Sinais e Sistemas de Tempo Discreto

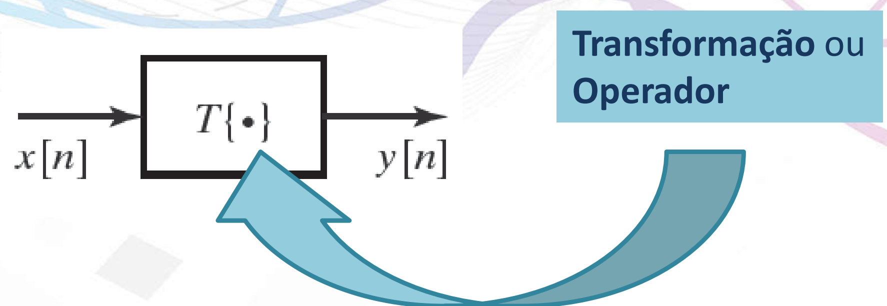
SISTEMAS DE TEMPO DISCRETO

Sistemas de tempo discreto

- Um sistema de tempo discreto é definido matematicamente como

$$y[n] = T\{x[n]\}$$

- Representação de um sistema de tempo discreto



Oppenheim • Schafer

Processamento em tempo discreto de sinais

3ª edição

Sistemas de tempo discreto

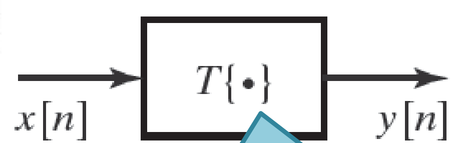
- Um sistema de tempo discreto é definido matematicamente como

$$y[n] = T\{x[n]\}$$

O operador T mapeia a sequencia de entrada x[n] em uma sequencia de saída y[n]

- Representa ... screto

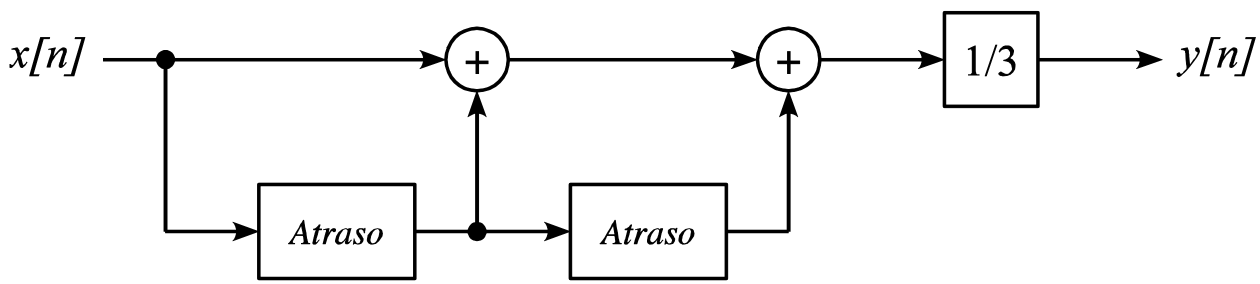
Transformação ou **Operador**



Exemplo

- **Sistema de médias móveis** (*moving average system*)
  - Faz a média aritmética entre uma entrada e suas duas antecessoras

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



Sinais e Sistemas de Tempo Discreto

SISTEMAS DE TEMPO DISCRETO

Classes importantes de sistemas

Sistemas de Tempo Discreto

Classes de Sistemas

- Quando aplicamos qualquer **restrição sobre a transformação T** surgem classes particulares de sistemas

- Serão abordados
  - Sistemas sem memória
  - Sistemas lineares
  - Sistemas invariantes no tempo
  - Causalidade
  - Estabilidade

Sistemas de Tempo Discreto
Classes de Sistemas

Memória

- Um sistema possui memória se as amostras da sequencia de saída dependem de amostras passadas tanto do sinal de entrada como do sinal de saída

Sist. com memória	Sist. sem memória
$y[n] = \frac{1}{3}(x[n] + x[n-1] + x[n-2])$	$y[n] = \left(2x[n] - x^2[n]\right)^2$
$y[n] = \sum_{k=-\infty}^{n} x[k] = \sum_{k=-\infty}^{n-1} x[k] + x[n] = y[n-1] + x[n]$	$y[n] = x[n]$

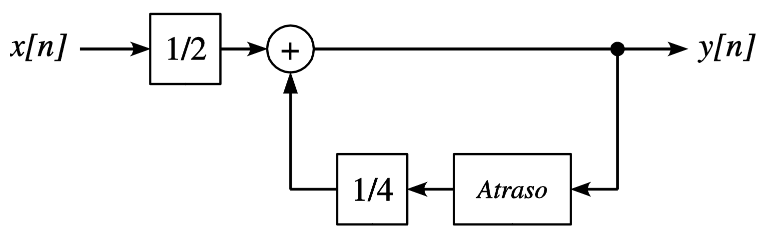
Sistemas de Tempo Discreto
Classes de Sistemas

Memória

Realimentação (Sistema com memória)

- Um sistema é realimentado se a amostra atual do sinal depende de amostras passadas do próprio sinal de saída
- **Exemplos**

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



Sistemas de Tempo Discreto

Classes de Sistemas

Sistemas Lineares

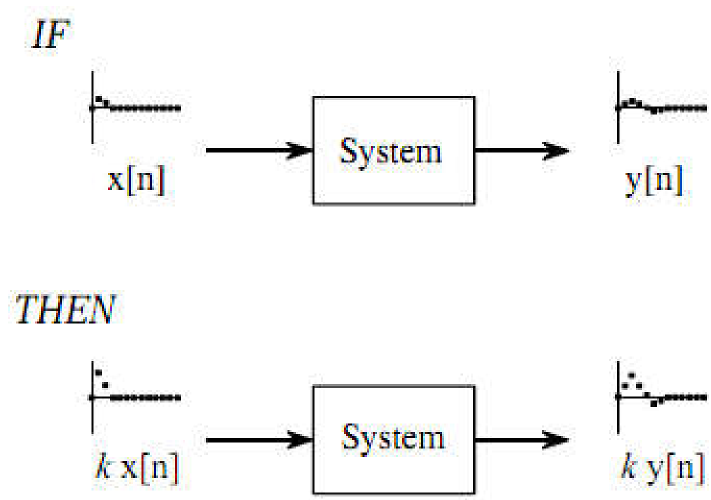
- Sistema que obedece a duas propriedades matemáticas
  - Aditividade
  - Homogeneidade

