

## Lecture 3

*Lecturer: Asst. Prof. M. Mert Ankarali*

## Difference Equations

In discrete-time domain, we have difference equations that replaces differential equations. We are mainly interested in LTI systems, that are represented by linear constant coefficient difference equations. Let  $x[k]$  and  $y[k]$  be the input and output respectively, then an LTI difference equation can be expressed as

$$a_0y[k] + a_1y[k-1] + \dots + a_Ny[k-N] = b_0x[k] + \dots + b_Mx[k-M]$$

$$\sum_{n=1}^N a_ny[k-n] = \sum_{n=1}^M b_nx[k-n]$$

Unlike ODEs difference equations are very easy to solve computationally or simulate in computer environment. Let's consider the following first-order difference equation

$$y[k] = \frac{1}{2}y[k-1] + x[k] \quad , x[k] = 0 \text{ \& } y[k] = 0, \text{ for } k < 0$$

Let's "simulate" the difference equation for  $x[k] = \delta[k]$ .

$$\begin{aligned} y[0] &= \frac{1}{2}y[-1] + x[0] = 0 + 1 = 1 \\ y[1] &= \frac{1}{2}y[0] + x[1] = \frac{1}{2} + 0 = \frac{1}{2} \\ y[2] &= \frac{1}{2}\frac{1}{2} = \frac{1}{4} \\ y[3] &= \frac{1}{2}\frac{1}{4} = \frac{1}{8} \\ &\vdots \\ y[k] &= \left(\frac{1}{2}\right)^k \end{aligned}$$

Now let's simulate for  $x[k] = u[k]$

$$\begin{aligned} y[0] &= 0 + 1 = 1 \\ y[1] &= \frac{1}{2} + 1 \\ y[2] &= \frac{1}{4} + \frac{1}{2} + 1 \\ y[3] &= \frac{1}{8} + \frac{1}{4} + \frac{1}{2} + 1 \\ &\vdots \\ y[k] &= \frac{1}{2^k} + \dots + \frac{1}{2} + 1 = 2 - \left(\frac{1}{2}\right)^k \end{aligned}$$

This is a great method for “simulating” using a computational approach, but in general it may be very hard to get a closed form expression. The most basic solution method is solving the difference equation directly in time domain by trying to find a “basis” for the solution space similar to the operation in ODEs. We try sequences/signals of the form  $\lambda^k$ ,  $k > 0$  to find a solution form for the homogeneous equation. Let’s apply this method for the first-order difference equation above

$$\begin{aligned} y[k] = \lambda^k \rightarrow y[k] - \frac{1}{2}y[k-1] &= 0 \\ \lambda^k - \frac{\lambda^{k-1}}{2} &= 0 \\ \lambda^{k-1} \left( \lambda - \frac{1}{2} \right) &= 0 \\ \lambda - \frac{1}{2} &= 0 \end{aligned}$$

Where the last equation is the characteristic equation of the difference equation. Since the characteristic equation has one root only, we obtain a solution of the form

$$y[k] = y_h[k] + y_p[k] = C \left( \frac{1}{2} \right)^k + y_p[k]$$

Let’s assume that for  $x[k] = u[k]$  particular solution has the form  $y_p[k] = A$  for  $k > 0$  then

$$A = \frac{1}{2}A + 1 \rightarrow A = 2$$

Now let’s find  $C$  using the fact that  $y[k] = 0$  for  $k < 0$

$$\begin{aligned} y[0] &= \frac{1}{2}y[-1] + x[0] \rightarrow y[0] = 1 \\ 1 &= C \left( \frac{1}{2} \right)^0 + 2 \rightarrow C = -1 \end{aligned}$$

Then the solution can be written as

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

**Example 1.1** Find the general form of the homogeneous solution for the following difference equation

$$y[k] - 3y[k-1] + 2y[k-2] = x[k]$$

**Solution:**

$$\begin{aligned} \lambda^2 - 3\lambda + 2 &= 0 \\ \lambda_1 = 1 \ \&\ \lambda_2 = 2 \\ y[k] &= C_1 + C_2 2^k, k > 0 \end{aligned}$$

**Example 1.2** Now let’s assume that  $y[k] = 0$  for  $k < 0$  and  $x[k] = 3^k$ , then find  $y[k]$  for  $k \geq 0$ .

