Final Report for MTHE 393

Sean Kato 20012208
Stefan Robb 20027476
Justin Yan 20029242
Maeve Buchan 20020510

April 8, 2019

1 Executive Summary

The required tasks for project completion can be divided into two categories: system analysis and project applications.

The first step of system analysis was verifying that the black box demonstrates the attributes of a linear time-invariant system. This analysis was completed in MATLAB as seen in *Section 5.1*. The subsequent progression involved creating a filter for the system that would successfully eliminate the majority of the noise produced by the unknown system; this filter can be seen in *Figure 12* of *Section 5.2*. Bode plots were then created to model the frequency response of the system, shown in *Section 5.3*. The Bode plots were generated by testing the system at a range of frequencies from $\omega = 10^{-4}$Hz to $\omega = 10^{4}$Hz. The magnitude and the difference between two zero-crossings were for recorded for each response. The magnitude and phase Bode plots were then generated using this data. A heuristic transfer function was then created in *Section 5.4* from the bode plot to model the behaviour of the unknown system. A PID controller was used to control the known heuristic system to attempt to meet the design specifications that are defined in *Section 5.9*. These controls were then applied to the unknown system in *Section 5.8*. After further tuning of the control variables, and the addition of a real-time filter, the black box system was finally controlled.

The selected project application is a model of the regulation of the internal temperature of a combustion chamber on a rocket. In this analysis, all relevant stakeholders were identified and a clear project scope has been defined in *Section 3*. In addition, the constraints of the application have been established.

Contents

2 Introduction

The objective of this project was to approximate and control the inner workings of an unknown system described as a "black box" and apply it to a complex real-world system. The approximation includes a linear time-invariant (LTI) state-space or transfer function that heuristically models this black box. This final result was achieved through a thorough analysis of system types, input/output behaviour, noise, causality. The application for the assumed LTI system being studied is the regulation of the internal temperature of combustion chamber on a interplanetary spacecraft.

The following report will elaborate on the application, methodology, project results, and additional impacts.

3 Application

3.1 Problem Definition

To generate enough thrust to lift a spacecraft off the earth and into outer space kerosene (fuel) is burnt with liquid oxygen (oxidizer). Kerosene has a very high auto ignition temperature. The temperature inside the combustion chamber can reach as high as 3000 degrees Celsius to generate the required thrust for the spacecraft. This temperature is too hot to house a sensor [1]. To control the thrust of the spacecraft we must control the temperature inside the combustion chamber, however the high temperatures can make for this to be a difficult task. A technical diagram of the control problem can be seen in *Figure 1*.

If the combustion chamber temperature is left unregulated, mechanical components of the rocket may cease to work or rapidly degrade in quality as a direct result of this extreme heat. To mitigate this risk, sensors are placed outside of the combustion chamber, a significantly cooler location, to measure the external temperature. Kalman filters are then used to accurately estimate the internal temperature [1]. This estimated internal temperature will be referenced against an approved temperature value. Based on the difference, the controller will determine how much cooling is required; the most common method of cooling being film cooling [2].

As fuel is burnt and the environment around the space craft changes the conditions of operation for the controller will vary. The "black box" in

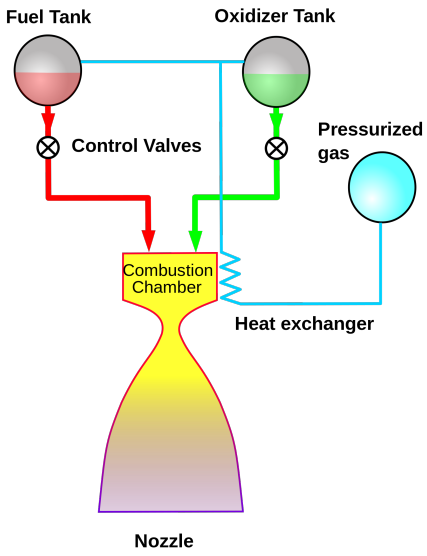


Figure 1: Technical Diagram of Process to Control Thrust in a Spacecraft.

this application is the combination of external factors influencing the difference in temperature between the interior of the combustion chamber and the exterior sensor. If one assumes constant thrust (resulting in a speed of at least 7.9km/s [4]), then one encounters compressible flow [3]. Moreover, other factors may include: ambient air temperature, humidity, and changing atmosphere. Thus, there is no clear way to directly model this system, justifying the black box assumption.

This assumption of constant thrust is the main constraint of this application, since as a rocket burns fuel, it is losing mass. If one has constant thrust whilst decreasing mass, then the speed is actually increasing over time. Thus, it is not an entirely realistic assumption that the trust of the rocket will be constant. A TBL chart for the stakeholder analysis of this application can be seen in *Appendix A*.

3.2 Triple Bottom Line

Spacecraft projects range from $62 million for smaller private spacecrafts to $200 billion for large federal spacecraft projects. With these associated costs

the success of missions are critical, the smallest malfunction of a piece of equipment on the spacecraft can lead to disastrous loss of time, lives, and resources. The most important aspect of the temperature controller is its safety and reliability. To ensure that the sensor is functional it will be tested under simulated conditions in many different circumstances to ensure that when required in abroad a live spacecraft it will never fail.

There are currently educational programs around the world informing young minds about the mysteries of our solar system and the science being used to solve them. These programs are key for generating interest in the industry and attracting talented youth to become future space engineers. In doing so, students that express an interest in space have a line of support to help them pursue their passion.

Up until December 2015, spacecrafts used to return a fraction of what left the Earth. Pieces of spacecrafts would disconnect along its trajectory in order for astronauts to ungracefully crash-land back to Earth in landing pods. The disconnected pieces of the spacecraft can be referred to as space debris and either float around in space or even fall back to Earth. In 2011, SpaceX announced its reusable launch system development program that would allow spacecrafts to return to Earth landing vertically in once piece for the first time ever. In December 2015, the Falcon 9 successfully achieved this feat. By doing so, the spacecraft could be refurbished and fit for another launch saving money, time and resources. For this to be possible, much of the equipment on the spacecraft must be modified for multiple launches. Combustion chambers controllers and sensors will be designed accordingly as it benefits stakeholders economically and environmentally.

Figure 2: SpaceX first ever successful landing of an autonomous spacecraft (on a boat).

3.3 Economic Analysis

The budgets for federal space agencies has been declining over the past couple years, however, since the emergence of private spacecraft manufacturers the industry has been on the rise [13]. Private manufacturers have been capitalizing on the economic opportunities that space offers. As soon as 2030, a space hotel will be launched allowing humans to take a private spacecraft to space and spend a few nights with an associated cost [14]. In addition, recent studies have shown that some uninhabitable planets may hold valuable resources to humans on Earth [15].

Currently, the space industry is very costly, so private manufacturers attempt to reduce costs by creating a economic system of businesses. One aspect of this system will include third-party manufacturers for individual pieces of equipment and systems for spacecrafts. This will allow manufacturers to hire fewer specialists in fields pertaining to space travel. This allows manufacturers to focus more on their individual goals while reducing costs, building time, and resources required. This economic system is efficient for all parties in the industry and is a solid model to build businesses.

When constructing the combustion chamber temperature controller, the business will only need to acquire the necessary resources to build sensors. This will allow for resources to be bought in larger quantities and more in-depth relationships to be built with the suppliers. Additionally, clients can experiment with many interchangeable resources to determine which may be

bests in specific conditions and situations. This will increase performance of the sensor as they can be optimized for their use. The sensors will be physically built in the factory by engineers using customized systems. Once the sensors are complete they will undergo rigorous testing and simulation to ensure they meet performance and safety specifications. After the sensors have been approved they will be distributed to clients and use in space missions around the world.

3.3.1 Cost Breakdown

An estimated cost breakdown for the start up of the business and the revenue for the first couple years was generated. Table 1 and Table 2 show the fixed costs and variable costs for the first year, respectively.

Manufacturing equipment	$340,000
Commercial space	$450,000
Licensing	$25,000
Initial technology	$100,000
Total fixed costs	$915,000

Table 1: Fixed Costs in the First Year.

Employee wages	$530,000/year
Testing and simulation	$12,000/sensor
Manufacturing costs	$500/sensor
Material costs [16]	$2,000/sensor
Additional Expenses	$10,000/year

Table 2: Variable Costs in the First Year.

A cost not included was debt, as it would be conditional on the amount of money the company decides to loan for their business venture.

