# Implementation of a Generic Dynamic System

Corrado Santoro

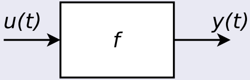
**ARSLAB - Autonomous and Robotic Systems Laboratory**
Dipartimento di Matematica e Informatica - Università di Catania, Italy
`santoro@dmi.unict.it`

Robotic Systems

# A System

Each **real system** (any object belonging to the real world) is an object with finite number of freedom degrees that evolves in time according to a certain law

A real system can be represented as a **black box** that can be externally stimulated from an input (that we call $u(t)$), producing a certan **effect** which is the output (that we call $y(t)$)




$$y(t) = f(u(t))$$

The behaviour is **totally represented** by function $f(\cdot)$

# Definition of **Dynamic System**

A **dynamic system** is a (physical) system where, given a certain time instant $t$, the output $y(t)$ depends on the current value $u(t)$ and the past of $u(t)$ and $y(t)$

A dynamic system (**time-continuous**) is described by a **differential equation** in the time domain:

$$y = f(\dot{y}, \ddot{y}, ..., u, \dot{u}, \ddot{u}, t)$$

# Definition of **Dynamic System**

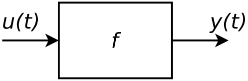
## Characteristics

A dynamic system (**time-continuous**) is described by a **differential equation** in the time domain:

$$y = f(\dot{y}, \ddot{y}, ..., u, \dot{u}, \ddot{u}, t)$$

- The **order** of a system is the highest derivative that appears
- A **linear system** is a system described by a linear differential equation
- If the coefficients are constants (not dependent on the time) the system is called **time-invariant**

# Dynamic System Implementation




Let us consider

$$\dot{y} + 3y = 5u$$

We can implement it by discretisation:

$$\frac{y(t+\Delta T) - y(t)}{\Delta T} + 3y(t) = 5u(t)$$

then

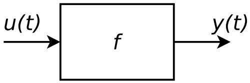
$$y(t+\Delta T) = y(t)(1 - 3\Delta T) + 5\Delta T\, u(t)$$

# Dynamic System Implementation

$$y(t + \Delta T) = y(t)(1 - 3\Delta T) + 5\,\Delta T\;u(t)$$

```
class MySystem:

    def __init__(self):
        self.y = 0

    def evaluate(self, delta_t, _input):
        self.y = self.y * (1 - 3 * delta_t) + 5 * delta_t * _input
        return self.y
```

# Dynamic System Implementation




Let us consider

$$\ddot{y} + 3\dot{y} + y = 5u$$

It's a second order system, how can be implemented it?

Using the incremental ratio of the second derivative is hard.... but we can use a "trick"

# Dynamic System Implementation

$$\ddot{y} + 3\dot{y} + y = 5u$$

The aim is to remove the second derivative and have variables that are **derivated only one time**

Let us introduce **two new variables $x_1$ and $x_2$** and assume:

$$\begin{aligned} x_1 &= y \\ x_2 &= \dot{y} = \dot{x}_1 \\ \dot{x}_2 &= \ddot{y} \end{aligned}$$

We obtain the following **equation system**:

$$\begin{cases} \dot{x}_2 + 3x_2 + x_1 = 5u \\ \dot{x}_1 = x_2 \end{cases}$$

# Dynamic System Implementation

$$\begin{cases} \dot{x}_2 + 3x_2 + x_1 & = & 5u \\ \dot{x}_1 & = & x_2 \end{cases}$$

Let's reorder:

$$\begin{cases} \dot{x}_1 & = & x_2 \\ \dot{x}_2 & = & -3x_2 - x_1 + 5u \end{cases}$$

Let's discretise:

$$\begin{cases} \frac{x_1(t+\Delta T)-x_1(t)}{\Delta T} & = & x_2(t) \\ \frac{x_2(t+\Delta T)-x_2(t)}{\Delta T} & = & -3x_2(t) - x_1(t) + 5u(t) \end{cases}$$

And reorder:

$$\begin{cases} x_1(t+\Delta T) & = & x_1(t) + x_2(t)\Delta T \\ x_2(t+\Delta T) & = & x_2(t) - 3\Delta T x_2(t) - \Delta T x_1(t) + 5\Delta T u(t) \end{cases}$$

