Controls Final Report

Alexander Hughes and Nathan Faber

May 2022

1 Introduction

This report outlines how we designed and implemented a control system to levitate a permanent magnet from above via an electromagnet. This system is shown in Figure: 1. This system is unstable. We will use our knowledge of controls to stabilize the system and balance the magnet.

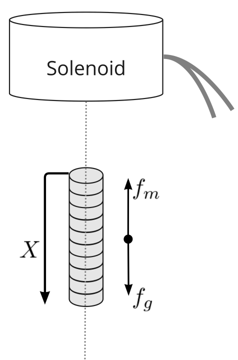

Figure 1: System Model

More specifically this system uses a 12v electromagnet and a Adafruit DRV8871 H-bridge motor driver.

2 Characterizing the Plant

From our initial research, magnetic fields are difficult to calculate and have complicated interactions with each other. However, we also knew that we only had the controls knowledge to control a linear system. To that end, we aggressively simplified the system from the beginning. We started with the assumption that the force on the magnet is proportional to its distance and the current in the solenoid, about some equilibrium point. We then had the challenge of characterizing the solenoid driver circuit; when an input signal is fed to the driver, does the current increase immediately or with some delay? Using an LCR meter, we measured the time constant of the solenoid (modeled as a resistor and inductor in series) to be 2.5ms. Under the assumption that the rest of our system would react much more slowly, we decided to ignore the driver circuit and treat the solenoid current as directly proportional to the input PWM signal.

After all these assumptions, we've linearized the solenoid-magnet system to have the following relationship between position, current, and force:

$$f_m = k_1 x + k_2 p + C_1 \tag{1}$$

where x is the magnet position in meters and p is the PWM signal expressed as a value between 0 and 1. The coefficients k_1 and k_2 are found by empirical calibration, and C is a constant. Since f_m is known to be highly nonlinear, the three constants are measured close to the desired equilibrium point.

We now have the tools to model the entire solenoid-magnet plant. The mechanical portion of the system matches the free body diagram in Figure 1. Writing out the equations of motion for a magnet with mass m,

$$mx'' = f_g - f_m \tag{2}$$

$$mx'' = mg - k_1 x - k_2 p - C_1 \tag{3}$$

$$mx'' + k_1 x = mg - k_2 p - C_1 \tag{4}$$

Unfortunately, the presence of constants mg and C interfere with our ability to find a transfer function from PWM signal to position. We solve this problem by expressing our inputs and outputs x and p as differences from an equilibrium value. We write the new variables as $\bar{x}$ and $\bar{p}$ respectively. By definition, at equilibrium

$$\bar{x}'' = \bar{x} = \bar{p} = 0 \tag{5}$$

and it follows that

$$mg - C_1 = 0 \tag{6}$$

$$m\bar{x}'' + k_1 \bar{x} = -k_2 \bar{p} \tag{7}$$

With this rearrangement, we can find a transfer function from normalized PWM input to normalized position:

$$ms^2 \bar{X}(s) + k_1 \bar{X}(s) = -k_2 \bar{P}(s) \tag{8}$$

$$\bar{X}(s)/\bar{P}(s) = \frac{-k_2}{ms^2 + k_1} \tag{9}$$

This model of the solenoid-magnet mechanical system is useful, but it is only part of the picture. Since we have decided to use the sensor reading as the output variable we are controlling, we need a transfer function from PWM input to Hall sensor voltage. We model the Hall sensor output as a function of both magnet position and PWM input; in tests we found that the solenoid creates a significant magnet field relative to that of the magnet. For a sensor voltage y (and a difference from an equilibrium value $\bar{y}$) we linearize about an equilibrium point:

$$y = k_3 x + k_4 p + C_2 \tag{10}$$

$$\bar{y} = k_3 \bar{x} + k_4 \bar{p} \tag{11}$$

Finally, we can combine the past equations into a single transfer function relating the input PWM signal to the Hall sensor output.

$$\bar{Y}(s) = k_3 \bar{X}(s) + k_4 \bar{P}(s) \tag{12}$$

$$\bar{Y}(s) = k_3 \frac{-k_2}{ms^2 + k_1} \bar{P}(s) + k_4 \bar{P}(s) \tag{13}$$

$$\bar{Y}(s)/\bar{P}(s) = \frac{k_4 \left(ms^2 + k_1\right) - k_2 k_3}{ms^2 + k_1} \tag{14}$$

$$\bar{Y}(s)/\bar{P}(s) = \frac{k_4 ms^2 + k_1 k_4 - k_2 k_3}{ms^2 + k_1} \tag{15}$$

