# Laboratory Experiment III: Implementation of a Position Controller

## MTRN3020 - Modelling and Control of Mechatronic Systems

I verify that the contents of this report are my own work

Zachary Hamid
z5059915
13/10/2017

# Contents

# 1 Introduction

The purpose of this experiment was to design and implement a discrete controller that controlled the position of a system comprised of a shaft arm that is driven by a motor via a gear box as well as apply that controller to the same system with slight alterations to determine the effect of designing a controller without knowing the exact system parameters. Certain design specifications had to be adhered to when designing the discrete controller.

The first part of the experiment mainly served the purpose of verifying the controller design to ensure any obtained results for the second part of the experiment were accurate.

The second part of the experiment dealt with applying the designed controller to a system with slightly different parameters than the controller had been designed for to observe the effects on its ability to control the system.

# 2 Aim

The aim of this laboratory experiment was to maintain a desired arm position, $\omega$, using a position controller implemented into a system consisting of a motor, gearbox, revolute joint and arm. The controller was to be designed using Ragazzini's method of direct analytical design for the internal velocity feedback loop of the Simulink diagram seen in Section 5, and the root-locus method to determine a suitable gain, $K$, for the external position feedback loop again seen in Section 5's Simulink diagram. The design had to satisfy the following specifications:

- A speed control time constant of $\tau = 37$ ms (from individual design data)
- A sampling frequency of $\boldsymbol{f_s} = \frac{1}{0.006} = 166.7$ Hz (from individual design data)
  - This corresponds to a sampling time of $T = 0.006$ s
- Zero steady state errors for both velocity and position loops
- Critically damped position controller response

# 3 Experimental Procedure

There were two parts to this experimental procedure.
For the first part, the crane arm of the system was set to a specific length related to the ROT335.m file; this is the system that was used to design the discrete controller. A sequence of desired angular positions was then input to the system at intervals of $400T = 400 * 0.006 = 2.4$s (T is the sampling time given in Section 2), and the designed controller ensured the crane arm rotated to these positions. The position sequence in encoder counts was:

$$3072 \rightarrow 1024 \rightarrow 2048 \rightarrow 2560 \rightarrow 0$$

Where the encoder counts $c = \dfrac{8192}{2\pi}$ rads.

The second part of the experiment had the crane arm length set to correspond with the ROT445.m file. This system was slightly different to the one the discrete controller was designed for as the crane arm length was different. The same sequence of desired positions was input to this system and the controller attempted to make the crane arm rotate to these positions.

# 4 Controller Design

To find the required discrete transfer function of the controller, the first thing that must be done is obtain some constants from experimental data related to the system. This is done using MATLAB's curve-fitting tool, *cftool*. By fitting to the first order response part of the data obtained from ROT335.m [motor speed (counts/s) against time (s)] using a fit equation of $f(t) = v(t) * A(1 - e^{-Bt})$, where $B = \frac{1}{\tau}$ and $v(t) = 24\text{V}$, the following fit was found:

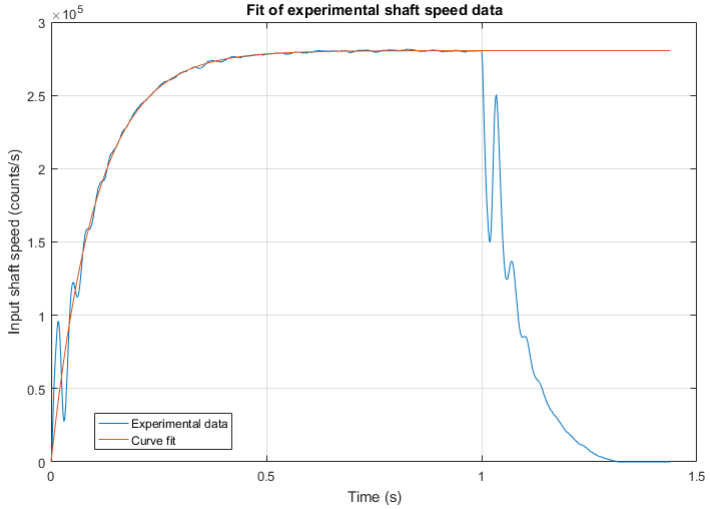


Figure 1: MATLAB *cftool* curve-fit to ROT335.m data

It can be seen that the *cftool* gives a good fit to the data, and so the constants obtained from this will be accurate enough for the controller design. From MATLAB's *cftool*, the constants were found to be:

$$A = 11687.5 \text{ counts/s/V} = 8.964200229206716 \text{ rad/s} \tag{1}$$

$$B = 9.624 \rightarrow \tau = \frac{1}{9.624} = 0.103906899418121 \text{ s} \tag{2}$$

