\delta(t+1)$$

is pictured as:



ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 21

D. Munson

Discrete-Time Fourier Transform (DTFT)

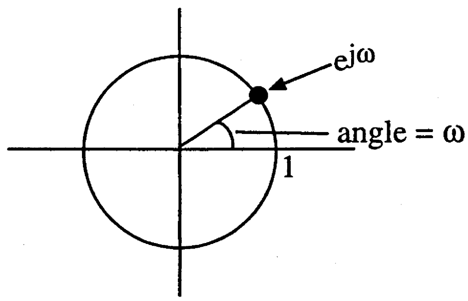
Def.

$$\boxed{X_d(\omega) = \sum_{n=-\infty}^{\infty} x_n \, e^{-j\omega n}}$$

real — if sum exists

X_d is a slice of $X(z)$ around the unit circle, i.e.,

$$X_d(\omega) = X(e^{j\omega})$$



The DTFT exists as a regular function iff ROC_X includes the unit circle. (For the case of sinusoidal sequences, where $X(z)$ has poles on the u.c., we will be able to define X_d in terms of impulse distributions.)

We will soon refer to the variable ω in $X_d(\omega)$ as digital frequency. In ECE 210 we used this same variable to represent analog frequency in the continuous-time Fourier transform. In ECE 310 we will use the variable Ω to denote continuous-time frequency. Of course, we could use any variables we like for the DTFT and FT. However, by selecting ω to represent digital frequency and Ω to represent analog frequency, we will be in agreement with most DSP textbooks.

Inverse DTFT:

$$\left[x_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} X_d(\omega) e^{j\omega n} \, d\omega \right]$$

Pf:

$$\frac{1}{2\pi}\int_{-\pi}^{\pi} X_d(\omega)e^{j\omega n}\,d\omega$$

$$= \frac{1}{2\pi}\int_{-\pi}^{\pi} \sum_{m=-\infty}^{\infty} x_m\, e^{-j\omega m}\, e^{j\omega n}\,d\omega$$

$$= \sum_{m=-\infty}^{\infty} x_m \underbrace{\frac{1}{2\pi}\int_{-\pi}^{\pi} e^{j\omega(n-m)}\,d\omega}$$

$$= \begin{cases} 1 & m = n \\ \left.\dfrac{e^{j\omega(n-m)}}{2\pi j(n-m)}\right|_{-\pi}^{\pi} & m \neq n \end{cases}$$

$$= \begin{cases} 1 & m = n \\ 0 & m \neq n \end{cases}$$

$$\Rightarrow \quad \frac{1}{2\pi}\int_{-\pi}^{\pi} X_d(\omega)\, e^{j\omega n}\,d\omega = x_n \qquad \checkmark$$

Properties of DTFT

1) Linearity:

$$\text{DTFT}\left[\{a\,x_n + b\,y_n\}\right] = a\,X_d(\omega) + b\,Y_d(\omega)$$

Pf: The DTFT is a special case of the z-transform.

2) Periodic: $X_d(\omega+2\pi) = X_d(\omega)$

$$\text{Pf: } X_d(\omega + 2\pi) = \sum_{n=-\infty}^{\infty} x_n\, e^{-j(\omega+2\pi)n}$$

$$= \sum_{n=-\infty}^{\infty} x_n\, e^{-j\omega n} = X_d(\omega) \qquad \checkmark$$

3) For real-valued x_n,

a) $\text{Re}\left[X_d(-\omega)\right] = \text{Re}\left[X_d(\omega)\right]$ (even)

b) $\text{Im}\left[X_d(-\omega)\right] = -\text{Im}\left[X_d(\omega)\right]$ (odd)

Pf: If x_n is real, then

$$\mathrm{Re}\left[X_d(\omega)\right] = \sum_n x_n \cos\omega n$$

$$= \sum_n x_n \cos(-\omega n) = \mathrm{Re}\left[X_d(-\omega)\right] \quad \checkmark$$

$$\mathrm{Im}\left[X_d(\omega)\right] = -\sum_n x_n \sin\omega n$$

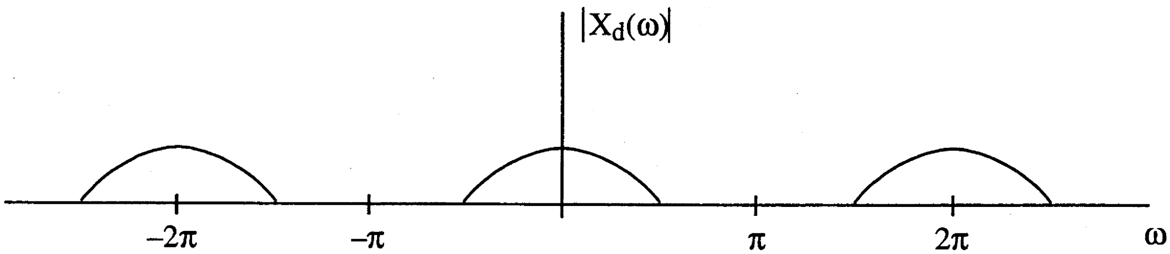
$$= \sum_n x_n \sin(-\omega n)$$

$$= -\mathrm{Im}\left[X_d(-\omega)\right] \quad \checkmark$$

4) For real-valued x_n,

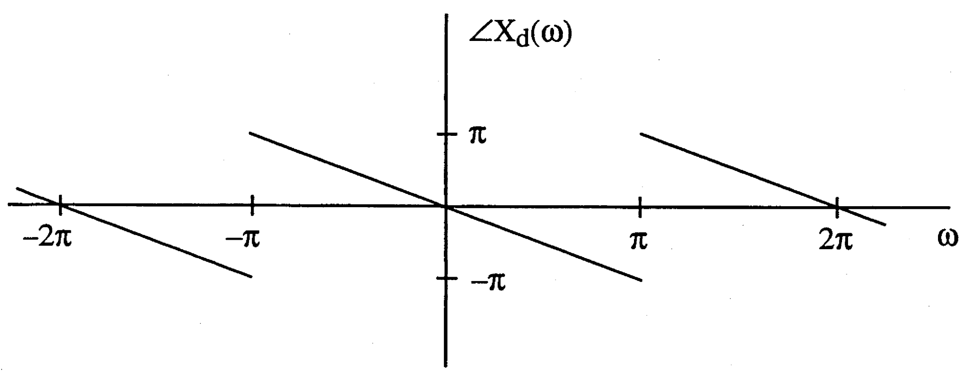
a) $|X_d(-\omega)| = |X_d(\omega)|$ (even)

For example, $|X_d(\omega)|$ might look like



b) $\angle X_d(-\omega) = -\angle X_d(\omega)$ (odd)

For example, $\angle X_d(\omega)$ might look like



Pf:

$$|X_d(\omega)| = \left[\left(\sum_n x_n \cos\omega n\right)^2 + \left(-\sum_n x_n \sin\omega n\right)^2\right]^{1/2}$$

$$= \left[\left(\sum_n x_n \cos(-\omega n)\right)^2 + \left(-\sum_n x_n \sin(-\omega n)\right)^2\right]^{1/2}$$

$$= |X_d(-\omega)|$$

$$\angle X_d(\omega) = \text{atan}\left[\frac{-\sum_n x_n \sin\omega n}{\sum_n x_n \cos\omega n}\right]$$

$$= \text{atan}\,\frac{\sum_n x_n \sin(-\omega n)}{\sum_n x_n \cos(-\omega n)}$$

$$\underset{\substack{\uparrow \\ \text{since} \\ \text{atan is odd}}}{=} -\text{atan}\left[\frac{-\sum_n x_n \sin(-\omega n)}{\sum_n x_n \cos(-\omega n)}\right]$$

$$= -\angle X_d(-\omega)$$

5) Shifting:

$$\text{DTFT}\left[\{x_{n\pm k}\}_{n=-\infty}^{\infty}\right] = e^{\pm j\omega k} X_d(\omega)$$

Pf: The DTFT is a special case of the z-transform with $z = e^{j\omega}$.

6) Modulation:

a) $y_n = e^{j\omega_o n} x_n \leftrightarrow Y_d(\omega) = X_d(\omega - \omega_o)$

b) $y_n = (\cos \omega_o n)\, x_n \leftrightarrow Y_d(\omega) = \frac{1}{2}\left[X_d(\omega - \omega_o) + X_d(\omega + \omega_o)\right]$

Proof of a):

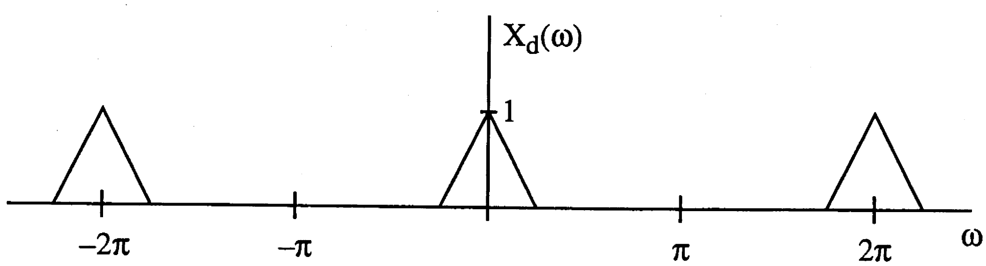
$$Y_d(\omega) = \sum_{n=-\infty}^{\infty} e^{j\omega_o n} x_n e^{-j\omega_o n}$$

$$= \sum_{n=-\infty}^{\infty} x_n e^{-j(\omega-\omega_o)n}$$

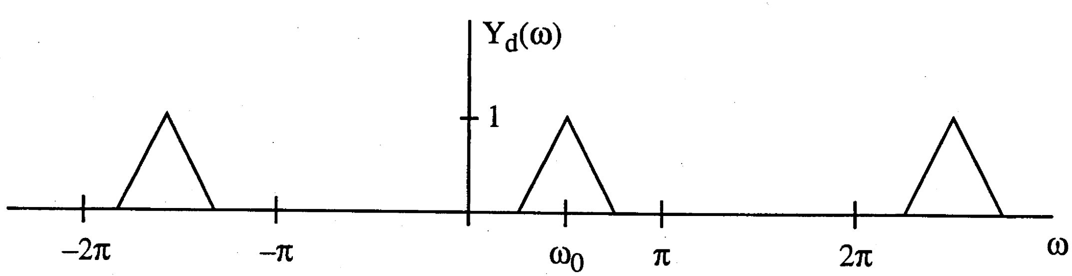
$$= X_d(\omega-\omega_o) \quad ✓$$

Example

If $X_d(\omega)$ looks like



Then $Y_d(\omega) = \text{DTFT}\left[e^{j\omega_o n} x_n\right]$ is



Proof of b):

$$Y_d(\omega) = \text{DTFT}\left[\frac{1}{2}\left(e^{j\omega_o n} + e^{-j\omega_o n}\right)x_n\right]$$

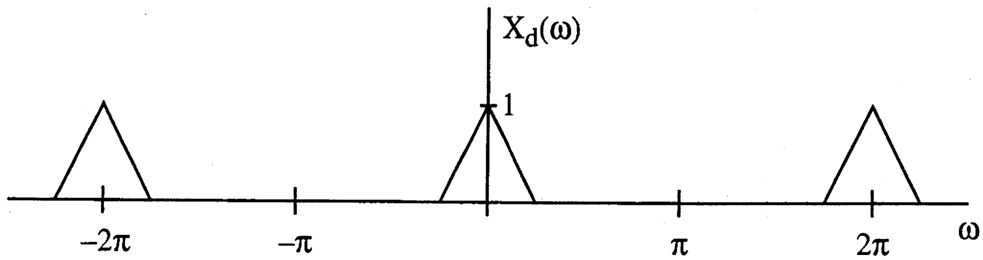
$$= \frac{1}{2}\text{DTFT}\left[e^{j\omega_o n} x_n\right] + \frac{1}{2}\text{DTFT}\left[e^{-j\omega_o n} x_n\right]$$

$$= \frac{1}{2}X_d\left(\omega-\omega_o\right) + \frac{1}{2}X_d\left(\omega+\omega_o\right) \quad ✓$$

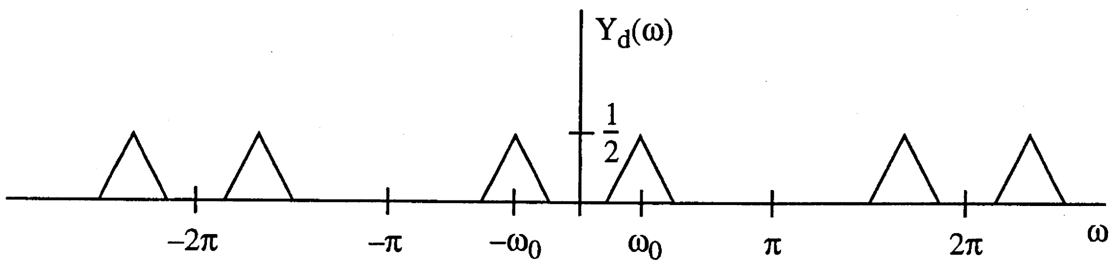
where the second inequality follows by linearity of the DTFT, and the third inequality follows by part a).

Example

If $X_d(\omega)$ looks like



then $Y_d(\omega) = \text{DTFT}\left[(\cos\omega_o n)x_n\right]$ is



7) Convolution:

$$y_n = h_n * x_n \leftrightarrow Y_d(\omega) = H_d(\omega)\, X_d(\omega)$$

$\uparrow$
assuming DTFTs are defined

Pf: The DTFT is a special case of the z-transform, so 7) follows from the z-transform convolution property.

8) Parseval:

$$\sum_{n=-\infty}^{\infty} |x_n|^2 = \frac{1}{2\pi}\int_{-\pi}^{\pi} |X_d(\omega)|^2\, d\omega$$

Pf:

$$\sum_{n=-\infty}^{\infty} x_n x_n^* = \sum_{n=-\infty}^{\infty} x_n \left[\frac{1}{2\pi}\int_{-\pi}^{\pi} X_d(\omega) e^{j\omega n}\, d\omega\right]^*$$

$$= \sum_{n=-\infty}^{\infty} x_n \frac{1}{2\pi} \int_{-\pi}^{\pi} X_d^*(\omega) e^{-j\omega n} \, d\omega$$

$$= \frac{1}{2\pi} \int_{-\pi}^{\pi} X_d^*(\omega) \sum_{n=-\infty}^{\infty} x_n \, e^{-j\omega n} \, d\omega$$

$$= \frac{1}{2\pi} \int_{-\pi}^{\pi} \left| X_d(\omega) \right|^2 d\omega \quad \checkmark$$

ECE 310 **DIGITAL SIGNAL PROCESSING** **D. Munson**
University of Illinois **Lecture 22**

Example

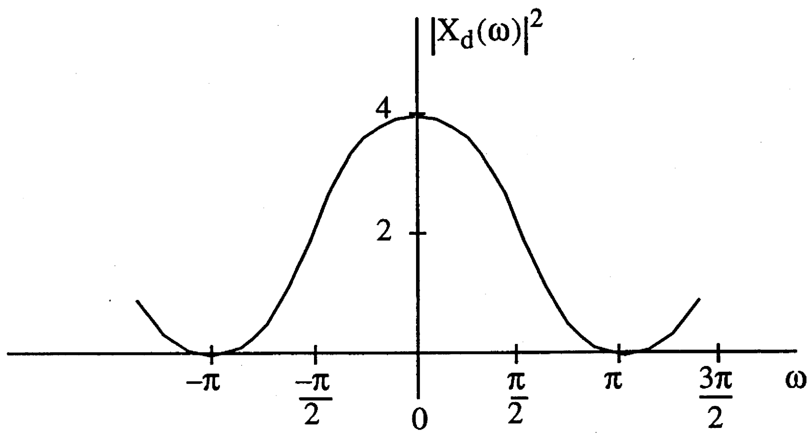
$$x_n = \delta_n + \delta_{n-1}$$

$$X_d(\omega) = 1 + e^{-j\omega}$$

$$|X_d(\omega)|^2 = (1 + \cos\omega)^2 + \sin^2\omega$$

$$= 1 + 2\cos\omega + \cos^2\omega + \sin^2\omega$$

$$= 2 + 2\cos\omega$$



Since $|X_d(\omega)|^2$ is both periodic with period 2π and symmetric around the origin, $|X_d(\omega)|^2$ is completely determined by its values on the interval $0 \le \omega \le \pi$. This is similarly true for $\angle X_d(\omega)$. Because of this, when $\{x_n\}$ is real (so that $|X_d(\omega)|$ and $\angle X_d(\omega)$ have even and odd symmetry, respectively) we will often plot $|X_d(\omega)|$ and $\angle X_d(\omega)$ on just the interval $0 \le \omega \le \pi$.

To find $\angle X_d(\omega)$ in this example, write

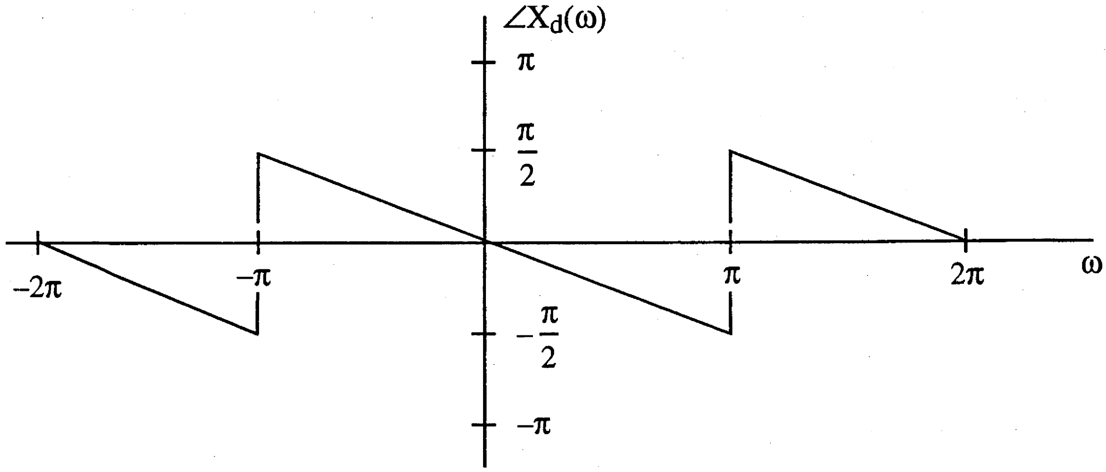
$$X_d(\omega) = 1 + e^{-j\omega} = e^{-j\frac{\omega}{2}}\left(e^{j\frac{\omega}{2}} + e^{-j\frac{\omega}{2}}\right)$$

$$= e^{-j\frac{\omega}{2}}\, 2\cos\left(\frac{\omega}{2}\right) \qquad (*)$$

Now, since $\cos\left(\frac{\omega}{2}\right) > 0$ for $-\pi < \omega < \pi$, equation (*) expresses X_d in polar form, so that

$$\angle X_d(\omega) = -\frac{\omega}{2} \qquad -\pi < \omega < \pi .$$

The phase is plotted as



Notice that the phase of $X_d(\omega)$ is an odd function.

Example

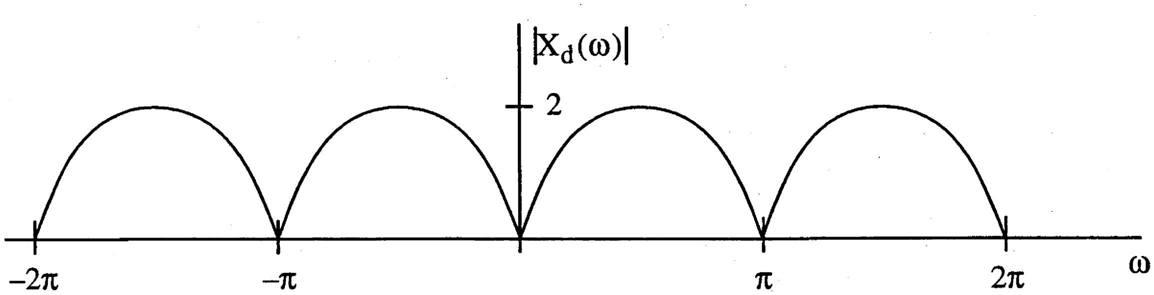
Plot $|X_d(\omega)|$ and $\angle X_d(\omega)$ for $x_n = \delta_{n-1} - \delta_{n+1}$

$$X_d(\omega) = e^{-j\omega} - e^{j\omega}$$

$$= -2j \sin\omega$$

$$\Rightarrow \quad |X_d(\omega)| = 2|\sin\omega| .$$

This is plotted as



Notice that $|X_d(\omega)|$ is again even, and its appearance is totally specified by $|X_d(\omega)|$ on just the interval $0 \le \omega \le \pi$. The phase of $X_d(\omega)$ is found by noting

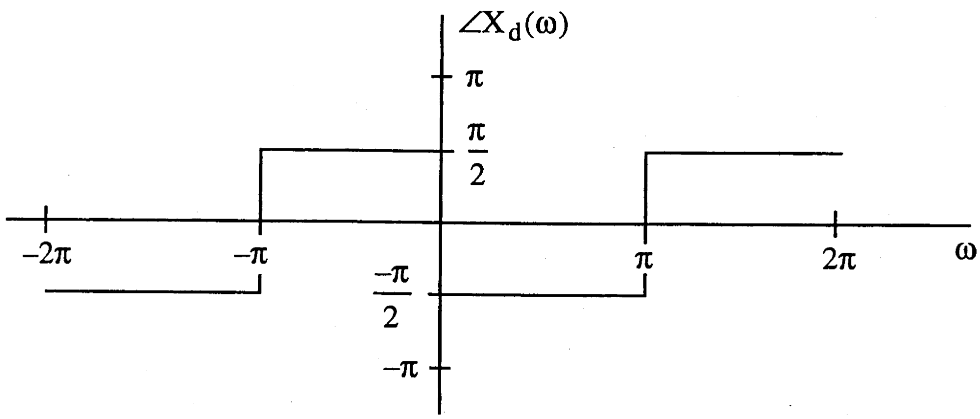
$$X_d(\omega) = \begin{cases} -2j\sin\omega & \{\omega : \sin\omega > 0\} \\ 2j\,|\sin\omega| & \{\omega : \sin\omega < 0\} \end{cases}$$

$$= \begin{cases} e^{-j\frac{\pi}{2}}\,2\sin\omega & 0 < \omega < \pi \\ e^{j\frac{\pi}{2}}\,2\,|\sin\omega| & -\pi < \omega < 0 \end{cases}$$

Both the top and bottom lines within the bracket are written in polar form, since $\sin\omega > 0$ for $0 < \omega < \pi$ and $|\sin\omega| > 0$. Thus,

$$\angle X_d(\omega) = \begin{cases} \frac{-\pi}{2} & 0 < \omega < \pi \\ \frac{\pi}{2} & -\pi < \omega < 0 \end{cases}$$

which is plotted as



Notice that $\angle X_d(\omega)$ is odd.

Cautionary Note: The symmetry properties of $|X_d(\omega)|$ and $\angle X_d(\omega)$ hold only for <u>real-valued</u> x_n. If x_n is not real, then the usual symmetry will not be present, as shown in the example below.

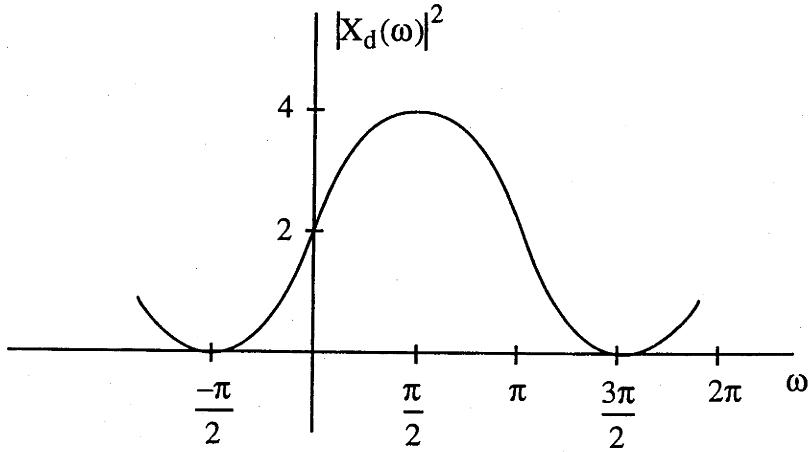
Example $\quad x_n = \delta_n + j\,\delta_{n-1}$

$$X_d(\omega) = 1 + j\,e^{-j\omega}$$

$$|X_d(\omega)|^2 = (1 + \sin\omega)^2 + \cos^2\omega$$

$$= 1 + 2\sin\omega + \sin^2\omega + \cos^2\omega$$

$$= 2 + 2\sin\omega$$



Here, we see that $|X_d(\omega)| = |X_d(-\omega)|$ does not hold.

Example $x_n = a^n u_n$, with 'a' real-valued and $|a| < 1$

Use

$$X_d(\omega) = X(z)\Big|_{z=e^{j\omega}}$$

X_d is well defined here since ROC_X includes the unit circle. We have

$$X(z) = \frac{z}{z-a}, \quad |z| > |a|$$

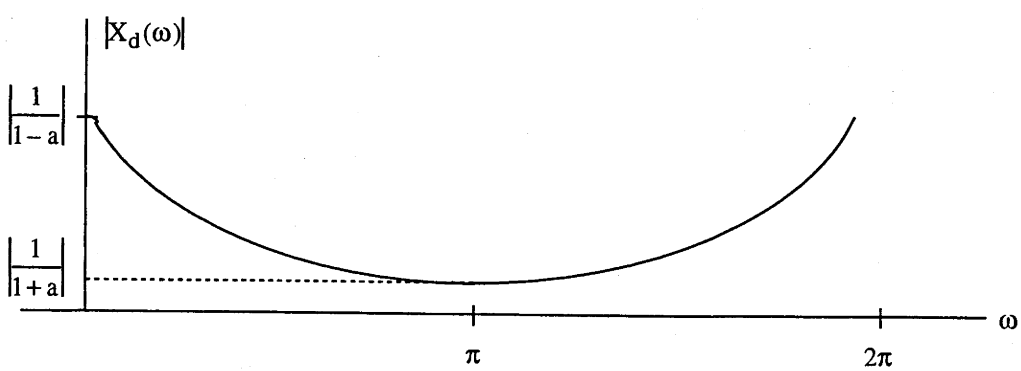
$$\Rightarrow X_d(\omega) = \frac{e^{j\omega}}{e^{j\omega} - a} = \frac{1}{1 - a\,e^{-j\omega}}$$

$$\Rightarrow |X_d(\omega)|^2 = \frac{1}{(1 - a\cos\omega)^2 + (a\sin\omega)^2}$$

$$= \frac{1}{1 - 2a\cos\omega + a^2\cos^2\omega + a^2\sin^2\omega}$$

$$= \frac{1}{1 + a^2 - 2a\cos\omega}$$

Picture assuming $0 < a < 1$:



Example $x_n = \cos \omega_o n \quad \forall\, n$

X(z) is undefined on the unit circle.

$$\text{So,}\quad X_d(\omega) = \sum_{n=-\infty}^{\infty} \cos(\omega_o n)\, e^{-j\omega n} = \,??$$

Claim:

$$X_d(\omega) = \pi\left[\delta(\omega-\omega_o)+\delta(\omega+\omega_o)\right] \qquad |\omega| \le \pi \qquad (*)$$

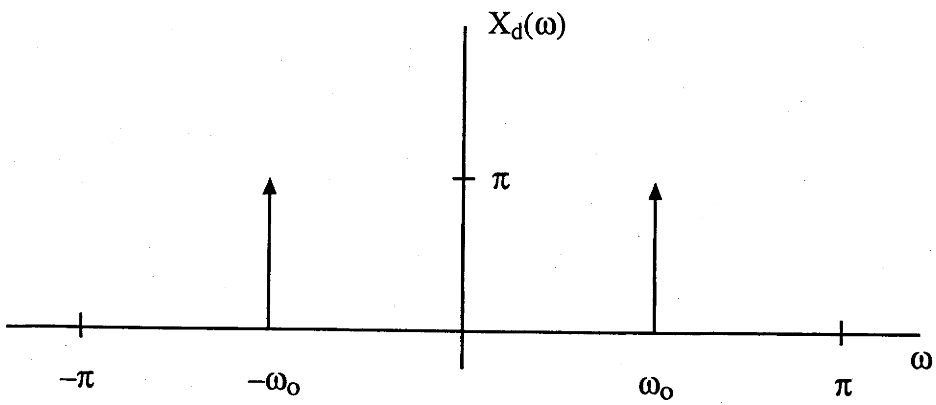
in the sense that using this in the DTFT^{-1} formula gives $\{x_n\}$. Here, $X_d(\omega)$ is a distribution, not a function.

Check that DTFT^{-1} of (*) gives cosine sequence:

$$\frac{1}{2\pi}\int_{-\pi}^{\pi} X_d(\omega)\, e^{j\omega n}\, d\omega = \frac{\pi}{2\pi}\int_{-\pi}^{\pi}\left[\delta(\omega-\omega_o)+\delta(\omega+\omega_o)\right] e^{j\omega n}\, d\omega$$

$$= \frac{1}{2}\left[e^{j\omega_o n}+e^{-j\omega_o n}\right] = \cos\omega_o n \qquad ✓$$

So:



Comment: If we had put tall, narrow rectangles in the above integral in place of $\delta(\omega-\omega_o)$ and $\delta(\omega+\omega_o)$, we would have obtained an approximation to $\cos(\omega_o n)$. The approximation improves as the rectangles get narrower and taller. You can even work this out in a closed-form!

Example

$$x_n = \begin{cases} \cos\omega_o n & 0 \le n \le N-1 \\ 0 & \text{else} \end{cases}$$

$$X_d(\omega) = \sum_{n=0}^{N-1} \cos(\omega_o n)\, e^{-j\omega n}$$

$$= \frac{1}{2} \sum_{n=0}^{N-1} \left[e^{j\omega_o n} + e^{-j\omega_o n} \right] e^{-j\omega n}$$

$$= \frac{1}{2} \sum_{n=0}^{N-1} e^{-j(\omega-\omega_o)n} + \frac{1}{2} \sum_{n=0}^{N-1} e^{-j(\omega+\omega_o)n}$$

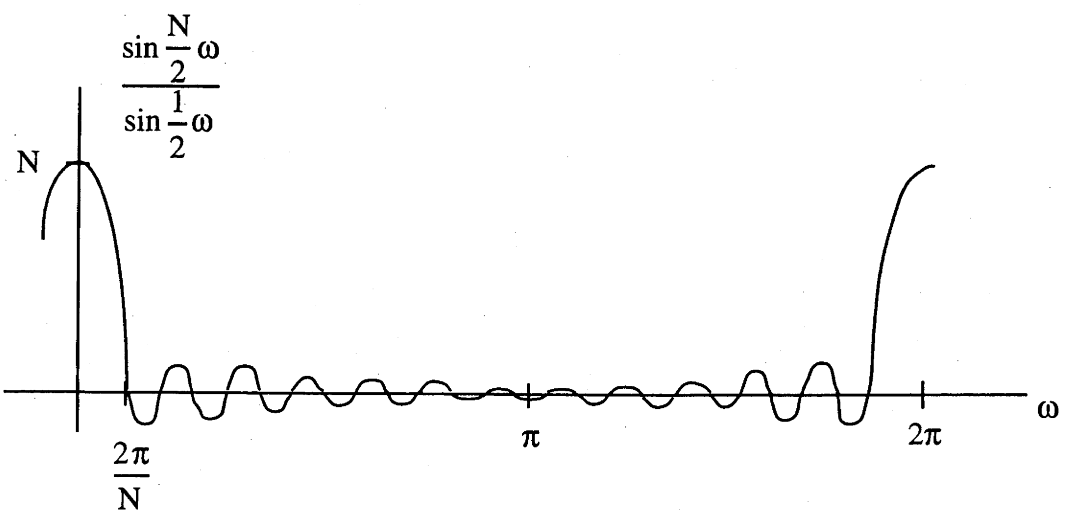
$$= \frac{1}{2} \frac{1-e^{-j(\omega-\omega_o)N}}{1-e^{-j(\omega-\omega_o)}} + \frac{1}{2} \frac{1-e^{-j(\omega+\omega_o)N}}{1-e^{-j(\omega+\omega_o)}}$$

$$= \frac{1}{2} \frac{e^{-j(\omega-\omega_o)\frac{N}{2}}}{e^{-j(\omega-\omega_o)\frac{1}{2}}} \frac{e^{j(\omega-\omega_o)\frac{N}{2}} - e^{-j(\omega-\omega_o)\frac{N}{2}}}{e^{j(\omega-\omega_o)\frac{1}{2}} - e^{-j(\omega-\omega_o)\frac{1}{2}}} + \frac{1}{2} \frac{e^{-j(\omega+\omega_o)\frac{N}{2}}}{e^{-j(\omega+\omega_o)\frac{1}{2}}} \frac{e^{j(\omega+\omega_o)\frac{N}{2}} - e^{-j(\omega+\omega_o)\frac{N}{2}}}{e^{j(\omega+\omega_o)\frac{1}{2}} - e^{-j(\omega+\omega_o)\frac{1}{2}}}$$

$$= \frac{1}{2} e^{-j(\omega-\omega_o)\frac{N-1}{2}} \underbrace{\frac{\sin\left[(\omega-\omega_o)\frac{N}{2}\right]}{\sin\left[(\omega-\omega_o)\frac{1}{2}\right]}}_{\uparrow} + \frac{1}{2} e^{-j(\omega+\omega_o)\frac{N-1}{2}} \frac{\sin\left[(\omega+\omega_o)\frac{N}{2}\right]}{\sin\left[(\omega+\omega_o)\frac{1}{2}\right]} \qquad (\Delta)$$

periodic sinc centered at $\omega = \omega_o$

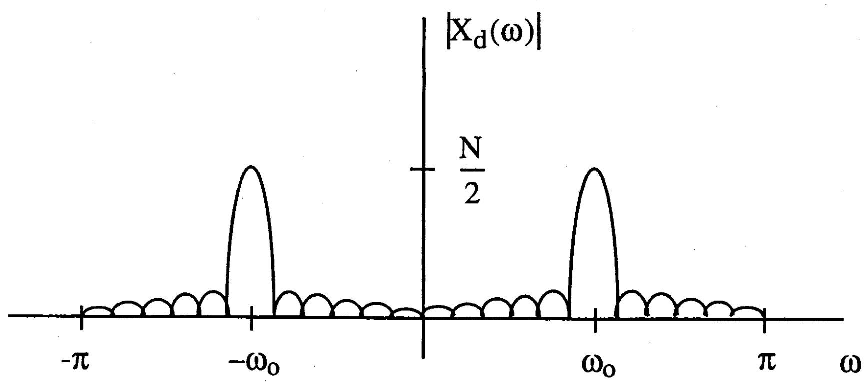
Periodic sinc :



For large N, the main lobe in the periodic sinc is narrow and tall so that the terms in (Δ) don't overlap much and

$$|X_d(\omega)| \approx \frac{1}{2}\left|\frac{\sin\left[(\omega-\omega_o)\frac{N}{2}\right]}{\sin\left[(\omega-\omega_o)\frac{1}{2}\right]}\right| + \frac{1}{2}\left|\frac{\sin\left[(\omega+\omega_o)\frac{N}{2}\right]}{\sin\left[(\omega+\omega_o)\frac{1}{2}\right]}\right|$$

Picture:



As $N \to \infty$ this "approaches" two impulses.

ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 23

D. Munson

Frequency Response and Digital Frequency

Have:

$$Y_d(\omega) = \underset{\uparrow}{=} H_d(\omega)\, X_d(\omega)$$

from convolution property of DTFT, where $H_d(\omega)$ = DTFT of u.p.r.

$H_d(\omega)$ is called the <u>frequency response</u>. Why?

If input is $x_n = e^{j\omega_o n}$ then

$$y_n = \sum_{m=-\infty}^{\infty} h_m\, e^{j\omega_o (n-m)}$$

$$= e^{j\omega_o n} \sum_{m=-\infty}^{\infty} h_m\, e^{-j\omega_o m}$$

$$= H_d(\omega_o)\, e^{j\omega_o n} \qquad (\square)$$

So output is same as input except scaled by the constant $\underset{\substack{\uparrow \\ \text{depends on} \\ \text{input "frequency"}}}{H_d(\omega_o)}$,
i.e.,

$$e^{j\omega_o n} \longrightarrow \boxed{h_n} \longrightarrow H_d(\omega_o)\, e^{j\omega_o n}$$

$e^{j\omega_o n}$ is called an "eigen-sequence."

<u>Digression</u>

What does this mean:

$$e^{j\omega_o n} \longrightarrow \boxed{h(n)} \longrightarrow y(n) \ ?$$

If h(n) is real valued then this means:

$$y(n) = h(n) * \left[\cos \omega_o n + j \sin \omega_o n\right]$$

$$= h(n) * \cos \omega_o n + j\, h(n) * \sin \omega_o n$$

i.e., the diagram above is a concise representation of a pair of systems having real-valued inputs and outputs:

$$\cos \omega_o n \longrightarrow \boxed{h(n)} \longrightarrow y_R(n)$$

$$\sin \omega_o n \longrightarrow \boxed{h(n)} \longrightarrow y_I(n)$$

with

$$y(n) \;\stackrel{\Delta}{=}\; \left(y_R(n), y_I(n)\right)$$

$$= y_R(n) + j\, y_I(n)$$

Note that we can build this system.

If h(n) is complex-valued, then as we saw in Lecture 18, we can still build system:

$$\text{Write } h(n) = \underbrace{h_R(n) + j\, h_I(n)}_{\text{real}}$$

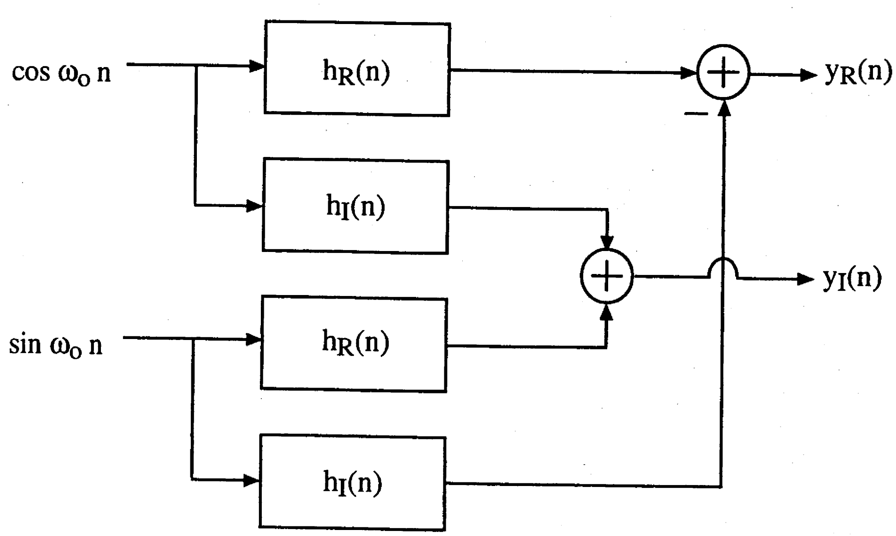
Then

$$y(n) = \left(h_R(n) + j\, h_I(n)\right) * \left(\cos\omega_o n + j \sin\omega_o n\right)$$

$$= h_R(n) * \cos \omega_o n - h_I(n) * \sin \omega_o n$$

$$+ j\left[h_I(n) * \cos\omega_o n + h_R(n) * \sin\omega_o n\right]$$

An implementation is:



(End of Digression)

Now, let's go back and use (□). This equation implies that the response to

$$\cos\omega_o n = \frac{1}{2}\left(e^{j\omega_o n} + e^{-j\omega_o n}\right)$$

is

$$y_n = \frac{1}{2}H_d(\omega_o)\, e^{j\omega_o n} + \frac{1}{2}\underbrace{H_d(-\omega_o)}\, e^{-j\omega_o n}$$

$$\sum_m h_m e^{j\omega_o m} = \left[\sum_m h_m e^{-j\omega_o m}\right]^*$$

for real h_m

$$= H_d^*(\omega_o)$$

$$\Rightarrow \quad y_n = \frac{1}{2}\overbrace{|H_d(\omega_o)|\, e^{j\angle H_d(\omega_o)}}^{H_d(\omega_o)}\, e^{j\omega_o n} + \frac{1}{2}\overbrace{|H_d(\omega_o)|\, e^{-j\angle H_d(\omega_o)}}^{H_d^*(\omega_o)}\, e^{-j\omega_o n}$$

$$= \frac{1}{2}|H_d(\omega_o)|\left\{e^{j[\omega_o n + \angle H_d(\omega_o)]} + e^{-j[\omega_o n + \angle H_d(\omega_o)]}\right\}$$

$$= |H_d(\omega_o)| \cos\left(\omega_o n + \angle H_d(\omega_o)\right)$$

Picture:

$$\cos(\omega_o n) \longrightarrow \boxed{h_n} \longrightarrow |H_d(\omega_o)| \cos\left(\omega_o n + \angle H_d(\omega_o)\right)$$

So, response to $\{\cos\omega_o n\}_{n=-\infty}^{\infty}$ is also a cos with

a) Same frequency
b) Amplitude $|H_d(\omega_o)|$
c) Phase $\angle H_d(\omega_o)$

<u>Note</u>: This result assumes H(z) is stable, because $H_d(\omega)$ exists only if ROC_H includes the unit circle.

Example

Given $y_n = x_n + 2x_{n-1}$

find the output due to $x_n = \cos\frac{\pi}{2} n \quad \forall\ n.$

<u>Solution</u>

$$Y_d(\omega) = X_d(\omega) + 2\,e^{-j\omega} X_d(\omega)$$

$$\Rightarrow H_d(\omega) = \frac{Y_d(\omega)}{X_d(\omega)} = 1 + 2\,e^{-j\omega}$$

Know $y_n = \left|H_d\left(\frac{\pi}{2}\right)\right| \cos\left(\frac{\pi}{2} n + \angle H_d\left(\frac{\pi}{2}\right)\right)$

Have $H_d\left(\frac{\pi}{2}\right) = 1 + 2\ e^{-j\frac{\pi}{2}} = 1 - j\,2 = \sqrt{5}\ e^{-j63.43^\circ}$

$$\Rightarrow \quad y_n = \sqrt{5}\ \cos\left(\frac{\pi}{2} n - 63.43^\circ\right)$$

Example

Given $H(z) = \dfrac{z}{z - \frac{1}{2}}$

find the output due to $x_n = \cos\frac{\pi}{4} n \quad \forall\ n.$

<u>Solution</u>

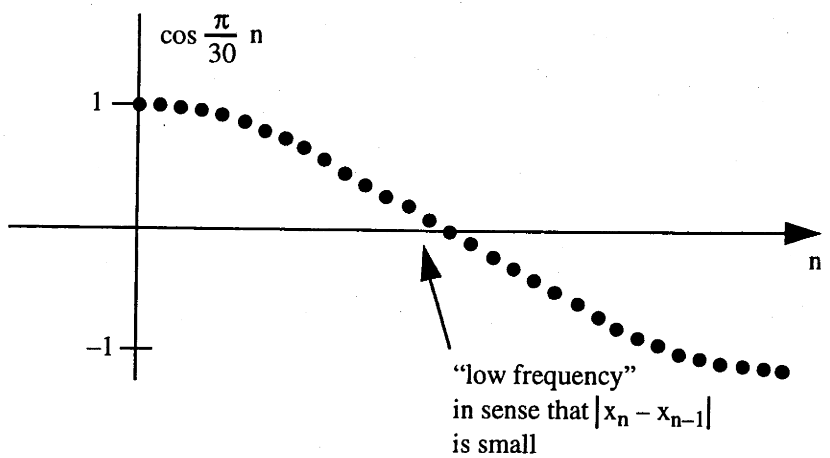
$$H_d(\omega) = \frac{1}{1 - \frac{1}{2} e^{-j\omega}}$$

$$\Rightarrow H_d\left(\frac{\pi}{4}\right) = \frac{1}{1 - \frac{1}{2} e^{-j\frac{\pi}{4}}} = \frac{1}{1 - \frac{\sqrt{2}}{4} + j\frac{\sqrt{2}}{4}} = 1.36\ e^{-j28.68^\circ}$$

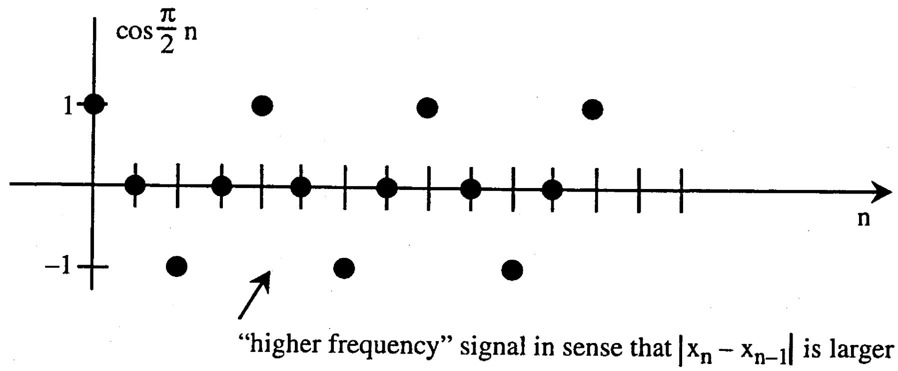
$$\Rightarrow \quad y_n = 1.36 \cos\left(\frac{\pi}{4} n - 28.68^\circ\right)$$

But, why call ω_o in $\cos(\omega_o n)$ "frequency?"

Suppose $\omega_o = \frac{\pi}{30}$. Plot $x_n = \cos \frac{\pi}{30} n$:



Suppose $\omega_o = \frac{\pi}{2}$. Plot $x_n = \cos \frac{\pi}{2} n$:



Lowest digital frequency:

$\omega_o = 0 \Rightarrow x_n = \cos(0 \bullet n) = 1 = \text{constant}.$

Highest digital frequency:

$\omega_o = \pi \Rightarrow x_n = \cos(\pi n) = (-1)^n$ so that $|x_n - x_{n-1}|$ is maximized for sinusoid of unit amplitude.

Why is $H_d(\omega)$ periodic with period 2π? Has to be! Note:

$$x_n = \cos\omega_o n \longrightarrow \boxed{H_d(\omega)} \longrightarrow |H_d(\omega_o)| \bullet \cos\left(\omega_o n + \angle H_d(\omega_o)\right)$$

$$\Updownarrow$$

$$x_n = \cos(\omega_o + 2\pi)n \longrightarrow \boxed{H_d(\omega)} \longrightarrow |H_d(\omega_o + 2\pi)| \bullet \cos\left[(\omega_o + 2\pi)n + \angle H_d(\omega_o + 2\pi)\right]$$

But, these two inputs with "different frequencies" are in fact identical sequences. So,

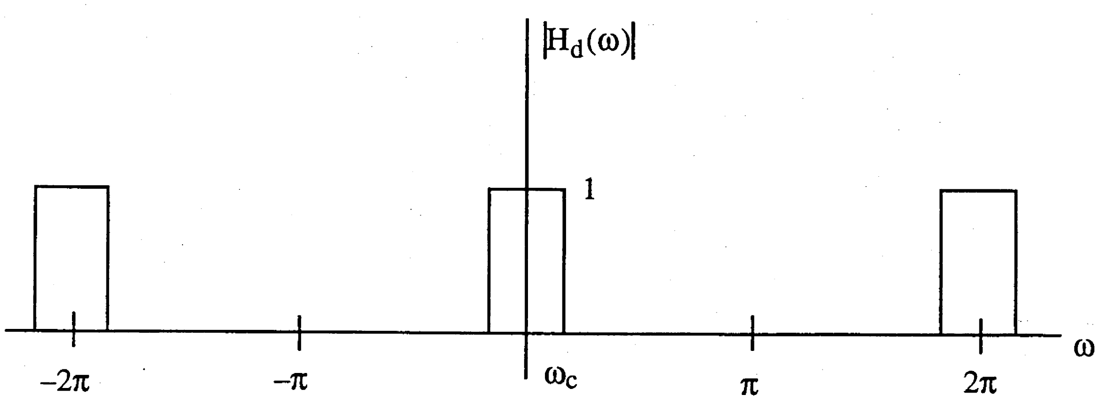
$$\Rightarrow \quad |H_d(\omega_o)| \quad = \quad |H_d(\omega_o + 2\pi)|$$

and

$$\angle H_d(\omega_o) \quad = \quad \angle H_d(\omega_o + 2\pi)$$

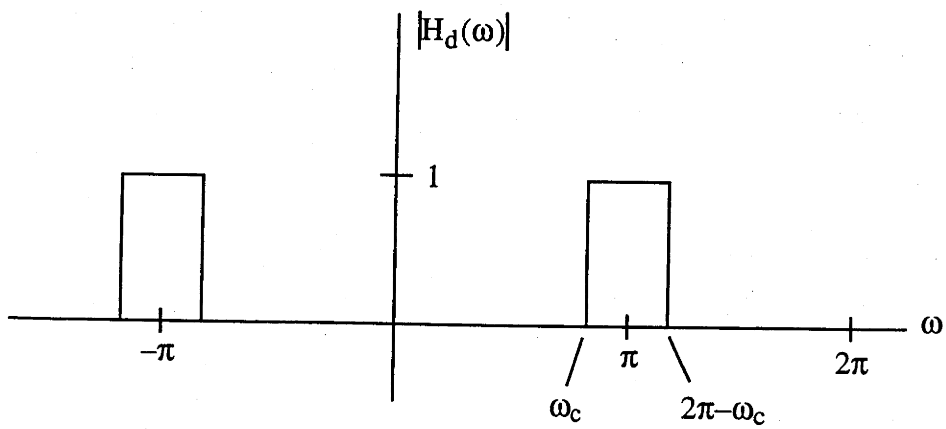
$$\Rightarrow \quad H_d(\omega_o) \quad = \quad H_d(\omega_o + 2\pi)$$

Ideal digital low-pass filter (LPF) frequency response:



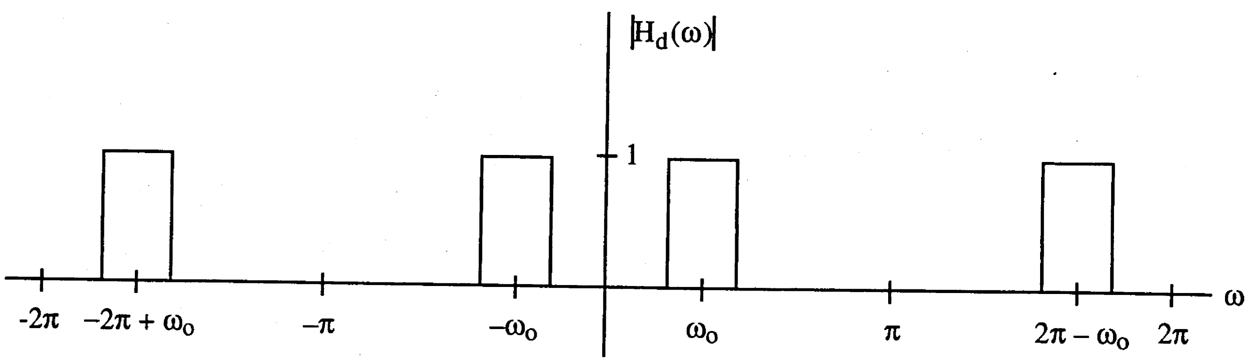
"Passes" all sinusoids having $|\omega_o| \le \omega_c$. Completely attenuates all others.

Ideal digital high-pass filter (HPF) frequency response:



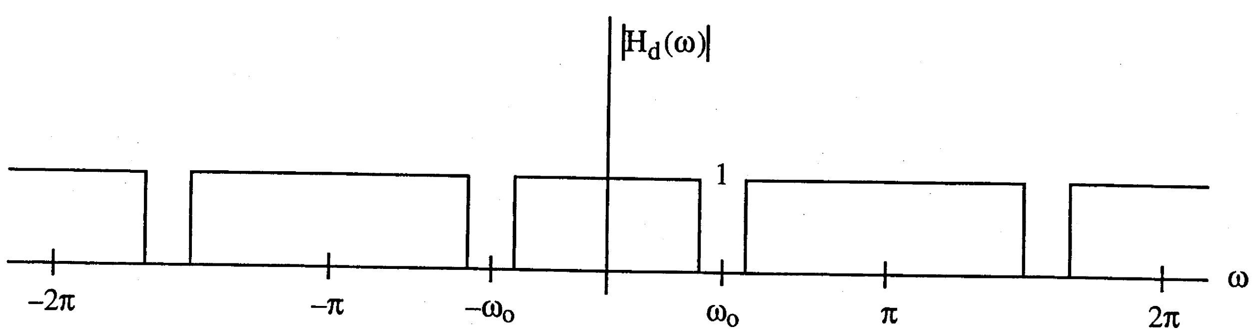
"Passes" all sinusoids having $\omega_c \le |\omega_o| \le 2\pi - \omega_c$. Attenuates all others.

Ideal digital band-pass filter (BPF) frequency response:



Passes all frequencies in band centered at ω_o. Attenuates all others.

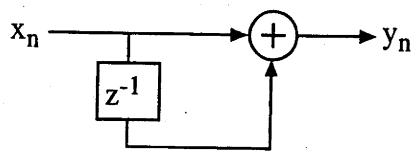
Ideal digital band-stop filter (BSF) frequency response:



Attenuates all frequencies in band centered at ω_o. When the stop-band is narrow, this is also called a notch filter.

Actual frequency responses using finite-order $H(z)$ give only an approximation to ideal $H_d(\omega)$.

Example



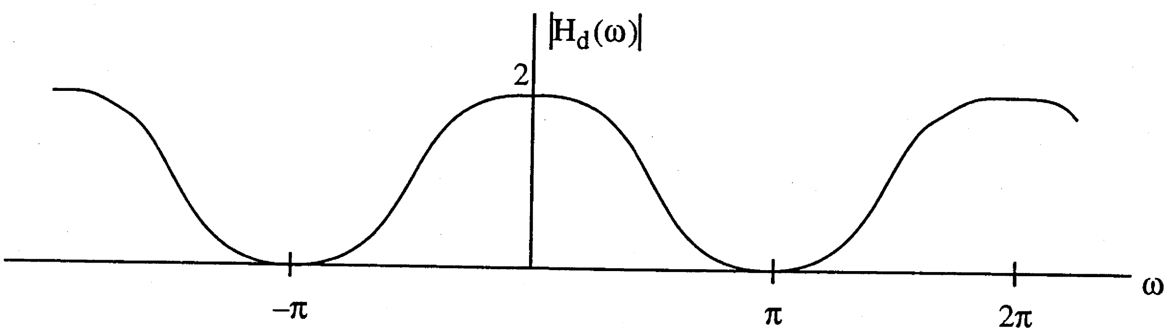
Have:

$$y_n = x_n + x_{n-1}$$

$$\Rightarrow \quad H(z) = 1 + z^{-1}$$

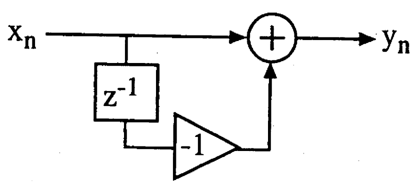
$$\Rightarrow H_d(\omega) = 1 + e^{-j\omega}$$

$$\Rightarrow |H_d(\omega)| = \sqrt{2 + 2\cos\omega}$$



So, this is a crude LPF.

Example



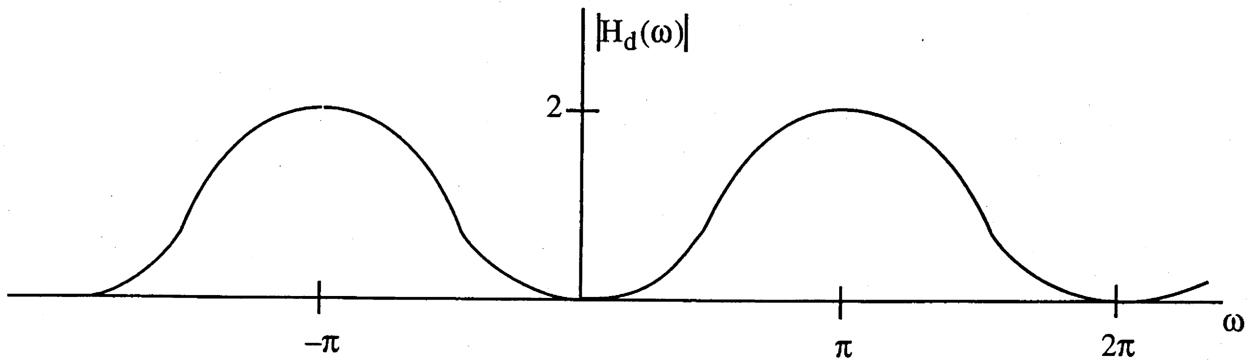
Have:

$$y_n = x_n - x_{n-1}$$

$$\Rightarrow H(z) = 1 - z^{-1}$$

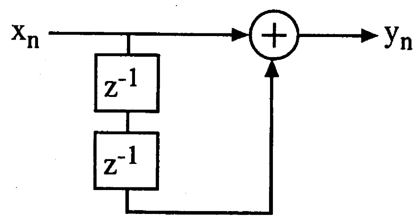
$$\Rightarrow H_d(\omega) = 1 - e^{-j\omega}$$

$$|H_d(\omega)| = \sqrt{2 - 2\cos\omega}$$



So, this is a crude HPF.

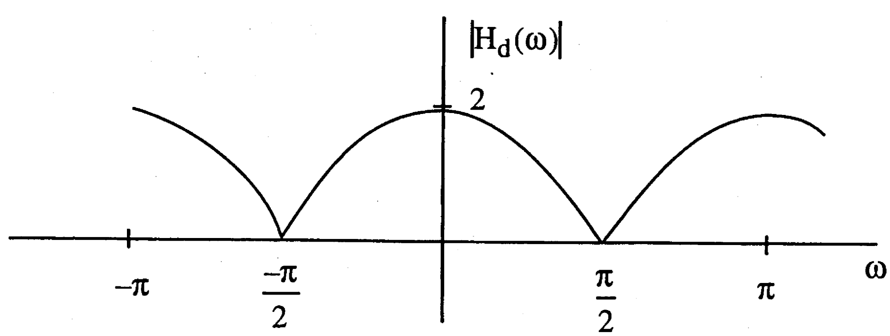
Example



Have:

$$
\begin{aligned}
H_d(\omega) &= 1 + e^{-j2\omega} \\
&= e^{-j\omega}\left(e^{j\omega} + e^{-j\omega}\right) \\
&= e^{-j\omega}\, 2\cos\omega
\end{aligned}
$$

$\Rightarrow |H_d(\omega)| = 2\ |\cos\omega|$



So, this is a crude BSF with stopband centered at $\omega = \dfrac{\pi}{2}$.

Note: In these last few examples, we have looked at frequency responses of simple nonrecursive filters. We can achieve responses that are much closer to ideal (as close as we would like) by considering nonrecursive filters with more coefficients, and through use of recursive filters. The design of such filters will be an important topic later in the course.

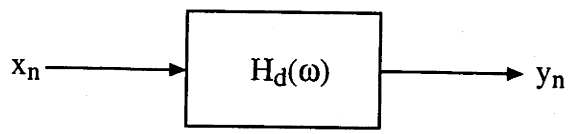
ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 24

D. Munson

Phase of Frequency Response

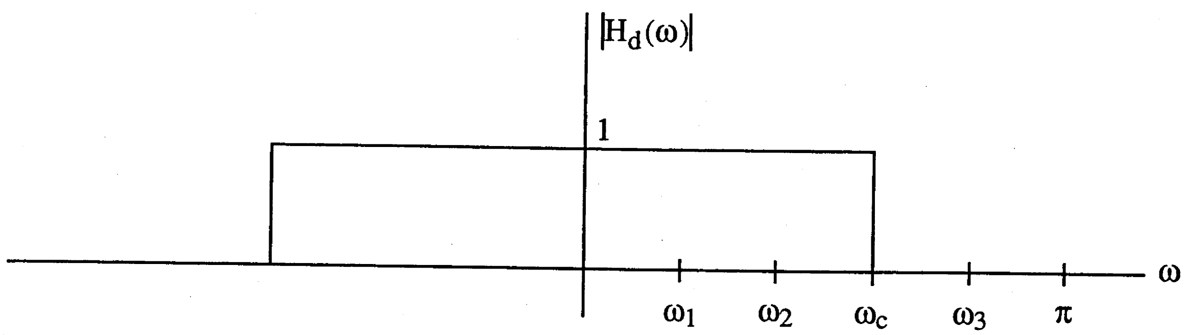
Suppose



with

$$x_n = \cos\omega_1 n + \cos\omega_2 n + \cos\omega_3 n$$

and



Suppose values of ω_1, ω_2 and ω_3 are unknown, but do know $\omega_1, \omega_2 < \omega_c$ and $\omega_3 > \omega_c$. Furthermore, suppose ω_3 is a contaminating sinusoid and you wish to recover just the sinusoids at ω_1 and ω_2.

Thus, want

$$y_n = \cos\omega_1 n + \cos\omega_2 n \qquad (*)$$

How does $\angle H_d(\omega)$ affect y_n?

Know:

$$y_n = \cos\left(\omega_1 n + \angle H_d(\omega_1)\right) + \cos\left(\omega_2 n + \angle H_d(\omega_2)\right)$$

If want (*) then need

$$\angle H_d(\omega) = 0 \text{ for all } \omega \text{ in passband.}$$

$$\Rightarrow H_d(\omega) = |H_d(\omega)|\, e^{j0} = |H_d(\omega)|$$

$$\Rightarrow \quad h_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_d(\omega) e^{j\omega n}\, d\omega$$

$$= \frac{1}{2\pi} \int_{-\omega_c}^{\omega_c} 1\, e^{j\omega n}\, d\omega$$

$$= \frac{\omega_c}{\pi} \operatorname{sinc}(\omega_c n)$$

$\uparrow$

noncausal, and large for $n < 0$.

If we wish to design a causal filter, this type of h_n cannot be well approximated.

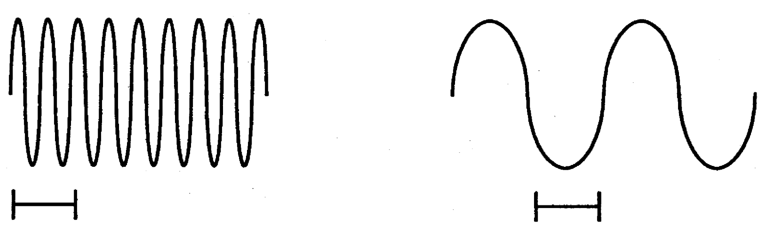
Suppose we are willing to accept a delayed version of the two lower-frequency sinusoids. That is, suppose instead of (*), we are satisfied with

$$y_n = \cos[\omega_1 (n - M)] + \cos[\omega_2 (n - M)]$$

What $\angle H_d(\omega)$ and h_n does this correspond to?

