EEE4118F
Controller Implementation and GA Report

Prepared by:
Kashan Pillay
PLLKAS005

Prepared for:
Professor Edward Boje
Department of Electrical Engineering
University of Cape Town

May 2, 2025

Declaration

1. I know that plagiarism is wrong. Plagiarism is to use another's work and pretend that it is one's own.
2. I have used the IEEE convention for citation and referencing. Each contribution to, and quotation in, this report from the work(s) of other people has been attributed, and has been cited and referenced.
3. This report is my own work.
4. I have not allowed, and will not allow, anyone to copy my work with the intention of passing it off as their own work or part thereof.

May 2, 2025

Kashan Pillay

Date

Contents

Chapter 1

Introduction

This laboratory project encompasses the design and implementation of a cascaded (speed and position) control system for a servo. The system must obtain and sustain the desired performance levels despite variations in friction, inertia, load as well as dynamic gain characteristics. Figure 1.1 below shows the designated system in the laboratory (Workstation 2). The letters on the image refer to the following: (A) - Operational Amplifier Circuit, (B) - Power Supply, (C) - DC Motor, (D) - Tachometer and (E) - Output Potentiometer making this a DC Servo.

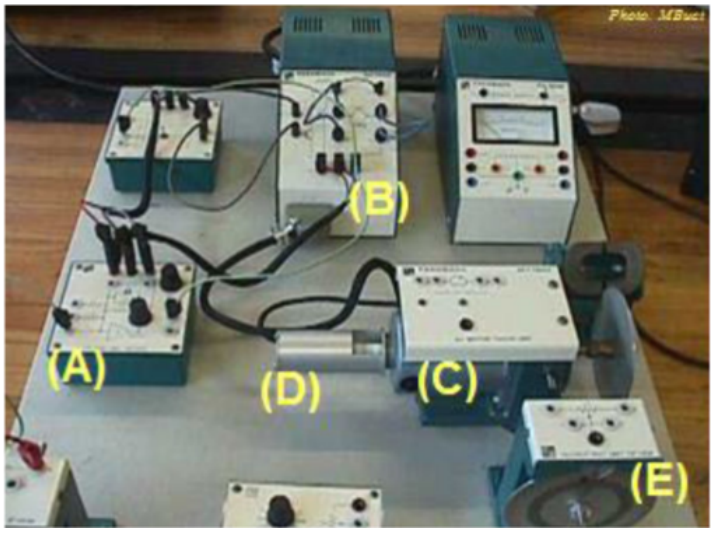


Figure 1.1: The Laboratory Equipment Utilised (Servo System).

The primary objective or aim of this laboratory can be broken up in to two steps. The first is to conduct system identification and system modeling of the assigned work station servo system, with the goal of deriving a dynamic model that can be validated against measurement data using Simulink or the Matlab Toolbox. The second step is to develop (design and simulate) and implement (test and tune) a robust controller. The end goal should result in a proper engineering design of a digital controller for the system, simulated, tested and validated by practical implementation.

Figure 1.2 below shows the block diagram used to model the system.

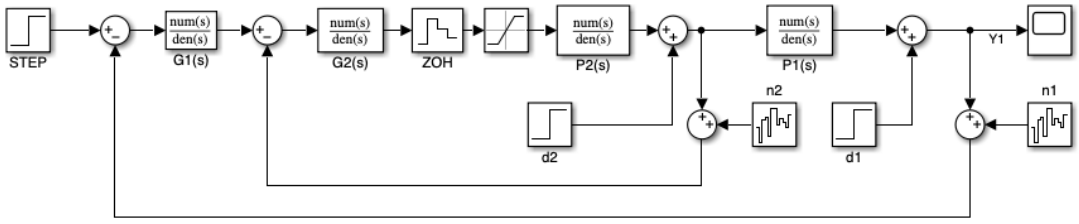


Figure 1.2: A Block Diagram of the Considered System.

Chapter 2

System Identification and Modelling

2.1 Ramp Test

The first step in the system identification process involved performing a ramp test in order to ascertain the dead-band region, so as to avoid this region when performing the coming steps. Figure 2.1 shows the results obtained which clearly indicated that to avoid the dead-band region, a step value of at least 2.15V is required.

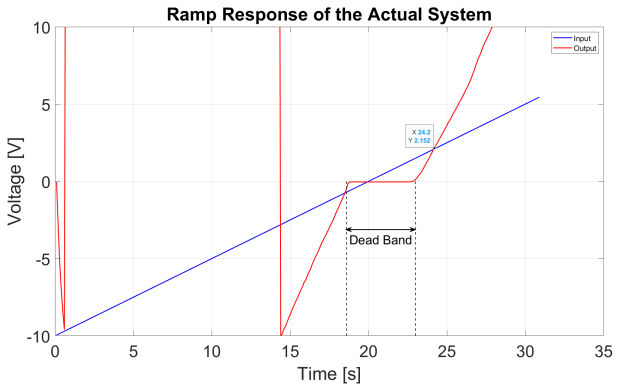


Figure 2.1: Results: Ramp Test Performed on Servo System.

