Observer-Based Robot Arm Control System

Project Proposal

By:
Alex Certa
Ron Gayles
Nick Vogel

Advisor:
Dr. Dempsey

September 27, 2011

Project Summary

The goal of this project is to use Ellis's method of observer-based control to control complex configurations of a robot arm system. Observer-based control simulates the system and fuses any available sensors to estimate the state of the system. Ellis's method is based largely on PID-style controllers, making it a relatively simple style of control for someone with experience in PID tuning or for someone without advanced control knowledge. Two systems are being used for this project. One is a horizontal arm configuration with two degrees of freedom, with a spring system associating the second degree of freedom to the first. The other configuration is the pendulum configuration. This configuration is vertical and thus has to deal with force of gravity.

The systems have both been tested with traditional control schemes, and models have been developed to approximate the response of the system. The actual observer and observer-based controller will be constructed in Simulink and run via Quarc from Quanser Consulting to control the system. The two systems should quickly and consistently move to commanded positions, and the systems should be able to consistently hand objects between them.

Detailed Project Description

Background Information

Control theory is applied in a vast variety of different fields including heating & cooling systems, cruise control, assembly line automation, and nuclear reactor control[3]. Most applications still use Proportional-Integral-Differential (PID) control because it is relatively simple to use and generally provides sufficient results for many applications while being easy to understand[2].

PID control consists essentially of three parallel paths summed in the forward path. The first, the proportional path, simply is a constant times the error. The second is integral, which is a constant times the integral of the error. The third term is the derivative term, which consists of a constant times the derivative of the error. The three paths can be seen in Figure 1-1.

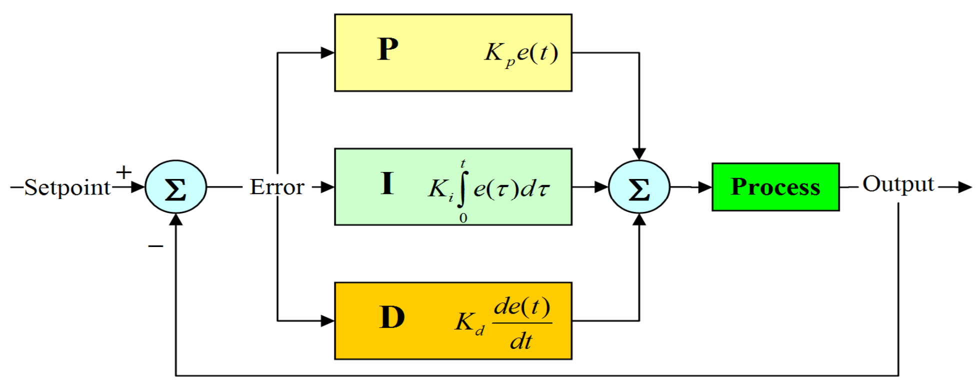

Figure 1-1 PID Control Diagram.

In practice, the PID controller cannot perfectly replicate the theory[2]. The derivative term is not realizable in practice and so instead it is approximated by a first order high pass filter with a relatively high corner frequency. The frequency domain equation for the derivative term is kp*s/s+p1, where p1 is large enough for it to approximate a differentiator for the specific application.

Observer-based control is one of the newer, more advanced concepts in control. Observers simulate a system based on available data, including sensor readings and command input and controller output. The method of simulating and controlling using this concept varies from observer to observer, but one of the most common types of observers used is based on linear algebra.

One method of observer-based control that stands out in particular is the Kalman Filter. This filter uses knowledge of the variance and covariance of the noise of different sensors in order to minimize the mean squared error of the measurement. This method, while being in some sense 'optimal', requires knowledge of statistics, linear algebra, and much time spent studying the sensor outputs.

George Ellis, however, proposes a system that is largely based on PID control[1]. The general control system set up can be seen in Figure 2-1.

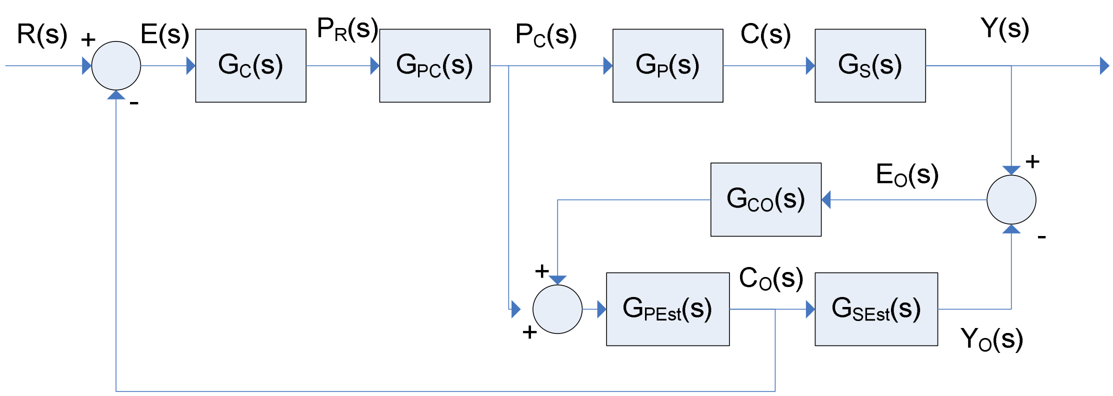

Figure 2-1 Ellis's Method of Observer-based Control[1]

R(s) is the set point input of the system. E(s) is the estimated error of the system, which is then fed into a PID controller Gc(s), and through the power converter Gpc(s). This signal is sent both to the actual system Gp(s) and to the observer. The system is then monitored by the sensors, which have a transfer function Gs(s). The output is then compared to the output of the observer, and the difference is sent through a PID controller Gco(s) and added to the input to help drive the error of the system to zero. The observer then estimates Gp(s) in Gpest(s), and this signal is then used to control the system. The observer also estimates the sensors in Gsest(s) in order to provide an output without significant phase lag.

Functional Description and Block Diagrams
With that background, the motive of the project is clear: evaluate the usefulness of Ellis's Method of observer-based control as a simple alternative for complicated robot arm systems.