Answer: $\angle H_d(\omega) = -M\omega$ ~ linear phase

Note: It makes sense that we need to shift a higher frequency signal more in phase to get the same delay:



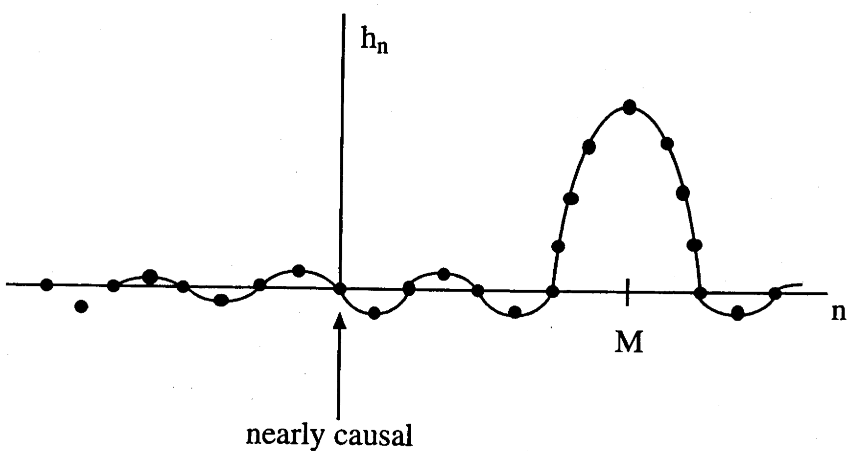
Now, what is h_n for the case with linear phase?

$$h_n = \frac{1}{2\pi}\int_{-\pi}^{\pi} H_d(\omega)\, e^{j\omega n}\, d\omega$$

$$= \frac{1}{2\pi}\int_{-\omega_c}^{\omega_c} 1 \bullet e^{-j\omega M}\, e^{j\omega n}\, d\omega$$

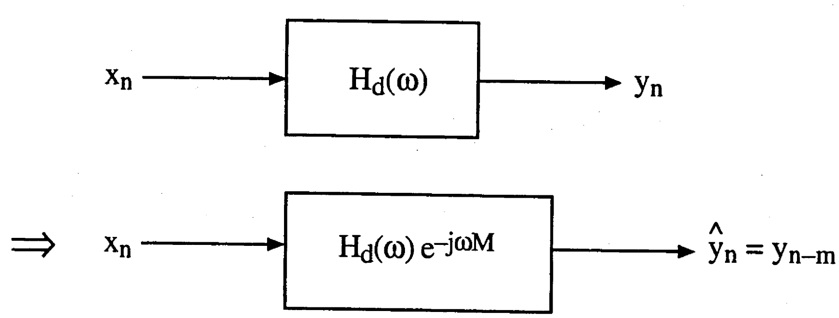
$$= \frac{1}{2\pi}\int_{-\omega_c}^{\omega_c} e^{j\omega(n-M)}\, d\omega$$

$$= \frac{\omega_c}{\pi}\, \mathrm{sinc}\left[\omega_c\,(n - M)\right]$$



By truncating this to the left of the origin, we get a causal h_n and this changes $|H_d(\omega)|$ only slightly.

In general, linear phase just adds a delay (which is often acceptable):



Proof:

$$\hat{y}_n = \frac{1}{2\pi}\int_{-\pi}^{\pi} \underbrace{H_d(\omega)\, e^{-jM\omega}\, X_d(\omega)}_{\hat{Y}_d(\omega)}\, e^{j\omega n}\, d\omega$$

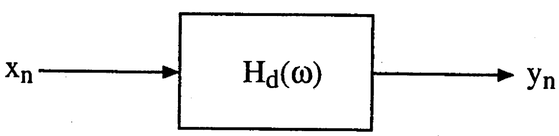
$$= \frac{1}{2\pi}\int_{-\pi}^{\pi} \underbrace{H_d(\omega)\, X_d(\omega)}_{Y_d(\omega)}\, e^{j\omega(n-M)}\, d\omega$$

$$= y_{n-M} \quad ✔$$

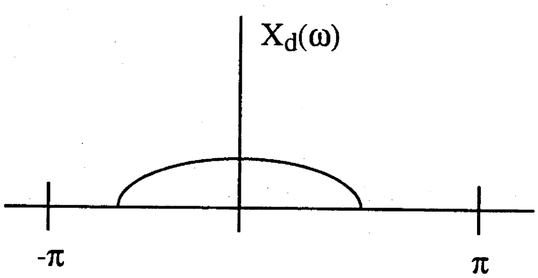
What about nonlinear phase? Answer: Usually don't want it.

Example

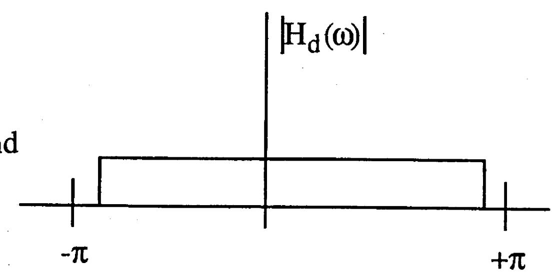
If



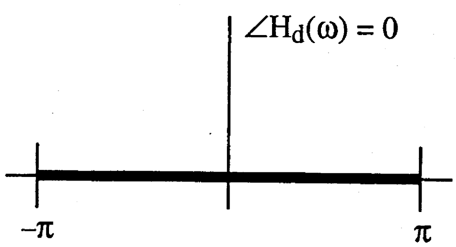
with



and



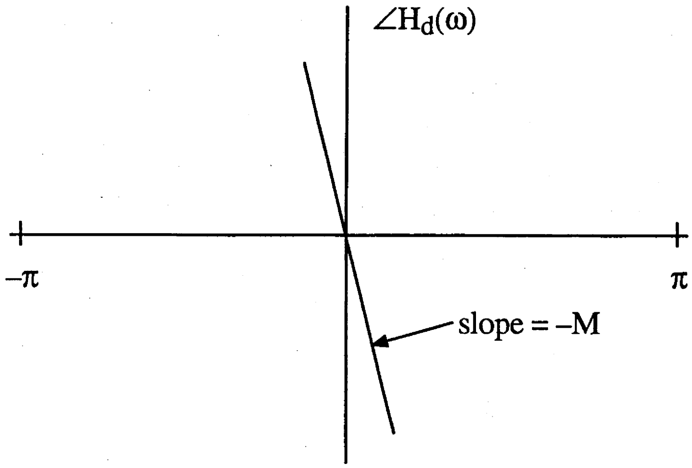
then consider three possibilities for the phase: zero, linear, and nonlinear.



$\Rightarrow H_d(\omega) = |H_d(\omega)|$

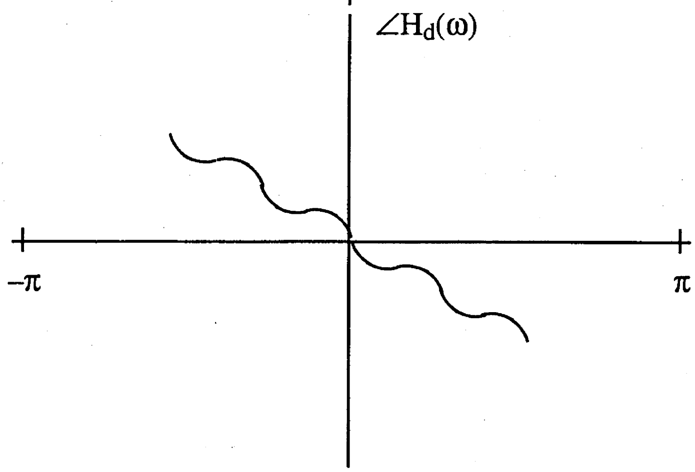
$\Rightarrow Y_d(\omega) = X_d(\omega)$

$\Rightarrow y_n = x_n$



⇒ All frequencies delayed by same amount

⇒ $y_n = x_{n-M}$



⇒ y_n not even close to x_n because different frequencies get delayed by different amounts.

Definition:

Will say $H_d(\omega)$ is <u>linear phase</u> if $H_d(\omega) = \underbrace{|H_d(\omega)|}_{\text{nonnegative}} e^{-j\omega M}$

<u>Comment</u>:

Later in the course, we will consider frequency responses having <u>generalized linear phase</u> where

$$H_d(\omega) = R(\omega)\, e^{-j\omega M}$$

with $R(\omega)$ real-valued, but not necessarily nonnegative.

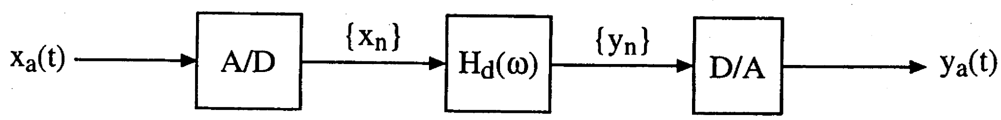
ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 25

D. Munson

Analog Frequency Response of a Digital Processor

Consider

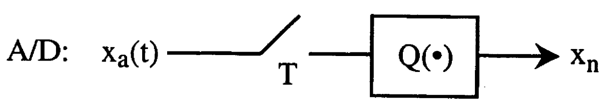


This overall system has an analog input and an analog output. We wish to discover how the analog frequency response depends on $H_d(\omega)$. So, find

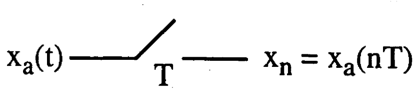
$$H_a(\Omega) = \frac{Y_a(\Omega)}{X_a(\Omega)}.$$

We will see that the formula for $Y_a(\Omega)$ in terms of $X_a(\Omega)$ is very complicated, and that in general we can't find this ratio. However, it is possible to find this ratio if we assume that $x_a(t)$ is bandlimited and that we sample above the Nyquist rate. To find $Y_a(\Omega)$ in terms of $X_a(\Omega)$, consider the components in the overall system one at a time, in the frequency domain. Begin with the A/D.

1) Sampling



In the analysis, we will neglect the quantizer. Consider



We will show:

$$\boxed{X_d(\omega) = \frac{1}{T} \sum_{n=-\infty}^{\infty} X_a\left(\frac{\omega + 2\pi n}{T}\right)} \qquad (\Diamond)$$

extremely important

Proof:

We will use the fact that

$$\frac{1}{\tau}\sum_{n=-\infty}^{\infty} e^{jn\frac{2\pi}{\tau}t} = \sum_{n=-\infty}^{\infty} \delta_a(t-n\tau) \qquad (*)$$

This equality holds in a <u>distributional sense</u>, i.e.,

$$\int_{-\infty}^{\infty} f_a(t)\frac{1}{\tau}\sum_{n=-\infty}^{\infty} e^{jn\frac{2\pi}{\tau}t}\,dt = \int_{-\infty}^{\infty} f_a(t)\sum_{n=-\infty}^{\infty} \delta_a(t-n\tau)\,dt$$

for any continuous function $f_a(t)$.

Now, prove (◊):

$$X_d(\omega) = \sum_{n=-\infty}^{\infty} x_n e^{-j\omega n}$$

$$= \sum_{n=-\infty}^{\infty} x_a(nT)e^{-j\omega n}$$

$$= \sum_{n=-\infty}^{\infty} \frac{1}{2\pi}\int_{-\infty}^{\infty} X_a(\Omega)\, e^{j\Omega nT}\, d\Omega\; e^{-j\omega n}$$

$$= \frac{1}{2\pi}\int_{-\infty}^{\infty} X_a(\Omega)\sum_{n=-\infty}^{\infty} e^{jn(\Omega T-\omega)}\, d\Omega$$

$$= \frac{1}{T}\int_{-\infty}^{\infty} X_a(\Omega)\sum_{n=-\infty}^{\infty} \delta_a\left(\Omega-\frac{\omega+2\pi n}{T}\right) d\Omega$$

↑

let $t=\Omega-\frac{\omega}{T},\ \tau=\frac{2\pi}{T}$ in (*)

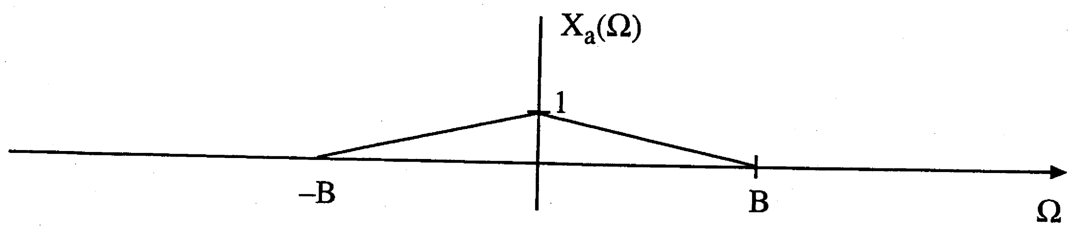
$$= \frac{1}{T}\sum_{n=-\infty}^{\infty} X_a\left(\frac{\omega+2\pi n}{T}\right) \qquad ✔$$

Memorize this result! We will use it repeatedly throughout the course.

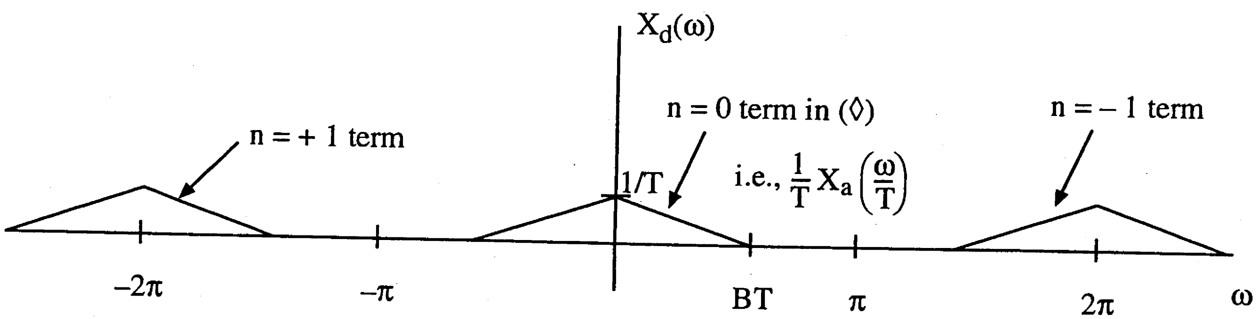
Now, what does (◊) mean?

Example

Suppose $x_a(t)$ is bandlimited to B rad/sec



If $T < \frac{\pi}{B}$ then terms in (◊) don't overlap ~ "no aliasing" and get:



Given $X_a(\Omega)$, you <u>must</u> be able to draw this picture!

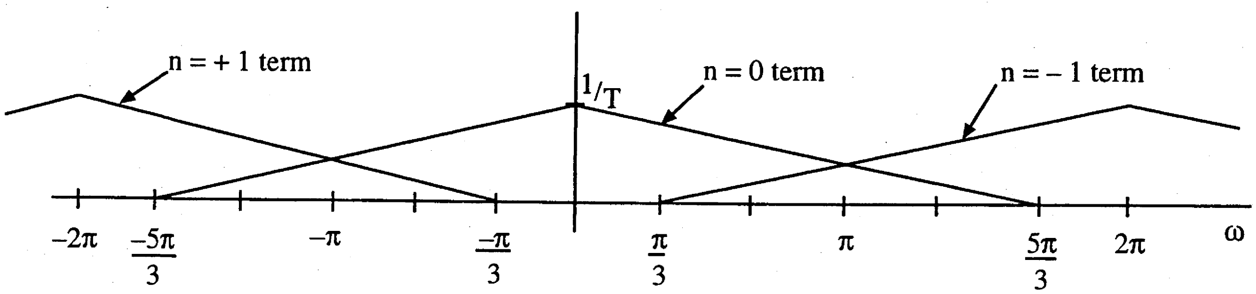
How do we know the n = 0 term is confined to $|\omega| \le BT$? Answer: The n = 0 term is $\frac{1}{T}X_a\left(\frac{\omega}{T}\right)$, which first hits zero when its argument equals ± B, i.e., when $\frac{\omega}{T} = \pm B \Rightarrow \omega = \pm BT$.

We assumed $T < \frac{\pi}{B} \Rightarrow BT < \pi$, so that the n = 0 term is confined to $|\omega| < \pi$ as shown above. If $BT > \pi$ then the various terms overlap, which is called <u>aliasing</u>. The condition $BT < \pi$ (no aliasing) is equivalent to the Nyquist criterion:

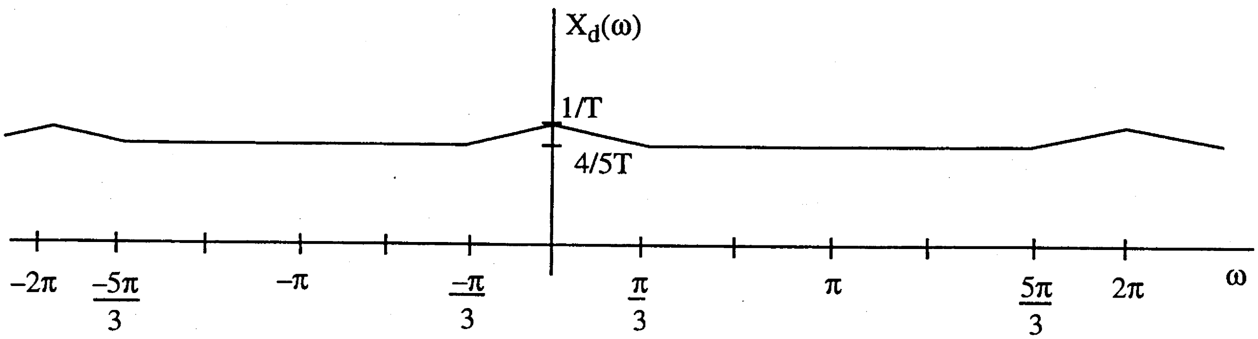
$$
\begin{aligned}
BT < \pi \quad &\Rightarrow \quad T < \frac{\pi}{B} \\
&\Rightarrow \quad \frac{1}{T} > \frac{B}{\pi} \\
&\Rightarrow \quad \frac{1}{T} > 2\left(\frac{B}{2\pi}\right)
\end{aligned}
$$

where 1/T is the sampling frequency in samples per second, and $B/2\pi$ is the bandwidth of $x_a(t)$ in Hz.

Suppose in the above example we choose $T = \frac{5}{3}\left(\frac{\pi}{B}\right)$, so that we have aliasing. In this case $BT = \frac{5}{3}\pi$ and $X_d(\omega)$ will be the sum of



Thus, $X_d(\omega)$ is



Obviously, $X_d(\omega)$ is no longer a periodic repetition of X_a. This is because some of the high frequencies in $X_a(\Omega)$ have been moved down to low frequencies in $X_d(\omega)$, i.e., high frequencies are masquerading as low frequencies. Hence, the term aliasing.

If we choose T even larger than above, then additional terms ($n = \pm 2$, etc.) in equation (◊) will overlap on the central interval $|\omega| \leq \pi$. In determining the shape of $X_d(\omega)$, we can concern ourselves only with $|\omega| \leq \pi$, since we know that $X_d(\omega)$ is periodic outside this central period.

Note: Can use equation (◊) to prove sampling theorem. If $x_a(t)$ is bandlimited and you sample above the Nyquist rate, then from the second plot on p. 25.3:

$$\begin{array}{cc} X_d(\omega) = & \frac{1}{T} X_a\left(\frac{\omega}{T}\right) \quad \text{for } |\omega| \leq \pi \\ \updownarrow & \updownarrow \\ \{x_n\} & x_a(t) \end{array}$$

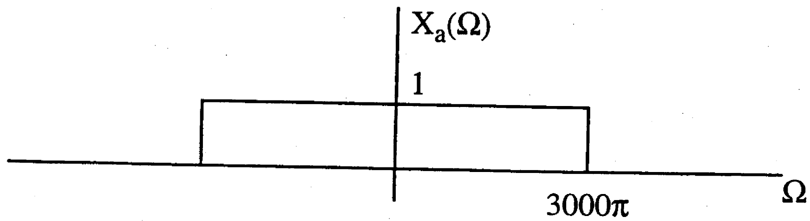
So, basically

$$x_a(t) = \mathcal{F}^{-1}\left[\mathrm{DTFT}\{x_n\}\right]$$

Working through the algebra gives the usual sinc reconstruction formula for $x_a(t)$ in terms of the x_n. You may see this as a homework problem!

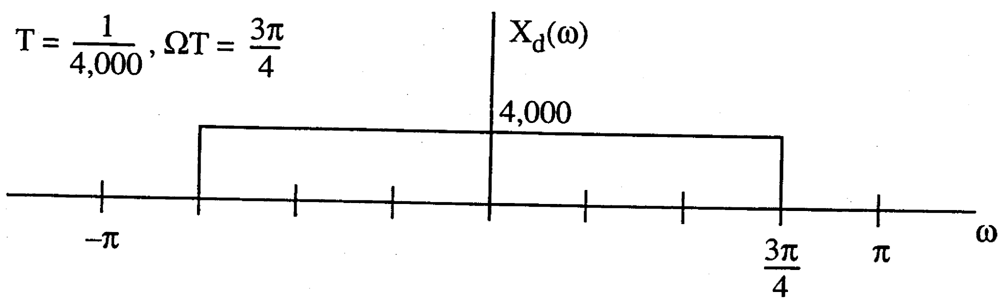
Example

Illustrate that a higher sampling frequency shrinks the DTFT. Suppose

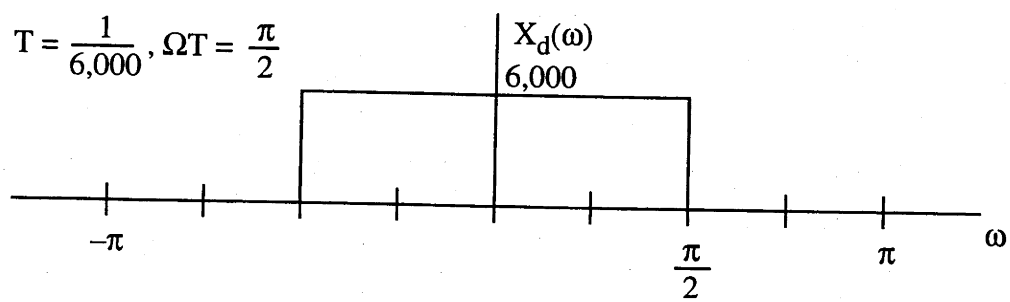


If $x_n = x_a(nT)$, sketch $X_d(\omega)$ for $T = \frac{1}{4{,}000}$, $T = \frac{1}{6{,}000}$, and $T = \frac{1}{12{,}000}$.

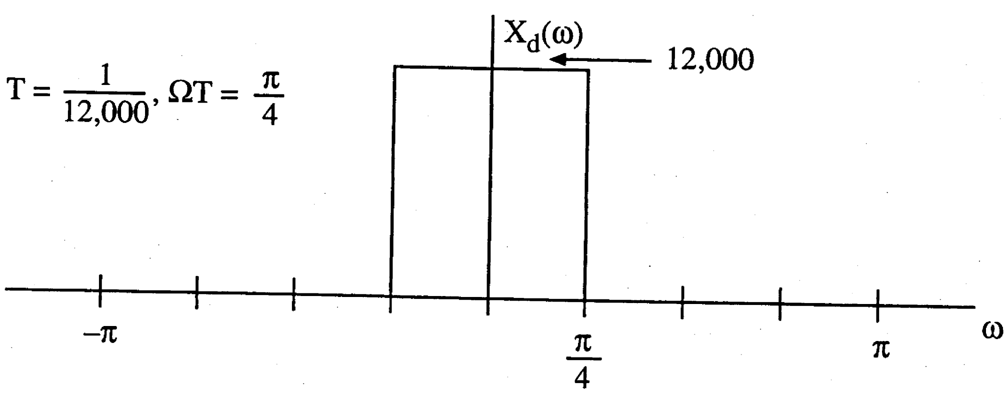
$T = \frac{1}{4{,}000}$, $\Omega T = \frac{3\pi}{4}$



$T = \frac{1}{6{,}000}$, $\Omega T = \frac{\pi}{2}$



$T = \frac{1}{12{,}000}$, $\Omega T = \frac{\pi}{4}$



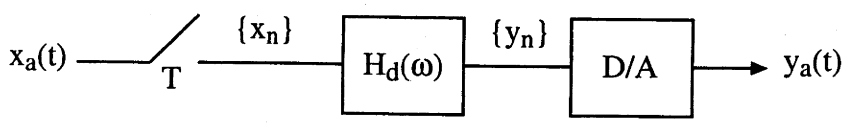
So, as T decreases (sampling frequency increases), the DTFT shrinks and also grows in amplitude.

ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 26

D. Munson

Now, get back to study of the system:



2) Digital Filter

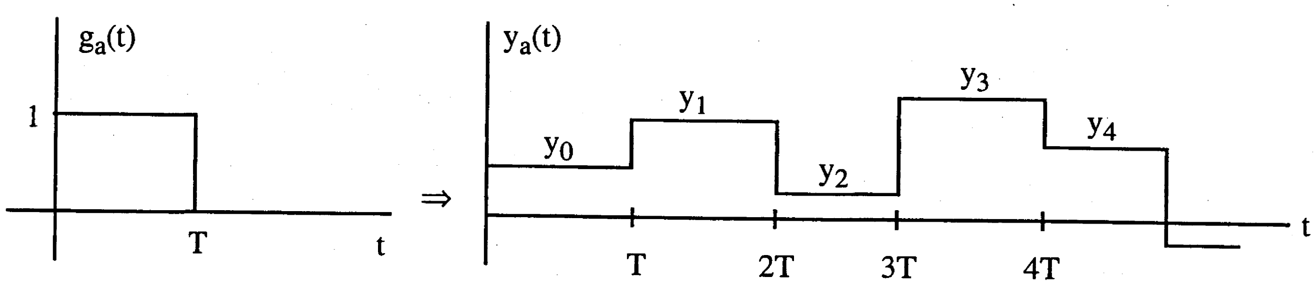
$$Y_d(\omega) = H_d(\omega)\, X_d(\omega)$$

$$\underset{\substack{\uparrow \\ \text{by } (\Diamond)}}{=} \frac{1}{T} H_d(\omega) \sum_{n=-\infty}^{\infty} X_a\left(\frac{\omega + 2\pi n}{T}\right) \qquad (\Diamond\Diamond)$$

3) D/A: Model as

$$y_a(t) = \sum_{n=-\infty}^{\infty} y_n\, g_a(t - nT) \qquad (\square)$$

so that $y_a(t)$ is a weighted pulse train. Later, we will consider the case where $g_a(t)$ is a rectangular pulse so that $y_a(t)$ is a staircase function:



This type of D/A is very common and is called a zero-order hold (ZOH). An ideal D/A, however, would use $g_a(t) = \text{sinc}\left(\frac{\pi}{T} t\right)$ so that ($\square$) would implement the ideal sinc reconstruction formula. As we shall see later, it is not possible to realize a D/A that exactly implements the sinc reconstruction formula. However, the ideal D/A can be approximated as closely as desired (except for a delay) by using a ZOH, followed by an appropriate analog low-pass filter. We will study this later.

For now, assume that the D/A is modeled by (□) and that $g_a(t)$ can be arbitrary. This D/A model will cover all situations that we will encounter.

Now, use (□) to find $Y_a(\Omega)$ in terms of $Y_d(\omega)$:

$$Y_a(\Omega) = \int_{-\infty}^{\infty} y_a(t)\, e^{-j\Omega t}\, dt$$

$$= \int_{-\infty}^{\infty} \sum_{n=-\infty}^{\infty} y_n\, g_a(t - nT)\, e^{-j\Omega t}\, dt$$

$$= \sum_{n=-\infty}^{\infty} y_n \int_{-\infty}^{\infty} g_a(t - nT)\, e^{-j\Omega t}\, dt$$

$$= \sum_{n=-\infty}^{\infty} y_n\, G_a(\Omega)\, e^{-j\Omega nT}$$

$$= G_a(\Omega) \sum_{n=-\infty}^{\infty} y_n\, e^{-j\Omega Tn}$$

$$\Rightarrow \boxed{Y_a(\Omega) = G_a(\Omega)\, Y_d(\Omega T)}$$

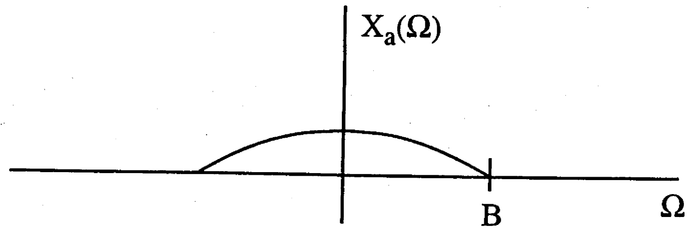
This is an important equation. You should either memorize it or remember how to quickly derive it.

Substituting for Y_d from (◊◊) gives us the expression for $Y_a(\Omega)$ in terms of X_a:

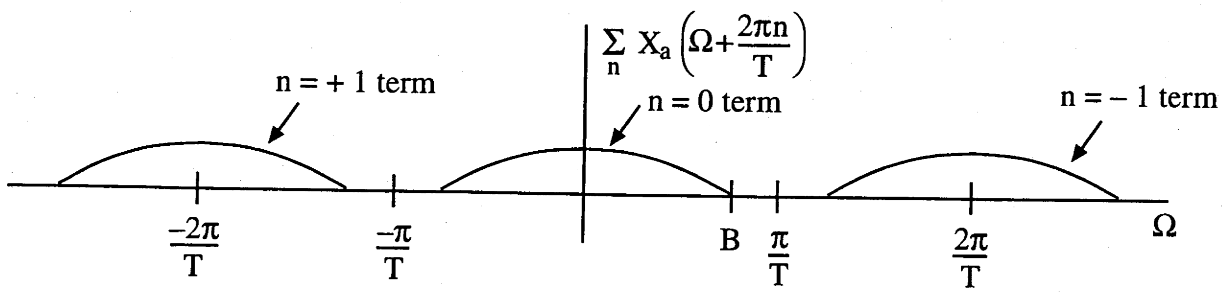
$$\boxed{Y_a(\Omega) = \frac{1}{T} G_a(\Omega) H_d(\Omega T) \sum_{n=-\infty}^{\infty} X_a\left(\Omega + \frac{2\pi n}{T}\right)} \qquad (\text{h})$$

Now, take a rest!

This equation is cumbersome and in general we cannot solve for $\frac{Y_a(\Omega)}{X_a(\Omega)}$. Indeed, we cannot define $H_a(\Omega)$ because, although the overall system is linear, in general it is <u>shift-varying</u> so that the system is not describable by a frequency response. Fortunately, (h) simplifies tremendously if we assume a bandlimited input with Nyquist-rate sampling, and an ideal D/A converter. Under these conditions the overall system is shift-invariant and it can be described by a frequency response. Let's consider this. Suppose $x_a(t)$ is bandlimited to B rad/sec and we choose $T < \frac{\pi}{B}$. Then, supposing



gives



$\Rightarrow$ No aliasing, so that:

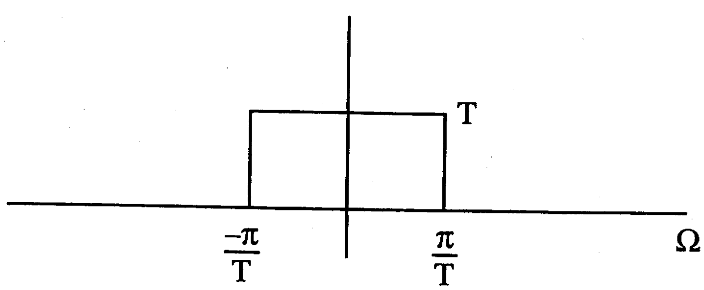
$$\sum_{n=-\infty}^{\infty} X_a\left(\Omega+\frac{2\pi n}{T}\right) = X_a(\Omega), \quad |\Omega| \le \frac{\pi}{T}$$

Now, assume an <u>ideal D/A</u> so that

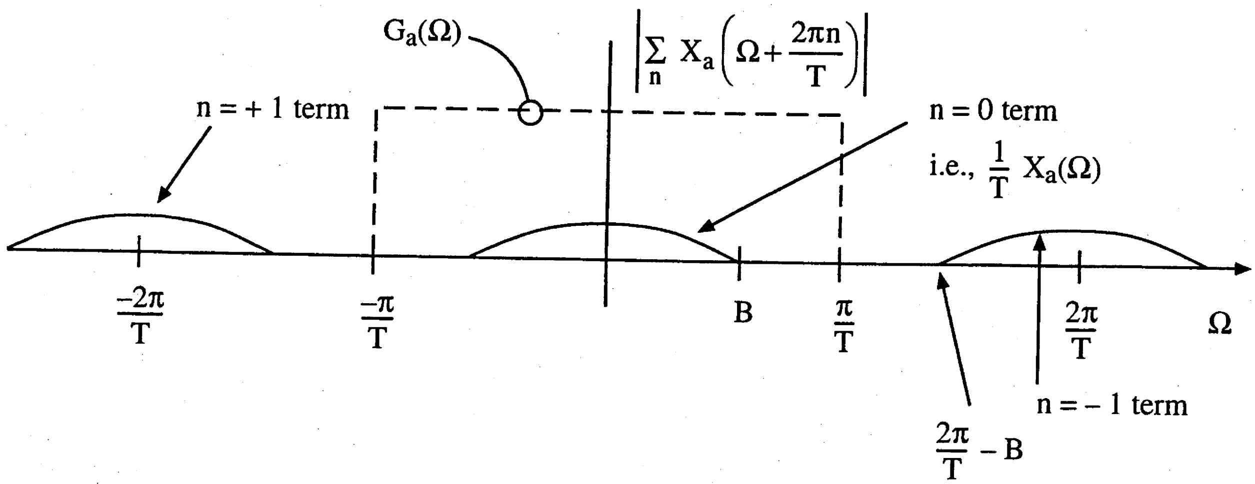
$$g_a(t) = \operatorname{sinc}\frac{\pi t}{T} = \frac{\sin\frac{\pi t}{T}}{\frac{\pi t}{T}}$$

Considering the Fourier transform of this pulse, we see that for an ideal D/A, $G_a(\Omega)$ has the shape of an ideal LPF:

$$G_a(\Omega) = \begin{cases} T & |\Omega| \le \frac{\pi}{T} \\ 0 & |\Omega| > \frac{\pi}{T} \end{cases}$$



Now, let's picture the terms that multiply H_d in (h):



So:

$$G_a(\Omega)\sum_{n=-\infty}^{\infty} X_a\left(\Omega+\frac{2\pi n}{T}\right) = \begin{cases} T \bullet X_a(\Omega) & |\Omega| \le \frac{\pi}{T} \\ 0 & |\Omega| > \frac{\pi}{T} \end{cases}$$

Using this in (⊢) gives:

$$Y_a(\Omega) = \begin{cases} H_d(\Omega T)X_a(\Omega) & |\Omega| \le \frac{\pi}{T} \\ 0 & |\Omega| > \frac{\pi}{T} \end{cases}$$

The analog frequency response of the A/D, digital filter, and D/A is

$$H_a(\Omega) = \frac{Y_a(\Omega)}{X_a(\Omega)}$$

$$\Rightarrow \quad \boxed{H_a(\Omega) = \begin{cases} H_d(\Omega T) & |\Omega| \le \frac{\pi}{T} \\ 0 & |\Omega| > \frac{\pi}{T} \end{cases} \qquad (\star)}$$

This is the <u>entire connection</u> between analog and digital filtering! This equation is extremely important!

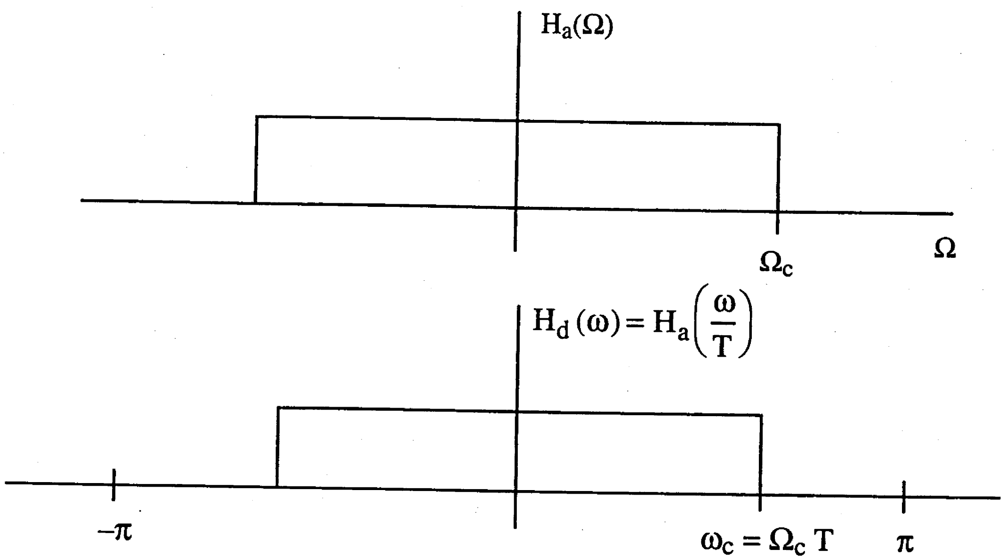
With a change of variable, $\omega = \Omega T$, ($\star$) becomes:

$$H_a\left(\frac{\omega}{T}\right) = \begin{cases} H_d(\omega) & |\omega| \le \pi \\ 0 & |\omega| > \pi \end{cases}$$

$$\Rightarrow \quad \boxed{H_d(\omega) = H_a\left(\frac{\omega}{T}\right) \qquad |\omega| \leq \pi \qquad (☆☆)}$$

So, given a desired $H_a(\Omega)$, $H_d(\omega)$ has the same shape, but on just the center interval $|\omega| \leq \pi$.

If the desired analog cutoff frequency is $\Omega = \Omega_c$ how do we choose the digital cutoff?



Remember:

$$\boxed{\omega_c = \Omega_c T}$$

This equation is very handy in specifying cutoff frequencies of digital filters. Likewise, it can be used to find the analog cutoff of a digital system operating with parameters T and ω_c, i.e., $\Omega_c = \omega_c/T$.

ECE 310 **DIGITAL SIGNAL PROCESSING** **D. Munson**
University of Illinois **Lecture 27**

Example $x_a(t)$ BL to 50 kHz

Implement analog LPF with

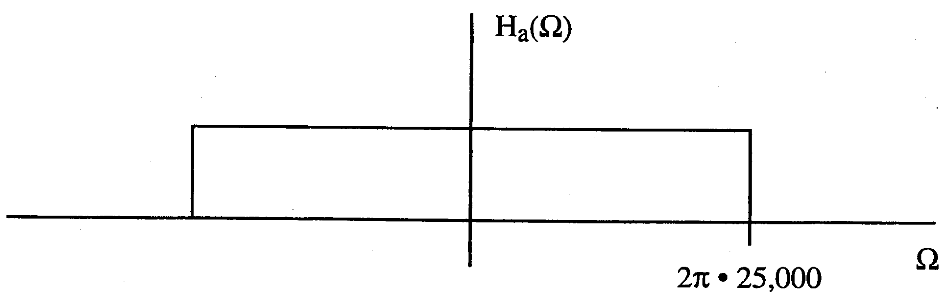
$$\Omega_c = 2\pi\,(25{,}000) \text{ rad/sec.}$$

Choose T according to Nyquist: $\frac{1}{T} = 100{,}000$ samples/sec.

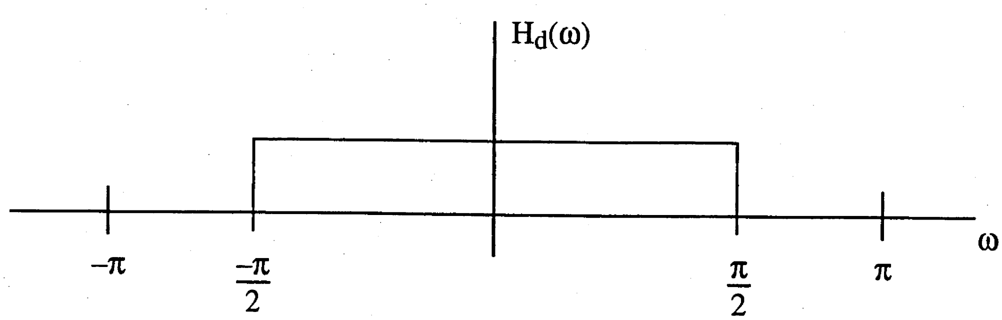
$$\Rightarrow T = 10^{-5}$$

$$\omega_c = \Omega_c T = 2\pi\,(25{,}000)\,10^{-5} = \frac{\pi}{2}$$

So,



is realized by $T = 10^{-5}$ and using



Note: In this example the ratio of the desired analog cutoff to the analog bandwidth was $\frac{1}{2}$. Likewise, the passband of $H_d(\omega)$ filled half of the digital frequency band $|\omega| \le \pi$. This proportional relationship will always hold if we sample <u>at</u> the Nyquist rate.

Question: Why did we sample <u>at</u> the Nyquist rate instead of above it? Why not sample at a rate much greater than 100,000 samples per second? Answer: This would increase hardware cost since it would require a faster A/D and digital filter.

Example

$x_a(t)$ is BL to $2\pi \times 10^6$ rad/sec. Implement analog HPF with $f_c = 250{,}000$ Hz.

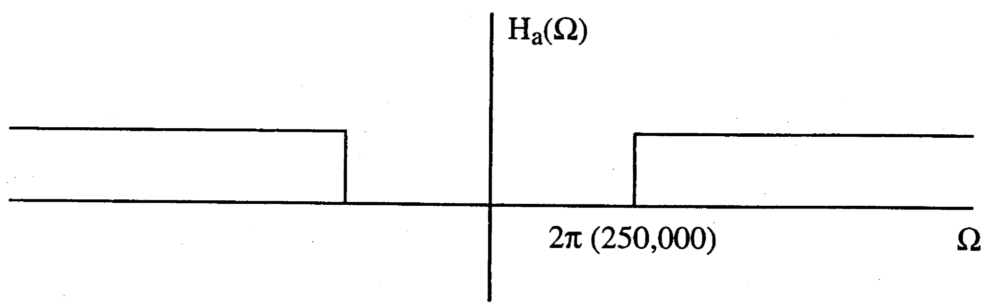
Choose $\frac{1}{T} = 2\left[\frac{2\pi \times 10^6}{2\pi}\right] = 2 \times 10^6$

$$\Rightarrow T = \frac{1}{2 \times 10^6}$$

$$\omega_c = \Omega_c T = 2\pi\,(250{,}000)\,\frac{1}{2 \times 10^6}$$

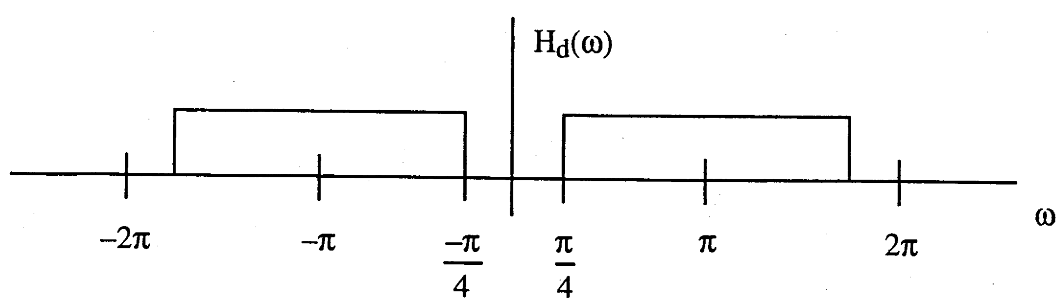
$$= \frac{\pi}{4}$$

So,



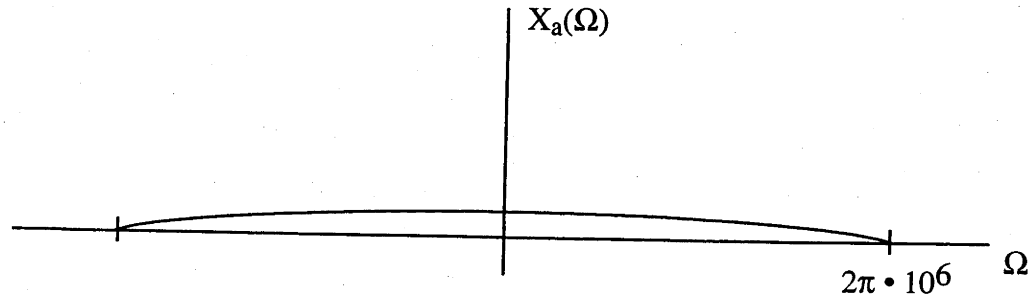
is realized by choosing $T = \frac{1}{2 \times 10^6}$

and using:

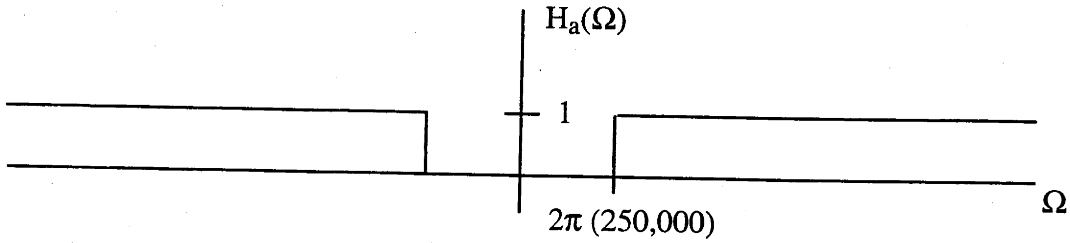


Note: In this situation the overall digital system implements a bandpass filter, but this is equivalent to a HPF because $x_a(t)$ is bandlimited and sampled according to Nyquist.

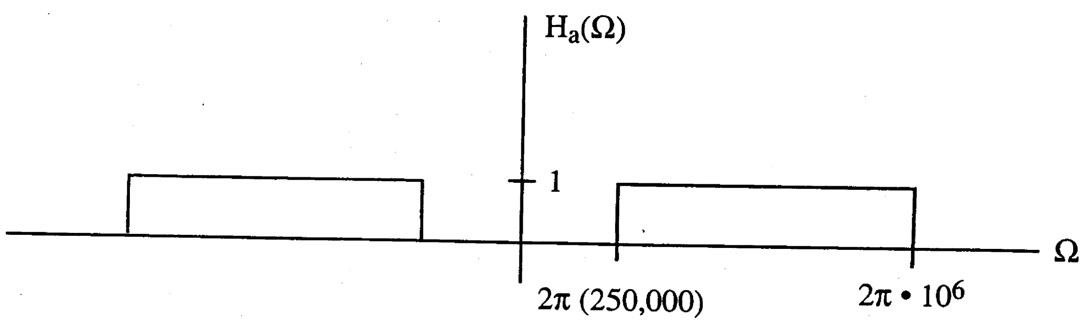
Filtering



with



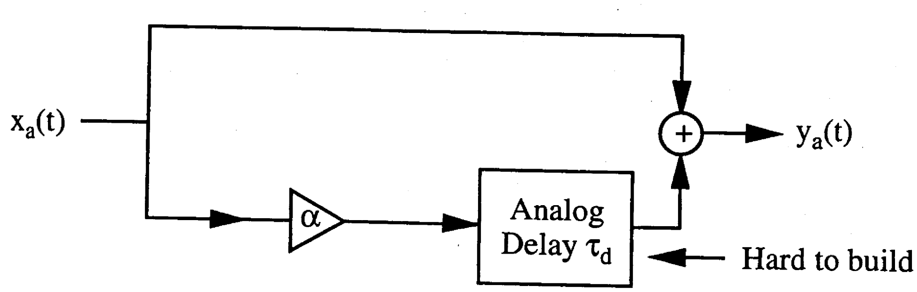
or



gives exactly the same result. The digital system implements the latter (D/A cuts off all frequencies above $2\pi \bullet 10^6$).

Example

Design a digital version of an echo generator:



So, we want a digital system that implements

$$y_a(t) = x_a(t) + \alpha\, x_a\,(t-\tau_d)$$

Assume $x_a(t)$ is bandlimited to 20 kHz.

a) Choose sampling period T.

b) Find desired analog response $H_a(\Omega)$.

c) Find needed digital filter response $H_d(\omega)$.

d) Assuming $\tau_d = kT$, draw a block diagram of the digital filter.

<u>Solution</u>

a) $T < \dfrac{\pi}{\Omega} = \dfrac{\pi}{2\pi \bullet 20\text{ kHz}} = \dfrac{1}{40{,}000}$

Choose

$$\boxed{T = \frac{1}{40{,}000}}$$

b) $Y_a(\Omega) = X_a(\Omega) + \alpha\, X_a(\Omega)\, e^{-j\Omega\tau_d}$

$$\Rightarrow \quad \boxed{H_a(\Omega) = 1 + \alpha\, e^{-j\Omega\tau_d}}$$

c) To find $H_d(\omega)$ in this example, we cannot simply apply $\omega_c = \Omega_c T$, because $H_a(\Omega)$ is not a LPF, HPF, or BFF. There is no Ω_c! Instead, must go back to (☆☆). From (☆☆):

$$H_d(\omega) = H_a\left(\frac{\omega}{T}\right) \qquad |\omega| \le \pi$$

$$= \boxed{1 + \alpha\, e^{-j\omega\frac{\tau_d}{T}}}$$

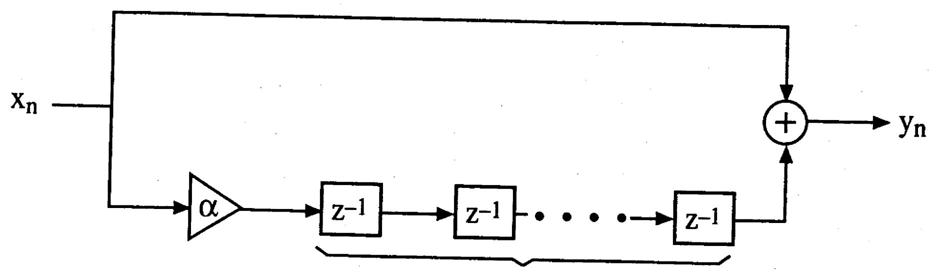
d) If $\tau_d = kT$ then:

$$H_d(\omega) = 1 + \alpha\, e^{-j\omega\frac{kT}{T}}$$

$$= 1 + \alpha\, e^{-jk\omega}$$

$$\Rightarrow H(z) = 1 + \alpha\, z^{-k}$$

So, the digital filter structure is:



This result completely agrees with our intuition. We could have guessed this!

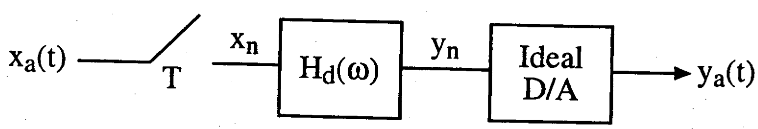
Notes:

1) Using this digital filter between an A/D and D/A implements the desired $H_a(\Omega)$.

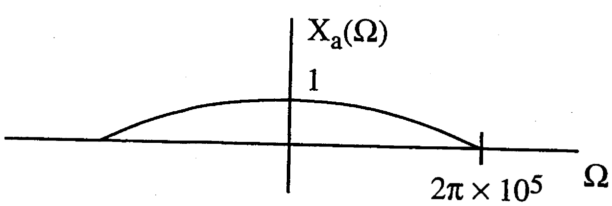
2) If $\tau_d \neq kT$ then H(z) is not a rational function in the variable z^{-1}, and we can only approximate the desired $H_d(\omega)$. Filter design (approximation) will be a major topic later in the course.

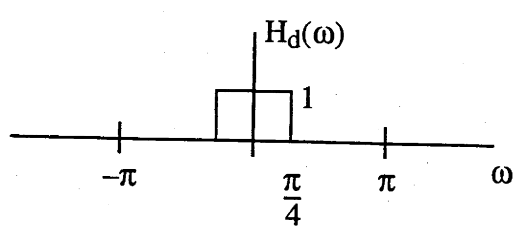
Example

Consider the following system



with $T = \dfrac{1}{3\times10^5}$ and

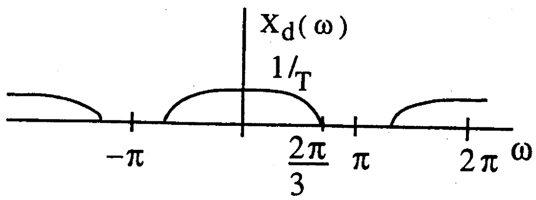


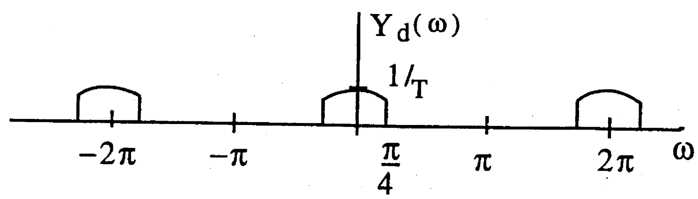


Sketch $X_d(\omega)$, $Y_d(\omega)$, and $Y_a(\Omega)$.

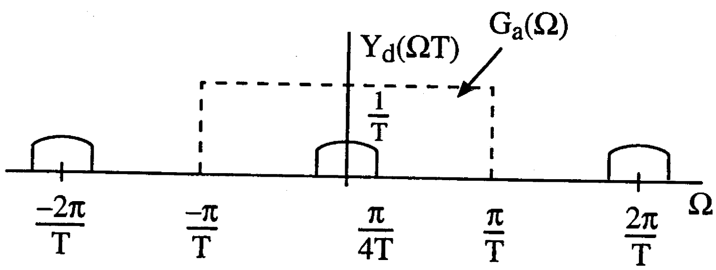
27.6

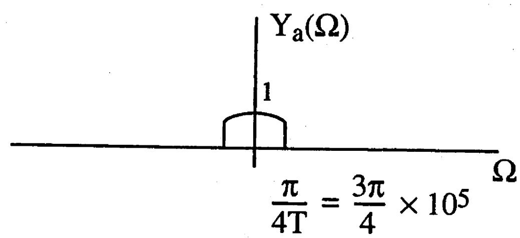
Solution





For an ideal D/A, $Y_a(\Omega) = G_a(\Omega)\, Y_d(\Omega T)$ with $G_a(\Omega) = \begin{cases} T & |\Omega| \leq \frac{\pi}{T} \\ 0 & \text{else} \end{cases}$. Thus,





Notice that if the D/A is nonideal, and $G_a(\Omega)$ does not cut off abruptly at $\pm \frac{\pi}{T}$, but instead is nonzero for $|\Omega| > \frac{\pi}{T}$, then $Y_a(\Omega)$ will have undesired high-frequency components due to the periodic nature of Y_d. Thus, a critical job of the D/A is to suppress these high-frequency replicas.

ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 28

D. Munson

Implementation of Ideal D/A

Consider

$$y_n \longrightarrow \boxed{\text{D/A}} \longrightarrow y_a(t)$$

Recall that any D/A we encounter in this course can be modeled by

$$y_a(t) = \sum_{n=-\infty}^{\infty} y_n \, g_a(t - nT) \tag{1}$$

and that the Fourier-domain relation is

$$Y_a(\Omega) = G_a(\Omega) \, Y_d(\Omega T) \tag{2}$$

For the ideal D/A, we have $g_a(t) = \text{sinc}\left(\frac{\pi}{T} t\right)$, giving

$$y_a(t) = \sum_{n=-\infty}^{\infty} y_n \, \text{sinc}\left[\frac{\pi}{T}(t - nT)\right] \tag{3}$$

and

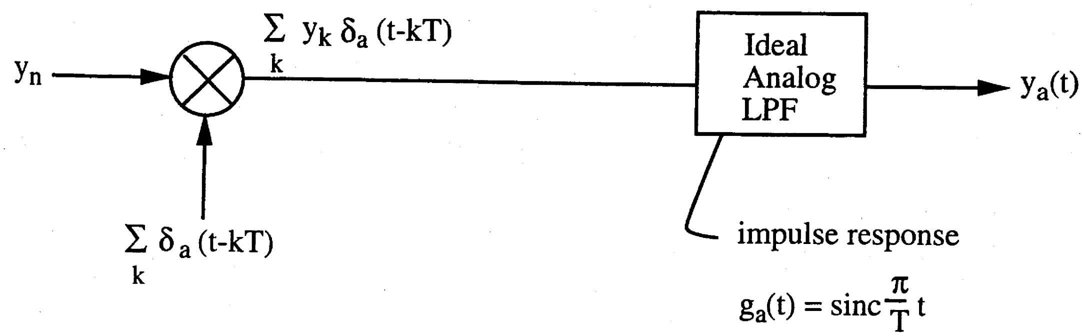
$$G_a(\Omega) = \begin{cases} T & |\Omega| \le \frac{\pi}{T} \\ 0 & \text{else} \end{cases}$$

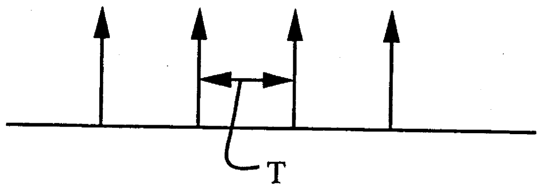
so that (2) gives

$$Y_a(\Omega) = \begin{cases} T\,Y_d(\Omega T) & |\Omega| \le \frac{\pi}{T} \\ 0 & \text{else} \end{cases} \tag{4}$$

How might we implement the ideal D/A, described by (3)?

Conceptually, we might think along the lines of:



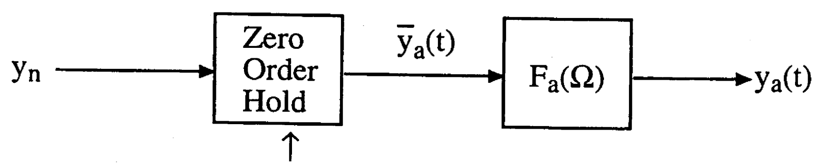


Then:

$$y_a(t) = g_a(t) * \sum_n y_n \, \delta_a(t-nT) = \sum_n y_n \, g_a(t-nT) \text{ as desired.}$$

For an actual implementation, we might consider approximating the impulse train by a periodic sequence of very tall, narrow pulses. **However, this would be difficult in practice. As a result, D/A's are not implemented as suggested above!**

In practice, the ideal D/A is approximated with the following two-stage system:

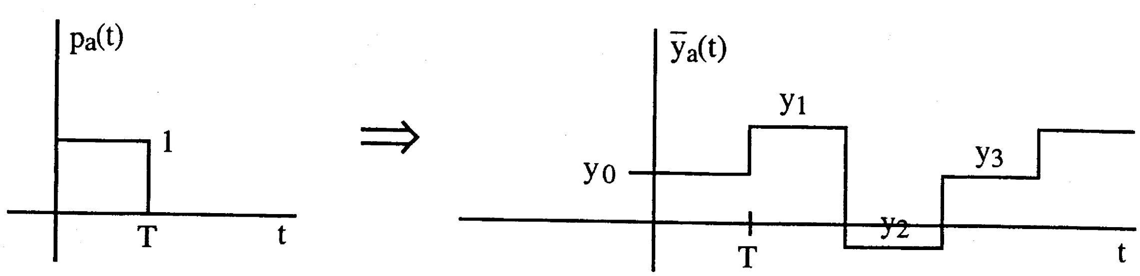


in manufacturer's catalog just the ZOH may be called a D/A

What is a zero-order hold (ZOH)? It is a D/A that uses rectangular pulses, i.e.,

$$\bar{y}_a(t) = \sum_n y_n \, p_a(t-nT) \tag{5}$$

where



Thus, the ZOH output is a staircase approximation to the desired $y_a(t)$. This staircase must be smoothed by $F_a(\Omega)$ to produce the proper $y_a(t)$.

The Fourier-domain relation for the ZOH has the form given by (2), but now

$$
\begin{aligned}
G_a(\Omega) &= \int_0^T 1 \bullet e^{-j\Omega t}\, dt \\
&= \left. \frac{e^{-j\Omega t}}{-j\Omega} \right|_0^T = \frac{e^{-j\Omega T} - 1}{-j\Omega} \\
&= \frac{e^{-j\Omega\frac{T}{2}}\left(e^{-j\Omega\frac{T}{2}} - e^{j\Omega\frac{T}{2}}\right)}{-j\Omega} = e^{-j\Omega\frac{T}{2}}\, \frac{2\sin\frac{\Omega T}{2}}{\Omega} \\
&= T\, e^{-j\Omega\frac{T}{2}} \operatorname{sinc} \frac{\Omega T}{2}
\end{aligned}
$$

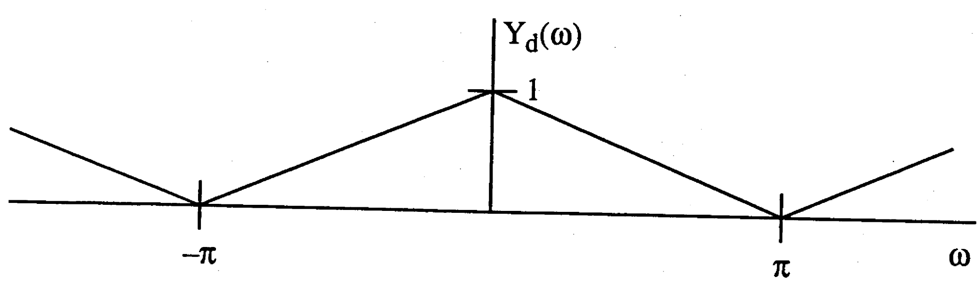
Thus,

$$
\overline{Y}_a(\Omega) = T\, e^{-j\Omega\frac{T}{2}} \operatorname{sinc}\left(\frac{\Omega T}{2}\right) Y_d(\Omega T) \tag{6}
$$

Before deciding how to choose $F_a(\Omega)$, which follows the ZOH, let's see how the effect of the ZOH differs from the ideal D/A, in the Fourier domain.

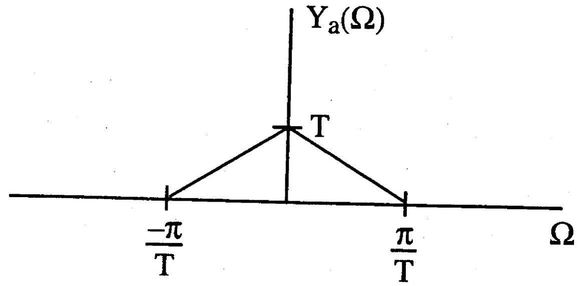
Example

Suppose have

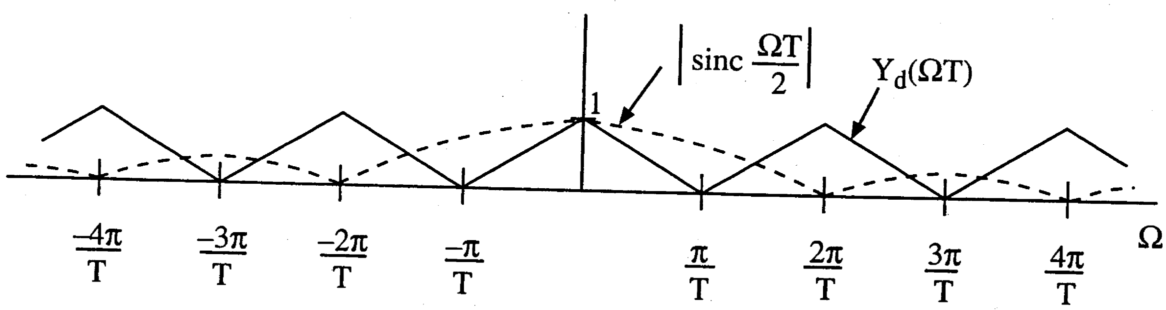


Sketch $Y_a(\Omega)$, the Fourier transform of the output of an ideal D/A, and $\overline{Y}_a(\Omega)$, the Fourier transform of the output of a ZOH.

