Systems

Linear systems

Advanced topics in machine learning

Francesco Corona

Department of Computer Science
Federal University of Ceará, Fortaleza

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Systems

A system

A **system** can be defined as a set of elements (or components) that cooperate in order to perform a specific functionality that would be otherwise impossible to attain for the individual components alone

This is definition is very fine, but it does not highlight an important fact

- The dynamical behaviour of the system

For us a central paradigm is that a system is subjected to external stimuli

- Stimuli influence the temporal evolution of the system itself

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Systems (cont.)

A system (reloaded)

A system is a physical entity, typically consisting of different interacting components, that responds to external stimuli producing a *determined/specific* dynamical behaviour

We study how to mathematically model a broad variety of systems

Our scope is to analyse their dynamical behaviour

- ⇝ We want to operate them appropriately
- ⇝ The design of control devices
- ⇝ Under external stimuli

The methodological approach shall be formal and system independent

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

General concepts

Systems

General topics

There is a wide spectrum of problems that spin around systems theory

⇝ System modelling, identification and analysis

- System control, optimisation and verification
- System diagnosis

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Modelling, identification and analysis

General concepts

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Modelling

To study a system, the availability of a **mathematical model** is crucial

⇝ A quantitative description of the behaviour of the system

The model is often constructed on the knowledge of the component devices

- A knowledge of the laws the system obeys to must be available, too

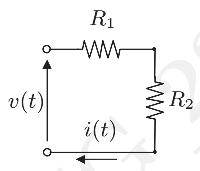
Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Modelling (cont.)

Example

Consider the electric circuit consisting of two serially arranged resistors

The current flow $i(t)$ [A] thru the system depends on tension $v(t)$ [V]



- $R_1 = 1$ [Ω]
- $R_2 = 3$ [Ω]

Both resistors can be assumed to follow Ohm's law[1]

$$\rightsquigarrow \quad v(t) = (R_1 + R_2)i(t) = 4i(t)$$

■

[1] The potential difference ('voltage') across an ideal conductor is proportional to the current that flows through it. The proportionality constant is called 'resistance'.

Identification

At times, we only have an incomplete knowledge on the system's devices

- The model must be constructed from observations
- (Observations of the system behaviour)

Case A) We have a knowledge on the type/number of component devices

- Not all of their parameters are known
- System observations are available

⇝ **Parametric identification**

⇝ **White-box identification**

Case B) We have no knowledge on the components and their parameters

- Observations of the system are available

⇝ **Black-box identification**

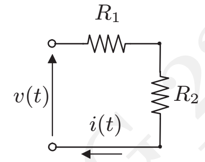
Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Identification (cont.)

Example

Consider the electric circuit consisting of two serially arranged resistors

The current flow $i(t)$ [A] thru the system depends on tension $v(t)$ [V]



- $R_1 = ?$
- $R_2 = ?$

Both resistors can still be assumed to follow Ohm's laws,

$$v(t) = (R_1 + R_2)i(t) = Ri(t)$$

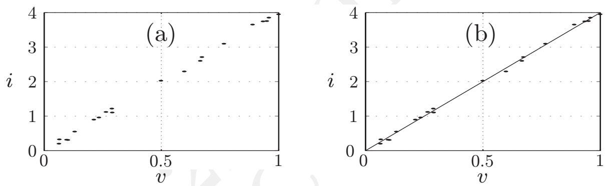
R is now an unknown system parameter

⇝ It can be identified from data

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Identification (cont.)

We can observe the system by collecting N pairs of measurements $\{(v_k, i_k)\}_{k=1}^{N}$



Often, such points will not be perfectly aligned along a line with slope R

⇝ **Disturbances** alter the behaviour to the system

⇝ **Measurement errors** are always present

We choose R corresponding to the line that *best* approximates the data

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Analysis

Systems analysis is about forecasting the future behaviour of a system

⇝ Based on the external stimuli it is subjected to

The availability of a mathematical model of the system is fundamental

- Needed to approach the problem in a quantitative manner

Analysis (cont.)

Example

The marine ecosystem is described thru time evolution of its fauna and flora

- Birth-growth-dead processes

The behaviour of the system is influenced by many factors

- Climate conditions, availability of food, ...
- Human predators, pollutants, ...
- ..., and so on

They *recently* spoke of reducing CO_2 emissions by injecting it into the sea

- CO_2 dissolves in sea water

The lack of a valid model limits our understanding about the system

- We do not know the response of the ecosystem

■

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Control, optimisation and validation

General concepts

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Control

The objective of **control** is to impose a desired behaviour on a system

We need to explicitly formulate what we mean by 'desired behaviour'

⇝ The **specifications** that such behaviour must satisfy

We need to design a device for implementing this task, the **controller**

⇝ The scope of a controller is to stimulate the system

⇝ Drive its evolution toward the desired behaviour

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Control (cont.)

Example

Consider a conventional network for the distribution of drinking water

- Water pressure must be kept constant throughout the net

We can measure pressure at various network locations

- Locations have nominal (target) pressure values

Specs suggest instantaneous variations within $\pm 10\%$ of nominal value

We identify two stimuli that act on the system (and modify its behaviour)

- The flow-rate of water withdrawn from the network
- The pressure imposed by the network pumps

We cannot control water withdrawals, they are understood as disturbances

Pump pressures can be manipulated to meet specifications

- This manipulation is done by the controller

■

Optimisation

Achieve a certain system's behaviour, while optimising a performance index

- **Optimisation** can be understood as a special case of control

We impose a desired behaviour, while optimising a **performance index**

- The index measures the quality of the behaviour of the system
- (Economic or operational terms)

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Optimisation (cont.)

Example

Consider a conventional suspension system of a conventional car

It is designed to satisfy two different needs

- ⇝ Appropriate passengers' comfort
- ⇝ Good handling in all conditions

Modern cars have suspensions based on 'semi-active' technology

- A controller (dynamically) changes the dumping factor
- It guarantees (a compromise between) the two needs

The optimiser takes into account of cabin and wheel oscillations

■

AUG 22, 2018 — FC

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Validation

Suppose that a mathematical model of a system under study is available

- Suppose that a set of desired properties can be formally expressed

Validation allows to see whether the model satisfies the properties

AUG 22, 2018 — FC

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Validation (cont.)

Example

Consider a conventional elevator

The system is controlled to guarantee that it responds correctly to requests

- The controller is an abstract machine
- Programmable logic controller (PLC)

Formal verification can be used to guarantee the correct functioning

■

AUG 22, 2018 — FC

Fault diagnosis

General concepts

Fault diagnosis

Systems deviate from nominal behaviour because of occurrence of faults

- ⇝ We need to detect the presence of an anomaly
- ⇝ We need to identify the typology of fault
- ⇝ We need to devise a corrective action

Fault diagnosis

Fault diagnosis (cont.)

Example

The human body is a complex system subjected to many potential faults

- We conventionally call them diseases

Consider the presence of fever, or another anomalous condition

- Symptoms reveal the presence of a disease

A doctor, once identified the pathology, prescribes a therapy ∎

Classification

Systems

Classification

The diversity of systems leads to a number of methodological approaches

- Each approach pertains a particular class of models

Conventional methodological approaches and model/system classification

By typology

- ⇝ Time-evolving systems
- Discrete-event systems
- Hybrid systems

By representation

- Input-output models
- ⇝ State-space models

Systems

UFC/DC
CK0255|TIP8244
2018.2

General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Time-evolving systems

Time-evolving systems

The system/model behaviour is described with functions, or signals

- The independent variable is time (t or k)

⇝Continuous time-evolving systems

- The time variable varies continuously

⇝ Discrete time-evolving systems

- The time variable takes discrete values

The signal can take values in a discrete set, **digital time-evolving systems**

Systems

UFC/DC
CK0255|TIP8244
2018.2

General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Time-evolving systems

Systems by typology

Systems

UFC/DC
CK0255|TIP8244
2018.2

General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Time-evolving systems (cont.)

The evolution of such systems is completely based on the passage of time

The functions of system behaviour satisfy differential/difference equations

- A specification of the relation between functions and their changes

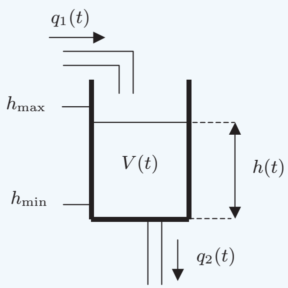
Time-evolving systems(cont.)

