Senior Capstone Project Report for

Internal Model Controller Design for a Robot Arm

By Vishal Kumar

Project Advisor: Dr. Gary L. Dempsey

5/15/08

Project Abstract

This project is centered around controlling the Quanser Consulting Plant SRV-02 with a Linear Internal Model Controller. The Quanser Plant consists of a stand supporting the base, an arm connected to the base with springs, and a base housing a DC motor driving the arm. The plant consists of 2 degrees of freedom originating at the motor in the base and the springs at the arm creating a 4th order plant. The accurate disturbance detection of Internal Model Controller can help us design a controller to manage the 4th order Quanser Plant despite its' non-linearities and external disturbances.

Table of Contents

Introduction

This paper will combine the description of the project developed in Project Deliverable I, II and III and include the latest results to provide a developed Senior Project Report. Preliminary computer simulations results, analytical evaluations, system identification results, considerations for extra equipment, final controller design and evaluation are included. In essence, this report summarizes all research and work completed on the project in the Fall '07 and Spring '08 semester.

Physical and Functional Description

The Quanser Consulting Plant SRV-02 consists of an arm, base, and stand. The stand contains a motor driving the arm through a gear train in the base. When electrical energy is supplied to the motor the result is mechanical energy, torque, at the output shaft. The gear train moves the arm and this creates the 1st degree of freedom (DOF). There are also springs attached from the arm to the base. This along with friction in the rotary flexible joint forces the arm to move independently creating a 2nd DOF (Dempsey, 2007). DOF's add to the degree of the system plant, more on this in the functional description. The top down view of the system is shown in Figure 1 below. A side view of the system is shown in Figure 2 on the next page.

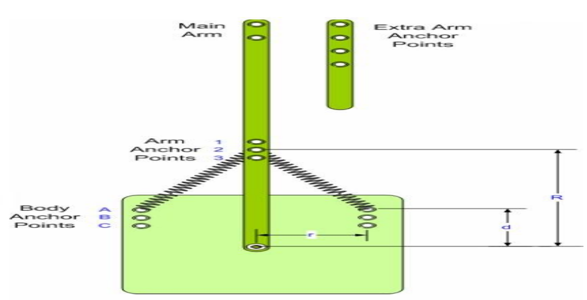


Figure 1 - Top-down view of the system showing the arm and base. Note the output shaft of the motor connected to the arm through a gear train in the center of the base Edwards).

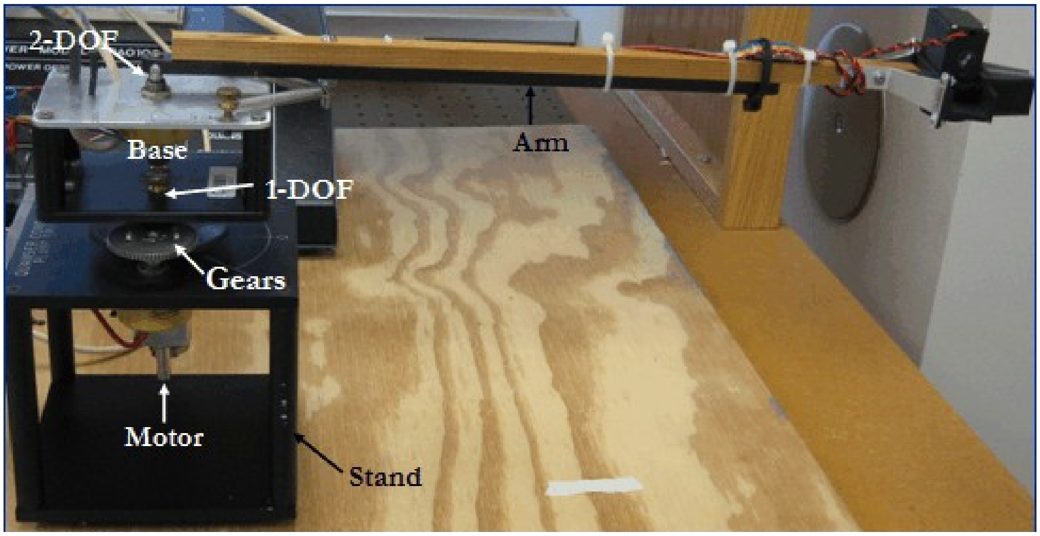


Figure 2 – Quanser Consulting System SRV-02 – Stand, base, arm and gears are shown. There are arrows pointing out where the DOFs originate (Dempsey, 2007)

The hardware shown above communicates with a 1.46Ghz Pentium-based computer with an internal A/D and D/A acquisition card. The system can be described using a high level system block diagram as shown in Figure 3 below. Lastly, the system is connected to a sophisticated power amplifier for driving the entire system shown as the amplifier in Figure 3 below.

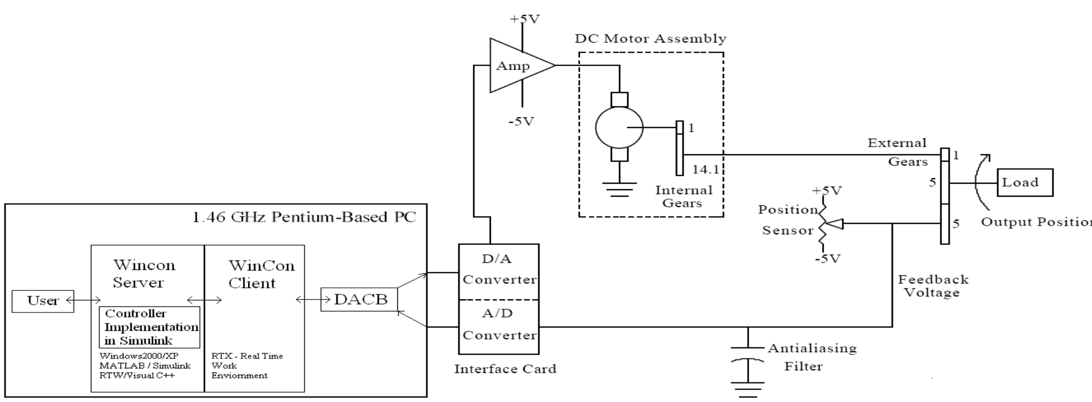


Figure 3 – High level block diagram. Note the gears, feedback voltage loop, DACB interface card housed in the PC and the anti-aliasing filter.

Software Interface

As shown in Figure 3, the PC contains a user interface for the Quanser Plant using WinCon and Simulink on a Windows based PC. WinCon enables you to create and control a real-time process entirely through Simulink and execute it entirely independent of Simulink. Diagnostics and position sensor output can be measured numerically, graphically, and is collected on the computer in real time. This is possible with the A/D and D/A converters on the Data Acquisition and Control Buffer (DACB) communicating with WinCon using the Real Time Execution (RTX) Workshop installed in Simulink. By implementing controllers in Simulink it is possible to measure the real time response of the plant.

Functional Requirements and Performance Specifications

This project flow will be such that controller complexity will increase with every step in effort of achieving all performance specifications. These specifications will be the standard of comparison for each controller design. The list below shows these set of specifications.

Percent Overshoot	5% max
Time to Peak(max)	50ms max
Time to settle	200ms max
Closed Loop Bandwidth	2Hz min
Peak Closed Loop Frequency Response	3dB max
Gain Margin	5.0 min
Phase Margin	60 degrees min
Steady State Error	1 degree max
Controller Execution Time	1ms max

The controller development flow, where each step can be considered a functional requirement is listed below.

