Systems Dynamics

Course ID: 267MI – Fall 2018

Thomas Parisini
Gianfranco Fenu

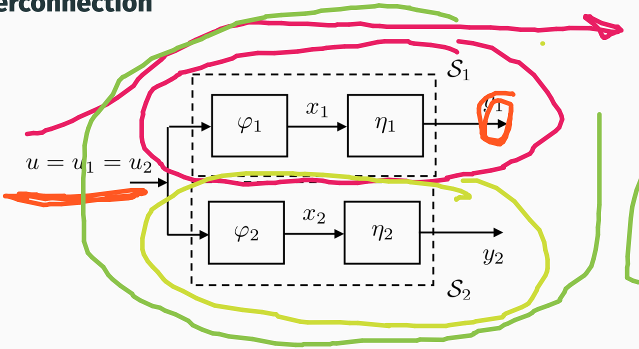
University of Trieste
Department of Engineering and Architecture

267MI –Fall 2018

Lecture 1
Generalities: systems and models

Parallel interconnection

$y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$



$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

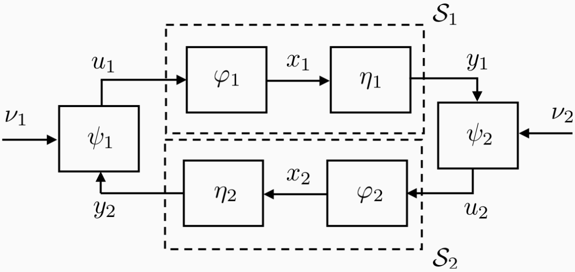
$$\mathcal{S} = \{T = T_1 = T_2, U = U_1 = U_2, \Omega = \Omega_1 = \Omega_2, X = X_1 \times X_2, Y = Y_1 \times Y_2, \\ \Gamma = \Gamma_1 \times \Gamma_2\}$$

$$X \begin{cases} (x_1(t), x_2(t)) = (\varphi_1(t, t_0, x_1(t_0), u(\cdot)), \varphi_2(t, t_0, \overset{x_2}{\cancel{x_1}}(t_0), u(\cdot))) \\ \qquad \to \varphi_2(t, t_0, x_2(t_0), \eta_1(t, \underline{\varphi_1(t, t_0, x_1(t_0), u(\cdot))}))) \\ (y_1(t), y_2(t)) = (\eta_1(t, x_1(t)), \eta_2(t, x_2(t))) \end{cases}$$

CASCATA

Feedback interconnection

General scheme:

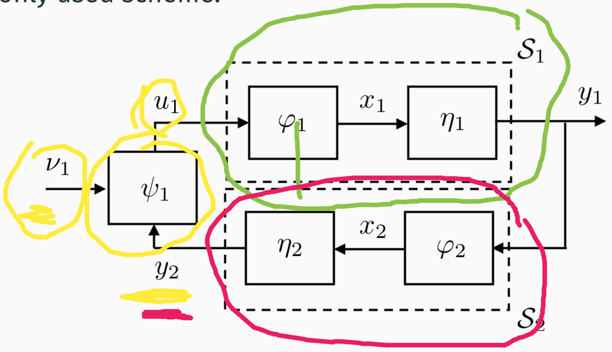


$$u_1(t) = \psi_1(y_2(t), \nu_1(t), t)$$

$$u_2(t) = \psi_2(y_1(t), \nu_2(t), t)$$

Feedback interconnection

Commonly used scheme:

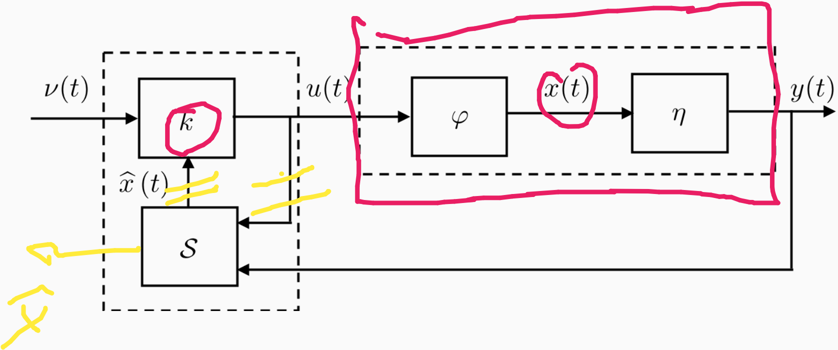


$$\mathcal{S} = \{T = T_1 = T_2, U = V_1, \Omega = \Omega_{\nu_1}, X = X_1 \times X_2, Y = Y_1, \Gamma = \Gamma_1\}$$