The poles and zeros of this system are:

$$p_{1,2} = \pm\sqrt{-\frac{k_1}{m}} \tag{16}$$

$$z_{1,2} = \pm\sqrt{\frac{k_2 k_3 - k_1 k_4}{k_4 m}} \tag{17}$$

Compared with the pure solenoid-magnet mechanical system, the introduction of the sensor into the plant adds two zeros. Whether the system can be easily controlled depends on whether the zeros are real or imaginary, which we determined using empirical values for our various coefficients.

3 Calibrating the System

Our system characterization gave us a form of equation with which to model our system but did not generate values for the constants in the various relationships. We performed a system calibration to find empirical coefficients for our system around an equilibrium point.

In order to calibrate the system, we first needed to choose an equilibrium point. In order to simplify our implementation we chose to only power the solenoid in the attracting direction. We therefore chose the equilibrium point with the solenoid attracting at half power, as this would give us our input the most room in the negative and positive directions. We incrementally moved the magnets closer to the solenoid at half power until the magnets jumped up to the solenoid. The position of the magnets just before this occurred is our equilibrium position. We then varied first solenoid power and then position from his equilibrium point, using a tared scale to measure the net change in force resulting from these actions. Figure 2 shows the equipment used for calibration. Figures 3 and 4 show the results from these experiments. The data collected are reasonably linear over the chosen ranges, which is encouraging for our linear control approach.

