Digital Signal Processing

Discrete-Time Signals and Systems (1)

Moslem Amiri, Václav Přenosil

Embedded Systems Laboratory
Faculty of Informatics, Masaryk University
Brno, Czech Republic

amiri@mail.muni.cz
prenosil@fi.muni.cz

March, 2012

Discrete-Time Signals

- A discrete-time signal $x(n)$ ($\equiv x_a(nT)$) is a function of an independent variable that is an integer
  - We assume that $x(n)$ is defined for every n for $-\infty < n < \infty$
  - $x(n)$ is not defined for non-integer values of n

- **Unit sample sequence** or **unit impulse**

$$\delta(n) \equiv \begin{cases} 1, & \text{for } n = 0 \\ 0, & \text{for } n \neq 0 \end{cases}$$

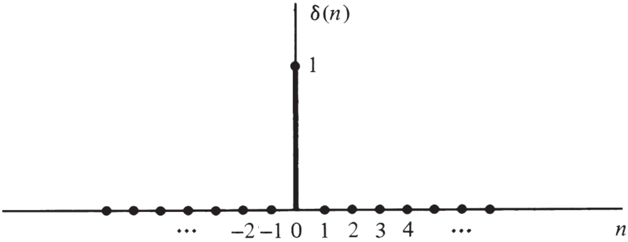


Figure 1: Graphical representation of the unit sample signal.

- **Unit step signal**

$$u(n) \equiv \begin{cases} 1, & \text{for } n \geq 0 \\ 0, & \text{for } n < 0 \end{cases}$$

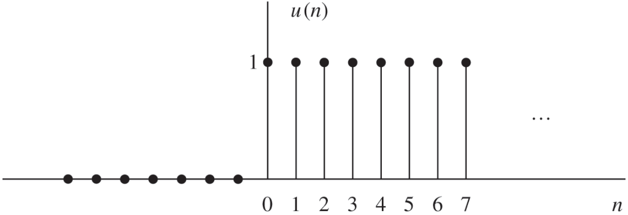


Figure 2: Graphical representation of the unit step signal.

- **Unit ramp signal**

$$u_r(n) \equiv \begin{cases} n, & \text{for } n \geq 0 \\ 0, & \text{for } n < 0 \end{cases}$$

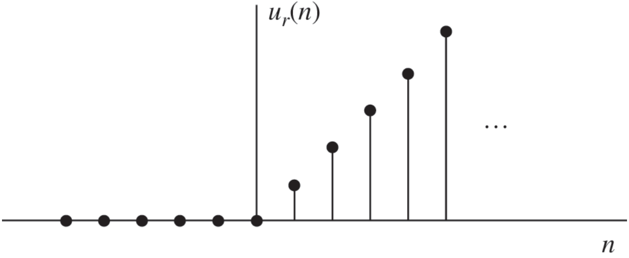


Figure 3: Graphical representation of the unit ramp signal.

- **Exponential signal**

$$x(n) = a^n \quad \text{for all } n$$

  - If a is real, $x(n)$ is a real signal

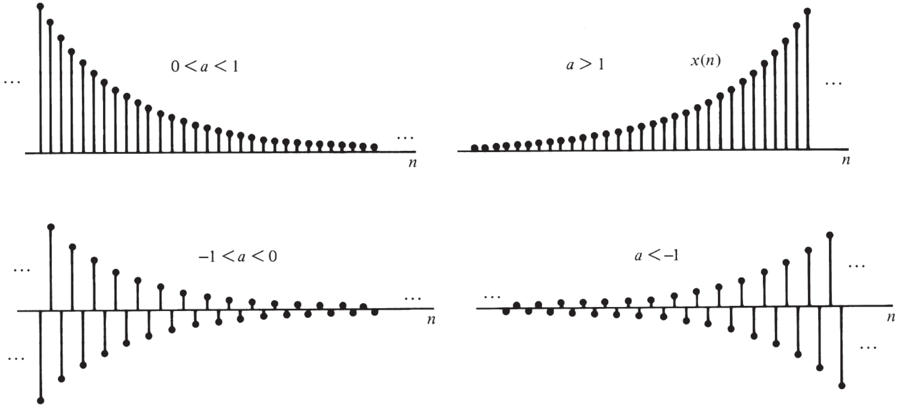


Figure 4: Graphical representation of exponential signals.

