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.

Cameron Murray
Z3417671
12th October 2012

1. INTRODUCTION

The experiment involves the design of a gearbox and motor driving controller system abstracted into both a position and speed controller. Students were required to derive their personalised speed and position controllers from given time constant/sample design parameters before attending a laboratory session. The governing coefficients from the determined transfer function (TF) could then be utilized to manipulate the extension length of axially mounted armature that functions perpendicularly to the motors rotational axes. The inertial force magnitude acting on the revolute hinge can then be exploited by altering the armatures extension.

The parameters of the controller were inserted into computer software to calibrate and direct an electric motor system to produce a data set relating the position and velocity of the armature against time and voltage inputs. The purpose being to predict real life controller performance against a theoretical simulation, as the logged data sets were distributed for simulation in Simulink to help analyse any theoretical consistencies/data trends.

2. AIM

The experiments aim was to predict real life controller performance against theoretical simulations by way of a implementing a speed and position controller to log and analyze a series of two discrete data sets relating armature position to time (T) for a variable voltage.
Following the rig demonstration and data logging process, utilization of MATLABs “Simulink” can be used to formulate a consistent block diagram describing the system to simulate the rigs environment and performance. This block diagram makes use of input parameters/data to derive graphs of a chosen loading pattern (namely the unique loading patterns assigned to each student) to validate experimental/theoretical consistencies. Following this method again with a unique, assigned erroneous transfer function for the system is then performed to identify and investigate the discrepancies between the two runs against their respective simulations.

3. THE EXPERIMENTAL PROCEDURE

Following the derivation of a unique speed & position controller transfer function from given time constant/sample parameters, the coefficients acquired from forming the corresponding final difference equation can be input into the control computers UI to calibrate and drive the motor/armature system. Namely, the armature was required to rotate through five predetermined displacements at 2 second intervals.

The procedure can be abstracted into a series of two sections, the first of which (A) is performed to ensure the validity of the design procedure. Namely, under inspection, if a 1_{st} order response (FOR) produces a time constant aligning with the given unique time constant design-parameter and a zero steady state error (SSE) is produced by the actual response, then the controller design can be considered valid. This section of the experiment was performed using the fixed armature length described by our design parameters/coefficients, namely 225mm.

(B) involves substituting the working TF from the simulink diagram with the TF of an unique and intentionally incorrect system (this time at an armature length fixed at 335mm) assigned in the design briefing on moodle to produce a position output log. Performing an experimental run with another series of predetermined intervals with a different armature length, (equivalent to using incorrect design parameters) resulted in some obvious response deviations that will be discussed in later sections.

4. CONTROLLER DESIGN CALCULATION

The unique design tau time is 38ms. The sample time is 4ms.

Start by attaining a FOA to the zero load/open loop results:
By utilizing the `lsqcurvefit` function MATLAB offers, we can fit a curve to the zero load data to find the values of τ and *A*.:

$$y(t) = A\left(1 - e^{-\frac{t}{\tau}}\right), \tag{1}$$

```
x = lsqcurvefit(@myfun,[281100,1],time,speed)

A = 281100;

tau = 0.08188;
```

By substituting in these values and introducing an integrator, the voltage/counts transfer function is determined as

$$A = 281100$$

$$G'p(s) = \frac{281100/24}{s(1 + 0.08188s)}$$

$$Gp(s) = \frac{8.98337}{s(1 + 0.08188s)} \tag{2}$$

By creating appropriate variables and utilizing the following MATLAB functions finds a discrete version of the continuous TF, which allows for *Gp(z)*. to be formed:

```
num = [8.9802];

den = [tau 1 0];

    = [0.08188 1 0];

sample = 0.004

[numd, dend] = c2dm(num,den,sampleT,'zoh');
```

The roots of numd produce the numerator of A(z) likewise the roots of dend help form the denominator:

$$A(z) = \frac{0.0008636(z + 0.9838)}{(z - 1)(z - 0.9523)} \tag{3}$$

Gp(z) can then be established by immediate substitution as:

$$Gp(z) = A(z)\frac{(z - 1)}{zT}$$

$$Gp(z) = \frac{0.21575(z + 0.9838)}{(z)(z - 0.9523)} \tag{4}$$

Utilizing the personalized time constant value provided on moodle and requiring a zero SSE, which is unity DC gain, then $F(z) = 1$. As the zero has the potential to result in ringing, the numerator needs to take in its value. Since $s = -1/\tau_d$, the placement is then e^{-T/τ_d}. To form F(z) we use:

$$F(z) = \frac{(1 - e^{-\frac{T}{\tau_d}})}{(1 - z_1)} \frac{(z - z_1)}{(z - e^{-\frac{T}{\tau_d}})}$$

