

ACDB5011 – Instrumentation and Control

Control System Design

Using mathematical modelling software to determine, simulate and validate a closed loop transfer function for a CIWS control system from a base process transfer function.

Shaun Malone (2030723)
Autumn 2025

Contents

Abstract.....	2
Introduction.....	2
Phalanx CIWS.....	2
Control Systems and Mathematical Modelling	3
Introduction	3
Open v Closed Loop Systems	3
Response and Stability	4
System Set-Up	6
Objective	6
Tuning the PID Controller	7
Initial Simulation.....	9
Determining the Whole-System Transfer Function.....	10
Results and Discussion	11
Validation of Whole-System Transfer Function	11
Control System Stability Analysis	12
Pole-Zero and Root Locus Analysis	12
Nyquist Plot Analysis	15
Conclusions	16
Appendices	17
A. Derivation of Whole-System Transfer Function	17
References	20

Abstract

2

This report will demonstrate the versatility and potential of mathematical modelling to design, test and evaluate control systems accurately, that would be costly to build and test incorrectly in a physical trial. A closed-loop control system was designed and created using MATLAB and Simulink to control the transient and steady-state responses of the angular positioning system of a Phalanx Close-In Weapon System (CIWS), required by brief to be at the correct angle within two seconds, without overshooting its target by more than 5% during the initial feedback loop. A whole-system transfer function was required and needed to be validated against the initial design, and the final design had to be sufficiently analysed for stability and compliance. A plant transfer function for the system, $G(s)$, was provided. The investigation was successfully completed, with a full closed-loop control system designed to fit the requirements of response time (0.647 seconds) and overshoot (3.24%). A whole-system transfer function was derived mathematically and validated in MATLAB and then, using Pole-Zero analysis and Nyquist plots, the stability of the system was thoroughly analysed – and found to be stable. The investigation verified the benefits of using mathematical models to test and evaluate control systems without the need for costly builds or trials.

Introduction

Phalanx CIWS

The Phalanx Close-In Weapon System (CIWS) is a multifunctional, computer-controlled, radar-guided gun and turret system that is designed to use radar to detect and eliminate incoming threats to naval vessels. It has the capability to carry out functions that would ordinarily be carried out by several systems [1].

Phalanx uses a combination of both a ‘pulse-Doppler’ radar, and a network of electro-optical sensors to scan its surroundings and seek out threats, the complexity of its tracking system allowing it to distinguish between genuine threats and noise [2].

Once a threat is detected, the positioning control system kicks in to action, with the ability to track fast moving airborne objects such as enemy anti-naval missiles as well as enemy vessels on the surface of the water.

With a unit cost of an estimated \$12m [3], the Phalanx is not cheap. There is shrinking appetite for physical test and evaluation of the control systems that drive such advanced weapon systems – where the cost of mistakes can be significant.

Control Systems and Mathematical Modelling

Introduction

A control system is defined as something that ‘*consists of subsystems and processes (or plants) assembled for the purpose of obtaining a desired output with desired performance, given a specified input*’ [4]. They exist everywhere, both in man-made engineering systems, such as the central heating systems that heat family homes, and in nature, where the human brain runs a vast network of control systems to regulate the body’s processes and keep the body healthy.

In the case of Phalanx, control systems allow the rapid movement of an extremely heavy motor-powered weapon platform for the purposes of precise and accurate tracking of threats, something that humans could never achieve alone to the same level of precision and accuracy.

Open v Closed Loop Systems

Typically, a control system can either be closed-loop or open-loop. An open-loop system is simpler – whereby an input signal is passed on to a controller via a transducer, and the controller in turn drives a process, which produces an output. Figure 1(a) shows how this would look as a block diagram.

The process is simple, but open to interference from disturbances. With no associated feedback process, any noise would be added to the process, and the output would not match the input [5]. The typical impacts of disturbance are zero drift, and sensitivity drift. Zero drift, sometimes called bias, is where disturbance changes the zero reading of a system. This produces a ‘constant error that exists over the full range of measurements’ [6]. Sensitivity drift refers to the ‘amount by which an instrument’s sensitivity varies as conditions change’ [6].

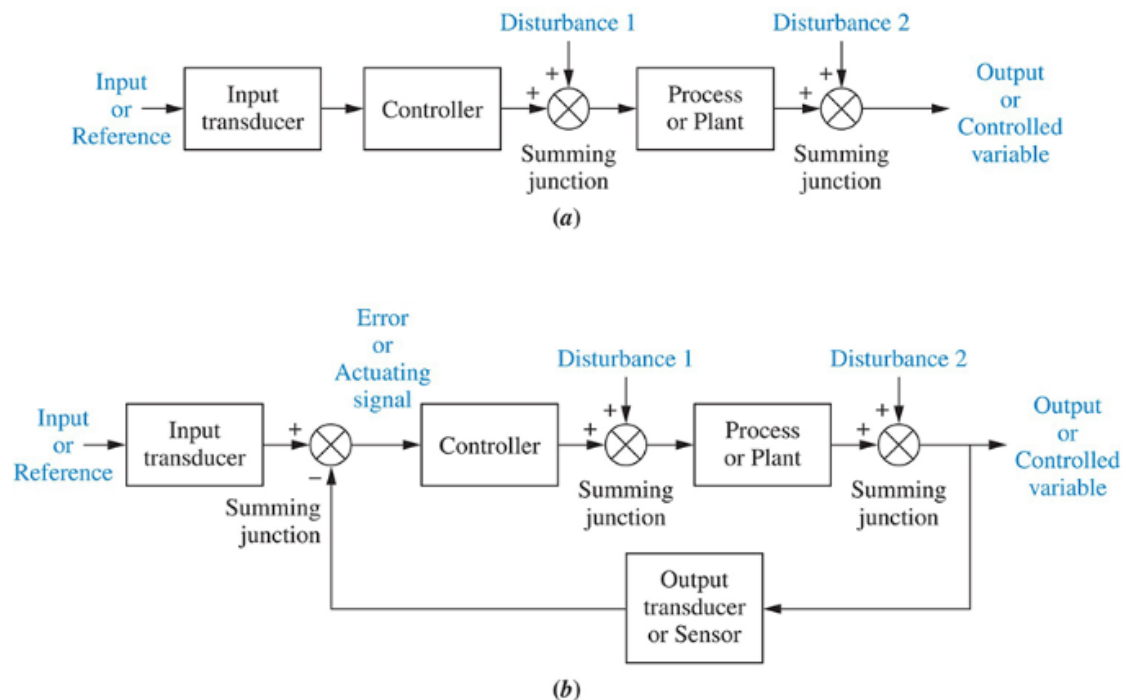


Figure 1: Image showing a typical open-loop (a) and closed-loop (b) control system. This is labelled as Figure 1.5 in [5].

Closed-loop systems deal with this issue with the addition of a feedback loop. This time, the signal at the point of output is sent back in to the initial summing junction, which calculates the difference between the input and output signal to produce an error signal, as shown in Figure 1(b). If the error signal value is not zero, process is repeated until it is, meaning that the output signal matches the input signal. This means that any noise and interference can be overcome as there is a system in place to check the alignment of input and output.

Engineers will often have to consider the trade-off between the need for a control system that is simple, and one that guarantees accuracy. In the case of Phalanx, accuracy outweighs simplicity, and so the positioning system will use a closed-loop system.

