DEPARTMENT OF MECHANICAL & AEROSPACE ENGINEERING

University of Strathclyde Engineering

Automotive Cruise Control Design

Class: ME528 - Control Systems Design

Author: Niamh O'Neill-Berest

Student Number: 201919421

Word Count: 3222

November 2023

Abstract

The aim of this project was to create a cruise controller for a car. In Part 1, simplified dynamics of the car were used, and in Part 2 a more realistic setup was used. Matlab was used to iterate through possible gain values for the controller and several different disturbances were modelled to test the controller. Finally, a review of the coursework was conducted.

Contents

List of Figures

List of Tables

Introduction

0.1 What is a PID Controller?

PID stands for Proportional, Integral, Derivative. It is a 3 part controller which uses closed-loop feedback to feed the actual output of a system back to the set-point to give the error [1]. The simplified idea is that the controller reads a sensor and then generates an actuator output according to the P, I and D gains that were set. These gains need to be tuned to ensure the system remains stable, even when disturbances are encountered. Noise from the chosen sensor also needs to be considered, as if this is not accounted for stability can be affected. [2]

0.2 Cruise Control

Modern cruise control was invented by Ralph Teetor in the late 1940s, and was first available as a luxury option in Chrysler cars [3]. The purpose of cruise control is to maintain the car at a certain speed, and it works by adjusting the throttle of the car according to a sensor reading.

1 Part 1

1.1 Method

1.1.1 Car Dynamics

Before creating the system block diagram in Simulink, the transfer function of the system was determined. The steps for this can be seen below, starting with Newton's Second Law.

$$F = m\ddot{x} \tag{1.1}$$

Then, the damping of the vehicle is introduced through the $c\dot{x}$ term, as well as an incline disturbance with the $mg\sin\theta$ term.

$$F = m\ddot{x} + c\dot{x} - mg\sin\theta \tag{1.2}$$

The equation is then divided through by mass, with a_n used to represent F/m and $u(t)$ used to represent the control torque.

$$\ddot{x} + a_n\dot{x} = u(t) - g\sin\theta \tag{1.3}$$

To get the transfer function, the disturbing force is subtracted from the control torque to get the driving force for the car dynamics.

$$\frac{V(s)}{E(s)} = \frac{1}{s + a_n} \tag{1.4}$$

For a typical 4 door saloon car driving in top gear, a_n is typically around 0.02, so this value was used moving forward.

1.1.2 Block Diagram

Once the dynamics of the car and the system transfer function had been found, a block diagram was created, as shown in Figure 1.1.

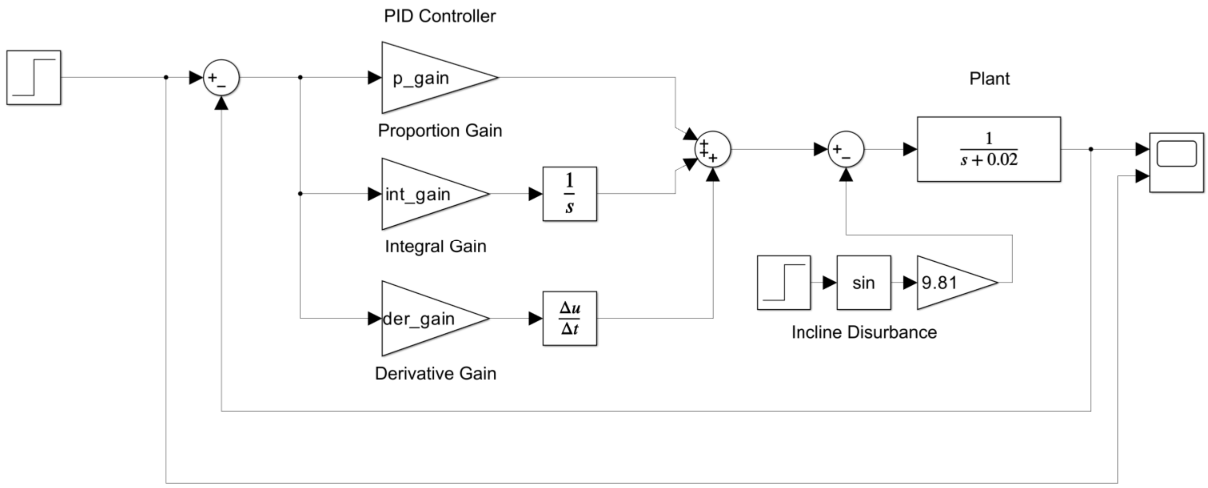

Figure 1.1: Simulink block diagram for Part 1.

The block diagram started with a step input for the set-point velocity. Then the PID controller was built, and the gain values were set using a Matlab script. The incline disturbance was subtracted at this point, with the angle of the incline also being set using a Matlab script. Finally, the plant was placed and a scope was used to plot the actual velocity of the system alongside the desired velocity of the system. Later in the project, a slight variation was used for both the set point velocity and incline disturbance, using a saturated ramp function as can be seen in Figure 1.2. This allowed for a more realistic disturbance scenario.

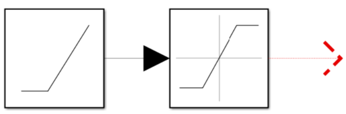

Figure 1.2: Example of a saturated ramp input, used in Part 1.

1.1.3 PID Gain Selection

Once the block diagram was in place, work could begin on determining suitable proportional, integral and derivative gain values for the controller. The first step for this was to investigate the 7 different types of PID controller - P, I, D, PI, ID, PD and PID. At this stage, the incline disturbance was removed to simplify the problem. As the gain values were defined using a Matlab script, this script was adapted to run the simulation for every possible combination of a certain set of values - in the first iteration, this was 0, 0.001, 0.01, 0.1 and 1. This exercise gave an idea of which magnitudes worked best for each gain, and also showed which gains had a larger impact on the overall performance of the controller.

Once the optimal magnitudes for each gain were determined, a more fine-tuned study was conducted in a similar manner, where each gain was increased and decreased slightly until the best values were found. This part of the study was repeated several times, using different input and disturbance cases. A description of each case can be found in Table 1.1.

Test Case ID	**Velocity Set Point**	**Disturbance**
1	Step to 60mph	No Incline
2	Step to 60mph	Step to 15° Incline
3	Ramp to 60mph over 6 seconds	No Disturbance
4	Ramp to 60mph over 6 seconds	Step to 15° Incline
5	Ramp to 60mph over 6 seconds	Ramp to 15° incline over 60 seconds

Table 1.1: Table showing the 5 initial test cases used.

For all of these test cases, the first performance indicator used to assess the controller was the peak response value of the system. A minimum and maximum allowable velocity was selected, and any controller which fell out of this range was not taken further. After this a more subjective approach was taken in the assessment of the controllers, looking at the rise time and the settling time. The maximum velocity was first calculated as shown in Equation 1.5 as this is the how the legal limit is calculated [4]. However this returned too many values, and a more stringent condition of no more than 6 percent overshoot or 4 percent undershoot allowable was then used.

$$V_{max} = 1.01V_{speedlimit} + 2 \tag{1.5}$$

1.1.4 Stability

The stability of the system was assessed using 3 methods - Routh Hurwitz, Bode, and Root Locus. These were all evaluated using a Matlab script, which calculated the Routh Hurwitz criterion and used Matlab's built in $pid()$ function to assist in creating the system transfer function, and the $bode()$ and $rlocus$ functions to create the Bode and Root Locus plots.

1.1.5 Additional Disturbance Modelling

To ensure that the final controller was robust and could tackle a varied set of disturbances, two more disturbance cases were used to test the system. The first of these was a very short amplitude, high frequency $sine$ wave, to simulate a long and very bumpy stretch of road. The second of these was a high amplitude, lower frequency $sine$ wave, used to simulate driving through long stretches of up-and-down roads that would be experienced when driving over a series of hills.

1.2 Results

1.2.1 PID Values for Step Input

Using a step input set point velocity of 60mph, and no incline disturbance, the only controllers which could meet the peak velocity requirements can be found in Table 1.2.

K_P	K_I	K_D	V_{peak}
1	0	0	58.824
1	0	0.001	58.824
1	0	0.01	58.824
1	0	0.1	58.824
1	0	1	58.831
1	0.001	0	58.942
1	0.001	0.001	58.942
1	0.001	0.01	58.943
1	0.01	0	59.662
1	0.01	0.001	59.663
1	0.01	0.01	59.666
1	0.01	0.1	59.699
1	0.01	1	59.964
1	0.1	0	63.184
1	0.1	0.001	63.185
1	0.1	0.01	63.213
1	0.1	0.1	63.559

Table 1.2: Initial ”good” values for the simple PID controller.

Some of these controllers overshot, and some undershot. The scope output of an undershot controller can be seen in Figure 1.3.

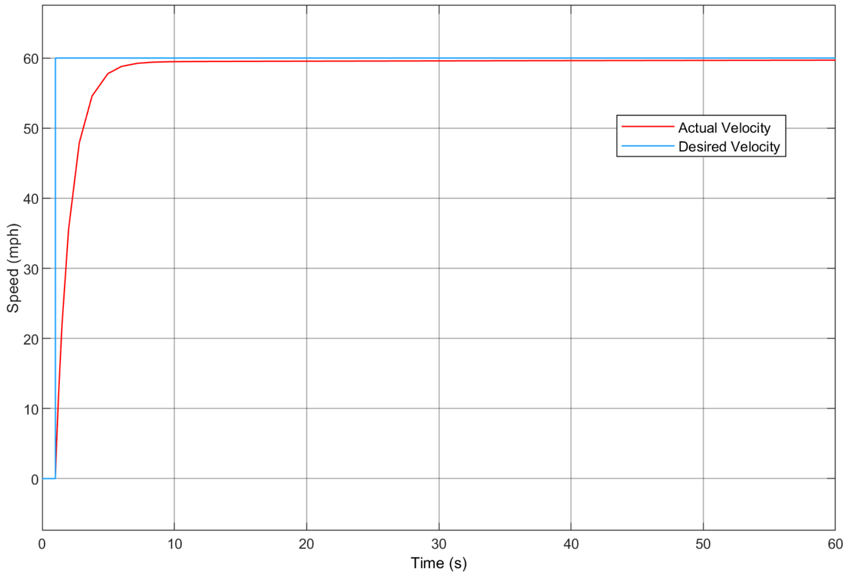

Figure 1.3: Example of an undershot response, using the gains P=1, I=0.01, D=0.1 .

