Thrust Vector Control (TVC) Desktop Learning Center

Kayla C. Aloyo[1]
Carnegie Mellon University, Pittsburgh, PA, 15213

Marissa A. Good[2]
Massachusetts Institute of Technology, Cambridge, MA, 02139

Krystal N. Mike[3]
Arizona State University, Tempe, AZ, 85281

and

Brian J. Williams[4]
Bradley University, Peoria, IL, 61625

Abstract

Thrust vector control (TVC) is often employed to guide rockets in the desired flight trajectory. For instance, the J2-X has two orthogonal actuators that are used to gimbal the engine nozzle and in turn vector the thrust, keeping the rocket on the flight path. Our project was to build and design a learning center that would demonstrate TVC of Ares I, as well as TVC in general. We began with a dual-axis double inverted pendulum mechanical set-up that was developed by a group of interns last year. The set-up was composed of two linear voice-coil actuators attached to a rod which acted as a lever arm through which we applied force to the model rocket in order to balance it. The first step in our project was to develop a mathematical model to characterize the dual-axis double inverted pendulum system using Lagrangian mechanics. The model then fed into a linear quadratic regulator controller. From this model we determined the measurements necessary to control the system, which were obtained via two tri-axial rate gyroscopes, two potentiometers, and two linear encoders. The measurements taken from these instruments were incorporated into a LabVIEW VI which calculated the necessary force to be sent to our actuators to control the system. Using this closed-loop feedback system ensured the system was being controlled appropriately.

Nomenclature

	=	lagrangian
T	=	potential energy
V	=	kinetic energy
q	=	angle
	=	angular velocity
	=	angular acceleration
	=	torque
I	=	moment of inertia
m	=	mass
g	=	gravity
l	=	length
$\mathbf{A}$	=	state matrix
$\mathbf{B}$	=	input matrix

---

[1] Mechanical Engineering
[2] Aerospace Engineering
[3] Mechanical Engineering
[4] Mechanical Engineering

Introduction

Thrust Vector Control (TVC) systems stabilize rockets by manipulating thrust direction to guide rockets in the desired flight trajectory. When a rocket is launched, it faces immediate instability due to its design as an inverted pendulum, an inherently unstable system. However in order for a rocket to successfully reach its destination, this instability needs to be countered. Thus a control system is applied to the rocket to keep it on the flight path. There are several types of TVC systems that can be applied to control the rocket trajectory. Our model was based on one type of TVC, gimballing, by which the motion of the rocket is controlled by gimballing the rocket nozzle(s), and in turn vectoring the thrust. This keeps the rocket on the desired flight trajectory.

Our task for the summer was to create a desktop learning center that would demonstrate the principles of TVC. We developed a learning center based off of a dual-axis double inverted pendulum mechanical system (Block 1996). This mechanical system is physically complex and requires the application of a robust control system in order to maintain proper control. The mechanical system began with the model rocket, an Ares I, which was attached by a ball bearing to a rod below. This rod was then attached to two orthogonal linear voice-coil actuators. Several measurement devices were also added to the system: two tri-axial rate gyroscopes, two potentiometers, and two linear encoders. A DAQ board was used to collect data from the various measurement devices. This data was then fed into LabVIEW, where the raw data was converted into usable information. We established a mathematical model of the system using Lagrangian mechanics, derived from the principles of conservation of momentum and conservation of energy (Spong 1994). Once the mathematical model was qualified, linearization techniques were utilized to decouple the two axes of the system. By separating the two axes, we were able to vastly simplify the mathematical model as it is far easier to control two separate axes when compared to two coupled axes. We applied Linear Quadratic Regulator (LQR) control to the system which determined the gains of the system and helped to ensure stability. Using the data from the measurement devices as the inputs to the control system, we attempted to control the unstable system.

I. System Model

The TVC demonstration device was defined as a spherical double inverted pendulum, with one active joint on the driving arm and one passive joint at the rocket. A diagram of the device is as shown in Fig. 1.