The business hopes to sell approximately 10 sensors in their first year. Contract talks with federal space agencies and private manufacturers have led to the business believing that 6 sensors will be sold in the private industry and 4 in the federal industry [17]. The business expects the number of sensors demanded in the private industry to increase dramatically and the number of sensors demanded in the federal industry to decrease.

Based on cost estimates the sensors will cost approximately $75,000 to manufacture. Exact cost will depend on the intended spacecraft size and purpose. The larger the spacecraft and its combustion chamber, the more expensive the sensor will be. The business plans on selling the sensors at around $90,000 for an approximate profit of $15,000 per sensor. With these profits the business would be able to pay of any debt it took on initial fixed costs in 6 years. After that the business can begin reinvesting their profits back into the business.

3.4 Regulatory Concerns

The technology for the spacecrafts must pass the most strict regulations and tests in order to be deemed safe for flight. The temperature control system in the combustion engine must operate correctly to ensure that the rocket can safely leave the atmosphere and return. The most important feature to its target audience is its reliability and safety. These are two necessities to the success of the combustion engine temperature control system.

4 Methodology

The greatest problem the team faced was understanding the black box. The process of overcoming this challenge has been outlined in *Section 5.1 and 5.2* and the methodology behind the approach is outlined in all of *Section 4*. To ensure the team met our target goals, the project was broken down into weekly tasks. *Table 3* outlines the project schedule.
The team diligently met twice a week to ensure that these tasks were completed and share progress. To develop a process to account for the random effects in the model, the team brainstormed possibilities on a white board and evaluated the most feasible and effective options using an evaluation matrix. A similar process was used to identify applications for the project. A sticky note design thinking process was completed to identify the environmental,social, economic and regulatory aspects of the project for each application.

Date	Goals
Week 7	Develop a process to account for the random effects in the model
Week 8	Understand and identify the environmental, social, and economic impacts of the project Understand the regulatory aspects of the project
Week 9	Develop a frequency response model for the system
Week 10	Review feedback from interim report, prepare for interview and validate transfer function
Week 11	Progress on control law design for model system, Implement model control law on actual system in simulation
Week 12	Finalize report and presentation

Table 3: Project timeline.

Figure 3: Sticky Note Design Thinking.

The most effective way to distribute tasks was to allow each team member to work with their strengths and interests individually but review all the work together. Implementing this strategy allowed the team to complete tasks efficiently while ensuring the whole team is on the same page and understands every aspect of the project.

5 Design

5.1 LTI Verification

Perhaps the most important underlying assumption of the model is that it resembles a linear time-invariant (LTI) system. For a system to be LTI, one must satisfy both linearity and time-invariance properties. Linearity, by definition, is when a function satisfies both additivity and homogeneity. In other words, the following equation must hold:

$$f(au + v) = af(u) + f(v) \tag{1}$$

for constant a, and variables u,v. A system is time-invariant if a time-shifted input correlates to the same time-shift in the output. In other words:

$$Response(f(t - \delta), t - \delta) = Response(f(t - \delta)) \tag{2}$$

with δ equal to the corresponding time-shift. To validate this assumption, the model was run 20 times with three different inputs, the third being the sum of the first two inputs. The average values of each respective input were then calculated. To minimize the effect of white noise on the output 20 unique trials were averaged. To validate these properties, the difference of the first two inputs and the third input were plotted, with the goal of being relatively close to the constant zero function.

To elaborate with specific examples, let $f_1(t) = 5$, $f_2(t) = 5t$, and $f_3(t) = 5(1 + t)$. The averaged and filtered values of $f_1(t) + f_2(t)$ and $f_3(t)$ are shown in *Figure 4*.

To better see the relationship between the functions, the difference between $f_1(t) + f_2(t)$ and $f_3(t)$ is displayed in *Figure 5*.

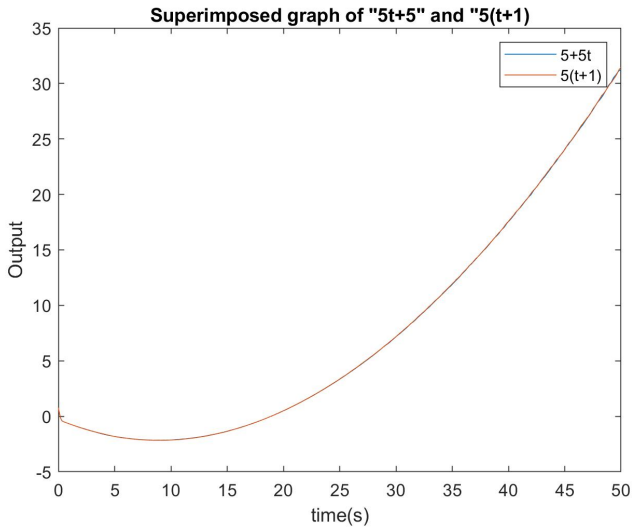


Figure 4: Averages of $f_1(t) + f_2(t)$ and $f_3(t)$.

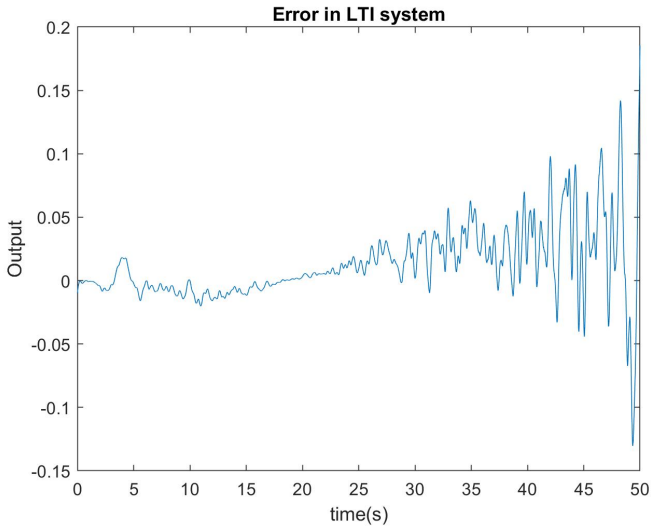


Figure 5: Difference between $f_1(t) + f_2(t)$ and $f_3(t)$.

The magnitude of the error in *Figure 5* is approximately bounded by 0.2, which remains within a reasonable tolerance of the zero function. Thus, the system mimics the behaviour $f(5 + 5t) = f(5) + f(5t) = 5f(1 + t)$, demonstrating both linearity and time-invariance.

To further validate the assumption of an LTI system, the aforementioned process was repeated with $f_1(t) = sin(5)$, $f_2(t) = sin(5t)$, and $f_3(t) = sin(5(1+t))$. The average values of $f_1(t) + f_2(t)$ and $f_3(t)$ are displayed in *Figure 6*. Again, the difference between $f_1(t) + f_2(t)$ and $f_3(t)$ is displayed

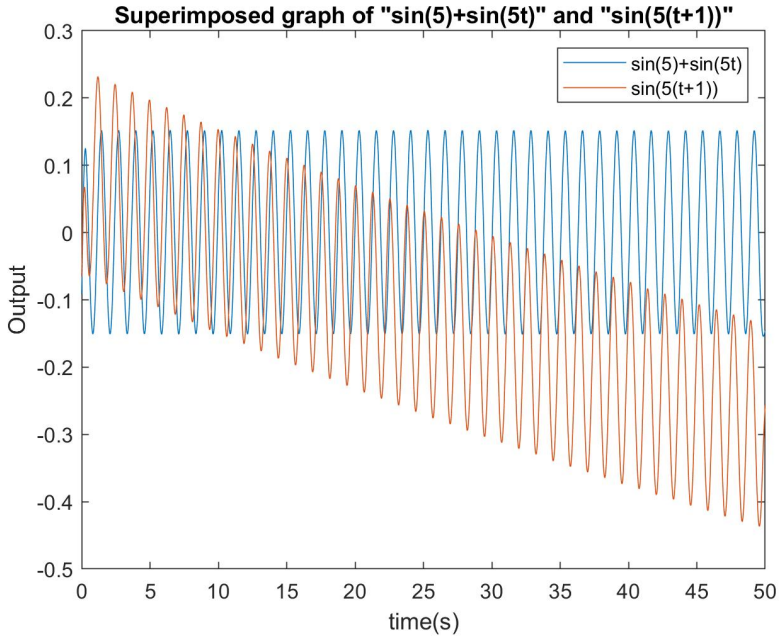


Figure 6: Averages of $f_1(t) + f_2(t)$ and $f_3(t)$.

in *Figure 5*. Despite the magnitude of the error in *Figure 7* having a larger bound of 0.5 versus the bound of 0.2 in the previous example, a clear LTI relationship is displayed in *Figure 7*. Hence, one can further justify the LTI assumption of the model being studied.