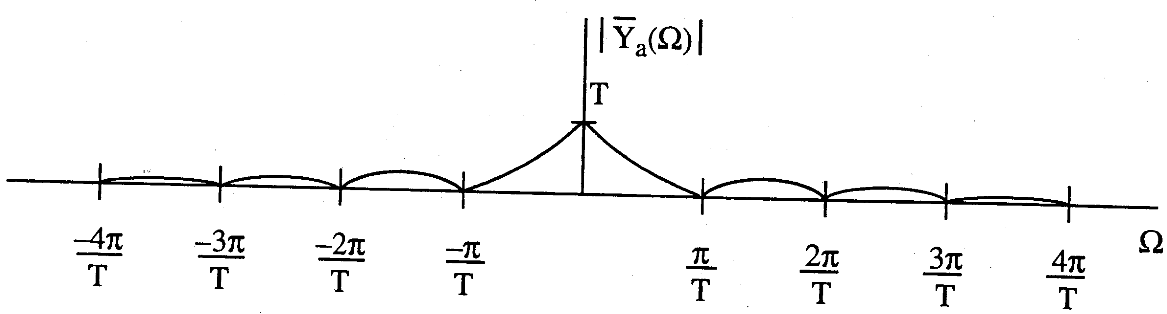
Using (4), we have for the ideal D/A:



For the ZOH, let's plot $|\overline{Y}_a(\Omega)|$. The terms in (6) look like:



$|\overline{Y}_a(\Omega)|$ is T times the product of the above two curves:



Notice that unlike $Y_a(\Omega)$ for the ideal D/A, $|\overline{Y}_a(\Omega)|$ for the ZOH has frequency content that extends all the way to $\Omega = \pm\,\infty$. This is not surprising, since $\overline{y}_a(t)$, for the ZOH, is a staircase function with discontinuities. Sharp edges (discontinuities) always correspond to a frequency content extending to $\pm\,\infty$.

Now, if we have $\overline{Y}_a(\Omega)$ from the ZOH, how do we choose $F_a(\Omega)$ to produce $Y_a(\Omega)$? The above sketches suggest that we need $F_a(\Omega)$ to be a LPF with cutoff at $\Omega_c = \pm\,\frac{\pi}{T}$. To investigate this thoroughly, note that for the ZOH system we have

$$Y_a(\Omega) = F_a(\Omega)\,\overline{Y}_a(\Omega)$$

$$= F_a(\Omega)\,T\,e^{-j\frac{\Omega T}{2}}\,\text{sinc}\left(\frac{\Omega T}{2}\right) Y_d(\Omega T) \tag{7}$$

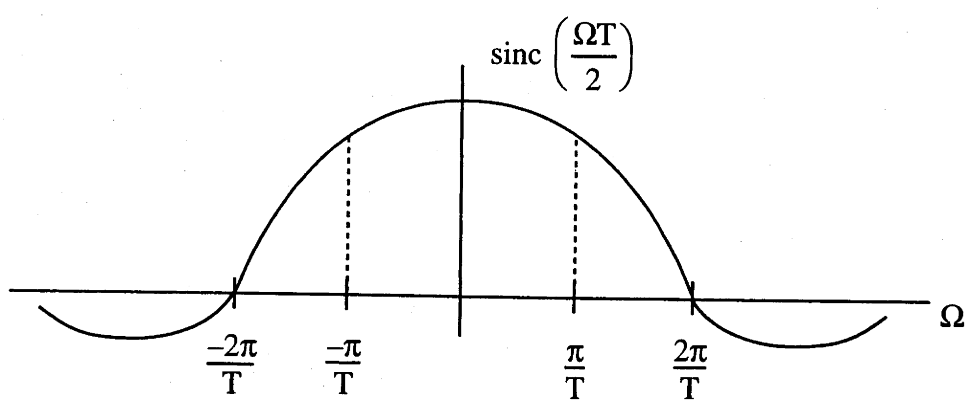
For the ideal D/A, the relation is given by (4). To have (7) correspond to (4) we must have

$$F_a(\Omega)\,T\,e^{-j\frac{\Omega T}{2}}\,\text{sinc}\left(\frac{\Omega T}{2}\right) Y_d(\Omega T) = \begin{cases} T\,Y_d(\Omega T) & |\Omega| \le \frac{\pi}{T} \\ 0 & \text{else} \end{cases}$$

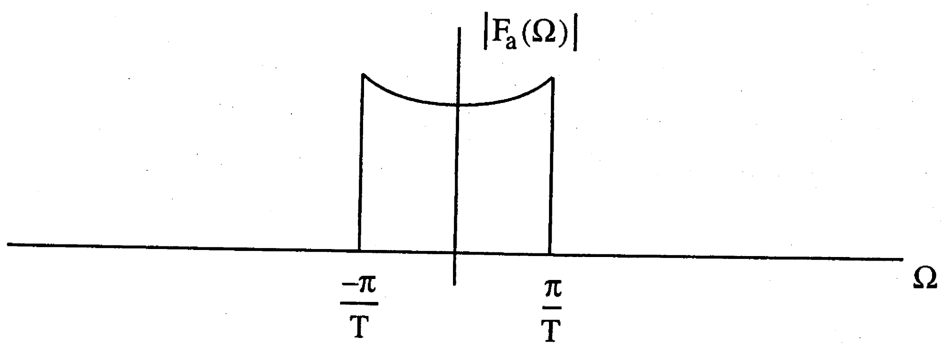
or

$$\boxed{F_a(\Omega) = \begin{cases} \dfrac{e^{j\frac{\Omega T}{2}}}{\text{sinc}\left(\frac{\Omega T}{2}\right)} & |\Omega| \le \frac{\pi}{T} \\ 0 & |\Omega| > \frac{\pi}{T} \end{cases}}$$

The first zero-crossing of sinc $\frac{\Omega T}{2}$ occurs when $\frac{\Omega T}{2} = \pi \Rightarrow \Omega = \frac{2\pi}{T}$.

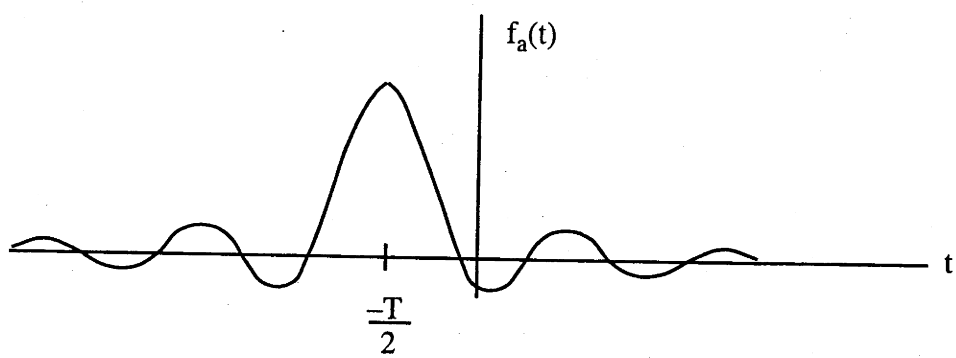


So, $|F_a(\Omega)|$ looks like:

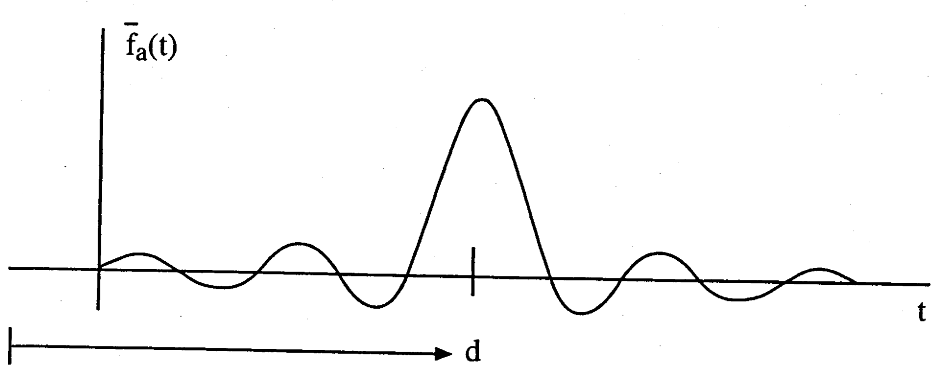


Thus, the ideal $F_a(\Omega)$ is a LPF that emphasizes the higher frequencies in its passband. (Surprising!)

$F_a(\Omega)$ has finite support $\Rightarrow$ $f_a(t)$ has infinite support. $f_a(t)$ might look something like:



In practice, we would use a filter with a causal impulse response $\bar{f}_a(t)$ with $\bar{f}_a(t) \approx f_a(t-d)\, u_a(t)$ (delayed and truncated version of $f_a(t)$).

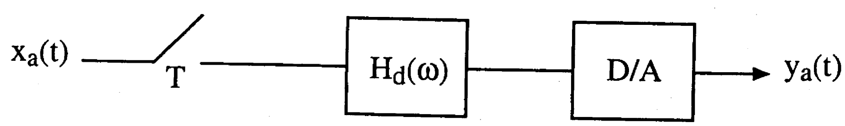


Using $\bar{f}_a(t)$ will delay the desired output by d seconds, but this is no problem in most applications if d is small.

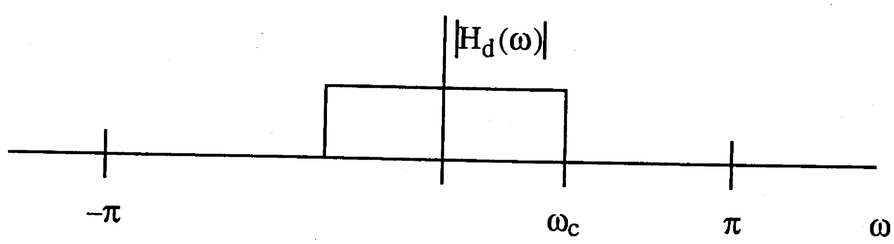
Notes:

1. In cheaper D/As, we may use a very simple R-C network to crudely approximate the desired $F_a(\Omega)$.

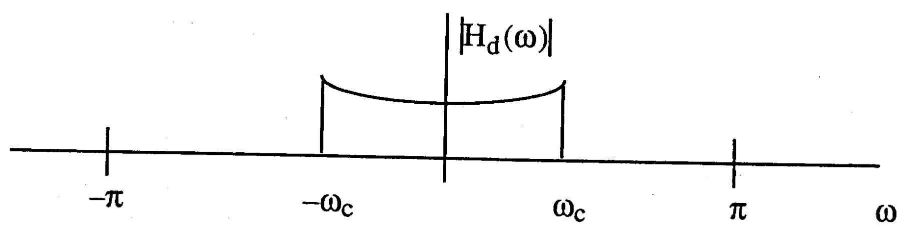
2. The high-frequency emphasis within the passband of $F_a(\Omega)$ can be performed digitally as part of the digital filter function. For example, if wish to realize an analog LPF using



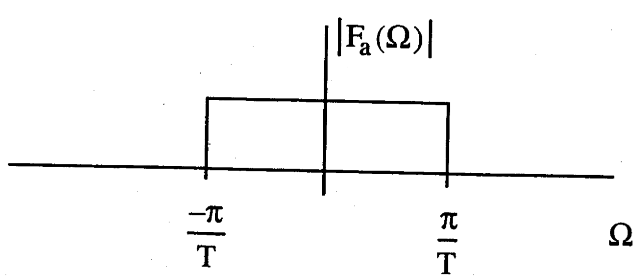
then instead of using



could use



In this case, we still need an $F_a(\Omega)$ after the ZOH, but now $F_a(\Omega)$ can be a regular LPF with a flat response in the passband:



ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 29

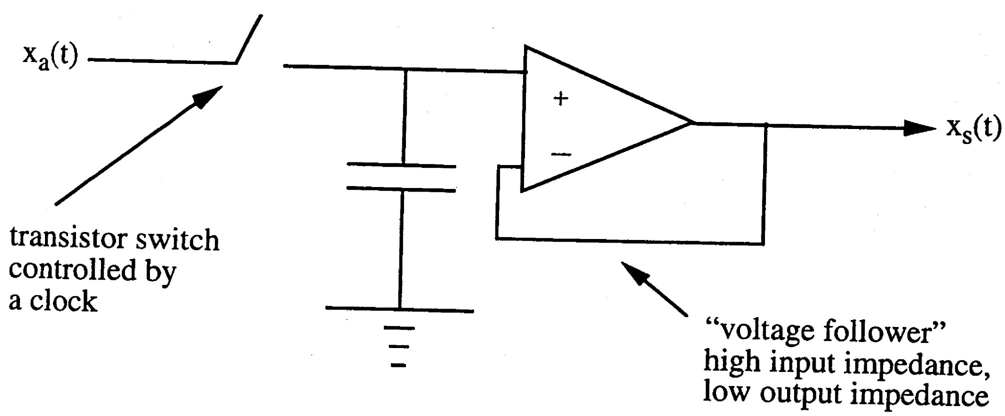
D. Munson

A/D and D/A Circuits

A/D consists of sample and hold followed by a quantizer.

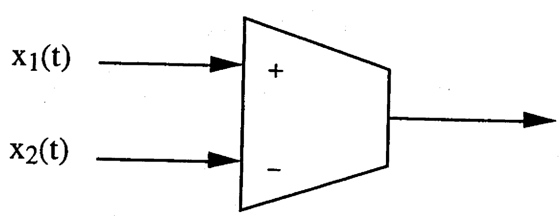
In catalogs, just the quantizer is called an A/D (unless A/D is referred to as a "sampling A/D"). As we shall see, the sample and hold is very simple, whereas the quantizer is much more complicated.

Sample and Hold:



A/D (Quantizer)

Uses comparators:

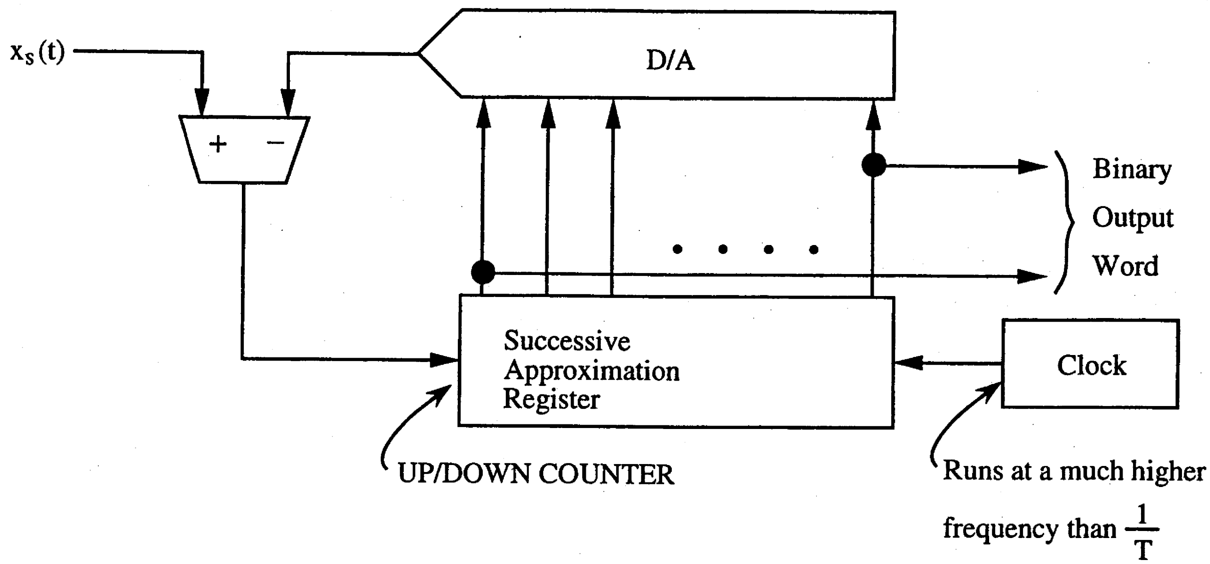


$$\begin{cases} 1 & x_1(t) \geq x_2(t) \\ 0 & x_1(t) < x_2(t) \end{cases}$$

Two popular types of A/D's:

a) Successive Approximation

~ for low and medium sampling rates; uses a D/A!



Here, $x_s(t)$ is the input from the sample and hold. The above system quantizes $x_s(t)$ to fit into a computer register. The comparator output signal causes the up-down counter to either increment or decrement, at a high rate, until it contains a binary approximation of $x_s(t)$. When the counter has settled around the correct digital representation of $x_s(t)$, it simply toggles back and forth in its least significant bit until the value of $x_s(t)$ changes.

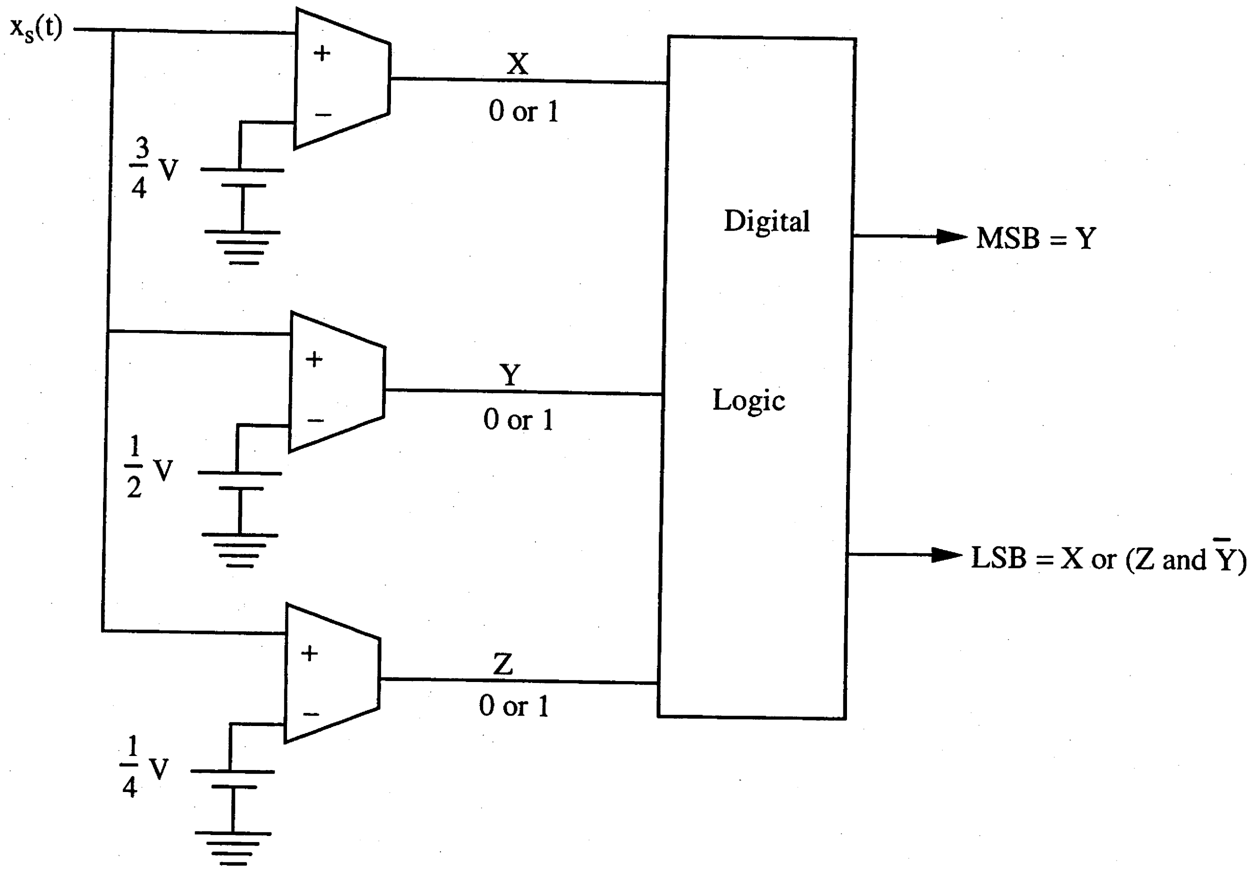
Succcessive approximation A/D's are fairly slow (and thus used for low and medium bandwidth applications) because it may take several clock cycles for the counter to settle on a new value of $x_s(t)$.

b) Parallel or Flash A/D

For high speed (8 bits/sample at 500 MHz is currently possible).

Uses 2^N-1 comparators for N-bit output word.

Example 2 bit quantizer:

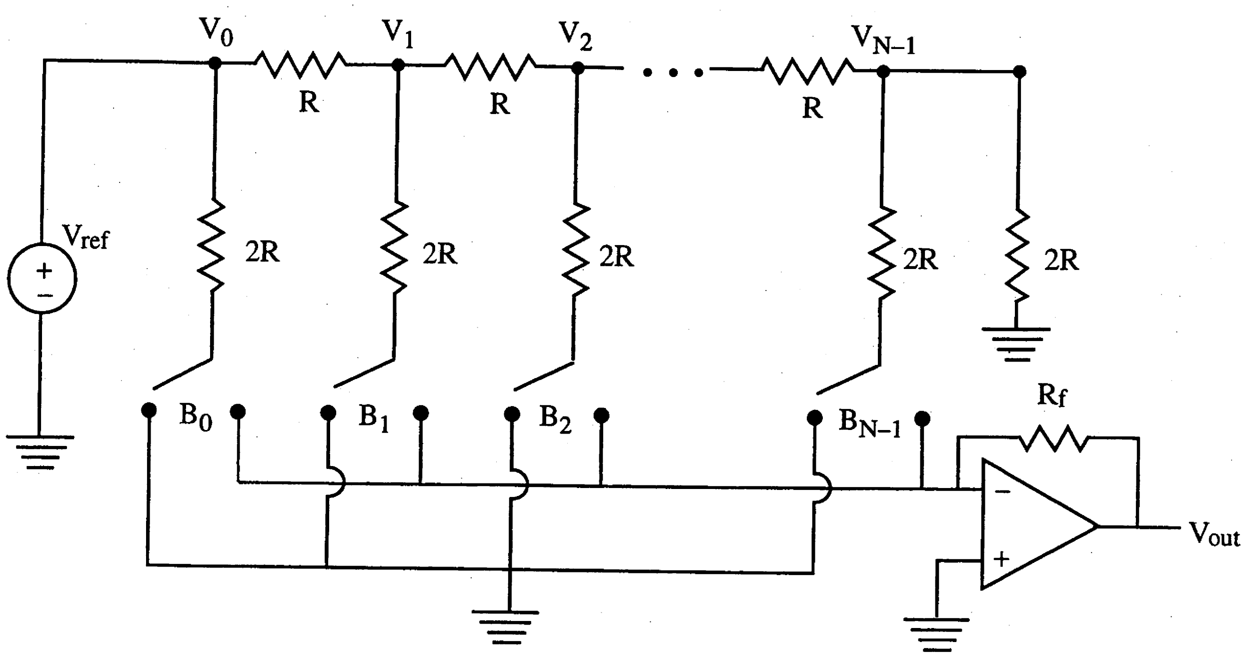


Here, $0 \le x_s(t) < \frac{1}{4}$ is mapped to (0, 0), $\frac{1}{4} \le x_s(t) < \frac{1}{2}$ is mapped to (0, 1), $\frac{1}{2} \le x_s(t) < \frac{3}{4}$ is mapped to (1, 0), and $x_s(t) \ge \frac{3}{4}$ is mapped to (1, 1).

D/A Converters ~ Zero Order Hold (ZOH)

The contents of a binary register containing y_n are the input to a D/A. Let $[B_0, B_1, B_2, \ldots, B_{N-1}]$ be the binary representation of y_n. The B_i change with period T as y_{n+1} replaces y_n in the D/A input register.

One popular type of D/A uses a resistor ladder (can also use a capacitor ladder):



The switches are transistors, where the B_i control whether the transistors conduct to ground (left position, $B_i = 0$) or to the op amp (right position, $B_i = 1$). The op amp then adds all signals input to its minus terminal, with a weighting determined by the resistor values. To find the exact relationship between V_{out} and the B_i, first apply KCL at Node N–1 at the upper right, to give:

$$\frac{V_{N-1}}{2R} + \frac{V_{N-1}}{2R} + \frac{V_{N-1} - V_{N-2}}{R} = 0$$

$$\Rightarrow V_{N-1} + V_{N-1} - V_{N-2} = 0 \Rightarrow V_{N-2} = 2\,V_{N-1}$$

Similarly: $V_{n-1} = 2V_n \quad n = 1, 2, \ldots, N-2$

$$\Rightarrow V_n = V_{N-1}\, 2^{N-1-n}$$

Using KCL at the minus terminal of the op amp gives:

$$\frac{1}{2R} \sum_{i=0}^{N-1} B_i V_i = \frac{0 - V_{out}}{R_f}$$

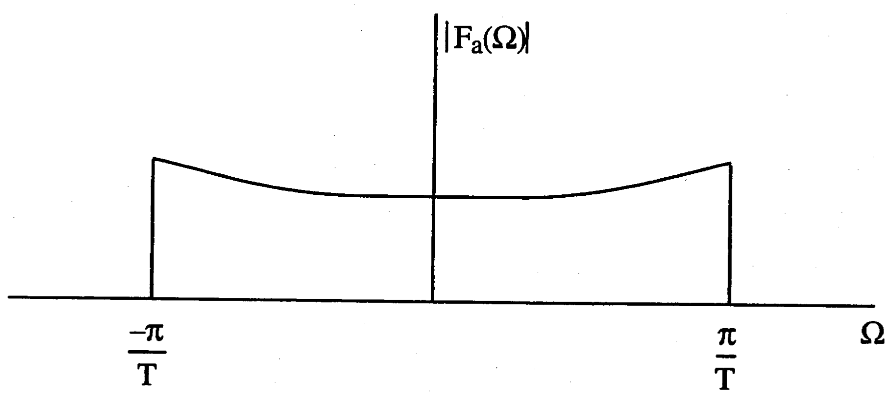
where each B_i is 0 or 1.

$$\Rightarrow \quad V_{out} = \frac{-R_f}{2R} \sum_{i=0}^{N-1} B_i \, V_{N-1} \, 2^{N-1-i}$$

$$= \frac{-R_f}{2R} V_{N-1} \left[2^{N-1} B_0 + 2^{N-2} B_1 + \ldots + 2 \, B_{N-2} + B_{N-1}\right]$$

So, V_{out} is proportional to the number stored in the binary register representing y_n. This number changes according to a clock ($y_n \rightarrow y_{n+1}$), so $V_{out}(t)$ is a staircase function (edges won't be perfectly square, though — op amp has a nonzero rise time).

The ZOH is followed with the analog LPF, below, as discussed previously.



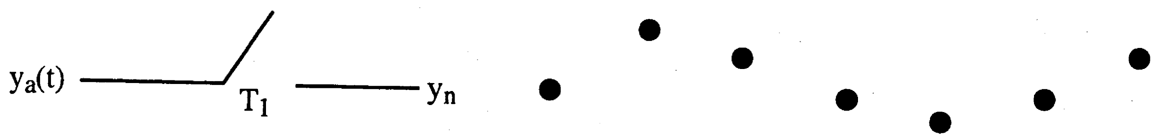
ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 30

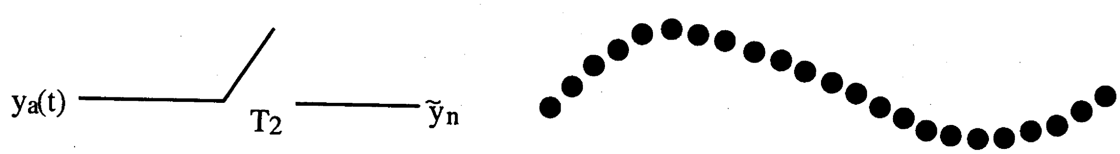
D. Munson

Digital Interpolation

Suppose have $y_n = y_a(nT_1)$ as pictured:



and want $\tilde{y}_n = y_a(nT_2)$ where $T_2 = \frac{T_1}{L}$, and L is integer :



Thus, $\{\tilde{y}_n\}$ are denser samples of $y_a(t)$. How do we get $\{\tilde{y}_n\}$ from $\{y_n\}$? Could use

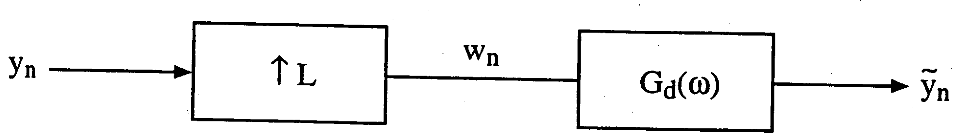
$$y_a(t) = \sum_{k=-\infty}^{\infty} y_k \operatorname{sinc}\left[\frac{\pi}{T_1}(t - kT_1)\right]$$

to get

$$\tilde{y}_n = y_a(nT_2) = \sum_{k=-\infty}^{\infty} y_k \operatorname{sinc}\left[\frac{\pi}{T_1}(nT_2 - kT_1)\right]$$

But, this involves an infinite sum (which must be truncated in practice) and evaluation of the sinc function. Alternatively, we might try something simpler, such as a piecewise linear or polynomial approximation to $y_a(t)$, but these methods are not particularly accurate.

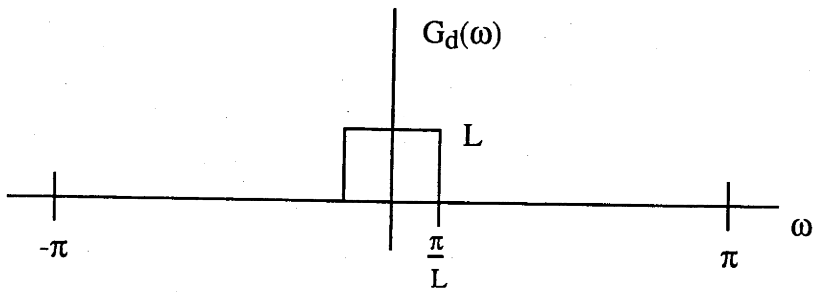
Alternative Digital Approach



where the first box is an up-sampler that inserts L-1 zeros between each pair of inputs:

$$w_n = \{0, 0, \ldots, 0, y_{-1}, \underbrace{0, 0, \ldots, 0}_{\text{L-1 zeros}}, y_0, 0, 0, \ldots, 0, y_1, 0 \ldots\}$$

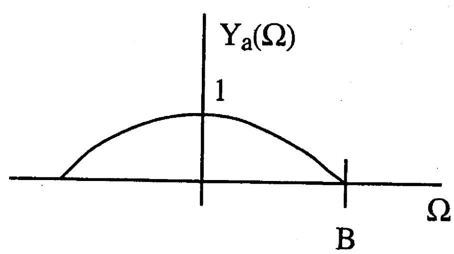
and $G_d(\omega)$ is an ideal LPF with cutoff π/L and passband gain L:



Why does this work??

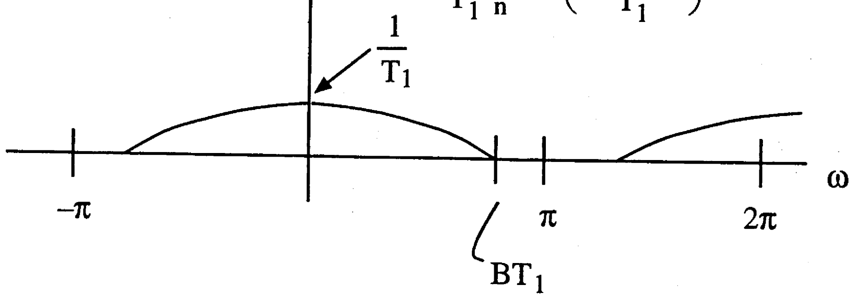
Analyze the problem in the Fourier domain.

First, note that if



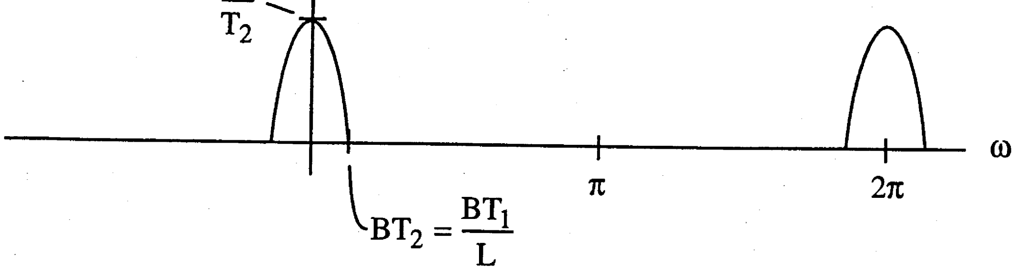
then sampling $y_a(t)$ at times nT_1 $\left(\text{with } T_1 < \frac{\pi}{B}\right)$ would give $\{y_n\}$ with

$$Y_d(\omega) = \frac{1}{T_1} \sum_n Y_a\left(\frac{\omega + 2\pi n}{T_1}\right)$$



Sampling $y_a(t)$ on the denser grid nT_2 would give $\{\tilde{y}_n\}$ with

$$\tilde{Y}_d(\omega) = \frac{1}{T_2}\sum_n Y_a\left(\frac{\omega+2\pi n}{T_2}\right)$$



(Sampling at a higher frequency <u>shrinks</u> the DTFT of the A/D output).

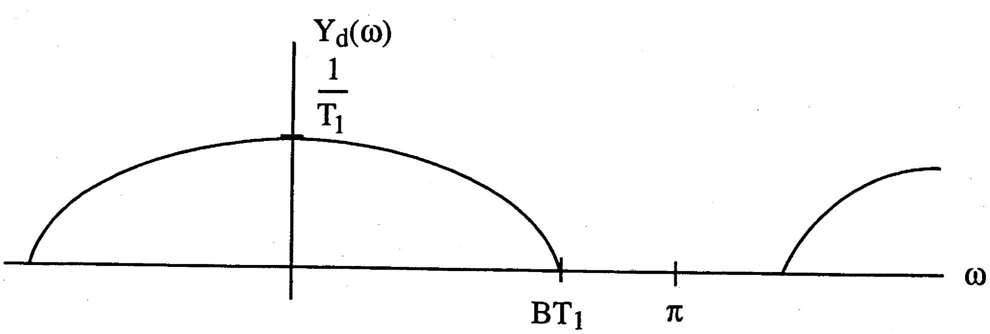
Now, show that the above digital interpolation approach gives $\tilde{Y}_d$ from Y_d (and therefore $\tilde{y}_n$ from y_n).

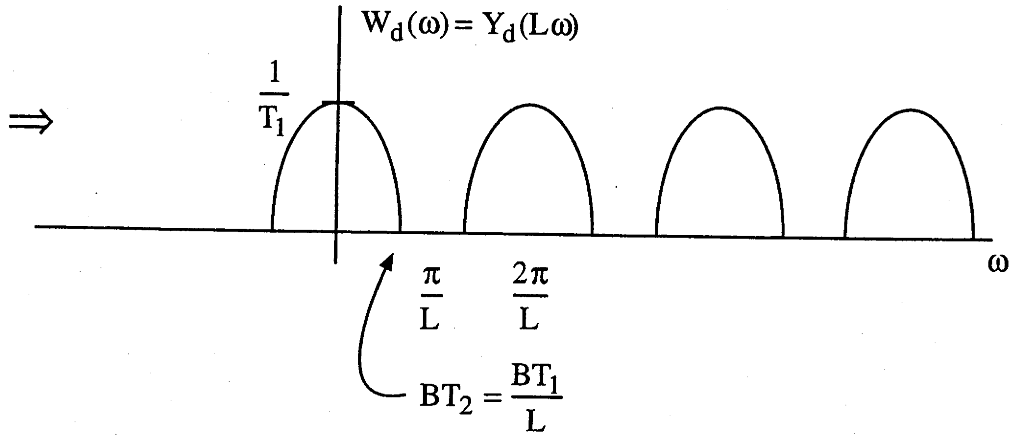
We have

$$W_d(\omega) = \sum_n w_n\, e^{-j\omega n} = \sum_n y_n\, e^{-j\omega L n}$$

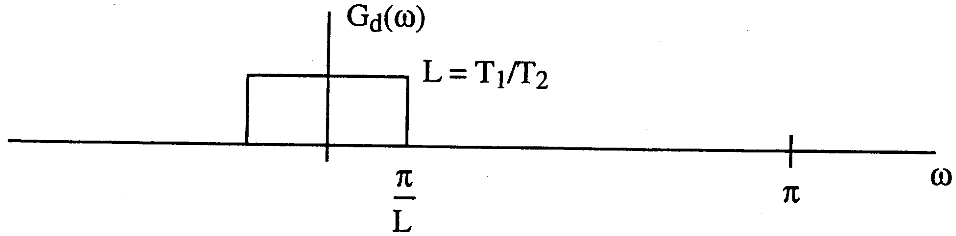
$$\Rightarrow \quad \boxed{W_d(\omega) = Y_d(L\omega)}$$

So,



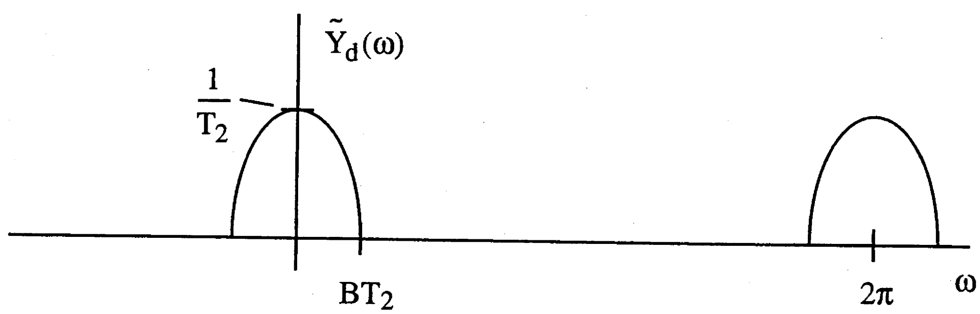


Now, since



we have

$\tilde{Y}_d(\omega) = G_d(\omega)\, W_d(\omega)$ given by



as desired. Thus, in principle, the digital interpolator will compute $\{\tilde{y}_n\}$ from $\{y_n\}$ exactly. In practice, the quality of the interpolator depends on the quality of $G_d(\omega)$), i.e., on how close $G_d(\omega)$ is to the ideal low-pass shape.

Comments:

1) L-1 out of every L inputs to $G_d(\omega)$ are zero. This saves many multiplications for L large! This is readily apparent for nonrecursive $G_d(\omega)$, but is also true for some recursive $G_d(\omega)$.

2) There exist efficient digital interpolation schemes for $T_2 = \alpha\, T_1$, where α is any real number (doesn't have to be $\frac{1}{L}$).

A Further Look at Up-Sampler

A digital interpolator uses an up-sampler as one of its components.

$$x_n \longrightarrow \boxed{\uparrow L} \longrightarrow y_n$$

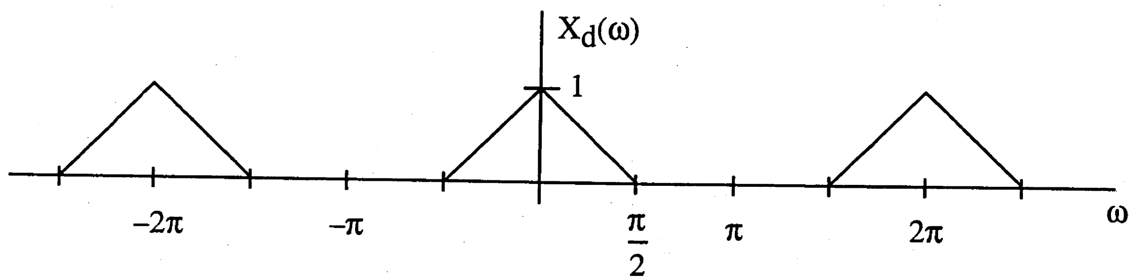
We have shown that $Y_d(\omega)$ is a squashed version of $X_d(\omega)$, namely

$$Y_d(\omega) = X_d(L\omega).$$

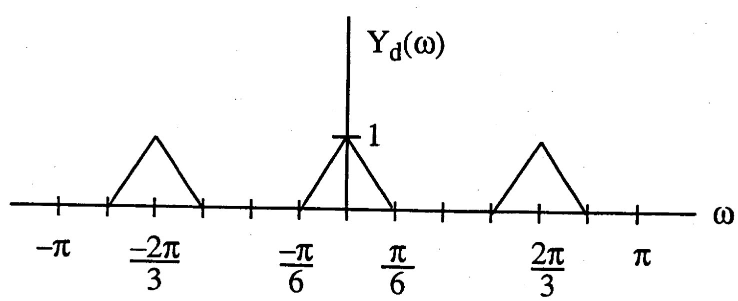
Notice that the amplitude of Y_d is the same as the amplitude of X_d. This makes intuitive sense since the energy in the y_n sequence is the same as that of the x_n sequence, because the up-sampler inserts just zeros between the x_n elements.

Example (Up-Sampler)

Suppose $L = 3$. Sketch $Y_d(\omega)$, assuming



The entire ω axis is squashed by a factor of 3 to give



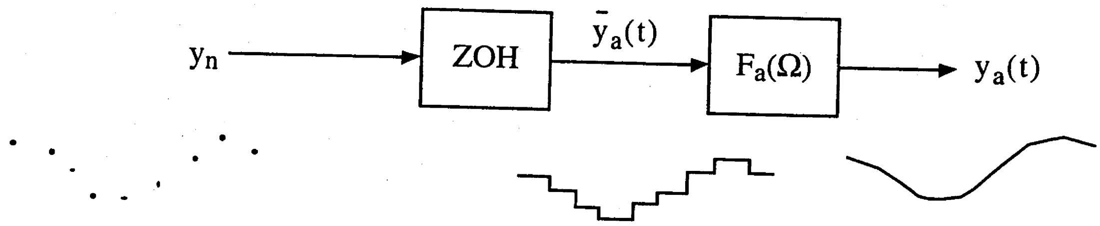
ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 31

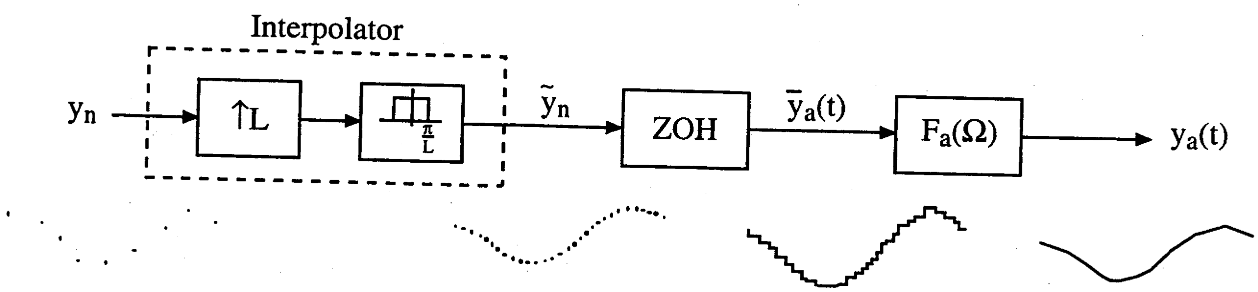
D. Munson

Oversampling D/A

Used in C-D players, for example. Idea is to simplify analog filter in D/A by using interpolation prior to the D/A. Interpolating $\{y_n\}$ prior to the D/A permits the use of a ZOH with a smaller step-size. This ZOH puts out a finer staircase approximation to $y_a(t)$, which relaxes the requirements on $F_a(\Omega)$. So, instead of this:

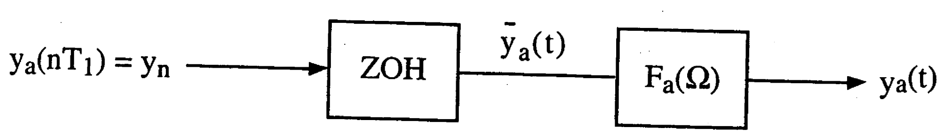


do this:



As you can imagine, a far simpler filter $F_a(\Omega)$ can be used in the second system to produce $y_a(t)$ from $\bar{y}_a(t)$, since $\bar{y}_a(t)$ is much smoother in the second system than in the first system. We gain considerable insight into this via the following analysis.

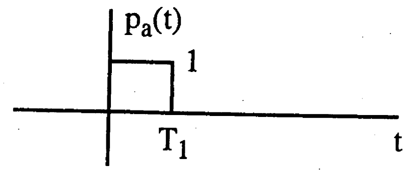
Our analysis of the oversampling D/A is facilitated by <u>first</u>, considering a usual D/A, assuming sampling period of T_1. The standard way to reconstruct $y_a(t)$ from $y_n = y_a(nT_1)$ is:



where

$$\bar{y}_a(t) = \sum_n y_n \; p_a(t - nT_1)$$

with



and

$$\overline{Y}_a(\Omega) = P_a(\Omega)\, Y_d(\Omega T_1)$$

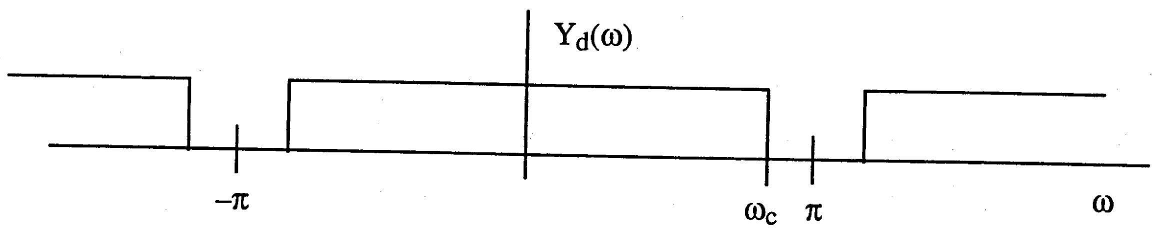
↑

from analysis of general D/A

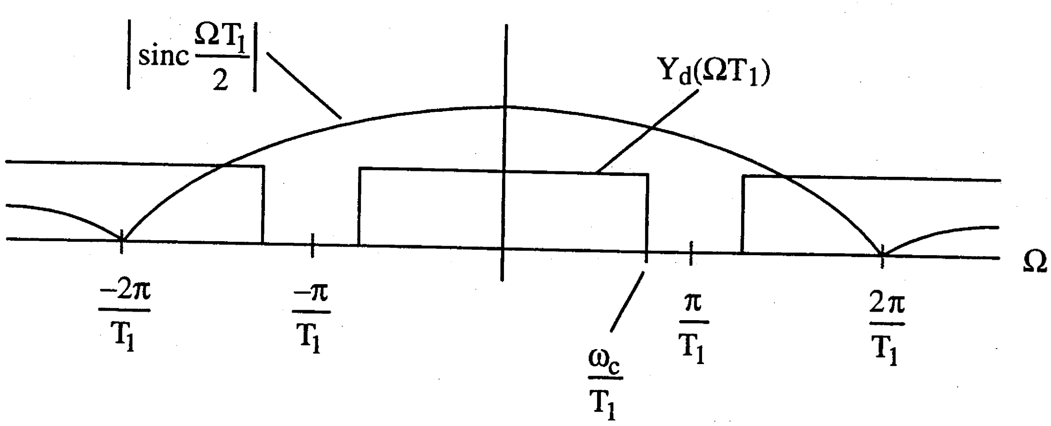
so that

$$\overline{Y}_a(\Omega) = T_1 \operatorname{sinc} \frac{\Omega T_1}{2}\, e^{-j\frac{\Omega T_1}{2}}\, Y_d(\Omega T_1) \qquad (\square)$$

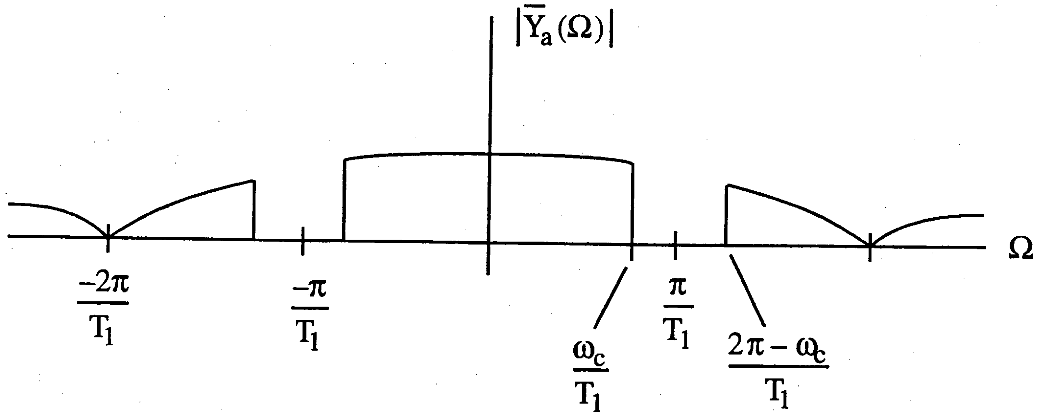
As a specific example, assume



Then $|\overline{Y}_a(\Omega)|$ is the <u>product</u> of the following two curves:



giving

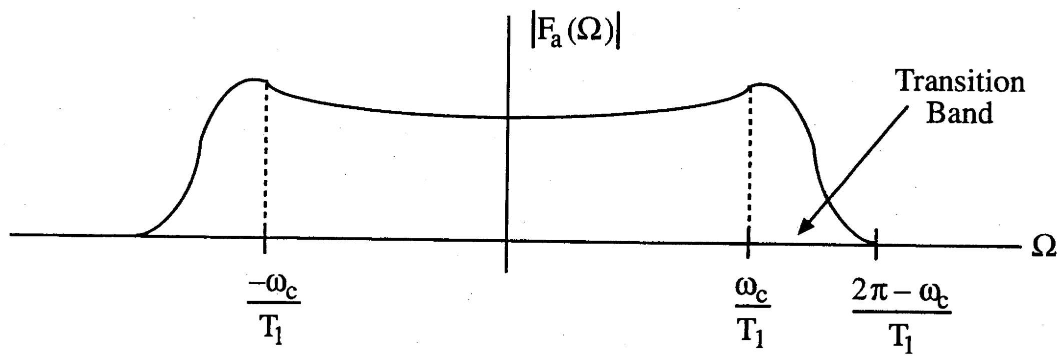


Now, as we know, $F_a(\Omega)$ should be a LPF with a

$$\frac{1}{\operatorname{sinc}\frac{\Omega T_1}{2}}$$

shape in its passband. For the situation above, with $\omega_c < \pi$, there is room for a <u>transition band</u> of $F_a(\Omega)$ on the interval $\frac{\omega_c}{T_1} < |\Omega| < \frac{2\pi-\omega_c}{T_1}$. A finite-order (realizable) $F_a(\Omega)$ needs room for a transition band (transition cannot be infinitely sharp). A wider transition band permits a lower order (less complicated) $F_a(\Omega)$.

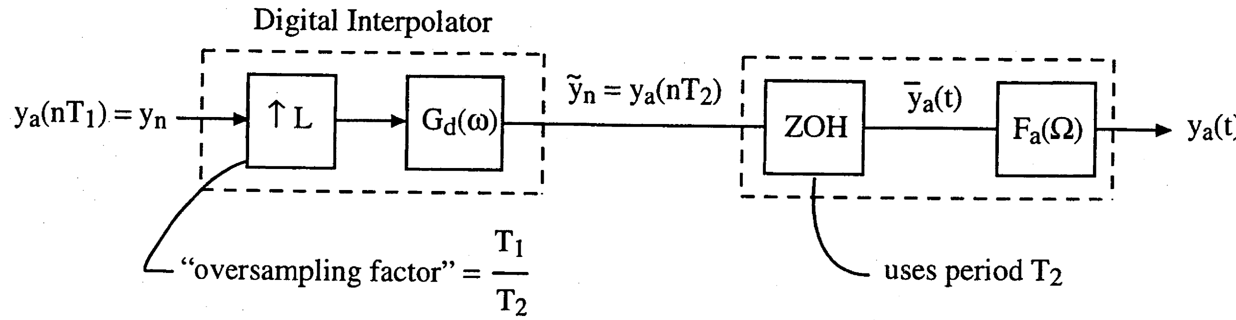
A realizable $F_a(\Omega)$ might look like:



This filter is permitted a transition bandwidth of

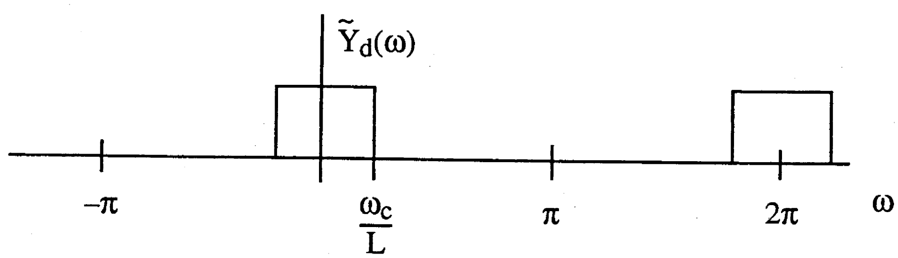
$$\frac{2\pi-\omega_c}{T_1} - \frac{\omega_c}{T_1} = \frac{2(\pi-\omega_c)}{T_1}$$

Now, consider oversampling D/A:

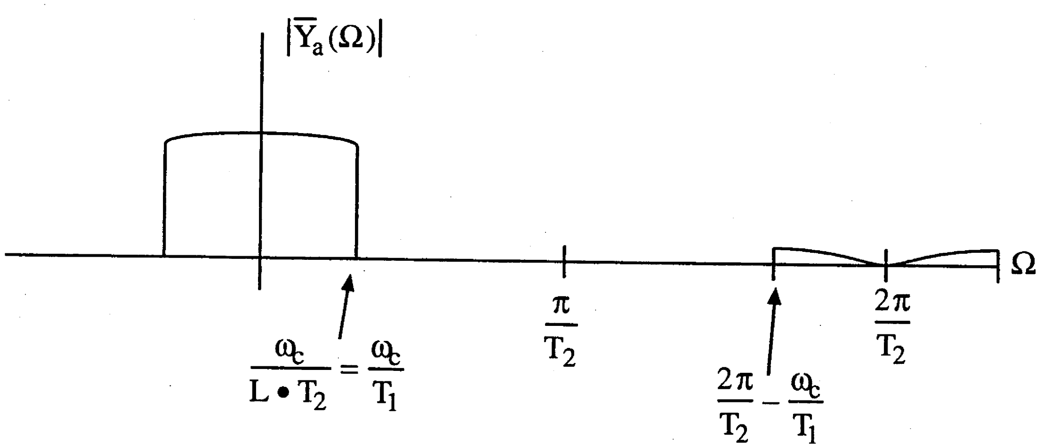


Due to the interpolation, the above ZOH puts out a finer staircase approximation with narrow steps (width T_2). Thus, we expect that $F_a(\Omega)$ can be simpler in this scheme. Let's analyze this in the frequency domain:

The interpolator squashes the DTFT of y_n:



So, $|\bar{Y}_a(\Omega)|$ now looks like the curve below (use eqn. (❑) except with T_2 instead of T_1 and $\tilde{Y}_d$ instead of Y_d):



Thus, the transition band of $F_a(\Omega)$ can now be much wider.

$$\text{Transition BW} = \frac{2\pi}{T_2} - \frac{\omega_c}{T_1} - \frac{\omega_c}{T_1}$$

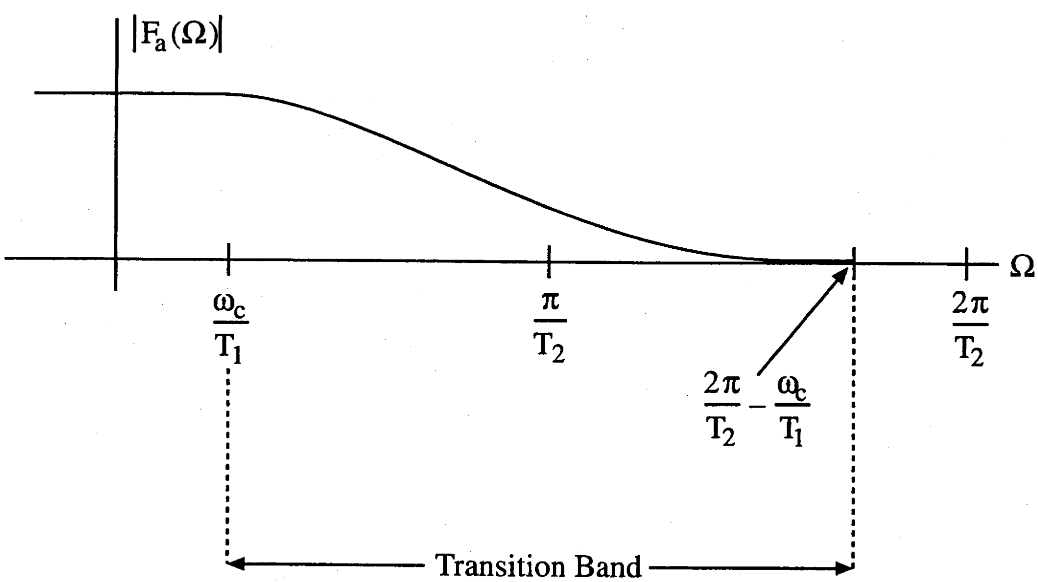
$$= \frac{2(L\pi - \omega_c)}{T_1} >> \frac{2(\pi - \omega_c)}{T_1}$$

↑ from before for regular D/A

so that implementation of $F_a(\Omega)$ can be far simpler.

Also, from the picture above we see that the center pulse of $\overline{Y}_a(\Omega)$ is almost flat and that the artifact centered at $\frac{2\pi}{T_2}$ is nearly zero, so even a fairly crude $F_a(\Omega)$ will do a good job. $F_a(\Omega)$ should have a nearly flat response in its passband, can have a huge transition band, and needs only moderate attenuation in its stopband.

$F_a(\Omega)$ in an oversampling D/A can look like:

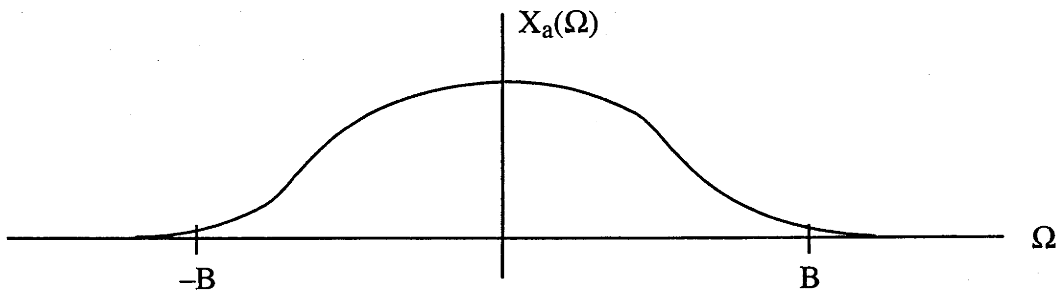


Oversampling A/D

A different type of oversampling is sometimes used to limit aliasing in the A/D. We will examine this as the second method, described below, for preventing aliasing at the A/D.

Prevention of Aliasing at A/D

Suppose $x_a(t)$ is nearly (not exactly) BL to B rad/sec.

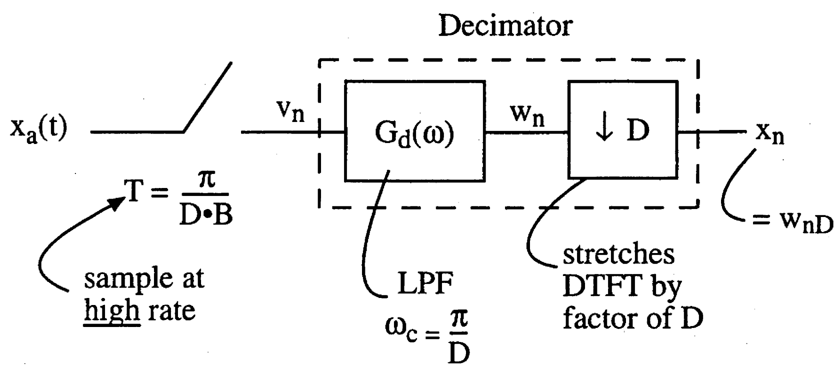


Here, B is an "effective band limit," but sampling with $T = \frac{\pi}{B}$ will still cause measurable aliasing.

How do we prevent aliasing at the sampler? Two possibilities:

1) Precede the A/D with an analog "antialiasing" LPF with cutoff B rad/sec. Then sample using $T < \frac{\pi}{B}$. This approach is very common.

2) Alternatively, sample at a high rate with $T = \frac{\pi}{B \bullet D}$ where D is an integer and is large enough to virtually prevent aliasing (choose D so that $X_a(\Omega)$ is virtually limited to $D \bullet B$ rad/sec). Then digitally LPF with cutoff $\omega_c = \frac{\pi}{D}$. Then decimate by a factor of D (discard D–1 of every D samples). This is called an oversampling A/D:



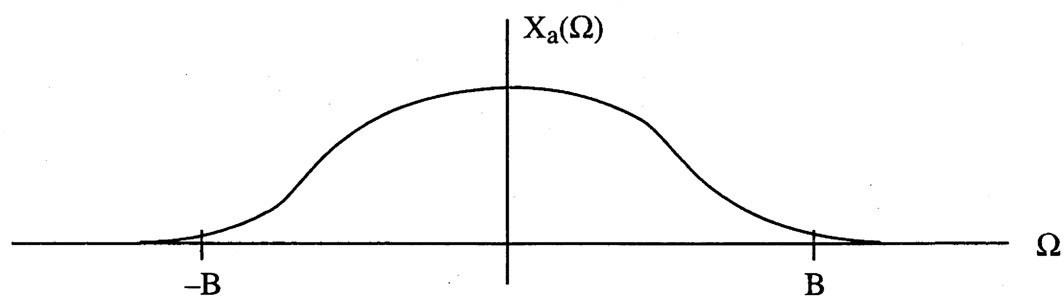
Ordinarily, sampling at such a high rate would be an expensive proposition, since this could create a very high data rate. The decimator, however, reduces the sampling rate back down by a factor of D. Note that implementation of $G_d(\omega)$ is not nearly so complicated as you might expect. Since D-1 of every D outputs of $G_d(\omega)$ will be discarded, only every Dth output need be computed!

Choosing between 1) and 2) is simply an issue of whether you put the complexity in the analog or digital part of your system.

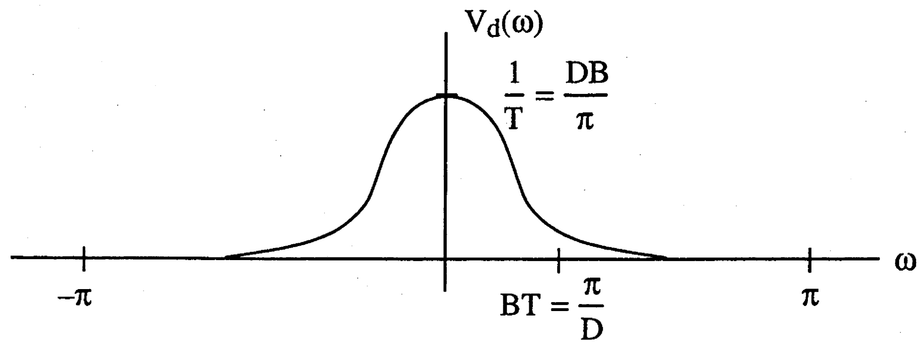
You generally do need 1) or 2) in practice..