The assumption was made that the spherical motion of the pendulum could be separated into two orthogonal planar pendulum systems that could be controlled independently. This assumption was justified by a literature review and by examining the underlying equations of motion for the system (Rong 2000). Lagrangian mechanics were used to examine the behavior of the system, by considering the potential and kinetic energies of the system. Because the total kinetic and potential energies could be mathematically described separately along the orthogonal planes, we were able to separate the system motion into two planar systems. The equations of motion for each planar pendulum system were identical along the two orthogonal coordinate planes.

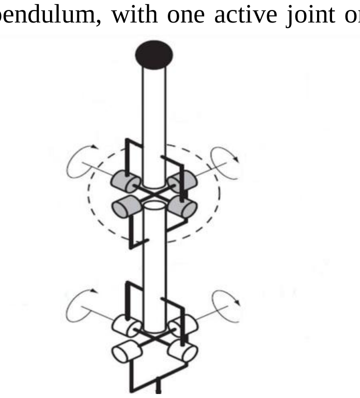

Figure 1. System Model.

A. Mathematical Model

The equations of motion were derived for the system using Lagrangian Mechanics. The Lagrangian Method is a study of the energy state in a dynamic system, which examines the difference in potential and kinetic energy in the system. This method, compared to the Newtonian Method, is advantageous for this application because it allows arbitrary coordinates to be selected which reduces the complexity of equations. Also the primary drivers of system dynamics can be observed directly, making the Lagrangian a more efficient approach for the TVC system model. These two factors were the reasons the Lagrangian method was chosen over Newtonian Mechanics.

To derive the equations of motion for the planar system, the generalized coordinates were selected. These coordinates needed to define the system uniquely; that is, the generalized coordinates needed to completely describe the motion of the system. The coordinate system of the simplified 2-dimensional case is defined in Fig. 2.

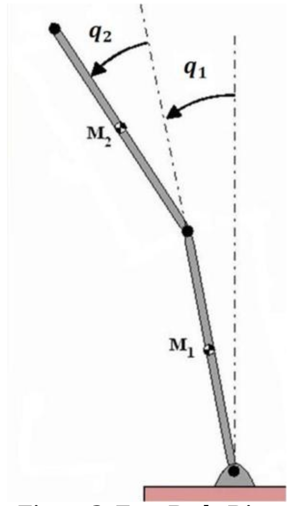

Figure 2. Free Body Diagram

The Lagrangian is given by Eq. (1).

(1)

In order to determine the Lagrangian, the total potential and kinetic energies of the system were determined. Using this method, the Lagrangian was found and put into Eqs. (2-3).

(2)

(3)

The equations of motion for the system were then computed by Eqs. (4-5).

(4)

(5)

However, in order to analyze and ultimately control the motion of the system, the equations needed to be decoupled. The acceleration terms, and , were separated using matrix algebra to find equations that uniquely identify and . In order to use matrix algebra, Eqs. (4-5), were changed into the matrix form given by Eq. (6).

(6)

Where

To decouple and , the matrices were algebraically manipulated by Eq. (7).

(7)

With and known, the system equations of motion were in the proper form to be inserted into a system model.

B. System Model Development

The choice was made to develop a linear controller for the TVC demo system. A linear controller was developed because similar researched systems have been controlled by linear controllers. The operating envelope of the TVC device was limited to 10 degrees off the zenith axis. As a result, the small-angle approximation applied to this case allowed a linear controller to be used. A larger operating envelope would require the use of nonlinear controllers, but the added complexity was not justified for this case.

The system was modeled in state space due to its flexibility in handling a multi-input, multi-output system (Block 1996). Modeling in the time domain allowed nonlinearities and nonzero initial conditions to be easily adapted to a controller model. The state variables were chosen as follows.

These particular state variables were chosen because they followed the generalized coordinates chosen for the system and all four states could be directly measured by the apparatus. Once the equations of motion were written in terms of the state variables, they could be written in terms of the linear state space form, given by Eq. (8), where **A** describes the system dynamics and **B** describes the torque input effect on the system.

(8)

The linearized **A** and **B** matrices were evaluated at the operating point, where $-$, to arrive at the final state space form. After **A** and **B** were evaluated, the linear state space model for one plane of the TVC system was defined by the following matrices, Eq. (9).

(9)

Calculating the eigenvalues of the A matrix showed positive poles, which indicated the system was unstable and required a controller to establish stability.

