

MATLAB Simulink - PID

Kun-Yen Chiu

Problem 1

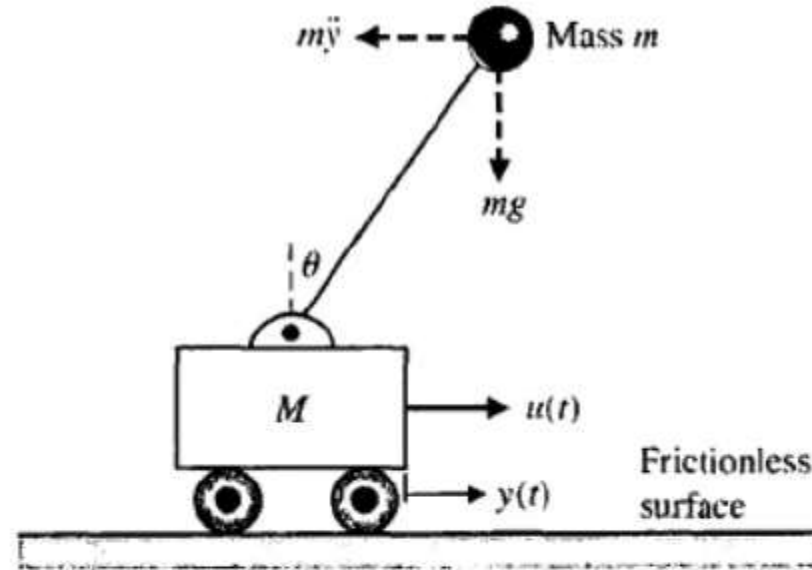


Figure 1 A cart and an inverted pendulum.

Force (horizontal):

$$(M + m)\ddot{y}(t) = u(t) - (mL\ddot{\theta}(t) \cos \theta(t) - mL\dot{\theta}(t)^2 \sin \theta(t))$$

Torque:

$$mL^2\ddot{\theta}(t) = mgL \sin \theta(t) - mL\ddot{y}(t) \cos \theta(t)$$

Assume that $M=100\text{Kg}$, $m=10\text{Kg}$, $L=1\text{m}$, $g=9.81\text{m/s}^2$ and $G_c(s) = K_P + \frac{K_I}{s} + sK_D$,

*This is a guideline for Problem 1.

*The equations are not exactly the same, but same concepts

1) Use ode45 to simulate the output response in time-domain.

$$\ddot{y} = -\frac{mg}{M}\theta + \frac{1}{M}u$$

$$\ddot{\theta} = \frac{(M+m)g}{ML}\theta - \frac{1}{ML}u$$

$$x_1 = y, x_2 = \dot{y}, x_3 = \int \theta dt, x_4 = \theta, x_5 = \dot{\theta}$$

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = -\frac{mg}{M}x_4 + \frac{1}{M}u$$

$$\dot{x}_3 = x_4$$

$$\dot{x}_4 = x_5$$

$$\dot{x}_5 = \frac{(M+m)g}{ML}x_4 - \frac{1}{ML}u$$

$$\text{PID control } U(s) = G_c(s)(R(s) - \theta(s))$$

$$u(t) = -K_P\theta(t) - K_I \int \theta(t)dt - K_D \frac{d\theta(t)}{dt} = -K_P x_4 - K_I x_3 - K_D x_5$$

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = -\frac{mg}{M}x_4 + \frac{1}{M}(-K_P x_4 - K_I x_3 - K_D x_5)$$

$$\dot{x}_3 = x_4$$

$$\dot{x}_4 = x_5$$

$$\dot{x}_5 = \frac{(M+m)g}{ML}x_4 - \frac{1}{ML}(-K_P x_4 - K_I x_3 - K_D x_5)$$

$$x_{10} = 0, x_{20} = 0, x_{30} = 0, x_{40} = 1, x_{50} = 0$$

q(s) can be obtained by (sl-A).

$$q(s) = MLs^3 - K_D s^2 - [(M+m)g + K_P]s - K_I$$

Problem 1 - Reference Code

```
clear all
clc
% Parameters:
m=10;
M=100;
L=1;
g=9.81;

% PID tuning
KP = ???
KI = ???
KD = ???

% Define reference signal
r = 0;
fun = @(t,x) [x(2); -m*g*x(4)/M; x(4); x(5); (M+m)*g*x(4)/(M*L)] + (-KP*x(4)-KI*x(3)-KD*x(5))*[0; 1/M; 0; 0; -1/(M*L)];
T = linspace(0, 15, 1e3);
x0 = [0; 0; 0; 1; 0];
[t, Y] = ode45(@(t,x) fun(t,x), T, x0);

figure
%%%%%%%%plot Yourself%%%%%%%%
xlabel('Time [s]')
ylabel('Angular rotation [rad]')

syms x;
digits(6)
eqn = M*L*x^3-KD*x^2-((M+m)*g+KP)*x-KI == 0;
s = solve(eqn,x);
vpa(s)
```

Problem 2

2) Use Simulink to simulate the output response in s-domain.

