

Development of a Close Quarter Collision Protected Aerial Drone

Angus Steele

November 28, 2018

Abstract

In this work a solution to flight inside a confined environment is proposed and implementation through simulation. The goal of the design is enable a vehicle to navigate in an unknown confined space avoiding collision with walls and unexpected obstacles. The selection of an appropriate aircraft design is done, followed by a controller design and implementation of a flight strategy including an obstacle avoidance routine. The goal of the project was proven through in depth mathematical modelling of the system and the environment. Simulation and testing was performed using a model generated in Matlab and Simulink.

The best suited configuration of a quadrotor was chosen based on a rigorous analysis of existing rotorcraft, highlighting the shortcomings and benefits of each design. The craft designed needed to be streamlined to ensure maximum capabilities in a narrow space. The vehicle was mathematically modelled based on the chosen configuration and sensors that would be used.

The flight control was designed to be robust against disturbances and have tight position tracking to ensure stable flight inside an confined environment and avoid collisions. The controllers were designed and shown to be able to withstand disturbances the vehicle will be challenged of facing.

The flight strategy was developed to optimise the use of such a platform in a narrow space similar to that seen inside a mining environment. A heading alignment strategy was added to the existing controllers ensuring the drone maintains a set heading based on it's current velocity. The final consideration was a thorough design and implementation of an obstacle avoidance routine. The system had to ensure no collisions would occur and ensure suitability for the implementation of a higher route planning strategy in future work.

The simulated craft was shown to be capable of navigating inside an unknown environment and successfully avoid collisions with the use of a waypoint generator.

Contents

1	Introduction	7
1.1	Project Background	7
1.2	Problem Statement	8
1.3	Application Requirements	8
1.3.1	Controlled Indoor Flight	8
1.3.2	Method of Expansion for Industrial Applications	9
2	Literature Review	10
2.1	Flight Theory	10
2.1.1	Fundamental Principles of Flow	11
2.1.2	Basic Rotor Theory	12
2.1.3	Momentum Theory and Thrust Basics	13
2.1.4	Disk and Power Loading	15
2.1.5	Electrical Power to Thrust	16
2.2	Analysis of Conventional Rotor Wing Configurations	17
2.2.1	Helicopter	18
2.2.2	Coaxial Rotors	19
2.2.3	Tandem Rotors	20
2.2.4	Multirotor Designs	21
2.2.5	Tilt Rotors	22
2.2.6	Discussion	23
2.3	Quadrotor Flight Dynamics	23
2.3.1	Coordinate Systems, Rotations and Nomenclature	23
2.3.2	Kinetics and Kinematics	26
2.3.3	Mass Model and the Inertia Tensor	27
2.3.4	Rotor Generated Forces and Moments	28
2.3.5	Disturbances	30
2.3.6	Instrumentation	33
2.4	Review of Existing Flight Control Strategies	34
2.4.1	Controller Architecture	34
2.5	Collision Protection Techniques	35
2.5.1	Impact Resistance	36
2.5.2	Rolling Cage	36
2.6	Collision Avoidance Techniques	36
2.6.1	Sensing Techniques and Requirements	36
2.6.2	Collision Avoidance Algorithms	37

3 Platform Design	40
3.1 System Hardware Architecture	40
3.1.1 Platform Construction	41
3.1.2 Electronics Interface	42
3.2 Platform Construction	44
3.2.1 Design Considerations	44
3.2.2 Chosen Concepts	45
3.2.3 Concept Comparison	48
3.3 Electronic Design	50
3.3.1 Flight Controller	51
3.3.2 On Board Computer	52
3.3.3 Location	54
3.3.4 Object Avoidance	54
4 Mathematical Modelling and System Identification	56
4.1 Dynamic Flight Model	56
4.2 System Identification	57
4.2.1 Mass and Inertia	57
4.2.2 Thrust and Moment Profiles	57
4.2.3 Drag Coefficients	58
4.2.4 Sensor Constants	59
4.2.5 Wind Model	59
4.3 Simulation Configuration	60
4.3.1 Motor Mixer	60
5 Controller Design	62
5.1 Design Goals	62
5.2 Flight Control Strategy	63
5.3 Altitude Controller	64
5.3.1 Heave Dynamics	64
5.3.2 Heave Controller	65
5.3.3 Climb Rate Controller	68
5.3.4 Altitude Hold Controller	70
5.4 Horizontal Control	75
5.4.1 Roll and Pitch Rate Dynamics	75
5.4.2 Roll and Pitch Rate Controllers	76
5.4.3 Tilt Angle Controller	81
5.4.4 Linear Velocity Control	85
5.4.5 Global Position Tracking Control	88
5.5 Heading Controller	90
5.5.1 Yaw Rate Dynamics	91
5.5.2 Yaw Rate Controller	91
5.5.3 Yaw Angle Controller	94
6 Flight Strategy and Obstacle Avoidance	99
6.1 Flight Strategy	99
6.1.1 Waypoint Generation	99
6.1.2 Yaw Alignment	100
6.1.3 Yaw Alignment Discussion	101

6.2	Obstacle Avoidance	103
6.2.1	Mathematical modelling	103
6.2.2	Obstacle Avoidance Methodology	105
6.2.3	Obstacle Avoidance Discussion	109
7	Simulated Flight Tests	112
7.1	title	112
7.2	Aims and Objectives	112
7.3	Testing Methodology and Results	113
7.3.1	Fully Integrated and Controlled System	113
7.3.2	Basic Obstacle Avoidance Routine	115
7.3.3	Obstacle Avoidance Navigation	118
7.3.4	Generic Mission Flight	123
7.3.5	Limitations of the Design	127
8	Conclusions and Recommendations	129
8.1	Summary and Conclusions	129
8.2	Recommendations	130

List of Tables

2.1	Standard Nomenclature	26
2.2	MAV Weight Data (Adapted from [38])	27
3.1	Table representing the size comparison of the two concepts	50
3.2	Table representing the end comparison of the two concepts	50
4.1	Measured Moments of Inertia	57
4.2	Measured Moments of Inertia	58
4.3	Drag Coefficients	59
4.4	IMU Sensor Coefficients	59
4.5	GPS Coefficients	59
5.1	Thrust Headroom Controller Percentages	64
5.2	Heave Controller Limits	67
5.3	Climb Rate Controller Limits	70
5.4	Altitude Hold Controller Limits	73

Glossary

1. Micro Aerial Vehicles (MAV)
2. Unmanned Aerial System (UAS)
3. Unmanned Aerial Vehicle (UAV)
4. Disk Loading (DL)
5. Power Loading (PL)
6. Degree of Freedom (DOF)
7. On Board Computer (OBC)
8. Ground Control Station (GCS)

Chapter 1

Introduction

This chapter sets to lay out as an introduction to the research. By discussing the end use case of the design, a few critical research questions can be raised. A problem statement will be drawn out by these questions. The chapter will then be concluded with a list of extracted requirements that the system needs to address and ultimately accomplish.

1.1 Project Background

Tracked and wheeled robots are beginning to reach their limitations, and society is in need of more complex and versatile vehicles. For a land robot to successfully navigate an extremely complex or cluttered environment, the designer must look at creating a legged robot. Legged designs introduce complexity into any system due to the intensive control theory required. There has been some great progress in legged designs, such as Big Dog created by Boston Dynamics [33]. Nevertheless, deploying currently available legged platforms could cost valuable time with lengthy navigation routines. An alternative approach would be to use an aerial platform that could do overhead surveillance. A drone could complete the required task by flying over the complexities in the operating environment. However, with conventional flight techniques and platforms, almost any sort of collision would cause a failure and the system would not be able to complete its mission. This limits the approach to only work in a situation where the drone would be able to fly over the obstacles.

Many of the desired use cases are not open aired and the platform will be required to fly below and even through obstacles to complete its task. A good example of this is in search and rescue missions. An aerial vehicle will be required to navigate through damaged or even collapsed buildings. The same platform could be used in a mining environment. Used to assist miners in assessing unexplored and potentially dangerous areas. From the late 1800s South Africa has had a massive mining community, with coal, gold and diamond mining being used as a major source of income and job creation for the country. In response to this, two South African research institutes have agreed to a joint collaboration in solving some of these aspects for an underground mine environment. This project involves both The University of Stellenbosch (US) and the Council of Scientific and Industrial research (CSIR). A mining environment has many applications for a collision resistant drone. Such as the mapping of unknown and potentially hazardous environments. The drone would be able to fly in, conduct a survey of the environment and feed that information back to the miners, ensuring a safer work environment while minimising costly delays.

1.2 Problem Statement

There are many potential applications for a drone capable of close quarter flight. CEPAD was seen as a potential safety platform for use in mines. Unsafe underground territories create a need for unmanned vehicles to do inspections. These areas are currently been mapped by trained professionals who risk their lives going into these unsecured regions. Using land vehicles proves difficult and slow in the complex terrains, creating a need for an alternative solution.

Designing any aerial drone introduces many complexities, including obtaining the required aerodynamics to achieve stable flight. There are modules that one can buy to stabilise the craft, but in a confined indoor space, this specification gets enhanced with the need to stabilise itself after a collision or due to flight near surfaces as shown in [36, 19]. Several strategies will need to be investigated to assist the device in navigating these confined environments. The platform should attempt to maintain a set distance from the walls, floors and other obstructions. For an indoor application it is important that the device can fly in a GPS constrained environment. Although this factor will not be solved in the scope of this project, the design of CEPAD should consider some of the factors involved to ensure expansion into that research can be done with relatively small changes to the work accomplished here.

To ensure the platform can be extended to industrial applications, certain external factors and peripherals need to be included. The device will need to be able to complete some of this missions autonomously, especially when line of sight and potentially, communications are lost. The platform must be able to handle and interface to an array of sensors for each specific mission. The drone will need to be small to increase its accessibility in confined spaces, this will limit payload and flight time. To complete a useful mission the platform must have a sufficient flight time by allowing a larger power source on board the craft.

1.3 Application Requirements

The problem statement above briefly introduces certain needs CEPAD must solve, this section that follows attempts to address each of these points and define them more specifically as key requirements for the system as a whole. To begin, this statement of requirements can be started by creating two high level objectives, achieving flight in a confined indoor space and providing the ability to complete industrial applications.

1.3.1 Controlled Indoor Flight

The identified requirements of the system begin by providing an aerial platform with the ability to fly in an enclosed, confined space. To do this, the platform must be able to position itself from any obstructions above, below or beside itself. This will require that the platform can measure its proximity to the surroundings in all directions. The flight controller must therefore be able to include additional sensor inputs into its control laws and other real time processes. When a new obstruction is detected it must have the ability to steadily move away and reposition itself, while not straying too far off the mission plan.

In order for the platform to complete missions in this environment, the drone needs to withstand the disturbances introduced by flight close to obstructions as well as collisions

with these obstructions. This creates a requirement to mechanically protect the platform from irreversible damage caused by a collision. The flight controllers on board must be able to stabilise the platform post collision. Additional requirements are introduced due to disturbances by being in close proximity and not necessarily colliding with, walls, floors and other obstructions. The flight control must be equipped to handle the near wall/ground effect.

1.3.2 Method of Expansion for Industrial Applications

If the above requirements are met, the platform could be expanded into an array of industrial and research applications. Most of these use cases will require additional flight modes, sensors and other peripherals. Although not all these factors will be proven for in this project, they must be considered so that expansion into these realms can be done with minimal rework being needed on the platform.

Since most of these missions will require some level of autonomy, the chosen flight controls must include an autopilot flight mode that allows the user to switch between manual and automatic mode. There must be a method to send flight data back to a ground control station for real time analysis of the mission. The ground control station should be able to update or halt the mission plan during deployment. Since additional sensors will be required, the platform must include some interface to handle the sensor data and relay the live sensor data back to a ground control station. The platform must provide a mechanism of mounting an array of different sensing equipment on board. This includes accounting for the extra thrust and electrical power requirements added by the sensors. Finally the drone must be able to stay in the air long enough to complete a mission. With indeterminable mission lengths at this point, the device must be able to expand it's battery capacity to account for longer missions.

Chapter 2

Literature Review

This chapter seeks to evaluate existing research in the field of rotorcraft design and obstacle avoidance strategies. To critically review some of the high level concepts in rotorcraft design, a brief evaluation is given of flight theory and how they effect design decisions for rotorcraft. After an understanding of flight theory is obtained, it is necessary to evaluate how this theory is utilised in creating rotorcraft. Armed with a better understanding of flight generation for rotorcraft, an analysis of traditional rotorcraft configurations is completed. Due to the hazardous nature of the mission environments, existing collision protection techniques are then discussed. At this point, the reader should feel confidence that the platform design is grounded with a solid understanding of existing techniques and requirements. Now that a platform has been developed, the next step is to review some of the methods used to control multi-rotor platforms. Once stable flight methods have been evaluated and discussed, the researcher reviewed existing methods for obstacle avoidance as well as the requirements of implementing an on-board obstacle avoidance system.

2.1 Flight Theory

Although there are many different forms of flight, each form will have a very similar force diagram as seen in Figure 2.1.

Weight is directly determined by the object's mass and the relevant gravity coefficient,

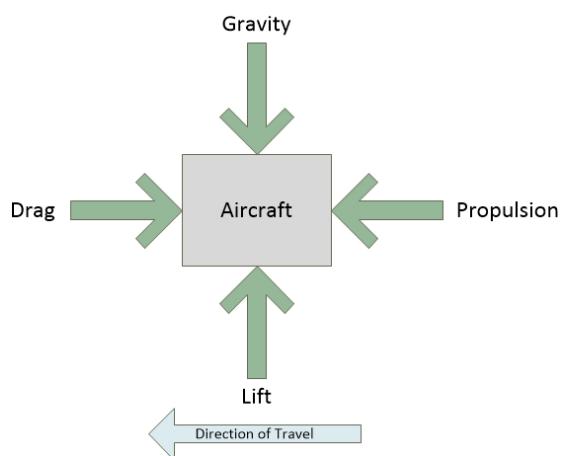


Figure 2.1: Force diagram of basic components of flight

$W = mg$. Lift counteracts this weight in an attempt to boost the body into the air. Upward acceleration is only achieved once lift exceeds weight, if they are equal the body will be in a state of hover. From the lift equation seen in (2.1) only a body with velocity can obtain lift. In a rotorcraft, the rotating blades move through the air and generate lift, thereby negating the requirement for the body to have any linear velocity. Unlike in a fixed wing aircraft if the vehicle is stationary, zero lift is generated, thus a vertical take-off is not possible.

$$F_{Lift} = C_L \left(\frac{1}{2} \rho V^2 \right) S \quad (2.1)$$

To propel the body forward the propulsion must exceed the value of the drag force, which acts directly against its velocity vector. When no propulsion force is present, the craft will continue to lose speed due to drag. Similar to lift, drag varies with velocity as shown in equation (2.2).

$$F_{Drag} = C_D \left(\frac{1}{2} \rho V^2 \right) A \quad (2.2)$$

The coefficient of drag (C_D) is determined by the objects shape and ultimately the way it interacts with the air flow.

Figure 2.1 better describes a fixed wing aircraft, the lift and propulsion forces in a rotorcraft can be seen as the components of thrust which generate either a vertical or horizontal force. To better understand the design behind rotorcraft, the principles behind this thrust generation are discussed.

2.1.1 Fundamental Principles of Flow

Rotors generate thrust by pulling quiescent air through the rotor plane. The principles that best govern these flow dynamics and force generation are discussed below.

Continuity Equation and Bernoulli's Principle

Bernoulli observed that the mass flow in a closed system remains constant, as shown in (2.3). This principle states that in a closed system, the product of density (ρ), area (A) and velocity (v) for a flowing system will remain constant [37]. Based on this finding it was discovered that the mass flow of a flowing medium will follow the laws of continuity in the form of equation (2.4).

$$\Delta \rho A v = 0 \quad (2.3)$$

The Bernoulli equation is a statement of the conservation of energies present in a flowing system [37]. Equation 2.4 considers a pipe with a flowing liquid and states that the energy will remain unchanged in a closed system. The sum of these energies will contain the kinetic energy of the liquid as well as the energy present through pressure. Bernoulli's equation can be rewritten in the form (2.5) to represent that any changes in pressure, can result in a change of the velocities.

$$P_0 + \frac{1}{2} \rho v_0^2 = P_1 + \frac{1}{2} \rho v_i^2 \quad (2.4)$$

$$P_2 - P_1 = \frac{1}{2} \rho (v_\infty^2 - v_0^2) \quad (2.5)$$

Reynold's Number

Moving different objects in the same environment will create different results of flow, in the same breath moving the same object through different environments will also create various results of flow. Osborne Reynold attempted to mathematically determine these effects and quantify what caused a system to have turbulent flow opposed to laminar flow and vice versa. During his research he created a dimensionless constant known as the Reynold's number, as shown in equation 2.6 [30].

$$Re = \frac{\rho v L}{\mu} \quad (2.6)$$

In its simplest form, the Reynold's number of an object is the ratio between inertial forces ($\rho v L$) and viscous forces (μ) of the gas. In a system where the viscous forces dominate ($Re < 10^3$), there will be laminar flow and when there are higher inertial forces ($Re > 10^4$) the flow will be turbulent. Since turbulent flow will decrease stability and increase drag forces, Reynold's numbers have become a very important part in correctly modelling and designing aircraft [30].

2.1.2 Basic Rotor Theory

The rotor is responsible for all the aspects of flight and generates the lift, forward propulsion and the means to control the orientation of the craft [21]. It is for this reason that an in depth understanding of rotor characteristics and performance was done. The original research into rotor analysis was done with helicopters, but the rotor theory basics are relevant to any rotating winged craft.

Due to Newton's third law, any rotating blade will cause a rotation in the opposite direction to that motion. This applied force will drive the vehicle to rotate around that spinning axis and creates the need for a counter torque mechanism. This is common with most rotorcraft and can be visualised in figure 2.2, in the form of the tail rotor as seen in conventional helicopters. The quadrotor handles this by having an equal number of oppositely rotating rotors.

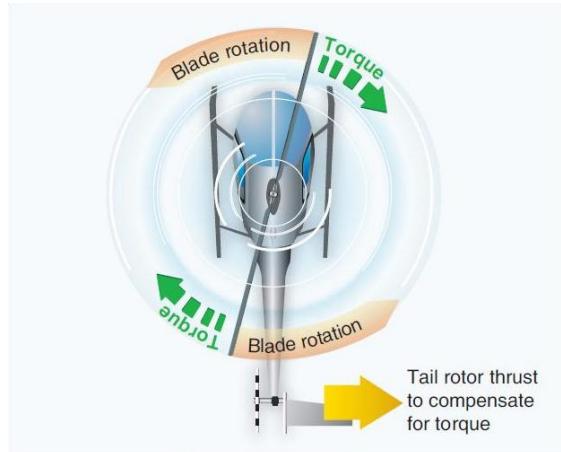


Figure 2.2: Image Illustrating generation of Counter Torque (Taken from [10])

The capability of any part of a rotor to produce lift is influenced by the local blade position and pressure at that point [21].

Angular velocity equations state that the speed of any part of the rotor varies along the length of the rotor. With the maximum velocity sitting at the rotor tip. As the rotor spins, the blade's angle of attack shifts. This angle is defined as an azimuth angle (ψ) and is measured relative to air flow. The azimuth angle is 0° down stream and sits at 180° when it faces directly upstream. As the rotorcraft adds a horizontal component to its hover or vertical flight, the relative speed of the individual rotor segments now adheres to equation (2.7). As visualised in figure 2.3, the relative velocity at the any part of the rotor is affected by the azimuth angle of the blade (ψ), translational speed of the craft (V_∞), angular speed of the rotor (Ω) and the considered distance along the rotor blade (r) [21] [1].

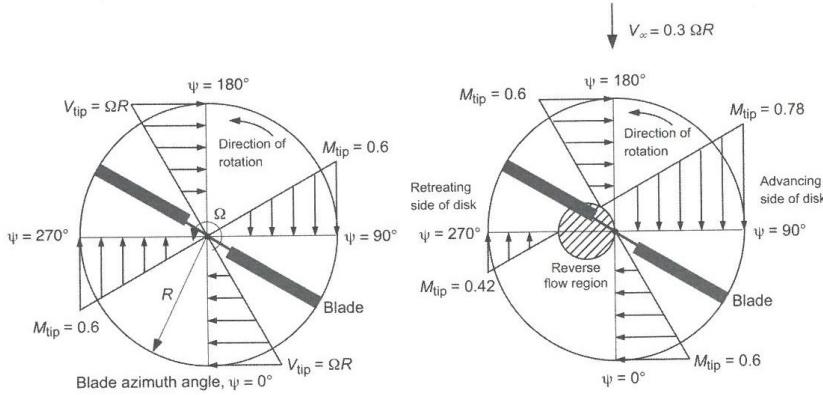


Figure 2.3: Velocity components of a rotor [21]

$$V_r = \Omega r + V_\infty \sin(\psi) \quad (2.7)$$

What this relationship shows is that during forward flight the tip velocity, relative to the ground, changes even if the rotor rotates at a constant speed. This complicates the rotor dynamics at higher speeds and limits the top speed of the craft. On the retreating edge ($\psi = 270^\circ$: $\sin(\psi) = -1$) if $\Omega r <= V_\infty$ the rotor would effectively be going backwards and the helicopter is at risk of stalling out, this is known as a stall condition [21] [1], while the advancing edge is reaching its maximum speed by approaching Mach conditions and sever instability.

2.1.3 Momentum Theory and Thrust Basics

As mentioned above the rotors of a rotorcraft are responsible for generating all the forces that manoeuvre the vehicle. These forces are induced by pushing air through the rotor disk. With a fixed wing aircraft the analysis of the blades is simplified because the only air flow produced is from the translational velocity of the entire craft. Analysis of blade performance in a rotorcraft can be more challenging as the rotation of the blades must be considered along side the overall speed of the vehicle. As the craft manoeuvres in space, the air flow through the rotor has significant complexities which complicates the analysis. Since the rotorcraft is expected to perform in a variety of flight styles it is important to understand these models, and their flaws.

To simplify, initially consider a helicopter in a hovering state ($\text{Weight}(W) = \text{Thrust}(T)$). Figure 2.4, taken from [21], helps visualise the induced air flow by showing how the rotor smooths out the air by forcing it through the disk area. This more uniform air creates an edge known as the slipstream or wake boundary, with the surrounding air remaining dormant [21]. Inside the wake boundary, the average velocity of the air is tangible and effective, where outside the slipstream edge, the average air velocity is negligible and obsolete. The force required to push that mass of air through the disk space is, by Newton's third law, returned by the air unto the rotor. Thus giving the rotor blades a thrust component.

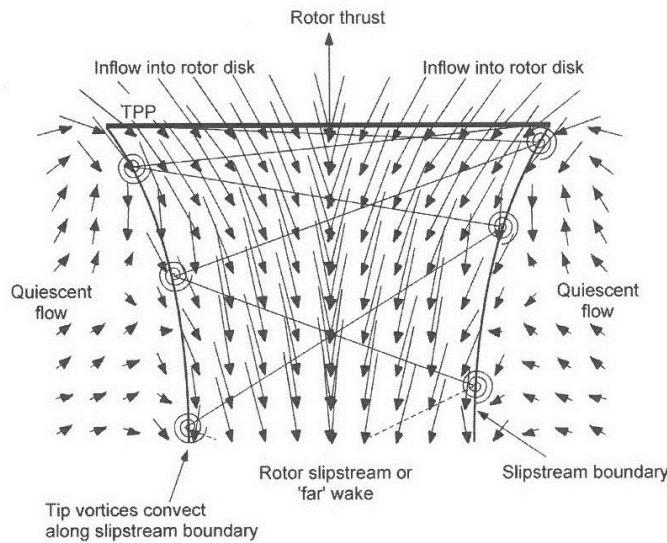


Figure 2.4: Visualisation of Induced Air Flow Through A Rotor [21]

Rankine-Froude's Momentum Theory looks at this induced velocity as well as the displacement of air through the propeller, and attempts to quantify the induced thrust. While figure 2.4 helps visualise the principle, the variable naming convention for the equations is shown in figure 2.5 below. Labels 0, 1, 2 and ∞ refer to the locations of quiescent flow, inflow directly before the rotor, airflow immediately after the disk and the slipstream¹ or far wake condition respectively. The velocities are shown as the induced velocity in and out the rotor (v_i), the far wake velocity (v_∞) and finally v_0 represents the zone with zero flow rate. There is no velocity jump across the rotor, the energy being fed into the system by the rotor is represented by a pressure change.

As described above, it is by forcing the air through the disk that lift is generated. The mass flow rate of this air can then be described by (2.8), where (ρ) is the density of air and A is the area of one full blade rotation. The rate at which this mass of air is displaced becomes a crucial variable in rotor dynamics and is directly proportional to thrust (T). This relationship can be quantified as shown in (2.9). Thrust can also be calculated by finding the difference in pressures over the rotor disk as in (2.10). Since acceleration is the difference in v_∞ and v_0 , (2.11) can also be obtained.

¹Generally far wake is considered as 1 full rotor diameter distance away [21].

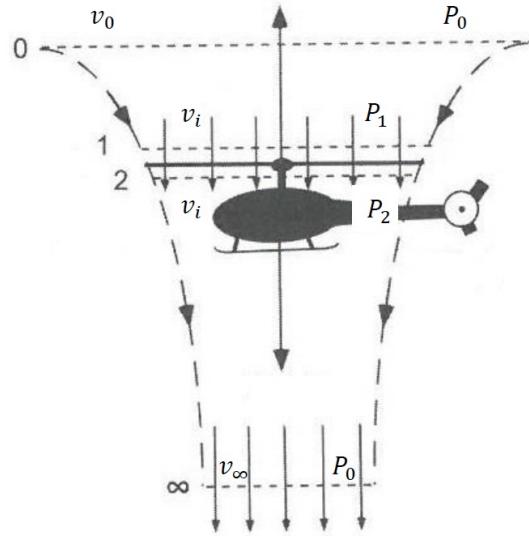


Figure 2.5: Momentum Theory in Hover (Adapted From [21])

$$\dot{m} = \rho A v_i \quad (2.8)$$

$$T = \dot{m} a \quad (2.9)$$

$$T = A(P_2 - P_1) \quad (2.10)$$

$$T = \rho A v_i (v_\infty - v_0) \quad (2.11)$$

Equation (2.11) demonstrates the negative effect of having active ambient air. This could be caused by surrounding turbulent airflow, wind or even translational movements and will need to be considered in the controller design.

2.1.4 Disk and Power Loading

Disk Loading

Disk loading (DL) is a term seen often in the world of rotorcraft, it is a simple but important ratio between thrust and the area a rotating disk makes. It is represented in (2.12). Since the pressure drop across each rotor is considered uniform, the disk loading for each rotor will equate to the pressure drop across that disk.

$$DL\left(\frac{N}{m^2}\right) = \frac{T}{A} = \frac{1}{2} \rho v_\infty^2 \quad (2.12)$$

For multi-rotor craft, the disk loading is assumed uniform across all rotors [21]. The overall disk loading of a single rotorcraft such as a traditional helicopter will be lower than that of a multi-rotor craft of a similar size [1]. Figure 2.6 shows some examples of disk loading values for a variety of rotor configurations, as shown disk loading is also a measure of hover efficiency.

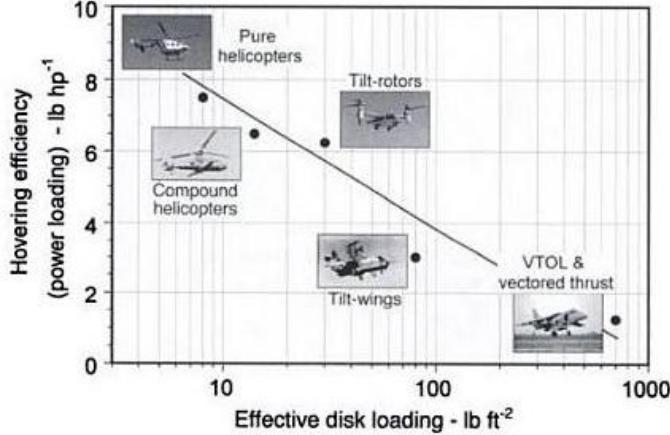


Figure 2.6: Image representing, various Disk Loading values for varying rotorcraft (Taken from [21])

A higher disk loading value results in larger values for induced velocities as well as the required power to hover. This means that the larger the blades, the higher the efficiency. More force will be generated by pushing large quantities of air slowly, than forcing small amounts of air through at high speeds. Of course with bigger blades, comes larger rotational inertia and geometry as well as the craft being less immune to gusts and interferences. A larger blade also creates faster tip velocities, which will limit the speed of the craft severely [21].

Power Loading

Power is given by the product of both Thrust and the induced velocity at the blade. It can be written as shown in equation (2.13). What this ratio shows is that the ideal power is in cubic proportion to the induced velocity at the rotor. Therefore to reduce required power the rotor's induced velocity must be small, which can be accomplished by a significant increase in disk area [21].

$$P = 2\rho Av_i^3 \quad (2.13)$$

Another important ratio is between thrust and power, it is called power loading (PL) and is shown in equation (2.14). Power loading can be seen as a measure of craft efficiency.

$$PL\left(\frac{N}{kW}\right) = \frac{T}{P} \quad (2.14)$$

From equations (2.12) and (2.14) it can be shown that power loading is inversely proportional to disk loading. Therefore a craft with a lower disk loading will generally be a more efficient platform.

2.1.5 Electrical Power to Thrust

Equation (2.13) gives a quantitative approach to solving for aerodynamic power (P_i). If electrical power is taken as $P_e = VI$, where V is the applied voltage and I is the sourced current, with an efficiency of η then $P_i = \eta VI$. Noting that $P_i = Tv_i$ and using equation (2.13), a relationship between thrust and P_e can be formed and is represented in equation (2.15).

$$T = (2\rho A)^{\frac{1}{3}} (\eta P_e)^{\frac{2}{3}} \quad (2.15)$$

Equation (2.15) brings to light a very important relationship which states that thrust grows at a slower rate than the electrical input power to the system.

$$T \propto P_e^{\frac{2}{3}}$$

2.2 Analysis of Conventional Rotor Wing Configurations

Some of the fundamental theories described above relate to the basics behind various rotor configurations and even varying flight techniques. Each different arrangement of blades introduces certain advantages and disadvantages to the system. Not every configuration will be applicable for all operations and it is important to determine what criteria are critical for the intended application. An analysis of varying rotor configurations is done below and follows a similar trend to that seen in [7], [4] and [38]. The main weighted criterion for the discussion were listed in no particular order as:

1. Flight time and efficiency
2. Geometry and size
3. Drone Manoeuvrability
4. Control algorithms
5. Mechanical complexity

Before the analysis can be done, certain operating parameters of the different craft, surrounding the above mentioned criteria, need to be understood. There have been discussions regarding how rotor blades produce lift, this section discusses the real world implementation of those blades.

The same way that a car tyre is the only way the energy from the engine is translated into motion, the rotor in a rotorcraft is responsible for taking the kinetic energy from the motors and translating it into flight. Typically a rotorcraft will be designed with either fixed pitched, or variable pitched rotors. A fixed pitched rotor is a rotor that has an optimally selected, unchangeable pitch and therefore a fixed angle of attack. This of course means that since the angle of attack is fixed for the blade, an increase in speed will be required for a change in lift. With a variable pitched blade, the pilot can change the angle of attack to increase the forces. As the angle of attack increases, the blade will produce more lift without changing the speed of the motor. However, as the pitch increases, so does the drag of the blade. This then requires more motor power to keep the blade moving through the air. The power requirements for either system are fairly similar, the advantages of a varying pitch is a single rotor has the potential for more dynamic force applications. The downfall however is the high level of complexity in the mechanical design. Both of these factors become pertinent in the final decision making of the platform design.

It is also known that any rotating member will produce a counter rotating torque to the static body, which means that any system with only one rotor will have inherent instability in the yaw axis. The end goal is to have a craft that can fly stably and accurately in 3 dimensions.

Having only a single, fixed pitched rotor allows only for control in the amount the craft flies up or down, as well as this fore mentioned instability. There are many different methods to obtain full six degrees of flight freedom. The following discussion tries to address each point listed above for different traditional methods.

