Discrete-Time Signals & Systems

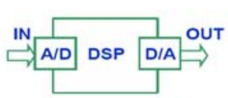
Maryam Mahsal Khan (Lecturer)
- B.Sc (CSE) – UET Peshawar
- M.Sc(EE) – UTP Malaysia

Outlines

- Introduction
- What are Discrete-Time Signals?
- What are Discrete-Time Systems?
- Linear Time Invariant System & its Properties
- Linear Constant Coefficient Difference Equation

Introduction

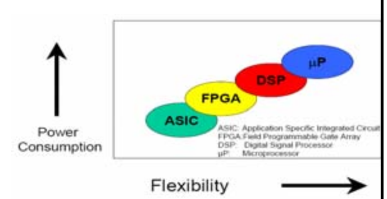
- Digital Signal Processing (def)
  - *Processing of digital signals using various algorithms.*
  - *Representation, transformation and manipulation of signals and the information they contain.*
- Applications
  - Communications
  - Space Exploration
  - Medicine
  - Archaeology
  - Speech
  - Image processing
  - e.g. cellular phone, fax/modem, radio, disk drives



- Primary Concern of DSP – Real Time Signal Processing.
- Analog based system – sensitive to temperature changes

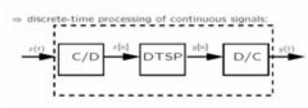
Introduction

- DSP based system – less affected by environmental conditions
  - Cost Effective
  - Easily reprogrammed
  - Precise mathematical Operations
  - High Processing Speed
  - Accuracy
  - Storage

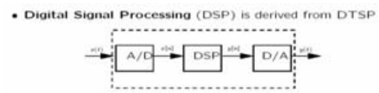


Difference b/w DSP and DTSP

- DTSP
  - Sampling of analog signals at discrete instants of time
  - Magnitude is the same as that of the analog signal

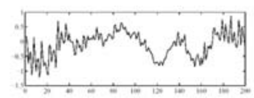


- DSP
  - Sampling of analog signals at discrete instants of time
  - Quantization to the nearest integer value.

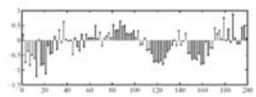


Signal

- *Physical quantity that varies with some independent variable (s) like time*. e.g.
  - Speech, Seismic signal, ECG , EEG etc.
- Types



Continuous Time Signal
(defined for every value of time)



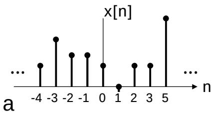
Discrete Time Signal
(defined at certain specific value of time)

Discrete-Time Signals

- A discrete-time signal is an indexed sequence of real or complex numbers. Thus, it is function of an integer-valued variable, n and is denoted by x[n].

$$x[n] = (\ldots 2\ 4\ 3\ 3\ 2\ 0\ 2\ 2\ 5 \ldots)$$
$$\uparrow$$

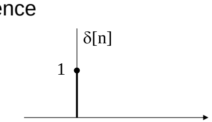
- A discrete-time signal is undefined for non-integer values of n.
- A real-valued signal x[n] will be represented graphically in the form of a *lollipop* plot as shown.



Unit Impulse Sequence $\delta[n]$

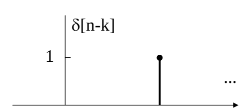
- The *unit impulse* (or *unit sample*) sequence δ[n], is defined as

$$\delta[n] = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases}$$



- The delayed/shifted unit impulse/sample sequence δ[n-k] is defined by

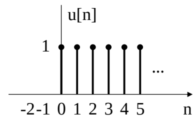
$$\delta[n-k] = \begin{cases} 1 & n = k \\ 0 & n \neq k \end{cases}$$



