CHAPTER 4

Discrete-Time Systems - I
Time-Domain Representation

These lecture slides are based on "Digital Signal Processing: A Computer-Based Approach, 4th ed." textbook by S.K. Mitra and its instructor materials. U.Sezen
Introduction

- A discrete-time system processes a given input sequence \( x[n] \) to generates an output sequence \( y[n] \) with more desirable properties.

- In most applications, the discrete-time system is a single-input, single-output system.

![Discrete-Time System Diagram]

- Mathematically, the discrete-time system is characterized by an operator \( \mathcal{H}(\cdot) \) that transforms the input sequence \( x[n] \) into another sequence \( y[n] \) at the output.

- The discrete-time system may also have more than one input and/or more than one output.

Examples

- 2-input, 1-output discrete-time systems:

  ![Modulator and Adder Diagram]

- 1-input, 1-output discrete-time systems:

  ![Multiplier, Unit Delay, and Unit Advance Diagram]
Examples

- A more complex example of an one-input, one-output discrete-time system is shown below

\[ y[n] = \sum_{\ell=-\infty}^{n} x[\ell] = \sum_{\ell=-\infty}^{n-1} x[\ell] + x[n] = y[n-1] + x[n] \]

- The output \( y[n] \) at time instant \( n \) is the sum of the input sample \( x[n] \) at time instant \( n \) and the previous output \( y[n-1] \) at time instant \( n-1 \) which is the sum of all previous input sample values from \( -\infty \) to \( n-1 \)

- The system cumulatively adds, i.e., it accumulates all input sample values
Input-output relation can also be written in the form

\[
y[n] = \sum_{\ell=-\infty}^{-1} x[\ell] + \sum_{\ell=0}^{n} x[\ell]
\]

\[
= y[-1] + \sum_{\ell=0}^{n} x[\ell], \quad n \geq 0
\]

The second form is used for a causal input sequence, in which case \(y[-1]\) is called the initial condition

M-point Moving-Average System:

\[
y[n] = \frac{1}{M} \sum_{k=0}^{M-1} x[n - k]
\]

Used in smoothing random variations in data

In most applications, the data \(x[n]\) is a bounded sequence, so

M-point average \(y[n]\) is also a bounded sequence

If there is no bias in the measurements, an improved estimate of the noisy data is obtained by simply increasing \(M\)

A direct implementation of the \(M\)-point moving average system requires \(M - 1\) additions, 1 division, and storage of \(M - 1\) past input data samples
A more efficient implementation is developed next

\[
y[n] = \frac{1}{M} \left( \sum_{k=1}^{M} x[n-k] + x[n] - x[n-M] \right)
\]

\[
= \frac{1}{M} \left( \sum_{k=0}^{M-1} x[n-1-k] + x[n] - x[n-M] \right)
\]

\[
= \frac{1}{M} \left( \sum_{k=0}^{M-1} x[n-1-k] \right) + \frac{1}{M} (x[n] - x[n-M])
\]

Hence

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

Computation of the modified \textit{M-point moving average system} using the recursive equation now requires 2 additions and 1 division

\textbf{An application:} Consider

\[
x[n] = s[n] + d[n]
\]

where \(s[n]\) is the signal corrupted by a noise \(d[n]\)
Example: \( s[n] = 2[n(0.9)^n] \) and \( d[n] \) is a random signal.

**Exponentially Weighted Running Average Filter:**

\[
y[n] = \alpha y[n-1] + x[n], \quad 0 < \alpha < 1
\]

- Computation of the running average requires only 1 addition, 1 multiplication and storage of the previous running average.
- Does not require storage of past input data samples.
- For \( 0 < \alpha < 1 \), the exponentially weighted average filter places more emphasis on current data samples and less emphasis on past data samples as illustrated below.

\[
y[n] = \alpha(\alpha y[n-2] + x[n-1]) + x[n]
= \alpha^2 y[n-2] + \alpha x[n-1] + x[n]
= \alpha^2(\alpha y[n-3] + x[n-2] + \alpha x[n-1]) + x[n]
= \alpha^3 y[n-3] + \alpha^2 x[n-2] + \alpha x[n-1] + x[n]
\]
Linear interpolation: Employed to estimate sample values between pairs of adjacent sample values of a discrete-time sequence