2.2.1 Helicopter

A conventional helicopter is still the most widely used configuration for large rotorcraft [7]. It consists of a single main rotor, coupled with a smaller counter rotating rotor located in the tail as seen in figure 2.7, this is to counteract the developed counter torque as shown in Figure 2.2.

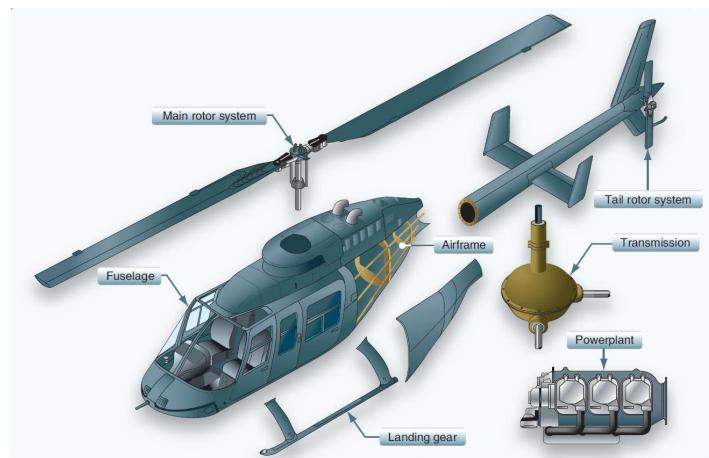


Figure 2.7: Main Components of a helicopter (Taken from [10])

The main rotor of a standard helicopter has very low disk loading which gives it excellent hover efficiency. Since the desired end product will mostly be in a state of hover or at least slow lateral movement, this will yield good results for flight time. To achieve yaw stability this configuration makes use of a small tail rotor to counter act the induced moments. The extended tail rotor requires energy which it will draw from the motor while also adding a significant amount of length and weight to the craft. Since the single rotor only gives the pilot thrust control and the tail rotor gives measurable yaw control, there is need for more control surfaces to do more manoeuvring. To implement this most helicopters use a variable pitched rotor system. Cyclic control of this pitch allows the pilot to adjust the angle of attack of the rotor blades while they rotate, thus a forward pitch can be applied by increasing the lift on the left². This set up is mechanically very complex and takes intensive control algorithms and laws to give stable control.

Even though the classic helicopter image is always seen as a main rotor with a smaller rotor at the tail, there are many different types of anti torque tail set ups. The ducted fan approach increases the efficiency of the tail rotor by channelling the air flow of the rotor.

²This is true for an American style helicopter. The French design requires am increase of lift to the right

The NOTAR design [22] as seen in Figure 2.8 manipulates the airflow generated by the main rotor and directs it to counter act the induced torque. A tip-jet design eliminates the torque applied to the airframe and therefore no tail rotor is required [7].

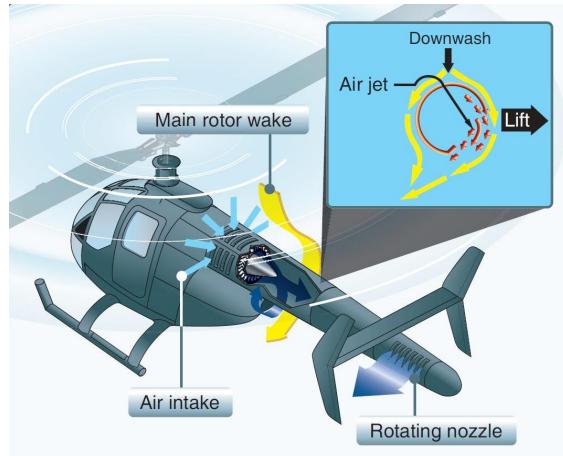


Figure 2.8: Image demonstrating the NOTAR system (Taken from[10])

There have been many attempts at improving the standard helicopter design. These improvements have taken the form of adding rotors, designing hybrid aircraft and complex mechanical designs to harvest advantages of both the fixed wing and VTOL craft. Some have even tried to combine multiple features as Flanigan [11] did in his design of a tip-jet, compound, tilt rotor aircraft. In an attempt to keep the mechanical complexity to a minimum, not all configurations were investigated.

2.2.2 Coaxial Rotors

A coaxial configuration consists of two counter rotating blades located about the same centre of rotation that both use the same drive system. This eliminates the need for a tail rotor as the torque applied by both rotors cancel each other out, as shown in figure 2.9.

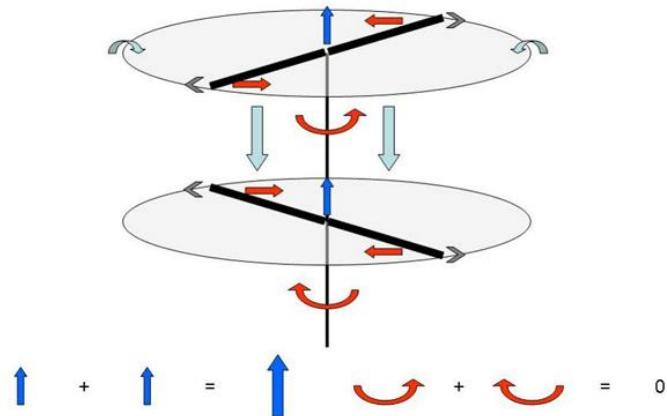


Figure 2.9: A standard Coaxial rotor set up and the induced forces

Localising the blades around a single point helps with the geometry of the craft as it is a more compact design. Using fixed pitched rotors, this platform will only give yaw

and over all thrust control. Bohorquez et al in [4] attempted a number of lateral control methods, eventually settling on aerodynamic flaps to control the flow of the downwash, that and other methods are shown in figure 2.10. Briod et all also used the same set up in his team's design of the Gimball [6].

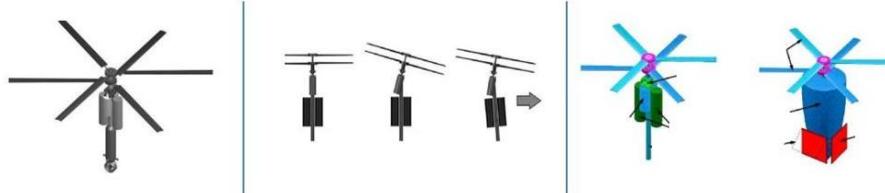


Figure 2.10: Different methods of lateral control in a Coaxial MAV (Adapted from [4])

The control flaps are the most common used form of lateral control for small coaxial MAVs. They introduce little mechanical complexity and do not require excessive power to use. The flaps do however decrease efficiency of the system by interfering with the rotor airflow. If designed correctly the flaps should only influence the system while in use. For hover and vertical flight the impact will be negligible. As a control surface the flap is quite rudimentary and will require some more advanced control methods as well as in depth testing to obtain smooth flight transitions. Due to its compactness the design can have considerable manoeuvrability if the control algorithms are designed effectively. Each flap will require an actuator, this will increase total weight, power consumption and required mechanics [4].

The overlap of the rotors also induces an inefficiency into the system. Johnson in [17], says there is much debate in how the loss of power is calculated. He states two of his preferred methods, the method chosen has a better approximation for small overlaps and is shown in (2.16) [17]. ΔP is the interference power (considered here as a fraction of total power) and m is the overlap fraction and is calculated in (2.17) [17].

$$\frac{\Delta P}{P} = \left(\frac{2}{2 - m} \right)^{1/2} - 1 \quad (2.16)$$

$$m = \frac{2}{\pi} \left[\cos^{-1} \frac{l}{2R} - \frac{l}{2R} \sqrt{1 - \frac{l^2}{4R^2}} \right] \quad (2.17)$$

These quantities assume a small vertical separation so that the inflow rates of both rotors can be considered the same. To calculate the overlap function, the rotor radius R is needed as well as the separation distance l .

2.2.3 Tandem Rotors

A tandem rotorcraft is sometimes referred to as a dual rotor, as it consists of two blades to generate thrust and thereby decreasing disk loading and increase the lift capacity. In a tandem configuration the blades sit in the front and the rear of the craft. Tandems are often used in applications that require heavier loads than the traditional rotorcraft can effectively offer. A tandem configuration is demonstrated in Figure 2.11³, the blades spin

³Image taken from <https://www.snafu-solomon.com/2011/11/ch-46-flight-ops-aboard-uss-new-orleans.html>

in opposite directions to counteract the other's rotational torque. Pitch and Yaw control are readily available through manipulation of the rotor speeds, while roll control is not as easily accomplished with this design and generally require variable pitch rotors [31]. Using two smaller blades also decreases the effects of interferences such as gusts on the craft.



Figure 2.11: A military CH-46E Sea Knight, example of a typical tandem rotor

As described in (2.15) the thrust of the system increases slower than the electrical power input into the system. In a standard configuration, doubling the electrical power would only increase the thrust by a factor of ≈ 1.587 . Where as doubling the amount of rotors being driven will double both the thrust and the electrical power. This gives the tandem arrangement the capability of lifting heavier loads with relatively low power consumption, as well as demonstrating low power consumption for hover and slow translatory flight. Having twin blades does increase the size of the craft, but the elimination of the tail rotor sees the size being similar to that of a classic helicopter.

2.2.4 Multirotor Designs

Drones have joined other remote controlled vehicles in the world of hobbyists. Of all the different designs the multirotor is the most popular. The four rotor design is generally chosen due to its incredible stability and manoeuvrability. Similar to the tandem, quadrotors have very good disk loading and thus can be used to lift heavy loads, there are even products that have 8 rotors to seriously increase the payload capability. This does however relate to a more power hungry system and a less efficient hover.

As shown in Figure 2.12, a quad rotor consists of two pairs of counter rotating propellers. Each shaft will be driven by its own motor and will contribute to the overall thrust and moment generation of the craft. Having the freedom to control each blade independently gives the pilot advanced manoeuvrability, with minimal mechanical complexities. This also reduces the complexity of the control algorithms as six degrees of freedom can be obtained by simply adjusting the speed of the motors. Besides the poor hover efficiency, the biggest downside of the multirotor designs is their size and weight. Each blade requires a drive system and space to rotate without interference. This generally limits the flight time of multirotors.

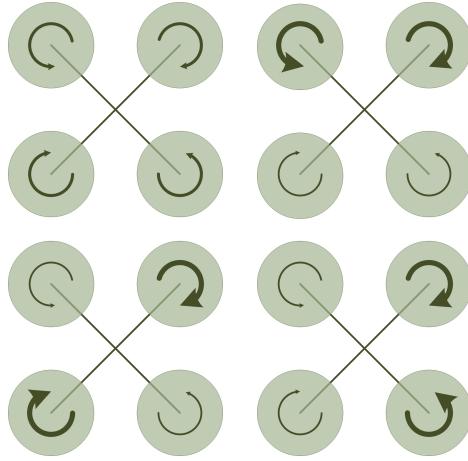


Figure 2.12: Quadrotor configuration [29]

2.2.5 Tilt Rotors

A tilt rotor is a very sophisticated system that attempts to harness the benefits of both the fixed and rotor wing aircraft. With the addition of a pivoting axis for each blade the craft has the forward flying speeds of a fixed wing craft while still being able to take off and land vertically like a rotorcraft. The tilt rotor's major downfall is related to the required highly complex and intricate mechanical design [7].

VTOL applications require a larger blade to decrease the disk loading, while in forward flight a smaller diameter blade is desired to increase the efficiency of propulsion. Hager [14] developed a telescopic system that transforms the blades to get the optimal benefits out of each configuration, shown in figure 2.13⁴. These and other improvements have established the tilt rotor as a competitive design in the field of aeronautic transportation [7].

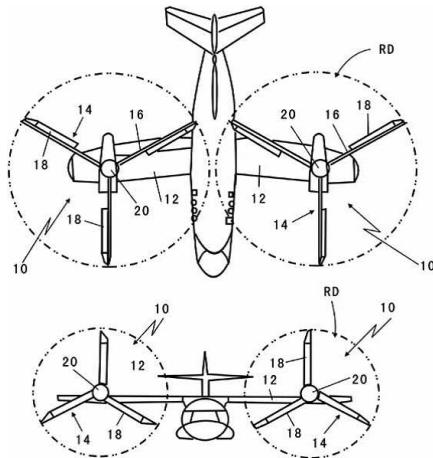


Figure 2.13: Hager's design for a telescopic tilt rotor system [10]

⁴(Taken from [10])

2.2.6 Discussion

2.3 Quadrotor Flight Dynamics

This section will discuss some of the methods and limitations pertaining to modelling the flight dynamics of a rotorcraft. Most of the discussion will surround multirotors, specifically quadrotors, as the majority of the literature is based on these designs [25, 9, 32, 15, 28]. Due to the mechanical complexity of swashplate designs, the discussion is assuming a fixed pitched rotor set up.

Before control laws can be applied there must be a dynamic model of the craft. To create the model there must be a good understanding of the factors that effect these dynamics as well as the mathematical methods for deriving the equations. A brief introduction to the nomenclature and axis systems is done and is followed by a discussion into modelling rotorcraft forces and moments. After the model can be obtained mathematically it is important to discuss the physical implementation of obtaining the data, and the instrumentation required. Unfortunately its very rare to have a flying environment that is void of disturbances, this section is closed with a discussion about the various disturbances that effect the flight dynamics of rotorcraft, including some specific environmental disturbances.

2.3.1 Coordinate Systems, Rotations and Nomenclature

As the rotorcraft manoeuvres through space, two separate sets of axes are created. Each axis system is important and transforming easily between these frames is necessary. Some of these methods are described in this section, as well as the various means of representing these rotations. This section begins by describing these different frames, namely the inertial and body frames.

Inertial and Body Frame

The inertial, or North East Down (NED), aligns itself with the North and East directions on a compass. The third axis will align with gravity as a positive Z component. This frame assumes that the earth is flat and non-rotating and this frame's origin can be defined arbitrarily.

The body frame aligns itself with the body of the drone, with the nose of the craft facing in the positive X direction and the Z axis is defined perpendicular to the rotor plane with thrust generated in a negative Z direction. The origin of the body frame is defined as the center of mass for the drone.

Figure 2.14 is a visual representation of both frames.

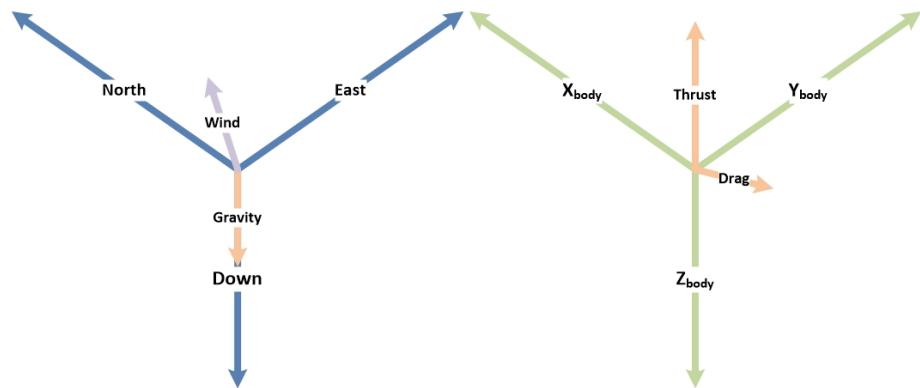


Figure 2.14: The inertial and body frames

In order to relate the motion of the craft in the body frame to the inertial frame, it is necessary to be able to represent the rotation between these frames.

Euler Angles

The most intuitive way to represent the rotation between two frames, is by looking at the rotation between each corresponding axis. These are known as the Euler angles and are made up of a roll (ϕ), pitch (θ) and yaw (ψ) angles. Euler angles provide a very intuitive understanding of the rotation between the different frames. This is best explained with Figure 2.15.

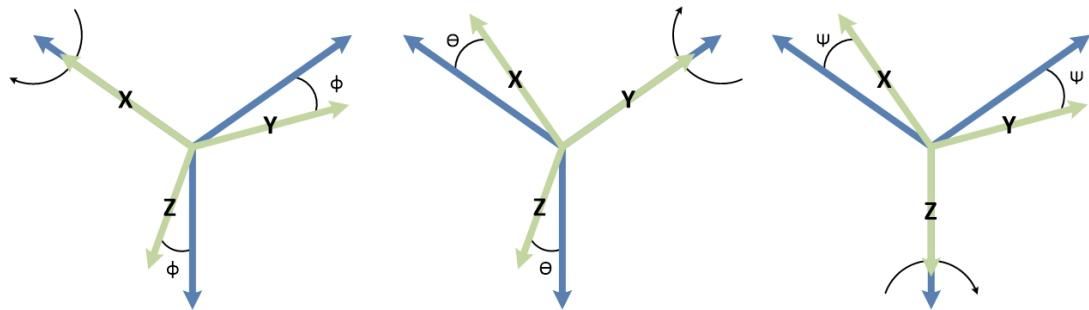


Figure 2.15: Individual rotations around the X, Y and Z axes respectively.

The yaw angle is defined as a pure rotation around the Z-Axis. Roll and pitch are defined as pure rotations around the X-Axis and Y-Axis respectively. The Euler angle representation does have limitations, such that any 3 Euler angles could represent a different rotation, based on the order it is applied. For this project, a Euler 3-2-1 sequence will be followed. There is also a chance of a singularity at extreme angles, this is not a concern for this project, as it will only be necessary to ever complete small rotations [13, 28].

According to Euler's theory, any two varying coordinate axes can always be related to one another by a single rotation.

Direct Cosine Matrix

The direct cosine matrix (DCM), provides a simple method for transforming references between two different frames. This is necessary for converting the NED frame to the body frame and vice versa. The DCM is calculated by following 3 individual rotations and multiplying their results together. A 3-2-1 Euler sequence will transform first using yaw then pitch and finally roll. Each transformation is represented as a 3x3 Matrix representing a rotation around one of the axes.

In the case of rotating from the body to the NED frame, the matrix takes the form as shown in equation (2.18) [25, 28] where $C_x = \cos(x)$ and $S_x = \sin(x)$. The matrix is also orthogonal, which means that $\mathbf{R}^{-1} = \mathbf{R}^T$. \mathbf{R}^T would be the rotation from the inertial frame to the body frame [25, 28, 13].

$$\mathbf{R} = \begin{bmatrix} C_\psi C_\theta & C_\psi S_\theta S_\phi - S_\psi C_\phi & C_\psi S_\theta C_\phi + S_\psi S_\phi \\ S_\psi C_\theta & S_\psi S_\theta S_\phi + C_\psi C_\phi & S_\psi S_\theta C_\phi - C_\psi S_\phi \\ -S_\theta & C_\theta S_\phi & C_\theta C_\phi \end{bmatrix} \quad (2.18)$$

The DCM does provide a mathematically simple method for creating relationships between frames, however this method is computationally taxing as it is forced to recalculate the matrix and the multiplications on every loop.

Quaternions

The quaternion representation manages to minimise the computation required to calculate transformations, as well as remove the singularity found in the Euler representation [13]. One of the major downsides of quaternions is that they are difficult to interpret intuitively. A quaternion follows the form seen in (2.19) and contains a scalar value q_w and a vector component $[q_x \ q_y \ q_z]$. This representation is broken up into a rotation angle, and a rotation axis.

$$\mathbf{q} = \begin{bmatrix} q_w \\ q_x \\ q_y \\ q_z \end{bmatrix} \quad (2.19)$$

Quaternions come with their own set of mathematical rules and laws which will not be discussed here. However it should be noted that there are techniques that provide simple conversion from and to Euler angles and thus the DCM.

Nomenclature

The naming convention used, follows Moller's notation [28] and is shown in Table 2.1. It makes sense that the global position and velocity of the craft be described in the NED frame, however the forces and moments will be generated in the body frame. Since there is now a simple relationship between the two frames, it is possible to relate the body frame forces and movements, into earth frame translations. The variables shown in Table 2.1 are visualised in Figure 2.16. The variables are all defined in the body frame and are shown, along with their positive directions. The right hand and thumb rule were used to dictate direction.

X, Y, Z	Force vector along the respective body frame axis
L, M, N	Moment around the respective body frame axis
U, V, W	Linear velocity along each body frame axis
P, Q, R	Angular velocity around each body frame axis

Table 2.1: Standard Nomenclature

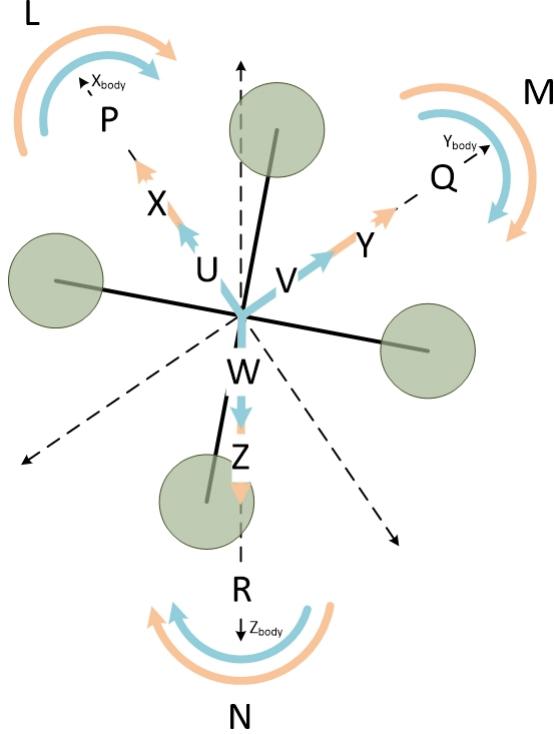


Figure 2.16: Typical naming convention of body forces, moments and velocities for a quadrotor

The body frame forces, moments and velocities can be seen, and are described in (2.21) - (2.26). Where X, Y, Z are the forces in each body axis, with the rotor thrust being produced in the negative Z direction. L, M, N are the moments around the x, y, z axes respectively and U, V, W are the velocities aligned with the x, y, z axes respectively.

Using the rotation matrix described in (2.18), a relationship for North, East and Down velocities can be made and is described in (2.20).

$$\begin{bmatrix} \dot{N} \\ \dot{E} \\ \dot{D} \end{bmatrix} = \mathbf{R} \begin{bmatrix} U \\ V \\ W \end{bmatrix} \quad (2.20)$$

2.3.2 Kinetics and Kinematics

Assuming the system can be considered as a rigid body, allows for the use of normal Newtonian mechanics to create the equations of motion. This method will also use the Euler angles described above [25, 23, 28].

The derivations of these calculations are well documented in literature [3]

$$X = m(\dot{U} - VR + WQ) \quad (2.21)$$

$$Y = m(\dot{V} - UR + WP) \quad (2.22)$$

$$Z = m(\dot{Q} - UQ + VP) \quad (2.23)$$

$$L = \dot{P}I_{xx} + QR(I_{zz} - I_{yy}) \quad (2.24)$$

$$M = \dot{Q}I_{yy} + PR(I_{xx} - I_{zz}) \quad (2.25)$$

$$N = \dot{R}I_{zz} + PQ(I_{yy} - I_{xx}) \quad (2.26)$$

2.3.3 Mass Model and the Inertia Tensor

Mass Model

In any aerial vehicle mass is always an important design criterion. Every aspect of the platform must be designed to be the lightest it possibly can. Having a light weight craft is one part of the design criterion, another would be ensuring that the weight is geometrically spread out correctly, as well as functionally distributed appropriately. The table below was adapted from [38] and demonstrates the latter point by showing the weight distribution of three separate crafts. Depending on the different criteria for the craft, different functional blocks will be allocated a certain percentage of weight. For example if the project requires a longer flight time, a higher percentage would be given to the power source and possibly less to the external payload. Generating a good mass model before designing helps better understand the requirements for the craft and could be a deciding factor in the construction.

Component	0.3kg	1.8kg	3.7kg
Rotor System	11.0	11.2	13.9
Tailboom Assembly	8.0	9.1	7.8
Main Rotor Motor	15.4	10.5	8.1
Fuselage/Structure	7.0	15.1	12.0
Main Transmission	2.0	3.4	3.4
Landing Gear	2.3	3.4	2.9
Control System	5.7	18.3	9.3
Avionics	29.4	2.4	1.6
Power Source	19.2	26.6	41.0

Table 2.2: MAV Weight Data (Adapted from [38])

Inertia Tensor

It was also mentioned that the weight needs to be geometrically positioned correctly, the point of this would be to create as much symmetry in the craft as possible. If this is done correctly the principle axes of inertia will align very closely with the body of the craft, simplifying calculations later on and helping find and define the principle axes. The inertia tensor is a matrix that is a representation of a rigid body's resistance to rotations in 3D space. For the general case the inertia tensor takes the form as shown in equation (2.27). The inertia tensor is very dependant on a craft's symmetry, and is symmetric itself. In other words, $I_{xy} = I_{yx}$, $I_{xz} = I_{zx}$ and $I_{zy} = I_{yz}$ and therefore if a

craft is symmetric about the X, Y and Z axes, then the assumption can be made that $I_{xy} = I_{yx} = I_{xz} = I_{zx} = I_{yz} = I_{zy} = 0$ [25, 8].

$$\mathbf{I} = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{yx} & I_{yy} & -I_{yz} \\ -I_{zx} & -I_{zy} & I_{zz} \end{bmatrix} \quad (2.27)$$

In order to successfully model a rotorcraft, the inertia tensor must be known and will be defined around the centre of rotation of that rotorcraft. The method for obtaining the inertia tensor is described in the system identification of this project.

2.3.4 Rotor Generated Forces and Moments

The forces and moments generated by the rotors are discussed here. It is assumed that the rotors will only generate a force perpendicular to their plane while the moments are dependant on the placement of the rotors.

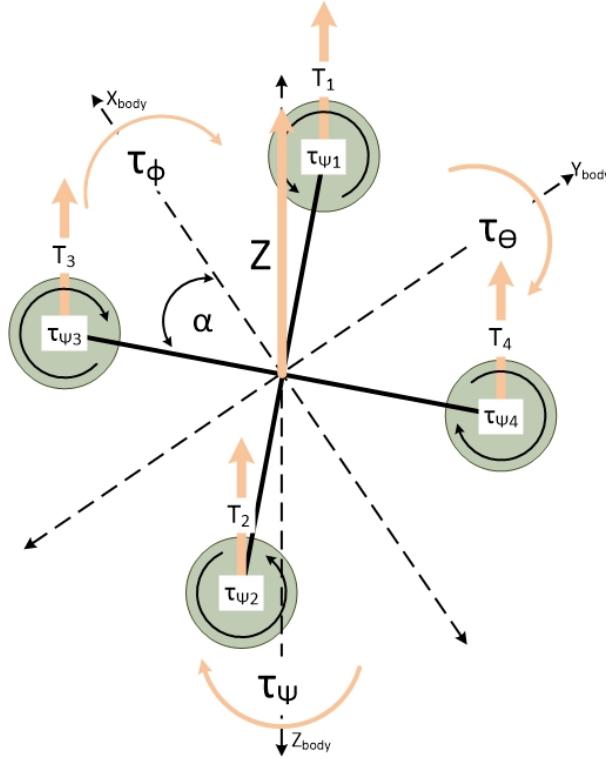


Figure 2.17: Forces and moments acting in the body frame on an X-Configuration quadrotor

Actuators

As shown in Figure 2.17, all the forces generated by the quadcopter are a product of the four rotors. The rotors convert mechanical energy from the motors into aerodynamic power. The motors convert electrical energy into mechanical energy based on the motor commands sent from the controller. Both the motors and rotors can not react instantly to new commands, this lag introduces a timing delay constant into the system [28].

If the lag timing constant is defined as τ , thrust generated by motor x as T_x and the command sent to that motor as T_{xR} . Then (2.28) can be created and applies to all four motors.

$$T_x = -\dot{T}_x \tau + T_{xR} \quad (2.28)$$

Controlled Body Forces

Figure 2.17 assumes that all of the rotors lie in the same plane, and only provide a unidirectional force. This assumption allows the easy creation of a total force Z , which is shown in (2.29) as the sum of all four motor thrusts.

$$Z = (T_1 + T_2 + T_3 + T_4) \quad (2.29)$$

To command this value, a virtual actuator can be created δ_Z which commands all four rotor thrusts. Equation (2.28) demonstrates the lag to generate these thrusts and the same lag dynamics will apply to δ_Z , thus creating (2.30).

$$Z = -\dot{Z}\tau + \delta_Z \quad (2.30)$$

Controlled Body Moments

A quadrotor generates a moment around it's own axis through varying the speed of each motor. The torque generated is also dependant on the spacing for the type of quadrotor used. A standard X-Configuration quadrotor is shown in Figure 2.17 and was used for this analysis. To induce a torque around the X-axis, the sum of the two left rotors subtracted from the sum of the two right rotors must be non-zero. Similarly the front and back rotor summations must not be equal to induce a torque around the Y-axis. As shown in 2.17, each rotor also creates a moment around the Z-axis. This induced torque is a product of the rotors lift to drag ratio and the chord length and is represented in (2.31).

$$\tau_{\psi x} = \frac{r_D}{R_{LD}} \times T_x \quad (2.31)$$

Assuming that each rotor has the same characteristics and are spaced evenly, these moments can be mathematically expressed as shown in (2.34), where l is the distance from the centre of the rotor to the centre of gravity, r_D is the chord length and R_{LD} is the lift to drag ratio for the rotors.

$$L = \frac{r_D}{R_{LD}} \times (T_3 + T_4 - T_1 - T_2) \quad (2.32)$$

$$M = (T_1 + T_3 - T_4 - T_2) \times l \cos(\alpha) \quad (2.33)$$

$$N = (T_2 + T_3 - T_1 - T_4) \times l \sin(\alpha) \quad (2.34)$$

Virtual actuators can be created to command these moments, namely δ_ψ , δ_θ and δ_ϕ . However these commands will be subject to the same time delay experienced by the rotors. Therefore (2.36) - (2.37) can be used to represent the relationship between these commanded values and the actual moment [23, 28].

$$L = -\dot{L}\tau + \delta_\psi \quad (2.35)$$

$$M = -\dot{M}\tau + \delta_\theta \quad (2.36)$$

$$N = -\dot{N}\tau + \delta_\phi \quad (2.37)$$

2.3.5 Disturbances

Drag

Drag is a damping force whose quantity is relative to the speed of the object, and always opposes the direction of motion. Drag is defined here in the body frame and from (2.2) and the discussion drag, the equations for three dimensional drag can be calculated. As shown in (2.38) - (2.40), the effect of drag can be reduced through mechanical design and flight strategy, by reducing the area of the plane facing towards the direction of the motion.

$$F_{Dx} = C_{DX} \left(\frac{1}{2} \rho U^2 \right) A_{YZ} \quad (2.38)$$

$$F_{Dy} = C_{DY} \left(\frac{1}{2} \rho V^2 \right) A_{XZ} \quad (2.39)$$

$$F_{Dz} = C_{DZ} \left(\frac{1}{2} \rho W^2 \right) A_{XY} \quad (2.40)$$

Due to an offset between the centre of gravity and the centre of pressure, the drag forces can also create undesired moments. Equations (2.41) - (2.41) can be derived from the Figure 2.18, where d_x, d_y, d_z are the offsets of the centre of pressure. F_{Dx}, F_{Dy}, F_{Dz} are the forces generated by drag act along the coinciding body axis. M_{Dx}, M_{Dy}, M_{Dz} are the moments generated by the drag forces and the offset of the centre of pressure, they act around the coinciding axis. A_{YZ}, A_{XZ}, A_{XY} are the surface areas facing the corresponding plane in the body frame with C_{DX}, C_{DY}, C_{DZ} as the corresponding drag coefficients.

$$M_{Dx} = F_{Dz} \times d_y - F_{Dy} \times d_z \quad (2.41)$$

$$M_{Dy} = F_{Dx} \times d_z - F_{Dz} \times d_x \quad (2.42)$$

$$M_{Dz} = F_{Dy} \times d_x - F_{Dx} \times d_y \quad (2.43)$$

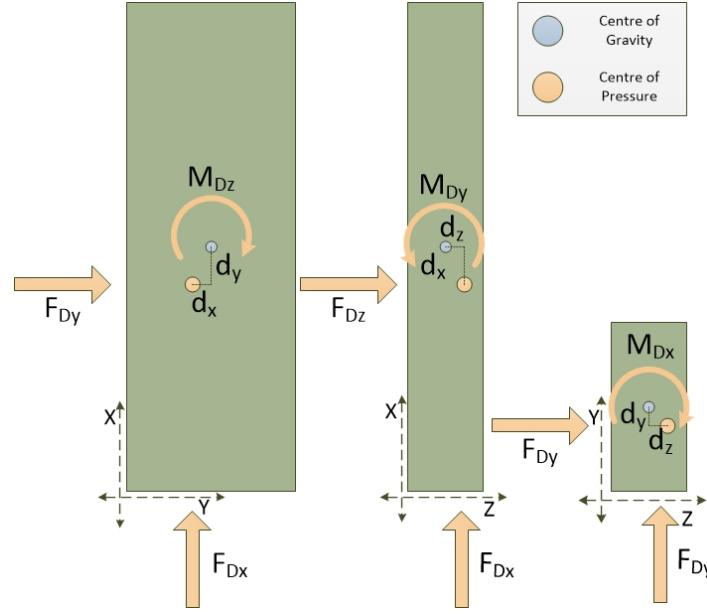


Figure 2.18: Typical moments created by drag forces

Airflow Characteristics

In the preceding section on flight theory, the importance of air density, pressure and the creation of rotor wake boundary are discussed. The negative effects of disrupting airflow as well as the need for controlling this disturbance has been well documented in literature [36, 20, 15].