Unit Step Sequence $u[n]$

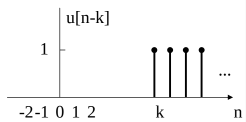
- The *unit step* sequence u[n], is defined as

$$u[n] = \begin{cases} 1 & n \geq 0 \\ 0 & n < 0 \end{cases}$$



- The shifted unit step sequence u[n-k] is defined as

$$u[n-k] = \begin{cases} 1 & n \geq k \\ 0 & n < k \end{cases}$$



Relation of Unit Impulse with Unit Step Sequence

- From the definitions of δ[n] and δ[n-k], it is readily seen that

$$x[n]\delta[n] = x[0]\delta[n]$$
$$x[n]\delta[n-k] = x[k]\delta[n-k]$$

- Note that δ[n] and u[n] are related by

$$u[n] = \sum_{k=-\infty}^{n} \delta[k],$$
$$\delta[n] = u[n] - u[n-1]$$

- Any sequence x[n] can be expressed as

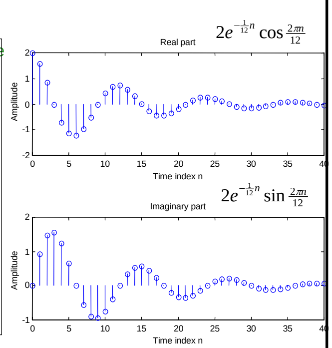
$$x[n] = \sum_{k=-\infty}^{\infty} x[k]\delta[n-k]$$

Complex-valued Exponential Sequences

```
% Generation of a complex exp. sequence
```

$$x[n] = 2\exp(-\tfrac{1}{12} + j\tfrac{\pi}{6})n$$

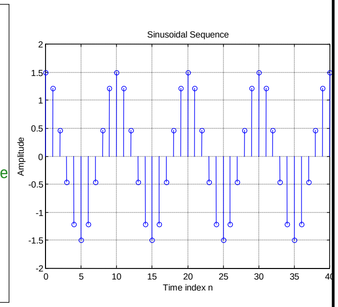
```
clf;
c = -(1/12)+(pi/6)*i;
K = 2;
n = 0:40;
x = K*exp(c*n);
subplot(2,1,1);
stem(n,real(x));
xlabel('Time index n');ylabel('Amplitude');
title('Real part');
subplot(2,1,2);
stem(n,imag(x));
xlabel('Time index n');ylabel('Amplitude');
title('Imaginary part');
```



Ref: Program P1_2, DSP Lab using Matlab, Mitra

Sinusoidal Sequences

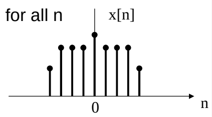
```
% Generation of a sinusoidal sequence
n = 0:40;
f = 0.1;
phase = 0;
A = 1.5;
arg = 2*pi*f*n - phase;
x = A*cos(arg);
clf;          % Clear old graph
stem(n,x);% Plot the generated sequence
axis([0 40 -2 2]);
grid;
title('Sinusoidal Sequence');
xlabel('Time index n');
ylabel('Amplitude');
axis;
```



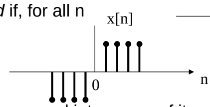
Ref: Program P1_4, DSP Lab using Matlab, Mitra

Symmetric Sequences

- A real-valued sequence x[n] is said to be *even* if, for all n
  - x[n] = x[-n]



- whereas it is said to be *odd* if, for all n
  - x[n] = -x[-n]



- Any signal x[n] can be decomposed into a sum of its even part $x_e[n]$, and its odd part $x_o[n]$, as follows:

  $x[n] = x_e[n] + x_o[n]$,

  where

  $x_e[n] = 1/2 \{x[n]+x[-n]\}$,

  $x_o[n] = 1/2 \{x[n]-x[-n]\}$.

Example:
x[n]=(3 6 2 10 1 8 5)
x[-n]=(5 8 1 10 2 6 3)

x_e[n]=(4 7 1.5 10 1.5 7 4)
x_o[n]=(-1 -1 0.5 0 0.5 1 1)

Periodic and Aperiodic Sequences

- In the discrete-time case, a periodic sequence is a sequence for which

$$x[n] = x[n+N], \text{ for all } n, \text{ where the period } N \text{ is an integer.}$$

- If this condition for periodicity is tested for the discrete-time sinusoid, then

$$A\cos(\omega_0 n + \phi) = A\cos(\omega_0 n + \omega_0 N + \phi),$$

which requires that

$$\omega_0 N = 2\pi k, \text{ where } k \text{ is an integer.}$$

Exercises

- Determine whether or not the following signals are periodic and, for each sequence that is periodic, determine the fundamental period.

