Lab 5b:

Wieger, Bryan
26612985

Gunel, Beliz
24203242

Rodewald, Keenan
25514977

Wednesdays 5-8pm

1 Purpose

As per the lab 5b assignment sheet: The goal of this lab is to design and implement an analog controller for the magnetic levitation system that you identified in last weeks lab. Our aim is to design an analog compensator such that the open loop systems DC gain (which include the compensator gain and the plant) is 2 A/m and the compensator pole/zero ratio is approximately 20.

2 Pre-Lab

During the lab section, the system identification steps were repeated to find system values of the new station. The system was characterized as follows using methods identical to Lab 5a:

$$a = 858.8571$$

$$K_i = 0.2456$$

$$K_x = 36.0640$$

$$K_c = .1710$$

2.1 Root Locus and Frequency Response Plotting

The root locus of the system defined by the transfer function found in lab 5a and the system values found above can be seen in figure 1 From the root locus, it is evident that the system is never fully stable. At

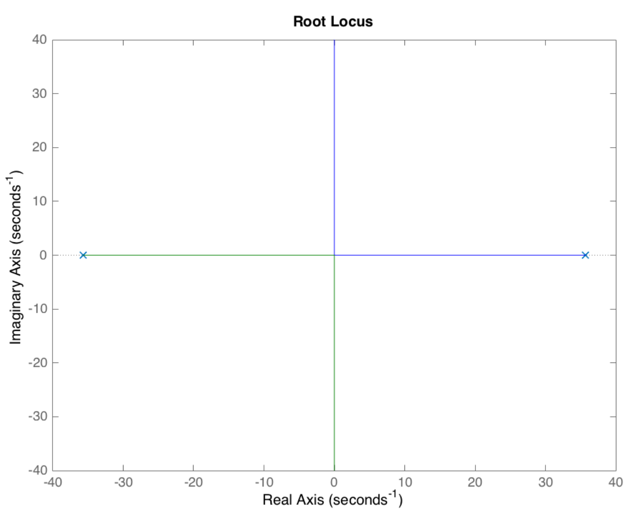

Figure 1: Root locus plot for the magnetic levitation system with system values found in the Pre-Lab

best for a range of K values it will be marginally stable, since the Loci are on the $j\omega$ axis. This finding is reinforced by the bode plot seen in figure 2 In the bode plot, it is evident that the phase is $-180\,\deg$ for all frequencies. This means that the phase margin is 0 and the system is marginally stable.

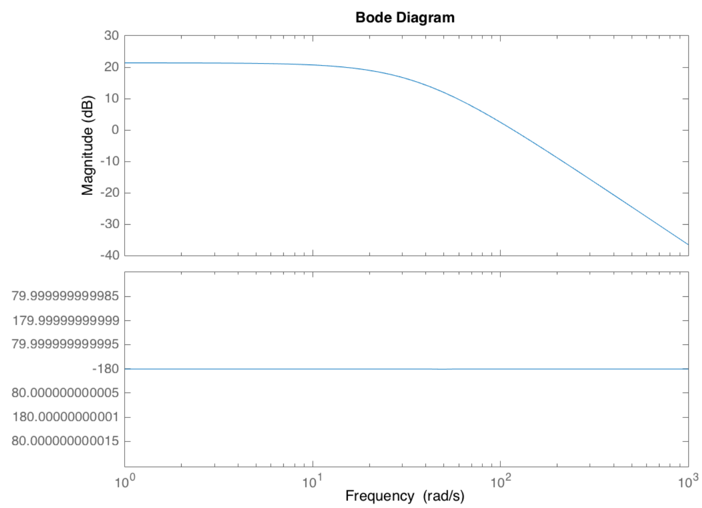

Figure 2: Bode plot of magnetic levitation system

2.2 Controller Design

Using sistool, an iterative approach was taken to choose pole and zero locations to achieve a phase margin of 60 deg as suggested by the lab GSI. Based on work in sistool, the desired compensator pole was found to be 340 and the zero location was placed at 17. According to sistool this produced a phase margin of 64.7 deg and a gain margin of -1.36.

Using these pole and zero locations, the op-amp circuit was analyzed to determine resistor and capacitor values. Using the labels of fig 3

$$R_1 = 58.49k\Omega$$

$$R_2 = 3.08k\Omega$$

$$C_1 = 0.95\mu f$$

Note that while these were the values found using Matlab, in the actual lab we ended up having to change these values in order to make a functioning controller, as will be discussed later in the lab.