Some Elementary Discrete-Time Signals

- Exponential signal

$$x(n) = a^n \quad \text{for all } n$$

  - If a is complex

$$a \equiv re^{j\theta}$$
$$x(n) = r^n e^{j\theta n} = r^n(\cos\theta n + j\sin\theta n)$$

  - $x(n)$ can be represented by separately plotting real part and imaginary part as functions of n

$$x_R(n) \equiv r^n \cos\theta n$$
$$x_I(n) \equiv r^n \sin\theta n$$

  - Alternatively, $x(n)$ can be represented by separately plotting amplitude and phase functions

$$|x(n)| = A(n) \equiv r^n$$
$$\angle x(n) = \phi(n) \equiv \theta n$$

  By convention, $\phi(n)$ is plotted over $-\pi < \theta \leq \pi$ or $0 \leq \theta < 2\pi$

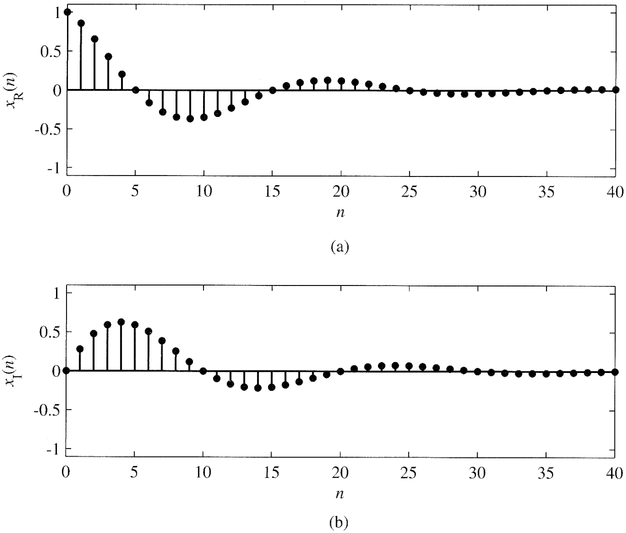
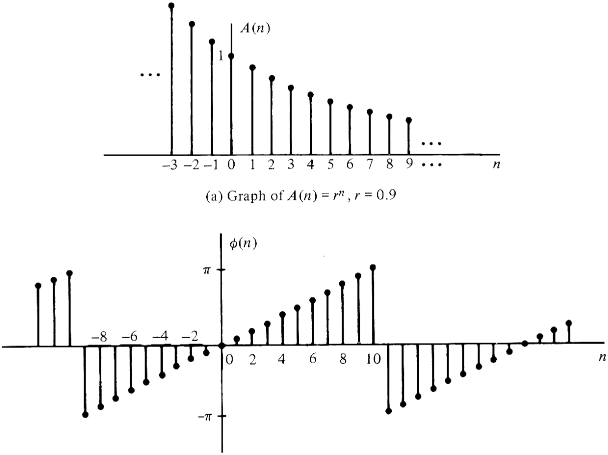


Figure 5: Graph of the real ($x_R(n) \equiv r^n \cos \theta n$) and imaginary ($x_I(n) \equiv r^n \sin \theta n$) components of a complex-valued exponential signal for $r = 0.9$ and $\theta = \pi/10$.

Some Elementary Discrete-Time Signals



(b) Graph of $\phi(n) = \frac{\pi}{10} n$, modulo 2π plotted in the range $(-\pi, \pi)$

Figure 6: Graph of amplitude and phase function of a complex-valued exponential signal: (a) graph of $A(n) = r^n, r = 0.9$; (b) graph of $\phi(n) = (\pi/10)n$, modulo 2π plotted in the range $(-\pi, \pi]$.

- **Energy signals and power signals**
  - Energy E of a signal $x(n)$

$$E \equiv \sum_{n=-\infty}^{\infty} |x(n)|^2$$

  - If E is finite, $x(n)$ is called an **energy signal**
  - Many signals with infinite energy have a finite average power
  - Average power of $x(n)$

$$P = \lim_{N \to \infty} \frac{1}{2N+1} \sum_{n=-N}^{N} |x(n)|^2$$

  - If P is finite (and nonzero), $x(n)$ is called a **power signal**

Classification of Discrete-Time Signals

Example

- Power and energy of unit step sequence

$$P = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=0}^{N} u^2(n) = \lim_{N\to\infty} \frac{N+1}{2N+1} = \lim_{N\to\infty} \frac{1+1/N}{2+1/N} = \frac{1}{2}$$

It is a power signal (its energy is infinite)

Example

- Power and energy of complex exponential sequence $x(n) = Ae^{j\omega_0 n}$

$$P = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} A^2 = \lim_{N\to\infty} \frac{(2N+1)A^2}{2N+1} = A^2$$

It is a power signal

Classification of Discrete-Time Signals

- **Periodic signals and aperiodic signals**
  - $x(n)$ is **periodic** with period N ($N > 0$) if and only if
    $$x(n+N) = x(n) \text{ for all } n$$
  - Smallest value of N is called fundamental period
  - If there is no value of N that satisfies above equation, signal is called **aperiodic**
  - Remember $x(n) = A \sin 2\pi f_0 n$ is periodic if $f_0 = \frac{k}{N}$ = rational number
  - Energy of a periodic signal over a single period is finite if it takes on finite values
    - It is infinite for $-\infty \leq n \leq \infty$
  - Average power of a periodic signal is finite
    - Equal to average power over a single period
  - If $x(n)$ is periodic with fundamental period N and takes on finite values
    $$P = \frac{1}{N} \sum_{n=0}^{N-1} |x(n)|^2$$
    - Periodic signals are power signals