(a) $x[n] = \cos(0.125\pi n)$

(b) $x[n] = \mathrm{Re}\{e^{jn\pi/12}\} + \mathrm{Im}\{e^{jn\pi/18}\}$

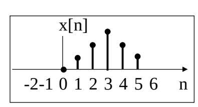
(c) $x[n] = \sin(\pi + 0.2n)$

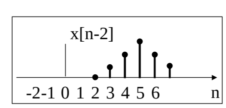
(d) $x[n] = e^{j\frac{\pi}{16}n} \cos(n\pi/17)$

Signal Manipulation: Shifting

- Note: The operations below are order-dependent.

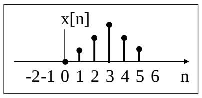
- **Shifting**
  - If $y[n]=x[n-n_0]$, x[n] is shifted to the right by n_0 samples (delay), given n_0 positive

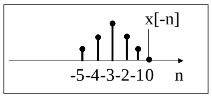




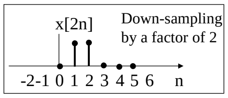
Signal Manipulation: Reversal, Time scaling

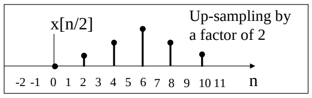
- **Reversal**
  - Given y[n]=x[-n] (simply involves "flipping" the sequence x[n] w.r.t. to index n)





- **Time scaling**
  - Given y[n]=x[Mn] or y[n]=x[n/N] where M and N are positive integers.





Signal Manipulation: Addition, Multiplication, Scaling

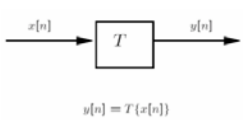
- **Addition**
  - The sum of 2 sequences, y[n]=a[n]+b[n] is formed by the pointwise addition of the two sequences.
- **Multiplication (or modulation)**
  - The product of 2 sequences, y[n]=a[n]b[n] is formed by the pointwise product of the two sequences.
- **Scaling (or scalar multiplication)**
  - Amplitude scaling by a constant c, y[n] = cx[n] is accomplished by multiplying every sample value by c.

n	-1	0	1	2	3	4
a[n]	2	-1	0	4	7	3
b[n]	3	5	-2	7	4	-5
Addition a[b]+b[n]	5	4	-2	11	11	8
Multiplication a[b]b[n]	6	-5	0	28	28	-15
Scaling 3 a[n]	6	-3	0	12	21	9

Discrete Time System

- A discrete-time system is defined mathematically as a transformation or operator that ***maps*** an input sequence with value *x*[*n*] into an output sequence with value *y*[*n*].

$$y(n) = T\{x(n)\}$$



Discrete-Time System Examples

- **Example 2.3** The ideal delay system

$$y[n] = x[n - n_d]$$

where n_d is a fixed positive integer called the delay of the system.

- **Example 2.4** Moving average system

$$y[n] = \frac{1}{M_1 + M_2 + 1} \sum_{k=-M_1}^{M_2} x[n-k]$$

$$= \frac{1}{M_1 + M_2 + 1} \{x[n + M_1] + x[n + M_1 - 1] + \cdots + x[n] + x[n-1] + \cdots + x[n - M_2]\}.$$

Discrete-Time System Properties

- Memoryless Systems
- Linear Systems
- Time invariant Systems
- Causal Systems
- Stable Systems

Memoryless System

- Output y[n] at every value of n depends only on the input x[n] at the same value of n.
- **Example 2.5** - A memoryless system

$$y[n] = (x[n])^2$$

- Any other memoryless system examples?
  - Daily temperature measurement.
- Any other systems with memory?
  - Five-day or recent temperature average.

