EE128 Lab 5: Magnetic Levitation

Lucine Oganesian, Edward Lai, Hillary Chen
Lab Section 104

November 6, 2015

1 Purpose

From the lab: To design and implement an analog controller for a magnetic levitation (MagLev) system. To design a controller, we need a linearized model of the plant to be controlled."

2 System Identification

a. Prelab

(a)

$$\frac{Y - Y_0}{10e3} = \frac{Y_0 - Y_1}{10e3}$$
$$Y - Y_0 = Y_0 - Y_1$$

Noting that $Y_0 = Y_{ref}$, then

$$Y_1 = 2Y_{ref} - Y$$

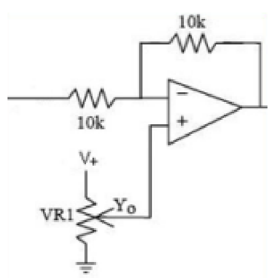

Figure 1: Output offset circuitry Op Amp

(b)

$$\frac{Y_1}{Z_1} = \frac{-Y_2}{10^4}$$

Solving for Z_1 (the impedence of the parallel circuit before the op amp):

$$Z_[1] = \frac{1}{\frac{1}{R_1} + \frac{1}{R_2 + \frac{1}{Cs}}} = \frac{R_1(R_2 + \frac{1}{Cs})}{R_2 + \frac{1}{Cs} + R_1} = \frac{R_1(R_2Cs + 1)}{R_2Cs + 1 + R_1Cs}$$

$$\frac{Y_2}{Y_1} = \frac{-10^4(Cs(R_1 + R_2) + 1)}{R_1(R_2Cs + 1)}$$

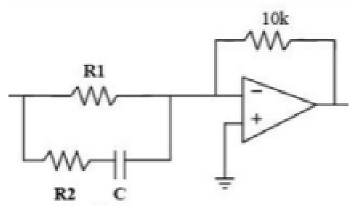

Figure 2: Controller circuitry Op Amp

(c)

$$\frac{Y_2}{10^4} = \frac{-Y_i}{10^4} - \frac{Y_0}{10^4}$$
$$V_0 = -(Y_2 + Y_i)$$

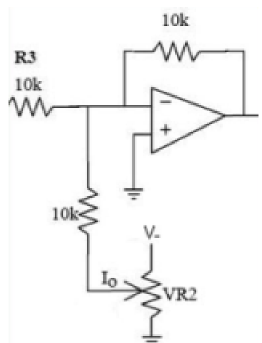

Figure 3: Current offset circuitry Op Amp

(d) linearized plant is characterized by $m\ddot{x} = K_i \delta I + K_x \delta x$ and $y = a\delta x$

$$ms^2 X = K_i I + K_x X$$

$$Y = aX$$
$$X = \frac{K_i I}{ms^2 - K_x}$$
$$Y = \frac{aK_i I}{ms^2 - K_x}$$
$$G(s) = \frac{Y(s)}{I(s)} = \frac{aK_i}{ms^2 - K_x}$$

b. Lab
In order to linearize the plant, we needed to take several measurements consisting of small deviations around an equilibrium point.

(a) We first chose an equilibrium point to be 3.7mm as the equilibrium point. This equilibrium point was found by raising the stage and scale apparatus such that the steel bearing covered approximately half the light reaching the photoresistor.

(b) We then measured the resistance of the photoresistor when it was fully covered and when it was fully uncovered by the steel bearing. This was measured to be 2.51kΩ and 0.737kΩ respectively. We then choose a value of R so that the voltage divider is most sensitive between 0 and 7 Volts. Our value chosen was $R = 1.5k\Omega$.

(c) For calculating a, the change in voltage across the photoresistor as a function of displacement, we moved the stage up and down in small increments and measured the output voltage. The resulting values were:

Offset(m)	Voltage of y
-0.0005	3.27
-0.0004	3.34
-0.0002	3.56
0	3.86
0.0002	4.09
0.0004	4.21
0.0005	4.5