Factor-of-4 interpolation

\[
\begin{align*}
\text{y}[n] &= x_u[n] + \frac{1}{2} (x_u[n - 1] + x_u[n + 1]) \\
\text{Factor-of-3 linear interpolator} &\quad y[n] = x_u[n] + \frac{1}{3} (x_u[n - 1] + x_u[n + 2]) + \frac{2}{3} (x_u[n - 2] + x_u[n + 1])
\end{align*}
\]
Factor of 2 linear 2-D interpolator

Median Filter: The median of a set of \((2K + 1)\) numbers is the number such that \(K\) numbers from the set have values greater than this number and the other \(K\) numbers have values smaller.

Median can be determined by rank-ordering the numbers in the set by their values and choosing the number at the middle.

Example: Consider the set of numbers

\[\{2, -3, 10, 5, 1\}\]

Rank-ordered set is given by

\[\{-3, -1, 2, 5, 10\}\]

Hence,

\[\text{med}\ \{2, -3, 10, 5, 1\} = 2\]
- Implemented by sliding a window of odd length over the input sequence \( \{x[n]\} \) one sample at a time
- Output \( y[n] \) at instant \( n \) is the median value of the samples inside the window centered at \( n \)
- Finds applications in removing additive random noise, which shows up as sudden large errors in the corrupted signal
- Usually used for the smoothing of signals corrupted by impulse noise

Example:

(a) Original data
(b) Impulse noise corrupted data
(c) Median filtered noisy data
Classification

- Linear System
- Shift-Invariant System
- Causal System
- Stable System
- Passive and Lossless Systems

Definition: If $y_1[n]$ is the output due to one input $x_1[n]$ and $y_2[n]$ is the output due to another input $x_2[n]$ then for an input

$$x[n] = \alpha x_1[n] + \beta x_2[n]$$

the output is given by

$$y[n] = \alpha y_1[n] + \beta y_2[n]$$

Above property must hold for any arbitrary constants $\alpha$ and $\beta$ and for all possible inputs $x_1[n]$ and $x_2[n]$.
Example: Consider two accumulators with

\[ y_1[n] = \sum_{\ell=-\infty}^{n} x_1[\ell] \quad \text{and} \quad y_2[n] = \sum_{\ell=-\infty}^{n} x_2[\ell] \]

For an input

\[ x[n] = \alpha x_1[n] + \beta x_2[n] \]

the output is

\[ y[n] = \sum_{\ell=-\infty}^{n} (\alpha x_1[\ell] + \beta x_2[\ell]) \]

\[ = \alpha \sum_{\ell=-\infty}^{n} x_1[\ell] + \beta \sum_{\ell=-\infty}^{n} x_2[\ell] \]

\[ = \alpha y_1[n] + \beta y_2[n] \]

Hence, the above system is linear

Example: The median filter described earlier is a nonlinear discrete-time system.

To show this, consider a median filter with a window of length 3

Output \( y_1[n] \) of the filter for an input \( x_1[n] \),

\[ \{x_1[n]\} = \{3, 4, 5\}, \quad 0 \leq n \leq 2 \]

is

\[ \{y_1[n]\} = \{3, 4, 4\}, \quad 0 \leq n \leq 2 \]

Output \( y_2[n] \) of the filter for another input \( x_2[n] \),

\[ \{x_2[n]\} = \{2, -1, -1\}, \quad 0 \leq n \leq 2 \]

is

\[ \{y_1[n]\} = \{0, -1, -1\}, \quad 0 \leq n \leq 2 \]
However, the output $y[n]$ for the input, $x[n] = x_1[n] + x_2[n]$, 

$$\{x[n]\} = \{5, 3, 4\}, \quad 0 \leq n \leq 2$$

is 

$$\{y[n]\} = \{3, 4, 3\}, \quad 0 \leq n \leq 2$$

**Note:**

$$\{y_1[n] + y_2[n]\} = \{3, 3, 3\} \neq \{y[n]\}$$

- Hence, the median filter is a **nonlinear** discrete-time system

---

**Shift-Invariant System**

- For a shift-invariant system, if $y_1[n]$ is the response to an input $x_1[n]$, then the response to an input 

$$x[n] = x_1[n - n_0]$$

is simply 

$$y[n] = y_1[n - n_0]$$

where $n_0$ is any positive or negative integer

- The above relation must hold for any arbitrary input and its corresponding output
In the case of sequences and systems with indices \( n \) related to discrete instants of time, the above property is called **time-invariance property**.