Classification of Discrete-Time Signals

- **Symmetric (even) and antisymmetric (odd) signals**
  - Real-valued signal $x(n)$ is **symmetric (even)** if
  $$x(-n) = x(n)$$
  - $x(n)$ is **antisymmetric (odd)** if
  $$x(-n) = -x(n)$$
    - $x(0) = 0$
  - Any arbitrary signal can be expressed as sum of one even and one odd signal components
  $$x_e(n) = \frac{1}{2}[x(n) + x(-n)]$$
  $$x_o(n) = \frac{1}{2}[x(n) - x(-n)]$$
  $$x(n) = x_e(n) + x_o(n)$$

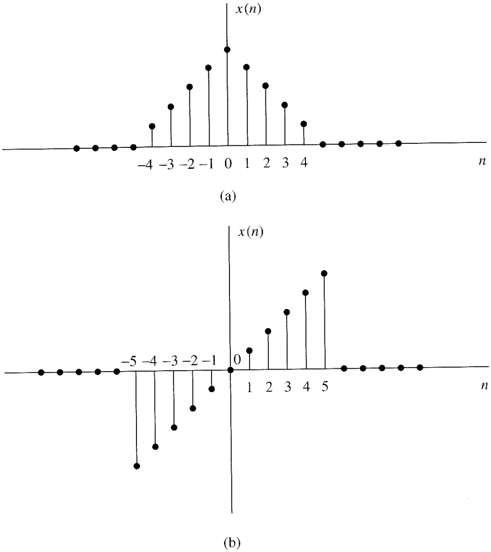


Figure 7: Example of even (a) and odd (b) signals.

Simple Manipulations of Discrete-Time Signals

- Transformation of independent variable (time)
  - $x(n)$ is **shifted** in time by replacing n by $n - k$
    - If $k > 0 \longrightarrow$ delay of signal by k units of time
    - If $k < 0 \longrightarrow$ advance of signal by $|k|$ units in time
  - $x(n)$ is **folded** or **reflected** about time origin $n = 0$ by replacing n by $-n$
  - Operations of folding (FD) and time delaying (TD) (or advancing) a signal are not commutative

$$TD_k[x(n)] = x(n-k), \quad k > 0$$
$$FD[x(n)] = x(-n)$$
$$TD_k\{FD[x(n)]\} = TD_k[x(-n)] = x(-n+k)$$
$$FD\{TD_k[x(n)]\} = FD[x(n-k)] = x(-n-k)$$

  - $x(n)$ is **time scaled** or **down-sampled** by replacing n by μn where μ is an integer
    - If $y(n) = x(2n)$

$$\text{we know } x(n) = x_a(nT)$$
$$y(n) = x(2n) = x_a(2Tn)$$

    Hence this time-scaling operation is equivalent to changing sampling rate from $1/T$ to $1/2T \longrightarrow$ a down-sampling operation

Simple Manipulations of Discrete-Time Signals

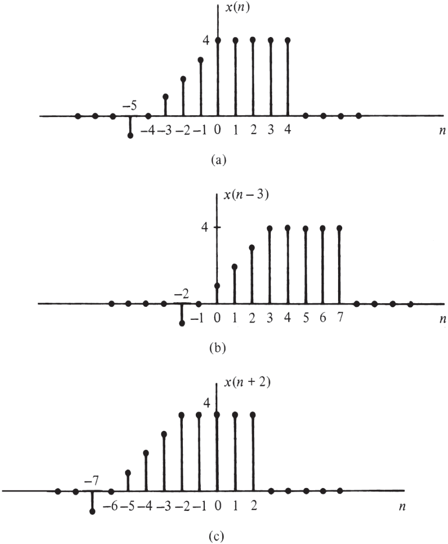


Figure 8: Graphical representation of a signal, and its delayed and advanced versions.

Simple Manipulations of Discrete-Time Signals

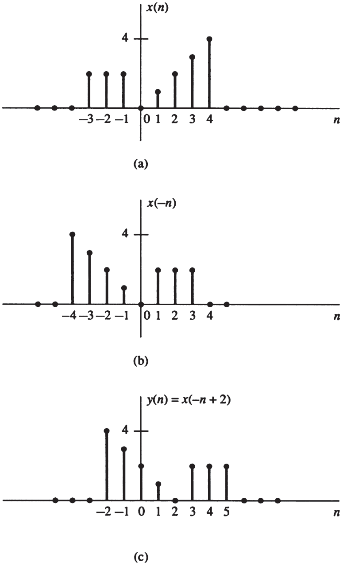


Figure 9: Graphical illustration of the folding and shifting operations.