Analysis of Oversampling A/D

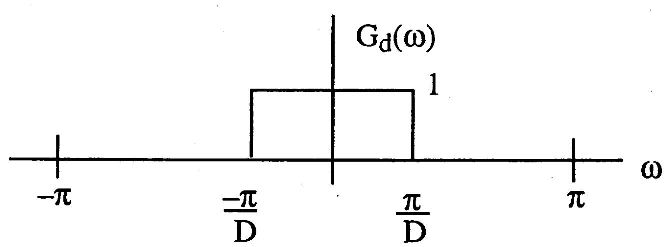
We will show that Option 2 (oversampling approach) produces exactly the same output $\{x_n\}$ as does Option 1. Suppose:



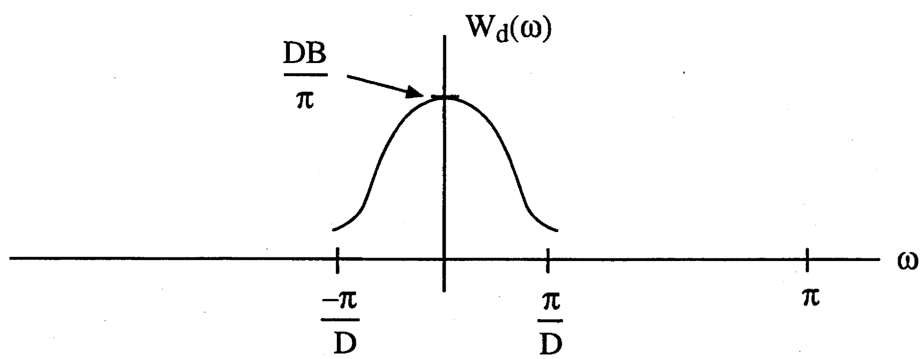
If $T = \dfrac{\pi}{D \bullet B}$ then



We have



So:



Now, what is the relationship between $X_d(\omega)$ and $W_d(\omega)$?

<u>Digression</u>

Note

$$\frac{1}{D}\sum_{k=0}^{D-1} e^{j\frac{2\pi}{D}kn} = \begin{cases} 1 & n = mD \\ \dfrac{1}{D}\dfrac{1-e^{j\frac{2\pi}{D}nD}}{1-e^{j\frac{2\pi n}{D}}} & n \neq mD \end{cases}$$

$$= \begin{cases} 1 & n = mD \\ 0 & n \neq mD \end{cases}$$

Now,

$$X_d(\omega) = \sum_{n=-\infty}^{\infty} x_n e^{-j\omega n}$$

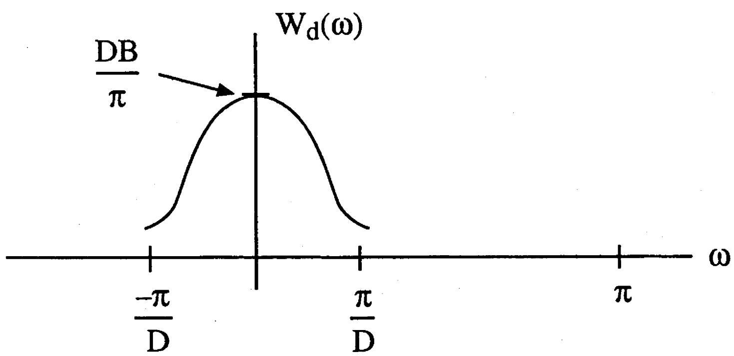
$$= \sum_{n=-\infty}^{\infty} w_{nD} e^{-j\omega n} = \sum_{\substack{n=mD \\ m=-\infty}}^{\infty} w_n e^{-j\omega \frac{n}{D}}$$

$$\underset{\substack{\uparrow \\ \text{trick from} \\ \text{digression}}}{=} \sum_{n=-\infty}^{\infty} w_n \underbrace{\frac{1}{D}\sum_{k=0}^{D-1} e^{j\frac{2\pi}{D}kn}}_{= 0 \text{ unless } n = mD} e^{-j\omega\frac{n}{D}}$$

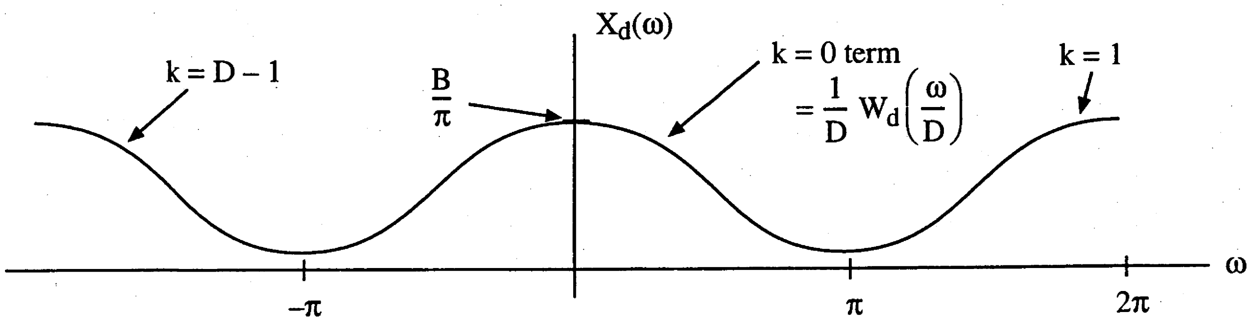
$$= \frac{1}{D}\sum_{k=0}^{D-1} \sum_{n=-\infty}^{\infty} w_n e^{-jn\left(\frac{\omega-2\pi k}{D}\right)}$$

$$\Rightarrow \quad \boxed{X_d(\omega) = \frac{1}{D}\sum_{k=0}^{D-1} W_d\left(\frac{\omega - 2\pi k}{D}\right)} \qquad (\Delta)$$

Now, had



Using this $W_d(\omega)$ in (Δ) gives



Note: This X_d is just what we would have obtained if we had analog low-pass filtered $x_a(t)$ to B rad/sec and then sampled with period $T = \frac{\pi}{B}$!

Thus, 2) does an equivalent job to 1).

Note:

How can (Δ) produce a periodic $X_d(\omega)$? (Δ) has only a finite number of terms in its sum. Answer: Each term is a periodic DTFT, not a FT as in Eq. ($\Diamond$).

k=0 term in (Δ) has pulses centered at 0, $\pm 2\pi D$, $\pm 4\pi D$, etc.

k=1 term has pulses centered at 2π, $2\pi \pm 2\pi D$, $2\pi \pm 4\pi D$, etc.

⋮

$k = D - 1$ term has pulses centered at $(D-1)2\pi$, $(D-1)2\pi \pm 2\pi D$, $(D-1)2\pi \pm 4\pi D$, etc.

A Further Look at Down-Sampler

A decimator uses a down-sampler as one of its components:

$$x_n \longrightarrow \boxed{\downarrow D} \longrightarrow y_n$$

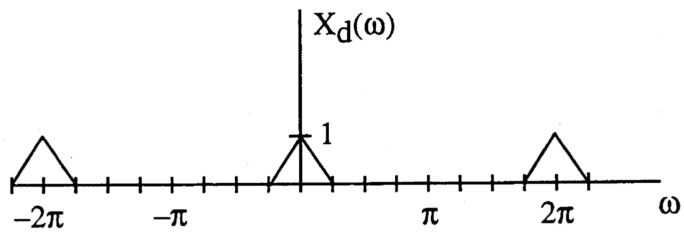
The down-sampler essentially stretches X_d. However, if $X_d(\omega)$ is not limited to $|\omega| < \frac{\pi}{D}$, then aliasing also occurs. Specifically,

$$Y_d(\omega) = \frac{1}{D} \sum_{k=0}^{D-1} X_d\left(\frac{\omega - 2\pi k}{D}\right). \qquad (\Delta)$$

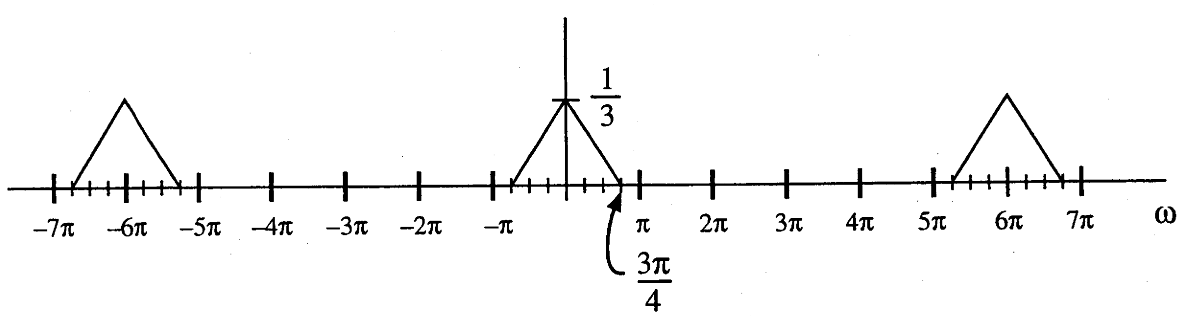
Notice the scaling in amplitude by $\frac{1}{D}$. This factor is not surprising, given that in the time domain, the down-sampler discards D–1 out of every D samples. By contrast, the up-sampler does not discard any samples, and inserts only zero-valued samples, so that there is no amplitude scaling in the Fourier domain for the up-sampler.

Example (Down-Sampler)

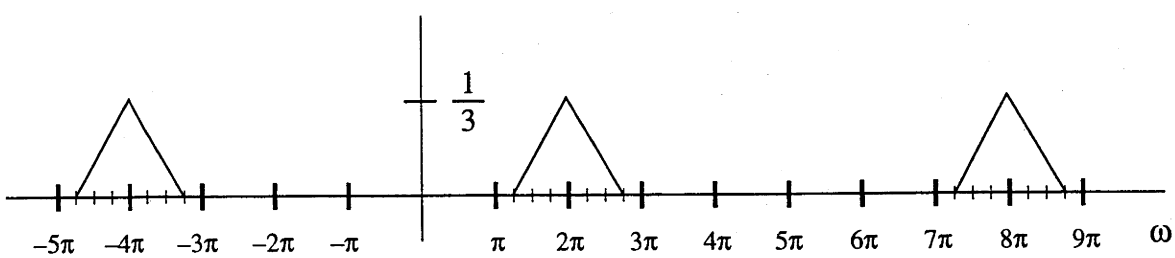
Suppose D = 3. Sketch $Y_d(\omega)$, assuming



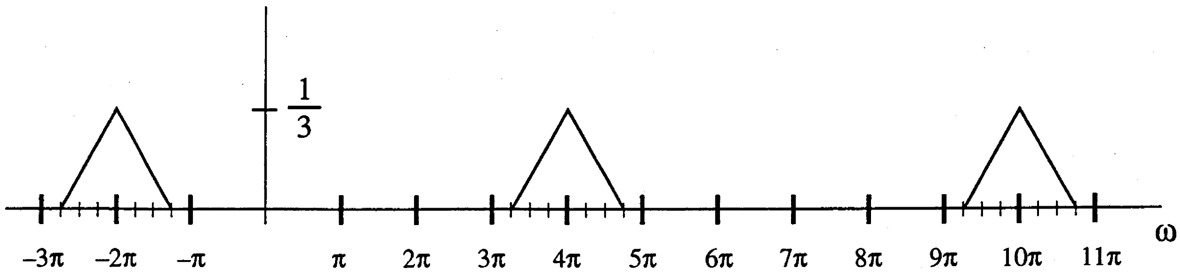
Then the k = 0 term in (Δ) is



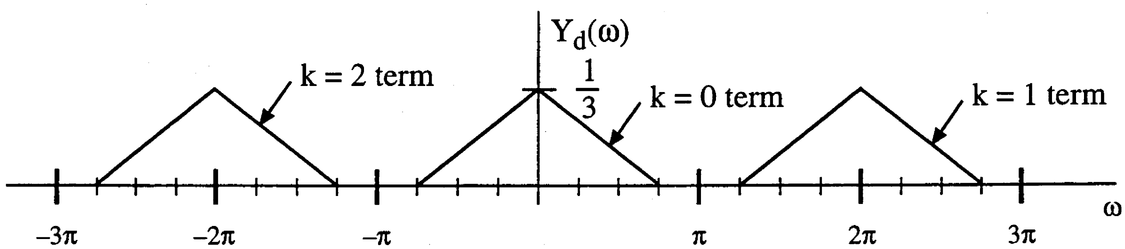
The k = 1 term in (Δ) is a 2π-shifted version of the above, namely



Likewise, the $D - 1 = 2$ term in (Δ) is a 4π-shifted version of the $k = 0$ term:



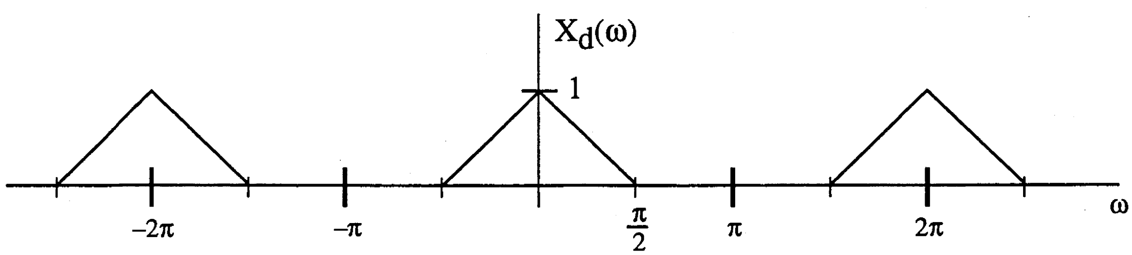
Adding the three previous plots together gives $Y_d(\omega)$:



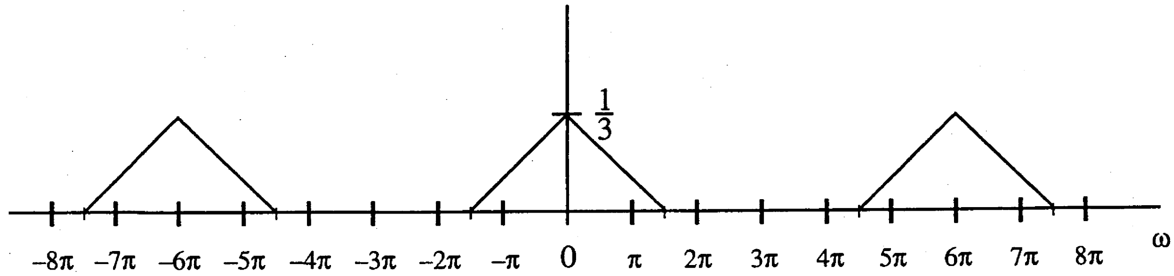
Note that the various terms in (Δ) interlace to produce a 2π-periodic $Y_d(\omega)$. In this example there was no need to plot the $k = 0, 1, 2$ terms, since the $k = 0$ term, alone, determines the shape of $Y_d(\omega)$ for $|\omega| < \pi$. In the next example, the downsampler causes aliasing, so that the terms in (Δ) overlap. This situation is more complicated than in the previous example.

Example (Down-Sampler)

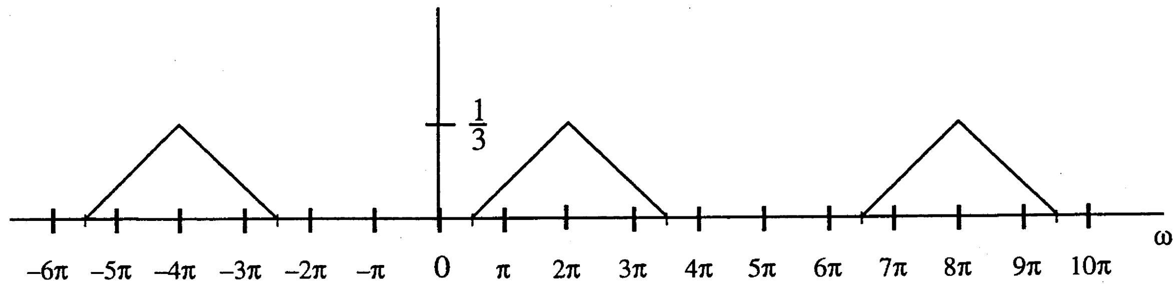
Suppose $D = 3$ as before, but now with



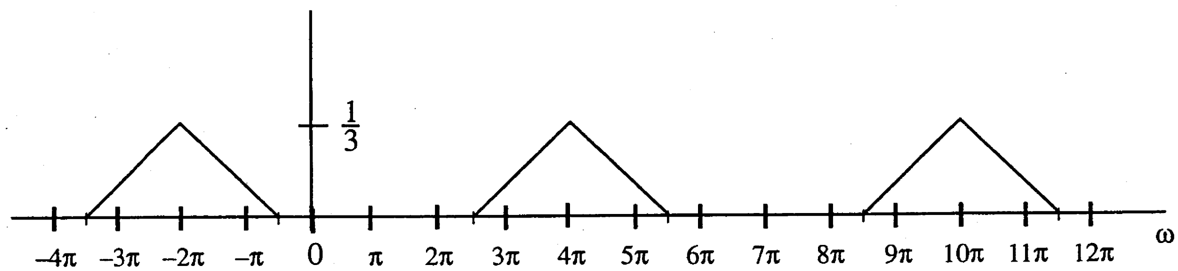
The $k = 0$ term in (Δ) is:



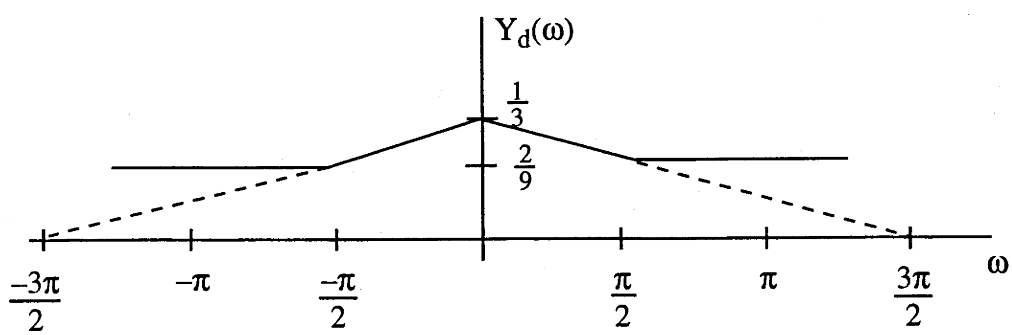
Notice that the center pulse extends beyond $\omega = \pm\,\pi$, which is an indication of aliasing. The $k = 1$ term in (Δ) is a 2π-shifted version of the above plot, namely:



The $k = 2$ term in (Δ) is a 4π-shifted version of the $k = 0$ term:



Adding the $k = 0, 1, 2$ terms gives $Y_d(\omega)$, which we plot only for $|\omega| \leq \pi$:



In this example, we have aliasing because $X_d(\omega)$ extends beyond $\omega = \pm\,\frac{\pi}{D} = \frac{\pi}{3}$. In a decimator, the job of the LPF that precedes the down-sampler is to cut off $X_d(\omega)$ at $\omega = \frac{\pi}{D}$ to prevent this aliasing.

ECE 310 **DIGITAL SIGNAL PROCESSING** **D. Munson**
University of Illinois **Lecture 33**

Classes of Digital Filters

FIR – Finite impulse response; $\{h_n\}$ finite in length

IIR – Infinite impulse response; $\{h_n\}$ infinite in length

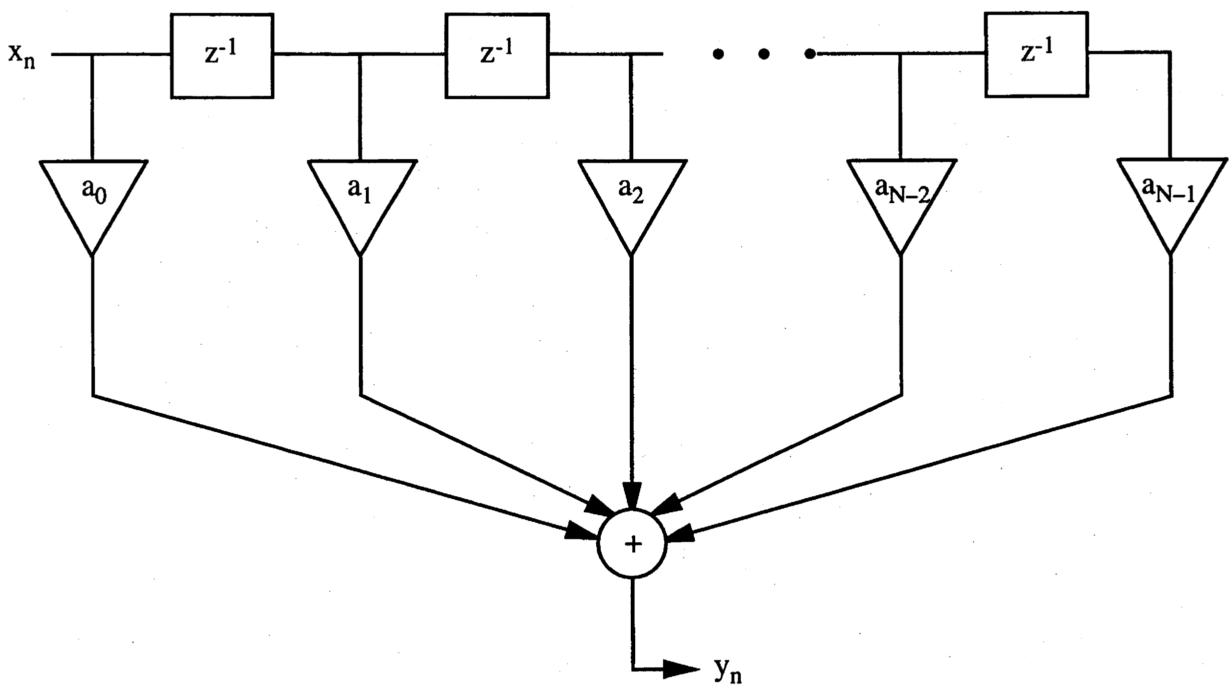
<u>FIR Filter Structures</u>

$$h_n = \{a_0, a_1, a_2, \ldots, a_{N-1}, 0, 0, \ldots\}$$

$$\Rightarrow \quad H(z) = \sum_{n=0}^{N-1} a_n \, z^{-n}$$

TF is a polynomial in z^{-1}

Direct Form Structure:

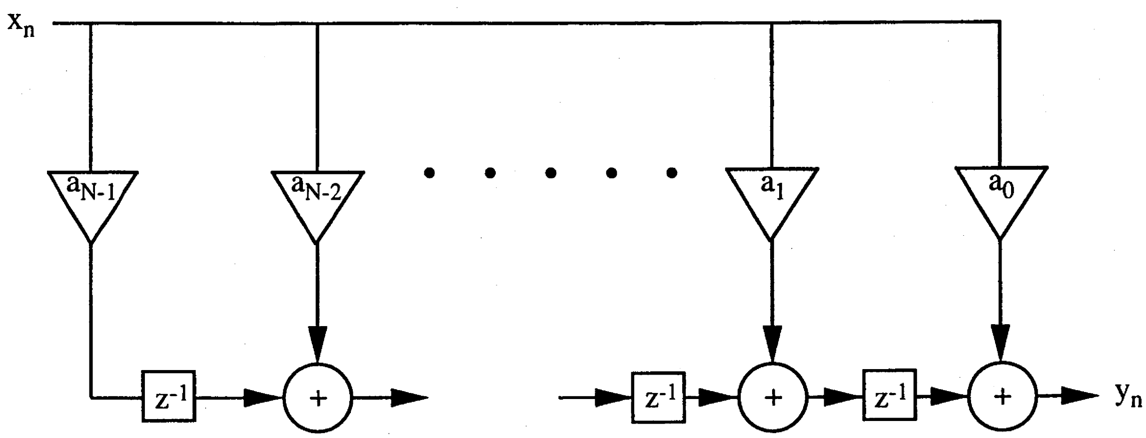


Due to the arrangement of the delays, this is also called a transversal filter or tapped delay-line filter.

The implementation of a transfer function is not unique. The transfer function describes only the input-output properties of the system. For any transfer function, there are an infinite number of possible realizations of that transfer function. For example, consider the transpose-form structure.

Transpose Form:

(obtained by reversing all flows)



This structure has the same transfer function as the Direct Form structure and is very commonly used.

Advantage: Easier to fully parallelize. No adder tree at output as in Direct Form.

FIR filters are nearly always implemented nonrecursively as in the above diagrams. Theoretically, though, FIR filters can be recursive, as shown in the following example.

Example

Disguise the FIR transfer function as a rational function with non-unity denominator:

$$H(z) = a_0 + a_1 z^{-1} + a_2 z^{-2}$$

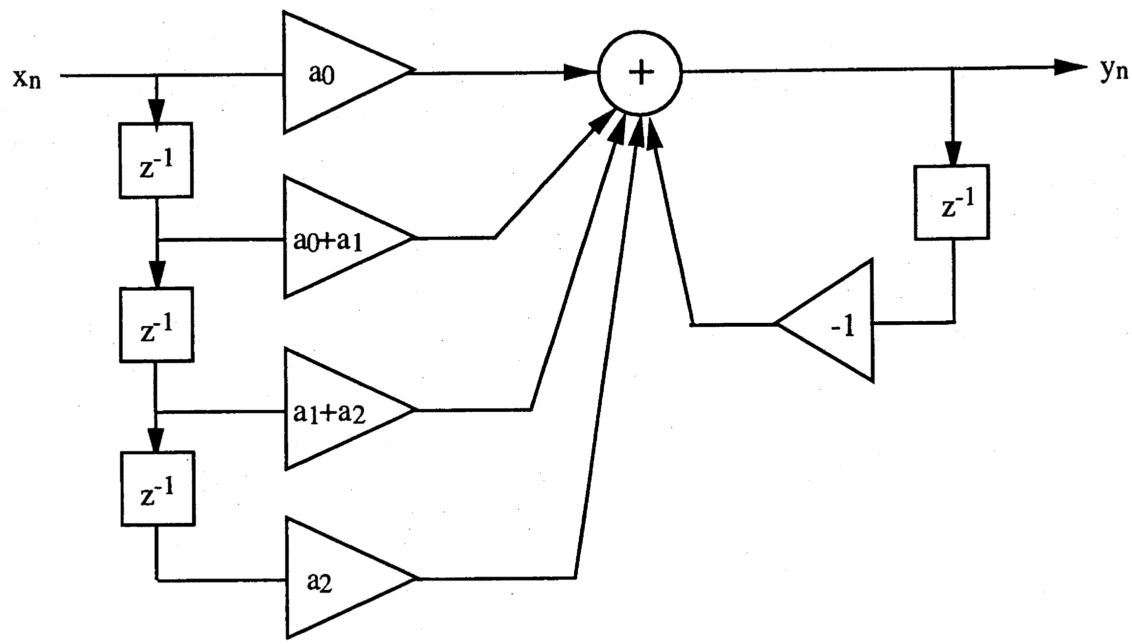
$$= \underbrace{\frac{1+z^{-1}}{1+z^{-1}}}_{\text{for example}} \left(a_0 + a_1 z^{-1} + a_2 z^{-2}\right)$$

$$= \frac{a_0 + z^{-1}(a_0 + a_1) + z^{-2}(a_1 + a_2) + a_2 z^{-3}}{1+z^{-1}}$$

$$\Rightarrow Y(z)\left[1+z^{-1}\right] = [\text{NUM}]\, X(z)$$

$$\Rightarrow Y(z) = -z^{-1} Y(z) + [\text{NUM}]\, X(z)$$

Filter structure:



This is a recursive structure (y_n depends directly on y_{n-1}) that realizes the transfer function H(z).

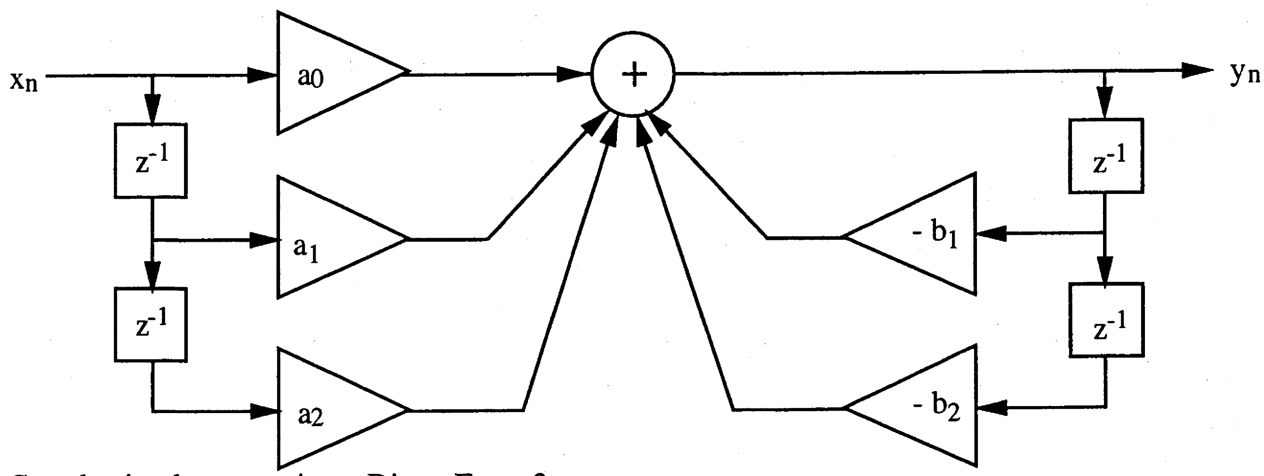
IIR Filter Structures

Transfer functions of IIR filters are not polynomials. We consider rational TF's. IIR filters must be recursive (otherwise they would require an infinite number of adders, multipliers, and delays).

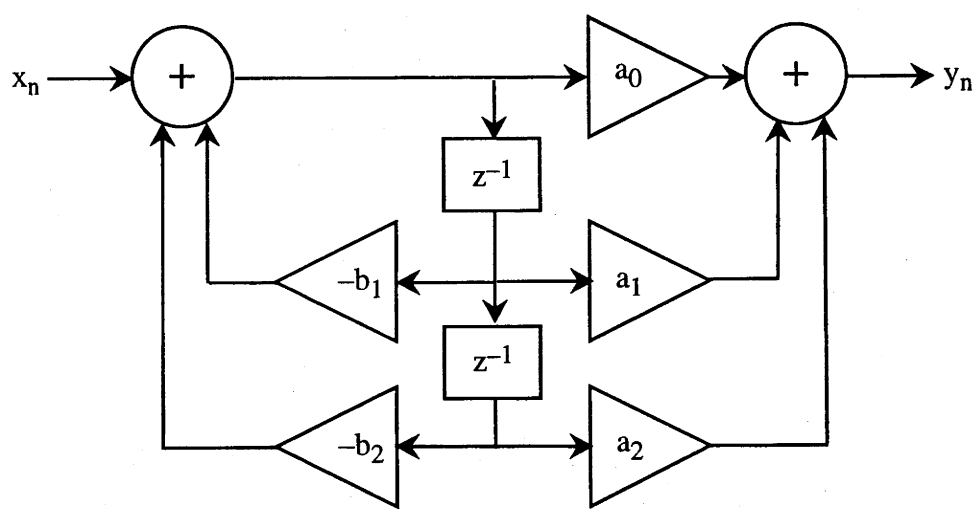
Consider a 2nd-order case:

$$H(z)= \frac{a_0 + a_1 z^{-1} + a_2 z^{-2}}{1 + b_1 z^{-1} + b_2 z^{-2}}$$

Direct Form 1 structure:



Can also implement using a Direct Form 2 structure:



We showed earlier that this structure has the same transfer function H(z) as the Direct Form 1 structure.

IIR filters are always recursive. FIR filters are implemented in nonrecursive form.

Implementation of Higher-Order Digital Filters

(Order of filter = max {degree (Num), degree (Den)} = # delays required for a Direct Form 2 implementation)

High-order direct-form filters can have large error at the output due to multiplication roundoff. Also, the actual $H_d(\omega)$ may deviate considerably from the desired due to coefficient rounding.

Cascaded or parallel second-order sections exhibit smaller error than direct form. Also, splitting into lower-order sections can make filter easier to parallelize (e.g., second-order filter on a chip).

Cascade Form:

$$H(z) = \frac{a_0 + a_1 z^{-1} + \ldots + a_N z^{-N}}{1 + b_1 z^{-1} + \ldots + b_N z^{-N}}$$

$$= \frac{a_0 z^N + a_1 z^{N-1} + \ldots + a_N}{z^N + b_1 z^{N-1} + \ldots + b_N}$$

Write as:

$$H(z) = a_0 \prod_{i=1}^{N} \frac{z - z_i}{z - p_i}$$

Since a_i, b_i are real, if p_i is complex, there must be some $p_k = p_i^*$.

Pair up poles and zeros so that (assume N is even)

$$H(z) = a_0 \prod_{i=1}^{N/2} H_i(z)$$

where

$$H_i(z) = \frac{(z - z_k)(z - z_\ell)}{(z - p_m)(z - p_n)}$$

is a second-order filter section with $z_\ell = z_k^*$ and $p_n = p_m^*$.

Pair up complex conjugates so that all multiplier coefficients of the second-order sections are <u>real</u>, so filter can use real arithmetic. (If z_k is real, then pair up with any real z_ℓ; similarly for poles.)

For instance, suppose you factor H(z) and find two poles are

$$p_m = 1 + j, \; p_n = 1 - j$$

Then pair these poles together in the same $H_i(z)$ so that:

$$H_i(z) = \frac{(z - z_k)(z - z_\ell)}{[z - (1 + j)][z - (1 - j)]}$$

$$= \frac{(z - z_k)(z - z_\ell)}{z^2 - z - jz - z + jz + (1 + j)(1 - j)}$$

$$= \frac{(z - z_k)(z - z_\ell)}{z^2 - 2z + 2}$$

real filter coeffs.

H(z) implemented in cascade form looks like:

$$X(z) \longrightarrow \boxed{a_0 H_1(z)} \longrightarrow \boxed{H_2(z)} \longrightarrow \cdots \longrightarrow \boxed{H_{\frac{N}{2}}(z)} \longrightarrow Y(z)$$

where each $H_i(z)$ is a second-order section. If N is odd, then one of the above sections will be a first-order filter.

Parallel Form:

Expand H(z) in a PFE:

$$\frac{H(z)}{z} = \frac{A}{z} + \frac{B_1}{z - p_1} + \ldots + \frac{B_N}{z - p_N}$$

$$\Rightarrow \qquad H(z) = A + \frac{B_1 z}{z - p_1} + \ldots + \frac{B_N z}{z - p_N}$$

Again, pair up complex poles.

If $p_k = p_\ell^*$ then know that $B_k = B_\ell^*$ so that

$$\frac{B_k z}{z - p_k} + \frac{B_\ell z}{z - p_\ell} = \frac{B_\ell^* z}{z - p_\ell^*} + \frac{B_\ell z}{z - p_\ell}$$

$$= \frac{B_\ell^* z(z - p_\ell) + B_\ell z(z - p_\ell^*)}{\left(z - p_\ell^*\right)(z - p_\ell)}$$

$$= \frac{z^2\left(B_\ell^* + B_\ell\right) - z\left(B_\ell^* p_\ell + B_\ell p_\ell^*\right)}{z^2 - z\left(p_\ell^* + p_\ell\right) + p_\ell p_\ell^*}$$

coeffs. are all <u>real</u>

Parallel realization (assume N is even):

$$H(z) = A + \sum_{i=1}^{N/2} H_i(z)$$

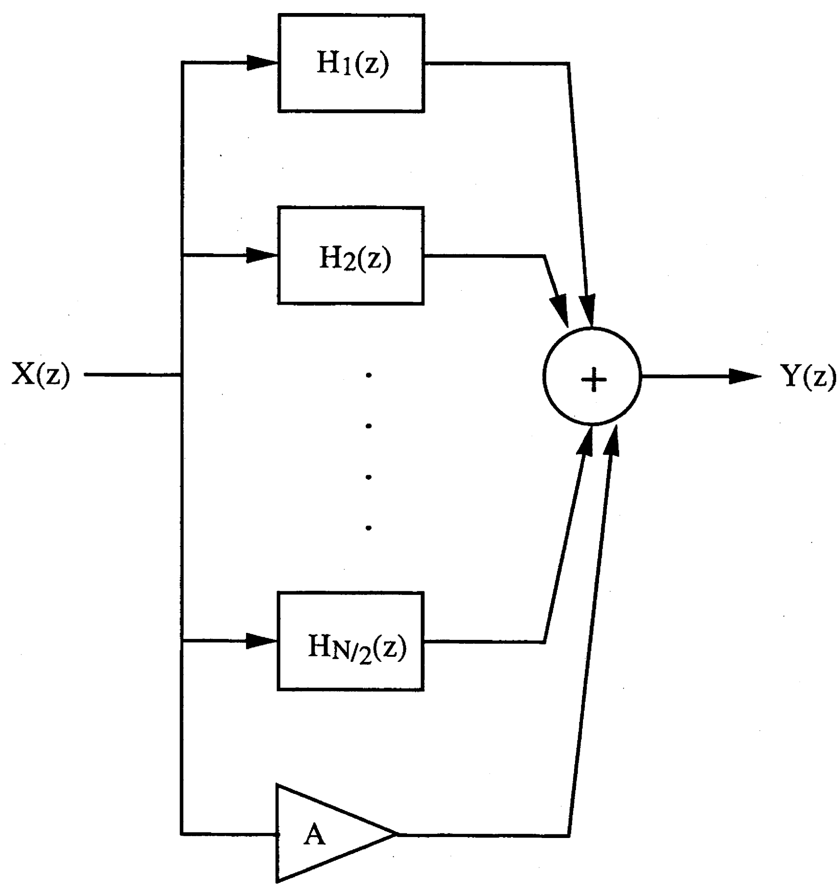
where

$$H_i(z) = \frac{a_{1i} z^2 + a_{2i} z}{z^2 + b_{1i} z + b_{2i}}$$

$$= \frac{a_{1i} + a_{2i} z^{-1}}{1 + b_{1i} z^{-1} + b_{2i} z^{-2}}$$

are second-order sections. Note that, due to the form of the numerator, each of these second-order sections requires one fewer multiplication than for cascade form.

Implementation:



If N is odd, then one of the above filter sections will be first-order.

Example

Suppose $\quad H(z) = \dfrac{z^4 + 1}{z^4 - \dfrac{1}{16}}$.

a) Draw a cascade structure of two second-order sections implementing H(z).

b) Repeat a), but for parallel form.

Solution

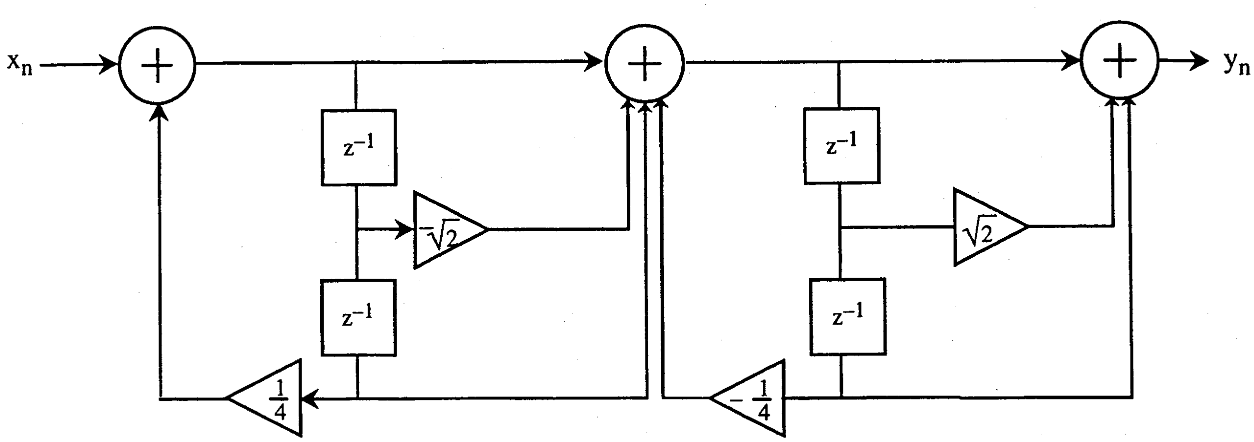
a) $$H(z) = \frac{\left(z - e^{j\frac{\pi}{4}}\right)\left(z - e^{-j\frac{\pi}{4}}\right)\left(z + e^{j\frac{\pi}{4}}\right)\left(z + e^{-j\frac{\pi}{4}}\right)}{\left(z - \frac{1}{2}\right)\left(z + \frac{1}{2}\right)\left(z - \frac{j}{2}\right)\left(z + \frac{j}{2}\right)}$$

$$= \frac{z^2 - \left(2\cos\frac{\pi}{4}\right)z + 1}{z^2 - \frac{1}{4}} \quad \frac{z^2 + \left(2\cos\frac{\pi}{4}\right)z + 1}{z^2 + \frac{1}{4}}$$

$$= \frac{z^2 - \sqrt{2}\,z + 1}{z^2 - \frac{1}{4}} \quad \frac{z^2 + \sqrt{2}\,z + 1}{z^2 + \frac{1}{4}}$$

$$= \frac{1 - \sqrt{2}\,z^{-1} + z^{-2}}{1 - \frac{1}{4}z^{-2}} \quad \frac{1 + \sqrt{2}\,z^{-1} + z^{-2}}{1 + \frac{1}{4}z^{-2}}$$

Using direct-form-2 second-order sections, the cascade structure is



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

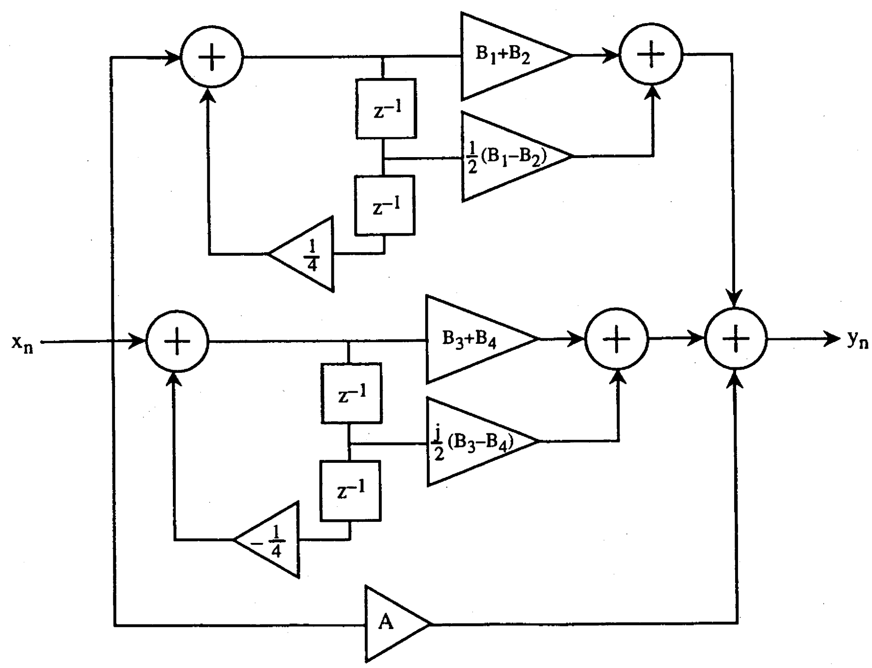
$$= \frac{A}{z} + \frac{B_1}{z-\frac{1}{2}} + \frac{B_2}{z+\frac{1}{2}} + \frac{B_3}{z-\frac{j}{2}} + \frac{B_4}{z+\frac{j}{2}}$$

$$\Rightarrow H(z) = A + \frac{B_1\,z}{z-\frac{1}{2}} + \frac{B_2\,z}{z+\frac{1}{2}} + \frac{B_3\,z}{z-\frac{j}{2}} + \frac{B_4\,z}{z+\frac{j}{2}}$$

$$= A + \frac{(B_1+B_2)\,z^2 + \frac{1}{2}(B_1-B_2)\,z}{\left(z-\frac{1}{2}\right)\left(z+\frac{1}{2}\right)} + \frac{(B_3+B_4)\,z^2 + \frac{j}{2}(B_3-B_4)\,z}{\left(z-\frac{j}{2}\right)\left(z+\frac{j}{2}\right)}$$

$$= A + \frac{(B_1+B_2) + \frac{1}{2}(B_1-B_2)\,z^{-1}}{1-\frac{1}{4}z^{-2}} + \frac{(B_3+B_4) + \frac{j}{2}(B_3-B_4)\,z^{-1}}{1+\frac{1}{4}z^{-2}}$$

Using direct-form-2 second-order sections, the parallel structure is



Note: In this diagram, A and the B_i are the coefficients in the PFE. B_4 is the complex conjugate of B_3, so all multiplier values are real. Both the cascade and parallel structures will implement the original transfer function H(z), and will generally do so with less error due to finite register length than a 4th-order direct-form implementation.

ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 34

D. Munson

Linear Versus Generalized Linear Phase

Will say $H_d(\omega)$ is linear phase if

$$H_d(\omega) = \underbrace{|H_d(\omega)|}_{\text{nonnegative}}\, e^{-j\omega M}$$

Fact:

A digital filter doesn't usually have exactly linear phase. But is easy to design FIR filters having what we will call generalized linear phase. Are two types.

Type 1:

$$H_d(\omega) = \underset{\substack{\uparrow \\ \text{real, but not nonnegative}}}{R(\omega)}\, e^{-j\omega M}$$

Type 2:

$$H_d(\omega) = R(\omega)\, e^{j(\alpha - \omega M)}$$

with $\alpha \neq 0$.

We will see that generalized linear phase corresponds to having linear phase over the passband.

FIR Versus IIR Filters

Advantage of FIR: Easy to design with generalized linear phase (linear phase over passband).

Advantage of IIR: Can't have exactly linear phase or generalized linear phase, but IIR can often meet $|H_d(\omega)|$ specification with a much lower order filter.

Generalized Linear Phase Property of FIR Filters

Type 1 Generalized Linear Phase

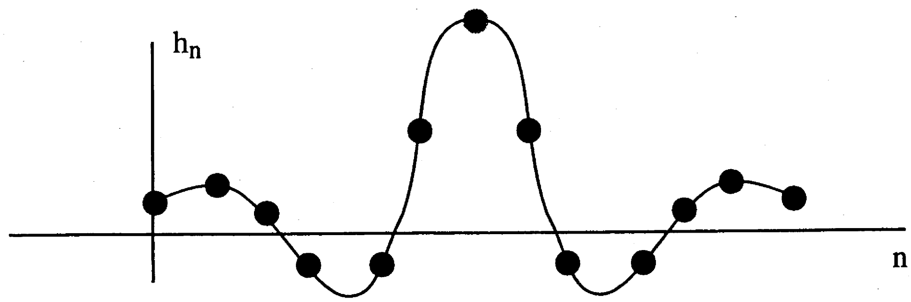
Theorem:

An FIR filter with real-valued unit pulse response $\{h_n\}_{n=0}^{N-1}$ has Type 1 generalized linear phase with $H_d(\omega) = R(\omega)\, e^{-j\omega M}$ iff

$$h_n = h_{N-1-n} \begin{cases} n = 0, 1, \ldots, \dfrac{N}{2} - 1 & \text{(N even)} \\[2ex] n = 0, 1, \ldots, \dfrac{N-1}{2} & \text{(N odd)} \end{cases}$$

where $M = \dfrac{N-1}{2}$ and $R(\omega)$ is real and even.

Picture:



Proof:

We give the proof in one direction only. Assuming filter coefficients with even symmetry, we show that $H_d(\omega)$ has the stated form. Now, assume N is odd. Let $M = \dfrac{N-1}{2}$. Given $h_n = h_{N-1-n}$, show that $H_d(\omega)$ has Type 1 generalized linear phase.

$$H_d(\omega) = \sum_{n=0}^{N-1} h_n e^{-j\omega n}$$

$$= \sum_{n=0}^{M-1} h_n e^{-j\omega n} + h_M e^{-j\omega M} + \sum_{n=M+1}^{N-1} h_n e^{-j\omega n}$$

$$= e^{-j\omega M} \left[h_M + \sum_{n=0}^{M-1} h_n e^{-j\omega(n-M)} + \sum_{n=M+1}^{N-1} h_n e^{-j\omega(n-M)} \right] \qquad (\square)$$

Now, making the change of variable $n = N{-}1{-}k$ and using $M = \dfrac{N-1}{2}$, the second sum in $(\square)$ can be written as

$$\sum_{k=M-1}^{0} h_{N-1-k} e^{-j\omega(M-k)}$$

Thus,

$$H_d(\omega) = e^{-j\omega M}\left[h_M + \sum_{n=0}^{M-1}\left(h_n\, e^{-j\omega(n-M)} + h_{N-1-n}\, e^{-j\omega(M-n)}\right)\right] \qquad (\square\square)$$

and using $h_{N-1-n} = h_n$, we have

$$H_d(\omega) = e^{-j\omega M}\underbrace{\left[h_M + 2\sum_{n=0}^{M-1} h_n \cos\omega(n-M)\right]}_{\stackrel{\Delta}{=} R(\omega) \,\sim\, \text{real valued}}$$

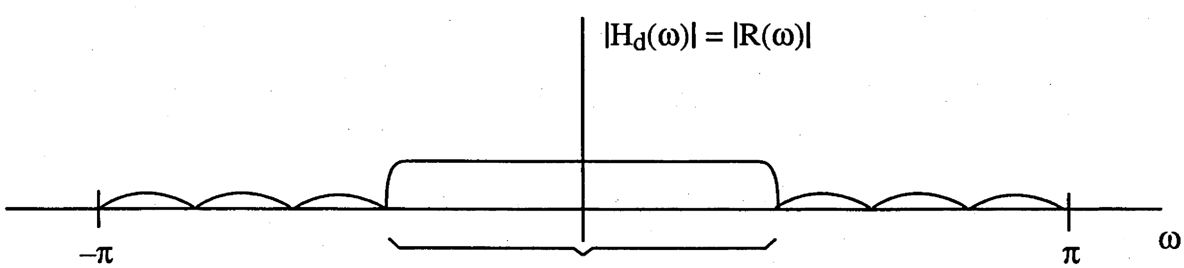
Note:

$$|H_d(\omega)| = |R(\omega)|$$

$$\angle H_d(\omega) = \begin{cases} -\omega M & \{\omega : R(\omega) > 0\} \\ \underset{\substack{\uparrow \\ -1 = e^{\pm j\pi}}}{-\omega M \pm \pi} & \{\omega : R(\omega) < 0\} \end{cases} \qquad (\Delta)$$

$(\Delta) \Rightarrow$ phase is linear except where $R(\omega)$ changes sign, in which case the phase jumps by π.

This implies that a generalized linear-phase filter has linear phase over the passband since



R(ω) can't change sign in passband since $|H_d(\omega)| = |R(\omega)|$ $\neq 0$ in passband

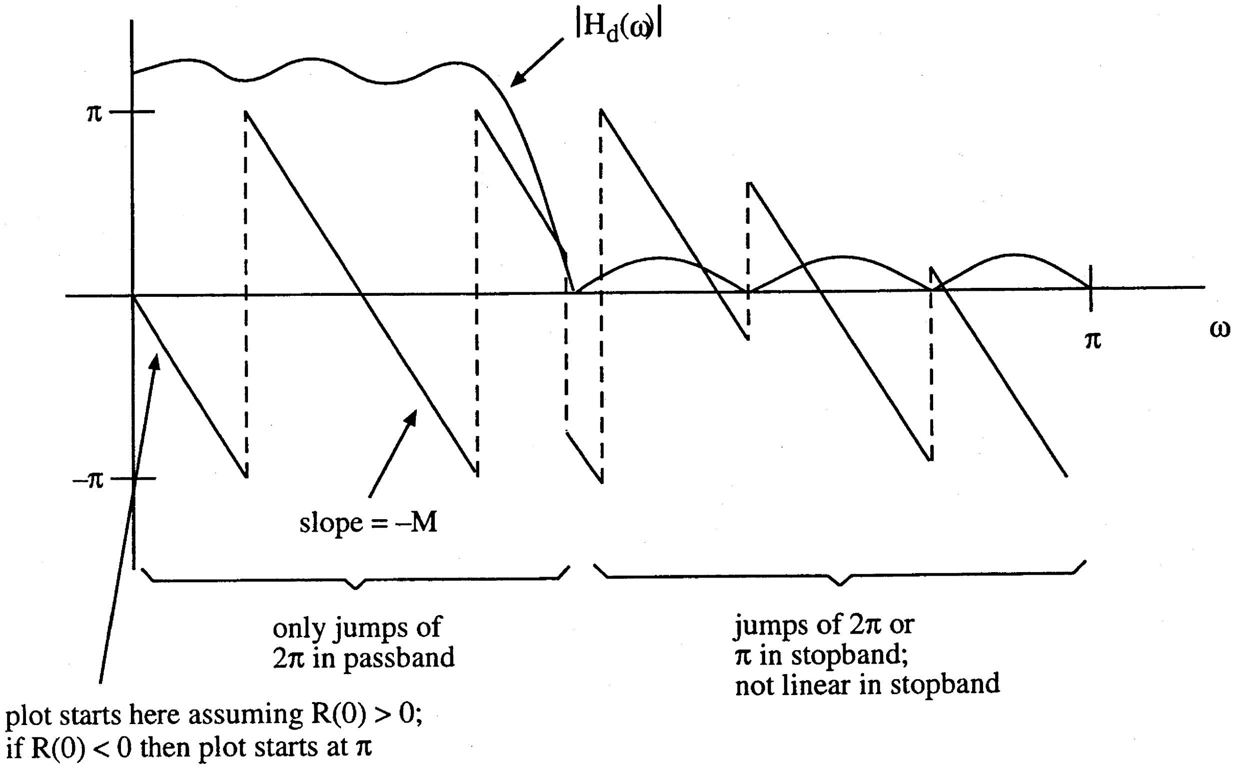
To prove the theorem in the other direction, start with

$$H_d(\omega) = \underset{\substack{\uparrow \\ \text{real and even}}}{R(\omega)}\, e^{-j\omega M}.$$

Then can show $h_n = h_{N-1-n}$.

Note: If N is even then we still take $M = \frac{N-1}{2}$ and the proof is nearly the same as above.

Phase characteristic of a generalized linear phase FIR filter:



In the next lecture, we will consider Type 2 generalized linear phase:

$$H_d(\omega) = \underset{\substack{\uparrow \\ \text{real, odd}}}{R(\omega)}\, e^{j(\alpha - \omega M)}$$

with $\alpha \neq 0$.

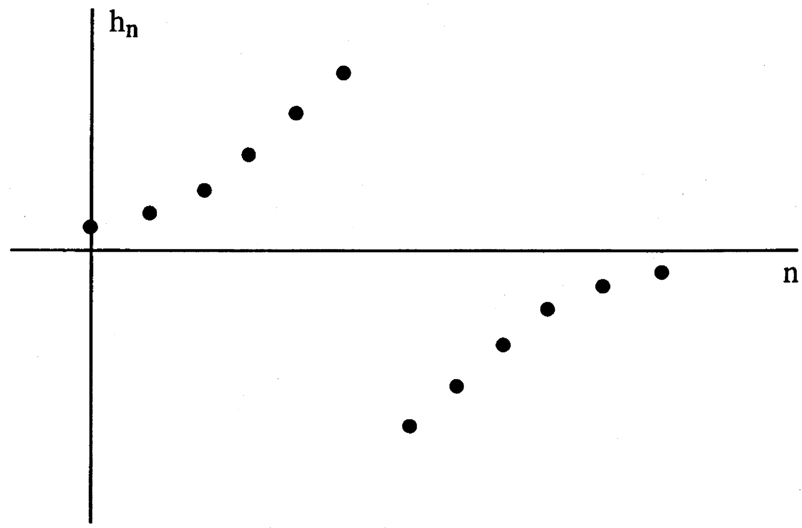
In this case, for a filter $\{h_n\}_{n=0}^{N-1}$ and $\alpha \neq 0$, can show must have $\alpha = \frac{\pi}{2}$ and have odd coefficient symmetry, i.e.,

$$h(n) = -h(N-1-n)$$

and
$$\angle H_d(\omega) = \begin{cases} \frac{\pi}{2} - \omega M & \{\omega : R(\omega) > 0\} \\ \frac{-\pi}{2} - \omega M & \{\omega : R(\omega) < 0\} \end{cases}$$

with $M = \frac{N-1}{2}$ for N even and N odd.

Picture:



ECE 310 **DIGITAL SIGNAL PROCESSING** **D. Munson**
University of Illinois **Lecture 35**

Type 2 Generalized Linear Phase

This type of generalized linear phase corresponds to antisymmetric (odd), rather than symmetric (even) filter coefficients.

Theorem:

An FIR filter with real-valued unit pulse response $\{h_n\}_{n=0}^{N-1}$ has Type 2 generalized linear phase with $H_d(\omega) = R(\omega)\, e^{j(\frac{\pi}{2}-\omega M)}$ iff

$$h_n = -\,h_{N-1-n}$$

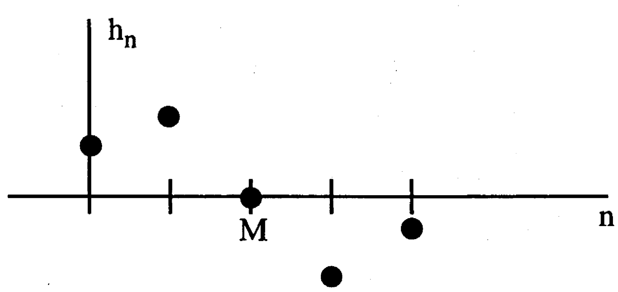
where $M = \dfrac{N-1}{2}$ and $R(\omega)$ is real and odd.

Proof:

We give the proof in one direction only. Assuming filter coefficients with odd symmetry, we show that $H_d(\omega)$ has the stated form. Now, assuming N is odd and taking $M = \dfrac{N-1}{2}$ we have from before:

$$H_d(\omega) = e^{-j\omega M}\left[h_M + \sum_{n=0}^{M-1}\left(h_n\, e^{-j\omega(n-M)} + h_{N-1-n}\, e^{-j\omega(M-n)}\right)\right] \qquad (\square\square)$$

Given that $h_n = -\,h_{N-1-n}$ we must have $h_M = 0$. Can see this pictorially:



Obviously, we cannot have odd coefficient symmetry unless $h_M = 0$.

Now, setting $h_M = 0$ and using $h_{N-1-n} = -\,h_n$ in $(\square\square)$, we have

$$H_d(\omega) = e^{-j\omega M}\left[\sum_{n=0}^{M-1} h_n\left(e^{-j\omega(n-M)} - e^{j\omega(n-M)}\right)\right]$$

$$= e^{-j\omega M}(-j2)\sum_{n=0}^{M-1} h_n \sin\omega(n-M)$$

$$= e^{j(\frac{\pi}{2}-\omega M)}\underbrace{\left(-2\sum_{n=0}^{M-1} h_n \sin\omega(n-M)\right)}_{=R(\omega),\text{ which is real and } \underline{\text{odd}}}$$

So, for the antisymmetric coefficient case, we have $|H_d(\omega)| = |R(\omega)|$, but now $R(\omega)$ is a linear combination of sines (odd) instead of cosines (even) and

$$\angle H_d(\omega) = \begin{cases} \frac{\pi}{2} - \omega M & \{\omega : R(\omega) > 0\} \\ \frac{-\pi}{2} - \omega M & \{\omega : R(\omega) < 0\} \end{cases}$$

To prove the theorem in the other direction, start with

$$H_d(\omega) = \underset{\substack{\uparrow \\ \text{real and odd}}}{R(\omega)}\, e^{j\left(\frac{\pi}{2}-\omega M\right)}$$

Then can show $h_n = -h_{N-1-n}$.

<u>Note</u>: If N is even instead of odd, we still take $M = \frac{N-1}{2}$ and the proof is nearly the same as above.

Example

Determine whether a filter with the unit-pulse response

$$\{h_n\} = \{1, \underset{\uparrow}{-1}, 1\}$$

has generalized linear phase, and if so, whether it has linear phase.

Note: Linear phase $\Rightarrow$ generalized linear phase, but not vice versa.

<u>Solution</u>

h_n is symmetric about its midpoint $\Rightarrow$ Have Type 1 generalized linear phase. To check whether we have linear phase, we must find the phase of the frequency response:

$$H_d(\omega) = 1 - e^{-j\omega} + e^{-j2\omega}$$

$$= \underset{\uparrow\, e^{-j\omega M} \text{ where } M = 1 \text{ in this example}}{e^{-j\omega}}\ (e^{j\omega} - 1 + e^{-j\omega})$$

$$= e^{-j\omega}\ \underbrace{(2\cos\omega - 1)}_{R(\omega)}$$

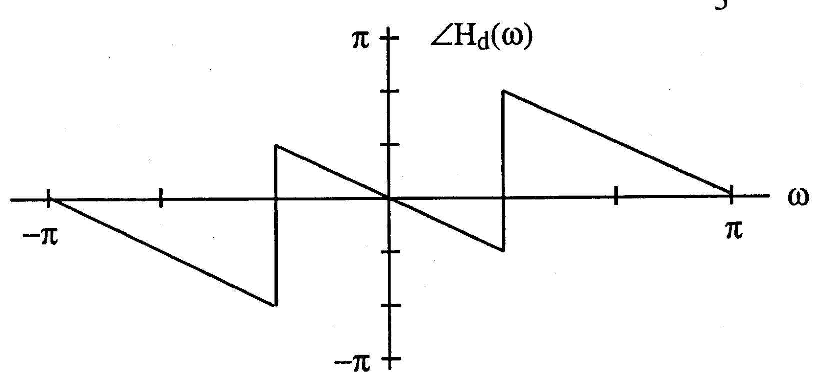
We see that $R(\omega)$ changes sign on $-\pi < \omega < \pi$. Thus, we know that this filter does <u>not</u> have linear phase. Let's find the phase:

$$\angle H_d(\omega) = \begin{cases} -\omega & \{\omega : 2\cos\omega - 1 > 0\} \\ -\omega + \pi & \{\omega : 2\cos\omega - 1 < 0\} \end{cases}$$

Thus, for $|\omega| < \pi$ we have

$$\angle H_d(\omega) = \begin{cases} -\omega & |\omega| < \frac{\pi}{3} \\ -\omega + \pi & \frac{\pi}{3} < |\omega| < \pi \end{cases}.$$

This is obviously not linear because $\angle H_d(\omega)$ takes jumps of π at $\omega = \pm\,\frac{\pi}{3}$.



Summary: In this example, $\angle H_d(\omega)$ is generalized linear phase but it is not linear phase.

Example

Changing the previous example to

$$\{h_n\} = \left\{\frac{1}{4}, -1, \frac{1}{4}\right\}$$

results in a filter that not only has generalized linear phase — it also has linear phase. Students are encouraged to work this out as an exercise.