Using Figures 2.4 and 2.5 as references Airflow can be seen as the stream of air from v_0 to v_∞ through v_i . Equation 2.11 states that thrust is directly proportional to the relationship between these velocities and any deviation in these velocities will vary the thrust of the rotor in question. v_0 is only zero when the craft is in a state of pure hover, completely stationary, and there is no wind. Increasing the speed of the craft will increase the v_0 component creating a variation in the overall thrust, the same can be said for any condition that contains a tangible wind factor.

As investigated by [15] mechanical intrusions will have an effect on the far wake velocity, thus also effecting the generated thrust. In the design of STARMAC by Hoffmann et al [15] the frame was designed to be very configurable so that the effects of the mechanical design could be quantified. Originally the rotors were shrouded and quite close to the centre of gravity of the craft. The shrouds were a distance of 5% rotor radius and when removed the yaw tracking improved from $\pm 10^\circ$ to $\pm 3^\circ$. When not included in the dynamic model the obstruction in the air stream will cause lower and less stable values of thrust, affecting the accuracy of the model.

When the rotors were located close to the centre of gravity they obtained some attitude disturbances that were eliminated by moving the rotors further away. It was also observed that any object that lies close to the rotor tip, created intense arbitrary disturbances and should be avoided [15].

[36] attempts to model some of the disturbances for a single rotor craft hovering near wall, but as stated by [20] it is not viable to accurately quantify these disturbances, however their presence must not be neglected. As Figure 2.19 shows, these can be modelled as a disturbance to the input force and moments.

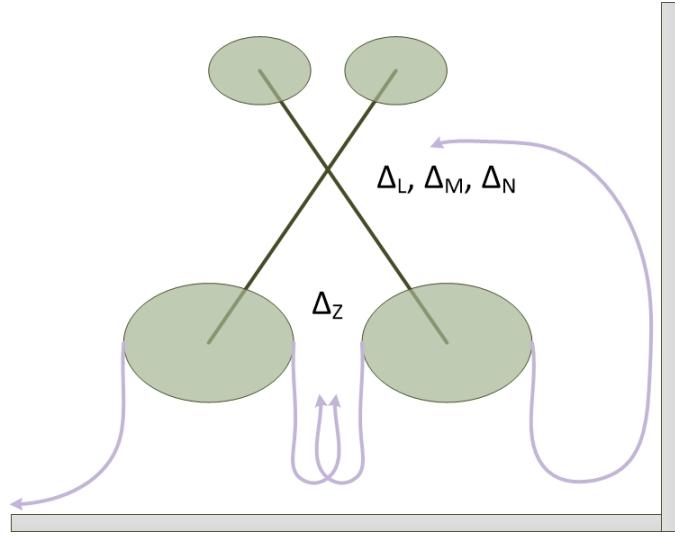


Figure 2.19: Velocity components though the rotor for, no wall (left) and near wall (right) conditions (Taken from [36])

As demonstrated by [36], there is also an induced moment acting on the rotor as the rotor approaches the wall. Figure 2.20 is an image generated by [36], it demonstrates the change in airflow on a rotor close to a wall region.

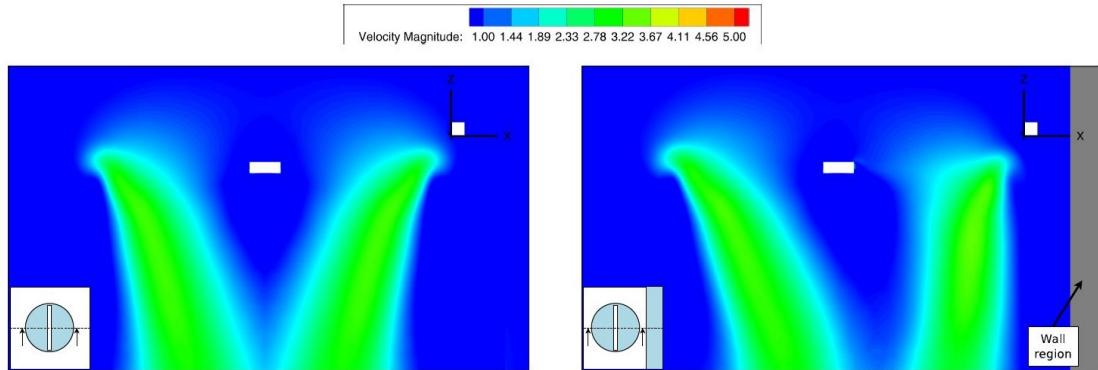


Figure 2.20: Velocity components though the rotor for, no wall (left) and near wall (right) conditions (Taken from [36])

In [36], Robinson et al used the script c as their unit of measure for distance to the wall, c is chord length of the airfoil. The graph shown in Figure 2.21 shows how the moment felt by the craft varies with the distance to the wall. The direction of the moment will be perpendicular to the wall. As above, this disturbance can also be modelled as a variation to the input moments to the system.

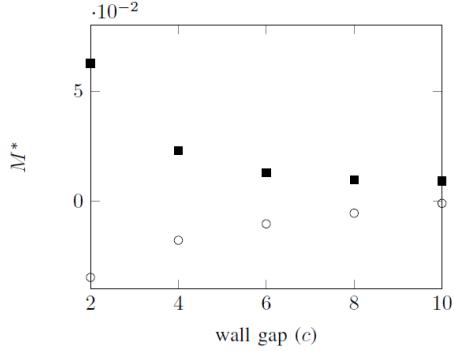


Figure 2.21: Graph showing relationship between distance from the wall and moment felt by the craft (Taken from [36])

In [36] they conclude their paper by describing a proposed method of control which will be investigated further in this text. This dynamic flight model requires sensing of specific craft variables, typical sensing methods and requirements are discussed in the proceeding section.

2.3.6 Instrumentation

The need for a well instrumented craft is intuitive and well documented in literature [28, 15]. With modern advancements in sensing technology there is now a number of methods to obtain the required information. However, some of these are very costly and require specific operational environments. For this purpose, only the inertial measurement unit (IMU) and the global positioning system (GPS) are discussed here.

Inertial Measurement Unit

Traditionally an IMU will consist of an accelerometer, gyroscope and possibly a magnetometer. The accelerometer has the ability to measure accelerations in the body axis. It should be noted that an accelerometer in rest, sitting along the inertial axis will provide a reading of $[0 \ 0 \ g]^T$. The gyroscope measures the rotational accelerations of the craft, thus measuring 6 degrees of freedom in total. Unfortunately gyroscopes suffer from sensor drift and needs compensation. The accelerometer's gravity offset can be used to align with the vertical axis, however a magnetometer is required to account for drift in the horizontal plane. Various filtering and fusion techniques are used to combine these readings, the most popular of which is a Kalman Filter [28, 15].

Global Positioning System

The most common method of measuring a drone's position is with the use of a GPS. The GPS readings can also be used to create an estimate of the craft's linear velocity in the inertial frame. When combined, multiple GPS unit's can be used to obtain higher precision. A major downside of a GPS is the dependence on a satellite link, this dependence severely limits the operational environments for this sensor. Other methods of localisation include using known maps and proximity sensing to provide the rotorcraft with position estimates. As discussed, these techniques are out of scope for this project and

will not be considered. This project will assume an earth position and velocity estimate are present, subject to filtered noise provided through band limited white noise [28]

Measurement Noise

From the discussion it is evident that sensors will not be without error. The noise associated with the measurements will be different for both varying sensor types and manufacturers and will depend on the chosen sensors. The characterisation of the noise for this project is discussed in the system identification chapter. The environment in which these sensors operate also has a significant effect on operation. Moller characterises this measurement noise as a random band limited white noise block. and adds a low pass filter to the GPS measurements [28].

2.4 Review of Existing Flight Control Strategies

The object of this section is gain a better of understanding of successful controller designs and implementations. A talented pilot could fly a rotorcraft with minimal assistance from an on board controller, however to successfully create an autonomous vehicle capable of flight in a confined environment, a robust controller design is required. The discussion is broken into two sections, first a look into high level controller architectures is done after which a look into some successful real world implementations is investigated. The different axis systems used are discussed in a previous section and transition between these axes becomes a crucial part of the controller architecture.

2.4.1 Controller Architecture

Traditional tracking control for a rotorcraft is distributed between three controllers, namely an altitude, heading and horizontal controller [28, 27]. The specific loop structure is determined by which feedback is available to close the loops. These loops are made up of combinations of traditional control laws such as PID controllers and Lead-Lag compensators.

The altitude controller is straight forward, dynamics are derived directly from Newton mechanics. A traditional structure is the use of three cascaded loops. The most inner loop can control the linear acceleration of the craft in the body frame and it's loop can be closed by use of an accelerometer. The next loop is used to control the climb rate of the craft in the earth frame, with the final loop controlling the final position of the craft. The altitude loop is closed by some form of altimeter, traditionally a barometer can be used to estimate altitude. For more precise altitude control a sensor can be used to give a relative height above the ground. The velocity loop is generally closed by some approximation of velocity through differentiation of absolute position. In some cases optical flow sensors could be used to obtain a velocity measurement directly.

The heading controller is responsible for controlling the orientation of the craft. Traditional quadcopters can fly omni-directionally irrespective of their heading, allowing for less stringent design requirements on this controller. Orientation is traditionally described as a Euler angle off of a given axis obscuring the axis systems and need for rotations. The heading controller can be simply broken into two loops, an inner rate loop and an outer angle loop. Both of which can be closed by the use of an IMU combining magnetometers and gyroscopes.

The final and most complex controller structure is the horizontal control. This is the only controller that is required to control both angular and translational control loops introducing the need for inferring desired angles from a given translational reference. The controller generally is also required to work in both axis systems, requiring some form of geometric transformation. The translational portion of the controller is split up into a North and East controllers, while the inner angular system is broken up into roll and pitch controllers. The translational portion is further broken down into an outer position loop which feeds setpoints to an inner linear velocity loop. Both of these loops can be closed by a GPS unit. The angular portion of the controller is then also split into a inner rate loop which is potentially closed by a gyroscope and an angular position loop which utilises feedback from an IMU.

Tracking control is used to obtain stable flight and reject disturbances and unmodelled errors. However certain disturbances might hinder the performance of these controllers and produce instability that traditional control law can not account for without negatively affecting the tracking control. Feed forward control could be utilised if the disturbances can be predicted and their effects have been quantified, this however is rarely the case.

Disturbance Based Observer Controller

Another method of rejecting extreme disturbances, while limiting the adverse effects on the tracking control, is to estimate the disturbance and then control based on that estimation. In an investigation of the near wall effects, Robinson et al propose a disturbance based observer which can be used to predict and counteract these disturbances [36, 35]. Figure 2.22 outlines the structure of such a controller, and demonstrates how it is interfaced with traditional tracking control. The main benefit of this structure is the loop is only activated when the state estimations do not match the state outputs.

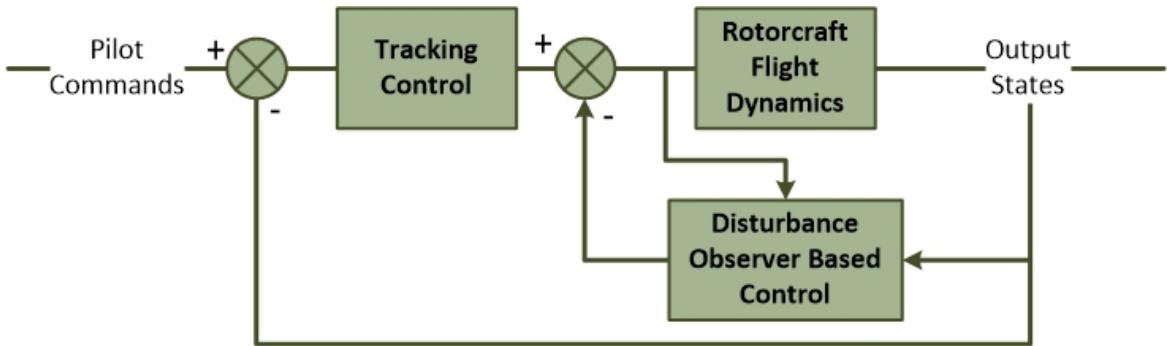


Figure 2.22: Disturbance Observer Based Controller Structure

For a system like this to be effective, the observer will need to measure the disturbances in real time requiring fast computation. Lee et al implemented an adaptive control strategy which uses an IMU and a Vicon system to estimate the disturbances and feeds them into the system [20]. This requires a complex lab set up and can not be used in any environment, however their experiments did validate the effectiveness of such a system.

2.5 Collision Protection Techniques

Collision protection is an addition to any platform that will assist during an impact. An advanced enough collision protection technique could negate the risk of impact in a

confined environment. Any obstruction in a rotor's path will either destroy the part, or stop it from spinning thus disabling the thrust generation capacity of that rotor. Collision protection in its simplest form is a simple shroud that can protect the rotors.

2.5.1 Impact Resistance

Impact resistance is a technique used by a variety of fields in the world today. Included in this is something as simple as the shocks or suspension in a car, they are designed to allow the automotive to withstand sudden impacts. Generally these techniques use a component that has some tangible spring constant that can be used to absorb the energy felt during an impact.

Klaptoz et al investigated such a system for an aerial vehicle [18]. Using a combination of drop tests and forced collisions it was validated that an impact resistance shroud could limit the force felt by the platform. Light weight materials can be used to create the design however there inevitably will be additional weight and size added to the platform.

2.5.2 Rolling Cage

Although impact resistance can limit the effect of a collision Briod et al set out to create a system truly robust to collisions [5, 6, 19]. Using a coaxial rotorcraft they developed a system whereby, using a three axis gimbal, an external shell will passively rotate around the craft on impact. Thus almost negating the effect of the collision completely. Although a proven solution to confined indoor flight, the system is optimised for a coaxial rotorcraft limiting the payload and flight time capabilities of the design.

2.6 Collision Avoidance Techniques

Some protection techniques were described above to assist the craft in the event of a collision. However, these techniques add weight, complexity and in some instances obstruct potential sensors. Near wall disturbances have also been discussed and can severely impact a rotor craft flying in a confined space. Using the more simple protection techniques, it is still desirable to avoid collisions all together. Collision and obstacle avoidance sensing techniques can also be used to aid in a higher route planning algorithms. For successful collision avoidance, the craft must be able to collect data about its surroundings. This section begins by describing some of these sensing techniques and finishes by investigating some of the methods for utilising this data.

2.6.1 Sensing Techniques and Requirements

Traditionally the collision avoidance sensing techniques can be broken into two main categories. Namely some form of proximity measurement can be done, or a higher level more intelligent image processing system can be applied [12]. This discussion analyses each of these hierarchies separately.

Proximity Measurement

The first and most common would be using a time of flight (TOF) sensor, such as an ultrasonic transducer. Very similar to bats, a transmitter module will emit a pulse and

based on the time it takes for the signal to return, the distance can accurately be measured. A sonar is dependant on a lot of variables and would require calibration for the environment it is in. Due to the method with which the modules acquire information the drone will receive a speed limitation. Since the sonar is dependant on the density of the medium it travels through, disturbance created by the rotors could severely affect the performance of the sensors.

Another option in the same category would be to use an infra red (IR) transceiver in the same configuration. The system uses strobed IR pulses to monitor the distance of an object, the system depends on the ratio of reflection for the IR spectrum. If an object has a high refraction or absorption ratio the signal can get lost. Ambient light can also cause interference, although most modules will have some form of built in filter. Both these methods are also sensitive to the angle at which they are measuring.

Using the same base technology TOF cameras have been developed such as Microsoft's Kinect. Instead of a single point, a TOF camera projects an IR grid into the area it wishes to explore. By measuring deviations in the expected grid structure, distance information can be inferred. In an IR rich environment, TOF cameras can get overwhelmed and produce unreliable results. Due to the projection, the power requirements and weight of such a system can also be high.

Image Processing

The term image processing has been used here to discuss a system that can extract data through a video feed. The process is often referred to as machine vision and can be trained to detect objects present in the image feed. A general method is to do edge detection, allowing the craft to differentiate between objects and safe flight zones. These configurations generally require a lot of processing power and can add considerable weight and electrical power requirements to an aerial vehicle.

Although more processing is required, very good results can be obtained from this set up. Generally these systems are slow and have a high power draw due to the multiple required modules. However, reliability from a robust image processing unit will be more reliable and consistent than a different system. Barry et al developed a system that minimises some of these negative effects [2]. Using this design MIT could obtain object avoidance at speeds of 31mph, proving the effectiveness of the object avoidance system.

It is also well known that a camera feed can provide other advantages, such as allowing the pilot to fly without line of sight. Stereo vision can also produce 3D information from multiple 2D images, creating useful data for the operators. Optical flow algorithms can also be applied to a camera feed to estimate the current velocity of the craft. A major downfall of optical systems is the requirement for well illuminated areas. A dimly lit area, or even a dust filled environment could produce faulty results hindering the performance of the collision avoidance system.

2.6.2 Collision Avoidance Algorithms

The theory behind obstacle avoidance algorithms is well documented and understood in literature. An obstacle avoidance routine will allow the craft to deviate off of the planned motion in order to avoid collisions. A successful implementation will attempt to limit this deviation as much as possible. Some of the main considerations for these algorithms is the amount of computational power required and the speed at which these computations

can be done, as well as the required sensing information.

Some of the typical sensors used in collision avoidance have been described above. A combination of these sensors can also be constructed to give even more robust and reliable results. The investigation into existing collision avoidance algorithms can be limited by proposing a chosen sensing technique. Due to the complexities involved in image processing implementations, this work will only investigate techniques using traditional proximity measurements.

Bug Algorithm

The bug algorithm was developed by Lumelsky in [24]. The principle behind the bug algorithm is to allow the robot to deviate off of the desired path based on immediate proximities to objects. Thus not requiring any previous knowledge of the surroundings. The robot will allow itself to contour around an object while still attempting to get back onto the correct path. Once the path is found, the robot will continue along, until another object is detected.

This method is useful, but limited. The robot will not attempt to optimise the route and this can be an inefficient method of solving the problem. The constant requirement for scanning the total environment and creating new paths can be time consuming and computationally costly.

Subsequently the bug algorithm has been developed further and multiple variations in the principle have been adapted. These adaptations are based on the bug principle, however utilise more advanced contour creation and state transitions, allowing for more versatility and optimised route planning. Figure 2.23 is an image showing the Bug2 algorithm at work with the craft being commanded from location S to location T. As shown, when the vehicle is in close proximity to an object it will create a contour around it until the desired path is rediscovered.

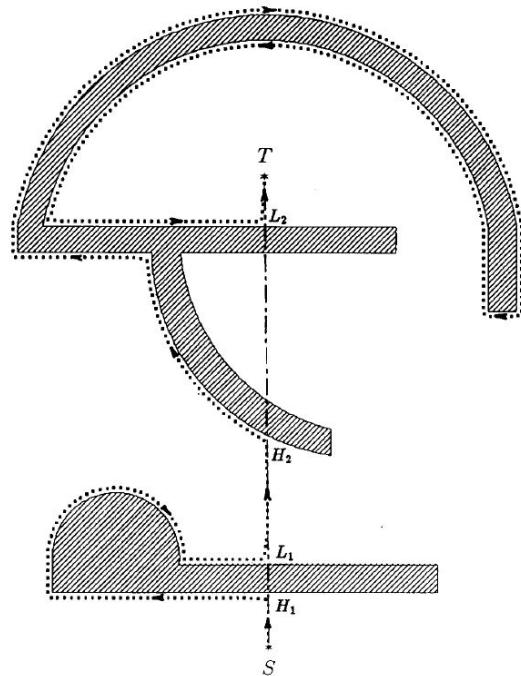


Figure 2.23: Demonstration of the Bug2 algorithm (Image taken from [24])

Potential Field Method

Another widely used and documented algorithm is the potential field method [34]. This method works on the basis of creating attractive potentials, in the form of goals and unattractive potentials, which can be the obstacles or no fly zones. A simple way of describing this method is to make a comparison with a charged particle moving through a potential field. In this case, the aircraft is considered a positive charge and the end goal or position is considered a negative charge. The goal will provide an attraction force based on the robot's proximity to the end point. Obstacles, in this analogy, can then be thought of as other positive particles which will produce a negative, or repelling force to the platform. The algorithm will then calculate the attractive and repelling forces based on immediate sensing data. The sum of all of these forces can then be given to the craft and used as a command to allow for successful navigation in an unknown environment.

An important consideration is the generation of these forces based on simple proximity measurements. Traditionally there will be a single goal which will attract with a parabolic like force, that increases quadratically as the robot moves further away from the goal. This set up will also ensure that the force felt by the robot decreases as the setpoint is approaching, limiting overshoot in the final system. Although the platform can expect one goal, there might be multiple objects in it's vicinity all creating repelling forces. The repelling force is then calculated by summing all of these forces.

A major limitation of the potential field method is the assumed holonomy of the platform. A non-holonomic vehicle will struggle to implement a potential field method as the direction of the induced forces may not be reachable for the platform in given scenarios. Another downfall of this method is the local minima created when the sum of repulsion forces equals to the sum of attractive forces. Figure 2.24 is used to illustrate a scenario where this local minima is created. Careful sensor placement can limit the occurrence of these situations, however it will be virtually impossible to remove this risk completely.

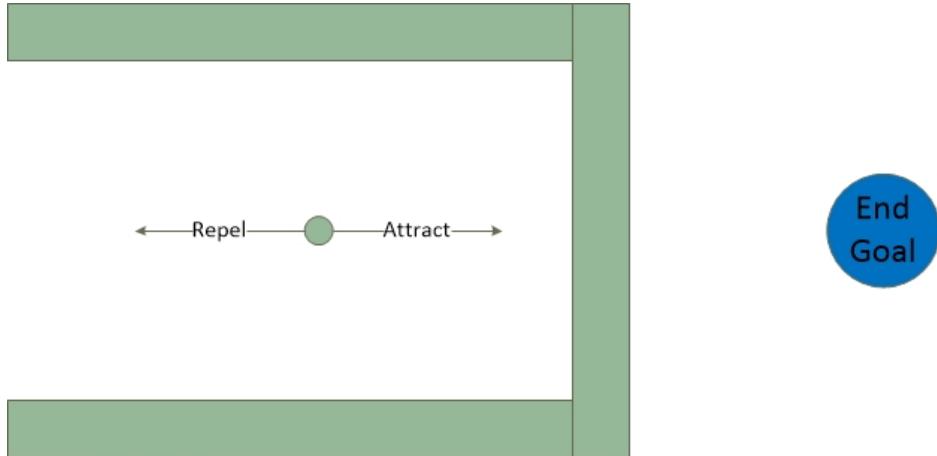


Figure 2.24: Local minima seen with potential field method

Chapter 3

Platform Design

This chapter contains the details of physically a quadcopter for use in a confined space. The hardware design decisions are preceded by giving an overview of the total system design. After the high level design has been done, this chapter looks into different platform designs and rotor configurations. Based on analysis of empirical data and knowledge learnt through an analysis of conventional rotors in Section 2.2, the best suited rotor configuration is identified. Once a suitable platform has been chosen, a look into the required electronics is done. This includes all the electronics required for stable flight, as well as sensor requirements for operations and flight strategy. The chapter is concluded with a summary of the platform decisions made and used in the simulation.

The final design will be modelled and used to validate the proposed outcomes for this work. Although this chapter will discuss the various aspects of a quadrotor, it will focus on the components required for creating an accurate simulation of the system.

3.1 System Hardware Architecture

The following section gives an overview behind the system design of CEPAD . The section begins by working through the hardware system architecture shown in Figure 3.1 and identifying the objectives and roles of each subsystem in the design. The first consideration is the size and mounting needs of the system, provided by the mechanical platform. This gives a good start to elaborate on the needed capabilities of the proposed method of propulsion. The next considerations are the electronic modules required to run all the necessary peripherals and obtain all the required sensor data.

The system architecture for CEPAD is laid out below in Figure 3.1. The hardware system comprises of 3 main subsystems, namely Mechanical Construction, Propulsion and Electronics. The role each subsystem plays in meeting the system requirements is discussed in more detail below.

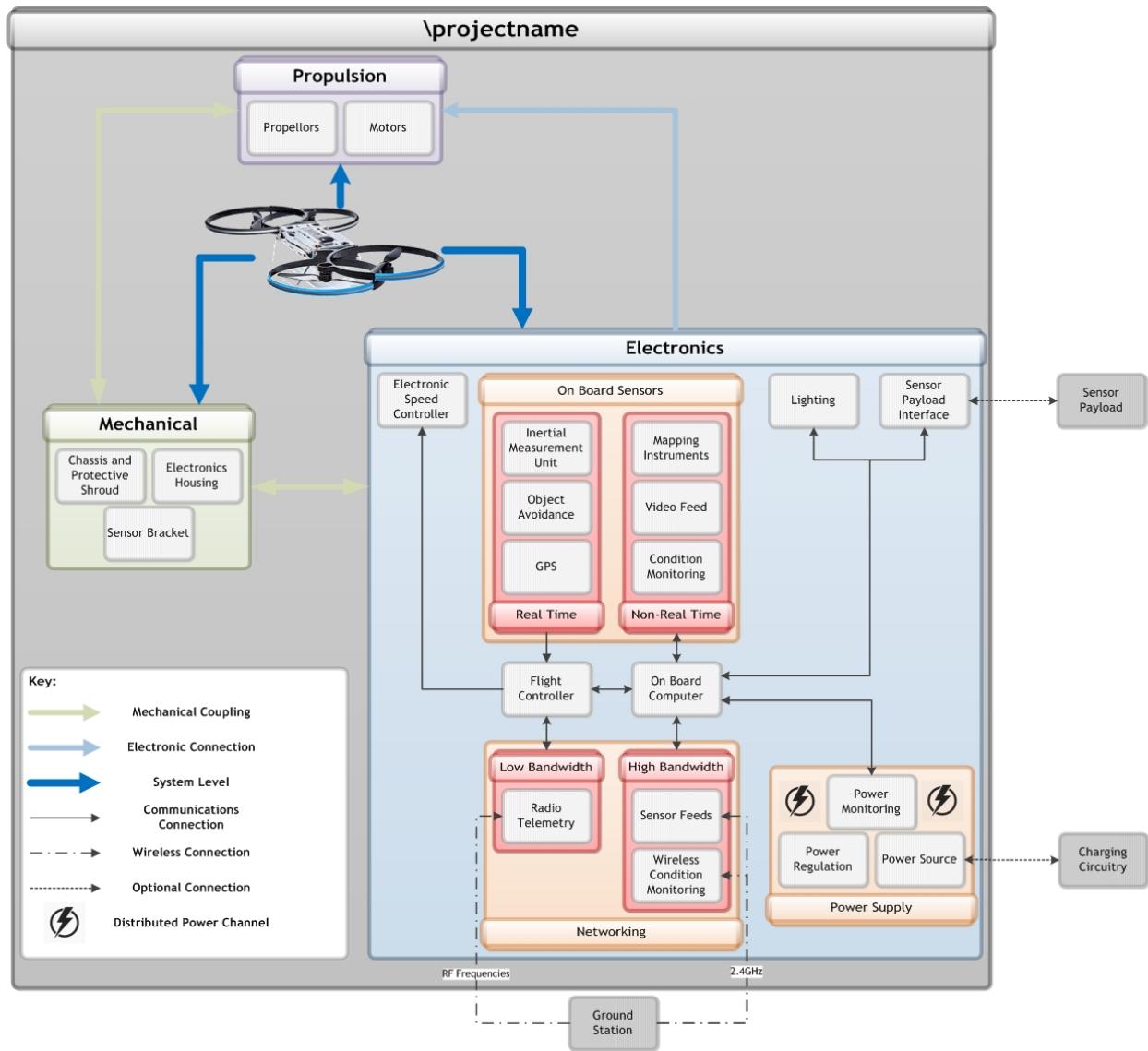


Figure 3.1: System Architecture

3.1.1 Platform Construction

The platform is the physical embodiment of the craft. In this context this includes the mechanical construction of the craft, as well as the flight generating components.

Mechanical Construction

The mechanical construction of the system introduces size and weight restrictions for the rest of the system. It should provide adequate mounting for the necessary peripherals described in the sections following, while remaining as light weight as possible. The structure needs to protect the rotors from collisions by means of a protective shroud and include lightweight landing gear. The chassis must also be such that the weight is distributed as symmetrically as possible. The propulsion system must be housed and provided with rigidity for steady flight control.

Propulsion and Flight Characteristics

The propulsion system comprises of the motors and propellers. This subsystem needs to supply enough thrust and overhead for steady control, as well as enough lifting capacity to carry any additional, mission sensors. The craft should be able to handle an external sensor payload. To ensure capable disturbance rejection, the total maximum thrust should include sufficient overhead above the hover requirement.

The craft must efficiently hover and fly at low speeds. The craft should have multiple control surfaces to accurately counter disturbances. Due to the intended use case in narrow corridors, the craft will fly with a set orientation and thus does not require the abilities for sharp turns of manoeuvres.

3.1.2 Electronics Interface

At the core of any reliable robotic platform is a thoroughly planned electronics system, for CEPAD this is the most complex subsystem containing the most modules. It is responsible for all intelligence and power generation. This section breaks this subsystem into more discrete parts. It begins by separating the two computing modules into real time and non-real time components. The real time components include some of the on board sensors which are required for stable and fast control, which all feed into the flight controller. For the drone to successfully autonomously navigate an environment, it will need to better understand its surroundings. The flight controller will analyse this data and based on flight software will control the electronic speed controllers (ESCs). The on board computer (OBC) will process and handle all non flight critical sensor information and can be considered as the mission computer. The OBC must also have provision to connect to a sensor pack via a predetermined interface. Another useful and necessary function, will be the ability to illuminate a dark area.

The network design will also consist of 2 discrete wireless links. Each link is dedicated to one of the computing modules. The real time node will be interfaced through a low bandwidth, high range and high reliability interface. With the non-real time node linking to the high bandwidth interface which will have lower range and reliability requirements. Both of these interfaces will link to a ground station of some sort.

No system can operate without a sufficient power source. In the case of any robotic system it is important that every aspect of the design is optimised. The capacity of the power supply is limited by weight and should be optimised to the lifting capacity of the platform. Once the battery chemistry has been decided upon, work can be done to design a robust power management system. Limiting the leakage of every subsystem as well as choosing efficient modules are both part of an effective power management system. The analysis of the power system is important since it generally contributes a very high percentage to the overall mass of the craft.

Flight Controller

The flight controller is responsible for handling all the flight critical sensor data. Using a motor mixing algorithm the flight controller will then output signals to the electronic speed controllers. There are multiple options for flight controllers, all tailored to different applications, platforms and flying styles; each option has their own level of reconfigurability. For this application it is important that the chosen flight controller allows access to both the inner control loops as well as allows additional sensor data to be added into the

architecture. The flight controller is also responsible for handling the pilot commands and waypoints sent via the radio link, as well as transmitting in flight diagnostic information back to the ground control station.