Example

Continuous time-evolving systems

Consider a tank whose volume of liquid $V(t)$ [m^3] varies in time

- Variation is due to input and output flow rates
- (By two externally operated pumps)



Tank cannot be emptied or filled

$$\rightsquigarrow \quad \frac{d}{dt}V(t) = q_1(t) - q_2(t)$$

- Output flow $q_2(t) \geq 0$ [m^3s^{-1}]
- Input flow $q_1(t) \geq 0$ [m^3s^{-1}]

The differential equation is a relation between continuous-time functions

$$V(t), q_1(t), q_2(t)$$

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Time-evolving systems(cont.)

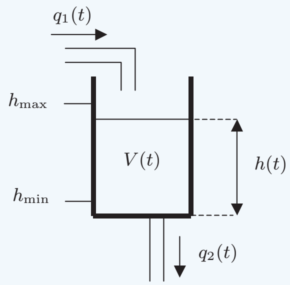
Example

Discrete time-evolving systems

Consider a tank whose volume of liquid $V(t)$ [m^3] varies in time

- Measurements are not continuously available
- Acquisitions only every Δt units of time

We are interested in the system behaviour at $0, \Delta t, 2\Delta t, \dots, k\Delta t, \dots$



We consider discrete-time variables

- $V(k) = V(k\Delta t)$
- $q_1(k) = q_1(k\Delta t)$
- $q_2(k) = q_2(k\Delta t)$

For $k = 0, 1, 2, \dots$

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Time-evolving systems(cont.)

We approximate the derivative with the difference quotient

$$\frac{d}{d}V(t) \approx \frac{\Delta V}{\Delta t} = \frac{V(k+1) - V(k)}{\Delta t}$$

Multiplying both sides of $\frac{\Delta V}{\Delta t} = q_1(k) - q_2(k)$ by Δt yields

$$\rightsquigarrow \quad V(k+1) - V(k) = q_1(k)\Delta t - q_2(k)\Delta t$$

The difference equation is a relation between discrete-time functions

$$V(k), q_1(k), q_2(k)$$

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Discrete-event systems

Systems by typology

AUG 22, 2018 — FC

Discrete-event systems

Discrete-event systems

Dynamic systems whose *states* take logical or symbolic values (not numeric)

The behaviour is characterised by the occurrence of instantaneous events

⇝ [At irregular (perhaps unknown) times]

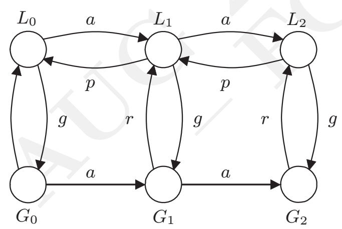
Discrete-event systems (cont.)

Example

Discrete-event systems

Consider a depot where parts are awaiting to be processed by some machine

- The number of parts awaiting to be processed cannot be larger than 2
- The machine can be either healthy (working) or faulty (stopped)



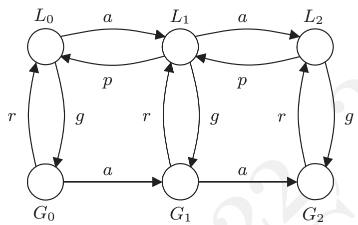
The state of the system

- Number of awaiting parts
- Status of the machine

The events of the system

- Changes in state

Discrete-event systems (cont.)



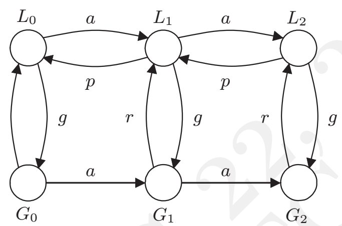
Six possible states

- L_0, L_1 and L_2
- G_0, G_1 and G_2

- L_0, the machine is working and the depot is empty
- L_1, the machine is working and there is one part in the depot
- L_2, the machine is working and there are two parts in the depot

- G_0, the machine is not working and the depot is empty
- G_1, the machine is not working and there is one part in the depot
- G_2, the machine is not working and there are two parts in the depot

Discrete-event systems (cont.)

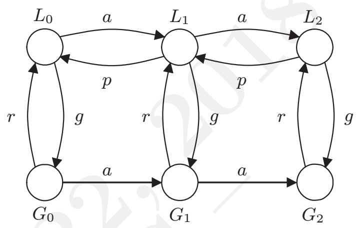


Four possible events

- a and p
- g and r

- a, a new part arrives to the depot
- p, the machines takes one part from the depot
- g, the machine gets faulty
- r, the machine gets fixed

Discrete-event systems (cont.)



Event a can only occur when the depot does not have two parts

$$a \rightsquigarrow \begin{cases} L_i \to L_{i+1} \\ G_i \to G_{i+1} \end{cases}$$

Event p can only occur when the deport is not empty

$$p \rightsquigarrow \left\{ L_i \to L_{i-1} \right.$$

Event g and r determine the switches $L_i \to G_i$ and $G_i \to L_i$, respectively

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Hybrid systems

Systems by typology

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Hybrid systems

A model that combines time-evolving dynamics and discrete-event dynamics

$\rightsquigarrow$ Thus, they are the most general class of dynamical systems

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

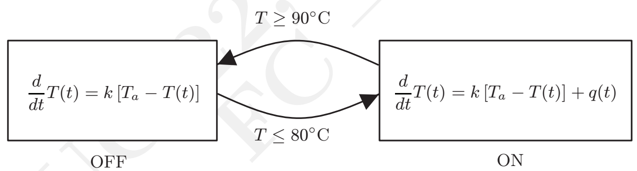
Hybrid systems (cont.)

Example

Hybrid systems

Consider a modern sauna, a cabin where the temperature is regulated

- A thermostat controls a stove used as heat generator
- Keep the temperature between 80°C and 90°C



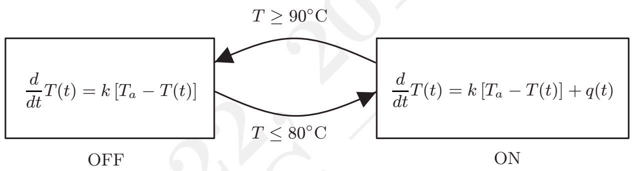
The thermostat can be represented using a discrete-event model

- Switch the heater ON
- Switch the heater OFF

The cabin can be represented using a time-evolving model

- Temperature $T(t)$

Hybrid systems(cont.)



When the thermostat is OFF

$T(t)$ in the cabin decreases, heat exchanged with the outside $[T_a < T(t)]$

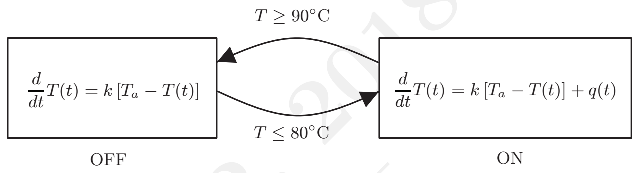
$$\frac{d}{dt}T(t) = k[T_a - T(t)], \quad \text{with } k > 0$$

Systems

UFC/DC
CK0255|TIP8244
2018.2

General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Hybrid systems(cont.)



When the thermostat is ON

$T(t)$ in the cabin increases, heat generated by the stove $q(t)$

$$\frac{d}{dt}T(t) = k[T_a - T(t)] + q(t)$$

The state of the system is $x = (l, T)$

- A logical variable $l \in \{\mathrm{ON}, \mathrm{OFF}\}$, representing the discrete state
- A real function $T(t) \in \mathcal{R}$, representing the continuous state

Systems

UFC/DC
CK0255|TIP8244
2018.2

General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Representation

Systems

Systems

UFC/DC
CK0255|TIP8244
2018.2

General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Representation

We provide fundamental concepts for the analysis of time-evolving systems

- Evolution generates from the passing of time
- Focus on continuous-time systems

We introduce the two main forms that are used for describing such systems

⇝ Mathematic formulations and example(s)

We conclude with a classification, based on some system/model properties

Representation (cont.)