The overall goals of the project are:

- Learn the Quanser software package and real time control via Simulink.
- Obtain a mathematical model for the pendulum arm and the horizontal arm.
- Design controllers for each system using classical control methods.
- Design a controller for each system that uses observers to predict the plant's response.
- Evaluate the performance of the two control methods and compare the result.

The workstation for each robot arm consists of the following components:

- PC with Matlab and Simulink
- Motor with Quanser Control System
- Linear Power Amplifier
- Robot arm with Gripper
- SRV-02 Rotary Servo Plant
  - One robot arm will also contain a SRV-02 Rotary Flexible Joint to add another degree of freedom.
  - The pendulum robot arm contains a rotary encoder, and the level robot arm contains a potentiometer to measure position.

One robot arm will be configured vertically in a pendulum-like fashion to incorporate the effects of gravity on the arm shown in Figure 4-1. The other robot arm will be placed horizontally and will have a flexible joint to add a second degree of freedom that is independent from the base of the system, shown in Figure 4-2. A closed-loop PID control system will be implemented in Simulink and will use Quanser software to allow real-time control of the robot arms through Simulink.

System Inputs:
- Internal Commands (position and velocity)
- Potentiometer Position Feedback (2 DOF arm configuration)
- Rotary Encoder (pendulum arm configuration)

System Outputs:
- Position
- Velocity

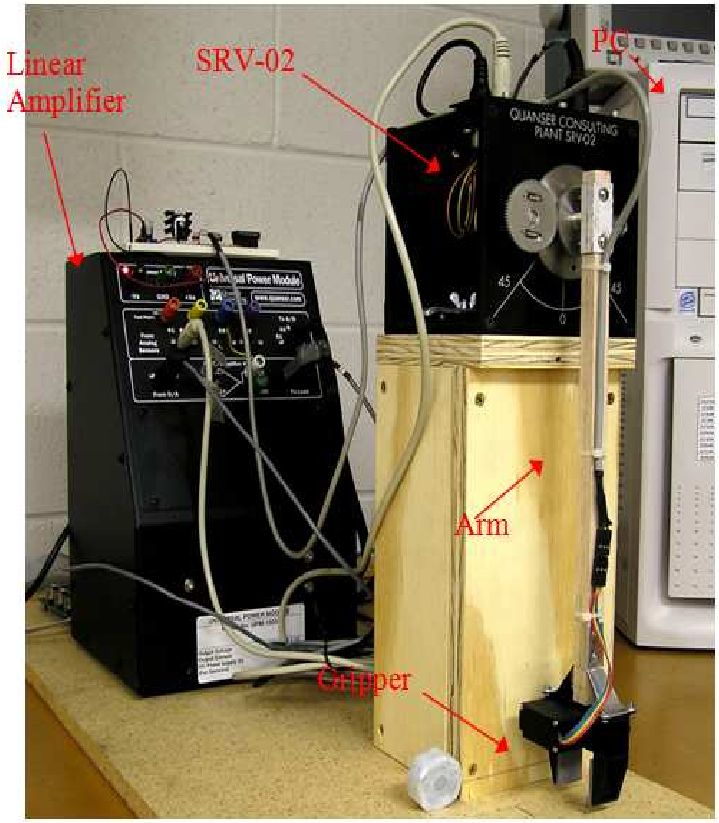

Figure 4-1 Pendulum Robot Arm Configuration

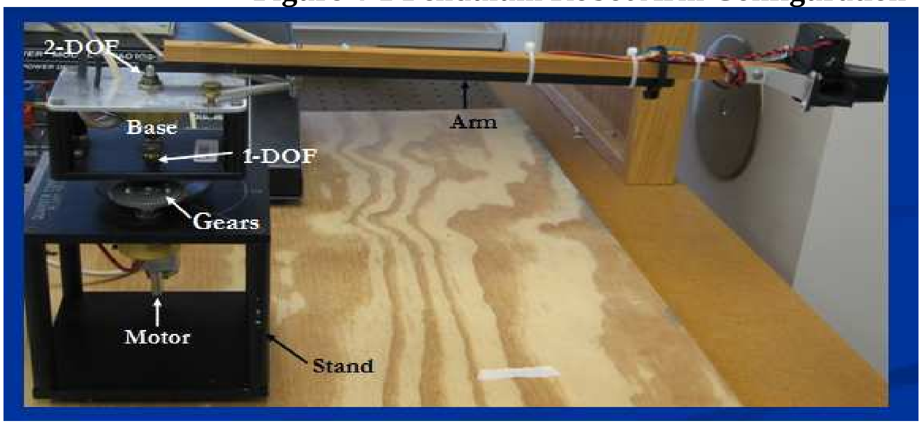

Figure 4-2 2-Degree of Freedom Robot Arm

The high level block diagram for the project is shown in Figure 5-1. The command signal is set in Simulink, which then sends a signal to the implemented arm controller, which then sends the signal to the arm. Sensors connected to the arm then send feedback to the controller allowing closed loop control. The power electronics involved with the robot arm controller, the robot arm itself and the sensors all introduce external disturbances including power supply noise, changes in load, friction, and quantization error.

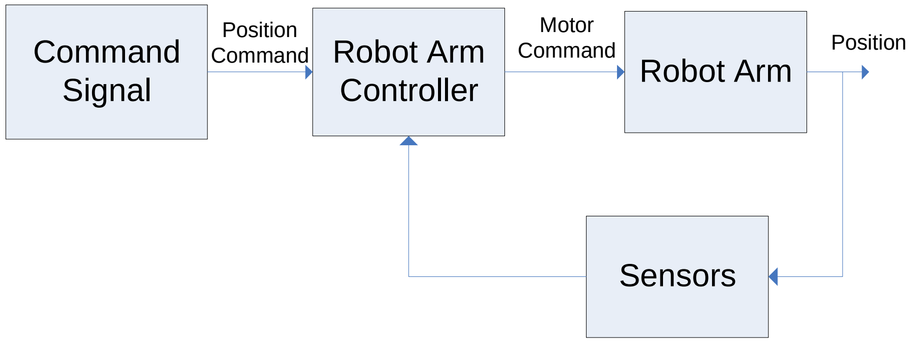

Figure 5-1 Overall System Block Diagram

Figures 5-2 and 6-1 show detailed views of the robot arm systems, the sensors, and the power electronics involved in the control of the arms.

