Testing of Closed Loop Control of Eddy Current Actuated Test Platform in One Dimension

Garrick Lau
gl259@cornell.edu
Space Systems Design Studio

Advisors:
Brandon Hencey
Mason Peck

15 May 2014

I Abstract

General Topic: Relative Navigation in spacecraft is an essential tool that has yet to be perfectly mastered. Many methods of relative navigation, such as the use of GPS receivers on the two space vehicles, carry along inherent noticeable errors in measurement. However, due to the metallic structure, especially non-magnetic aluminum, of most space vehicles, the induction of an eddy current in the target spacecraft's surface provides a method of relative navigation that can be utilized ubiquitously.

Specific Question: We are looking to design a test platform that can be rebuilt in any space systems laboratory for a spacecraft that attempts to implement eddy current actuation. This process will evolve to three dimensions of closed loop control, but the project is currently designing for one translational degree of freedom.

Method: The platform is run in real-time through Simulink and XPC Target. The actuation is fed back by position sensors; these measurements create the control law that cause a microcontroller to actuate motors holding electromagnets. The direction of the magnetic field of these electromagnets controls the dynamics of a plate attached to a cart on a one-dimensional air track.

Results: Creating a physically realizable system is a challenge that no one has actually been able to perform. The system I created, I was unable to linearize. As well, testing was unable to be performed due to a shut down of the lab before work was complete. However, the entire process of testing is set up and will be discussed in this document.

II Introduction

An eddy current is an induced loop of electric charge in a non-magnetic surface by a exogenous magnetic field. Other magnetic fields could thus impart a force on this eddy current through the law of the Lorentz Force. Due to the fact that most spacecraft are composed of an aluminum frame, using normal magnetic forces are not reliable methods but imparting an eddy current in the target is practical.

For this project, the team has the objective of controlling the position of a frictionless moving cart using eddy current actuation; this semester was limited to one-dimensional movement, allowed by placing the cart on an air track. This paper presents the part of the project that aimed at implementing actuators that control the movement of the magnets that would cause varying forces on a plate attached with its area vector parallel to the air track. As can be seen in figures 1 and 3, the magnet in the middle, labeled in this paper as magnet 2, remains stationary, as its purpose is to induce an eddy current in the plate that is attached to the cart. The two side motors, labeled magnet 1 for the left magnet (when facing in the direction of the magnets to the cart) and magnet 3, sit atop motors that turn in response to the proximity of the cart to the magnets. Specifically, the IR sensor, as seen in figure 3, that measures the proximity of the cart, sits underneath magnet 2.