1. Single Loop – Proportional , Proportional–Derivative Controller
2. Single Loop – FD Design for P, PD, PI controllers
3. Internal Model Controller
4. Internal Model Controller with Artificial Neural Networks

Research Significance

The research focus of this project will be minimizing the effects of external disturbances from the 2 degrees of freedom, from the rotary flexible joint, and non-linearities arising through out the system specifically at the gear train using the Quanser SRV-02 plant. From research, Internal Model Controller design is the best solution. However, the project cost of this approach compared with more conventional methods is not clear. Thus elements such as cost, performance, complexity, precision, accuracy, and design time will be explored in this project.

Internal Model Control

After conventional system identification of Gp(s), the process or plant model, and several controller design iterations, stated in the Functional Requirements, Internal Model Controller (IMC) design will begin. Refer to Figure 4 below where Quanser plant is shown as Gp(s). The Internal Model

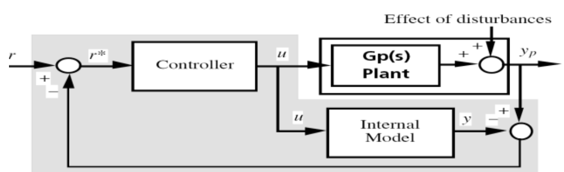


Figure 4 - Block Diagram for Internal Model Controller

Controller produces the difference between Gp(s) and the 'internal model' of Gp(s) providing the effects of the disturbances. The disturbance is then minimized by the controller. For minimum model mismatch, the controller is the inverse of the model without time delay. Traditional internal models are simulated by linear system design technique or even 2nd order. However, a fourth order model would be better because non-linearities exist from gear backlash, static friction and coulomb friction. The internal model architecture for a non linear plant needs to be designed through tuning.

System Identification without arm

System Identification was performed experimentally on the Quanser Plant without the arm. The plant could be modeled accurately as a 2nd order system with time delay. The plant gain, time delay, and pole locations were identified to be...

$$Gp(s)=\frac{69*e^{(-.350s)}}{(s(s/50+1))}$$

The figure below shows a comparison of the step response for the identified plant and experimental results. The system approximation is good.

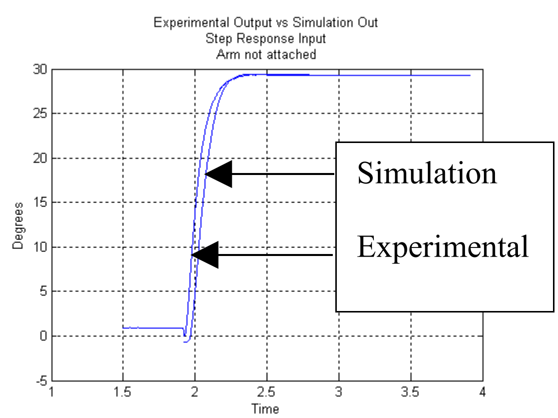


Figure 6 – System Identification Results without arm

System Identification with arm

System Identification has been performed experimentally on the Quanser Plant with the arm. The plant gain, time delay, and pole locations were identified to be...

$$Gp(s)=\frac{45.73*e^{(-.110s)}}{(s(s/30+1))}$$

The figure below shows a comparison of the step response for the identified plant and experimental results.

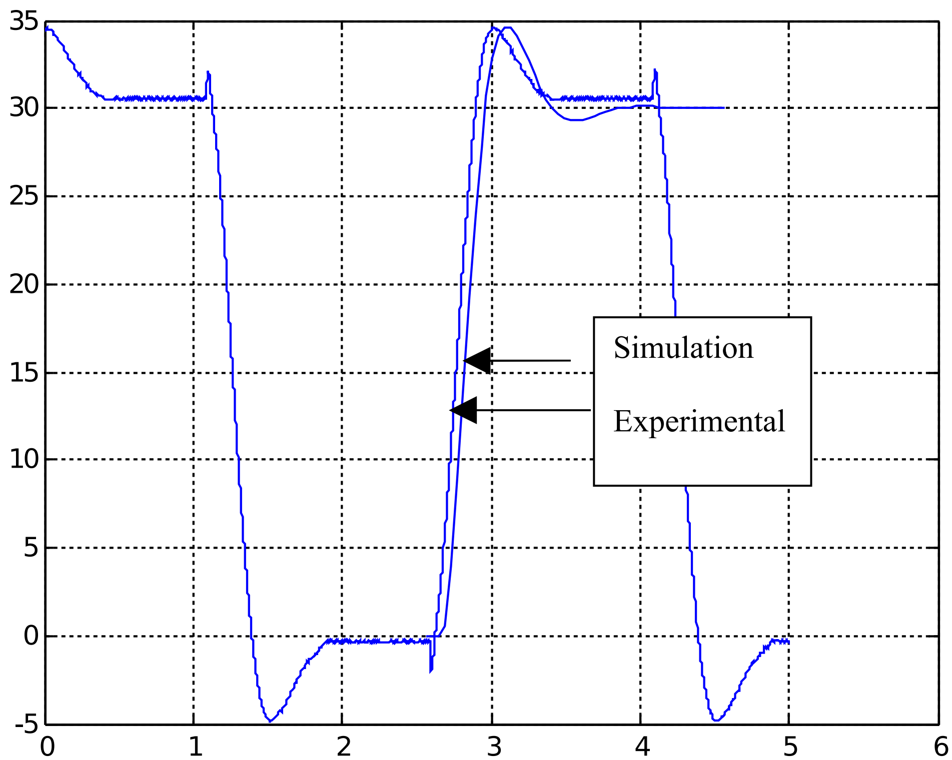


Figure 7 – System Identification Results with arm

P and PD controller design

Controller design without the arm on the plant was performed. P and PD controllers were implemented.

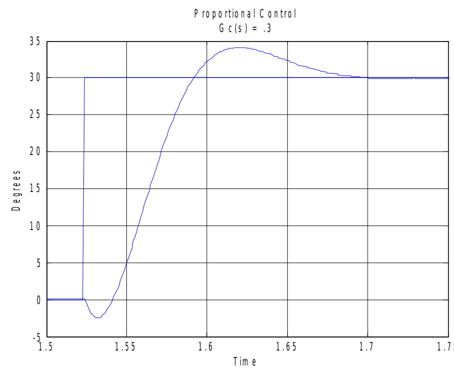


Figure 7 – Proportional controller results

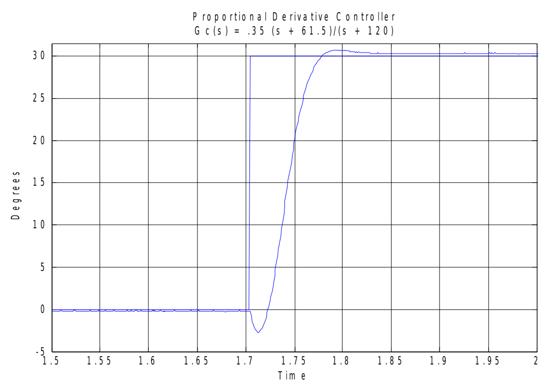


Figure 8 – Proportional Derivative controller results

The data for these two controllers is shown in Table 1 below. Note the increase in performance for overshoot, settling time and time to peak from the proportional to proportional derivative controller.