The scope output of an overshot controller can be seen in Figure 1.4.

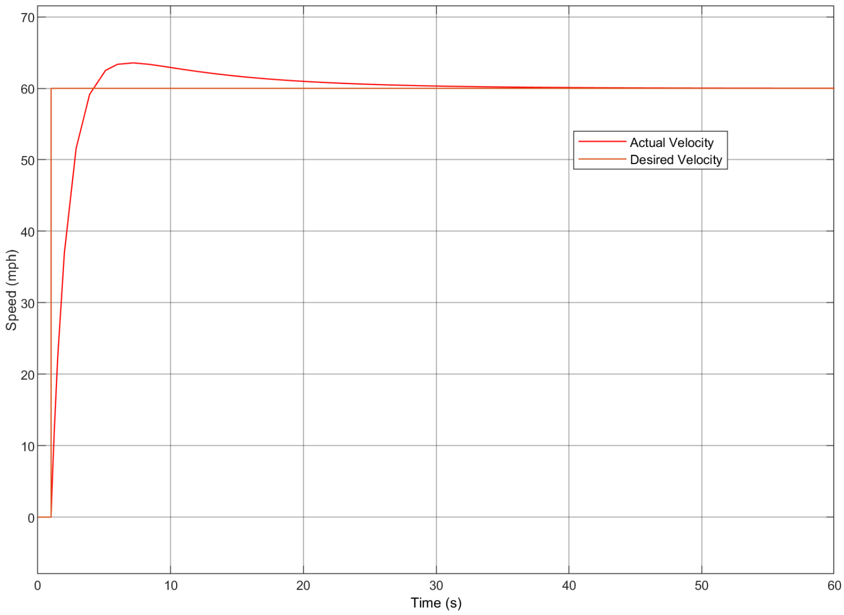

Figure 1.4: Example of an overshot controller, using the gains P=1, I=0.1, D=0.1 .

By observing the difference in responses when adjusting the gains, some general observations were made. For example, when increasing K_P, it was seen that overshoot decreased but settling time increased. When increasing K_I, overshoot increased but settling time decreased, and when increasing K_D the effect was a slight decrease in oscillations. Exaggerated examples of this can be seen in Figures 1.6, 1.7 and 1.8.

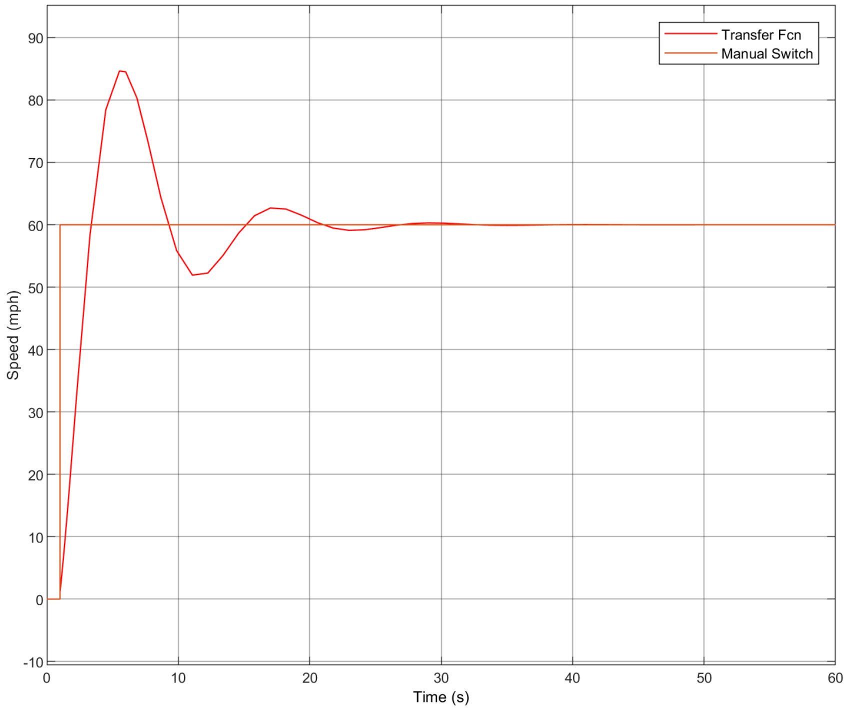

Figure 1.5: Baseline controller used to compare PID gain values. P = 0.5, I = 0.5, D = 0.5

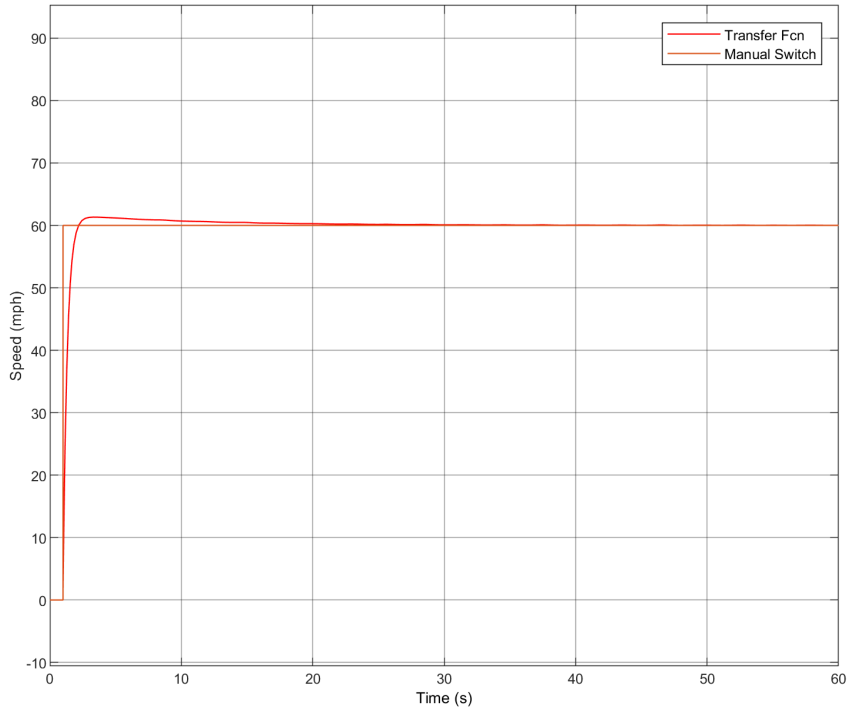

Figure 1.6: Increased proportional gain to 5.

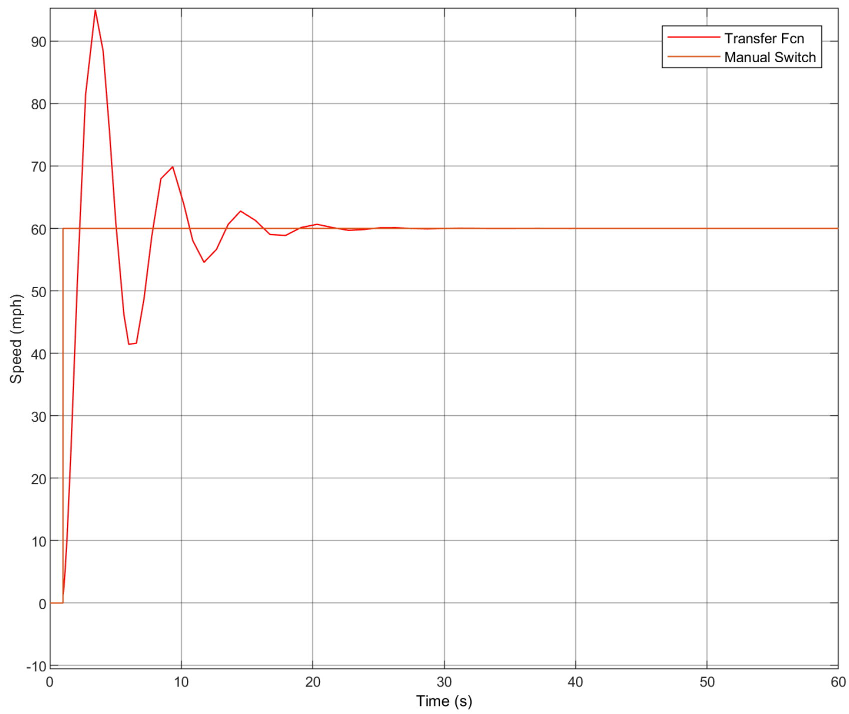

Figure 1.7: Increased integral gain to 2.