Fundamental step to use formal techniques to study time-evolving systems

⇝ We describe the system behaviour in terms of functions

There are two possible such model/system descriptions

- **Input-output** (**IO**) representation

⇝ **State-space** (**SS**) representation

We define the mathematical elements and properties of these representations

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Input-output representation

Representation

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Input-output representation

The quantities involved in the input-output (IO) representation of a system

Causes

- Quantities that are generated outside the system
- Their evolution influences the system behaviour
- They are not influenced by the system behaviour

Effects

- Quantities whose behaviour is influenced by the causes
- Their evolution is influenced by nature of the system

By convention,

$$\rightsquigarrow \begin{cases} \textbf{Causes} & \rightsquigarrow \quad \textbf{Inputs} \\ \textbf{Effects} & \rightsquigarrow \quad \textbf{Outputs} \end{cases}$$

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Input-output representation (cont.)

A system

The system S can be seen as an operator or a processing/computing unit

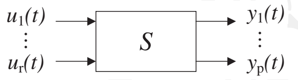
- It assigns a specific evolution to the output variables
- One for each possible evolution of the input variables

Input-output representation (cont.)

A system can have more than one (r) input and more than one (p) output

- Both inputs and outputs are assumed to be measurable/observable

A graphical IO system representation



r **inputs** (in $\mathcal{R}^r$)

$$\mathbf{u}(t) = \big[u_1(t) \cdots u_r(t)\big]^T$$

p **outputs** (in $\mathcal{R}^p$)

$$\mathbf{y}(t) = \big[y_1(t) \cdots y_p(t)\big]^T$$

Manipulable inputs

- They can be used for control

Non-manipulable inputs

- They are called disturbances

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Input-output representation (cont.)

Example

A car (IO representation)

Consider a car

Let its position and speed be the output variables

- They are both measurable

As input variables, we can consider wheel and gas position

- They are both measurable and manipulable

By acting on the input variables, we influence the output behaviour

- The changes depend on the specific system (car)
- (More precisely, by its dynamics)

Wind speed could be considered as an additional input variable

- It may be measurable but it is hardly manipulable

$r = 3$ inputs and $p = 2$ outputs (A MIMO system)

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Input-output representation (cont.)

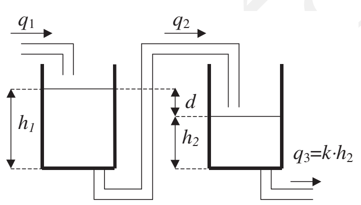
Example

Two tanks (IO representation)

Consider a system consisting of two cylindric tanks, both of base B [m^2]

- Output flow-rate from tank 1 is the input flow-rate to tank 2

⇝ q_2



First tank

- Input flow-rate q_1 [m^{3}s^{-1}]
- Output flow-rate q_2 [m^{3}s^{-1}]
- h_1 is the liquid level [m]

Second tank

- Input flow-rate q_2 [m^{3}s^{-1}]
- Output flow-rate q_3 [m^{3}s^{-1}]
- h_2 is the liquid level [m]

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Input-output representation (cont.)

Suppose that flow-rates q_1 and q_2 can be set to some desired value (pumps)

Also suppose that q_3 depends linearly on the liquid level in the tank

- $q_3 = kh_2$ [m^{3}s^{-1}], with k [m^{2}s^{-1}]
- k appropriate constant

Inputs, q_1 and q_2

⇝ Measurable and manipulable

⇝ They influence the liquid levels in the tanks

Output, $d = h_1 - h_2$

⇝ Measurable, but not manipulable

⇝ It is influenced indirectly only through inputs

State-space representation

Representation

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

State-space representation

For a given behaviour of the inputs, S defines the behaviour of the outputs

- ⇝ The output at time t is not only dependent on the inputs at time t
- ⇝ It also depends on the past behaviour (evolution) of the system

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

State-space description(cont.)

Example

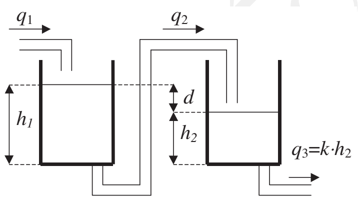
Two tanks (SS representation)

Consider a system consisting of two cylindric tanks, both of base B $[\text{m}^2]$

Let $d_0 = h_{1,0} - h_{2,0}$ be the output variable at time t_0

- ($h_{1,0}$ and $h_{2,0}$ are the liquid levels at time t_0)

Suppose that all input variables (q_1 and q_2) are zero at time t_0



- $q_{1,0} = 0$
- $q_{2,0} = 0$

Output $d(t)$ at any time $t > t_0$ depends on input values $q_1(t)$ and $q_2(t)$

- Over the entire interval $[t_o, t]$

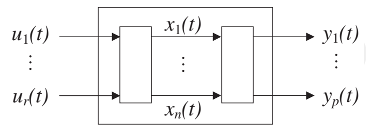
Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

State-space representation (cont.)

We can take this into account by introducing an *intermediate* variable

A variable that *exists* between inputs and outputs

- The **state** variable of the system



r **inputs** (in $\mathcal{R}^r$)

$\mathbf{u}(t) = [u_1(t) \cdots u_r(t)]^T$

n **states** (in $\mathcal{R}^n$)

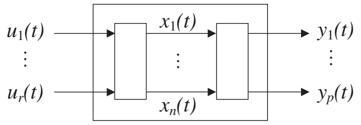
$\mathbf{x}(t) = [x_1(t) \cdots x_n(t)]^T$

p **outputs** (in $\mathcal{R}^p$)

$\mathbf{y}(t) = [y_1(t) \cdots y_p(t)]^T$

The state condenses information about past and present of the system

State-space representation (cont.)



The **state vector** $\mathbf{x}(t) = [x_1(t) \cdots x_n(t)]^T$ has n components

⇝ We say that n is the order of the system

- (In this representation)

Definition

State variable

The state of a system at time t_0 is a variable that contains the necessary information to univocally determine the behaviour of output $\mathbf{y}(t)$ for $t \geq t_0$

- *Given the behaviour of input $\mathbf{u}(t)$ for $t \geq t_0$ and the state itself at t_0*

State-space description (cont.)

In general, it is possible to select different physical entities as state variables

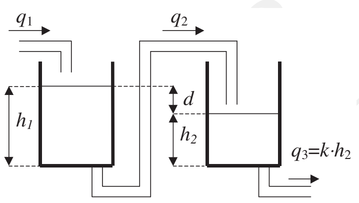
- The state variable is neither univocally defined, nor it is determined
- It is anything that can be seen as an *internal cause* of evolution
- (In general)

State-space representation (cont.)

Example

Two tanks (SS representation)

Consider a system consisting of two cylindric tanks, both of base B [m^2]



First tank

- Input flow-rate q_1 [m^{3}s^{-1}]
- Output flow-rate q_2 [m^{3}s^{-1}]
- h_1 is the liquid level [m]

Second tank

- Input flow-rate q_2 [m^{3}s^{-1}]
- Output flow-rate q_3 [m^{3}s^{-1}]
- h_2 is the liquid level [m]

Let $d_0 = h_{1,0} - h_{2,0}$ be the output variable at time t_0

- $h_{1,0}$ and $h_{2,0}$ are the liquid levels at time t_0

As state variable, select the volume of liquid in the tanks, $V_1(t)$ and $V_2(t)$

State-space representation (cont.)

We shall see that we are able to evaluate the output $d(t)$ for $t > t_0$

⇝ Need to know the initial state, $V_{1,0}$ and $V_{2,0}$, at t_0

⇝ Need to know the input, $q_1(t)$ and $q_2(t)$, in $[t_0, t]$

State-space representation (cont.)

Common to choose as state those variables that characterise system energy

- For a cylindric tank of base B and liquid level $h(t)$, the potential energy at time t is $E_p(t) = 1/2\rho g V^2(t)/B$, with ρ the density of the liquid and $V(t) = Bh(t)$. $V(t)$ or equivalently $h(t)$ can be used as state variable
- For a spring with elastic constant k, the potential energy at time t is $E_k(t) = 1/2kz^2(t)$ with $z(t)$ the spring deformation with respect to an equilibrium position. $z(t)$ can be used as state variable
- For a mass m moving with speed $v(t)$ on a plane, the kinetic energy at time t is $E_m(t) = 1/2mv^2(t)$. $v(t)$ can be used as state of the system

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

State-space description(cont.)

Consider a system in which there is energy stored, its state is not zero

- The system can evolve even in the absence of external inputs

The state can be understood as a possible (internal) cause of evolution

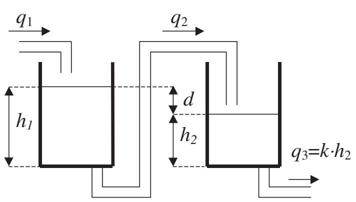
Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

State-space representation (cont.)