Sistemas de Tempo Discreto

Classes de Sistemas

Sistemas Lineares

Homogeneidade



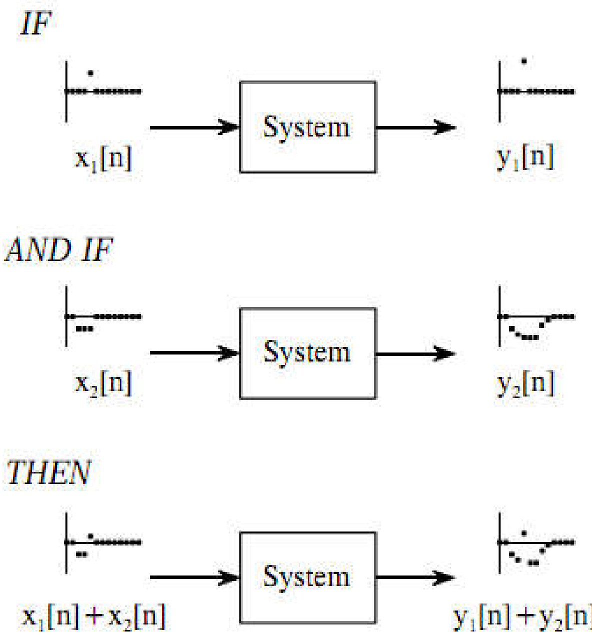
Um sistema é dito homogêneo se uma mudança de amplitude no sinal de entrada gera uma mudança de amplitude idêntica no sinal de saída

Sistemas de Tempo Discreto

Classes de Sistemas

Sistemas Lineares

Aditividade



Um sistema é dito aditivo se 2 sinais podem ser somados sem interação entre eles, em outras palavras, se $x_1[n]$ resulta em $y_1[n]$ e $x_2[n]$ resulta em $y_2[n]$ então $x_1[n]+x_2[n]$ resulta em $y_1[n]+y_2[n]$

Exemplo

- Verifique se sistema abaixo é linear

$$y[n] = \log(x[n])$$

Exemplo

- Verifique se sistema abaixo é linear

$$y[n] = \log(x[n])$$

Como o sistema não é aditivo e nem homogêneo ele não é linear

Verificar se $y[n] = \log(x[n])$ é linear.

Para verificar se um sistema é linear é preciso verificar se ele é aditivo e homogêneo.

homogêneo

$$y[n] = T\{c\,x[n]\} = c\,T\{x[n]\}$$

aplicar escala ao sinal de entrada

Vamos considerar $x_1[n] = c\,x[n]$

$$y_1[n] = \log(c\,x[n]) = \log(c) + \log x[n] \neq c\,\underbrace{\log(x[n])}_{T\{x[n]\}}$$

$\log_a b.c = \log_a b + \log_a c$

∴ não é homogêneo

aditivo

$$\underbrace{T\{x_1[n] + x_2[n]\}}_{\mathrm{I}} = \overbrace{\underbrace{T\{x_1[n]\}}_{y[n]} + \underbrace{T\{x_2[n]\}}_{y_2[n]}}^{\mathrm{II}}$$

I $T\{x_1[n] + x_2[n]\} = \log(x_1[n] + x_2[n])$

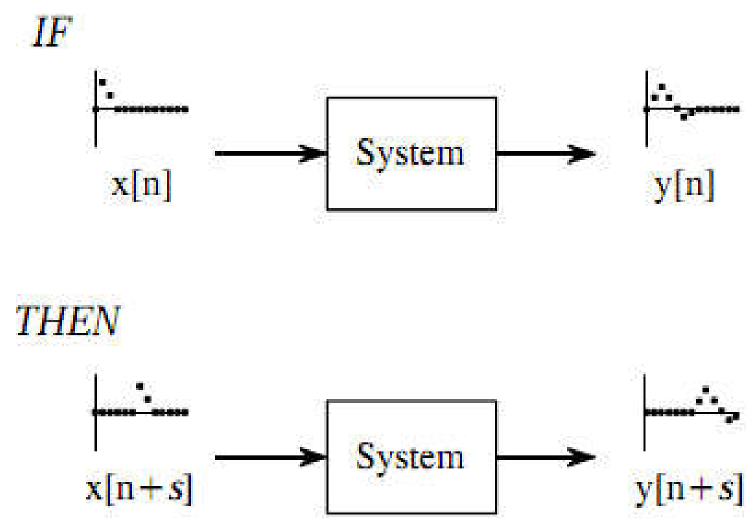
II $T\{x_1[n]\} + T\{x_2[n]\} = \log(x_1[n]) + \log(x_2[n])$

$$= \log(x_1[n] . x_2[n])$$

$$\neq$$

∴ não é aditivo

Invariância a deslocamento (*Shift invariance*)



Não é uma propriedade de linearidade mas é uma propriedade de PDS, se você vir o termo SL usado em PDS você deve assumir invariância no tempo também. **Um sistema é dito invariante no tempo se um deslocamento na entrada causa um deslocamento IDÊNTICO na saída**, matematicamente: se x[n] resulta em y[n] e então x[n+s] resulta em y[n+s]

Invariância a deslocamento (*Shift invariance*)

- Um deslocamento em x[n] causa um deslocamento equivalente em y[n]

$$x_1[n] = x[n - n_0] \longrightarrow y_1[n] = y[n - n_0].$$

Exemplo

- Verifique se o acumulador é invariante no tempo

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

Calculo y em n-n0
Calculo y para x1[n]=x[n-n0]
Comparo

Exemplo

Verifique se o acumulador é invariante ao deslocamento.

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

$y[n]$ em $n-n_0$

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

$n-n_0$

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

$$y[n] = \sum_{k=-\infty}^{n} x[k] \leftarrow x_1[n]$$

$$y_1[n] = \sum_{k=-\infty}^{n} x_1[n] = \sum_{k=-\infty}^{n} x[\boxed{k-n_0}]$$

↳ $x[n-n_0]$

↳ k_1 mudança de índice no somatório

Com a mudança de índices, é preciso ajustar os limites do somatório

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

como k_1 é apenas índice $\boxed{y_1[n] = \sum_{k=-\infty}^{n-n_0} x[k]}$

Logo o acumulador é invariante ao deslocamento

TAREFA

- Verifique se o compressor (ou reamostrador) dado por $y[n]=T\{x[n]\}=x[Mn]$ é invariante no tempo

Causalidade

- Um sistema é causal se a saída, em qualquer tempo, depender dos valores da entrada somente nos instantes presente e passados. Também chamado de sistema NÃO ANTECIPATIVO

- **Exemplos**

Não causal	Causal
$y[n] = \boxed{x[n+1]} + x[n]$ $y[n] = \frac{1}{3}(x[n-1] + x[n] + \boxed{x[n+1]})$ Amostras futuras	$y[n] = \frac{1}{3}(x[n] + x[n-1] + x[n-2])$ Apenas amostras passadas e presentes

Sistemas de Tempo Discreto

Classes de Sistemas

Causalidade

- Um sistema é causal se a saída, em qualquer tempo, depender dos valores da entrada somente nos instantes presente e passados.