Response and Stability

System Response

When designing a control system, two types of response must be considered – transient and steady-state.

A system's transient response is defined as the immediate response of a system following receipt of an input [7]. In the case of the Phalanx, the transient response stage is the period in which the positioning system begins moving towards the required bearing following threat detection. The steady-state response comes after this, when the system settles [7], where the final output should match the desired input.

It is important when analysing a system's response, that both the transient response and steady-state responses meet the requirements. If the system only satisfies one, for example repositioning quickly but settling at the wrong bearing, or having good accuracy but taking minutes to reach the output state, then the system as a whole becomes redundant.

5

A system's response, based on the inputs and controls applied to it, can be undamped, critically damped, overdamped or underdamped. Damping refers to the amplitude limiting of a wave as it progresses. Figure 2 shows this visually.

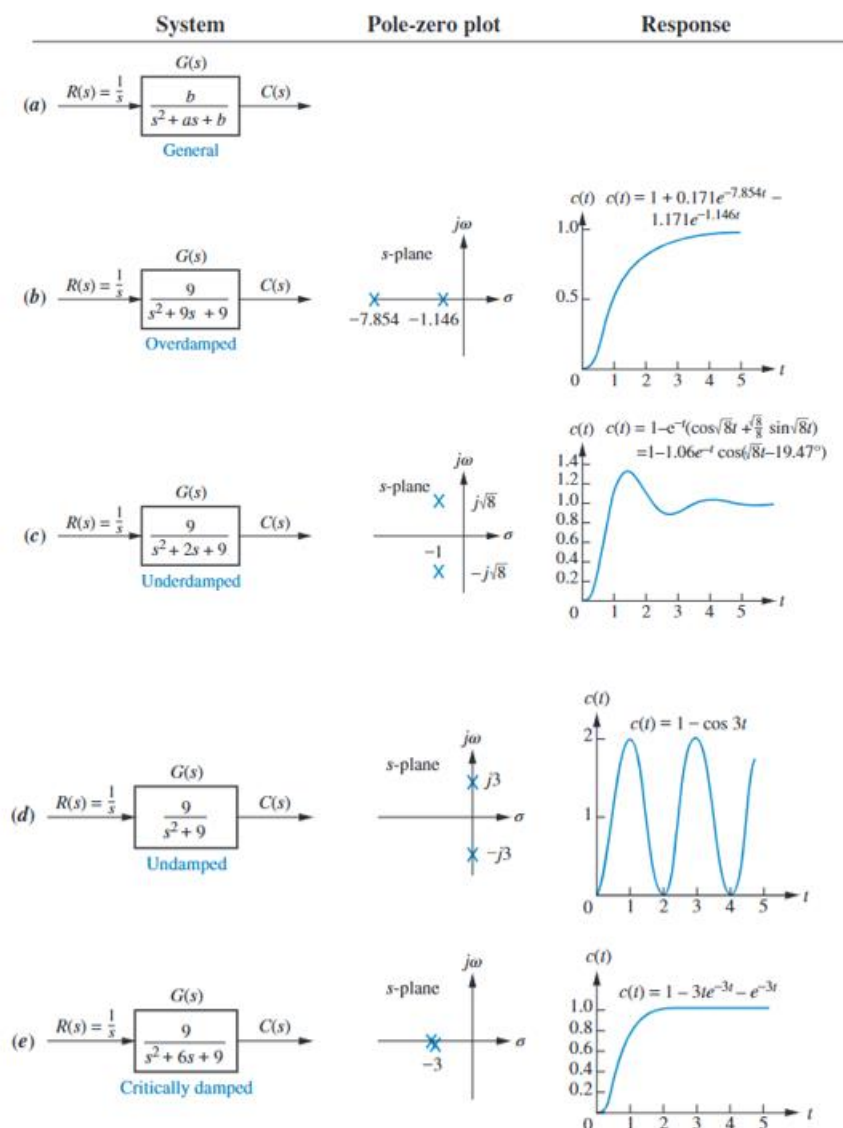


Figure 2: Various levels of damping during system response [8].

Stability

Stability is another key component of any system. For most systems, the total response of a system is a combination of its transient and steady-state responses. The system will be designed so that it receives an input, moves towards the required final state, before

the energy of the system ultimately dissipates to zero as it reaches the desired steady-state [9]. These can be termed as a forced and natural response, i.e. the response as a direct result of the input, and the system's natural tendency based on its innate properties. Stability analysis is important in any control system model because if the right conditions are met, the natural response can grow exponentially in ever-increasing oscillations rather than diminishing to zero or remaining steady. If this continues, the natural response can grow so large that the system is no longer under control [9]. Instability causes ill-response and, at its extreme, physical damage to the system and its components, which can be incredibly costly to repair. This is why advance mathematical modelling of systems plays a crucial role in modern system design.

System Set-Up

Objective

The objective of the control system in this report was to govern the positioning system of a Phalanx CIWS turret following detection of a threat. Using a combination of MATLAB and Simulink, the goal was to design a closed-loop control system that governed the motion of the turret, in-line with specific requirements set out by the brief. The key requirements of the brief were:

- The system must have a full response time of under two seconds.
- The initial overshoot of the system must not be more than 5% before correction.
- The system must be verifiably stable.

The base design for the system is shown in Figure 3. It shows an input step signal connected to an error signal, which feeds the difference signal in to a PID controller, which drives the process given by the process transfer function shown in the plant (in this case the CIWS turret positioning motor) box – and finally feeding in to an output.

As a closed-loop system has been used, there is a feedback loop which returns part of the output signal to the actuator, which determines whether or not the desired output has been achieved, and runs the process again if it has not, until a satisfactory output is provided.

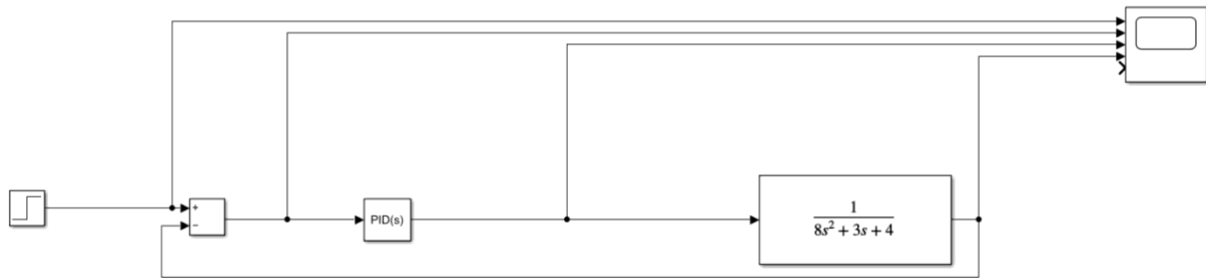


Figure 3: Initial control system design including a step signal, error signal, PID controller, process transfer function and output - with feedback loop.

For the purpose of this report, the initial plant transfer function was given. It took the form $\frac{1}{ax^2+bx+c}$, where a, b and c were given as 8, 3 and 4 – derived from a UWTSD student number. This function was the basis for finding the whole system transfer function later-on. In a real-life CIWS system, this function would be based on a number of parameters, such as the weight of the system, the size of the motor required and the energy transfers in the system among other hardware-related decisions.