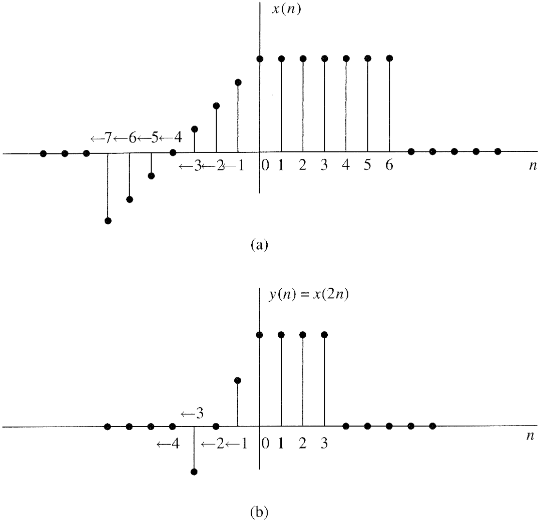


Figure 10: Graphical illustration of down-sampling operation.

Simple Manipulations of Discrete-Time Signals

- Amplitude modifications
  - **Amplitude scaling** by a constant A
    $$y(n) = Ax(n), \quad -\infty < n < \infty$$
  - **Sum** of two signals
    $$y(n) = x_1(n) + x_2(n), \quad -\infty < n < \infty$$
  - **Product** of two signals
    $$y(n) = x_1(n)x_2(n), \quad -\infty < n < \infty$$

- Discrete-time system
  - A device or algorithm that operates on a discrete-time signal called **input** or **excitation**, according to some well-defined rule, to produce another discrete-time signal called **output** or **response** of system
  - Input signal $x(n)$ is **transformed** by system into output signal $y(n)$

$$y(n) \equiv \tau[x(n)]$$

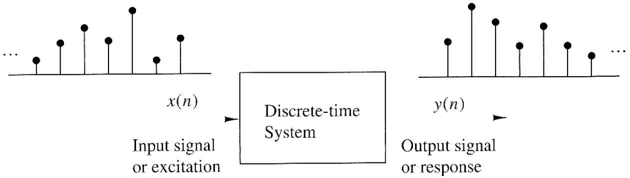


Figure 11: Block diagram representation of a discrete-time system.

Input-Output Description of Systems

- Input-output description of a system
  - Consists of a mathematical expression or a rule defining relation between input and output signals
  - The only way to interact with system is by using its input and output terminals
  - System is assumed to be a black box
  - Exact internal structure of system is either unknown or ignored

Example

- Response of following systems to the input signal

$$x(n) = \begin{cases} |n|, & -3 \leq n \leq 3 \\ 0, & \text{otherwise} \end{cases}$$

$$x(n) = \{\ldots, 0, 3, 2, 1, \underset{\uparrow}{0}, 1, 2, 3, 0, \ldots\}$$

1. $y(n) = x(n)$ (identity system)
   $y(n) = x(n) = \{\ldots, 0, 3, 2, 1, \underset{\uparrow}{0}, 1, 2, 3, 0, \ldots\}$
2. $y(n) = x(n-1)$ (unit delay system)
   $y(n) = \{\ldots, 0, 3, 2, \underset{\uparrow}{1}, 0, 1, 2, 3, 0, \ldots\}$
3. $y(n) = x(n+1)$ (unit advance system)
   $y(n) = \{\ldots, 0, 3, 2, 1, 0, \underset{\uparrow}{1}, 2, 3, 0, \ldots\}$

Example (continued)

4. $y(n) = \frac{1}{3}[x(n+1) + x(n) + x(n-1)]$ (moving average filter)
   $y(n) = \{\ldots, 0, 1, \frac{5}{3}, 2, 1, \underset{\uparrow}{\frac{2}{3}}, 1, 2, \frac{5}{3}, 1, 0, \ldots\}$
   E.g., $y(0) = \frac{1}{3}[x(-1) + x(0) + x(1)] = \frac{1}{3}[1 + 0 + 1] = \frac{2}{3}$
5. $y(n) = \text{median}\{x(n+1), x(n), x(n-1)\}$ (median filter)
   $y(n) = \{\ldots, 0, 2, 2, 1, \underset{\uparrow}{1}, 1, 2, 2, 0, \ldots\}$
6. $y(n) = \sum_{k=-\infty}^{n} x(k) = x(n) + x(n-1) + x(n-2) + \cdots$ (accumulator)
   $y(n) = \{\ldots, 0, 3, 5, 6, \underset{\uparrow}{6}, 7, 9, 12, \ldots\}$

Input-Output Description of Systems

- For some systems, output at $n = n_0$ depends not only on input at $n = n_0$, but on input values before and after $n = n_0$
- E.g., for accumulator

$$y(n) = \sum_{k=-\infty}^{n} x(k) = \sum_{k=-\infty}^{n-1} x(k) + x(n) = y(n-1) + x(n)$$

