Title: **Automatic Control V: PID Tuning and Simulation for Mag. Lev. System**
Course: **Laboratory Experiments 4 (CSE411)**
Lab No.: **15** ***Category:*** **Automatic Control**
Date: **18/4/2023** ***Due:*** **15/5/2023** ***Time:*** **4 Hours**

Objectives:

1. To get familiar with MATLAB Control System SISO Tool.
2. To understand how to design a PID controller using MATLAB.
3. To practice the discrete implementation of a PID controller.

Hardware Requirements:

- ✓ Feedback 33-006 Magnetic Levitation System.
- ✓ PC.

Software Requirement:

- ✓ MATLAB R2018a or higher.
- ✓ Simulink.

Pre-lab:

1. What are the effects of P-I-D components of the PID controller?
2. What are dominant poles?
3. What is controller emulation?
4. How to verify performance of controllers?

Part 1: Using MATLAB SISO Tool to Design a PID Controller for Mag Lev System:

1. Define the system transfer function as:

$$G(s) = \frac{1658}{s^2 - 872.9}$$

2. Send the system transfer function to the MATLAB SISO tool:

```
>> sisotool(G)
```

3. Investigate the root locus and open loop bode (Figure 1):

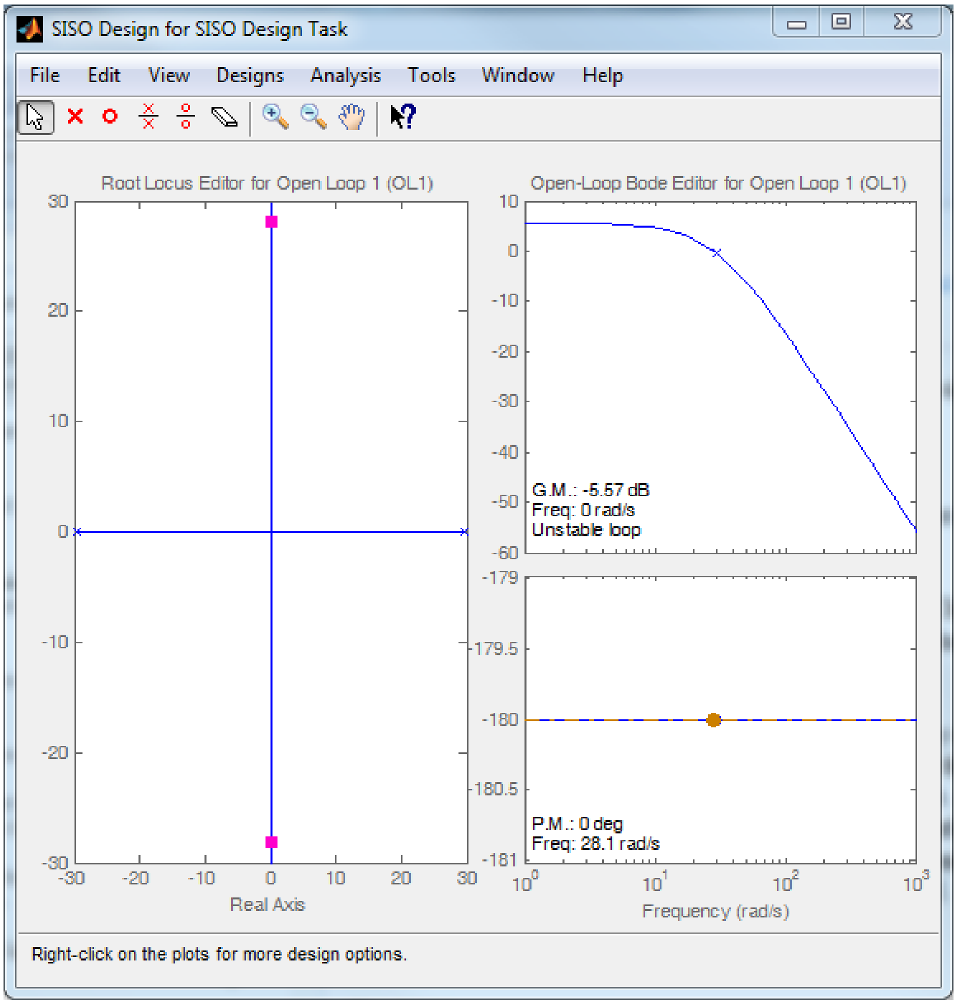

Figure 1 Analysis Plots of the Open Loop System

4. Choose the control architecture (Figure 2).
5. Define the PID controller poles and zeros at the Compensator Editor tab (Figure 3):