  Tam

- **Exer**

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

$y[0] = x^2[0]$ presente

$y[1] = x^2[1]$ presente

$y[2] = x^2[2]$ presente

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

$y[0] = x[0]$ presente

$y[1] = x[1]$ presente

$y[2] = x[4]$ futura

Não cau	
$y[n$ ...	... $-2])$
	3
$y[n] = \frac{1}{3}(x[n-1] + x[n] + \boxed{x[n+1]})$	
Amostras futuras	Apenas amostras passadas e presentes

Estabilidade

A estabilidade requer que, <u>para toda entrada limitada</u>, exista um valor fixo positivo e finito B_y tal que

$$|y[n]| \le B_y < \infty, \quad \text{para todo } n.$$

BIBO = bounded-input, bounded output, ou seja, entrada limitada, saída limitada

Exemplo

- Verificar se os seguintes sistemas são estáveis

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

$b)\ y[n] = \sum_{k=-\infty}^{n} u[k]$ (acumulador)

Exemplo

Verifique se os seguintes sistemas são estáveis.

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

Considerando que $x[n]$ seja uma entrada limitada

$$|x[n]| \leq B_x < \infty$$

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

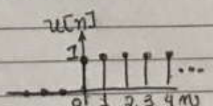
$$B_y = B_x^2$$

$\therefore$

$$|y[n]| \leq \underbrace{B_x^2}_{B_y} < \infty$$

$y[n]$ é limitado

b) $y[n] = \sum_{k=-\infty}^{n} u[k]$ (acumulador)



$x[n]$ é limitada, $B_x = 1$

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

$$y[n] = \sum_{k=0}^{n} u[k] = (1)(n+1) = n+1$$

depende de n

não é limitada

Sinais e Sistemas de Tempo Discreto

SISTEMAS LINEARES E INVARIANTES NO TEMPO - LTI

Sistemas LTI

Lineares

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

Invariantes no tempo

$$x_1[n] = x[n - n_0] \longrightarrow y_1[n] = y[n - n_0].$$

$$p[n] = a_{-3}\delta[n+3] + a_1\delta[n-1] + a_2\delta[n-2] + a_7\delta[n-7]$$

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

Seq é um somatório de impulsos deslocados

Sistemas LTI

Lineares

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

Invariantes no tempo

$$x_1[n] = x[n - n_0] \longrightarrow y_1[n] = y[n - n_0].$$

$$\Downarrow$$

$$p[n] = a_{-3}\delta[n+3] + a_1\delta[n-1] + a_2\delta[n-2] + a_7\delta[n-7]$$

APLICAR UMA TRANSFORMAÇÃO T EM p[n]

Sistemas LTI

Lineares

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

Invariantes no tempo

$$x_1[n] = x[n - n_0] \longrightarrow y_1[n] = y[n - n_0].$$

$$p[n] = a_{-3}\delta[n+3] + a_1\delta[n-1] + a_2\delta[n-2] + a_7\delta[n-7]$$

$$T\{p[n]\} = T\{a_{-3}\delta[n+3] + a_1\delta[n-1] + a_2\delta[n-2] + a_7\delta[n-7]\} =$$

$$= a_{-3}\boxed{\mathrm{T}\{\delta(n+3)\}} + a_1\boxed{\mathrm{T}\{\delta(n-1)\}} + a_2\boxed{\mathrm{T}\{\delta(n-2)\}} + a_7\boxed{\mathrm{T}\{\delta(n-7)\}}$$

Sistemas LTI

Resposta Impulsiva

• A resposta de um sistema a qualquer entrada **pode ser expressa em função da resposta do sistema à seqüência δ[n]**

$$h[n] = H\{\delta[n]\}$$

Sistemas LTI

$$y[n] = T\left\{ \underbrace{\sum_{k=-\infty}^{\infty} x[k]\delta[n-k]}_{\mathbf{x[n]}} \right\}.$$

Resposta impulsiva

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]\boxed{T\{\delta[n-k]\}} = \sum_{k=-\infty}^{\infty} x[k]h_k[n].$$

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

Uma consequência dessa equação é que **um sistema LIT é completamente caracterizado por sua resposta ao impulso *h*[*n*]** no sentido de que, dadas as sequências *x*[*n*] e *h*[*n*] para todo *n*, é possível usar a equação acima para calcular cada amostra da sequência de saída *y*[*n*].

Sistemas LTI

- A **invariância no tempo** implica que se h[n] é a resposta a δ[n], então a resposta a δ[n-k] é h[n-k]
- Se $h[n] = H\{\delta[n]\} \Rightarrow h[n-k] = H\{\delta[n-k]\}$

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

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

$$\boxed{y[n] = x[n] * h[n].}$$

Soma de convolução

Sistemas LTI

- Soma de convolução: **Resultado direto da linearidade e invariancia no tempo!**

- A convolução pega duas seqüências x[n] e h[n] e produz uma terceira seqüência y[n]