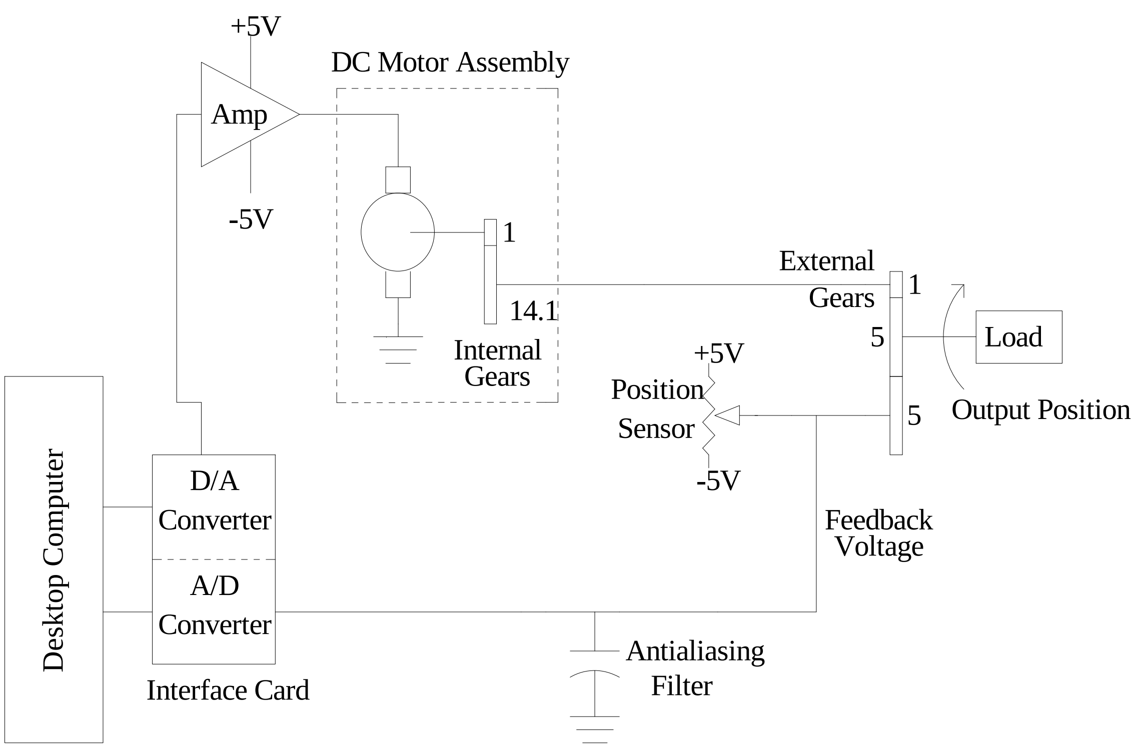

Figure 5-2. Quanser Electromechanical Plant with Potentiometer

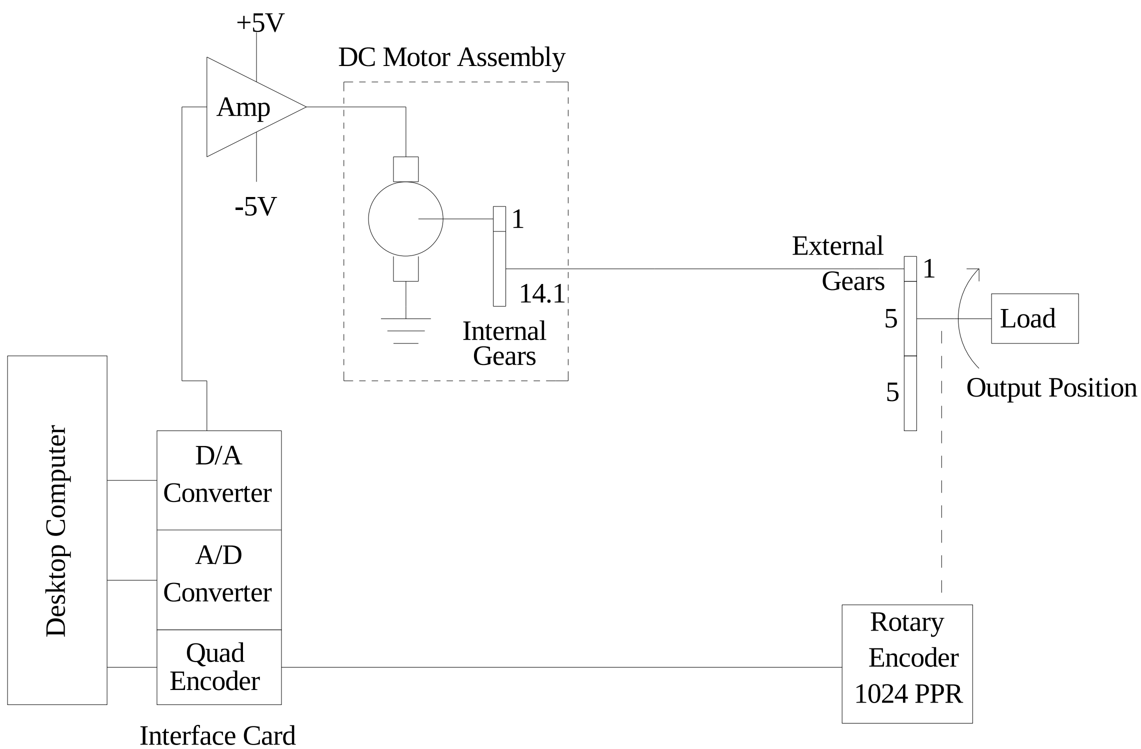

Figure 6-1. Quanser Electromechanical Plant with Rotary Encoder

The first, simple type of controller used is a single loop controller shown in Figure 6-2. Gc represents a controller transfer function, which varies from simply a gain, representing a proportional controller, to a PID controller or even more complicated system with more poles and zeroes. Gp is the transfer function of the system being controlled, and H is the transfer function of the sensors used. The more complicated final system is shown in Figure 2-1.