Linear Systems

- The class of linear systems is defined by the principle of superposition as follows:
  - Additivity property

$$T\{x_1[n] + x_2[n]\} = T\{x_1[n]\} + T\{x_2[n]\}$$

  - Homogeneity or scaling property

$$T\{ax[n]\} = aT\{x[n]\}$$

  - Two properties can be combined into the principle of superposition

$$T\{ax_1[n] + bx_2[n]\} = aT\{x_1[n]\} + bT\{x_2[n]\}$$

Example 2.6 The Accumulator System

- An accumulator system defined by the equation

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

  - Given two arbitrary inputs $x_1[n]$ and $x_2[n]$, and their corresponding outputs:

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

  - When the input is $x_3[n] = ax_1[n] + bx_2[n]$, the output

$$y_3[n] = \sum_{k=-\infty}^{n} x_3[k] = a\sum_{k=-\infty}^{n} x_1[k] + b\sum_{k=-\infty}^{n} x_2[k] = y_1[n] + y_2[n].$$

Superposition principle

Example : Linear Systems

Example: Let $y(n) = x^2(n)$ (i.e., $T\{\cdot\} = (\cdot)^2$). Then,

$$T\{x_1(n) + x_2(n)\} = x_1^2(n) + x_2^2(n) + 2x_1(n)x_2(n) \neq x_1^2(n) + x_2^2(n)$$

Hence, this system is nonlinear!

Time-Invariant Systems

- A time-invariant (shift-invariant) system is a system for which a time shift or delay of the input sequence causes a corresponding shift in the output sequence.
- Specifically, the system satisfies

$$x[n] \Rightarrow y[n], \quad \text{then} \quad x[n-n_0] \Rightarrow y[n-n_0]$$

Example 2.8 The Accumulator as a Time-Invariant system

- Given a accumulator system

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

  - We define a shifted input $x_1[n] = x[n-n_0]$.
  - First, $y[n-n_0] = \sum_{k=-\infty}^{n-n_0} x[k]$
  - Second, $y_1[n] = \sum_{k=-\infty}^{n} x_1[k]$
  - Substituting the change of variables $k_1 = k - n_0$

$$y_1[n] = \sum_{k_1=-\infty}^{n} x[k-n_0] = \sum_{k_1=-\infty}^{n-n_0} x[k_1] = y[n-n_0]$$

Causal Systems

- **Definition** - A system **L** is causal if and only if each output is a function of prior inputs only. In other words, if $y[n] = \mathrm{L}[x[n]]$ then *y*[n] is a function of {…, *x*[*n*-2], *x*[n-1], *x*[n]} only.
- Example 2.9 Forward and Backward Difference Systems
  - Consider the *forward difference* system defined by

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

    - The system is not causal, since the current value of the output depends on a future value of the input.
  - Consider the *backward difference* system defined as

$$y[n] = x[n] - x[n-1]$$

    - The system is causal, since the output depends only on the present and past values of the input.

Stable Systems

- The system **L** is bounded-input, bounded-output stable, or BIBO stable if and only if every bounded input results in a bounded output.
- Formally - The system L is BIBO stable if and only if for some $A \in R$, $A>0$, it is true that

$$|x[n]| < A \text{ for all } n \text{ (} x \text{ is bounded)}$$

then there exists $B \in R$, $B>0$ such that

$$|y[n]| < B \text{ for all } n \text{ (} y \text{ is bounded)}$$

where

$$y[n] = \mathrm{L}[x[n]].$$

Example 2.11 Testing for Stability

- Given a memoryless system A

$$y[n] = (x[n])^2, \text{ for each value of } n.$$

- Given an accumulator system B

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

- Given the forward and backward difference systems

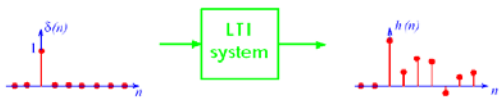
$$y[n] = x[n+1] - x[n]$$

$$y[n] = x[n] - x[n-1]$$

Linear Time-Invariant Systems

Linear Time-Invariant (LTI) system is a system that is both linear and time-invariant (sometimes referred to as a Linear Shift-Invariant (LSI) system)