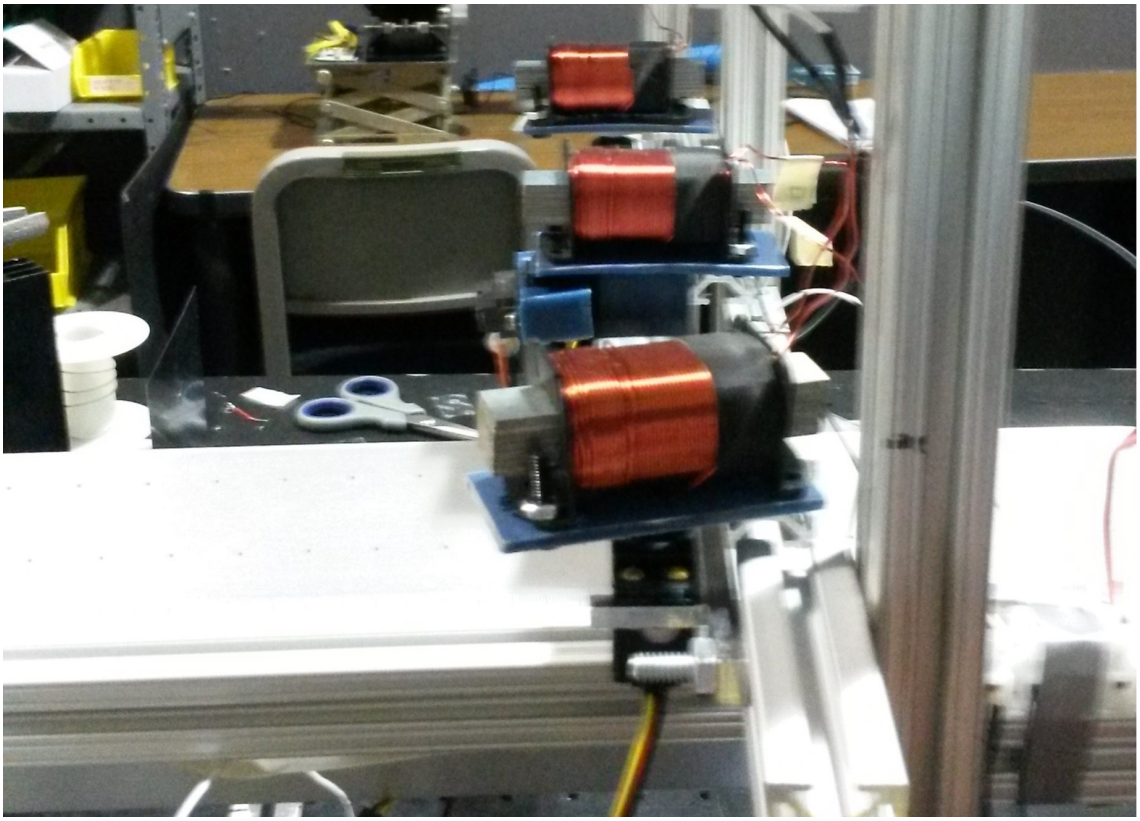

Figure 1: The test setup used for this project

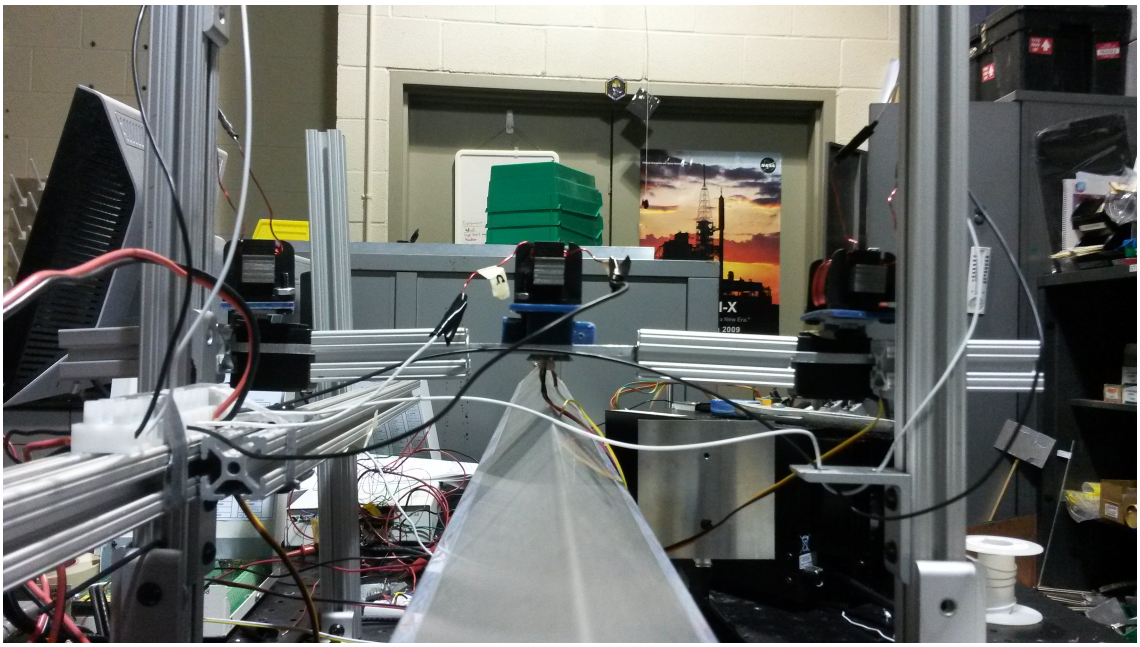

Figure 2: Axial View of Test Setup, in $-\hat{z}$ Direction

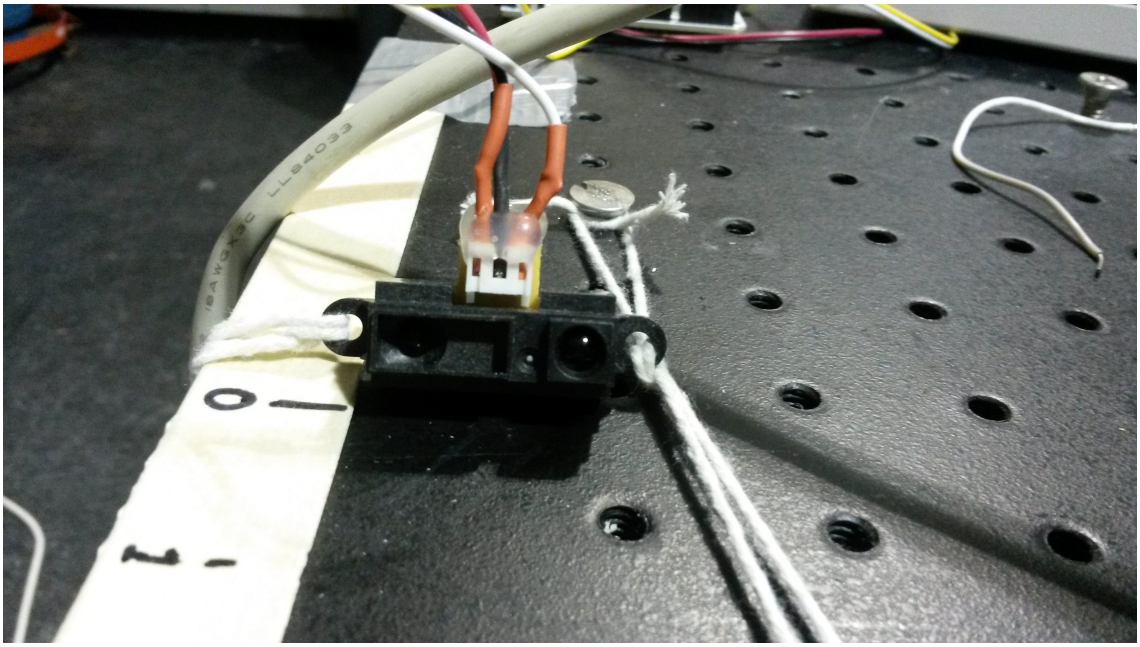

Figure 3: The IR Sensor

The end goal for this part of the project was a closed loop control of the actuators based on, possibly, several inputs, but primarily from the IR sensor. Another possibility was the integration of an accelerometer reading on the cart. This will be useful for measuring the force that the magnets impart on the cart, as well as provide another method of measuring velocity of the cart other than a discrete time derivative of the position. In an attempt

to model the plant, which currently remains almost a black box, the theory of the eddy current dynamics was worked through to provide a state space system that modeled the cart's movement based on a control input of the magnets and their positions. The control input is a result of a controller design that has yet to be implemented; this semester, the idea of a Linear Quadratic Regulator (LQR) controller was floated, due to the desire to dampen oscillations of the state and the control input. In addition, a Kalman Filter was suggested, but was taken back due to the fact that the assumptions made in the modeling of the dynamics would compound from linearization, and thus a Kalman Filter using a state space formed from these assumptions would create a fairly inaccurate estimation.

Throughout this paper are derivations of and processes of control; For those students who are to continue this work, it may be helpful to use this paper as a guide; for this audience, please note specifically the caveats and assumptions that may have to be altered.

This paper is meant mostly as a guide for future members of the project. The theory of the dynamics is given a head start, and the control theory is laid out, and thus it is hoped this will be helpful for the future of this project.

III Theory

i Eddy Current Dynamics

The nature of eddy currents is unfortunately a mysterious entity in application. The theory behind eddy currents requires the use of Lenz's Law, Ampere's Law, and the Lorentz magnetic force. However, the dynamics are not entirely known, which is why there is extensive testing required for the eddy current actuated spacecraft. Nevertheless, the theory allows for the creation of a controller based on linearized forms of the equations of motion. These equations of motion may not be entirely accurate, due to assumptions that must be made for application, and may logically be edited in the future. The linearization attempts have not been entirely successful, but the process attempted this semester should provide a good stepping stone for future members of the project.