Table 1 – P and PD controller results

		PD Controller		P controller
	Experimental	Hand Calculations	Experimental	Hand Calculations
Gain	0.61	35.0067	0.303	17.4
Tp	0.09	0.036	0.1	0.084
Ts	0.1	0.06	0.16	0.13
% O.S.	4%	5%	16.20%	4.60%

Experimental K = Hand K * pi/180

Frequency Domain Proportional Controller Design

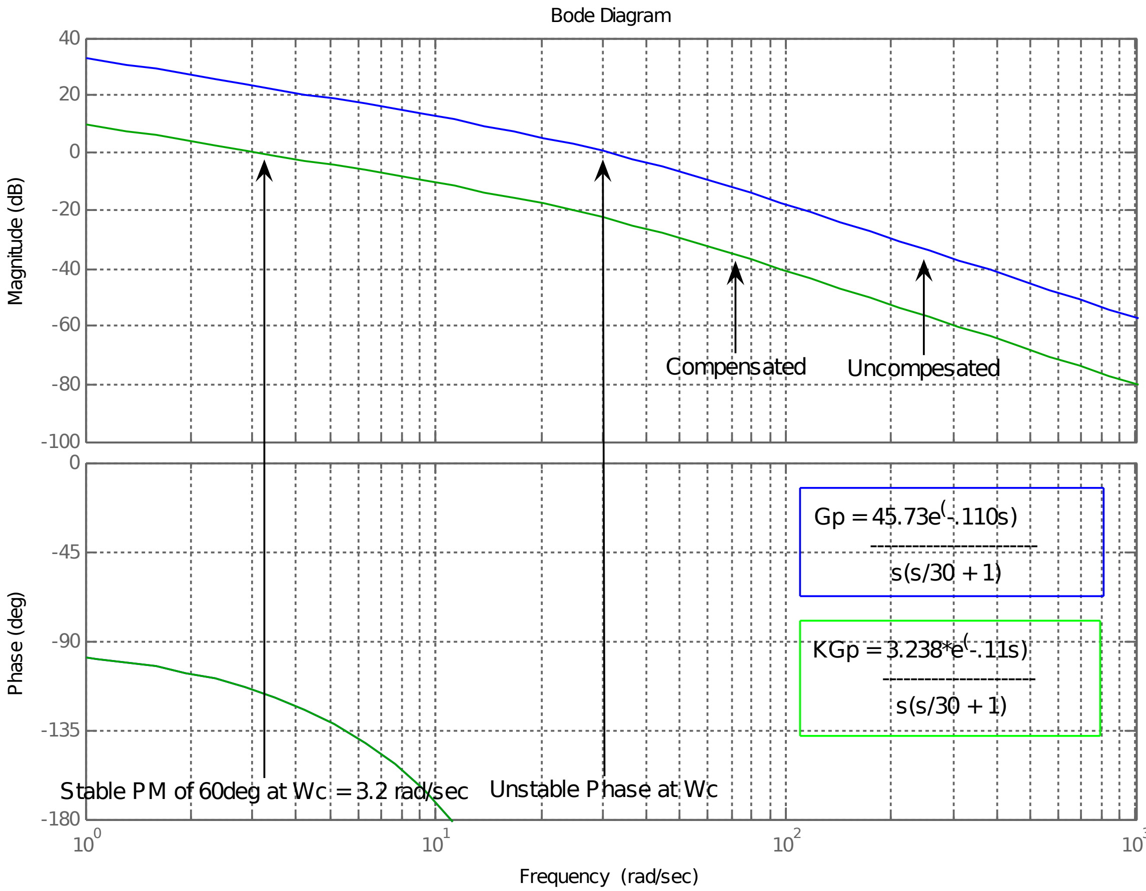


Figure 9 - Bode Plot of compensated and uncompensated plant

<u>Time Domain predictions using Frequency Domain Parameters</u>
Frequency Domain Parameters:
Cutoff Frequency (Wc) = 3.2 rad/sec; PM = 60 deg, GM = 11.4dB

Prediction Parameters:
Maximum Peak O.S. (Mp), Closed-Loop Bandwidth(B.W.), Damping Ratio(del), %Overshoot(%O.S.), Natural Frequency(Wn), Damped Frequency(Wd), Settling Time(Ts), Time to Peak(Tp),

Mp	= sin(PM)	= .866
B.W.	= Wc	= 3.2 rad/sec
del	= PM/100	= .6
%O.S.	= 100*e^((-del*pi)/sqrt(1-del^2))	= 9.47%
Wn	= (sqrt(sqrt(4*del^4+1) -2*del^2)/Wc)^(-1)	= 4.47 rad/sec

Wd = Wn*sqrt(1-del^2) = 3.57 rad/sec
Ts = 4/(del*Wn) = 1.49sec
Tp = pi/(Wn*sqrt(1-del^2)) = .87sec

