Data-driven model learning of dynamical systems

-

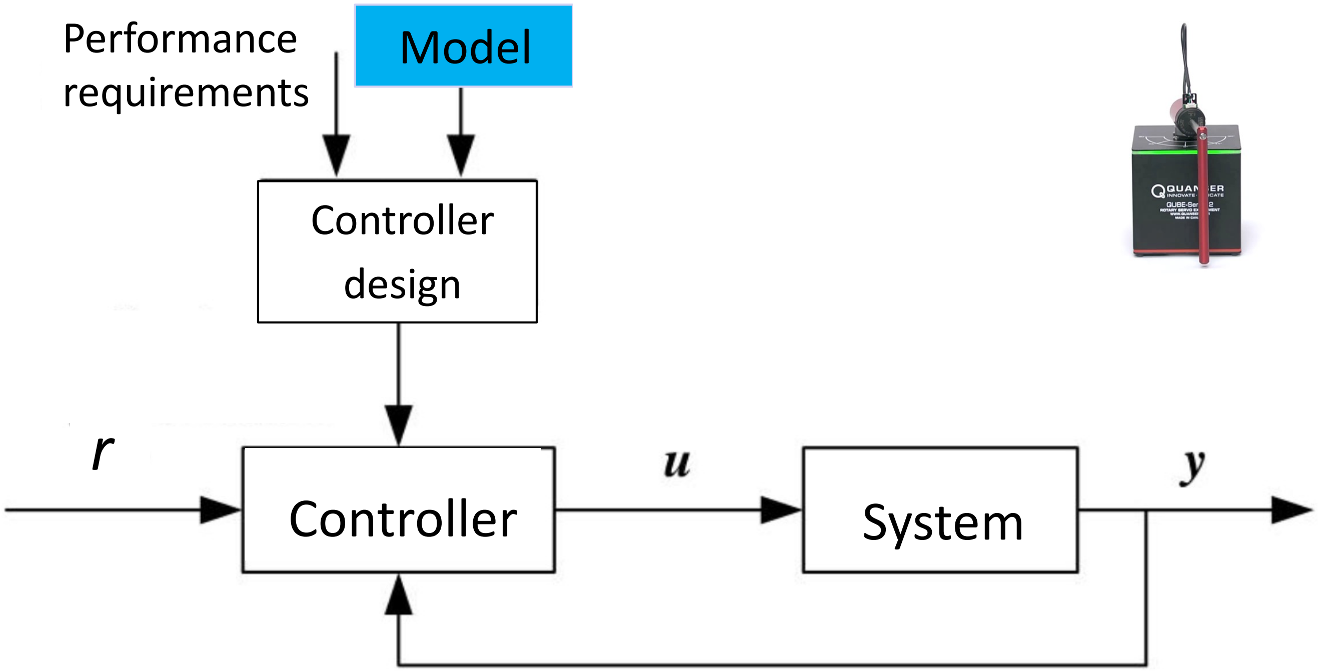
A case study with the QUBE servo 2

Hugues GARNIER

hugues.garnier@univ-lorraine.fr

Model identification for control
What is it all about?

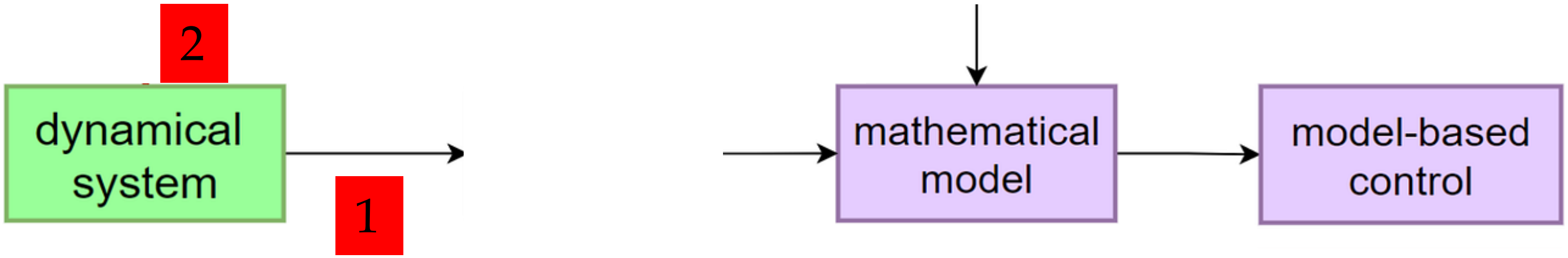
- Control is an interdisciplinary branch of engineering and mathematics that deals with dynamical systems with inputs, and how their behavior is modified by feedback