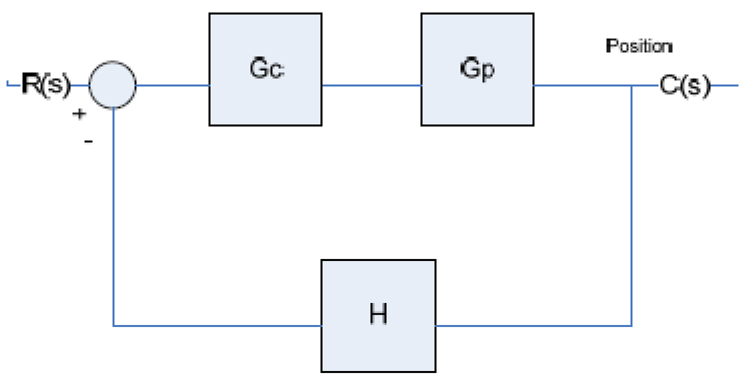

Figure 6-2 Single Loop Controller

Functional Requirements & Performance Specifications

The high level block diagram for the project is shown in Figure 5-1. The command signal to the system will be a value that assigned in Simulink. This value will be limited to plus or minus 90 degrees. The position command will be passed to the controller via Simulink. The controller will generate a digital control signal, which will be converted to an analog signal in the range of ±5 volts via a D/A converter. The position of the arm will be measured two different ways in the two platforms. The 2-DOF platform will use a potentiometer to measure position. The analog position signal will be fed into an A/D converter. The digital signal will then be compared to the reference signal to generate an error signal. The pendulum arm platform will use a rotary encoder to measure position, which is fed into the quad encoder interface to the computer. The signal is then compared with the reference signal to generate an error signal to drive the controller.

The level 2-DOF configuration shall be controlled using different types of classical controllers and observer-based controllers. The system shall also perform disturbance rejection for a load. The specifications for the performance of this system for a step command of 90 degrees are as follows:

- The overshoot of the arm shall be less than or equal to **15%**
- The settling time of the arm shall be less than or equal to **2s**
- The phase margin shall be at least **50 degress**
- The gain margin shall be at least **3.5**
- The sample time shall be **10 ms**
- The steady state error of the system shall be less than **2 degrees**

The pendulum arm configuration will go through the same design process as above. The system shall perform disturbance rejection for a load. The specifications for this configuration given a 90 degree step command are as follows:

- The overshoot of the arm shall be less than or equal to **15%**
- The settling time of the arm shall be less than or equal to **2s**
- The phase margin shall be at least **50 degrees**
- The gain margin shall be at least **3.5**
- The sample time shall be **10 ms**
- The steady state error of the system shall be less than **1 degree**

For both of these systems, the specifications shall hold for loaded conditions. The controllers will be designed to work with the existing robot arm system, and A/D and D/A converters.

Completed Work for Pendulum Configuration

The work completed was done by two groups working separately on the pendulum robot arm system and the two degree of freedom robot arm system.

The preliminary research on the pendulum robot arm system was done in this order: first, the motor system was identified and controlled; second, the robot arm system was identified and a linear model constructed; third, proportional control was implemented; finally, PID control was implemented, then a non-linear model for the motor was constructed.

The first step taken was to control a relatively simple system: the Quanser station with the arm not attached. The system was identified using knowledge of the workings of motors along with data from the data sheet to construct a system block diagram. The model is shown in Figure 8-1

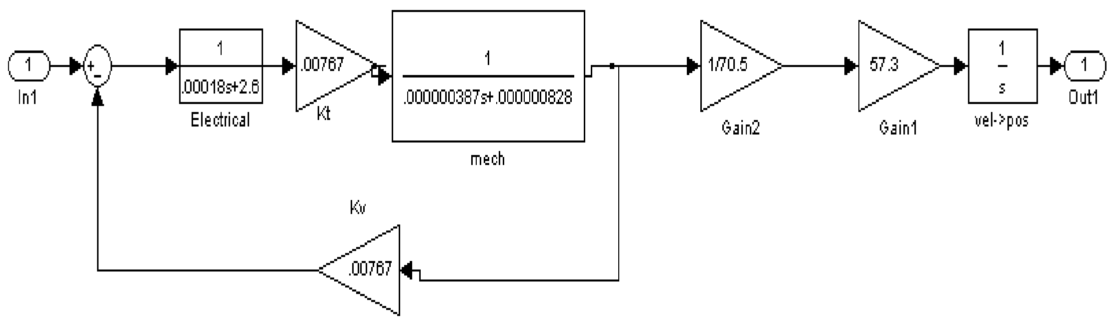

Figure 8-1 motor and gear train linear model

This model closely matched the response seen in the actual system. Based on this model, a proportional controller was constructed. The maximum velocity of the system appeared to be approximately 450 degrees per second. With a 180 degree input, there is a 0.25 degree steady state error, and it took 0.42 seconds to reach 90% of the way to 180 degrees. A feed forward model was then constructed. The poles for the model are at 0, -13000 and -60. Since the pole at -13000 is much further than the other poles, it can be ignored for this design.

A feed forward network attempts to cancel the low frequency effects of the plant. In this case at low frequencies the transfer function approaches 102/s. Thus the network for the feed forward design should be s/102. This is a pure differentiator which cannot be implemented in practice, and so a pole is placed more than a decade from the relevant poles. The gain was then tuned for optimum results.

For a 180 degree input, the time to reach 90% of the value was the same, however, it reached steady state 7% faster and only had 0.1 degree of error, which is much smaller than the potential gear backlash and thus can be ignored.

The proportional and feed forward controller transient responses can be seen in Figure 9-1 and 9-2.