The flight controller's computing time capabilities have a direct influence on the system. This timing needs to be understood and introduces requirements on the controller designs. Any additional computing the craft needs should be handled by the OBC and not the flight controller. This will reduce the risk of unmodelled timing delays and unrealistic controller design goals.

Electronic Speed Controller

The above mentioned flight controller sends a motor control command to the electronic speed controller (ESC). The ESCs then in turn directly control the speed of each motor. Each motor should have a dedicated ESC. This part needs to be chosen based on the maximum amount of current it can handle. At 100% throttle the current draw of the motor should not exceed 75% of the ESC's limit. Another important consideration is the ESC's refresh rate and computing speed. The flight controller will be rapidly sending data to the ESC. The quicker the module can respond, the more robustly the platform will be controlled. Possible timing delays can be accounted for by ensuring the controllers have sufficient phase margins.

On Board Computer

Where the flight controller ensures the craft maintains steady flight, the On Board Computer (OBC) handles all higher level processing and flight strategy. This will include interfacing to the rest of the on board sensors, the sensor pack as well as handling the high bandwidth networking to the ground station. The OBC should not be required for flight purposes, and can thus be considered as a dedicated mission computer. The OBC should be as light weight and low power as possible while being sufficiently powerful to do real time analysis of camera data for future missions. The OBC must have multiple standard communication ports to easily interface to other sensors. There should also be a highly reliable, and fast interface between the OBC and the flight controller, thus allowing more complex flight strategies to be put in place, while not risking the robustness of the tracking control.

On Board Sensors

The on board sensors can be split up into two distinct categories. The first set of sensors are to enable stable flight of the craft and are used to close the control loops. Examples of such sensors would be inertial measurement units (IMU), global positioning units (GPS) and other relative velocity measuring sensors.

The second set of on board sensors are required for enabling more intelligent autonomous flying strategies. Examples of these sensors include proximity measurements to enable obstacle avoidance and localisation. Other types of sensors from this set could include condition monitoring of the craft's batteries.

Sensor Pack

The actual sensor pack in question will remain generic. So that the platform can cater for an array of sensing devices and applications. To ensure compatibility, the sensor pack needs to be powered and communicated to. It will have access to interface directly to the OBC. Typical examples of sensor pack application would be mapping equipment and stored video feed for post flight inspections.

3.2 Platform Construction

3.2.1 Design Considerations

Before a proper analysis can be done, specific design criteria need to be outlined; this includes parameters such as the required flight time and manoeuvring decisions.

Physical Restrictions and Requirements

One of the major components CEPAD will have to overcome is to navigate these unknown areas and not only survive collisions, but also reject the disturbances introduced by being close to the walls. Since mine shafts are predominantly long and narrow, the same approach will be looked at for the design of the drone. To optimise the size of rotors that can be used the platform will also be designed to be long and narrow. Since the drone will be required to navigate very confined spaces, the smaller the drone the better. The minimum size of the drone is limited by the need for adequate flight time as well as payload capacities which require a higher disk loading.

Manoeuvring Decisions

The manoeuvring decisions are dependant on the type of environment and type of missions required by the platform. These decisions influence the final design of the craft. The end use case for CEPAD will include mapping of unknown areas. To complete this, it will simplify the procedure if the drone keeps its orientation during flight. Due to the nature of the environment, fast speeds will not be used regularly. Therefore CEPAD will be designed to have slow, steady and controlled movements.

Disturbances

Apart from the difficulty of navigating and manoeuvring through an unmapped area there are other disturbances introduced into the system. Some common disturbances found were outlined in Section 2.3.5. Due to the nature of the tunnels, wind gusts are created that funnel through these passageways. These winds will produce large undesired drag moments and forces. The effect of coming close to a wall or floor has been discussed in the literature study. Since the areas will be unknown and complex, collisions and bumps are extremely likely. The flight strategy will try and ensure the drone does not collide with the environment. However, the drone must be able to withstand a collision and maintain its orientation as best as possible.

Thrust Overhead

The total overhead of a rotorcraft is a percentage above the thrust required for hover. This value determines a craft's ability to manoeuvre, with a higher value giving it more freedom and a greater ability to resist disturbances. With these benefits the system does become very sensitive and more difficult to control and stabilise. Since the craft will be in confined narrow passages, the craft does not need to be fast moving. Rather a "slow and steady" pace will be approached. The craft does however need enough power to counteract the disturbances described above. These considerations lead to a value of 150%, with 100% being enough thrust to hover.

Flight Time

Flight time is dependant on efficiency and power requirements of the system as well as the capacity of the on-board power source. This becomes a typical optimisation problem. By adding a larger power source the weight is increased and therefore the platform requires more power to keep itself aloft. Weight is a determining factor for any aerial system and influences flight time, for this reason the weight of every subsystem must be optimised. To ensure the craft can complete a mission it will need sufficient flight time. Initial discussions have set 30 minutes as the bottom limit. The original platform might not be able to reach this goal, but once the platform is performing adequately adjustments can be made to the system to optimise weight and power consumption.

3.2.2 Chosen Concepts

The traditional configurations of drones struggle to handle disturbances introduced by being in a confined space. Traditional configurations are also not optimised for fitting inside a long narrow space. For these reasons a few unique designs were considered and are discussed below. For the comparison, it was assumed that thrust to RPM and RPM to current are linear relationships.

Two concepts were selected as final candidates. This next section walks through some of the important factors considered and ultimately, why certain decisions were made. The naming convention used is shown in Figure 3.2. On the left of Figure 3.2 represents the "*Unlike Size Quad*" and the right of Figure 3.2 is the "*Overlapping Quad*".



Figure 3.2: Rendering of initial concept of the unlike rotor size quadcopter (Left). Malloy Aeronautics Hoverbike Concept (Right) (Picture taken [26]).

Concept 1 - The Overlapping Quad

The overlapping quad is a concept pursued by Malloy Aeronautics [26]. They used the design in an initial concept of their hover bike personnel carrier, which requires a large payload capacity. The design uses an H-formation for it's rotors, except the rotors are brought in to limit the width of the craft to the point that they overlap, as shown in Figure 3.2. Each overlapping pair will have both spin directions, this feature is shown in Figure 3.3.

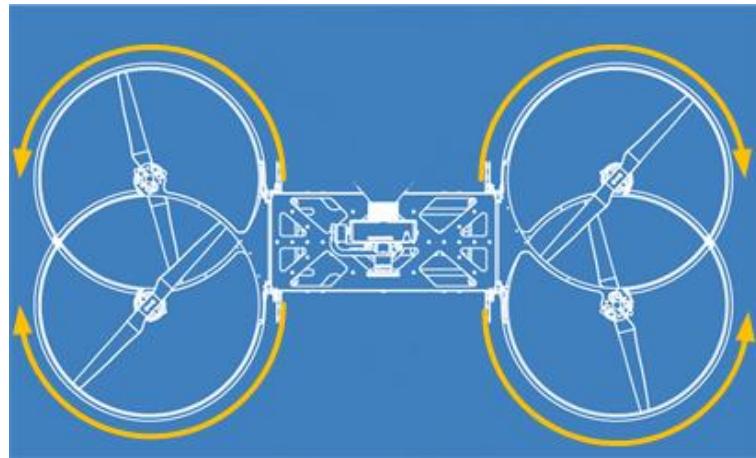


Figure 3.3: Overlapping concept, visual representation of rotor pairs. Image modified from [26]

Overlapping rotors introduce an inefficiency into the system. Figure 3.4 illustrates how the percentage overlap affects the total efficiency of the rotors.

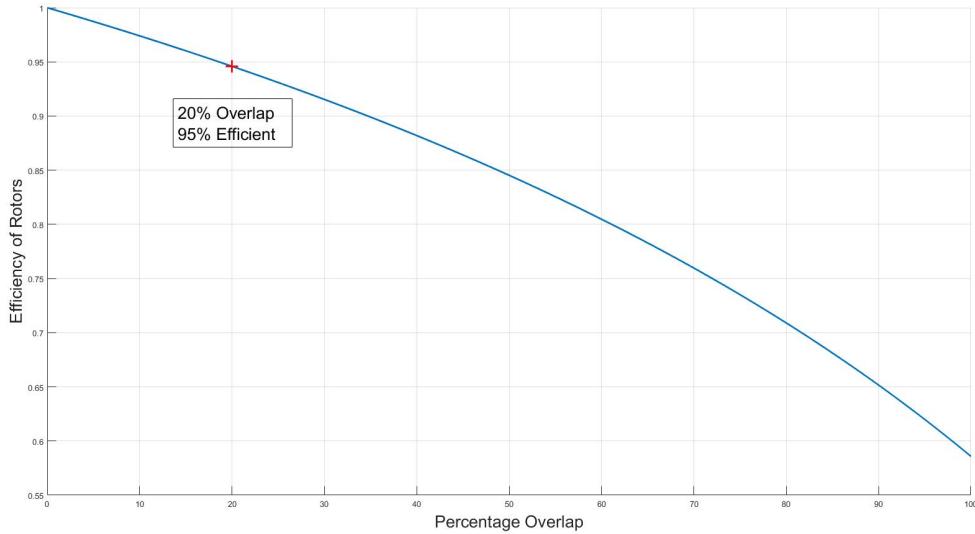


Figure 3.4: Graph representing the effects of overlapping rotors in a quadrotor

A large benefit to this design is the power to size ratio. This benefit can be utilised through handling larger payloads, or a larger battery pack increasing total flight time.

Concept 2 - The Unlike Size Quad

The unlike size quad is an original design that uses the standard cross formation, except it has two pairs of different size rotors. This means that the counter rotating pairs will be set up as shown in Figure 3.5, with each rotation direction having one big and one small rotor.

To maintain a common disk loading in the system, the thrust requirement will be lower on the smaller blades and larger on the bigger blade. The smaller the side rotors get, the higher the thrust requirements of the larger rotors become, this limits the rotor size ratio. Initial calculations, factoring in thrust overhead, overall size of the craft and minimum thrust allowed to rotors, set the ratio between 65% – 80%. When approaching the lower bound, the thrust requirement for hover alone leaves very little room for manoeuvring or disturbance rejection. The upper bound reaches a point where the size difference is so negligible the design's narrow intent is lost.

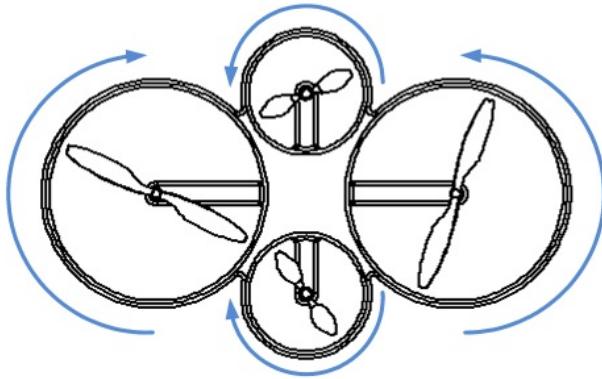


Figure 3.5: Unlike Size Quad visual representation of rotor pairs

The second important choice is how far to put the rotors away from the centre. As the craft gains translational speed, the air doesn't come in directly from the top any more as shown in 2.4. Instead it now starts to come in at an angle. As the craft then manoeuvres and changes it's orientation, the air starts coming in at more extreme angles. If the electronics housing has a lot of height it can interfere with the wake boundary and create an inefficiency. This inefficiency is also based on how far the rotors are from the housing. Therefore, before this decision can be made the limits of how small the centre electronics housing needs to be decided.

Through discussions and observations of current systems a minimum limit of $75mm \times 75mm$ was set. To avoid overlap, the distance from the centre of the big rotor and the centre of the craft will be at the very least R . Including space for the housing increases this.

Since the craft needs to remain narrow, the side rotors can be brought in as well as shrunk. This would require pushing the two larger rotors slightly more out. Bringing the side rotors in closer to the middle housing will reduce efficiency, so a compromise must be chosen between width and efficiency. Since the side rotors don't contribute as much to the system as their bigger counterparts, they have the option of a lower separation distance. The lower separation distance can also be justified by the lower need and use of roll moments and side translations.

3.2.3 Concept Comparison

The comparison of the two concepts includes hover efficiency, thrust, electrical power requirements, size and manoeuvrability. The final decision will need to be made with certain assumptions in mind. These assumptions as well as the method of comparison are described below.

Assumptions and Method

Since both designs use 4 rotors they can be compared relative to a well known design, the standard quadrotor. For each configuration certain parameters need to be decided before a comparison can be done.

If hover is a thrust of 100%, the overhead was set at 50% of that, to a total of 150%. The mass in question includes provision for a sensor pack of undecided mass.

The unlike size quad had its small rotors set at 75% of the larger ones, this decision is explained further in the text. To give a quantifiable reference, R was set at $254mm$. With that assumption the unlike size quad moves its side rotors in to a distance of $300mm$ and the larger rotors were moved to $508mm = 2R$ away. The overlapping quad set a separation distance of $350mm$ which created an overlap factor of 20%.

Rotor Area

If a rotor size of R is assumed for the rotors¹ then the total area for a standard quad will be $A_{std} = 4\pi R^2$. The reduction in radius of the two smaller rotors leads of the unlike size quad leads to a decrease in area of 78.13% of A_{std} .

Thrust and Power Considerations

The decision to set the smaller rotors to 75% was based on observation of thrust ratios between the rotors. The final value of thrust required from each rotor as a percentage is shown in Figure 3.6. The points marked are at minimum, hover and maximum. The total thrust available to the unlike size quad is $\approx 78\%$ of the thrust available to the standard design. This reduced value also comes at a weight reduction. The overlapping quad has an equal total rotor area but an inefficiency is introduced by the overlap as according to (2.16). Therefore the overlapping quad has 95% of the total thrust capacity, without the weight reduction.

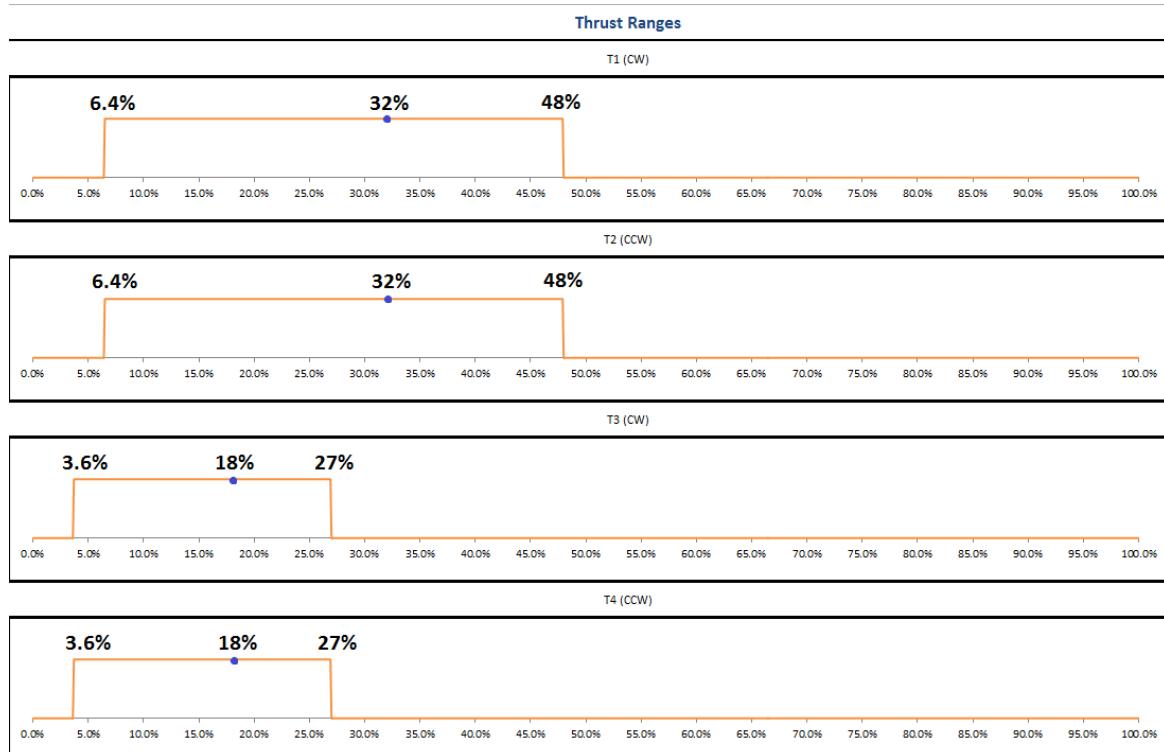


Figure 3.6: Graphical representations of the thrust ratios for the unlike size quad

The values for electrical power were calculated according to how much energy would be needed to obtain the same thrust as the standard design. The inefficiency introduced

¹The big rotors in the case of the unlike size quad

by the overlap relates to a reduction in thrust of $\Delta T_{overlap} = 5.36\%$, therefore $\Delta P_{overlap} = 14.21\%$ is needed to overcome this loss, based on (2.15).

For the unlike size quad, the reduction in rotor size leads to a substantial loss in aerodynamic power, even with a small reduction in inertia. To regain that power, the rotors need to be pushed harder, this leads to an increase in electrical power. A value of $\Delta P_{unlikesize} = 18.5\%$ is calculated.

Size and Manoeuvrability

The size was calculated as though the drone made a rectangular box and with the rough values above, Table 3.2.3 puts those values in a tabular format.

Concept	Length (mm)	Breadth (mm)
Unlike Size	1524	981
Overlapping	1308	858
Standard	1524	1524

Table 3.1: Table representing the size comparison of the two concepts

As shown both crafts are similar in size, with the overlapping design being slightly shorter as well as more narrow.

Discussion

The quantifiable values are culminated below in Table 3.2.3. The winner of each parameter is written in bold.

Concept	Disk Loading	Total Thrust	Electrical Power	Length (mm)	Width (mm)
Unlike Size	78.13%	78.13%	118.57%	1524	981
Overlapping	80.18%	94.64%	114.21%	1308	858
Standard	100%	100%	100%	1524	1524

Table 3.2: Table representing the end comparison of the two concepts

The overlapping quadrotor introduces a significant size reduction to the standard quadrotor while losing minimal thrust to inefficiencies. This allows for a larger more powerful drone to be deployed in tight spaces. The extra required electrical power can be nullified by the ability to carry a larger power source thus increasing flight time.

Malloy Aeronautics sell what is called a *Drone-3 Kit*, it includes various accessories to help developers use the platform. Based on the above analysis and conclusions the overlapping quadrotor design will be used as the design going forward.

3.3 Electronic Design

At the core of any reliable system is a thoroughly planned electronics system. There are multiple aspects this subsystem must handle. Due to the rapidly increasing market and interest in multirotors there has been a wave of developers and designers creating devices for these specific purposes. The first and foremost is monitoring and controlling flight dynamics.

The on board computer will handle the data streams and control the non-real time peripherals. For the drone to successfully navigate an environment it will need to better understand its surroundings. To achieve successful navigation, a few onboard sensors have been looked at. Another design consideration is the provision for a sensor pack. This section of the study will also involve looking at potential external sensors and providing accessibility for them.

3.3.1 Flight Controller

There are multiple options for flight controllers, all tailored to different applications and flying styles. This section identifies some possible options

Custom Design

Given the time and resources most final products will look at custom designing some hardware and electronics. The benefits of custom design include complete control over the operation and functionality of the design as well as reduced cost when scaling up. The development of a custom design however can be very taxing and costly.

It is also common for a research lab to dedicate time into developing such modules. By using MITs custom board, Cutler in his masters dissertation demonstrates this [16]. *The University of Stellenbosch's Electronics System Lab* produced a custom design of their own which is currently under redesign and unfortunately cannot be used for this project.

Pixhawk

The need for a commercial solution becomes evident. Due to the growing hobbyist community, some flight controllers are difficult to modify and are designed for use as a "plug and play" module. Fortunately there is also a large designer community which has created the need for more configurable modules.

Figure 3.7 shows the *Pixhawk*, which is marketed as an autopilot module for fixed wing and rotor wing aircraft. It is specifically tailored for research and is listed as open-hardware². Due to the open platform it has created an experienced community with good documentation and other forms of assistance.

²<https://store.3dr.com/products/3dr-pixhawk>



Figure 3.7: Pixhawk Flight Controller

It features a 32 bit STM32F427 processor, running at 168MHz with 256Kb of RAM and 2Mb of Flash memory. It comes equipped with a full inertial measurement unit (IMU), consisting of a gyroscope, accelerometer and magnetometer. The Pixhawk also includes an integrated barometer and has an additional 32 bit co-processor that acts as a failsafe. There are multiple interfacing capabilities as well as a built in power protection unit³.

The Pixhawk has been designed for robotic applications thus is light weight and power efficient. It operates on a real time operating system called *NuttX* which has Unix characteristics but is much lighter than a different operating system such as Linux. This provides a lot of reconfigurability which is needed in this project.

3.3.2 On Board Computer

The On Board Computer (OBC) is required to handle all higher level processing. In an aerial application weight and power consumption are both important considerations. Based on current available commercial products a few were selected. The advantages and disadvantages of each are described below.

Raspberry Pi 3 Model B

The Raspberry Pi has become a well known and respected piece of hardware. It has generated a large community and thus resources are readily available and the device can be bought locally. The new version of the device runs a 1.2GHz 64-bit quad-core ARMv8 CPU and includes built in Bluetooth and WiFi modules, an image from the Raspberry Pi website is shown in Figure 3.8⁴.

³<https://pixhawk.org/modules/pixhawk>

⁴<https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>



Figure 3.8: Raspberry Pi 3 Model B

It has a large 40 pin GPIO connector, 4 USB connections and an Ethernet port. The Pi's main advantage would be the access to the online community constantly updating Raspberry Pi resources and forums. With the community also comes example projects and large variety of compatible hardware and open source software. With a maximum current of 2.5A at 5V, the 12.5W computer is relatively low powered which suits this application. However there are more powerful machines that can run more intense algorithms at the cost of more power.

Odroid XU4

Hardkernel has designed a compact high processing power unit called the Odroid XU4. It has gained respect in some developer communities due to it's incredible processing power. It can run both Android, Ubuntu and other similar Linux based operating systems. Hardkernel has generated an immense of documentation and wiki pages, all available on their website⁵. They also have a team of developers creating new devices to interface with the devices.

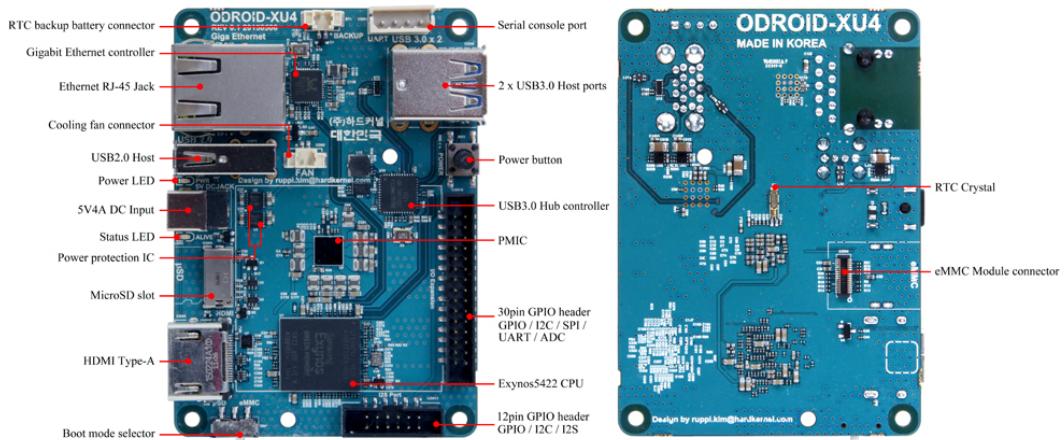


Figure 3.9: The Odroid XU4

The XU4 hosts a Samsung Exynos5422 Cortex-A15 processor running at a 2GHz clock

⁵<http://www.hardkernel.com/>

speed and an additional Cortex-A7 Octa core processing unit. This allows for incredible processing capabilities and speed, an image from Hardkernel's Website is shown in Figure 3.9. The rated power supply for the unit is a 4A at 5V module. Including a 20W module on board will add a significant drain on the power source. It comes with 2 stable USB 3.0 ports and an additional USB 2.0 connection.

The Odroid is also frequently used at the CSIR, opening up experience and knowledge on the devices.

3.3.3 Location

An important part of any robotic application is localisation. As stated earlier, an underground environment limits the use of traditional GPS. Stellenbosch University as well as the CSIR are both funding research into localisation in a GPS constraint environment. Until such a time that these technologies are readily available, this project will work with traditional GPS devices.

As discussed previously, this project will not be designing for a GPS constraint environment. Fortunately there are many commercial GPS modules available for purchase. Each with their own advantages and disadvantages. For this project the module needs to be lightweight and low power. The GPS needs to create speed and translation data, this information is crucial when generating maps of unknown areas. The PixHawk website recommends the 3DR uBlox GPS kit⁶. Although many alternatives exist, the open source hardware status creates ease of integration and support. With these considerations, this module has been chosen.

It weighs a total of 16.8gs and has a low 8.5mm profile. This module will perform adequately and fulfil the design needs of the project.

3.3.4 Object Avoidance

Due to the nature of the project, object avoidance is an important attribute to include in the design. Obstacle avoidance requires that the drone has some level of an understanding of the environment. This information can then be used in higher processing nodes to actively avoid objects and plan routes. A few different sensor options are observed below, each with a set of advantages and disadvantages.

Laser Range Finders

Laser range finders can be found in a multitude of high end robotic platforms often in the form of a Lidar. They exhibit high accuracy and can be bought as full 360°modules. Lidars are expensive and heavy making them less suitable for this application.

Ultrasonic Proximity Sensors

Ultrasonic sensors are already used in aircraft, generally as downward facing to accurately predict altitude. Relatively speaking ultrasonic sensors are cheap and light weight and consume little power. They are however prone to measurement errors when used against non-uniform surfaces and require additional processing to ensure that multiple sensors do not influence each other. These sensors produce a measurement cone, rather than a

⁶<https://store.3dr.com/products/3dr-gps-ublox-with-compass>

directed pinpoint measurement providing a large coverage area per sensor. For this work ultrasonic proximity measurement will be used.

Chapter 4

Mathematical Modelling and System Identification

This chapter follows the process of mathematically modelling the quadrotor system. The nomenclature for the kinetics and kinematics has been discussed in Section 2.3. The chapter begins with the dynamic flight model of the craft, focusing on all forces and moments generated by the rotors. The chapter then continues to describe the process of system identification followed to generate the required constants and system parameters for an accurate model. The external disturbances are discussed next, specifically the wind models used in this work. The chapter concludes with an overview of the non-linear simulation generated from the discussion.

4.1 Dynamic Flight Model

The dynamic flight model of the craft must cater for all six of the degrees of freedom the craft experiences. The dynamics of this system is modelled as three rotational and three translational degrees of freedom [28]. To continue deriving the equations of motion, the following assumptions are made: the aircraft is a rigid body, the aircraft has constant mass and I_{xy}, I_{xz}, I_{yz} are all negligibly small.

The Newton-Euler method of defining the accelerations uses the inertial frame to define the linear accelerations, and the body frame to define the rotational accelerations. Using Newton's first law and the rotation matrix described in (2.18), the expression for the linear acceleration can be developed and is shown in (4.1).

$$\begin{bmatrix} \ddot{N} \\ \ddot{E} \\ \ddot{D} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} + \mathbf{R} \begin{bmatrix} 0 \\ 0 \\ \frac{\mathbf{T}}{m} \end{bmatrix} \quad (4.1)$$

The rotational accelerations of the craft can be similarly described using the moments and the simplified inertia tensor. These rotational rates are described in (4.2)

$$\begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} I_{xx}N \\ I_{yy}M \\ I_{zz}L \end{bmatrix} \quad (4.2)$$

4.2 System Identification

In order to correctly model the system, a thorough system identification needs to be completed. This entails real world measurements of the chosen platform or substantiated evidence from literature for aircraft of similar size and characteristics. The methods and results from these experiments are covered in this section.

4.2.1 Mass and Inertia

Using a calibrated scale the mass of the rotorcraft measured at 3.352Kg.

To calculate moments of inertia, the Bifilar Pendulum method was used. The method is thoroughly described in literature and involves tying the drone from the ceiling allowing it rotate around one axis. Since it is desired to measure the inertias along three axes, three separate test set ups were required. Images of the test set up for a single axis is shown in 4.1.



Figure 4.1: Bifilar Pendulum for Inertia Measurement

Each axis was measured 10 times and the values were averaged out to obtain the final values represented in Table 4.2.1. To give a representation of measurement variance, the standard deviation is provided along side.

Parameter	Averaged Measured Value	Standard Deviation
I_{xx}	0.025027578	0.001063842
I_{yy}	0.169260024	0.000142928
I_{zz}	0.170196714	0.000527406

Table 4.1: Measured Moments of Inertia

4.2.2 Thrust and Moment Profiles

In order to correctly validate the thrust characteristics of the drone, each motor rotor pair needed to be evaluated. Each pair was marked and coupled to a load cell. The ESCs

were configured to send varying PWMs to the motors. The commands sent to the ESCs and the measured thrust values are plotted together in Figure 4.2. Table 4.2 gives the exact maximum and minimum values of each rotor motor pair.

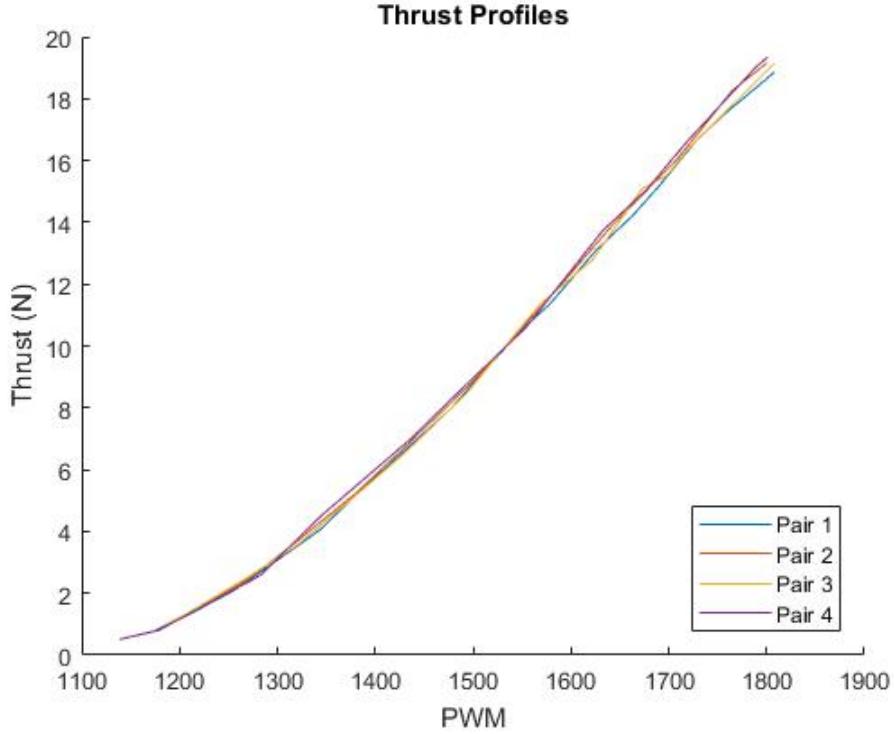


Figure 4.2: Thrust Ranges for Motor Rotor Pairs

Pair	Max Thrust	Min Thrust
1	18.8558	0.7852
2	19.1489	1.1889
3	19.1434	0.9511
4	19.3369	0.5087

Table 4.2: Measured Moments of Inertia

The motor lag constant for a similar sized craft and rotors was investigated in [28] and found to be 0.125s. Each rotor is also subject to high frequency noise caused by undesired vibrations. The high frequency noise is modelled as a bandwidth limited white noise applied to each motor separately.

4.2.3 Drag Coefficients

The drag coefficients chosen are that typically expected for a flat plate with an effective frontal area, as that seen in the fuselage of the craft. A C_D of 1 is chosen.