*To do it in a **systematic way**, a **model-based approach** for control design has been developed*

H. Garnier

The two main ways to determine a mathematical model for control design



Model-based control design
Case study: Rotary speed control of the QUBE servo 2

(a) QUBE-Servo 2 with Inertia Disc Module

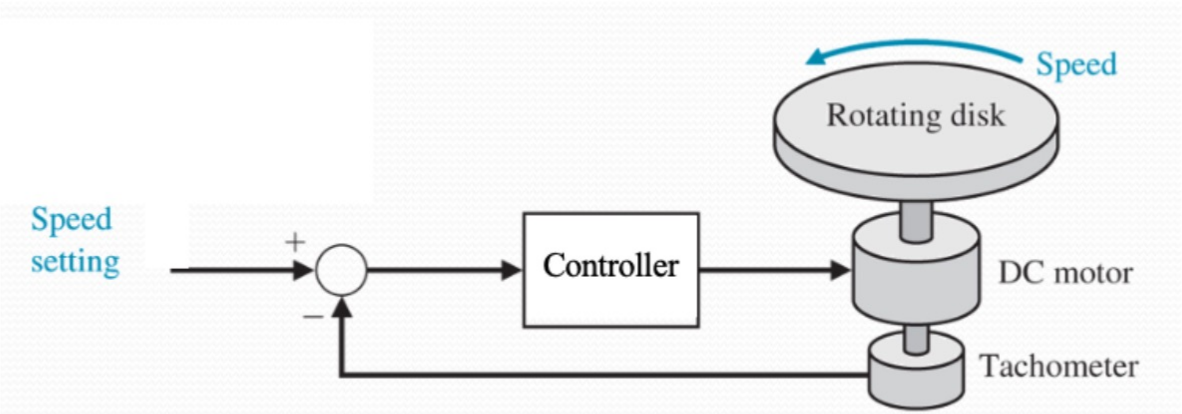
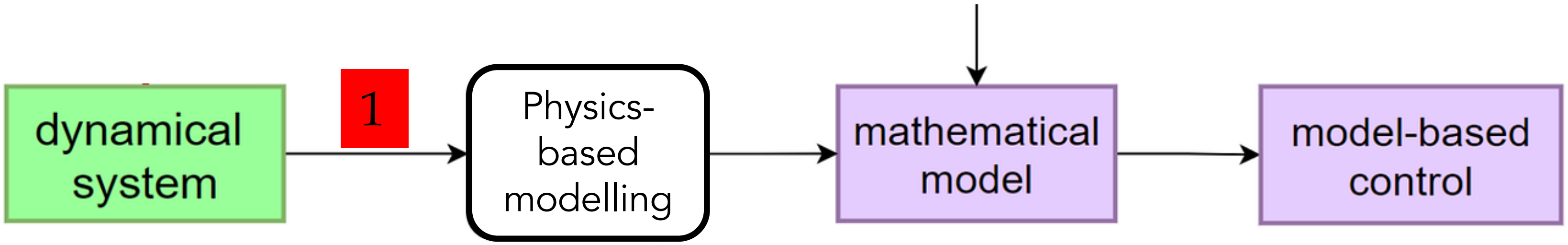
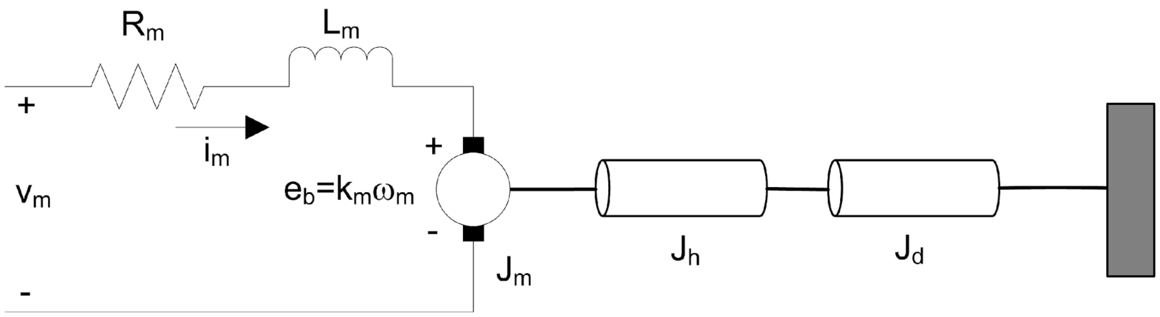