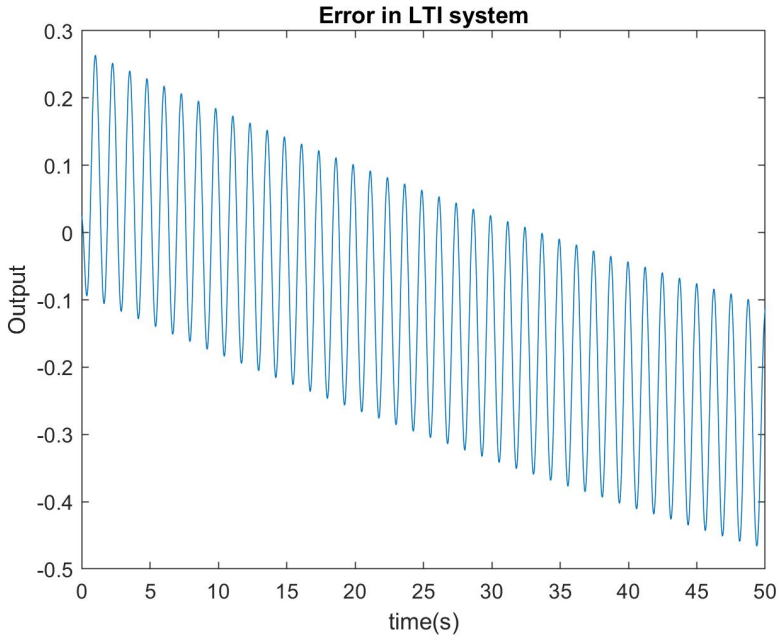


Figure 7: Difference between $f_1(t) + f_2(t)$ and $f_3(t)$.

5.2 Filtering

The initial filtering process began by applying a Savitzy-Golay filter to an averaged sinusoidal function. The idea behind the filtering was to eliminate random noise from the system by preforming 20 iteration of the "Blackbox" and taking the average value at each time point. The results were a much smoother function albeit with little reasoning and validation for the process.

After more research and reflection the filtering process re-begun. To create a valid filter it was essential to first determine the nature of the noise produced. After researching various types, such as pink, low frequency, impulsive, ect, the noise was determined to be proportional and white. This came as the amplitude of the random noise is proportional to the amplitude of the signal as well as there is an equal distribution of noise above and below the function average. The proportionality characteristic can be seen below in *Figure 8*.

A general sinusoidal function, sin(t), was iterated 30 times and averaged to eliminate some randomness of the white noise. This function can be seen in *Figure 9*. This new averaged function was then subtracted from a random

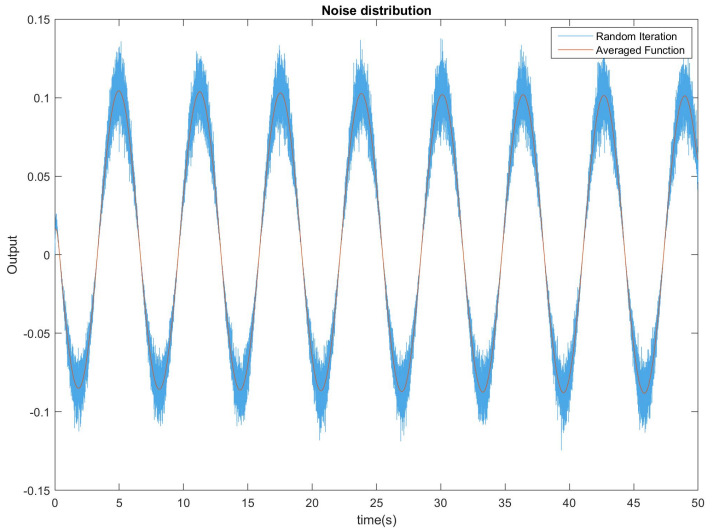


Figure 8: Proportional noise in system.

iteration to demonstrate the even noise distribution, *Figure 9*. This result confirms that the white noise is added to the "true value" of the signal about μ=0.

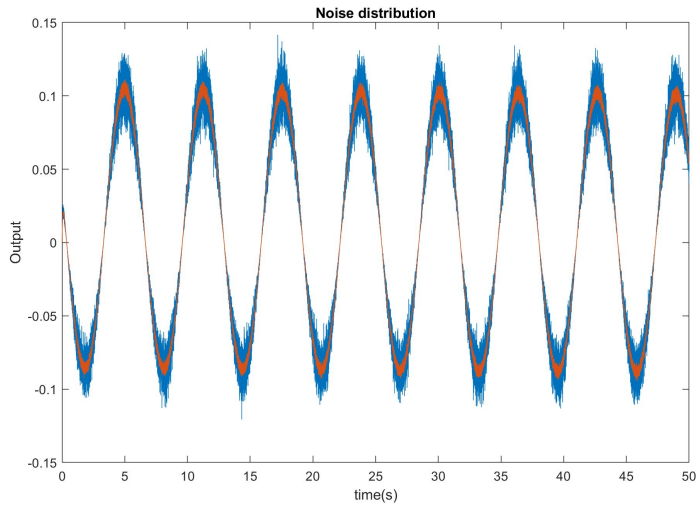


Figure 9: Averaged data over 30 iterations.

The distribution of the noise was evaluated and determined to be Gaussian/normal in distribution. This classification was achieved through researching the different types of noise distribution present in systems and comparing the system's results. *Figure 10* was retrieved from [11] and gives a visual representation of the different types of distribution and fits for noise that were analyzed.

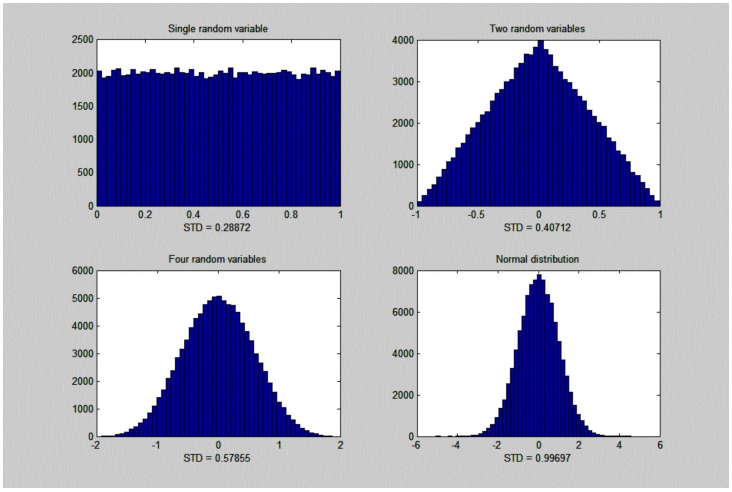
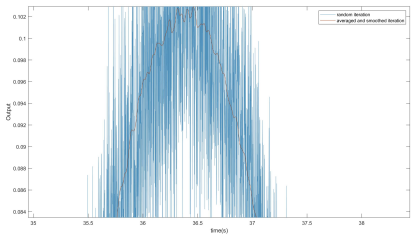
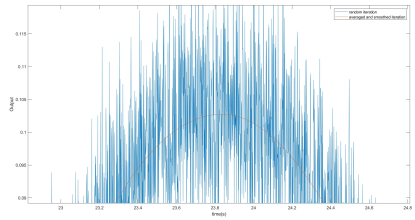


Figure 10: Graph of histogram and distributions.

Following this identification stage the filtering (or "data smoothing") process began. The first step was to research different means of smoothing data such as by: moving average, Savitzy-Golay, median, and finally the Gaussian filter. Based on the nature of the noise and its distribution and experimental trials the Gaussian filter was selected as the desired filter. A built in MATLAB function was utilized to smooth the signal using a Gaussian-weighted moving average over a window of 500. This smoothing function operates by assigning a weighted average of neighbouring elements to a specific data point. The central data point will then be given the greatest Gaussian value, with values decreasing proportional to distance. The window of 500 determines the number of neighbouring points that will be calculated in the average. The higher the window ultimately leads to a smoother resulting function, as seen in *Figure 11* below.



(a) Gaussian filter with window of 50.



(b) Gaussian filter with window of 500.

Figure 11: Plots of Gaussian smoothing function with different window inputs.

The filtered output was then superimposed on a random iteration to validate its systematic fit, this can be seen in *Figure 12*.

The next steps for filtering would include more research into Kalman filters and their implementation into the system. As Gaussian filters are non-causal it would be beneficial to implement an optimal estimation algorithm with causal properties to improve the filtering process. The filter was not chosen due to comprehension limitations in the restricted time period.