At each stage, for the purpose of system analysis, separate branches at each step were connected up to the output scope so the system could be analysed continuously to fault-find if any errors occurred.

Tuning the PID Controller

PID (proportional-integral-derivative) controllers are a comprehensive means of controlling a system's response to deliver the most desirable output. They are a combination of three different control mechanisms: proportional control, integral control and derivative control. Figure 4 shows this combination and how it drives the process mathematically.

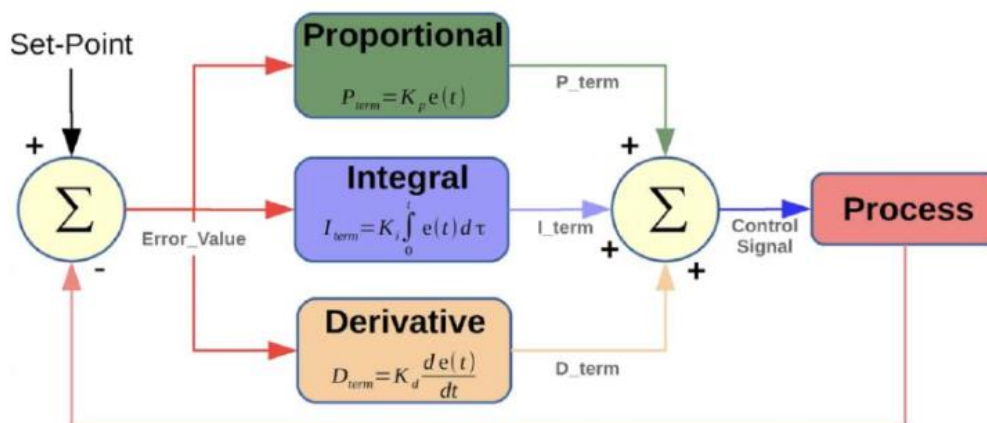


Figure 4: Image showing components of a PID controller [10].

Each part plays a key role in regulating system response. The proportional component is responsible for controlling the current error in the system. The integral component is

responsible for controlling previous error, and therefore decreasing response time and increasing the speed of convergence on the final point, and the derivative component acts to control future possible error, and therefore controls system stability. [10]

Using MATLAB and Simulink's tuning wizard within PID controllers, values for K_p , K_i and K_d were found. The tuning wizard automatically showed likely system response when values were placed in to the boxes, allowing for the gathering of values that met the system's requirements. Figure 5 shows the values for P, I, D and N, and figure 6 shows the controller's system analysis based on the given values.

Controller parameters

Source: internal

Proportional (P): 141.3575

Integral (I): 16.4627 ☐ Use I*Ts (optimal for codegen)

Derivative (D): 119.0622 ☐ Use externally sourced derivative

Filter coefficient (N): 1079.5407 ☒ Use filtered derivative

Automated tuning

Select tuning method: Transfer Function Based (PID Tuner App)

☒ Enable zero-crossing detection

Figure 5: PID controller tuning wizard showing values for P, I, D and N that meet the system requirements.

	Block	
P	141.3575	▲
I	16.4627	
D	119.0622	
N	1079.5407	
		▼

Performance and Robustness		
	Block	
Rise time	0.128 seconds	▲
Settling time	0.647 seconds	
Overshoot	3.24 %	
Peak	1.03	
Gain margin	Inf dB @ Inf rad/s	
Phase margin	86.1 deg @ 15 rad/s	
Closed-loop stability	Stable	▼

Figure 6: Tuning wizard analysis of the system given the found values of P, I, D and N.

The system analysis from the PID tuning wizard shown in figure 5 shows that the response time to reach steady state is 0.647 seconds, and that the initial overshoot is 3.24% and, crucially, that the closed-loop system is stable.

Initial Simulation

The output scope was set up to run a simulation of the system for ten seconds, and figure 7 shows the output visually.

9

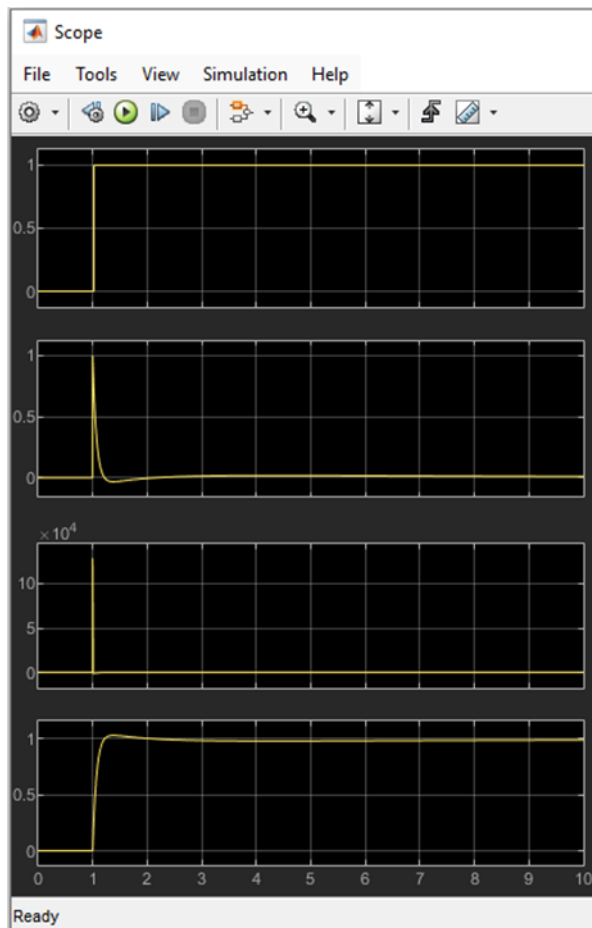


Figure 7: Scope analysis showing signals at each stage of the system. From top to bottom, 1) step signal, 2) error signal, 3) PID signal, 4) output signal.

Each window represents a scope of the signal at each component in the system. The first window shows the step signal with the input signal of 1, the second shows the error signal, initially 1 and decreasing toward zero, the third shows the PID controller with an initial spike and then falling to close to zero, and the fourth showing the output signal, initially overshooting and then correcting to converge on 1 within a short period of time.

This simulation analysis is the basis on which future system analysis will be compared against.

Determining the Whole-System Transfer Function

The next step of the analysis process is the determination of a whole-system transfer function. This is a complex, multi-step process involving multi-order polynomials.

The first step is to take the process transfer function, reshown in figure 8, and use it to find the forward path transfer function by multiplying it with the transfer function that governs the PID controller.

10

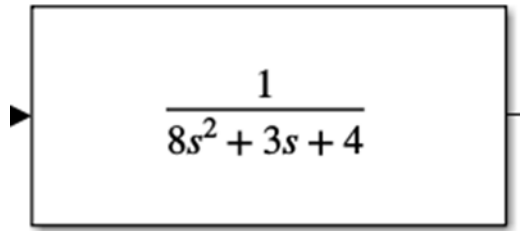


Figure 8: Process transfer function for CIWS turret.

The PID controller transfer function is given as:

$$PID\ TF = P + I\left(\frac{1}{s}\right) + D\frac{s}{N^{1/2} + 1}$$

Equation 1: Transfer function for a PID controller [11].