The surface area of the drone was calculated by taking physical measurements of the drone and calculating the surface area from the measurements. Table 4.3 records the calculated surface areas to be used in the calculation of each drag component. The last value in the table is the centre of gravity (COG) which was calculated to be located at the

centre of the X and Y axis indicating symmetry along those axes, with a 3.5 cm offset in the Z-Axis. The centre of pressure (COP) is assumed as the centre of the craft, creating the required offset for the drag moments described in 2.3.5.

Parameter	Value
A_x	0.0354
A_y	0.0693
A_z	0.4
C_D	1
COG	[0 0 0.035]
COP	[0 0 0]

Table 4.3: Drag Coefficients

4.2.4 Sensor Constants

Investigation into the sensors used by the PixHawk, the proposed flight controller module, gives an indication of the sensor accuracy, resolution, noise and speed the drone measurements will be subject to.

The PixHawk uses a combination of gyroscopes, accelerometers and magnetometers. The sensors used for the characterisation are the 6 DOF BMI055 which contains a 3-Axis accelerometer and a gyroscope along with the IST8310, 3-Axis magnetometer. Table 4.4 lists the extracted data.

Sensor	Offset	Resolution	Noise	Maximum Bandwidth
Accelerometer	$\pm 70 \text{ mg}$	0.98 mg	$150 \mu\text{g}/\sqrt{\text{Hz}}$	1000 Hz
Gyroscope	$\pm 1^\circ/\text{s}$	0.1 $^\circ/\text{s}$	$0.014^\circ/\text{s}/\sqrt{\text{Hz}}$	1000 Hz
Magnetometer	$\pm 0.3 \mu\text{T}$	0.3 $\mu\text{T}/\text{LSB}$	NA	200 Hz

Table 4.4: IMU Sensor Coefficients

For location and velocity measurements the GPS sensor used is the Neo-M8N developed by UBlox. Table 4.5 lists the relevant information obtained from the sensor datasheet.

Sensor	Accuracy	Maximum Bandwidth
Position	2.5 m	10 Hz
Velocity	0.05 m/s	10 Hz

Table 4.5: GPS Coefficients

From the information gathered the sensors can now be modelled accordingly in Simulink. The IMU is modelled using the 3-Axis accelerometer block and the 3-axis gyroscope block in Simulink. The noise is set by using the above documented noise ratios. The GPS model utilises the Band-Limited White Noise Block in Simulink to create appropriate sensor noise.

4.2.5 Wind Model

The wind model is broken into two portions, a constant wind and a more erratic, unpredictable wind both of which flow in the NED frame. The constant wind is defined as a

configurable constant in the North East Down frame. The gust component is modelled as shown in Figure 4.3. The band limited white noise block is passed through a first order low pass filter to create a more realistic dynamic wind. The gain of the filter can be adjusted to observe effects of larger wind gusts on the craft.

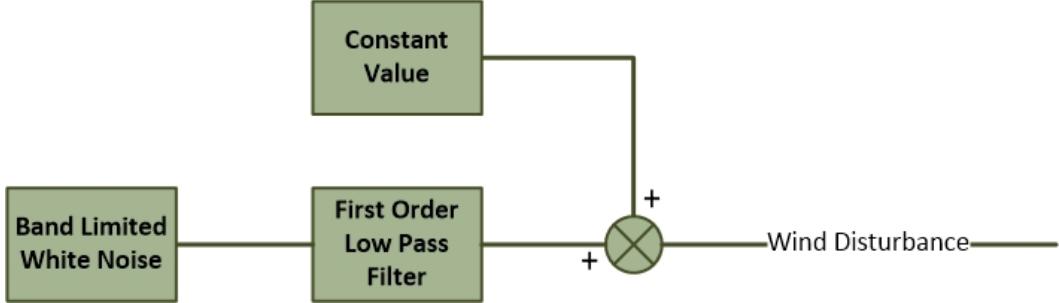


Figure 4.3: Wind Model

The direction is commanded in the NED frame and rotated to apply the applicable forces to the body frame. The direction is modelled with a constant offset and varied by adding a small random number, smoothed by a low pass filter to create a more realistic wind direction variance.

4.3 Simulation Configuration

This section describes the process of creating the non linear simulation of the aircraft. The drone was modelled in Simulink using a combination of blocks included in the aerospace and control toolboxes, as well as a set of custom blocks. The model is required to successfully predict the position of the craft after being subjected to forces and moments. Using the dynamic flight model created above the bodies linear and rotational accelerations can be calculated by inputting forces and moments into the system. These accelerations can be subsequently integrated to achieve rates and positions. No model is 100% accurate, however the information recorded in this chapter, when included, should create a solid estimation of the real world craft dynamics.

4.3.1 Motor Mixer

The motor mixer is an important part of the control software running on the onboard computer and thus should be accurately modelled. This portion of code commands the ESCs of the drone to create the desired moments and forces. Section 2.3.4 explains how the rotors have control of four virtual actuators, these are δ_Z , δ_θ , δ_ψ and δ_ϕ . The motor mixer is responsible for converting these desired moments and forces into actual values of thrust for each rotor.

As Figure 4.4 describes, each motor thrust value is made up of a summation of four thrust values, T_Z , T_θ , T_ψ and T_ϕ . Each value describes the rotor's contribution to the generation of that specific virtual actuator.

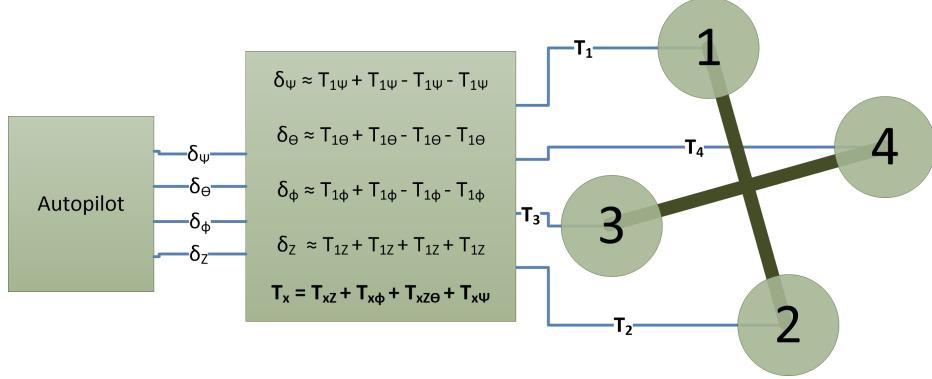


Figure 4.4: Motor Mixer

The motor mixer mathematics is based on the contribution provided by each rotor's thrust to the relevant virtual actuator. Assuming that each rotor only contributes a Z-Force it can be said that each rotor's thrust contributes 100% of its power to the virtual actuator δ_Z . The roll and pitch virtual actuators are moment contributions and are calculated by evaluating the distance off the centre of gravity. Using the rotor arm length (l) and the angle (α) a simple trigonometric relationship dictates the roll and pitch contributions. Finally the yaw contribution is calculated by taking into consideration the drag created by each rotor as it generates thrust. This relationship includes the lift to drag ratio (R_{LD}) of each rotor and the effective chord length (r_D) where the drag force is applied. Using all of these relationships the motor mixer's contribution matrix can be created and is shown in (4.3). The signs in the matrix are dictated by the body frame notation of positive directions and the placement of the rotors. The yaw contributions are dependant on the rotation direction of the rotors.

$$ContributionMatrix = \begin{bmatrix} -1 & -1 & -1 & -1 \\ l \sin \alpha & -l \sin \alpha & l \sin \alpha & -l \sin \alpha \\ -l \cos \alpha & l \cos \alpha & l \cos \alpha & -l \cos \alpha \\ -\frac{r_D}{R_{LD}} & -\frac{r_D}{R_{LD}} & \frac{r_D}{R_{LD}} & \frac{r_D}{R_{LD}} \end{bmatrix} \quad (4.3)$$

The values for thrust can subsequently be calculated by multiplying the desired forces and moments by the inverse of the contribution matrix.

Chapter 5

Controller Design

This chapter follows the stages of designing a controller for a quadrotor and begins by discussing the overall strategy. An analysis of existing rotorcraft control systems has been done in Section 2.4. The system must be able to follow waypoint commands, containing a position reference in the North, East and Down frame. As well as align with a desired heading, this reference will be in the form of a Euler angle. The detailed design goals are discussed before the controller design is begun.

5.1 Design Goals

The design goals for the controller system are presented here. The controllers must ensure that the system is capable of meeting all of the prescribed requirements.

1. The controller must be able to track a position reference in the North, East and Down frame. The limited space in which the vehicle must fly requires a final system with sufficient damping and minimal overshoot.
2. The transient response of the velocity control system must be rapid to ensure fast responses to changing setpoints caused by unexpected obstacles.
3. The controller's must be robust to unmodelled system errors. To accomplish this sufficient phase margin is required to handle unmodelled timing delays and sufficient gain margin is required to handle unmodelled errors in actuation and plant gains.
4. The output of the controllers must never exceed the thrust capabilities of the system.
5. The controllers should be designed as to not demand large angles for the craft.
6. The velocity of the craft must be limited for flight inside a confined environment.
7. The controllers must be capable of rapidly rejecting disturbances.
8. The designed controller bandwidth must not exceed the bandwidth capable of the rotor motor system.

The stipulated design goals are appropriate for a vehicle expected to fly in a confined space and differ from traditional flight which requires high speed and fast position tracking.

5.2 Flight Control Strategy

Figure 5.1 represents the high level control strategy for this project. This discussion will break the system into three main components: an altitude controller, a horizontal flight controller and a heading controller.

The altitude controller starts by getting a position reference in the earth frame. This reference is fed through a Proportional Integral (PI) controller to create a desired climb rate. The climb rate reference is converted into the body frame and is then controlled using a Proportional (P) controller to produce a body acceleration reference. The heave controller can only produce a force perpendicular to the rotor, thus the Z-Axis acceleration component is taken and fed through an additional PI heave controller. The output of this inner heave controller would be the virtual actuator δ_Z .

The horizontal controller gets both a North and an East position reference and uses PI controllers to create velocity setpoints in the earth frame. These setpoints are converted to the body frame where the linear velocity P controller works and outputs acceleration references in the body frame. Using the body acceleration references, roll and pitch rate setpoints can be created using a tilt angle controller. These rate set points are fed into the inner rate loop lead lag controllers which output the virtual actuators δ_ϕ and δ_θ .

Lastly the heading controller is discussed. Correct design of the tilt angle controller will decouple the vehicle's horizontal controller from a dependency on the heading of the craft. This allows seamless implementation of a heading controller. The heading controller is broken down into two parts: a yaw angle and a yaw rate controller. The angle loop uses a PI control architecture while rate loop utilises a simple P controller. The output of the yaw rate controller is the virtual actuator δ_ψ .

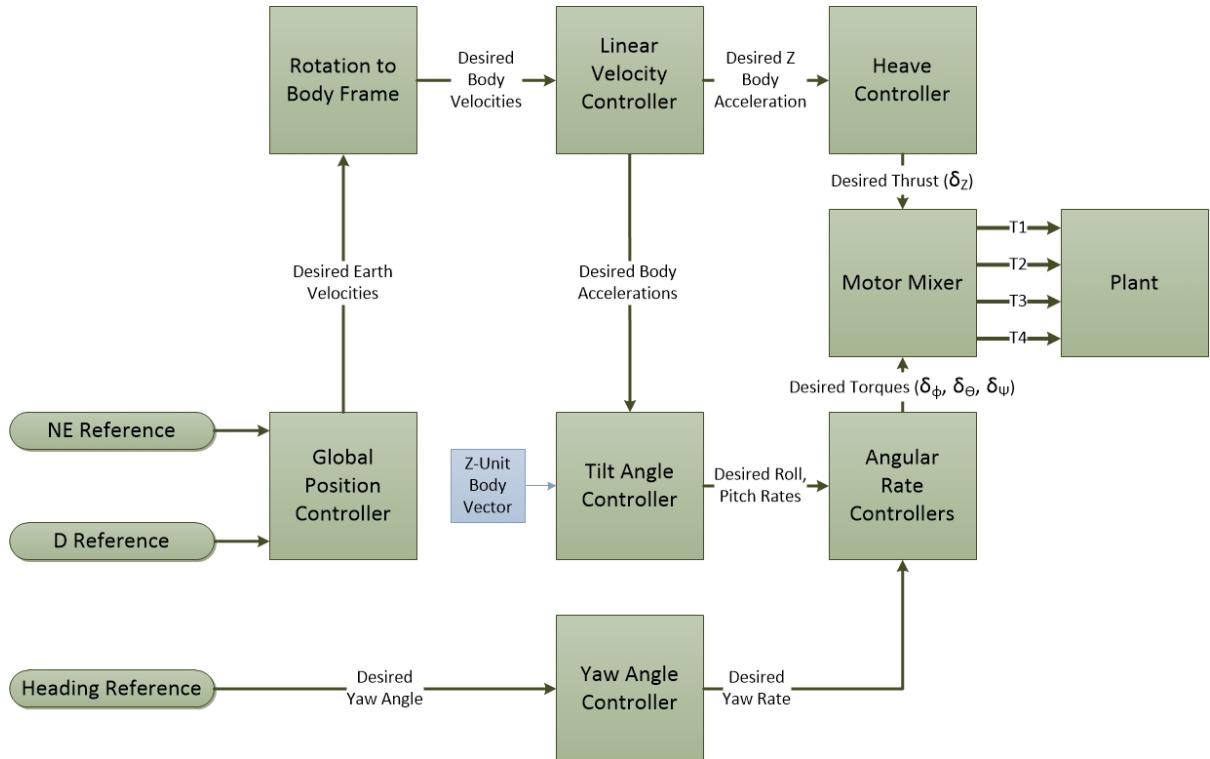


Figure 5.1: High Level Control Strategy

Each controller utilises the same rotor system to produce their outputs. This depen-

dency on a single generation system allows for interference between the controllers. To ensure that no loop can saturate another, each system is given a percentage of headroom in which it can work, as shown in Table 5.1. The controller design must ensure that the thrust commanded during a step response is within those limits.

Controller	Percentage	Allowed Thrust Per Rotor
Altitude	64%	12.16N
North	8%	1.52N
East	12%	2.28N
Heading	6%	1.14N
Safety Factor	10%	1.9N

Table 5.1: Thrust Headroom Controller Percentages

5.3 Altitude Controller

This section discusses the design and implementation of the altitude controller which is responsible for controlling the desired height of the craft. The craft is required to fly in confined spaces and must be able to track a setpoint with zero steady state error and negligible overshoot. In order for the altitude system to handle disturbances, an integrator term will be required. The system must be able to respond quickly to commands, however it must not exceed its thrust utilisation percentage. To do this the system will require an upper limit which should not be reached. A lower limit is then introduced so the craft does not descend too quickly.

The overall altitude controller is structured as a set of cascaded control loops, with the most inner loop controlling the aircraft's acceleration and the most outer loop controlling the desired altitude in the earth frame. The system must be able to respond to disturbances quickly. Therefore, the inner heave loop utilises a PI controller to follow a desired vertical acceleration reference. The climb rate P controller is responsible for generating these acceleration references and is fed a desired linear vertical velocity reference by the altitude hold P controller. In order to reject measurement errors in the inner loops, the altitude hold controller makes use of a limited integrator that does effect the bandwidth of the system. Before the controller's can be designed, an analysis of the system's heave dynamics must first be performed.

5.3.1 Heave Dynamics

Using Newton mechanics at near hover conditions for the aircraft, the heave dynamics can be derived and are shown in (5.1). Where \dot{W} is the current acceleration of the craft in the Z-Axis and m is the vehicle's mass. Z is defined as the current instantaneous force being produced by the rotors.

$$\dot{W} = \frac{Z}{m} \quad (5.1)$$

The state variable of the system is chosen as Z with the output of this plant being \dot{W} . Using the transfer function for motor-rotor lag dynamics seen in (2.28) and the dynamics seen in (5.1), the state space equation for the system can be derived and is shown in (5.2) and (5.3).

$$[\dot{Z}] = -\left[\frac{1}{\tau}\right] [Z] + \left[\frac{1}{\tau}\right] [\delta_Z] \quad (5.2)$$

$$[\dot{W}] = -\left[\frac{1}{m}\right] [Z] \quad (5.3)$$

Subsequently the transfer function can be calculated and the result is shown in (5.4). The negative gain of the transfer function must be noted and is caused by the direction of the defined axes, with the rotors producing a negative Z-Axis force.

$$G(s)_{heave} = \frac{\frac{-1}{m \times \tau}}{s + \frac{1}{\tau}} \quad (5.4)$$

The rotor motor lag of 0.125 s stipulated in the system identification chapter produces the pole at $\frac{-1}{\tau} = -8$ and indicates the maximum response capabilities and timing constant of the rotor system.

5.3.2 Heave Controller

The heave controller is responsible for commanding the δ_Z virtual actuator to achieve a desired Z-Axis acceleration in the body frame. The heave controller is the fastest controller in the altitude system and should utilise as much bandwidth as the rotor-motor system allows. The altitude controller wishes to reject disturbances quickly and thus a PI architecture was initially chosen as shown in Figure 5.2. The system should be stable and exhibit a reasonable phase margin and therefore be able to compensate for margin loss when the outer loops are closed. The phase margin in the open loop system should at least 70°.

The integrator is used to reject disturbances, while the most left limiter shown in Figure 5.2 was added to stop integrator wind up and is not considered during the linear controller design. The proportional gain is used to move the closed loop poles and achieve the desired bandwidth. The dynamic response of the system can be investigated using the root locus and bode plots shown in Figure 5.3 and 5.4.

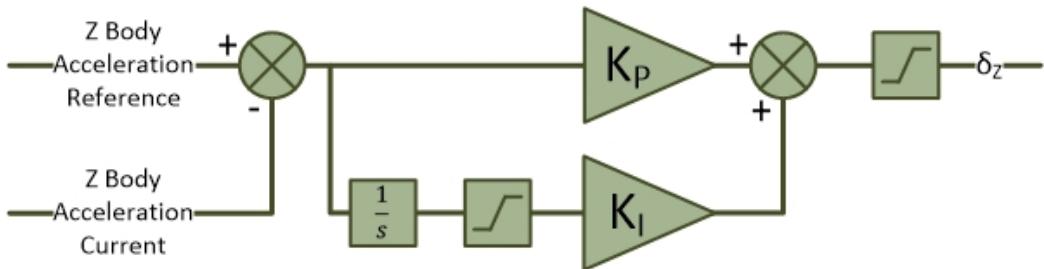


Figure 5.2: Heave Controller - Control Diagram

Figure 5.3 shows the root locus of the system with the PI controller included. The controller introduces a new open loop pole at the origin. To maintain a first order response, the zero is placed close to the plant pole. This placement will attenuate the open loop, plant pole's response. Finally the gain is varied until the closed loop responses are closely aligned with the naturally occurring open loop pole. The final closed loop poles

are a set of complex poles and are located at $-7.55 \pm 3.64i$. The frequency response can be evaluated using the bode plot in Figure 5.4.

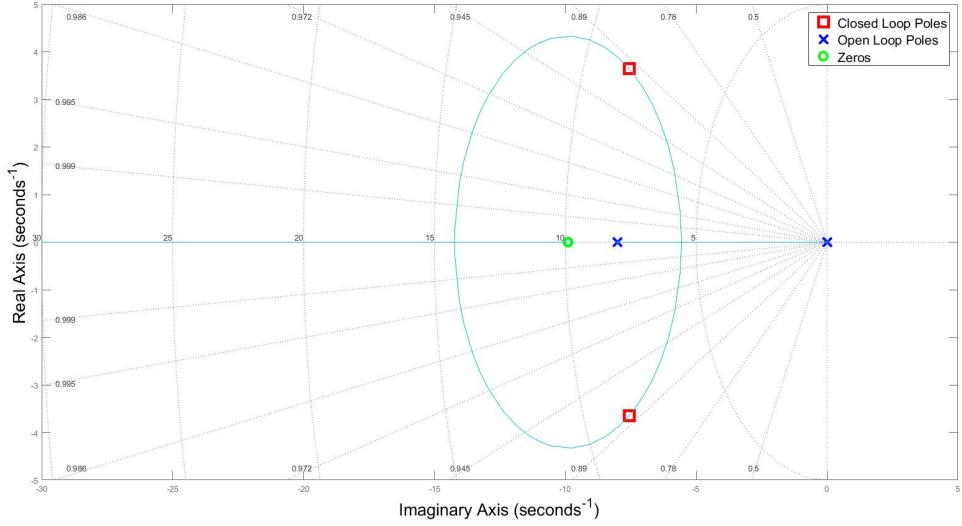


Figure 5.3: Heave Controller - Root Locus

The final cross over frequency is shown on the bode plot for the heave controller in Figure 5.4. The gain plot shows the controller adjust the crossover frequency to 7.99 rad/s which is close to the limit of the system. The controller also increases the phase of the system and has a final phase margin of 84°.

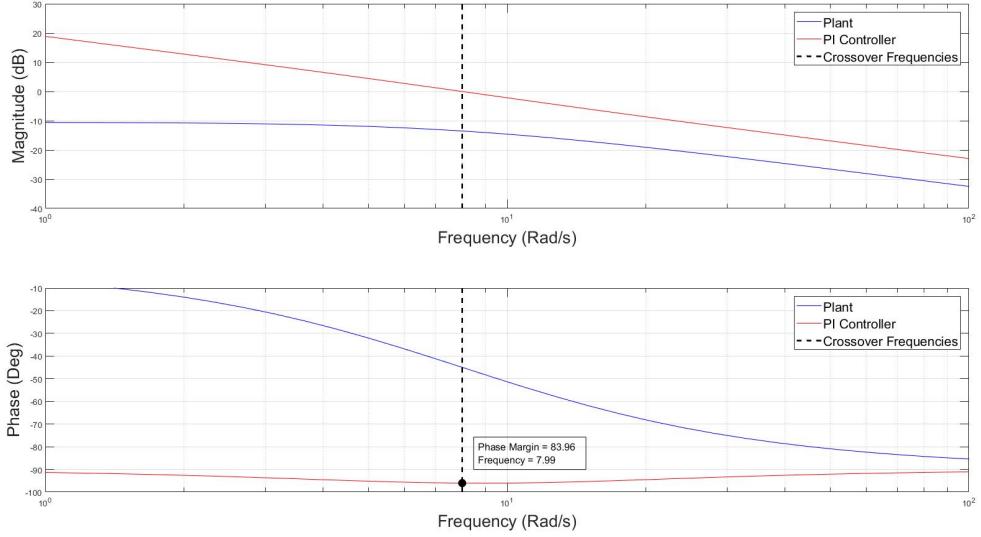


Figure 5.4: Heave Controller - Bode Plots

An additional non linear element in the form of a limiter is brought into the system to limit the maximum and minimum thrust commands. The maximum limit is used to ensure the heave controller does not saturate the motors, thereby creating headroom for

Limit Name	Min	Max
Integrator Wind Up	-1.5	1.5
Thrust Command	10	48.64

Table 5.2: Heave Controller Limits

the angular rate controllers. The lower limit is used to ensure the vehicle always descends at a steady pace. The maximum thrust allowances are displayed in table 5.1, the final limits chosen are shown in Table 5.2.

Heave Controller Discussion

Now that the system presents stable dynamic results in the frequency and Laplace domains, using the non-linear simulation, the time domain responses can be discussed in brief. The resultant step response, including the PI controller, is shown in Figure 5.5. To demonstrate the disturbance rejection capabilities of the design, a force of $10N$ is applied to the drone at $2.5s$.

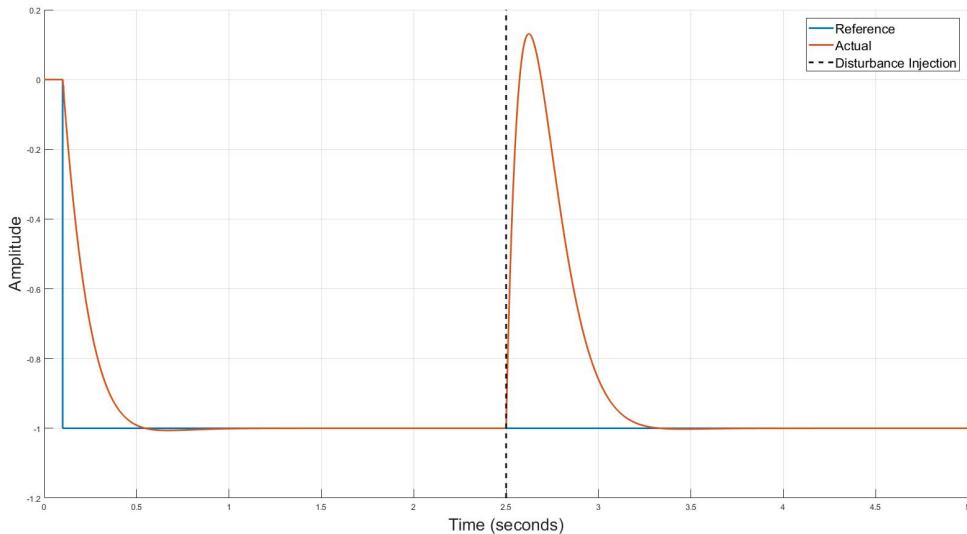


Figure 5.5: Heave Controller - Step Response

The heave controller, as the most inner loop, limits the response for the rest of the altitude control system. The proposed design brings the heave loop response close to the limits of the plant, thus producing a similar (but slower) timing constant to that of the motor-rotor system. The system reaches and settles within 5% of the reference by $0.31s$, is critically damped and presents negligible overshoot. The system also shows to be capable of tracking an acceleration setpoint with zero steady state error. As shown in Figure 5.5 the system can also respond quickly to a large, sudden and constant disturbance. The maximum rotor thrust commanded during this run is 3.35 N . Gravity will add on offset of mg to the acceleration setpoint, this needs to be rotated into the body frame as this controller provides force in the Z-Body Axis.

5.3.3 Climb Rate Controller

The climb rate controller is responsible for controlling the vertical velocity of the aircraft, in the earth frame. This introduces the need for rotating either the reference or the command into the body frame. The decision can be made by considering the frame in which the sensing information is provided. Most aircraft make use of some form of global positioning system and using differentiation can calculate speed. However, due to the application environment, this aircraft will most likely use a velocity measurement sensor relative to the body frame. The architecture for the climb rate controller is outlined in Figure 5.6.

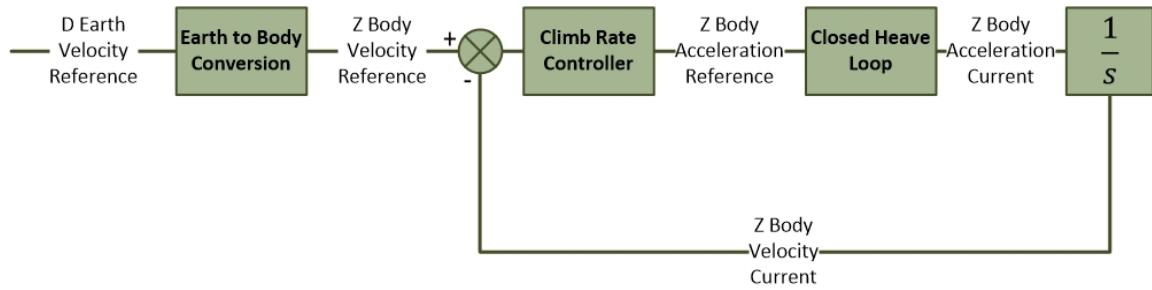


Figure 5.6: Climb Rate Controller Closed Loop

At near hover conditions the plant can be linearised and the rotation can be excluded. Figure 5.7 shows the simplified climb rate controller architecture. The computational time required for rotating the references can be considered by ensuring the controller design provides a reasonable phase margin. The speed of the climb rate controller is limited by the inner heave leave control loop. The controller must react quickly but there must still be a sufficient bandwidth ratio between the inner and outer loop. The controller must be able to track a setpoint with zero steady state error, but is not required to reject disturbances. The craft is required to produce a steady approach to position targets, the climb rate controller should then exhibit a damped first order response.

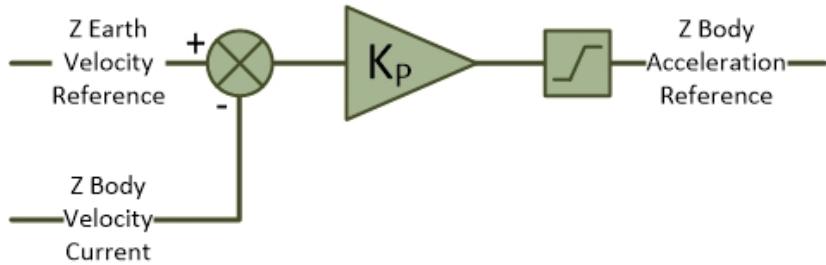


Figure 5.7: Climb Rate Controller

The open loop poles of the climb rate system are located at the closed loop pole positions of the inner heave system, while the mathematical relationship between acceleration and velocity yields an additional open loop pole at the origin. The free integrator in the plant ensures the system will track a step response with zero steady state error while the proportional gain is used to speed up the system and achieve the desired bandwidth. The dynamic response of the system is evaluated using the root locus and bode plots shown in Figure 5.8 and 5.9.

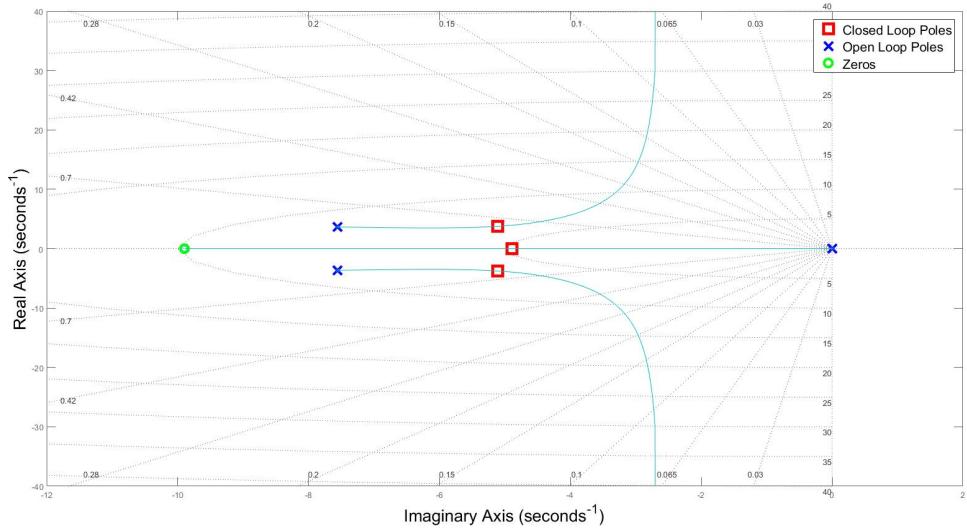


Figure 5.8: Climb Rate Controller - Root Locus

Figure 5.8 shows the location of the three final closed loop poles. There is a non dominant complex pair which is placed at $-5.11 \pm 3.76i$. The dominant pole is critically damped and located on the imaginary axis at -4.89 .

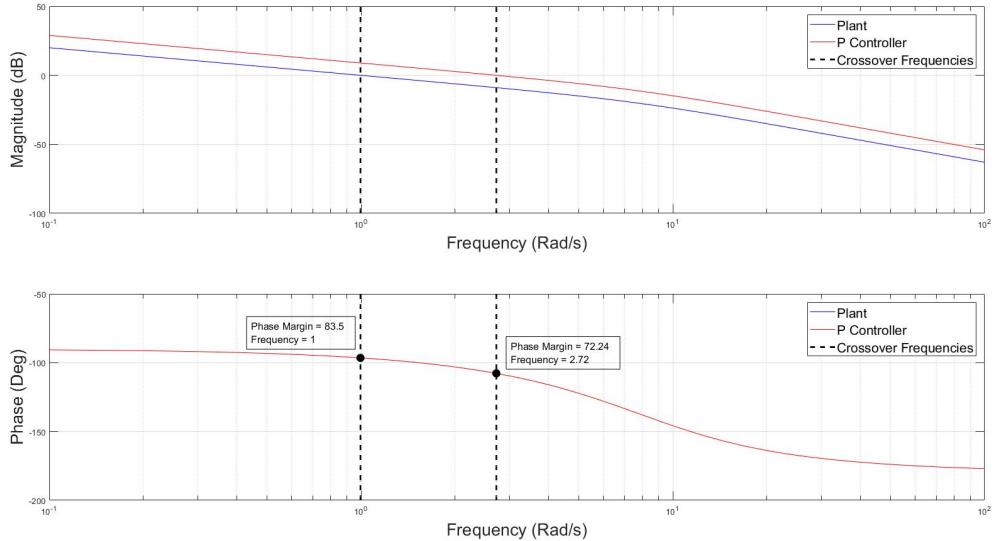


Figure 5.9: Climb Rate Controller - Open-Loop Bode Plots

The bode plot shown in 5.9 shows zero change in phase due to the controller architecture. The gain however is increased and moves the crossover frequency to 2.72 rad/s . The ratio of inner and outer loop crossover frequencies is then 2.91 , providing enough bandwidth between the inner and outer loops. The phase margin can then be calculated to be 72° .