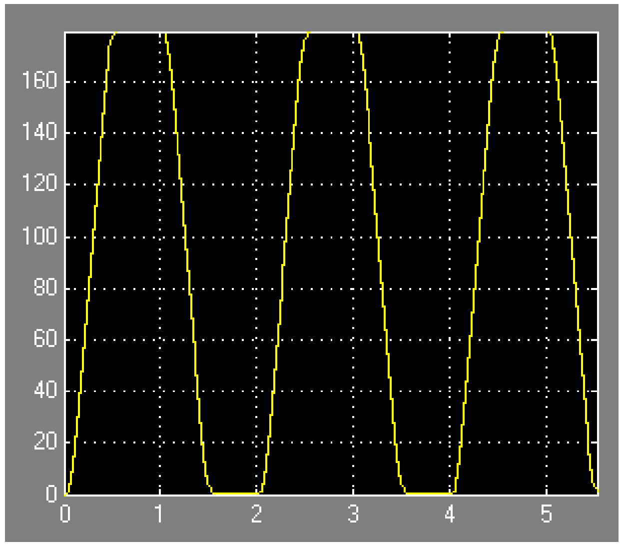

Figure 9-1 Transient Response of Proportional Controller for Motor System

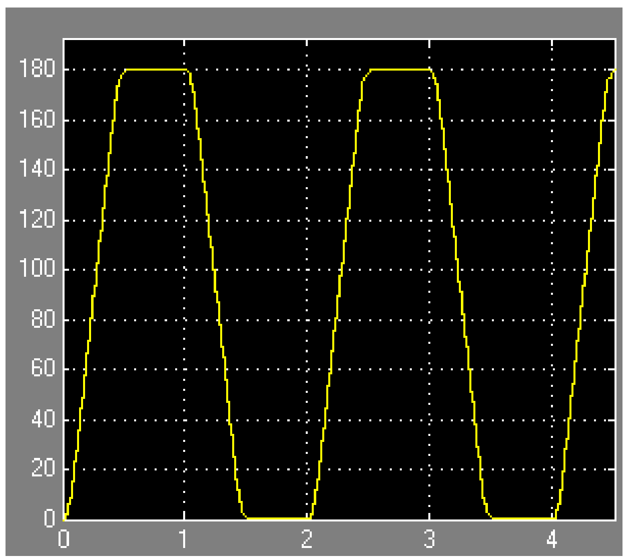

Figure 9-2 Transient Response for Feed Forward Controller for Motor System

After completing these controllers for the motor system, the arm was attached and the system including the arm was identified. Three primary methods were used to help identify the robot arm system. First, the steady state voltage response was measured to help determine the DC gain over the linear region. Second, a proportional controller with high gain was implemented and the 2nd order transients were measured, then the exact second order equations were used to estimate the open loop pole locations. Third, the frequency response was measured to compare the results of the model with the results of the system.

The voltage response of the arm was measured and is shown in Figure 10-1. The next step taken was to divide by the angle to get the DC gain at each DC value. The response is shown in Figure 10-2. Small angle approximation guarantees that the DC gain due to gravity should be a constant over small angles. Our DC gain as shown in Figure 10-2 is not constant but appears to be linear over a large portion or the range. This disparity can be accounted for by Coulomb friction. Through trial and error, a 0.13 constant reduction in voltage was found to account for the slope of the DC gain. The modified DC gain graph can be seen in Figure 10-3.

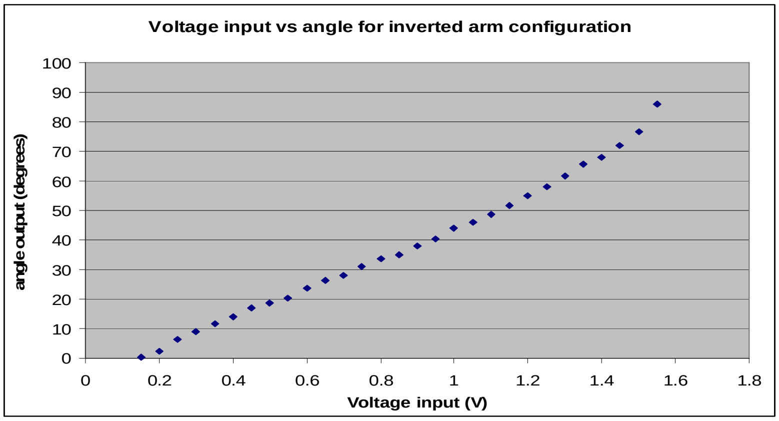

Figure 10-1 DC Voltage Response of Robot Arm System

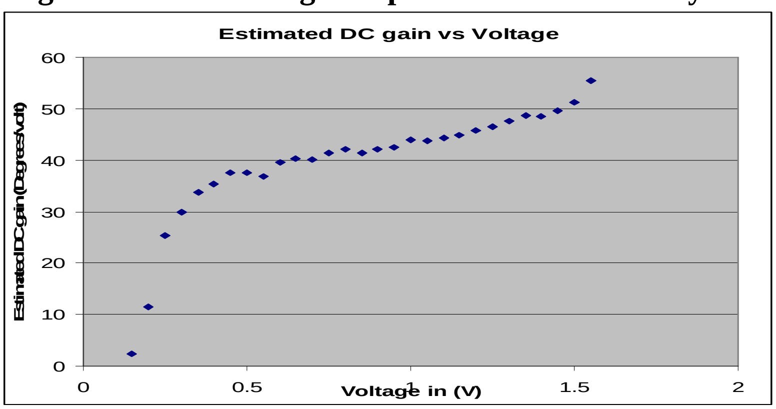

Figure 10-2 DC Gain Response of Robot Arm System

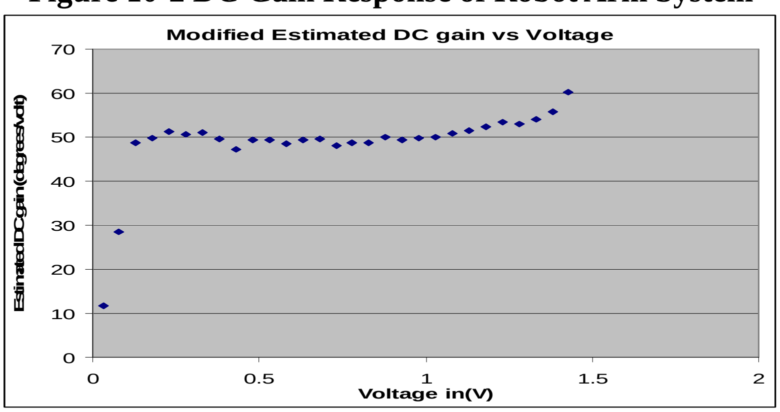

Figure 10-3 Modified DC Gain Response of Robot Arm System.