  - Given input signal $x(n)$ for $n \geq n_0$, output $y(n)$ for $n \geq n_0$

$$y(n_0) = y(n_0 - 1) + x(n_0)$$
$$y(n_0 + 1) = y(n_0) + x(n_0 + 1)$$

    and so on
  - The additional information required to determine $y(n)$ for $n \geq n_0$ is **initial condition** $y(n_0 - 1)$
  - With no excitation prior to n_0, initial condition is $y(n_0 - 1) = 0$
    - System is **initially relaxed**
  - Every system is relaxed at $n = -\infty$

Example

- Following accumulator is excited by sequence $x(n) = nu(n)$

$$y(n) = \sum_{k=-\infty}^{n} x(k)$$

- Output of system

$$y(n) = \sum_{k=-\infty}^{-1} x(k) + \sum_{k=0}^{n} x(k) = y(-1) + \sum_{k=0}^{n} x(k) = y(-1) + \frac{n(n+1)}{2}$$

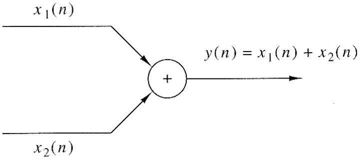
- If system is initially relaxed $\longrightarrow y(-1) = 0$

$$y(n) = \frac{n(n+1)}{2}, \quad n \geq 0$$

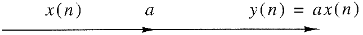
- If initial condition is $y(-1) = 1$

$$y(n) = 1 + \frac{n(n+1)}{2} = \frac{n^2+n+2}{2}, \quad n \geq 0$$

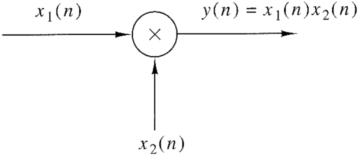
- Symbols used to denote different basic building blocks
  - **An adder**



    - This operation is **memoryless** (not necessary to store sequences)
  - **A constant multiplier** (memoryless operation)

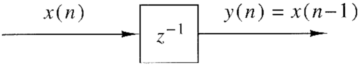


  - **A signal multiplier** (memoryless operation)

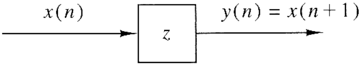


Block Diagram Representation of Discrete-Time Systems

- Symbols used to denote different basic building blocks
  - **A unit delay element** (requires memory)



  - **A unit advance element** (requires memory)



Example

- Using basic building blocks, sketch block diagram of

$$y(n) = \frac{1}{4}y(n-1) + \frac{1}{2}x(n) + \frac{1}{2}x(n-1)$$

  - Shown in Fig. 12 (a)

- A simple rearrangement

$$y(n) = \frac{1}{4}y(n-1) + \frac{1}{2}[x(n) + x(n-1)]$$

  - Shown in Fig. 12 (b)

Example (continued)

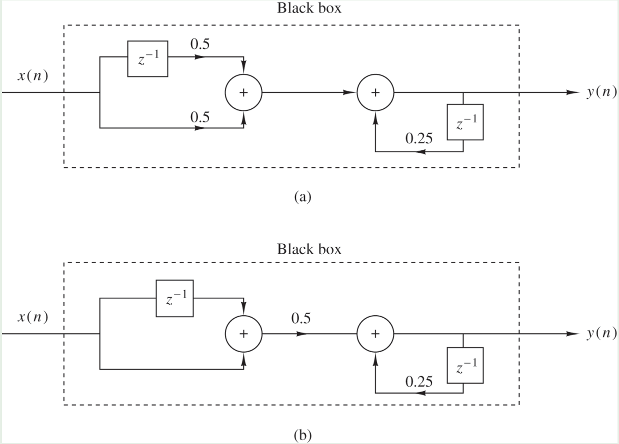


Figure 12: Block diagram realizations of the system $y(n) = 0.25y(n-1) + 0.5x(n) + 0.5x(n-1)$.

Classification of Discrete-Time Systems: Static vs Dynamic

- **Static** or memoryless system
  - Output at any instant n depends at most on input sample at same time, but not on past or future samples of input
- **Dynamic**
  - A system which is not static
    - Has memory
  - If output at time n is completely determined by input samples from $n-N$ to n ($N \geq 0$), system is said to have memory of duration N
  - $N = 0 \longrightarrow$ system is static
  - $0 < N < \infty \longrightarrow$ system has finite memory
  - $N = \infty \longrightarrow$ system has infinite memory

Classification of Discrete-Time Systems: Static vs Dynamic

Example

- Following systems are static

$$y(n) = ax(n)$$
$$y(n) = nx(n) + bx^3(n)$$

- Following systems are dynamic

$$y(n) = x(n) + 3x(n-1)$$

This system has finite memory

$$y(n) = \sum_{k=0}^{n} x(n-k)$$

This system has finite memory

$$y(n) = \sum_{k=0}^{\infty} x(n-k)$$

This system has infinite memory

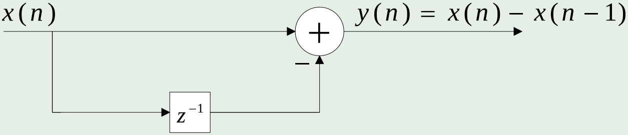
Classification of D-T Systems: Time-Invariant, Time-Variant