2.2 Step Test

The next step in this process is to perform a Step Test on the system. The purpose of this action is to excite the system with a known input - a step - and analyse the output response in order to characterise the system's behaviour.

To mitigate the effects of static friction, low-speed non-linearities, actuator saturation and insufficient excitation, the system was subjected to a step from a non-zero initial value. This can be seen in Figure 2.2 below.

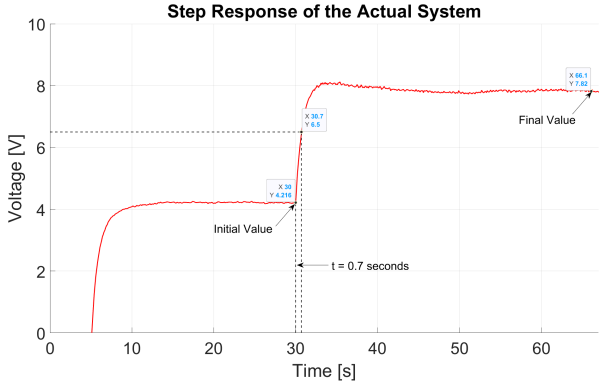
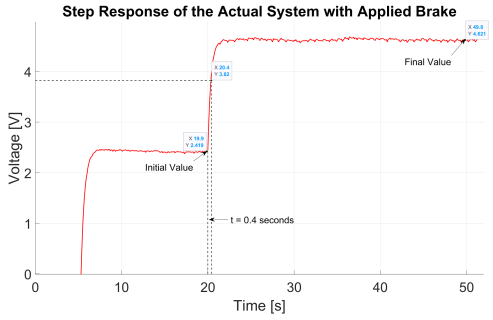
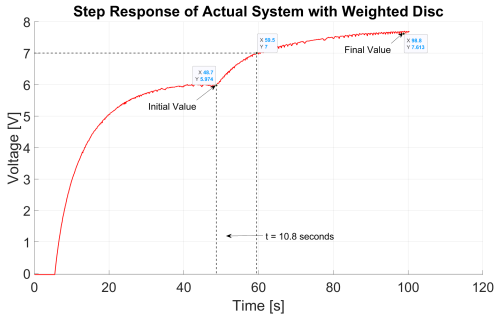


Figure 2.2: Results: Step Test Performed on Servo System.

Subsequently, in order to simulate a varying system, the step test was conducted another two times, each with a different effect. First a magnet brake (Eddy Current Braking) was utilised and the next a weighted disk was utilised (Added Inertia). The results from these two step tests can be seen below in Figure 2.5a and Figure 2.5b respectively.



(a) Results: Step Test Performed with addition of magnetic brake.



(b) Results: Step Test Performed with the addition of weighted disk.

Figure 2.3: Step Tests conducted in order to retrieve varying parameter values.

2.3 Calculations

2.3.1 System Transfer Function

As seen in Figure 2.2, the system response is similar to that of a first order system (the effect of a dominant pole) and so the system can be approximated to be first order. With reference to Figure 2.2, Figure 2.5a and Figure 2.5b the following calculations produce the system's transfer function.

Nominal Plant

$$AB = FinalValue - InitialValue = 7.82 - 4.22$$

$$A(1) = 3.6 \qquad \therefore A = 3.6$$

$$\tau = 30.7s - 30s = 0.7s$$

therefore:

$$P(s) = \frac{A}{1 + s\tau} = \frac{3.6}{1 + 0.7s}$$

Plant Uncertainties

Similarly to the calculations above, the following two transfer functions were determined: $P_{brake}(s) = \frac{A}{1+s\tau} = \frac{2.19}{1+0.4s}$ and $P_{weight}(s) = \frac{A}{1+s\tau} = \frac{4.63}{1+10.8s}$. From these transfer functions the uncertainties for A, $[2.19; 4.63]$ and τ, $[0.4; 10.8]$ were calculated. This allowed for the system to be identified and modelled as:

$$P_2(s) = \frac{A}{1 + s\tau} = \frac{3.6}{1 + 0.7s} \qquad A \in \{2.19, 4.63\}; \tau \in \{0.4, 10.8\}$$

2.3.2 System Frequency Response

Using the corner frequency (fc) below and 5 other frequencies, sinusoidal sweeps were conducted. Table 2.1 below indicates the various results.

$$f_c = \frac{1}{\tau} = \frac{1}{0.7} = 1.42 rad/s$$

Frequency [rad/s]	Magnitude [lin.]	Phase [°]
0.105	3.8163	-11.00
0.85	3.2547	-38.96
0.95	3.1345	-40.82
1.42	2.6109	-44.69
3.35	1.4928	-76.77
10.5	0.5393	-90.24

Table 2.1: Frequency Response Data

2.3.3 Velocity Position Scaling Factor (Integrator Block)

Using the position versus time plot (Figure 1.8 - blue), an average slope value was calculated. In order to determine the scaling factor value 'a', the following was done with Vspeed being the input to the integrator block and the average slope being the output (as seen in the block diagram in Figure 1.1(a)).:

$$a = \frac{Output}{Input} = \frac{\frac{\triangle y}{\triangle t}}{V_{speed}} = \frac{55.91}{3.6} = 15.53 \qquad \therefore P_1 = \frac{15.53}{s}$$

2.4 System Validation

In order to ensure the integrity, and later robustability, the identified and modelled system was validated. This was done by comparing the simulated and the recorded results for both the step response and frequency response.