II. Controller Design

The specific controller chosen for the TVC system was the Linear Quadratic Regulator (LQR) controller. The LQR controller was selected due to its robustness as it can respond easily to disturbances and allows multiple inputs and outputs for the system. The inputs were both angle and angular velocity of the rod and the rocket. One assumption made in our LQR controller was that the states given by **x** could be directly measured and thus could be controlled. By using LQR, a gain matrix, K, was obtained and used to determine the necessary output torque, , that would be applied to the arm by the actuator to return the system to the desired operating point as shown in Eq. (10).

(10)

The parameters that were adjusted in LQR were the weighting matrix, **Q**, and the controller gain, R. The weighting matrix allows the engineer to identify which state variables to control and what priorities to place on each. The controllability gain determines the response of the system and how much controller effort is used to balance the rocket. The LQR function in MATLAB uses the steady state **A** and **B** matrices from equation 9 to calculate the feedback gain matrix, **K,** which stabilizes the system.

Figure 3 describes the control system using Simulink. Simulink integrates the system model with the controller, and simulates the system's response to adjustments in the weighting matrix or controller gain. This allows the engineer to visually see the model working to balance the rocket when an impulse input is applied to the system.

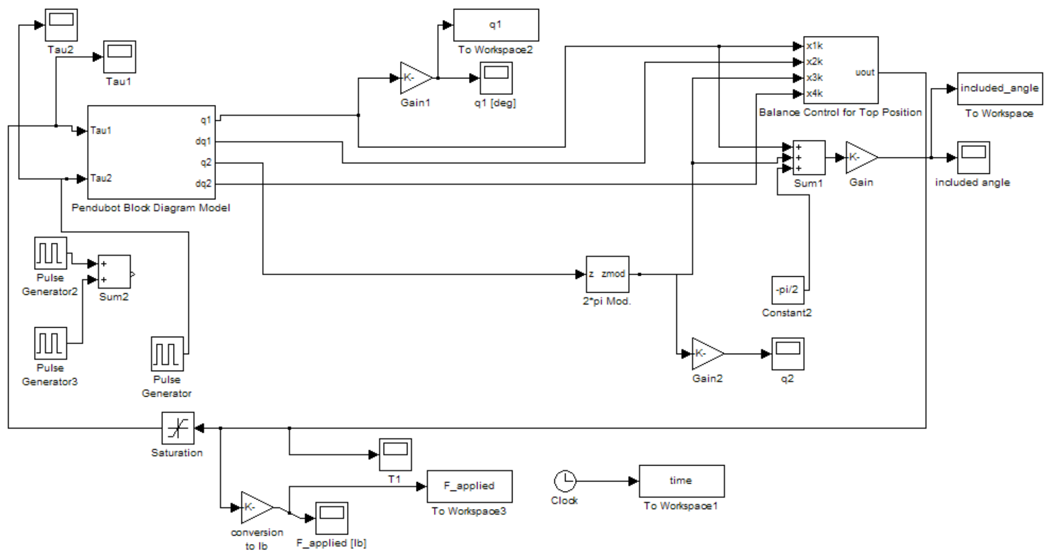

Figure 3. Simulink Model of System

Another MATLAB code was written to iterate over various combinations of **Q** and R to determine the optimal gain matrix for the controller. The **Q** matrix was adjusted by creating the weighting variable z and a magnitude variable m. These variables allow the **Q** matrix to change in a systematic manner during testing. One method used to determine the best combination was to vary one variable while keeping all others constant. Once the best variable was selected for that trial, another variable was tested with the others constant. This process was repeated until the best combination was found. To find the best combination, we needed to simulate a known disturbance torque. The criteria for best selection of variables were peak displacement, settling time, and maximum force required to stabilize the rocket.

In addition to finding the best combination, there were a few mechanical restrictions on the system. For instance, the angle of the rod q_1 was initially set at 90 with the rod along the vertical axis. Also q_1 could not exceed 107 or fall below 73 . The controller gain R could range between zero and infinity, and the actuators were limited to a peak force of 9.5 lb and a constant force of 3.2 lb. Table 1 and Fig. 4 show the results of an example where Z is 0.01, m is 0.5, and R is 2000. All of the requirements were met and therefore this combination of Q and R was valid as shown in Table 1. Figure 4 shows the angle of the rod q_1: the rod moves greater than -7 degrees from center and around 2 seconds it settles and the rocket converges to the vertical position.