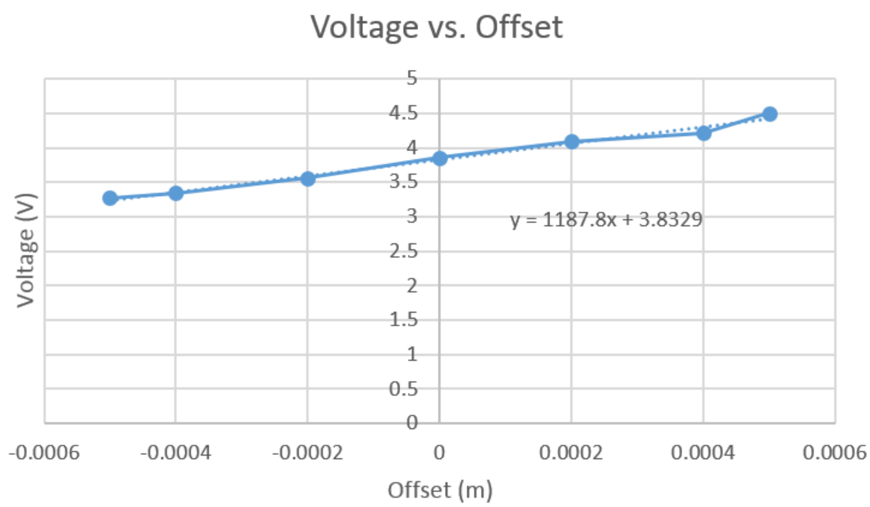

Figure 4: Output voltage vs offset height

To calculate a, we fit a line to our collected points and determined that $a = 1187.8V/m$

(d) For calculating K_i measurements were taken starting from a near weightless position then successively decrementing the current to the electromagnet (therefore, increasing weight values). To convert to newtons, we took the recorded weight values and computed $F = m_0g - mg$, where m_0 is the original mass of the ball (16.1 g) and m was the recorded mass.

Offset Current(A)	Force(N)
0	0.153
-0.1467	0.141
-0.1333	0.139
-0.2467	0.132
-0.4333	0.113

After fitting a line to the curve, $K_I = 0.0921N/A$.

(e) K_x was calculated in a similar way. Except instead of recording current supplied, we recorded the height (in mm) with respect to the weight. Our recorded values were:

Height(m)	Force(N)
0.0002	0.0006
0.00	-0.0033
-0.0005	-0.0141
-0.001	-0.0239
-0.0013	-0.0307

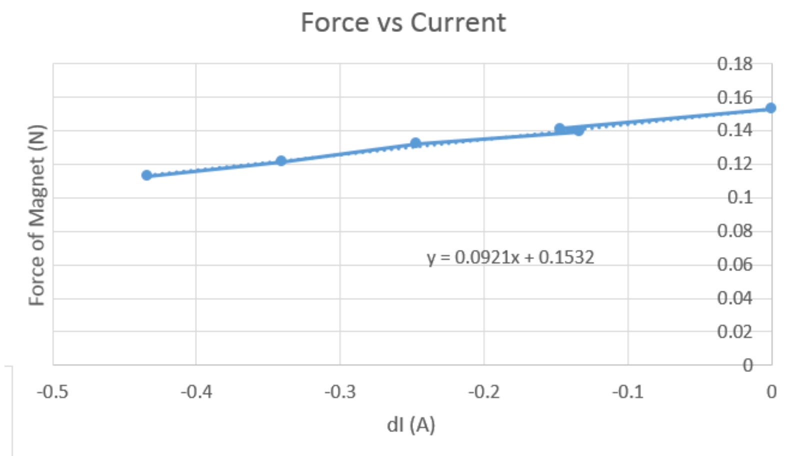

Figure 5: Calculated force of magnet vs offset of equilibrium current

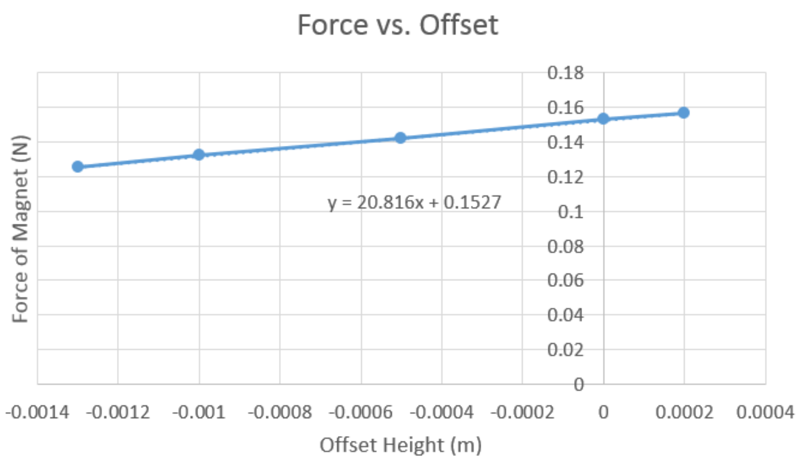

Figure 6: Calculated force of magnet vs offset of height

After fitting a line to the curve, $K_X = 20.816 N/m$.