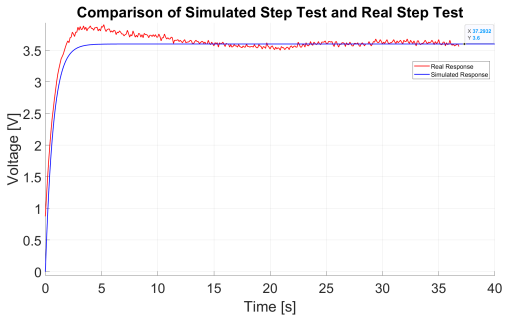
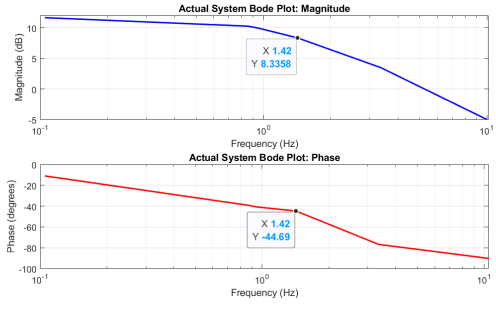
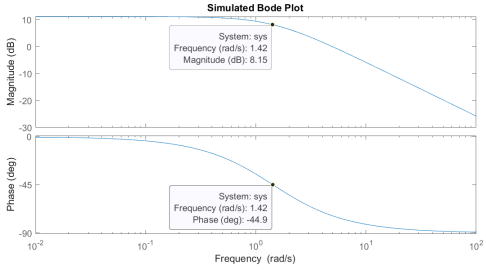


Figure 2.4: Simulation versus Results: Actual and Simulated Step Response in order to validate the system identification and modelling performed.



(a) Results: Frequency Response Data as per Table 2.1.



(b) Simulation: Utilising the calculated transfer function P_2.

Figure 2.5: A side by side comparison of the simulated and actual frequency response of the system to further validate the system identification and modelling.

Chapter 3

Controller Design

3.1 Design Specifications

Prior to the implementation or utilisation of any design tools the following specifications were interpreted:

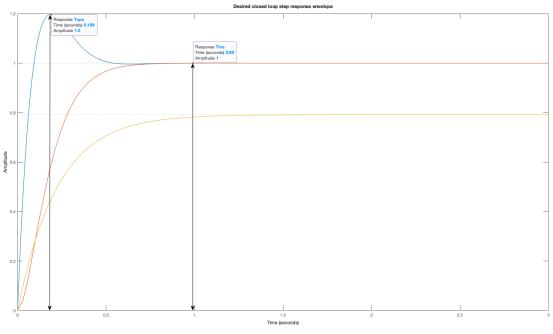
- Settling time < 1 second.
- Zero steady state error.
- Overshoot or undershoot $< 20\%$.
- Output disturbance rejection < 1 second.
- Stable for a varying plant.

Since the task at hand is robust digital feedback design, these transient response specifications were utilised to generate transfer functions as per Robust Control Engineering' by Mario Garcis-Sanz.

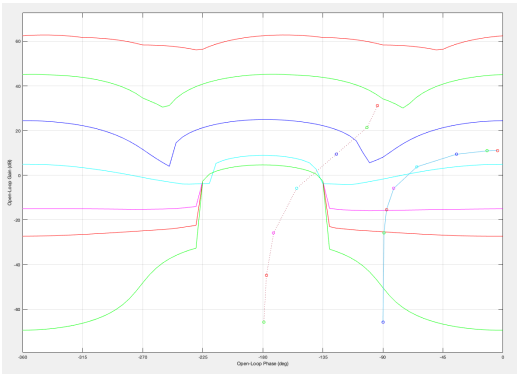
The lower boundary equation is $B_{\text{low}} = \frac{1-\varepsilon_l}{\left(\frac{s}{b_l}+1\right)^2}$, and for the given specifications, $\varepsilon_l = 0$ and $b_l = 7$. The upper boundary equation is $B_{\text{upper}} = \frac{\left(\frac{s}{b_u}+1\right)(1+\varepsilon_u)}{\left(\frac{s}{\omega_n}\right)^2+2\zeta\cdot\frac{s}{\omega_n}+1}$ with $\varepsilon_u = 0$, $b_u = 10.5$, $\zeta = 0.78$ and $\omega_n = \frac{1.25 b_u}{\zeta}$.

3.2 Velocity Loop

Prior to the implementation of any control, the system response (working with the nominal plant $P_2(s) = \frac{A}{\frac{s}{b}+1} = \frac{3.6}{1+0.7s}$) was as seen in Figure 3.1a. The red and blue are indicative of the lower and upper boundary while the orange is the plant. Using the QFT Toolbox, the Figure 3.1a was then plotted on a Inverse Nichols Chart, as seen in Figure 3.1b.



(a) A Simulation of the System without any control mechanism.