Hence:

$$F(z) = \frac{bo\ (z - z1)}{z(z - e^{T/Td})}$$

And so due to the ringing caused by the zero in the numerator of Gp(z):

$$F(z) = \frac{bo\ (z + 0.9838)}{z(z - 0.9001)}$$

If we substitute F(1) = 1 we can find *bo:*

$$F(1) = 1 = \frac{bo\ (1.9838)}{(0.0999)} \qquad , \qquad bo = \frac{1(0.0999)}{(1.9838)} = 0.050358$$

$$F(z) = \frac{0.050358\ (z + 0.9838)}{z(z - 0.9001)} \tag{5}$$

To calculate Gc(z):

$$Gc(z) = \frac{1}{Gp(z)} \frac{F(z)}{(1 - F(z))}$$

$$Gc(z) = \frac{0.004\ \text{[redacted]}\ (z - 0.9523)\ \text{[redacted]}}{0.21575\ \text{[redacted]}} \frac{(0.05036)(z)\ \text{[redacted]}}{z(z - 0.9001) - \ 0.05036(z + 0.9838)}$$

$$Gc(z) = \frac{0.23342z^2 - 0.2223z}{z^2 - 0.95046z - 0.04954} \tag{6}$$

where the roots of the numerator are 0 and 0.952361.

The roots of the denominator are 1.000 and -0.04954

Forming the final difference equation for the controller using these parameters:

$$m(k) = \ 0.95045m(k-1) - \ 0.04954m(k-2) + \ 0.23342e(\mathrm{k}) \ - 0.2223\mathrm{e}(\mathrm{k}-1) \tag{7}$$

The controller will then be generated after merging the complete velocity loop with the 2 K gain and the integrator, together with the gear ratio of 1/38.4.

$$G(z) = 2K.\frac{1}{38.4}.\frac{(z)(0.004)(z+0.9838)}{(z-1)(z-0.9001)}$$

$$G(z) = \frac{(0.00001049166)(z+0.9838)}{(z-1)(z-0.9001)} \tag{8}$$

Inputting this into MATLAB:

```
Gz = tf(0.00001049166*[1 0.9838], [1 -1.9001 0.9001])

rlocus(Gz)
```

Produces the subsequent figure (fig. 1):

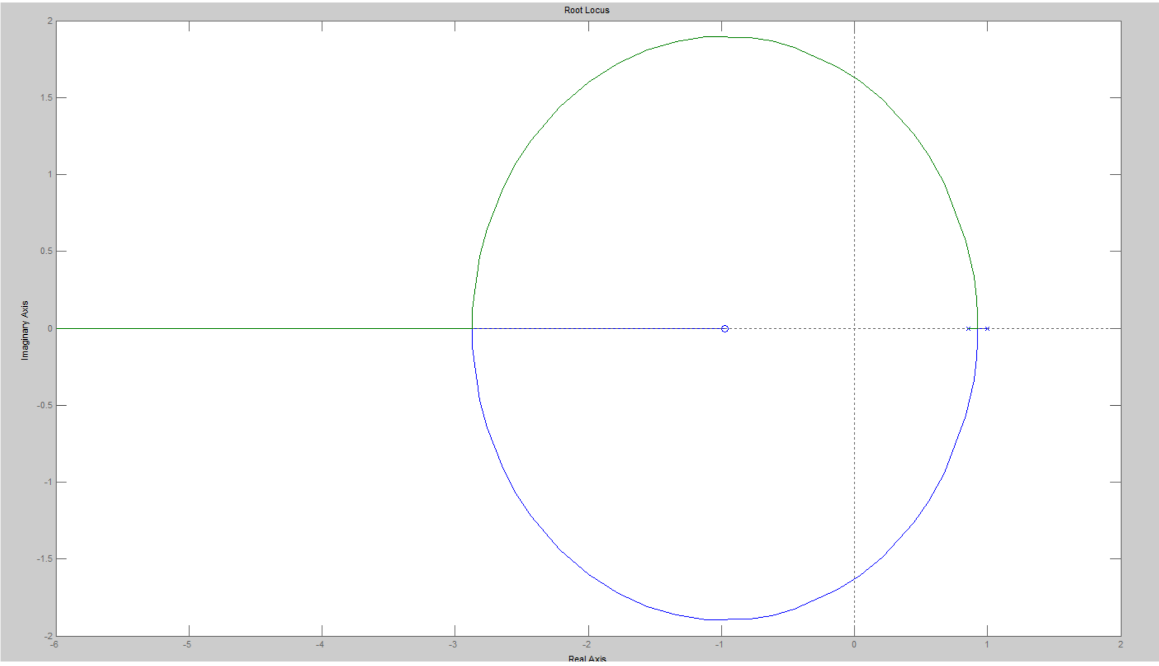


Figure 1: Root Locus Graph

Inquiring at the break-away point allows us to find:

$$K = 123.73; \ \ z = 0.9483$$

5. SIMULINK BLOCK DIAGRAM

With the intention of generating a model to relate to the real life data, Simulink can be utilized to produce a block diagram of the motor/generator system. Taking the values from the the appendix in the experiment guidelines, the proceeding BD as seen in Figure 1 was prepared:

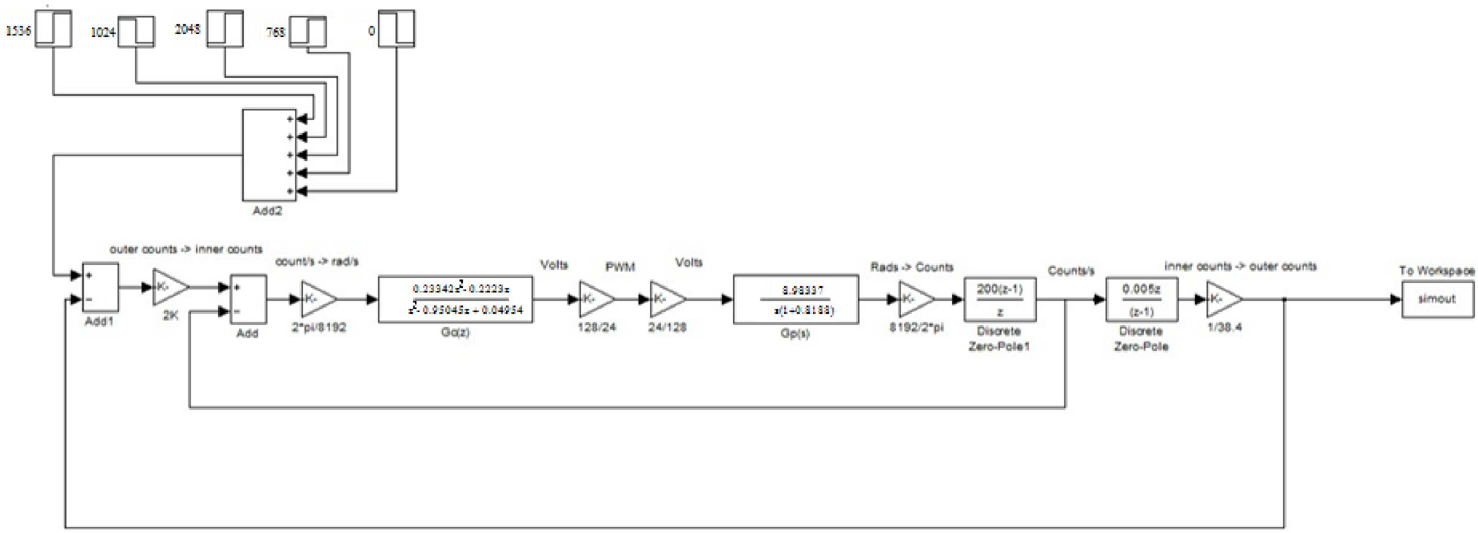


Figure 2: SIMULINK BD for the system.

6. PART A

After producing the block diagram in Simulink, appending a step function block will correctly simulate the steps activated during the experiment runs. To make use of the block diagram, the simulated data produced was superimposed on the respective experimental data in the hopes of verifying the design. The step function bank consists of five independently activated blocks with different amplitudes (derived from the unique assigned parameters) that were activated sequentially at 1600ms intervals.

Since the produced data will obviously not have the same number of vector elements, MATLAB's

`interp1` function is utilized to interpolate the simulated data's array length to be compatible with the time vector of the experimental data. After finalizing the array, the comparison can be made by simultaneously plotting the two data sets as shown below in figure 3.