Figure 1.2: Schematic of the feedback speed control of the rotating disk

First way to determine a mathematical model: Physics-based modelling



Example: Physics-based modelling of the QUBE servo 2



(a) QUBE-Servo 2 with Inertia Disc Module

The equations of motion are:

$$R_m i_m(t) + L_m \frac{di_m(t)}{dt} = v_m(t) - k_m \dot{\theta}_m(t)$$

$$J_{eq}\ddot{\theta}_m(t) + b\dot{\theta}_m(t) = k_t i_m(t)$$

$$J_{eq} = J_m + J_h + J_d$$
$$= J_m + \frac{1}{2} m_h r_h^2 + \frac{1}{2} m_d r_d^2$$

Simplifying assumptions:

- Since the motor inductance L_m is much less than its resistance, it can be ignored
- The viscous coefficient b is assumed very small, it can be ignored
- Static friction effects neglected

Input voltage-to angular velocity transfer function

$V_m(s)$ →  → $\Omega_m(s)$

(a) QUBE-Servo 2 with Inertia Disc Module

$$\frac{\Omega_m(s)}{V_m(s)} = G(s) = \frac{k_t}{J_{eq}R_m s + k_t k_m}$$

$$G(s) = \frac{\frac{1}{k_m}}{1 + \frac{J_{eq}R_m}{k_t k_m} s}$$

$$G(s) = \frac{K}{Ts+1}$$

$$K = \frac{1}{k_m}; \quad T = \frac{J_{eq}R_m}{k_t k_m}$$

Example: Physical modelling of the QUBE servo 2

Physical parameters need to be determined from mechanical and electrical datasheet of the QUBE servo 2

Symbol	Description	Value
DC Motor		
R_m	Terminal resistance	8.4Ω
k_t	Torque constant	0.042 N.m/A
k_m	Motor back-emf constant	0.042 V/(rad/s)
J_m	Rotor inertia	4.0×10^{-6} kg.m^2
L_m	Rotor inductance	1.16 mH
m_h	Load hub mass	0.0106 kg
r_h	Load hub mass	0.0111 m
J_h	Load hub inertia	0.6×10^{-6} kg.m^2
Load Disk		
m_d	Mass of disk load	0.053 kg
r_d	Radius of disk load	0.0248 m

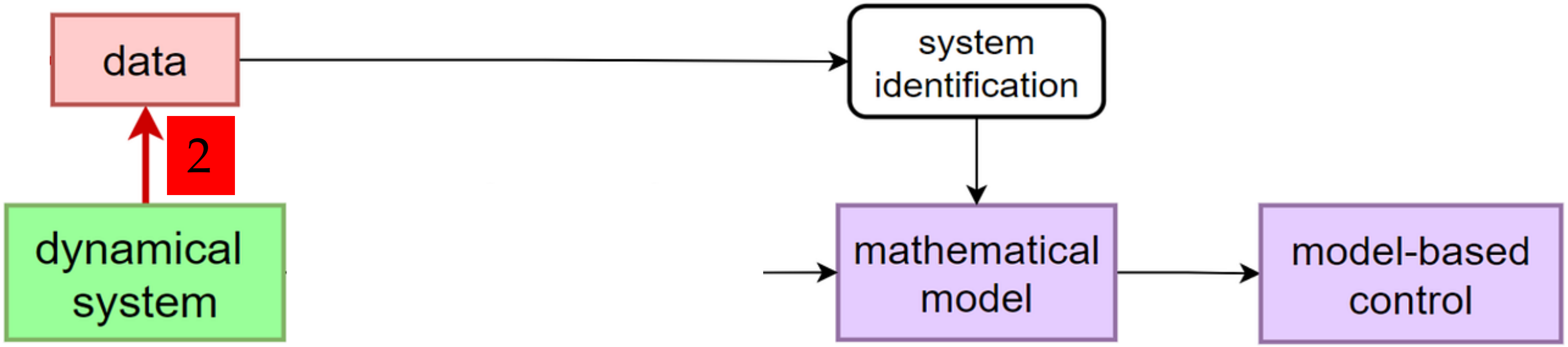
Table 1.1: QUBE-Servo 2 system parameters