(b) A Inverse Nichols plot of the system without any control (solid) and after the addition of an integrator (broken-line).

Figure 3.1: Simulation: Step Response and Frequency Response Simulations.

As seen in Figure 3.1b, the first step in the implementation of a controller, in order to meet the design

specification requiring zero steady state error, is by the use of an integrator. After the addition of the integrator, a gain addition is required in order to ensure that the low specification boundaries are not violated. Figure 3.2 shows how the addition of 29 (29.25dB) satisfies the low frequency boundaries. Since overshoot is a feature of predominantly the low frequencies, we can assume this specification to be met. The controller design so far is $G_2(s) = \frac{29}{2}(\frac{s}{b} + 1)$.

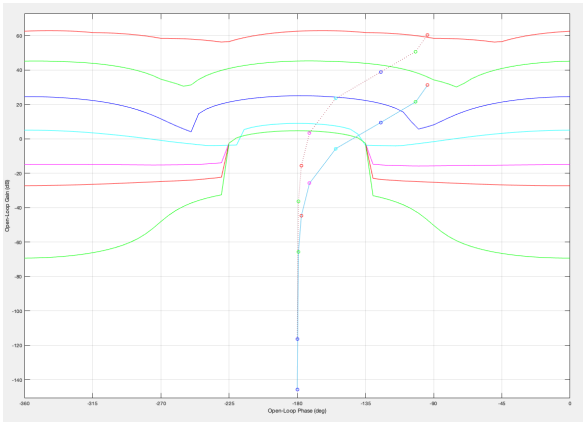


Figure 3.2: Digital Design: Addition of gain to satisfy low frequency specifications.

In order to ensure that the high frequency specifications are not violated, a phase lead at the 1rad/s point is considered. In order to accurately account for the effects of digital design (sampling) an additional 15 of phase is added. As suggested by the Inverse Nichols Plot on the QFT Toolbox, around 55 degrees of phase is required, resulting a total 70 degree phase lead being added. This is done as follows:

$$zero : (\frac{s}{b} + 1)$$

$$b = \frac{\omega}{tan(\alpha)} = \frac{1}{tan(55 + 15)} = 0.3639$$

As a result, the final G2 controller was:

$$G_2(s) = \frac{29}{s}(\frac{s}{0.3639} + 1)$$

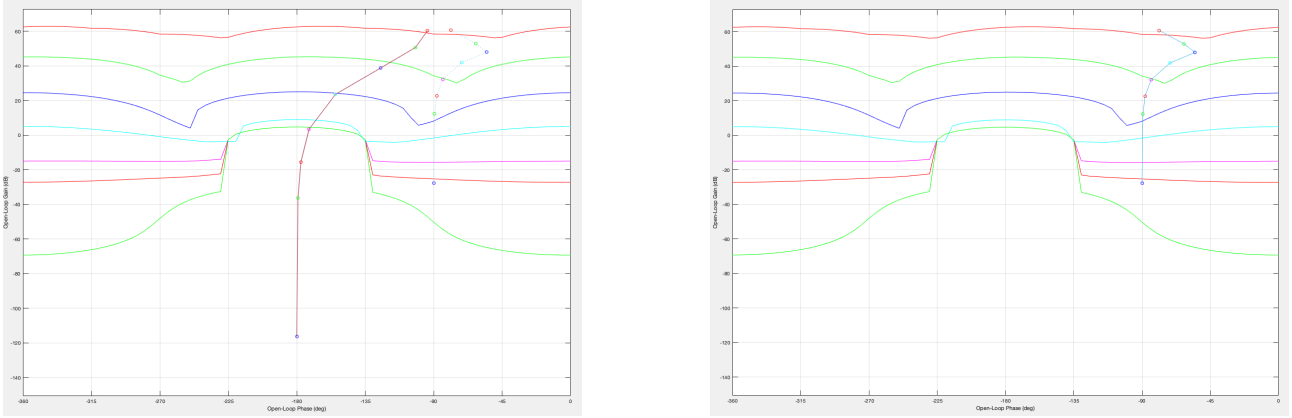
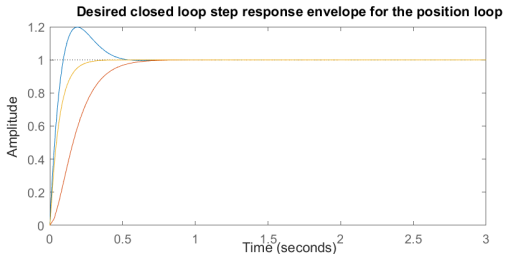


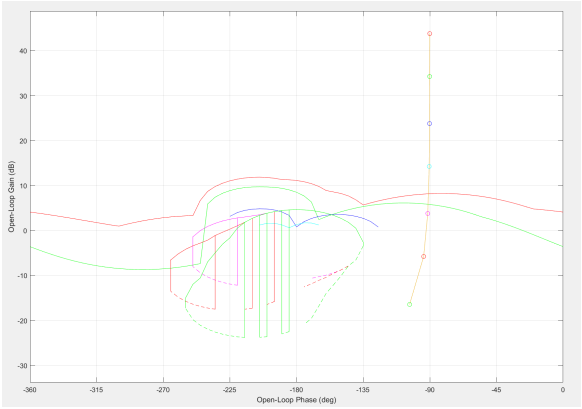
Figure 3.3: Two Inverse Nichols charts that show the proposed and then implemented phase lead alongside the integral and proportional aspects of the design.