- Nessa situação todos os valores de y[n] até n são calculados

Sistemas LTI

Lineares

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

Invariantes no tempo

$$x_1[n] = x[n - n_0] \longrightarrow y_1[n] = y[n - n_0].$$

$$\Downarrow$$

$$p[n] = a_{-3}\delta[n+3] + a_1\delta[n-1] + a_2\delta[n-2] + a_7\delta[n-7]$$

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

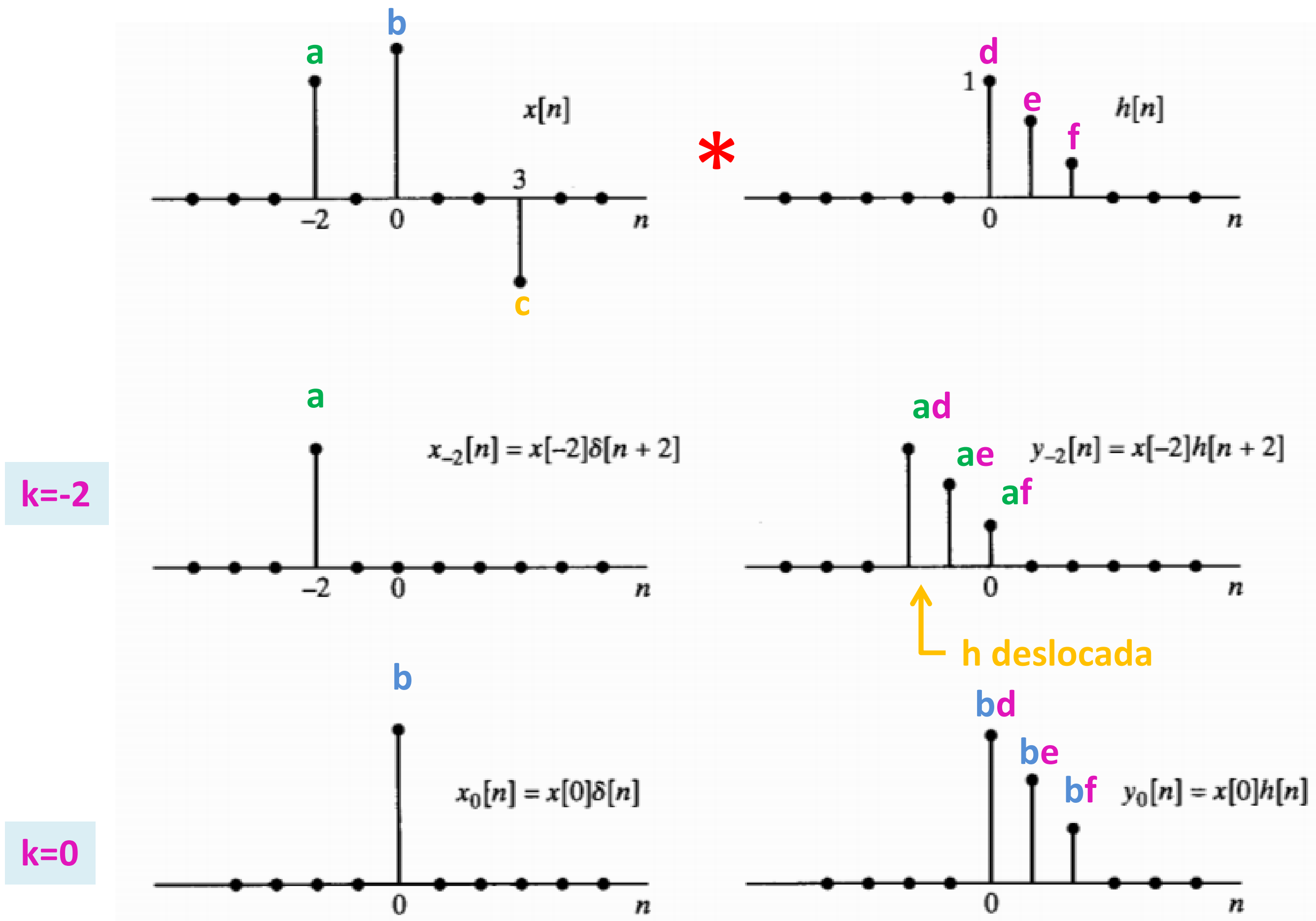
Soma de convolução
é consequência
direta de ser LTI

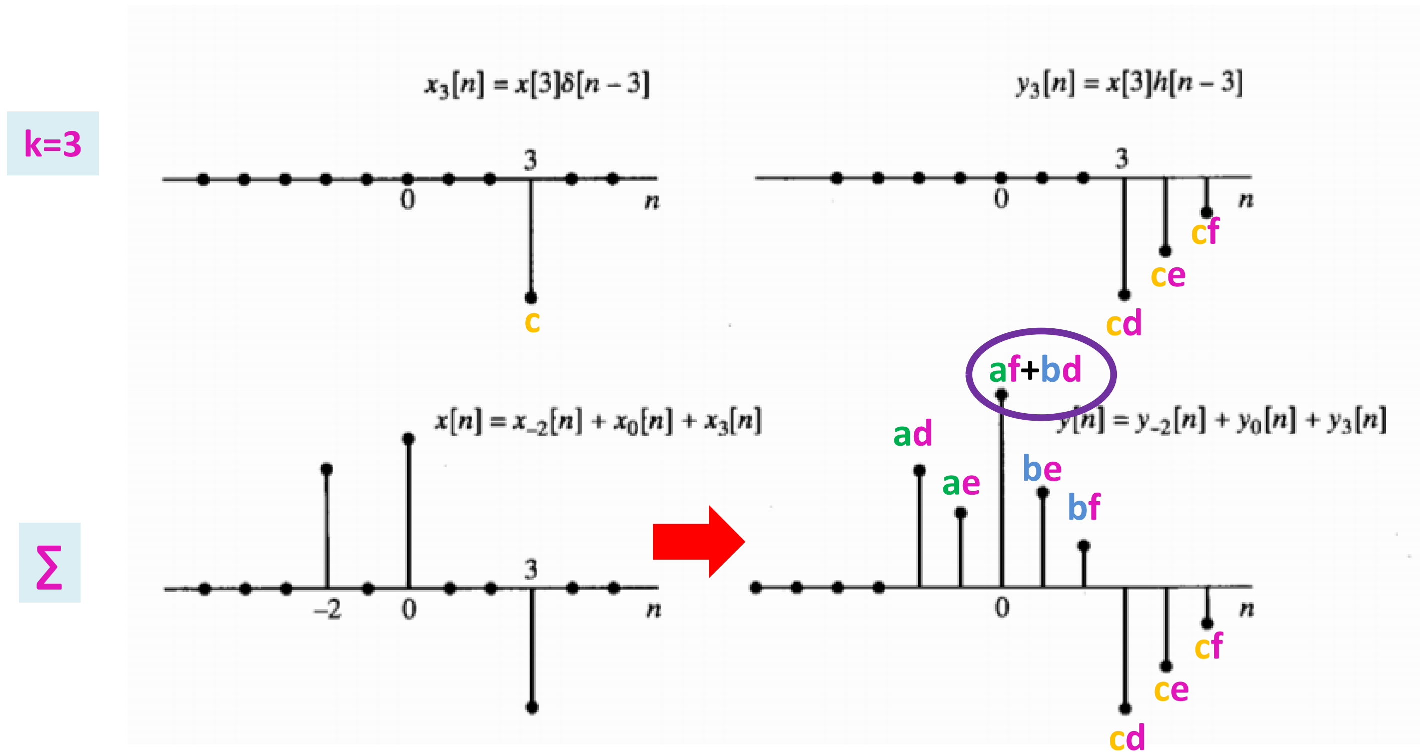
Sistemas LTI - Convolução

Como interpretar a convolução?