(a) QUBE-Servo 2 with Inertia Disc Module

$$J_{eq} = J_m + J_h + J_d$$
$$= J_m + \frac{1}{2} m_h r_h^2 + \frac{1}{2} m_d r_d^2$$

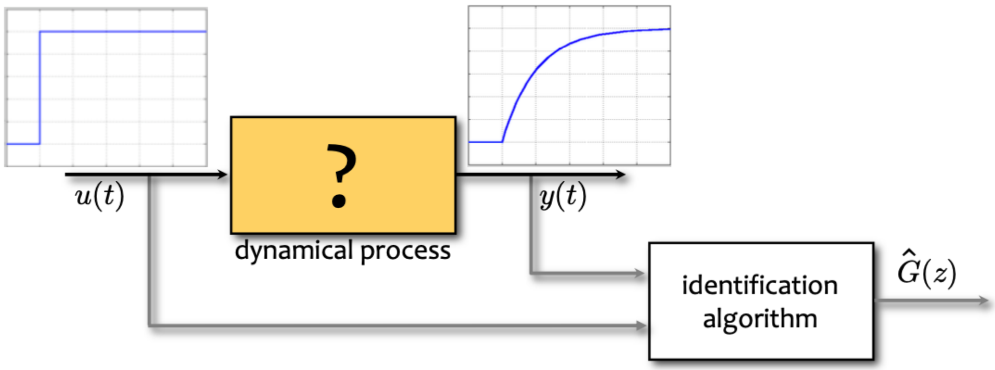
$$G(s) = \frac{\frac{1}{k_m}}{1 + \frac{J_{eq} R_m}{k_t k_m} s} = \frac{23.81}{1 + 0.0994 s}$$

