



A photograph of a modern, multi-story building with a glass facade, illuminated from within, set against a sunset sky. The building is reflected in a body of water in the foreground. The sky is a mix of orange, pink, and blue. The building has several towers and a central section with a large glass facade.

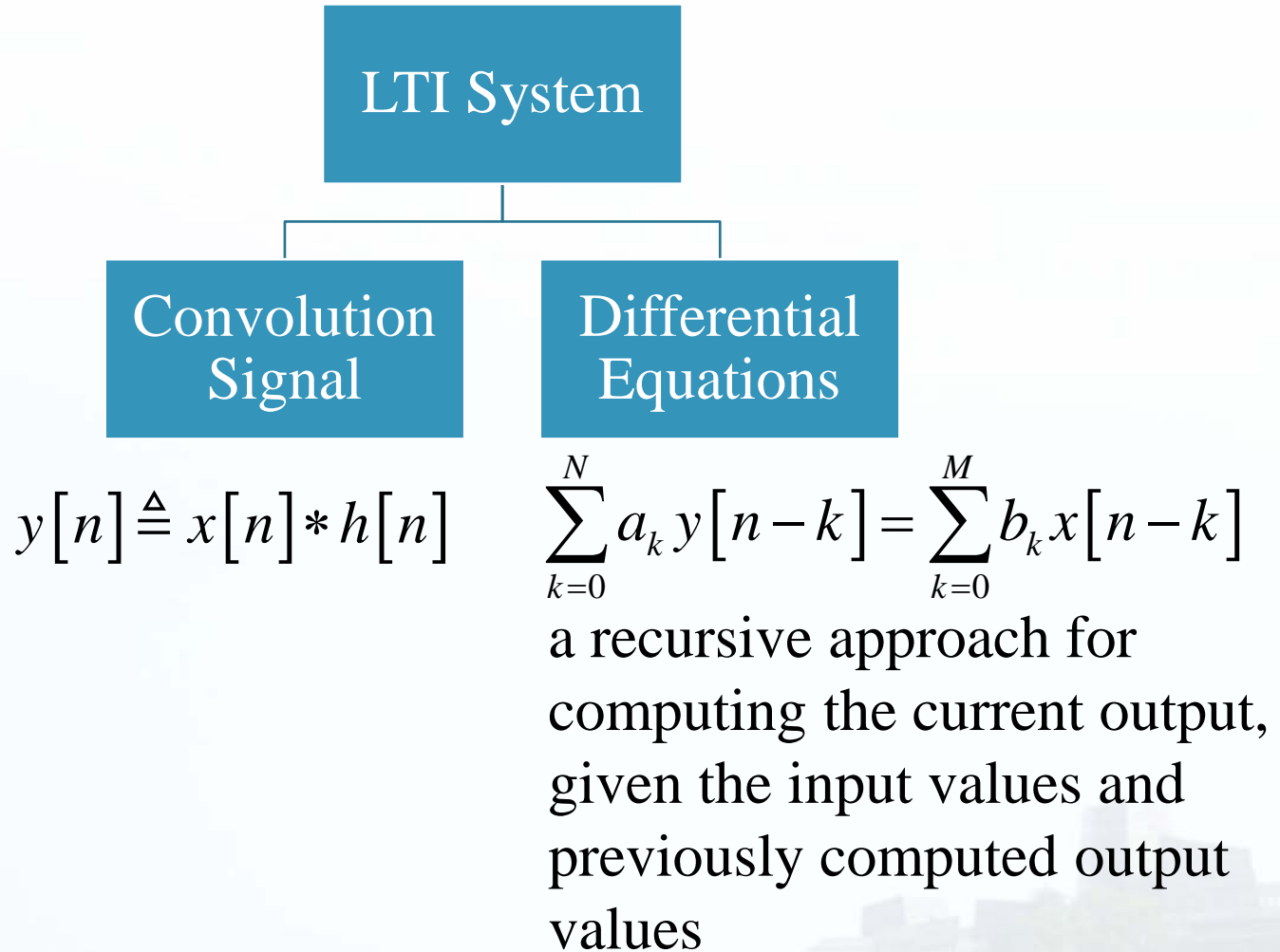
Pengolahan Sinyal Digital

Adhi Harmoko Saputro



Differential Equations

LTI System



Linear Constant-Coefficient Difference Equations

- In general, an N^{th} -order linear constant coefficient difference equation has the form

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

- The output is not uniquely specified for a given input
 - The initial conditions are required
 - Linearity, time invariance, and causality depend on the initial conditions
 - If initial conditions are assumed to be zero system is linear, time invariant, and causal

Linear Constant-Coefficient Difference Equations

- Example
 - Moving Average

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

- Difference Equation Representation

$$\sum_{k=0}^0 a_k y[n-k] = \sum_{k=0}^3 b_k x[n-k] \quad \text{where } a_k = b_k = 1$$

General Solution

- A general solution can be expressed as the sum of a homogeneous response (natural response), and a particular solution (forced response) of the system:

$$y[n] = y_h[n] + y_p[n]$$

- The concept of *initial rest* of the LTI causal system described by the difference equation here means that $x[n] = 0$ implies $y[n] = 0$

Recursive Solution

- In the discrete-time case, there are an alternative to find the Differential Equations

$$y[n] = \frac{1}{a_o} \left(\sum_{k=0}^M b_x x[n-k] - \sum_{k=1}^N a_k y[n-k] \right)$$