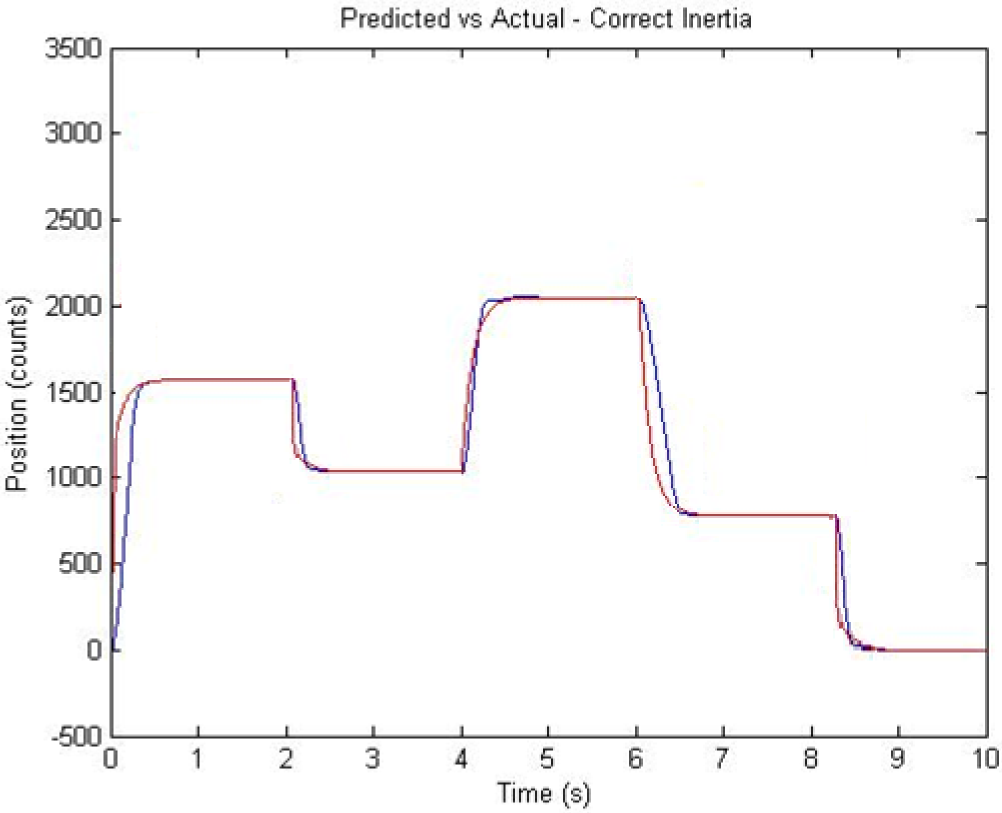


Figure 3: Overlaying of Exp. and Sim. theta vs time lines. Exp. in blue, Sim. in red.

The plot verifies the designs validity, as the experimental (blue line) is matching the theoretical (red line) quite well. There is some deviation and discrepancy in the rise time and in reaching steady state, arising from two sources, namely control effort saturation in that the power source driving the system cannot deliver infinite voltage to control the motors acceleration, and hence the acceleration is capped at a realistic limit. This is seen in the comparative figure (3) as the theoretical acceleration exceeds this limit hence accounting for the rise time inconsistency.

Any steady state delay seen in the graph can be associated with a stacked error of steady state positioning as the run carries on.

It is apparent from inspection that when the control effect is not subject to saturation, the experimental vs. theoretical data is consistently accurate further verifying the correct design.

It is also notable that the simulation value for tau is approximately *double* the desired tau derived from the root locus.

7. PART B

The use `lsqcurvefit` once more to find figures for A and τ for the arm length of 335mm. allows for the simulation of "incorrect" TF coefficients. By inputting our assigned data from ROT335, we acquire $\tau = 0.1039$ and A = 280500. By utilizing the form seen from the initial design segment:

$$G_{p1}(s) = \frac{A}{1 + \tau s}$$

An utilizing MATLAB `lsqcurvefit` function to fit the curve:

```
x = lsqcurvefit(@myfun,[280000,1],t335,s335)
```

The following Tau Values can be derived

```
A = 280500;

tau = 0.1039;
```

Hence the calculations are as follows:

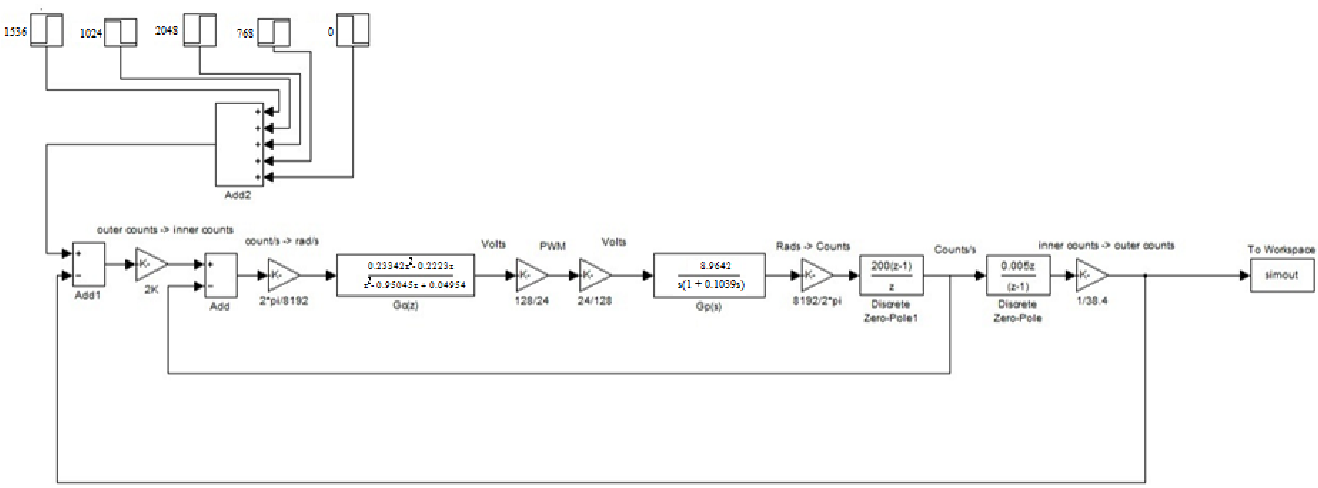
$$A = 280500$$

$$G'p(s) = \frac{\frac{280500}{24}}{s(1 + 0.1039s)} \cdot \frac{2\pi}{8192}$$

Linking the input Volt./input shaft speed with units rad/s gives,

$$Gp(s) = \frac{8.9642}{s(1 + 0.1039s)} \tag{9}$$

Using this plant we arrive at the following block diagram:



shown in Figure 4. The only difference between fig. 4 and fig. 2 is the change of the plant Gp(s).

Figure 4: Incorrect Gp(s) intentionally inserted into block diagram

We can now rerun the simulation with the altered TF to obtain a data set required for analysis by superimposing onto data distributed after the demonstration:
The subsequent figure (fig. 5) shows the result of the Simulink block diagram with an altered plant from Fig. 4 laid on top of the experimental data. What can be seen is the relationship between both data sets from the position controller model.