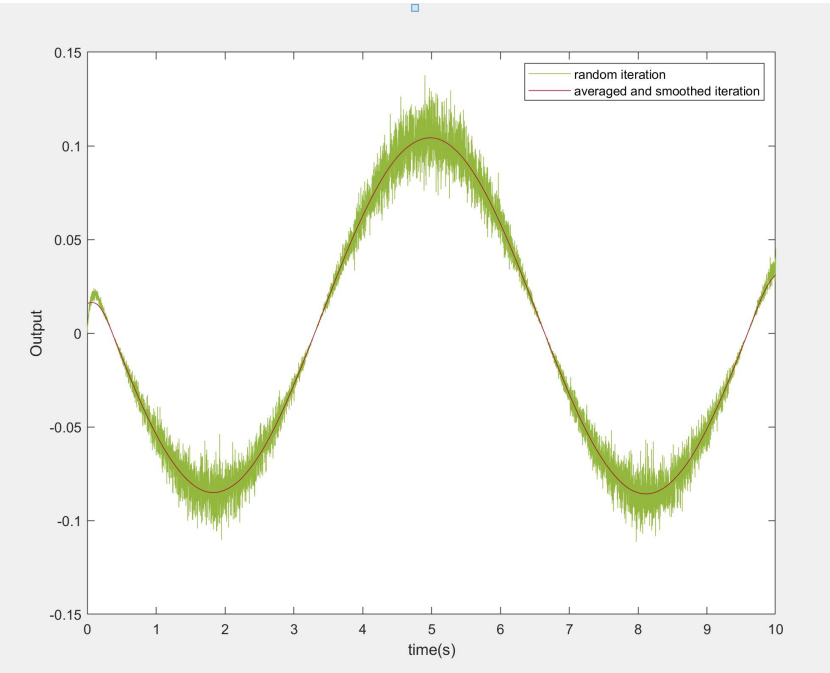


Figure 12: Filtered signal superimposed on a random iteration.

5.3 Bode Plots

The following Bode plots were generated by by analyzing the response of the system when sine waves with different frequencies were inputted into the system. The range of frequencies used were from $\omega = 10^{-4}$ to $\omega = 10^{4}$, with 60 points logarithmically spaced between the start and end frequencies. The "blackBox" was ran with a start-time of 0, a step-size of 0.01, and a refine output value of 1. The end-time varied for each frequency as the very low frequencies (in the region of 10^{-4} to 10^{-3}) required an end-time of up to 40,000 while the higher frequencies needed only an end-time of 1. *Figure 13* depicts the magnitude Bode plot that was generated.

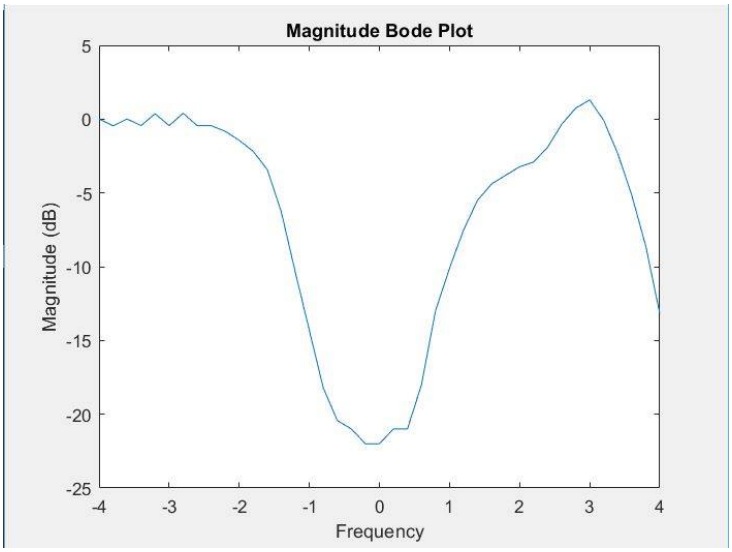


Figure 13: Magnitude Bode plot.

The magnitude Bode plot was generated by plotting $20log_{10}(|magnitude|)$ by $log_{10}(\omega)$, where magnitude was the amplitude of the outputted sinusoid. The magnitude of the sinusoid as determined $\frac{max-min}{2}$. Where max and min are the maximum and the minimum values achieved by the system. One problem that arose was that the filter had to be adjusted for each frequency. Setting the window size of the Gaussian filter to 500, which was used when testing the filter, was not sufficient when the data set for a single sin wave at the low frequencies had over 20,000,000 points. Thus, the window size was manually changed for each frequency based on the size of the output array.

Figure 14 shows the phase Bode plot that was generated.

The phase plot was generated by plotting the phase angle of the responses by $log_{10}(\omega)$. The phase angle ϕ (degrees) was found by:

$$\phi = |(ztd)|\omega\frac{180}{\pi} \tag{3}$$

Where zero-time difference (ztd) is the time between the two zero crossings. This was found by iterating through each data point collected from the responses and finding when the response crosses zero. A pair of zero crossings was then chosen to be used for the calculations. The pair that was chosen was chosen arbitrarily but sufficient time was given so that the system could settle.

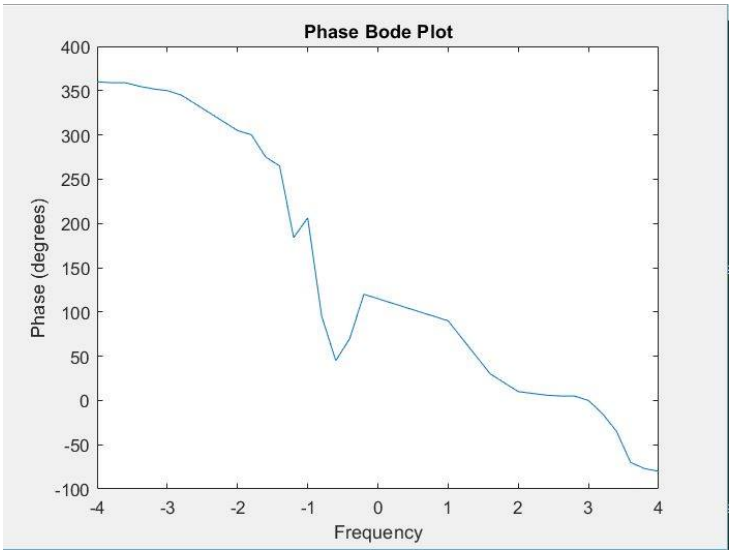


Figure 14: Phase Bode plot.

5.4 Creating the heuristic transfer function

After the completion of the experimental Bode plot, a heuristic function for the system was made. To create a transfer function from a Bode plot, one must first identify the notable increases and decreases in slope in the magnitude graph. Increases correspond to zeroes of the system, and decreases correspond to poles. Moreover, the slope itself (in multiples of 20) correlates to the degree of the zero or pole at a given location. For example, a slope of $+20$ represents a zero of degree one, whereas a slope of -40 represents a pole of degree two. By convention (and for BIBO stability), one first assumes that all of the zeroes and poles are negative. *Figure 15* shows the approximate locations of these changes in slope; red circles correspond to zero locations and black squares correspond to poles.

The above pole and zero locations were fine-tuned to ensure the generated transfer function magnitude curve matched the shape of the experimental one. The two curves were then plotted on the same graph, shown in *Figure 16*. *Figure 16* clearly shows that the heuristic curve is similar in shape, but shifted down by approximately 68 dB. To correct this issue, a gain constant, denoted k, was introduced. To get the appropriate value of k, one must first get the magnitude values associated with the leftmost frequency that both curves are defined. In this case, this frequency is 10^{-3}. The magnitude values

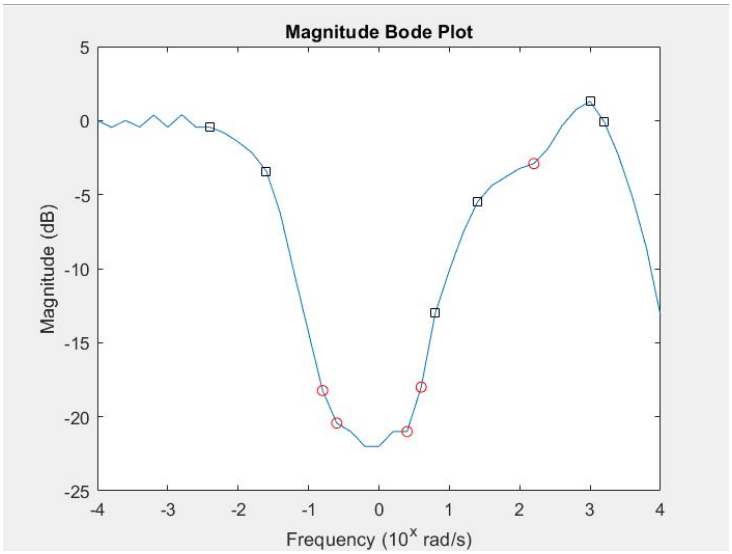


Figure 15: Identified poles and zeroes of the system; circles are zeroes and squares are poles.