3.3 Position Loop

For fast attenuation of output disturbances as well as noise, the inner loop can be simplified to this single transfer function, $\frac{G_2(s)P_2(s)}{1+G_2(s)P_2(s)}$ which in turn allows for the position loop plant to be written as $\frac{G_2(s)P_2(s)}{1+G_2(s)P_2(s)}P_1(s)$. Using this new plant, the same steps as mentioned above were completed in order to utilise the QFT Toolbox. However, prior to the implementation of any control methods, there were no violations of any of the boundaries across a wide frequency spectrum. This can be seen in Figure 3.4. As a result, the G1 controller is $G_1(s) = 1$. This is due to the fact that $G_2(s)$ provides for a sufficient transient response such that the system abides by the set out design requirements.



(a) Tracking Specifications for Position Loop with Nominal Plant Satisfied



(b) Inverse Nichols Chart Specifications with Nominal Plant Satisfied

Figure 3.4: Simulation: No boundary violations.

3.4 Theoretical Controller

As per the above justifications, the final theoretical velocity controller is $G_2(s) = \frac{29}{s}(\frac{s}{0.3639} + 1)$ and the final theoretical position look controller is $G_1(s) = 1$ for the plants $P_1(s) = \frac{15.53}{s}$ and $P_2(s) = \frac{3.6}{(0.7s+1)}$.

Chapter 4

Controller Simulation and Design Evaluation

4.1 Design Limitations

A limitation of the servo motor system is that of the maximum unsaturated step size of 10. This is represented by the saturation block as seen Figure 1.2. Due to this, the designed controller - while satisfying all the necessary continuous time domain specifications - results in what is known as a 'bang bang' controller. In order to mitigate this, an option could be to lower the gain of the velocity loop controller, while keeping the position loop controller as $G_1 = 1$. It can be seen in Figure 3.4, that a significant drop in gain is viable for the overall system (position loop), while ensuring that there are no violations of any specification bounds after taking into account both the velocity and position loops.

Figure 4.1 shows a Inverse Nichols plot of the new system - both the position and velocity loops - with the gain of controller G_2 decreased to 1, resulting in $G_2(s) = \frac{1}{s}(\frac{s}{0.3639} + 1)$ being the velocity loop controller.

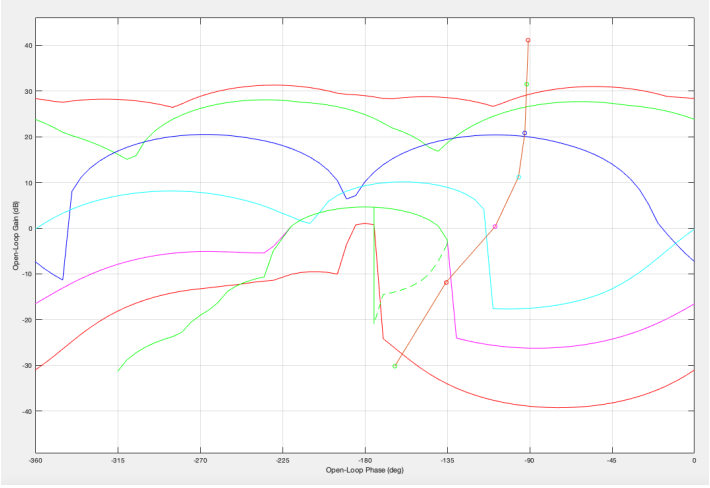


Figure 4.1: Design: Decreased Gain Controller G_2.

4.2 Design Simulations

Utilising the updated controller design, these simulations are conducted with $P_1(s) = \frac{15.53}{s}$, $P_2(s) = \frac{3.6}{0.7s+1}$ for $A \in \{2.19, 4.63\}; \tau \in \{0.4, 10.8\}$, $G_1(s) = 1$ and $G_2(s) = \frac{1}{s}(\frac{s}{0.3639} + 1)$. The purpose of these simulations is to evaluate the performance of the controller prior to practical implementation and testing.

4.2.1 Nominal Plant

The purpose of the output disturbance is to evaluate the performance of the controller under a known disturbance and the purpose of the varied parameters is to test the robustness of the controller.

In Figure 4.2, a step (at 0.5s) was applied to the nominal plant values (as per the block diagram in Figure 1.2) and the response captured as seen. It can be seen that the overshoot was less than 20% (it was 5%) with a settling time, as well as output disturbance rejection, being less than 1 second. The output disturbance was modelled as a step (0.5V) at 5 seconds. The system rejects the output disturbance within 1 second and is able to return to the previous value.