<u>Time Domain Evaluation in Simulation</u>

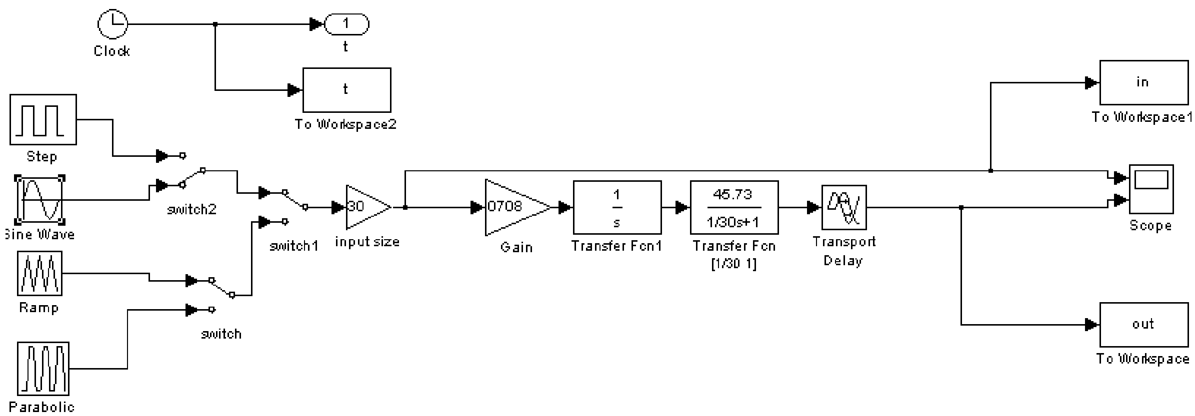


Figure 10- Simulink Model used for Simulation

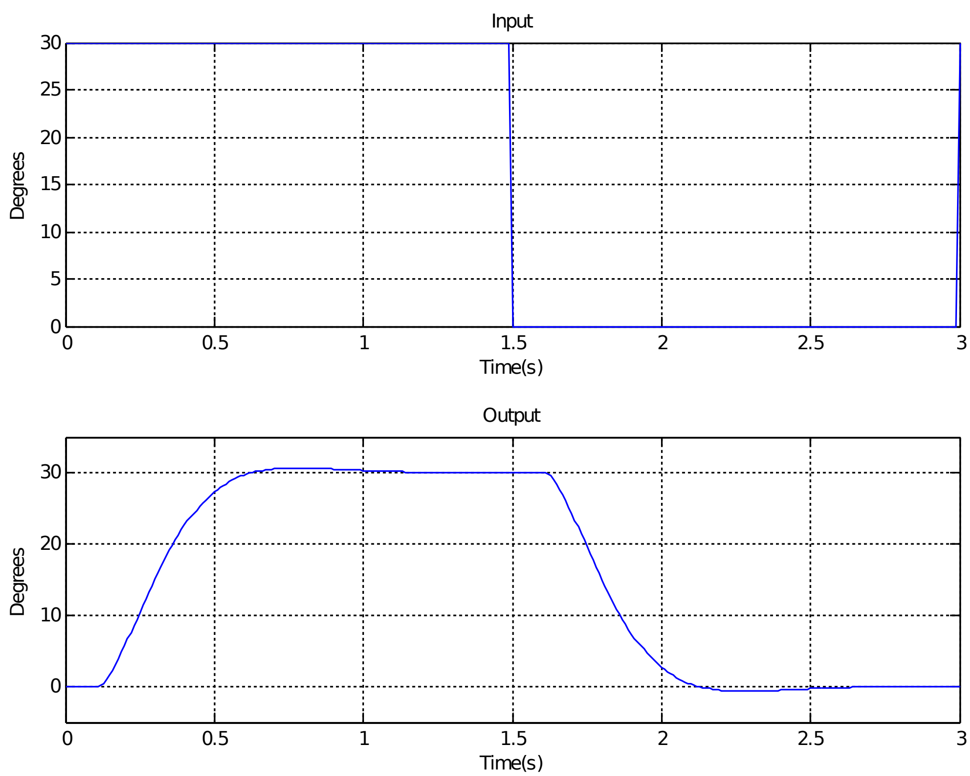


Figure 11 - Time Domain Input and Output

Time domain Measurements

Time to Settle with a tolerance of +-2% = .85sec

Time to Peak = .77sec at a value of 30.67

Percent Overshoot = 2.2%

Frequency Domain Measurements

I ran multiple simulations in Simulink and recorded the input and output to develop a bode plot of the system. I developed my own data entry tool to calculate and store the gain at different frequencies. Refer to the plot of the data in figure 12 and table 2 for the data below.

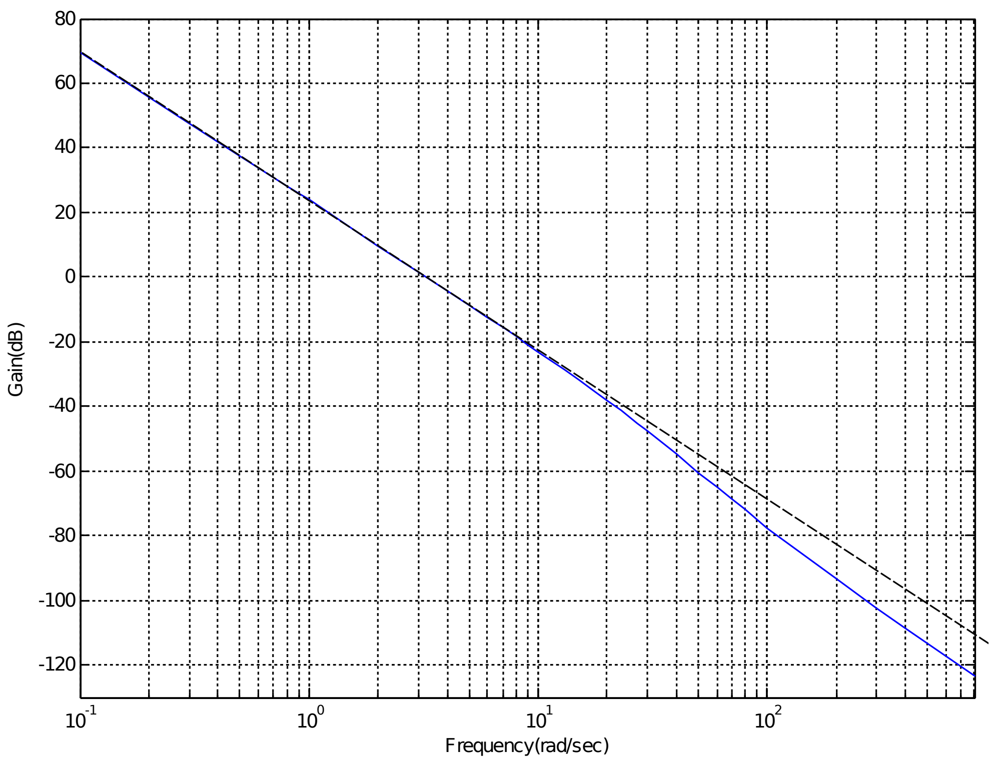


Figure 12 - Magnitude plot obtained by simulation

Table 2

(Frequency(rad/sec), Gain(dB))

0.1	69.549	8	-18.432	60	-64.86
0.3	47.576	9	-20.874	70	-68.693
0.5	37.359	10	-23.074	80	-71.988
0.8	27.956	13	-28.643	90	-74.866
0.9	25.6	15	-31.748	100	-77.417
1	23.878	17	-34.505	200	-93.566
2	9.612	20	-38.172	300	-102.55
3	1.4753	23	-41.391	400	-108.76
4	-4.3162	27	-45.167	800	-123.39
5	-8.8281	30	-47.693		
6	-12.533	40	-54.743		
7	-15.684	50	-60.305		

From this data and graph we can determine 2 things: Cutoff Frequency and location of poles. The cutoff is at 3.2rad/sec as predicted mathematically. There is a pole at the origin and one more at near w =30 rad/sec.

Cutoff Frequency = 3.2 rad/sec

Closed-loop B.W. = Cutoff Frequency = 3.2rad/sec

The next 2 variables determined are Phase Margin and Gain Margin. These require a phase plot of the system. Directly obtaining the phase plot is hard. So I will use the equation:

Phase at w in deg = -90 –atan(w/30) - w*.110*57.3

This semilogx plot of this equation will give us the phase from the magnitude plot.

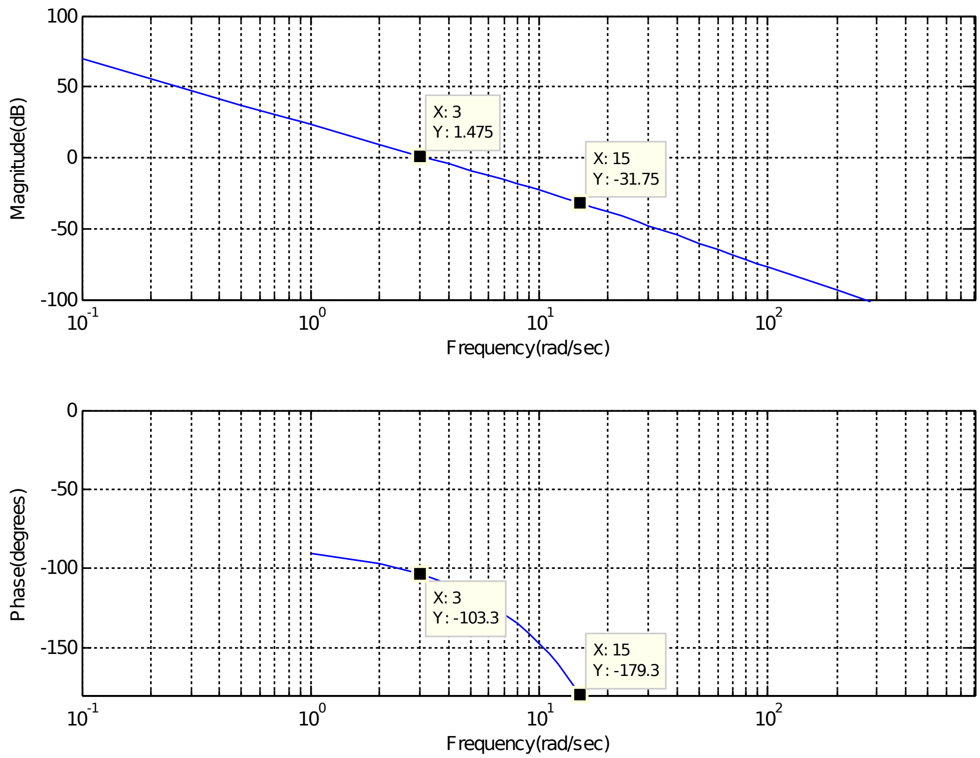


Figure 13 - Bode plot obtained from simulation

Now we are able to measure the last 2 parameters.