$$\begin{cases} (x_1(t), x_2(t)) = (\varphi_1(t, t_0, x_1(t_0), \psi_1(\nu_1(\cdot), y_2(\cdot)), \varphi_2(t, t_0, x_2(t_0), y_1(\cdot)))) \\ y(t) = y_1(t) = \eta_1(t, x_1(t)) \end{cases}$$

Feedback Interconnection: a Notable Example

A notable example of feedback interconnection is the **state control law + state observer** scheme (will be dealt with in the *Control Theory* course)



Finite-dimensional Regular Systems

A dynamic systems is **regular** if:

- U, Ω, X, Y, Γ are normed vector spaces
- $\varphi(\cdot,\cdot,\cdot,\cdot)$ is a continuous function with respect its arguments
- $\dfrac{\mathrm{d}}{\mathrm{d}t}\varphi(t, t_0, x_0, u(\cdot))$ does exist and it is continuous for all values of the arguments where $u(\cdot)$ is continuous

The state movement $\varphi(t, t_0, x_0, u(\cdot))$ of a regular finite-dimensional dynamic system is the **unique solution** of a suitable vector differential equation

$$\begin{cases} \dot{x}(t) = f(x(t), u(t), t) \\ x(t_0) = x_0 \end{cases}$$

φ

and

$$y(t) = g(x(t), u(t), t)$$

Finite-dimensional Discrete-time Dynamic Systems

Discrete-time dynamic systems obtain by sampling a continuous-time regular system

- U, X, Y finite-dimensional normed vector spaces
- $\Omega = \{u(\cdot) : \text{ piecewise constant } u_i(\cdot)\,,\; i = 1, \ldots, m\}$
- Sampling time ΔT:

$$u(k) = u(t)\,,\; t_0 + k\Delta T \le t < t_0 + (k+1)\Delta T,\; k = 0, 1, \ldots$$
$$y(k) = y(t_0 + k\Delta T),\; k = 0, 1, \ldots$$

Then:

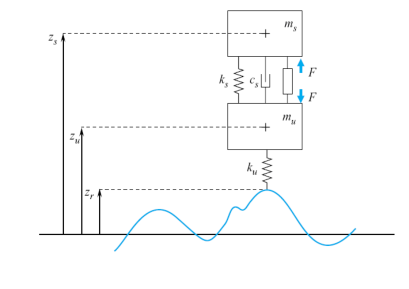
$$\begin{cases} x(k+1) = f_d(x(k), u(k), k) \\ y(k) = g_d(x(k), u(k), k) \end{cases}$$

where (from composition property of φ):

$$f_d(x(k), u(k), k) = \varphi(t_0 + (k+1)\Delta T, t_0 + k\Delta T, x(k), u(k))$$
$$g_d(x(k), u(k), k) = \eta(x(k), u(k), t_0 + k\Delta T)$$

An example: continuous-time model of a car suspension

From a real vehicle ...

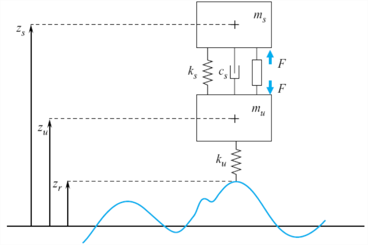


to a simplified *quarter-car model*

quarter-car model hypotheses

- vehicle as assembly of four decoupled parts
- each part consists of
  - the *sprung mass*: a quarter of the vehicle mass, supported by a suspension actuator, placed between the vehicle and the tyre
  - the *unsprung mass*: the wheel/tyre sub-assembly
- the model allows only for vertical motion: the vehicle is moving forward with an almost constant speed

Continuous-time model of a car suspension (cont.)