Second way to determine a mathematical model: Data-driven system identification



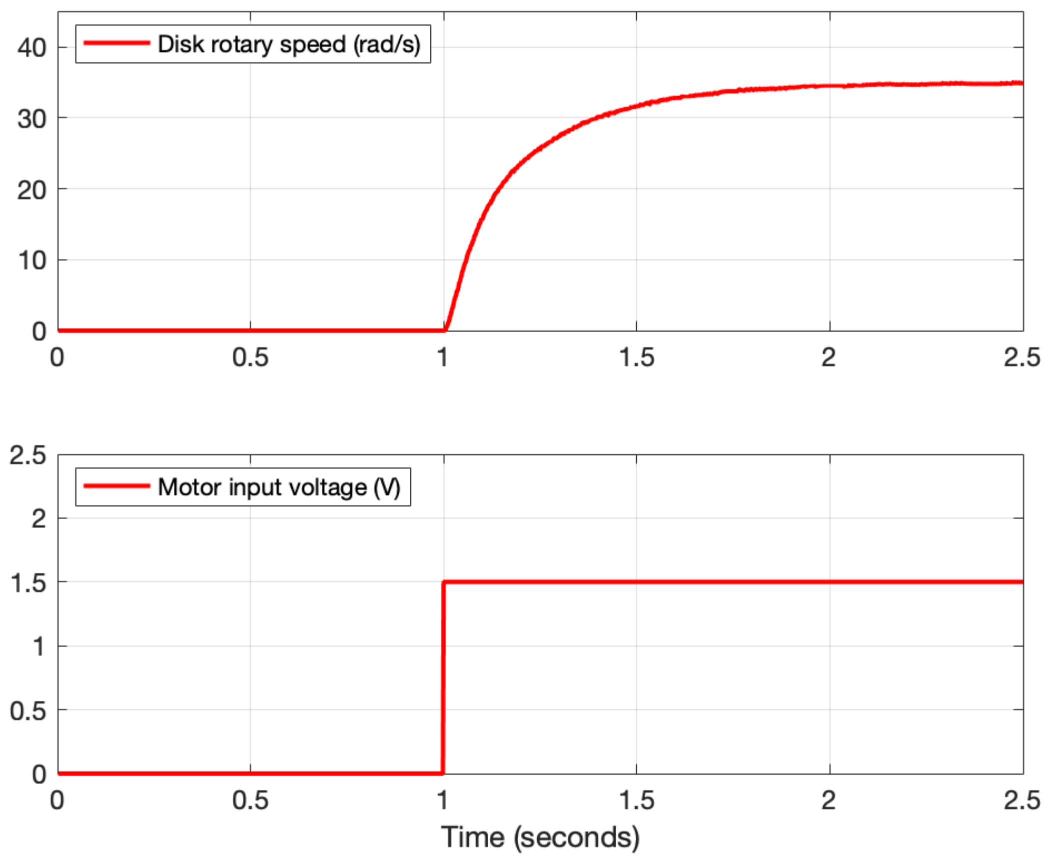
H. Garnier

Basic system identification method
Step response-based model identification



- Excite the process with a step $u(t)$, record output response $y(t)$
- Observe the shape of $y(t)$
  (1st-order response ? 2nd-order undamped response ? Any delay ? ...)

Example: Step response identification of the QUBE servo 2

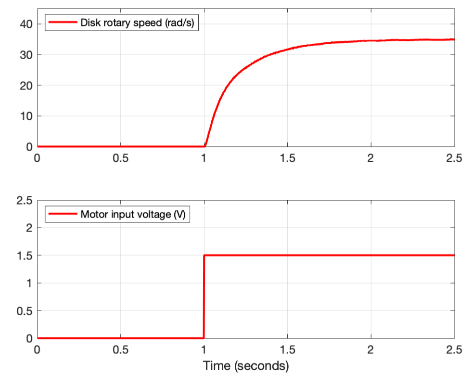


(a) QUBE-Servo 2 with Inertia Disc Module

Example: Step response identification of the QUBE servo 2 by using the CONTSID toolbox

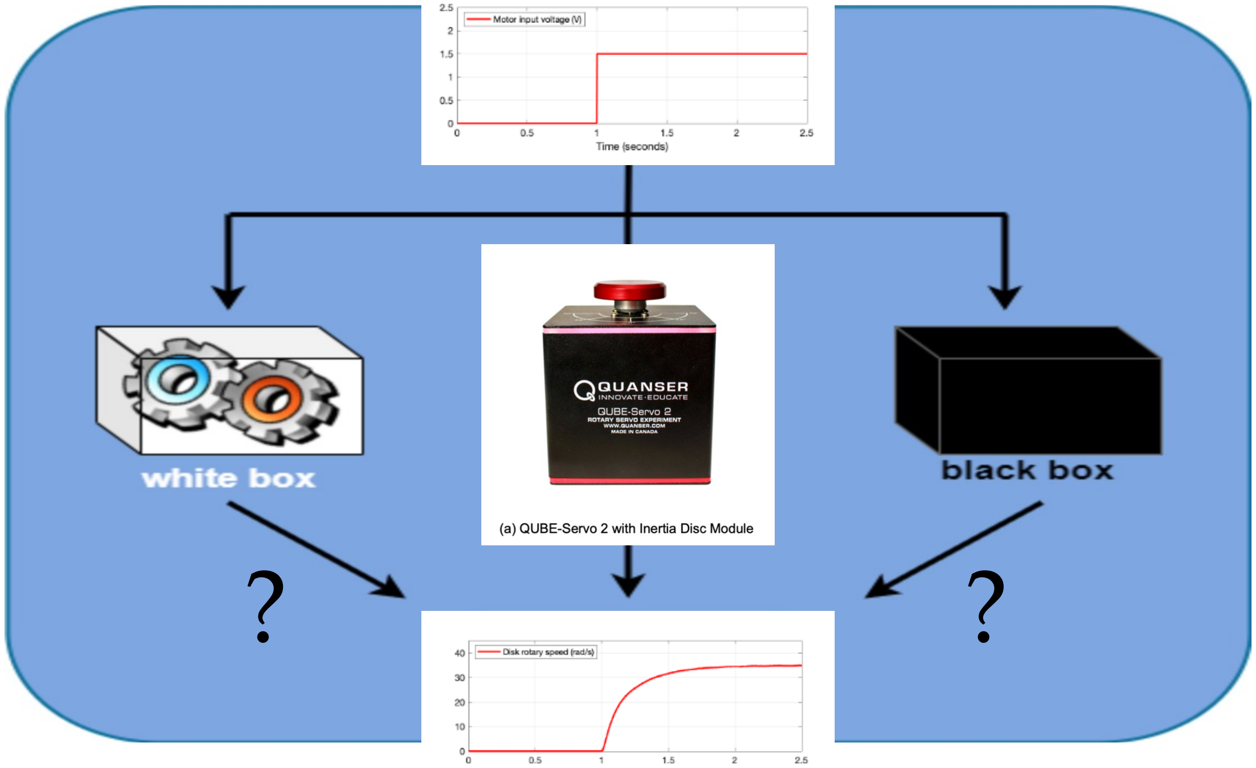
```
load data_step_Qube_speed
t=speed_data(1,:)'; % time-instants
y=speed_data(2,:)'; % rotary speed in rad/sec
u=speed_data(3,:)'; % motor input voltage in V
Ts=t(2)-t(1);       % Sampling period in sec

data=iddata(y,u,Ts);
idplot(data)

Model_type=idproc('P1'); % Simple first-order model
M=procsrivc(data,Model_type)
Process model with transfer function:
             Kp
  G(s) = ----------
          1+Tp1*s

       Kp = 22.98
      Tp1 = 0.18072
```