Trial	Z (ratio) (0 < Z < 1)	m	R	Max F [lb] (Cont: 3.2, Peak 9.5)	Max Included Angle [Deg]	Max q1 [deg] (limit: 107)	Ts (within 0.05, found from graphs)
6	0.01	0.5	2000	5.5134	7.1924	103.954	2.14

Table 1. Excel Trial 6

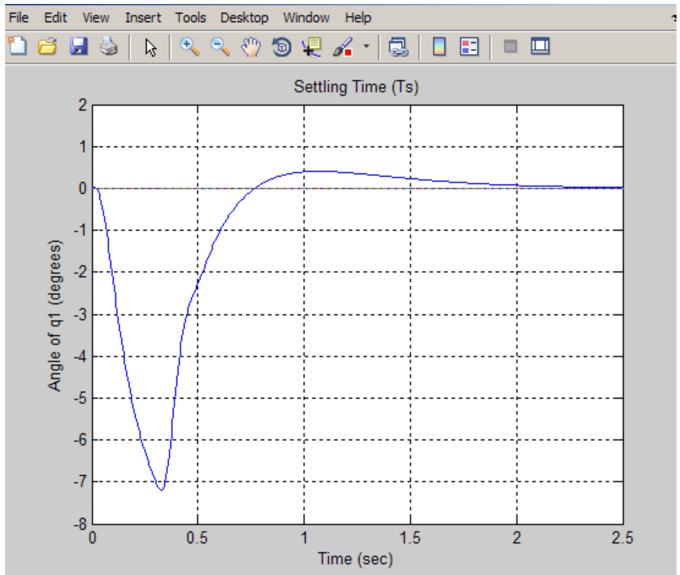

Figure 4. Plot Used to Determine Settling Time

The LQR controller proved to be a valid controller for our system. The final step was to do the actual testing and integrating with hardware. During testing, a few **K** gain matrices were tested to determine which gains would work best for our system. Figure 6 shows the control system block diagram. By using LabVIEW, closed-looped feedback was achieved.

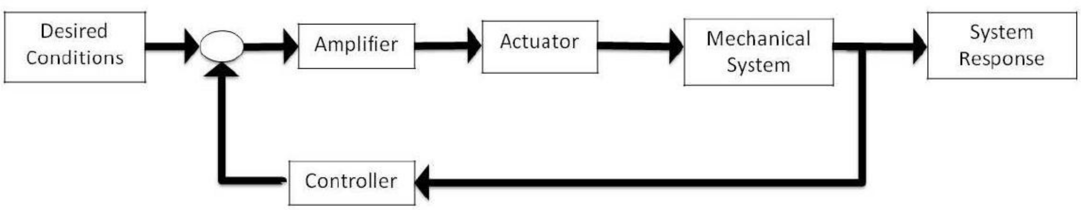

Figure 5. Control System Block Diagram

III. Physical Model

The following section contains information pertinent to the design of the physical system of the TVC desktop learning center. Discussion of the design choice, components, and set-up are all included.

A. Design Choice

This project was first assigned to a group of interns in the Summer of 2009. They played the major role in the design we selected for the desktop learning center. They considered various design options for the learning center and ultimately settled on a dual-axis inverted pendulum system controlled by two linear voice-coil actuators. Although they made substantial headway on the project, they were unable to attain full control of the system as much of their summer was spent building the physical model leaving them little time to fully develop the control system. When we were assigned this project, we first considered all of the possible designs of the system. First priority went to evaluation of the current design, as the majority of the necessary components had already been purchased or built. However we did not want to limit ourselves to last year's design so we considered a few other options. We considered a cart system, a pendubot model, and a pulley system. The option we were most intrigued by was the cart system operating in two separate axes. The cart system is frequently applied to inverted pendulums in one-axis, so it seemed like a viable option for our system. However after much consideration, we ultimately decided to use the set-up developed last summer as it had several advantages: it provided a good physical representation of actual TVC; most of the components had been previously purchased or built so we were able to immediately begin working on the control system rather than spending much of our internship designing a new physical model; we could speak to last year's group about the system; and our mentors had a good understanding of the setup.