$$C_{PID}(s) = K_P + \frac{K_I}{s} + K_D.s = \frac{K_D.s^2 + K_P.s + K_I}{s}$$

6. Use the Automated Tuning for PID to get initial values for your PID design (Figure 4).
7. Analyze the effects of the designed controller using the Analysis Plots tab (Figure 5).

8. Use the Graphical Tuning tab to further tune your PID parameters (Figure 6).

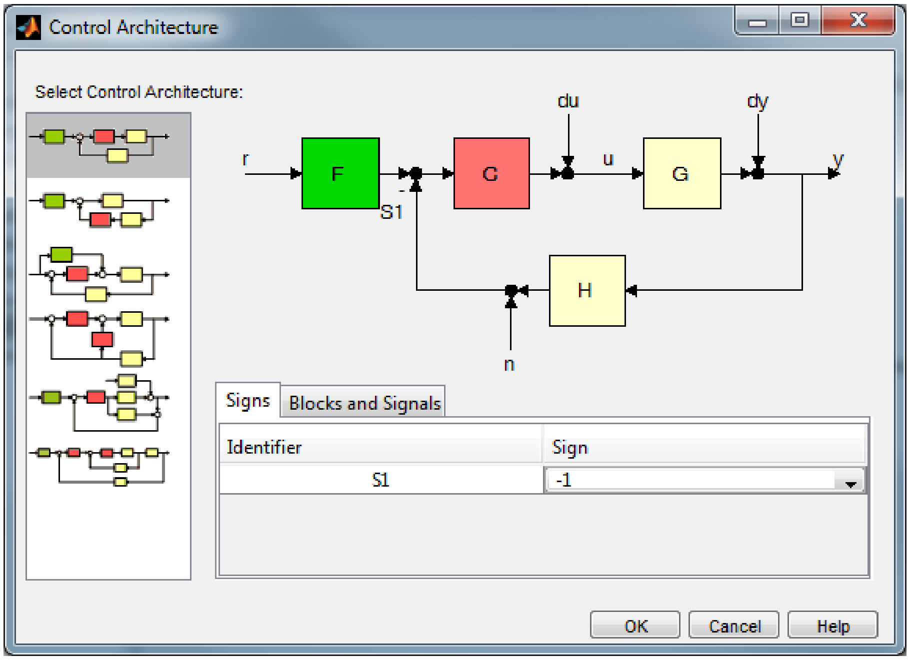

Figure 2 Control Loop Architecture

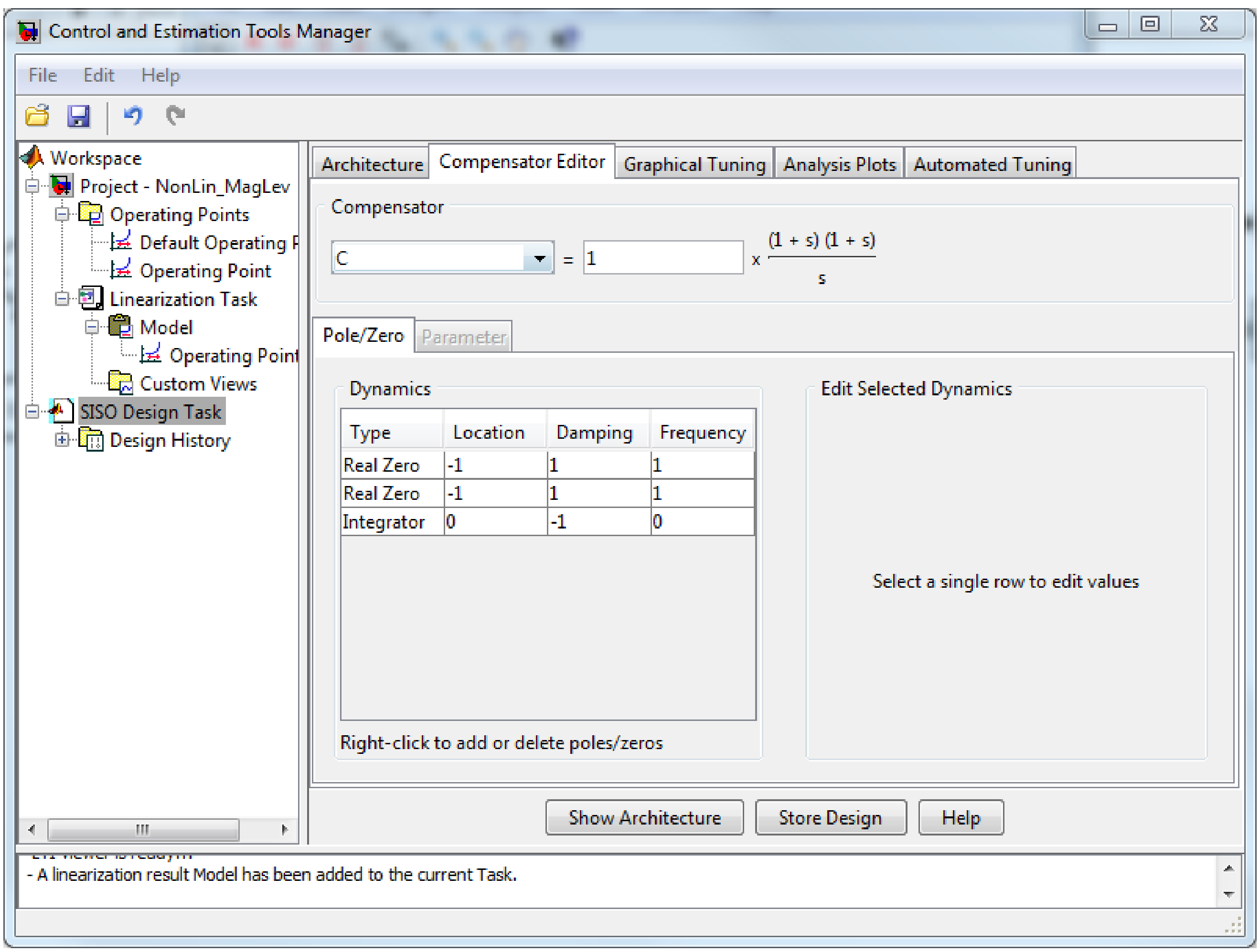

Figure 3 Controller Poles and Zeros