$$\boxed{y[n] = x[n] * h[n].}$$

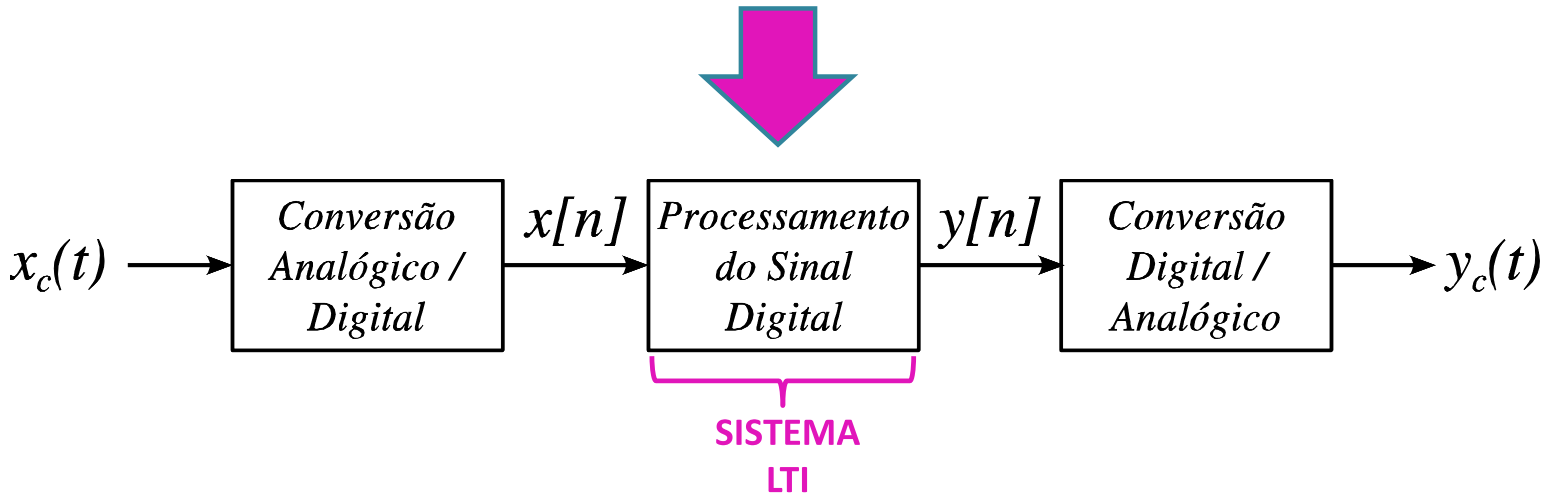
y[n], n-ésimo valor de y é obtido multiplicando-se a seqüência de entrada (x[k]δ[n-k]) pela seqüência h[n-k] e somando esses valores



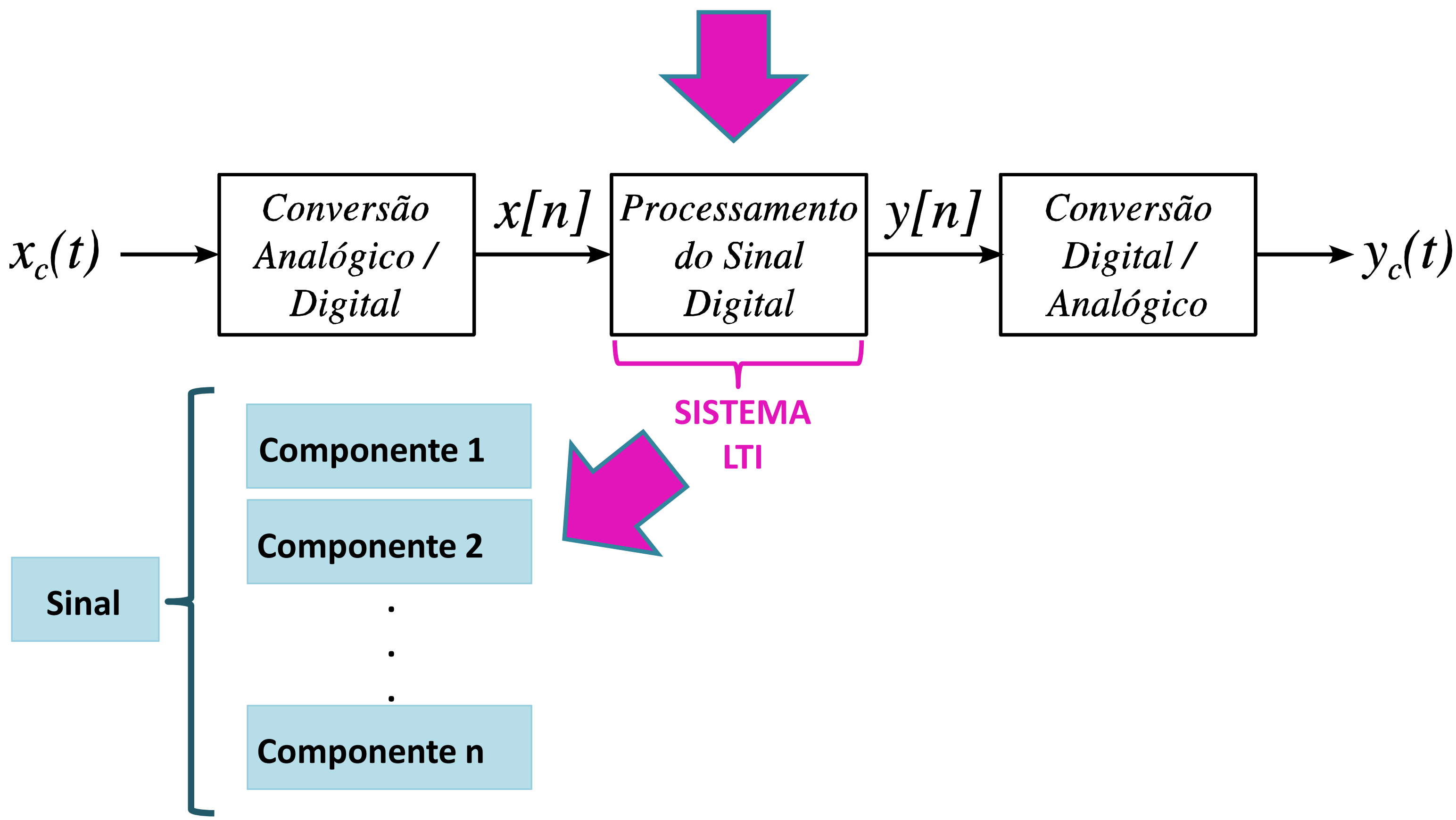


k=3
$x_3[n] = x[3]\delta[n-3]$
$y_3[n] = x[3]h[n-3]$
c
cd
ce
cf
Σ
$x[n] = x_{-2}[n] + x_0[n] + x_3[n]$
$y[n] = y_{-2}[n] + y_0[n] + y_3[n]$
af+bd
ad
ae
be
bf
cd
ce
cf

