ME 421L
Lab 10: Single Pendulum Gantry (SPG)

Pre-Lab:

1- How many actuators and sensors in the Single Pendulum Gantry (SPG)?
2- Linearize the Equations of Motions (EOM) of the system. (Eqns: B-17 & B-18 p. 33)
Use the hint in page 6 of your handout.
3- The SPG system needs to be represented in the state space form, i.e.

$$\frac{\partial}{\partial t} X = A X + B U$$ ….. Input equation

$$Y = C X + \mathrm{D}\, U$$ ……. Output equation

Where
$$X^T = \left[x_c(t),\ \alpha(t),\ \frac{d}{dt} x_c(t),\ \frac{d}{dt}\alpha(t) \right]$$

The objective is to control the displacement of the pendulum tip along the x-axis, x_t.

$x_t = x_c + L_p \sin(\alpha)$ è vector C

4- Explain why it is a SISO?

MATLAB:

5- start a MATLAB script file and save it as LastName1_LastNam2_SPG.m in your section folder

6-In the MATLAB file, define the following matrices for the SPG open loop State Space representation:

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1.5216 & -11.6513 & 0.0049 \\ 0 & -26.1093 & 26.8458 & -0.0841 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 0 \\ 0 \\ 1.5304 \\ -3.5261 \end{bmatrix}$$

$$C = [1,\ L_p,\ 0,\ 0] \quad \text{and} \quad \mathrm{D} = [0]$$

Where Lp is 25.25 inches *** 25.4/1000** = ……. meter

7- Define your Open Loop System from the provided State Space matrices using the MATLAB command ***ss***:

sys_OL=ss(A,B,C,D);

8- Find the poles of the open loop system using the MATLAB command eig:

eig(A) or ***poles_OL=eig(A)***

9- Find the pole-zero locations of the SISO system using the MATLAB command:

[z,p,k]=ss2zp(A,B,C,D) ***OR*** ***[z,p,k]=ss2zp(sys_OL)***

10- Plot the pole-zero locations of the SISO system in the s-plane using the MATLAB command:

pzmap(sys_OL)

11- From the open loop poles locations, what can you infer about the open loop system stability?

- Pole Placement Design:

12- Go through *pp. 9-10* of the SPG handout and use the design performance requirements in p. 4, to find the dominating pair of complex conjugate poles, p1 and p2.
For controllability check, you can use the MATLAB command ***Co=ctrb(A,B)*** and then ***rank(Co)***.

13- Go through steps 3 – 8, *pp. 12-13* in the SPG handout

The place command in step 4 can be used as follow:

a- First put your calculated poles in a vector in the form:

p=[p1, p2, p3, p4]

b- Use ***K=place(A,B,p)*** to find the state-feedback gain vector.

c- Define the closed loop matrices as:

   a. A_{CL}= A-B*K
   b. B_{CL}=B*K(1)
   c. C_{CL}=C
   d. D_{CL}=D

d- Repeat steps from 7-10 in this handout for the closed loop system.

e- The step command can be used as follow:
step(sys_OL) & step(sys_CL)

f- In the resulting plots, right click to show the system characteristics (overshoot and settling time). Point at the resulting locations to get the values.

g- Re-adjust the values of p3, and p4 until the requirements are satisfied.

SIMULINK

Follow the SPG handout, pp. 13-18.

Real Time Implementation

Follow the SPG handout, pp. 19-26.

- Take screenshots for your work.
- Show your MATLAB code and its outputs.