Example

Two tanks (SS representation, reloaded)

Consider any of the tanks in the two cylindric tanks system, base B [m^2]



First tank

- Input flow-rate q_1 [m^{3}s^{-1}]
- Output flow-rate q_2 [m^{3}s^{-1}]
- h_1 is the liquid level [m]

Second tank

- Input flow-rate q_2 [m^{3}s^{-1}]
- Output flow-rate q_3 [m^{3}s^{-1}]
- h_2 is the liquid level [m]

Each of the tanks can store a certain amount of potential energy

- The amount depends on the liquid volumes (levels)

The entire (two-tank) system has order 2

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Mathematical model

Representation

Mathematical model

System analysis studies the relations between the system inputs and outputs (IO) or, alternatively, between the system inputs, states and outputs (SS)

We are given certain input functions

- ⇝ Interest in understanding how states and outputs evolve in time

We need a model to describe quantitatively the system behaviour

- The relations between inputs (states) and outputs

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Mathematical model (cont.)

Input-output model

The relationship between the system output $\mathbf{y}(t) \in \mathcal{R}^p$ and its derivatives, the system input $\mathbf{u}(t) \in \mathcal{R}^r$ and its derivatives (a **differential equation**)

State-space model

It describes how the evolution $\dot{\mathbf{x}}(t) \in \mathcal{R}^n$ of the system state depends on the state $\mathbf{x}(t) \in \mathcal{R}^n$ itself and on the input $\mathbf{u}(t) \in \mathcal{R}^r$ (the **state equation**)

It describes how the system output $\mathbf{y}(t) \in \mathcal{R}^p$ depends on system state $\mathbf{x}(t) \in \mathcal{R}^n$ and on system input $\mathbf{u}(t) \in \mathcal{R}^r$ (the **output transformation**)

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Input-output model

Mathematical model

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Input-output model

The IO model of a SISO system is given as a differential equation

$$h\Big[\underbrace{y(t), \dot{y}(t), \ldots, y^{(n)}(t)}_{\text{output}}, \underbrace{u(t), \dot{u}(t), \ldots, u^{(m)}(t)}_{\text{input}}, \underbrace{t}_{\text{time}}\Big] = 0$$

- $\dot{y}(t) = \dfrac{d}{dt}y(t), \ldots, \dot{y}^2(t) = \dfrac{d}{dt^2}y(t)$ and $\ldots, y^{(n)}(t) = \dfrac{d^n}{dt^n}y(t)$
- $\dot{u}(t) = \dfrac{d}{dt}u(t), \ldots, \dot{u}^2(t) = \dfrac{d}{dt^2}u(t)$ and $\ldots, u^{(m)}(t) = \dfrac{d^m}{dt^m}u(t)$

h is a multi-parametric function that depends on the system

- n is the maximum order of derivation of the output
- m is the maximum order of derivation of the input

The **order of the system** (**model**) is n

Input-output model (cont.)

Example

Consider a system model given by the differential equation

$$\overbrace{2\underbrace{\dot{y}(t)y(t)}_{\text{output}} + 2\sqrt{\underbrace{t}_{\text{time}}}\underbrace{u(t)\ddot{u}(t)}_{\text{input}}}^{h} = 0$$

We have,

- Output order of derivation, $n = 1$
- Input order of derivation, $m = 2$

Function h links y and $\dot{y}$, and u and $\ddot{u}$, and t is the independent variable

The relationship *explicitly* depends on the independent variable (time)

$$\rightsquigarrow \quad \sqrt{t}$$

■

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Input-output model (cont.)

The IO model of a MIMO system with p outputs and r inputs

$$\begin{cases} h_1\Big[\underbrace{y_1(t), \dot{y}_1(t), \ldots, y_1^{(n_1)}(t)}_{\text{output 1}}, \underbrace{u_1(t), \dot{u}_1(t), \ldots, u_1^{(m_{1,1})}(t)}_{\text{input 1}}, \ldots, \underbrace{u_r(t), \ldots, u_r^{(m_{1,r})}(t)}_{\text{input r}}, t\Big] = 0 \\ h_2\Big[\underbrace{y_2(t), \dot{y}_2(t), \ldots, y_2^{(n_2)}(t)}_{\text{output 2}}, \underbrace{u_1(t), \dot{u}_1(t), \ldots, u_1^{(m_{2,1})}(t)}_{\text{input 1}}, \ldots, \underbrace{u_r(t), \ldots, u_r^{(m_{2,r})}(t)}_{\text{input r}}, t\Big] = 0 \\ \vdots \\ h_p\Big[\underbrace{y_p(t), \dot{y}_p(t), \ldots, y_p^{(n_p)}(t)}_{\text{output p}}, \underbrace{u_1(t), \dot{u}_1(t), \ldots, u_1^{(m_{p,1})}(t)}_{\text{input 1}}, \ldots, \underbrace{u_r(t), \ldots, u_r^{(m_{p,r})}(t)}_{\text{input r}}, t\Big] = 0 \end{cases}$$

h_i $(i = 1, \ldots, p)$ are multi-parametric functions depending on the system

- n_i, max order of derivation of the i-th component of output $y_i(t)$
- m_i, max order of derivation of the i-th component of input $u_i(t)$

A total of p differential equations

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

State-space model

Mathematical model

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

State-space model

The SS model of a SISO system is NOT a differential equation of order n

State equation

$$\begin{cases} \dot{x}_1(t) = f_1\big[x_1(t), \ldots, x_n(t), u(t), t\big] \\ \dot{x}_2(t) = f_2\big[x_1(t), \ldots, x_n(t), u(t), t\big] \\ \vdots \\ \dot{x}_n(t) = f_n\big[x_1(t), \ldots, x_n(t), u(t), t\big] \end{cases}$$

It links the derivative of each state with the other states and the input

Output transformation

$$y(t) = g\big[x_1(t), \ldots, x_n(t), u(t), t\big]$$

It further links the output with each state variable and the input

f_i with $i = 1, \ldots, n$ and g are multi-parametric functions

- They depend on (are) the dynamics of the system

State-space model (cont.)

Let $\dot{\mathbf{x}}(t)$ be the vector whose components are the derivatives of the state

$$\dot{\mathbf{x}}(t) = \frac{d}{dt}\mathbf{x}(t) = \begin{bmatrix} \dot{x}_1(t) \\ \vdots \\ \dot{x}_n(t) \end{bmatrix} = \begin{bmatrix} \frac{\mathrm{d}}{\mathrm{d}t}x_1(t) \\ \vdots \\ \frac{\mathrm{d}}{\mathrm{d}t}x_n(t) \end{bmatrix}$$

Systems
UFC/DC
CK0255|TIP8244
2018.2

General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

State-space model (cont.)

State-space model

$$\rightsquigarrow \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\big[\mathbf{x}(t), u(t), t\big] \\ y(t) = g\big[\mathbf{x}(t), u(t), t\big] \end{cases}$$

$\mathbf{f}$ is a vectorial function whose i-th component is f_i, with $i = 1, \ldots, n$

Systems
UFC/DC
CK0255|TIP8244
2018.2

General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

State-space model (cont.)

The SS model of a MIMO system with r inputs and p outputs

State equation

$$\begin{cases} \dot{x}_1(t) = f_1\big[x_1(t), \ldots, x_n(t), u_1(t), \ldots, u_r(t), t\big] \\ \dot{x}_2(t) = f_2\big[x_1(t), \ldots, x_n(t), u_1(t), \ldots, u_r(t), t\big] \\ \quad\vdots \\ \dot{x}_n(t) = f_n\big[x_n(t), \ldots, x_n(t), u_1(t), \ldots, u_r(t), t\big] \end{cases}$$

Output transformation

$$\begin{cases} y_1(t) = g_1\big[x_1(t), \ldots, x_n(t), u_1(t), \ldots, u_r(t), t\big] \\ y_2(t) = g_2\big[x_1(t), \ldots, x_n(t), u_1(t), \ldots, u_r(t), t\big] \\ \quad\vdots \\ y_p(t) = g_p\big[x_1(t), \ldots, x_n(t), u_1(t), \ldots, u_r(t), t\big] \end{cases}$$

Multi-parametric functions depending on the system, $\mathbf{f}$ and $\mathbf{g}$

- f_i with $i = 1, \ldots, n$
- g_i with $i = 1, \ldots, p$

Systems
UFC/DC
CK0255|TIP8244
2018.2

General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

State-space model (cont.)