Multiplying the two gives:

$$G = \frac{(DN + P)s^2 + Is + (P + I)N}{8s^4 + (3 + 8N)s^3 + (4 + 3N)s^2 + 4Ns}$$

Equation 2: Forward path transfer function (G) from multiplication of Figure 7 and Equation 1.

The whole-system transfer function can be found using the formula in equation 3.

$$System\ TF = \frac{G}{1 + GH}$$

Equation 3: Formula for the whole system transfer function [12].

In this system, $H = 1$.

Therefore, substituting G in to the numerator and denominator, expanding, cancelling and simplifying, and then substituting in values for P, I, D and N, gives the eighth order polynomial governing the entire control system from start to finish, calculated by hand, verified in MATLAB and shown in figure 9. A full mathematical derivation of the function is shown in Appendix A.

TF =

$$\frac{1.03e06 s^6 + 1.114e09 s^5 + 1.737e09 s^4 + 1.205e09 s^3 + 7.172e08 s^2 + 7.68e07 s}{64 s^8 + 1.383e05 s^7 + 7.577e07 s^6 + 1.17e09 s^5 + 1.822e09 s^4 + 1.233e09 s^3 + 7.358e08 s^2 + 7.68e07 s}$$

11

Figure 9: Eighth order polynomial transfer function for the entire control system.

Results and Discussion

Validation of Whole-System Transfer Function

To validate the whole-system transfer function as correct within the control system, a new box was added to the Simulink system diagram, labelled $\frac{Y(s)}{U(s)}$, which represents the system transfer function. This branch bypassed all other boxes in the system, going straight from the step signal to the output. When visualised in the scope, the function would be validated as correct if the new scope window matched the output window (window four) from the original simulations.

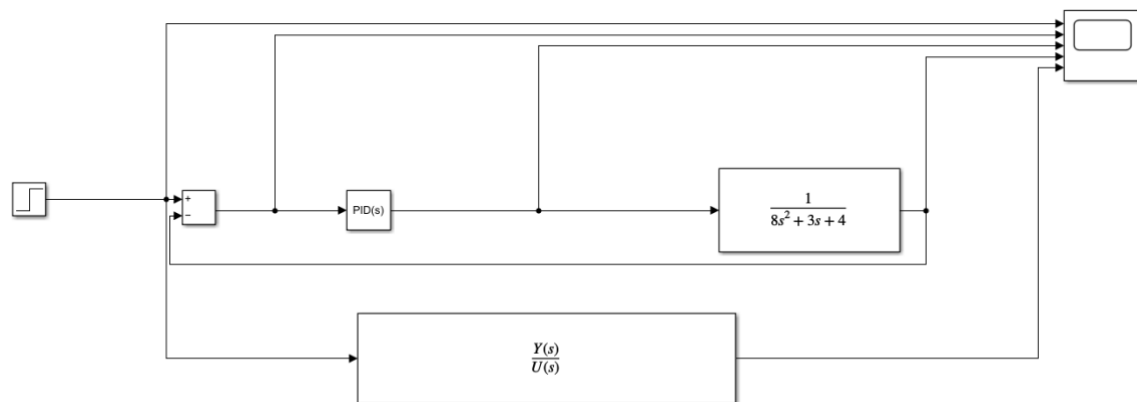


Figure 10: Simulink block diagram of control system showing new branch with whole-system transfer function.

Figure 10 shows the new system box diagram, and figure 11 shows the scope view following a new ten-second run of the system.

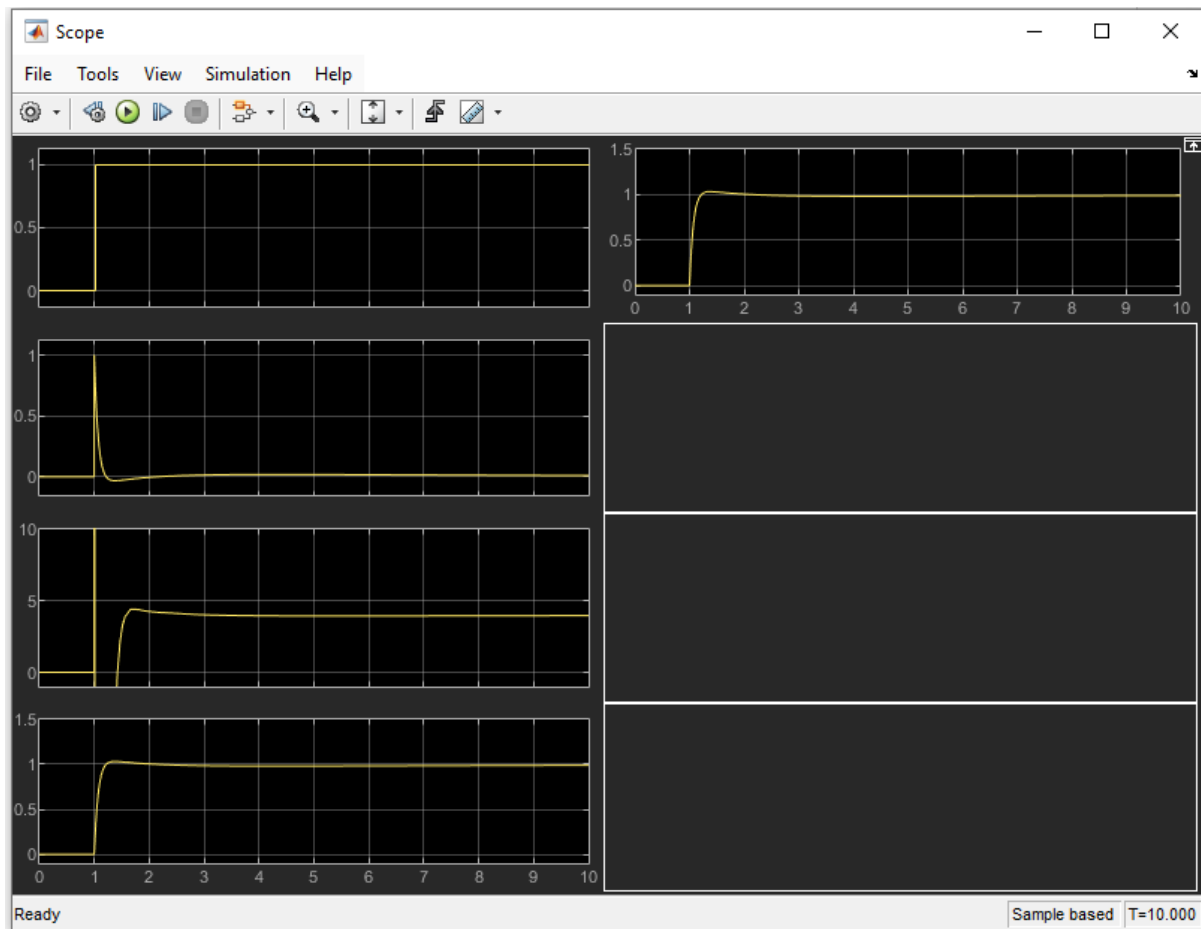


Figure 11: Scope following new simulation, with the new fifth window showing the output of the path with only the system transfer function in it.