- A relaxed system τ is **time invariant** or **shift invariant** if and only if
$$x(n) \xrightarrow{\tau} y(n)$$
implies that
$$x(n-k) \xrightarrow{\tau} y(n-k)$$
for every input signal $x(n)$ and every time shift k
- To determine if any given system is time invariant
  1. Excite system with an arbitrary sequence $x(n)$, which produces $y(n)$
  2. Delay input sequence by some amount k and recompute output
  $$y(n,k) = \tau[x(n-k)]$$
  3. If $y(n,k) = y(n-k)$, for all possible k, system is time invariant. If not, even for one k, system is time variant

Classification of D-T Systems: Time-Invariant, Time-Variant

Example

- Is this system time invariant or time variant?



- Input-output equation of system

$$y(n) = \tau[x(n)] = x(n) - x(n-1)$$

Delaying input by k units, it is clear from block diagram that

$$y(n,k) = x(n-k) - x(n-k-1)$$

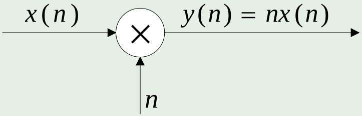
On the other hand, delaying $y(n)$ by k units

$$y(n-k) = x(n-k) - x(n-k-1)$$

Since $y(n,k) = y(n-k)$, system is time invariant

Example

- Is this system time invariant or time variant?



- Input-output equation of system

$$y(n) = \tau[x(n)] = nx(n)$$

Response of this system to $x(n-k)$ is

$$y(n,k) = nx(n-k)$$

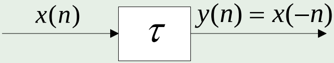
If we delay $y(n)$ by k units

$$y(n-k) = (n-k)x(n-k) = nx(n-k) - kx(n-k)$$

Since $y(n,k) \neq y(n-k)$, system is time variant

Example

- Is this system time invariant or time variant?



- Input-output equation of system

$$y(n) = \tau[x(n)] = x(-n)$$

Response of this system to $x(n-k)$ is

$$y(n,k) = \tau[x(n-k)] = x(-n-k)$$

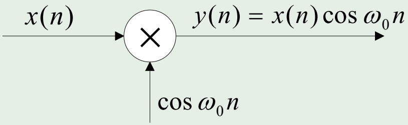
If we delay $y(n)$ by k units

$$y(n-k) = x(-n+k)$$

Since $y(n,k) \neq y(n-k)$, system is time variant

Example

- Is this system time invariant or time variant?



- Input-output equation of system

$$y(n) = x(n)\cos\omega_0 n$$

Response of this system to $x(n-k)$ is

$$y(n,k) = x(n-k)\cos\omega_0 n$$

If we delay $y(n)$ by k units

$$y(n-k) = x(n-k)\cos\omega_0(n-k)$$

Since $y(n,k) \neq y(n-k)$, system is time variant

Classification of D-Time Systems: Linear vs Nonlinear

- A system is linear if and only if

$$\tau[a_1 x_1(n) + a_2 x_2(n)] = a_1 \tau[x_1(n)] + a_2 \tau[x_2(n)]$$

  for any arbitrary input sequences $x_1(n)$ and $x_2(n)$, and any arbitrary constants a_1 and a_2
- A linear system satisfies **superposition principle**
  - This principle requires that response of system to a weighted sum of signals be equal to the corresponding weighted sum of responses of system to each of individual input signals

Classification of D-Time Systems: Linear vs Nonlinear

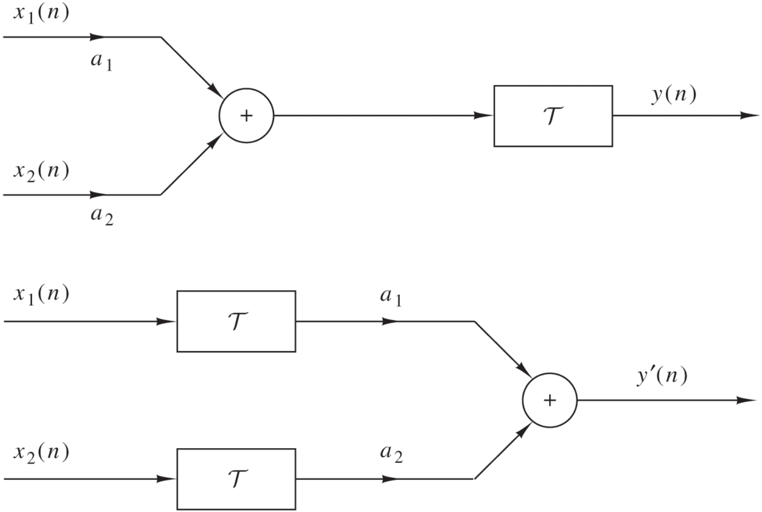


Figure 13: Graphical representation of the superposition principle. τ is linear if and only if $y(n) = y'(n)$.