State-space model

$$\rightsquigarrow \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\big[\mathbf{x}(t), \mathbf{u}(t), t\big] \\ \mathbf{y}(t) = \mathbf{g}\big[\mathbf{x}(t), \mathbf{u}(t), t\big] \end{cases}$$

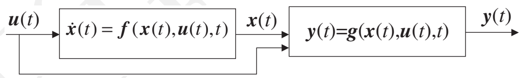
State-space model (cont.)

The state equation is a set of n first-order ordinary differential equations

- Regardless of the fact that the system is SISO or MIMO

The output transformation is a scalar or vectorial algebraic equation

- Depending on the number p of output variables



Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

State-space model (cont.)

The state-space representation of a system is central in our methods

- It offers a consistent framework for analysing systems
- Analysis of systems of arbitrary degree of complexity

Conversion of a scalar n-th order ordinary differential equation

- $\rightsquigarrow$ n first-order ordinary differential equations
- $\rightsquigarrow$ A first-order vector equation, dimension n

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

State-space model (cont.)

Example

Consider the third-order scalar equation

$$\dddot{x}(t) + c_3\ddot{x}(t) + c_2\dot{x}(t) + c_1 x(t) = bu(t)$$

We define,

$$\mathbf{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} = \begin{bmatrix} x(t) \\ \dot{x}(t) \\ \ddot{x}(t) \end{bmatrix}$$

Thus,

$$\rightsquigarrow \quad \dot{x}_3(t) + c_3 x_3(t) + c_2 x_2(t) + c_1 x_1(t) = bu(t)$$

This equation can be first integrated to get $x_3(t)$ [$\rightsquigarrow \ddot{x}(t)$]

- Two more integrations to get $x_2(t)$ [$\rightsquigarrow \dot{x}(t)$] and x_1 [$\rightsquigarrow x(t)$]

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

State-space model (cont.)

The three scalar differential equations that must be solved

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \end{bmatrix} = \begin{bmatrix} x_2(t) \\ x_3(t) \\ [-c_3 x_3(t) - c_2 x_2(t) - c_1 x_1(t) + bu(t)] \end{bmatrix} = \begin{bmatrix} f_1[\mathbf{x}(t), u(t), t] \\ f_2[\mathbf{x}(t), u(t), t] \\ f_3[\mathbf{x}(t), u(t), t] \end{bmatrix}$$

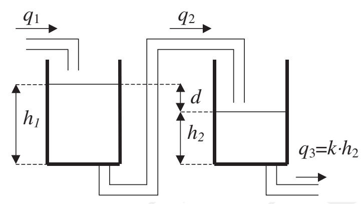
A single vector (state) equation

Examples

Systems

Examples

Example



First tank

- Input flow-rate q_1 $[\mathrm{m}^3\mathrm{s}^{-1}]$
- Output flow-rate q_2 $[\mathrm{m}^3\mathrm{s}^{-1}]$
- h_1 is the liquid level [m]

Second tank

- Input flow-rate q_2 $[\mathrm{m}^3\mathrm{s}^{-1}]$
- Output flow-rate q_3 $[\mathrm{m}^3\mathrm{s}^{-1}]$
- h_2 is the liquid level [m]

- $u_i = q_i$ with $i = 1, 2$, the input variables
- $y = d$, the output variable
- $x_1 = V_1$ and $x_2 = V_2$, the state variables

We are interested in the IO and the SS representations of the system

Examples (cont.)

For an incompressible fluid, by mass conservation

$$\begin{cases} \dfrac{\mathrm{d}\,V_1(t)}{\mathrm{d}t} = q_1(t) - q_2(t) \\ \dfrac{\mathrm{d}\,V_2(t)}{\mathrm{d}t} = q_2(t) - q_3(t) = q_2(t) - kh_2(t) \end{cases} \tag{1}$$

We can set $h_1 = V_1/B$, $h_2 = V_2/B$, and $q_3 = kh_2$

$$\rightsquigarrow \begin{cases} \dot{h}_1(t) = \dfrac{1}{B}q_1(t) - \dfrac{1}{B}q_2(t) \\ \dot{h}_2(t) = \dfrac{1}{B}q_2(t) - \dfrac{1}{B}q_3(t) = \dfrac{1}{B}q_2(t) - \dfrac{k}{B}h_2(t) \end{cases}$$

Examples (cont.)

Moreover, we have $y(t) = d(t) = h_1(t) - h_2(t)$

$$\dot{y}(t) = \dot{d}(t) = \dot{h}_1(t) - \dot{h}_2(t) = \underbrace{\Big[\frac{1}{B}q_1(t) - \frac{1}{B}q_2(t)\Big]}_{\dot{h}_1(t)} - \underbrace{\Big[\frac{1}{B}q_2(t) - \frac{k}{B}h_2(t)\Big]}_{\dot{h}_2(t)}$$

$$= \frac{1}{B}q_1(t) - \frac{2}{B}q_2(t) + \frac{k}{B}h_2(t)$$

$$= \frac{1}{B}u_1(t) - \frac{2}{B}u_2(t) + \frac{k}{B}[h_1(t) - y(t)]$$

We used $u_1(t) = q_1(t)$ and $u_2(t) = q_2(t)$

Examples (cont.)

$$\dot{y}(t) = \frac{1}{B}u_1(t) - \frac{2}{B}u_2(t) + \frac{k}{B}h_1(t) - \frac{k}{B}y(t)$$

By taking the second derivative of $y(t)$, we have

$$\begin{aligned}\ddot{y}(t) &= \frac{1}{B}\dot{u}_1(t) - \frac{2}{B}\dot{u}_2(t) + \frac{k}{B}\dot{h}_1(t) - \frac{k}{B}\dot{y}(t)\\ &= \frac{1}{B}\dot{u}_1(t) - \frac{2}{B}\dot{u}_2(t) + \underbrace{\frac{k}{B^2}u_1(t) - \frac{k}{B^2}u_2(t)}_{\frac{k}{B}\dot{h}_1(t)} - \frac{k}{B}\dot{y}(t)\end{aligned}$$

The IO system representation is a ordinary differential equation

$$\rightsquigarrow \quad \ddot{y}(t) + \frac{k}{B}\dot{y}(t) = \frac{1}{B}\dot{u}_1(t) - \frac{2}{B}\dot{u}_2(t) + \frac{k}{B^2}u_1(t) - \frac{k}{B}u_2(t)$$

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Examples (cont.)

$$\ddot{y}(t) + \frac{k}{B}\dot{y}(t) - \frac{1}{B}\dot{u}_1(t) + \frac{2}{B}\dot{u}_2(t) - \frac{k}{B^2}u_1(t) + \frac{k}{B}u_2(t) = 0$$

The obtained model is in the general IO form

$$\begin{cases} h_1\Big[\underbrace{y_1(t), \dot{y}_1(t), \ldots, y_1^{(n_1)}(t)}_{\text{output 1}}, \underbrace{u_1(t), \dot{u}_1(t), \ldots, u_1^{(m_1,1)}(t)}_{\text{input 1}}, \ldots, \underbrace{u_r(t), \ldots, u_r^{(m_1,r)}(t)}_{\text{input r}}, t\Big] \\ \quad = 0 \\ \vdots \\ h_p\Big[\underbrace{y_p(t), \dot{y}_p(t), \ldots, y_p^{(n_p)}(t)}_{\text{output p}}, \underbrace{u_1(t), \dot{u}_1(t), \ldots, u_1^{(m_p,1)}(t)}_{\text{input 1}}, \ldots, \underbrace{u_r(t), \ldots, u_r^{(m_p,r)}(t)}_{\text{input r}}, t\Big] \\ \quad = 0 \end{cases}$$

- $\rightsquigarrow$ $p = 1,\ n_1 = 2$
- $\rightsquigarrow$ $r = 2,\ m_1 = m_2 = 1$

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Examples (cont.)

The SS system representation is derived from mass conservation

$$\rightsquigarrow \begin{cases} \dfrac{\mathrm{d}\,V_1(t)}{\mathrm{d}t} = q_1(t) - q_2(t) \\ \dfrac{\mathrm{d}\,V_2(t)}{\mathrm{d}t} = q_2(t) - q_3(t) = q_2(t) - kh_2(t) \end{cases}$$

The state equation is obtained by setting $h_1 = x_1/B$ and $h_2 = x_2/B$

The output transformation,

$$y(t) = \frac{1}{B}x_1(t) - \frac{1}{B}x_2(t)$$

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Examples (cont.)