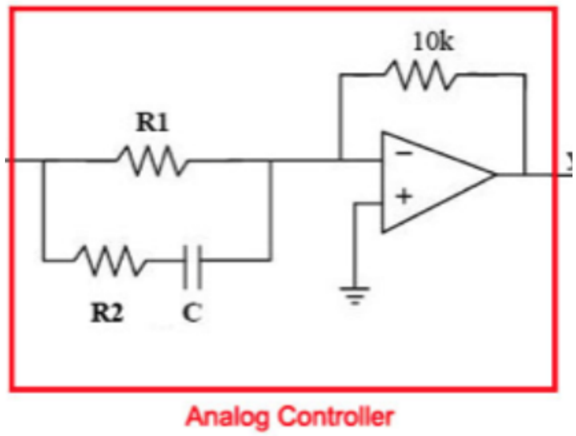

Figure 3: Lead Compensator opamp circuit

3 Lab

3.1 System Identification

3.1.1 Finding a

The data used to re-identify system values is listed as follows.

mm	Volt
8	3.63
9	4.28
7.5	3.02
7	2.58

This data is plotted along with the linear fit found in 4. a was found to be 858.8571.

3.1.2 Finding K_i

In order to find K_i, the current through the magnetic coil was varied and the ball weight at the equilibrium position was measured.

Current [A]	Ball Weight [g]
1.467	-1.4
1.39	0.7
1.29	2.3
1.187	5.9

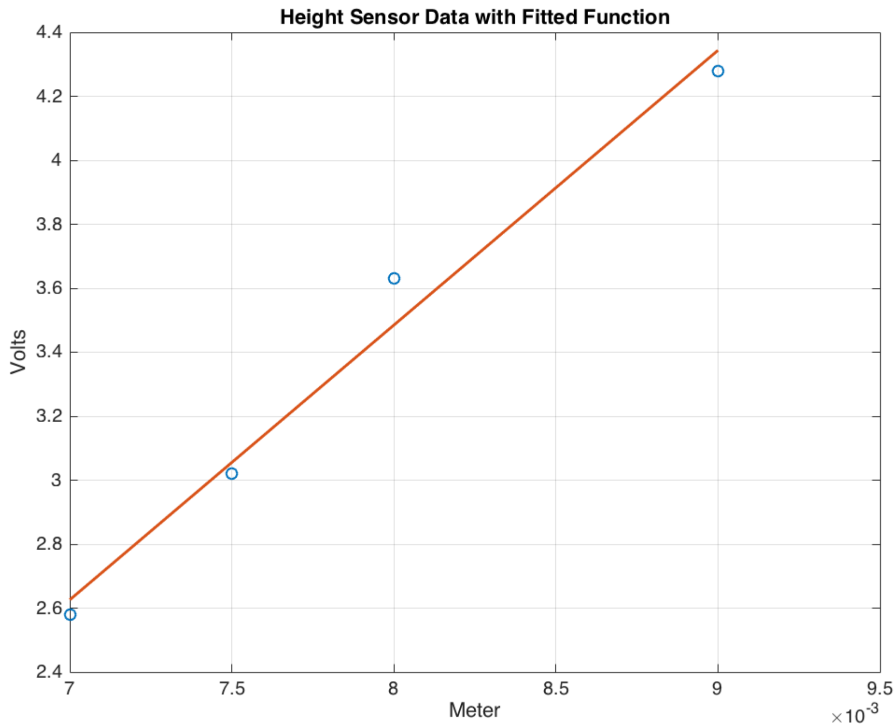

Figure 4: Plot of experimental sensor voltage vs ball height [mm]

This data is plotted alongside the fitted line in figure 5. K_i was found to be 0.2456.

3.1.3 Finding K_x

In order to find K_x, the ball height was varied and the ball weight was measured at each height.

Height[mm]	Ball Weight [g]
8	-0.2
7.5	3.2
7	4.5
6.5	5.5

This data is plotted alongside the fitted line in figure 6. K_x was found to be 36.0640.