Example

Determine whether $H_d(\omega)$ corresponding to

$$\{h_n\} = \{-1, 3, 1\}$$

has generalized linear phase.

Solution

At first it appears that this filter might have Type 2 generalized linear phase. However, this is not the case because the middle coefficient is nonzero. Let's examine $H_d(\omega)$:

$$\begin{aligned} H_d(\omega) &= -1 + 3e^{-j\omega} + e^{-j2\omega} \\ &= e^{-j\omega}\left(-e^{j\omega} + 3 + e^{-j\omega}\right) \\ &= e^{-j\omega}\left(3 - j2\sin\omega\right) \end{aligned}$$

Notice that this cannot be put in the form $e^{j(\frac{\pi}{2}-\omega M)}$ $R(\omega)$ where $R(\omega)$ is real. The nonzero middle coefficient prevents this!

Example

Determine whether $H_d(\omega)$ corresponding to

$$\{h_n\} = \{1, -1\}$$

has generalized linear phase and linear phase.

Solution

$H_d(\omega)$ has Type 2 generalized linear phase since h_n is antisymmetric. Find the phase of $H_d(\omega)$ to check whether we have linear phase:

$$\begin{aligned} H_d(\omega) &= 1 - e^{-j\omega} \\ &= e^{-j\frac{\omega}{2}}\left(e^{j\frac{\omega}{2}} - e^{-j\frac{\omega}{2}}\right) \\ &= e^{-j\frac{\omega}{2}}\, 2\,j\sin\frac{\omega}{2} \\ &= e^{j(\frac{\pi}{2}-\frac{\omega}{2})}\, \underbrace{2\sin\frac{\omega}{2}}_{R(\omega)} \end{aligned}$$

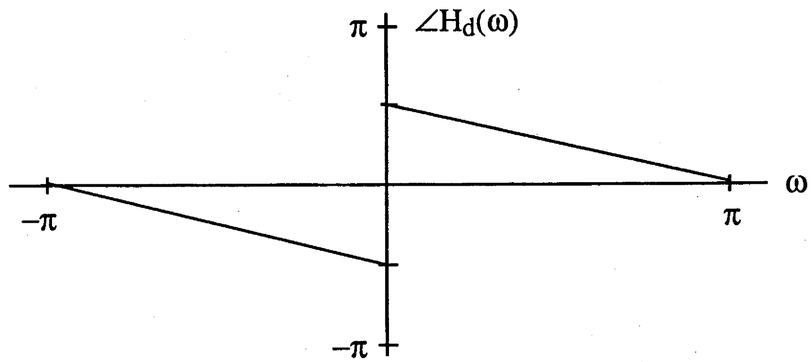
Here $R(\omega)$ is odd, so it must change sign at $\omega = 0$. This implies that we do not have linear phase. Let's find the phase:

$$\angle H_d(\omega) = \begin{cases} \frac{\pi}{2} - \frac{\omega}{2} & \left\{\omega : \sin\frac{\omega}{2} > 0\right\} \\ -\frac{\pi}{2} - \frac{\omega}{2} & \left\{\omega : \sin\frac{\omega}{2} < 0\right\} \end{cases}$$

Thus, for $|\omega| < \pi$ we have

$$\angle H_d(\omega) = \begin{cases} \frac{\pi}{2} - \frac{\omega}{2} & 0 < \omega < \pi \\ -\frac{\pi}{2} - \frac{\omega}{2} & -\pi < \omega < 0 \end{cases}$$

This is clearly not linear, as shown in the following plot.



Since $R(\omega)$ will be odd for any antisymmetric filter, we conclude that filters with antisymmetric coefficients cannot have linear phase.

Example

Given

$$h_n = \{\underset{\uparrow}{1}, 1\}$$

does $H_d(\omega)$ have generalized linear phase? How about linear phase?

Solution

Because of coefficient symmetry, $H_d(\omega)$ has Type 1 generalized linear phase. To check linear phase, look at:

$$H_d(\omega) = 1 + e^{-j\omega}$$

$$= e^{-j\frac{\omega}{2}}\left(e^{j\frac{\omega}{2}} + e^{-j\frac{\omega}{2}}\right)$$

$$= e^{-j\frac{\omega}{2}} \underbrace{2\cos\frac{\omega}{2}}_{R(\omega)}$$

Here $R(\omega)$ does not change sign on $-\pi < \omega < \pi$ and we have

$$\angle H_d(\omega) = -\frac{\omega}{2} \qquad |\omega| \leq \pi$$

$\Rightarrow$ Strictly linear phase.

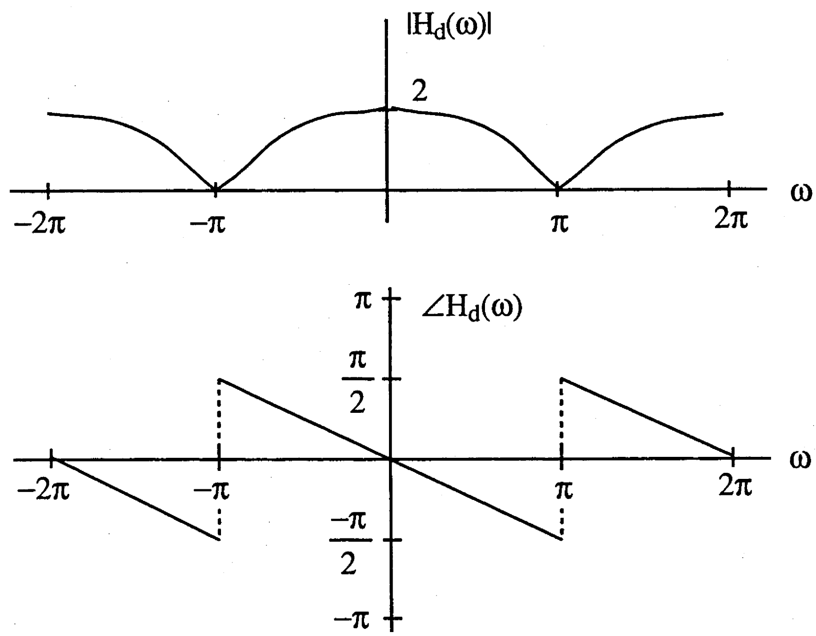
Of course, $\angle H_d(\omega)$ is periodic outside $|\omega| < \pi$.

We have:

$$|H_d(\omega)| = 2\cos\frac{\omega}{2} \qquad |\omega| \leq \pi$$

$$\angle H_d(\omega) = -\frac{\omega}{2} \qquad |\omega| \leq \pi$$

So:



Here we do have jumps of π at ω = odd multiples of π, but we will still call this linear phase.

Impact of Coefficient Symmetry on Realizable Frequency Responses

Depending on whether $\{h_n\}_{n=0}^{N-1}$ are symmetric or antisymmetric, and N is even or odd, there can be restrictions on the types of filters that can be realized.

Example

If N is even (number of coefficients is even) and $\{h_n\}_{n=0}^{N-1}$ are symmetric, then you can't realize a high-pass filter! Why not? Because, for this case $H_d(\pi) = 0$, so that $\omega = \pi$ can't be in the passband for this type of filter. Let's show this.

For N even and h_n symmetric, we have

$$H(z) \;=\; h_0 + h_1 z^{-1} + h_2 z^{-2} + \ldots + h_2 z^{-(N-3)} + h_1 z^{-(N-2)} + h_0 z^{-(N-1)}.$$

Then

$$H_d(\pi) \;=\; H(-1) = h_0 - h_1 + h_2 - \ldots - h_2 + h_1 - h_0 = 0$$

In practice, it pays to be aware of these types of constraints, but the problem is easily resolved. For example, in designing a high-pass filter with symmetric coefficients, we would simply take N to be odd.

Let's now address this problem in more generality by considering some short FIR filters to see what restrictions exist on $H_d(0)$ and $H_d(\pi)$ as a function of coefficient symmetry and the value of N.

$$H_d(\omega) \;=\; a_0 + a_1 e^{-j\omega} + a_1 e^{-j2\omega} + a_0 e^{-j3\omega} \qquad \text{(even symmetry, N even)}$$

$$\Rightarrow \begin{cases} H_d(0) = 2a_0 + 2a_1 \\ H_d(\pi) = 0 \end{cases}$$

$$H_d(\omega) \;=\; a_0 + a_1 e^{-j\omega} + a_2 e^{-j2\omega} + a_1 e^{-j3\omega} + a_0 e^{-j4\omega} \quad \text{(even symmetry, N odd)}$$

$$\Rightarrow \begin{cases} H_d(0) = 2a_0 + 2a_1 + a_2 \\ H_d(\pi) = 2a_0 - 2a_1 + a_2 \end{cases}$$

$$H_d(\omega) \;=\; a_0 + a_1 e^{-j\omega} - a_1 e^{-j2\omega} - a_0 e^{-j3\omega} \qquad \text{(odd symmetry, N even)}$$

$$\Rightarrow \begin{cases} H_d(0) = 0 \\ H_d(\pi) = 2a_0 - 2a_1 \end{cases}$$

$$H_d(\omega) = a_0 + a_1 e^{-j\omega} + 0\, e^{-j2\omega} - a_1 e^{-j3\omega} - a_0 e^{-j4\omega} \quad \text{(odd symmetry, N odd)}$$

$$\Rightarrow \begin{cases} H_d(0) = 0 \\ H_d(\pi) = 0 \end{cases}$$

In general, we conclude

Symmetry	**N**	**Unrealizable Filters**
even	even	high-pass, bandstop
even	odd	no restriction
odd	even	low-pass, bandstop
odd	odd	low-pass, high-pass, bandstop

Notice that a bandstop filter has its stopband located between 0 and π and therefore has passbands centered at both $\omega = 0$ and $\omega = \pi$. Thus, if either a lowpass or highpass filter cannot be realized, this implies that a bandstop filter cannot be realized.

ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 36

D. Munson

DIGITAL FILTER DESIGN

FIR Design Methods

1) Windowing

   a) Truncation
   b) General windowing

2) Frequency Sampling ~ will cover later in connection with the DFT

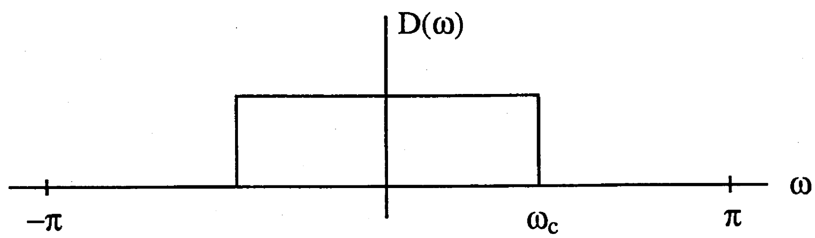
3) Computer-Aided Optimization

   a) Parks - McClellan (√) ~ widely used
   b) Linear Programming

1a) Truncation

Illustrate by example.

Suppose we want a LPF with frequency response:

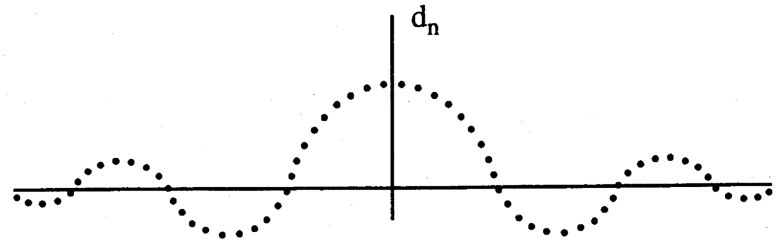


We might choose the filter coefficients $\{h_n\}$ to be

$$d_n = \frac{1}{2\pi}\int_{-\pi}^{\pi} D(\omega)\, e^{j\omega n}\, d\omega \;=\; \frac{1}{2\pi}\int_{-\omega_c}^{\omega_c} 1\, e^{j\omega n}\, d\omega = \frac{\omega_c}{\pi}\,\mathrm{sinc}\,\omega_c n$$

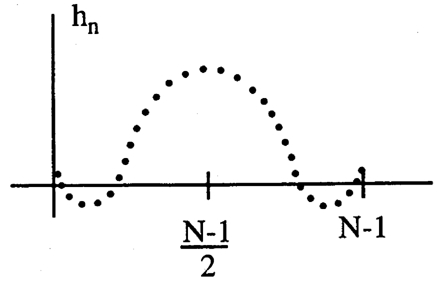
(Then frequency response $H_d(\omega) = \text{DTFT}\left[\{d_n\}\right] = D(\omega)$ exactly.)

Look at the d_n:



This sequence is infinite in length and noncausal. So, let's choose (assume N odd, for now):

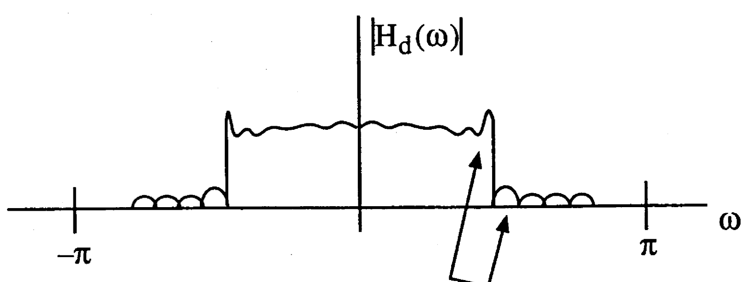
$$h_n = \begin{cases} d_{n-\frac{N-1}{2}} & 0 \le n \le N-1 \\ 0 & \text{else} \end{cases}$$



If N is large, then h_n consists of all the larger values of d_n and

$$H_d(\omega) \approx D(\omega)\, e^{-j\omega\frac{N-1}{2}}\ .$$

<u>But</u> this approximation is poor in the following sense. The frequency response will look like:



The ripples are due to the Gibbs' phenomenon. Get tall ripples at sharp transitions. As N increases, ripples become narrower and more numerous, but the heights of the ripples nearest the discontinuity remain large. We will soon develop our own explanation for why this occurs. For now, it is worth noting that the DTFT

$$\sum_{n=-\infty}^{\infty} d_n\, e^{-j\omega n}$$

is actually a Fourier series expansion of $D(\omega)$. We had not thought of the DTFT in this way before, because there was no advantage in doing so. Here, though, we note that $H_d(\omega)$ is obtained by <u>truncating</u> this Fourier series (i.e., choosing $h_n = d_{n-\frac{N-1}{2}}$ for $0 \le n \le N-1$.)

Gibbs and other mathematicians studied truncation of Fourier series and showed that ripples will occur around locations where the periodic function is discontinuous. These ripples can be made narrower and to bunch up around the points of discontinuity by taking N larger. However, a larger N does not reduce the heights of the ripples!

How can we reduce the ripple heights?

1b) General Windowing

Given $D(\omega)$ and the corresponding infinite-length $\{d_n\}$, choose the coefficients $\{h_n\}_{n=0}^{N-1}$ to be (again, assume N odd for now):

$$h_n = w_n \, d_{n-\frac{N-1}{2}} \qquad 0 \le n \le N-1$$

where $w_n = 0$, $n \notin [0, N-1]$, is a <u>window</u> sequence that gently tapers to zero.

For truncation we used

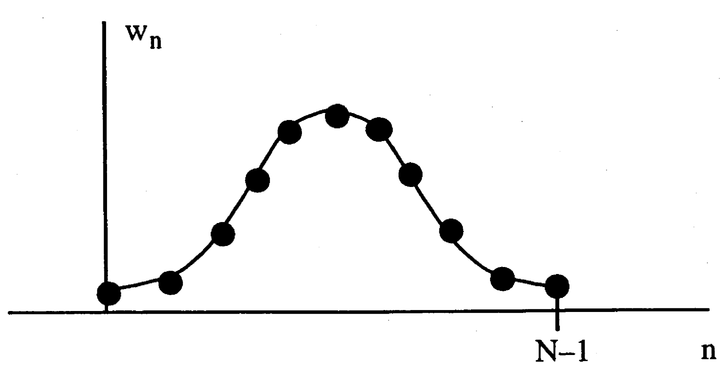
$$w_n = \begin{cases} 1 & 0 \le n \le N-1 \\ 0 & \text{else} \end{cases}$$

which does *not* taper gently to zero. Smoother windows can lead to <u>much lower ripple</u>.

An example of a good window is the Hamming window:

$$w_n = .54 - .46 \cos\left(\frac{2\pi n}{N-1}\right), \quad 0 \le n \le N-1.$$

This is plotted below



How does the window choice affect the frequency response of the designed filter $\{h_n\}_{n=0}^{N-1}$?

Let

$$g_n = d_{n-\frac{N-1}{2}} \, .$$

Then:

$$h_n = w_n \, g_n$$

desired u.p.r. after shifting

window

coeffs. of designed FIR filter

In the frequency domain this corresponds to:

$$\begin{aligned}
H_d(\omega) &= \sum_n w_n \, g_n \, e^{-j\omega n} \\
&= \sum_n w_n \frac{1}{2\pi} \int_{-\pi}^{\pi} G_d(\theta) e^{j\theta n} \, d\theta \; e^{-j\omega n} \\
&= \frac{1}{2\pi} \int_{-\pi}^{\pi} G_d(\theta) \sum_n w_n \, e^{-jn(\omega-\theta)} \, d\theta \\
&= \frac{1}{2\pi} \int_{-\pi}^{\pi} G_d(\theta) \, W_d(\omega-\theta) \, d\theta \qquad (*)
\end{aligned}$$

Notice that this has the form of a convolution. Since the integrand is periodic and we integrate over only a single period, this is called a periodic convolution.

Since our goal is to have $H_d(\omega) \approx G_d(\omega)$, we see from $(*)$ that we would like

$$W_d(\omega) = 2\pi \, \delta(\omega)$$

But, $W_d(\omega) = 2\pi \, \delta(\omega) \Rightarrow$

$$w_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} 2\pi \, \delta(\omega) \, e^{j\omega n} \, d\omega = 1 \qquad \text{for all } n$$

$\Rightarrow w_n$ is not a window sequence!

w_n nonzero only on $n \in [0, N-1]$ results in (at best) either:

i) $W_d(\omega)$ is a narrow pulse around $\omega = 0$, but has high sidelobes.

or

ii) $W_d(\omega)$ is wider around the origin but the sidelobes are lower.

So, we have a tradeoff.

From (*) we see that the ripple in $H_d(\omega)$ is caused by integrating the product of G_d and W_d as the ripples in W_d are shifted across the discontinuity of G_d. Likewise, the width of the transition band for H_d will depend on the width of the mainlobe (center lobe) of W_d. We conclude:

High sidelobes of $W_d \Rightarrow$ Large ripple in H_d.
Wide center lobe of $W_d \Rightarrow$ Wide transition band in H_d.

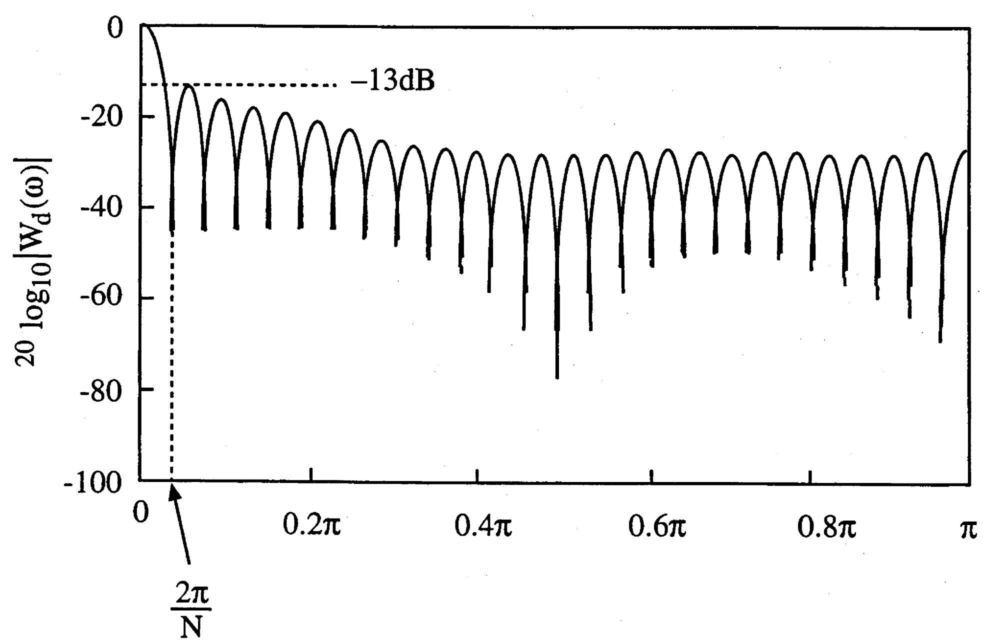
Our goal is to achieve moderately low sidelobes and a moderately narrow transition width in W_d simultaneously.

What does $W_d(\omega)$ look like for some common windows?

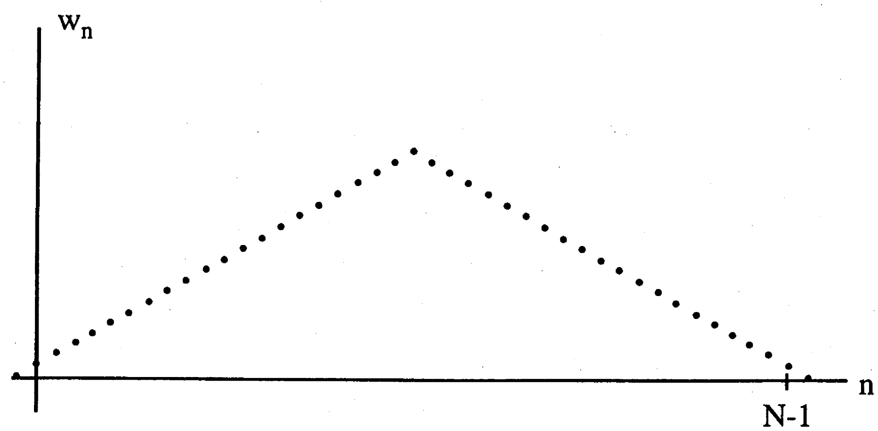
Truncation

$$|W_d(\omega)| = \left|\sum_{n=0}^{N-1} 1\, e^{-j\omega n}\right| = \left|\frac{\sin \frac{N}{2}\omega}{\sin \frac{1}{2}\omega}\right|$$

Plot on log scale:

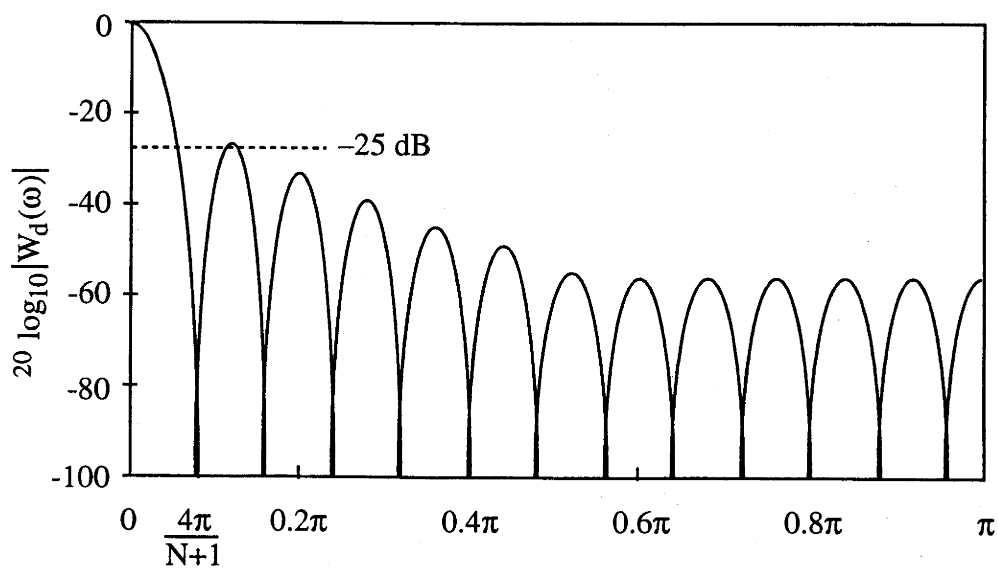


Triangular (Bartlett):



$$|W_d(\omega)| = \left| \frac{\sin \frac{N+1}{4} \omega}{\sin \frac{1}{2} \omega} \right|^2$$

Plot on log scale:

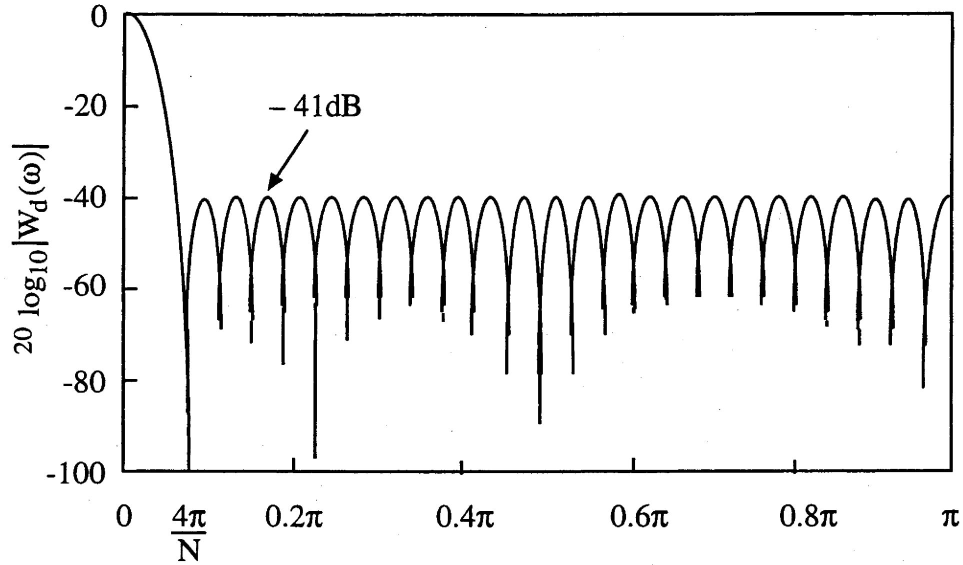


From this plot, we see that the triangular window has a mainlobe that is twice as wide as that of the rectangular window, but the highest sidelobe is reduced by 12 db.

Hamming:

$$w_n = .54 - .46 \cos\left(\frac{2\pi n}{N-1}\right) \quad 0 \le n \le N-1$$

Plot of $|W_d(\omega)|$ on a log scale:



From this plot, we see that the Hamming window has essentially the same mainlobe width as the triangular window, and sidelobes that are reduced an additional 16 dB to –41 dB. Thus, the Hamming window is preferred over the triangular window. Comparing the Hamming window to the truncation window, we see that the highest sidelobe is reduced by 28 dB (more than a factor of 10) at the expense of increasing the mainlobe width by a factor of 2.

Best window: <u>Kaiser</u>

$$w_n = I_o\left[\beta\left(1-\left[\left(n-\frac{N-1}{2}\right)\Big/\frac{N-1}{2}\right]^2\right)^{1/2}\right] \qquad 0 \le n \le N-1$$

I_o is the zeroth-order modified Bessel function of the first kind:

$$I_o(x) = \frac{1}{\pi}\int_0^{\pi} e^{\pm x\cos\theta}\, d\theta = \frac{1}{\pi}\int_0^{\pi} \cosh(x\cos\theta)\, d\theta$$

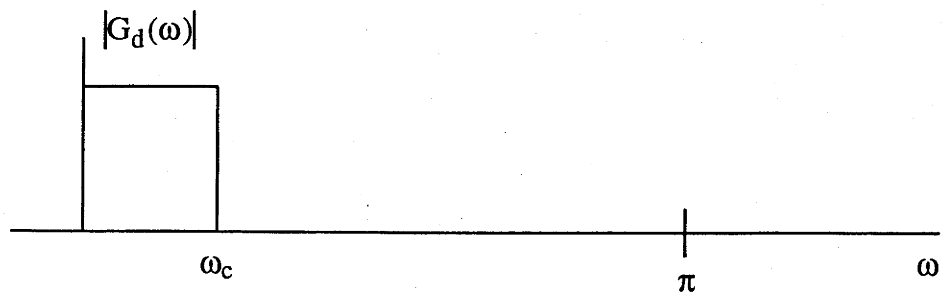
The choice of β affects the tradeoff between the mainlobe width and sidelobe heights. β is user specified.

The Kaiser window can achieve slightly narrower mainlobe with the same sidelobe height as Hamming window.

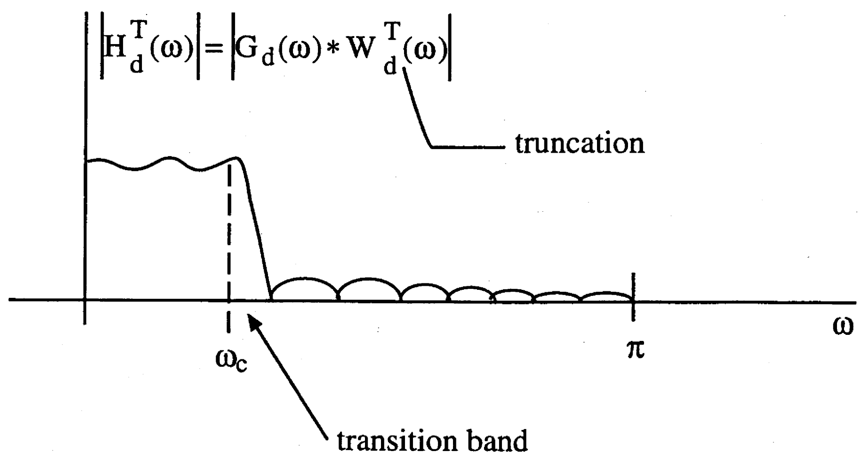
ECE 310 **DIGITAL SIGNAL PROCESSING** **D. Munson**
University of Illinois **Lecture 37**

Typical Frequency Responses Using Window Design

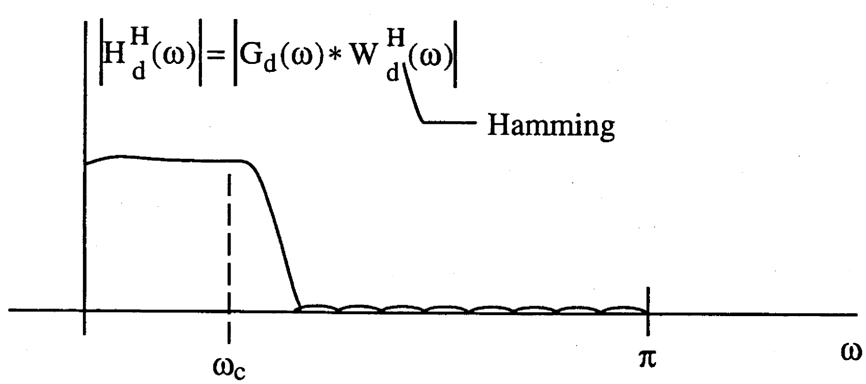
If the desired magnitude response is



then the frequency response of a truncation design may look like:



and the frequency response of a Hamming design will look like:



So, Hamming widens the transition band by a factor of two, but greatly reduces ripple.

Note: The actual filter design procedure is

$$h_n = w_n\, g_n \qquad 0 \le n \le N-1.$$

The above Fourier-domain concepts are to help us visualize the resulting $H_d(\omega)$.

Now, so far we have considered only the case with N odd, where we defined

$$g_n = d_{n-\frac{N-1}{2}}$$

where $D(\omega)$ is the desired $H_d(\omega)$.

How do we find g_n if N is even? Answer: Select g_n as shown in the following procedure, which works for both N odd and N even.

General Window Design Procedure

To design a generalized linear phase $\{h_n\}_{n=0}^{N-1}$ with $|H_d(\omega)| \approx D(\omega)$ do this:

1) Let $G_d(\omega) = D(\omega)\, e^{-j\frac{N-1}{2}\omega}$

2) Find $g_n = \text{DTFT}^{-1}\left[G_d(\omega)\right]$

3) Let $h_n = w_n\, g_n$.

Notes:

1) For N odd this procedure gives $g_n = d_{n-\frac{N-1}{2}}$ as before. For N even, steps 1) and 2) give $\{g_n\}$ as an interpolated set of values lying between $\left\{d_{n-\frac{N}{2}}\right\}$ and $\left\{d_{n-\frac{N-2}{2}}\right\}$.

2) We wish to know whether $H_d(\omega)$, designed via the window method, will have generalized linear phase. The answer is ordinarily yes, since $\{h_n\}$ will be symmetric or antisymmetric if $\{w_n\}$ is symmetric and $\{g_n\}$ is either symmetric or antisymmetric.

FIR Window Design Examples

Example

Design generalized linear-phase, low-pass FIR filters having coefficients $\{h_n\}_{n=0}^{29}$ and cutoff $\omega_c = \frac{\pi}{4}$ using the window design procedure with both truncation and Hamming windows.

Solution

$$G_d(\omega) = D(\omega)\,e^{-j\frac{N-1}{2}\omega}$$

$$= \begin{cases} e^{-j\frac{29}{2}\omega} & |\omega| \le \frac{\pi}{4} \\ 0 & \frac{\pi}{4} < |\omega| \le \pi \end{cases}$$

$$\Rightarrow g_n = \frac{1}{2\pi} \int_{-\frac{\pi}{4}}^{\frac{\pi}{4}} e^{-j\frac{29}{2}\omega}\, e^{j\omega n}\, d\omega$$

$$= \frac{1}{2\pi} \left. \frac{e^{j\omega\left(n-\frac{29}{2}\right)}}{j\left(n-\frac{29}{2}\right)} \right|_{-\frac{\pi}{4}}^{\frac{\pi}{4}}$$

$$= \frac{\sin\frac{\pi}{4}\left(n-\frac{29}{2}\right)}{\pi\left(n-\frac{29}{2}\right)}$$

$$= \frac{1}{4}\,\mathrm{sinc}\left[\frac{\pi}{4}\left(n-\frac{29}{2}\right)\right]$$

Now, for truncation $h_n = w_n\, g_n$ with $w_n = \begin{cases} 1 & 0 \le n \le 29 \\ 0 & \text{else} \end{cases}$.

So:

$$\boxed{h_n = \frac{1}{4}\mathrm{sinc}\,\frac{\pi}{4}\left(n-\frac{29}{2}\right) \quad 0 \le n \le 29}$$

For the Hamming window design we have

$$\boxed{h_n = \left[.54 - .46\cos\frac{2\pi n}{29}\right]\frac{1}{4}\mathrm{sinc}\,\frac{\pi}{4}\left(n-\frac{29}{2}\right) \quad 0 \le n \le 29}$$

Example

Design generalized linear phase high-pass FIR filters having coefficients $\{h_n\}_{n=0}^{60}$ and cutoff $\omega_c = \frac{2\pi}{3}$ using the window design procedure with both truncation and Hamming windows.

Solution

$$G_d(\omega) = \begin{cases} e^{-j30\omega} & \frac{2\pi}{3} \le |\omega| \le \pi \\ 0 & |\omega| < \frac{2\pi}{3} \end{cases}$$

$$\Rightarrow g_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} \underbrace{G_d(\omega) e^{j\omega n}}_{\text{periodic with period } = 2\pi} \, d\omega$$

$$= \frac{1}{2\pi} \int_{0}^{2\pi} G_d(\omega) e^{j\omega n} \, d\omega \qquad (*)$$

$$= \frac{1}{2\pi} \int_{\frac{2\pi}{3}}^{\frac{4\pi}{3}} e^{-j30\omega} e^{j\omega n} \, d\omega \qquad (\square)$$

$$= \left. \frac{e^{j\omega(n-30)}}{2\pi j(n-30)} \right|_{\frac{2\pi}{3}}^{\frac{4\pi}{3}} \qquad (\Delta)$$

$$= \frac{1}{2\pi j(n-30)} \left[e^{j\frac{4\pi}{3}(n-30)} - e^{j\frac{2\pi}{3}(n-30)} \right]$$

$$= \frac{1}{2\pi j(n-30)} e^{j\pi(n-30)} \left[e^{j\frac{\pi}{3}(n-30)} - e^{-j\frac{\pi}{3}(n-30)} \right]$$

$$= \frac{1}{2\pi j(n-30)} (-1)^n \, 2j \sin\frac{\pi}{3}(n-30)$$

$$= (-1)^n \frac{1}{3} \operatorname{sinc} \frac{\pi}{3}(n-30)$$

So, using a truncation window gives:

$$\boxed{h_n = (-1)^n \frac{1}{3} \text{sinc} \frac{\pi}{3}(n-30) \quad 0 \le n \le 60}$$

and applying a Hamming window gives:

$$\boxed{h_n = \left[.54 - .46\cos\frac{2\pi n}{60}\right](-1)^n \frac{1}{3} \text{sinc} \frac{\pi}{3}(n-30) \quad 0 \le n \le 60}$$

Notes:

1) Since $G_d(\omega)$ is nonzero on <u>two</u> subintervals of $-\pi \le \omega < \pi$ for the high-pass case, it can save algebra if we use periodicity to rewrite the inverse DTFT as in (*) across the interval $0 \le \omega \le 2\pi$. This trick is straightforward if N is odd. For N even, however, there are two extra things to think about. First, in view of the table on p. 35.8, we should use an odd-symmetric design with Type 2 generalized linear phase. Second, for N even, the slope of the phase, $-\frac{N-1}{2}$, is noninteger and the phase will take a jump of π at $\omega = \pi$. Thus, $G_d(\omega)$ seemingly will have two different forms on the interval $0 \le \omega \le 2\pi$. See the next example for details.

2) Since the denominator of (Δ) is zero at n = 30, we cannot presume that (Δ) follows from ($\square$) at n = 30. Thus, we must be careful to check that our expressions for $\{h_n\}$ hold at n = 30. Our final expression for g_n, which follows from (Δ), gives:

$$g_{30} = (-1)^{30} \bullet \frac{1}{3} \bullet 1 = \frac{1}{3}$$

Evaluating ($\square$) at n = 30 gives

$$g_{30} = \frac{1}{2\pi}\left[\frac{4\pi}{3} - \frac{2\pi}{3}\right] \bullet 1 = \frac{1}{3}$$

which agrees with (Δ). Thus, our expressions for h_n are valid for $0 \le n \le 60$.

Now, let's change N in the previous example from N = 61 to N = 62 and see how the algebra associated with the design changes for N even.

Example

Design a generalized linear phase high-pass FIR filter having coefficients $\{h_n\}_{n=0}^{61}$ and cutoff $\omega_c = \frac{2\pi}{3}$ using the window design procedure with a Hamming window.

Solution

A filter with real-valued unit-pulse response satisfies $G_d(\omega) = G_d^*(-\omega)$. Thus, for an antisymmetric design with Type 2 generalized linear phase, we have

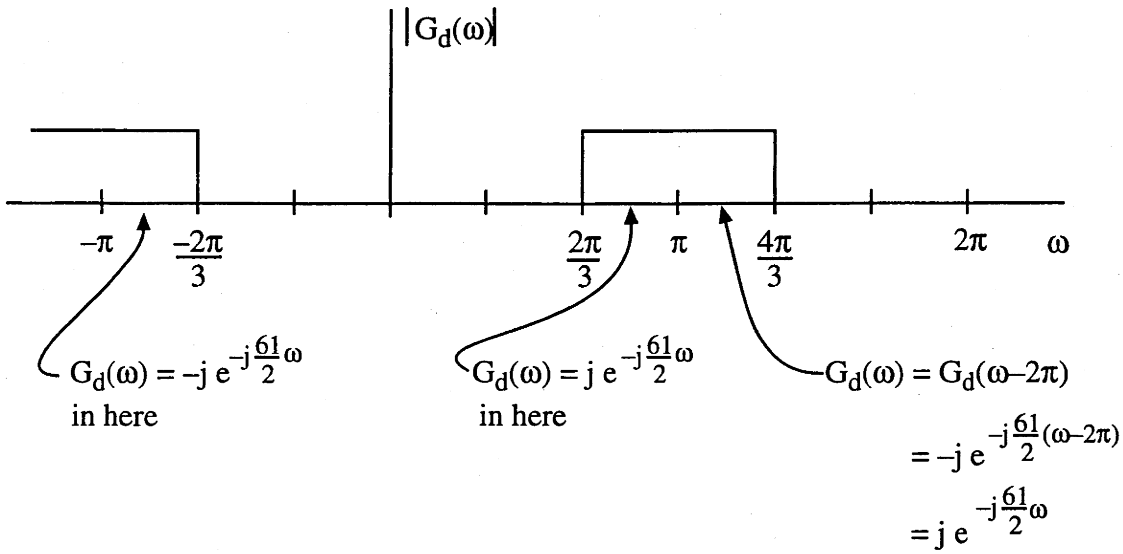
$$G_d(\omega) = \begin{cases} e^{j\left(\frac{\pi}{2}-\frac{61}{2}\omega\right)} & \frac{2\pi}{3} \le \omega < \pi \\ 0 & |\omega| < \frac{2\pi}{3} \\ e^{j\left(-\frac{\pi}{2}-\frac{61}{2}\omega\right)} & -\pi < \omega \le -\frac{2\pi}{3} \end{cases}$$

$$= \begin{cases} j\, e^{-j\frac{61}{2}\omega} & \frac{2\pi}{3} \le \omega < \pi \\ 0 & |\omega| < \frac{2\pi}{3} \\ -j\, e^{-j\frac{61}{2}\omega} & -\pi < \omega \le -\frac{2\pi}{3} \end{cases} \qquad (*)$$

As in the last example we can write

$$g_n = \frac{1}{2\pi} \int_{\frac{2\pi}{3}}^{\frac{4\pi}{3}} G_d(\omega)\, e^{j\omega n}\, d\omega \qquad (**)$$

Equation (*) specifies $G_d(\omega)$ on $|\omega| < \pi$. To find $G_d(\omega)$ for $\pi < \omega < \frac{4\pi}{3}$ consider



So, $G_d(\omega)$ in (**) maintains the same form across the full range of integration in (**) and we have

$$
\begin{aligned}
g_n &= \frac{1}{2\pi}\int_{\frac{2\pi}{3}}^{\frac{4\pi}{3}} j e^{-j\frac{61}{2}\omega}\, e^{j\omega n}\, d\omega \\
&= \left.\frac{e^{j\omega\left(n-\frac{61}{2}\right)}}{2\pi\left(n-\frac{61}{2}\right)}\right|_{\frac{2\pi}{3}}^{\frac{4\pi}{3}} \\
&= \frac{1}{2\pi\left(n-\frac{61}{2}\right)}\left[e^{j\frac{4\pi}{3}\left(n-\frac{61}{2}\right)} - e^{j\frac{2\pi}{3}\left(n-\frac{61}{2}\right)}\right] \\
&= \frac{1}{2\pi\left(n-\frac{61}{2}\right)}\, e^{j\pi\left(n-\frac{61}{2}\right)}\left[e^{j\frac{\pi}{3}\left(n-\frac{61}{2}\right)} - e^{-j\frac{\pi}{3}\left(n-\frac{61}{2}\right)}\right] \\
&= \frac{1}{2\pi\left(n-\frac{61}{2}\right)}(-1)^n\, 2\sin\left[\frac{\pi}{3}\left(n-\frac{61}{2}\right)\right] \\
&= (-1)^n\, \frac{1}{3}\,\mathrm{sinc}\,\frac{\pi}{3}\left(n-\frac{61}{2}\right)
\end{aligned}
$$

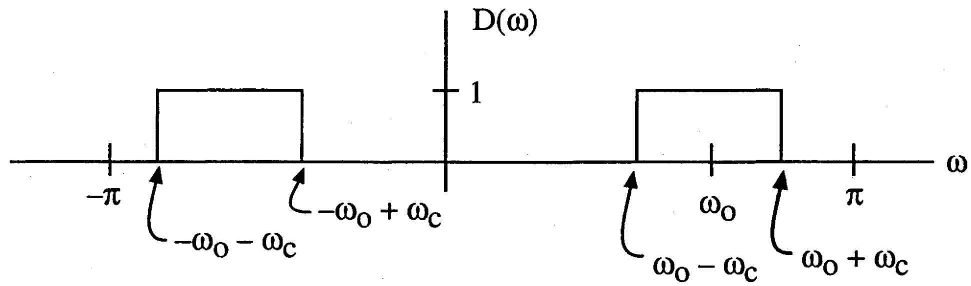
Applying the Hamming window gives

$$\boxed{h_n \;=\; \left[.54 - .46\cos\frac{2\pi n}{61}\right](-1)^n \frac{1}{3}\,\text{sinc}\,\frac{\pi}{3}\left(n-\frac{61}{2}\right) \qquad 0 \le n \le 61}$$

similar to the example on pages 37.4, 37.5.

Example

Window design of bandpass filter. Find $\{h_n\}_{n=0}^{N-1}$ so that $|H_d(\omega)|$ approximates



For lowpass filter had $d_n = \frac{\omega_c}{\pi}$ sinc ω_c n. By modulation property, here for BP case we expect $d_n \approx (\cos\omega_o n)\,\frac{\omega_c}{\pi}\text{sinc}\,[\omega_c n]$. Let's see:

$$d_n = \frac{1}{2\pi}\int_{-\omega_o-\omega_c}^{-\omega_o+\omega_c} 1e^{j\omega n}\,d\omega + \frac{1}{2\pi}\int_{\omega_o-\omega_c}^{\omega_o+\omega_c} 1e^{j\omega n}\,d\omega$$

$$= \frac{e^{j\omega n}}{2\pi jn}\Bigg|_{-\omega_o-\omega_c}^{-\omega_o+\omega_c} + \frac{e^{j\omega n}}{2\pi jn}\Bigg|_{\omega_o-\omega_c}^{\omega_o+\omega_c}$$

$$= \frac{e^{jn(-\omega_o+\omega_c)} - e^{jn(-\omega_o-\omega_c)} + e^{jn(\omega_o+\omega_c)} - e^{jn(\omega_o-\omega_c)}}{2\pi jn}$$

$$= \frac{1}{2\pi jn}\left[e^{-jn\omega_o}\left(e^{jn\omega_c} - e^{-jn\omega_c}\right) + e^{jn\omega_o}\left(e^{jn\omega_c} - e^{-jn\omega_c}\right)\right]$$

$$= \frac{1}{\pi n}\left[e^{-jn\omega_o}\sin\omega_c n + e^{jn\omega_o}\sin\omega_c n\right] = \frac{\sin\omega_c n}{\pi n}\,2\cos\omega_o n$$

$$= 2(\cos\omega_o n)\frac{\omega_c}{\pi}\text{sinc}\,\omega_c n \qquad \leftarrow \text{what we expected.}$$

Now, to design $\{h_n\}_{n=0}^{N-1}$ with linear phase, need to incorporate shift to give:

$$\boxed{g_n = 2\cos\omega_o\left(n - \frac{N-1}{2}\right)\frac{\omega_c}{\pi}\,\text{sinc}\;\omega_c\left(n - \frac{N-1}{2}\right) \qquad 0 \le n \le N-1}$$

Windowed coefficients are then

$\boxed{h_n = w_n\, g_n \qquad 0 \le n \le N-1}$ ~ coefficients for FIR filter where w_n is a Hamming or other window.

ECE 310 **DIGITAL SIGNAL PROCESSING** **D. Munson**
University of Illinois **Lecture 38**

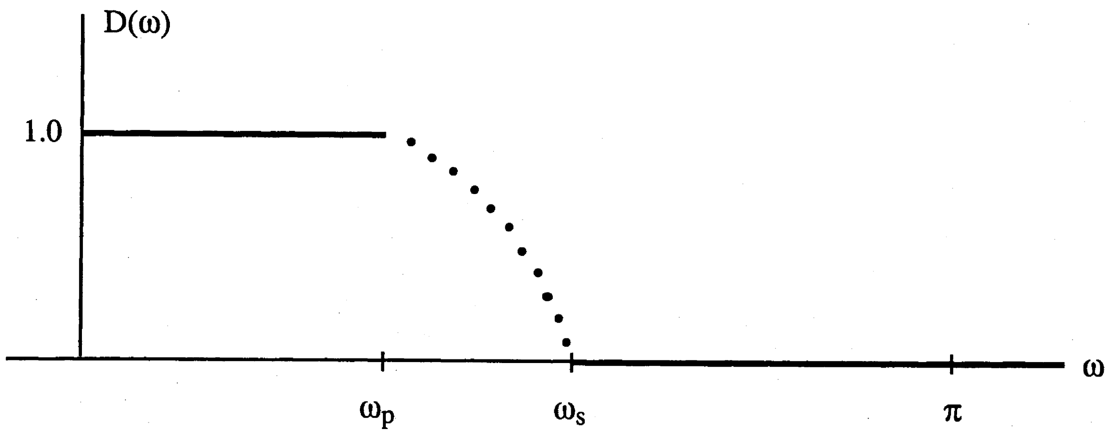
Parks-McClellan

Parks and McClellan developed a computer program for solving the following problem:

Define the error

$$E(\omega) = \underbrace{W(\omega)}_{\text{arbitrary weighting}}[\underbrace{D(\omega)}_{\text{desired response}} - \underbrace{R(\omega)}_{\text{from } H_d(\omega) = R(\omega)e^{j\left(\underset{0 \text{ or } \frac{\pi}{2}}{\alpha} - \frac{N-1}{2}\omega\right)}}]$$

Let ω_p, ω_s be the passband and stopband cutoff frequencies so that $D(\omega)$ might look like:



Ordinarily $D(\omega)$ is unspecified on $\omega_p < \omega < \omega_s$ because we are not concerned with the precise shape of $H_d(\omega)$ or $R(\omega)$ in this <u>transition band</u>.

The Parks-McClellan algorithm finds $\{h_n\}_{n=0}^{N-1}$ that minimizes

$$\max_{\substack{0 \le \omega \le \omega_p \\ \omega_s \le \omega \le \pi}} |E(\omega)|$$

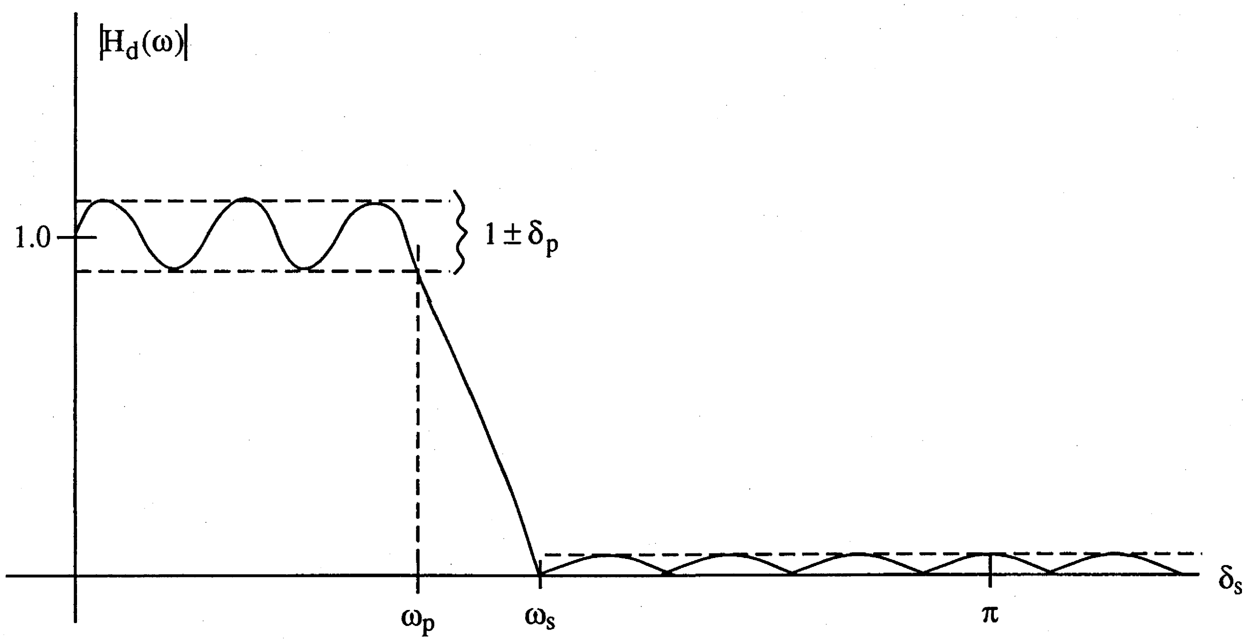
This error measure is called the minimax or Chebyshev error norm.

In the standard P-M algorithm, $W(\omega)$ can be selected to have one value, W_p, on the passband and another, W_s, on the stopband. Frequently, W_s is chosen larger than W_p so that the designed filter will have a smaller stopband error than passband error.

The program user specifies:

N, ω_p, ω_s, W_p, and W_s

The designed filter has <u>equiripple</u> behavior:



How are the ripple heights related to W_p and W_s?

Answer: They satisfy $\delta_p W_p = \delta_s W_s$.

Thus, $\dfrac{\delta_p}{\delta_s} = \dfrac{W_s}{W_p}$.

What filter order is required to meet given specifications?

Answer: It has been found empirically that

$$N \approx \frac{-10\log_{10}\left(\delta_p \delta_s\right) - 13}{2.324\left(\omega_s - \omega_p\right)}$$

Note: The filter order is not too sensitive to δ_p, and δ_s. <u>But</u>, N is inversely proportional to the transition bandwidth! Halving the transition bandwidth doubles the required filter length!

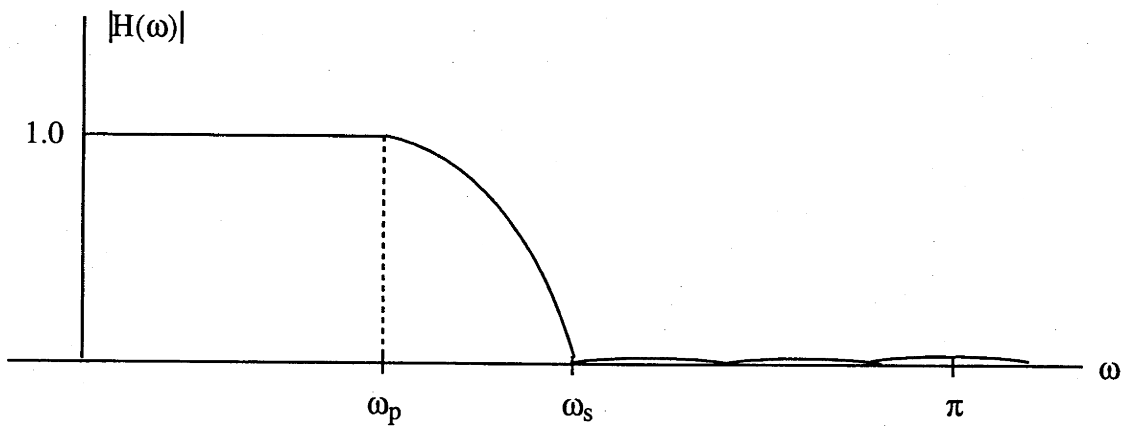
All students will get experience with the Parks-McClellan algorithm in a Matlab assignment.

Linear Programming

Linear programming is a fairly general optimization algorithm that can solve the Parks-McClellan problem and many others. LP is slower, but its generality can be exploited to incorporate time-domain or additional frequency-domain constraints.

For example, it is possible to eliminate ripple in the passband by constraining the derivative of $R(\omega)$ to be ≤ 0 in the passband.

The result is a monotone decreasing response in the passband, with ripple remaining in the stopband:



Incorporating the monotone passband constraint increases the required filter order slightly.

An LP package, called METEOR, has been developed by Steiglitz, Parks, and Kaiser for FIR filter design. The software is available by ftp from Prof. Steiglitz in the Department of Computer Science, Princeton University.

ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 39

D. Munson

IIR Filter Design

1) Based on Analog Prototype

   a) Impulse invariant design
   b) Bilinear transformation (✓) ~ widely used

2) Computer-Aided Optimization

Designs in category 1) proceed by first designing an analog filter having a frequency response with the desired shape, and then "transforming" it to a digital filter. To use these design methods, we must first learn just a bit about analog filter design.

Elements of Analog Filter Design

Notation:

$$H_L(s) = \int_{-\infty}^{\infty} h_a(t)\, e^{-st}\, dt \qquad \text{(Laplace transform)}$$

$$\Rightarrow \quad H_a(\Omega) = H_L(j\Omega) \qquad \text{(Fourier transform)}$$

Consider only lowpass Butterworth, Chebyshev, and elliptic (Cauer) filters.

For each of these types of filters, $H_L(s)$ is found <u>indirectly</u> from a specified $|H_a(\Omega)|^2$.

We need $H_L(s)$ because, later, this is what will be transformed into $H(z)$.

For Butterworth, Chebyshev, and elliptic filters, $|H_a(\Omega)|^2$ has the form:

$$|H_a(\Omega)|^2 = M(\Omega^2) = \frac{1}{1+\underset{\substack{\uparrow \\ \text{rational with} \\ \text{real coeffs.}}}{F(\Omega^2)}}$$

How do we get $H_L(s)$ from $|H_a(\Omega)|^2$? That is, how do we find $H_L(s)$ satisfying

$$|H_L(j\Omega)|^2 = |H_a(\Omega)|^2 = M(\Omega^2)? \qquad (*)$$

Answer:

1) First find poles and zeros of $M(-s^2)$ where s is a complex variable. Since M has real coefficients and is a function of s^2, the poles and zeros will have symmetry around both the real and imaginary axii.

2) Take $H_L(s)$ to be the left-half-plane pole factors (for stability) and left-half-plane zero factors (for smallest delay, called "minimum phase").

<u>But</u>, does this work?

Need to show

$$|H_L(j\Omega)|^2 = M(\Omega^2) \qquad (1)$$

Have:

$$\underbrace{H_L(s)}_{\text{LHP factors}}\ \underbrace{H_L(-s)}_{\text{RHP factors}} = M(-s^2)$$

which implies

$$H_L(j\Omega)\ H_L(-j\Omega) = M(\Omega^2)$$

So, (1) will be true if

$$H_L(-j\Omega) = H_L^*(j\Omega) \qquad (2)$$

This follows, though, because the poles and zeros of $H_L(s)$ are symmetric around the real axis, and therefore occur in complex-conjugate pairs. For any pole pair or zero pair $(s-p)\ (s-p^*)$ in $H_L(s)$, we have

$$(s-p)\ (s-p^*)\Big|_{s=-j\Omega} = (-j\Omega-p)\ (-j\Omega-p^*) = [(j\Omega-p^*)\ (j\Omega-p)]^* = \left[(s-p)(s-p^*)\Big|_{s=j\Omega}\right]^*$$

which proves (2), and therefore (1).

Example

$$|H_a(\Omega)|^2 = M(\Omega^2) = \frac{1}{1+\Omega^2}$$

Find $H_L(s)$.

We have

$$M(-s^2) = \frac{1}{1-s^2}$$

$$= \frac{1}{(1-s)\underbrace{(1+s)}_{\text{LHP factor}}}$$

$$\Rightarrow \quad \left[H_L(s) = \frac{1}{s+1}\right]$$

Let's check to see if $|H_a(\Omega)|^2$ = above $M(\Omega^2)$:

$$|H_a(\Omega)|^2 = |H_L(j\Omega)|^2$$

$$= \left|\frac{1}{j\Omega+1}\right|^2 = \frac{1}{|j\Omega+1|^2}$$

$$= \frac{1}{\Omega^2+1} = M(\Omega^2) \quad ✔$$

Example

Suppose $M(\Omega^2) = \dfrac{2+\Omega^2}{1+\Omega^4}$ (not for a B, C, or E filter!)

Then

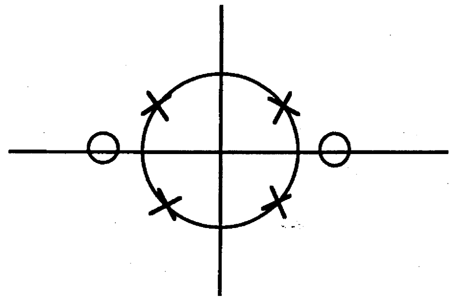
$$M(-s^2) = \frac{2-s^2}{1+s^4}.$$

$$= \frac{\left(\sqrt{2}-s\right)\left(\sqrt{2}+s\right)}{(s+\gamma)(s+\gamma^*)(s-\gamma)(s-\gamma^*)}$$

$$\uparrow$$

$$\gamma = e^{j\frac{\pi}{4}} = \frac{1+j}{\sqrt{2}}$$

Pole-zero diagram:



$\Rightarrow$ Take $H_L(s) = \dfrac{\sqrt{2}+s}{(s+\gamma)(s+\gamma^*)}$ (LHP factors)

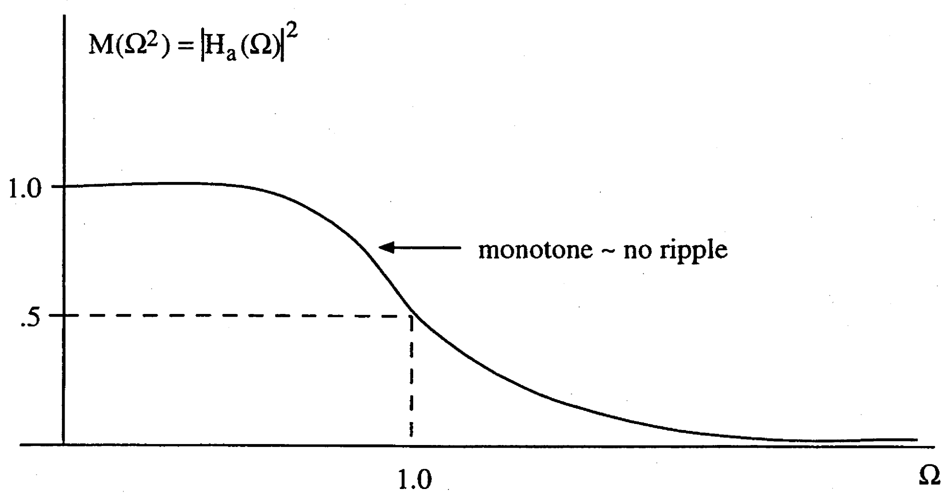
Can check that $|H_L(j\Omega)|^2 = M(\Omega^2)$.

$M\left(\Omega^2\right)$ for B, C, and E Filters

Butterworth

$M(\Omega^2) = \dfrac{1}{1+F(\Omega^2)}$ with $F(\Omega^2) = \Omega^{2n}$ for nth-order filter

Result:



$M(\Omega^2) = .5$ at $\Omega = \Omega_c = 1.0$. For a general cutoff frequency Ω_c, use $F(\Omega^2) = \left(\dfrac{\Omega}{\Omega_c}\right)^{2n}$.

Here, we are defining the cutoff frequency to be the value of Ω where $|H_a(\Omega)|^2$ is reduced to one-half its maximum height, or correspondingly, $|H_a(\Omega)|$ reaches $1/\sqrt{2}$ times its maximum value. This definition of cutoff frequency is common, particularly for smooth frequency responses that contain little or no ripple.

Optimality: This $M(\Omega^2)$ has maximum # of derivatives = 0 at the origin for its order. Thus, the response is very flat across lower frequencies.

Can show poles of $M(-s^2) = H_L(s)\, H_L(-s)$ are equally spaced on the unit circle. This fact helps in factoring $M(-s^2)$.