There is only one degree of motion, and therefore the state space of the system consists of $\dot{x}$ and $\ddot{x}$ of the cart. The equation of motion for the cart is governed by the two forces of the same form, from the side magnets.

$$m\ddot{x} \;=\; I_p \, \underset{\sim}{l_p} \times \underset{\sim}{B}_1 + I_p \, \underset{\sim}{l_p} \times \underset{\sim}{B}_3 \tag{1}$$

where

$$\begin{aligned}
m &\triangleq \textit{mass of cart, including the plate atop the cart} \\
I_p &\triangleq \textit{Current in loop in plate} \\
\underset{\sim}{l_p} &\triangleq \textit{Vector following current in plate} \\
\underset{\sim}{B}_1 &\triangleq \textit{magnetic field enacting a force on eddy current loop due to magnet } 1 \\
\underset{\sim}{B}_3 &\triangleq \textit{magnetic field enacting a force on eddy current loop due to magnet } 3
\end{aligned}$$

The first step is to derive the expression for the current induced in the plate; this is derived from Faraday's Law, and continued as in the following derivation.

$$\begin{aligned}
\epsilon &= -\frac{\partial \Phi_B}{\partial t} \\
I_p &= \frac{1}{R_p}\epsilon
\end{aligned} \tag{2}$$

where ϵ is the induced EMF due to the middle magnet, and R_p is the resistance in the plate. This quantity may be difficult to quantify due to anisotropic properties of the plate, but we can assume that measuring the resistance with a multimeter will suffice.

$$I_p = -\frac{1}{R_p}\frac{\partial}{\partial t}\int\int \underset{\sim}{B} \cdot d\underset{\sim}{A}$$

The largest ambiguities in this problem are highlighted in this equation. At a specified distance, magnetic fields from a dipole magnet have several directions because there are

several field lines. The solution to this issue should be to integrate over all of the field lines, which unfortunately becomes difficult as at each position on the plate, the direction is unknown. It is due to this issue that we assume that the direction of the magnetic field is approximately anti-parallel to the area considered (the area of the current loop in the plate), as the area vector is towards the magnets and the magnetic field is from the magnets. This is the first caveat that may have to be a revised assumption. As we will see later in this derivation, this may make sense due to the direction of the magnetic fields in directions perpendicular to the axial direction may cancel out.

There also exists an ambiguity in the area of the loop; each magnetic field line should theoretically form its own loop, but the fact that there are an infinite number of magnetic field lines may allow us to assume that a large loop is formed. This is the second caveat that future work may need to edit.

$$\begin{aligned} I_p &= \frac{1}{R_p}\frac{\partial B}{\partial t}A \\ &= \frac{1}{R_p}\frac{\partial B}{\partial t}\pi r_l^2 \end{aligned}$$

where r_l is the radius of the induced current loop. Again, this is not exactly known, but assume that the radius of the current loop envelopes the width of the area of the plate, due to the second assumption that the infinite number of magnetic field lines create a large current loop.

At this point, there are two properties of electromagnets that will have to be considered. The first is that the ferromagnetic core, which has a permeability constant μ greater than 1, enhances the strength of the magnetic field at an axial distance away from the electromagnet by a factor of that permeability constant. The second is that the electromagnet's magnetic field at that axial distance is ruled by the Biot-Savart Law, given by equation 3.

$$\begin{aligned} \underset{\sim}{B} &= \frac{\mu\mu_0}{4\pi}\int I_2 \frac{\underset{\sim}{dl_2}\times \underset{\sim}{r_{p/dl_2}}}{|\underset{\sim}{r_{p/dl_2}}|^3} \\ &= \frac{\mu\mu_0}{4\pi} I_2 \int_{dl=0}^{dl=2\pi r_2}\int_{r=\sqrt{x^2+r_2^2}}^{r=\sqrt{(x+L)^2+r_2^2}} \left(\frac{\underset{\sim}{dl_2}\times \underset{\sim}{dr_{p/dl_2}}}{|\underset{\sim}{r_{p/dl_2}}|^3}\right) \end{aligned} \tag{3}$$

where r_2 is the radius of the electromagnet, $\underset{\sim}{dl_2}$ is a section of the electromagnet's surrounding wire, L is the length of the electromagnet axially, and $\underset{\sim}{dr_{p/dl_2}}$ is the distance of the plate from each $\underset{\sim}{dl_2}$. At this point, we will use a coordinate frame:

- $\hat{\theta}$ is in the direction of current in the wire coils
- $\hat{r}$ is in the direction of the radius of the electromagnet, from the center
- $\hat{z}$ is in the axial direction, $\hat{r}\times\hat{\theta}$, or from the magnets to the cart

and state that $\tilde{x}$ is the axial distance of the plate from the differential section of the coil being examined.

$$\begin{aligned}\underset{\sim}{B} &= \frac{\mu\mu_0}{4\pi} I_2 2\pi r_2 \int \left(\frac{\hat{\theta} \times (r_2\hat{r} + \tilde{x}\hat{z})}{|\underset{\sim}{r}_{p/dl_2}|^3} \right) \\ &= \frac{\mu\mu_0}{2} r_2 I_2 \int \left(\frac{-r_2\hat{z} + \tilde{x}\hat{r}}{|\underset{\sim}{r}_{p/dl_2}|^3} \right)\end{aligned}$$

Here we can see that if we integrate over the circumference of the electromagnet (as was done by pulling out the term $2\pi r_2$, the radial direction of this equation cancels out. This upholds the first assumption made above, and thus this may be affirmed. However, this takes in another assumption- that the distance is simplified to the distance to the center of plate instead of that from the electromagnet wire section to the induced loop in the plate.

$$\begin{aligned}\underset{\sim}{B} &= \frac{\mu\mu_0}{2} r_2 I_2 \int_x^{x+L} \left(\frac{-r_2\hat{z}}{(r_2^2 + \tilde{x}^2)^{1.5}} \right) \\ &= -\frac{\mu\mu_0}{2} r_2^2 I_2 \left(\frac{1}{r_2} \frac{\tilde{x}}{\sqrt{r_2^2 + \tilde{x}^2}} \right) \Bigg|_x^{x+L} \hat{z} \\ \underset{\sim}{B} &= -\frac{\mu\mu_0}{2} r_2 I_2 \left(\frac{x+L}{\sqrt{r_2^2 + (x+L)^2}} - \frac{x}{\sqrt{r_2^2 + x^2}} \right) \hat{z} \end{aligned} \tag{4}$$

