BINGHAMTON
UNIVERSITY
State University of New York

EEO 401
Digital Signal Processing

Prof. Mark Fowler

Note Set #4

- Difference Equations – Review
- Reading Assignment: Sect. 2.4 of Proakis & Manolakis

Recall: Convolution is a Method to Compute Output

Could we use this to actually implement (i.e., build) a DT system???

Consider practical scenario of causal system with input that "starts" @ $n = 0$

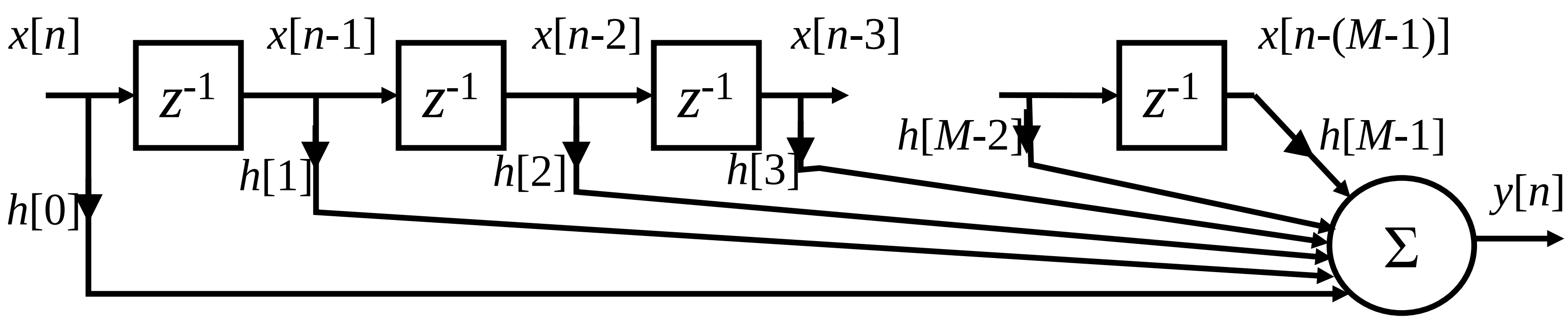
$$y[n] = \sum_{i=0}^{n} h[i]x[n-i]$$

If this is an IIR system then we run into trouble as n grows large… The amount of computation needed to compute output samples grows larger and larger!!!

Now… if instead we had an order M FIR system then we could use this:

$$y[n] = \sum_{i=0}^{M-1} h[i]x[n-i]$$

Now… we do not have any trouble with computational complexity growing without bound! Each output sample needs only M multiplications and $M - 1$ additons.



IIR Systems, Recursion and Difference Equations

So…. We can use convolution to implement an FIR system but not an IIR system!

For some IIR systems we can overcome this limitation. Some IIR systems can be implemented using a "recursive" form.

As a motivating example recall the accumulator system:

$$y[n] = \sum_{i=0}^{n} x[i]$$

Can verify that this is an IIR system with impulse response $h[n] = u[n]$

We can rewrite this as $$y[n] = \underbrace{\left[\sum_{i=0}^{n-1} x[i]\right]}_{=y[n-1]} + x[n]$$

$$\boxed{y[n] = y[n-1] + x[n]}$$

Recursive!

So… to compute the current output sample $y[n]$ we need the most recent past output sample $y[n$-1] and the current input sample $x[n]$

Generalizing this idea leads to the concept of **Difference Equations**.

LTI System

A general N^{th}-order Linear, Constant-Coefficient Difference Equations looks like this:

$$\boxed{y[n]+a_1 y[n-1]+\ldots+a_N y[n-N]=b_0 x[n]+b_1 x[n-1]+\ldots+b_M x[n-M]}$$

$$\Rightarrow \boxed{y[n]+\sum_{k=1}^{N} a_k y[n-k]=\sum_{k=0}^{M} b_k x[n-k]}$$

Rewriting in recursive form gives:

$$\boxed{y[n]=-\sum_{k=1}^{N} a_k y[n-k]+\sum_{k=0}^{M} b_k x[n-k]}$$

Clearly, such a system can be computed with a fixed and finite amount of operations needed for each output sample. Later we will dig deeper into the implementation/computational issues of such systems.

Exploring Simple First-Order Difference Equation

$$y[n] = ay[n-1] + x[n] \qquad \text{with input } x[n] \text{ for } n \geq 0$$

Clearly, to compute $y[0]$ we need to know $y[-1]$ so we assume we know that "initial condition" (IC).

For this simple system we explore the recursive solution and deduce the general form for the output:

$$y[0] = ay[-1] + x[0]$$

$$y[1] = ay[0] + x[1] = a^2 y[-1] + ax[0] + x[1]$$

$$y[2] = ay[1] + x[2] = a^3 y[-1] + a^2 x[0] + ax[1] + x[2]$$

$$\vdots$$

$$y[n] = \underbrace{a^{n+1} y[-1]}_{\text{Depends only on IC}} + \underbrace{\sum_{k=0}^{n} a^k x[n-k]}_{\text{Depends only on Input}}, \quad n \geq 0$$

Zero-Input Response

Zero-State Response

$$y[n] = \underbrace{a^{n+1}y[-1]}_{\text{Depends only on IC}} + \underbrace{\sum_{k=0}^{n} a^k x[n-k]}_{\text{Depends only on Input}}, \quad n \geq 0$$

Zero-Input Response

Zero-State Response

- Zero-Input Response is the response to the ICs when the input is zero
  - Also called "Natural Response"
  - Note here it is exponential…. And in general will involve exponentials
    - Similar to why exponentials show up when solving differ<u>ential</u> Eqs
- Zero-State Response is the response to the input when the ICs are zero
  - Also called "Forced Response"
  - From its form here we see that it has the form of a convolution
  - That is true in general for Linear, Constant-Coefficient Difference Eqs

$$\sum_{k=0}^{n} a^k x[n-k] = \sum_{k=0}^{n} h[k]x[n-k] \qquad\qquad h[n] = a^n u[n]$$

We can verify that this Difference Equation has this as its impulse response.

By definition, the impulse response is the <u>zero-state</u> response to an impulse at the input.

$$h[0] = ah[-1] + x[0] = a \times 0 + 1 = 1$$

$$h[1] = ah[0] + x[1] = a \times 1 + 0 = a$$

$$h[2] = ah[1] + x[2] = a \times a + 0 = a^2$$

$$h[3] = ah[2] + x[3] = a \times a^2 + 0 = a^3$$

$$\vdots$$

$$h[n] = a^n u[n]$$

Big Picture

For the Linear, Constant-Coefficient Difference Eqs of interest the solution will always end up like this

$$y[n] = y_{zi}[n] + y_{zs}[n], \quad n \geq 0$$

Will involve exponentials involving roots of "Characteristic Polynomial"

Will be convolution between impulse response and input

The methods discussed in this chapter for solution are valid but very limited in their capabilities to (i) handle more complex systems and (ii) provide much insight into understanding system behavior.

The z-transform methods we will see in Ch. 3 are much more powerful!

Note… the book also discusses a related approach of breaking the solution into homogeneous and particular solutions… we'll skip that!