As can be seen in figure 10, there is a perfect match between the output wave shown in the fourth window, and the output wave shown in the fifth window, where the only box along the branch is of the whole system transfer function. This verifies the derived function's accuracy, as it is doing the job of all three control components in the original box diagram and producing the same result.

Control System Stability Analysis

Pole-Zero and Root Locus Analysis

Pole-Zero plots play a key role in system analysis, as they determine all of the points in the system where the transfer function tends to zero, or to infinity. In a pole-zero plot, the poles are defined as the values of the transform variable, s , that cause the transfer function to become infinite, and the zeroes are the values of the transform variable, s , that cause the transfer function to become zero [13].

Figure 12 shows the pole-zero plots of the process function, the PID controller, the forward path function (G), and the whole-system transfer function.

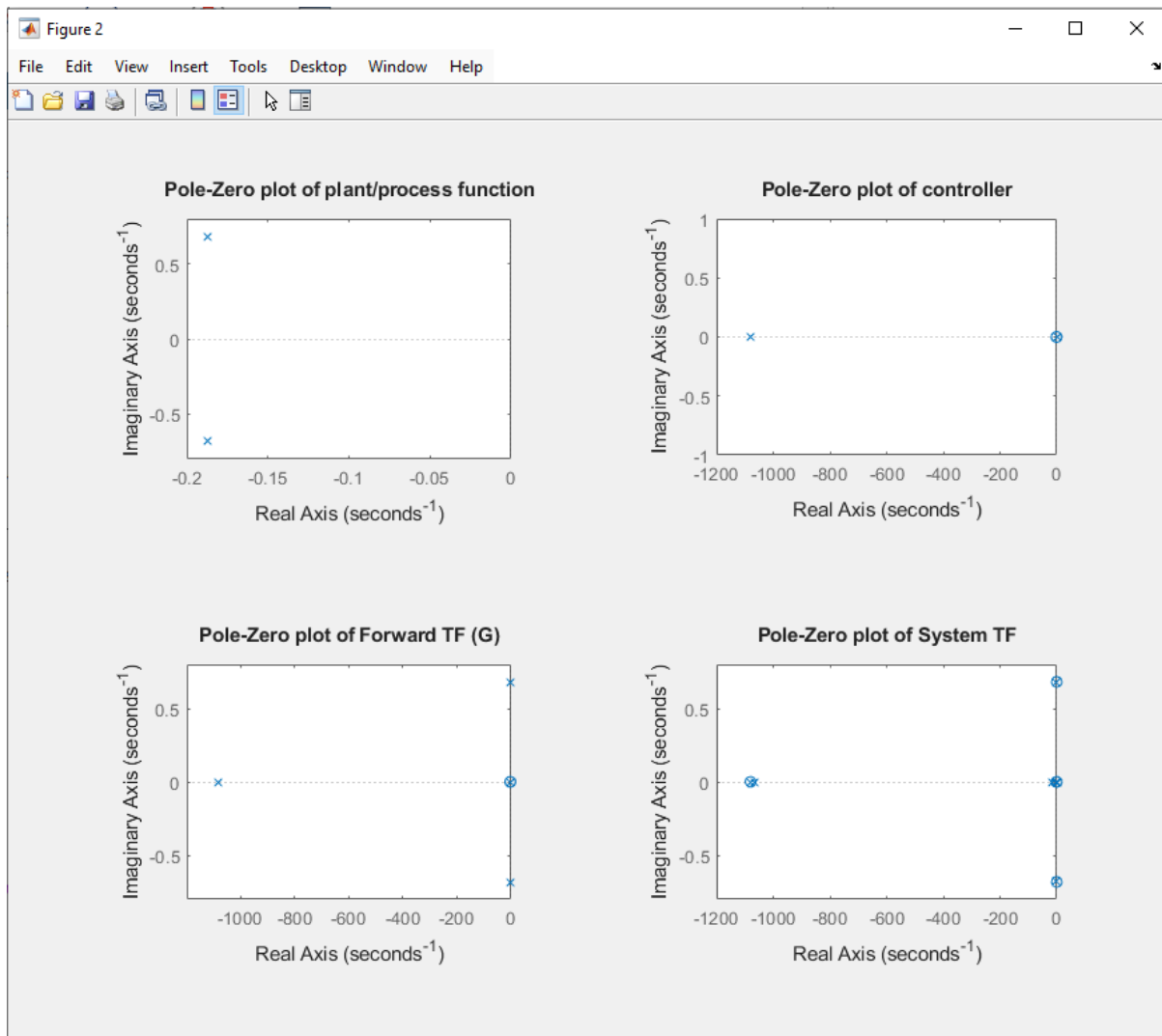


Figure 12: Pole-Zero plots of various parts of the CIWS control system.

Figure 12 shows the relationship between all four parts of the system. As the forward path function (G) is found by multiplying the PID function with the process function, the pole-zero plot for G shows a combination of the poles and zeros found in the PID and process plots. In extension, the system transfer function plot visually represents the relationship between it and G as shown in equation 3.

The pole-zero plot analysis of the system transfer function shows the values of poles and zeros as shown by MATLAB in figure 12. The numerator function produces a list of all zero locations in the system, and the denominator function produces a list of all pole locations in the system.

All poles and zeros lie on the left hand (negative real) side of the plot, which indicates that the system is stable, and that oscillations will decrease over time in a damped sinusoidal wave form. The poles that lie directly on the real axis are overdamped, and the single complex conjugate pair of poles, where one lies on the positive y-axis and one on the negative y-axis, demonstrates an underdamped response [8], as shown in Figure 2.

```
>> roots(cell2mat(TF.numerator))

ans =

    1.0e+03 *

    0.0000 + 0.0000i
   -1.0799 + 0.0000i
   -0.0011 + 0.0000i
   -0.0002 + 0.0007i
   -0.0002 - 0.0007i
   -0.0001 + 0.0000i

>> roots(cell2mat(TF.denominator))

ans =

    1.0e+03 *

    0.0000 + 0.0000i
   -1.0799 + 0.0000i
   -1.0648 + 0.0000i
   -0.0142 + 0.0000i
   -0.0012 + 0.0000i
   -0.0002 + 0.0007i
   -0.0002 - 0.0007i
   -0.0001 + 0.0000i
```

Figure 13: List of all poles and zeros in the system transfer function.

Root Locus plots show a representation of the entire possible set of poles in a system across a varied range of parameters [14].