The resulting SS representation of the system

$$\rightsquigarrow \begin{cases} \dot{x}_1(t) = u_1(t) - u_2(t) \\ \dot{x}_2(t) = -\dfrac{k}{B}x_2(t) + u_2(t) \\ y(t) = \dfrac{1}{B}x_1(t) - \dfrac{1}{B}x_2(t) \end{cases}$$

Examples (cont.)

The model is in the general SS form

State equation

$$\begin{cases} \dot{x}_1(t) = f_1\big[x_1(t), \dots, x_n(t), u_1(t), \dots, u_r(t), t\big] \\ \dot{x}_2(t) = f_2\big[x_1(t), \dots, x_n(t), u_1(t), \dots, u_r(t), t\big] \\ \quad\vdots \\ \dot{x}_n(t) = f_n\big[x_n(t), \dots, x_n(t), u(t), t\big] \end{cases}$$

Output transformation

$$\begin{cases} y_1(t) = g_1\big[x_1(t), \dots, x_n(t), u_1(t), \dots, u_r(t), t\big] \\ y_2(t) = g_2\big[x_1(t), \dots, x_n(t), u_1(t), \dots, u_r(t), t\big] \\ \quad\vdots \\ y_p(t) = g_p\big[x_1(t), \dots, x_n(t), u_1(t), \dots, u_r(t), t\big] \end{cases}$$

Examples (cont.)

In general, the choice of the states is not unique

- We could have chosen the levels as states
- $x_1 = h_1$ and $x_2 = h_2$

$$\rightsquigarrow \begin{cases} \dot{x}_1(t) = -Bu_1(t) - Bu_2(t) \\ \dot{x}_2(t) = -kx_2(t) + Bu_2(t) \\ y(t) = x_1(t) - x_2(t) \end{cases}$$

■

System properties

Systems

System properties

We discuss a set of fundamental properties of time-evolving models

- Proper v Improper
- Linear v Non-linear
- With v Without delay
- Dynamical v Instantaneous
- Stationary v Non-stationary
- *Lumped v Distributed parameters*

Yet another way of classifying dynamical systems/models

System properties

Properties

Systems

UFC/DC
CK0255|TIP8244
2018.2

General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Dynamical v Instantaneous

Definition

Instantaneity

*A system is said to be **instantaneous** if the value of the output $\mathbf{y}(t) \in \mathcal{R}^p$ at time t only depends on the value of the input $\mathbf{u}(t) \in \mathcal{R}^r$ at time t*

*A system is said to be **dynamical**, otherwise*

Systems

UFC/DC
CK0255|TIP8244
2018.2

General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Dynamical v Instantaneous (cont.)

Proposition

IO representation - SISO

A necessary and sufficient condition for a SISO system to be instantaneous is that the IO relationship is expressed by an equation in the form

$$h\big[y(t), \cancel{\dot{y}(t), \ldots, y^{(n)}}(t), u(t), \cancel{\dot{u}(t), \ldots, u^{(m)}}(t), t\big] = 0$$

$$\rightsquigarrow \quad h\big[y(t), u(t), t\big] = 0$$

$\rightsquigarrow$ *The order of the derivatives of y and u is zero ($n = m = 0$)*

Systems

UFC/DC
CK0255|TIP8244
2018.2

General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Dynamical v Instantaneous (cont.)

$$h\big[y(t), u(t), t\big] = 0$$

If a SISO system is instantaneous, the IO relation is an algebraic equation

This is necessary, but it is not sufficient for a system to be instantaneous

Consider a differential equation as IO representation of a SISO system

$\rightsquigarrow$ Then, the system is certainly dynamical

Dynamical v Instantaneous systems (cont.)

Example

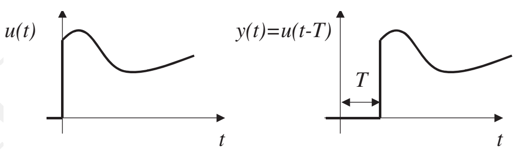
Counter-intuition

Consider the algebraic equation as IO representation of a SISO system

$$y(t) = u(t - T), \quad \text{with } T \in \mathcal{R}^+$$

The output $y(t)$ at time t does not depend on the input $u(t)$ at time t

- It depends on the input value $u(t - T)$, at a preceding moment



Such a system is not instantaneous, it is dynamical

⇝ Finite time delay system

Dynamical v Instantaneous (cont.)

Proposition

IO representation - MIMO

A necessary and sufficient condition for a MIMO system with r inputs and p outputs to be instantaneous is that the IO representation is expressed by a system of equations in the form

$$\begin{cases} h_1\big[y_1(t), u_1(t), u_2(t), \dots, u_r(t), t\big] = 0 \\ h_2\big[y_2(t), u_1(t), u_2(t), \dots, u_r(t), t\big] = 0 \\ \quad \cdots \\ h_p\big[y_p(t), u_1(t), u_2(t), \dots, u_r(t), t\big] = 0 \end{cases}$$

Dynamical v Instantaneous (cont.)

If a MIMO system is instantaneous, then the following conditions are true

⇝ The order of the derivatives of y_i is $n_i = 0$, for all $i = 1, \dots, p$

⇝ The order of the derivatives of u_i is $m_{j,i} = 0$, for all $j = 1, \dots, p$ and $i = 1, \dots, r$

The IO relation can be expressed as a system of p algebraic equations

If any of the IO relations is a differential equation, the system is dynamical

Dynamical v Instantaneous

Proposition

SS representation

Consider a model of a system expressed in SS form

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\big[\mathbf{x}(t), \mathbf{u}(t), t\big] \\ \mathbf{y}(t) = \mathbf{g}\big[\mathbf{x}(t), \mathbf{u}(t), t\big] \end{cases}$$

A necessary and sufficient condition for a system to be instantaneous is that the SS model is zero-order (i.e., there exists no state vector)

$$\rightsquigarrow \quad \mathbf{y}(t) = \mathbf{g}\big[\mathbf{u}(t), t\big]$$

Dynamical v Instantaneous

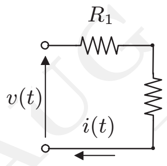
Example

Consider the two serially arranged resistors

$$v(t) = (R_1 + R_2)i(t) = Ri(t)$$

The system is instantaneous

The IO representation corresponds to the SS output transformation



$$v(t) = (R_1 + R_2)i(t) = Ri(t)$$

$$\rightsquigarrow \quad y(t) = \frac{1}{R}u(t)$$

The order of the system is zero (no device to store energy)

Linear v Nonlinear

Properties

Linear v Nonlinear

Definition

Linearity

*A system is said to be **linear** if it obeys the superposition principle*

*A system is said **nonlinear**, otherwise*

Superposition principle

Consider some system

- Let the system response to cause c_1 be equal to effect e_1
- Let the system response to cause c_2 be equal to effect e_2

The system response to cause $(\alpha c_1 + \beta c_2)$ equals effect $(\alpha e_1 + \beta e_2)$

- (whatever the constants α and β)

Linear v Non-linear (cont.)

Proposition

IO representation - SISO

A necessary and sufficient condition for a SISO system to be linear is that the IO representation is expressed by a linear differential equation

$$a_0(t)y(t) + a_1(t)\dot{y}(t) + \cdots + a_n(t)y^{(n)}(t) = b_0(t)u(t) + b_1(t)\dot{u}(t) + \cdots + b_m(t)u^{(m)}(t)$$

The coefficients of the IO representation are, in general, time dependent

Linear v Non-linear (cont.)

Linear differential equation

Consider the differential equation

$$h\big[y(t), \dot{y}(t), \dots, y^{(n)}(t), u(t), \dot{u}(t), \dots, u^{(m)}(t), t\big] = 0$$

The equation is linear if and only if the function h is a linear combination of the output and its derivatives, and of the input and its derivatives

$$\alpha_0(t)y(t) + \alpha_1(t)\dot{y}(t) + \dots + \alpha_n(t)y^{(n)}(t) + \beta_0(t)u(t) + \beta_1(t)\dot{u}(t) + \dots + \beta_m(t)u^{(m)}(t) = 0$$

⇝ A zero-sum weighted sum of inputs, outputs, and derivatives

Systems
UFC/DC
CK0255|TIP8244
2018.2

General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Linear v non-linear (cont.)