- This is often how digital filters are implemented on a computer or a digital signal processor board
- The response of a differential equation
 - The differential equation is discretized at a given sampling rate to obtain a difference equation
 - The response of the difference equation is computed recursively

Example

- Consider the difference equation

$$y[n] - \frac{5}{6}y[n-1] + \frac{1}{6}y[n-2] = 3x[n] - 2x[n-1]$$

- Obtaining the recursive form of the difference equation

$$y[n] = \frac{5}{6}y[n-1] - \frac{1}{6}y[n-2] + 3x[n] - 2x[n-1]$$

- Assuming initial rest and that the input is an impulse,
 - $y[-2] = y[-1] = 0$

Example

- the recursion can be started

$$\begin{aligned}y[0] &= \frac{5}{6}y[-1] - \frac{1}{6}y[-2] + 3x[0] - 2x[-1] \\&= \frac{5}{6}(0) - \frac{1}{6}(0) + 3(1) - 2(0) = 3\end{aligned}$$

$$\begin{aligned}y[1] &= \frac{5}{6}y[0] - \frac{1}{6}y[-1] + 3x[1] - 2x[0] \\&= \frac{5}{6}(3) - \frac{1}{6}(0) + 3(0) - 2(1) = \frac{1}{2}\end{aligned}$$

Example

$$\begin{aligned}y[2] &= \frac{5}{6}y[1] - \frac{1}{6}y[0] + 3x[2] - 2x[1] \\&= \frac{5}{6}\left(\frac{1}{2}\right) - \frac{1}{6}(3) + 3(0) - 2(0) = -\frac{1}{12}\end{aligned}$$

Matlab Implemantation

- A function called `filter` is available to solve difference equations numerically, given the input and the difference equation coefficients.

- In its simplest form this function is invoked by

$$y = \text{filter}(b,a,x)$$

- where

- $b = [b_0, b_1, \dots, b_M]$; $a = [a_0, a_1, \dots, a_N]$ are the coefficient arrays

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

Matlab Implemantation

- To compute and plot impulse response, MATLAB provides the function `impz`.
- When invoked by

$$h = \text{impz}(b, a, n)$$

- Computes samples of the impulse response of the filter at the sample indices given in `n` with numerator coefficients in `b` and denominator coefficients in `a`.

Matlab Example

- Given the following difference equation

$$y(n) - y(n - 1) + 0.9y(n - 2) = x(n);$$

- Calculate and plot the impulse response $h(n)$ at $n = -20, \dots, 100$.
- Calculate and plot the unit step response $s(n)$ at $n = -20, \dots, 100$.
- Is the system specified by $h(n)$ stable?

Matlab Example

- From the given difference equation the coefficient arrays are
 - $b = [1]; a = [1, -1, 0.9];$

```
>> b = [1]; a = [1, -1, 0.9]; n = [-20:120];  
>> h = impz(b,a,n);  
>> subplot(2,1,1); stem(n,h);  
>> title('Impulse Response');  
>> xlabel('n'); ylabel('h(n)')
```


Matlab Example

```
>> n=-20:100;  
>> x = (n-0) >= 0; % x = stepseq(0,-20,120);  
>> s = filter(b,a,x);  
>> subplot(2,1,2); stem(n,s)  
>> title('Step Response');  
>> xlabel('n'); ylabel('s(n)')
```

Matlab Example

- To determine the stability of the system, we have to determine $h(n)$ for all n .
- Although we have not described a method to solve the difference equation
- we can use the plot of the impulse response to observe that $h(n)$ is practically zero for $n > 120$. Hence the sum

$$\sum |h[n]|$$

```
>> sum(abs(h))
```

```
ans = 14.8785
```

which implies that the system is stable.

Example

- Let us consider the convolution given in previous example. The input sequence is of finite duration $x(n) = u(n) - u(n - 10)$
- while the impulse response is of infinite duration

$$h(n) = (0.9)^n u(n)$$

- Determine $y(n) = x(n) * h(n)$.

Example

- If the LTI system, given by the impulse response $h(n)$, can be described by a difference equation, then $y(n)$ can be obtained from the filter function.
- From the $h(n)$ expression

$$(0.9)h[n-1] = (0.9)(0.9)^{n-1}u[n-1] = (0.9)^n u[n-1]$$

$$\begin{aligned}h[n] - (0.9)h[n-1] &= (0.9)^n u[n] - (0.9)^n u[n-1] \\&= (0.9)^n (u[n] - u[n-1]) = (0.9)^n \delta[n] \\&= \delta[n]\end{aligned}$$

Example

- The last step follows from the fact that $\delta(n)$ is nonzero only at $n = 0$.
- By definition $h(n)$ is the output of an LTI system when the input is $\delta(n)$.
- Hence substituting $x(n)$ for $\delta(n)$ and $y(n)$ for $h(n)$, the difference equation is

$$y[n] - (0.9)y[n-1] = x[n]$$

Example

- MATLAB's filter function can be used to compute the convolution indirectly.

```
>> b = [1]; a = [1,-0.9];  
>> n = -5:50;  
>> x = stepseq(0,-5,50) - stepseq(10,-5,50);  
>> y = filter(b,a,x);  
>> subplot(2,1,2); stem(n,y);  
>> title('Output sequence')  
>> xlabel('n'); ylabel('y(n)'); axis([-5,50,-0.5,8])
```




Terima Kasih