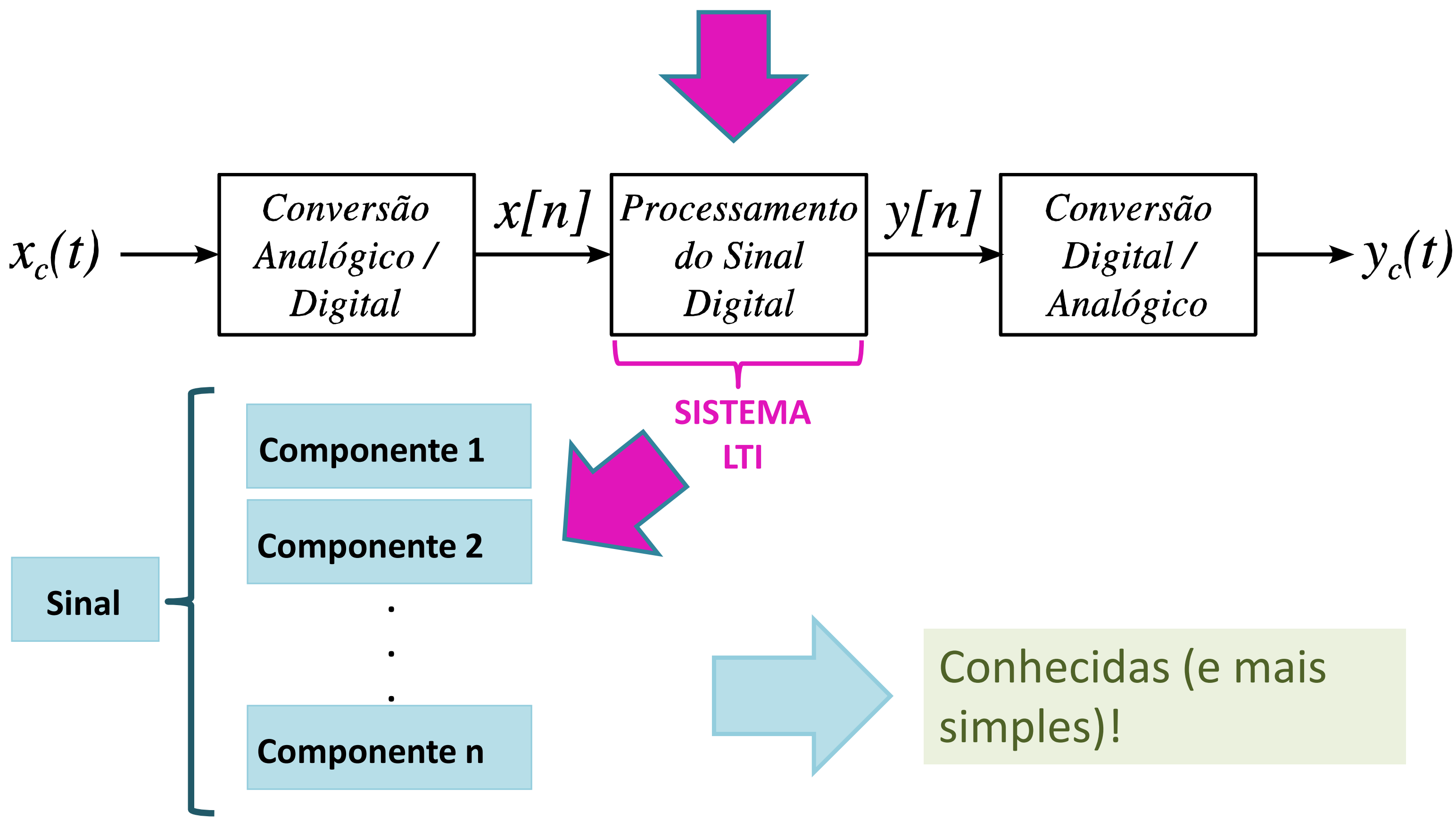
Processamento Digital de Sinais (PDS)



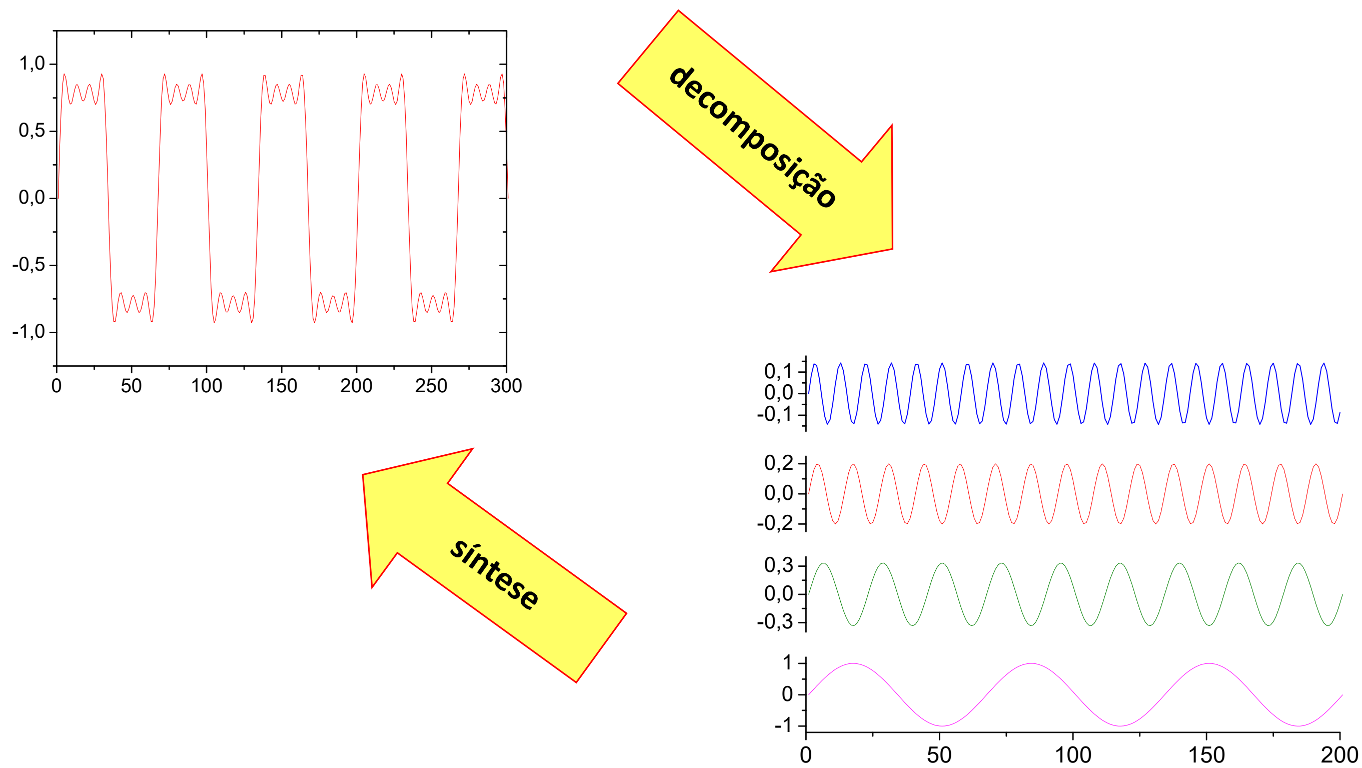
Processamento Digital de Sinais (PDS)



Processamento Digital de Sinais (PDS)



Como aplicar estes conceitos a PDS?