Consider a MIMO system in IO representation

Each function h_i, $i = 1, \dots, p$, must be a linear combination of the i-th component of the output and its n_i derivatives, and the input and its derivatives

⇝ The condition is necessary and sufficient

AUG 22, 2018 _ FC

Systems
UFC/DC
CK0255|TIP8244
2018.2

General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Linear v non-linear (cont.)

Proposition

SS representation

A necessary and sufficient condition for a system to be linear is that state equation and output transformation in the SS model are linear equations

$$\begin{cases} \dot{x}_1(t) = a_{1,1}(t)x_1(t) + \dots + a_{1,n}(t)x_n(t) + b_{1,1}u_1(t) + \dots + b_{1,r}(t)u_r(t) \\ \quad \dots \\ \dot{x}_n(t) = a_{n,1}(t)x_1(t) + \dots + a_{n,n}(t)x_n(t) + b_{n,1}u_1(t) + \dots + b_{n,r}(t)u_r(t) \\ \\ y_1(t) = c_{1,1}(t)x_1(t) + \dots + c_{1,n}(t)x_n(t) + d_{1,1}u_1(t) + \dots + d_{1,r}(t)u_r(t) \\ \quad \dots \\ y_p(t) = c_{p,1}(t)x_1(t) + \dots + c_{p,n}(t)x_n(t) + d_{p,1}u_1(t) + \dots + d_{p,r}(t)u_r(t) \end{cases}$$

AUG 22, 2018 _ FC

Systems
UFC/DC
CK0255|TIP8244
2018.2

General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Linear v non-linear (cont.)

$$\rightsquigarrow \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t) \\ \mathbf{y}(t) = \mathbf{C}(t)\mathbf{x}(t) + \mathbf{D}(t)\mathbf{u}(t) \end{cases}$$

⇝ $\mathbf{A}(t) = \{a_{i,j}(t)\} \in \mathcal{R}^{n \times n}$

⇝ $\mathbf{B}(t) = \{b_{i,j}(t)\} \in \mathcal{R}^{n \times r}$

⇝ $\mathbf{C}(t) = \{c_{i,j}(t)\} \in \mathcal{R}^{p \times n}$

⇝ $\mathbf{D}(t) = \{d_{i,j}(t)\} \in \mathcal{R}^{p \times r}$

Coefficient matrices **A**, **B**, **C** and **D** are, in general, time dependent

■

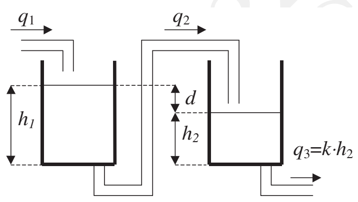
AUG 22, 2018 _ FC

Linear v non-linear (cont.)

Example

Consider a system consisting of two cylindric tanks, both of base B $[\mathrm{m}^2]$

- Output flow-rate from tank 1 is the input flow-rate to tank 2, $q_2(t)$



First tank

- Input flow-rate q_1 $[\mathrm{m}^3\mathrm{s}^{-1}]$
- Output flow-rate q_2 $[\mathrm{m}^3\mathrm{s}^{-1}]$
- h_1 is the liquid level $[\mathrm{m}]$

Second tank

- Input flow-rate q_2 $[\mathrm{m}^3\mathrm{s}^{-1}]$
- Output flow-rate q_3 $[\mathrm{m}^3\mathrm{s}^{-1}]$
- h_2 is the liquid level $[\mathrm{m}]$

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Linear v non-linear (cont.)

The SS representation of the system, from mass conservation

$$\rightsquigarrow \begin{cases} \rightsquigarrow \begin{cases} \dfrac{\mathrm{d}V_1(t)}{\mathrm{d}t} = q_1(t) - q_2(t) \\ \dfrac{\mathrm{d}V_2(t)}{\mathrm{d}t} = q_2(t) - q_3(t) = q_2(t) - kh_2(t) \end{cases} \\ \rightsquigarrow \begin{cases} y(t) = \dfrac{1}{B}x_1(t) - \dfrac{1}{B}x_2(t) \end{cases} \end{cases}$$

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Examples (cont.)

We obtained,

$$\rightsquigarrow \begin{cases} \dot{x}_1(t) & = u_1(t) - u_2(t) \\ \dot{x}_2(t) & = \dfrac{-k}{B}x_2(t) + u_2(t) \\ \dot{y}(t) & = \dfrac{1}{B}x_1(t) - \dfrac{1}{B}x_2(t) \end{cases}$$

We have,

$$\mathbf{A}(t) = \begin{bmatrix} 0 & 0 \\ 0 & \dfrac{-k}{B} \end{bmatrix}, \quad \mathbf{B}(t) = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix}$$

$$\mathbf{C}(t) = \begin{bmatrix} \dfrac{1}{B} & \dfrac{-1}{B} \end{bmatrix}, \quad \mathbf{D}(t) = \begin{bmatrix} 0 & 0 \end{bmatrix}$$

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Linear v non-linear (cont.)

Example

Counter-intuition

Consider the system described by the IO model

$$y(t) = u(t) + 1$$

The system violates the superposition principle

- It is thus nonlinear

Consider two constant inputs

- $u_1(t) = 1$
- $u_2(t) = 2$

We can calculate the outputs

- $\rightsquigarrow$ $y_1(t) = u_1(t) + 1 = 2$
- $\rightsquigarrow$ $y_2(t) = u_2(t) + 1 = 3$

Linear v non-linear (cont.)

Consider the combined input

$$u_3(t) = u_1(t) + u_2(t) = 3$$

The resulting output

$$y_3(t) = u_3(t) + 1 = 4$$

We thus have,

$$y_3(t) = 4 \neq y_1(t) + y_2(t) = 5$$

The IO representation is a nonlinear algebraic equation

- (Blame the +1 on the RHS for nonlinearity)

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Linear v non-linear (cont.)

Example

Counter-intuition

Consider the system described by the IO model

$$\dot{y}(t) + y(t) = \sqrt{t-1}\,u(t)$$

The system is linear

Consider the IO representation for a linear SISO system

$$a_0(t)y(t) + a_1(t)\dot{y}(t) + \cancel{\cdots + a_n(t)y^{(n)}(t)}$$
$$= b_0(t)u(t) + \cancel{b_1(t)\dot{u}(t) + \cdots + b_m(t)u^{(m)}(t)}$$

- ⇝ $a_0(t) = 1$
- ⇝ $a_1(t) = 1$
- ⇝ $b_0(t) = \sqrt{t-1}$

System $\dot{y}(t) + y(t) = \sqrt{t-1}\,u(t)$ is thus linear

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Stationary v non-stationary

Properties

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Stationary and non-stationary systems

Definition

Stationarity

A system is said to be stationary (or time-invariant), if it obeys the translation principle

A system is said to be non-stationary (or time-varying), otherwise

Translation principle

Consider some system

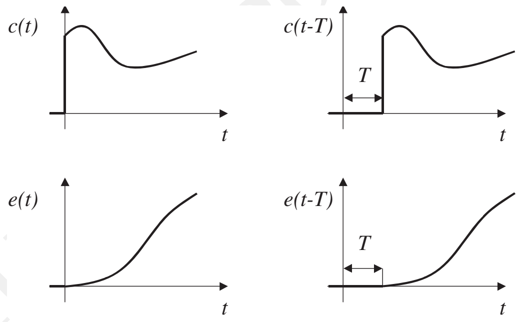
Let the system response to a cause $c_1(t)$ be equal to an effect $e_1(t)$

System response to cause $c_2(t) = c_1(t-T)$ equals effect $e_2(t) = e(t-T)$

Stationary and non-stationary systems (cont.)

Let the same cause be applied to system S at 2 different times

- ⇝ $t = 0$
- ⇝ $t = T$



The resulting effects are analogous

- Shifted by time interval T

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Stationary and non-stationary systems (cont.)

In nature, no system is stationary

Yet, there exists a wide range of variations that can be neglected

- Over large time intervals

Over such intervals, the systems can be considered as stationary

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Stationary and non-stationary systems (cont.)