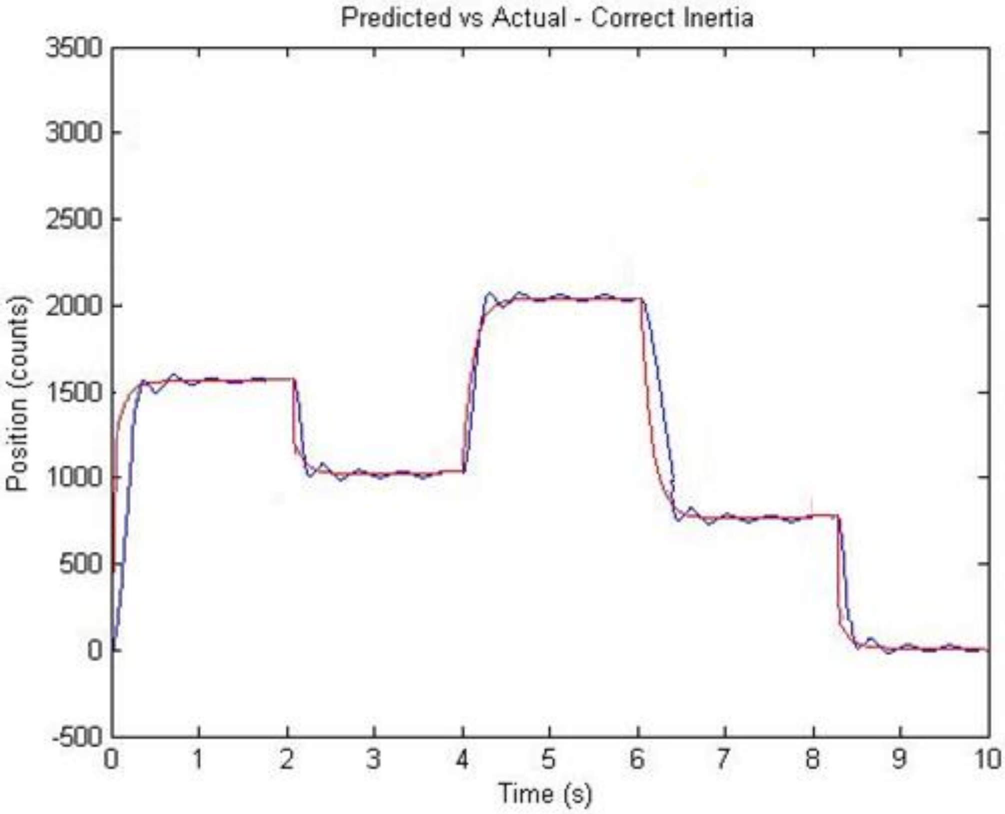


Figure 5: Overlaying Exp. Vs. Sim. theta vs time plots. Exp in blue and Sim. In red.

.

The figure, whilst maintaining the same consistent trends seen in section (A), can be observed to have some accuracy limitations, as such is not entirely correct. The Exp. results roughly agree with the theoretical results, as some definite and evident overshoot with oscillation (which leads to a slow steady state approach) can be seen.

It is also noteworthy that the higher inertia in the run resulting from the longer armature length also amplifies the overshoot. This is not a present factor in the correct TF design. The actual error itself results from the inertia differential in the two rig setups, i.e. the time/energy magnitudes required to halt the lengthened armature on its correct target were not met, as the rig's build parameters were insufficient. As the experiment ran on, it can be seen that this offshoot decreased at a consistent rate as the control effort's had insufficient power to notably overshoot the desired displacements.

Consistent with what was seen in fig.3, there is a gradient differential between the Exp. vs. Actual rise time degrees which can be accounted by the same control effort saturation. By analysing MATLAB's SIMULINK block diagram, the design uncertainties should be relatively well handled by the controller, unfortunately as a result of this motor impulse saturation, the armatures inertial values are compromised and ill-handled in realtime.

8. CONLUSION

Comparing both Fig. 3 and Fig.5, it can be seen that the actual and theoretical data trends do not precisely agree, instead, at points of sudden displacement errors arise with issues in speed controller aspect. As discoursed at length, the accountable cause is control effort saturation, resulting from the maximal voltage to drive motor acceleration not being accounted for in the SIMULINK design.

The standout inconsistency seen in Fig. 5 is the obvious damped oscillations about the desired position that leads to a slow steady state, this is not taken into account in the SIMULINK model. As noted, this can be accounted for by the change in inertia due to the difference in armature lengths (between the two runs/different plant TF). This results in the control effort to be noticeably inadequate at stopping the armature at a desired position. However, whilst observable, one can conclude from this that the steady state error should be quite small.

By inspecting the plots, we can append trendlines to approximate the final steady state rate, as seen in figure 6. This, as can be seen, is approached very quickly, even when using the incorrect TF

Figure 6: Overlaying Exp. Vs. Sim. theta vs time plots., with relative trendlines. Green showing potential Exp. Settling form, Purple showing Sim. data desired steady state position.

Other errors that may have contributed to the difference between expected and actual results includes variability in plant parameters. Armature inertia, motor inertia, armature length and air resistances are areas of potential error and may have been different during the test compared to what is predicted. Human error and instability in the rig can affect plant parameters as complete stationary behavior was assumed in the simulation. Any errors of this nature would change the plant transfer function slightly and contribute to a difference between expected and measured results.

Some further errors that could account for some discrepancy and result variability include: Back EMF coefficients may have differed slightly throughout the experiment. This inconsistency can definitely alter the plant TF. Other discrepancy origins include armature resistance and inductance, viscous damping, motor inertia, differing torque constants.

On the contrary, the data derived from the rig does hold a stable curve at points which the desired positioning is encountered. It is possible to see that depending how the experiments are run, they can both be seen as a success as they reach their desired positioning, with some lag time due to oscillations in part (B). Any inconsistencies in rise time are comparatively small even when taking into account the differences between the Actual and Experimental curves, as the armature finds it's mark fast (less than a second for full stability, stabilising contributes to a maximum of 2.6% of traverse time). Still, this could still be a result of variability in the plant parameters including both motor and armature inertias with (almost negligible) air friction effects.

All things considered, looking at the controller in terms of its purpose, it can be deemed correct in terms of its capability of maintaining accurate and relatively fast control.