The sequence $\{h(n)\}$ is commonly referred to as impulse response of the LTI system



Linear Time-Invariant System

Let $h(n)$ be the response of the system to $\delta(n)$.
Due to the time-invariance property, the response to $\delta(n-k)$ is simply $h(n-k)$ $\Longrightarrow$

$$\begin{aligned} y(n) &= T\{x(n)\} \\ &= T\left\{\sum_{k=-\infty}^{\infty} x(k)\delta(n-k)\right\} \\ &= \sum_{k=-\infty}^{\infty} x(k)T\{\delta(n-k)\} \\ &= \sum_{k=-\infty}^{\infty} x(k)h(n-k) = \{x(n)\} * \{h(n)\} \quad \text{convolution sum} \end{aligned}$$

LTI Causality

- **Definition** - A system L is causal if and only if each output is a function of prior inputs only. In other words, if $y[n] = \mathrm{L}[x[n]]$ then *y*[n] is a function of {…, *x*[*n*-2], *x*[n-1], *x*[n]} only.

- **Theorem** (LTI Causality) - An LTI system L with impulse response h[n] is causal if and only if

$$h[n] = 0 \text{ for } n < 0.$$

Proof: An input *x*[*n*] results in an output

$$y[n] = \sum_{m=-\infty}^{\infty} h[n-m]x[m] = \sum_{m=-\infty}^{n} h[n-m]x[m] + \sum_{m=n+1}^{\infty} h[n-m]x[m].$$

The second term will be zero for any input *x*[*n*] if and only if

$$h[n-m] = 0 \text{ for } m = n+1, n+2, ..., \infty$$

or (by change of variable)

$$h[m] = 0 \text{ for } m = -\infty, ..., -3, -2, -1.$$

- *Such a system is often referred to as being non anticipated.*

Example: LTI Causality

- *Causal System*

$$h[n] = u[n] - u[n-1]$$

- *Non-Causal System*

$$h[n] = u[n+1] - u[n]$$

LTI Stability

- **Theorem (LTI BIBO stability)** - An LTI system L is BIBO stable, *if and only if* it has an absolutely summable unit pulse response *h*[*n*]:

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

Proof of "if": Let the input $x(n)$ be bounded so that $|x(n)| < L_x$, $\forall n \in [-\infty, \infty]$. Then

$$|y(n)| = \left| \sum_{k=-\infty}^{\infty} h(k)x(n-k) \right|$$

$$\leq \sum_{k=-\infty}^{\infty} |h(k)||x(n-k)|$$

$$\leq L_x \sum_{k=-\infty}^{\infty} |h(k)| \implies |y(n)| < \infty \text{ if } \sum_{k=-\infty}^{\infty} |h(k)| < \infty$$

Response of LTI Systems

- All LTI systems are described by the convolution sum.
- The impulse response is a complete characterization of the properties of a specific LTI system.
- Basic properties of convolution
  - Commutative

$$y[n] = x[n] * h[n] = h[n] * x[n]$$

  - Distributive

$$x[n] * (h_1[n] + h_2[n]) = x[n] * h_1[n] + x[n] * h_2[n]$$

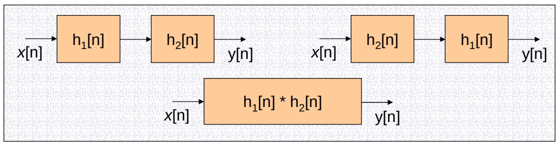
  - Cascade connection

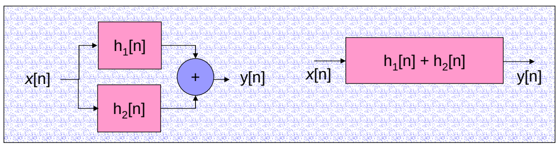
$$x[n] * h_1[n] * h_2[n] = x[n] * (h_1[n] * h_2[n])$$