Time-invariance property ensures that for a specified input, the output is independent of the time the input is being applied.

**Example:**

Consider the upsampler:

\[
x[n] \xrightarrow{\uparrow L} x_u[n]
\]

with an input-output relation given by

\[
x_u[n] = \begin{cases} 
  x[n/L], & n = 0, \pm L, \pm 2L, \ldots \\
  0, & \text{otherwise}
\end{cases}
\]

For an input \( x_1[n] = x[n - n_0] \) the output \( x_{1,u}[n] \) is given by

\[
x_{1,u}[n] = \begin{cases} 
  x_1[n/L], & n = 0, \pm L, \pm 2L, \ldots \\
  0, & \text{otherwise}
\end{cases}
\]

\[
= \begin{cases} 
  x[n/L - n_0], & n = 0, \pm L, \pm 2L, \ldots \\
  0, & \text{otherwise}
\end{cases}
\]
However from the definition of the up-sampler

\[
x_u[n - n_0] = \begin{cases} 
  x_1[(n - n_0)/L], & n = 0, \pm L, \pm 2L, \ldots \\
  0, & \text{otherwise}
\end{cases}
\]

\[\neq x_{1,u}[n]\]

- Hence, the upsampler is a **time-varying** system

---

**Linear Time-Invariant (LTI) System**

- **Linear Time-Invariant (LTI)** system is a system satisfying both the linearity and the time-invariance property

- LTI systems are mathematically easy to analyze and characterize, and consequently, easy to design

- Highly useful signal processing algorithms have been developed utilizing this class of systems over the last several decades
Causal System

- In a causal system, the \( n_0 \)-th output sample \( y[n_0] \) depends only on input samples \( x[n] \) for \( n \leq n_0 \) and does not depend on input samples for \( n > n_0 \).

- Let \( y_1[n] \) and \( y_2[n] \) be the responses of a causal discrete-time system to the inputs \( x_1[n] \) and \( x_2[n] \), respectively.

Then

\[
x_1[n] = x_2[n] \quad \text{for } n < N
\]

implies also that

\[
y_1[n] = y_2[n] \quad \text{for } n < N
\]

- For a causal system, changes in output samples do not precede changes in the input samples.

Examples:

- Examples of causal systems:

\[
\begin{align*}
y[n] &= \alpha_1 x[n] + \alpha_2 x[n - 1] + \alpha_3 x[n - 2] + \alpha_4 x[n - 3] \\
y[n] &= b_0 x[n] + b_1 x[n - 1] + b_2 x[n - 2] + a_1 y[n - 1] + a_2 y[n - 2] \\
y[n] &= y[n - 1] + x[n]
\end{align*}
\]

- Examples of noncausal systems:

\[
\begin{align*}
y[n] &= x_u[n] + \frac{1}{2}(x_u[n - 1] + x_u[n + 1]) \\
y[n] &= x_u[n] + \frac{1}{3}(x_u[n - 1] + x_u[n + 2]) + \frac{2}{3}(x_u[n - 2] + x_u[n + 1])
\end{align*}
\]
▶ A noncausal system can be implemented as a causal system by delaying the output by an appropriate number of samples.

▶ For example, a causal implementation of the factor-of-2 interpolator is given by:

\[ y[n] = x_u[n - 1] + \frac{1}{2}(x_u[n - 2] + x_u[n]) \]

---

**Stable System**

▶ There are various definitions of stability.