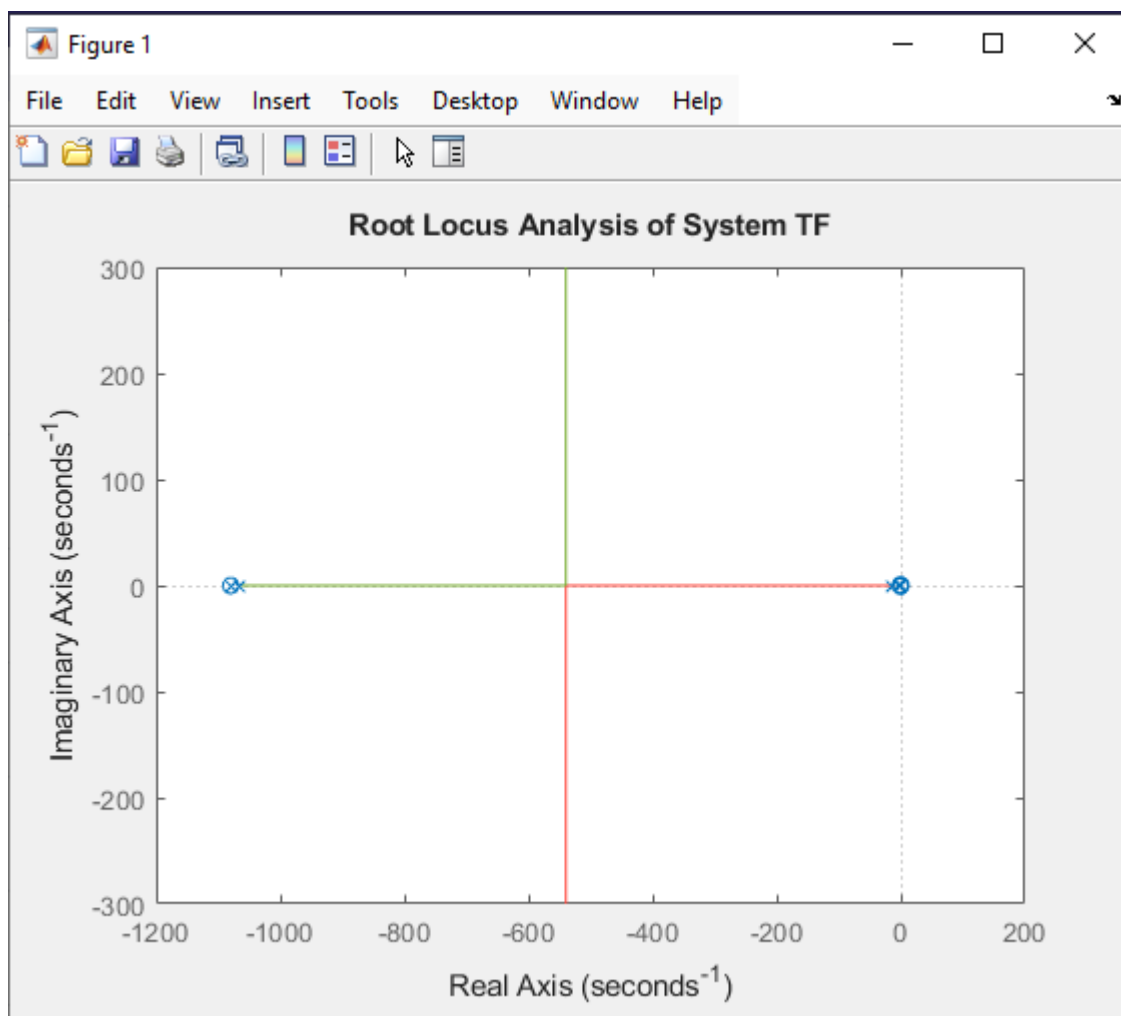


Figure 14: Root Locus plot of CIWS system.

Figure 14 shows the Root Locus plot of the system. Verifying the information taken from the pole-zero analysis, it shows that, whilst the system can move between and

overdamped and underdamped response when parameters change, there is no change in gain that cause the system to become unstable, as all poles are found on the left side of the complex frequency axis.

The system demonstrates robust stability in the time-domain response.

Nyquist Plot Analysis

The final stability analysis tool used in this report is a Nyquist plot. Nyquist plots are a form of frequency domain analysis over which stability can be determined. It is created by mapping all real values of frequency and connecting them.

The key point for determining the stability of a system using a Nyquist plot is to determine where the plot intersects with the negative real axis, and the number of encirclements of the point -1 on the negative real axis [15].

Nyquist plots can be used to determine the gain and phase margins of the system, i.e. what changes to gain and phase are required to bring about instability in the system?

Gain margin is defined as the change in gain required at 180° phase shift to cause instability in the system, and phase margin is defined as the phase shift required at unity gain to cause instability in the system [15].

Figure 15 shows the means by which these are calculated. The gain margin is determined from the distance within the unit circle between -1 and the point where the plot crosses the negative real axis. The phase margin is the angle, anti-clockwise from the negative real axis, representing the phase shift required to cause instability.

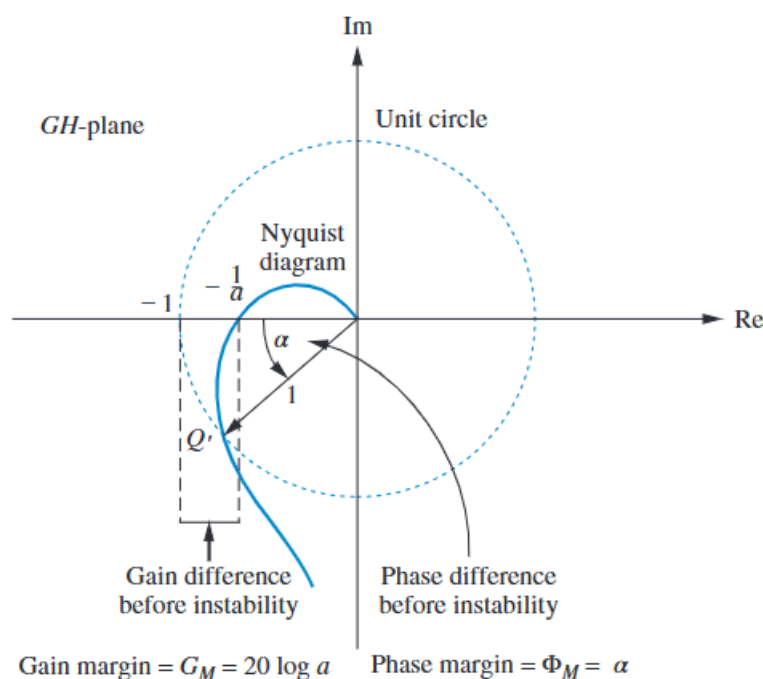


Figure 15: Nyquist plot showing how phase and gain margin are calculated using a unit circle [15].

Figure 16 shows the CIWS control system Nyquist plot.