As shown in Figure 5.7 there is a limiter applied to the acceleration commands. This limit is present due to the confined operational environment and ensures that the climb

Limit Name	Min	Max
Acceleration Command	$-4m/s^2$	$4m/s^2$

Table 5.3: Climb Rate Controller Limits

rate controller does not saturate the horizontal velocity controllers. The final limits are shown in Table 5.3

Climb Rate Controller Discussion

The dynamic response shows sufficient phase margin to handle unmodelled timing delays. While the ratio between the inner controller ensures this controller will not be influenced by the inner loop. The step response of the closed loop system is shown in Figure 5.10. The system has a 5% settling time of $0.747s$ and shows negligible overshoot. At $5s$ a disturbance of $10N$ is placed on the rotors and the system demonstrates the ability to continue tracking the desired setpoint with zero steady state error.

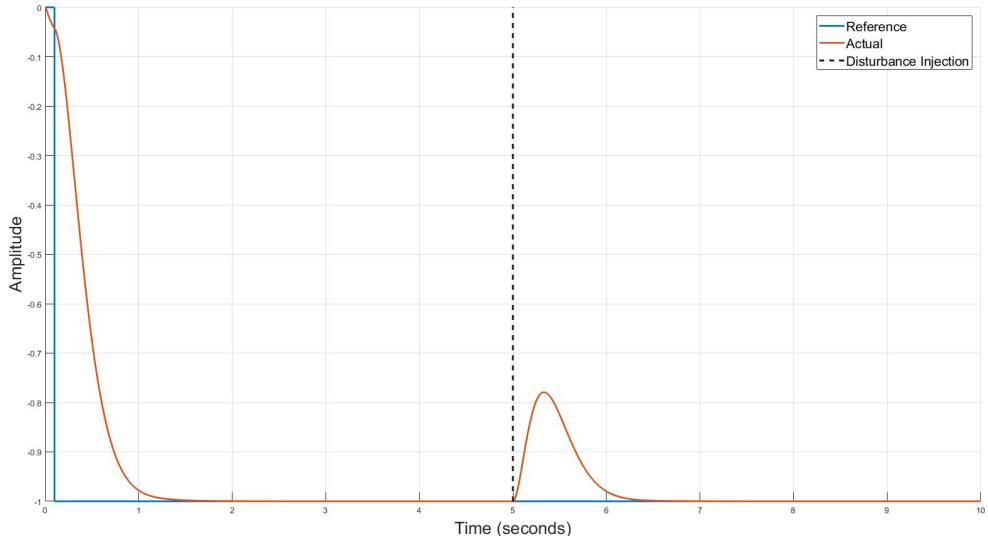


Figure 5.10: Climb Rate Controller - Step Response

5.3.4 Altitude Hold Controller

The final stage of the vertical control system is the altitude hold controller. This controller receives a desired altitude in the earth frame and outputs a reference velocity, also in the earth frame. The closed control loop block diagram is shown in Figure 5.11.

The altitude hold controller must be able to reject measurement errors in the inner loops, this can be achieved by adding an integrator into the controller. The system must also be able to track a set point with zero steady state error and must show a damped response with little overshoot. The system must be able to react quickly to commands, but is limited by the bandwidth of the climb rate system. The final bandwidth of this loop must be such that this controller is not influenced by the inner climb rate loop. To ensure this, the altitude controller should be a magnitude of at least 2 smaller than the climb rate controller. As the most outer loop this system will have unmodelled errors,

the controller must be robust and exhibit sufficient gain margin and a phase margin of at least 60° .

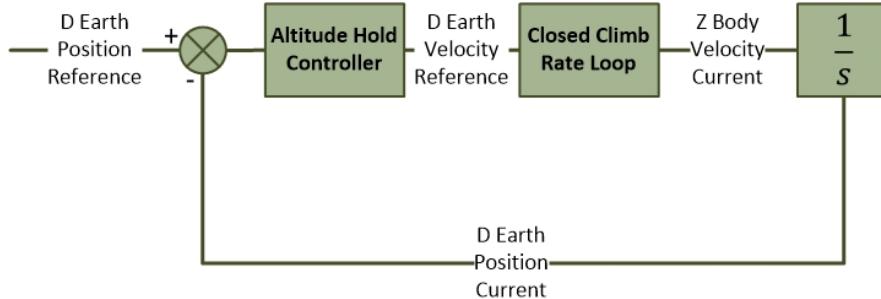


Figure 5.11: Altitude Hold Controller Closed Loop

The chosen controller architecture is shown in Figure 5.12. The proportional gain is used to vary the bandwidth to be within the limits of the system. An limited integrator is added to reject measurement errors in the inner loops, this component is represented by the faded integrator shown in Figure 5.12 is not considered during linear analysis. The integrator shall be limited in such a way as to limit the interference of the proportional gain. This approach reduces the maximum disturbance rejection this controller can handle. To increase the bandwidth of the disturbance rejection capabilities, a PID controller architecture was also considered and the analysis was done for both control laws.

The system's dynamic response is analysed using the root locus shown in Figure 5.13 and the bode plot shown in Figure 5.14.

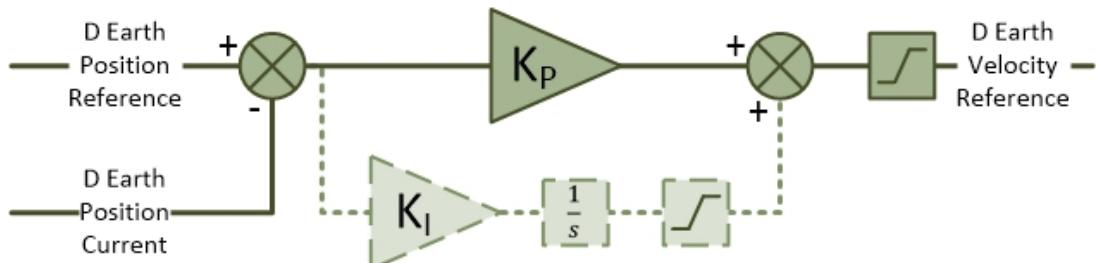


Figure 5.12: Altitude Hold Controller

Figures 5.13 and 5.14 evaluate a P controller against a PID controller. The dominant closed loop poles of the P controlled system are placed at $-2.03 \pm 0.58i$ and are slightly under damped with a damping ratio of 0.96.

The PID controller adds an additional pole and two additional zeros into the system. The closed loop poles of the PID controlled system are located at -6.64 , $-3.59 \pm 5.03i$ and $-0.64 \pm 0.47i$. The two new zeros are placed at -0.57 and -2.06 .

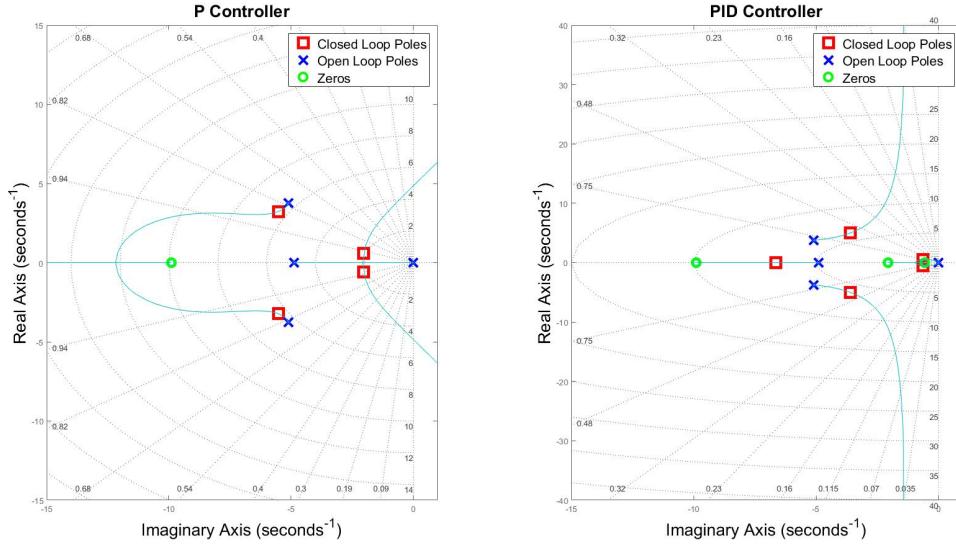


Figure 5.13: Altitude Hold Controller - Root Locus

The bode plot shows the PID controller producing a final cross over frequency of 1.90Rad/s , this response is too fast for the inner climb rate system and will need to be redesigned or discarded. The P controller exhibits a cross over frequency of 0.91Rad/s , this produces a ratio of 3.03 between the inner and outer loops and a phase margin of 71.5° . The phase of the system using a P controller crosses the 180° mark at 4.81Rad/s and has a gain margin of 18.7dB .

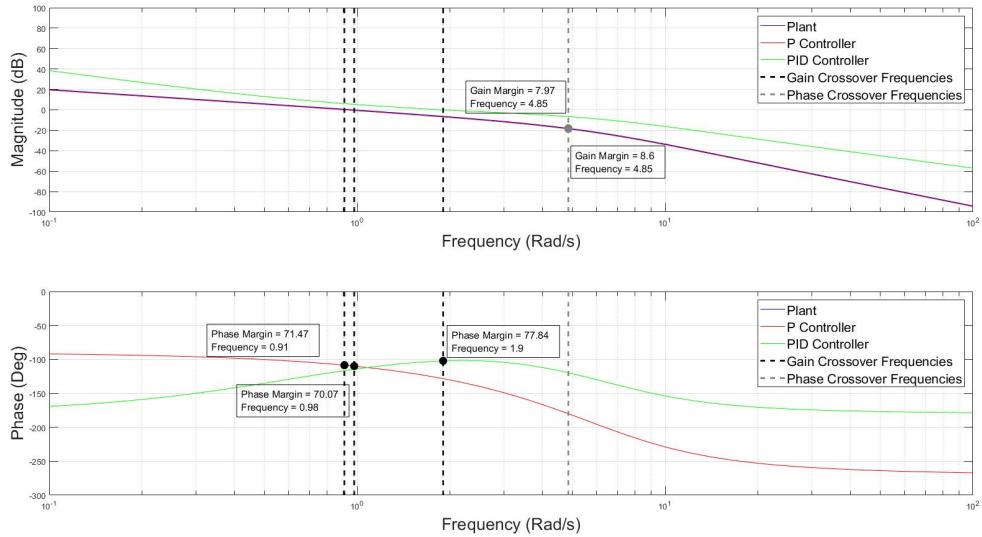


Figure 5.14: Altitude Hold Controller - Bode Plots

To finalise the design, the two limiters are discussed. The first limiter is used to limit the effect of the integrator on the system as well as stop integrator wind up. The second limiter is used to limit the climb rate commands sent to the inner controllers. Both sets of limits are shown in Table 5.4

Limit Name	Min	Max
Integrator Wind Up	-0.09	0.09
Climb Rate Command	-5	5

Table 5.4: Altitude Hold Controller Limits

Altitude Hold Controller Discussion

Although both the P and PID controllers exhibit stable dynamic responses, the PID controller exhibited too fast a response and will be influenced by the inner controllers. The system including only a P controller exhibits a step response as shown in Figure 5.15, a disturbance of $10N$ is applied to the rotors at $10s$. The system has a 5% settling time of $2.29s$ and tracks the set point with zero steady state error. The system handles the disturbance with a maximum overshoot of $0.01m$ and is critically damped.

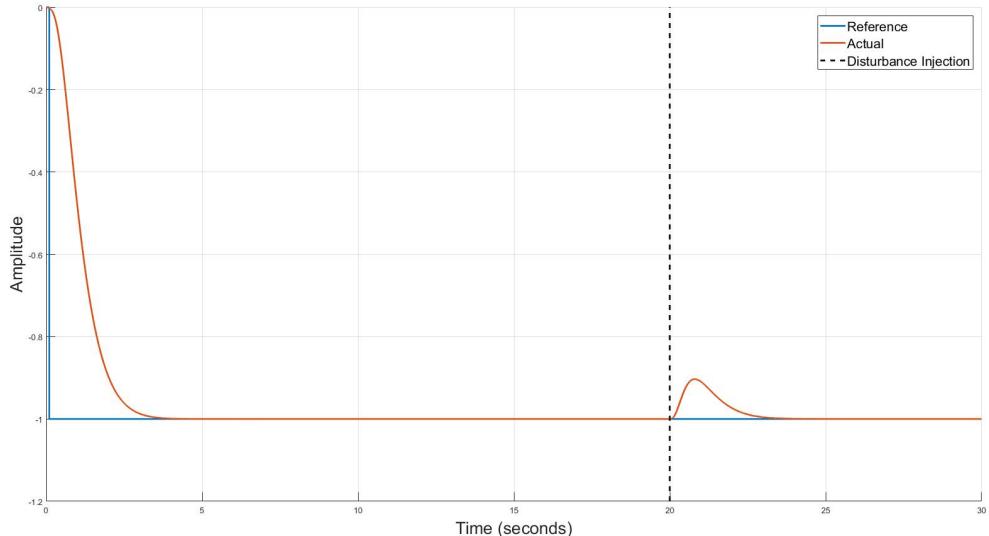


Figure 5.15: Altitude Hold P Controller - Step response

However, if a measurement disturbance is present in the inner loops, this system will not track a setpoint with zero steady state error. To demonstrate this a constant offset of $0.05m/s$ is placed on the Z-Axis velocity measurement, Figure 5.16 shows the current system cannot account for this disturbance. As shown in 5.12, a limited gain integrator is introduced into the system to help the system track steady state error.

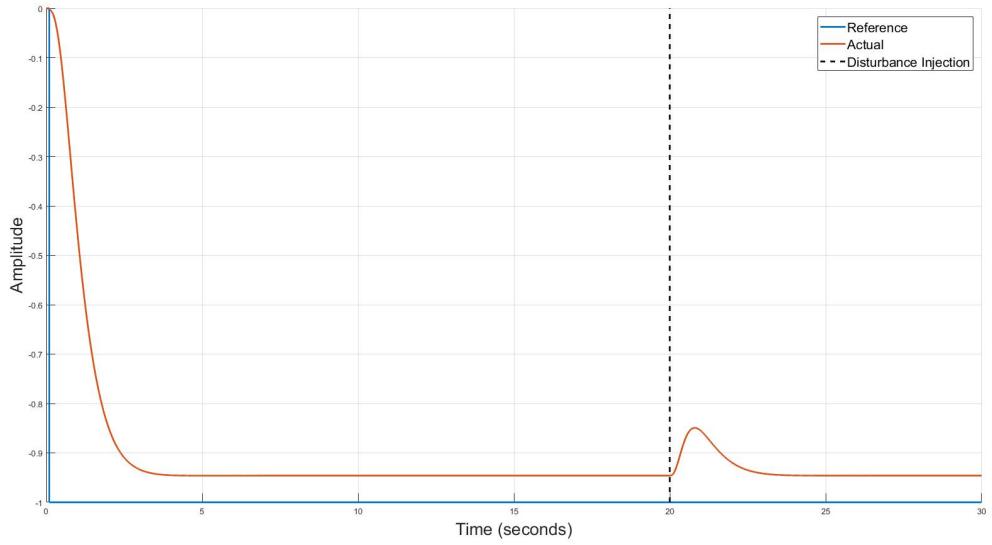


Figure 5.16: Altitude Hold P Controller - Step response with inner loop measurement offset

The new controller must be limited in such a way as to exhibit a similar transient response as the existing P controller. Figure 5.17 shows the step response of the new system both with and without the $0.05m/s$ offset in the velocity measurement. As shown, the P controller including a limited I component introduces more overshoot into the system. The limits are designed to ensure the new controller introduces less than 10% overshoot into the system.

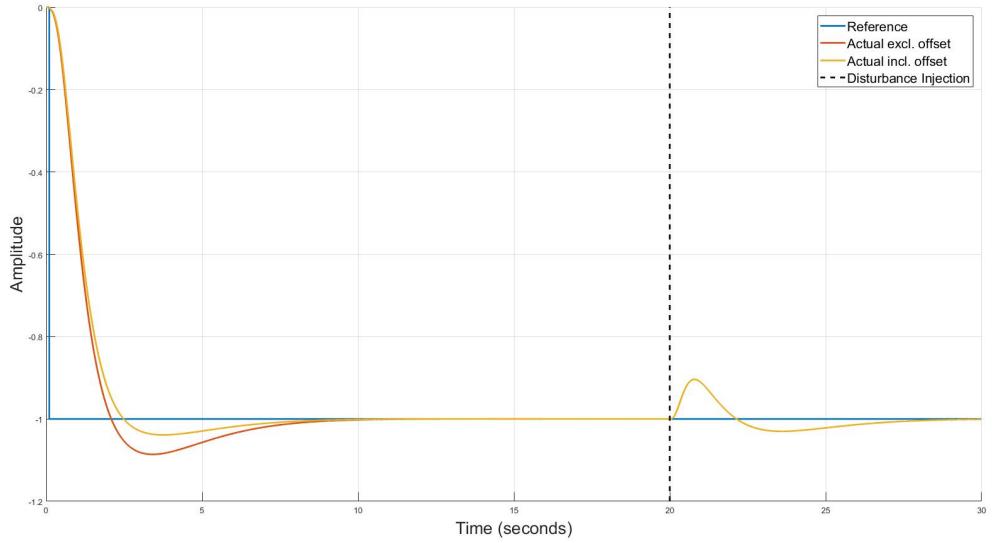


Figure 5.17: Altitude Hold P with Limited I Controller - Step Responses

5.4 Horizontal Control

This section describes the horizontal controller. This system is responsible for controlling the craft's North and East position in the earth frame, to do this the controller's most inner loop commands the pitch and roll rates of the craft. The narrow, confined spaces in which the craft must fly means it is very important for the horizontal controller to respond quickly to commands and disturbances. The limited space also limits the amount of allowed overshoot, requiring a well damped final system. The system must ensure it stay within the thrust limits as to not affect the other controllers. The controller will need to be able reject disturbances caused by wind, sensor offsets and unbalanced rotors. To handles these disturbances and other, the system will require an integrator in the controller. The integrator should be fast enough as to ensure the system stays within it's narrow, permissible flight region, even during disturbances.

The horizontal controller is designed as two sets of four cascaded control loops, one set for roll and one set for pitch. The most inner loop controls either the roll or pitch rate of the craft by commanding the virtual actuators δ_ϕ and δ_θ respectively. The desired angular rates are in turn commanded by the tilt angle controller. The tilt angle controller is responsible for converting desired translational accelerations into desired roll and pitch angles. These acceleration setpoints are commanded by the linear velocity controller which receives it's setpoint from the most outer global position controller. The global position controller will receive it's setpoint from the waypoint generation method described in a proceeding section. This section begins by deriving the plant dynamics for roll and pitch.

5.4.1 Roll and Pitch Rate Dynamics

The roll and pitch acceleration dynamics can be derived using Newton mechanics at near hover conditions and the craft's inertia around the X-axis (I_{xx}) and the Y-axis (I_{yy}) respectively, the result is shown in (5.5) and (5.6).

$$\dot{p} = \frac{L}{I_{xx}} \quad (5.5)$$

$$\dot{q} = \frac{M}{I_{yy}} \quad (5.6)$$

The rotors introduce an additional timing delay into the dynamics, as shown in (2.28). The state space equation for both systems can be derived using the current angular rates (p & q) and the current angular moments (L & M) as the system states. The state space representation for roll is shown in (5.7) and (5.8). The transfer function for roll acceleration can subsequently be calculated from the state space representation. Integrating the result produces the transfer function for roll rate as shown in (5.9). The same approach is followed for deriving the pitch rate dynamics shown in 5.10.

$$\begin{bmatrix} \dot{L} \\ \dot{p} \end{bmatrix} = \begin{bmatrix} \frac{1}{\tau} & 0 \\ \frac{1}{I_{xx}} & 0 \end{bmatrix} \begin{bmatrix} L \\ p \end{bmatrix} + \begin{bmatrix} \frac{1}{\tau} \\ 0 \end{bmatrix} \delta_\phi \quad (5.7)$$

$$y = [0 \quad 1] \begin{bmatrix} L \\ p \end{bmatrix} \quad (5.8)$$

$$G(s)_{roll} = \frac{\frac{1}{\tau I_{xx}}}{s(s + \frac{1}{\tau})} \quad (5.9)$$

$$G(s)_{pitch} = \frac{\frac{1}{\tau I_{yy}}}{s(s + \frac{1}{\tau})} \quad (5.10)$$

The roll and pitch plants both have a naturally occurring integrator, an open loop pole at $-\frac{1}{\tau}$ and no naturally occurring zeros. As shown, the plant gain is inversely proportional to the specific axis inertia. The design of the craft creates a smaller pitching plant gain than rolling plant gain. The design however, gives the pitching moment a longer torque arm, creating a larger actuation torque.

5.4.2 Roll and Pitch Rate Controllers

The roll and pitch rate controllers are the most inner loops of the horizontal controller and they command the δ_ϕ and δ_θ virtual actuators respectively. As the most inner controllers the outer loops of the horizontal controller are limited by the response and bandwidth of this system. Both these systems should then utilise as much bandwidth as the physical motor-rotor system allows. Section 2.3.5 describes some of the expected disturbances where induced moments can be caused in multiple scenarios. Including being in near proximity with a wall or having unbalanced rotor pairs. The final system must track a set point with zero steady state error. To meet the horizontal controller's requirement of fast disturbance rejection, the roll and pitch rate controllers, as the fastest controllers, should include an integrator.

The integrator will slow the system down, which can be subsequently sped up with a lead compensator. However, the final commanded motor thrusts must be validated against the limits provided in Table 5.1. The final controller architecture is shown in Figure 5.18. The controller gain values must be chosen such that the inner rate system is robust to unmodelled errors and have a sufficient gain and phase margin.

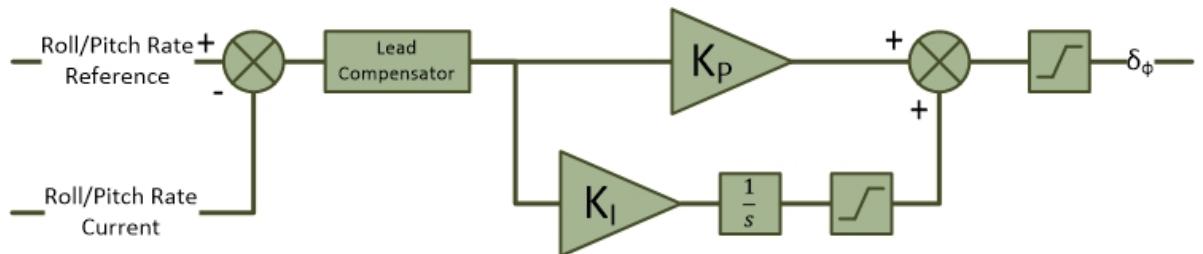


Figure 5.18: Roll and Pitch Rate Controller Design

First, the dynamic response of the controlled roll system is evaluated using the root locus shown in Figure 5.19. To maintain good damping, the two dominant poles are kept

close to the imaginary axis and have a final damping ratio of 0.9. Where the placement of the slower zero dictates how much influence the integrator can have on the system.

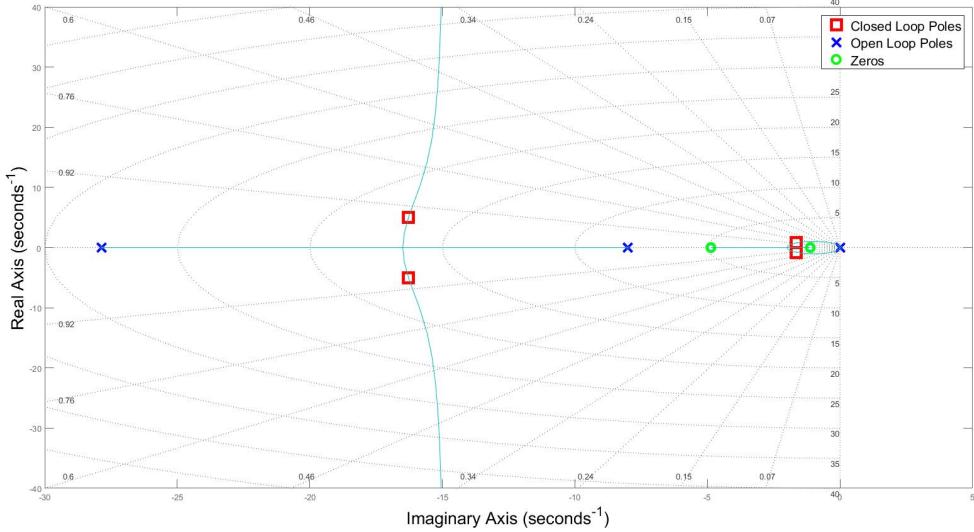


Figure 5.19: Roll Rate Controller - Root Locus

The frequency response of the roll system is then investigated using the Bode plot shown in Figure 5.20. Unity feedback is compared against the chosen controller. The high natural gain of the rolling system gives unity feedback a very fast result with too much bandwidth for the physical system to match. There is also an insufficient phase margin of 25° in the system and no offset disturbance rejection. The controller adds an integrator to reject disturbances, this however reduces phase even more and slows the system. The lead compensator is then used to increase the phase and bandwidth and reach a final phase margin of 80.6° with a crossover frequency of 4.75 rad/s.

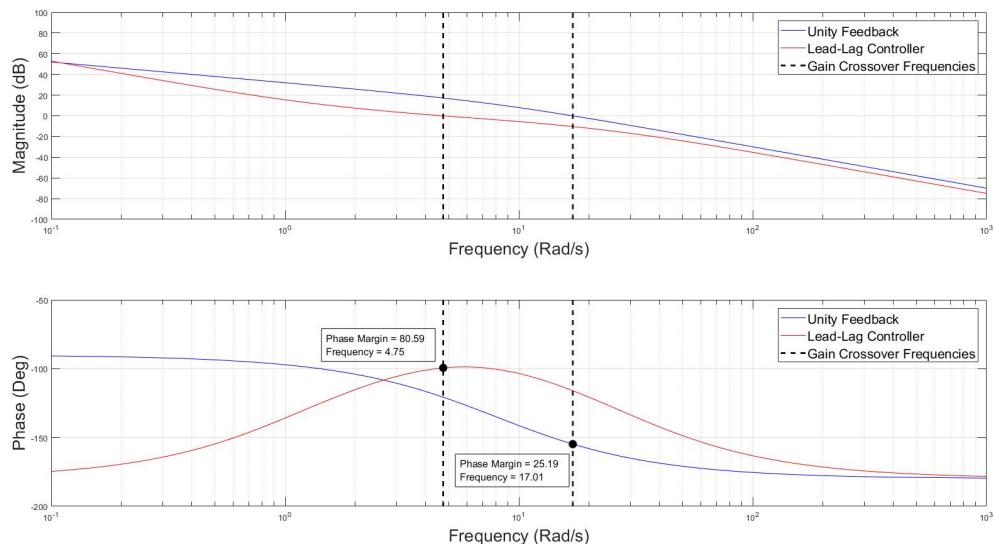


Figure 5.20: Roll Rate Controller - Bode Plot

Next the pitch rate system is evaluated. The loci and closed loop poles of the controlled pitch system can be shown to be similar to the roll system. However, the pitching plant is naturally slower and has less gain than the rolling system. The pitch rate controller is thus required to have more gain than the roll rate controller. The Bode plot shown in Figure 5.21 is used to evaluate the frequency response of the pitch rate system. Unity feedback is compared against the implemented Lead-Lag controller. As shown the natural system with unity feedback produces a much lower crossover bandwidth than the natural roll system. As with the roll rate controller, the integrator included in the controller reduces phase and bandwidth in the system but also enables disturbance rejection. To speed up the system, a similarly placed lead compensator is used. This increases the phase and bandwidth to reach a final phase margin of 82.2° with a crossover frequency of 4.72 rad/s.

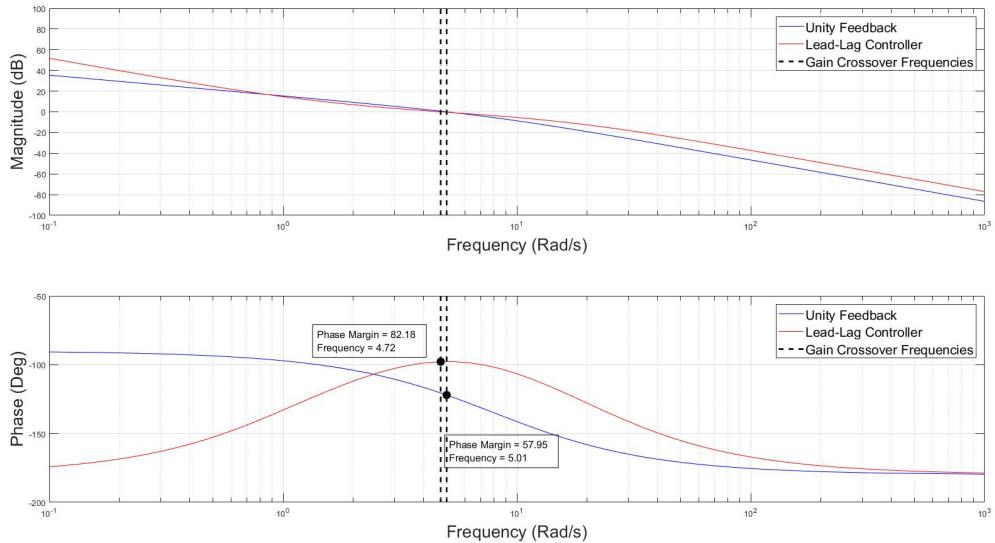


Figure 5.21: Pitch Rate Controller - Bode Plot

Roll and Pitch Rate Controller Discussion

The dynamic responses of the roll and pitch rate systems is shown to be robust, damped and they both utilise the full bandwidth allowed by the system characteristics. The integrator term in both controllers will ensure that the rate loop can handle steady state disturbances. To stop integrator wind up however, the controllers include a saturation on the integral term. Although both systems have the same controller architecture, the physical design of the craft means the roll system will have a larger plant gain. It is desired that the roll and pitch rate systems have similar closed loop responses which means the pitch rate controller needs to have increased gain compared to the roll rate controller. This unfortunately leaves the roll rate controller to be more susceptible to disturbances. The flight strategy should take this into consideration and negate rolling disturbances as much as possible.

This characteristic can be shown in the time domain using step responses and the maximum impulse required of the motors. To enable a comparison between the systems, the step response of both the roll and pitch rate system is shown in Figure 5.22. To simulate a disturbance, a 0.05 Nm loss in torque is applied to the roll system at 5 s. The

pitching system experiences a disturbance of 0.2 Nm. Both disturbances are calculated and equivalent to two rotors, on the same side, instantaneously losing 0.25 N of thrust.