Because this comes out to be in the axial direction, the current in the plate comes out to be, after deriving equation 4, given that we apply a sinusoidal current through the electromagnet coils,

$$\begin{aligned} I_p = -\frac{\pi r_l^2 \mu\mu_0 r_2}{2R_p} \Bigg(&\Big(\frac{x+L}{\sqrt{r_2^2 + (x+L)^2}} - \frac{x}{\sqrt{r_2^2 + x^2}}\Big)\dot{I}_2 + \\ &\Big(\frac{r_2^2}{(r_2^2 + [x+L]^2)^{1.5}} - \frac{r_2^2}{(r_2^2 + x^2)^{1.5}}\Big) I_2 \dot{x} \Bigg) \end{aligned} \tag{5}$$

The next step, deriving the formula for $\underset{\sim}{l}_p$, relies on the assumption concerning the area of the loop. Because we assume that the loop covers the width of the plate, which we do arbitrarily label this loop radius as r_l, we can write this loop as the following in the $\hat{\theta}$ direction, as the plate is consistently in the same $\hat{\theta}$ direction as in the electromagnet.

$$\underset{\sim}{l}_p = 2\pi r_l \hat{\theta} \tag{6}$$

The final step to writing out equation 1 is equations for the magnetic fields from the side electromagnets ($\underset{\sim}{B}_1$ and $\underset{\sim}{B}_3$). We can start from equation 3, and representing both magnetic

fields by a subscript i, as the symmetry allows us to write the equations for both magnets the same. The integration that allows equation 4 to form is executed from $\left(\frac{x}{cos\phi_i}\right)$ to $\left(\frac{x}{cos\phi_i}+L\right)$, where ϕ_i is the angle from the $\hat{z}$ direction that the ith magnet is pointed. Note that, because the electromagnets being used are the same, $r_i = r_2$. Also note that because the magnets are tilted, we introduce $\hat{y}$, the direction perpendicular to the air track, and along the vector that connects all of the magnets. This exists only show the entirety of the $\underset{\sim}{r}$ vector, as, like the $\hat{z}$ direction part of this vector, it cancels out due to the symmetry of the two magnets on either side of the middle magnet 2.

$$
\begin{aligned}
\underset{\sim}{B_i} &= \frac{\mu\mu_0}{4\pi}\int I_i \frac{\underset{\sim}{dl_i}\times\underset{\sim}{r_{p/dl_i}}}{|\underset{\sim}{r_{p/dl_i}}|^3} \\
&= \frac{\mu\mu_0}{4\pi} I_i \int_{dl=0}^{dl=2\pi r_i}\int_{r=\sqrt{x^2/cos^2\phi_i+r_i^2}}^{r=\sqrt{(x/cos\phi_i+L)^2+r_i^2}} \left(\frac{\underset{\sim}{dl_i}\times\underset{\sim}{dr_{p/dl_i}}}{|\underset{\sim}{r_{p/dl_i}}|^3}\right) \\
&= \frac{\mu\mu_0}{4\pi} I_i 2\pi r_i \int \left(\frac{\hat{\theta}\times(r_i\hat{r}+\tilde{x}\hat{z}+x\,tan(\phi)\hat{y})}{|\underset{\sim}{r_{p/dl_i}}|^3}\right) \\
&= \frac{\mu\mu_0 r_i I_i}{2}\int_{\tilde{x}=x^2/cos^2\phi_i}^{\tilde{x}=x/cos\phi_i+L} -\frac{r_i}{(r_i^2+\tilde{x}^2)^{1.5}}d\tilde{x}\hat{z} \\
&= \frac{\mu\mu_0 r_i I_i}{2}\left(\frac{\tilde{x}}{\sqrt{r_i^2+\tilde{x}^2}}\right)\Bigg|_{x/cos\phi_i}^{x/cos\phi_i+L}\hat{z} \\
&= \frac{\mu\mu_0 r_i I_i}{2}\left(\frac{x/cos\phi_i+L}{\sqrt{r_i^2+(x/cos\phi_i+L)^2}}-\frac{x/cos\phi_i}{\sqrt{r_i^2+x^2/cos^2\phi_i}}\right)\hat{z} \qquad (7)
\end{aligned}
$$

The combination of all of these three parts gives us the total force imparted on the cart by the magnets. As we can see, this needs refinement, as the end of the work done on this process shows a peculiar direction from the cross product. It is probable that the issue lies in the derivation of equation 7.

$$
\begin{aligned}
m\ddot{x} = -\frac{\pi^2 r_l^3\mu\mu_0 r_2}{R_p}\Bigg(&\left(\frac{x+L}{\sqrt{r_2^2+(x+L)^2}}-\frac{x}{\sqrt{r_2^2+x^2}}\right)\dot{I}_2+ \\
&\left(\frac{r_2^2}{(r_2^2+[x+L]^2)^{1.5}}-\frac{r_2^2}{(r_2^2+x^2)^{1.5}}\right)I_2\dot{x}\Bigg)\hat{\theta}\times \\
&\sum_{i=1,3}\frac{\mu\mu_0 r_i I_i}{2}\left(\frac{x/cos\phi_i+L}{\sqrt{r_i^2+(x/cos\phi_i+L)^2}}-\frac{x/cos\phi_i}{\sqrt{r_i^2+x^2/cos^2\phi_i}}\right)\hat{z} \qquad (8)
\end{aligned}
$$

ii Feedback Loop

As can be seen in equation 8, linearization of the dynamics is a hurdle. Because the controller synthesis was not reached yet, this process was neglected temporarily. For the students

who will take over this part of the project, it is important to note that the linearization must be done as follows.

$$
\begin{aligned}
f &= m\ddot{x} \\
\dot{z} &= Az + B_u u \\
y &= Cz
\end{aligned}
$$

where

$$
\begin{aligned}
z &= \begin{bmatrix} x \\ \dot{x} \end{bmatrix} \\
u &= \begin{bmatrix} \phi_1 \\ \phi_3 \\ I_2 \\ I_1 \\ I_3 \end{bmatrix} \\
y &= x \\
A &= \begin{bmatrix} 0 & 1 \\ \dfrac{df}{dx} & \dfrac{df}{d\dot{x}} \end{bmatrix} \\
B_u &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ \dfrac{df}{d\phi_1} & \dfrac{df}{d\phi_3} & \dfrac{df}{dI_2} & \dfrac{df}{dI_1} & \dfrac{df}{dI_3} \end{bmatrix} \\
C &= \begin{bmatrix} 1 & 0 \end{bmatrix}
\end{aligned}
$$

These plant dynamics are taken care of by the actual physical system, nonlinearized, but these state space matrices are used to generate the controllers.

iii Observer-Based Control

The linearization of the equations of motion is necessary for the feedback loop, as the observer-based controller needs a linearized system to send an estimation to the code. This method was considered at first due to the fact that the IR sensor gave a noisy measurement, seen through the serial monitor in the compiler of the microcontroller (Arduino). This was also seen physically in the initial linear motor control (angle of the motors linearly related to the distance signal from the IR sensor), which showed that at a constant distance, the motors would wobble at fairly high frequencies, although fortunately not at high amplitudes. The solution to this issue was placing a low pass filter in the simulink code. Because of the heavily assumption-dependent linearization, it was considered unwise to utilize the observer-based control to estimate the state at the next time step.