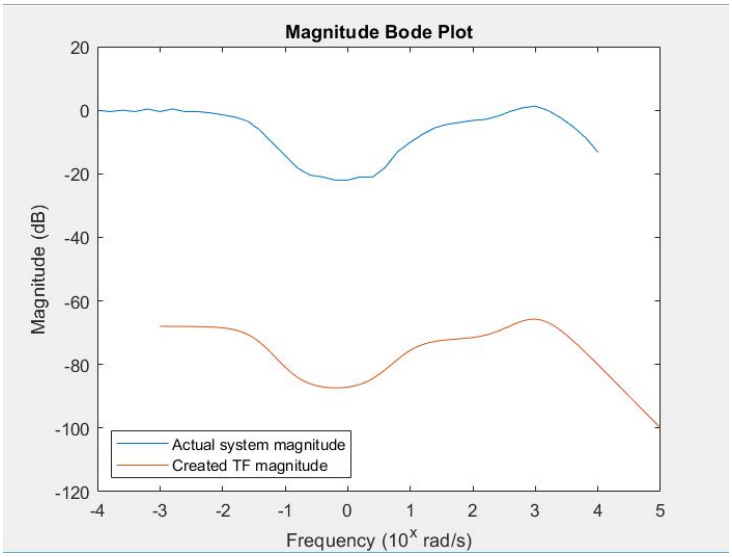


Figure 16: Experimental (blue) and heuristic (orange) magnitude curves without gain constant.

for the experimental (M_E) and theoretical (M) curves at this frequency are $M_E = -0.4441$ and $M = -68$, respectively. Now, define $\Delta = M_E - M$.

Then, k is given by:

$$k = 10^{\frac{\Delta}{20}} = 2395.1 \tag{4}$$

Multiplying the transfer function by this value of k yields the desired result, shown in *Figure 17*.

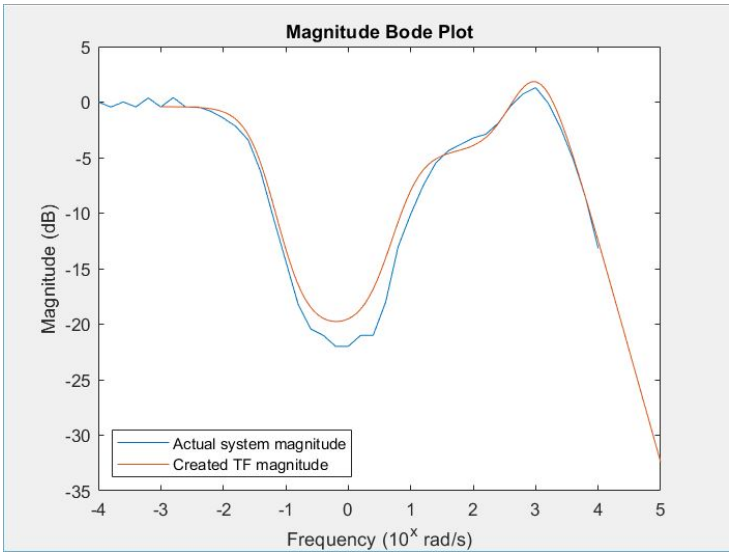


Figure 17: Experimental (blue) and heuristic (orange) magnitude curves with gain constant.

Now that the magnitude curves are aligned, one must turn their attention to the phase curves. Using all negative zeroes and poles, the two phase curves were plotted in *Figure 18*. *Figure 18* shows that the general shape of the two phase curves are similar, but that they are off by 360° at low frequencies. To combat this problem, the sign of the zeroes were changed. In general, negative zeroes have an angle of 0° and positive zeroes have an angle of 180°. Thus, since *Figure 18* showed that the heuristic curve was off by 360° at low frequencies, the sign of of the first two zeroes were changed (i.e change the sign of the two lowest frequency zeroes to positive). The effect of these sign changes is shown in *Figure 19*. *Figure 19* clearly demonstrates that the heuristic transfer function and experimental phase curves have the same shape. Since both the magnitude and phase curves match up, one can approximate the system using the heuristic transfer function, given by:

$$T_P(s) = \frac{2395.1(s+0.1)(s+0.1585)(s+2.512)(s+3.981)(s+251.2)}{(s+0.03981)^2(s+6.31)(s+10)(s+1000)^2} \tag{5}$$

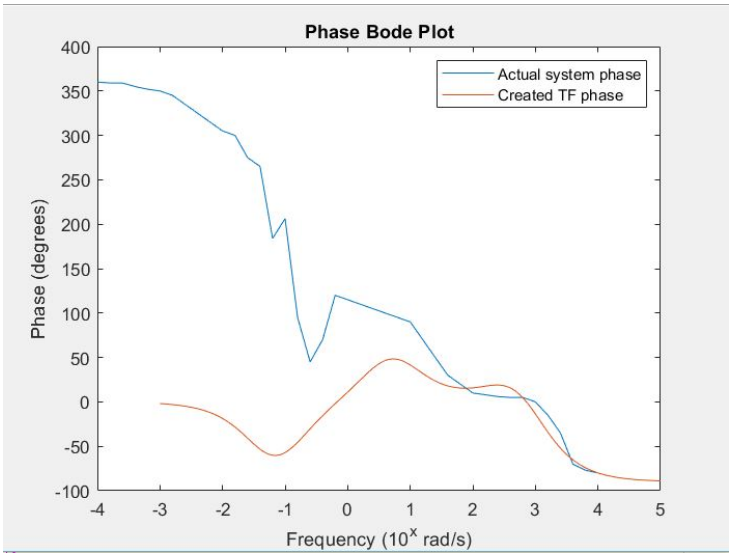


Figure 18: Experimental (blue) and heuristic (orange) phase curves off by 360 degrees at low frequencies.

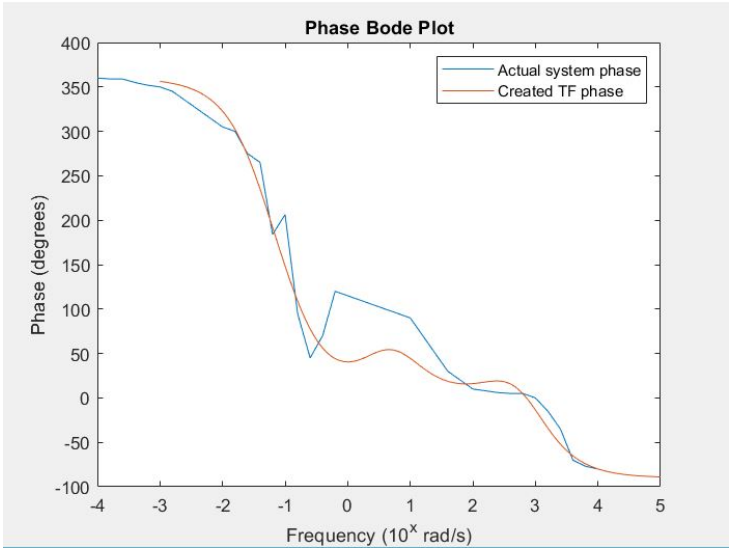


Figure 19: Experimental (blue) and heuristic (orange) phase curves.

5.5 IBIBO stability and validation of the transfer function

Before further validating that (5) models the behaviour of the system, one must first check stability. Indeed, first observe that (5) has no poles in $\mathbb{C}^-$.

Then, the Nyquist plot of (5) is displayed in *Figure 20*.