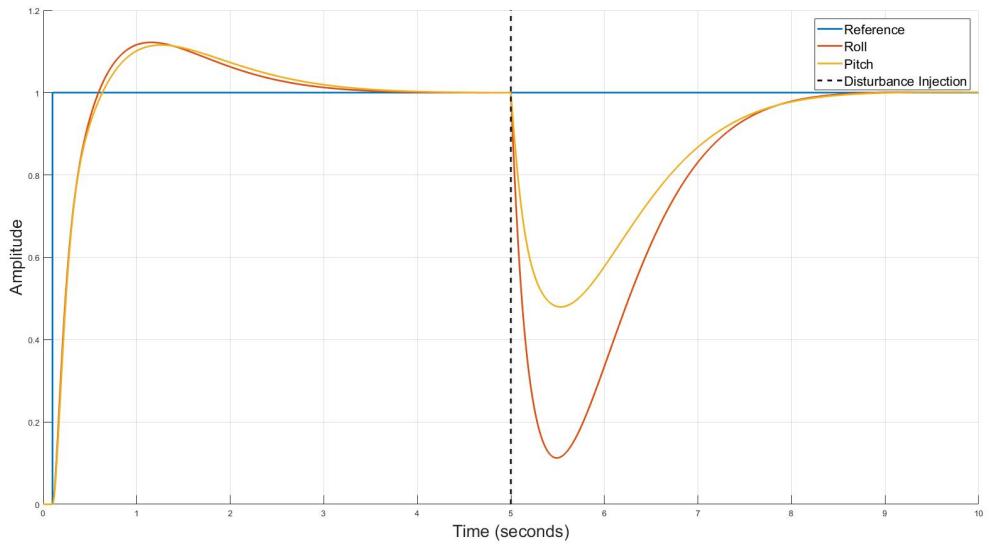


Figure 5.22: Pitch Rate Controller - Step Responses

Both the roll and pitch closed loop systems have similar transient responses. The pitch rise time of 0.33 s is very similar to the roll rise time of 0.32 s. The pitch 5% settling time is measured at 2.2 s which is also very close to the rolling settling time of 2.1 s. Both systems are similarly damped and have overshoot of 12 %. Limiting the integral term can reduce the overshoot however, this will also limit the disturbance rejection capabilities of the system. Both the roll and pitch systems handle the disturbance successfully and settle back within 5 % of the setpoint in 2.6 s. As expected, the roll system has more difficulty handling the disturbance.

The commands sent to the rotors during the roll step are shown in Figure 5.23 with a maximum commanded thrust of 0.49 N. Similarly, the pitching motor outputs are shown, in Figure 5.24, to have a maximum thrust of 0.69 N.

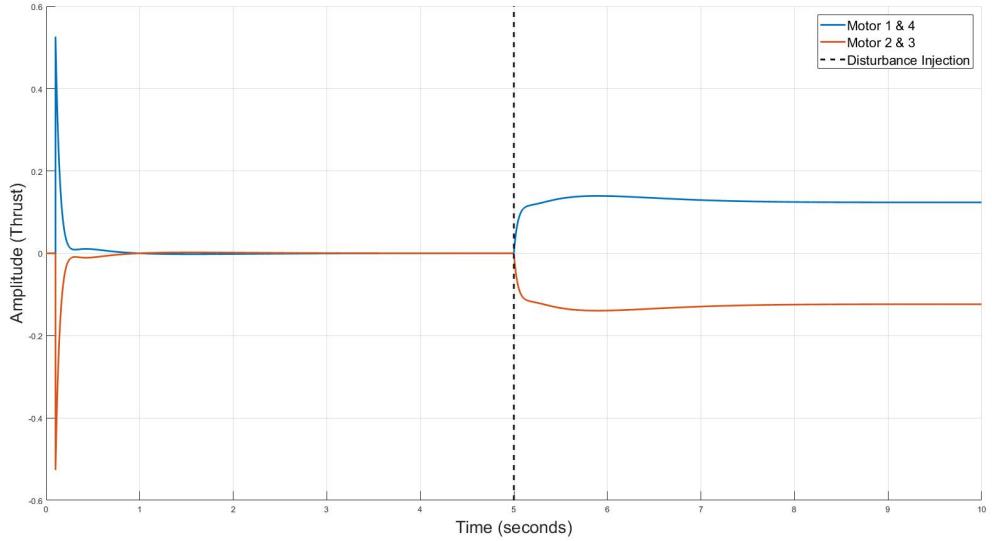


Figure 5.23: Roll Rate Controller - Motor Commands

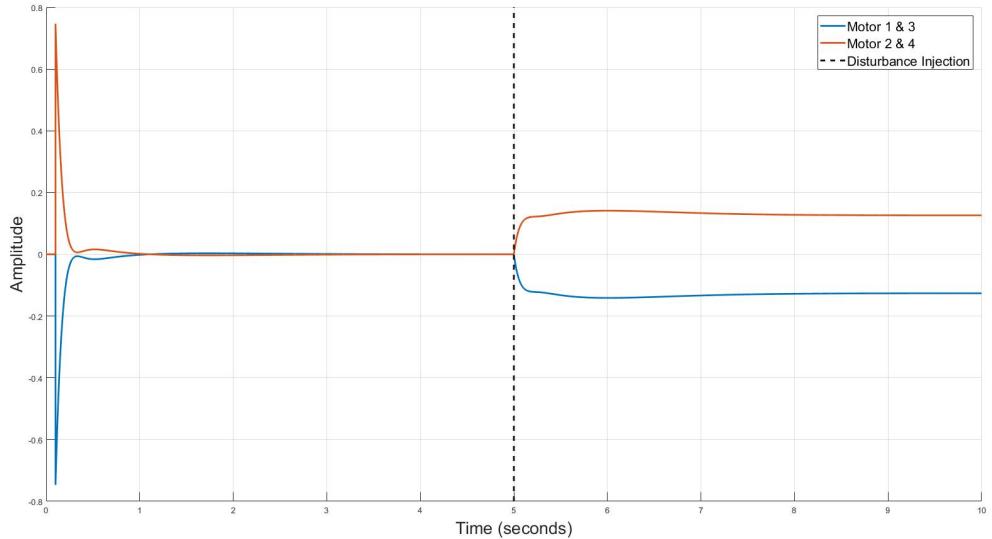


Figure 5.24: Pitch Rate Controller - Motor Commands

Intuitively it can be strange that the pitching step response produces larger motor outputs than the rolling step. The longer pitch actuator arm would lead one to believe that the pitch system will command lower values of thrust. This is only true for a similar moment. The differing inertias entails that for a rate step response the rolling plant will produce a lower moment impulse as compared to the pitching system. To quantify the effect of the arm length versus the effect of the inertias both can be represented as ratios. The ratio between the roll and pitch arm lengths is 3.73 where the ratio between the inertias is 6.76. Therefore the pitch system has to work, ratio metrically, 1.81 times harder than the rolling system resulting in larger motor thrust outputs.

5.4.3 Tilt Angle Controller

The tilt angle controller is responsible for controlling the desired roll and pitch angles of the craft. The controller does this by commanding angular rates it calculates from a translational acceleration reference in the earth frame. Using the current angular position, this earth frame acceleration reference is converted to the body frame and used to calculate the error in angular positions for roll and pitch. This error is then fed through a Proportional gain. This section makes reference to Figure 5.25 and begins by explaining the method used for converting the acceleration reference into desired angular rates.

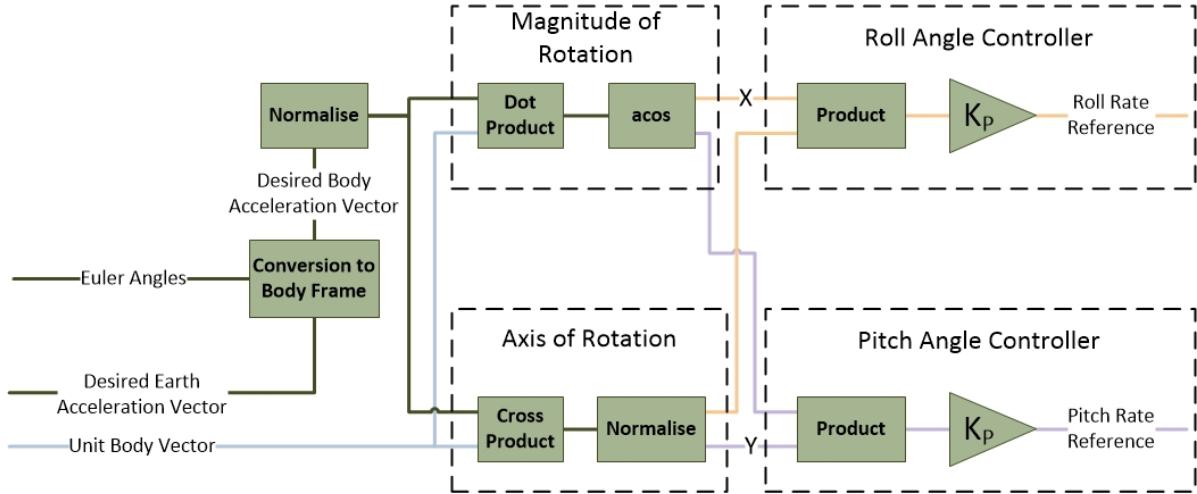


Figure 5.25: Tilt Angle Controller

Method of Conversion

The first step to calculating the desired roll and pitch angles is to convert the earth frame set point into a body frame reference. To enable the transformation, a rotation matrix is calculated from the current Euler angles as seen in (2.18). It is important to mention at this point that in order for accurate alignment, the desired earth acceleration vector must include a gravity component. The desired, now body, acceleration vector is normalised and then compared with a unit body vector. To remove any dependency on yaw, a unit Z body vector is created, which is perfectly aligned with the Z-Axis and thrust generation of the craft. Utilising the dot product shown in (5.11) the magnitude of the rotation can be calculated. Unit vectors are used, so simply taking the arc cosine of the result will produce the magnitude of rotation. The axis of rotation can subsequently be calculated by using the cross product shown in (5.12) and normalising the output to remove any magnitude. Figure 5.26 is used a visual aid for the preceding description.

$$\vec{a} \cdot \vec{b} = |ab| \cos \alpha \quad (5.11)$$

$$\vec{a} \times \vec{b} = \vec{c} \quad (5.12)$$

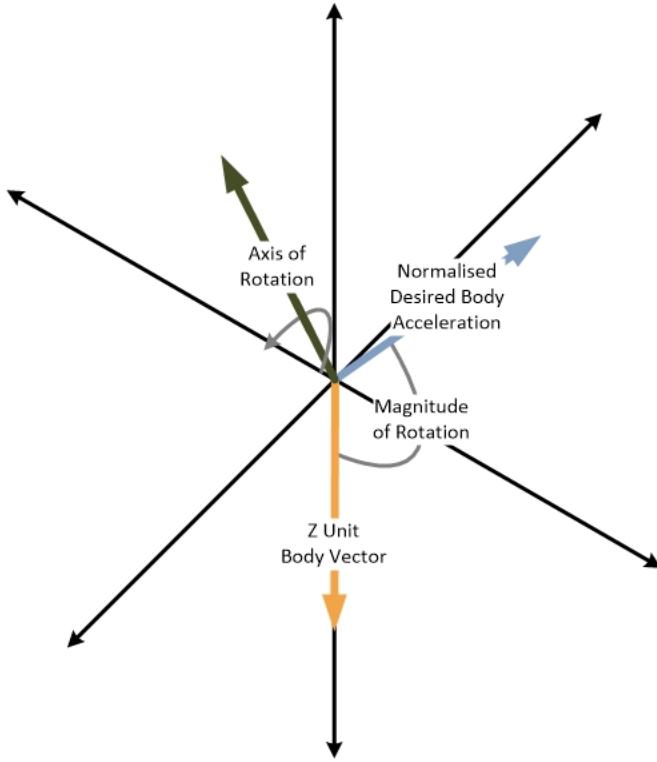


Figure 5.26: Conversion Technique using Dot and Cross Products

Roll and Pitch Angle Controllers

The linear analysis of the tilt angle controller is done by simplifying the system as shown in Figure 5.27. The additional time required to calculate the setpoints and the rotation matrix can be considered in the design by ensuring sufficient phase margin. As mentioned in the rate controller section, the outer controllers are limited by the inner loop bandwidth. The roll and pitch angle controllers must account for this by having a slower system with less bandwidth. For practical systems, the ratio between the inner and outer loop should be in the region of 2 – 4. The inner rate controller includes an integrator term and can handle disturbances, thus allowing for a less complex angle control law.

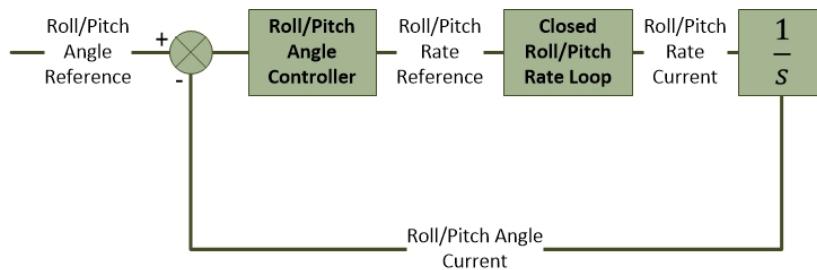


Figure 5.27: Roll and Pitch Angle Simplified Closed Loops

The roll angle loop's frequency response is shown in the Bode plot in Figure 5.28. Unity feedback is compared against the chosen controller. The integration between rate and position increases the phase in the lower frequencies producing sufficient phase, allowing for a simple Proportional (P) control law in the angle loop. The final phase margin is 78° . The controller adds a bit more gain than unity feedback and increases the

bandwidth while pushing the crossover frequency to 1.26 rad/s. There is now a ratio of 3.8 between the inner and outer loop. From observation there is still more room for a faster system and increased bandwidth. However, the damping decreases as the system is pushed harder creating a need for more complex control law, with little gain benefit.

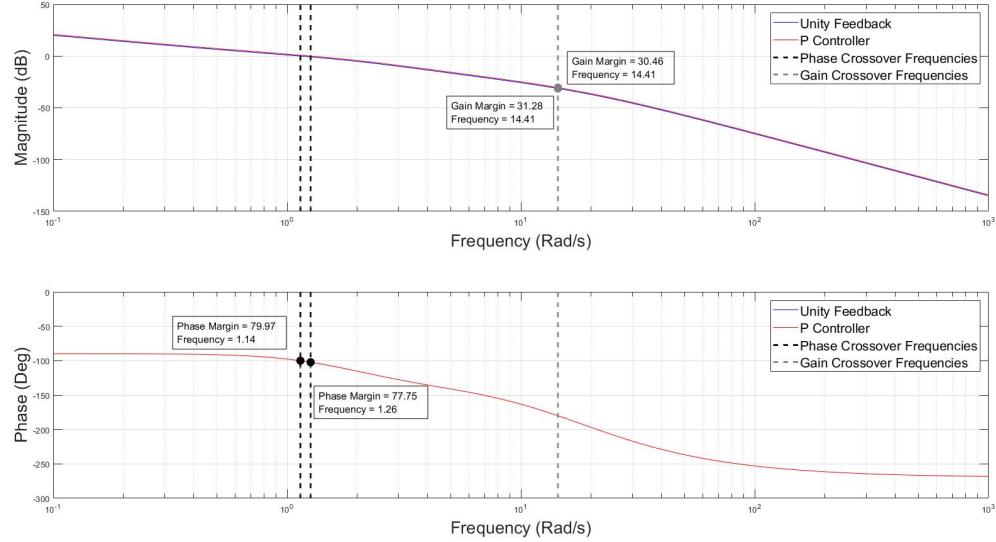


Figure 5.28: Roll Angle Controller - Bode Plots

Tilt Angle Controller Discussion

The time domain response of the system is evaluated and discussed next. To draw a comparison between the roll and pitch systems their step responses are plotted together in Figure 5.29. As desired, the roll and pitch angle transient responses are almost identical. The final rise time for both systems is 1.65 s and they both successfully have zero steady state error. The same disturbances used in the rate loop are applied to this system at 10 s. As shown, both systems handle the disturbance successfully. Although, as expected, the pitch system deviates less from the reference.

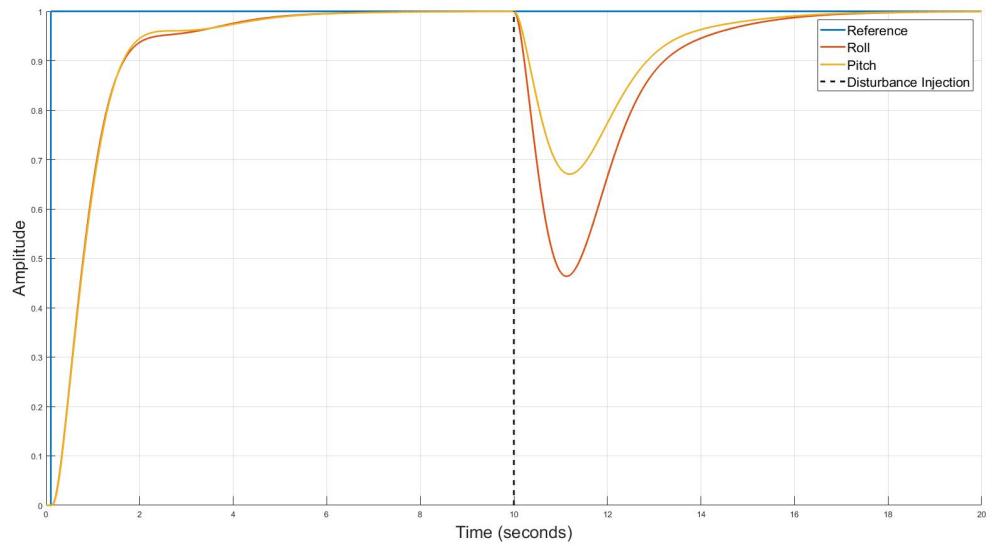


Figure 5.29: Roll and Pitch Angle Controller - Step Responses

Finally the commands sent to the motors are evaluated in Figures 5.30 and 5.31. The maximum thrust commanded by the pitch system is just less than 0.8 N. As expected the roll system commands a lower maximum of 0.57 N.

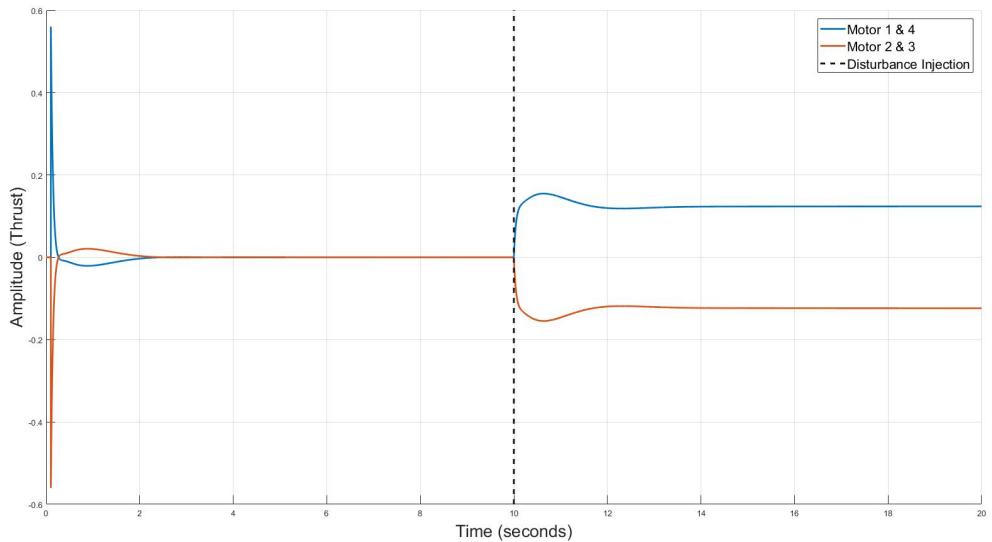


Figure 5.30: Roll Angle Controller - Motor Commands

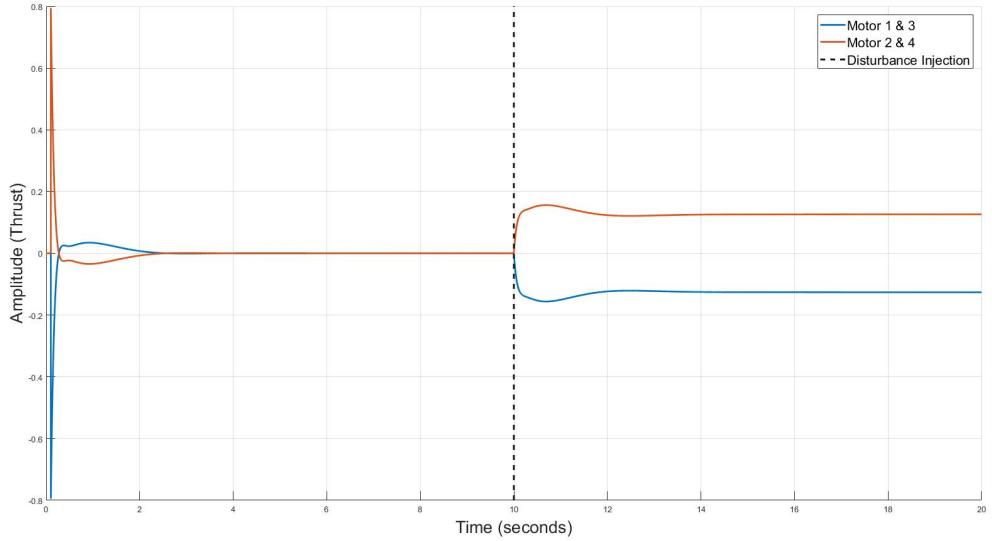


Figure 5.31: Pitch Angle Controller - Motor Commands

5.4.4 Linear Velocity Control

This section follows the design of the linear velocity controller. This controller is responsible for controlling the translational velocities of the craft along the North and East axis. This controller will receive a reference from the outer position loop and feed an acceleration command to the tilt angle controller. The tilt angle controller implementation successfully abstracts the angular position from the acceleration reference. However, the relationship between the pitch angle of the craft and North acceleration reference still requires a linearisation for the controller design.

For simplification the craft is assumed to be travelling at a maintained height in a Northern direction, with zero heading. The relationship between Northern acceleration of the craft can then be defined trigonometrically by the pitch angle of the craft as seen in (5.13).

$$\ddot{N} = g \times \tan \theta \quad (5.13)$$

At low angles, which is expected for the craft, $\tan \theta$ can be approximated to θ allowing for the linearisation seen in (5.14) and (5.15). The closed loop diagram can then also be simplified as shown in Figure 5.32.

$$\ddot{N} \approx g \times \theta \quad (5.14)$$

$$\theta \approx \frac{\ddot{N}}{g} \quad (5.15)$$

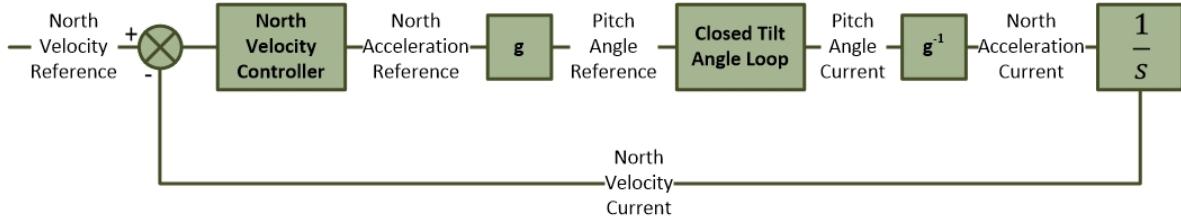


Figure 5.32: North East Simplified Closed Loops

The allowed bandwidth of the linear velocity controller is limited by the bandwidth of the tilt angle controller. The free integrator in the linear velocity loop will ensure that the system will track a set point with zero steady state error. However, there are expected disturbances which require more complex control than proportional control to reject. The Bode plots in Figure 5.33 assist with the design by allowing easy analysis of phase and gain in the system.

A traditional PI architecture increases the low frequency gain, however was not suitable due to the loss in phase and damping. Instead a lag compensator could be designed to limit the overshoot while enabling some disturbance rejection. The process of designing the lag compensator went through the following steps. First a proportional controller is designed to achieve the desired bandwidth, ω_{des} . The zero of the compensator is then placed far enough to negate any effect on the bandwidth. The pole has been placed to optimise both limiting overshoot and enabling disturbance rejection.

As desired the P and lag controlled systems exhibit the same crossover frequency bandwidth and negligibly different high gain profiles. Both the lag compensator and the PI controller increase the low bandwidth gain at the cost of some phase. However, the phase benefits of the lag compensator compared to the PI controller can be seen clearly. The final system is designed to have a crossover frequency of 0.42 rad/s and a phase margin of 65°. This bandwidth is a ratio 2.7 slower than the slowest loop in the tilt angle controller.

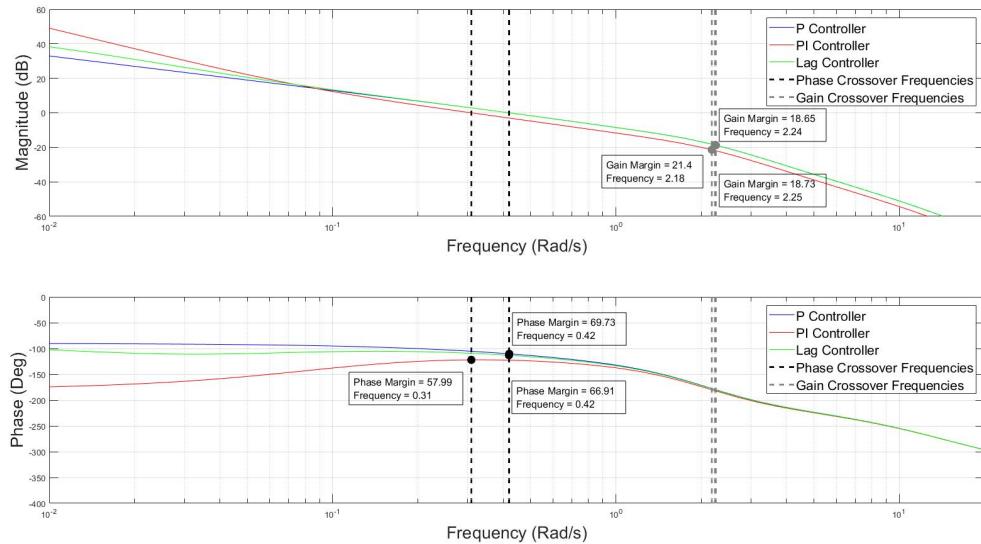


Figure 5.33: North Velocity Controller - Bode Plots

The final placement of the lag compensator can be shown on the root locus in Figure

5.34. The compensator zero has been placed at $\frac{\omega_{des}}{10}$ with the pole a factor of 4 slower.

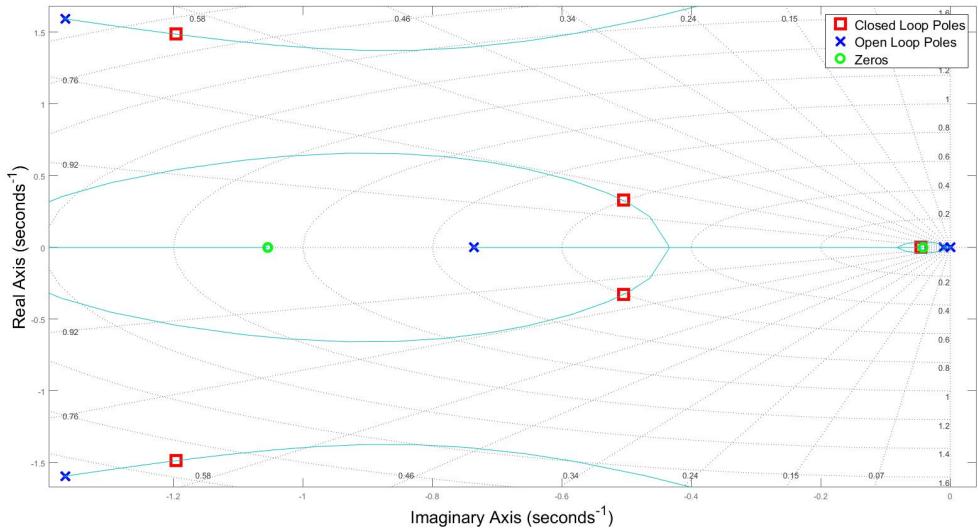


Figure 5.34: North Velocity Controller - Root Locus Plot

Linear Velocity Controller Discussion

The three controllers all exhibit a stable dynamic response, the differences in gain and phase were identified and discussed in the Bode plot. The time domain responses and differences can now be evaluated and discussed. The step response of the P, PI and lag controllers are shown in Figure 5.35. The proportional controller has a fast transient response with a rise time of 3.3 s. The P controller exhibits good phase margin and shows little overshoot. The loss in phase of the PI controller presents itself as a large overshoot of 20%. The integrator introduces a long tail into the system and slows the transient response to a rise time of 4.9 s. The lag compensator has some overshoot and a very similar transient response to the P controller.

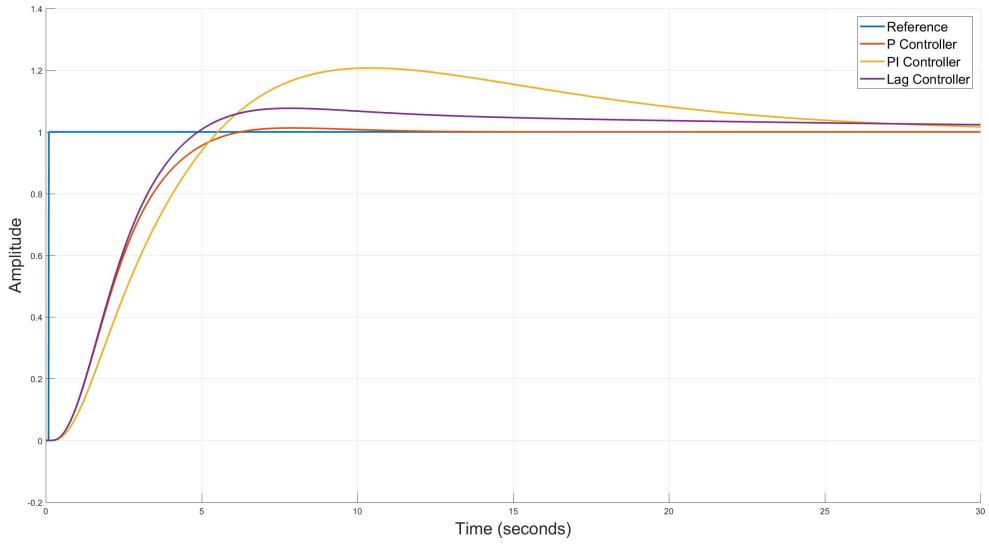


Figure 5.35: North Velocity Controller - Step Responses

The benefit of the lag compensator over the P controller can be seen when a disturbance is introduced into the system. Figure 5.36 is used to show the effect of a constant disturbance in the system by adding an external force at 30 s. The P controller is unable to reject the disturbance. The increased low bandwidth gain of the lag compensator manages to reduce the disturbance and as expected the PI controller successfully rejects the disturbance completely.

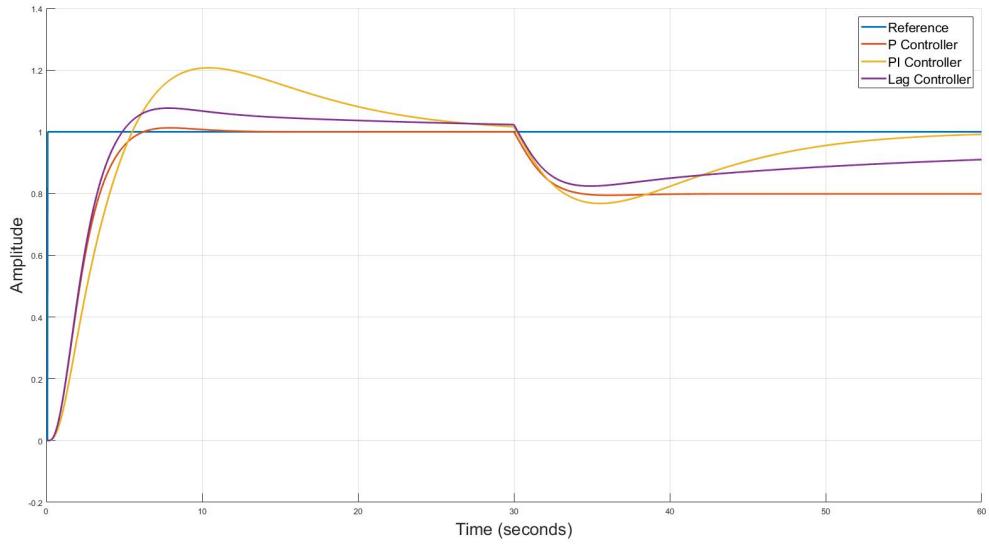


Figure 5.36: North Velocity Controller - Step Responses With a Disturbance

5.4.5 Global Position Tracking Control

The position controller is the most outer loop of the horizontal controller and will be fed a reference from a waypoint generator or some other, high level flight strategy. There is

sufficient disturbance rejection in the inner loops allowing the position controller to use a simple proportional controller. The final bandwidth should utilise the full potential of the inner loop velocity system. The craft should approach a set point steadily and with little overshoot, the final system should thus be well damped.