Using the above results, a transfer function relating the applied voltage and the input shaft angular velocity (in rad/s) can be derived as:

$$G'_P(s) = \frac{A}{1 + \tau s} \tag{3}$$

However, the velocity can't be sampled, so an integrator must be added to the transfer function in order to relate the applied voltage to rad, so the final plant is:

$$G_P(s) = \frac{A}{s(1 + \tau s)} \tag{4}$$

Incorporating a zero-order hold as well as the differentiator from the block diagram given in the lab document[1], the following expression for $G_p(z)$ is obtained as:

$$G_P(z) = \mathcal{Z}\left[\frac{(1 - e^{-sT})}{s} G_P(s) \frac{(z - 1)}{zT}\right] \tag{5}$$

Where $T = 0.006$s, from Section 2 above. Simplifying Eq. (5) and manipulating the transfer function inside the z-transform brackets so the z-transform tables can be used gives:

$$G_P(z) = \frac{(z-1)}{zT}(1-z^{-1})\mathcal{Z}\left[\frac{G_P(s)}{s}\right] = A\frac{(z-1)^2}{z^2T}\mathcal{Z}\left[\frac{1/\tau}{s^2(s+1/\tau)}\right] \tag{6}$$

Applying the z-transform tables for the form $\dfrac{a}{s^2(s+a)}$:

$$G_P(z) = A\frac{(z-1)^2}{z^2T}\frac{z[((T/\tau)-1+e^{-T/\tau})z + (1-e^{-T/\tau}-(T/\tau)e^{-T/\tau})]}{(1/\tau)(z-1)^2(z-e^{-T/\tau})} \tag{7}$$

To reduce the above result, we set $\alpha = T/\tau$, $\beta = e^{-T/\tau}$, and factorise the numerator:

$$G_P(z) = \frac{A}{\alpha}(\alpha - 1 + \beta)\frac{(z + \frac{(1-\beta-\alpha\beta)}{(\alpha-1+\beta)})}{z(z-\beta)} \tag{8}$$

Calculating the zero above using result (2) and $T = 0.006$ s from Section 2, to determine if it is ringing or not we have:

$$\frac{(1-\beta-\alpha\beta)}{(\alpha-1+\beta)} = \frac{0.001604373430646}{0.001635552747503} = 0.980936526257202 \tag{9}$$

This is a stable but ringing zero since it is a positive value $< 1$, so it must be eliminated in the numerator of $F(z)$ to ensure that it isn't cancelled in the plant. The pole of (9) must also be calculated to determine if it is ringing:

$$-\beta = -e^{-T/\tau} = -0.943891552747503 \tag{10}$$

The above pole is a stable pole that doesn't ring since it is negative and $> -1$, this means it doesn't need to be included in the numerator of $1 - F(z)$.

Keeping these variables to retain accuracy by eliminating rounding error as a factor, to determine $F(z)$, we first find the desired closed-loop s-domain root $s = -\frac{1}{\tau_d}$, where $\tau_d = 0.037$s from Section 2 above, and so the z-domain root is:

$$z = e^{sT} = e^{-T/\tau_d} \tag{11}$$

From this result we have the starting point for $F(z)$ as:

$$F(z) = \frac{b_0(z+z_1)}{(z-e^{-T/\tau_d})} \tag{12}$$

Where $z_1$ is the ringing zero from (8). Now, checking whether the causality constraint holds; the pole-zero deficiency of $F(z)$, $\mathcal{D}\{F(z)\} = 0$, but the pole-zero deficiency of $G_P(z)$, $\mathcal{D}\{G_P(z)\} = 1$. Since $\mathcal{D}\{F(z)\} < \mathcal{D}\{G_P(z)\}$, the causality constraint is violated. Therefore the new $F(z)$ must be:

$$F(z) = \frac{b_0(z+z_1)}{z(z-e^{-T/\tau_d})} \tag{13}$$

In order to determine the last unknown, $b_0$, the desire of zero steady-state error is used, that is $F(z) = 1$, this happens at $z = 1$ and therefore:

$$F(1) = \frac{b_0(1+z_1)}{(1-e^{-T/\tau_d})} = 1 \rightarrow b_0 = \frac{(1-e^{-T/\tau_d})}{(1+z_1)} \tag{14}$$

The above will be left as $b_0$ for now as explained before to remove rounding errors. Now everything required to design the controller has been found, we can now derive the controller equation using the following expression (by Ragazzini's method):

$$G_c(z) = \frac{1}{G_p(z)} \frac{F(z)}{1 - F(z)} \tag{15}$$

Substituting all known values, the following is obtained:

$$G_c(z) = \frac{\alpha}{A(\alpha - 1 + \beta)} \frac{z(z - \beta)}{(z + \frac{(1-\beta-\alpha\beta)}{(\alpha-1+\beta)})} \frac{\frac{b_0(z+z_1)}{z(z-e^{-T/\tau_d})}}{(1 - \frac{b_0(z+z_1)}{z(z-e^{-T/\tau_d})})} \tag{16}$$

For ease of simplification (and to retain accuracy), setting $C = \frac{\alpha}{A(\alpha-1+\beta)}$, $D = z_1 = \frac{(1-\beta-\alpha\beta)}{(\alpha-1+\beta)}$, $E = e^{-T/\tau_d}$, then (17) becomes:

$$G_c(z) = C \frac{z(z - \beta)}{(z + D)} \frac{\frac{b_0(z+D)}{z(z-E)}}{(1 - \frac{b_0(z+D)}{z(z-E)})} \tag{17}$$

This simplifies down to:

$$G_c(z) = C b_0 \frac{z^2 - \beta z}{z^2 - (E + b_0)z - b_0 D} \tag{18}$$

And so, substituting all known values, the final discrete transfer function for the controller is:

$$\boxed{G_c(z) = \frac{M(z)}{E(z)} = \frac{0.297627056627521z^2 - 0.280927664619820z}{z^2 - 0.925871953630202z - 0.074128046369798}} \tag{19}$$

The above equation forms the controller block used in the Simulink diagram (shown in the next section). In order to obtain the equation that was used to find the coefficients used in the experiment, we divide the above through by $z^2$, rearrange for M(z) and take the inverse z-transform to get:

$$m(k) = 0.925871953630202 * m(k-1) + 0.074128046369798 * m(k-2) + 0.297627056627521 * e(k) - 0.280927664619820 * e(k-1) \tag{20}$$

Where;

- $m(k)$ is the current voltage
- $m(k-1)$ is the previously sampled voltage
- $m(k-2)$ is the voltage of 2 samples prior
- $e(k)$ is the currently measured error in input shaft velocity
- $e(k-1)$ is the error in input shaft velocity in the previous sample

In addition to the controller, the gain shown in the Simulink diagram in the next section, $K$, must be found. This is done by combining the internal velocity loop with a gain of $2K$, a discrete integrator block and the gear ratio of $\frac{1}{38.4}$ as seen in the block diagram from the lab document[1]. The internal velocity loop can be found by applying the negative feedback loop:

$$G_{VL}(z) = \frac{G_c(z)G_p(z)}{1 + G_c(z)G_p(z)} \tag{21}$$

Combining this with the other parts of the block diagram gives the following transfer function:

$$G(z) = \frac{2KT}{38.4}\frac{z}{z-1}\frac{G_c(z)G_p(z)}{1+G_c(z)G_p(z)} \tag{22}$$