Chebyshev

$$F(\Omega^2) = \varepsilon^2\, C_n^2(\Omega)$$

where ε is a real constant chosen by the designer and $C_n(\bullet)$ is the nth-order Chebyshev polynomial:

$$C_n(\Omega) = \begin{cases} \cos\left(n \cos^{-1}(\Omega)\right) & |\Omega| \le 1 \\ \cosh\left(n \cosh^{-1}(\Omega)\right) & |\Omega| > 1 \end{cases}$$

with

$$\cosh t = \frac{e^t + e^{-t}}{2}$$

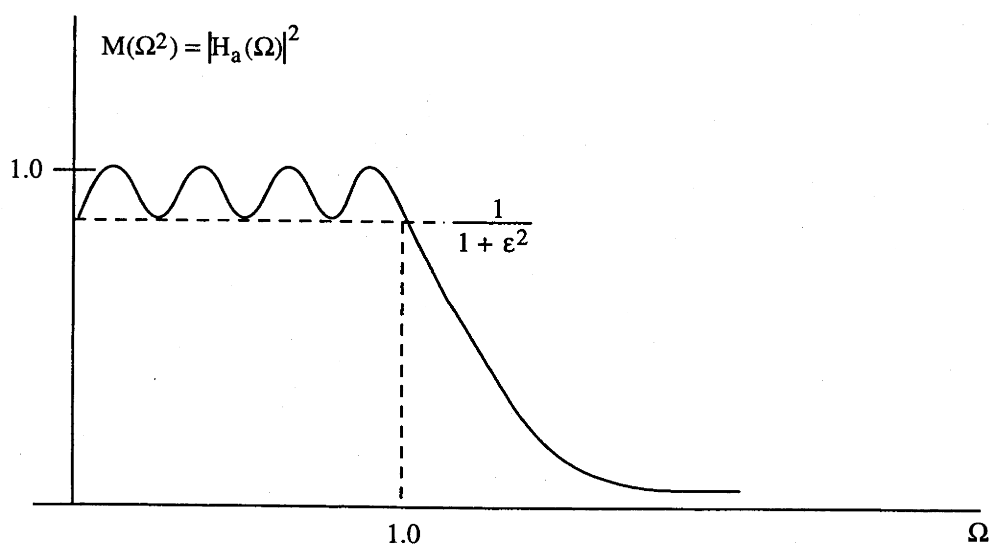
Can show:

$$C_0(\Omega) = 1,\ C_1(\Omega) = \Omega,\ C_2(\Omega) = 2\Omega^2 - 1,$$

and that there is a recursion relation:

$$C_{n+1}(\Omega) = 2\Omega\, C_n(\Omega) - C_{n-1}(\Omega)$$

Result:



For this type of filter, the cutoff frequency is defined to be the value of Ω where $|H_a(\Omega)|^2$ first drops below $1/(1+\varepsilon^2)$ or, correspondingly, $|H_a(\Omega)|$ first drops below $1/\sqrt{1+\varepsilon^2}$.

This is a "Type 1" Chebyshev filter. Its response is equiripple in the passband and monotone decreasing in the stopband. It has a narrower transition band than a Butterworth filter.

Tradeoff: Smaller ε gives smaller passband ripple but a wider transition band.

Poles of $M(-s^2) = H_L(s)\,H_L(-s)$ lie on an ellipse.

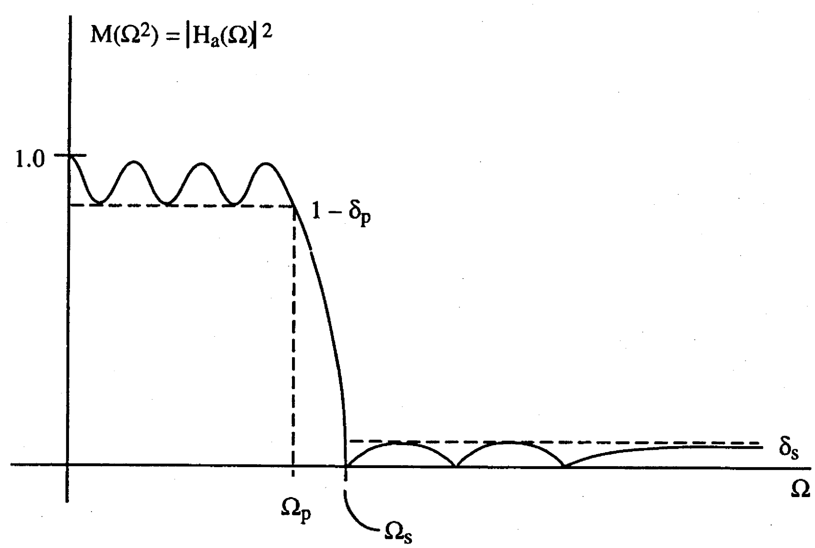
There is also a "Type 2" Chebyshev filter that has a monotone response in its passband and equiripple behavior in its stopband.

<u>Elliptic</u>

$$F(\Omega^2) = \varepsilon^2\, J_n^2(\Omega) \text{ where } J_n \text{ is the Jacobi elliptic function.}$$

The defining formulas for J_n are so cumbersome that they are not presented here.

Result:



The response is equiripple in both the passband and stopband.

Elliptic filters are optimal in the sense that for a given n, δ_p, δ_s, Ω_p, the transition bandwidth $\Omega_s - \Omega_p$ is the smallest possible.

$\angle H_a(\Omega)$ for B, C, and E filters is reasonably linear till you get near the edge of the passband, where it can be quite nonlinear.

The phase response is closest to linear for B, then C. Elliptic is worst.

All-pass filters are sometimes cascaded onto elliptic filters to compensate for the nonlinear phase of elliptic filters.

All-pass filters have $|H_a(\Omega)|$ = constant and the coefficients are chosen to shape $\angle H_a(\Omega)$ in a desired way.

ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 40

D. Munson

1b) Bilinear Transformation

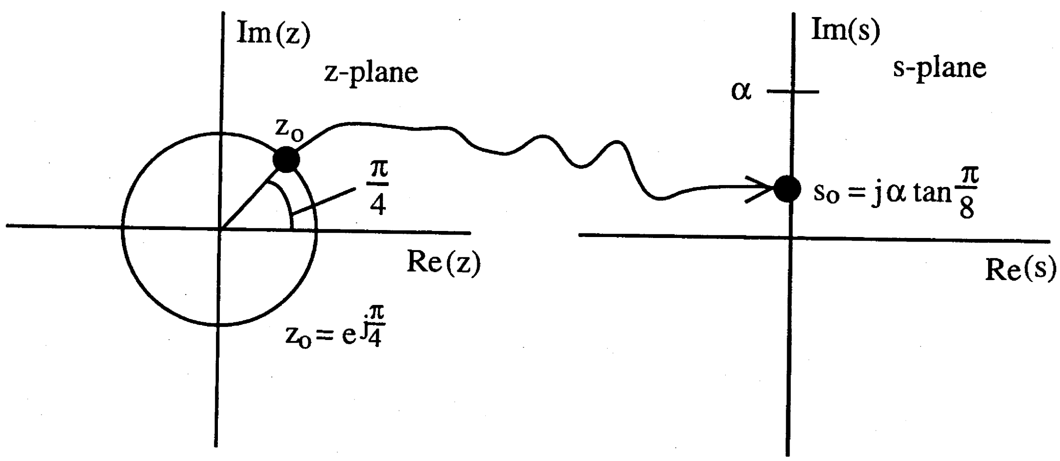
Start with analog prototype $H_L(s)$.

Take

$$H(z) = H_L(s)\Big|_{s=\alpha\frac{1-z^{-1}}{1+z^{-1}}}$$

α is a real positive constant that we will be able to choose to control the cutoff frequency of the digital filter.

$s = \alpha \dfrac{1-z^{-1}}{1+z^{-1}}$ is a <u>bilinear transformation</u> (BLT) of the z-plane to the s-plane. For example, the point z_o maps to the point s_o as shown below.



To see this, note:

$$s_o = \alpha\,\frac{1-z_o^{-1}}{1+z_o^{-1}} = \alpha\,\frac{1-e^{-j\frac{\pi}{4}}}{1+e^{-j\frac{\pi}{4}}} = \alpha\,\frac{e^{-j\frac{\pi}{8}}}{e^{-j\frac{\pi}{8}}}\;\frac{e^{j\frac{\pi}{8}}-e^{-j\frac{\pi}{8}}}{e^{j\frac{\pi}{8}}+e^{-j\frac{\pi}{8}}}$$

$$= \alpha\,\frac{2j\sin\frac{\pi}{8}}{2\cos\frac{\pi}{8}} = j\,\alpha\tan\frac{\pi}{8}$$

So, if we design a digital filter using the BLT, then $H(z_o) = H_d\left(\frac{\pi}{4}\right)$ will have the same value as $H_L(s_o) = H_a\left(\alpha \tan \frac{\pi}{8}\right)$.

We must have a much broader understanding than this, however. Questions:

i) Stable $H_L(s) \Rightarrow$ stable $H(z)$?

ii) How is $H_d(\omega) = H\left(e^{j\omega}\right)$ related to $H_L(s)$?

Answer i) by considering a point $s = s_o$ and determining what z it gets mapped to.

BLT mapping is $s = \alpha \dfrac{1-z^{-1}}{1+z^{-1}}$

$$\Rightarrow \quad s\left(1+z^{-1}\right) = \alpha\left(1-z^{-1}\right)$$

$$\Rightarrow \quad z^{-1}(s+\alpha) = \alpha - s$$

$$\Rightarrow \quad z = \frac{\alpha+s}{\alpha-s}$$

So, a point $s_o = \sigma_o + j\,\Omega_o$.

gets mapped to:

$$z_o = \frac{\alpha+\sigma_o+j\Omega_o}{\alpha-\sigma_o-j\Omega_o} \qquad (*)$$

i.e., $H(z_o) = H_L(s_o)$.

From (*)

$$|z_o| = \left[\frac{(\alpha+\sigma_o)^2+\Omega_o^2}{(\alpha-\sigma_o)^2+\Omega_o^2}\right]^{1/2}$$

So:

$$|z_o| \begin{cases} <1 & \sigma_o < 0 \\ =1 & \sigma_o = 0 \\ >1 & \sigma_o > 0 \end{cases}$$

giving:

a) Left-half s-plane is mapped <u>inside</u> the unit circle in the z-plane.

b) Right-half s-plane is mapped <u>outside</u> the unit circle in the z-plane.

c) Imaginary axis in s-plane is mapped <u>onto</u> the unit circle in the z-plane.

Note:

$\left[a) \Rightarrow \text{stable } H_L(s) \text{ results in stable } H(z)\right]$

$\left[c) \Rightarrow H_d(\omega) = H(e^{j\omega}) \text{ depends only on } H_L(j\Omega) = H_a(\Omega)\right]$

What is the relationship between $H_d(\omega)$ and $H_a(\Omega)$? $H_d(\omega)$ is given by

$$H_d(\omega) = H(e^{j\omega}) = H_L(s)\Big|_{s=\alpha\frac{1-e^{-j\omega}}{1+e^{-j\omega}}}$$

Note:

$$\alpha\,\frac{1-e^{-j\omega}}{1+e^{-j\omega}} = \alpha\,\frac{e^{-j\omega/2}}{e^{-j\omega/2}}\,\frac{e^{j\omega/2}-e^{-j\omega/2}}{e^{j\omega/2}+e^{-j\omega/2}}$$

$$= \alpha\,\frac{2j\sin\frac{\omega}{2}}{2\cos\frac{\omega}{2}} = j\,\alpha\,\tan\frac{\omega}{2}$$

So,

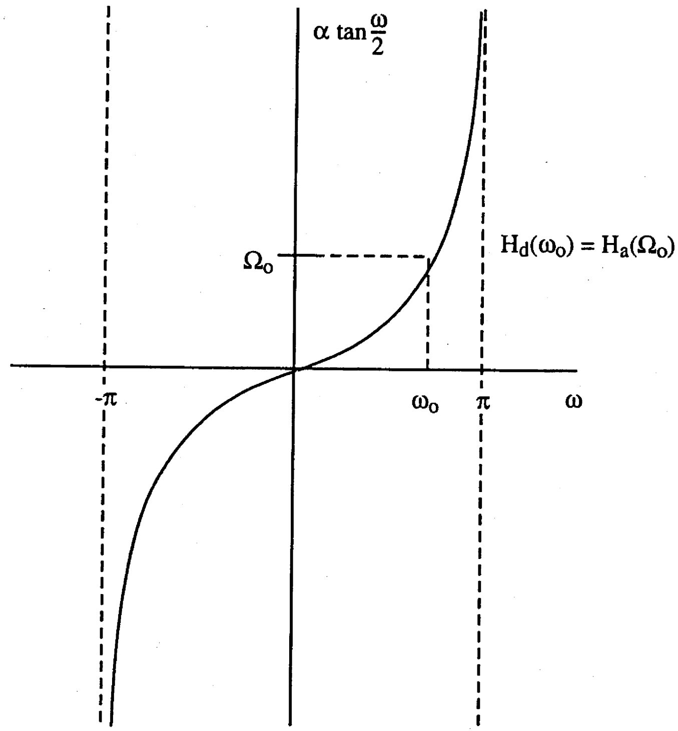
$$H_d(\omega) = H_L(s)\Big|_{s=j\alpha\tan\frac{\omega}{2}}$$

Since $H_a(\Omega) = H_L(j\Omega)$ we have

$$\boxed{H_d(\omega) = H_a\left(\alpha\tan\frac{\omega}{2}\right)} \qquad (\square)$$

This equation tells us exactly how, when using the bilinear transformation design method, the frequency response of the designed digital filter will depend on the frequency response of the analog prototype.

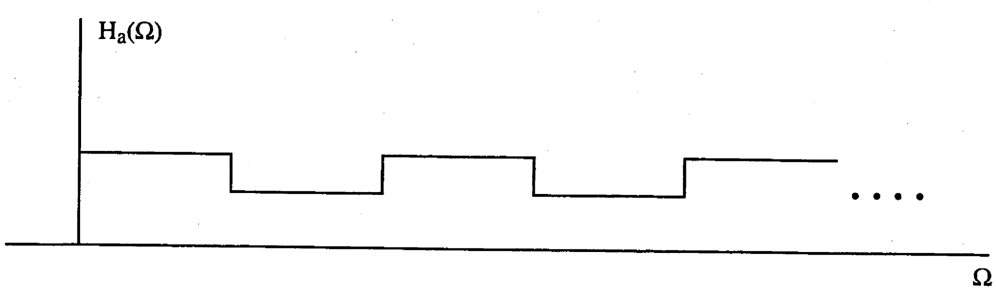
Pictorial interpretation of (❑):



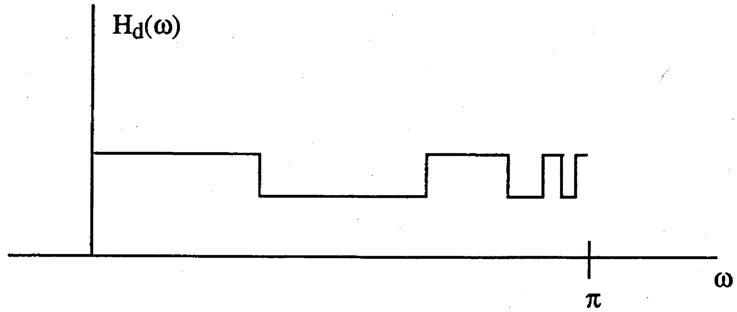
So, $H_d(\omega)$ takes on exactly the same set of values as $H_a(\Omega)$, but there is a squashing of the analog frequency axis, according to the above curve. This mapping is nonlinear, and it has to be, since the infinite-length analog frequency axis $-\infty < \Omega < \infty$ is mapped onto the finite-length interval $-\pi < \omega < \pi$. Because of this, $H_d(\omega)$ won't look like $H_a(\Omega)$ in general. It will be a warped version. This effect is sometimes called "frequency warping."

Example

Applying the BLT to an analog prototype having frequency response:



results in:



In general, to design $H_d(\omega)$ having a desired shape, we would need to design $H_a(\Omega)$ so that after application of the BLT, the frequency warping produces the desired $H_d(\omega)$. Thus, we would need to "prewarp."

Let the desired $H_d(\omega)$ be $D(\omega)$.

$$(\square) \Rightarrow \text{want } H_a\left(\alpha \tan\frac{\omega}{2}\right) = D(\omega)$$

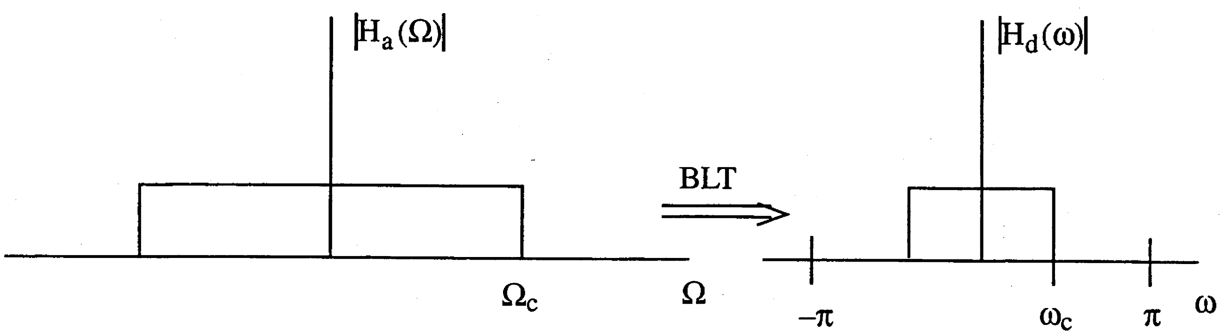
$$\Rightarrow H_a(\Omega) = D\left(2\tan^{-1}\frac{\Omega}{\alpha}\right) \qquad (\Delta)$$

If we *could* design $H_L(s)$ to satisfy (Δ), the BLT, would then give

$$H_d(\omega) = D(\omega)\,.$$

This is problematic, however. The design of H_L with a general shape will require computer optimization. Thus, we may as well design H_d directly, using computer optimization.

Fortunately, for LPF's, BPF's and HPF's, there is no such problem. For these kinds of filters, frequency warping just affects the cutoff frequencies, e.g.,



From ($\square$):

$$\boxed{\Omega_c = \alpha \tan\frac{\omega_c}{2}} \qquad (\square\square)$$

So, we can simply pick Ω_c to give the desired ω_c. Equivalently, if $H_a(\Omega)$ is normalized to $\Omega_c = 1$, we can choose α in the BLT to give the desired ω_c.

This suggests two alternative, but equivalent, design procedures:

1. a) Choose α arbitrarily, say $\alpha = 1$.

   b) Then design the analog prototype to have cutoff Ω_c given by (❑❑).

   c) Apply the BLT.

2. a) Use an analog prototype with Ω_c.

   b) Choose α to give the desired ω_c.

   From (❑❑),

$$\alpha = \Omega_c \cot\left(\frac{\omega_c}{2}\right)$$

   so that the BLT method becomes

$$H(z) = H_L(s)\Big|_{s = \Omega_c \cot\frac{\omega_c}{2}\,\frac{1-z^{-1}}{1+z^{-1}}}$$

Analog prototype filters are usually designed with normalized cutoff frequencies $\Omega_c = 1.0$. In this case, the bilinear transformation method of design reduces to

$$\boxed{H(z) = H_L(s)\Big|_{s=\cot\frac{\omega_c}{2}\,\frac{1-z^{-1}}{1+z^{-1}}}} \qquad \textbf{(BLT)}$$

In this course, we will use the second option, with Eq. (BLT), to perform the bilinear transformation method of design.

ECE 310 **DIGITAL SIGNAL PROCESSING** **D. Munson**
University of Illinois **Lecture 41**

Example

Design a first-order Butterworth digital filter with $\omega_c = \frac{\pi}{4}$, using the BLT method

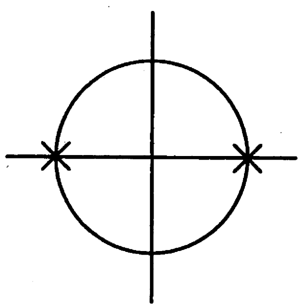
Solution:

First, find the analog prototype:

$$|H_a(\Omega)|^2 = B(\Omega^2) = \frac{1}{1+\Omega^2}$$

$$B(-s^2) = \frac{1}{1-s^2} = \frac{1}{1-s} \bullet \frac{1}{1+s}$$

Pole locations:



Take LHP factor for $H_L(s)$:

$$H_L(s) = \frac{1}{s+1}$$

Apply (BLT):

$$H(z) = \left.\frac{1}{s+1}\right|_{s=\underbrace{\cot\frac{\pi}{8}}_{2.4142}\frac{1-z^{-1}}{1+z^{-1}}}$$

$$\Rightarrow H(z) = \frac{1}{2.4142\,\dfrac{1-z^{-1}}{1+z^{-1}}+1}$$

$$= \frac{1+z^{-1}}{2.4142\,(1-z^{-1})+1+z^{-1}}$$

$$= \frac{1+z^{-1}}{3.4142-1.4142\,z^{-1}}$$

$$= \boxed{\frac{.2929+.2929\,z^{-1}}{1-.4142\,z^{-1}}}$$

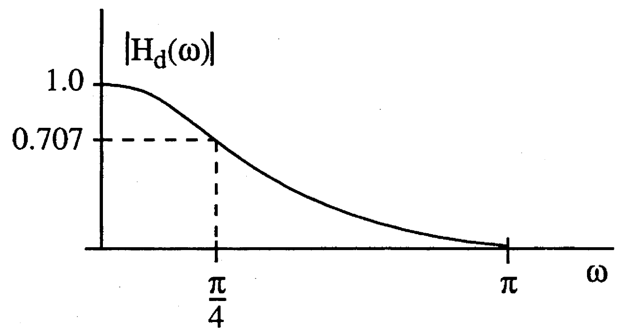
Now, since H(z) resulted from a BLT design using an analog prototype filter with $|H_a(0)| = 1$ and $|H_d(\infty)| = 0$, we know

$$|H_d(0)| = |H_a(0)| = 1 \quad \text{and} \quad |H_a(\pi)| = |H_a(\infty)| = 0$$

Furthermore, since the digital cutoff is $\frac{\pi}{4}$, we should have

$$\left|H_d\left(\frac{\pi}{4}\right)\right| = \frac{1}{\sqrt{2}}$$

Thus, $|H_d(\omega)|$ should look like



The correctness of the values of $|H_d(\omega)|$ at $\omega = 0, \pi$ can be easily verified from the transfer function H(z). Note that

$$H_d(0) = H(e^{j0}) = H(1) = \frac{.2929+.2929}{1-.4142} = 1.0$$

$$H_d(\pi) = H(e^{j\pi}) = H(-1) = \frac{.2929-.2929}{1-.4142} = 0$$

For $\omega \neq 0$ or π we can always evaluate the magnitude response via the lengthier calculation

$$|H_d(\omega)| = \frac{|.2929+.2929\,e^{-j\omega}|}{|1-.4142\,e^{-j\omega}|}$$

$$= \frac{\sqrt{(.2929+.2929\cos\omega)^2+(.2929\sin\omega)^2}}{\sqrt{(1-.4142\cos\omega)^2+(.4142\sin\omega)^2}}$$

Finally, a plot of $\angle H_d(\omega)$ would show that the phase is nearly linear for $|\omega| < \frac{\pi}{4}$ and becomes more nonlinear for $|\omega| \approx \frac{\pi}{4}$ and larger.

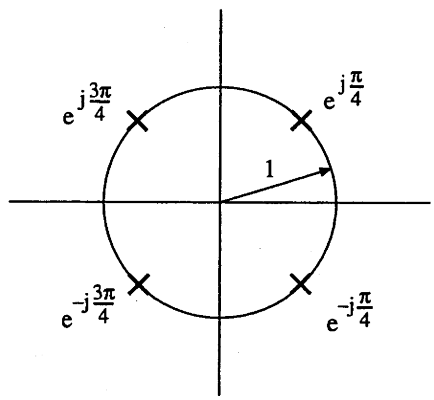
Example

Similar to previous example, but now design a second-order Butterworth digital filter with $\omega_c = \frac{\pi}{4}$.

To find analog prototype, note

$$|H_a(\Omega)|^2 = M(\Omega^2) = \frac{1}{1+\Omega^4}$$

$$\Rightarrow B(-s^2) = \frac{1}{1+s^4}$$



Left-half-plane poles are used for $H_L(s)$:

$$H_L(s) = \frac{1}{\left(s - e^{j\frac{3\pi}{4}}\right)\left(s - e^{-j\frac{3\pi}{4}}\right)} = \frac{1}{s^2 + \sqrt{2}\,s + 1}$$

$$H(z) = H_L(s)\Big|_{s=\cot\frac{\pi}{8}\frac{1-z^{-1}}{1+z^{-1}}}$$

$$= \frac{1}{(2.4142)^2 \dfrac{(1-z^{-1})^2}{(1+z^{-1})^2} + 2.4142\,\sqrt{2}\,\dfrac{1-z^{-1}}{1+z^{-1}} + 1}$$

$$= \frac{(1+z^{-1})^2}{(2.4142)^2(1-2z^{-1}+z^{-2}) + 2.4142\sqrt{2}(1-z^{-2}) + (1+2z^{-1}+z^{-2})}$$

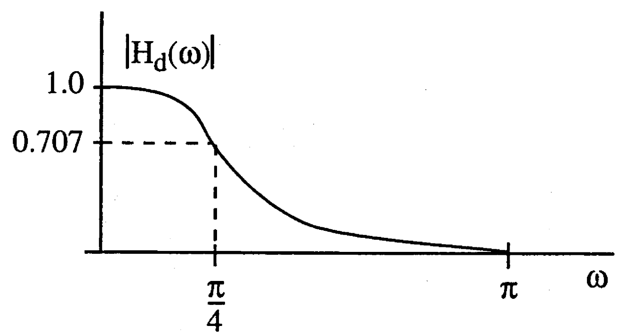
$$= \frac{1+2z^{-1}+z^{-2}}{10.23-9.66z^{-1}+3.41z^{-2}}$$

$$= \frac{0.098+0.196z^{-1}+0.098z^{-2}}{1-0.944z^{-1}+0.333z^{-2}}$$

Now, once again, from the shape of the analog prototype $H_a(\Omega)$ we know

$$H_d(0) = 1, \quad \left|H_d\left(\frac{\pi}{4}\right)\right| = \frac{1}{\sqrt{2}}, \quad |H_d(\pi)| = 0$$

However, for the second-order filter, the frequency response makes a sharper transition around $\omega_c = \frac{\pi}{4}$ and looks like

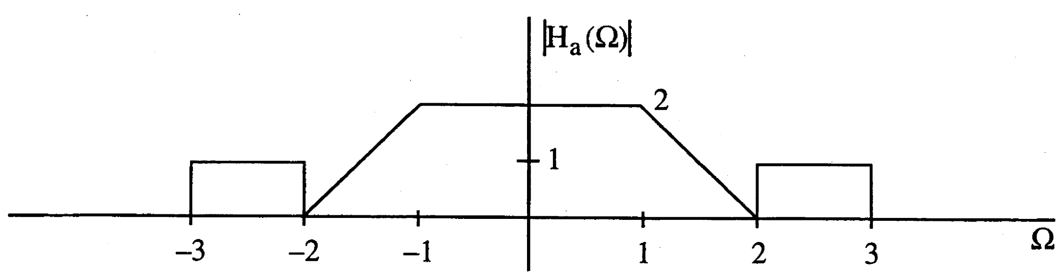


The phase $\angle H_d(\omega)$ will again be quite linear across the passband and more nonlinear across the stopband.

Let's try to get a better feel for the BLT mapping by considering one further example.

Example

Suppose that an analog prototype filter $H_L(s)$ has the frequency response



and that the bilinear transformation is used to produce a digital filter with $H(z) = H_L(s)\Big|_{s=\frac{1-z^{-1}}{1+z^{-1}}}$.

Sketch $|H_d(\omega)|$. Label all critical frequencies and amplitudes.

Solution

$H_d(\omega)$ is a squashed version of $H_a(\Omega)$, described by

$$H_d(\omega) = H_a\left(\alpha \tan\frac{\omega}{2}\right) = H_a\left(\tan\frac{\omega}{2}\right)$$

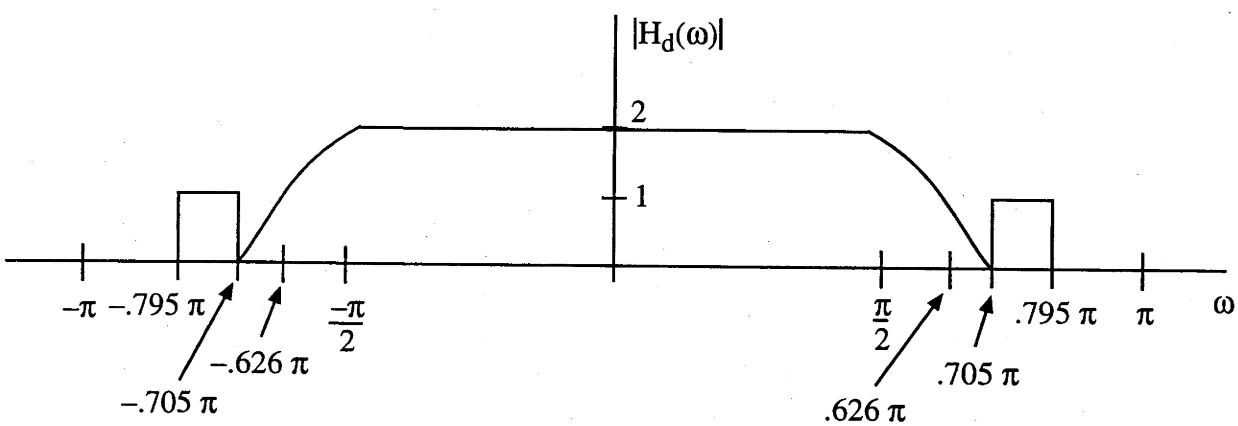
Thus, the value of $H_d(\omega)$ equals the value of $H_a(\Omega)$ at $\Omega = \tan\frac{\omega}{2}$. So a feature (e.g., jump) in $H_a(\Omega)$ that occurs at $\Omega = \Omega_0$ will occur in $H_d(\omega)$ at

$$\omega_0 = 2\tan^{-1}\Omega_0$$

The interesting features in $H_a(\Omega)$ occur at $\Omega_0 = 1, 2,$ and 3. In addition let's also consider $\Omega_0 = 1.5$. The corresponding ω_0 are given in the table below.

Ω_0	1	1.5	2	3
ω_0	$0.5\,\pi$	$0.626\,\pi$	$0.705\,\pi$	$0.795\,\pi$

Thus, $|H_d(\omega)|$ looks like



Notice that $H_d(\omega)$ takes on exactly the same set of values taken on by $H_a(\Omega)$. It just does so at different frequencies, which causes a change in shape.

2. Computer-Aided Optimization

Used for IIR designs with general magnitude and/or phase specifications (i.e., $H_d(\omega)$ not LPF, HPF, or BPF).

Can see Chapter 6, Programs for Digital Signal Processing, IEEE Press, 1979.

This book contains descriptions of several useful programs. Among these, the Deczky algorithm is best known. It approximates arbitrary magnitude and/or phase characteristics by minimizing $E + \hat{E}$ where

$$E = \int_0^{\pi} W(\omega)\left[|H_d(\omega)| - |D(\omega)|\right]^p \, d\omega$$

and

$$\hat{E} = \int_0^{\pi} \hat{W}(\omega)\left[\frac{d}{d\omega}\angle H_d(\omega) - \frac{d}{d\omega}\angle D(\omega)\right]^p \, d\omega$$

$W(\omega)$ and $\hat{W}(\omega)$ are user-specified weights.

p is chosen to be an even integer.

For large p, the solution will be equiripple (close for $p = 6$).

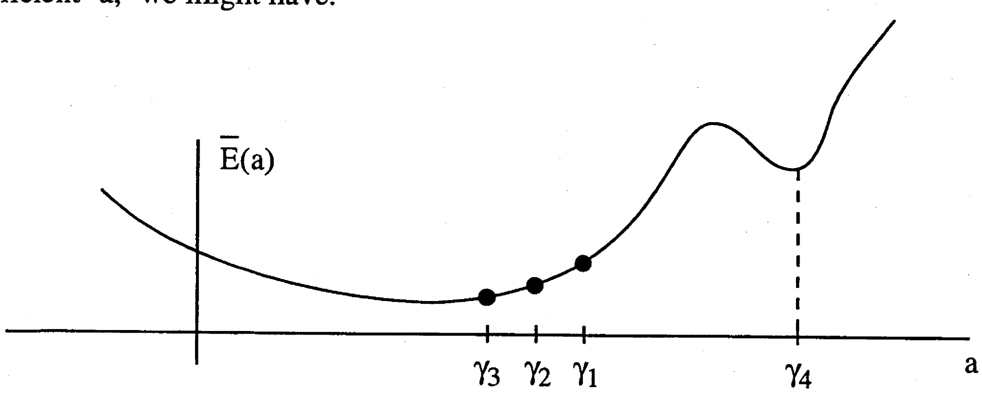
E and $\hat{E}$ are nonlinear functions of the filter coefficients.

Deczky uses the Fletcher-Powell gradient-based search.

Start with initial guess for $\{a_i\}$, $\{b_i\}$. Let $\bar{E} = E + \hat{E}$.

Compute $\left\{\frac{\partial \bar{E}}{\partial a_i}\right\}$, $\left\{\frac{\partial \bar{E}}{\partial b_i}\right\}$. Update coefficients.

The error function $\bar{E}$ is not convex. Thinking of $\bar{E}$ as a function of a single filter coefficient 'a,' we might have:



If you start with $a = \gamma_1$, then using derivative information will lead you to $a = \gamma_2$ and $a = \gamma_3$ in successive iterations. But, if you start at $a = \gamma_4$, then you get stuck in a local minimum. Global optimality is not guaranteed.

Program may require a lot of CPU time ~ especially for joint phase and magnitude specifications.

Digital Frequency Transformation

By making a "change of variable" in $H_L(s)$ or $H(z)$, we can transform an analog or digital LPF to a LPF having a different cutoff frequency, or to a HPF or BPF.

We will consider only digital transformations (transformations of H(z)).

Procedure:

Let H(z) be the transfer function of a low-pass digital filter. Then simply substitute for z^{-1} in H(z), using the expressions below, to produce filters having the described characteristics.

<u>Lowpass → Lowpass</u>

$$z^{-1} \rightarrow \frac{z^{-1} - \beta}{1 - \beta z^{-1}}$$

where

ω_c' = cutoff of new filter

$$\beta = \frac{\sin\left[(\omega_c - \omega_c')/2\right]}{\sin\left[(\omega_c + \omega_c')/2\right]}$$

<u>Lowpass → Highpass</u>

$$z^{-1} \rightarrow - \frac{z^{-1} - \beta}{1 - \beta z^{-1}}$$

ω_c' = cutoff of new filter

$$\beta = \frac{\cos\left[(\omega_c + \omega_c')/2\right]}{\cos\left[(\omega_c - \omega_c')/2\right]}$$

<u>Lowpass → Bandpass</u>

$$z^{-1} \rightarrow - \frac{z^{-2} - \beta_1 z^{-1} + \beta_2}{\beta_2 z^{-2} - \beta_1 z^{-1} + 1}$$

ω_ℓ = lower cutoff of BPF

ω_u = upper cutoff of BPF

$\beta_1 = 2\gamma K/(K+1)$

$$\beta_2 = (K-1)/(K+1)$$

$$\gamma = \frac{\cos\left[(\omega_u + \omega_\ell)/2\right]}{\cos\left[(\omega_u - \omega_\ell)/2\right]}$$

$$K = \cot\frac{\omega_u - \omega_\ell}{2}\tan\frac{\omega_c}{2}$$

ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 42

D. Munson

Relationship Between Pole and Zero Locations and Frequency Response

In digital filter design we choose the coefficients of

$$H(z) = \frac{a_0 + a_1 z^{-1} + \cdots + a_N z^{-N}}{1 + b_1 z^{-1} + \cdots + b_N z^{-N}}$$

to shape the frequency response $H_d(\omega) = H(e^{j\omega})$ in a desired way. Since $H(z)$ can also be written in terms of its poles and zeros as

$$H(z) = a_0 \prod_{i=1}^{N} \frac{z - z_i}{z - p_i}$$

this provides an alternative parameterization of $H(z)$. Choosing the pole and zero locations of the filter is basically equivalent to choosing the $\{a_i\}$ and $\{b_i\}$. In this connection, it is worth exploring how the pole and zero locations affect the shape of $H_d(\omega)$.

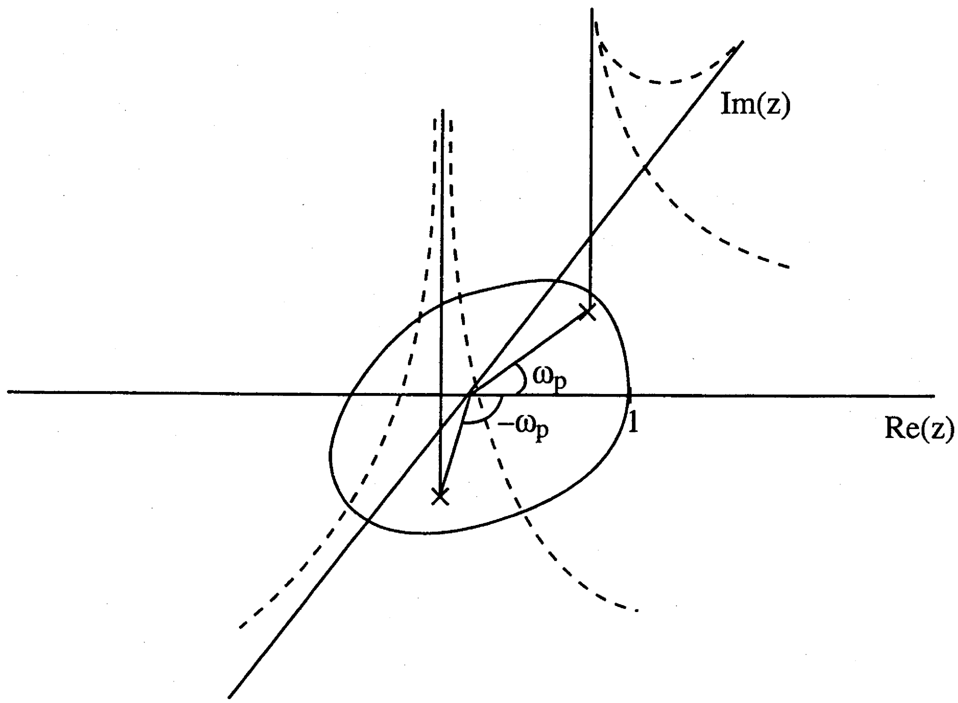
The general shape of $|H_d(\omega)|$ often can be visualized from knowledge of the pole and zero locations of $H(z)$. This is especially true for situations where poles are near the unit circle and zeros are either on or near the unit circle.

Example

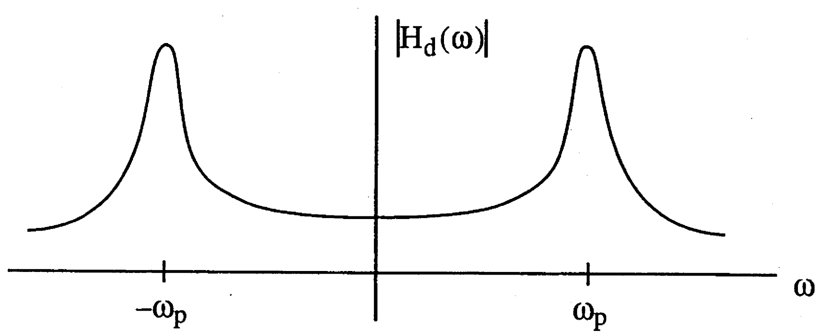
Consider a causal, stable all-pole filter with

$$H(z) = \frac{1}{z^2 - \left(2\alpha\cos\omega_p\right)z + \alpha^2} = \frac{1}{\left(z - \alpha e^{j\omega_p}\right)\left(z - \alpha e^{-j\omega_p}\right)}$$

and $0 < \alpha < 1$. $H(z)$ approaches ∞ at the pole locations and approaches zero for $|z|$ large. Thus, we expect $|H(z)|$ to look somewhat like a two-pole circus tent:



Here, the pole locations are marked by $\times$ and are at a distance α from the origin. Since H(z) is causal, its ROC is $\{z : |z| > \alpha\}$ and the above tent covers only this set of z. (The tent is undefined elsewhere.) Since $\alpha < 1$, ROC_H includes the unit circle. Thus, the frequency response $H_d(\omega) = H(e^{j\omega})$ is well defined and is a circular slice of the circus tent, around the unit circle. Now, since H(z) is infinite at $z = \alpha\, e^{\pm j\omega_p}$, we expect that H(z) will be large for z near the poles. If α is nearly one then the poles are close to the unit circle and $H_d(\omega)$ will be large for ω such that $e^{j\omega}$ is close to the poles. This suggests that $|H_d(\omega)|$ will look like

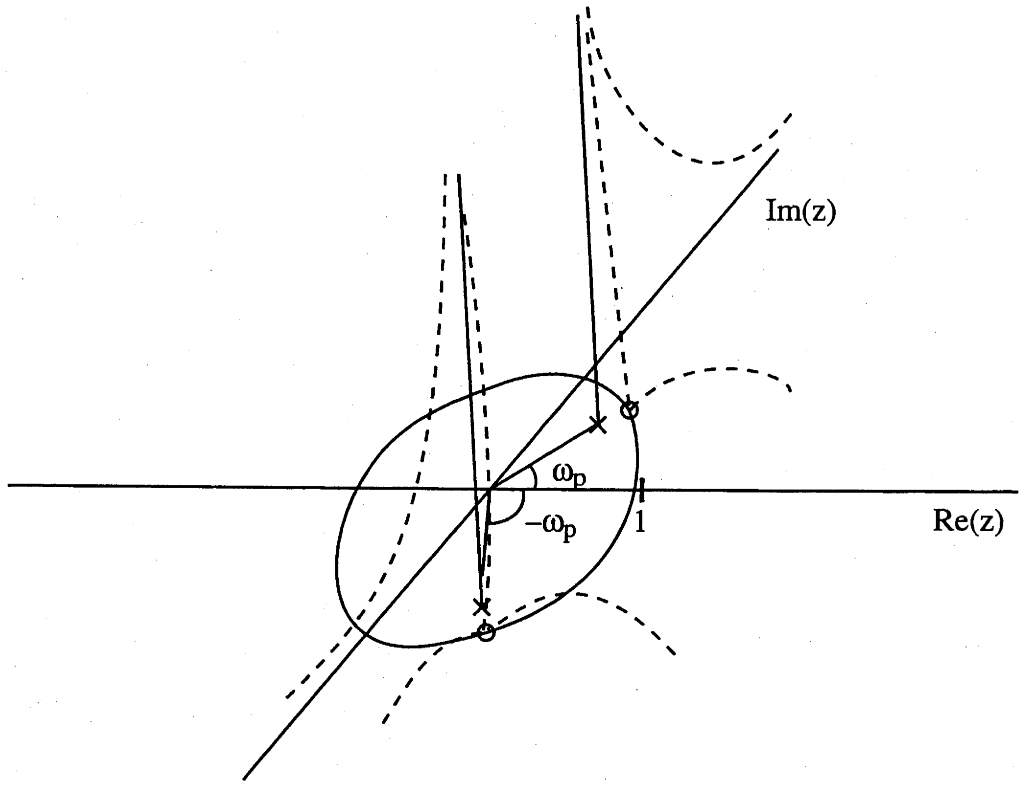


where the peaks occur near $\omega = \omega_p$.

Example

$$H(z) = \frac{z^2 - \left(2\cos\omega_p\right)z + 1}{z^2 - \left(2\alpha\cos\omega_p\right)z + \alpha^2} = \frac{\left(z - e^{j\omega_p}\right)\left(z - e^{-j\omega_p}\right)}{\left(z - \alpha e^{j\omega_p}\right)\left(z - \alpha e^{-j\omega_p}\right)}$$

As before, H(z) approaches ∞ at the pole locations $z = \alpha\, e^{\pm j\omega_p}$. H(z) = 0 at the zero locations $z = e^{\pm j\omega_p}$. Thus, $|H(z)|$ is similar to the previous two-pole circus tent except it "touches the ground" at the zero locations $z = e^{\pm j\omega_p}$ as shown below, where poles are indicated by $\times$ and zeros are represented by $\circ$.



Since the pole and zero locations are close together, it may seem difficult to determine what $H_d(\omega)$ = {H(z) for z on the unit circle} will look like. A simple observation helps greatly, though. Suppose

$$H(z) = \frac{(z - z_1)(z - z_2)}{(z - p_1)(z - p_2)} \qquad (\square)$$

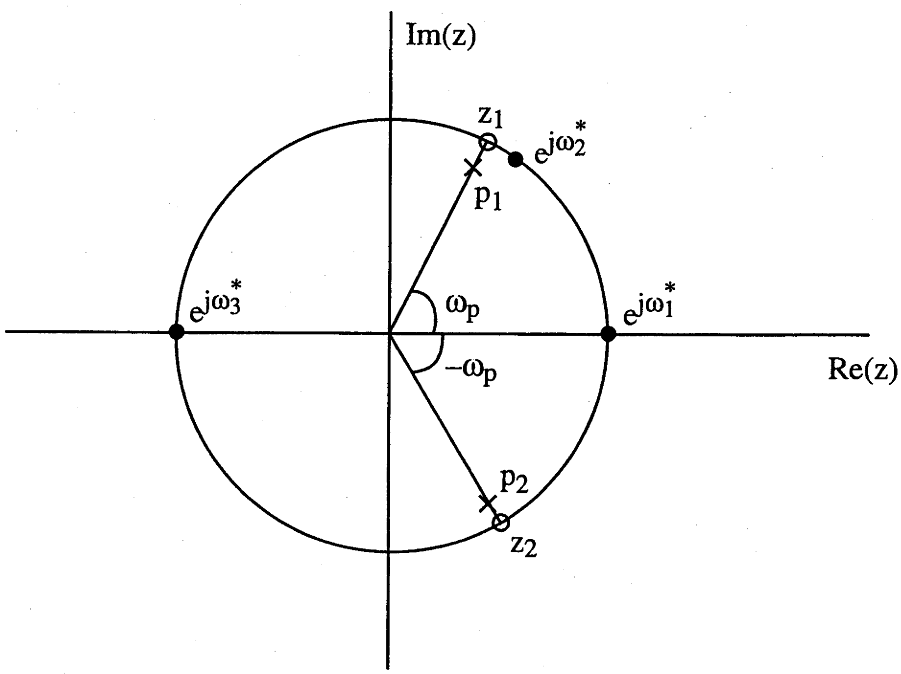
so that

$$|H_d(\omega)| = |H(e^{j\omega})| = \frac{|e^{j\omega} - z_1|\,|e^{j\omega} - z_2|}{|e^{j\omega} - p_1|\,|e^{j\omega} - p_2|}$$

Then, the value of $|H_d(\omega)|$ at some specific frequency ω^* is

$$\frac{|e^{j\omega^*} - z_1|\,|e^{j\omega^*} - z_2|}{|e^{j\omega^*} - p_1|\,|e^{j\omega^*} - p_2|}.$$

The factor $\left|e^{j\omega^*} - z_i\right|$ is the <u>distance</u> in the complex plane between $e^{j\omega^*}$ and z_i. Likewise, the factor $\left|e^{j\omega^*} - p_i\right|$ is the distance in the complex plane between $e^{j\omega^*}$ and p_i. Thus, $\left|H_d(\omega^*)\right|$ is the product of the distances between $e^{j\omega^*}$ and z_1, and between $e^{j\omega^*}$ and z_2, divided by the product of the distances between $e^{j\omega^*}$ and p_1, and between $e^{j\omega^*}$ and p_2. In our example, these distances can be visualized for three different frequencies ω_i^* by examining the figure below.



We see that the distances from $e^{j\omega_1^*}$ to the zero z_1 at $e^{j\omega_p}$ and to the neighboring pole p_1 are nearly the same. Likewise, the distances from $e^{j\omega_1^*}$ to the zero z_2 at $e^{-j\omega_p}$ and its neighboring pole p_2 are nearly the same. Thus,

$$\left|H_d\left(\omega_1^*\right)\right| = \frac{\left|e^{j\omega_1^*} - z_1\right|}{\left|e^{j\omega_1^*} - p_1\right|} = \frac{\left|e^{j\omega_1^*} - z_2\right|}{\left|e^{j\omega_1^*} - p_2\right|} \approx (1)(1) = 1$$

The same result holds for any ω_1^* that is close to zero.

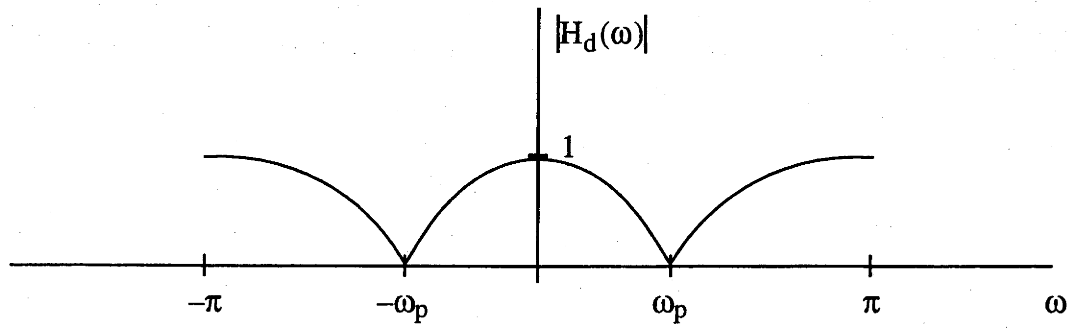
Similarly, in the case of $e^{j\omega_3^*}$ we see that distances to all poles and zeros are nearly equal. Thus.

$$\left|H_d\left(\omega_3^*\right)\right| \approx 1,$$

which holds for any ω_3^* roughly satisfying $\frac{\pi}{2} \le \left|\omega_3^*\right| \le \pi$. The distance between $e^{j\omega_2^*}$ and the zero at $e^{j\omega_p}$ approaches zero as $\omega_2^* \to \omega_p$. Thus

$$\left|H_d\left(\omega_2^*\right)\right| \approx 0$$

for ω_2^* close enough to ω_p. These considerations lead to the $|H_d(\omega)|$ sketched below.



Here, we can find the precise values of $|H_d(0)|$ and $|H_d(\pi)|$ by using

$$H_d(0) = H(1) = \frac{2 - 2\cos\omega_p}{1 + \alpha^2 - 2\alpha\cos\omega_p}$$

$$H_d(\pi) = H(-1) = \frac{2 + 2\cos\omega_p}{1 + \alpha^2 + 2\alpha\cos\omega_p}\ .$$

The above frequency response is that of a crude notch filter where signal components near $\omega = \omega_p$ are greatly attenuated and signal components at other frequencies are passed with nearly unit amplitude. The nulls (notches) in $H_d(\omega)$ at $\omega = \omega_p$ are caused by the zeros of H(z) at $z = e^{\pm j\omega_p}$.

It is only slightly more difficult to gain a rough idea of what $\angle H_d(\omega)$ will look like. From (❑) we have

$$\angle H_d(\omega) = \angle(e^{j\omega} - z_1) + \angle(e^{j\omega} - z_2) - \angle(e^{j\omega} - p_1) - \angle(e^{j\omega} - p_2)$$

Each term $(e^{j\omega} - z_i)$ or $(e^{j\omega} - p_i)$ is a vector in the complex plane. $\angle(e^{j\omega} - z_i)$ and $\angle(e^{j\omega} - p_i)$ are simply the angles of these vectors with respect to the positive real axis. This interpretation can be helpful when trying to visualize $\angle H_d(\omega)$ for low-order filters. In general, however, Matlab should be used to plot both $|H_d(\omega)|$ and $\angle H_d(\omega)$ for higher-order filters

Example

For an FIR lowpass filter, use Matlab to find the zero locations of H(z). (All poles are at $z = 0$.) You will find that zeros in $|H_d(\omega)|$ within the stopband are caused by zeros of H(z) on the unit circle. Other zeros of H(z) are strategically placed off the unit circle to give a flat response in the passband.

Example

For a Butterworth lowpass filter you will find that some poles are located near the unit circle $e^{j\omega}$ for ω corresponding to the cutoff frequency.

Example

For an elliptic lowpass filter you will find poles near $e^{j\omega}$ for ω = cutoff frequency, and some zeros <u>on</u> the unit circle at locations corresponding to the stopband.

ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 43

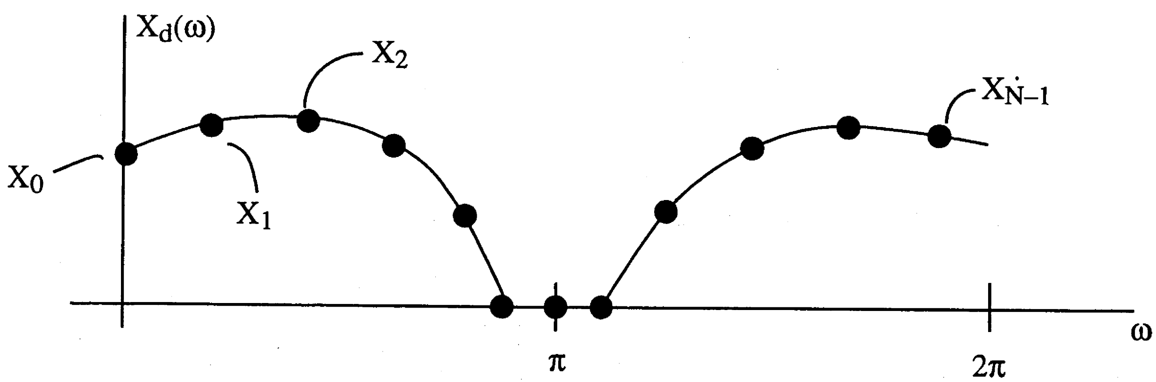
D. Munson

Discrete Fourier Transform (DFT)

The DFT of $\{x_n\}_{n=0}^{N-1}$ is a set of evenly spaced samples of $X_d(\omega)$ on $\omega \varepsilon\ [0, 2\pi)$, and is denoted by $\{X_m\}_{m=0}^{N-1}$. Specifically,

$$\boxed{X_m = X_d\left(\frac{2\pi}{N}m\right) = \sum_{n=0}^{N-1} x_n\, e^{-j\frac{2\pi}{N}mn} \qquad m=0, 1, \ldots, N-1} \tag{1}$$

Picture:



Note that the last DFT sample, X_{N-1} is not at $\omega = 2\pi$. It is to the left of $\omega = 2\pi$.

Inverse DFT:

$$\boxed{x_n = \frac{1}{N} \sum_{m=0}^{N-1} X_m\, e^{j\frac{2\pi}{N}nm} \qquad n=0, 1, \ldots, N-1}$$

Proof of DFT^{-1}:

$$\frac{1}{N}\sum_{m=0}^{N-1} X_m\, e^{j\frac{2\pi}{N}nm} = \frac{1}{N}\sum_{m=0}^{N-1}\sum_{\ell=0}^{N-1} x_\ell\, e^{-j\frac{2\pi}{N}\ell m}\, e^{j\frac{2\pi}{N}nm}$$

$$= \frac{1}{N} \sum_{\ell=0}^{N-1} x_\ell \sum_{m=0}^{N-1} e^{j\frac{2\pi}{N}m(n-\ell)} \qquad (*)$$

$$\text{Second sum} = \begin{cases} N & \ell = n \\ \dfrac{1-e^{j\frac{2\pi}{N}(n-\ell)N}}{1-e^{j\frac{2\pi}{N}(n-\ell)}} = 0 & \ell \neq n \end{cases}$$

So, the second sum can be written as $N\delta_{\ell-n}$. Substituting into (*) gives

$$\frac{1}{N}\sum_{m=0}^{N-1} X_m \, e^{j\frac{2\pi}{N}nm} = \frac{1}{N}\sum_{\ell=0}^{N-1} x_\ell N \delta_{\ell-n}$$

$$= x_n \quad ✔$$

Properties of the DFT

1) If $\{x_n\}_{n=0}^{N-1}$ is real, then

a) $|X_m| = |X_{<N-m>_N}| \qquad 0 \le m \le N-1 \qquad$ (magnitude of $\{X_m\}$ is even)

b) $\angle X_m = -\angle X_{<N-m>_N} \qquad 0 \le m \le N-1 \qquad$ (angle of $\{X_m\}$ is odd)

where $<k>_N = k \bmod N$, i.e.,

$<k>_N = r \quad \text{if} \quad k = \ell N + r \quad \text{with} \quad 0 \le r \le N-1$

e.g.,

$$<7>_4 = 3, \; <4>_4 = 0, \; \underset{-2(4)+3}{\underset{\uparrow}{<-5>_4 = 3}}, \; \underset{-1(4)+2}{\underset{\uparrow}{<-2>_4 = 2}}$$

Note: It is sometimes helpful to use $<-k>_N = N - <k>_N$.

In the above DFT property, computing the index modulo N makes a difference only for $m = 0$ since $<N-m>_N = N - m$ for $1 \le m \le N-1$, and $<N-0>_N = 0$.

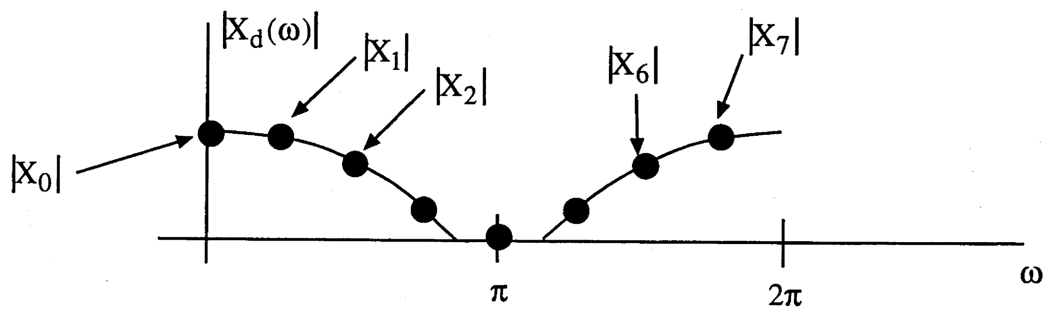
Proof: Follows from $X_m = X_d\left(\frac{2\pi m}{N}\right)$ and

$$|X_d(\omega)| \underset{\substack{\uparrow \\ \text{even}}}{=} |X_d(-\omega)| \underset{\substack{\uparrow \\ \text{periodic}}}{=} |X_d(2\pi - \omega)|$$

$$\angle X_d(\omega) \underset{\substack{\uparrow \\ \text{odd}}}{=} -\angle X_d(-\omega) \underset{\substack{\uparrow \\ \text{periodic}}}{=} -\angle X_d(2\pi-\omega)$$

Example

Suppose have real-valued $\{x_n\}_{n=0}^{7}$. Then $|X_m|, 0 \leq m \leq 7$ might look like the DTFT samples below



Obviously:

$$|X_1| = |X_7|$$

$$|X_2| = |X_6|$$

etc.

Similarly:

$$\angle X_1 = -\angle X_7$$
$$\angle X_2 = -\angle X_6$$

etc.

2) Time Shift

$$\text{DFT}^{-1}\left[\left\{X_m\, e^{-j\frac{2\pi}{N}mk}\right\}_{m=0}^{N-1}\right] = \left\{x_{<n-k>_N}\right\}_{n=0}^{N-1}$$

This property states that multiplication of a DFT X_m by a complex exponential $e^{-j\frac{2\pi}{N}mk}$ corresponds to a circular shift by an amount k.

Proof:

$$\frac{1}{N}\sum_{m=0}^{N-1} X_m\, e^{-j\frac{2\pi}{N}mk}\, e^{j\frac{2\pi}{N}nm} = \frac{1}{N}\sum_{m=0}^{N-1} X_m\, e^{j\frac{2\pi}{N}m(n-k)}$$

$$= \frac{1}{N}\sum_{m=0}^{N-1} X_m\, e^{j\frac{2\pi}{N}m<n-k>_N} \qquad (*)$$

$$= x_{<n-k>_N} \qquad 0 \le n \le N-1$$

since if $(n-k) = q\,N + r,\quad 0 \le r \le N-1$

$$\Rightarrow e^{j\frac{2\pi}{N}m(n-k)} = e^{j\frac{2\pi}{N}m(qN+r)} = e^{j\frac{2\pi}{N}mr} = e^{j\frac{2\pi}{N}m<n-k>_N}$$

Note: The step in Eq. (*) is necessary because x_{n-k} is not defined outside the range $0 \le n - k \le N - 1$.

Example

Let $\{x_n\}_{n=0}^{4} = \{1, 3, 5, 7, 9\}$

Then

(k= 1) $\mathrm{DFT}^{-1}\left[\left\{X_m\, e^{-j\frac{2\pi}{5}m}\right\}\right] = \{9, 1, 3, 5, 7\}$

(k = 2) $\mathrm{DFT}^{-1}\left[\left\{X_m\, e^{-j\frac{2\pi}{5}2m}\right\}\right] = \{7, 9, 1, 3, 5\}$

(k = 3) $\mathrm{DFT}^{-1}\left[\left\{X_m\, e^{-j\frac{2\pi}{5}3m}\right\}\right] = \{5, 7, 9, 1, 3\}$

etc.

3) $\mathrm{DFT}\left[\{X_m\}_{m=0}^{N-1}\right] = \left\{N \bullet x_{<N-n>_N}\right\}_{n=0}^{N-1}$

This DFT property states that applying a forward DFT twice gives $\{x_n\}$ back but scaled by N and flipped around (except the n = 0 term x_0).

Example

Let $\{X_m\}_{m=0}^{5}$ be the DFT of

$$\{x_n\}_{n=0}^{5} = \{2, 4, 6, 8, 10, 12\}.$$

Then DFT $\left[\{X_m\}_{m=0}^{5}\right] = 6 \bullet \{2, 12, 10, 8, 6, 4\} = \{12, 72, 60, 48, 36, 24\}$.

This property gives a way to compute an inverse DFT using a forward DFT algorithm, which is handy on some rare occasions.

<u>Proof of Property 3)</u>:

Let $\{y_n\}_{n=0}^{N-1} = \text{DFT}\left[\{X_m\}_{m=0}^{N-1}\right]$

$$\Rightarrow \qquad y_n = \sum_{m=0}^{N-1} X_m \, e^{-j\frac{2\pi}{N}nm}$$

$$= \sum_{m=0}^{N-1} \sum_{\ell=0}^{N-1} x_\ell \, e^{-j\frac{2\pi}{N}m\ell} \, e^{-j\frac{2\pi}{N}nm}$$

$$= \sum_{\ell=0}^{N-1} x_\ell \sum_{m=0}^{N-1} e^{-j\frac{2\pi}{N}m(\ell+n)}$$

$$= \sum_{\ell=0}^{N-1} x_\ell \sum_{m=0}^{N-1} e^{-j\frac{2\pi}{N}m<\ell+n>_N}$$

But, the second sum above can be written as

$$\sum_{m=0}^{N-1} e^{-j\frac{2\pi}{N}m<\ell+n>_N} = \begin{cases} N & <\ell+n>_N = 0 \\ 0 & <\ell+n>_N \neq 0 \end{cases}$$

Since $\ell \in \{0, 1, \ldots, N-1\}$ we have $<\ell+n>_N = 0$ if $\ell = <-n>_N = <N-n>_N$.