However, if future members of the project were to generate more reliable plant dynamics, they should be aware of Kalman Filtering, as seen in figure 4. The estimate is output as $\hat{x}$, which is then fed into the LQR controller law $u = K_r\hat{x}$.

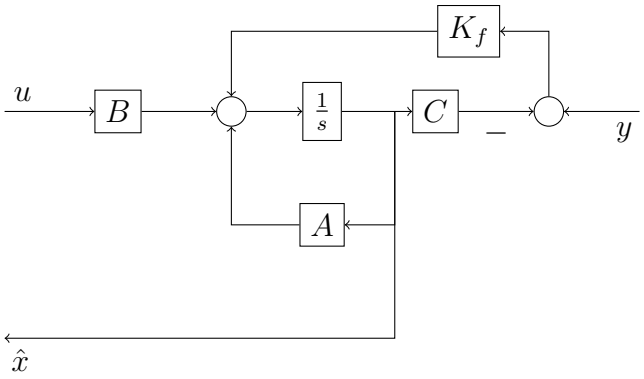

Figure 4: Kalman Filter

iv Control Based on Cost

The LQR controller, as briefly mentioned above, would be fed the estimate (or in case the Kalman Filter is not used) the measured position and derived velocity of the cart. The LQR controller is designed around the idea of minimizing cost J, which is defined as

$$J = \int_0^\infty x^T Q x + u^T R u \tag{9}$$

where Q and R are weightings, dependent on how much each cost (the wiggle of the motors and fluctuation of the currents in u^2, or the error in the state from desired state in x^2) is worth.

The controller K_r can be synthesized using the MATLAB command *lqr*, or MATLAB can be used to optimize a set of Linear Matrix Inequalities.

$$\begin{bmatrix} YA^T + AY - BR^{-1}B^T & YC^T \\ CY & -1 \end{bmatrix} \leq 0$$

$$\begin{bmatrix} \gamma & z_0^T \\ z_0 & Y \end{bmatrix} \geq 0$$

where γ is the value of the cost J, and Y is used to develop the LQR controller. For theoretical understanding, this matrix is defined as $Y = X^{-1}$, where X is variable in the Lyapunov function V:

$$V = x^T X x, \text{ , where}$$
$$\dot{V} < 0 \text{ for stability}$$

And the optimal LQR controller is synthesized as

$$K_r = R^{-1} B^T X$$

IV Method

i Motor Choice

The beginning of the semester was dedicated to determining best selection of motors for actuation of the magnets' directions. This was not a difficult task, but it was very important for several reasons. The first was ease of control; spending a large amount of time or resources determining how to control the actuator would be a waste of time. Second was cost; this test platform is meant to be reproduced, so if an inexpensive motor meets the criteria for the project, there is no reason not to use it over a more expensive one. Third was method of measurement; in order to ensure that the motors could be used in closed loop, a simple sending of a signal to a motor will not be useful. This is because the repulsion and attraction of a plate holding an eddy current is dependent on the angle at which the magnets are turned. Without a measurement of the angle at which the magnets are pointed, there is no way of determining force on the eddy current. A hobbyist servo met all of these conditions, and thus two were to be utilized to hold and direct the two side motors. The servo chosen, a HS-422, was appropriate because a commanded PWM signal corresponded to a given angle at which it would turn, due to the internal h bridge in servos.

Figure 5: Motor Chosen: HS-422 Hobbyist Servo

ii Construction of Test Setup

There are many factors that go into the force placed on the eddy currents by the side magnets. These include the perpendicular distance from the axis of the air track, but it seemed reasonable to place the magnets at the same distance at which they were before the magnet directions were actuated, and when they were only movable by hand. In the magnet slot drawing below, I decided not to cut a hole in the middle and instead create three walls that would hold the motor. This makes the machining easier and the fit of the motor more snug. These drawings are seen in figures 6-9.