In order to simplify this calculation, it should be noted that by definition:

$$G_c(z) = \frac{1}{G_p(z)}\frac{F(z)}{1-F(z)} \tag{23}$$

Rearranging the above, we get:

$$G_c(z)G_p(z)[1-F(z)] = F(z) \tag{24}$$

$$\rightarrow G_c(z)G_p(z) = F(z) + F(z)G_c(z)G_p(z) \tag{25}$$

$$\rightarrow G_c(z)G_p(z) = F(z)[1+G_c(z)G_p(z)] \tag{26}$$

$$\therefore F(z) = \frac{G_c(z)G_p(z)}{1+G_c(z)G_p(z)} \tag{27}$$

Applying result (27) to Eq. (22):

$$G(z) = \frac{2KT}{38.4}\frac{zF(z)}{z-1} = \frac{2KT}{38.4}\frac{b_0 z + b_0 z_1}{z^2 - (1+e^{-T/\tau_d})z + e^{-T/\tau_d}} \tag{28}$$

Using this open-loop transfer function without the gain $K$ with MATLAB's *rlocus* function, the following root locus is obtained:

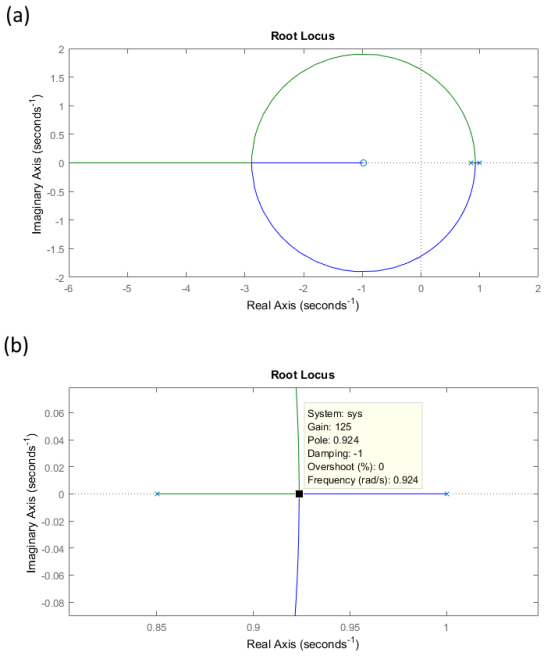


Figure 2: (a) Root locus of Eq. (28); (b) Zoomed in (a) at breakaway point

At the breakaway point, as can be seen from (b) above, we have a gain of $K = 125$ at $z = 0.924$, to find the time constant of this we have:

$$z = e^{sT} = e^{0.006s} \tag{29}$$

$$s = \frac{ln(0.924)}{0.006} = -\frac{1}{\tau} \tag{30}$$

$$\therefore \tau = 75.9079 \text{ ms} \tag{31}$$

Note that the breakaway point is chosen to satisfy one of the design requirements for the controller which is for the response to be critically damped (as seen in Section 2).

# 5 Simulink Diagram

To generate a simulation of the output shaft angular position and input shaft angular velocity, a Simulink diagram for the physical system and discrete controller combination was created. There were two different Simulink diagrams used to generate the results in Section 6. Part A and B of Section 6 used slightly different Simulink diagrams as the experiment was performed on slightly different physical systems. The first Simulink diagram below corresponds to the system represented by the ROT335.m data. The discrete controller $G_c(z)$ takes in the input shaft speed error and outputs the voltage that needs to be applied to the motor. This voltage signal is then zero-order held and passed in to the plant transfer function as described in Eq. (4). The output of this transfer function is then differentiated to obtain the input shaft speed, which is both sent to the workspace for plotting and also fed back into a summing joint. This signal is also integrated and multiplied by the gear ratio of the system to obtain the output shaft angular position which is also both sent to the workspace and fed back into a summing joint.

The "desired position input" repeating stair block describes a sequence of step inputs that correspond to the series of desired shaft positions outlined in Section 3.