▶ We consider here the bounded-input, bounded-output (BIBO) stability, i.e.,

If \( y[n] \) is the response to an input \( x[n] \) and if

\[ |x| \leq B_x \quad \text{for all values of } n \]

then

\[ |y| \leq B_y \quad \text{for all values of } n \]

where \( B_x < \infty \) and \( B_y < \infty \).
**Example:** The $M$-point moving average filter is **BIBO stable**

\[ y[n] = \frac{1}{M} \sum_{k=0}^{M-1} x[n - k] \]

For a bounded input we have

\[ |y[n]| = \left| \frac{1}{M} \sum_{k=0}^{M-1} x[n - k] \right| \leq \frac{1}{M} \sum_{k=0}^{M-1} |x[n - k]| \leq \frac{1}{M} (M B_x) \leq B_x \]

---

**Passive and Lossless Systems**

- A discrete-time system is defined to be **passive** if, for every finite-energy input $x[n]$, the output $y[n]$ has, at most, the same energy, i.e.

  \[ \sum_{n=-\infty}^{\infty} |y[n]|^2 \leq \sum_{n=-\infty}^{\infty} |x[n]|^2 < \infty \]

- For a **lossless** system, the above inequality is satisfied with an equal sign for every input.
Example: Consider the discrete-time system defined by $y[n] = \alpha x[n - N]$ with $N$ a positive integer.

Its output energy is given by

$$\sum_{n=-\infty}^{\infty} |y[n]|^2 = |\alpha|^2 \sum_{n=-\infty}^{\infty} |x[n]|^2$$

Hence, it is a passive system if $|\alpha| \leq 1$ and is a lossless system if $|\alpha| = 1$.

Impulse and Step Responses

The response of a discrete-time system to a unit sample sequence $\{\delta[n]\}$ is called the unit sample response or simply, the impulse response, and is denoted by $\{h[n]\}$.

The response of a discrete-time system to a unit step sequence $\{\mu[n]\}$ is called the unit step response or simply, the step response, and is denoted by $\{s[n]\}$.
Example: The impulse response $h[n]$ of the discrete-time accumulator

$$y[n] = \sum_{\ell=-\infty}^{n} x[\ell]$$

is obtained by setting

$$x[n] = \delta[n]$$

resulting in

$$h[n] = \sum_{\ell=-\infty}^{n} \delta[\ell]$$

$$= \mu[n]$$

Example: The impulse response $h[n]$ of the factor-of-2 interpolator

$$y[n] = x_u[n] + \frac{1}{2}(x_u[n-1] + x_u[n+1])$$

is obtained by setting

$$x_u[n] = \delta[n]$$

and is given by

$$h[n] = \delta[n] + \frac{1}{2}(\delta[n-1] + \delta[n+1])$$

The impulse response $\{h[n]\}$ is thus a finite-length sequence of length 3:

$$\{h[n]\} = \{0.5, 1, 0.5\}, \quad -1 \leq n \leq 1$$
Input-Output Relationship

▶ A consequence of the linear, time-invariance property is that an LTI discrete-time system is completely characterized by its impulse response

▶ Thus, knowing the impulse response one can compute the output of the system for any arbitrary input

Example:

▶ Let $h[n]$ denote the impulse response of a LTI discrete-time system, and compute its output $y[n]$ for the input:

$$x[n] = 0.5 \delta[n + 2] + 1.5 \delta[n - 1] - \delta[n - 2] + 0.75 \delta[n - 5]$$

As the system is linear, we can compute its outputs for each member of the input separately and add the individual outputs to determine $y[n]$

Since the system is time-invariant

<table>
<thead>
<tr>
<th>input</th>
<th>output</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\delta[n + 2]$</td>
<td>$h[n + 2]$</td>
</tr>
<tr>
<td>$\delta[n - 1]$</td>
<td>$h[n - 1]$</td>
</tr>
<tr>
<td>$\delta[n - 2]$</td>
<td>$h[n - 2]$</td>
</tr>
<tr>
<td>$\delta[n - 5]$</td>
<td>$h[n - 5]$</td>
</tr>
</tbody>
</table>
Likewise, as the system is linear

\[
\begin{align*}
\text{input} & \quad \text{output} \\
0.5 \delta[n+2] & \rightarrow 0.5 h[n+2] \\
1.5 \delta[n-1] & \rightarrow 1.5 h[n-1] \\
-\delta[n-2] & \rightarrow -h[n-2] \\
0.75 \delta[n-5] & \rightarrow 0.75 h[n-5]
\end{align*}
\]

Hence, because of the linearity property we get

\[
y[n] = 0.5 h[n+2] + 1.5 h[n-1] - h[n-2] + 0.75 h[n-5]
\]