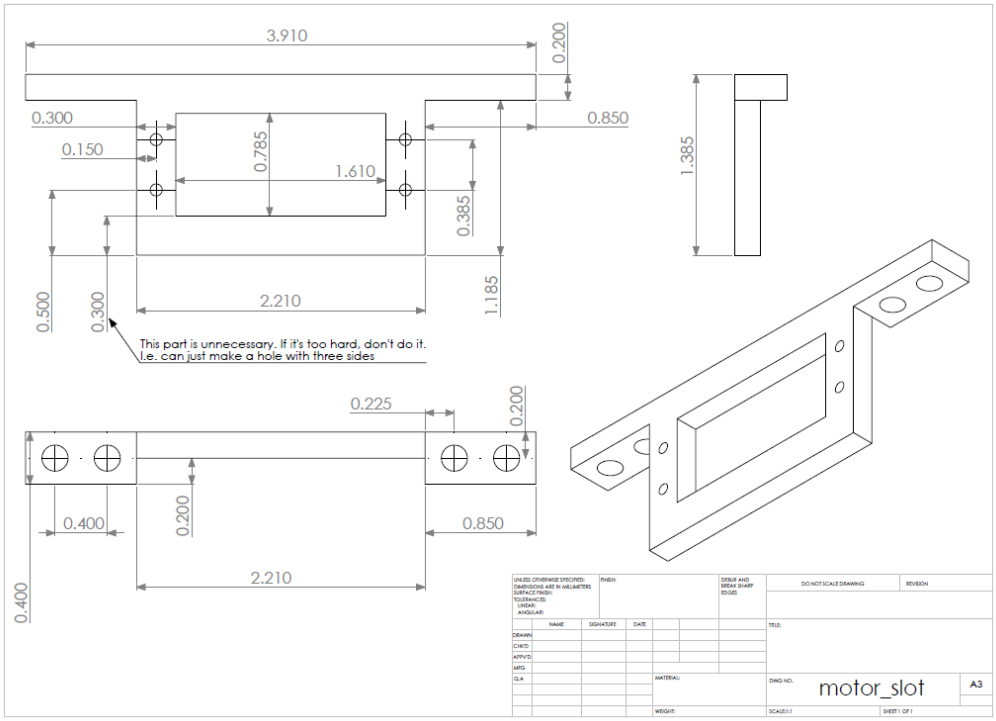

Figure 6: Slot Holding Side Magnets

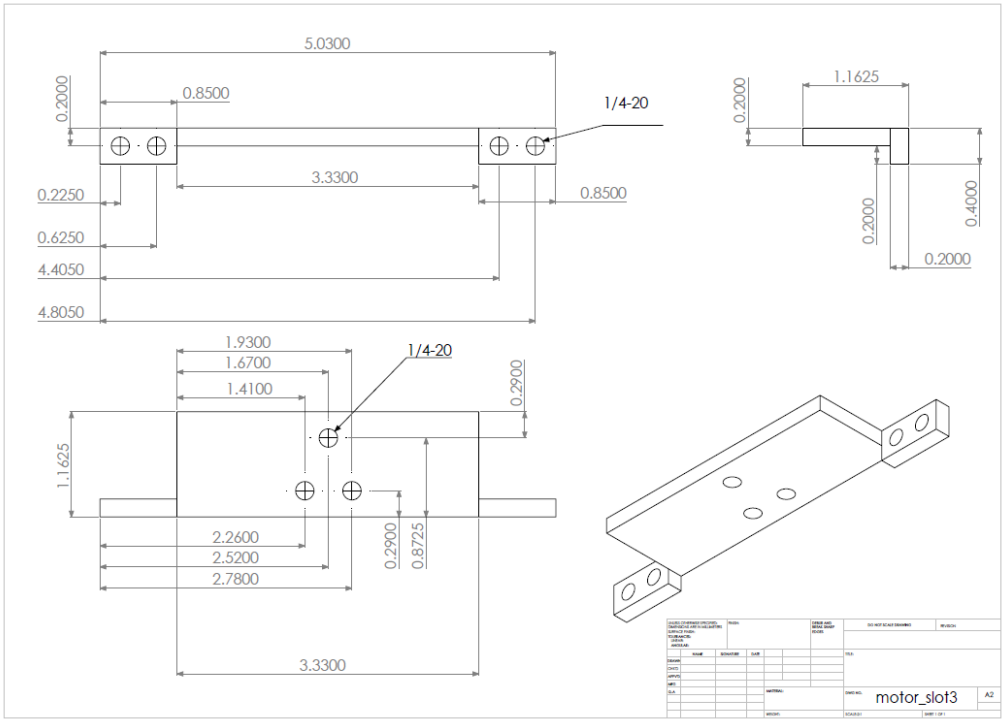

Figure 7: Platform Holding Middle Magnet

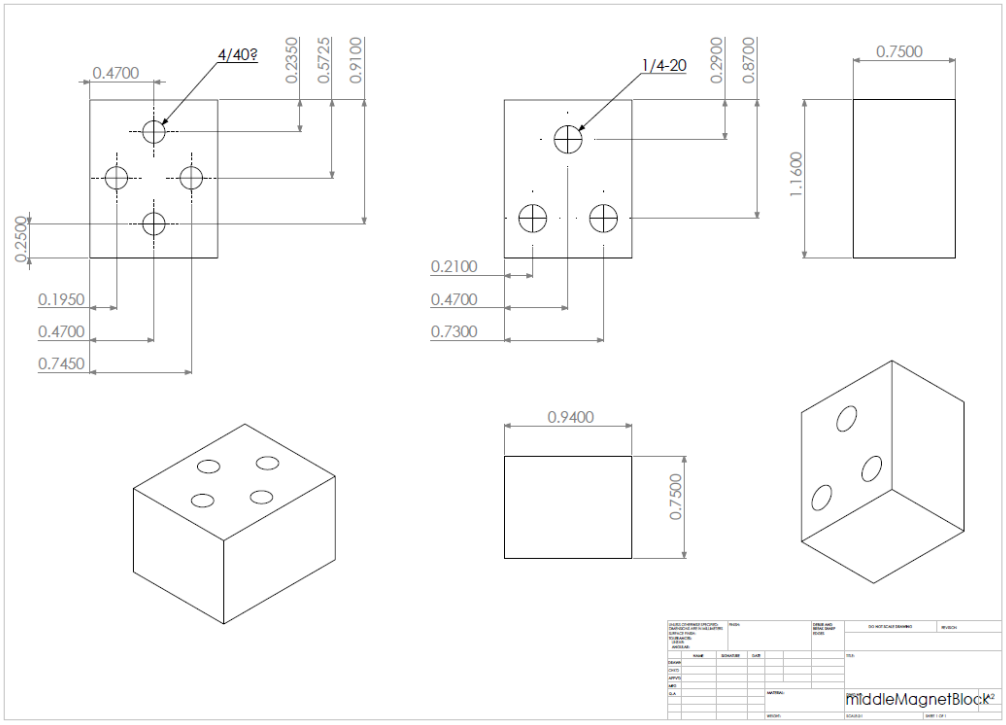

Figure 8: Block that sits atop Above Motor Platform

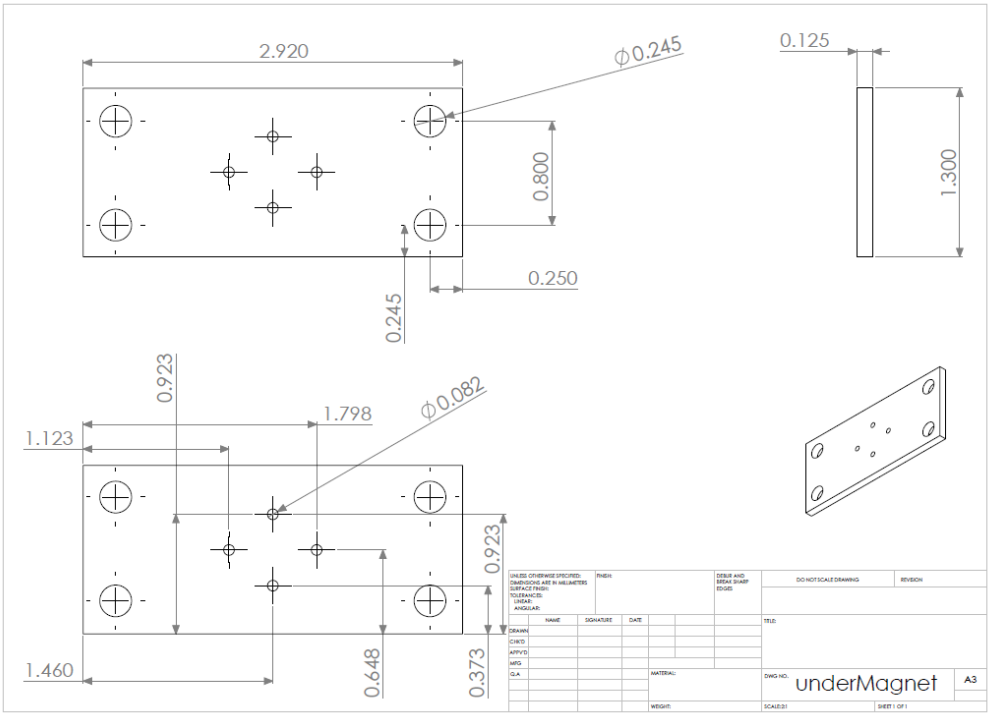

Figure 9: Platform Directly Holding Magnets

iii Motor Control Overview

A large part of the semester was also spent on controlling the motor, to prepare its use with the PCI 6259 data acquisition card. This was done initially with an Arduino Leonardo in four steps.

1. Use a potentiometer to control the direction of a single motor; full turn of the potentiometer one way turns the motor to zero degrees and full turn of the pot the other way turns the motor to 180 degrees

2. Use a potentiometer to control the direction of a single motor by connection to the SCB box (through simulink via the PCI card)

3. Use IR sensor to control the direction of a single motor. This is trickier than using a potentiometer because of the nonlinear nature of the measurements taken by an IR sensor. Between one and three inches from the sensor, the analog signal output has a higher range than at any other distance. Under one inch the signal is unreliable, and farther than approximately three inches the signal decreases noticeably in range.

4. Use the IR sensor, motors, and cart in the loop- the measurements taken by the IR sensor, the control input being calculated by the Simulink model, the control input output by the motors and current through the electromagnets, and the plant in the cart on the air track

The potentiometer-controlled motor code from parts 1 and 2 is shown below.

```
/*
  pot2servo
  control position of servo by potentiometer turn
*/

#include <Servo.h>

Servo hs422; // creates object that is a servo

void setup(){
  Serial.begin(9600); //set comm b/w Ard&comp to 9600 bits/sec
  hs422.attach(7); // attaches servo to digital pin 9
  pinMode(13,OUTPUT);
}

void loop(){
  digitalWrite(13,LOW);
  int potValue= analogRead(A0); // get value of potentiometer
  //potentiometer 0 to 1023; servo 0 to 180
  float servoValue= potValue * 180.0/1023.0;
  //int pos= round(servoValue); // cast servoValue float to pos int
```

```
  float pos= servoValue;