Sinais e Sistemas de Tempo Discreto

PROPRIEDADES DOS SISTEMAS LTI

Propriedades dos sistemas LTI

- Sistemas LTI são **descritos pela soma de convolução**

- As propriedades $\rightarrow$ definidas em função das propriedades da soma de convolução

- **A resposta impulsiva caracteriza completamente os sistemas LTI**

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

Propriedades dos sistemas LTI

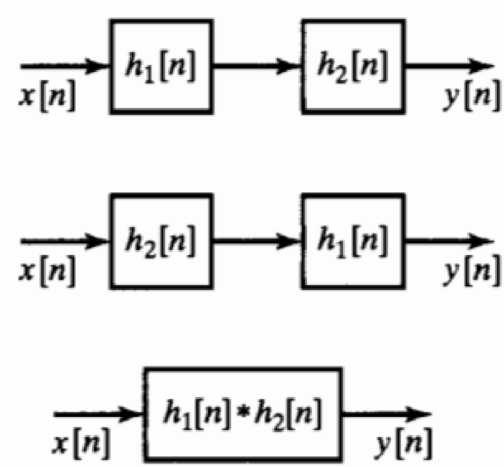
Comutatividade → **Cascata (série)**

- Em uma cascata de sistemas, a saída de um é entrada do outro

- A resposta impulsiva de uma combinação independe da ordem em que os sistemas estão em cascata

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

$$h[n] = h_1[n] * h_2[n].$$

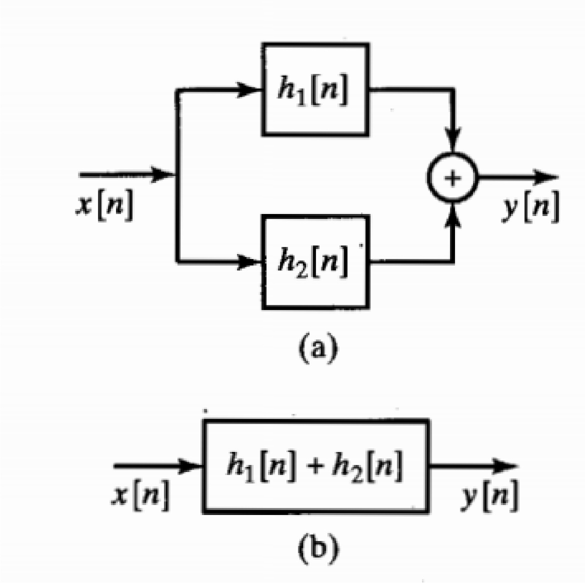


Propriedades dos sistemas LTI

Distributividade → **Paralelo**

- Em sistemas paralelos, a saída é somada

$$h[n] = h_1[n] + h_2[n].$$



Propriedades dos sistemas LTI

Estabilidade

Condição suficiente para estabilidade?

$$|y[n]| = \left| \sum_{k=-\infty}^{\infty} h[k]x[n-k] \right| \leq \sum_{k=-\infty}^{\infty} |h[k]|\,|x[n-k]|.$$

$$|x[n]| \leq B_x, \quad \Rightarrow \quad |y[n]| \leq B_x \sum_{k=-\infty}^{\infty} |h[k]|.$$

h[k] ser limitada é uma condição suficiente para estabilidade

$$S = \sum_{k=-\infty}^{\infty} |h[k]| < \infty.$$

Propriedades dos sistemas LTI

Estabilidade

Condição $|A + B| \leq |A| + |B|$ ra estabilidade?

$$|y[n]| = \left| \sum_{k=-\infty}^{\infty} h[k]x[n-k] \right| \leq \sum_{k=-\infty}^{\infty} |h[k]|\,|x[n-k]|.$$

$$|AB| = |A||B|$$

$$|x[n]| \leq B_x, \qquad |y[n]| \leq B_x \sum_{k=-\infty}^{\infty} |h[k]|.$$

h[k] ser limitada é uma condição suficiente para estabilidade

$$S = \sum_{k=-\infty}^{\infty} |h[k]| < \infty.$$

Exemplo

- Verificar se os seguintes sistemas são estáveis

$a)\ y[n] = (n - k)u[n - k]$

$b)\ y[n] = \delta(2n - k)$

Exemplo

Para um sistema ser estável basta que sua resposta impulsiva seja limitada.

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

a) $y[n] = |(n-k)|\, u[n-k]$

$$S = \sum_{k=-\infty}^{\infty} |(n-k)|\, u[n-k]$$

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

$$= \sum_{k=-\infty}^{n} |\underbrace{(n-k)}_{t}|$$

t	$=$	n	$-$	k
$-\infty$		n	$-$	∞
0	$=$	n	$-$	n

$$= \sum_{t=-\infty}^{0} |t| = \sum_{t=0}^{\infty} |t|$$

↳ é infinito

∴ não é estável

b) $y[n] = \delta[2n-k]$

↙ tem um ponto igual a 1

$\delta[2n-k] = \begin{cases} 1, & 2n = k \\ 0, & 2n \neq k \end{cases}$

↓

$$S = \sum_{k=-\infty}^{\infty} |\delta[2n-k]| = 1$$

↳ é uma soma finita

∴ o sistema é estável

Propriedades dos sistemas LTI

Causalidade

- Num sistema causal a contribuição de amostras futuras do sinal é nula

$$\boxed{h[n] = 0, \qquad n < 0,}$$

- Vantagem: Convolução é comutativa, logo

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

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

Propriedades dos sistemas LTI

Causalidade

- h[k] deve ser zero sempre que k<0

$$\sum_{k=-\infty}^{-1} x[k]h[n-k] = 0$$

Consequentemente

$$x[n] * h[n] = \sum_{k=0}^{\infty} x[k]h[n-k]$$

h[k] deve ser zero sempre que k<0

Sinais e Sistemas de Tempo Discreto

EQUAÇÕES DE DIFERENÇAS LINEARES COM COEFICIENTES CONSTANTES

Equações de diferenças lineares com coeficientes constantes

Oppenheim • Schafer

Processamento em tempo discreto de sinais

3ª edição

- Uma classe importante de sistemas LIT consiste naqueles sistemas para os quais a entrada $x[n]$ e a saída $y[n]$ satisfazem uma equação de diferenças linear de N-ésima ordem com coeficientes constantes na forma