Assume that $\theta(t) \rightarrow 0^\circ \Rightarrow \sin \theta(t) \approx \theta(t), \cos \theta(t) \approx 1, \dot{\theta}(t) \approx 0,$

$$(M + m)\ddot{y}(t) + mL\ddot{\theta}(t) = u(t)$$

$$\ddot{y}(t) + L\ddot{\theta}(t) - g\theta(t) = 0$$

$$G(s) = \frac{-1/ML}{s^2 - \frac{(M+m)g}{ML}}$$

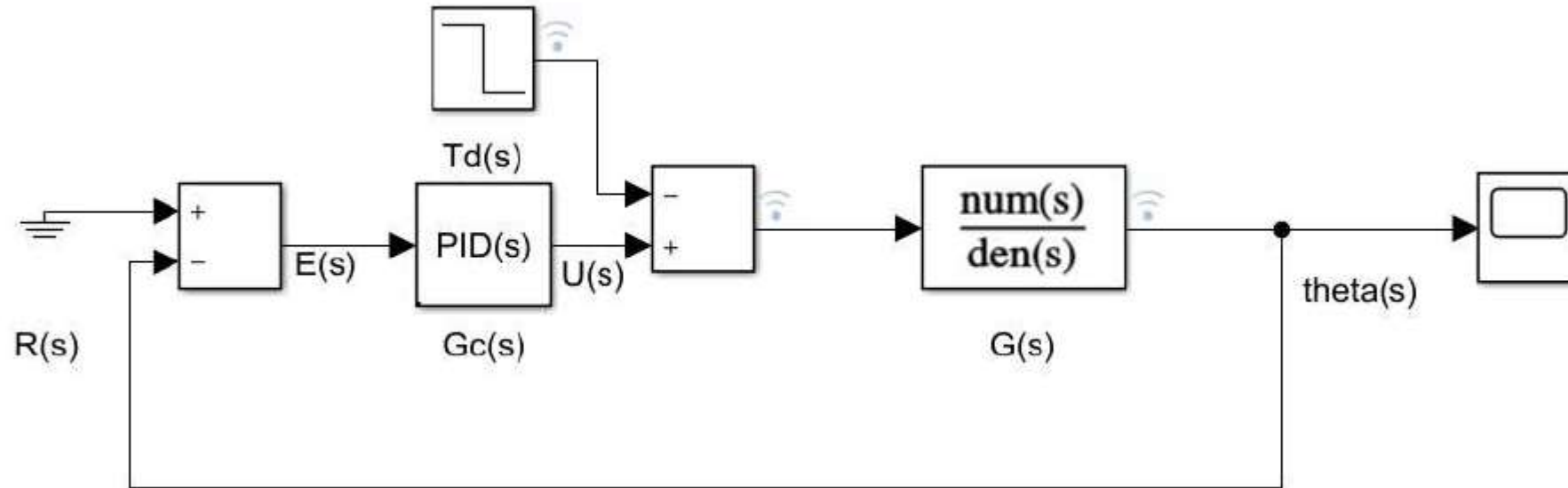
Applying PID control to the inverted pendulum.

Assume that $T_d(s) = 1(T_d(t) = \delta(t))$

Create Block Diagram

You will need the following function blocks:

- > PID Controller
- > Add
- > Ground
- > Step
- > Transfer Fcn
- > Scope (see output result)



Set parameters

Block Parameters: PID Controller

PID 1dof (mask) (link)

This block implements continuous- and discrete-time PID control algorithms and includes advanced features such as anti-windup, external reset, and signal tracking. You can tune the PID gains automatically using the 'Tune...' button (requires Simulink Control Design).

Controller: PID Form: Parallel

Time domain:

☒ Continuous-time

☐ Discrete-time

Discrete-time settings

Sample time (-1 for inherited): -1

Compensator formula

$$P + I \frac{1}{s} + D \frac{N}{1 + N \frac{1}{s}}$$

Main Initialization Output Saturation Data Types State Attributes

Controller parameters

Source: internal

Proportional (P): KP

Integral (I): KI

Derivative (D): KD

☒ Use filtered derivative

Filter coefficient (N): 1000

Automated tuning

Select tuning method: Transfer Function Based (PID Tuner App) Tune...

☒ Enable zero-crossing detection

OK Cancel Help Apply

Block Parameters: Transfer Fcn

Transfer Fcn

The numerator coefficient can be a vector or matrix expression. The denominator coefficient must be a vector. The output width equals the number of rows in the numerator coefficient. You should specify the coefficients in descending order of powers of s.