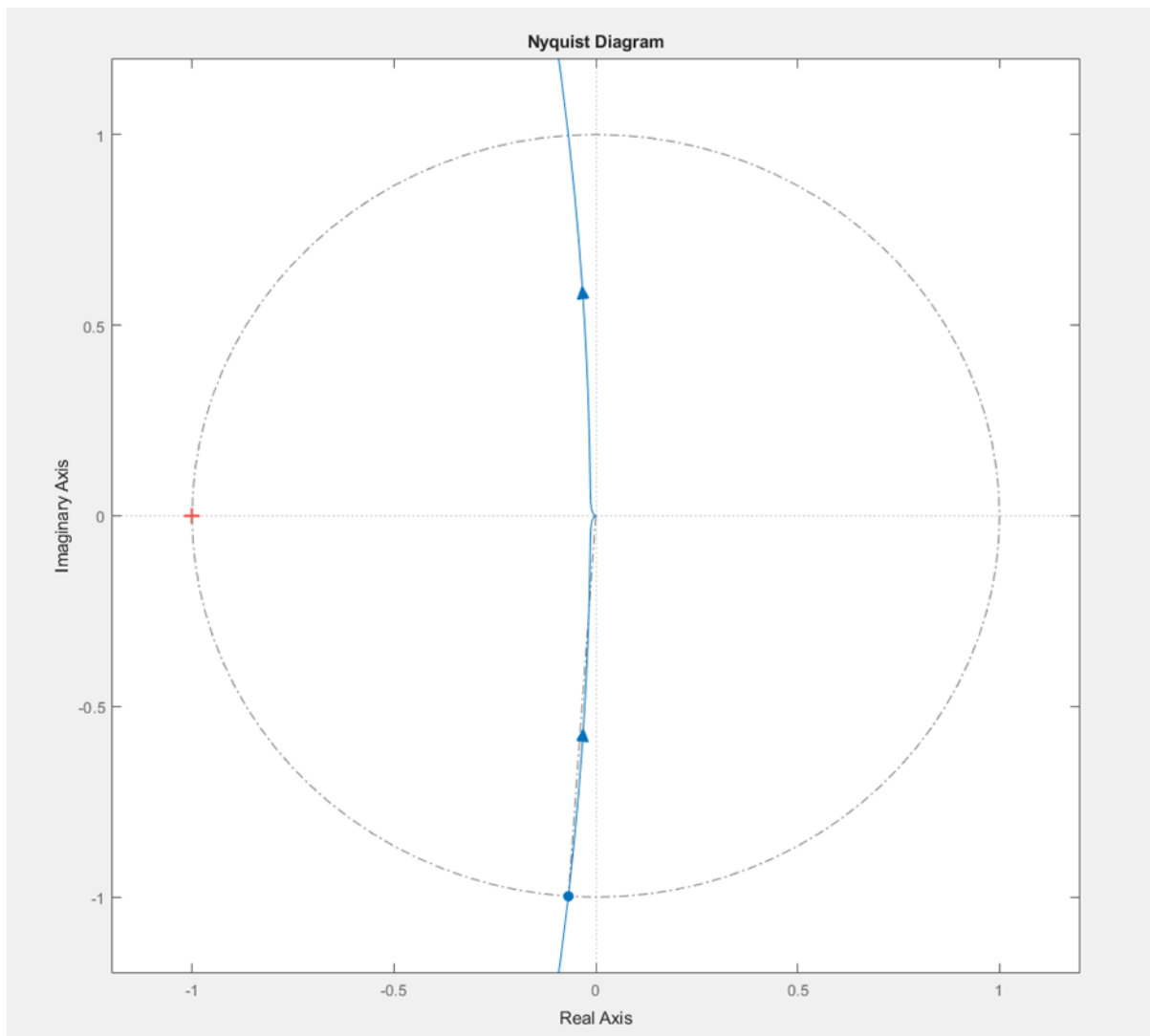


Figure 16: Nyquist plot of Phalanx CIWS control system.

From this plot, it can be seen that the plot does not cross the negative real axis, and therefore this means the gain margin is infinite – meaning that no change in gain can bring about instability in the system. Using the MATLAB inspection tool, the angle at which the plot crosses the unit circle measured from the same axis is 86.1° , which means that the system becomes prone to instability at 86.1 degrees of phase shift. The system has a high degree of stability in the frequency response.

Conclusions

In conclusion, this report has demonstrated clear, tangible benefits to using mathematical modelling tools to design and test control systems before releasing them into real-world operation. Using MATLAB and Simulink, a viable control system for a Phalanx CIWS positioning system could be designed, tested and analysed to test for stability and identify area of potential fault, such as the phase and gain margins required

Shaun Malone (2030723)

BEng – Ordnance, Munitions & Explosives

to bring about instability. Using methods such as those investigated in this report prevents costly real-world evaluation of systems, that can prove costly if anything goes wrong. In reality, a control system for a Phalanx CIWS derived from scratch would make the task more complicated and require more time, but that only further demonstrates the benefits of having access to complex mathematical tools to complete the design, test and evaluation of control systems in modern engineering projects.

Appendices

A. Derivation of Whole-System Transfer Function

Beginning with the given process transfer function $G(s)$.

$$G(s) = \frac{1}{as^2 + bs + c}$$

Values for a , b and c are given but will be substituted in later.

The transfer function determining a PID controller is given as below.

$$PID\ TF = P + I\left(\frac{1}{s}\right) + D\frac{s}{\left(\frac{1}{N}\right)s + 1}$$

Values for P , I , D and N were found using MATLAB's controller tuning software to fit the system requirements. They will be substituted in later.

The first step is to find the forward path transfer function, G .

$$G = G(s) \times PID\ TF$$

Simplifying the third term of the PID TF:

$$D\frac{s}{\left(\frac{1}{N}\right)s + 1} \rightarrow \frac{Ds}{\left(\frac{1}{N}\right)s + 1} \rightarrow \frac{Ds}{\left(\frac{s}{N}\right) + 1} \rightarrow \frac{Ds}{\left(\frac{s}{N}\right) + \left(\frac{N}{N}\right)} \rightarrow \frac{Ds}{\left(\frac{s + N}{N}\right)} \rightarrow \frac{DNs}{s + N}$$

$$PID\ TF = P + I\left(\frac{1}{s}\right) + \frac{DNs}{s + N}$$

Then, turning PID TF in to a combination of fractions with a common denominator. The denominator will be the products of the separate denominators, so $s(s + N)$

$$P \rightarrow \frac{P(s(s + N))}{s(s + N)}$$

$$\frac{I}{s} \rightarrow \frac{I(s + N)}{s(s + N)}$$

$$\frac{DNs}{s+N} \rightarrow \frac{DNs^2}{s(s+N)}$$

Combining into a single fraction gives:

$$\frac{P(s^2 + sN) + I(s + N) + DNs^2}{s^2 + sN}$$

Expanding and then giving in descending powers of s:

$$Ps^2 + PNs + Is + IN + DNs^2 \rightarrow (DN + P)s^2 + (PN + I)s + IN$$

$$PID\ TF = \frac{(DN + P)s^2 + (PN + I)s + IN}{s^2 + sN}$$

Therefore:

$$\begin{aligned} G &= G(s) \times PID\ TF \\ G &= \frac{1}{as^2 + bs + c} \times \frac{(DN + P)s^2 + (PN + I)s + IN}{s^2 + sN} \\ G &= \frac{(DN + P)s^2 + (PN + I)s + IN}{(s^2 + sN)(as^2 + bs + c)} \end{aligned}$$

Expanding the denominator:

$$(s^2 + sN)(as^2 + bs + c) = as^4 + bs^3 + cs^2 + aNs^3 + bNs^2 + cNs$$

Factorising in to descending powers of s:

$$as^4 + (b + aN)s^3 + (c + bN)s^2 + cNs$$

Finally, G in its simplest form, in descending powers of s, is given as:

$$G = \frac{(DN + P)s^2 + Is + (P + I)N}{as^4 + (b + aN)s^3 + (c + bN)s^2 + cNs}$$