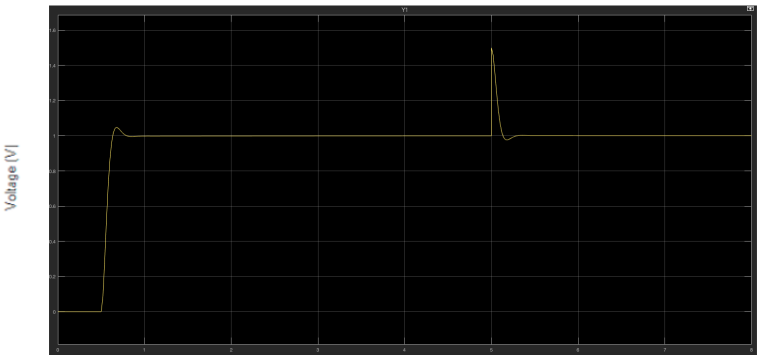


Figure 4.2: Simulation: The Nominal Plant with an Output Disturbance.

4.2.2 Plant Variations

In Figure 4.3 and 4.4, a step input was applied (1V at 0.5s) to the upper and lower varying transfer functions (when the weighted disk and magnetic brake were used respectively). The primary requirement for plant variations is stability and it can be seen in both plots that the requirement was fulfilled. However, it can be noted that for the added inertia, the system has a much longer settling time (slower response and high overshoot).

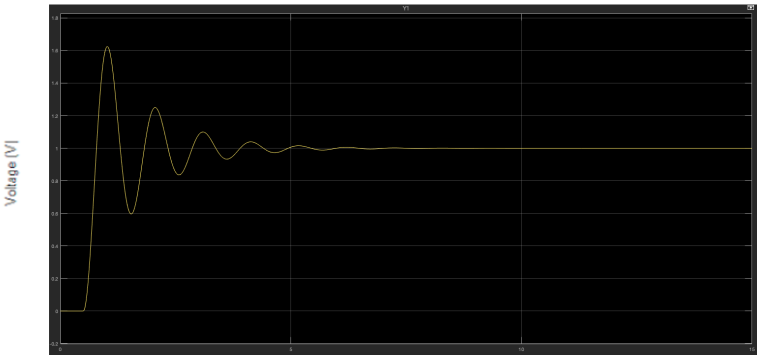


Figure 4.3: Simulation: Variation of the plant (Added Inertia).

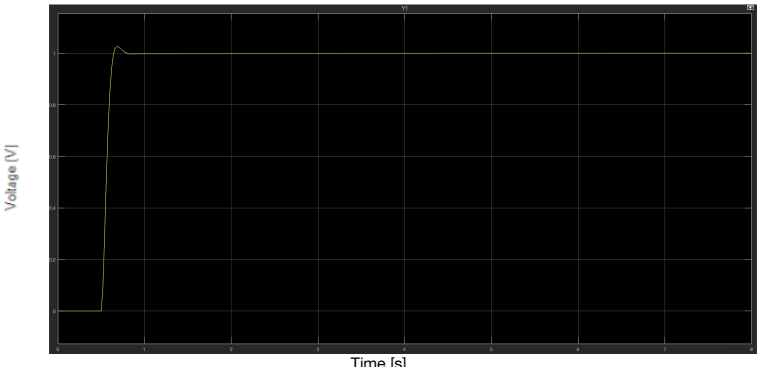


Figure 4.4: Simulation: Variation of the plant (Magnetic Brake).

Chapter 5

Controller Implementation and Testing

5.1 Digital Implementation

In order to implement the continuous-time controller digitally, it needs to be discretized (converted into discrete form). Since G_1 is simply 1, only G_2 needs to be converted. This is done as follows with a sampling time of 100ms:

$$\begin{aligned} G_2(s) = \frac{1}{s}(\frac{s}{0.3639} + 1) &= 1[\frac{1}{w}(\frac{w}{0.3639} + 1)] \\ &= 2.748 + \frac{1}{w} \qquad for \quad w = \frac{2(z-1)}{T(z+1)} \\ &= 2.748 + \frac{T(z+1)}{2(z-1)} \\ &= \frac{2.748(z-1) + \frac{T}{2}(z+1)}{(z-1)} \qquad for \quad T = 100ms \\ &= \frac{2.748z - 2.748 + 0.05z + 0.05}{z-1} \\ &= \frac{2.798z - 2.698}{z-1} \end{aligned}$$

From the above equation, the inverse z-transform properties can be utilised to produce the difference equation as follows:

$$\begin{aligned} H[z] = \frac{Y[z]}{X[z]} &= \frac{2.798z - 2.698}{z-1} \\ Y[z](z-1) &= X[z](2.798z - 2.698) \\ Y[z]z - Y[z] &= X[z](2.798z) - X[z](2.698) \\ Y[n+1] - Y[n] &= X[n+1](2.798) - X[n](2.698) \end{aligned}$$