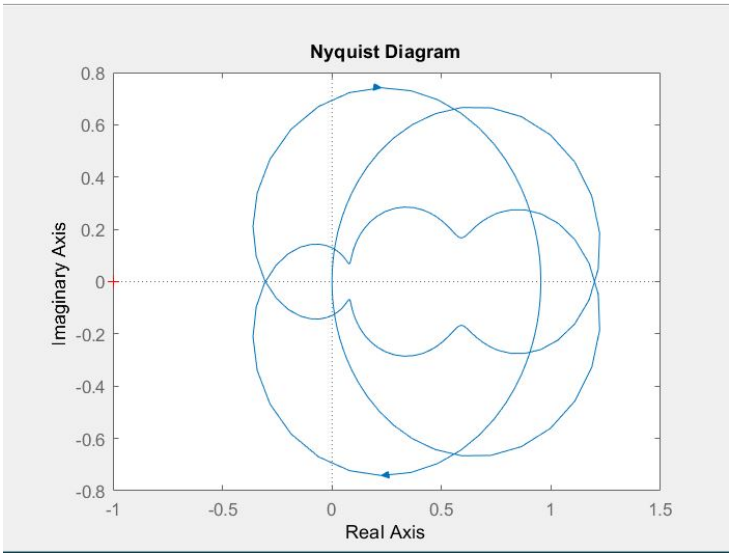


Figure 20: Nyquist plot of heuristic transfer function; no CCW encirclements of -1.

Since there are no poles in $\mathbb{C}^-$ and *Figure 20* shows that there are no CCW encirclements of -1, one can conclude that (5) is interconnected bounded input bounded output (IBIBO) stable.

Now that (5) has been determined to be IBIBO stable, one must then further validate that it appropriately models the system. To do so, the step responses of both the system and the heuristic transfer function are depicted in *Figure 21*.

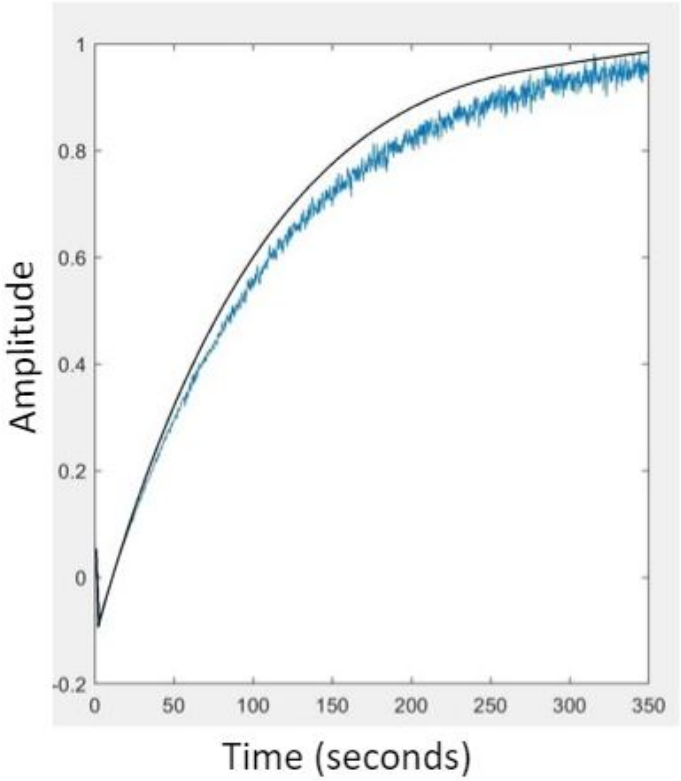


Figure 21: Step responses of the system (blue) and the heuristic transfer function (black).

Figure 21 clearly shows that the step responses are very similar. Indeed, the difference between the two step response is plotted in *Figure 22*.

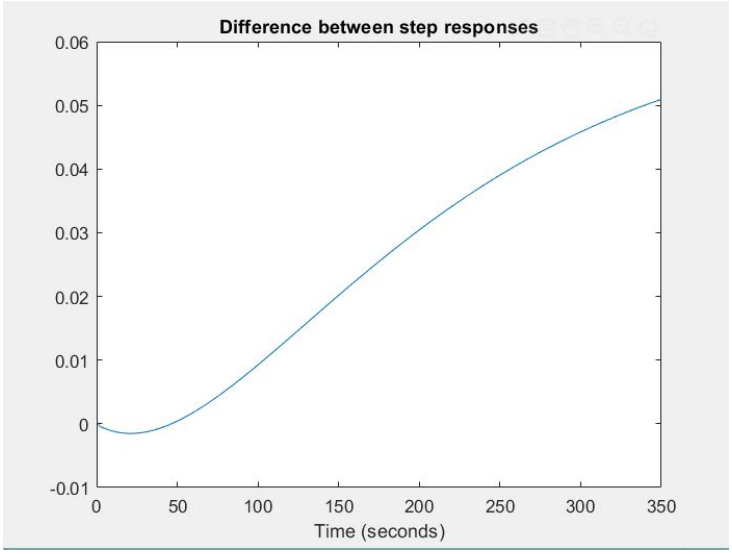


Figure 22: Difference between the two step responses.

Looking at the y-axis of *Figure 22* shows that the error between the step responses does not exceed 6%. In fact, as both the step responses converge to 1, the error will theoretically go back down to 0; hints of this can be seen in *Figure 22* in the form of a gradually decreasing slope.

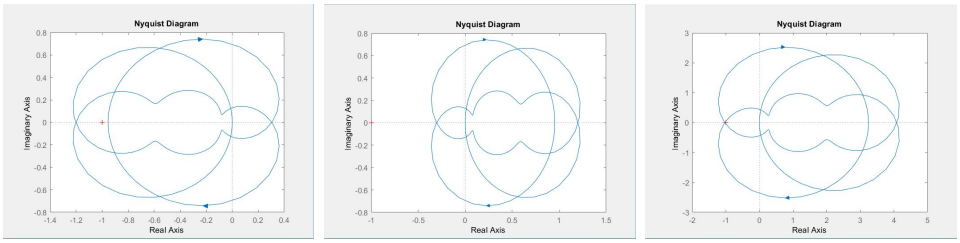
5.6 PID transfer function and restrictions on the PID variables

As is industry standard, a PID controller model was chosen to control the unknown system given by the temperature of the combustion chamber. The transfer function of the parallel PID controller is given by:

$$T_C(s) = P + \frac{I}{s} + \frac{DN}{1 + \frac{N}{s}} = \frac{(P + DN)s^2 + (NP + I)s + NI}{s(s + N)} \tag{6}$$

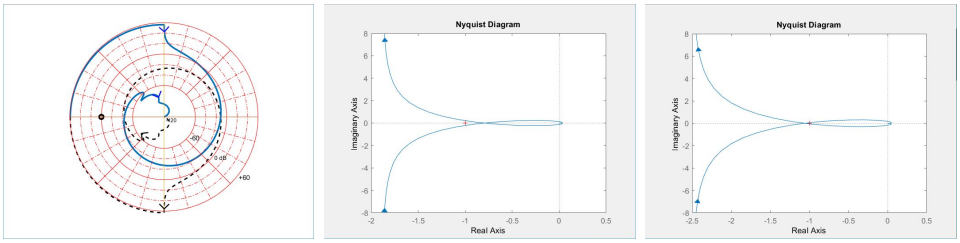
where P is the proportional constant, I is the integral constant, D is the derivative constant, and N is the filter coefficient. Now, define $T_L(s) = T_P(s)T_C(s)$. One then wishes to study the restrictions on P, I, D, and N so as to adhere to the stability of the heuristic system. To get a reasonable estimate of the bounds of P, I, and D, the filter coefficient was fixed at the default value of $N = 100$ and only one variable was changed at a time while

the other two were fixed at 0. Nyquist plots were then created to determine whether or not changing a particular value of a single variable made the system unstable. The Nyquist plots showing the bounds for P, I, and D are displayed in *Figures 23, 24, and 25*, respectively.



(a) (b) (c)

Figure 23: (a) Unstable with $P = -1$, $I = D = 0$; (b) Stable with $P = 1$, $I = D = 0$; (c) Unstable with $P = 3.4$, $I = D = 0$.



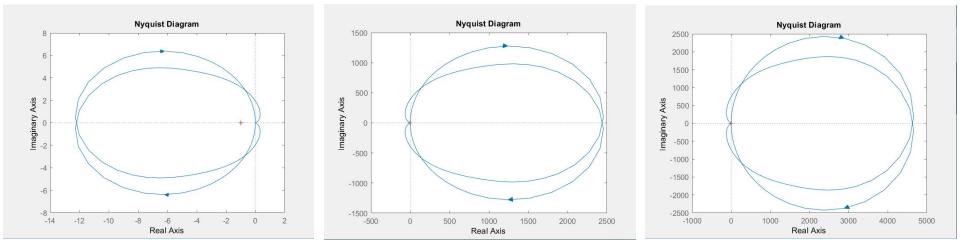
(a) (b) (c)

Figure 24: (a) Unstable with $P = 0$, $I = -0.01$, $D = 0$; (b) Stable with $P = 0$, $I = 0.03$, $D = 0$; (c) Unstable with $P = 0$, $I = 0.04$, $D = 0$.