The whole system transfer function is represented by the formula:

$$TF = \frac{G}{1 + GH}$$

In this system, H =1, so:

$$TF = \frac{G}{1 + G}$$

Subbing in G:

$$TF = \frac{\frac{(DN + P)s^2 + (PN + I)s + IN}{as^4 + (b + aN)s^3 + (c + bN)s^2 + cNs}}{1 + \frac{(DN + P)s^2 + (PN + I)s + IN}{as^4 + (b + aN)s^3 + (c + bN)s^2 + cNs}}$$

$$1 = \frac{as^4 + (b + aN)s^3 + (c + bN)s^2 + cNs}{as^4 + (b + aN)s^3 + (c + bN)s^2 + cNs}$$

$$TF = \frac{\left(\frac{(DN + P)s^2 + (PN + I)s + IN}{as^4 + (b + aN)s^3 + (c + bN)s^2 + cNs} \right)}{\left(\frac{as^4 + (b + aN)s^3 + (c + bN)s^2 + cNs + (DN + P)s^2 + (PN + I)s + IN}{as^4 + (b + aN)s^3 + (c + bN)s^2 + cNs} \right)}$$

$$\begin{aligned} TF &= \frac{(DN + P)s^2 + (PN + I)s + IN}{as^4 + (b + aN)s^3 + (c + bN)s^2 + cNs} \\ &\times \frac{as^4 + (b + aN)s^3 + (c + bN)s^2 + cNs}{as^4 + (b + aN)s^3 + (c + bN)s^2 + cNs + (DN + P)s^2 + (PN + I)s + IN} \end{aligned}$$

Collecting like terms on the second denominator:

$$\begin{aligned} TF &= \frac{(DN + P)s^2 + (PN + I)s + IN}{as^4 + (b + aN)s^3 + (c + bN)s^2 + cNs} \\ &\times \frac{as^4 + (b + aN)s^3 + (c + bN)s^2 + cNs}{as^4 + (b + aN)s^3 + (c + bN + DN + P)s^2 + (CN + PN + I)s + IN} \\ TF &= \frac{((DN + P)s^2 + (PN + I)s + IN) \times (as^4 + (b + aN)s^3 + (c + bN)s^2 + cNs)}{(as^4 + (b + aN)s^3 + (c + bN)s^2 + cNs) \times (as^4 + (b + aN)s^3 + (c + bN + DN + P)s^2 + (CN + PN + I)s + IN)} \end{aligned}$$

Multiplying numerators gives a sixth order expression in s.

$$\begin{aligned} \text{Numerator} &= (DN + P)a^6 + ((b + aN)(DN + P) + a(PN + I))s^5 \\ &+ ((c + bN)(DN + P) + (b + aN)(PN + I) + aIN)s^4 \\ &+ (cN(DN + P) + (c + bN)(PN + I) + IN(b + aN))s^3 \\ &+ (cN(PN + I) + IN(c + bN))s^2 + cIN^2 \end{aligned}$$

Multiplying denominators gives an eighth order expression in s.

Denominator

$$\begin{aligned}
 &= (a^2)s^8 + (2a(b + aN))s^7 \\
 &+ (aC + aDN + aP + ac + b^2 + a^2N^2 + 4abN)s^6 \\
 &+ (aCN + (b + aN)(c + bN) + (b + aN)(c + bN + DN + P) + a(cN \\
 &+ I))s^5 \\
 &+ ((b + aN)(cN) + (c + bN + DN + P)(c + bN) + (b + aN)(cN + I) \\
 &+ a(cN + I))s^4 \\
 &+ ((cN)(c + bN + DN + P) + (cN + I)(c + bN) + (b + aN)(P + I)N)s^3 \\
 &+ ((cN + I)(cN) + (c + bN)(P + I)N)s^2 + ((cN)(P + I)N)s
 \end{aligned}$$

Values for substitution, as per report:

$$\begin{aligned}
 P &= 141.3575 \\
 I &= 16.4627 \\
 D &= 119.0622 \\
 N &= 1079.5407 \\
 a &= 8 \\
 b &= 3 \\
 c &= 4
 \end{aligned}$$

Substituting and simplifying gives:

$$\begin{aligned}
 TF &= \frac{(1.03 \times 10^6)s^6 + (1.114 \times 10^9)s^5 + (1.737 \times 10^9)s^4 + (1.205 \times 10^9)s^3 + (7.172 \times 10^8)s^2 + (7.68 \times 10^7)s}{64s^8 + (1.383 \times 10^5)s^7 + (7.577 \times 10^7)s^6 + (1.17 \times 10^9)s^5 + (1.822 \times 10^9)s^4 + (1.233 \times 10^9)s^3 + (7.358 \times 10^8)s^2 + (7.68 \times 10^7)s}
 \end{aligned}$$

This is the whole system transfer function for the report, fully derived, as given in the report.

References

- [1] Raytheon , “Phalanx Weapon System,” RTX, 2025. [Online]. Available: <https://www.rtx.com/raytheon/what-we-do/sea/phalanx-close-in-weapon-system>. [Accessed 13 November 2025].
- [2] C. Ryan, “Phalanx CIWS - An Advanced Naval Close-In Weapon System,” Naval History, 19 June 2023. [Online]. Available: <https://navalhistoria.com/the-phalanx-ciws-an-advanced-naval-close-in-weapon-system/>. [Accessed 13 November 2025].

- [3] WeaponSystems.net, “20mm Mk15 Phalanx,” WeaponSystems.net, 2025. [Online]. Available: <https://weaponsystems.net/system/508-20mm%20Mk%2015%20Phalanx>. [Accessed 13 November 2025].
- [4] N. S. Nise, “Introduction to Control Systems,” in *Control Systems Engineering*, Hoboken, John Wiley & Sons, 2019, pp. 2-4.
- [5] N. S. Nise, “Control System Configurations,” in *Control Systems Engineering*, Hoboken, Wiley & Sons, 2019, pp. 6-8.
- [6] A. S. Morris and R. Langari, “Instrument Types and Performance Characteristics,” in *Measurement and Instrumentation (2nd Edition)*, Academic Press, 2016, pp. 13-44.
- [7] W. Bolton, “System Parameters,” in *Control Systems*, Newnes, 2002, pp. 85-98.
- [8] N. S. Nise, “Time Response: Second Order Systems,” in *Control Systems Engineering*, Hoboken, John Wiley & Sons, 2019, pp. 137-142.
- [9] N. S. Nise, “Analysis and Design Objectives,” in *Control Systems Engineering*, Hoboken, John Wiley & Sons, 2019, pp. 9-11.
- [10] R. Chen, “A Comprehensive Analysis of PID Control Applications in Automation Systems: Current Trends and Future Directions,” *Highlights in Science, Engineering and Technology*, vol. 97, pp. 126-132, 2024.
- [11] K. Iqbal, “PI, PD and PID Controllers,” in *Introduction to Control Systems*, Little Rock (University of Arkansas), LibreTexts, 2024, pp. 60-63.
- [12] K. Iqbal, “Stability of the Closed-Loop System,” in *Introduction to Control Systems*, Little Rock (University of Arkansas), LibreTexts, 2024, pp. 68-73.
- [13] N. S. Nise, “Poles, Zeros, and System Response,” in *Control Systems Engineering*, Hoboken, John Wiley & Sons, 2019, pp. 131-135.
- [14] N. S. Nise, “Root Locus Techniques,” in *Control Systems Engineering*, Hoboken, John Wiley & Sons, 2019, pp. 306-317.
- [15] N. S. Nise, “Gain Margin and Phase Margin via the Nyquist Diagram,” in *Control Systems Engineering*, Hoboken, John Wiley & Sons, 2019, pp. 456-462.