- state variables:
  - vertical positions of sprung and unsprung masses vs. the corresponding steady-state values
  - vertical speeds of masses
- inputs:
  - ground vertical position vs. the steady-state
  - active actuator force
- outputs:
  - sprung mass vertical acceleration
  - contact force between tyre and ground

$$
\begin{cases}
x_1(t) = z_s(t) - \bar{z}_s \\
x_2(t) = z_u(t) - \bar{z}_u \\
x_3(t) = \dot{x}_1(t) \\
x_4(t) = \dot{x}_2(t) \\
u_1(t) = z_r(t) - \bar{z}_r \\
u_2(t) = F(t) \\
y_1(t) = \ddot{x}_1 \\
y_2(t) = k_u\left(x_2(t) - u_1(t)\right)
\end{cases}
$$

Continuous-time model of a car suspension (cont.)

$$
\begin{cases}
\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{k_s}{m_s} & \frac{k_s}{m_s} & -\frac{c_s}{m_s} & \frac{c_s}{m_s} \\ \frac{k_s}{m_u} & -\frac{k_s + k_u}{m_u} & \frac{c_s}{m_u} & -\frac{c_s}{m_u} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & \frac{1}{m_s} \\ \frac{k_s}{m_u} & -\frac{1}{m_u} \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \\
\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} -\frac{k_s}{m_s} & \frac{k_s}{m_s} & -\frac{c_s}{m_s} & \frac{c_s}{m_s} \\ 0 & k_u & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 & \frac{1}{m_s} \\ -k_u & 0 \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}
\end{cases}
$$

Continuous-time car suspension: an example

Assuming

$$m_s = 400.0 \text{ kg} \qquad m_u = 50.0 \text{ kg} \qquad c_s = 2.0\,10^3 \text{ N s m}^{-1}$$
$$k_s = 2.0\,10^4 \text{ N m}^{-1} \qquad k_u = 2.5\,10^5 \text{ N m}^{-1}$$

the car suspension model becomes

$$\begin{cases} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1.0 \\ -50.0 & 50.0 & -5.0 & 5.0 \\ 400.0 & -5400.0 & 40.0 & -40.0 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 2.5\,10^{-3} \\ 5.0\,10^3 & -2.0\,10^{-2} \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \\ \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \boxed{\begin{bmatrix} -50.0 & 50.0 & -5.0 & 5.0 \\ 0 & 2.5\,10^5 & 0 & 0 \end{bmatrix}} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \boxed{\begin{bmatrix} 0 & 2.5\,10^{-3} \\ -2.5\,10^5 & 0 \end{bmatrix}} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \end{cases}$$

Sampled-time car suspension models

Let's get a **sampled-time** description of the same dynamic system:

- How does the sampled-time description correlate with the continuous-time model?
- What happens if we increase or decrease the sampling rate? Does the sampled-time model change with the sampling time?
- Does the sampled-time model describe the behaviour of the continuous-time dynamic system for **any possible choice** of the sampling time value?

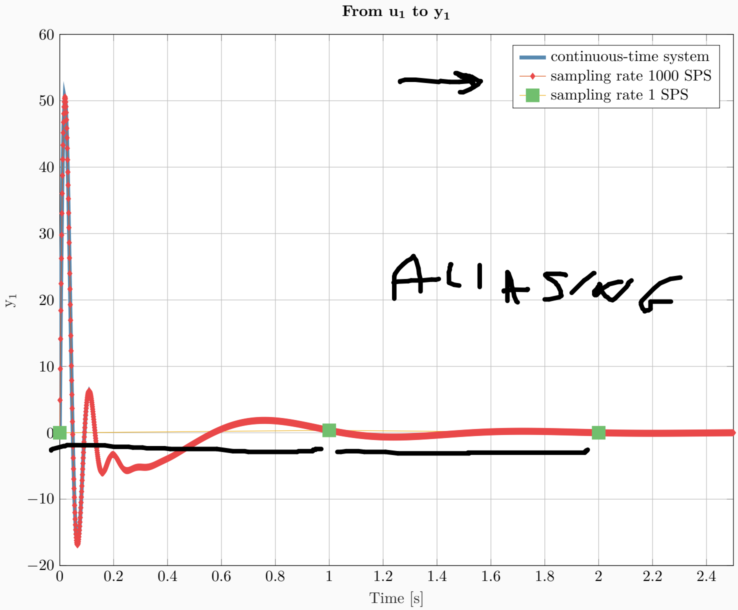
Using *1000 samples per second* as sampling rate