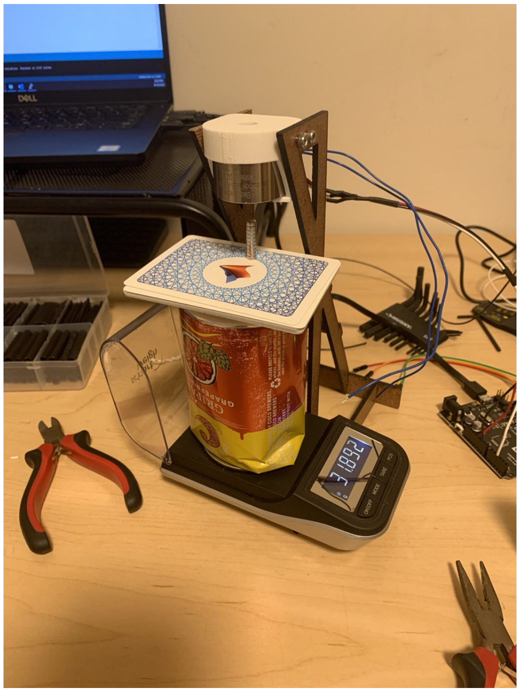

Figure 2: Calibration Setup

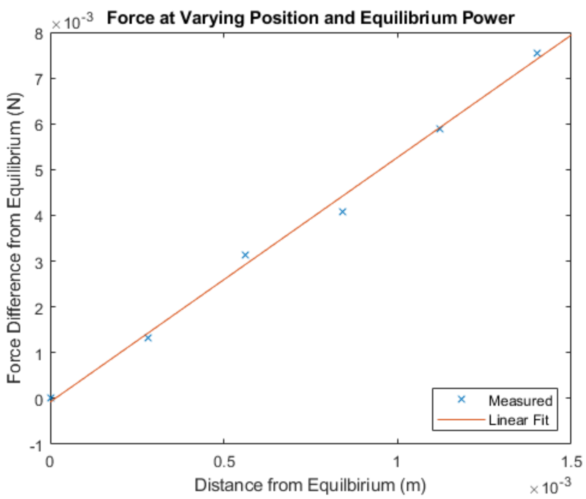

Figure 3: Force calibration results for varying position

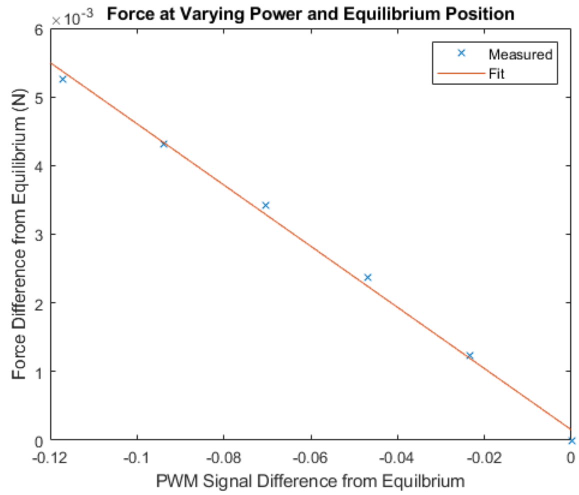

Figure 4: Force calibration results for varying power

In addition to these force measurements, we collected Hall sensor data while varying distance and power. Figures 5 and 6 show the measurements, which also appear reasonably linear over the chosen ranges.

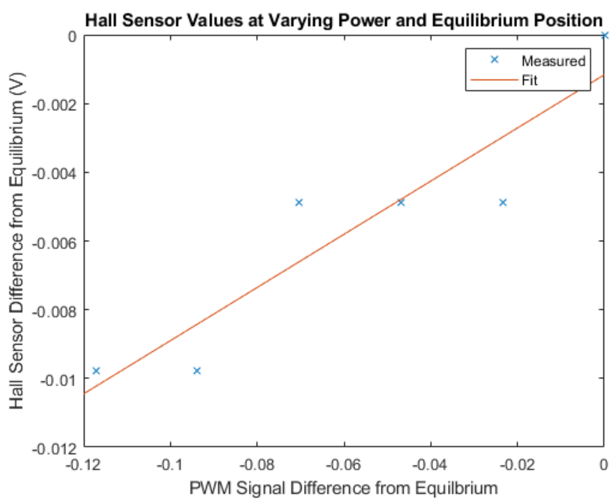

Figure 5: Hall sensor calibration results for varying power

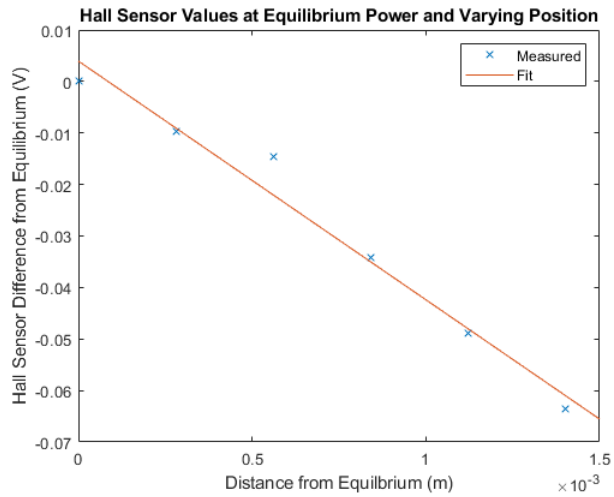

Figure 6: Hall sensor calibration results for varying position

The calibrated equations for our system dynamics are as follows:

$$\bar{f_m} = -5.344\bar{x} - 0.0446\bar{p} \tag{18}$$

$$\bar{y} = -46.34\bar{x} + 0.0774\bar{p} \tag{19}$$

Finally, the mass of the magnets is 0.00288kg.

Inserting these values into the transfer function identified previously, we obtain the plant transfer function:

$$\bar{Y}/\bar{P} = \frac{0.0773s^2 + 337.6}{s^2 - 1854} \tag{20}$$

Figure 7 shows the Pole-Zero plot for the plant of this system. It has a pole in the right half-plane, indicating its instability.

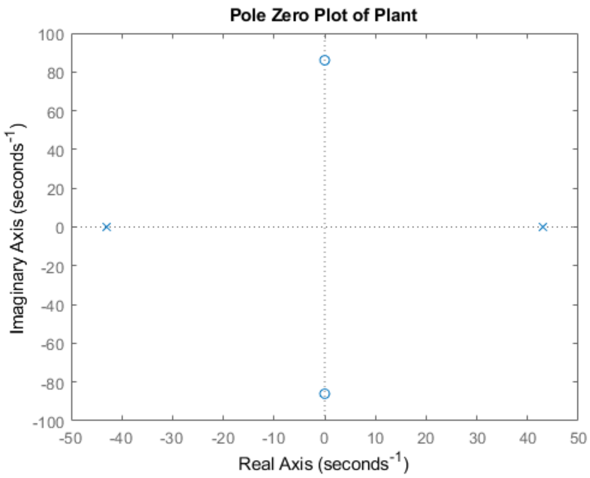

Figure 7: Plant Pole-Zero Plot

4 Simulation

After determining the plant for our system we wanted to simulate how the system behaves. This is needed to allow us to design a controller to stabilize the system. We chose to use Matlab and impulse responses to simulate our system. Using the plant transfer function we can simulate its impulse response. Figure 8 shows this response. As we expected, the plot shows that the open-loop system is unstable.