This is analogous to the controller algorithm:

$$u[i+1] = u[i] + e[i+1](2.798) - e[i](2.698)$$

The controller was implemented using VSCode(C/C++) using the discretized difference equation as seen above. The above mentioned sampling time was utilised in order to mitigate any associated aliasing or loss of performance taking into account limitations (such as resolution) of the DAC.

```
// Controller
double Error = rt_SetPoint - yt_PlantOutput; //Position Loop

double Error2 = Error - Velocity; // Velocity Loop

if (Error2 < 0.0999999999 & Error2 > -0.0999999999) { Error2 = 0; } // Mitigation of integral windup

double Output = prevOutput + 2.798 * Error2 - 2.698 * prevError2; // Implementation of Difference Equation

Output = Output * Gain;

double ut_Output = Output;

// Send output to DAQ
string ControlLoop = comboBoxControlLoop.Text;
if (ControlLoop == "OPEN")
{ ut_Output = rt_SetPoint; }

//INPUT DISTURBANCE
//double InputDisdurbance = 1;
//ut_Output = ut_Output + InputDisdurbance;

if (ut_Output > 9.8) { ut_Output = 9.8; } //Saturation Bounds (DAC Protection)
if (ut_Output < -9.8) { ut_Output = -9.8; }

DAQ.Output(0, ut_Output);

prevError2 = Error2; // creating the previous values required as per the Difference Equation.
prevOutput = Output;

ut_PlantInput = ut_Output;        // Read and Display Controller Output
textBoxControllerOutput_ut.Text = ut_PlantInput.ToString("0.000");
WriteData += "," + ut_PlantInput.ToString("0.000");
chart1.Series["u(t) Input"].Points.AddXY(XValue, ut_PlantInput);
DisplayError(Error);
```

Figure 5.1: How the controller was implemented in code.

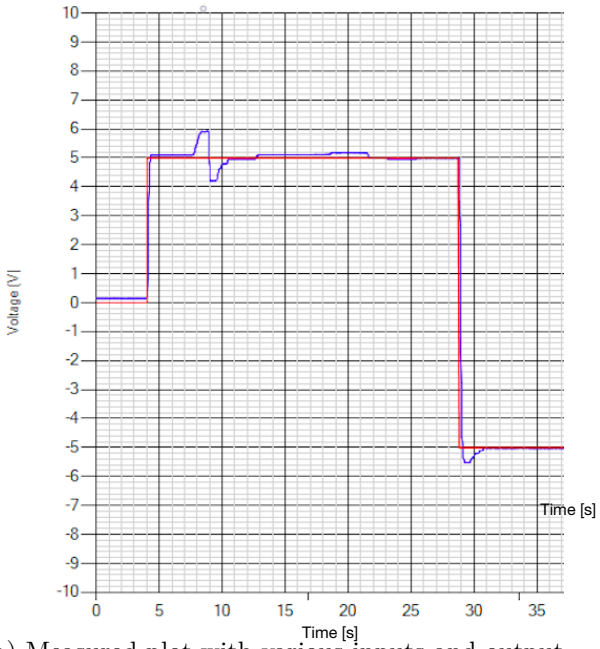
5.2 Testing Results

The purpose of the analysis of these results are to validate the results using real world conditions, to ensure that adequate precautions have been followed to account for the impact of discretization and to evaluate controller robustness and stability. While simulations offered a controlled environment in which to evaluate the transient behaviour, physical testing provides the platform to fine-tune the gain. Ultimately, this final step closes the triangle of knowledge (theory, simulation and measurement).

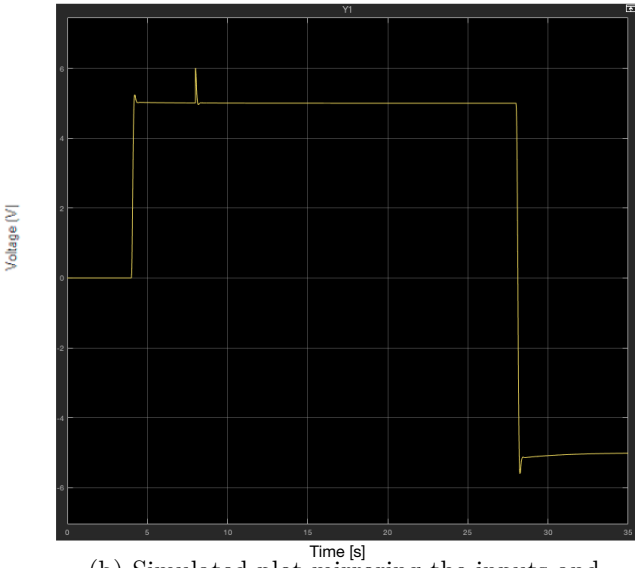
In the measured plots below, the red line is the set-point and the blue line is the system output.

5.2.1 Nominal Plant

Figure 5.2 shows a side-by-side comparison between the measured data and the simulation. The system underwent a set-point change to 5V at 4 seconds, and output disturbance of 1V at 8 seconds, and then a set-point change to -5V at 29 seconds. The simulation was then run under the same conditions. Firstly, Figure 5.2a shows that the implemented controller has zero steady state error (set point tracking) and is in keeping with that of the simulation 5.2b. The settling time and output disturbance rejection is less than 1 second -satisfying the design requirements - however slightly slower than the simulation. This can be attributed to the noise of the mechanical system. Lastly, the implemented controller had less than 20% of overshoot which is in keeping with that of the controller. The slight deviation can be attributed to non-linearities of the system (such as real-world friction). Ultimately, the design specifications under the nominal values are fulfilled.