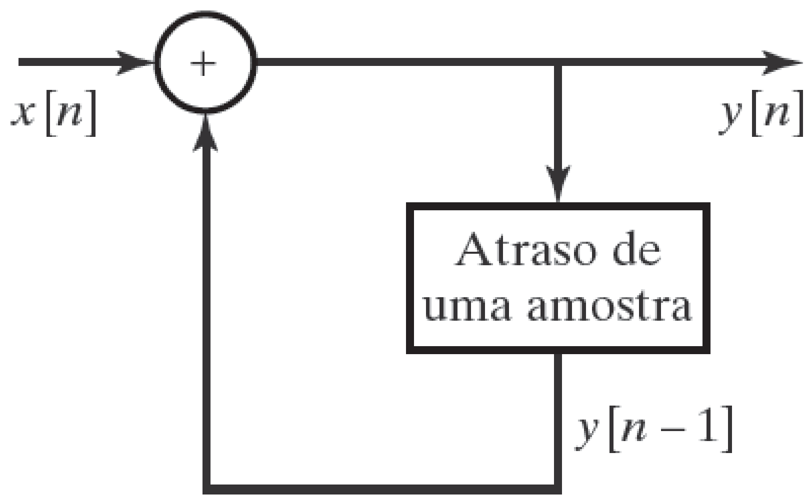
$$\boxed{\sum_{k=0}^{N} a_k y[n-k] = \sum_{m=0}^{M} b_m x[n-m]}$$

- A equação $y[n] = x[n] + y[n-1]$ e o diagrama de blocos na figura a seguir são chamados de representação recursiva do sistema.

Equações de diferenças lineares com coeficientes constantes

Oppenheim • Schafer
Processamento em tempo discreto de sinais
3ª edição

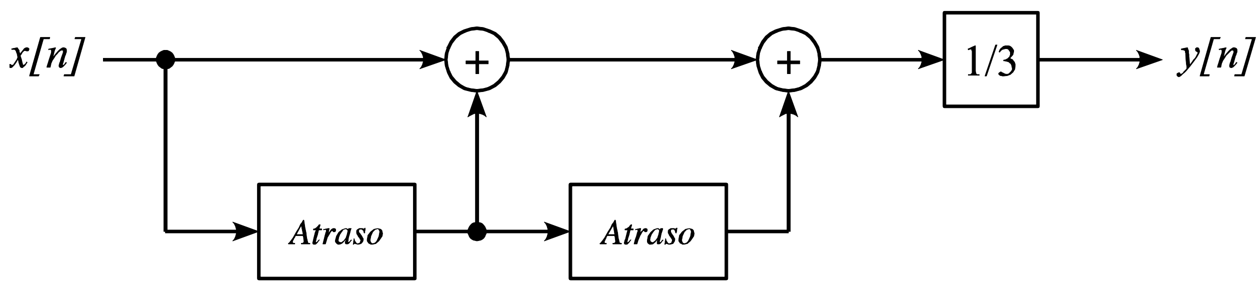
- Diagrama de blocos de uma equação de diferenças recursiva representando um acumulador.



Exemplo

- **Sistema de médias móveis** (*moving average system*)
  - Faz a média aritmética entre uma entrada e suas duas antecessoras

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



filter

- Eco → *time delay*

$$x[n] = y[n] + \alpha y[n - D]$$

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

Syntax

```
y = filter(b,a,x)
y = filter(b,a,x,zi)
y = filter(b,a,x,zi,dim)
[y,zf] = filter( __ )
```

Description

example

`y = filter(b,a,x)` filters the input data `x` using a rational transfer function defined by the numerator and denominator coefficients `b` and `a`.

If `a(1)` is not equal to `1`, then `filter` normalizes the filter coefficients by `a(1)`. Therefore, `a(1)` must be nonzero.

- If `x` is a vector, then `filter` returns the filtered data as a vector of the same size as `x`.
- If `x` is a matrix, then `filter` acts along the first dimension and returns the filtered data for each column.
- If `x` is a multidimensional array, then `filter` acts along the first array dimension whose size does not equal 1.

MATLAB

filter

- Eco → t

$x[n] = y[$

Syntax

```
y = filter(b,a,x
y = filter(b,a,x
y = filter(b,a,x
[y,zf] = filter(
```

Description

`y = filter(b,a,x)` f

coefficients `b` and `a`.

If `a(1)` is not equal to 1

- If x is a vector, then
- If x is a matrix, then
- If x is a multidimens

Equação de Diferenças

$$\underbrace{\sum_{k=0}^{N} a_k\, y[n-k]}_{\text{saídas}} = \underbrace{\sum_{m=0}^{M} b_m\, x[n-m]}_{\text{entradas}}$$

Comparando com o sistema dado

$$\underbrace{x[n]}_{\text{saída}} = \underbrace{y[n] + \alpha y[n-D]}_{\text{entradas}}$$

4196

$b_0 = 1$

$b_1 = 0$

$b_2 = 0$

$\vdots$

$b_{4196} = 0$

$b_{4197} = \alpha$

$\therefore$ b=[1, zeros(1,4196), α]

só tem $a_0 = 1 \rightarrow$ a=[1]

1 y

filter (b,a, sinal de entrada)

b=[1, zeros(1,D), α];

MATLA

filter

- Eco → *time delay*

$$x[n] = y[n] + \alpha y[n-D]$$

Editor - G:\Meu Drive\DISCIPLINAS\PROC SINAIS EM BIOS\MATLAB\ex_eco_filt.m

impseq.m | stepseq.m | ex_exp.m | ex_eco_filt.m | +

```
load ('handel.mat'); % sinal pre definido do MATLAB
sound(y,Fs); pause(10); %toca o sinal original
alpha =0.9; D=4196; %parâmetros do sinal de eco
b=[1,zeros(1,D),alpha]; %parâmetros do filtro
x=filter(b,1,y); %gera o sinal com eco
sound(x,Fs);
```

TAREFA

- A tarefa a seguir é para aqueles que estão com dificuldade em funções
  - Leiam o arquivo em anexo sobre funções*
  - Façam as funções a seguir que são simples
  - Em seguida refaça as funções feitas em sala de aula

*será disponibilizado

TAREFA

- Faça uma função para somar 2 números fornecidos pelo usuário $f(x, y) = x + y$

- Faça uma função que calcule o quadrado de um numero $f(x) = x^2$

- Faça uma função que dada a inclinação e o ponto de encontro com o eixo y trace o gráfico da reta

  $f(x) = ax + b$

- Faça uma função para somar 2 vetores fornecidos pelo usuário