From the above Nyquist plots, one can deduce the following general bounds:

- $0 < P < 3.4$;
- $0 < I < 0.04$;
- $0 < D < 38$.

It is important to note that these bounds are merely a guideline, since they may change as the values of the other variable change (i.e. when the other values are not 0).



(a) (b) (c)

Figure 25: (a) Unstable with $P = I = 0$, $D = -0.1$; (b) Stable with $P = I = 0$, $D = 20$; (c) Unstable with $P = I = 0$, $D = 38$.

5.7 Determining PID values for heuristic model

Once the restrictions for the PID values were determined, a closed-loop Simulink model was created for the heuristic system, shown in *Figure 26*. To quickly establish accurate values (and as a sanity check) for P, I, and D,

Figure 26: Heuristic closed-loop system.

Matlab's built-in *pidTuner* function was used. This function is a means of automatically fine-tuning the controller variables to optimize a known system in regards to design specifications; namely rise time, settling time, and overshoot. The most important thing to note here is that the PID tuner can only be used for known systems, i.e. systems where one knows the transfer function. Since the heuristic transfer function does not perfectly model the behaviour of the black box system, these computed values are merely guidelines for controlling the actual system. The PID tuner values were determined to be: $P = 1.809$, $I = 0.03408$, and $D = 24.01$, which are all within their respective bounds mentioned above (note that N was still fixed at 100). The controlled heuristic step response can be seen in *Figure 27*.

The design specifications of the system are shown in *Section 5.9*. Let rise time be the time it takes the step response to go from 10% to 90% of its steady value; in this case, the steady value is 1. Then, the rise time of

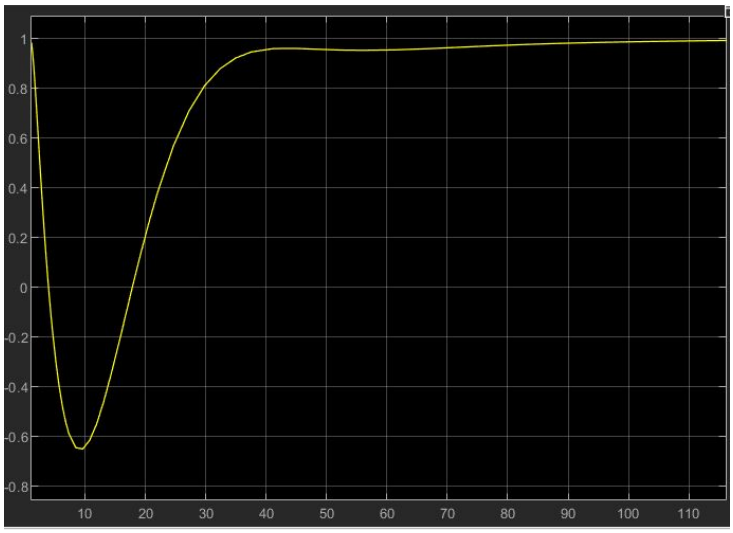


Figure 27: Controlled heuristic step response; time (seconds) on the x-axis, and amplitude on the y-axis.

the controlled step response of *Figure 27* was calculated to be approximately 15.1 seconds (off by less than 1% of the target value), shown in *Figure 28*.

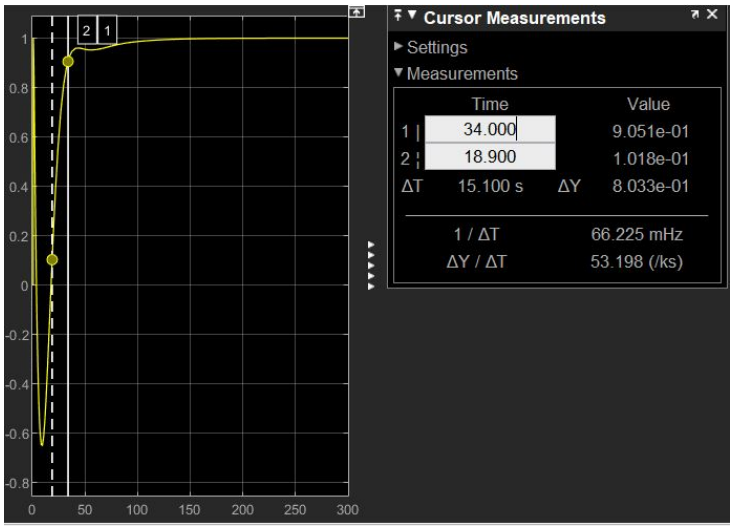


Figure 28: Controlled heuristic step response rise time; time (seconds) on the x-axis, and amplitude on the y-axis.

From both *Figures 27 and 28*, one can clearly see that there is no overshoot present, as the response does not exceed 1, thus meeting another design criteria. Lastly, the settling time was determined to be approximately 35 seconds (off by roughly 230%), shown in *Figure 29*. To validate that the PID

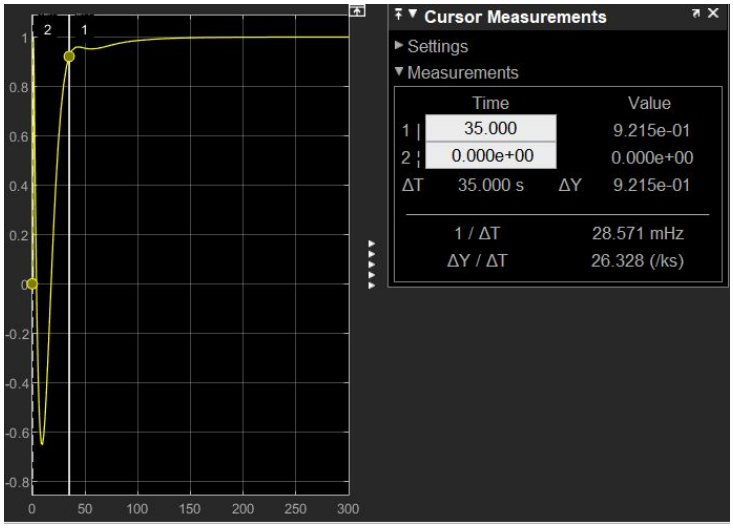


Figure 29: Controlled heuristic step response settling time; time (seconds) on the x-axis, and amplitude on the y-axis.

tuner actually optimized these design specification, the values of P, I, and D were varied systematically within their respective bounds. Unsurprisingly, altering any of these three values worsened the results.

5.8 Determing PID values for black box system

Before diving into the PID values for the black box system, one must first determine a means to real-time filter the noise of the system. A real-time filter is necessary since one theoretically does not have access to future values of the output, thus eliminating the use of any averaging functions/filters. Since the noise of the system was determined to be proportional white noise and since the step response of the system would ideally have a low frequency, it makes perfect sense to use a lowpass filter in this situation. Lowpass filters (also known as high cutoff filters) allow signals with a lower frequency than the cutoff frequency to pass through, and attenuate signals with a higher frequency than the cutoff. There are two main variables of a lowpass filter:

the cutoff frequency, and the filter order. The filter order is essentially the rate at which high frequencies are attenuated; a filter order too low may not filter the noise fast enough, and a filter order that is too high may cutoff information of the actual signal.

With all of this background in mind, the closed-loop Simulink model was created, shown in *Figure 30*.

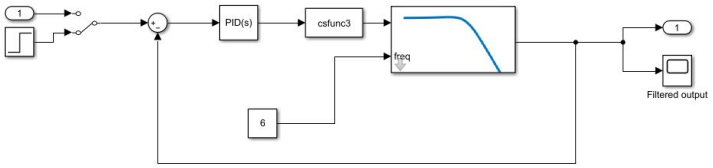


Figure 30: Closed-loop system.

By simply trial and error, it was determined that a cutoff frequency of 6 and a filter order of 2 is optimal for the system. The team then decided to test whether or not the above PID values work for the actual system. The resulting effect can be seen in *Figure 31*.