Phase Margin = 80 degrees

Gain Margin = 31dB

<u>Comparison of Predicted vs. Simulated measurement of parameters</u>

Parameters	Predicted	Simulated
Ts	1.49 sec	0.85 sec
%O.S.	9.47%	2.20%
Tp	.87sec	0.77 sec
Wc	3.2 rad/sec	3.2 rad/sec
PM	60 deg	80 deg
GM	11.4dB	31dB
BW	3.2 rad/sec	3.2 rad/sec

There is definitely a lot of error in Phase and Gain Margin. This is due to the inaccuracy of the data obtained from the magnitude plot. I'm not worrying about it, everything is within ballpark and this makes me aware that I should take experimental data very carefully to prevent error. The important part of the simulation iteration was to develop tools and methods for data collection using Simulink.

Implementation and Time domain comparison

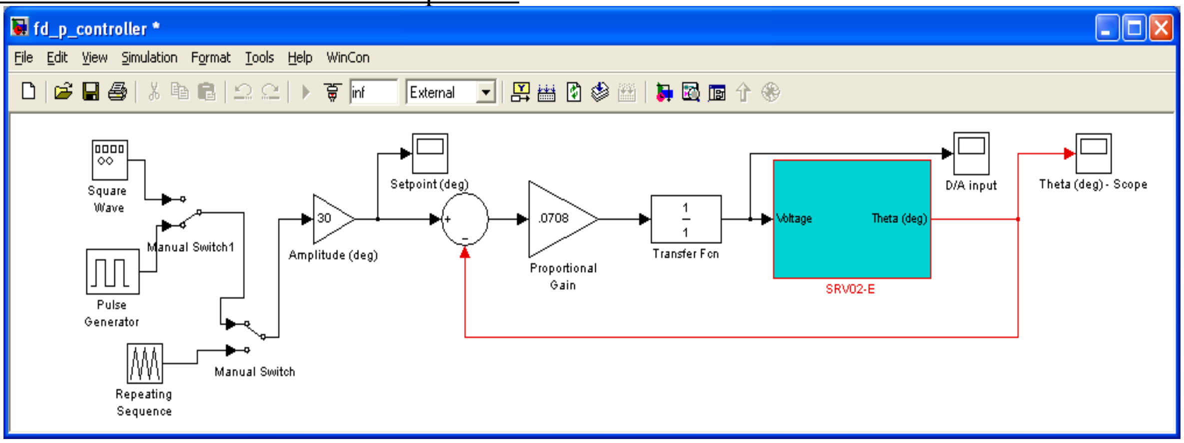


Figure 14 - Simulink model used for Experimental Implementation

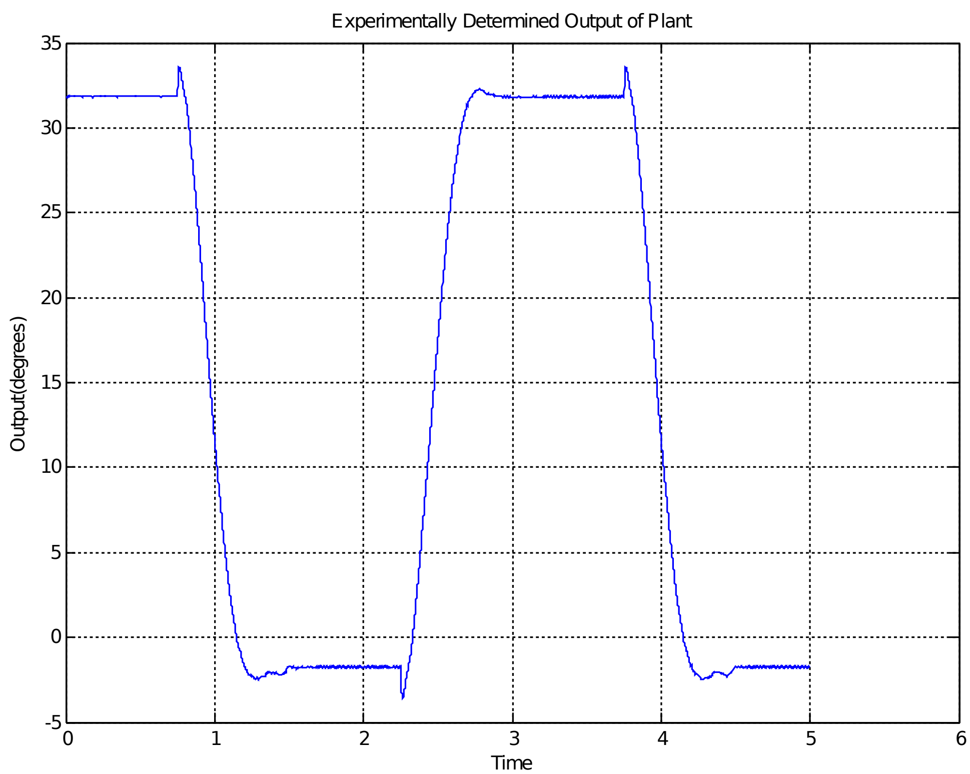


Fig 15 - Experimentally determined output of plant

From this figure we notice that there is a constant steady state error of 1.7 deg. The steady state value is 31.7 deg. The percent overshoot is 1.5%. The settling time within 1% of steady state is .68sec. The time to peak is .644sec. These are all the values of interest as of now. The experimental results are better than predicted and simulated. Refer to Table 3 on the next page.

Table 3 – Comparison of FD proportional controller design predictions

Parameters	Predicted	Simulated	Experimental
Ts	1.49 sec	0.85 sec	.68 sec
%O.S.	9.47%	2.20%	1.50%
Tp	.87sec	0.77 sec	.644sec
Wc	3.2 rad/sec	3.2 rad/sec	
PM	60 deg	80 deg	
GM	11.4dB	31dB	
BW	3.2 rad/sec	3.2 rad/sec	