(a) Measured plot with various inputs and output disturbances.

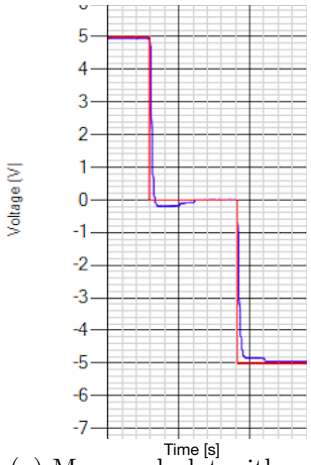


(b) Simulated plot mirroring the inputs and output disturbances of the measured results.

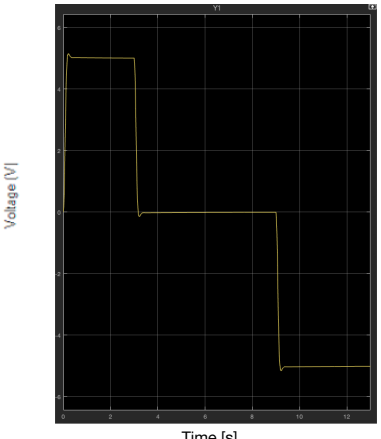
Figure 5.2: A comparison between measurement and simulation of the nominal plant

5.2.2 Plant Variations

As previously mentioned, the purpose of this test (both in simulation and in practice) is to further test the robustness of the controller. As seen in Figure 5.3, the set-point of the system was first set to 5, then 0 and then -5. The outcome of this test should, and does, show that the system is stable under these varied parameters. However, it should be noted that there is overshoot and undershoot,which is not present in the simulation. This deviation can be attributed to the effects of approximating the system to that of first order.



(a) Measured plot with various inputs.



(b) Simulated plot mirroring the inputs of the measured results.

Figure 5.3: A comparison between measurement and simulation of a varied parameter plant (magnetic brake).

5.2.3 External Control

This last plot, seen in Figure 5.4, shows the response (in blue) to the random set-point (red) which is controlled from the servo-motor system (externally). It can be seen that the system tracks the set-point with little to no error, a minimal settling time and miniscule overshoot. This further attests to the robustness and real-world applicability of the controller.

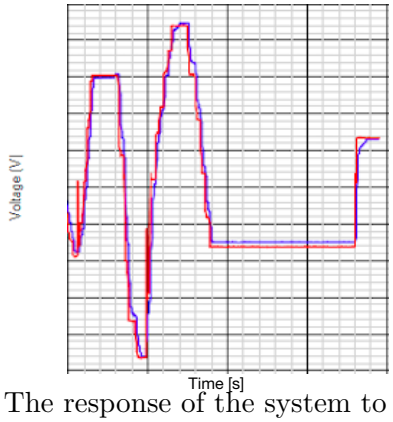


Figure 5.4: Measured: The response of the system to an external input (control).

Chapter 6

Discussion and Conclusion

6.1 Conclusion

This laboratory project - through continuous development over a fixed period - designed, simulated and implemented a controller for a servo motor system identified and modelled, as per the block diagram in Figure 1.2, using $P_1(s) = \frac{15.53}{s}$ and $P_2(s) = \frac{3.6}{0.7s+1}$ for $A \in \{2.19, 4.63\}; \tau \in \{0.4, 10.8\}$. The final tested and implemented controllers were $G_1(s) = 1$ and $G_2(s) = \frac{1}{s}(\frac{s}{0.3969} + 1)$. The control system was developed and tested in simulation under nominal conditions, with varied set points and output disturbances, and under varied parameter values. The design was then evaluated, edited and simulated again. It was clear from these simulations that the performance metrics (design specifications) such as zero steady state error, less than 20% overshoot and a settling time and output disturbance rejection in less than 1 second were evaluated and successfully fulfilled. Additionally, tests for robustness were conducted and were successful. Following this, the controllers were discretized and practically implemented - enabling real time control of the servo motor system. The implemented system, as with the updated simulated system, demonstrated adherence to the set out specifications and robustness expectations under nominal values and stability under varied parameter values. This, in turn, validates the overall design process (completing the triangle of knowledge from theory to simulation to implementation).

6.2 Discussion and Recommendations

While the simulations indicated both robustness and stability while successfully fulfilling the design requirements, there were various discrepancies which were observed in the translation from theoretical knowledge to practical implementation. The measured responses being slightly slower and producing slightly higher overshoot values can be attributed to limitations of the system such as non-linearities (like saturation and dead-band regions) as well as real-time processing delays and machine uncertainties. This was evident in spite of the efforts put in place to mitigate these setbacks.

There are various ways in which this design could be improved. Changing the sampling time (while not resulting in a value too fast such that is results in an increased computational load or amplification of noise) from 100ms to possibly 50ms. Additionally, the introduction of a phase shift in the position loop, instead of simple $G_1(s) = 1$, could allow for a lower gain while satisfying the high frequency boundaries. The lower gain simply allows for the system to conservatively avoid actuator saturation (+10/-10 in this instance).