  hs422.write(pos);
  Serial.println(pos);
  digitalWrite(13,HIGH);
  delay(10); //pause 30 ms
}
```

iv Motor Control in the Loop

Because the dynamics were not fully figured out, there was no actual controller synthesized. Hence, the input to output of the code was not finished, and instead the following code was developed to attempt a linear control; that is, angle of the side magnets was linearly dependent on the distance of the cart from the IR sensor. This required linear interpolation, as in the following code.

```
/*
  SCB2servo
  control position of servo by analog output from SCB box from
  simulink model: value is 0-5 in volts
*/

#include <Servo.h>

Servo hs422; // creates object that is a servo

void setup(){
  Serial.begin(9600); //set comm b/w Ard&comp to 9600 bits/sec
  hs422.attach(7); // attaches servo to digital pin 9
  pinMode(13,OUTPUT);
}

//float IRdist[] = {3.5,4.0,5.0 ,6.0,7.0 ,8.0 ,9.0 ,10.0,12.0,14.0,16.0};
//float IRvolt[] = {3.0,2.7,2.35,2.0,1.75,1.55,1.40,1.25,1.05,0.9 ,0.8};
float IRsig[] = {525.833,350.0,267.5,230.833,205.0,180.385,160.417,143.75};
float IRdist[] = {1.0,   2.0,   3.0, 4.0,    5.0,   6.0,   7.0,    8.0};

void loop(){
  digitalWrite(13,LOW);
  float cmdValue= analogRead(A0); // get value
  //IR from... 50... to ... 550?
  float IR_out = interp(cmdValue);
  float servoValue= (IR_out-1.0) * 180.0/7.0; // replace w/ 5?... cmdValue
  //int pos= round(servoValue); // cast servoValue float to pos int
  float pos= servoValue;
```

```
  hs422.write(pos);
  Serial.println(servoValue);
  digitalWrite(13,HIGH);
  delay(100); //pause 10 ms
}

float interp(float cmdValue){
  //interpolation based on cmdValue and IR signal array of
  //recorded data (IRsig)
  //interpolation returns distance value for loop to perform
  //its own interpolation
  //  to give the servo an appropriate angle
  //IR_out (voltIRout) is a distance out
  int indic = 0;
  int indx = 1;
  while(!indic && indx<11){
    float compValue = floor(cmdValue/IRsig[indx]);
    if (compValue > 0.0){
      indic=1;
    }
    else {
      indx ++;
    }
  }
  float voltIRout = ((IRdist[indx]-IRdist[indx-1])/
    (IRsig[indx]-IRsig[indx-1]))*(cmdValue-IRsig[indx-1])
    +IRdist[indx-1];
  return voltIRout;
}
```