$$
\begin{cases}
\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \\ x_4(k+1) \end{bmatrix} = \begin{bmatrix} 9.98\,10^{-1} & 2.05 \cdot 10^{-5} & 9.98 \cdot 10^{-4} & 2.47 \cdot 10^{-6} \\ 1.97 \cdot 10^{-4} & 0.99 & 1.98 \cdot 10^{-5} & 9.80 \cdot 10^{-4} \\ -4.89 \cdot 10^{-2} & 3.65 \cdot 10^{-3}8 & 9.95 \cdot 10^{-1} & 4.91 \cdot 10^{-3} \\ 3.91 \cdot 10^{-1} & -5.29 & 3.93 \cdot 10^{-2} & 0.96 \end{bmatrix} \cdot \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \end{bmatrix} \\
\qquad + \begin{bmatrix} 4.13 \cdot 10^{-6} & 1.23 \cdot 10^{-9} \\ 2.47 \cdot 10^{-3} & -9.85 \cdot 10^{-9} \\ 1.24 \cdot 10^{-2} & 2.44 \cdot 10^{-6} \\ 4.90 & -1.95 \cdot 10^{-5} \end{bmatrix} \cdot \begin{bmatrix} u_1(k) \\ u_2(k) \end{bmatrix} \\
\begin{bmatrix} y_1(k) \\ y_2(k) \end{bmatrix} = \begin{bmatrix} -50.0 & 50.0 & -5.0 & 5.0 \\ 0 & 2.5 \cdot 10^5 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k \end{bmatrix} \\
\qquad + \begin{bmatrix} 0 & 2.5 \cdot 10^{-3} \\ -2.5 \cdot 10^5 & 0 \end{bmatrix} \cdot \begin{bmatrix} u_1(k) \\ u_2(k) \end{bmatrix}
\end{cases}
$$

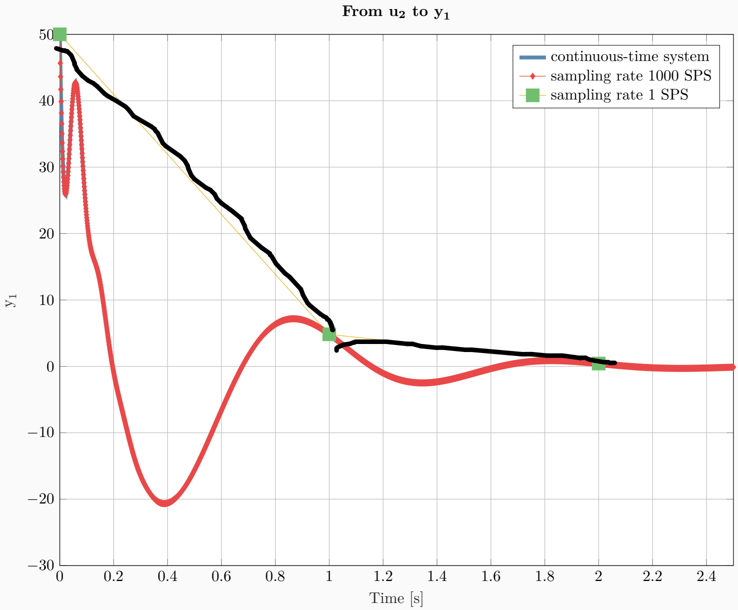
Instead, using *1 sample per second* as sampling rate