3 Designing and implementing a control system

a. Prelab
Once we had calculated our a, K_I, and K_X values, we needed to design our controller and pick the appropriate resistor and capacitor values. First, we plotted the root locus and frequency response of our system:
The plant is inherently unstable as it has two poles on the $j\omega$ axis to begin with and with increasing gain values, one pole shoots off into the right half plane. Therefore we needed to design a compensator for the plant.
Since we wanted to attract the pole that ends up in the RHP into the LHP (thus effectively improving the transient response), we decided a lead compensator was needed.
The lab specified a DC gain of 2, so

$$K_c = \frac{2 * K_x}{a * K_a * K_i} = 0.1903$$

To obtain a phase margin of 60 degrees, we moved the pole and zero of the compensator around until such a margin was achieved using the SISO tool. We found a suitable pole at -0.00034 ($p = 0.00034$) and

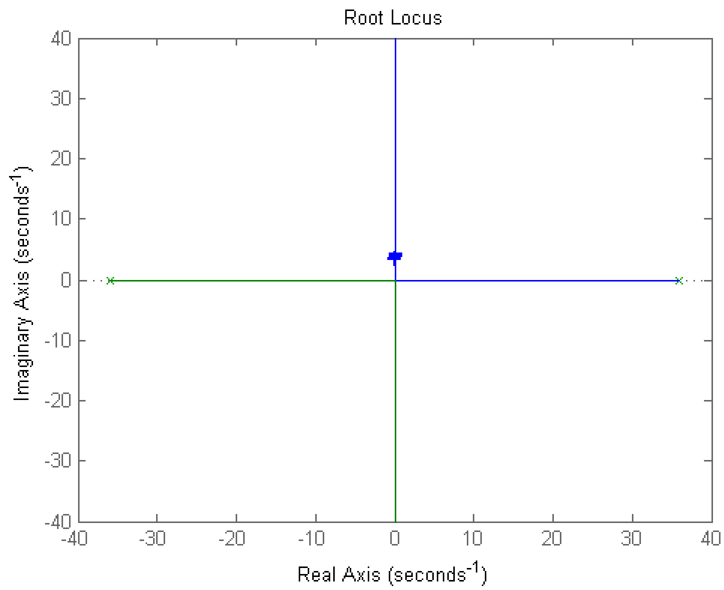

Figure 7: Root locus of the plant

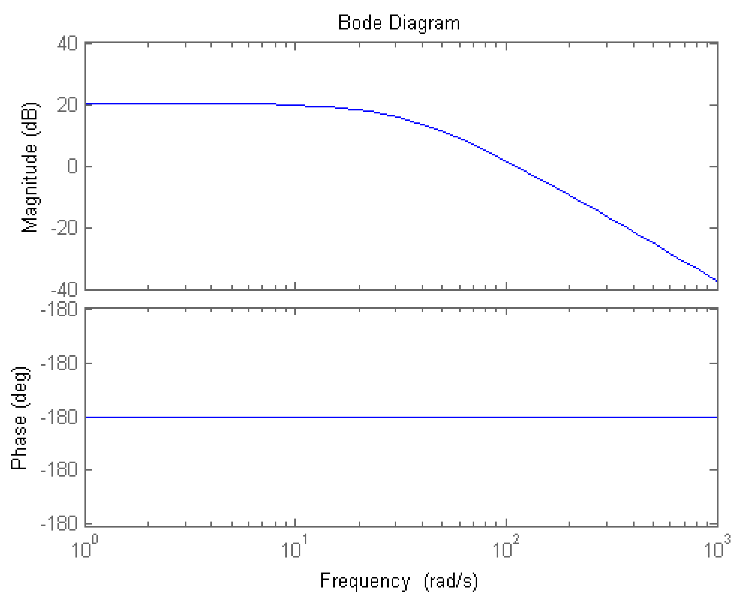

Figure 8: Frequency response of the plant

a suitable zero at 0.78 ($z = 0.78$), as seen below. DC gain of the resulting system was 6dB, as expected.