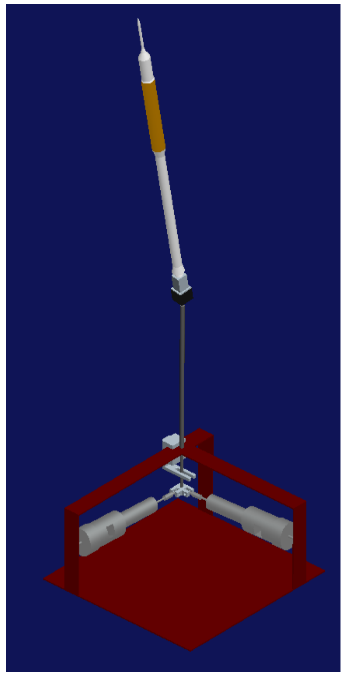

Figure 6. 2010 Pro-E Design

B. List of Components and Brief Explanation of Functioning

- 2 linear voice-coil actuators
  Voice-coil actuators operate based on the Lorentz Force Principle, which states that if a current-carrying conductor is placed in a magnetic field, the conductor will feel a force. A voice-coil actuator is composed of a conductor coil and a permanent magnet. Current is applied to the coil creating a change in the magnetic field and in turn producing a force proportional to the current. This force is determined by F=k*B*L*I*N (k=constant, B=magnetic flux density, I=current, L=length of conductor, N=number of conductors) (Black, Lopez and Morcos July 1993). The voice-coil actuators selected for this project had a continuous force of 3.2 pounds and a peak force of 9.6 pounds. This magnitude of force would be sufficient to control the physical system.
- Ares I model rocket (144:1 scale)
- 2 tri-axial rate gyroscopes
  Rate gyroscopes measure the angular velocity around three defined axes by measuring the precession of the gyroscope. Rate gyroscopes are composed of a spinning rotor mounted in a gimbal. By rotating the gyro case about the input axis, the gyro will precess. The speed with which you rotate the gyro case is proportional to the precession of the rotor. Thus by measuring the force of the precessing rotor, we are able to determine the angular velocity of about the input axis. Although our gyroscopes were tri-axial, we only utilized two axes from each of the gyros as our model only required angular velocity data about two axes (Integrated Publishing n.d.).
- 2 potentiometers
  Potentiometers consist of a resistor and a movable contact arm. In our case the resistor was a curved track along which the contact arm could move. A constant voltage of 5 V is applied to the potentiometer at the input terminal. As the angle of the contact arm is changed, the contact arm moves along the curved resistance track such that as the contact arm gets further from the input terminal, the total resistance of the circuit increases. As Ohm's Law states V=IR, so as the resistance increases, so does the voltage across the circuit. Therefore the angle of the contact arm can be determined by reading the voltage across the circuit (Harris 2002).
- 2 linear encoders

Linear encoders have two basic components, an optical sensor and a sensor strip. As the strip moves across the sensor, lines from the strip are counted. These lines are converted to voltage pulses and sent to the DAQ board. This allows us to determine the change in position of the drive shaft of the actuator, from which we can determine the angle of the rod.

- Data Acquisition (DAQ) System

C. Set-Up

The physical model consisted of an Ares I rocket, which sat atop a dual-axis ball bearing attached to an aluminum rod. The rocket was able to pivot about the ball bearing, but was limited in its ability to roll. Two linear voice-coil actuators were then attached to the base of the aluminum rod. The purpose of the aluminum rod was to increase the stroke of the actuators as the actuators were only capable of a stroke of 1.5 inches, which may not have been enough to control the rocket. By using the aluminum rod as a lever, we were able to effectively increase the stroke of the actuator from 1.5 inches to 6 inches in both axes. Each actuator was driven by an amplifier which took voltage as an input and output a current. This current then created a change in the magnetic field within the actuator's voice coil. The change in magnetic field results in an electromagnetic force which caused the actuator to advance or retreat, depending on the direction of the magnetic field. A guide vane was used us to prevent the aluminum rod from rotating due to forces applied by the actuators. This helped ensure that the system could be resolved into two separate axes, which vastly decreases the complexity of the mathematical model.

