Lecture 4 - Discrete Time Systems - Representations

July 6, 2015

Outline

Discrete Time System

For each time step $k \in \mathbb{Z}$ the system has:

- A vector of input $u[k]$
- A vector of output $y[k]$
- A vector of states $x[k]$

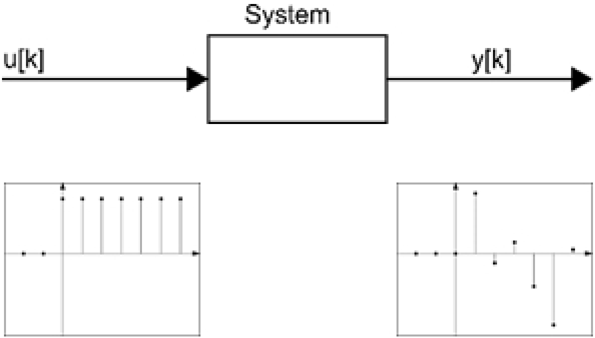


Figure:

How to represent a system?

- Block-diagram
- State space representation
- Difference/differential equation
- Impulse response
- Transferfunctions

Outline

Block diagram

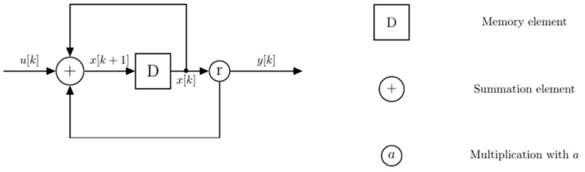
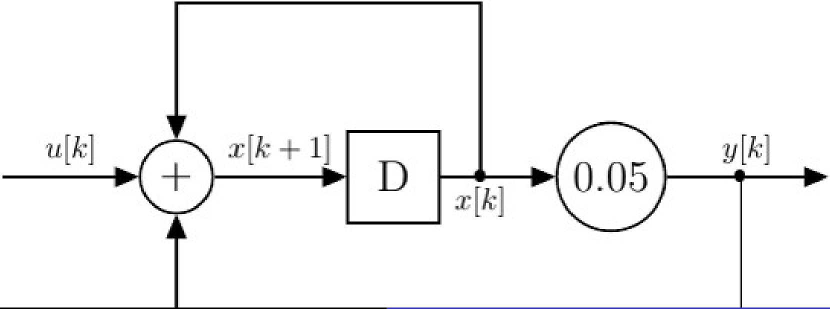


Figure: An example of a discrete time system

A block diagram is a visual representation of a system. All LTIs (Linear Time Invariant) systems can be constructed using these 3 building blocks(Memory element, summation element, multiplication element). Note that every memory element corresponds to one state variable.

Example: compond interes

$u[k]$:The deposits and withdrawals from the bank account $x[k]$:The current saldo on bank account(before deposit and interest) $y[k]$: The acquired interest of that year $x[k+1]$: The saldo on the next year = current saldo + interest + deposits



$u[k]$	$x[k+1]$	$x[k]$	$y[k]$
50	50	0	0
0	52.5	50	2.5
-25	30.13	52.5	2.62
0	31.63	30.13	1.51
0	33.21	31.63	1.58
30	64.87	33.21	1.66
0	68.12	64.87	3.24
0	71.52	68.12	3.41

Bad block diagrams

Delay-free loops: Delay-free loop: The issue is that this leads to an implicit connection $u[k]$ depends on $y[k]$,which is not yet known
You can easily rewrite this in an allowd shape
$y[k] = u[k] + 3y[k] \iff y[k] = -\frac{1}{2}u[k]$

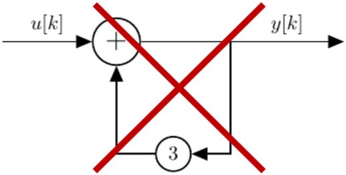


Figure: An example of a delay free loop

Connecting two outputs without using a sum: The issue is that this can lead to inconsistencies According to this block diagram the output of the systems S1 and S2 are equal There is no way to get around this.

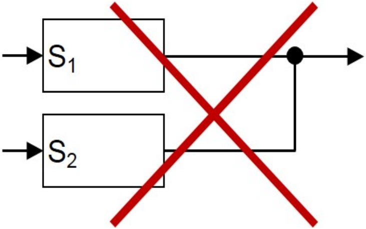


Figure:

Outline

State space representation

$$x[k+1] = Ax[k] + Bu[k]$$
$$y[k] = Cx[k] + Du[k]$$

This state space representation is again specific to LTI systems: Linear: its easy to see these systems are linear (see lecture about classification of dynamical systems) Time-invariant: the matrices A,B,C,D do not depend on time, if it were to be a time-variant system the matrices would be replaced by $A[k], B[k], C[k] and D[k]$.

From block diagram to state space

Blockdiagram:

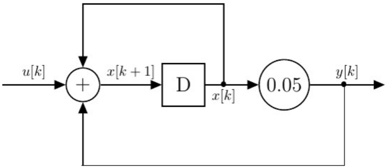


Figure:

State space representation:
In general Let the inputs of the memory element be and the outputs . Trace back to retrieve equations for $x_i[k+1]$ and $y_i[k]$
This results in:

$$x[k+1] = u[k] + 1.05x[k]$$
$$y[k] = 0.05x[k]$$

From state space to block diagram (DT)

$$x[k+1] = Ax[k] + Bu[k]$$
$$y[k] = Cx[k] + Du[k]$$

with $A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 3 & 0 \end{bmatrix}$, $B = \begin{bmatrix} 1 \\ 0 \\ 4 \end{bmatrix}$, $C = \begin{bmatrix} 5 & 1 & 0 \end{bmatrix}$ and $D = \begin{bmatrix} 1 \end{bmatrix}$

First add a delay element for every state $x_i[k]$