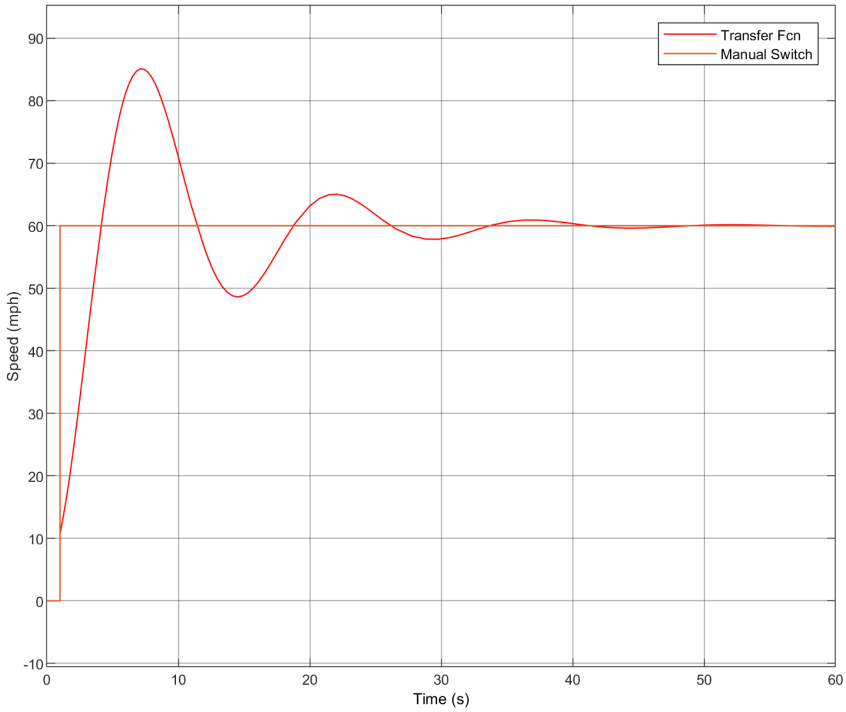

Figure 1.8: Increased derivative gain to 1.5.

Based on these results, more fine tuning of the controllers was conducted. A range of 0.5 - 2.0 was investigated for the proportional gain, 0.1 - 1.0 for the integral gain and 0.01 - 0.5 for the derivative gain. The successful ranges of each gain can be seen in Table 1.3.

K_P	1.2 - 2.0
K_I	0.1 - 0.2
K_D	0.01 - 0.45

Table 1.3: Table showing acceptable ranges for PID gains for a step input.

1.2.2 Disturbance and Ramp Input

The same gain values as before were then tested with the other 4 cases in Table 1.2, to see if there was any variance in performance. The highest and lowest settling time for each case is shown in Table 1.4, along with the peak velocity of the system.

Case	K_P	K_I	K_D	V_{peak}	t_{settle}
1 - best	2	0.15	0.01	61.344	54.953
1 - worst	2	0.1	0.5	61.340	58.426
2 - best	2	0.25	0.01	61.436	54.933
2 - worst	2	0.1	1.35	60.952	84.057
3 - best	1.9	0.2	0.4	61.630	13.179
3 - worst	2	0.1	1.35	60.684	52.680
4 - best	1.2	0.1	0.1	60.922	14.898
4 - worst	2	0.1	1.35	60.011	59.982
5 - best	1.3	0.15	0.3	61.629	15.638
5 - worst	1.9	0.1	1.35	60.016	56.194

Table 1.4: Table showing the performance of various controllers.

The final selected gain values for part 1 were 1.3, 0.15 and 0.3 for proportional, integral and derivative gains respectively.

1.2.3 Additional Disturbances and Final Responses

The final response of the system responding to a 15° ramp incline can be seen in Figure 1.9.

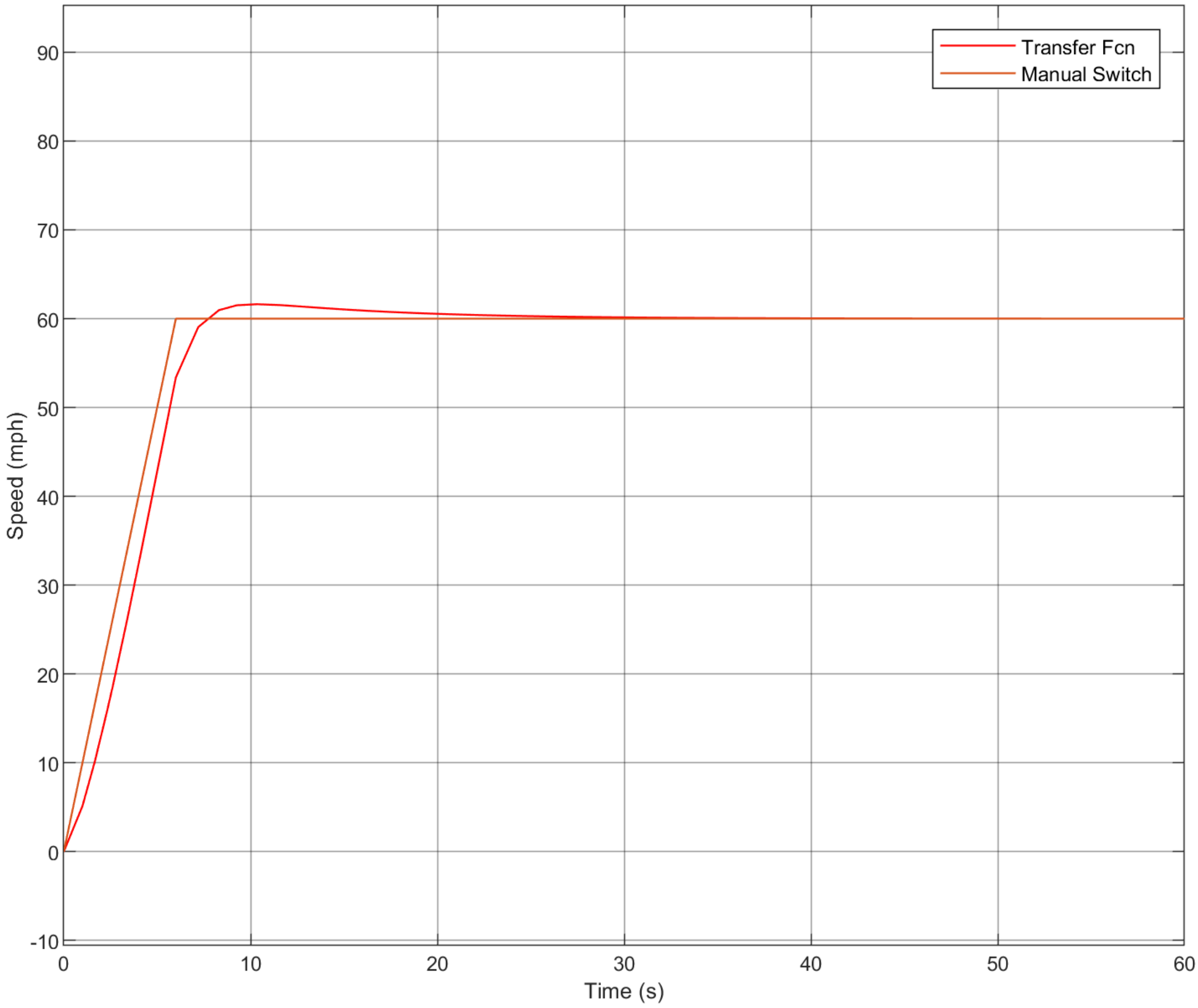

Figure 1.9: System response to 15° ramp incline disturbance,

The response of the system to a high frequency (1 rad/sec), low amplitude (0.1 m), can be seen in Figure 1.10.

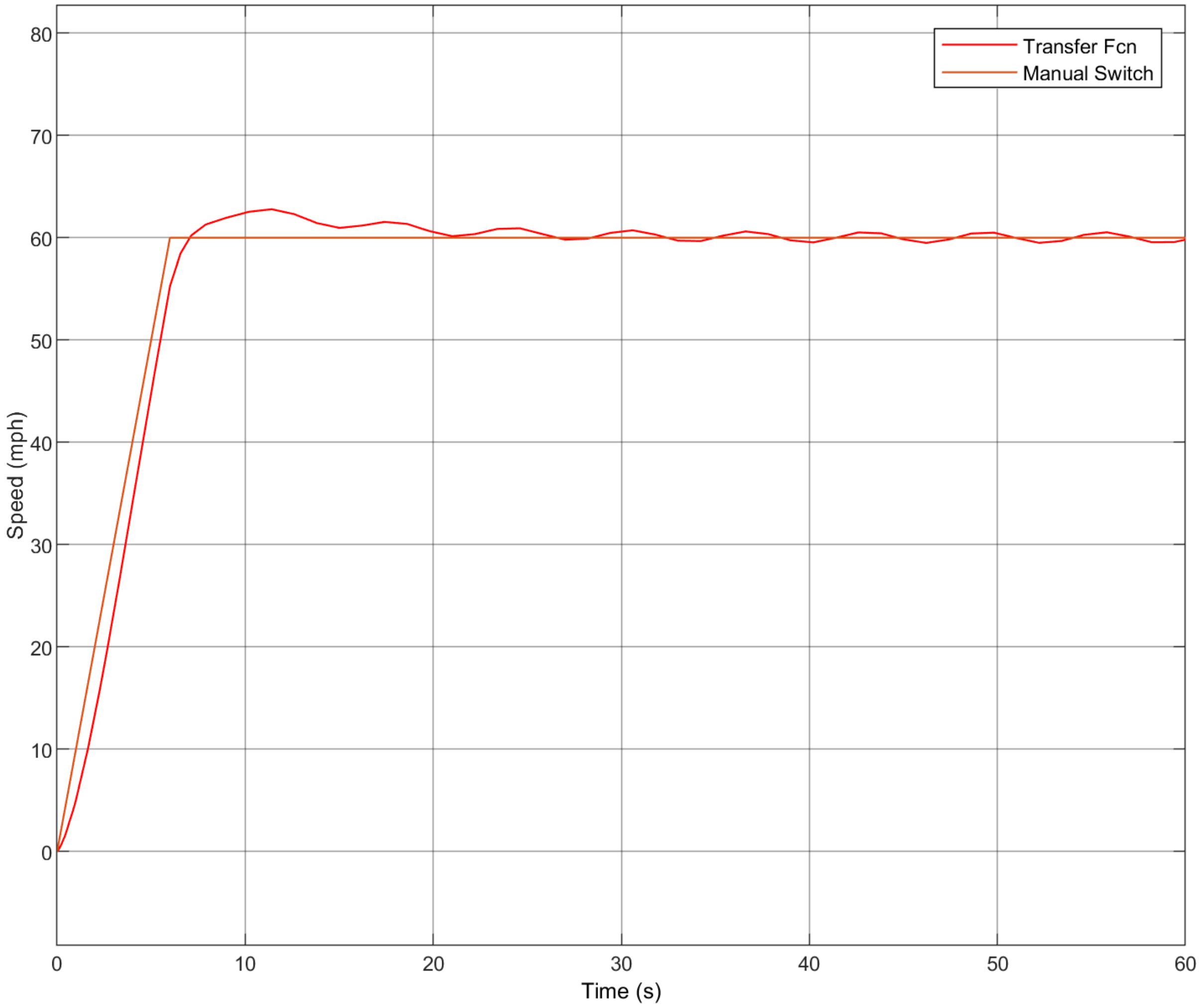

Figure 1.10: System response to the "bumpy road" disturbance scenario.