Figure 7. Desktop Learning Center

Various measurement devices were required in order to control the system. Two tri-axial rate gyroscopes were utilized to measure the angular velocities of the rocket and aluminum rod. Although the gyroscopes were capable of taking measurements about three axes, data was collected from only two axes as the third axis (roll) was prevented by our physical setup. Two potentiometers were placed at the joint of the rocket and the aluminum rod, allowing us to resolve the relative angle of the rocket to the aluminum rod. A linear encoder was mounted to each actuator, which gave us the position of the base of the aluminum rod at all times. Given this distance and the length of the aluminum rod, we were able to apply simple geometry to attain the angle of the rod relative to the z-axis of the system. Superposition of the angles of the rocket and the aluminum rod allowed us to obtain the absolute angle of the rocket in space. This absolute angle, along with angular velocity, was input into LabVIEW and sent through the control system. The proper force to be applied by each actuator was determined via the control system. This force was then sent as a voltage from the DAQ board to an amplifier. The amplifier converted the voltage to a current which drove the actuator to apply the proper force to bring the system towards stability. Measurements were taken at 1000 Hz in order to allow time for the voice coil to apply the proper force.

IV. Integration and Testing

After completion of the LQR Controller and making adjustments to the physical system, the integration and testing of the hardware with the mathematical model control system occurred.

A. Hardware Integration

LabVIEW, a National Instruments data acquisition program, is a graphical programming environment that integrates hardware with software, in order to develop measurement, test, and control systems. In LabVIEW, every command is created by dragging distinctive graphical icons and wiring them into terminals. Through an intuitive interface and expert help, we were able to apply the effectiveness of LabVIEW to our project. LabVIEW produces VI's, or virtual instruments, which can create a different task, collect data, compute mathematical functions, or command signals to devices. For our project, we designed eight different codes; one code to test every measurement device individually, as well as code that merges the mathematical model with data taken from each measurement device.

First, this program enabled us to write a code to acquire the voltage signal from the gyroscopes, potentiometers, and linear encoders and convert it to the desired units by calibration. The gyroscopes, which deliver the angular velocity of the rod and the angular velocity of the rocket (and), were calibrated by determining the voltage output for a given angular velocity. We then took the voltage output in the code, subtracted the offset and divided by the gain to determine the angular velocity in radians/sec. These units were necessary for the inputs of and .

After ensuring that the angular velocities procured in the LabVIEW code were adequate, we composed a code that acquired the voltage output of the potentiometers for both the x and y axes and converted it into the corresponding angle. This calibration was simpler as we were able to use a protractor to read the angle while obtaining the voltage. We applied a linear relationship between the voltage and angle such that LabVIEW would return the angle of the potentiometer. The potentiometer data gave us the angle of the rocket relative to the rod ( ) in radians.

In order to acquire the data for , the angle of the rod, we utilized the linear encoders that are attached to our actuators. We adapted the linear encoder code developed last year, with the help of Nick Johnston. The linear encoder code gave the distance traveled by the actuator when the motor shaft was extended or retracted. Simple trigonometry was applied to find the angle of the rod using the distance provided by the code and the given rod length.

After all four inputs were acquired, calibrated, and tested individually; a master code for all measurement devices was created. This ensured that merging all the data into one code would work simultaneously and would return the expected results. We tested this code first without placing the mathematical model into LabVIEW. One final step to control one axis was to merge a code of the three measurement devices and the mathematical model. The four inputs, , , and were multiplied by the **K** matrix which delivered the torque needed to ensure stability. Dividing by the lever arm allowed us to convert the torque into force which was necessary in order to command our actuators. Because our actuators require current as an input in order to create force, we needed an amplifier to convert the voltage output from the DAQ board to a current that could be sent to our actuator. To determine the voltage-force relationship between our DAQ board and actuator, we employed load cell calibration. We sent known voltages from the DAQ board to the amplifier. The amplifier converted the voltage into a proportional current which was sent to the actuator. The load cell then recorded the force applied to it by the actuator. This data was plotted, which allowed us to determine the relationship between voltage commanded from the DAQ board to force applied by the actuator. This relationship, which was linear, was incorporated into our LabVIEW VI so that given a desired force, we could immediately calculated the related voltage to be output by the DAQ board. Every code written was duplicated in order to test the measurement devices and math in both axes. A master code containing the x-axis full control and the y-axis full control VI were then placed together to receive 8 inputs and deliver voltages to two separate axes.