The results of these findings are summarized in the table below:

a	858.56
K_i	0.2456
K_x	36.064
K_c	0.1710

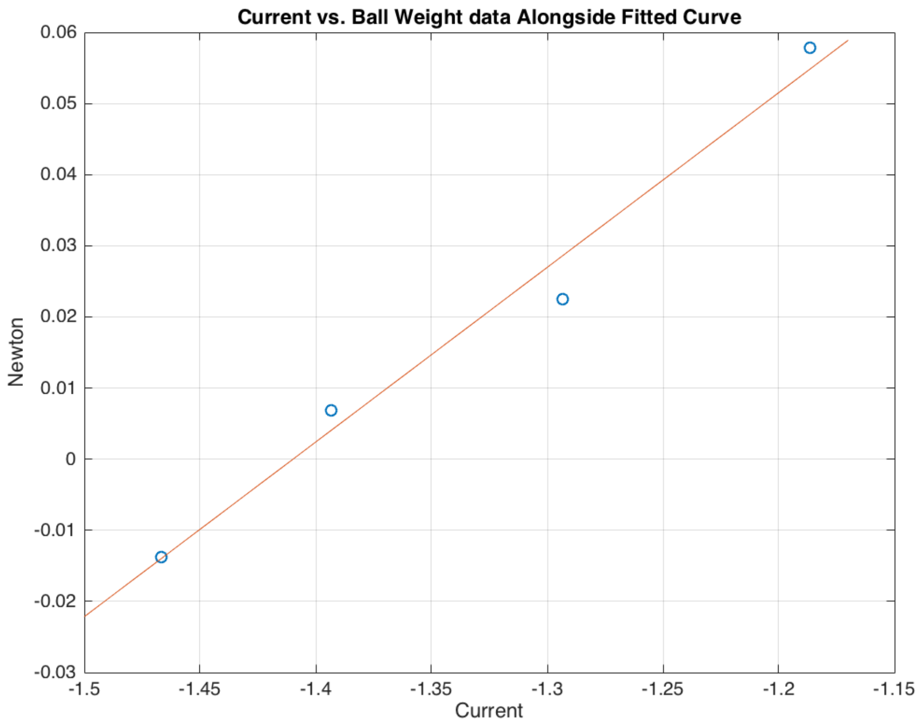

Figure 5: Plot of experimental current vs ball weight [N] with the fitted function in red

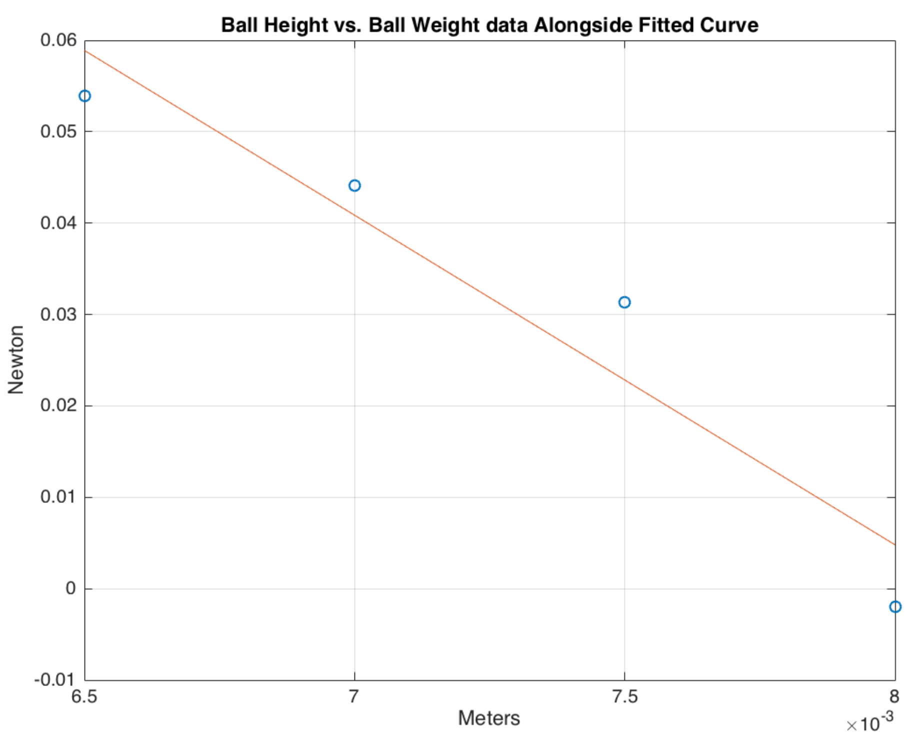

Figure 6: Plot of experimental ball height vs ball weight [N] with the fitted function in red

3.2 Controller Design Strategy

The controller was initially designed using Matlab to find pole and zero positions that resulted in desired PM and GM values. This approach was not fully successful and did not result in a stable system.