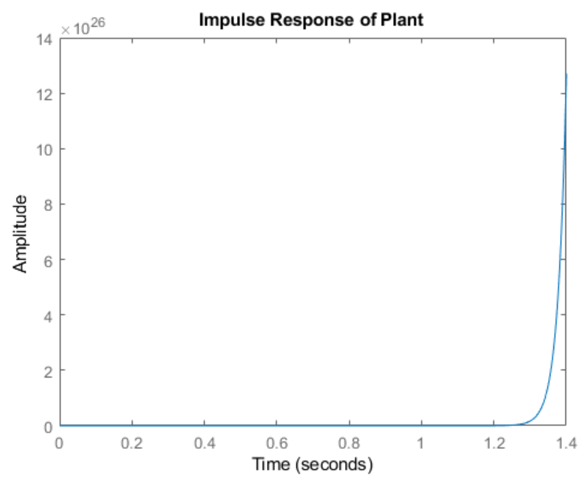

Figure 8: Plant Impulse Response

This impulse response is a baseline. When designing the controller we will simulate the system with an impulse response to look at the controlled system behavior.

5 Controller Design

Our open loop system is unstable and has poles in the right half-plane. Our controller needs to adjust the system such that the closed loop poles are not there. We analyzed using lead-control since this is a physically realizable controlled similar to Proportional-Derivative Control. A lead control adds a pole and a zero on the real axis where the zero is closer to the imaginary axis. We proposed placing the pole to the left of our system pole on the real axis, and the zero in the left half-plane to create a root locus that indicated closed loop stability. We also didn't want the pole extremely negative as that translates to high gains for high frequencies, which is something we want to avoid since the electromagnet injects noise based on it's PWM input signal.

We chose the pole to be at -50 and the zero to be at -30 on the real axis of the pole zero plot. Figure 9 shows the root locus for the system with lead control.

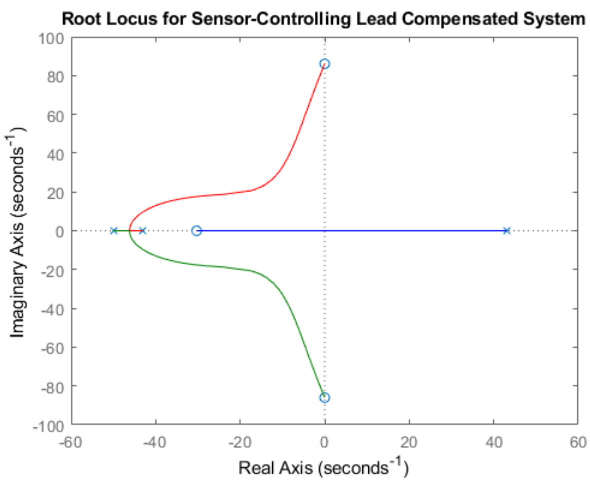

Figure 9: Lead Control Root Locus

The root locus plot indicates that for some values of K the entire system is stable! This implies that if we can control our system in this way, with the proper K value we will stabilize the magnet. Matlab allows us to look at the K value that moved all poles to the left half-plane, this value was scaled to an appropriate value that correlated with the PWM scaling in our implementation. For reference the gain in our system code is 800, and when we increased the gain we ended up going unstable which the root locus implies.

The system block diagram is shown in Figure 10.