Parameters

Numerator coefficients:

[-1/(M*L)]

Denominator coefficients:

[1 0 -(M+m)*g/(M*L)]

Absolute tolerance:

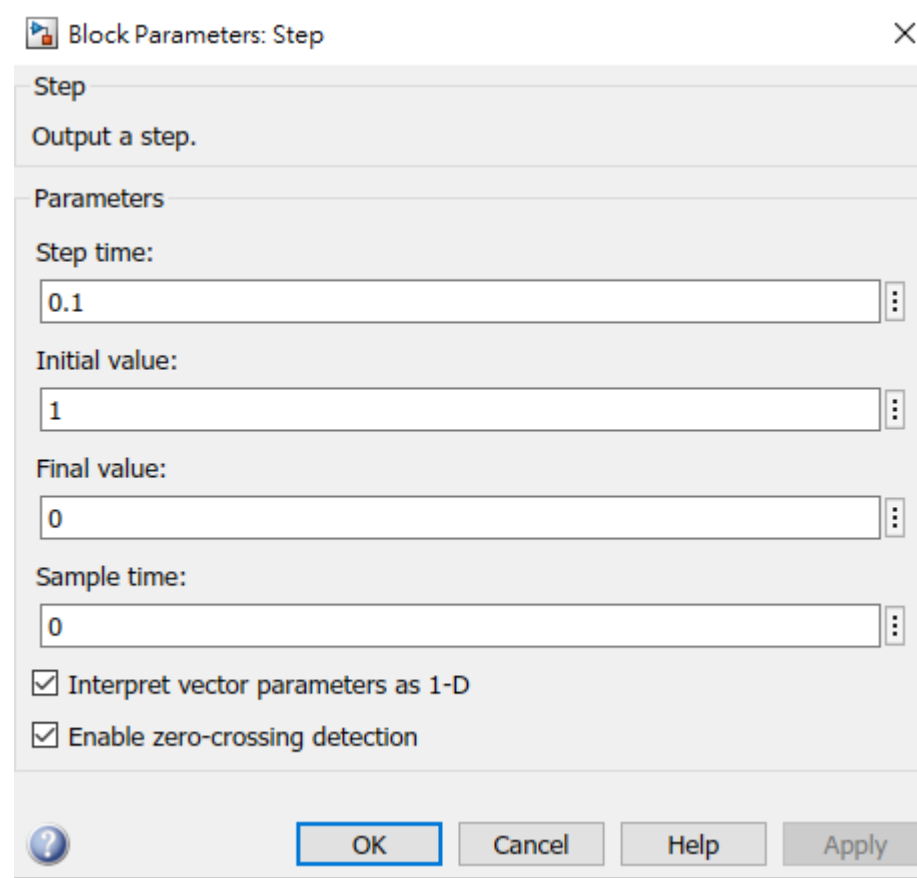
auto

State Name: (e.g., 'position')

'Angular_rotation'

OK Cancel Help Apply

Set Parameters (Cont.)



A screenshot of the 'Block Parameters: Step' dialog box. The dialog has a title bar with a question mark icon, the text 'Block Parameters: Step', and a close button. The main area is divided into sections. The first section is titled 'Step' and contains the text 'Output a step.'. The second section is titled 'Parameters' and contains four input fields: 'Step time:' with value '0.1', 'Initial value:' with value '1', 'Final value:' with value '0', and 'Sample time:' with value '0'. Each input field has a small icon with three vertical dots to its right. Below these fields are two checked checkboxes: 'Interpret vector parameters as 1-D' and 'Enable zero-crossing detection'. At the bottom of the dialog are four buttons: a help button (question mark icon), 'OK', 'Cancel', and 'Apply'.

Block Parameters: Step

Step

Output a step.

Parameters

Step time:

0.1

Initial value:

1

Final value:

0

Sample time:

0

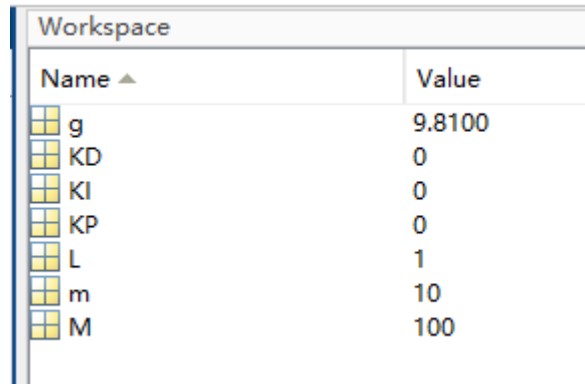
☒ Interpret vector parameters as 1-D

☒ Enable zero-crossing detection

OK Cancel Help Apply

How to run Simulink with matlab parameter

- In parameter file, choose the parameter K_D , K_I , and K_P
- Next, Run the parameter.m file, in order to save the data in workspace



The screenshot shows the MATLAB Workspace window with a table of variables and their values. The variables are g, KD, KI, KP, L, m, and M. The values are 9.8100, 0, 0, 0, 1, 10, and 100 respectively. Each variable name has a small yellow icon to its left.

Name ▲	Value
g	9.8100
KD	0
KI	0
KP	0
L	1
m	10
M	100

- After run the parameter.m, you can press run in Simulink