This initial design was then iterated upon in order to get the ball to remain stable. In an attempt to push the compensator poles and zeros further into the LHP (and thus promote stability) new resistor and capacitor values were used as follows:

R_1	$10.5k\Omega$
R_2	500Ω
C_1	$1\mu f$

With these values, the feedback control system performed its job and the ball was able to levitate. When placing the ball, there was very little transient behavior observed. Once the scale was lowered, the ball would stay in place without any visible oscillation. These component values corresponded to a compensator pole at $p = 2000$ and a zero at $z = 90.91$. This corresponds to a PM of 57.3 and a GM of -20.9dB.

Thus these controller values should not work theoretically because the $GM < 1$, however there were likely errors in the system identification steps that manifest themselves when computing the margins analytically. Clearly it is important not to rely solely on computed values.

4 Appendix

4.1 Code

Listing 1: "Code used to find answers and generate plots for the prelab"

```
%% h(x) a

mm = [8
9
%8.5
7.5
7
]

volt = [3.63
4.28
%4.94
3.02
2.58
]

meter = .001*mm

plot(meter, volt, 'o'),grid, xlabel('meter'), ylabel('Volts')
blah = polyfit(meter,volt,1)
slope = blah(1)
a = slope
a = abs(a)

%% Plots for report
    plot(meter, volt, 'o'),grid, xlabel('Meter'), ylabel('Volts')

    x_test = linspace(0.007, 0.009)
    y_test = x_test * blah(1) + blah(2)
    hold on
    plot(x_test, y_test, 'LineWidth', 1.5)
    title('Height Sensor Data with Fitted Function')
%% ki

current = [-1.466666667
-1.393333333
-1.293333333
-1.186666667
]

grams = [-1.4
0.7
2.3
5.9
]

kg = .001*grams

newton = 9.8*kg

plot(current, newton, 'o'), grid, xlabel('current'), ylabel('newton')
blah = polyfit(current,newton,1)
```

```
slope = blah(1)
ki = slope
Ki = abs(ki)

%% Plot for Report

plot(current, newton, 'o'), grid, xlabel('current'), ylabel('newton')

x_test = linspace(-1.17, -1.5);
y_test = x_test * blah(1) + blah(2);

hold on
plot(x_test, y_test)
title('Current vs. Ball Weight data Alongside Fitted Curve')
xlabel('Current')
ylabel('Newton')

%% kx

mm = [8
7.5
7
6.5
]

grams = [-0.2
3.2
4.5
5.5
]

kg = .001*grams

newton = 9.8*kg

meters = .001*mm

plot(meters, newton, 'o'), grid, xlabel('meters'), ylabel('newton')
blah = polyfit(meters,newton,1)
slope = blah(1)
kx = slope
Kx = abs(kx)

%% Plot for report
plot(meters, newton, 'o'), grid, xlabel('Meters'), ylabel('Newton')

x_test = linspace(.0065, .008);
y_test = x_test * blah(1) + blah(2);

hold on
plot(x_test, y_test)
title('Ball Height vs. Ball Weight data Alongside Fitted Curve')

%% Controller Design

a = 858.8571;
Ki = 0.2456;
Kx = 36.0640;
Ka = 2;
m = 0.0283;
Kc = 2*Kx/(a*Ka*Ki)
```

```
% our Kc = .1710

%% Siso - ing
s = tf('s');
G = (a*Ka*Ki)/(m*(s^2-(Kx/m)));
sisotool(G);

%% Plot Root Locus and Frequency Response for Prelab

figure
rlocus(G)

figure
bode(G)

%% finding the element values

% eq1 = 10^4 /r1 * [(r1 +r2)Cs + 1]/[r2Cs + 1]
% eq2 = Kc * [(s/z +1)/(s/p + 1)]
% eq1 = eq2 -> 1/z = (r1+r2)*C and 1/p = r2*C

p = 340
```