In addition, in order to control the current that runs through the electromagnets, the following code was developed, also controlling the motor angle but this time by choosing a linear relation to the signal and not the actual distance (for simplicity).

```
/*
  IR2SCB2servo
  control position of servo by analog output from SCB box from
  simulink model: value is 0-5 in volts
*/

#include <Servo.h>

Servo leftMagnet; // creates object that is a servo
Servo rightMagnet;
```

```
int proxFlag = 4;

void setup(){
  Serial.begin(9600); //set comm b/w Ard&comp to 9600 bits/sec
  leftMagnet.attach(7);
  rightMagnet.attach(8);
  pinMode(proxFlag,OUTPUT);
}

void loop(){
  int cmdValue= analogRead(A0); // get value

  float servoValue= 180.0-cmdValue * 90.0/682;
  //float servoValue= cmdValue * 90.0/512.0 + 90.0;

  float rightPos= servoValue;
  float leftPos= 180.0-servoValue;
    //float rightPos= servoValue;
  //float leftPos= 180.0-servoValue;

  rightMagnet.write(rightPos);
  leftMagnet.write(leftPos);
  //Serial.println(cmdValue);

  //if too far, low so that current runs opposite direction
  if (cmdValue<341) {
    digitalWrite(proxFlag,LOW);
  }
  else {
    digitalWrite(proxFlag,HIGH);
  }
  Serial.println(cmdValue);
  delay(100); //pause 100 ms
}
```

The simulink code used is seen in figure 10, utilizing the PCI 6259 data acquisition card to take an analog signal from the IR sensor and output a current signal to the electromagnet coils. The PWM signal was handled by the Arduino due to the high current draw of the servos, too high for the specifications of the PCI card. However, code was developed for outputting analog signal that can be linearly interpolated by the Arduino for the motor PWM signal. This would be useful in order to include the motor control input into the loop, integrated in the control input array u. The signal sent to the Arduino would just have to be scaled to a value that the microcontroller could interpolate PWM signals from.

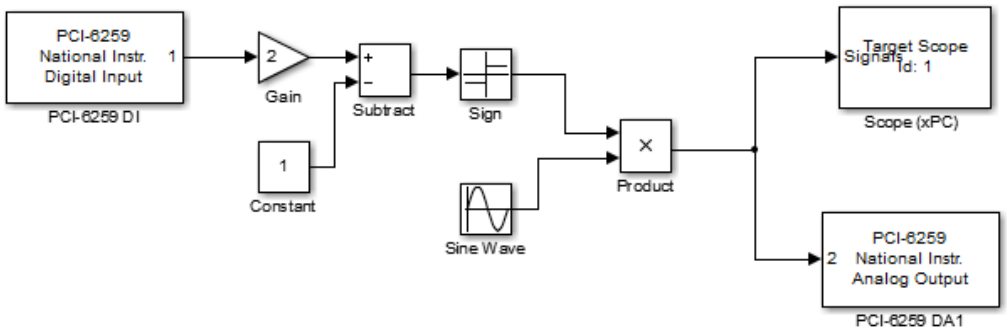

Figure 10: Simulink Code

v Hardware and Software Used

The HS422 servos have been mentioned in the first part of the Methods Section; they draw 140 mA maximum, although rated to 150 mA. This is too much to draw from the PCI 6259 data acquisition, which is why the direct motor control is handled by the Arduino, the pins on which draw only 40 mA. The motor control input is calculated and then converted to a 0-5V signal by the Simulink code, and is sent through an analog output pin on the SCB-68 box, a data acquisition input/output box, to an analog in pin on the Arduino. The control input is calculated in Simulink through the synthesized controller, not yet utilized, from the distance and velocity measurement signals interpolated to an actual distance in the block diagram. As well, this calculation creates, through the controller, the current control inputs.

This is all performed using XPC Target, a toolbox on Simulink that allows a simulation to be run in real-time. The code is compiled and sent to a target computer, which is hooked up the PCI 6259 card, which in turn is connected to the SCB box. The target computer runs the code until its commanded stop time at the actual sampling time given in the block diagram (0.00025 seconds), instead simulating a response for the code time length.

V Conclusion

Although no test results were accomplished, the setup has been finished; the methods of control with no synthesized controller (instead, a simple algebraic relation between the motor angles and the distance from the IR sensor) is complete. Although the scope of the project was not completely seen through, the one-dimensional control of a cart using eddy current actuation has two steps left. First, the accurate modeling of the dynamics will cause one to linearize the plant, and thus allow for controller synthesis using the theory described in this report. Second, testing of this controller will affirm this controller synthesis and system dynamics, and thus the project can be moved on to two dimensional control.

VI Appendix: Take Away for Future Members of the Project

i Motor Control

It is important to note that a servo cannot hook up to an h bridge because the servo already has an h bridge in its own internal circuit. The wires that hook up to control the servo cannot sensibly connect to the ports of an h bridge. The VCC of the servo should always be in the range given in its data sheet, and the PWM signal to the servo is actually controlled already in the circuitry of the motor. Attempting to send a PWM signal to the in wire of the servo will do nothing useful.

ii XPC Target

There exists documentation on XPC Target on the trello website, and possibly as well will be included in the box folder. It is important to note that one cannot expect to be able to run a Simulink diagram in real time on the target machine just from a simple compilation, as the model parameters must altered to particular settings.

iii Plant Linearization

The section III.i gives the process for finding the actual plant dynamics for the force on the cart, but this may be inaccurate. If any future project members attempt this process, this section may be helpful despite inaccuracies in assumptions or calculations. The linearization process is outlined in III.ii.

iv Controller Synthesis

If the plant dynamics are accurately determined and linearization is possible, a LQG controller (LQR plus a Kalman Filter) should be utilized. For this purpose, either use the CVX toolbox for MATLAB (in order to set the LMIs shown in III.iv) or use internal MATLAB commands that give you the controller directly.

v Noisy Measurements

It is expected that the measurement devices (sensors) will be inaccurate. Before placing a filter on the signal (which is done by a block in Simulink), use an oscilloscope. In fact, the oscilloscope is the most useful tool in the lab; utilize it often.

vi Machining

If a project member comes onto the team and stays for two semesters, beware the spring semester, as MAE 2250 students have priority in the machine shop for extended time periods

during this semester. In addition, working with plastic is painful; if something must be machined for permanence, use aluminum; if something must be made quickly, choose Plexiglas or some other plastic.

vii PCI Card

Note that the PCI 6259 card has inadequate documentation, which does not give the maximum current that can be drawn from it, via the SCB box for our purposes. It is thus that you must be extremely careful when deciding to control any external hardware, checking the specs beforehand as well as using a multimeter. It is, however, definitely safe up to 40 mA, the current draw that the Arduino's pins require.

VII Acknowledgments

I would like to thank:

Cornell University for providing the resources for the lab and lab space

Mason Peck and Brandon Hencey for supervising this project

and, of course, Benjamin Reinhardt, Iona Brockie, and Dan LaChapelle for their assistance with my part of this project