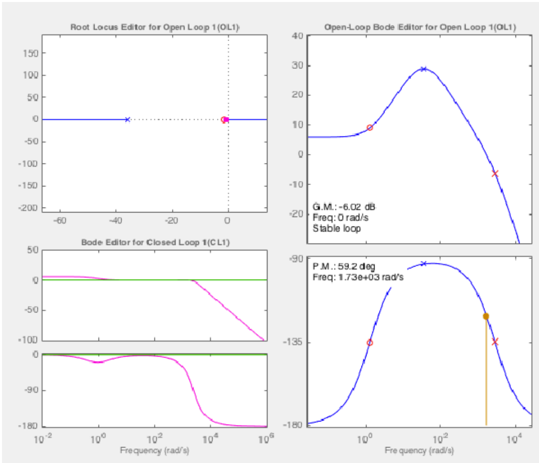

Figure 9: Lead complensator location on plot to create a phase margin of 60 degrees. 59.6 degree phase margin was achieved.

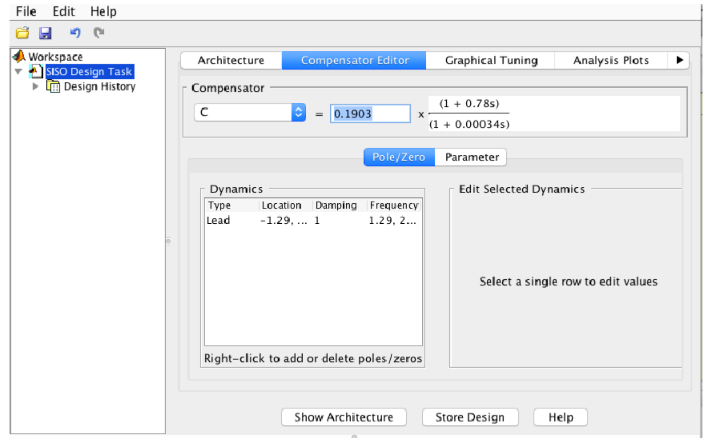

Figure 10: Location of pole and zero

From this we were able to determine the controller gain:

$$G_c = K_C * \frac{1+s/z}{1+s/p} = \frac{0.2454s + 0.1914}{2941s + 1}$$

Finally, we calculated values for R_1, R_2, and C:

$$R_1 = 10^4/K_C = 5.2549 * 10^4 \Omega$$

$$C = \frac{1/z - 1/p}{R_1} = 14.8\mu F$$

$$R_2 = \frac{1}{p * C} = 22.916\Omega$$

b. Lab

Actually implementing the controller system proved to be quite difficult. Despite checking each subcircuit independently and then checking the complete circuit once all three modules were connected, our controller would not working. We performed system identification again only to yield similar results.

In the end, using $R_1 = 52.14k\Omega$, $R_2 = 22.916\Omega$, and $C = 10\mu F$ yielded a working controller and our maglev worked. These were the third set of resistor and capacitor values we used. We think the first set did not work due to changes in the system after our initial measurements. After that we think that the second set of resisitors did not work due to incorrect wiring of the potentiometer. The third set of values worked successfully even though our designed controller had the zero significantly closer to the pole than had been suggested (suggestion was at s=-20). One possible explanation for this might have been that we were using measurements from the system identification that were not completely accurate, and therefore the only way to achieve a successful controller was to create something that was as close to an ideal PD controller as possible. However this also probably contributed the the inherent noise in our final product; there was only a very narrow range around the equilibrium where it successfully floated (i.e. it was not particularly robust). Unfortunately we ran out of time to play around with our controller design and our resistor values, however future steps would include adjusting resistor and capacitor values to increase robustness.