Thus, the second sum can be written as $N \bullet \delta_{\ell-<N-n>_N}$ so that

$$y_n = \sum_{\ell=0}^{N-1} x_\ell \, N \bullet \delta_{\ell-<N-n>_N}$$

$$= N\, x_{<N-n>_N} \qquad 0 \le n \le N-1 \quad ✔$$

4) Parseval

$$\sum_{m=0}^{N-1} |X_m|^2 = N \bullet \sum_{n=0}^{N-1} |x_n|^2$$

This property shows how to compute the energy of a sequence in terms of its DFT.

<u>Proof</u>:

$$\sum_{m=0}^{N-1} |X_m|^2 = \sum_{m=0}^{N-1} X_m X_m^*$$

$$= \sum_{m=0}^{N-1} X_m \left[\sum_{n=0}^{N-1} x_n \, e^{-j\frac{2\pi}{N}nm} \right]^*$$

$$= \sum_{n=0}^{N-1} x_n^* \sum_{m=0}^{N-1} X_m \, e^{j\frac{2\pi}{N}nm}$$

$$= \sum_{n=0}^{N-1} x_n^* \bullet N \bullet x_n$$

$$= N \sum_{n=0}^{N-1} |x_n|^2 \quad ✔$$

5) Cyclic Convolution

$$C_m = A_m \bullet B_m \qquad 0 \le m \le N-1$$

if and only if

$$c_n = \sum_{\ell=0}^{N-1} a_\ell \, b_{<n-\ell>_N}$$

This summation is called a <u>cyclic convolution</u>. It will be described later in great detail. It differs from the regular "linear convolution" because taking the index $n-\ell$ modulo N causes the b_ℓ

sequence to "wrap back around" the a_ℓ sequence in a circular fashion. Our interest in this property arises because it suggests that a <u>cyclic</u> convolution can be computed as

$$\{c_n\} = \text{DFT}^{-1}\left\{\left[\text{DFT}\{a_n\}\right]\bullet\left[\text{DFT}\{b_n\}\right]\right\}. \qquad (*)$$

Later we will study extremely fast algorithms for computing DFTs and inverse DFTs, and we will also show that by zero-padding $\{a_n\}$ and $\{b_n\}$, a cyclic convolution can produce a linear convolution. Together, these results will provide a fast method for performing regular, linear convolution.

<u>Proof of Cyclic Convolution Property</u>

Show that (*) results in a cyclic convolution:

$$\begin{aligned}
\{c_n\} &= \text{DFT}^{-1}\left[\{A_m B_m\}_{m=0}^{N-1}\right] \\
&= \frac{1}{N}\sum_{m=0}^{N-1}\left(\sum_{k=0}^{N-1} a_k\, e^{-j\frac{2\pi}{N}mk}\right) B_m\, e^{j\frac{2\pi}{N}mn} \\
&= \sum_{k=0}^{N-1} a_k \frac{1}{N}\sum_{m=0}^{N-1} B_m\, e^{j\frac{2\pi}{N}m(n-k)} \\
&= \sum_{k=0}^{N-1} a_k \underbrace{\frac{1}{N}\sum_{m=0}^{N-1} B_m\, e^{j\frac{2\pi}{N}m\langle n-k\rangle_N}}_{b_{\langle n-k\rangle_N}} \\
&= \sum_{k=0}^{N-1} a_k\, b_{\langle n-k\rangle_N} \qquad 0 \le n \le N-1 \qquad ✔
\end{aligned}$$

Note: We needed the next-to-last step, where we inserted the modulo notation, because b_ℓ is defined only for $0 \le \ell \le N-1$.

ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 44

D. Munson

Frequency Sampling FIR Filter Design

Given $G_d(\omega)$ ~ desired frequency response with linear phase

Idea: Choose $H_d(\omega)$ to agree with $G_d(\omega)$ at $\omega = \frac{2\pi}{N}m,\ 0 \le m \le N-1$.

Take $\{h_n\}_{n=0}^{N-1}$ to be $\mathrm{DFT}^{-1}\left[\left\{G_d\left(\frac{2\pi m}{N}\right)\right\}_{m=0}^{N-1}\right]$, i.e.,

$$\boxed{h_n = \frac{1}{N}\sum_{m=0}^{N-1} G_d\left(\frac{2\pi}{N}m\right)e^{j\frac{2\pi}{N}nm} \quad 0 \le n \le N-1} \qquad \text{(FS)}$$

Using $\{h_n\}_{n=0}^{N-1}$ as our FIR filter coefficients will give a frequency response $H_d(\omega)$ satisfying

$$\underset{\text{actual}}{H_d(\omega)} = \underset{\text{desired}}{G_d(\omega)} \text{ at } \omega = \frac{2\pi m}{N} \qquad 0 \le m \le N-1$$

Why?

$$H_d(\omega)\Big|_{\omega=\frac{2\pi}{N}m} = \mathrm{DFT}\left[\{h_n\}\right]$$

$$\underset{\text{by (FS)}}{=} \mathrm{DFT}\left[\mathrm{DFT}^{-1}\left[\left\{G_d\left(\frac{2\pi}{N}m\right)\right\}\right]\right]$$

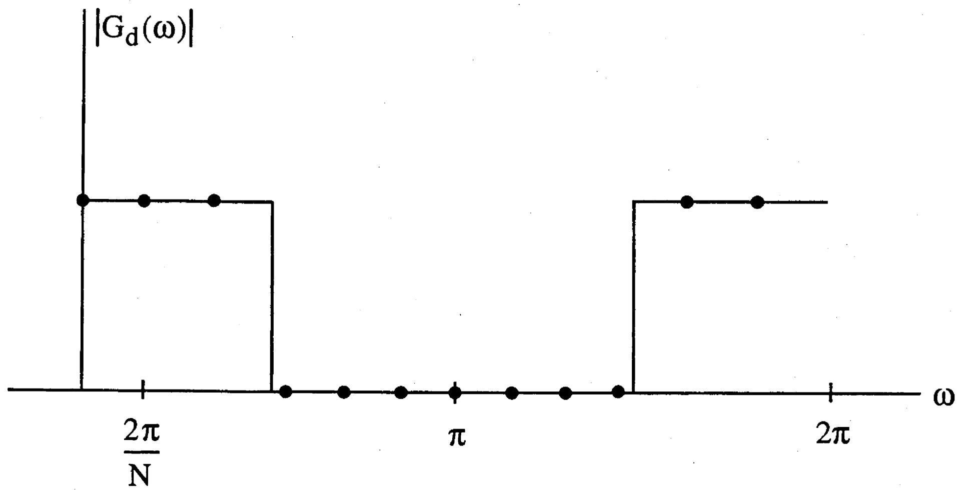
$$= G_d\left(\frac{2\pi}{N}m\right) \qquad ✔$$

<u>But</u> in general $H_d(\omega) \ne G_d(\omega)$ for $\omega \ne \frac{2\pi}{N}m$.

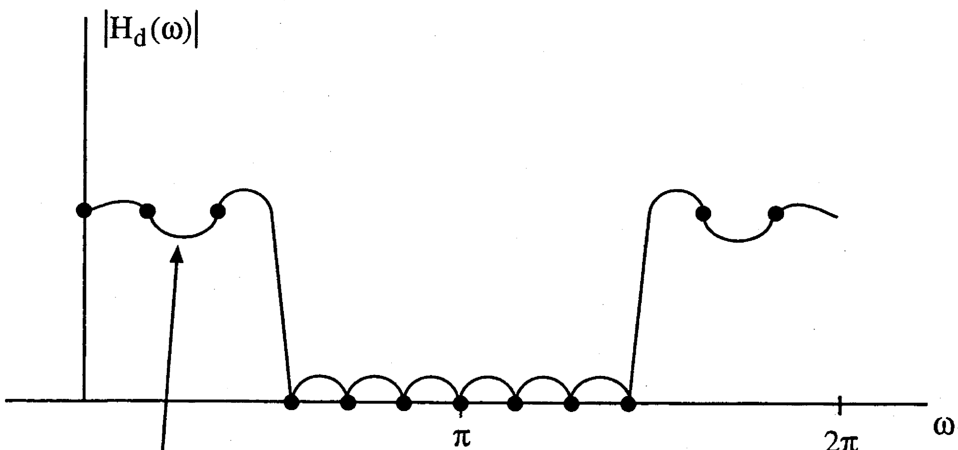
This approach <u>can be</u> too simplistic.

Example

Let $G_d(\omega)$ be an ideal LPF with linear phase and delay $\frac{N-1}{2}$.

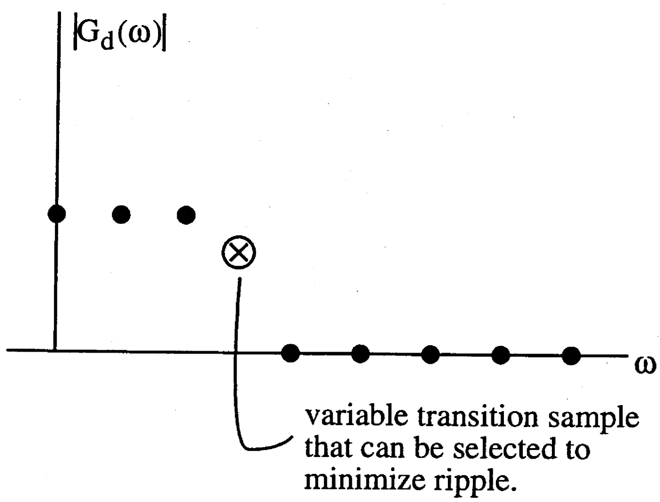


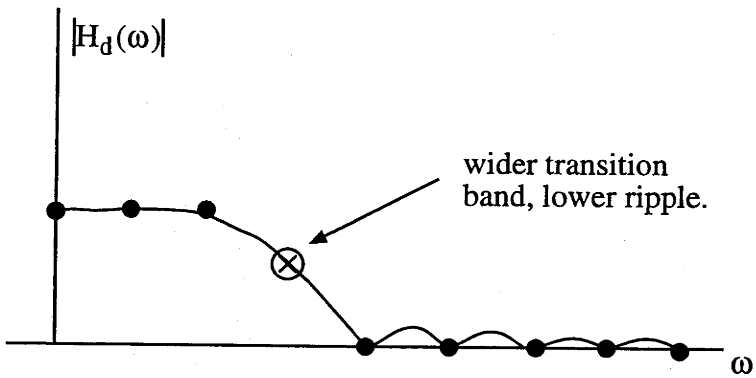
Using (FS), will produce an $H_d(\omega)$ that agrees with $G_d(\omega)$ at the points shown above. The resulting $H_d(\omega)$ might look like:



agrees with $G_d(\omega)$ at uniformly spaced points, but can have large ripple.

There is a modified frequency sampling design procedure that works better. In the modified procedure, one or more frequency samples in the filter's transition band are left unconstrained as free variables. The values of these free variables are then selected via linear programming to minimize some measure of ripple. This results in a wider transition band, but lower ripple, as suggested in the figures below.





Note: If you take $G_d(\omega)$ to have zero phase, the designed H_d will still pass through the correct samples of G_d, but the response will be terrible off the grid $\omega = \frac{2\pi m}{N}$. Hoping for zero phase (no delay) in a causal filter is a pipe dream!

Example

Use the frequency sampling method to design a linear-phase HPF $\{h_n\}_{n=0}^{60}$, with cutoff $\omega_c = .6\pi$.

Solution

$$N = 61 \Rightarrow \text{phase} = -\frac{N-1}{2}\,\omega = -30\,\omega$$

$$\Rightarrow \quad G_d(\omega) = \begin{cases} 0 & 0 \le \omega < .6\pi \\ e^{-j30\omega} & .6\pi \le \omega \le 1.4\pi \\ 0 & 1.4\pi < \omega \le 2\pi \end{cases}$$

$$\Rightarrow \quad G_d\left(\frac{2\pi m}{61}\right) = \begin{cases} 0 & 0 \le m \le 18 \\ e^{-j30\frac{2\pi}{61}m} & 19 \le m \le 42 \\ 0 & 43 \le m \le 60 \end{cases}$$

Now, $\{h_n\}$ is the inverse DFT of $\left\{G_d\left(\frac{2\pi m}{61}\right)\right\}$ as given by

$$h_n = \frac{1}{61}\sum_{m=0}^{60} G_d\left(\frac{2\pi m}{61}\right) e^{j\frac{2\pi}{61}nm}$$

$$= \frac{1}{61}\sum_{m=19}^{42} e^{-j30\frac{2\pi}{61}m}\, e^{j\frac{2\pi}{61}nm}$$

$$= \frac{1}{61}\sum_{m=19}^{42} e^{j\frac{2\pi}{61}(n-30)m}$$

effect of linear phase is to circularly shift the $\{h_n\}$ since replacing $n-30$ with $\langle n-30\rangle_{61}$ does not change this equation

$$= \frac{1}{61}\sum_{k=0}^{23} e^{j\frac{2\pi}{N}(n-30)(k+19)}$$

$k = m - 19$

$$= \frac{1}{61}\, e^{j\frac{2\pi}{61}(n-30)19}\, \frac{1-e^{j\frac{2\pi}{61}(n-30)24}}{1-e^{j\frac{2\pi}{61}(n-30)}}$$

$$= \frac{\frac{1}{61}\, e^{j\frac{2\pi}{61}(n-30)31}}{e^{j\frac{2\pi}{61}(n-30)}} \bullet \frac{e^{-j\frac{2\pi}{61}(n-30)12} - e^{j\frac{2\pi}{61}(n-30)12}}{e^{-j\frac{\pi}{61}(n-30)} - e^{j\frac{\pi}{61}(n-30)}}$$

$$= \frac{1}{61}\, e^{j\pi(n-30)}\, \frac{-2j\sin\left(\frac{2\pi}{61}(n-30)12\right)}{-2j\sin\left(\frac{\pi}{61}(n-30)\right)}$$

$$= \boxed{\frac{1}{61}(-1)^n \underbrace{\frac{\sin\left(24\frac{\pi}{61}(n-30)\right)}{\sin\left(\frac{\pi}{61}(n-30)\right)}}_{\text{sampled periodic sinc}} \qquad 0 \le n \le 60}$$

Here, the factor $(-1)^n$ makes this a high-pass filter rather than a low-pass filter. Since this term multiplies a periodic sinc, which is similar to a sinc, we expect that the resulting frequency response will not be too different from that obtained using the window design procedure with a truncation window.

Let's next consider the design of a low-pass filter $\{h_n\}_{n=0}^{N-1}$ where N is even. This example will be more complicated than the previous one in two respects. First, since N is even, the formula for $G_d(\omega)$ will take two different forms on $0 \le \omega < 2\pi$. Second, since this is a low-pass filter, there will be two separate bands where $G_d(\omega)$ is nonzero on $0 \le \omega < 2\pi$.

Example

Frequency sampling design of $\{h_n\}_{n=0}^{21}$, linear-phase LPF with $\omega_c = \frac{\pi}{2}$.

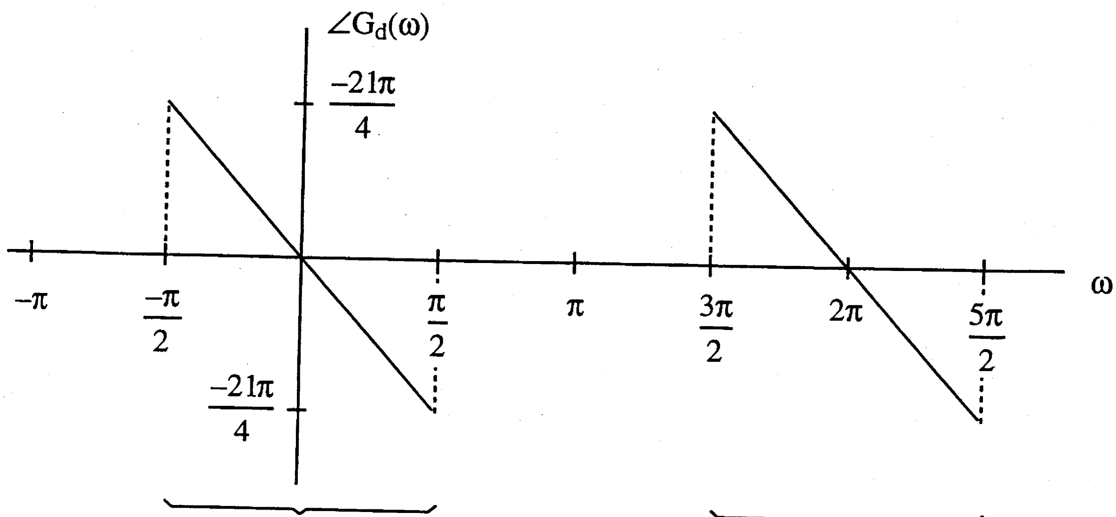
<u>Solution</u>

$$N = 22 \Rightarrow \text{phase} = -\frac{N-1}{2}\omega = \frac{-21}{2}\omega$$

$$\Rightarrow \quad G_d(\omega) = \begin{cases} e^{-j\frac{21}{2}\omega} & |\omega| \le \frac{\pi}{2} \\ 0 & \frac{\pi}{2} < |\omega| \le \pi \end{cases}$$

But, for a frequency sampling design, we need samples of $G_d(\omega)$ on $\omega \varepsilon [0, 2\pi)$. For N even, with $\frac{N-1}{2}$ noninteger, we must be careful! Consider the phase of $G_d(\omega)$. Here for clarity, we will not wrap the phase inside the interval $(-\pi, \pi)$ and will instead show its linear extension.

44.6



$G_d(\omega) = e^{-j\frac{21}{2}\omega}$ in here

$$G_d(\omega) = e^{-j\frac{21}{2}(\omega-2\pi)}$$

$$= e^{j\pi}\, e^{-j\frac{21}{2}\omega}$$

$$= -e^{-j\frac{21}{2}\omega}$$

in here

Note: Minus sign doesn't occur if N is odd.

So:

$$G_d(\omega) = \begin{cases} e^{-j\frac{21}{2}\omega} & 0 \le \omega \le \frac{\pi}{2} \\ 0 & \frac{\pi}{2} < \omega < \frac{3\pi}{2} \\ -e^{-j\frac{21}{2}\omega} & \frac{3\pi}{2} \le \omega \le 2\pi \end{cases}$$

$$\Rightarrow G_d\left(\frac{2\pi m}{22}\right) = \begin{cases} e^{-j\frac{21}{2}\left(\frac{2\pi m}{22}\right)} & 0 \le m \le 5 \\ 0 & 6 \le m \le 16 \\ -e^{-j\frac{21}{2}\left(\frac{2\pi m}{22}\right)} & 17 \le m \le 21 \end{cases}$$

Thus,

$$h_n = \frac{1}{22}\sum_{m=0}^{5} e^{-j\frac{21}{2}\left(\frac{2\pi m}{22}\right)} e^{j\frac{2\pi}{22}mn} + \frac{1}{22}\sum_{m=17}^{21} - e^{-j\frac{21}{2}\left(\frac{2\pi m}{22}\right)} e^{j\frac{2\pi}{22}mn}$$

$$= \frac{1}{22}\sum_{m=0}^{5} e^{j\frac{2\pi}{22}\left(n-\frac{21}{2}\right)m} - \frac{1}{22}\sum_{m=17}^{21} e^{j\frac{2\pi}{22}\left(n-\frac{21}{2}\right)m}$$

Making the change of variable $\ell = m - 22$, the second sum can be rewritten as

$$\sum_{\ell=-5}^{-1} e^{j\frac{2\pi}{22}\left(n-\frac{21}{2}\right)(\ell+22)} = \sum_{\ell=-5}^{-1} e^{j\frac{2\pi}{22}\left(n-\frac{21}{2}\right)\ell} \underbrace{e^{-j21\pi}}_{=-1}$$

Thus,

$$h_n = \frac{1}{22}\sum_{m=-5}^{5} e^{j\frac{2\pi}{22}\left(n-\frac{21}{2}\right)m}$$

$$\underset{k=m+5}{=} \frac{1}{22}\sum_{k=0}^{10} e^{j\frac{2\pi}{22}\left(n-\frac{21}{2}\right)(k-5)}$$

$$= \frac{1}{22} e^{j\frac{2\pi}{22}\left(n-\frac{21}{2}\right)(-5)} \frac{1-e^{j\frac{2\pi}{22}\left(n-\frac{21}{2}\right)11}}{1-e^{j\frac{2\pi}{22}\left(n-\frac{21}{2}\right)}}$$

$$= \frac{1}{22} e^{-j\frac{10\pi}{22}\left(n-\frac{21}{2}\right)} \frac{e^{j\frac{2\pi}{22}\left(n-\frac{21}{2}\right)\frac{11}{2}}}{e^{j\frac{2\pi}{22}\left(n-\frac{21}{2}\right)\frac{1}{2}}} \bullet \frac{-j2\sin\frac{11\pi}{22}\left(n-\frac{21}{2}\right)}{-j2\sin\frac{\pi}{22}\left(n-\frac{21}{2}\right)}$$

$$= \boxed{\frac{1}{22}\frac{\sin\frac{\pi}{2}\left(n-\frac{21}{2}\right)}{\sin\frac{\pi}{22}\left(n-\frac{21}{2}\right)} \qquad 0 \le n \le 21}$$

Thus, h_n is a sampled, periodic sinc, which is similar to the sinc-type design that would have resulted from a window design using a truncation window.

In general, the formula for the coefficients of either an FIR LPF or HPF, designed using the frequency sampling technique, are given by

$$\text{LPF:} \quad h_n = \frac{1}{N} \frac{\sin\left(\gamma \frac{\pi}{N}\left(n - \frac{N-1}{2}\right)\right)}{\sin\left(\frac{\pi}{N}\left(n - \frac{N-1}{2}\right)\right)}$$

$$\text{HPF:} \quad h_n = (-1)^n \frac{1}{N} \frac{\sin\left(\gamma \frac{\pi}{N}\left(n - \frac{N-1}{2}\right)\right)}{\sin\left(\frac{\pi}{N}\left(n - \frac{N-1}{2}\right)\right)}$$

where γ is the number of samples $G_d\left(\frac{2\pi m}{N}\right)$, $0 \le m \le N-1$, that are nonzero (number of samples in the passband).

ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 45

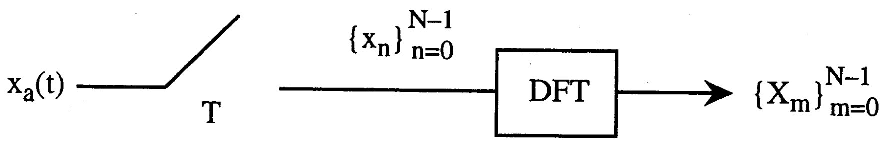
D. Munson

DFT Spectral Analysis

Problem: Given $x_a(t)$, compute approximate samples of $X_a(\Omega)$.

Proposed scheme:

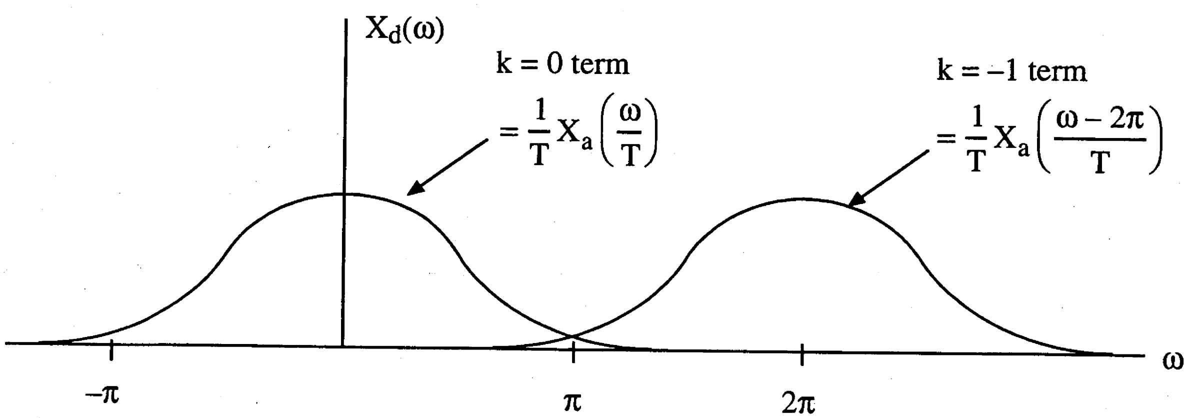
Here we will assume $x_a(t)$ has finite support on [0, (N–1) T] and is <u>nearly</u> bandlimited.



In the Fourier domain we know:

$$X_d(\omega) = \frac{1}{T}\sum_{k=-\infty}^{\infty} X_a\left(\frac{\omega + 2\pi k}{T}\right)$$

If $x_a(t)$ is <u>nearly</u> bandlimited and T is small enough, there is little aliasing so that:



The DFT, $\{X_m\}_{m=0}^{N-1}$ is a set of samples of $X_d(\omega)$ on $[0, 2\pi)$: $X_m = X_d\left(\frac{2\pi m}{N}\right), 0 \le m \le N-1.$

Thus, we have (for N odd):

$$\Rightarrow X_m \approx \begin{cases} \dfrac{1}{T} X_a\left(\dfrac{2\pi m}{NT}\right) & 0 \le m \le \dfrac{N-1}{2} \\ \dfrac{1}{T} X_a\left(\dfrac{2\pi(m-N)}{NT}\right) & \dfrac{N-1}{2} < m \le N-1 \end{cases}$$

So, $\{X_m\}_{m=0}^{N-1}$ are approximate, scaled samples of $X_a(\Omega)$.

It is important to note that the first half of the DFT gives samples of $X_a(\Omega)$ for $\Omega > 0$, while the second half of the DFT gives samples of $X_a(\Omega)$ for $\Omega < 0$. This peculiarity arises due to the definition of the DFT: $\{X\}_{m=0}^{N-1}$ is a set of samples of $X_d(\omega)$ on $[0, 2\pi)$ rather than on $[-\pi, \pi]$.

DFT Spectral Analysis of Sinusoids

Suppose that

$$x_a(t) = \sum_{i=1}^{M} A_i \cos(\Omega_i t)$$

and we have available only $x_n = x_a(nT)$, $0 \le n \le N-1$.

From $\{x_n\}_{n=0}^{N-1}$, we wish to determine M, $\{\Omega_i\}$, and $\{A_i\}$.

One approach to this problem is to use DFT spectral analysis.

Application areas: This problem arises in numerous applications, especially involving rotating machinery. For example, an acoustic transducer coupled to a piece of rotating machinery will output a periodic signal (sum of sinusoids) plus, perhaps, a smaller nonperiodic component. A frequency (i.e., DFT) analysis of this signal can indicate whether the machinery requires maintenance or replacement. Similarly, in an underwater setting, ships can be identified through DFT analysis of acoustic signals they emit, which are collected by hydrophones.

Consider a single sinusoidal component:

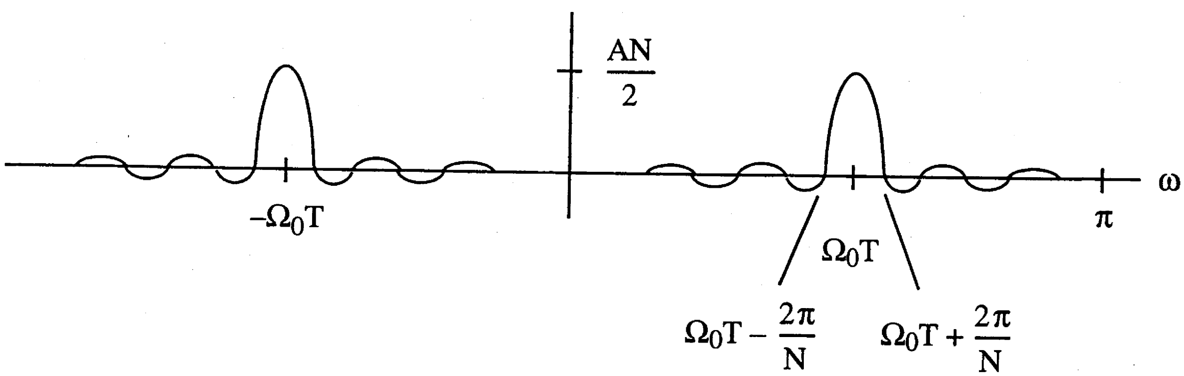
$$x_n = A \cos \Omega_0 nT \qquad 0 \le n \le N-1$$

The DFT of $\{x_n\}_{n=0}^{N-1}$ is a set of samples of the DTFT:

$$X_d(\omega) = \sum_{n=0}^{N-1} A \cos \Omega_0 nT\, e^{-j\omega n}$$

$$= \underbrace{e^{-j(\omega-\Omega_0 T)\frac{N-1}{2}} \frac{\frac{A}{2}\sin\left[(\omega-\Omega_0 T)\frac{N}{2}\right]}{\sin\left[(\omega-\Omega_0 T)\frac{1}{2}\right]}}_{T_1(\omega)} + \underbrace{e^{-j(\omega+\Omega_0 T)\frac{N-1}{2}} \frac{\frac{A}{2}\sin\left[(\omega+\Omega_0 T)\frac{N}{2}\right]}{\sin\left[(\omega+\Omega_0 T)\frac{1}{2}\right]}}_{T_2(\omega)}$$

The second equality follows from a DTFT example on p. 22.6, with $\Omega_0 T$ substituted for ω_0. The two periodic sincs above have peaks at $\omega = \pm\, \Omega_0 T$ and look like



The width of each main lobe is $\frac{4\pi}{N}$. If N is large enough, these pulses don't overlap much and

$$\begin{aligned}
|X_d(\omega)| &= |T_1(\omega) + T_2(\omega)| \\
&\approx |T_1(\omega)| + |T_2(\omega)| \\
&= \frac{A}{2}\left|\frac{\sin\left[(\omega-\Omega_0 T)\frac{N}{2}\right]}{\sin\left[(\omega-\Omega_0 T)\frac{1}{2}\right]}\right| + \frac{A}{2}\left|\frac{\sin\left[(\omega+\Omega_0 T)\frac{N}{2}\right]}{\sin\left[(\omega+\Omega_0 T)\frac{1}{2}\right]}\right|
\end{aligned}$$

Thus $\{|X_m|\}_{m=0}^{N-1}$ will provide approximate samples of the magnitude of the above plot, from which we can estimate Ω_0 and A.

If x_n is a sum of several sinusoids, we will have multiple peaks and we can estimate M, Ω_i, and A_i as:

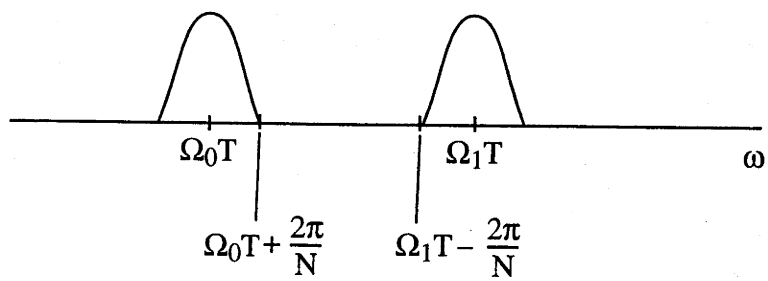
$$\begin{aligned}
M &= \text{number of peaks on } [0, \pi] \\
\Omega_i &= (\text{location of ith peak})/T \\
A_i &= \frac{2}{N}(\text{height of ith peak})
\end{aligned}$$

Actually, sinusoidal spectral analysis, via the DFT, is not quite this straightforward.

Here are some frequent problems:

1) Two Ω_i are so close together that two peaks add together into a single peak. Solution: If possible, use a larger N (i.e., collect data over a longer observation interval) so that peaks will be narrower and won't overlap.

   Suppose you have the frequencies Ω_0 and Ω_1. Then the main lobes of the periodic sincs look like



To clearly distinguish the two peaks, we might require that the main lobes not overlap, i.e.,

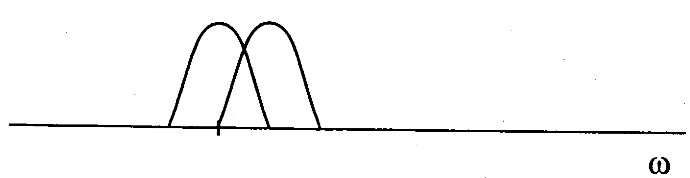
$$\Omega_1 T - \frac{2\pi}{N} > \Omega_0 T + \frac{2\pi}{N}$$

$$\Rightarrow \; (\Omega_1 - \Omega_0)T > \frac{4\pi}{N}$$

$$\Rightarrow NT > \frac{4\pi}{\Omega_1 - \Omega_0}$$

Thus, the necessary observation interval, NT, increases as the frequency separation, $\Omega_1 - \Omega_0$, decreases.

In practice the above condition on NT is too conservative, since we can still discern two peaks even if there is some overlap. If we instead require that there be no more than 50% overlap as shown below



then the required condition is

$$\Omega_1 T - \frac{2\pi}{N} > \Omega_0 T$$

giving

$$NT > \frac{2\pi}{\Omega_1 - \Omega_0}.$$

2) Two Ω_i are fairly close together and one A_i is much smaller than the other, causing the small peak to be buried in the sidelobes of the high peak. Solution: Window $\{x_n\}_{n=0}^{N-1}$ prior to computing the DFT, to reduce sidelobes! Unfortunately, though, this is not a cure-all. Windowing widens the main lobes so that closely spaced sinusoids will be harder to distinguish.

The most important difficulty in spectral analysis is:

3) Noise!! ~ are alternate methods for sinusoidal spectral analysis that would work <u>perfectly</u> if there were no noise.

<u>Fact</u>: Achievable resolution in spectral analysis ultimately depends on noise, and NT ~ length of observation interval of $x_a(t)$.

Zero-Padding in DFT Spectral Analysis

Suppose have $\{x_n\}_{n=0}^{N-1}$ with DTFT $X_d(\omega)$.

Know DFT $\left[\{x_n\}_{n=0}^{N-1}\right] = X_d\left(\frac{2\pi m}{N}\right) \quad 0 \le m \le N-1$

<u>Question</u>: If pad $\{x_n\}_{n=0}^{N-1}$ with M zeros and then compute an N + M point DFT, how is the result related to $X_d(\omega)$?

Let:

$$\tilde{x}_n = \begin{cases} x_n & 0 \le n \le N-1 \\ 0 & N \le n \le N+M-1 \end{cases}$$

Claim:

$$\tilde{X}_m = X_d\left(\frac{2\pi}{N+M} m\right) \quad 0 \le m \le N+M-1$$

i.e., zero padding gives more densely spaced samples of exactly the same DTFT.

Proof:

$$\tilde{X}_m = \sum_{n=0}^{N+M-1} \tilde{x}_n \, e^{-j\frac{2\pi}{N+M}nm} \qquad 0 \le m \le N+M-1$$

$$= \sum_{n=0}^{N-1} x_n \, e^{-j\frac{2\pi}{N+M}nm}$$

$$= X_d\left(\frac{2\pi m}{N+M}\right) \quad ✔$$

Two potential reasons for zero-padding in DFT spectral analysis:

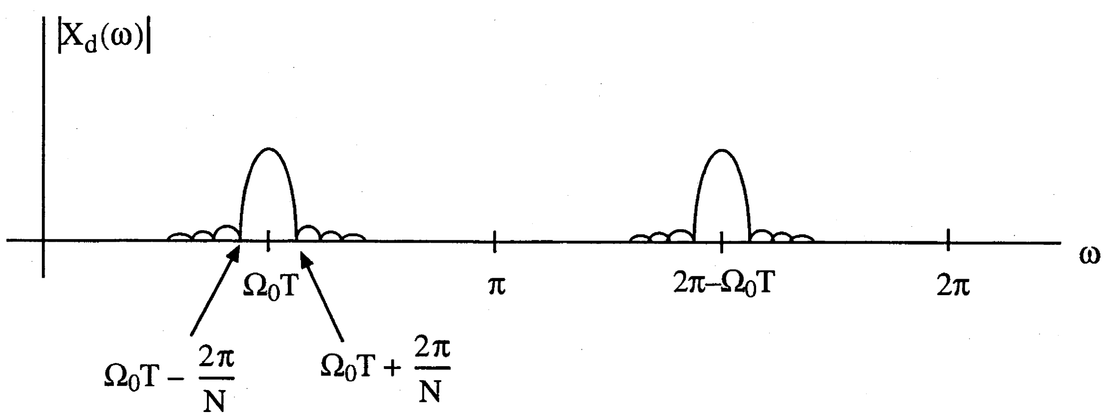
1) Want more densely spaced samples of X_d.

2) Your sequence length, N, is not a length (e.g., power of 2) for which you have a fast DFT available. So, you zero pad out to the nearest length for which you have an FFT.

Item 2) is a consideration if N is large or if you need to compute many length-N DFTs.

Comments on Computer Assignment: DFT Spectral Analysis of Sinusoids

If $x_a(t) = \cos \Omega_0 t$, we have available $\{x_n\}_{n=0}^{N-1}$ where $x_n = \cos \Omega_0 nT$. The magnitude of the DTFT of this sequence looks like



Computing the DFT $\{X_m\}_{m=0}^{N-1}$ gives samples of X_d. Depending on the length of the DFT and the value of $\Omega_0 T$, you may or may not have DFT samples at or near the peaks of the mainlobes and sidelobes. Zero-padding prior to the DFT will give denser samples of X_d and will therefore give a better representation of X_d.

Windowing can help in DFT spectral analysis. Note that given $\{x_n\}_{n=0}^{N-1}$, you have already implicitly used a truncation window.

You could use a different window by multiplying $\{x_n\}_{n=0}^{N-1}$ by a window sequence, such as Hamming, prior to computing the DFT. This will widen main lobes (by a factor of two) but greatly reduce sidelobes.

ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 46

D. Munson

Fast Fourier Transform (FFT)

FFTs comprise a class of algorithms for quickly computing the DFT.

DFT:

$$X_p = \sum_{n=0}^{N-1} x_n \bullet \underset{\uparrow}{W_N^{np}} \qquad 0 \le p \le N-1$$

$$W_N \triangleq e^{-j\frac{2\pi}{N}}$$

A straightforward computation requires:

$N^2 \otimes$, $N(N-1) \oplus$

where these multiplications and additions are generally complex.

There are many different FFTs. We will consider only radix-2 decimation-in-time and decimation-in-frequency algorithms.

Radix-2 FFTs, where the sequence length N is restricted to be a power of two, require only $0(N \log_2 N)$ computations.

Decimation-in-Time Radix-2 FFT

Suppose $N = 2^M$

Idea: Divide input sequence into two groups, those elements of $\{x_n\}$ with n even and those with n odd. Then combine the size N/2 DFTs of these two subsequences to calculate the first half of $\{X_m\}_{m=0}^{N-1}$ and the second half of $\{X_m\}_{m=0}^{N-1}$.

Let
$$\left.\begin{aligned} y_n &= x_{2n} \\ z_n &= x_{2n+1} \end{aligned}\right\} \quad 0 \le n \le \frac{N}{2} - 1$$

Show $\left\{X_p\right\}_{p=0}^{N-1}$ can be obtained from the $\frac{N}{2}$ point DFTs $\left\{Y_p\right\}_{p=0}^{\frac{N}{2}-1}$ and $\left\{Z_p\right\}_{p=0}^{\frac{N}{2}-1}$.

Splitting a size N problem into two size $\frac{N}{2}$ problems will reduce computation because

$$\left(\frac{N}{2}\right)^2 + \left(\frac{N}{2}\right)^2 = \frac{N^2}{2} < N^2$$

Our strategy will then be to divide each size $\frac{N}{2}$ problem into two size $\frac{N}{4}$ problems, etc.

Derivation Relating X_p to Y_p and Z_p:

$$X_p = \sum_{k=0}^{\frac{N}{2}-1} \left(x_{2k}\, W_N^{2kp} + x_{2k+1}\, W_N^{(2k+1)p} \right)$$

$$= \sum_{k=0}^{\frac{N}{2}-1} y_k\, W_{N/2}^{kp} + W_N^p \sum_{k=0}^{\frac{N}{2}-1} z_k\, W_{N/2}^{kp} \qquad (1)$$

since $W_N^{2kp} = e^{-j\frac{2\pi}{N}2kp} = e^{-j\frac{2\pi}{N/2}kp} = W_{N/2}^{kp}$

For $p = 0, 1, \ldots, \frac{N}{2} - 1$, the first sum in (1) is Y_p, and the second sum is $W_N^p\, Z_p$.

$$\Rightarrow \quad \left[X_p = Y_p + W_N^p\, Z_p \qquad 0 \le p \le \frac{N}{2} - 1 \right] \qquad (2)$$

What about X_p for $p > \frac{N}{2} - 1$? We can get these by using (1) to write:

$$X_{p+\frac{N}{2}} = \sum_{k=0}^{\frac{N}{2}-1} y_k\, W_{N/2}^{k\left(p+\frac{N}{2}\right)} + W_N^{p+\frac{N}{2}} \sum_{k=0}^{\frac{N}{2}-1} z_k\, W_{N/2}^{k\left(p+\frac{N}{2}\right)}$$

Note that:

$$W_{N/2}^{k\left(p+\frac{N}{2}\right)} = W_{N/2}^{kp}\, W_{N/2}^{k\frac{N}{2}} = W_{N/2}^{kp} \bullet 1$$

and

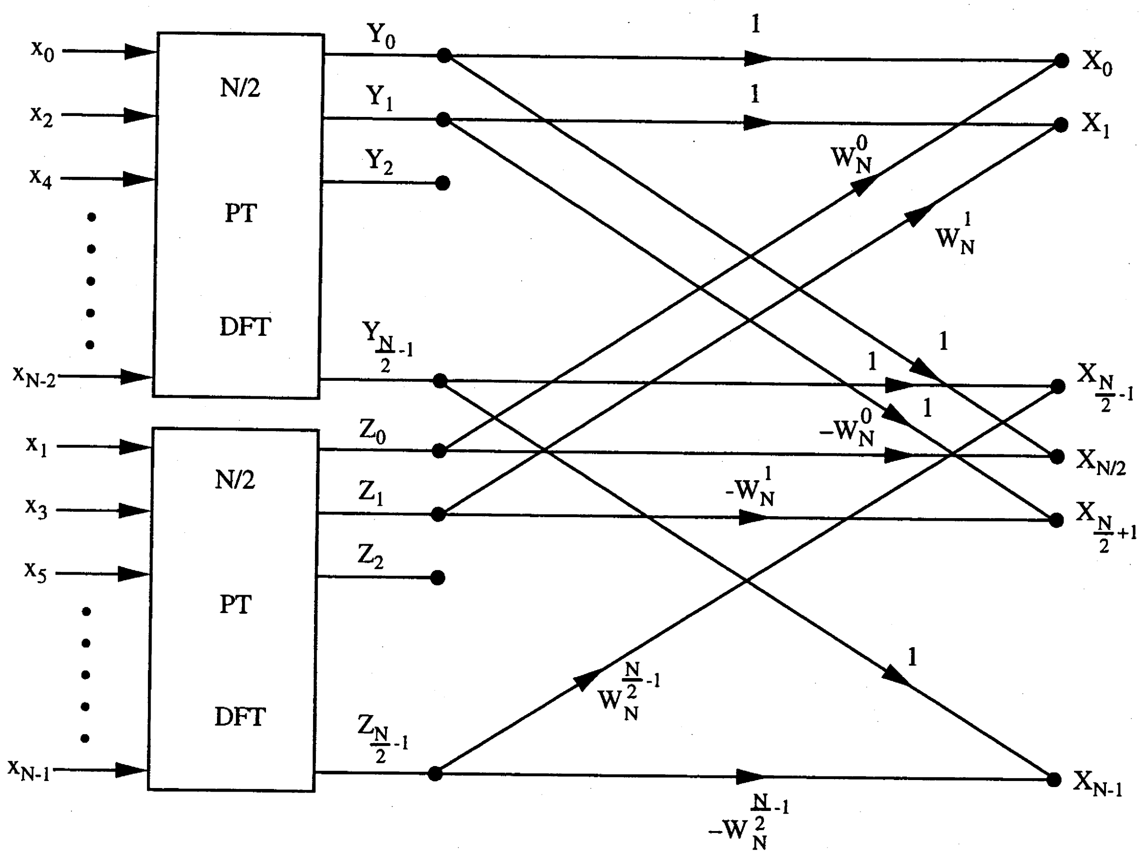
$$W_N^{p+\frac{N}{2}} = W_N^p\, e^{-j\frac{2\pi}{N}\frac{N}{2}} = -W_N^p$$

So:

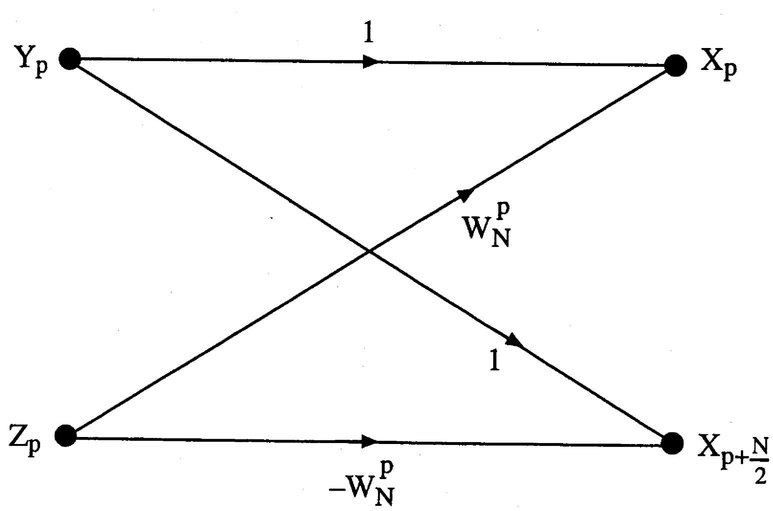
$$W_{p+\frac{N}{2}} = \sum_{k=0}^{\frac{N}{2}-1} y_k\, W_{N/2}^{kp} - W_N^p \sum_{k=0}^{\frac{N}{2}-1} z_k\, W_{N/2}^{kp}$$

$$\Rightarrow \quad \left[X_{p+\frac{N}{2}} = Y_p - W_N^p\, Z_p \qquad 0 \le p \le \frac{N}{2} - 1 \right] \qquad (3)$$

(2) and (3) show how to compute an N point DFT using two $\frac{N}{2}$ point DFTs. These two equations are the essence of the FFT and describe the following flow graph:



The operation to combine the $\frac{N}{2}$ point DFT outputs Y_p and Z_p is called a <u>butterfly</u>:



$0 \le p \le \frac{N}{2} - 1$

This butterfly diagram summarizes (2) and (3).

Our overall strategy will be to:

Replace the N-point DFT by $\frac{N}{2}$ butterflies preceded by the $\frac{N}{2}$-point DFTs.

Replace each $\frac{N}{2}$-point DFT by $\frac{N}{4}$ butterflies preceded by two $\frac{N}{4}$-point DFTs.

⋮

Replace each 4-point DFT by two butterflies preceded by two 2-point DFTs.

Replace each 2-point DFT by a single butterfly preceded by two one-point DFTs. But, a one-point DFT is the identity operation, so a two-point DFT is just a single butterfly.

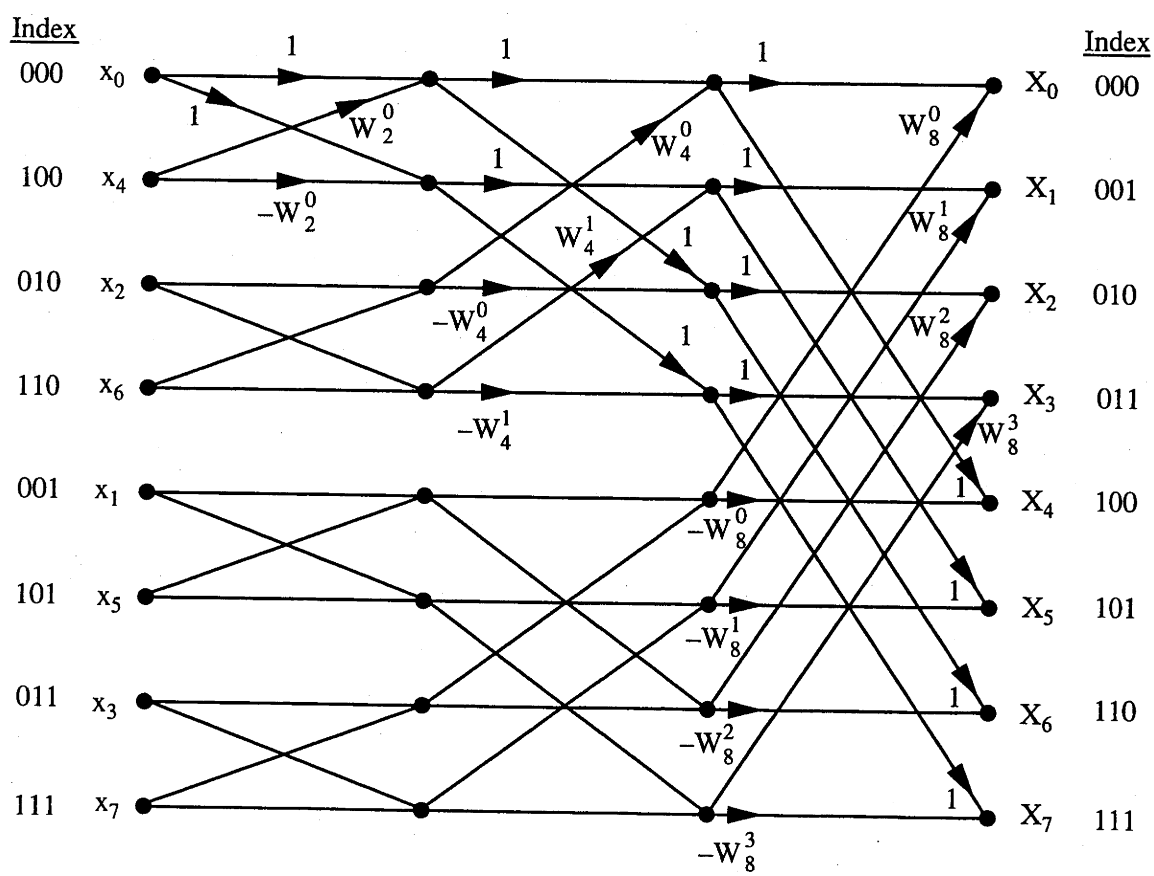
Since $N = 2^M$, this recursion leads to $M = \log_2 N$ stages of $\frac{N}{2}$ butterflies each.

Thus, for a DSP chip that can perform one multiplication and one addition (one multiply-accumulate) in each clock cycle, a radix-2 DIT FFT requires

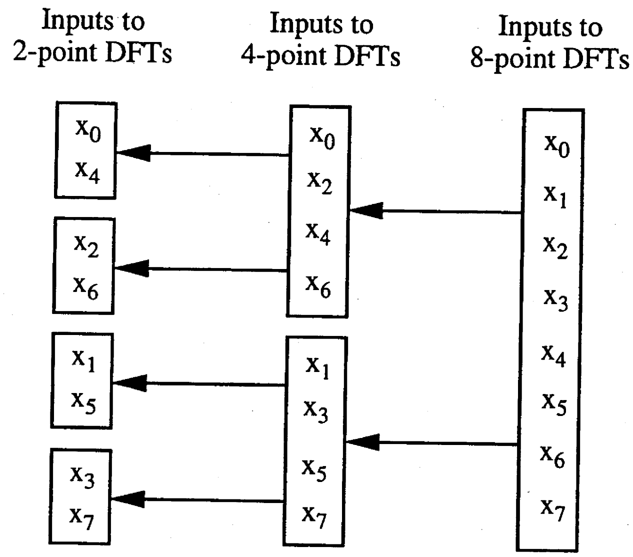
$$\boxed{N \log_2 N \text{ multiply-accumulates}}$$

which can be far less than the N^2 multiply-accumulates required by a straightforward DFT.

Example (N = 8, DIT FFT)



The input x_n is required in "bit-reversed" order. Why? This follows since to compute an N-point DFT using two N/2 point DFTs, we break up the input into even and odd points. We do this successively as we work backward in the flow diagram:



Note: FFT computation can be performed "in place." We need only one length-N array in memory since the output of a butterfly can be written back into the input locations.

Example ~ computational comparison

Suppose $N = 2^{14} = 16{,}384$.

Compare the number of multiply-accumulates in straightforward and DIT FFT implementations of the DFT.

Straightforward: $N^2 = 268{,}435{,}456$ multiply-accumulates

FFT: $N \log_2 N = 2^{14}(14) = 229{,}376$ multiply-accumulates

$$\text{Savings factor} = \frac{268{,}435{,}456}{229{,}376} = 1170!$$

Suppose that in 1964 a state-of-the-art computer required 10 hours to compute a straightforward length 2^{14} DFT. Then, in 1965, after publication of the FFT, this same computation could be performed in about 30 seconds!

Decimation in Frequency Radix – 2 FFT

Idea: Essentially is backwards from DIT. Separate $\{x_n\}_{n=0}^{N-1}$ into first half and second half and then compute even and odd points in $\{X_p\}_{p=0}^{N-1}$ separately, using two $\frac{N}{2}$-point DFTs.

Derivation of algorithm:

$$\begin{aligned}
X_p &= \sum_{n=0}^{N-1} x_n W_N^{np} \\
&= \sum_{m=0}^{\frac{N}{2}-1} x_m W_N^{mp} + \sum_{m=0}^{\frac{N}{2}-1} x_{m+N/2} W_N^{(m+N/2)p} \\
&= \sum_{m=0}^{\frac{N}{2}-1} \left(x_m + x_{m+N/2} W_N^{(N/2)p} \right) W_N^{mp} \qquad (10)
\end{aligned}$$

Look at even and odd points in X_p separately.

Evens:

(10) $\Rightarrow$

$$X_{2q} = \sum_{m=0}^{\frac{N}{2}-1} \left(x_m + x_{m+N/2} \bullet 1\right) W_{N/2}^{mq}$$

$$\Rightarrow \left[\underset{\substack{\uparrow \\ \text{even points in desired} \\ \text{length-N DFT}}}{\left\{X_{2q}\right\}_{q=0}^{N/2-1}} = \underset{\substack{\uparrow \\ \text{N/2 point DFT}}}{\text{DFT}} \left[\left\{ x_m + x_{m+N/2} \right\}_{m=0}^{N/2-1} \right] \right] \tag{11}$$

Odds:

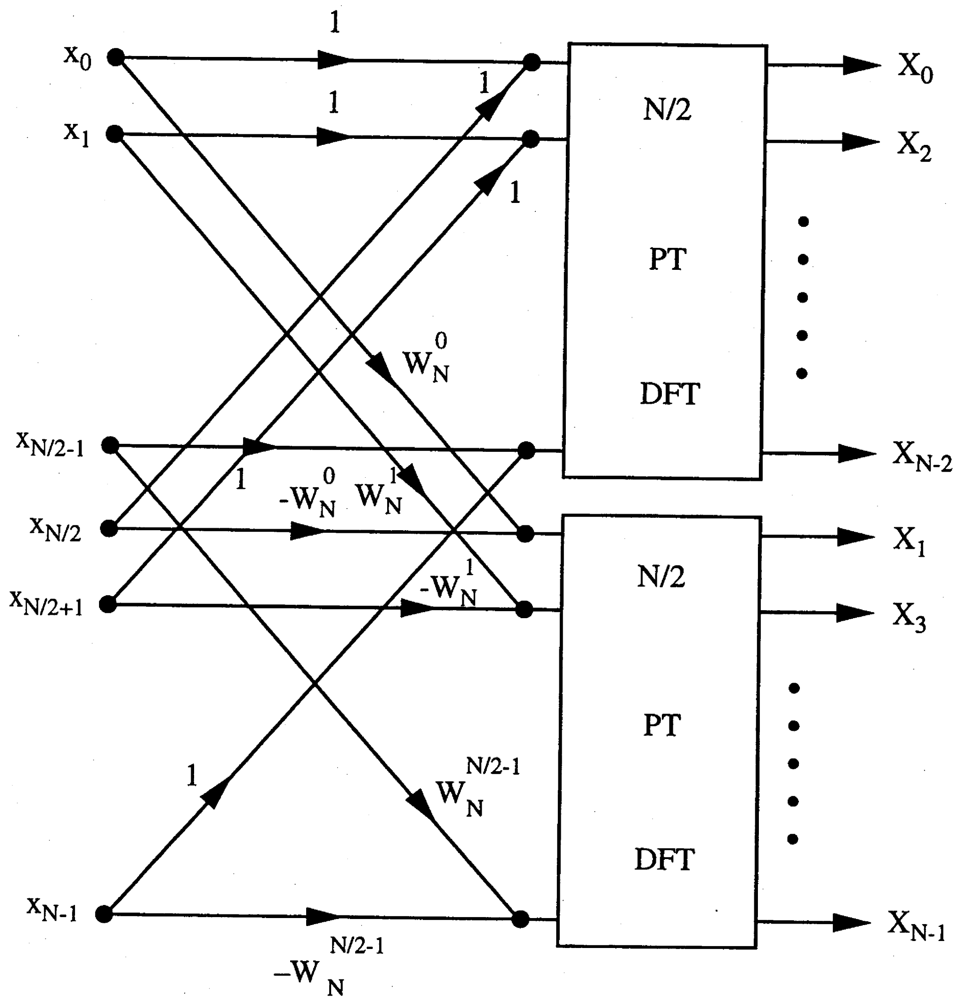
(10) $\Rightarrow$

$$X_{2q+1} = \sum_{m=0}^{\frac{N}{2}-1} \left(x_m + x_{m+N/2} W_N^{(N/2)(2q+1)}\right) W_{N/2}^{mq} W_N^m$$

$$= \sum_{m=0}^{\frac{N}{2}-1} \left[\left(x_m - x_{m+N/2}\right) W_N^m\right] W_{N/2}^{mq}$$

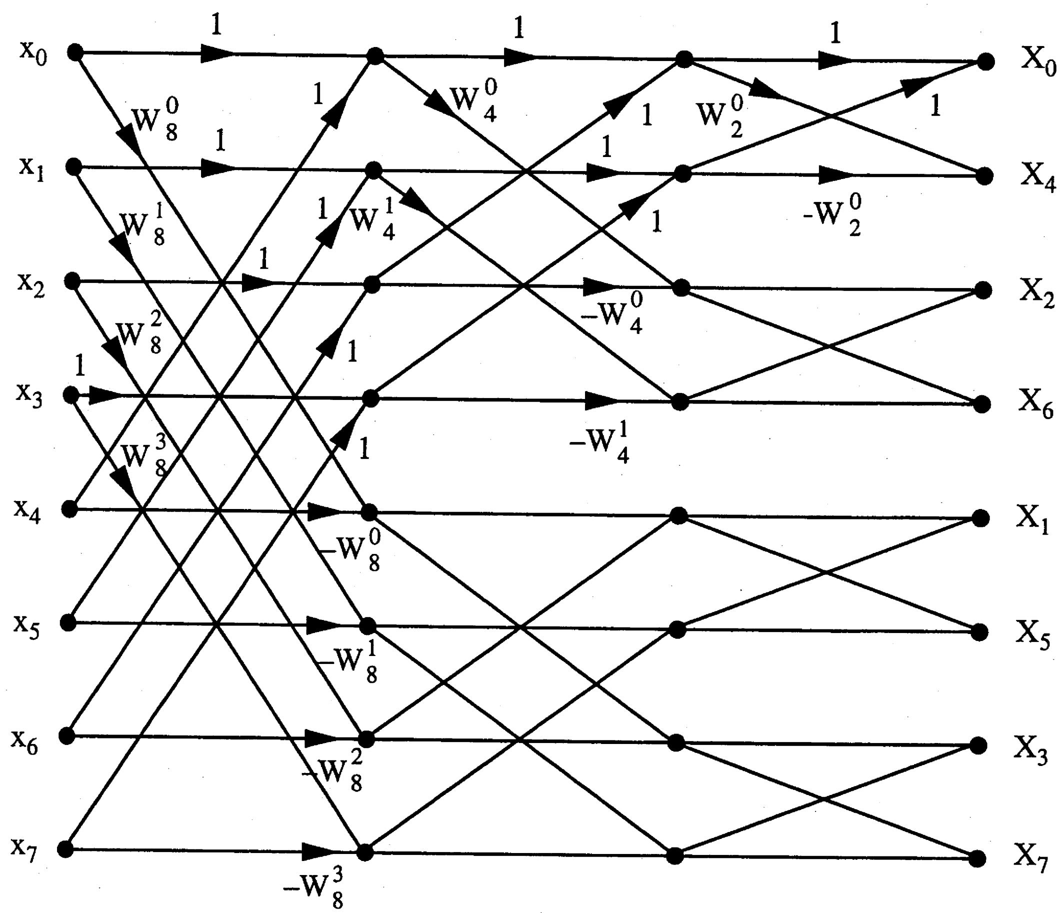
$$\Rightarrow \left[\underset{\substack{\uparrow \\ \text{odd points in desired} \\ \text{length-N DFT}}}{\left\{X_{2q+1}\right\}_{q=0}^{N/2-1}} = \text{DFT} \left[\left\{ (x_m - x_{m+N/2}) W_N^m \right\}_{m=0}^{N/2-1} \right] \right] \tag{12}$$

(11) and (12) give:



The complete DIF algorithm computes each $\frac{N}{2}$-point DFT using two $\frac{N}{4}$-point DFTs, etc. As in the DIT algorithm, we get $\log_2 N$ stages of $\frac{N}{2}$ butterflies each, but now the <u>output</u> appears in bit-reversed order.

Example (N = 8, DIF FFT)



The branch weights are found by using (11) and (12).

Note: As mentioned above, the <u>output</u> appears in bit-reversed order.

Comment: The DIF flow diagram is simply the transpose of the DIT diagram (switch input and output, and reverse all flows).

Other Comments:

1) FFT computer algorithms incorporate the reordering ("bit reversal") of input or output. You don't have to do this yourself.