The second order step response was then generated using a proportional controller with a gain of 0.45. This gain provided sufficient overshoot so that the second order specifications of percent overshoot, settling time, rise time, and time to first peak could be accurately measured. The percent overshoot was 46%. The rise time was 0.06 seconds. The settling time was 0.58 seconds. The time to first peak was 0.14 seconds. The graph of the transient response can be seen in Figure 11-1.

The robot arm system was assumed to be a second order system in the form k/(s/p1+1)(s/p2+1). Based on the equations for an exact second order system, which should prove a good approximation, the poles were found to be at 2.6 and 11. The results of the system with the proportional gain and the results of the model with the same gain are shown in Figure 11-1 and 11-2.

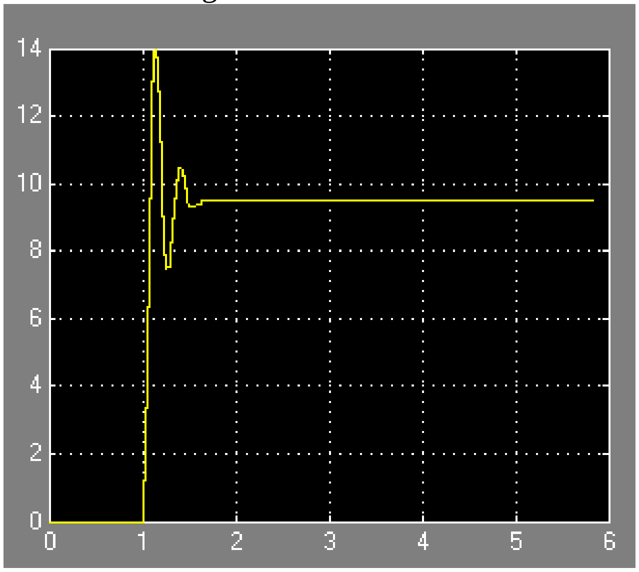

Figure 11-1 Transient Response of High Gain Proportional Controller for Robot Arm System

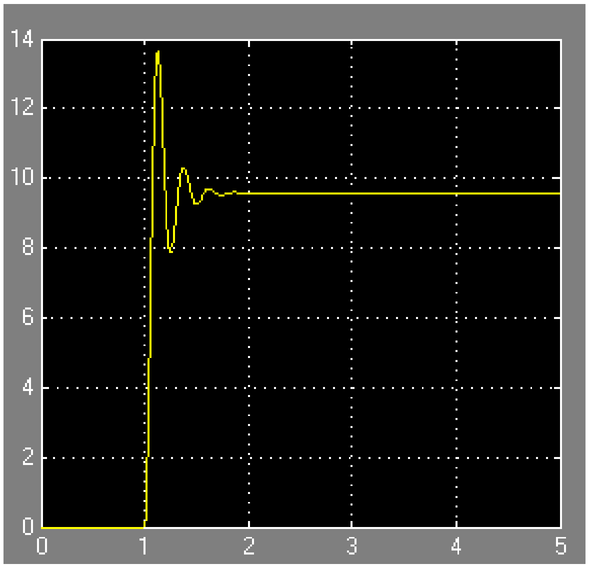

Figure 11-2 Transient Response of High Gain Proportional Controller for Robot Arm Linear Model

These responses are very similar, with only slight errors in the values at different times. The next step taken was to compare the system and the model open loop. The results are shown in Figure 12-1 and Figure 12-2. It is clear that the model and actual systems still have major differences, but these differences will be reduced when placing them in closed loop systems. The model has a much more gradual transition into steady state, but the actual system stops dead at a certain point. This difference can most likely be accounted for by static friction effects so that once the robot arm stops, it is much more difficult to get moving again.

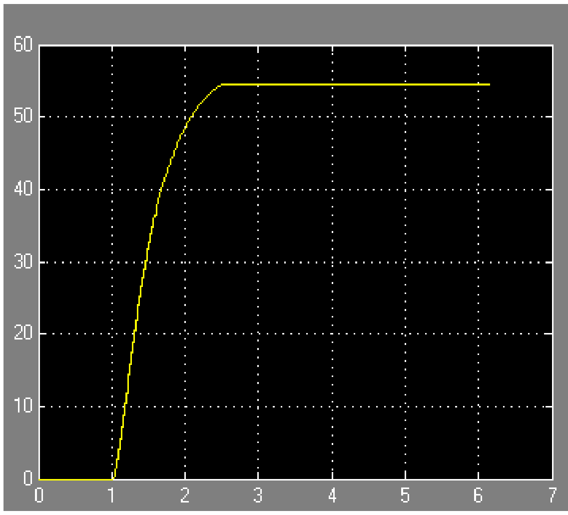

Figure 12-1 Open Loop Transient Response of Robot Arm System

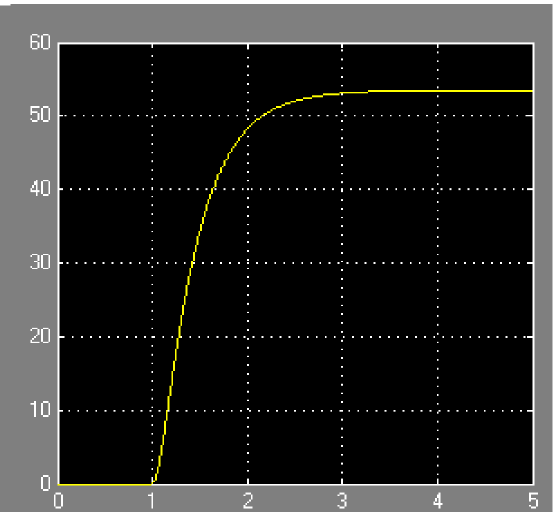

Figure 12-2 Open Loop Transient Response of Robot Arm Linear Model

Next, the magnitude frequency response of the robot arm system and the linear model were measured. The results are shown in Figure 12-3. The results were normalized to a nominal value of the DC gain. The system response is shown in blue, while the model response is shown in yellow. The system with a slight change to the DC gain is shown in pink.