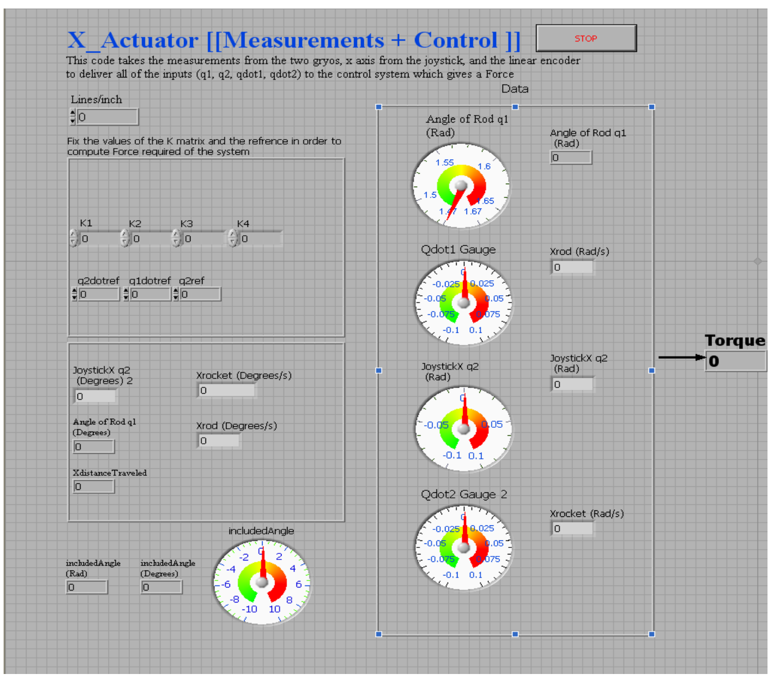

Figure 8. LabVIEW Front Panel

LabVIEW is also great for viewing the data we are receiving while running the program. The Front Panel, Fig. 8, or the user interface, contains control dialog boxes to be able to enter the **K** matrix and easily change the **K** values, displays indicators of the data in text boxes, as well as gauges to display changes in angles, and exhibits the outputs of the system (Torque, Force, and Voltage). The interface is a great way to view and collect the data in a user-friendly way, which enabled our group to intuitively understand the mechanical movement of our system.

B. Testing and Experimental Results

We began our testing by considering only uni-axis control of the system. We decided to consider each axis separately as this would allow us to decrease the complexity of our testing. Beginning with testing of the x-axis, we used a clamp to lock out motion in the y- axis, which we were not controlling. This greatly simplified the testing as we were able to isolate motion to the x-axis only. We input our initial **K** values, which had been calculated via simulation, into our LabVIEW VI, and, as expected, our initial testing was unsuccessful.

One problem we encountered was a discrepancy between the necessary force, and voltage, being calculated by our code and the possible force, and voltage, our actuator could handle. In response to this problem, we added a sequence of if-statements to the LabVIEW code that limited the output voltage range between -10 and 10 volts. Therefore, even if the calculated force, and corresponding voltage, was outside of this range, our DAQ board would automatically output the limit voltage, either +10 or -10 volts. This gave us better control of the system, as we were constantly outputting some voltage to the system. Another problem we encountered arose due to noise in our gyroscope and potentiometer signals. We ran independent tests on the measurement devices to determine what range of noise they were contributing to the system. Given the data from these tests, we tried various adjustments to limit this noise, and settled on applying a moving-average smoothing filter to the signals. We selected a half-width of 5, giving us an average of 10 data points for every voltage output. One problem with this type of signal is that it results in a time-delay in the output as you are no longer outputting a signal directly proportional to the input signal at a given time. Although this time delay was very small, it still may have caused an overall lag in our system response. Overall applying the filters dramatically improved the performance of the system, but we were still unable to attain control.