$$
\begin{cases}
\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \\ x_4(k+1) \end{bmatrix} = \begin{bmatrix} 1.17 \cdot 10^{-1} & -1.76 \cdot 10^{-2} & 4.65 \cdot 10^{-3} & 1.34 \cdot 10^{-4} \\ 7.75 \cdot 10^{-3} & -4.87 \cdot 10^{-3} & 1.07 \cdot 10^{-3} & 1.29 \cdot 10^{-5} \\ -1.79 \cdot 10^{-1} & -4.90 \cdot 10^{-1} & 9.94 \cdot 10^{-2} & 3.64 \cdot 10^{-4} \\ -4.84 \cdot 10^{-2} & -1.62 \cdot 10^{-2} & 2.91 \cdot 10^{-3} & -2.95 \cdot 10^{-5} \end{bmatrix} \cdot \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \end{bmatrix} \\
\qquad + \begin{bmatrix} 9.00 \cdot 10^{-1} & 4.41 \cdot 10^{-5} \\ 9.97 \cdot 10^{-1} & -3.88 \cdot 10^{-7} \\ 6.70 \cdot 10^{-1} & 8.96 \cdot 10^{-6} \\ 6.46 \cdot 10^{-2} & 2.42 \cdot 10^{-6} \end{bmatrix} \cdot \begin{bmatrix} u_1(k) \\ u_2(k) \end{bmatrix} \\
\begin{bmatrix} y_1(k) \\ y_2(k) \end{bmatrix} = \boxed{\begin{bmatrix} -50.0 & 50.0 & -5.0 & 5.0 \\ 0 & 2.5 \cdot 10^{5} & 0 & 0 \end{bmatrix}} \cdot \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k \end{bmatrix} \\
\qquad \boxed{+ \begin{bmatrix} 0 & 2.5 \cdot 10^{-3} \\ -2.5 \cdot 10^{5} & 0 \end{bmatrix}} \cdot \begin{bmatrix} u_1(k) \\ u_2(k) \end{bmatrix}
\end{cases}
$$

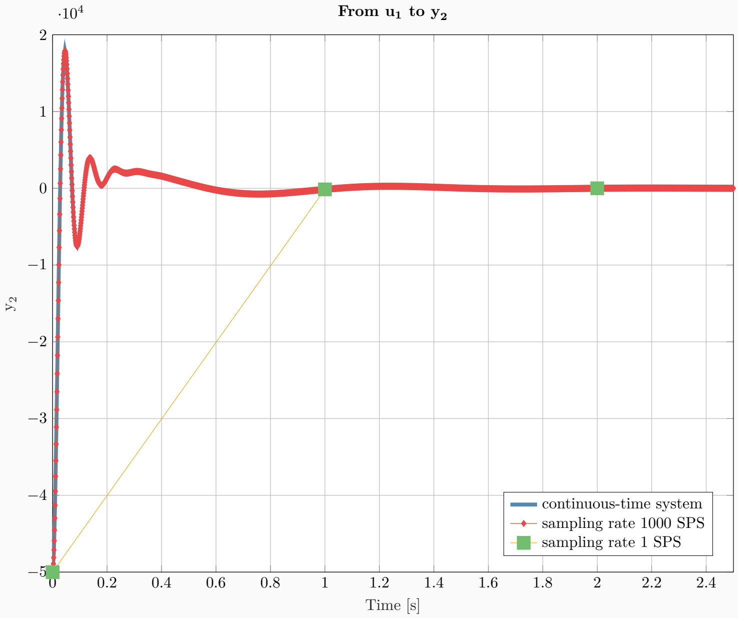
Step responses comparison



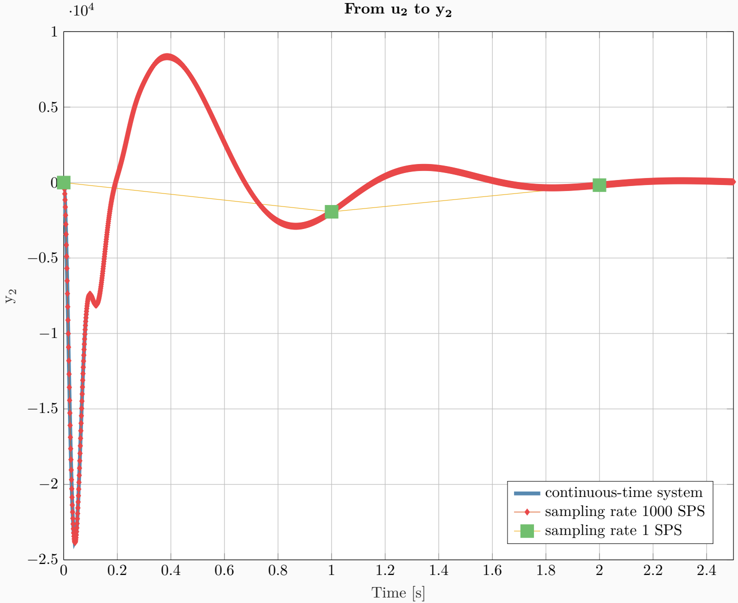
Step responses comparison (cont.)