In order to tune our parameters and component values, we always began with our ball at equilibrium (3.7mm). We adjusted the first potentiometer in order to correctly calibrate error to zero. After that we disconnected R_3 and adjusted the potentiometer in the current offset circuitry to make sure that the apparent weight at equilibrium was zero. Lastly, once R_3 was replaced back into the circuit, we placed the bearing around equilibrium position, while adjusting the gain of the controller (R1) to make it as stable

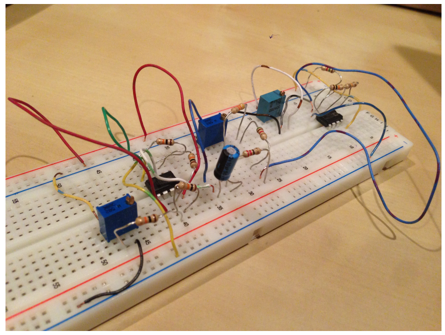

Figure 11: Image of our breadboard with complete circuit

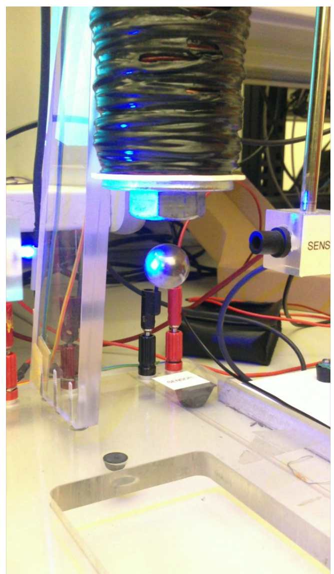

Figure 12: Levitating ball bearing

as possible.

A video of our working maglev can be found at https://youtu.be/PehrvzE2oHU.

4 Appendix

The following script was used when running the simulations and calculating step response information in the prelab.

```
%% voltage weight
clc
clear all
close all

I = [-0.3050, -0.2830,   -0.2850,   -0.2680,   -0.2540,   -0.2400] * (20/3);
I = I - I(1);
w = 9.8*(16.1*1e-3 - 1e-3*[0.5000, 1.7000, 1.9000, 2.6000, 3.7000, 4.6000]);

coeffs = polyfit(I, w, 1);
fittedY = polyval(coeffs, I);

KI = diff(fittedY) ./ diff(I);
```

```
figure()
plot(I, w)
hold on
plot(I, fittedY, '+')
title('Force vs Voltage supplied')
xlabel('Current (I)')
ylabel('Force (N)')

%% height weight
clc
clear all
close all

H = [3.9 3.7  3.2 2.7 2.4] * 1e-3;
H = H -H(2);
w = 1.6*1e-3 - 9.8*1e-3*[0.1 0.5000 1.6000 2.6000 3.3];

coeffs = polyfit(H, w, 1);
fittedY = polyval(coeffs, H);

KX = diff(fittedY) ./ diff(H);

figure()
plot(H, w)
hold on
plot(H, fittedY, '+')
title('Force vs Height, with -0.307V supplied')
xlabel('Height (m)')
ylabel('Force (N)')

%% a value
clc
clear all
close all

deltX = [-0.5000, -0.4000, -0.2000, 0, 0.2000, 0.4000, 0.5000] * 10^(-3);
V = [3.2700 3.3400, 3.5600, 3.8600, 4.0900, 4.2100, 4.5000];

coeffs = polyfit(deltX, V, 1);
fittedY = polyval(coeffs, deltX);

a = diff(fittedY) ./ diff(deltX);

figure()
plot(deltX, V)
hold on
plot(deltX, fittedY, '+')
title('displacement vs Voltage')
xlabel('displacement (delta x)')
ylabel('voltage')


%% controller
a = 1.0e+03 * 1.1878;
```

```
Ki = 0.092;
Kx = 20.8;
m = 16.1*10^(-3);
sys = tf([2 * a * Ki], [m 0 -Kx]);

% calculate gain value to use for controller gain
Kc = 2 * Kx / (2 * a * Ki);

% CALCULATED COMPONENT VALUES
% R1 =   5.2549e+04 Ohms
% R2 = 22.916 Ohms
% C = 1.4837e-05 = 14.8e-6 F

% ACTUAL COMPONENT VALUES
% position offset (pot) --> 8.026 kOhms
% potentiometer current offset --> 3.956 kOhms
% resistor (R1) 5K + 4.8K (pot) + 42.34 K
% resistor (R2) 21.9 Ohms
% cap 10microfarad
% 1.5K voltage divider of sensor
% coil isn't linear either; temperature (i.e. how the coil heats up could
% affect as well)
% heats up and shit
```

Listing 1: lab5script.m