**Solution:** First let’s find a particular solution. Let’s assume that  $y_p[k] = A3^k$ , then

$$\begin{aligned} A3^k - 3A3^{k-1} + 2A3^{k-2} &= 3^k \rightarrow A = 9/2 \\ y_p[k] &= 4.5 \ 3^k \end{aligned}$$

Now let's try to find  $C_1$  and  $C_2$

$$\begin{aligned}y[k] - 3y[k-1] + 2y[k-2] &= x[k] \\y[0] = x[0] &\rightarrow C_1 + C_2 = -3.5 \\y[1] - 3y[0] &= x[1] \rightarrow C_1 + C_2 2 = -7.5 \\C_1 &= 0.5 \text{ \& } C_2 = -4 \\y[k] &= 0.5 - 4 \cdot 2^k + 4.5 \cdot 3^k, k > 0\end{aligned}$$

**What about repeated roots?** Possible mini project question

**Example 2** Find the general form of the homogeneous solution for the following difference equation

$$y[k] + 4y[k-2] = x[k]$$

**Solution:**

$$\begin{aligned}\lambda^2 + 4 &= 0 \rightarrow \lambda_{1,2} = \pm 2j \\y[k] &= C_1(2j)^k + C_2(-2j)^k = C_1 2^k e^{jk\pi/2} + C_2 2^k e^{-jk\pi/2} \\y[k] &= \bar{C}_1 2^k \frac{e^{jk\pi/2} + e^{-jk\pi/2}}{2} + \bar{C}_2 2^k \frac{e^{jk\pi/2} - e^{-jk\pi/2}}{2j} \\y[k] &= \bar{C}_1 2^k \cos(k\pi/2) + \bar{C}_2 2^k \sin(k\pi/2)\end{aligned}$$

**How we can generalize this to arbitrary complex conjugate roots?** Possible mini project question

**What is the home message?** Similar to ODEs time domain solution of difference equations is generally “messy”.

## Z-transform & Difference Equations

### Difference Equations to Z-transform

Let's consider the following difference equation with  $y[n]$  and  $x[n]$  be the strictly causal input-output pair.

$$a_0 y[k] + a_1 y[k-1] + \dots + a_N y[k-N] = b_0 x[k] + \dots + b_M x[k-M]$$

Now let's assume that  $\mathcal{Z}\{x[k]\} = X(z)$  and  $\mathcal{Z}\{y[k]\} = Y(z)$ . If we take the Z-transform for the both sides of the equation by applying the shifting theorem we obtain

$$\begin{aligned}a_0 Y(z) + a_1 z^{-1} Y(z) + \dots + a_N z^{-N} Y(z) &= b_0 X(z) + \dots + b_M z^{-M} X(z) \\(a_0 + a_1 z^{-1} + \dots + a_N z^{-N}) Y(z) &= (b_0 + b_1 z^{-1} + \dots + b_M z^{-M}) X(z) \\\frac{Y(z)}{X(z)} = G(z) &= \frac{b_0 + b_1 z^{-1} + \dots + b_M z^{-M}}{a_0 + a_1 z^{-1} + \dots + a_N z^{-N}} \\&= z^{N-M} \frac{b_0 z^M + b_1 z^{M-1} + \dots + b_M}{a_0 z^N + a_1 z^{N-1} + \dots + a_N}\end{aligned}$$

Under “zero initial conditions” if we can find  $X(z)$ , then we can compute  $Y(z)$  using  $Y(z) = G(z)X(z)$ . After that we can take the inverse z-transform and compute  $y[k]$ .

**Example 3.1** Compute  $y[k]$  using the Z-transform method

$$y[k] = \frac{1}{2}y[k-1] + x[k]$$

$$y[k] = 0, \text{ for } k < 0 \text{ \& } x[k] = \delta[k]$$

**Solution:**

$$Y(z) = \frac{1}{2}Y(z)z^{-1} + X(z) \rightarrow \frac{Y(z)}{X(z)} = G(z) = \frac{z}{z-1/2}$$