Frequency Domain Proportional-Derivative Controller Design

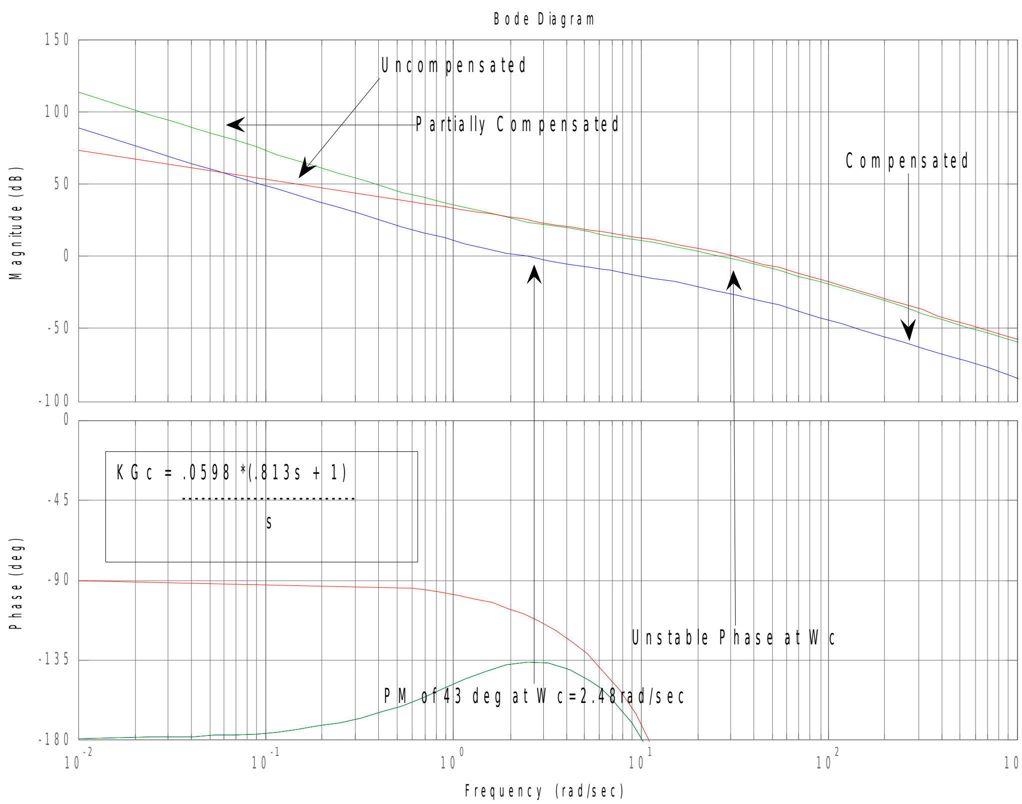


Figure 16 - Bode Plot of compensated and uncompensated plant

<u>Time Domain predictions using Frequency Domain Parameters</u>

Frequency Domain Parameters:

Cutoff Frequency(Wc) = 2.48 rad/sec; PM = 43 deg, GM = 13.4dB

Prediction Parameters: Maximum Peak O.S. (Mp), Closed-Loop Bandwidth(B.W.), Damping Ratio(del), %Overshoot(%O.S.), Natural Frequency(Wn), Damped Frequency(Wd), Settling Time(Ts), Time to Peak(Tp).

Mp = sin(PM) = .682
B.W. = Wc = 2.48 rad/sec
del = PM/100 = .43
%O.S. = 100*e^((-del*pi)/sqrt(1-del^2)) = 22.39%
Wn = (sqrt(sqrt(4*del^4+1) -2*del^2)/Wc)^(-1) = 2.97 rad/sec
Wd = Wn*sqrt(1-del^2) = 2.68 rad/sec
Ts = 4/(del*Wn) = 3.13sec
Tp = pi/(Wn*sqrt(1-del^2)) = 1.17 sec

Time Domain Evaluation in Simulation

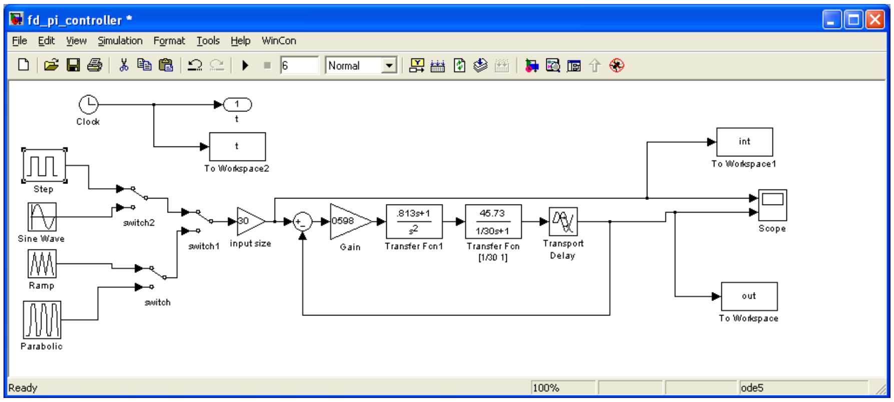


Figure 17 - Simulink Model used for Simulation

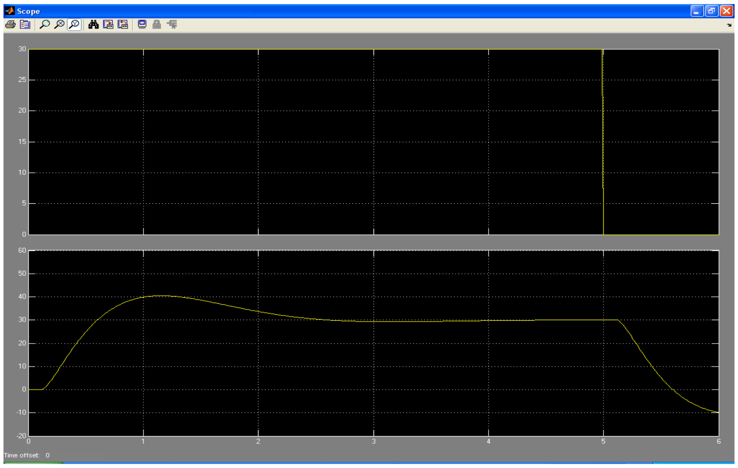


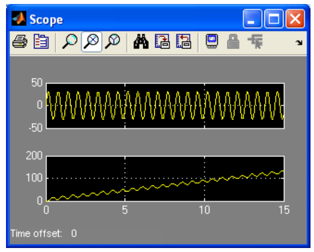
Figure 18 - Time Domain Input and Output

Time domain Measurements

Time to Settle with a tolerance of +-2% = 3.5sec
Time to Peak = 1.15sec at a value of 40.5deg
Percent Overshoot = 30%

Frequency Domain Measurements

I ran multiple simulations in Simulink and noticed the following with an additional integrator.



This ramping behavior of the output makes it difficult to take measurements for experimentally determining the Frequency Response. Therefore, I opt to let the Control Toolbox handle this. Using the LTI viewer tool, I received the following response.

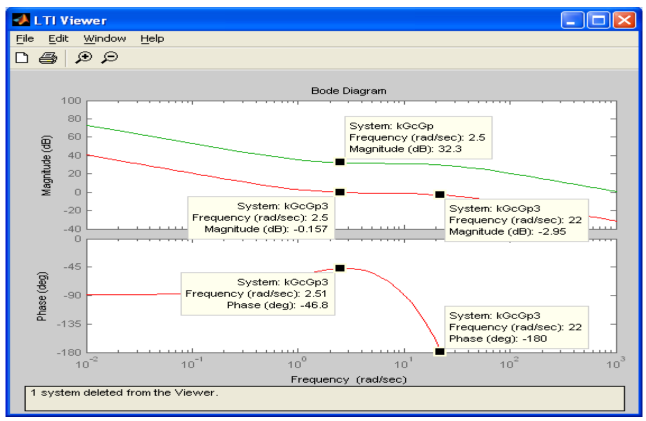


Figure 19 – Bode plot using LTI viewer

From the response above, I noticed that the predicted gain was not enough to meet the PM spec. So I reduced it a bit more for proper LTI viewer behavior. The new K is 1/16.7

Implementation and Time domain comparison

Time to Settle with a tolerance of +-2% = 5.31sec
Time to Peak = 1.06sec at a value of 40.5deg
Percent Overshoot = 30%

Table 4 - Comparison of Predicted vs. Simulated vs. Experimental measurement of parameters

Parameters	Predicted	Simulated	Experimental
Ts	3.13 sec	3.5 sec	3.6 sec
%O.S.	22.39%	30%	30%
Tp	1.17 sec	1.15 sec	1.06 sec
Wc	2.48 rad/sec	2.5 rad/sec	
PM	43deg	43 deg	
GM	13.4db	3db	
BW	2.48 rad/sec	2.5rad/sec	

Table 4 on the previous page shows a comparison between what was expected in design vs. performance in real-time. We see time to peak is a little better but settling time suffers at the cost of increased overshoot. This discrepancy can be attributed to the plant's non-linear characteristics which are not modeled in the Internal Model. After these two rigorous design iterations in the frequency domain it is possible to improve the controller even further using optimum phase margin design. Although comparison data has not been provided for the Optimum Phase Margin Proportional Integral controller, we can judge the optimum performance by simply looking at the bode plot in Figure 20 below. The optimum phase margin controller has the fastest cut-off frequency with a stable phase making it the ideal controller so far.