The response of the system to a low frequency (0.2 rad/s), high amplitude (1 m), can be seen in Figure 1.11.

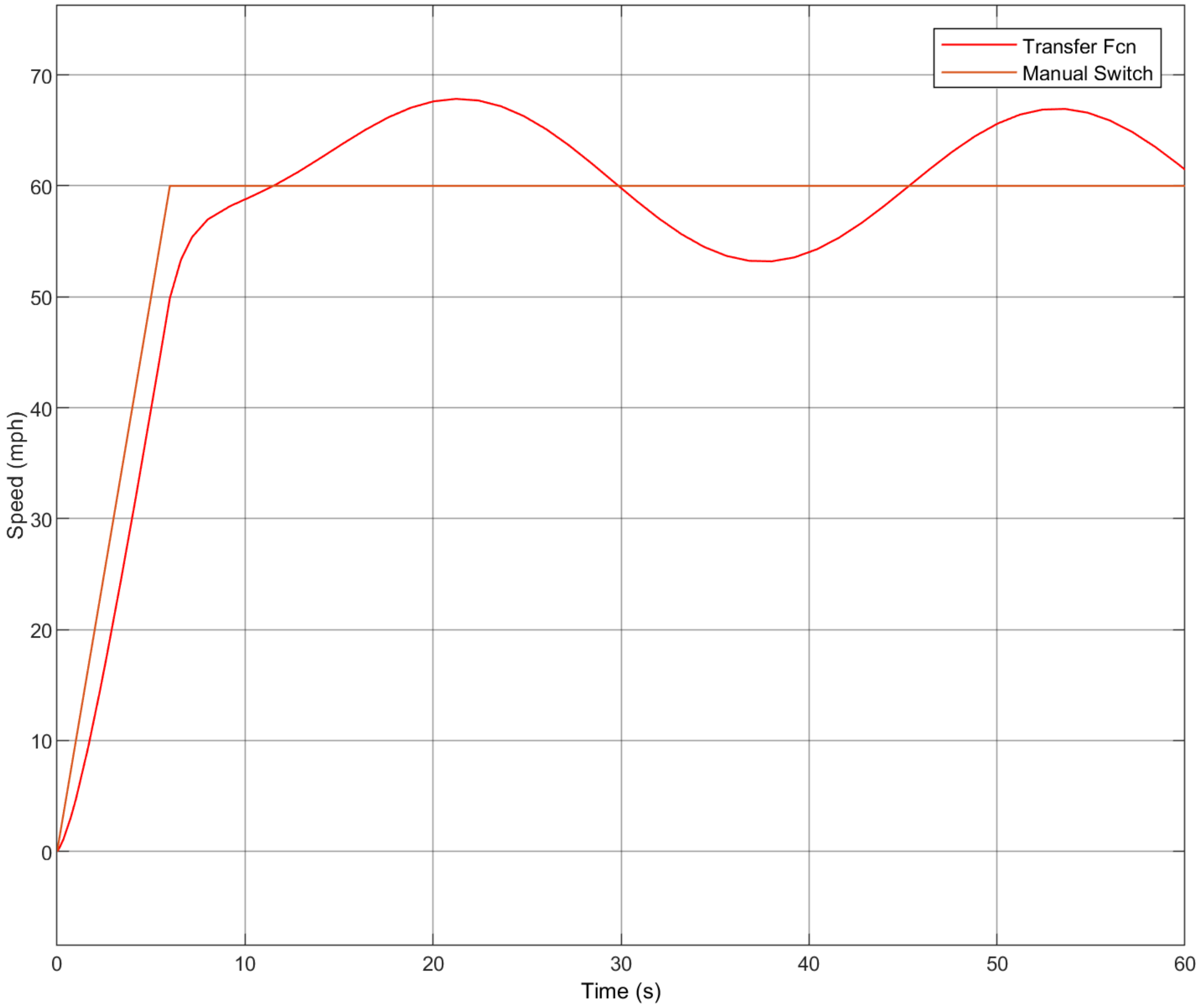

Figure 1.11: System response to the "hilly" disturbance scenario.

At this point, the integral gain was increased to 0.65 to reduce overshoot in the hilly scenario, as shown in Figure 1.12.

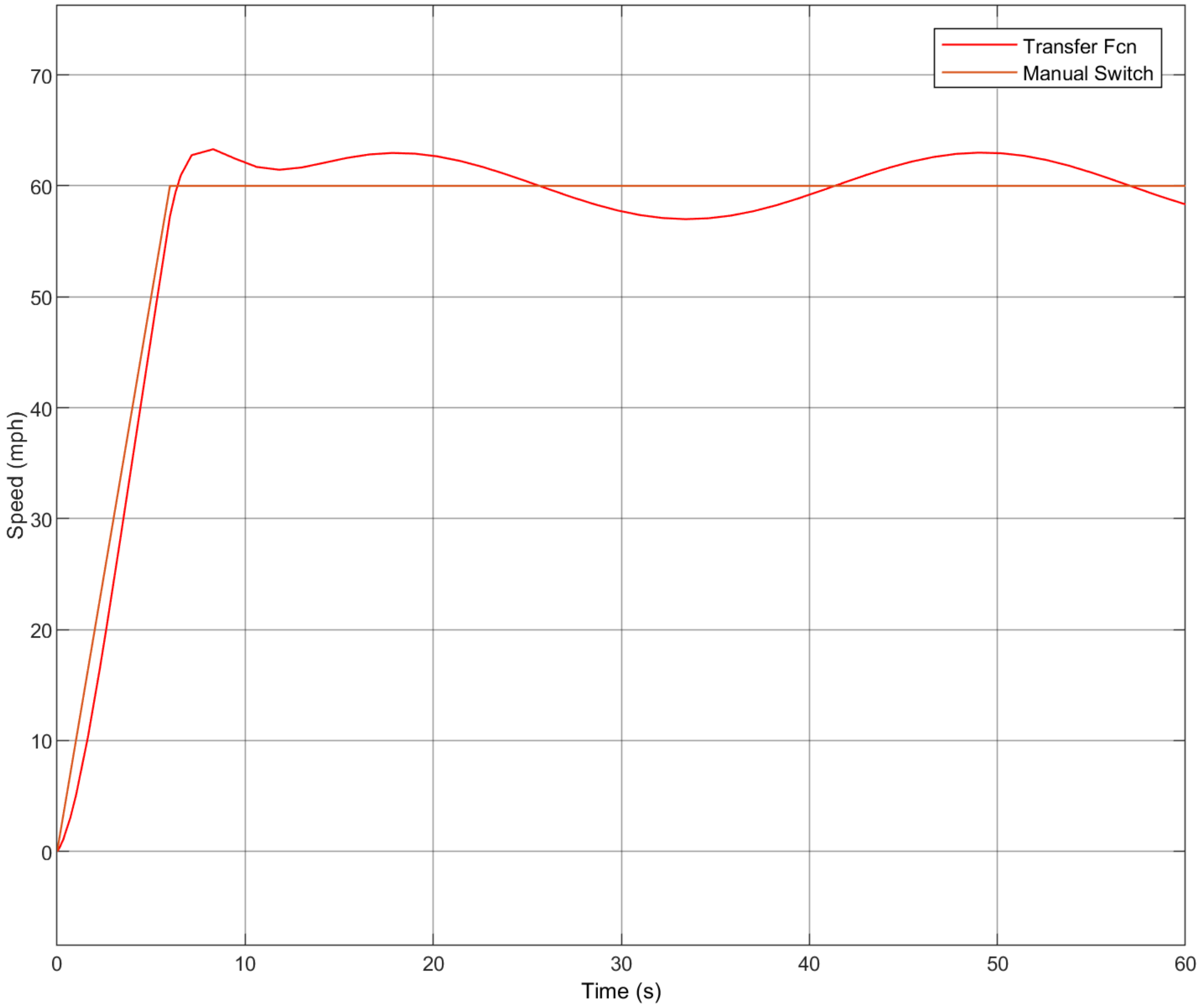

Figure 1.12: System response to "hilly" disturbance scenario with an increased integral gain.

1.2.4 Stability

Using the final selected gain values, three methods were used to evaluate the stability of the system. Due to the nature of generating the transfer function, this aspect of the project was done assuming no disturbing forces were present. The first of method used was Routh-Hurwitz, which was evaluated as follows:

$a_0 = 1.3$	$a_2 = 0.65$
$a_1 = 1.32$	$a_3 = 0$
$b_1 = 0.65$	

Table 1.5: Routh-Hurwitz Table

Where b_1 was calculated according to Equation 1.6:

$$b_1 = \frac{(a_1 a_2) - (a_0 a_3)}{a1} \tag{1.6}$$

Following this, a Root Locus plot was produced as can be seen in Figure 1.13.