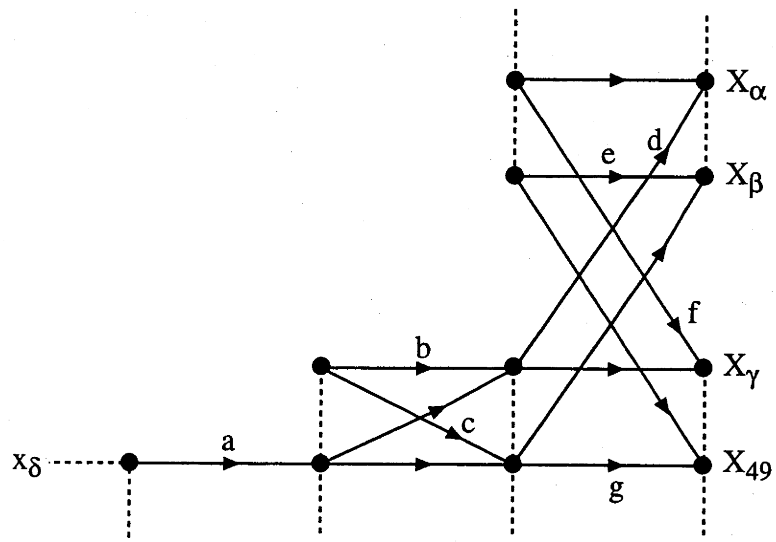
2) Can generalize Radix-2 approach to Radix-3, Radix-4, etc. with $N = 3^M$, $N = 4^M$, etc. For a Radix-4 DIT algorithm, break input up into four groups.

$$\begin{array}{c} x_0\\ x_1\\ x_2\\ x_3\\ x_4\\ x_5\\ x_6\\ x_7\\ x_8\\ \bullet\\ \bullet\\ \bullet\\ \bullet\\ \bullet\\ \bullet\\ \bullet\\ \bullet\\ \bullet\\ \bullet\\ x_{N-3}\\ x_{N-2}\\ x_{N-1} \end{array}
\qquad\qquad
\begin{array}{l}
\left.\begin{array}{c} x_0\\ x_4\\ x_8\\ \bullet\\ \bullet\\ x_{N-4} \end{array}\right\} \rightarrow \boxed{\text{N/4 PT DFT}}\\[2ex]
\left.\begin{array}{c} x_1\\ x_5\\ x_9\\ \bullet\\ \bullet\\ x_{N-3} \end{array}\right\} \rightarrow \boxed{\text{N/4 PT DFT}}\\[2ex]
\left.\begin{array}{c} x_2\\ x_6\\ x_{10}\\ \bullet\\ \bullet\\ x_{N-2} \end{array}\right\} \rightarrow \boxed{\text{N/4 PT DFT}}\\[2ex]
\left.\begin{array}{c} x_3\\ x_7\\ x_{11}\\ \bullet\\ \bullet\\ x_{N-1} \end{array}\right\} \rightarrow \boxed{\text{N/4 PT DFT}}
\end{array}$$

The outputs of the N/4-point DFTs can then be combined, using modified butterflies with 4 inputs and 4 outputs each, to calculate $\{X_m\}_{m=0}^{N-1}$.

Example

Shown below is part of a radix-2, 64-point DIT FFT. Determine the indices α–δ and the coefficients a–g.



Solution: Use Eqs. (2) and (3) from p. 47.2 in course notes:

$$X_p = Y_p + W_N^p Z_p \qquad 0 \le p \le \frac{N}{2} - 1$$

$$X_{p+\frac{N}{2}} = Y_p - W_N^p Z_p \qquad 0 \le p \le \frac{N}{2} - 1$$

$N = 64,\ \beta + \frac{N}{2} = 49 \ \Rightarrow \ \beta = \underline{17}$

$\gamma = 49 - \frac{N}{4} = \underline{33}$

$\alpha = 33 - \frac{N}{2} = \underline{1}$

δ is bit reversal of $49 = (110001)_2 \Rightarrow \delta = (100011)_2 = \underline{35}$

$d = W_{64}^{1} = e^{-j\frac{2\pi}{64}}$ $\qquad$ $g = -\, W_{64}^{17} = -e^{-j\frac{34\pi}{64}}$

$e = 1$ $\qquad$ $b = 1$

$f = 1$ $\qquad$ $c = 1$ $\qquad$ $a = 1$ $\leftarrow$ since this is a top branch in butterfly of 16 pt DFT

ECE 310
University of Illinois

DIGITAL SIGNALS AND SYSTEMS
Lecture 48

D. Munson

Fast Linear Convolution

Recall the cyclic convolution property of the DFT:

$$y_n = \sum_{m=0}^{N-1} h_m x_{<n-m>_N} \quad \text{iff} \quad Y_m = H_m X_m \quad 0 \le m \le N-1$$

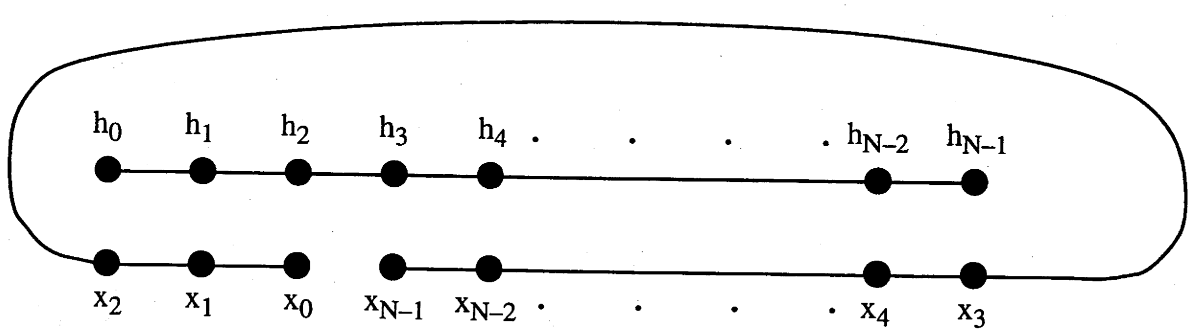
So, we can implement cyclic convolution via

$$\{y_n\} = \text{DFT}^{-1}\left[\text{DFT}\left[\{h_n\}\right] \bullet \text{DFT}\left[\{x_n\}\right]\right] \qquad (\Delta\Delta)$$

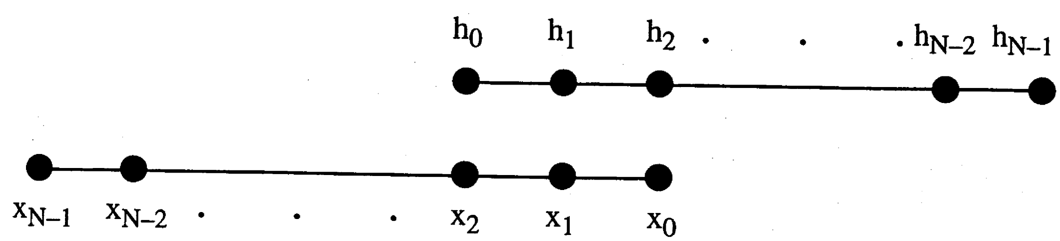
This can be done quickly for long sequence lengths using the FFT.

But, what is cyclic convolution?

To compute y_2:



We would rather implement a linear (regular) convolution:



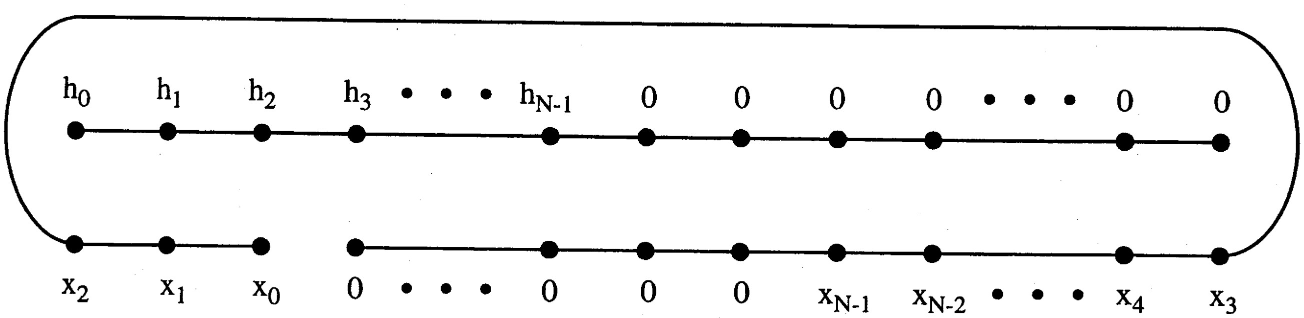
To compute a linear convolution via a cyclic convolution, we must eliminate the wrap-around of nonzero terms in the cyclic convolution. Use zero-padding with N – 1 zeros, i.e., let:

$$\hat{h}_n = \begin{cases} h_n & 0 \le n \le N-1 \\ 0 & N \le n \le 2N-2 \end{cases}$$

$$\hat{x}_n = \begin{cases} x_n & 0 \le n \le N-1 \\ 0 & N \le n \le 2N-2 \end{cases}$$

Now, cyclically convolve the zero-padded sequences.

The result is that $\{\hat{y}_n\}_{n=0}^{2N-2}$ will be a linear convolution of $\{h_n\}_{n=0}^{N-1}$ with $\{x_n\}_{n=0}^{N-1}$. For example, in computing $\hat{y}_2$, we will have:



Obviously, the zero-padding eliminates the wrap-around problem. Using an FFT with (ΔΔ), and zero-padded sequences, provides a fast means of performing linear convolution.

What if $\{h_n\}$ and $\{x_n\}$ are not of the same length?

If $\{h_n\}$ is of length M and $\{x_n\}$ is of length N, then pad each sequence to length N + M – 1 (or nearest larger power of 2 <u>if</u> you are using a radix-2 FFT).

Let's check and see that (ΔΔ), with zero padding, works for a specific example.

Example

$$h_n = \{\underset{\uparrow}{1}, 1, 1\}, \; x_n = \{\underset{\uparrow}{1}, -1, 1\}$$

To produce a linear convolution via (ΔΔ), first pad each sequence with N – 1 = 2 zeros:

$$\hat{h}_n = \{1, 1, 1, 0, 0\}$$

$$\hat{x}_n = \{1, -1, 1, 0, 0\}$$

Now,

$$\hat{H}_m = \sum_{n=0}^{4} \hat{h}_n \, e^{-j\frac{2\pi}{5}nm}$$

$$= 1 + e^{-j\frac{2\pi}{5}m} + e^{-j\frac{4\pi}{5}m}$$

Likewise,

$$\hat{X}_m = 1 - e^{-j\frac{2\pi}{5}m} + e^{-j\frac{4\pi}{5}m}$$

So,

$$\hat{Y}_m = \hat{H}_m \hat{X}_m = 1 \cancel{+ e^{-j\frac{2\pi}{5}m}} \cancel{+ e^{-j\frac{4\pi}{5}m}}$$

$$\cancel{- e^{-j\frac{2\pi}{5}m}} \cancel{- e^{-j\frac{4\pi}{5}m}} \cancel{- e^{-j\frac{6\pi}{5}m}}$$

$$+ e^{-j\frac{4\pi}{5}m} \cancel{+ e^{-j\frac{6\pi}{5}m}} + e^{-j\frac{8\pi}{5}m}$$

$$= 1 + e^{-j\frac{4\pi}{5}m} + e^{-j\frac{8\pi}{5}m}$$

$$= 1 + e^{-j\frac{2\pi}{5}2m} + e^{-j\frac{2\pi}{5}4m}$$

Since

$$\hat{Y}_m = \sum_{n=0}^{4} \hat{y}_n \, e^{-j\frac{2\pi}{5}m}$$

we see that

$$\hat{y}_n = \{1, 0, 1, 0, 1\}$$

It is easy to see that this is the correct <u>linear</u> convolution:

$$\begin{array}{ccccc} & & 1 & 1 & 1 \\ 1 & -1 & 1 & & \end{array}$$

Performing the usual shift and add operations gives the sequence $\{1, 0, 1, 0, 1\}$. ✔

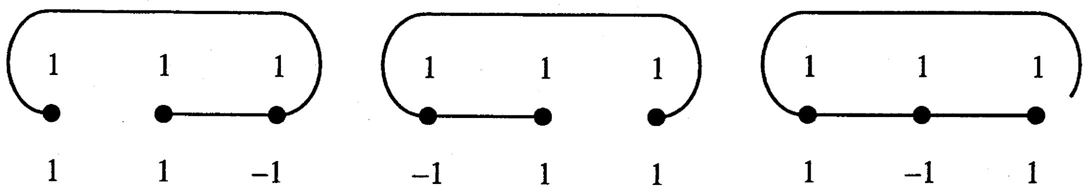
Now, what if we had not zero padded?

Then (ΔΔ) would have produced a <u>cyclic</u> convolution.

The cyclic convolution formula is

$$y_n = \sum_{m=0}^{2} h_m \, x_{<n-m>_3}$$

which is computed pictorially as



$y_0 = 1$ $\qquad$ $y_1 = 1$ $\qquad$ $y_2 = 1$

Let's check that (ΔΔ) without zero-padding gives this same result.

$$H_m = \sum_{n=0}^{2} h_n \, e^{-j\frac{2\pi}{3}nm}$$

$$= 1 + e^{-j\frac{2\pi}{3}m} + e^{-j\frac{4\pi}{3}m}$$

$$X_m = 1 - e^{-j\frac{2\pi}{3}m} + e^{-j\frac{4\pi}{3}m}$$

$$Y_m = H_m X_m$$

$$= 1 + \cancel{e^{-j\frac{2\pi}{3}m}} + \cancel{e^{-j\frac{4\pi}{3}m}}$$

$$- \cancel{e^{-j\frac{2\pi}{3}m}} - \cancel{e^{-j\frac{4\pi}{3}m}} - \cancel{e^{-j\frac{6\pi}{3}m}}$$

$$+ e^{-j\frac{4\pi}{3}m} + \cancel{e^{-j\frac{6\pi}{3}m}} + e^{-j\frac{8\pi}{3}m}$$

Interchanging the latter two terms and using 2π periodicity of the complex exponential gives

$$Y_m = 1 + e^{-j\frac{2\pi}{3}m} + e^{-j\frac{2\pi}{3}2m}$$

Matching up terms with

$$Y_m = \sum_{n=0}^{2} y_n \, e^{-j\frac{2\pi}{3}nm} = y_0 + y_1 \, e^{-j\frac{2\pi}{3}m} + y_2 \, e^{-j\frac{2\pi}{3}2m}$$

gives

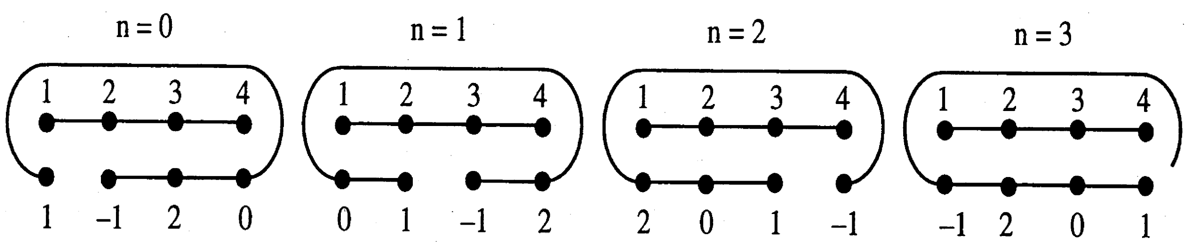
$$\{y_n\} = \underset{\uparrow}{\{1},\, 1,\, 1\} \qquad \checkmark$$

Note: We worked through this example to show that (ΔΔ) can give a linear convolution or a cyclic convolution, depending on whether we first zero pad. In practice, if N is large the DFTs and inverse DFT would be computed using FFTs. If N is small, then it is faster to perform the convolution in the sequence domain.

For practice at computing cyclic convolution in the sequence domain, consider the following example.

Example

Find $y_n = h_n \circledast x_n$ where $\{h_n\}_{n=0}^{3} = \{1, 2, 3, 4\}$ and $\{x_n\}_{n=0}^{3} = \{1, 0, 2, -1\}$.



$y_0 = 5$ $\qquad$ $y_1 = 7$ $\qquad$ $y_2 = 1$ $\qquad$ $y_3 = 7$

Example (Convolution via FFT)

Suppose that a sequence $\{x_n\}_{n=0}^{7000}$ is to be filtered with an FIR filter having coefficients $\{h_n\}_{n=0}^{1100}$.

a) Ignoring possible savings from coefficient symmetry, what is the total number of multiplications required to compute the output $\{y_n\}_{n=0}^{8100}$ by implementing the usual convolution formula with a direct-form filter structure?

b) Using the FFT method (with a radix-2 FFT and zero-padding to length 8192), how many complex multiply-accumulates (MAs) are required to compute $\{y_n\}_{n=0}^{8100}$? How many real MAs are required? (For simplicity, count <u>all</u> "multiplications" in an FFT, even those by ±1, ±j, as complex multiplications.)

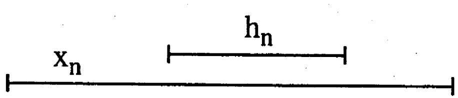
48.6

Solution

a) Output of regular convolution is composed of 3 parts:

\# of MAs is $1 + 2 + 3 + \ldots + 1100$

$$= \frac{(1100)(1101)}{2}$$



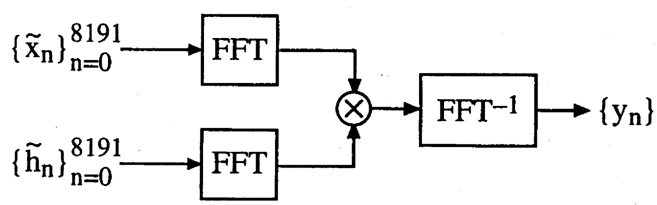
\# of MAs is 1101 (7001 – 1100)

\# of MAs is $1100 + 1099 + \ldots + 1$

$$= \frac{(1100)(1101)}{2}$$

Total # MAs = 1100(1101) + 1101(5901) = $\boxed{7{,}708{,}101}$

b) FFT method is



where $\{\tilde{x}_n\}$ and $\{\tilde{h}_n\}$ are zero-padded versions of $\{x_n\}$ and $\{h_n\}$.

\# complex MAs $= 3\ (N \log_2 N) + N = 3\ (8192 \bullet 13) + 8192 = \boxed{327,680}$

The number of real MAs required to implement a complex MA is generally 4. To see this, write out the detailed calculation (a) (b) + c where a, b, and c are all complex. Assuming this factor of 4 overhead, we have

\# real MAs $= 4\ (327,680) = \boxed{1,310,720}$

Thus, in this example the FFT approach requires fewer than 20% of the MAs required by a straightforward convolution.

ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 49

D. Munson

Block Convolution

Given $\{x_n\}_{n=0}^{N-1}$ and $\{h_n\}_{n=0}^{M-1}$, we have developed an approach for efficiently computing $y_n = h_n * x_n$ using zero-padding and FFTs. But, what if N >> M? If N, the length of the input, is really large, we are faced with two problems:

1) Very long FFTs will be required, which will lead to computational inefficiency.

2) There will be a very long delay in computing $\{y_n\}$ since our scheme requires that all of $\{x_n\}$ be acquired before any element in the output sequence can be computed.

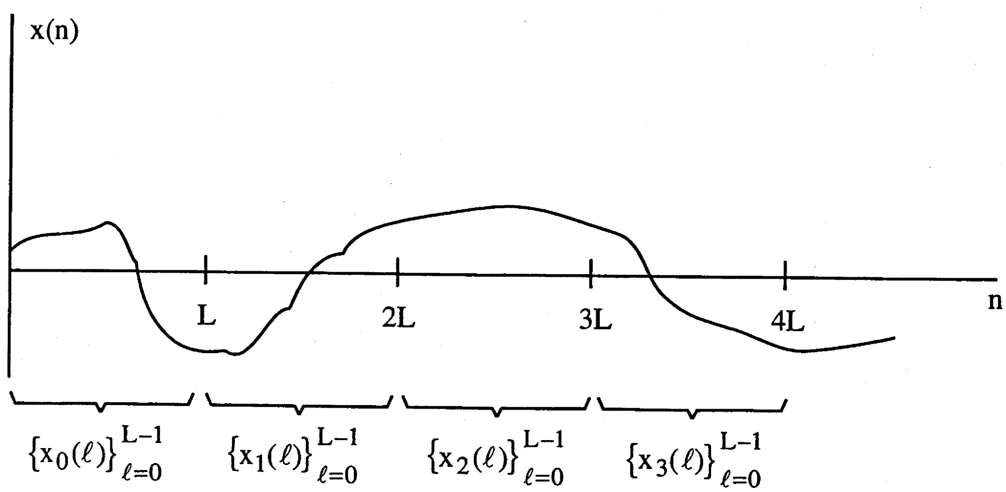
What to do? Answer: Segment the long input $\{x_n\}_{n=0}^{N-1}$ into shorter pieces, convolve the individual pieces with $\{h_n\}_{n=0}^{M-1}$ and then stitch together the results of the shorter convolutions to form $\{y_n\}$. There are two popular ways of doing this.

Method 1: Overlap and Add

Here, we divide up the input into nonoverlapping sections of length L. Let

$$x_k(\ell) = x(kL + \ell) \qquad 0 \le \ell \le L-1, \quad k = 0, 1, 2, \ldots$$

Picture:



We have

$$x(n) = \sum_k x_k(n - kL) \qquad 0 \le n \le N - 1.$$

Now, convolution is a linear operation, so

$$y(n) = h(n) * x(n) = h(n) * \left[\sum_k x_k(n - kL) \right] = \sum_k h(n) * x_k(n - kL)$$

Let $y_k(n) = h(n) * x_k(n)$. Then by shift-invariance,

$$y(n) = \sum_k y_k(n - kL) \; . \qquad (1)$$

We compute each $\{y_k(n)\}$ via the FFT as in the previous lecture. For simplicity, assume $M + L - 1$ is a sequence length for which we have an FFT algorithm. Then

1) Pad $\{x_k(n)\}_{n=0}^{L-1}$ with M–1 zeros to give $\{\tilde{x}_k(n)\}_{n=0}^{L+M-2}$.

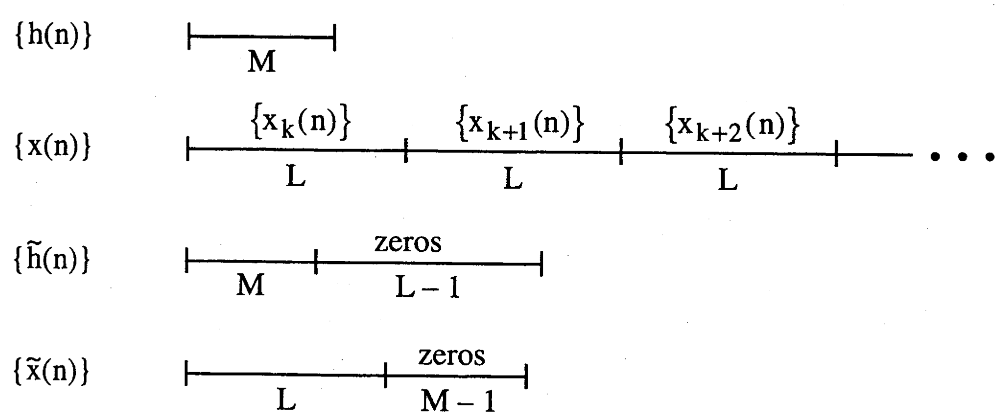
   Pad $\{h(n)\}_{n=0}^{M-1}$ with L–1 zeros to give $\{\tilde{h}(n)\}_{n=0}^{L+M-2}$.

2) Calculate the FFTs of $\{\tilde{x}_k(n)\}_{n=0}^{L+M-2}$ and $\{\tilde{h}(n)\}_{n=0}^{L+M-2}$.

3) Multiply FFTs together and take FFT^{-1} to give $\{y_k(n)\}_{n=0}^{L+M-2}$.

Finally, calculate $\{y(n)\}$ via (1) by adding together the appropriately shifted $\{y_k(n)\}$.

Pictorially:

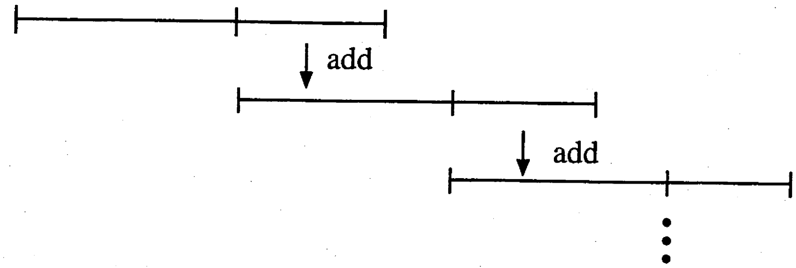


$y_0(n) = h(n) * x_0(n)$

$y_1(n) = h(n) * x_1(n)$

$y_2(n) = h(n) * x_2(n)$

$\vdots$



Sum of the shifted (by kL) $y_k(n)$ gives $\{y(n)\}$.

Example

Given $\{h(n)\}_{n=0}^{249}$ and $\{x(n)\}_{n=0}^{\infty}$ we wish to compute $\{y(n)\} = \{h(n)\} * \{x(n)\}$ using the FFT method. What is the best block length L, using the Overlap and Save method with radix-2 FFTs?

We have M = 250. Let K = FFT length. Then since K = L + M – 1, the block length will be L = K – 249. Each length-K FFT and inverse FFT requires $K \log_2 K$ MAs. Multiplication of FFTs requires K MAs. We shall assume that the FFT of $\{\tilde{h}(n)\}$ is precomputed once and stored. Thus, the amount of computation for each input block will be

$$2\,K \log_2 K + K = K\,(2\log_2 (K) + 1) \qquad \text{MAs.}$$

This amount of computation is needed to compute each $\{y_k(n)\}$ k = 0, 1, 2, ... from each input block $\{x_k(n)\}$ of length L = K – 249. Thus, the computation per input sample (or per output sample), ignoring the few additions needed to sum the overlapping $\{y_k(n)\}$ blocks, is

$$\frac{K[2\log_2 K + 1]}{K - 249} \qquad (2)$$

Trying some different values for the FFT length K, we find:

K	L	Complex MAs Per output
256	7	621.7
512	263	37.0
1024	775	27.7
2048	1799	26.2
4096	3847	26.6

K = FFT length

L = input block length

\# MAs given by (2)

For larger K, (2) approaches $(2 \log_2 K) + 1$, which grows with K.

Even allowing for the required complex arithmetic (4 real MAs per complex MA), the FFT approach offers considerable savings over a direct filter implementation, which would require 250 MAs per output.

Notes:

1) Based on the above table, and if we are at all concerned about delay, we would select an FFT block length of either 512 or 1024.

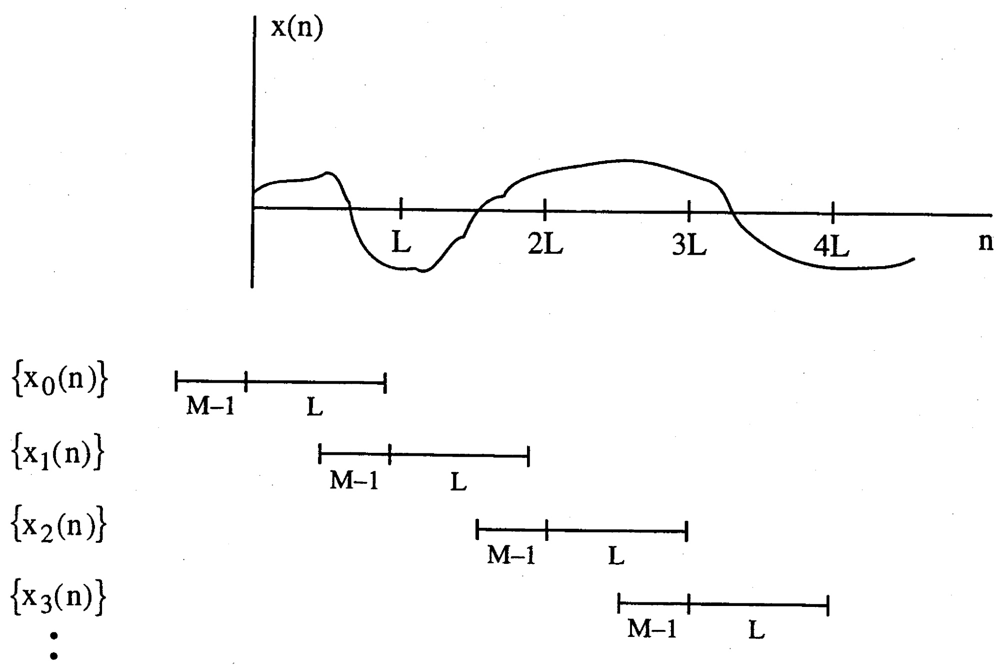
2) If $\{x_n\}$ and $\{h_n\}$ were complex-valued, then the direct filter implementation would require roughly 1000 MAs per filter output.

3) If a sequence is real, there are tricks that can be used to speed up computation (by a factor of approximately two) of its DFT. If both $\{x(n)\}$ and $\{h(n)\}$ are real, in which case $\{y(n)\}$ is real, these tricks can be used to reduce the number of MAs in the FFT approach by nearly a factor of two over the entries shown in the above table.

Method 2: Overlap and Save

Could just as easily be called Overlap and Discard.

Here, we define the $\{x_k(n)\}$ to be overlapping as shown below.



The first M–1 entries of $\{x_0(n)\}$ are filled with zeros. All other entries of $\{x_0(n)\}$ and all entries of all other subsequences $\{x_k(n)\}$ are filled with the values of $\{x(n)\}$ directly above. In general, each subsequence overlaps with its two neighboring subsequences. The algorithm to calculate $\{y(n)\}$ is then:

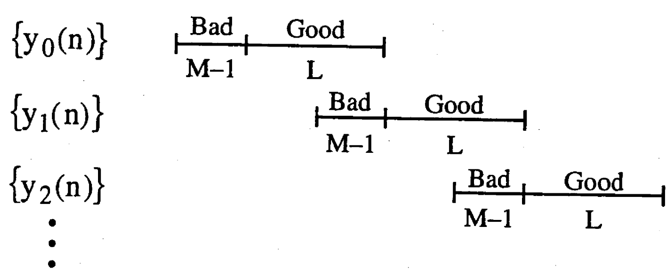
1) Zero-pad $\{h(n)\}_{n=0}^{M-1}$ with L–1 zeros to produce $\left\{\tilde{h}(n)\right\}_{n=0}^{M+L-2}$.

2) Cyclically convolve (via FFT) $\left\{\tilde{h}(n)\right\}_{n=0}^{M+L-2}$ with each $\{x_k(n)\}_{n=0}^{M+L-2}$ to give

$$y_k(n) = \tilde{h}(n) \circledast x_k(n). \qquad 0 \le n \le M + L - 2$$

The result is that the first M–1 samples of each $\{y_k(n)\}$ will be useless, but the last L samples will be samples of $\{y(n)\}$.

3) Assemble $\{y(n)\}$ as shown:



The "bad" samples are discarded and the "good" samples are concatenated to form $\{y(n)\}$.

ECE 310
University of Illinois

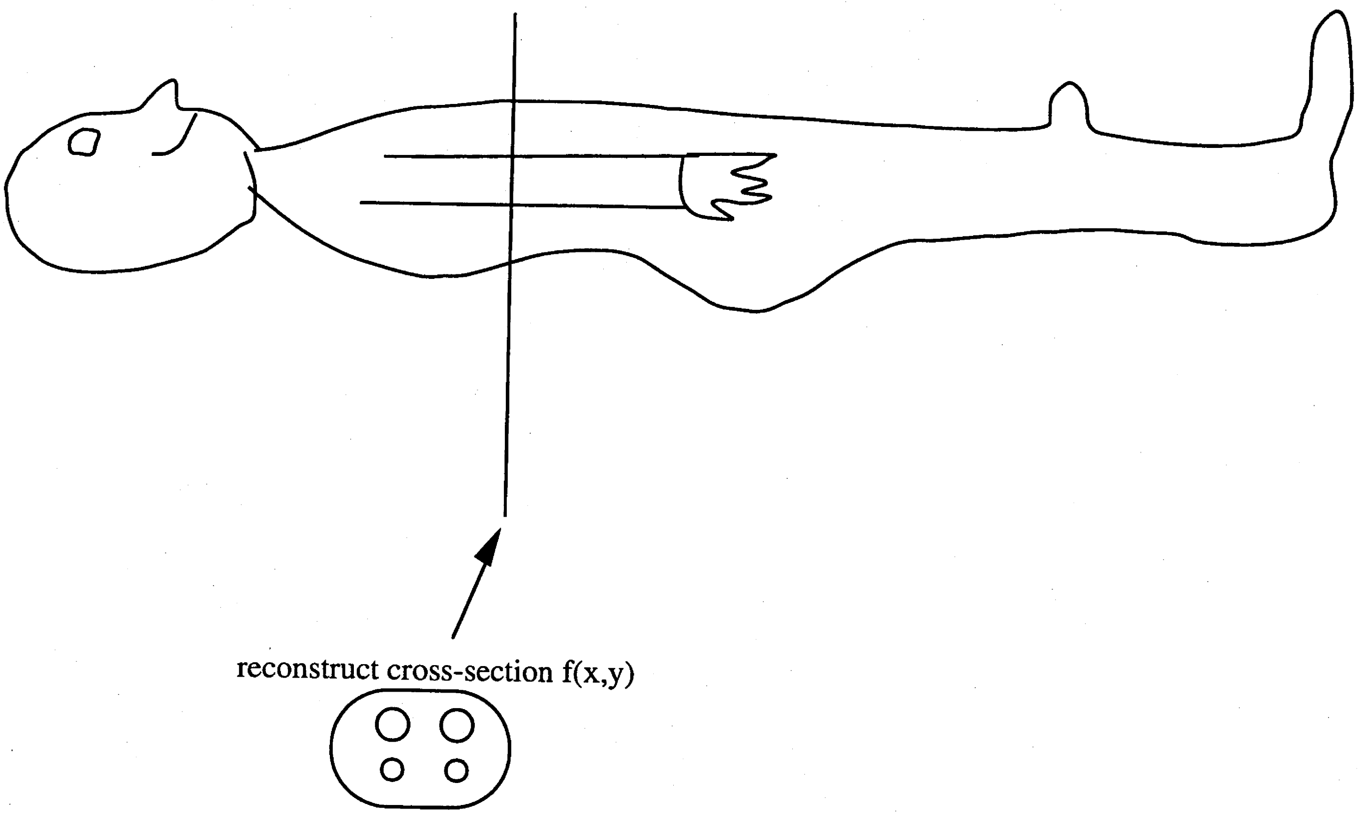
DIGITAL SIGNAL PROCESSING
Lecture 50

D. Munson

Application 1: Computer Tomography (CT)

Used extensively for medical imaging, nondestructive testing.

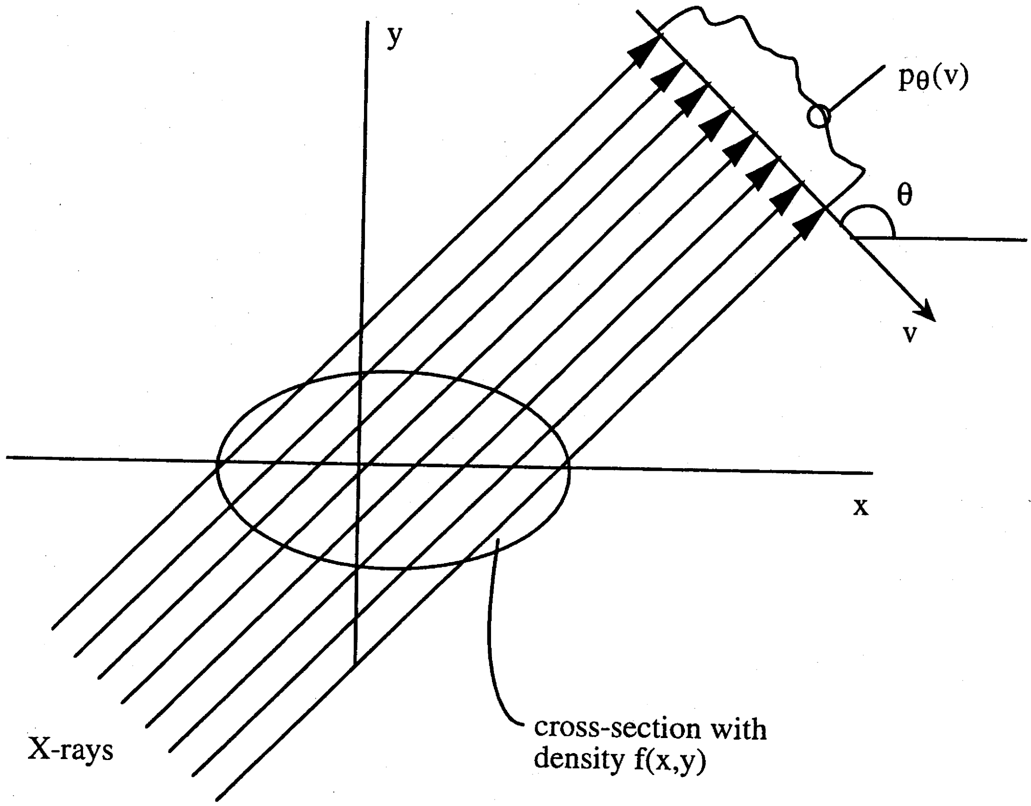
Objective is to reconstruct a cross-sectional view of a 3-D object:



Accomplish this by shining x-rays sideways through the object and collecting "projections" at various angles. The projection data is then processed digitally to produce the image of $f(x,y)$.

The oldest CT machines used narrow, parallel x-ray beams. Modern-day machines use a fan-beam geometry. Since the digital processing in both systems is similar, we will consider the parallel-beam case, which is a bit simpler mathematically.

Parallel beam geometry



$p_\theta(v)$ is a set of line integrals called a **projection.**

$p_\theta(v_o)$ is the integral of f(x,y) along the path of the x-ray at angle θ impinging at $v = v_o$.

Typically, projections $p_\theta(v)$ are collected through a full 360° by rotating the x-ray source(s) and detectors around the object being imaged.

How do we recover f(x,y) from the projections $p_\theta(v)$?

Define the 2-D Fourier transform of f(x,y) as

$$F(\Omega_1, \Omega_2) = \int_{-\infty}^{\infty}\int f(x,y)e^{-j(\Omega_1 x+\Omega_2 y)}\, dxdy$$

The inverse 2-D Fourier transform is

$$f(x,y) = \frac{1}{(2\pi)^2}\int_{-\infty}^{\infty}\int F(\Omega_1,\Omega_2)\, e^{j(\Omega_1 x+\Omega_2 y)}\, d\Omega_1\, d\Omega_2$$

Notation:

Let $F(\Omega_1,\Omega_2)$ in <u>polar</u> coordinates be written as

$$F_{pol}(r, \phi) = F(r\cos\phi, r\sin\phi)$$

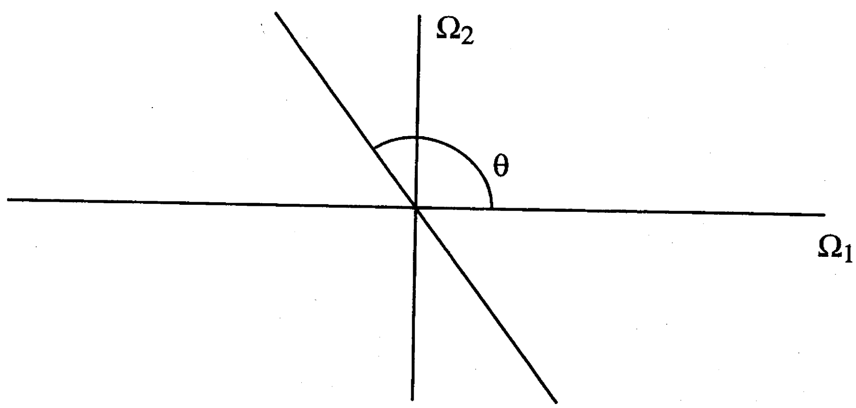
Then the following famous theorem forms the basis for reconstructing f(x,y) from its projections.

Projection-Slice Theorem

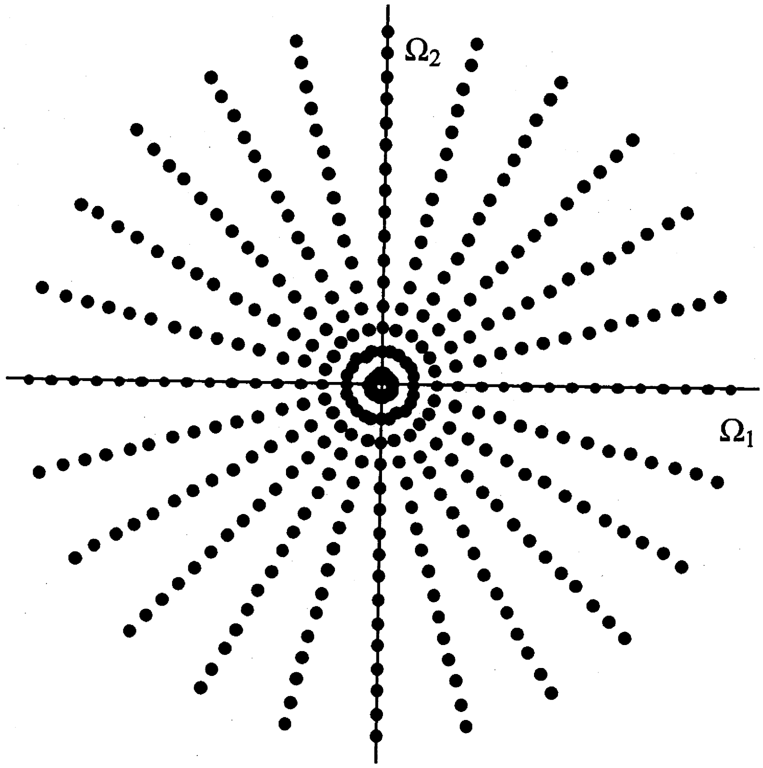
Let $P_\theta(\Omega)$ be the 1-D Fourier transform of $p_\theta(v)$. Then

$$\left[P_\theta(\Omega) = F_{pol}(\Omega,\theta)\right]$$

So, the Fourier transform of a projection is a radial slice of the 2-D Fourier transform of f(x,y) at angle θ:



Collecting sampled projections at many (usually hundreds) of angles and taking the DFT (via FFT) of each projection gives samples of $F(\Omega_1, \Omega_2)$ on a polar grid:



To reconstruct samples of f(x,y) we might try discretization of the inverse 2-D Fourier transform.

We had:

$$f(x,y) = \frac{1}{(2\pi)^2} \int_{-\infty}^{\infty}\int F(\Omega_1,\Omega_2)\, e^{j(\Omega_1 x+\Omega_2 y)}\, d\Omega_1\, d\Omega_2$$

Writing this integral in polar coordinates, and then discretizing, would give an approximate formula for f(nT,mT) in terms of the available polar samples of $F(\Omega_1, \Omega_2)$. Computing N^2 samples of f(x,y) from N^2 samples of F would require

$$N^2 \times N^2 = N^4 \otimes$$

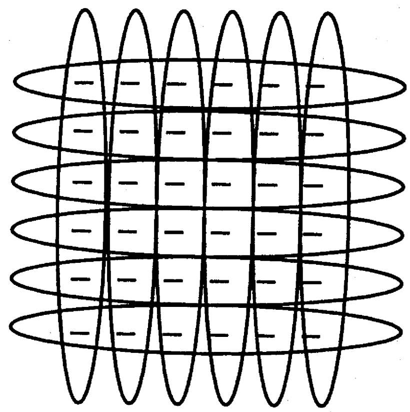
which is excessive. For example, if N = 512, this approach would require about 64 x 10^9 MAs.

A faster alternative would be to

1) Interpolate the polar Fourier data to a Cartesian grid.

2) Compute a 2-D FFT^{-1} (requires ~ $2N^2 \log_2 N$ MAs)

A 2-D DFT is implemented by a series of row FFTs, followed by a series of column FFTs:



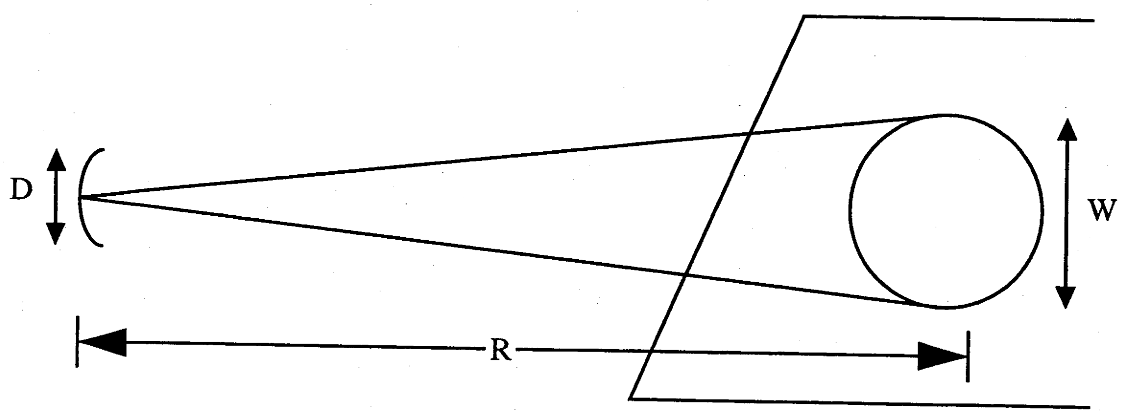
In practice, an accurate implementation of Step 1) requires more computation than the 2-D FFT^{-1}. The most popular image reconstruction algorithm for computer tomography is "convolution-back-projection" (also called filtered back-projection) which is essentially an accurate and efficient way to accomplish 1) and 2) in $0(N^3)$ MAs. Researchers are now working on convolution-back-projection algorithms that require only $0(N^2 \log_2 N)$ MAs.

Application 2: Synthetic Aperture Radar (SAR)

SAR is a high-resolution microwave imaging system. It is used widely in applications such as earth resources monitoring, military reconnaissance, planetary imaging, etc.

Same advantages of microwave imaging over optical: can penetrate fog, cloud cover, atmosphere of Venus, etc., and does not rely on illumination by the sun.

Disadvantage: It is hard to achieve optical resolution. Why? Because, although we can get high resolution in range via delay measurements, the cross-range resolution is seemingly limited by the antenna beamwidth, which can be very wide at microwave frequencies:



If D is the antenna diameter, R is the range to the scene, and λ is the wavelength of radiation, then the width of the antenna-footprint is

$$W = \frac{R\lambda}{D}$$

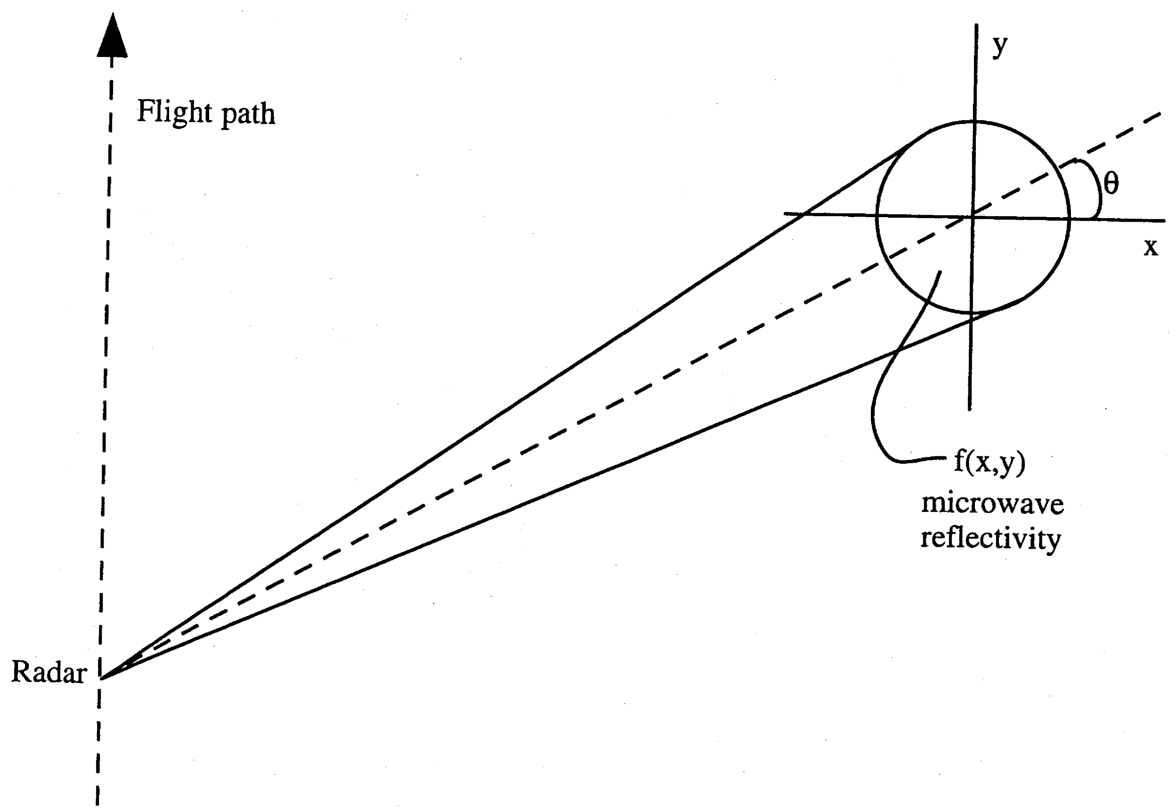
For high cross-range resolution, we want W to be small, but this is hard to get. R may be large and dictated by the imaging scenario, and you hope to use an antenna of practical size (D small).

For microwaves, λ is <u>much</u> larger than for visible light. So, with microwaves, D may need to be impractically-large to achieve a desired cross-range resolution.

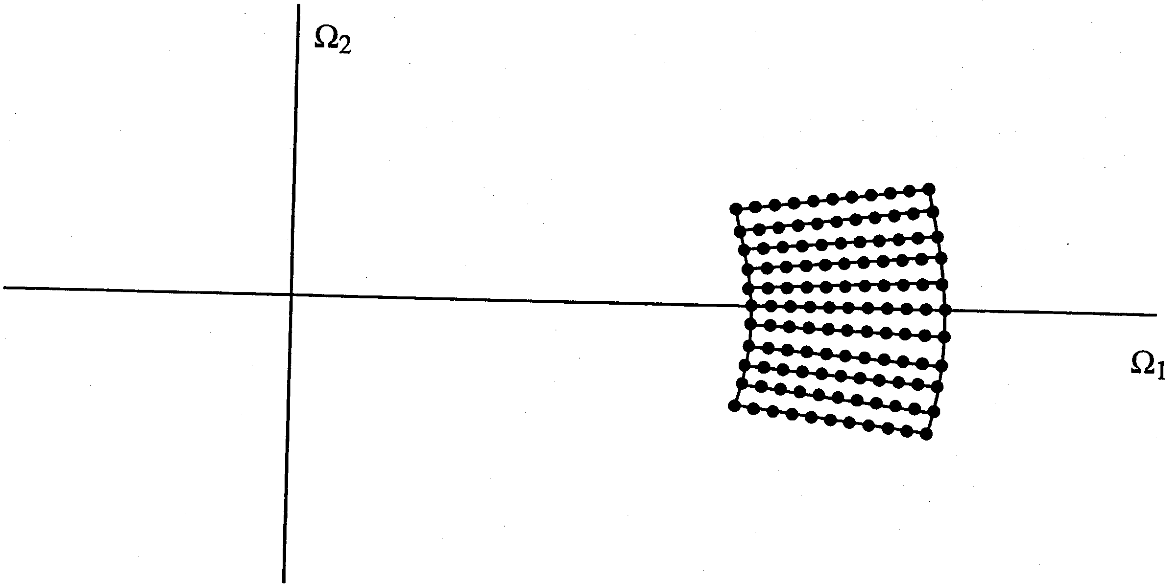
Solution:

Use small D with large W, but collect and process data from <u>many angles</u>. This is called <u>spotlight-mode SAR</u>.

Imaging geometry



If a linear FM waveform $\cos(\Omega_0 t + \alpha t^2)$ is transmitted, it can be shown that demodulated, sampled returns provide Fourier data on a polar grid:



The return collected at angle θ in the spatial domain gives Fourier data on the radial trace at the same angle θ in the Fourier domain. (Proof of this fact uses the projection-slice theorem.) The inner and outer radii of the Fourier data region are proportional to the lowest and highest frequencies, respectively, of the transmitted linear FM signal.

Reconstruction algorithm:

1) Polar-to-Cartesian interpolation

2) 2-D FFT^{-1}

3) Display magnitude of the result.

Typical resolution:

1 ft., or less, to 20 m, depending on the application

These resolutions are achievable at exceedingly long ranges (e.g., space-based monitoring of the earth).

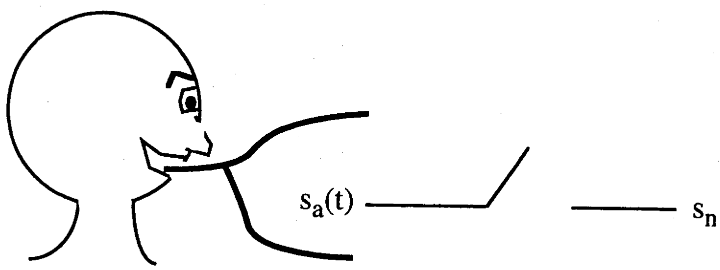
ECE 310
University of Illinois

DIGITAL SIGNAL PROCESSING
Lecture 51

D. Munson

Application 3: Speech Analysis/Synthesis

Consider an approach to speech coding called LPC – Linear Predictive Coding.
This scheme is used in many speech communication systems, automated answering systems and electronic games.



Speech samples are highly correlated, so that s_n can often be fairly-well predicted from its past values.

Suppose we wish to predict s_N from $s_{N-1}, s_{N-2}, \ldots, s_{N-K}$. Try linear prediction.

Estimate s_N by:

$$\hat{s}_N = \sum_{k=1}^{K} a_k s_{N-k}$$

The $\{a_k\}$ are called LPC coefficients.

Choose the $\{a_k\}$ to minimize $E\left\{\left(s_N - \hat{s}_N\right)^2\right\}$.

So, do this

$$\min_{\{a_i\}_{i=1}^{K}} E\left\{\left(s_N - \sum_{k=1}^{K} a_k s_{N-k}\right)^2\right\}$$

$$\Rightarrow \frac{\partial}{\partial a_i} E\left\{\left(s_N - \sum_{k=1}^{K} a_k s_{N-k}\right)^2\right\} = 0 \qquad i = 1, \ldots, K$$

$$\Rightarrow E\left\{2\left(s_N - \sum_{k=1}^{K} a_k s_{N-k}\right)\left(-s_{N-i}\right)\right\} = 0 \quad i = 1, \ldots, K$$

$$\Rightarrow \sum_{k=1}^{K} a_k \, E\left\{s_{N-k}\, s_{N-i}\right\} = E\left\{s_N\, s_{N-i}\right\} \qquad i = 1, 2, \ldots K \tag{1}$$

Suppose $\{s_n\}$ is short-term "wide-sense stationary." Then $E\{s_m\, s_n\}$ depends only on the separation between m and n, i.e., on $|m-n|$, not on m and n individually.

In this case we can write $E\ \{s_n\, s_m\}$ as some function $R_s(n-m) = R_s(m-n)$ where R_s is called the autocorrelation.

Substituting R_s into (1) gives

$$\sum_{k=1}^{K} a_k\, R_s(i-k) = R_s(i) \qquad i = 1, 2, \ldots, K$$

This set of K equations can be expressed in matrix form as

$$\begin{bmatrix} R_s(0) & R_s(1) & \cdots & R_s(K-1) \\ R_s(1) & R_s(0) & \ddots & \vdots \\ \vdots & \ddots & \ddots & R_s(1) \\ R_s(K-1) & \cdots & R_s(1) & R_s(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_K \end{bmatrix} = \begin{bmatrix} R_s(1) \\ R_s(2) \\ \vdots \\ R_s(K) \end{bmatrix} \tag{2}$$

Given the $R_s(i)$, this set of equations can be solved for the optimal $\{a_k\}_{k=1}^{K}$

We might approximate $R_s(i)$ as:

$$R_s(i) = \underset{\substack{\uparrow \\ \#\text{ terms in sum} = L_i}}{\frac{1}{L_i}} \sum_n s_n\, s_{n+i}$$

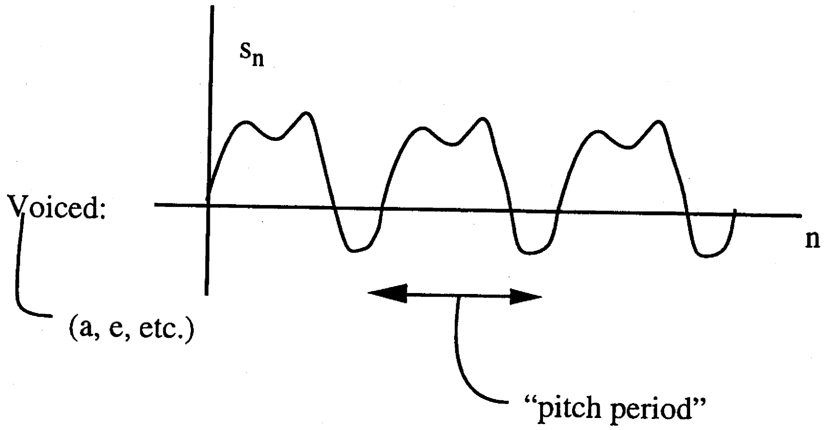
The solution of (2) would ordinarily require $0(K^3)$ operations.

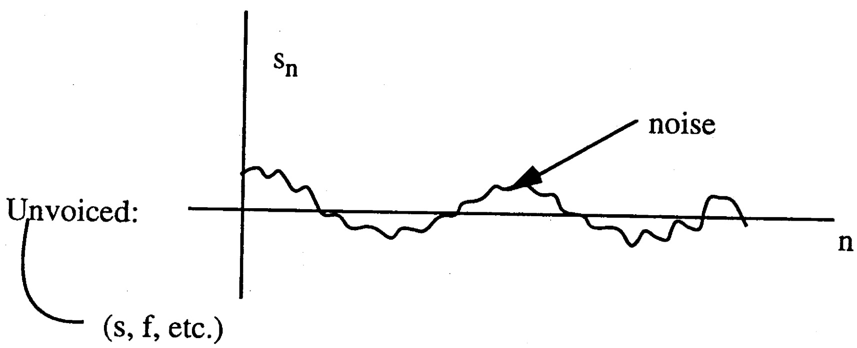
But, the matrix has a special Toeplitz structure $\Rightarrow$ faster algorithms exist.

The Levinson - Durbin algorithms require $0(K^2)$ operations.

Now, look at speech!

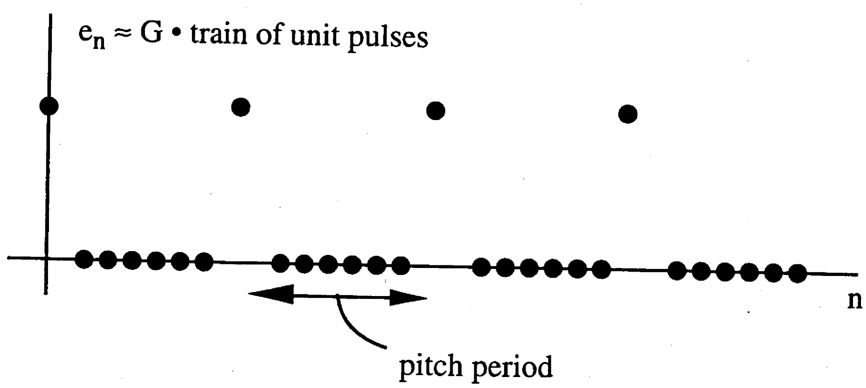
Classification of speech segments:





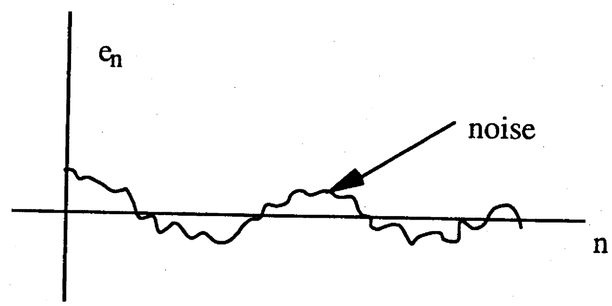
Now, suppose we have the optimal $\{a_k\}$and we look at the prediction error $e_n = s_n - \hat{s}_n$.

It turns out that for voiced sounds e_n is well approximated by a pulse train:



where G is a slowly varying gain.

For unvoiced sounds, e_n looks like noise:



Using our definition of e_n and the LPC model, we have

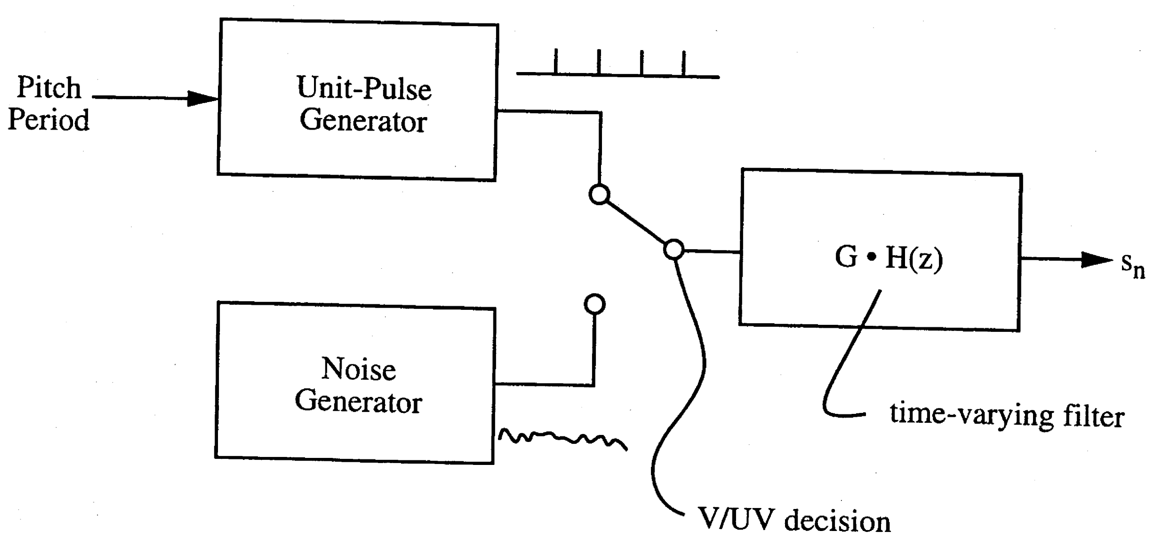
$$s_n = \hat{s}_n + e_n = \sum_{k=1}^{K} a_k s_{n-k} + e_n$$

Thus, we can obtain s_n from e_n via

$$e_n \longrightarrow \boxed{H(z)} \longrightarrow s_n$$

where $H(z) = \dfrac{1}{1 - \sum_{k=1}^{K} a_k z^{-k}}$

Standard Speech Model for Analysis/Synthesis:



This model provides the basis for a speech analysis/synthesis scheme:

1) Analyze each 20 msec segment of the speech waveform to get:

   a) V/UV decision

   b) Pitch period (if voiced)

   c) $\{a_k\}_{k=1}^{K}$ ~ by solving equations in (2)

   d) Gain ~ G

Transmit a) – d) every 20 msec. At the receiver, reconstruct an approximation to the original speech waveform by using the above model.

<u>Comparison with PCM</u>

PCM: Sample speech at ~ 8 kHz; use 7 bits/sample

$\Rightarrow$ 56 K bits/sec.

Analysis/Synthesis:

(assuming a fancier version of LPC than we just covered)

8 K bits/sec: very close to regular telephone quality

2 K bits/sec: very understandable, somewhat machine-like

600 bits/sec: understandable, quite machine-like

Thus, we see that the LPC scheme can greatly reduce the bit rate for both transmission and storage of speech.