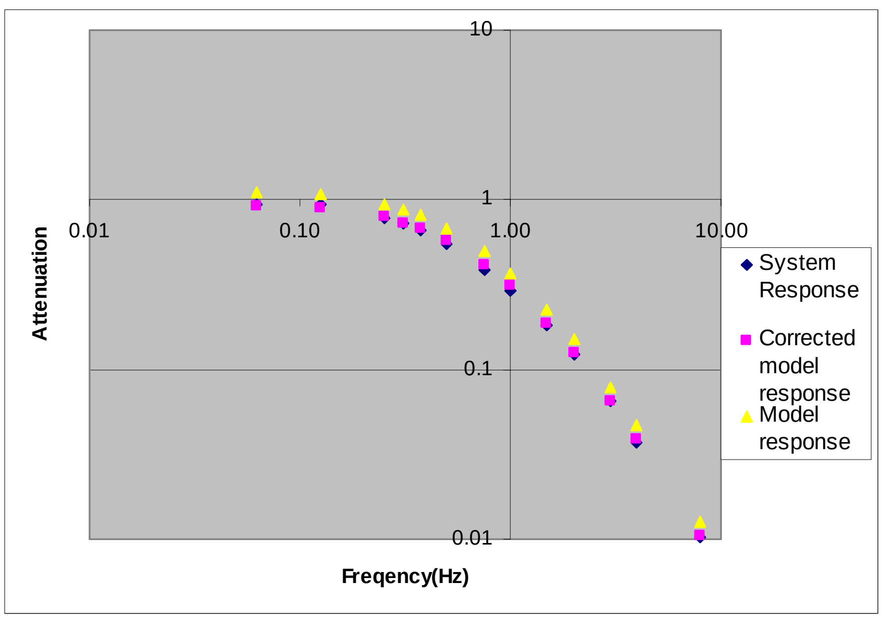

Figure 12-3 Frequency Response of System and Model

A proportional controller was then constructed with 15% overshoot for a 20 degree input. The steady state error was 2.5 degrees, the rise time was 0.12 seconds, and the settling time was .41 seconds. The transient response can be seen in Figure 13-1.

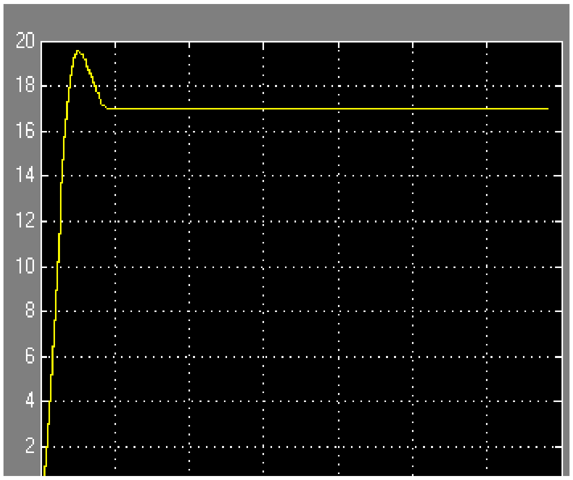

Figure 13-1 Transient Response for Proportional Control of Robot Arm

After the proportional controller was working, a PID controller was implemented to create an exact 2nd order system. The poles were cancelled with zeroes and a pole at the origin and another on the real axis were placed. Having the pole placed further out in theory makes the system faster, but with the D/A converter limitations the input has to be rate limited to avoid saturation. The system was tested with no saturation (with a 1 degree input), to see how the system would work without rate limitation. As expected, with the pole locations further out the settling time is lower for the same percent overshoot. However, for a 180 degree input the time spent in rate limitation dominates the actual transients, so the system is faster with a lower pole location that allows a larger rate limit. The results of this testing can be seen in Table 13-1. The final results of the PID controller are show in Figure 14-1.

Table 13-1 PID Pole Location Determination Data

Pole Location	Gain value	overshoot %	Settling time	Rate limitation	Rate limited settling time
-40	0.75	14.9	0.20	155	1.16
-80	1.5	15	0.10	148	1.20
-60	1.1	14.9	0.14	151	1.18
Rad/s			s	deg/s	
			1 deg input		180 deg input

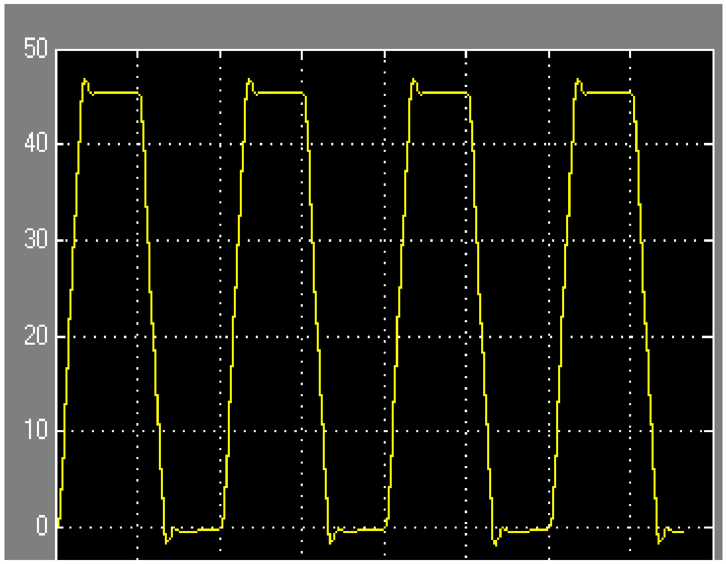

Figure 14-1 Transient response for PID controller

Completed Work for Level Arm Configuration

The horizontal 2-DOF robot arm underwent similar design stages to the pendulum configuration robot arm. At this point the progress we have completed is system identification, and sample rate selection. The system identification is broken down into two steps. First the system was without the effect of the springs, and then it was modeled with the effect of the springs. The two results were combined to build an accurate model of the system.