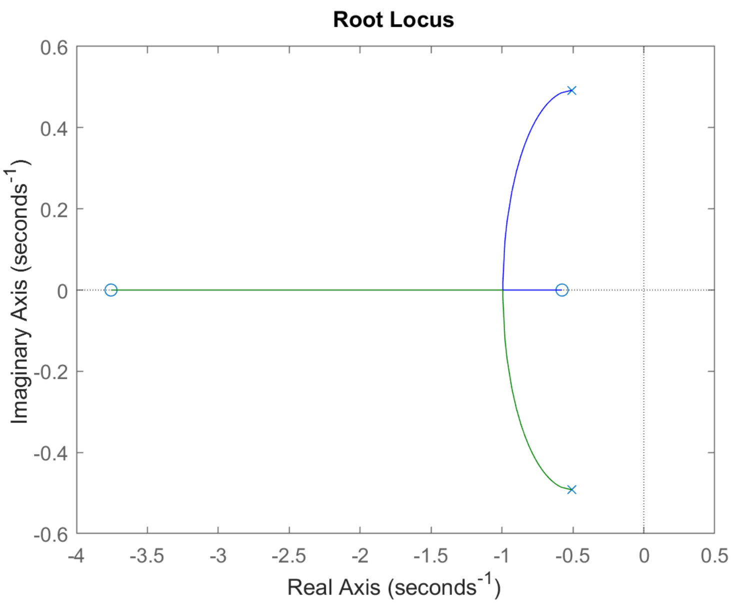

Figure 1.13: Root Locus plot.

Finally, a Bode plot was produced as shown in Figure 1.14.

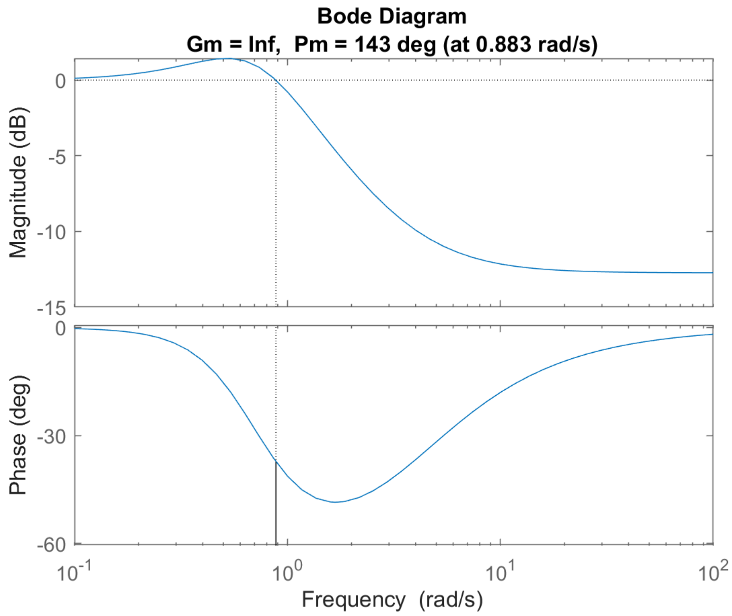

Figure 1.14: Bode plot showing gain and phase margins.

1.3 Discussion

1.3.1 Justification of Final PID Gain Values

The final gains of 1.3, 0.15 and 0.3 for proportional, integral and derivative gain respectively were selected as they produced a system response with a peak response within the allowable bounds and had a low settling time. These gains also responded well to both step and ramp inputs, as well as step and ramp inclines. The ramp input with ramp incline was given more weight in the decision making process as this scenario was the most realistic out of the original 5 testing cases.

1.3.2 Additional Disturbance Cases

Despite the controller performing well in the 5 cases outlined in Table 1.1, it did not perform as well in the low frequency, high amplitude test case. It could be seen that when the car was going down hill, it sped up quite significantly and when it was going uphill it slowed down just as much. Due to this response, the integral gain was increased to prevent overshoot, and this showed a much more satisfactory response.

1.3.3 Stability

As there were no sign changes in the first column of the Routh-Hurwitz array, this control system was deemed to be Routh-Hurwitz stable. The Root Locus plot also confirmed stability, as the real components of the roots are negative. The Root Locus plot also suggests that the system is underdamped as can be seen by the poles having an imaginary component. The oscillations are small however, as can be seen by the small imaginary component. Finally, the Bode plot showed that the system had an infinite gain margin, which means that no matter how much the gain of the controller is increased, the system will always be stable. This demonstrates how simplified the dyanmics of the system are, as this would not usually be true for most realistic systems. The phase margin of 143deg was also very high, pointing to a high degree of stability. As all of these methods show that the system is stable, it was deemed that the P=1.3, I=0.65, D=0.3 controller was suitable.

1.3.4 General Observations

It was observed that when using step inputs, having a high derivative gave the Simulink solver a lot of difficulty. This is likely due to the fact that the derivative part of the controller needs to be able to see future time steps and the step input can be very harsh for this. Much fewer problems were encountered when using a ramp input.

It was also observed that increasing the proportional gain can sometimes require an increased integral gain, as a high proportional gain can cause the controller to ”overreact” to disturbances. It was seen that the key to a good PID controller for this project was to strike the right balance between proportional and integral gains. The derivative gain didn't have much impact but did help with reducing oscillations slightly.

1.4 Conclusions

To conclude, a PID control system was created to control a simplified dynamic model of a car's cruise control. Several iterations and test cases were carried out, with the most valuable being the ramp input and ramp disturbance due to their realism. At this point, the ”best” PID gain values were selected based on settling time and peak response, and these were P = 1.3, I = 0.15, and D = 0.3. At this point more disturbance values were investigated and it was discovered that with high amplitude disturbance oscillations, the controller tended to overshoot. For this reason, the integral gain was increased to 0.65 which rectified this problem.

2 Part 2 - Non-Linear Dynamics

In Part 2, the dynamics of the car were modelled in more detail and the concept of feedback linearisation was introduced.

2.1 Method

2.1.1 More Advanced Dynamics

In Part 1, the damping of the car was covered by a single coefficient. In Part 2, this was covered in a lot more detail. To start with, the car as a whole was modelled as follows:

$$m\ddot{x} = F_T - F_d \tag{2.1}$$

Then, the disturbance force was split into 3 forces - aerodynamic damping force (F_a), rolling resistance force (F_r), and the gravitational force (F_g). These were summed together to give the total disturbance force, (F_d). Each was calculated as follows:

$$F_d = F_g + F_r + F_a \tag{2.2}$$

$$F_g = mg\sin(\theta)) \tag{2.3}$$

$$F_r = mgC_r \tag{2.4}$$

$$F_a = \frac{1}{2}\rho C_d v^2 \tag{2.5}$$

The traction force was also calculated separately using gear ratios and other IC engine characteristics.

2.1.2 Initial Block Diagram

To make the Simulink block diagram easier to work with, each segment of the system was separated into a subsystem block. The individual segments can be seen in the Figures below.

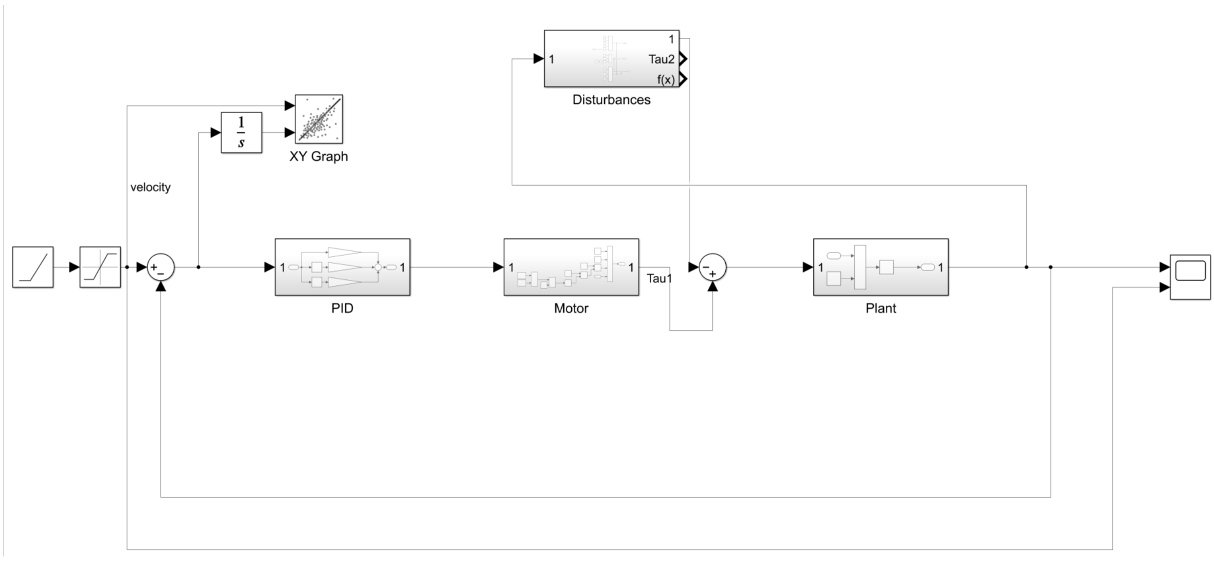

Figure 2.1: Initial block diagram for Part 2.