# Dynamic System Implementation

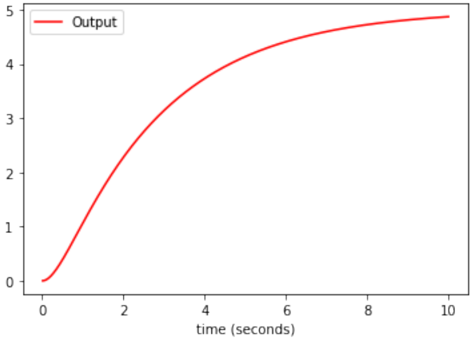
$$\begin{cases} x_1(t+\Delta T) & = & x_1(t) + x_2(t)\Delta T \\ x_2(t+\Delta T) & = & x_2(t) - 3\Delta T x_2(t) - \Delta T x_1(t) + 5\Delta T u(t) \end{cases}$$

```
class SecondOrderSystem:

    def __init__(self):
        self.x1 = 0
        self.x2 = 0

    def evaluate(self, delta_t, _input):
        new_x1 = self.x1 + delta_t * self.x2
        new_x2 = (1 - 3 * delta_t) * self.x2 - delta_t * self.x1 +
                 5 * delta_t * _input

        self.x1 = new_x1
        self.x2 = new_x2

        return self.x1
```

(examples/systems/second_order.ipynb)

# Dynamic System Implementation

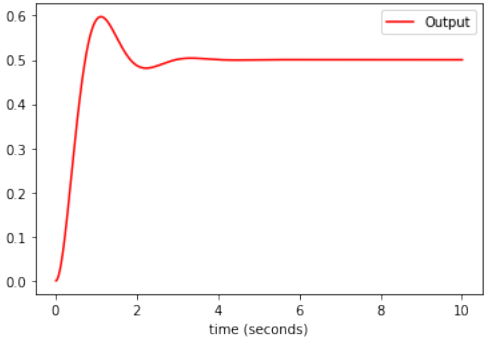
$$\ddot{y} + 3\dot{y} + y = 5u$$




(examples/systems/second_order.ipynb)

# Dynamic System Implementation

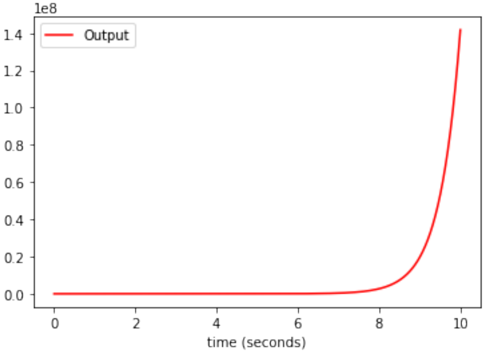
$$\ddot{y} + 3\dot{y} + 10y = 5u$$




(examples/systems/second_order_2.ipynb)

# Dynamic System Implementation

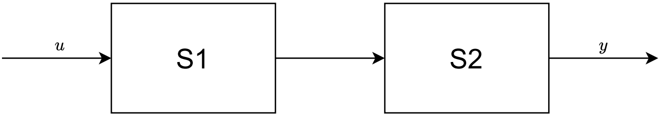
$$\ddot{y} + 3\dot{y} - 10y = 5u$$




(examples/systems/second_order_3.ipynb)

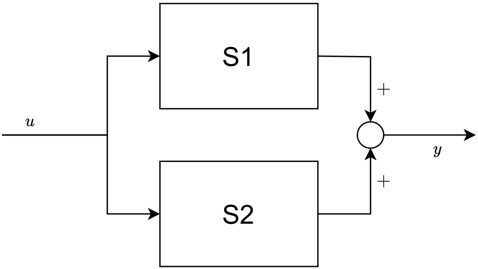
# Exercises

# Exercises




$$S1 : \ddot{y} + 4\dot{y} + 2y = 5u$$

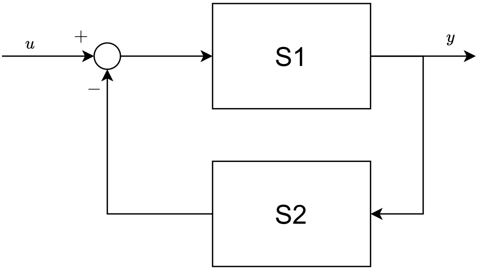
$$S2 : \dot{y} + 0.5y = u$$




$$S1 : \ddot{y} + 4\dot{y} + 2y = 5u$$

$$S2 : \dot{y} + 0.5y = u$$

# Exercises




$$S1 : \ddot{y} + 4\dot{y} + 2y = 5u$$

$$S2 : \dot{y} + 0.5y = u$$

# Implementation of a Generic Dynamic System

Corrado Santoro

**ARSLAB - Autonomous and Robotic Systems Laboratory**
Dipartimento di Matematica e Informatica - Università di Catania, Italy
`santoro@dmi.unict.it`

Robotic Systems