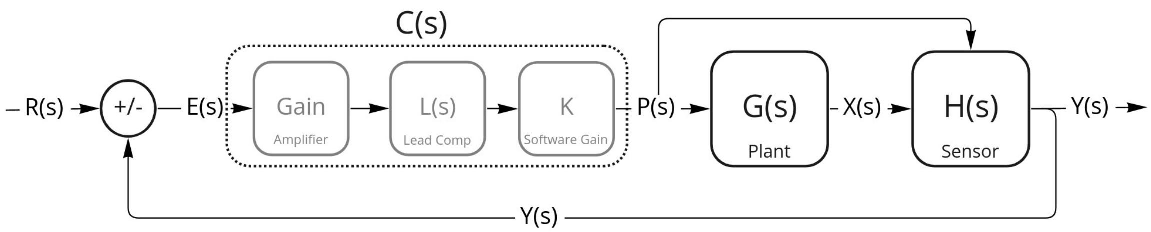

Figure 10: Lead Compensated System Block Diagram

For the pole and zero chosen, the transfer function of the lead compensator circuit we constructed is:

$$C_L(s) = \frac{s + 30.3}{s + 50} \tag{21}$$

The amplifier implemented in circuitry is:

$$C_A(s) = 4.318 \tag{22}$$

In total, then, the open loop transfer function of our controlled system with some gain K introduced by the microcontroller is

$$G_{OL}(s) = \frac{4.318K(s + 30.3)(0.0773s^2 + 337.6)}{(s + 50)(s^2 - 1854)} \tag{23}$$

Using the matlab plot shown in Figure 9 we chose a value of 1.4 for K, which approximately minimized the real components of our closed-loop poles.

We tested this controller(described by the transfer function above) in simulation. Figure 11 shows the success of this controller at rejecting the impulse and stabilizing the system. Our system will be starting from equilibrium by us positioning the magnet and then responding to very small impulses as various forces act on it. This impulse response gives us confidence that our system can stabilize.

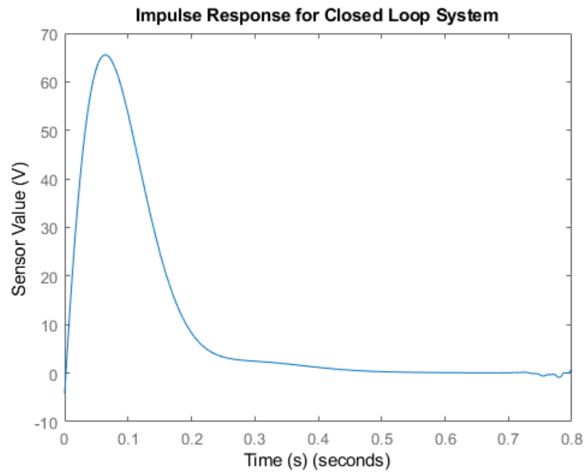

Figure 11: Lead-Compensated Impulse Response with stabilizing gain

6 Implementation

To implement the controller shown in the block diagram in Figure 10 we implemented the lead controller in circuitry. The signal coming into the lead-compensator was also amplified in circuitry to improve reading resolution(shown in the first gain block of our controller). The lead-compensated signal must be converted into a PWM signal to the electromagnet. This signal is translated into a PWM signal by another controller gain in software. This gain allows us to move our poles along the root locus of our open-loop system. This means that our micro controller is simply a proportional controller around our input signal and it simply scales the input into a valid PWM signal. Figure 12 shows our system balancing the magnet.

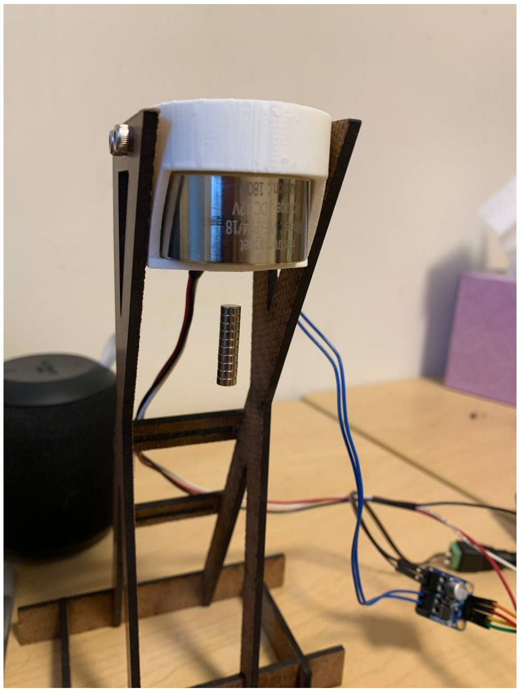

Figure 12: Hover

7 Appendix

- All code attached can be found here
- Previous work can be found in presentation slides here