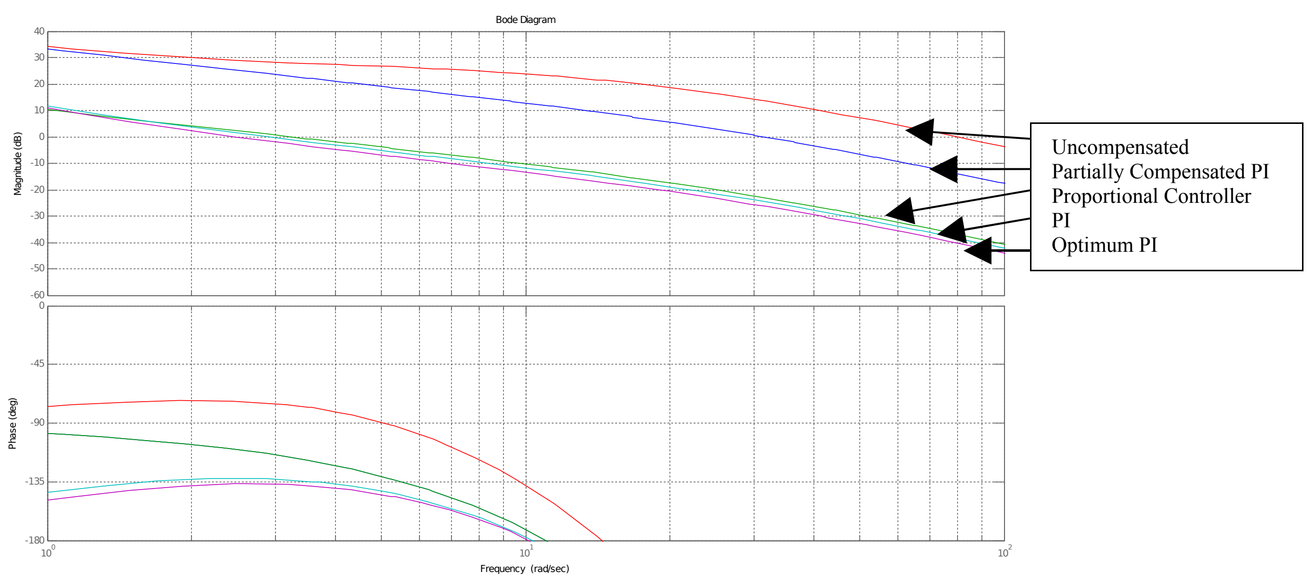


Figure 20 – Bode plot of Frequency Domain controllers

Internal Model Controller Revisit

The architecture relies on the Internal model principle which states that perfect control can be achieved only if the control system encapsulates some representation of the process. If the developed control scheme is based on the exact model of the process, then perfect control is theoretically possible. This is shown in figure 21 on the next page.

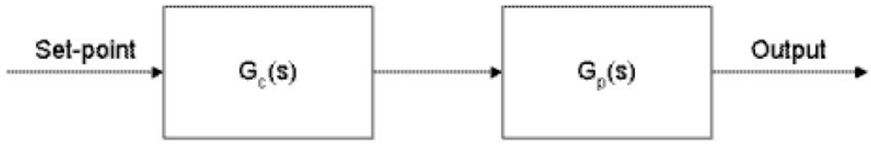


Figure 21 – Internal Model Control in Open Loop

If we let Gp(s) = approx(Gp(s)) and Gc(s) = approx(Gp(s)) ^ -1, then Gp(s)*Gc(s) = approx(Gp(s)) * approx(Gp(s)) ^ -1 = 1. This is the recipe for the perfect tracking system, input/output transfer function equals one. Note that ideal performance is achieved without feedback. Feedback is only necessary when knowledge about the process is incomplete. In practice, model mismatch is common since the process may not be invertible and-or the system is affected by unknown disturbances. To compensate for model mismatch and add robustness to the system, a feedback loop and a controller become part of the architecture as discussed in the IMC section on page 4. This is shown in Figure 22 below.

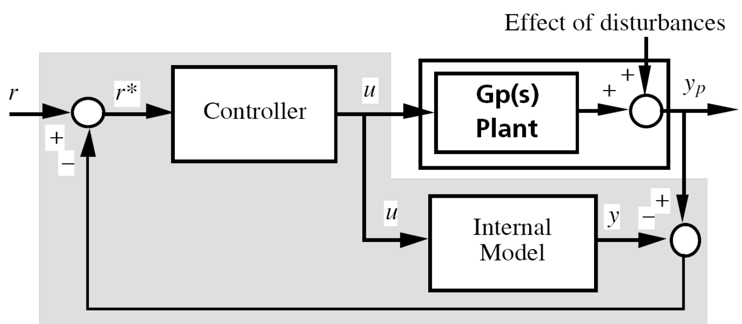


Figure 22 – Internal Model Controller in closed loop

Mathematical analysis of this architecture provides the following advantages.

- Provides time-delay compensation
- At steady-state, the controller will give offset free responses(perfect control at S.S)
- The controller can be used to shape both the input tracking and disturbance rejection responses
- The controller is the inverse of the plant without non-invertible components(time-delay)
- Perfect Tracking is achieved despite model-mismatch, as long as the controller is the perfect inverse of the model.

The encapsulation of the system process can be done in several different ways. They are as follows: 2nd order model(from classical control), State-space model(from modern control), adaptive and non adaptive Look-up tables and artificial neural networks(linear Adaline and non-linear multi-layer Perceptron). For this project, the 2nd order model of the system developed through system identification with the arm was used. The implemented architecture is shown in figure 23 below.