- Linearity condition (superposition principle)

$$\tau[a_1x_1(n) + a_2x_2(n)] = a_1\tau[x_1(n)] + a_2\tau[x_2(n)]$$

  - Suppose $a_2 = 0$

    $$\tau[a_1x_1(n)] = a_1\tau[x_1(n)] = a_1y_1(n)$$

    This is **multiplicative** or **scaling property** of a linear system
    - If $a_1 = 0$, then $y(n) = 0 \longrightarrow$ a relaxed, linear system with zero input produces a zero output
  - Suppose $a_1 = a_2 = 1$

    $$\tau[x_1(n) + x_2(n)] = \tau[x_1(n)] + \tau[x_2(n)] = y_1(n) + y_2(n)$$

    This is **additivity property** of a linear system
  - Extension of linearity condition

    $$x(n) = \sum_{k=1}^{M-1} a_k x_k(n) \xrightarrow{\tau} y(n) = \sum_{k=1}^{M-1} a_k y_k(n)$$

    where $y_k(n) = \tau[x_k(n)], \quad k = 1, 2, \ldots, M-1$
  - If a relaxed system does not satisfy superposition principle, it is **nonlinear**

Classification of D-Time Systems: Linear vs Nonlinear

Example

- Determine if $y(n) = nx(n)$ is linear or nonlinear
- For two inputs $x_1(n)$ and $x_2(n)$, outputs are

$$y_1(n) = nx_1(n)$$
$$y_2(n) = nx_2(n)$$

A linear combination of two input sequences results in output

$$y_3(n) = \tau[a_1x_1(n) + a_2x_2(n)] = n[a_1x_1(n) + a_2x_2(n)] = a_1nx_1(n) + a_2nx_2(n)$$

A linear combination of two output sequences results in output

$$a_1y_1(n) + a_2y_2(n) = a_1nx_1(n) + a_2nx_2(n)$$

Since right-hand sides of two above equations are identical, system is linear

Classification of D-Time Systems: Linear vs Nonlinear

Example

- Determine if $y(n) = x(n^2)$ is linear or nonlinear
- Response of system to two separate inputs $x_1(n)$ and $x_2(n)$

$$y_1(n) = x_1(n^2)$$
$$y_2(n) = x_2(n^2)$$

Output of system to a linear combination of $x_1(n)$ and $x_2(n)$

$$y_3(n) = \tau[a_1x_1(n) + a_2x_2(n)] = a_1x_1(n^2) + a_2x_2(n^2)$$

A linear combination of two output sequences

$$a_1y_1(n) + a_2y_2(n) = a_1x_1(n^2) + a_2x_2(n^2)$$

Since right-hand sides of two above equations are identical, system is linear

Classification of D-Time Systems: Linear vs Nonlinear

Example

- Determine if $y(n) = x^2(n)$ is linear or nonlinear
- Response of system to two separate inputs

$$y_1(n) = x_1^2(n)$$
$$y_2(n) = x_2^2(n)$$

Response of system to a linear combination of these two inputs

$$y_3(n) = \tau[a_1 x_1(n) + a_2 x_2(n)] = [a_1 x_1(n) + a_2 x_2(n)]^2 =$$
$$a_1^2 x_1^2(n) + 2 a_1 a_2 x_1(n) x_2(n) + a_2^2 x_2^2(n)$$

If system is linear, it will produce a linear combination of two outputs

$$a_1 y_1(n) + a_2 y_2(n) = a_1 x_1^2(n) + a_2 x_2^2(n)$$

Since right-hand sides of two above equations are not identical, system is nonlinear

Classification of D-Time Systems: Linear vs Nonlinear

Example

- Determine if $y(n) = Ax(n) + B$ is linear or nonlinear
- For two inputs $x_1(n)$ and $x_2(n)$, outputs are

$$y_1(n) = Ax_1(n) + B$$
$$y_2(n) = Ax_2(n) + B$$

A linear combination of $x_1(n)$ and $x_2(n)$ results in output

$$y_3(n) = \tau[a_1x_1(n) + a_2x_2(n)] = A[a_1x_1(n) + a_2x_2(n)] + B = a_1Ax_1(n) + a_2Ax_2(n) + B$$

If system were linear, its output would be

$$a_1y_1(n) + a_2y_2(n) = a_1Ax_1(n) + a_1B + a_2Ax_2(n) + a_2B$$

The two results are different and system fails to satisfy linearity test. Reason is not that system is nonlinear but with $B \neq 0$ system is not relaxed.