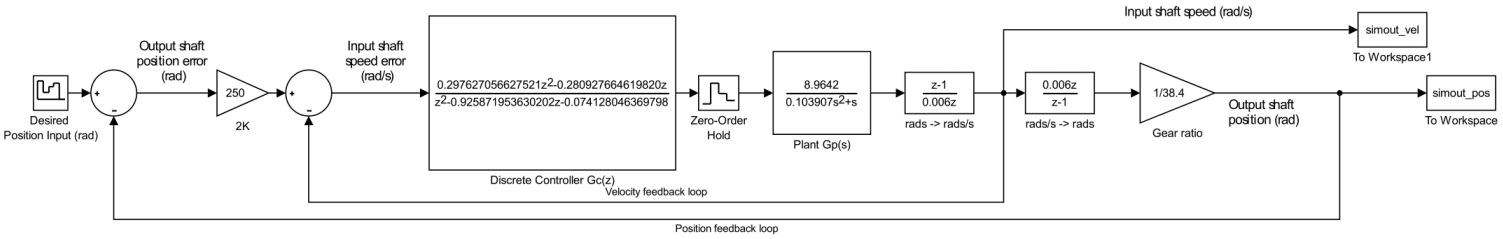


Figure 3: Simulink diagram of system and discrete controller

The changes made to the above Simulink diagram are highlighted by a pink square below. A different plant was used in this Simulink diagram as the physical system it is simulating has slightly different physical parameters to the above Simulink diagram. This transfer function was found the same way the first one was, by curve-fitting data to obtain A and $\tau$. The data that was used for this second transfer function was the test file ROT445.m (from individual data).

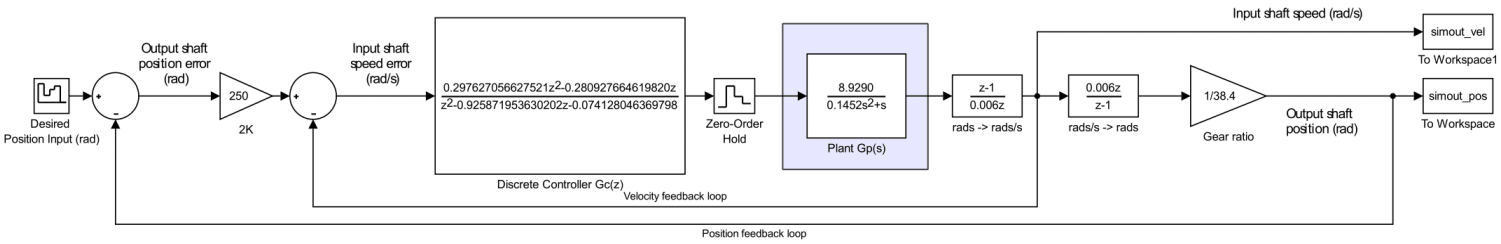


Figure 4: Simulink diagram of system and discrete controller with different $G_P(s)$

# 6 Results

## 6.1 Part A - Design System (ROT335)

For this part of the experiment, the output shaft arm was adjusted in such a way that it's parameters corresponded with those related to the ROT335.m data file. The desired outer shaft position sequence of:

$$3072 \rightarrow 1024 \rightarrow 2048 \rightarrow 2560 \rightarrow 0$$

as stated in Section 2 was input into the system at intervals of $400 * T = 2.4$s between each desired position. The simulated data generated by the Simulink diagram in Fig. 3 was superimposed onto the experimental data given by the A5059915.m data file yields the following:

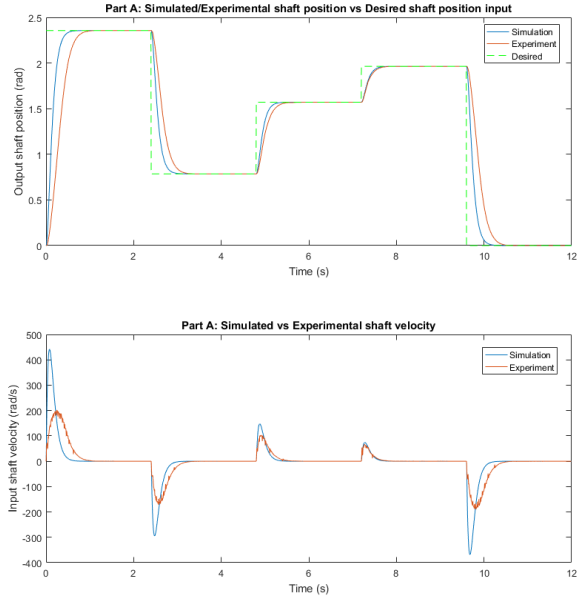


Figure 5: Simulink simulation vs. experimental data of desired position input sequence