Aside from those major problems, we also ran into a variety of hardware issues. The output voltage of one of our gyroscopes tended to drift over the course of our testing, making it necessary to frequently adjust the offset gain of that gyroscope. This was not difficult to do, but it added complexity to the testing as this signal had to be constantly monitored. We also had problems with our DAQ board acquiring signals properly. This problem was solved, but did cause us to lose some time during testing. After making the various hardware adjustments mentioned above, we moved onto tuning the system gains. We applied a variety of gains, primarily focusing on controlling the angle and angular velocity of the rocket. Certain gains seemed more promising than others, but ultimately, we were unable to gain uni-axis control due to time constraints.

V. Conclusion

A. Future Work

As described by the title of our project, the ultimate goal of this project was to design a learning center that would better educate individuals about Thrust Vector Control. Due to time constraints, we were unable to complete the learning center. Therefore future work could go into developing an interactive station that would provide a better understanding of TVC. This learning center could incorporate the following: videos of TVC controlling rockets in flight, a brief presentation of the physical principles of TVC, a virtual model of a rocket where the user can apply a force to any point on the rocket and observe how the TVC system would react to return the rocket to the correct flight trajectory, as well as our physical model responding to an input response. Creating a user-friendly learning center would be a great addition to the physical model we developed this summer.

There are also some technical adjustments that could be made to our model in order to improve its performance. We utilized an L-shaped bracket in order to restrict roll of the rod in our physical model, but this L-bracket introduced quite a bit of friction to the system. Replacing this L-bracket with a linear bearing would reduce friction but still provide the necessary control of roll. Further investigating the frequency response of our various mechanical devices would also help refine the system. By determining the frequency response of all of our measurement devices as well as our actuators, we could adjust the sampling rate such that we would be operating within the frequency response range of our system. Currently the sampling rate was picked somewhat arbitrarily so there is no way to determine whether or not our system is operating optimally. By determining the frequency response, we should see improved operation of the system.

Future work could go into tuning of the gains. Much of our internship was spent working on developing a mathematical model for the system, from which we found gains to apply to the system. However because the mathematical model was not an exact model of our system, these gains did not control the system. In order to control the system, a wide variety of gains should be applied. By applying these gains and testing the system response, data from these tests could be used to focus in on the most appropriate gains. This would result in system control.

B. Closing Remarks

In retrospect, this project was extremely unique in that our goal was to create a system that teaches others about Thrust Vector Control, while we were also learning many of those aspects ourselves. Although we were unable to complete dual-axis control of the system, the knowledge gained by researching this projects was immense. We ascertained the knowledge necessary to create an applicable control system, became fluent with the Langrangian, as well as became efficient at hardware integration and programming in LabVIEW and Matlab. Unfortunately, we believe our greatest downfall was not leaving enough time for integration of the physical system with the mathematical model. We believe that if we were more efficient in our process we would have had more time for testing and tuning, and would have been able to complete the project. In order to complete the project, time should be spent looking at developing more appropriate gains for the system. Also, replacing some of the hardware, such as the L-bracket, would decrease the unaccounted-for friction in the system, making it more similar to our mathematical model. Work could also go into developing the learning center itself, by incorporating visual aids to help in understanding our physical model.

Acknowledgments

We would like to thank our mentors, Blake Stuart and Dr. David Young, and our Principal Investigator, Lisa Bates, for their continuous support throughout this project.

References

Black, Bill, Mike Lopez, and Anthony Morcos. *Basics of Voice Coil Actuators.* BEI Motion Systems Company, San Marcos, California: PCIM, July 1993, 44-46.

Block, D. J. "Mechanical Design and Control of the Pendubot." *M.S. thesis, University of Illinois Urbana-Champaign*, 1996.

Harris, Tom. *How Joysticks Work.* June 25, 2002. http://electronics.howstuffworks.com/joystick4.htm (accessed August 2, 2010).

Integrated Publishing. http://www.tpub.com/neets/book15/63e.htm (accessed August 2, 2010).

Rong, M. W. "Control of a 2-DOF Inverted Pendulum." *M.S. thesis, Hong Kong University of Science and Technology*, 2000.

Spong, M. W. "The Control of Underactuated Mechanical Systems." *First International Conference on Mechatronics*, 1994.