First, the DC gain and pole locations of the system were determined from the step response of the system without springs. The step response is shown below in Figure 14-2.

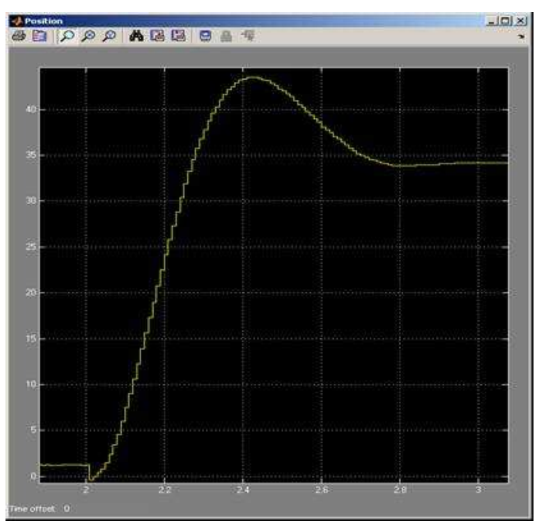

Figure 14-2 Step Response of Horizontal Configuration Without Springs

From this graph the pole locations are determined to be 0 and -10 rad/s. The DC gain was found in Simulink to be 1500deg/s. This gives an overall transfer function of Gp=1500/(s2+10s). Next, the springs were modeled with the step response of the system. The poles associated with the springs were determined from the displacement of the arm relative to the base. These turned out to be -4+-17j. The DC gain was found in Simulink to be .42 giving an overall transfer function of GD=0.42s/(s2+8s+289). The spring disturbance was modeled as a minor loop disturbance because it continually affects the output until steady state. The block diagram of the entire system is shown in Figure 15-1 below

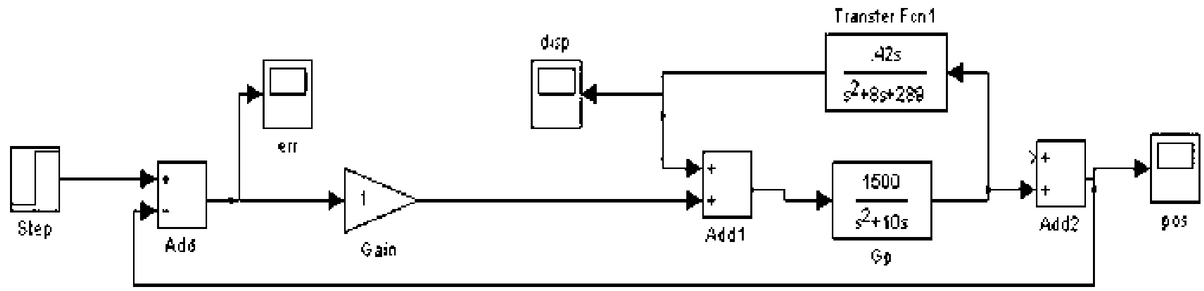

Figure 15-1 Block Diagram for the Arm with Spring Disturbance Modeling

After getting an accurate model of the plant, the model was digitized in order to allow implementation of digital controllers for the plant. The digital model was then analyzed via root locus and the sample frequency was adjusted to give a better response. This sample period was found to be 0.07s. At this sample period, the phase margin specification as well as the %OS specification was met with only proportional control.

<u>Schedule</u>

A Gantt Chart of the schedule of the project is available in Figure 14-2.

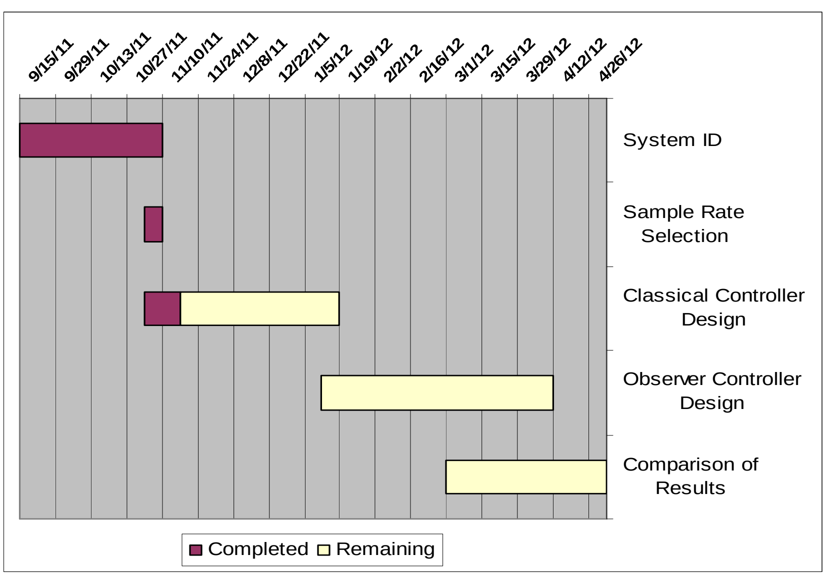

Figure 15-2 Schedule of tasks for observer based robot arm control project.

Required Equipment

No additional equipment is required for this project. The stations already in place including the two computers and the two Quanser motor stations are needed. In addition Matlab, Simulink, and Real-time Workshop are needed to interface and control the devices.

Conclusion

This project will provide an insight into the usefulness of the Ellis method of observer-based control to control complex robot arm configurations. The Ellis method will provide disturbance rejection and a method of augmenting degraded sensors or identifying broken sensors.

Initial investigations have already been performed into the systems used and baselines have been investigated by testing traditional controllers. More traditional controllers will be implemented. Then the observer-based controller will be designed and tested. The results will be tested, and based on these results, the value of Ellis' method can be judged.

Bibliography

[1] G. Ellis, *Observers in Control Systems*, San Diego, CA: Academic Press, 2002.
[2] G. Dempsey, *EE 432- Control System Theory Lecture Workbook*, Peoria, IL: Unpublished, 2011.
[3] A. Fouts and K. Liggett, "Observer-based Engine Cooling Control System (OBCOOL) Final Report," B.U., Peoria, IL, 2011