Proposition

IO representation

A necessary and sufficient condition for a system to be stationary

⇝ *The IO representation must not explicitly depend on time*

Consider the SISO system

$$h\big[y(t), \dot{y}(t), \dots, y^{(n)}(t), u(t), \dot{u}(t), \dots, u^{(m)}(t), t\big] = 0$$

Then, the stationary model becomes

$$\rightsquigarrow \quad h\big[y(t), \dot{y}(t), \dots, y^{(n)}(t), u(t), \dot{u}(t), \dots, u^{(m)}(t)\big] = 0$$

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Stationary and non-stationary systems (cont.)

Consider a linear SISO system

$$a_0(t)y(t) + a_1(t)\dot{y}(t) + \cdots + a_n(t)y^{(n)}(t) = b_0(t)u(t) + b_1(t)\dot{u}(t) + \cdots + b_m(t)u^{(m)}(t)$$

The model becomes a linear differential equation

$$\rightsquigarrow \quad a_0 y(t) + a_1 \dot{y}(t) + \cdots + a_n y^{(n)}(t) = b_0 u(t) + b_1 \dot{u}(t) + \cdots + b_m u^{(m)}(t)$$

The coefficients are constant

Stationary and non-stationary systems (cont.)

Theorem

SS representation

A necessary and sufficient condition for a system to be stationary

- ⇝ *The SS representation must not explicitly depend on time*
- *(Both state equation and output transformation)*

Consider the system

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\big[\mathbf{x}(t), \mathbf{u}(t), t\big] \\ \mathbf{y}(t) = \mathbf{g}\big[\mathbf{x}(t), \mathbf{u}(t), t\big] \end{cases}$$

Then, the stationary model becomes

$$\rightsquigarrow \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\big[\mathbf{x}(t), \mathbf{u}(t)\big] \\ \mathbf{y}(t) = \mathbf{g}\big[\mathbf{x}(t), \mathbf{u}(t)\big] \end{cases}$$

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Stationary and non-stationary systems (cont.)

Consider a linear system

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t) \\ \mathbf{y}(t) = \mathbf{C}(t)\mathbf{x}(t) + \mathbf{D}(t)\mathbf{u}(t) \end{cases}$$

The model becomes

$$\rightsquigarrow \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \end{cases}$$

The (elements of the) coefficient matrices **A**, **B**, **C** and **D** are constant

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Stationary and non-stationary systems (cont.)

Example

Consider the instantaneous and linear system

$$y(t) = tu(t)$$

The system is non-stationary

We can show this by using the translation principle

Consider the input

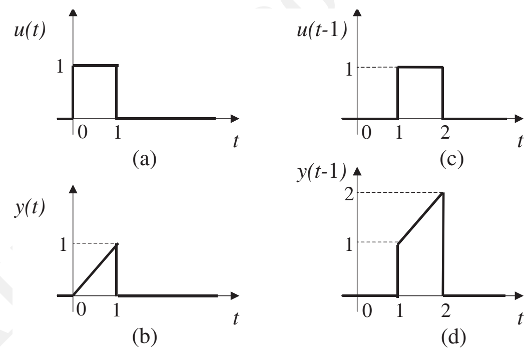
$$u(t) = \begin{cases} 1, & t \in [0, 1] \\ 0, & \text{elsewhere} \end{cases}$$

If the same input is applied with a delay, the output is not simply shifted

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Stationary and non-stationary systems (cont.)

The same input is applied with one (1) time-unit delay



Proper v improper

Properties

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Proper and improper systems

Definition

Appropriateness

A system is said to be ***proper****, if it obeys the causality principle*

The system is said to be ***improper****, otherwise*

Causality principle

The effect does not precede the generating cause

In nature, all systems are (obviously?) proper

- Only the model can be improper

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Proper v improper (cont.)

Proposition

IO representation - SISO

A necessary and sufficient condition for a SISO system to be proper

⇝ *The order of derivation of the output (n) is equal to or larger than the order of derivation of the input (m)*

$$h\big[y(t), \dot{y}(t), \dots, y^{(n)}(t), u(t), \dot{u}(t), \dots, u^{(m)}(t), t\big] = 0, \ \textit{with } n \geq m$$

A system where $n > m$ is said to be **strictly proper**

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Proper and improper systems (cont.)

The result can be extended to MIMO systems

$$\begin{cases} h_1\big[\underbrace{y_1(t), \dot{y}_1(t), \dots, y_1^{(n_1)}(t)}_{\text{output 1}}, \underbrace{u_1(t), \dot{u}_1(t), \dots, u_1^{(m_{1,1})}(t)}_{\text{input 1}}, \dots, \underbrace{u_r(t), \dots, u_r^{(m_{1,r})}(t)}_{\text{input r}}, t\big] = 0 \\ h_2\big[\underbrace{y_2(t), \dot{y}_2(t), \dots, y_2^{(n_2)}(t)}_{\text{output 2}}, \underbrace{u_1(t), \dot{u}_1(t), \dots, u_1^{(m_{2,1})}(t)}_{\text{input 1}}, \dots, \underbrace{u_r(t), \dots, u_r^{(m_{2,r})}(t)}_{\text{input r}}, t\big] = 0 \\ \vdots \\ h_p\big[\underbrace{y_p(t), \dot{y}_p(t), \dots, y_p^{(n_p)}(t)}_{\text{output p}}, \underbrace{u_1(t), \dot{u}_1(t), \dots, u_1^{(m_{p,1})}(t)}_{\text{input 1}}, \dots, \underbrace{u_r(t), \dots, u_r^{(m_{p,r})}(t)}_{\text{input r}}, t\big] = 0 \end{cases}$$

None of the equations must include derivatives of the input variables whose order is larger than the derivation order of corresponding output variables

$$n_i \geq \max_{j=1,\dots,r} m_{i,j}, \quad \text{for all } i = 1, \dots, p$$

A system is strictly proper if the inequality is strictly true, for all $i = 1, \dots, p$

Proper and improper systems (cont.)

Proposition

SS representation

Consider a system described by a SS model

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\big[\mathbf{x}(t), \mathbf{u}(t), t\big] \\ \mathbf{y}(t) = \mathbf{g}\big[\mathbf{x}(t), \mathbf{u}(t), t\big] \end{cases}$$

Such a system/model is always proper

A strictly proper system has an output transformation independent on $\mathbf{u}(t)$

$$\rightsquigarrow \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\big[\mathbf{x}(t), \mathbf{u}(t), t\big] \\ \mathbf{y}(t) = \mathbf{g}\big[\mathbf{x}(t), t\big] \end{cases}$$

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

Proper and improper systems (cont.)

The SS model of a linear, stationary and strictly proper system

$$\rightsquigarrow \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) \end{cases}$$

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

With v without delay

Properties

Systems
UFC/DC
CK0255|TIP8244
2018.2
General concepts
Modelling, identification and analysis
Control, optimisation and validation
Fault diagnosis
Classification
Representation
Input-output representation
State-space representation
The model
Examples
System properties
Dynamical v Instantaneous
Linear v Nonlinear
Stationary v non-stationary
Proper v improper
With v without delay

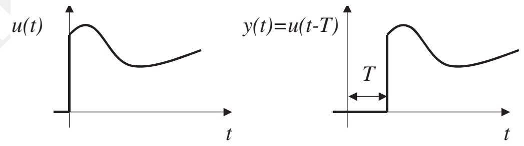
Systems with and without delay

Definition

Finite time delay

A ***finite delay*** *is a system whose output* $y(t)$ *at time* t *is equal to the input* $u(t-T)$ *at time* $t-T$

$\rightsquigarrow$ $T \in (0, +\infty)$ is called the **time delay**



Systems with and without delay (cont.)

Consider the algebraic equation describing a finite delay element

$$y(t) = u(t - T), \text{ with } T \in \mathcal{R}^+$$

- Such a system is not instantaneous
- The system is dynamic

The output at time t depends on the previous values of the input

Systems with and without delay (cont.)

Proposition

IO and SS representation

A necessary and sufficient condition for a system to be without a time delay

$\rightsquigarrow$ *All the signals in the model (IO or SS) must share the same argument*

Systems with and without delay (cont.)

Example

Consider a system described by the IO model

$$4\dot{y}(t) + 2y(t) = u(t - T)$$

The system has delay elements

- There are signals that are dependent on t
- There are signals that are dependent on $t - T$

Example

Consider a system described by the SS model

$$\begin{cases} \dot{x}(t) = x(t - T) + u(t) \\ y(t) = 7x(t) \end{cases}$$

The system/model has delay elements

- There are signals that are dependent on t
- There are signals that are dependent on $t - T$