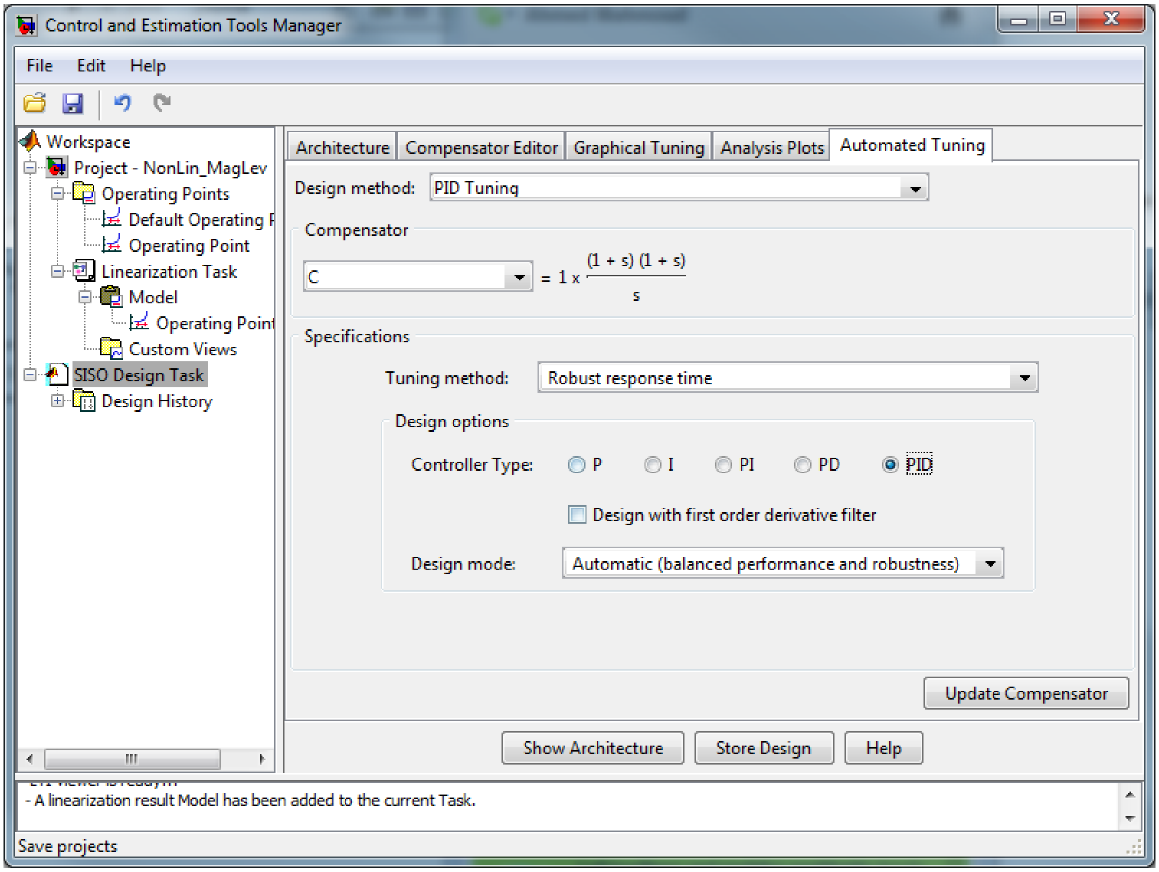

Figure 4 Automated Tuning of PID Controller

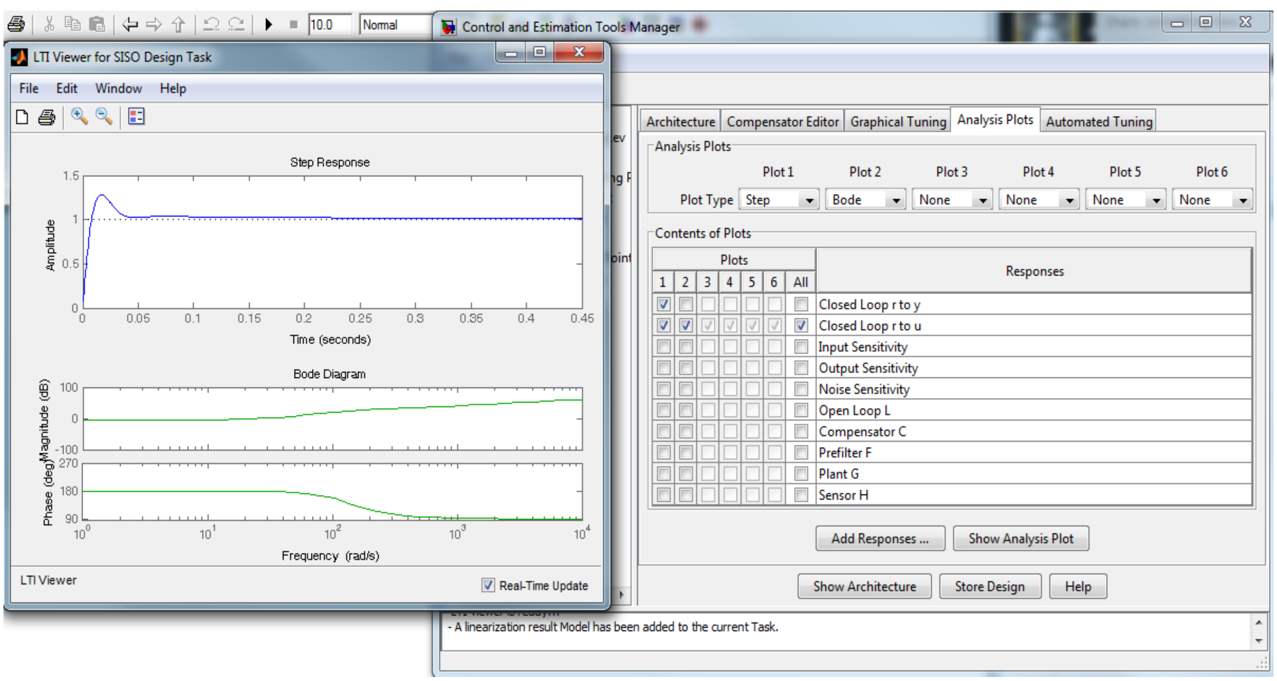

Figure 5 Analysis Plots

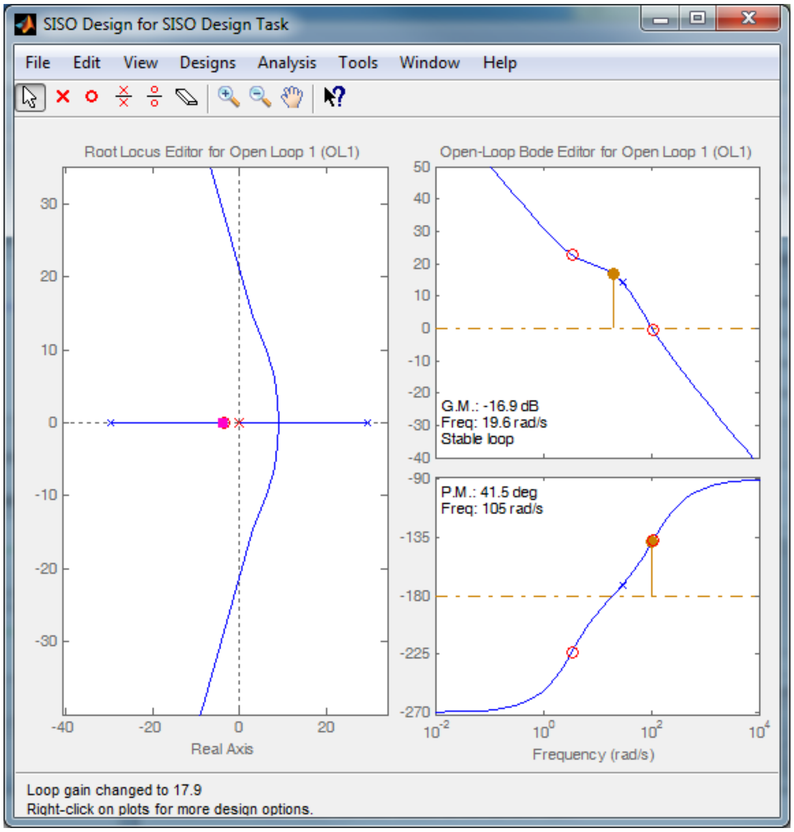

Figure 6 Graphical Tuning

9. Add your gains to a separate m file: K_P, K_I, K_D, and use saturation values for control signals as [-5,5] and integrator saturation limits as 60%.
10. Implement your PID in the Simulink model to test the controller with the nonlinear model (Figure 7).