Now, any arbitrary input sequence \(x[n]\) can be expressed as a linear combination of delayed and advanced unit sample sequences in the form

\[
x[n] = x[n] \ast \delta[n] = \sum_{k=-\infty}^{\infty} x[k] \delta[n-k]
\]

The response of the LTI system to an input \(x[k] \delta[n-k]\) will be \(x[k]h[n-k]\)

Hence, the response \(y[n]\) to the input above is given by

\[
y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k]
\]

which can be alternately written as

\[
y[n] = \sum_{k=-\infty}^{\infty} x[n-k]h[k]
\]
The summation

\[ y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n - k] = \sum_{k=-\infty}^{\infty} x[n - k]h[k] \]

is thus the convolution sum of the sequences \( x[n] \) and \( h[n] \) and represented compactly as

\[ y[n] = x[n] \ast h[n] \]

**Example:**

Consider an LTI discrete-time system with an impulse response \( h[n] \) generating an output \( y[n] \) for a input \( x[n] \):

\[ y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n - k] = x[n] \ast h[n] \]

Let us determine the output \( y_1[n] \) of an LTI discrete-time system with an impulse response \( h[n - N_0] \) for the same input \( x[n] \):

\[ y_1[n] = \sum_{k=-\infty}^{\infty} x[k]h[n - N_0 - k] = x[n] \ast h[n - N_0] \]

Hence,

\[ y_1[n] = y[n - N_0] \]
Convolution Sum Properties

- **Commutative property:**
  \[ x[n] \ast h[n] = h[n] \ast x[n] \]

- **Associative property:**
  \[ (x[n] \ast h[n]) \ast y[n] = x[n] \ast (h[n] \ast y[n]) \]

- **Distributive property:**
  \[ x[n] \ast (h[n] + y[n]) = x[n] \ast h[n] + x[n] \ast y[n] \]

In practice, if either the input or the impulse response is of finite length, the convolution sum can be used to compute the output sample as it involves a finite sum of products.

- If both the input sequence and the impulse response sequence are of finite length, the output sequence is also of finite length.

- If both the input sequence and the impulse response sequence are of infinite length, convolution sum cannot be used to compute the output.

- For systems characterized by an infinite impulse response sequence, an alternate time-domain description involving a finite sum of products will be considered.
Can be used to convolve two finite-length sequences

Consider the convolution of \( \{g[n]\}, 0 \leq n \leq 3, \) with
\( \{h[n]\}, 0 \leq n \leq 2, \) generating the sequence \( y[n] = g[n] \circledast h[n] \)

Samples of \( \{g[n]\} \) and \( \{h[n]\} \) are then multiplied using the conventional multiplication method without any carry operation as shown on the next slide.

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>( g[n] )</td>
<td>( g[0] )</td>
<td>( g[1] )</td>
<td>( g[2] )</td>
<td>( g[3] )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( h[n] )</td>
<td>( h[0] )</td>
<td>( h[1] )</td>
<td>( h[2] )</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>( g[0]h[0] )</td>
<td>( g[1]h[0] )</td>
<td>( g[2]h[0] )</td>
<td>( g[3]h[0] )</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The samples \( y[n] \) generated by the convolution sum are obtained by adding the entries in the column above each sample.

The samples of \( \{y[n]\} \) are given by

\[
\begin{align*}
y[0] & = g[0]h[0] \\
y[1] & = g[1]h[0] + g[0]h[1] \\
\end{align*}
\]
The method can also be applied to convolve any two finite-length two-sided sequences as explained below.

Consider two sequences \( \{x_1[n]\} \) with \( N_1 \leq n \leq N_2 \) and \( \{x_2[n]\} \) with \( N_a \leq n \leq N_b \) and we are asked to compute the convolution \( y_1[n] = x_1[n] \ast x_2[n] \) of these two sequences of size \( N_2 - N_1 + N_b - N_a + 1 \)

1. Create two causal sequences \( g[n] = x_1[n + N_1] \) with \( 0 \leq n \leq N_2 - N_1 \) and \( h[n] = x_2[n + N_a] \) with \( 0 \leq n \leq N_b - N_a \), where both have their first elements at \( n = 0 \).