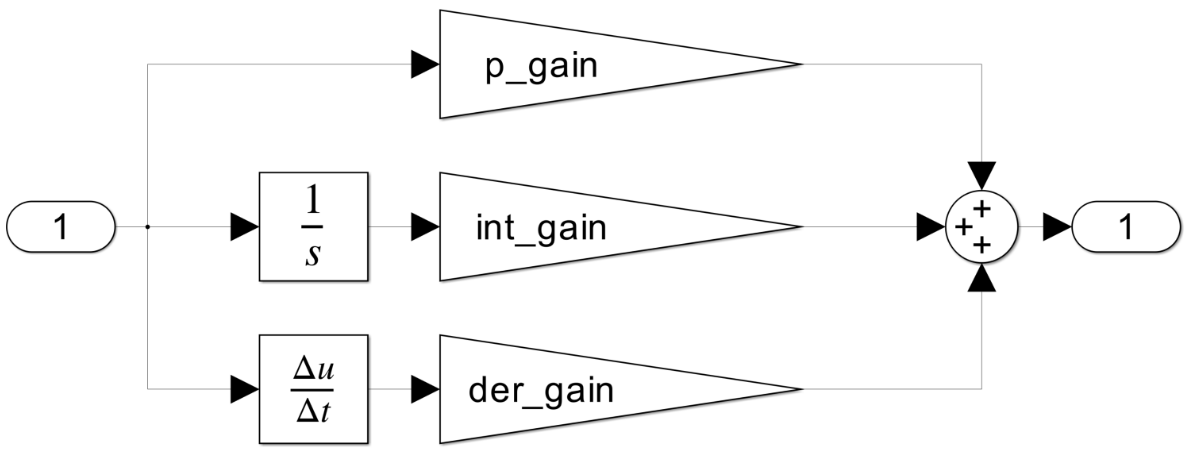

Figure 2.2: PID section of the Part 2 block diagram.

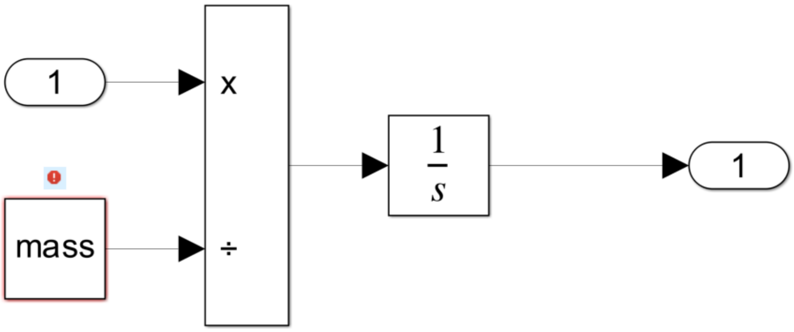

Figure 2.3: Plant section of the Part 2 block diagram.

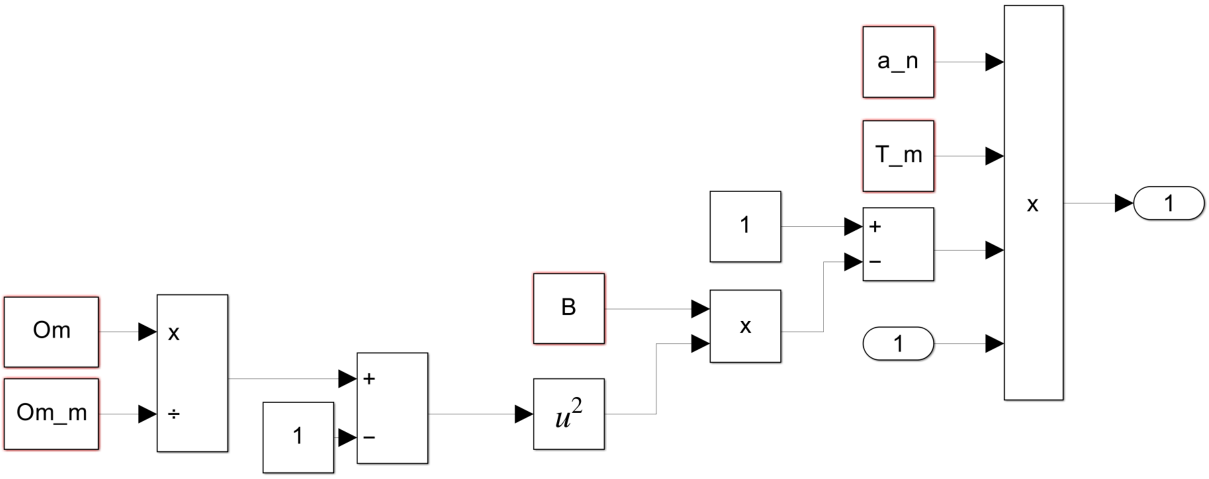

Figure 2.4: Motor section of the Part 2 block diagram.

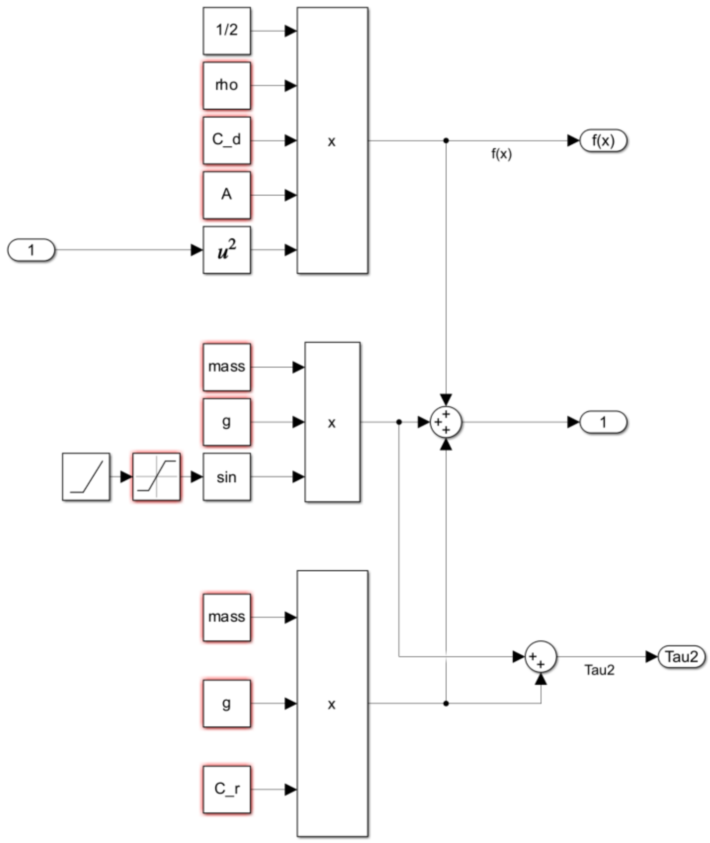

Figure 2.5: Disturbance force section of the Part 2 block diagram.

2.1.3 Phase-Space Plots and the Equilibrium Point

A phase-space plot was produced for each gear ratio, using Simulink's multiple simulations and data inspector tools. This allowd for the equilibrium point of each gear to be determined.

2.1.4 Modelling Gear Changes

For a more accurate representation, gear changes were then modelled using Simulink's switch block. This can be seen in Figure 2.6.

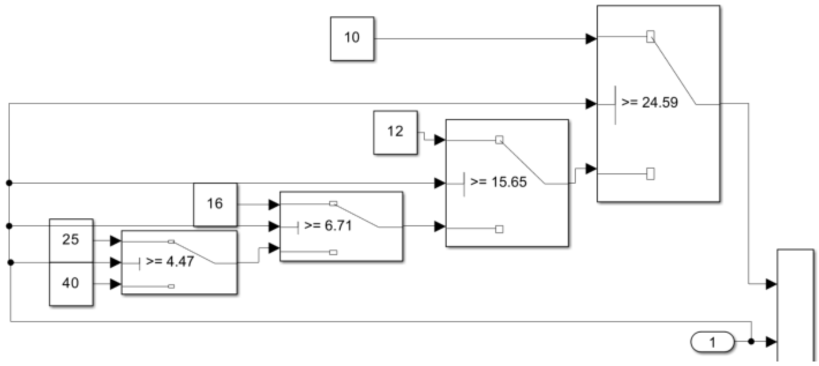

Figure 2.6: Section of block diagram showing the changes made to the motor block.

The speed range associated with each gear can be seen in Table 2.1.

Gear	Speed (mph)	Speed (m/s)
1	0 - 10	0 - 4.47
2	10 - 15	4.47 - 6.71
3	15 - 35	6.71 - 15.65
4	35 - 55	15.65 - 24.59
5	55+	29.06+

Table 2.1: Speed range for each gear.

2.1.5 Feedback Linearisation

Once the initial block diagram was made, and the equilibrium points found, the equation used in Part 1 was re-written as a second order equation, with the dependent variable changing from velocity to displacement. This is shown in Equation 2.6.

$$m\ddot{x} = a_n u T_m (1 - (\frac{\Omega}{\Omega_m} - 1)^2) - mgC_r - \frac{1}{2}\rho C_d \ddot{x}^2 - mg\sin\theta \tag{2.6}$$

From this, an equation representing the control torque, τ_c, was derived.

$$\tau_c = DV + f(x) - \tau \tag{2.7}$$

$$\tau_c = m(k_v(\dot{x}_sp - \dot{x}) + k_d(x_sp - x) + \ddot{x}_sp) - \frac{1}{2}\rho C_d \dot{x}^2 - \tau \tag{2.8}$$

Where:

$$\tau = a_n u T_m (1 - (\frac{\Omega}{\Omega_m} - 1)^2) - mgC_r - mg\sin\theta \tag{2.9}$$

This required the addition of 2 new gains - velocity gain, k_v, and displacement gain, k_d.

2.1.6 Block Diagram with Feedback Linearisation

The Simulink block diagram therefore required the addition of a control torque, as can be seen in the below Figures.