Step responses comparison (cont.)



Step responses comparison (cont.)



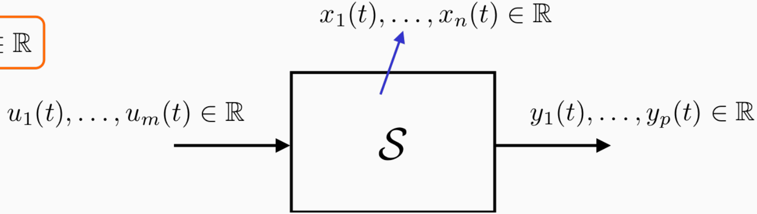
Sampled-time car suspension description (cont.)

Remarks

- by selecting **different sampling rate** we obtained **different representations** of the same continuous-time dynamic system
- **sampling** may **heavily distort the information**, giving a completely wrong discrete-time representation of the original continuous-time system: indeed the model obtained using *one sample per second* as the sampling rate is wrong!

Continuous-time State Equations

$\boxed{\forall t \in \mathbb{R}}$



State equations (dynamic)

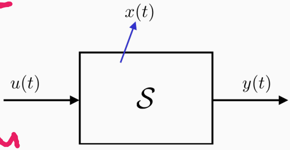
$$\begin{cases} \dot{x}_1(t) = f_1(x_1(t), \ldots, x_n(t), u_1(t), \ldots, u_m(t), t) \\ \quad \vdots \\ \dot{x}_n(t) = f_n(x_1(t), \ldots, x_n(t), u_1(t), \ldots, u_m(t), t) \end{cases}$$

Output equations (algebraic)

$$\begin{cases} y_1(t) = g_1(x_1(t), \ldots, x_n(t), u_1(t), \ldots, u_m(t), t) \\ \quad \vdots \\ y_p(t) = g_p(x_1(t), \ldots, x_n(t), u_1(t), \ldots, u_m(t), t) \end{cases}$$

Continuous-time State Equations (cont.)

$$u(t) = \begin{bmatrix} u_1(t) \\ \vdots \\ u_m(t) \end{bmatrix} \in \mathbb{R}^m, \quad y(t) = \begin{bmatrix} y_1(t) \\ \vdots \\ y_p(t) \end{bmatrix} \in \mathbb{R}^p$$



$$x(t) = \begin{bmatrix} x_1(t) \\ \vdots \\ x_n(t) \end{bmatrix} \in \mathbb{R}^n$$

$n \gg m$

$$f(x,u,t) = \begin{bmatrix} f_1(x,u,t) \\ \vdots \\ f_n(x,u,t) \end{bmatrix} \in \mathbb{R}^n$$

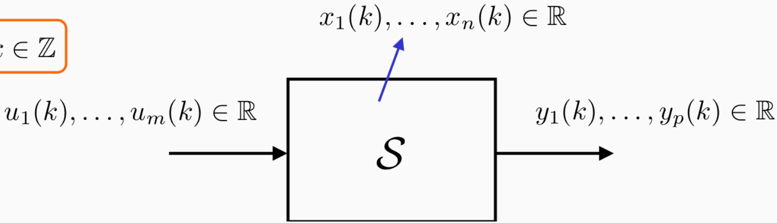
$$f(x,u,t) = \begin{bmatrix} f_1(x,u,t) \\ \vdots \\ f_n(x,u,t) \end{bmatrix} \in \mathbb{R}^n$$

Compact form

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

Discrete-time State Equations

$\boxed{\forall\, k \in \mathbb{Z}}$



State equations (dynamic)

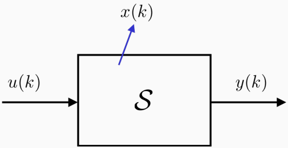
$$\begin{cases} x_1(k+1) = f_1(x_1(k), \ldots, x_n(k), u_1(k), \ldots, u_m(k), k) \\ \quad \vdots \\ x_n(k+1) = f_n(x_1(k), \ldots, x_n(k), u_1(k), \ldots, u_m(k), k) \end{cases}$$