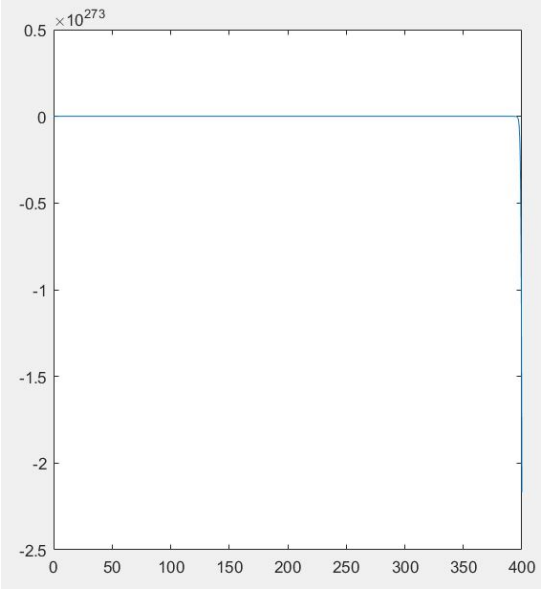


Figure 31: Closed-loop system response with $P = 1.809$, $I = 0.03408$, and $D = 24.01$; time (seconds) on the x-axis, and amplitude on the y-axis.

It is extremely evident upon observation of *Figure 31* that the system is nowhere close to being controlled. This large discrepancy is due to the fact that those PID variable values were specifically optimized for the heuristic transfer function, which is not a perfect representation of the actual system. Indeed, even the previously mentioned bounds on the variables do not hold. This was a notable setback as it essentially forced the team to rely on intuition of variable values from previous MTHE 332 labs. Using said intuition/experience, the team came up with some very general bounds on the variables:

- $1 < P < 10$;
- $0 < I < 1$;
- $0 < D < 1$.

After iterating through various combinations of different values, and using some human trial-and-error, the following PID values were determined: $P = 7.85$, $I = 0.1$, and $D = 0.3$. Once these values were chosen (by seeing their influences on the step response), the team began tampering with the filter coefficient variable, N; a value of $N = 25$ was decided. The controlled step response of the system can be seen in *Figure 32*.

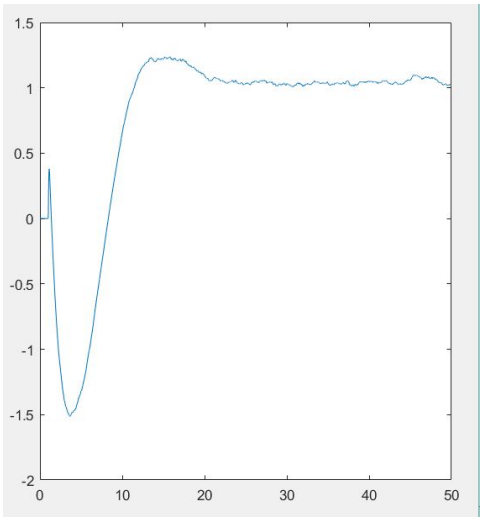


Figure 32: Controlled closed-loop system response with $P = 7.85$, $I = 0.1$, $D = 0.3$, and $N = 25$; time (s) on the x-axis, and amplitude on the y-axis.

5.9 Conclusions and evaluation

From the black box analysis that was conducted, a controller was developed that was able to control the system. The controlled system's targets were:

- Overshoot: 1.4
- Rise Time: 10
- Settling Time (s): 15
- Settling Time (ϵ): 0.08
- Steady State Error: $\pm$ 0.08

The final controlled system was able to achieve:

- Overshoot: 1.21
- Rise Time: 10.42
- Settling Time (s): 23
- Steady State Error: $\pm$ 0.08

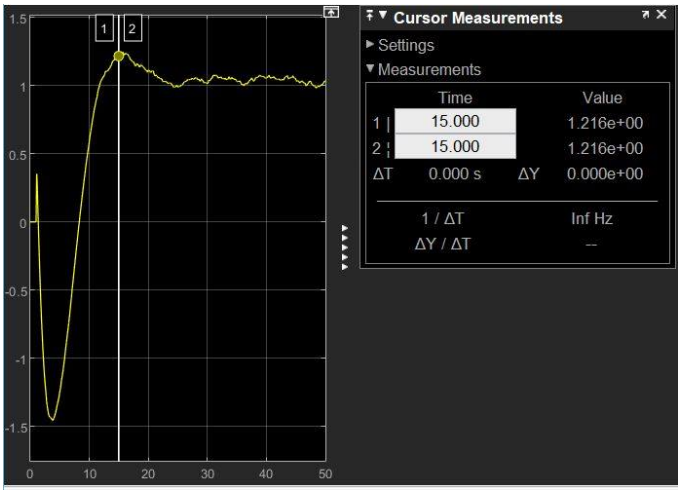


Figure 33: The system's overshoot; time (seconds) on the x-axis, and amplitude on the y-axis.

Thus, the controlled system was able to meet much of the target metrics. The system's overshoot was 1.21, *Figure 33*, which was below the target of 1.4. This means that the controller was able to stop the system, in this case the combustion chamber of the rocket, from overshooting the target temperature. The system's steady state error, shown in *Figure 35*, was within the target of ± 0.08. This means that the controller was able to ensure that the temperature of the combustion chamber was within an acceptable bound of the target temperature once the system had reached steady state.

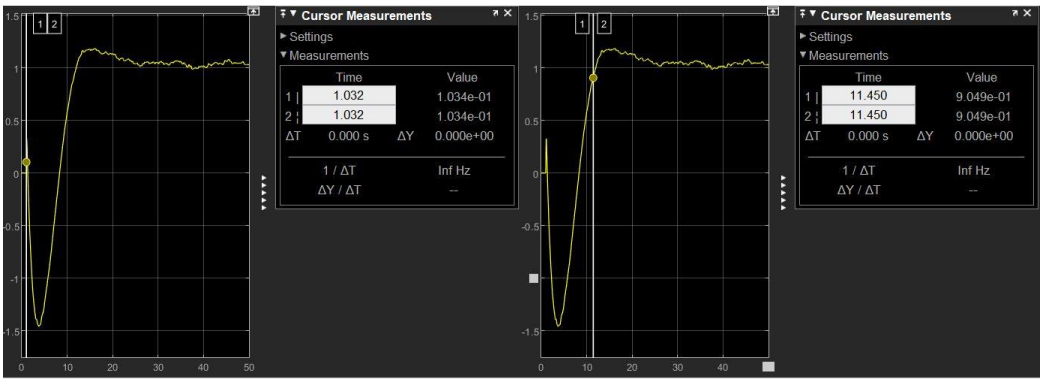


Figure 34: The system response at 10% (left) and 90% (right) of target value; time (seconds) on the x-axis, and amplitude on the y-axis.

However, the settling time and rise time of the system did not meet their respective target values. While the rise time, defined above in *Section 5.7* and depicted in *Figure 34*, could be improved, it would be at the cost of a longer settling time and overshoot. In terms of the application, a larger rise time would be preferable, as this would mean that the temperature of the chamber takes longer to heat up. The other target that the system was not able to meet was settling time, depicted in *Figure 35*. The system took 8 seconds longer to reach steady state than what was desired; this would mean that the controller would take 8 seconds longer to reach an acceptable temperature inside the combustion chamber. This could lead to failure of the engine over time, which would be fatal to the rocket.

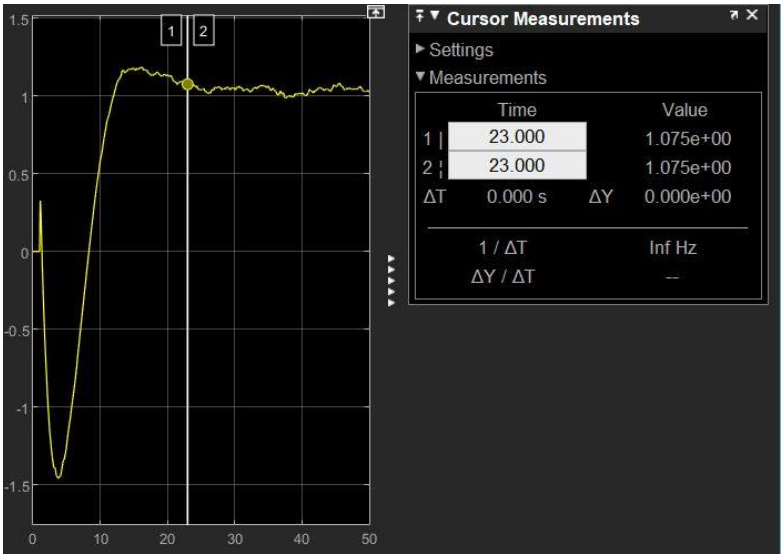


Figure 35: The system response once steady state has been achieved; time (seconds) on the x-axis, and amplitude on the y-axis.

6 Appendix A: TBL Stakeholder Chart

7 References