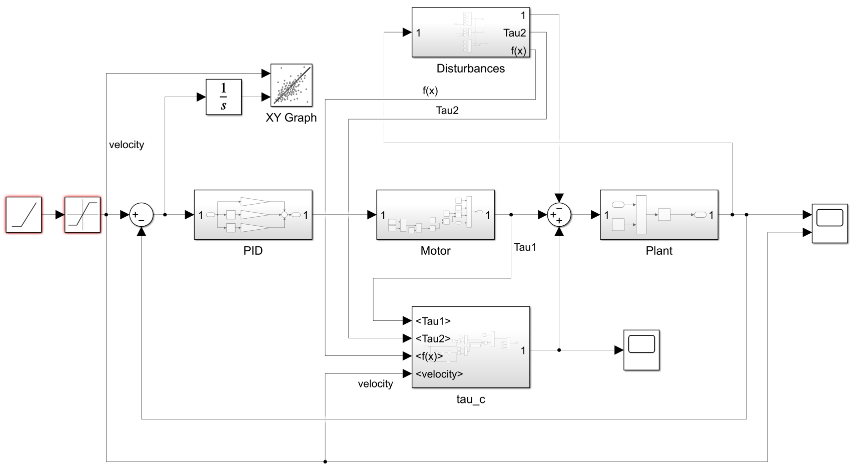

Figure 2.7: Block diagram featuring advanced dynamics and feedback linearisation.

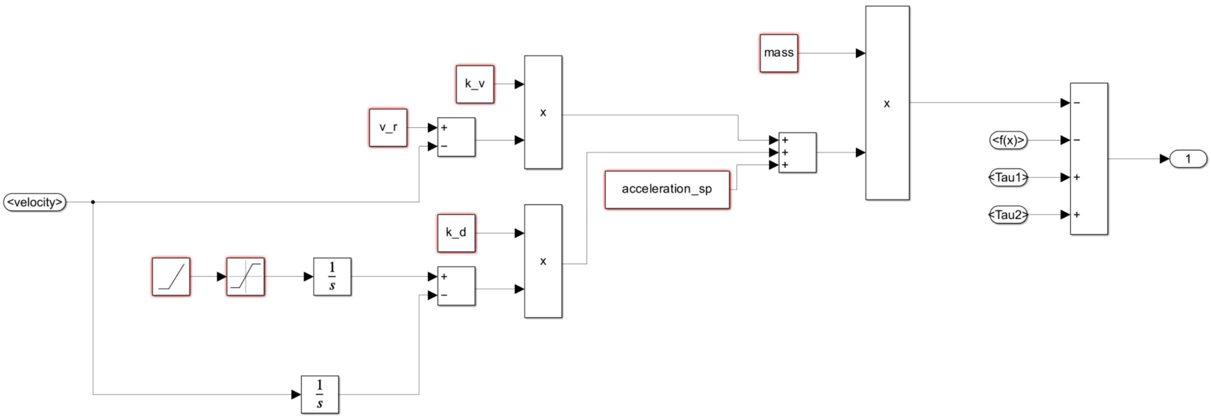

Figure 2.8: Segment of the feedback linearised block diagram showing the control torque.

An integral block was used to obtain the current total displacement of the car, and using the same ramp input as the velocity set point, the same method was used to obtain the current desired displacement of the car. A similar process as discussed in Part 1 was then used to determine appropriate values for the two additional gains.

2.1.7 Additional Disturbance Cases

This controller was also tested with the same cases discussed in Part 1 - firstly, a ”bumpy road” disturbance, modelled as a sine wave with low amplitude and high frequency, and then a ”hilly road” disturbance with a higher amplitude and lower frequency.

2.2 Results

Initially, the same gains found in Part 1 were used for Part 2 - P = 1.3, I = 0.65, D = 0.3. However, the derivative gain was not contributing a lot to the system response, and it frequently caused issues with the solver so was set to 0.

2.2.1 Feedback Linearisation

The two additional gains were selected by iterating through many combinations of values ranging from 1.0 - 15.0. An optimal configuration of k_v = 4, and k_d = 12 was chosen due to the low settling time and suitable v_{peak}.

2.2.2 Determining Equilibrium Points

A phase space diagram was created, showing all of the gear ratios (note that this was done before gear changes were introduced). This can be seen in Figure 2.9.

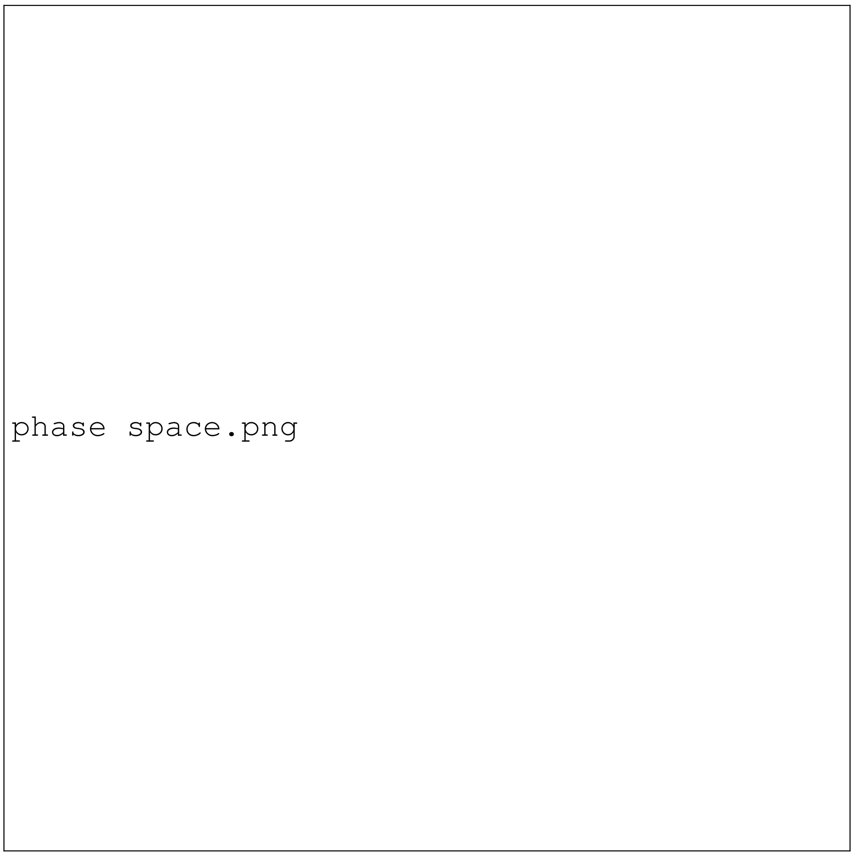

Figure 2.9: A phase space diagram for each gear ratio. The x-axis is speed (m/s), and the y-axis is the integral of the error.

2.2.3 System Response and Gear Modelling

The system responded to the initial test case by briefly producing a negative speed. It was believed that this could be due to the gear ratio used being inappropriate for the speed of the car, and therefore gear modelling was introduced as described in Section 2.1.4. A comparison of the system before and after this was carried out can be seen below.

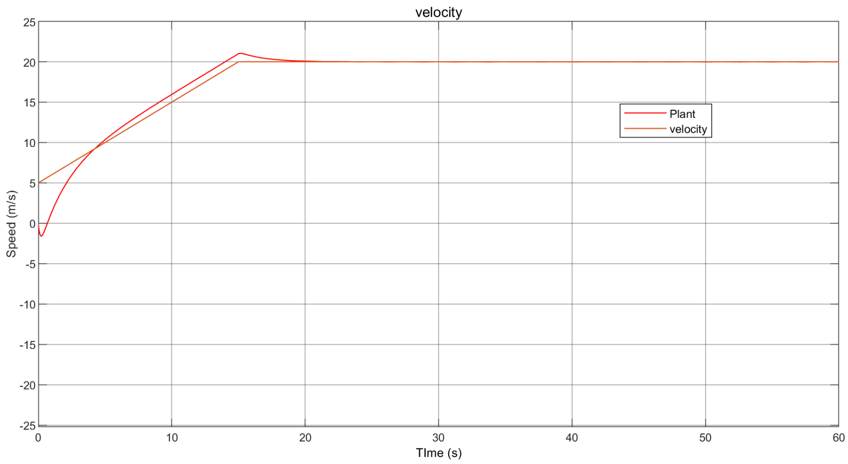

Figure 2.10: System response using a static gear ratio of 16.

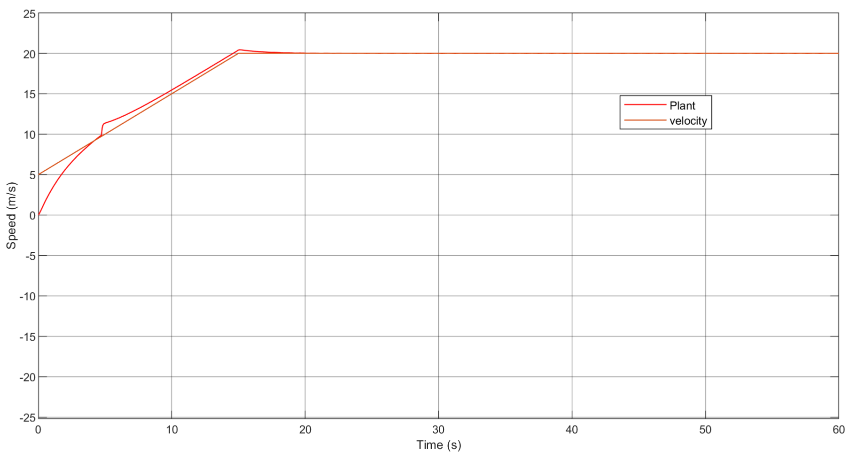

Figure 2.11: System response using a dynamic gear ratio.

2.2.4 Control Torque

At this point, the control torque was also plotted to allow for further analysis.