Output equations (algebraic)

$$\begin{cases} y_1(k) = g_1(x_1(k), \ldots, x_n(k), u_1(k), \ldots, u_m(k), k) \\ \quad \vdots \\ y_p(k) = g_p(x_1(k), \ldots, x_n(k), u_1(k), \ldots, u_m(k), k) \end{cases}$$

Discrete-time State Equations (cont.)

$$u(k) = \begin{bmatrix} u_1(k) \\ \vdots \\ u_m(k) \end{bmatrix} \in \mathbb{R}^m, \quad y(k) = \begin{bmatrix} y_1(k) \\ \vdots \\ y_p(k) \end{bmatrix} \in \mathbb{R}^p$$



$$x(k) = \begin{bmatrix} x_1(k) \\ \vdots \\ x_n(k) \end{bmatrix} \in \mathbb{R}^n$$

$$f(x,u,k) = \begin{bmatrix} f_1(x,u,k) \\ \vdots \\ f_n(x,u,k) \end{bmatrix} \in \mathbb{R}^n$$

$$f(x,u,k) = \begin{bmatrix} f_1(x,u,k) \\ \vdots \\ f_n(x,u,k) \end{bmatrix} \in \mathbb{R}^n$$

Compact form

$$\begin{cases} x(k+1) = f(x(k), u(k), k) \\ y(k) = g(x(k), u(k), k) \end{cases}$$

More Definitions and Properties

- **Time-invariant Dynamic Systems**

$$\begin{cases} \dot{x}(t) = f(x(t), u(t), \cancel{t}) \\ y(t) = g(x(t), u(t), \cancel{t}) \end{cases} \Longrightarrow \begin{cases} \dot{x}(t) = f(x(t), u(t)) \\ y(t) = g(x(t), u(t)) \end{cases}$$

$$\begin{cases} x(k+1) = f(x(k), u(k), \cancel{k}) \\ y(k) = g(x(k), u(k), \cancel{k}) \end{cases} \Longrightarrow \begin{cases} x(k+1) = f(x(k), u(k)) \\ y(k) = g(x(k), u(k)) \end{cases}$$

- **Strictly Proper Dynamic Systems**

$$\begin{cases} \dot{x}(t) = f(x(t), u(t), \cancel{t}) \\ y(t) = g(x(t), \cancel{u(t)}, \cancel{t}) \end{cases} \Longrightarrow \begin{cases} \dot{x}(t) = f(x(t), u(t), t) \\ y(t) = g(x(t), t) \end{cases}$$

$$\begin{cases} x(k+1) = f(x(k), u(k), \cancel{k}) \\ y(k) = g(x(k), \cancel{u(k)}, \cancel{k}) \end{cases} \Longrightarrow \begin{cases} x(k+1) = f(x(k), u(k), k) \\ y(k) = g(x(k), k) \end{cases}$$

così anche
Time-invariant

More Definitions and Properties (cont.)

- **Forced and Free Dynamic Systems**

Solo C. I.

$$\begin{cases} \dot{x}(t) = f(x(t), \cancel{u(t)}, t) \\ y(t) = g(x(t), \cancel{u(t)}, t) \end{cases} \Longrightarrow \begin{cases} \dot{x}(t) = f(x(t), t) \\ y(t) = g(x(t), t) \end{cases}$$

$$\begin{cases} x(k+1) = f(x(k), \cancel{u(k)}, k) \\ y(k) = g(x(k), \cancel{u(k)}, k) \end{cases} \Longrightarrow \begin{cases} x(k+1) = f(x(k), k) \\ y(k) = g(x(k), k) \end{cases}$$

It is worth noting that in case the input function $u(t)$, $\forall\, t$ or input sequence $u(k)$, $\forall\, k$ are **known beforehand**, the dynamic system can be re-written as a free one:

$$\begin{cases} \dot{x}(t) = f(x(t), u(t), t) = \tilde{f}(x(t), t) \\ y(t) = g(x(t), u(t), t) = \tilde{g}(x(t), t) \end{cases}$$

$$\begin{cases} x(k+1) = f(x(k), u(k), k) = \tilde{f}(x(k), k) \\ y(k) = g(x(k), u(k), k) = \tilde{g}(x(k), k) \end{cases}$$