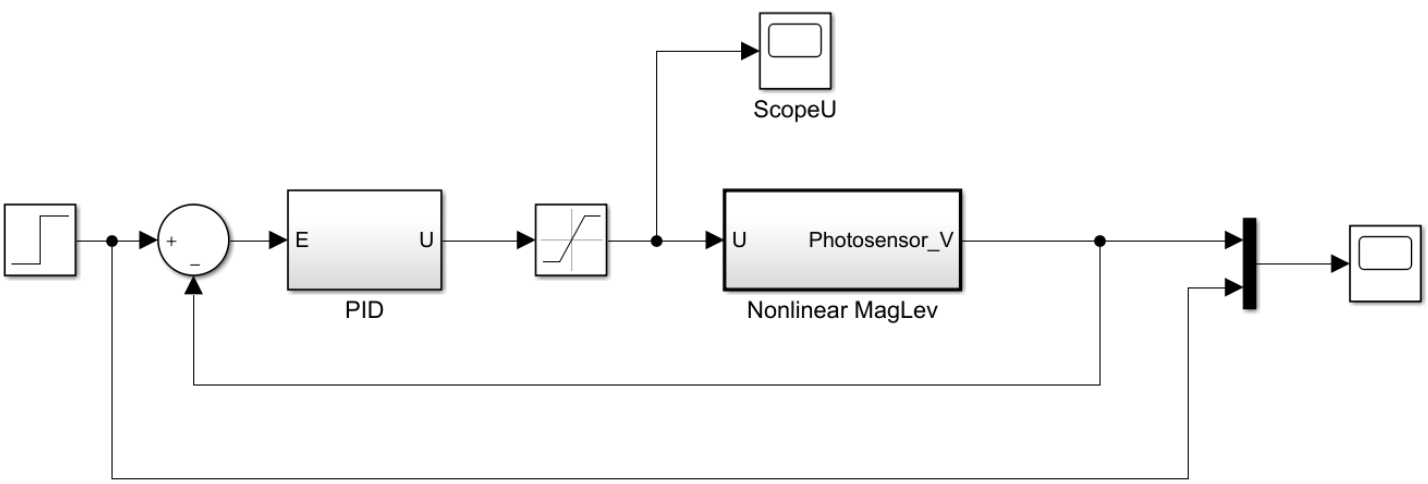

Figure 7 Simulation of the Mag Lev System with PID Controller

11. Compare the reference with output, and check control signal.

Part 2: Emulate PID for Discrete Implementation:

1. Use the following PID structure with your tuned parameters:

$$C_{PID}(s) = K_P + \frac{K_I}{s} + K_D.s = \frac{K_D.s^2 + K_P.s + K_I}{s}$$

2. Emulate the controller to get to the discrete version as:

$$u(n) = K_p e(n) + K_i \sum_{j=0}^{n} e(j) + K_d[e(n) - e(n-1)]$$

3. Implement the previous equation as your controller algorithm using MATLAB code in the MATLAB function block.

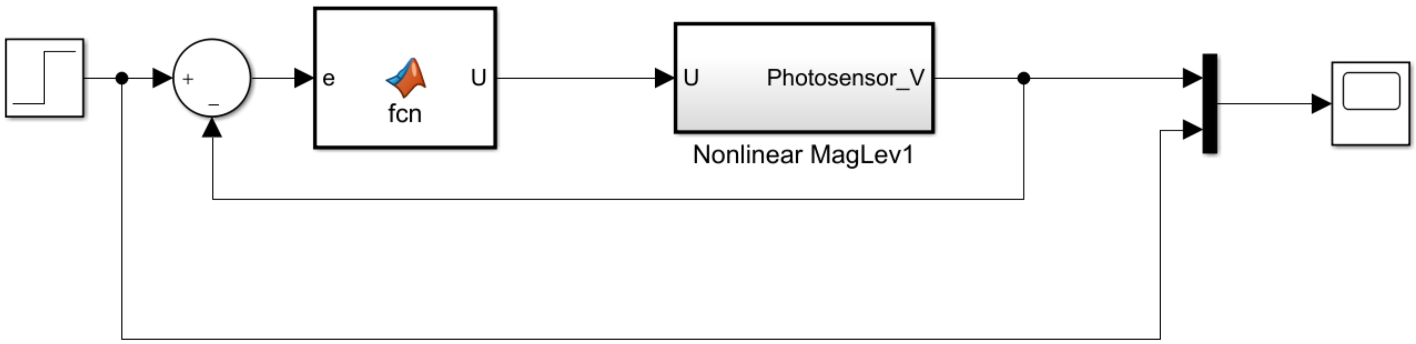

Figure 8 Discrete PID Implementation

Technical References:

1. *Magnetic Levitation System; Getting Started: 33-006*. Feedback Instruments.
2. *Magnetic Levitation Control Experiments 33-942S*. Feedback Instruments.
3. *Modeling and Control of a Magnetic Levitation System*. Marwan K. Abbadi and Winfred Anakwa.