Classification of D-Time Systems: Linear vs Nonlinear

Example

- Determine if $y(n) = e^{x(n)}$ is linear or nonlinear
- This system is relaxed
  If $x(n) = 0 \longrightarrow y(n) = 1$
  Hence system is nonlinear

Classification of D-Time Systems: Causal vs Noncausal

- A system is **causal** if its output at any time depends only on present and past inputs but not on future inputs

$$y(n) = F[x(n), x(n-1), x(n-2), \ldots]$$

  - If a system does not satisfy this definition, it is **noncausal**

Example

- These systems are causal

$$y(n) = x(n) - x(n-1)$$
$$y(n) = \sum_{k=-\infty}^{n} x(k)$$
$$y(n) = ax(n)$$

- These systems are noncausal

$$y(n) = x(n) + 3x(n+4)$$
$$y(n) = x(n^2)$$
$$y(n) = x(2n)$$
$$y(n) = x(-n) \xrightarrow{e.g.,\ n=-1} y(-1) = x(1)$$

Classification of D-Time Systems: Stable vs Unstable

- A relaxed system is bounded input-bounded output (BIBO) **stable** if and only if every bounded input produces a bounded output
  - $x(n)$ and $y(n)$ are bounded if there exist some finite numbers, M_x and M_y, such that for all n
  $$|x(n)| \leq M_x < \infty, \quad |y(n)| \leq M_y < \infty$$
  - If for bounded $x(n)$, output is unbounded (infinite), system is **unstable**

Example

- Consider nonlinear system
$$y(n) = y^2(n-1) + x(n)$$
We select bounded input
$$x(n) = C\delta(n)$$
where C is a constant. Assume $y(-1) = 0$. Output sequence is
$$y(0) = C, \quad y(1) = C^2, \quad y(2) = C^4, \quad \ldots, \quad y(n) = C^{2^n}$$
Output is unbounded when $1 < |C| < \infty$
System is BIBO unstable

- Systems can be interconnected in two ways to form larger systems
  - Cascade (series)
  - Parallel

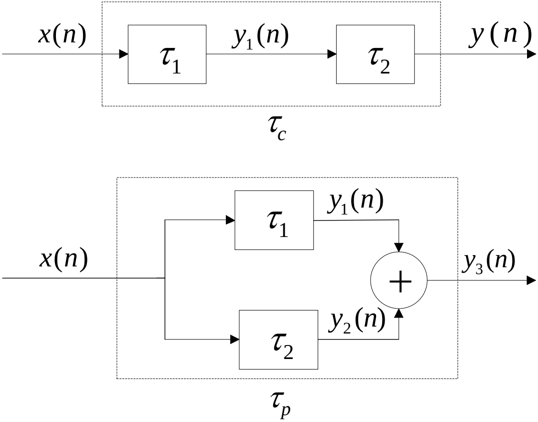


Figure 14: Cascade and parallel interconnections of systems.

- In cascade interconnection
  - Output of first system is
    $$y_1(n) = \mathcal{T}_1[x(n)]$$
    Output of second system
    $$y(n) = \mathcal{T}_2[y_1(n)] = \mathcal{T}_2[\mathcal{T}_1[x(n)]]$$
    Combining systems $\mathcal{T}_1$ and $\mathcal{T}_2$ into a single system $\mathcal{T}_c$
    $$\mathcal{T}_c \equiv \mathcal{T}_2\mathcal{T}_1 \longrightarrow y(n) = \mathcal{T}_c[x(n)]$$
  - For arbitrary systems $\mathcal{T}_1$ and $\mathcal{T}_2$
    $$\mathcal{T}_2\mathcal{T}_1 \neq \mathcal{T}_1\mathcal{T}_2$$
  - If systems $\mathcal{T}_1$ and $\mathcal{T}_2$ are linear and time invariant, then
    1. $\mathcal{T}_c$ is time invariant
       $$x(n-k) \xrightarrow{\mathcal{T}_1} y_1(n-k)$$
       and
       $$y_1(n-k) \xrightarrow{\mathcal{T}_2} y(n-k)$$
       thus
       $$x(n-k) \xrightarrow{\mathcal{T}_c = \mathcal{T}_2\mathcal{T}_1} y(n-k)$$
    2. $\mathcal{T}_2\mathcal{T}_1 = \mathcal{T}_1\mathcal{T}_2$

Interconnection of Discrete-Time Systems

- Output of parallel interconnection is

  $y_3(n) = y_1(n) + y_2(n) = \tau_1[x(n)] + \tau_2[x(n)] = (\tau_1 + \tau_2)[x(n)] = \tau_p[x(n)]$

  where $\tau_p = \tau_1 + \tau_2$
- Parallel and cascade interconnections can be used to construct larger, more complex systems
  - Conversely, a larger system can be broken down into smaller subsystems for purposes of analysis and implementation

References

John G. Proakis, Dimitris G. Manolakis, *Digital Signal Processing: Principles, Algorithms, and Applications*, Prentice Hall, 2006.