(a) QUBE-Servo 2 with Inertia Disc Module

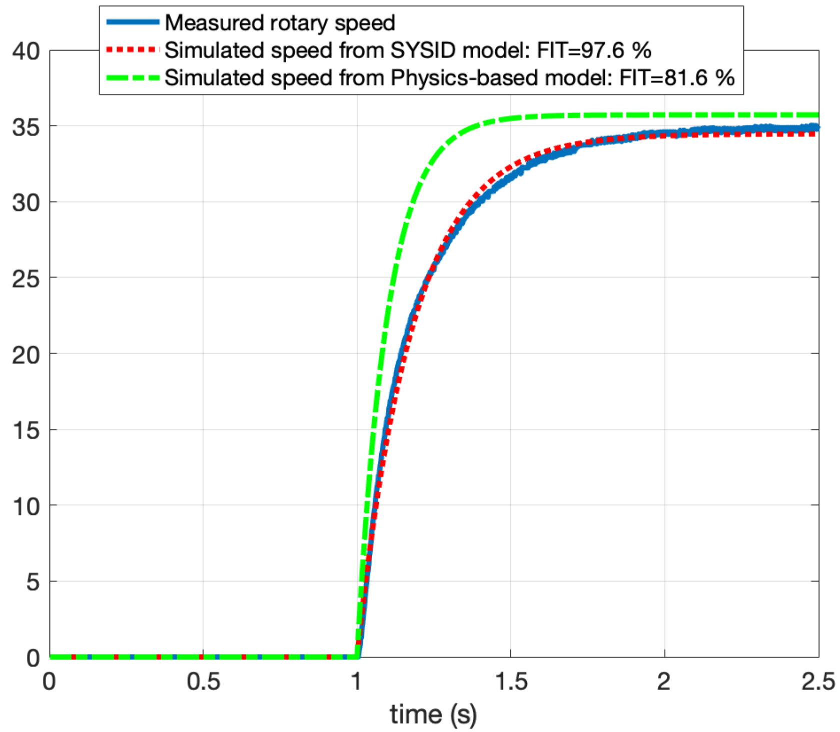


Model quality assessment

Let us compare the measured and model outputs for the step input



Comparison of the physics-based and identified models



(a) QUBE-Servo 2 with Inertia Disc Module

The fit is better for the System identification (SYSID) black-box model

Why is there a discrepancy for the Physics-based white-box model? Possible reasons:

- Physical parameters have tolerances and may be slightly off
- Un-modeled effects such as motor inductance,…

The two models capture quite well the main dynamic of the QUBE servo 2 system

Any of the two models could be used to design a PI control with good performance