2. Compute the convolution \( y[n] = g[n] \ast h[n] \) using the tabular method explained on the previous slide. Here \( \{y[n]\} \) is defined for \( 0 \leq n \leq N_2 - N_1 + N_b - N_a \).

3. Then, obtain the real convolution \( y_1[n] \) as

\[
y_1[n] = y[n - N_1 - N_b]
\]

Convolution Using MATLAB

The M-file `conv` implements the convolution sum of two finite-length sequences.

If

\[
a = [-2 \ 0 \ 1 \ -1 \ 3] \\
n = [1 \ 2 \ 0 \ -1]
\]

then `conv(a,b)` yields

\[
[-2 \ -4 \ 1 \ 3 \ 1 \ 5 \ 1 \ -3]
\]
Stability Condition of an LTI Discrete-Time System

- **BIBO Stability Condition:** A discrete-time is BIBO stable if and only if the output sequence \( \{y[n]\} \) remains bounded for all bounded input sequence \( \{x[n]\} \).

- An LTI discrete-time system is BIBO stable if and only if its impulse response sequence \( \{h[n]\} \) is absolutely summable, i.e.

  \[
  S = \sum_{n=-\infty}^{\infty} |h[n]| < \infty
  \]

- Proof can be found in the textbook.

**Example:**

- Consider an LTI discrete-time system with an impulse response

  \[ h[n] = \alpha^n \mu[n] \]

- For this system

  \[
  S = \sum_{n=-\infty}^{\infty} |\alpha|^n \mu[n] = \sum_{n=0}^{\infty} |\alpha|^n = \frac{1}{1 - |\alpha|}
  \]

- Therefore \( S < \infty \) if \( |\alpha| < 1 \) for which the system is BIBO stable.

- If \( |\alpha| = 1 \), the system is not BIBO stable.
Causality Condition of an LTI Discrete-Time System

- Let $x_1[n]$ and $x_2[n]$ be two input sequences with
  
  $x_1[n] = x_2[n]$ for $n \leq n_0$
  
  $x_1[n] \neq x_2[n]$ for $n > n_0$

  then the system is causal if the corresponding outputs $y_1[n]$ and $y_2[n]$ are also given by

  $y_1[n] = y_2[n]$ for $n \leq n_0$
  
  $y_1[n] \neq y_2[n]$ for $n > n_0$

- An LTI discrete-time system is causal if and only if its impulse response $\{h[n]\}$ is a causal sequence.
- Proof can be found in the textbook.

Examples:

- The discrete-time system defined by

  $y[n] = \alpha_1 x[n] + \alpha_2 x[n-1] + \alpha_3 x[n-2] + \alpha_4 x[n-3]$

  is a causal system as it has a causal impulse response

  $\{h[n]\} = \{[\alpha_1 \alpha_2 \alpha_3 \alpha_4]\}, \ 0 \leq n \leq 3$

  then the system is causal if the corresponding outputs $y_1[n]$ and $y_2[n]$ are also given by

- The discrete-time accumulator defined by

  $y[n] = \sum_{\ell=-\infty}^{n} x[\ell]$

  is a causal system as it has a causal impulse response given by

  $h[n] = \sum_{\ell=-\infty}^{n} \delta[\ell] = \mu[n]$
Examples:

- The factor-of-2 interpolator defined by

\[ y[n] = x_u[n] + \frac{1}{2}(x_u[n - 1] + x_u[n + 1]) \]

is noncausal as it has a noncausal impulse response given by

\[ \{h[n]\} = \{[0.5, 1, 0.5]\}, \quad -1 \leq n \leq 1 \]

- Note: A noncausal LTI discrete-time system with a finite-length impulse response can often be realized as a causal system by inserting an appropriate amount of delay.