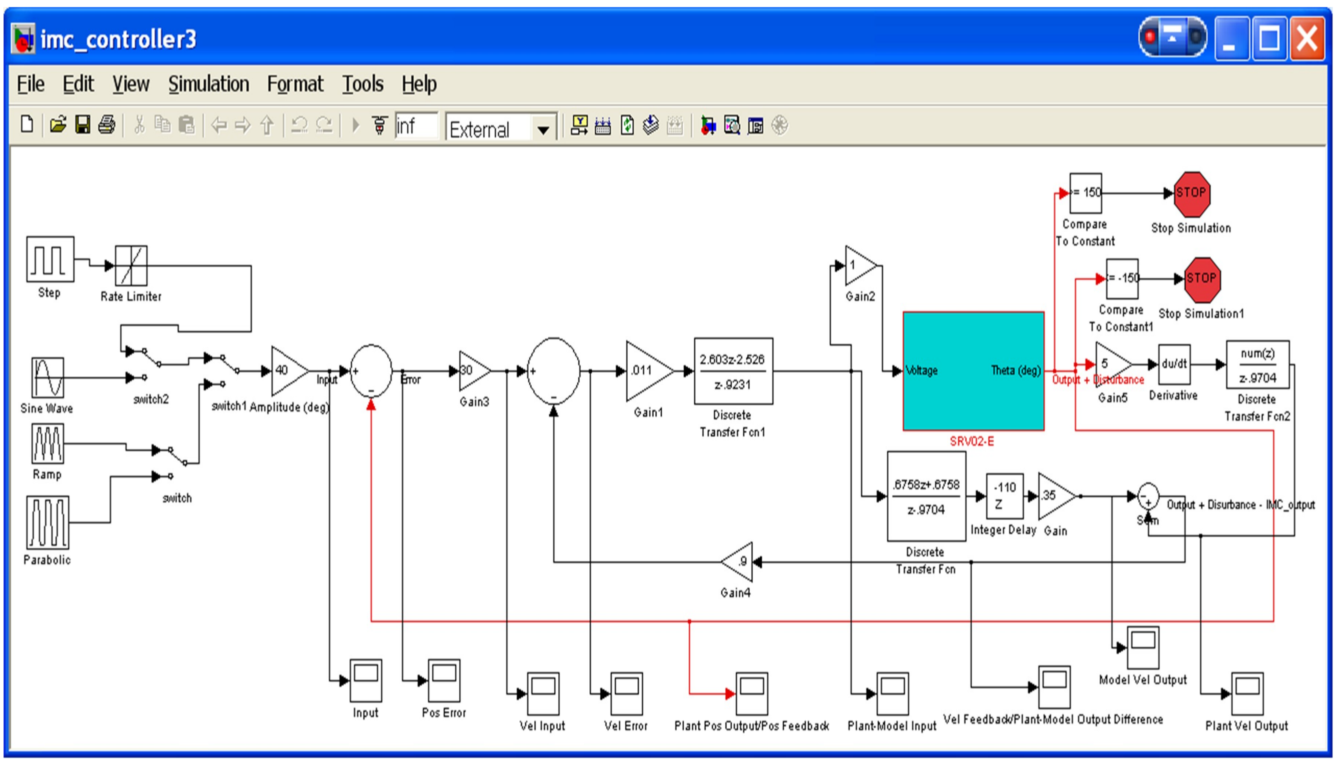


Figure 23 – Final IMC architecture with optimized tuning

There are a few outlying things in this architecture, not discussed before. Using the IMC architecture requires a stable plant. Through experiments it was noticed that the D/A of the system was being saturated. After saturation, the controller becomes unstable and loses control. It was determined that this saturation was due to the integration in the plant. Since the stable plant becomes unstable constantly due to the D/A limitation, another loop was added to the architecture to manage the integrator and stabilize the plant. The position output, as shown in figure 23 above, is differentiated to eliminate the integrator. This changes the model of the process and the controller for the process to exclude the integrator eliminating the D/A saturation problems. After this modification, the plant position was still experiencing steady state error. To eliminate this, a position feedback loop with a proportional gain was added. Then the whole architecture was thoroughly tuned to the edge of instability for an equal balance between speed and control. This was done by modifying the gains in figure 23 above. Each gain is closely tied with the amount of error received from each output(the plant output and the model output). To minimize error from model mismatch the amplitudes of all signals in the system were matched. This balances out the error feedback between the model and the plant for appropriate error minimization by the controller of the plant and in the position feedback loop.

The step response of this system, shown in figure 24 below, is encouraging.

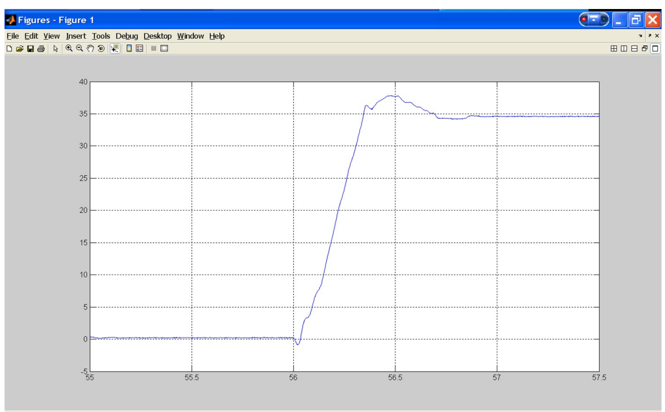


Figure 24 – Step response from tuned IMC architecture

The figure shows that the nonlinear backlash has been mostly eliminated. For disturbance rejection of the 2nd order of freedom, the input was rate limited to +-1.5 maximum slope shown as the rate limiter in figure 23. Further testing of this architecture shows external disturbance rejection and model-mismatch compensation is robust. Testing was done by adding a load to arm gripper.

Conclusion

Internal Model Control (IMC) provides excellent performance for stable plants. Due to integration in the plant model, meaning the plant is marginally stable/unstable, the controller architecture reaches limitations and has to be modified. As shown in the Simulink Block Diagram, the new architecture provides velocity and position feedback with Internal Model for the velocity of the plant. Literature analyzing controller design provides no insight for controlling unstable plants therefore the plant has be stabilized. After plant stabilization, the architecture has to be tuned for speech and stability. The result is a controller capable of handling non-linear plants with robust external disturbance rejection.

Bibliography

Dempsey, Dr. Gary,2007. "Internal Model Control For A Robot Arm With A Low-Frequency Resonant Mode." Memo. Aug. 2007.

Dempsey, Gary, Christopher Spevacek, and Manfred Meissner. Implementation of Conventional and Neural Controllers Using Position and Velocity Feedback. 2000. Dept. of Electrical and Computer Engineering, Bradley U. 18 Oct. 2000 <http://cegt201.bradley.edu/projects/proj2000/prjneurl/>.

Dempsey, Gary, 2006, Joseph Faivre, Kain Osterholt, and Adam Vaccari. Virtual Control Workstation for Adaptive Controller Evaluation. Peoria,IL: Department of Electrical and Computer Engineering, Bradley U. 2006.

Dempsey, Gary, Manfred Meissner, and Christopher Spevacek. Using a CMAC Neural Network in Noisy Enviornments. Peoria, IL: Dept. of Electrical and Computer Engineering, Bradley U, 2005

Edwards, Chris, and Emberly Smith. Design of a Simulink-Based 2-DOF Robot Arm Control Workstation. May 2007. Bradley U. 12 Oct. 2007 <http://cegt201.bradley.edu/projects/proj2007/twodofra/>.

Gu, Jun, James Taylor, and Derek Seward. "Proportional-Integral-Plus Control of an Intelligent Excavator." Computer-Aided Civil & Infrastructure Engineering 19.1 (Jan. 2004): 16-27. Academic Search Premier. EBSCO. 12 Oct. 2007 <http://search.ebscohost.com/login.aspx?direct=true&db=aph&AN=11581088&site=ehost-live>.

Le, Thuong D.,2003. Using A Neural Network Model For A Robot Arm To Design and Implement Conventional and Neural Controllers. Ed. Gary Dempsey. Peoria, IL: Dept. of Electrical and Computer Engineering, Bradley U. 18 Oct. 2007 <http://cegt201.bradley.edu/projects/proj2003/anncntrl>.

Osterholt, Kain, and Adam Vaccari. Simulink Based Control Workstation. Bradley U. 12 Oct. 2007 <http://cegt201.bradley.edu/projects/proj2005/simrobot/>.

Rivals, Isabelle, and Léon Personnaz. "Internal Model Control Using Neural Networks." Proceedings of the IEEE International Symposium on Industrial Electronics (June 1996).

The MathWorks, Inc., 2006, *Simulink (6.4.1), MATLAB (7.2), SimMechanics (2.4), Real-Time Workshop (6.4.1), and the Virtual Reality Toolbox (4.3). 18 Oct. 2007.* <http://www.mathworks.com>.