It can be seen from the above plots that the simulated value has a zero steady-state error in it's response to a desired position input which was one of the design requirements seen in Section 2. One noticeable thing from the plots above is that there is a discrepancy between the simulated and experimental plots, namely the simulated velocity spikes a lot sooner and with larger amplitude than the experimental data, and the simulated position approaches the desired position sooner than the experimental data.

This can be explained by the fact that there was saturation present in the physical systems that were used in the experiment. This saturation was not modelled in the Simulink diagrams due to its complicated nature of being non-linear. The validity of this explanation is evident from the velocity plot above, it can be seen that on the third rise that the simulation and experimental plots are almost identical for both the position and velocity plots, this is due to the fact that the velocity for the experiment hasn't been affected by saturation as much since it is a small value. It can also be observed that the error between simulated and experimental data gets larger for bigger changes in desired position, this suggests there is a factor of proportionality that is not modelled in the Simulink and this could be related to the saturation mentioned previously.

Calculating the time constant of the smallest rise in velocity of the position graph for both experimental and simulated data yields:

$$\tau_{Simulation} = 158.1\text{ms} \tag{32}$$

$$\tau_{Experimental} = 150.9\text{ms} \tag{33}$$

This is a percentage error of only:

$$PE = \frac{|\tau_{Experimental} - \tau_{Simulation}|}{\tau_{Simulation}} * 100\% = 4.55\% \tag{34}$$

It should be noted that a source of error for the calculation of the time constant comes from the approximation of a third order system (See Eq. (6)) as a first order system. However, the values obtained have a small enough error that the design can be considered accurate when saturation isn't a factor or when ignoring the fact that saturation was not modelled. However this $\tau_{Simulation}$ is approximately double what was expected by Eq. (31), and this must be due to unaccounted factors in the calculation or modelling process (however the lab document calculations and system modelling[2] were followed in this report precisely), but the simulation and experimental time constants (not including saturation) are similar enough that this controller can be considered to be correct.

## 6.2 Part B - Test System (ROT445)

For the second part of the experiment, as outlined in Section 3, the output shaft arm was readjusted to have physical parameters that related to the ROT445.m file (from individual data). This change of system is shown in Fig. 4, where a different plant, $G_P(s)$ is used in the simulation to represent the different system that the experiment is performed on. The same desired outer shaft arm position sequence as in Part A was applied to this new system and the following plots were obtained:

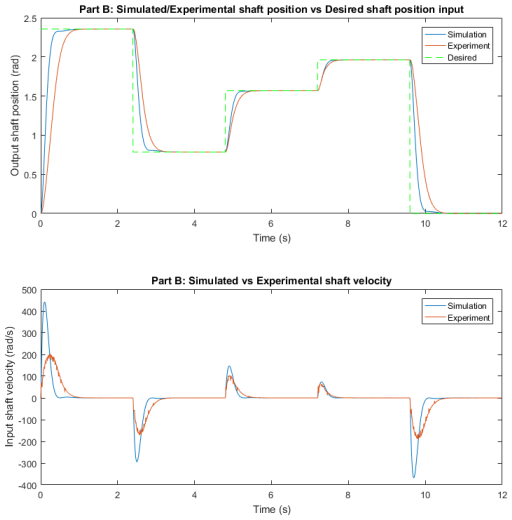


Figure 6: Simulink simulation vs. experimental data of desired position input sequence with different $G_P(s)$

Overall the differences between the simulted and experimental plots of Fig. 6 are very similar to the differences spoken about in Part A. However, zooming into the above plots:

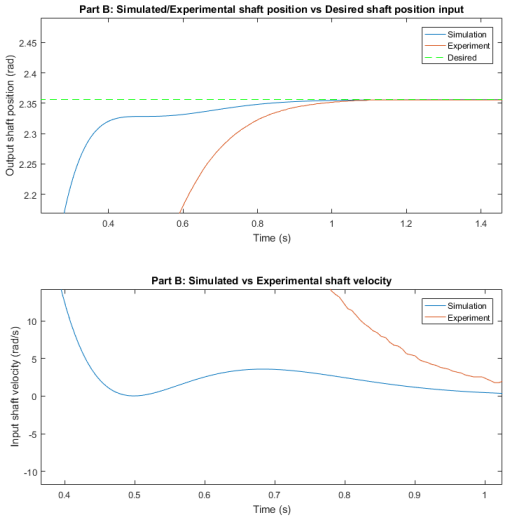


Figure 7: Simulink simulation vs. experimental data of desired position input sequence with different $G_P(s)$

It can be seen from the above figure that there are some slight oscillations occurring in the simulated response of Part B that don't occur in the simulated response of Part A, this is to be expected for two reasons. Firstly, the system that the controller was designed for is different to this system that the controller was tested on and so it's response isn't going to be optimal and the gain $K$ of the model was

chosen such that it was critically damped for the ROT335 system. So it will be under or overdamped for other systems, in this case it is underdamped as evidenced by the oscillations in the response. Secondly, as mentioned previously, there is saturation present in the physical system that was omitted in the Simulink model. This omission of the saturation of the system in the model means the behaviour of the simulated system will have some explainable discrepancies to the experimental behaviour.

To conclusively determine the difference between Part A and Part B, the difference in position is plotted for both the simulation and experimental data:

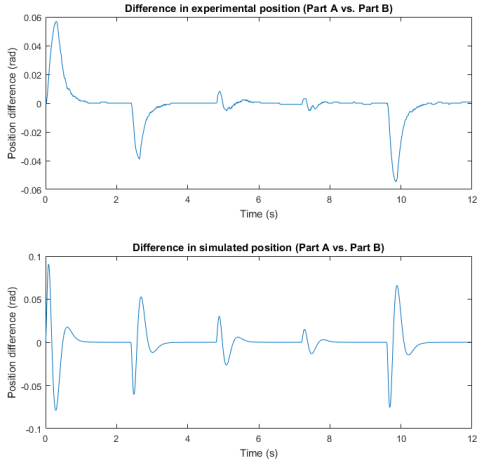


Figure 8: Difference in position for experimental and simulated data

As explained before, the difference is larger for greater changes in the desired position input due to saturation. The oscillation in the simulated figure is also due to all the factors explained above in this section. The maximum absolute difference for the experimental data was determined to be $d = 0.0568$ rad and for the simulated data, $d = 0.0906$ rad. These are relatively small values so the difference in data for part A and B is minimal. The response also still exhibits zero steady-state error, considering these facts the controller can be considered to work effectively for this different system with some note-able, albeit non-problematic differences.

# 7 Conclusion

Overall, the experiment was seemingly a success when ignoring the fact that saturation wasn't modelled for the simulation data. It was shown that when saturation was a near non-factor, the simulation and experimental data were almost perfectly in-line with only a 4.55% difference in time constants. The controller response did exhibit a $\tau$ that was approximately double what was desired but the calculation process and system modelling diagram from the lab document were followed correctly so this could be ignored. The controller designed exhibited critically damped behaviour with a zero steady-state response for the ROT335 system, as was desired in the design specifications. It also performed reasonably well on the ROT445 system, a system it wasn't designed specifically for, having only minimal oscillatory behaviour.

# References

[1] Katupitiya, J. (2017). Lab III : Position Controller Design Guide [pdf] p.2.
Available at:
https://moodle.telt.unsw.edu.au/pluginfile.php/2728245/mod_folder/content/0/PositionControllerDesignGuide2014.pdf
[Accessed Oct. 2017].

[2] Katupitiya, J. (2017). Lab III : Position Controller Design Guide [pdf] p.1-3.
Available at:
https://moodle.telt.unsw.edu.au/pluginfile.php/2728245/mod_folder/content/0/PositionControllerDesignGuide2014.pdf
[Accessed Oct. 2017].