  For example, a causal version of the factor-of-2 interpolator is obtained by delaying the input by one sample period

\[ y[n] = x_u[n - 1] + \frac{1}{2}(x_u[n - 2] + x_u[n]) \]
Simple Interconnection Schemes

Two simple interconnection schemes are:

- Cascade Connection
- Parallel Connection

**Cascade Connection**

\[ x[n] \xrightarrow{h_1[n]} h_2[n] \xrightarrow{y[n]} \quad \equiv \quad x[n] \xrightarrow{h_2[n]} h_1[n] \xrightarrow{y[n]} \]

\[ \equiv \quad x[n] \xrightarrow{h_1[n] \ast h_2[n]} y[n] \]

- Impulse response \( h[n] \) of the cascade of two LTI discrete-time systems with impulse responses \( h_1[n] \) and \( h_2[n] \) is given by

\[ h[n] = h_1[n] \ast h_2[n] \]

- **Note:** The ordering of the systems in the cascade has no effect on the overall impulse response because of the commutative property of convolution

- A cascade connection of two stable systems is stable

- A cascade connection of two passive (lossless) systems is also passive (lossless)
An application is in the development of an inverse system as explained below:

If the cascade connection satisfies the relation

\[ h_1[n] \ast h_2[n] = \delta[n] \]

then the LTI system \( h_1[n] \) is said to be the inverse of \( h_2[n] \) and vice-versa

An application of the inverse system concept is in the recovery of a signal \( x[n] \) from its distorted version \( \hat{x}[n] \) appearing at the output of a transmission channel.

If the impulse response of the channel is known, then \( x[n] \) can be recovered by designing an inverse system of the channel

\[ h_1[n] \ast h_2[n] = \delta[n] \]
Example:

- Consider the discrete-time accumulator with an impulse response $\mu[n]$. Its inverse system satisfy the condition
  
  $$\mu[n] \ast h_2[n] = \delta[n]$$
  
  - It follows from the above that $h_2[n] = 0$ for $n < 0$ and
    
    $$h_2[0] = 1$$
    
    $$\sum_{\ell=0}^{n} h_2[\ell] = 0 \quad \text{for } n \geq 1$$
  
  - Thus the impulse response of the inverse system of the discrete-time accumulator is given by
    
    $$h_2[n] = \delta[n] - \delta[n - 1]$$
    
    which is called a **backward difference system**

Parallel Connection

- Impulse response $h[n]$ of the parallel connection of two LTI discrete-time systems with impulse responses $h_1[n]$ and $h_2[n]$ is given by
  
  $$h[n] = h_1[n] + h_2[n]$$
Example:

- Consider the discrete-time system shown in the figure below

\[
\begin{align*}
    h_1[n] &= \delta[n] + 0.5\delta[n-1] \\
    h_2[n] &= 0.5\delta[n] - 0.25\delta[n-1] \\
    h_3[n] &= 2\delta[n] \\
    h_4[n] &= -2(0.5)^n\mu[n]
\end{align*}
\]

where

\[
\begin{align*}
    h_1[n] &= \delta[n] + 0.5\delta[n-1] \\
    h_2[n] &= 0.5\delta[n] - 0.25\delta[n-1] \\
    h_3[n] &= 2\delta[n] \\
    h_4[n] &= -2(0.5)^n\mu[n]
\end{align*}
\]

Simplifying the block-diagram we obtain

\[
\begin{align*}
    h_1[n] &= h_1[n] + h_2[n] \ast (h_3[n] + h_4[n]) \\
    &= h_1[n] + h_2[n] \ast h_3[n] + h_2[n] \ast h_4[n]
\end{align*}
\]

Overall impulse response \(h[n]\) is given by

\[
\begin{align*}
    h[n] &= h_1[n] + h_2[n] \ast (h_3[n] + h_4[n]) \\
    &= h_1[n] + h_2[n] \ast h_3[n] + h_2[n] \ast h_4[n]
\end{align*}
\]

Now,

\[
\begin{align*}
    h_2[n] \ast h_3[n] &= \left(\frac{1}{2}\delta[n] - \frac{1}{4}\delta[n-1]\right) \ast 2\delta[n] \\
    &= \delta[n] - \frac{1}{2}\delta[n-1]
\end{align*}
\]
and

\[ h_2[n] \odot h_4[n] = \left( \frac{1}{2} \delta[n] - \frac{1}{4} \delta[n-1] \right) \odot (-2(0.5)^n \mu[n]) \]

\[ = -(0.5)^n \mu[n] + \frac{1}{2} (0.5)^{n-1} \mu[n-1] \]

\[ = -(0.5)^n \mu[n] + (0.5)^n \mu[n-1] \]

\[ = -(0.5)^n (\mu[n] - \mu[n-1]) \]

\[ = -(0.5)^n \delta[n] \]

\[ = -\delta[n] \]