  - Parallel connection

$$x[n] * h_1[n] + x[n] * h_2[n] = x[n] * (h_1[n] + h_2[n])$$

LTI System Properties





Linear Constant Coefficient Difference Equations (LCCDE)

- Discrete-time LTI system can be characterized by linear constant coefficient difference equations (LCCDE's or DE).
- LCCDE may be regarded as the discrete analogues of the LCC differential equations central to continuous system theory.
- Example - A generic difference equation (DE):

$$\sum_{k=0}^{N} b[k]y[n-k] = \sum_{m=0}^{M} a[m]x[n-m].$$

which is a weighted sum of shifted outputs up to the current one, expressed as a weighted sum of shifted inputs.

Example 2.14 DE Representation of the Accumulator

- An example of the class of LCCDE is the accumulator system defined by

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

$$y[n-1] = \sum_{k=-\infty}^{n-1} x[k] \qquad y[n] = x[n] + \sum_{k=-\infty}^{n-1} x[k]$$

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

$$y[n] - y[n-1] = x[n]$$

Example 2.14 DE Representation of the Accumulator (Diagram)

$$y[n] - y[n-1] = x[n]$$

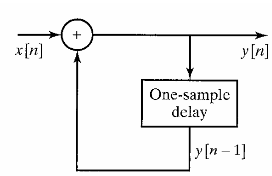


Figure 2.15 Block diagram of a recursive difference equation representing an accumulator.

Example 2.16 Recursive Computation of Difference Equations

- The difference equation satisfied by the input and output of a system is

$$y[n] = ay[n-1] + x[n].$$

Consider the input $x[n] = K\delta[n]$, where K is an arbitary number, and the auxiliary condition $y[-1] = c$.

$y[n] = ay[n-1] + x[n]$ $n \geq 0$

$y[0] = ac + K,$

$y[1] = ay[0] + 0 = a(ac + K) = a^2c + aK,$

$y[2] = ay[1] + 0 = a(a^2c + aK) = a^3c + a^2K,$

$y[n-1] = a^{-1}(y[n] - x[n])$ $n \leq -1$

$y[-2] = a^{-1}(y[-1] - x[-1]) = a^{-1}c,$

$y[-3] = a^{-1}(y[-2] - x[-2]) = a^{-2}c,$

$y[-4] = a^{-1}(y[-3] - x[-3]) = a^{-3}c,$

$y[n] = a^{n+1}c + a^nK$ for $n \geq 0$ $\quad$ $y[n] = a^{n+1}c$ for $n \leq -1.$

$$y[n] = a^{n+1}c + Ka^nu[n], \text{ for all } n.$$

Summary for LCCDE

- The output for a given input *is not uniquely specified*. Auxiliary information or conditions are required.
- If the auxiliary information is in the form of *N* sequential values of the output
  - later values can be obtained by rearranging the difference equation as a *recursive relation running forward in n*;
  - Prior values can be obtained by rearranging the difference equations as a *recursive relation running backward in n*.
- *Linearity*, *time-invariance*, and *causality* of the system will depend on the auxiliary conditions. If an additional condition is that the system is initially *at rest*, then it will be linear, time invariant, and causal.

$$\text{If } x[n] = 0 \text{ for } n < n_0, \text{ then } y[n] = 0 \text{ for } n < n_0.$$

Example 2.16 Recursive Computation of Difference Equations (Initial-rest)

- Given the system

$$y[n] = ay[n-1] + x[n].$$

  - Consider the input $x[n] = K\delta[n];$
  - Since with initial-rest conditions $x[n] = 0, n < 0 \Rightarrow y[-1] = 0;$
  - From the general output, we see the output is

$$y[n] = Ka^nu[n], \text{ for all } n.$$

  - If the input is $x[n] = K\delta[n - n_0]$, again with initial-rest conditions, then the recursive solution is carried out using the initial condition $y[n] = 0, n < n_0.$
  - The initial rest does not always mean that $y[-1] = \cdots = y[-N] = 0,$ but it does mean $y[n] = 0$ for $n < n_0$ if $x[n] = 0$ for $n < n_0.$