$$Y(z) = \frac{z}{z-1/2} \rightarrow y[k] = \left(\frac{1}{2}\right)^k$$

**Example 3.2** Now let's compute  $y[k]$  for  $x[k] = u[k]$

$$Y(z) = G(z)X(z) \rightarrow Y(z) = \frac{z^2}{(z-1/2)(z-1)}$$

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

$$y[k] = 2 - \left(\frac{1}{2}\right)^k$$

**Example 4** For the following difference equation, compute  $y[k]$  for  $x[k] = x[k] = 3^k u[k]$

$$y[k] - 3y[k-1] + 2y[k-2] = x[k]$$

**Solution:**

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

$$Y(z) = \frac{z^3}{(z-1)(z-2)(z-3)} = 0.5\frac{z}{z-1} - 4\frac{z}{z-2} + 4.5\frac{z}{z-3}$$

$$y[k] = (0.5 - 4 \cdot 2^k + 4.5 \cdot 3^k) u[k]$$

## Z-transform to Difference Equations

Sometimes the Z-domain transfer function of a system is given, and we may be supposed to find the difference equation representation. Let's assume that we have a general transfer function that can be represented in terms of ratio of two polynomials in  $z$  or  $z^{-1}$  as given below

$$\frac{Y(z)}{X(z)} = G(z) = \frac{b_0 + b_1z^{-1} + \dots + b_Mz^{-M}}{a_0 + a_1z^{-1} + \dots + a_Nz^{-N}} = z^{N-M} \frac{b_0z^M + b_1z^{M-1} + \dots + b_M}{a_0z^N + a_1z^{N-1} + \dots + a_N}$$

In this case, I prefer to work with the polynomials that are written in terms of  $z^{-1}$ . Let's manipulate the Z-domain equation to obtain

$$Y(z)(a_0 + a_1z^{-1} + \dots + a_Nz^{-N}) = X(z)(b_0 + b_1z^{-1} + \dots + b_Mz^{-M})$$

$$a_0Y(z) + a_1z^{-1}Y(z) + \dots + a_Nz^{-N}Y(z) = b_0X(z) + b_1z^{-1}X(z) + \dots + b_Mz^{-M}X(z)$$

Let's assume that  $\mathcal{Z}^{-1}\{Y(z)\} = y[k]$  and  $\mathcal{Z}^{-1}\{X(z)\} = x[k]$ . If we take the inverse Z-transform of both sides by applying the shifting theorem we obtain

$$a_0y[k] + a_1y[k-1] + \dots + a_Nz^{-N}y[k-N] = b_0x[k] + b_1x[k-1] + \dots + b_Mx[k-M]$$

We can use this conversion to “simulate” a given discrete time transfer function or realizing the given system (it may be a filter or controller) to implement on an embedded platform.

It can also be used for computationally finding the inverse Z-transform of a given z-domain rational function. The next example will illustrate this feature.

**Example 5** Find a computational solution for the inverse Z-transform of  $H(z) = \frac{z^{-1}}{1-2z^{-1}+z^{-2}}$  by using the conversion from Z-domain transfer function to difference equation concept.

**Solution:** Let's assume that  $H(z)$  is a “transfer function” not an arbitrary z-domain function. Then  $\mathcal{Z}^{-1}\{H(z)\} = h[k]$  becomes the impulse response of the “system”. Thus we can assume some imaginary input-output pair  $y[n]$  and  $x[n]$  where

$$\frac{Y(z)}{X(z)} = H(z)$$

If we can find a difference equation realization for  $H(z)$  then we can simulate the difference equation by assuming  $x[k] = \delta[k]$  (i.e. unit impulse input). So let's find a realization for the given  $H(z)$  as

$$\begin{aligned}\frac{Y(z)}{X(z)} &= \frac{z^{-1}}{1-2z^{-1}+z^{-2}} \\ Y(z) - 2z^{-1}Y(z) + z^{-2}Y(z) &= z^{-1}X(z) \\ y[k] - 2y[k-1] + y[k-2] &= x[k-1]\end{aligned}$$

Now let's simulate the above equation for  $x[k] = \delta[k]$

$$\begin{aligned}y[k] &= 2y[k-1] - y[k-2] + x[k-1] \\ y[0] &= 2y[-1] - y[-2] + x[-1] = 0 \\ y[1] &= 2y[0] - y[-1] + x[0] = 1 \\ y[2] &= 2y[1] - y[0] + x[1] = 2 \\ y[3] &= 2y[2] - y[1] + x[2] = 3 \\ y[4] &= 4 \\ &\dots \\ y[k] &= k\end{aligned}$$