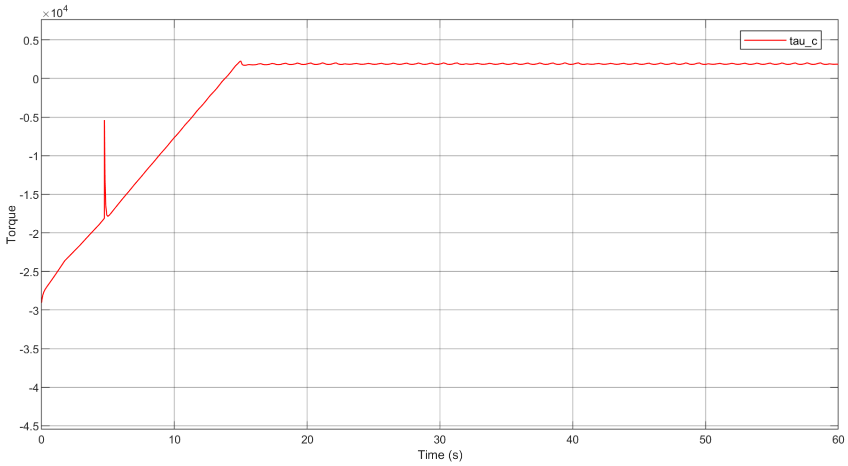

Figure 2.12: Control torque plotted against time.

2.2.5 Additional Disturbance Cases

The ”bumpy road” disturbance was set up as a sine wave with amplitude 0.1m and frequency 1rad/s. The system response can be seen below, as well as the phase-space plot and control torque over time. This was done with the gear change modelling.

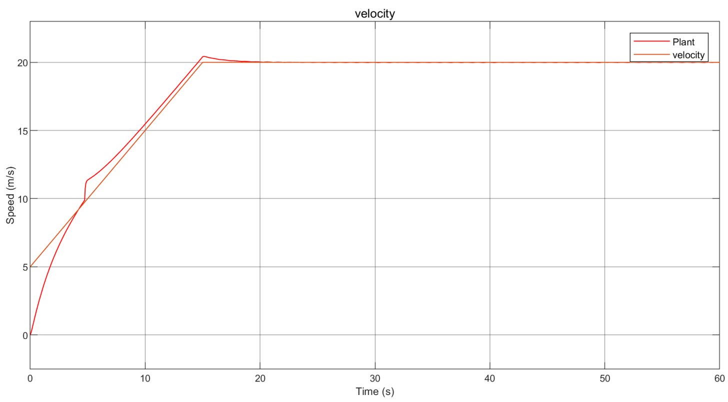

Figure 2.13: Bumpy road system response.

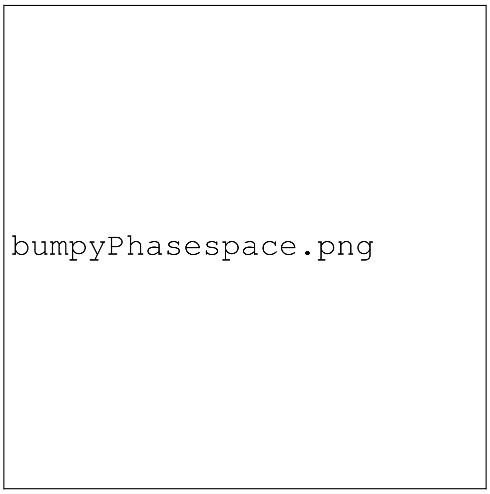

Figure 2.14: Bumpy road phase-space plot.

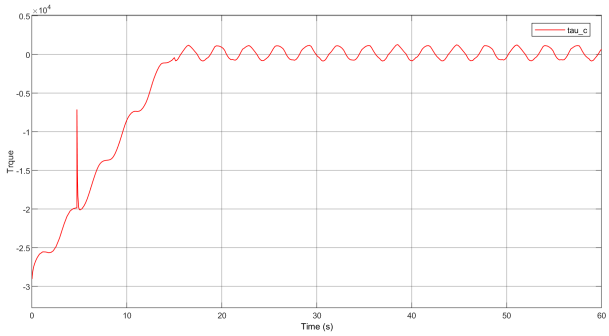

Figure 2.15: Control torque for the bumpy road scenario.

The "hilly road" disturbance was set up as a sine wave with amplitude 1m and frequency 0.2rad/s. The system response can be seen below, as well as the phase-space plot and control torque over time.

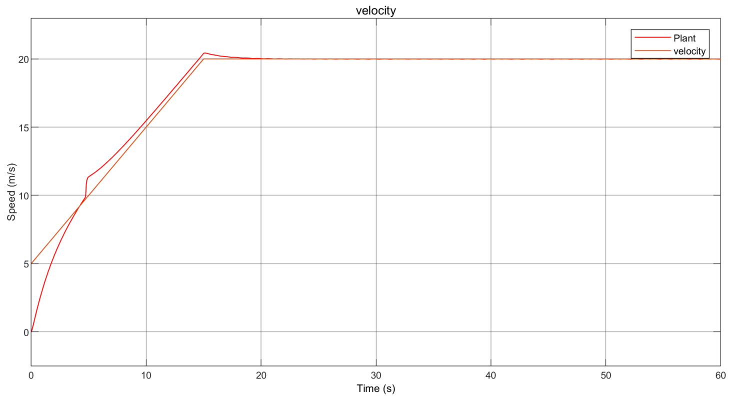

Figure 2.16: Hilly road disturbance system response.

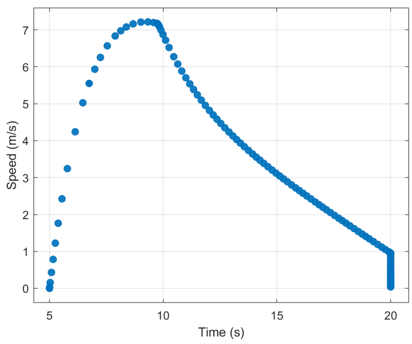

Figure 2.17: Hilly road disturbance phase-space plot.

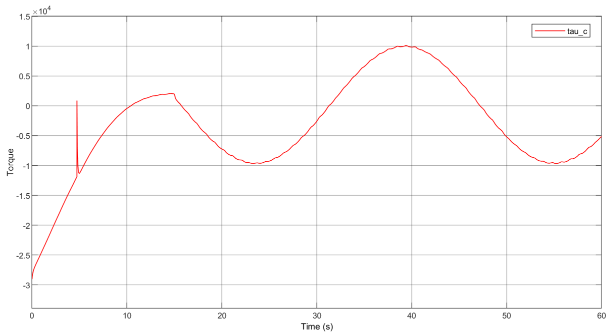

Figure 2.18: Control torque for the hilly road scenario.

2.3 Discussion

2.3.1 Equilibrium Points

As can be seen on the combined phase-space plot, the higher the gear used the higher the integral of the error for the final equilibrium point. All of the gears were able to come to equilibrium around the 20m/s set point speed, and once at this speed the integral of the error began to reduce. This was also seen for extra disturbance cases.

2.3.2 System Response and Control Torque

The inital system response, when using a gear ratio corresponding to 3rd gear throughout the simulation, was overall very good with low steady-steady state error, an acceptable V_{peak} and a low settling time. However, the controller output a negative speed for the first second. Once gear shifting was introduced, this was rectified and the controller produced an even better system response.

2.3.3 Additional Disturbance Modelling

As shown by the system response and control torque plots, the controller could handle both the hilly and bumpy road scenarios adequately. The amount of control torque required (both in the basic disturbance scenario and the two more demanding scenarios) is somewhat unrealistic and if further work were to be conducted work should be done to limit the amount of control torque able to be produced. This would likely have a large effect on the stability of the system and would require a lot more fine tuning of the gains.

2.4 Conclusion

To conclude, in Part 2 the dynamics of the car were modelled more realistically, which introduced some non-linear equations. Feedback linearisation was used to account for this. A gear shifting model was then added to improve the realism of the simulation, and two different road condition scenarios were used to test the controller. The controller was able to handle the disturbance scenarios very well, however to do this the control torque was in the magnitude of 10^3, which is somewhat unrealistic. Future work could be done on limiting the control torque to a more realistic value, and if that is done a lot more fine-tuning would have to be done to ensure stability.

3 Course Review

3.1 Part 1

I found Part 1 to be very straightforward with lots of places to dive deeper than the supplied theory, which made it very interesting. At times I struggled a little with choosing what to include or leave out, given the 20 page limit. A separate page allowance of 2-3 pages for a literature review could have been helpful for this, as a way to show any extra topics that we explored. Aside from that, I believe all of the lecture content up to stability was covered by this section of the coursework. Perhaps a little guidance on ideas for how to model more complicated disturbances could have been helpful, as I only worked out how to do much more complicated disturbance models (such as adding in specific road profiles) much later in the project, and as such did not have much time to write about this. However I do not think that this detracted much from the overall project.

3.2 Part 2

In Part 2, I believe all of the content from the lectures was covered by the project. However, I think there would have been a benefit to having one additional section here where we were given an allowable range for the control torque magnitude and some guidance on how to reduce the control torque whilst keeping the system stable. I also personally struggled a little with understanding the phase-space plots and equilibrium points and looking through the slides on this was still confusing for me, but in hindsight I should have asked about this during a tutorial or lecture and that would likely have sorted the problem.

3.3 Overall

I think I would have benefited from splitting the project into 2 submissions - a mid-semester Part 1 submission, and an end of semester Part 2 submission. I also would have enjoyed some freedom over the application for the controller - controlling a satellite ADCS system for example would have been more interesting to me, however I understand the difficulties in teaching that could arise from having every student doing a different project.

Aside from the above points, I have greatly enjoyed this course and found it well-balanced between theory and hands-on application. The content of the lectures was almost always useful for the stage in the project that I was at, which helped me to keep on top of the work.

References

[1] Omega, “Pid controllers.” [Online]. Available: https://www.omega.co.uk/prodinfo/pid-controllers.html

[2] NI, “Pid theory explained.” [Online]. Available: https://www.ni.com/en/shop/labview/pid-theory-explained.html

[3] TOMTOM, “A brief history of cruise control systems.” [Online]. Available: https://www.tomtom.com/newsroom/explainers-and-insights/a-brief-history-of-cruise-control-systems/

[4] R. L. Barristers, “Phage lambda: description & restriction map.” [Online]. Available: https://roadlawbarristers.co.uk/speeding-what-is-the-10-plus-2-rule-and-what-does-it-mean-2/#