Therefore

\[ h[n] = \delta[n] + \frac{1}{2} \delta[n-1] + \delta[n] - \frac{1}{2} \delta[n-1] - \delta[n] \]

\[ = \delta[n] \]

▶ An important subclass of LTI discrete-time systems is characterized by a linear constant coefficient difference equation of the form

\[ \sum_{k=0}^{N} d_k y[n - k] = \sum_{k=0}^{M} p_k x[n - k] \]

where \( x[n] \) and \( y[n] \) are, respectively, the input and the output of the system, and, \( \{d_k\} \) and \( \{p_k\} \) are constants characterizing the system

▶ The order of the system is given by \( \max(N, M) \), which is the order of the difference equation

▶ It is possible to implement an LTI system characterized by a constant coefficient difference equation as here the computation involves two finite sums of products
If we assume the system to be causal, then the output $y[n]$ can be recursively computed using

$$y[n] = -\sum_{k=1}^{N} \frac{d_k}{d_0} y[n-k] + \sum_{k=0}^{M} \frac{p_k}{d_0} x[n-k]$$

where $d_0 \neq 0$

- $y[n]$ can be computed for all $n \geq n_0$, knowing $x[n]$ and the initial conditions $y[n_0], y[n_1], \ldots, y[n-N]$

---

**Finite Impulse Response (FIR) Discrete-Time Systems**

Based on Impulse Response Length:

- If the impulse response $h[n]$ is of finite length, i.e.,

  $$h[n] = 0 \quad \text{for } n < N_1 \text{ and } n > N_2, N_1 < N_2$$

  then it is known as a **finite impulse response (FIR) discrete-time system**

- The convolution sum description here is

  $$y[n] = \sum_{k=N_1}^{N_2} h[k] x[n-k]$$
The output $y[n]$ of an FIR LTI discrete-time system can be computed directly from the convolution sum as it is a finite sum of products.

Examples of FIR LTI discrete-time systems are the moving-average system and the linear interpolators.

If the impulse response is of infinite length, then it is known as an infinite impulse response (IIR) discrete-time system.

The class of IIR systems we are concerned with in this course are characterized by linear constant coefficient difference equations.
Examples:

- **Example:** The discrete-time accumulator defined by
  \[ y[n] = y[n-1] + x[n] \]
  is seen to be an IIR system

- **Example:** The familiar numerical integration formulas that are used to numerically solve integrals of the form
  \[ y(t) = \int_0^t x(\tau) \, d\tau \]
  can be shown to be characterized by linear constant coefficient difference equations, and hence, are examples of IIR systems

If we divide the interval of integration into \( n \) equal parts of length \( T \), then the previous integral can be rewritten as

\[ y(nT) = y((n-1)T) + \int_{(n-1)T}^{nT} x(\tau) \, d\tau \]

where we have set \( t = nT \) and used the notation

\[ y(nT) = \int_0^{nT} x(\tau) \, d\tau \]

Using the trapezoidal method we can write

\[ \int_{(n-1)T}^{nT} x(\tau) \, d\tau = \frac{T}{2} \{ x((n-1)T) + x(nT) \} \]
Hence, a numerical representation of the definite integral is given by

\[ y(nT) = y((n-1)T) + \frac{T}{2} \{x((n-1)T) + x(nT)\} \]

Let \( y[n] = y(nT) \) and \( x[n] = x(nT) \)

Then

\[ y(nT) = y((n-1)T) + \frac{T}{2} \{x((n-1)T) + x(nT)\} \]

reduces to

\[ y[n] = y[n-1] + \frac{T}{2} \{x[n-1] + x[n]\} \]

which is recognized as the difference equation representation of a first-order IIR discrete-time system.

Nonrecursive and Recursive Systems

Based on the Output Calculation Process:

- **Nonrecursive System**: Here the output can be calculated sequentially, knowing only the present and past input samples.

- **Recursive System**: Here the output computation involves past output samples in addition to the present and past input samples.
Real and Complex Systems

Based on the Coefficients:

- **Real Discrete-Time System**: The impulse response samples are real valued
- **Complex Discrete-Time System**: The impulse response samples are complex valued