The Bode plot in Figure 5.37 is used for the design as it easily shows the phase and gain margins of the system. The plant is on the edge of stability and is compared with a controlled system utilising a P controller. The proportional gain is adjusted until there is sufficient phase margin of 69° and gain margin of 17 dB. The final cross over frequency is 0.16 rad/s, creating a ratio of 2.6 between the inner and outer loop.

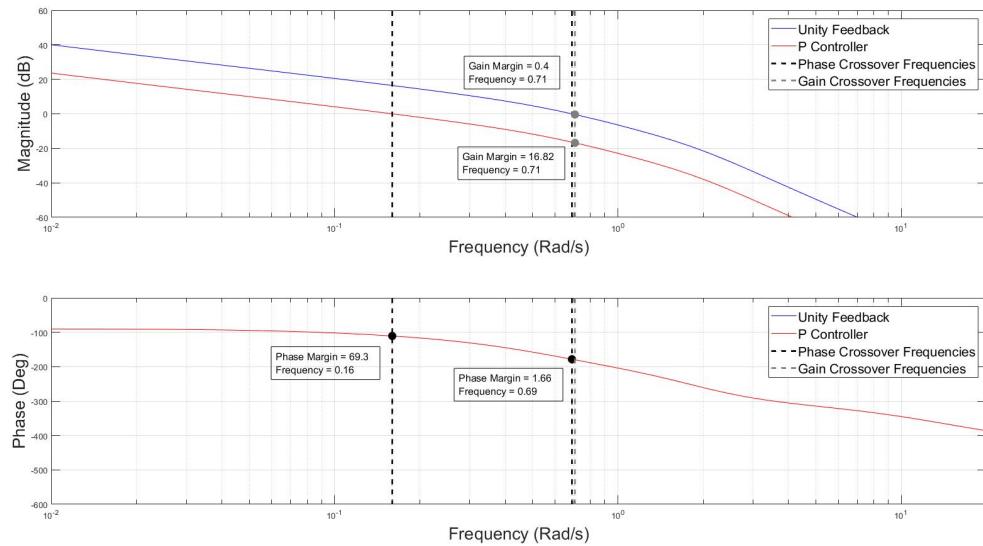


Figure 5.37: North Position Controller - Bode Plots

Position Controller Discussion

The proportional gain increase stability in the system and reduces the crossover frequency. Figure 5.38 represents the step response of the system. The system is shown to be well damped with no oscillatory motion in the response.

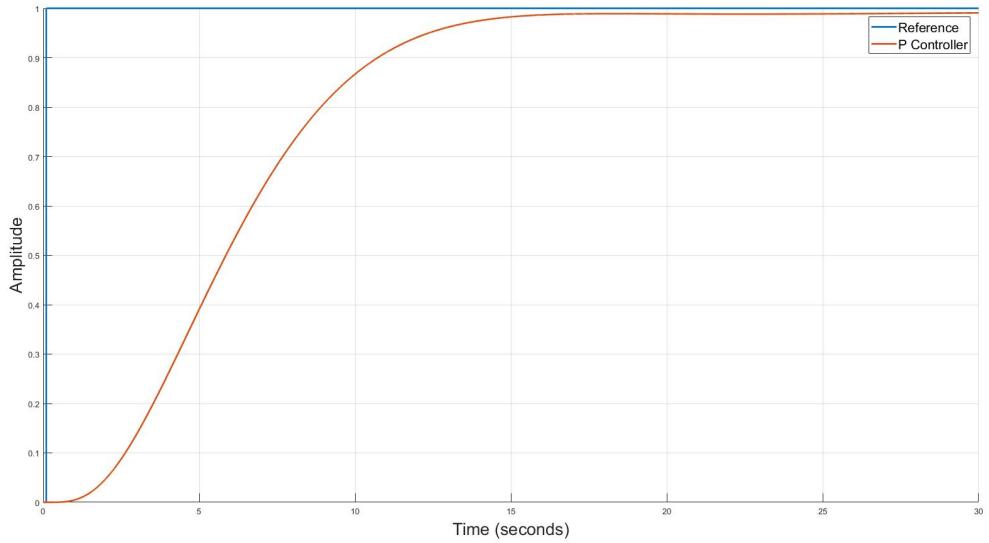


Figure 5.38: North Position Controller - Step Response

The position controller could be commanded with large step values. To prohibit commanding large velocity values a limiter is used. Figure 5.39 is used to show the effect the saturation has for a large step input command.

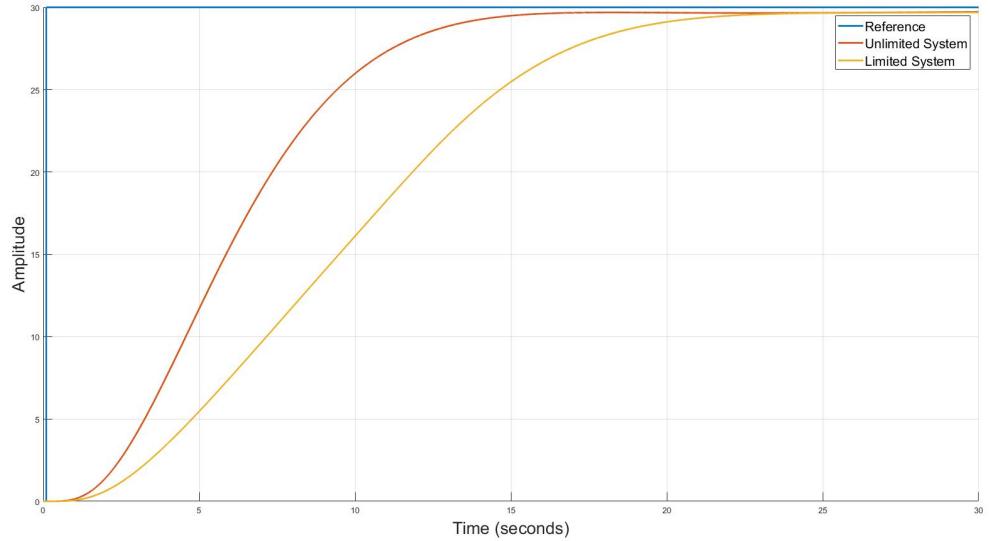


Figure 5.39: North Position Controller - Large Step Response With and Without a Limiter

5.5 Heading Controller

This section describes the heading controller which is responsible for aligning the craft with a desired yaw angle reference. An angle reference is given to a yaw angle controller which outputs a yaw rate reference. The inner yaw rate controller commands the yawing

moment around the Z-Axis of the craft. Both the vertical and the horizontal controllers have been designed to operate independently of the heading. The craft is however, expected to fly down a narrow channel. This calls for a method of aligning the body axis of the craft with a given heading in the earth frame. This heading could be given by a higher flight planning strategy.

The craft must be able to follow a heading setpoint with zero steady state error and have a reasonably damped response. The heading controller must also be able to reject disturbances and will require an integrator in the control law. The yaw controller must also consider that the yaw torque generation has a reduction gain due to the lift to drag ratio. The design of the craft also means the yaw rate dynamics produce the lowest plant gain with the largest inertia component. The juxtaposition of a low actuation torque and lower plant gain leads to the heading controller typically being slower and exhibit less bandwidth when compared to the other controllers. Before a controller can be designed, the plant dynamics must be derived.

5.5.1 Yaw Rate Dynamics

Using Newton mechanics at near hover conditions, the yaw dynamics for the craft can be derived, the result is shown in equation (5.16). \dot{r} is the rotational acceleration of the craft and N is the instantaneous moment experienced by the craft around the Z-Axis.

$$\dot{r} = \frac{N}{I_{ZZ}} \quad (5.16)$$

\dot{r} is chosen as the output of the system with the state variable chosen as N . From this, the space equation for the system can be derived and is shown in (5.17) and (5.18).

$$[\dot{N}] = -\left[\frac{1}{\tau}\right] [N] + \left[\frac{1}{\tau}\right] [\delta_\psi] \quad (5.17)$$

$$[\dot{r}] = -\left[\frac{1}{I_{ZZ}}\right] [N] \quad (5.18)$$

$$G(s)_{yaw} = \frac{\frac{1}{\tau I_{ZZ}}}{s(s + \frac{1}{\tau})} \quad (5.19)$$

From the state space representation, the transfer function for the yaw acceleration can be calculated. Integrating the result produces the transfer function for yaw rate, introducing a new pole into the system, the result is shown in (5.19). This plant now has two open loop poles, the first pole is due to the lag introduced by the motor rotor system, and lies at $\sigma = -\frac{1}{\tau} = 8$ with the second due to the integration of yaw acceleration to velocity.

5.5.2 Yaw Rate Controller

The yaw rate controller receives a yaw acceleration reference in radians per second (rad/s) and outputs the virtual actuator σ_ψ . The yawing moment is generated by air pressure on the rotors as they generate thrust, the reduction gain introduces the possibility of saturating the other controllers for large step inputs. However, as the most inner of the two heading loops, the yaw rate system limits the bandwidth of the outer yaw angle

loop. The yaw rate controller must then produce enough bandwidth for the yaw angle controller, while ensuring it is not commanding thrust values above the limits described in 5.1. The yawing moment torque generation is also less accurately modelled and the system must exhibit high stability with large gain and phase margins. The free integrator in the yaw rate system will produce zero steady state error. The design for this controller can then be designed as a simple P controller with a non-linear saturation as shown in Figure 5.40.

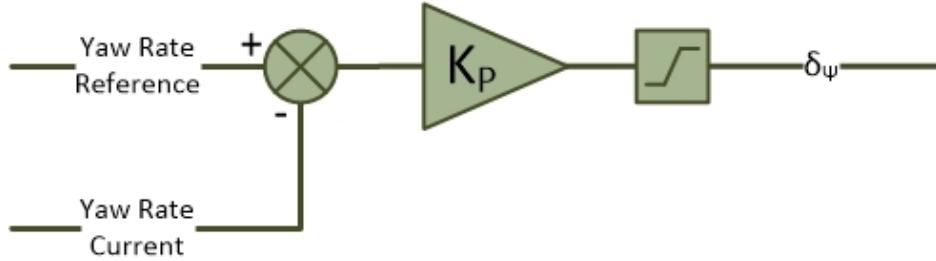


Figure 5.40: Yaw Rate Controller - Control Diagram

The dynamic response of the proportional (P) controller is evaluated using the root locus in Figure 5.41. The controller adds no new poles or zeros. The proportional gain is used to move the closed loops and achieve the desired bandwidth. The final closed loops poles sit at -4 ± 1.93 rad/s, the poles are slightly under damped with a damping ratio of 0.9.

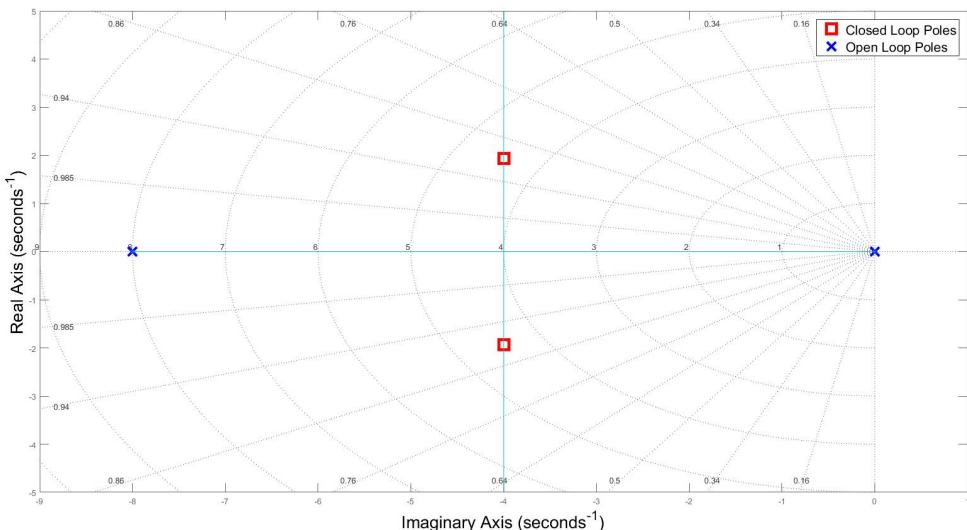


Figure 5.41: Yaw Rate Controller - Root Locus

The bode plot in Figure 5.42 is used to evaluate the frequency response of the system against unity feedback. As expected the controller introduces no phase change into the system. Unity feedback produces a crossover frequency of 4.99 rad/s which is too fast for the heading system. Reducing the gain of the system increases the phase margin to 74° and reduces the crossover frequency by nearly half to 2.37 rad/s.

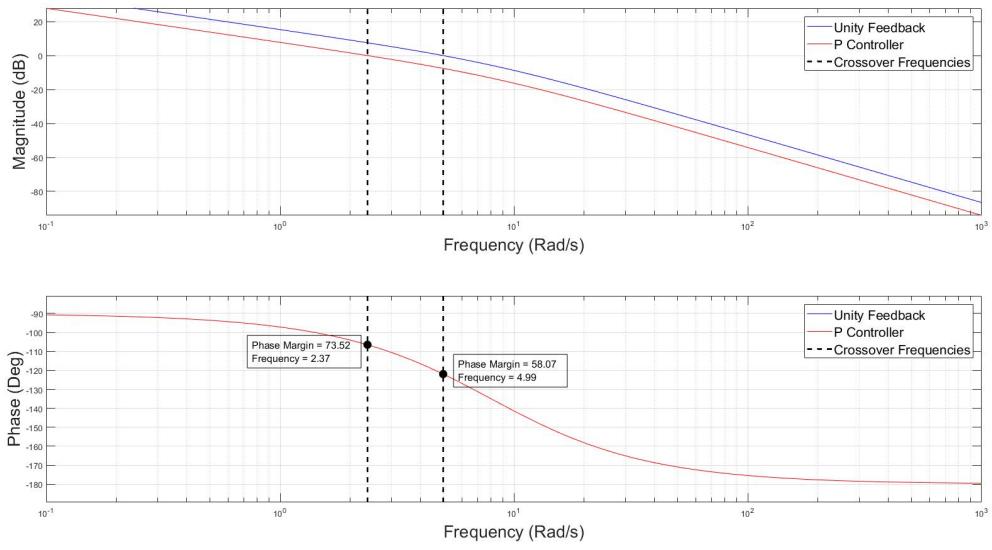


Figure 5.42: Yaw Rate Controller - Bode Plots

Yaw Rate Controller Discussion

The controller increases stability in the system and produces the step response shown in Figure 5.43. The system has a 5% settling time of $0.9s$ and negligible overshoot. The maximum thrust commanded per motor using this system is 0.26 N , which falls in the limits of this system. The low gain of the system makes this system susceptible to disturbances and calls for an outer angle loop.

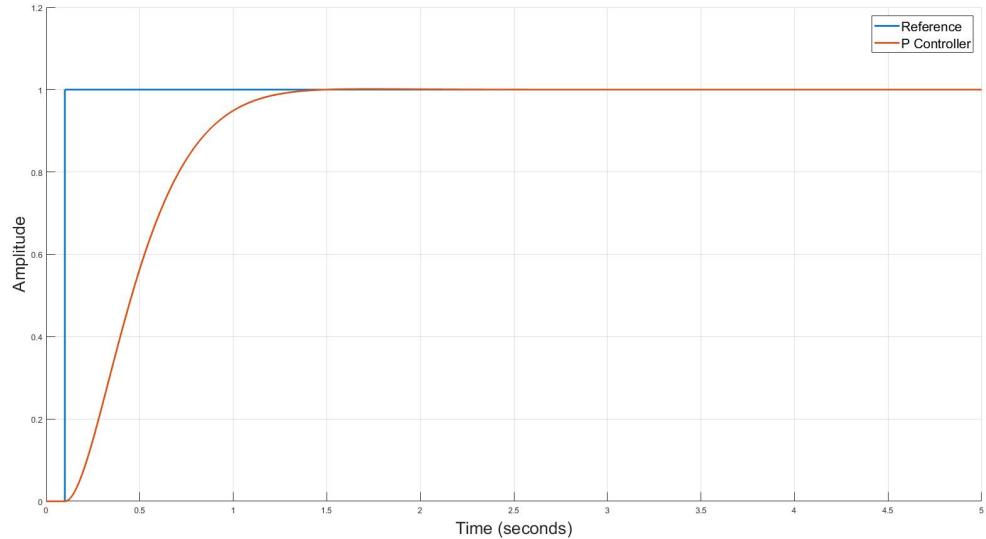


Figure 5.43: Yaw Rate Controller - Step Responses

5.5.3 Yaw Angle Controller

The yaw angle controller receives a heading reference in radians and outputs a yaw angle rate reference in radians per second to the inner rate controller as shown in Figure 5.44. This controller is limited by the inner loop and must ensure significant bandwidth between the inner and outer loops. The system must be able to reject disturbances and requires an integrator in the system. The system must be reasonably damped with limited overshoot.

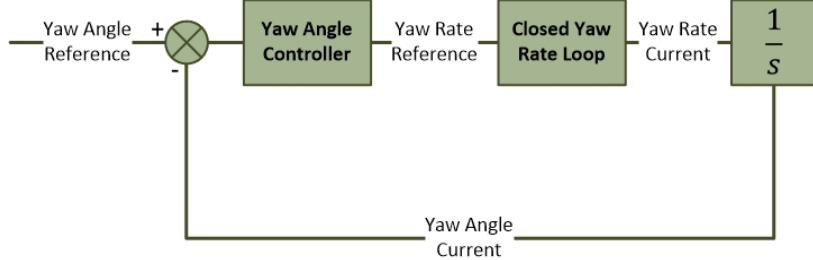


Figure 5.44: Yaw Angle PI Controller - Control Diagram

Initially a Proportional Integral (PI) controller was considered as shown in Figure 5.45. The proportional gain is used to move the closed loop poles and achieve the desired bandwidth. The integral term is introduced to account for expected disturbances as well as measurement errors in the rate loop. The PI controller adds a new zero and a new pole into the system.

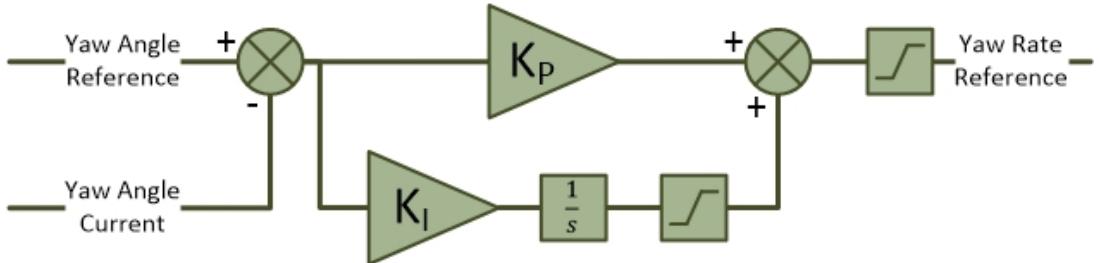


Figure 5.45: Yaw Angle PI Controller - Control Diagram

The second scheme was designed as a Proportional Integral Derivative (PID) controller as shown in Figure 5.46, the differential term is introduced to increase the phase of the system and adds an additional zero. The differential command is fed through a low pass filter to reduce noise on the command. As shown both controllers contain non linear elements that are not considered during the linear design. The components excluded during the analysis are all the limiters as well as the low pass filter seen in the differentiator portion of the PID leg.

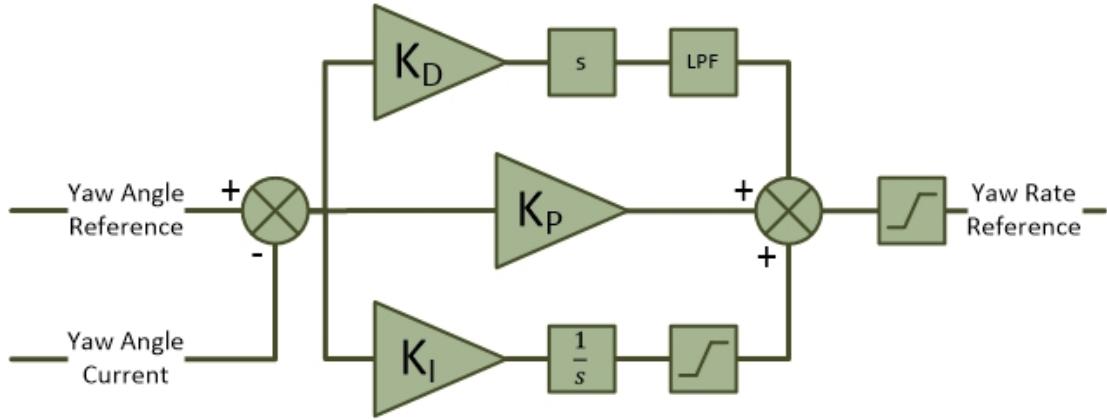


Figure 5.46: Yaw Angle PID Controller - Control Diagram

The dynamic response of each system can be evaluated using the root loci diagrams seen in Figure 5.47. The plant has two open loop poles at the closed loop yaw rate pole locations, as well as a new pole at the origin introduced by the mathematical relationship between speed and position. Both controllers introduce one new open loop pole at the origin and at least one zero. The PID controller introduces an additional zero into the system.

The final closed loop pole positions for the PI controller all lie on the imaginary axis and are located at -3.62 , -3.26 , -0.88 and -0.25 . The PID controller has a dominant complex pair of closed loop poles at $-0.52 \pm 0.51i$ with the other non-dominant closed loop pole pair sitting at $-3.48 \pm 2.54i$. The PI controller has critically damped dominant poles whereas the dominant poles for the PID controller are under damped with a damping ratio of 0.71.

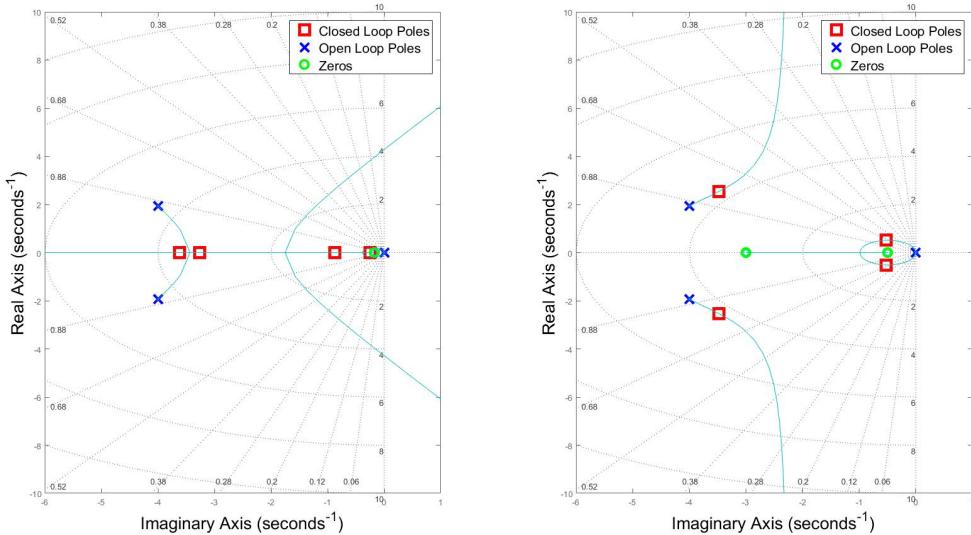


Figure 5.47: Yaw Angle Controller - Root Locus (Left:PI, Right:PID)

The frequency response of both systems is evaluated next. Using the open loop bode plots shown in Figure 5.48, unity feedback is compared with a PI and PID controller. The first zero in both cases is placed close to the origin to limit the effect the new zero

has on the system. The PID controller's second zero is placed to increase the phase of the system, allowing for more bandwidth in each leg of the controller. This additional phase allows for larger and more aggressive disturbance rejection, but will result in larger setpoints for the inner yaw rate loop.

The gain of each system has similar bandwidth around the cross over point. The extra zero in the PID controller reduces the gradient of the gain slope off and increases the total phase of the system. The PID controller exhibits the second largest phase margin of 61° which is found at 1.12 rad/s , the fastest of the three crossover frequencies. The PI controller achieves the desired bandwidth and has the slowest crossover frequency of 0.75 rad/s , this however relates to a lower phase margin of 59° . The final crossover frequency of the yaw rate system was 2.37 rad/s , resulting in a ratio with the PID controller of 2.12 and a ratio of 3.16 with the PI controller. A larger ratio implies less risk of attenuation for the outer loop.

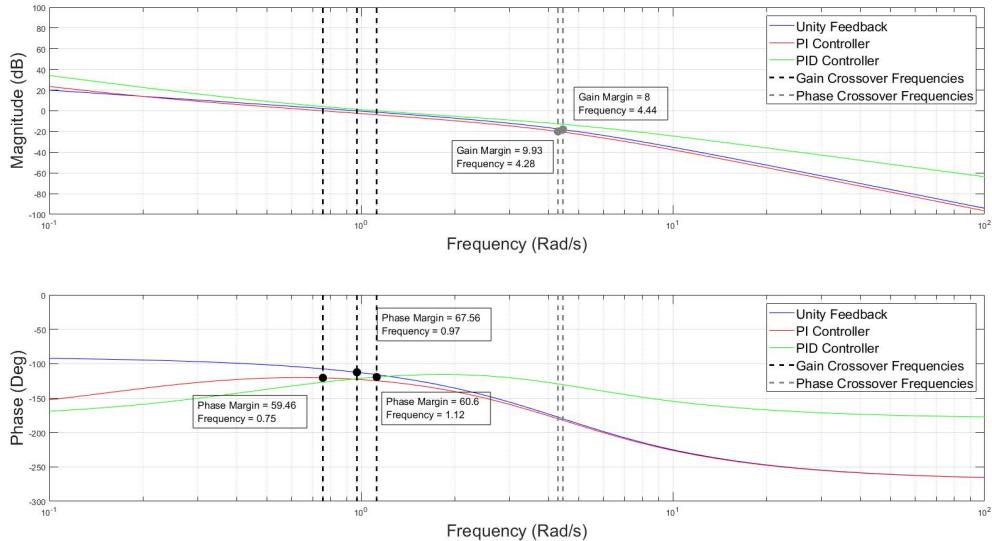


Figure 5.48: Yaw Angle Controller - Bode Plots

Yaw Angle Controller Discussion

Each controller was added to the non-linear simulation and evaluated in the time domain including the non linearities previously unconsidered. Figure 5.49 shows the results of both controllers with and without the limits as well as the additional low pass filter on the differential gain. The PID controller without any limits or low pass filtering had a 5 % settling time of $6.37s$, adding in the limits and filter decreases that time to $4.9s$. The PI controller had a settling time of $10.3s$ with no limits which was decreased to $8.8s$ by the addition of the limit on the integrator.

All three systems exhibit some overshoot. The limiters must be carefully chosen to reduce overshoot while also still allowing for substantial disturbance rejection. The linear PI and PID systems produce overshoot of 17% and 29% respectively. The limits for the integrators on both systems was set to $\pm 0.1 \text{ rad/s}$. This limit also becomes the maximum offset this controller can successfully correct for. Both systems had significant overshoot reduction, the PI controller now only had an 12% overshoot, with the limited PID system showing only 6% overshoot.

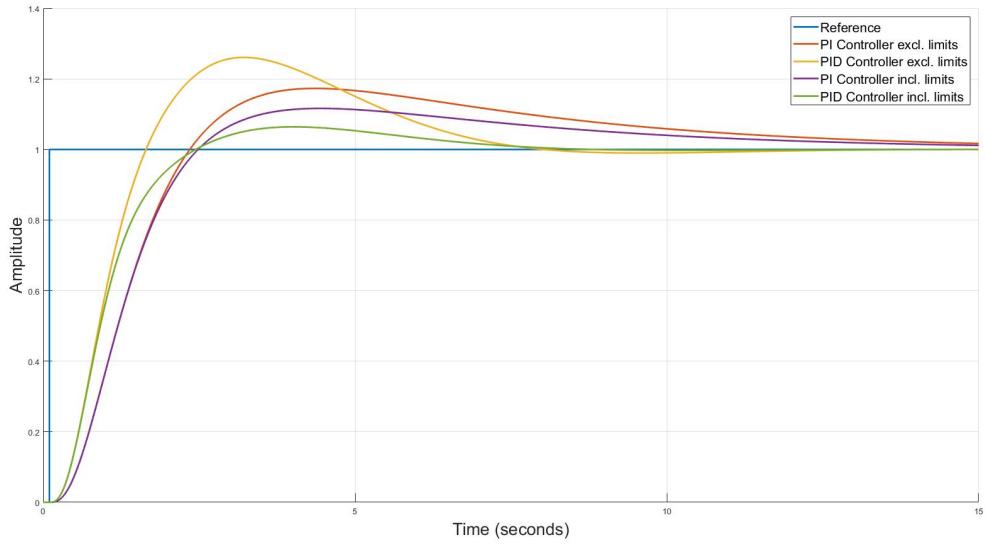


Figure 5.49: Yaw Angle Controller - Step Responses

As mentioned the limits introduce the maximum disturbance rejection capability of each system. Figure 5.50 shows how each limited system handles a measurement offset of 0.1 rad/s in the yaw rate loop.

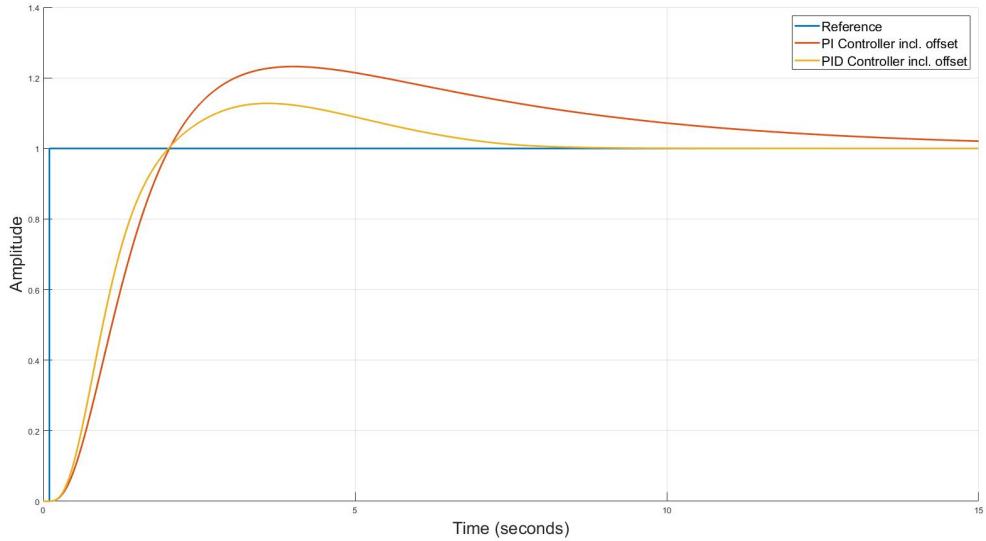


Figure 5.50: Yaw Angle Controller - Step Responses Including Inner Loop Measurement Offset

The final consideration is the impulse each system creates for the yaw rate controller, Figure 5.51 demonstrates both limited systems impulse responses. As seen the PID controller commands a larger initial setpoint, more than double that of the PI controller.

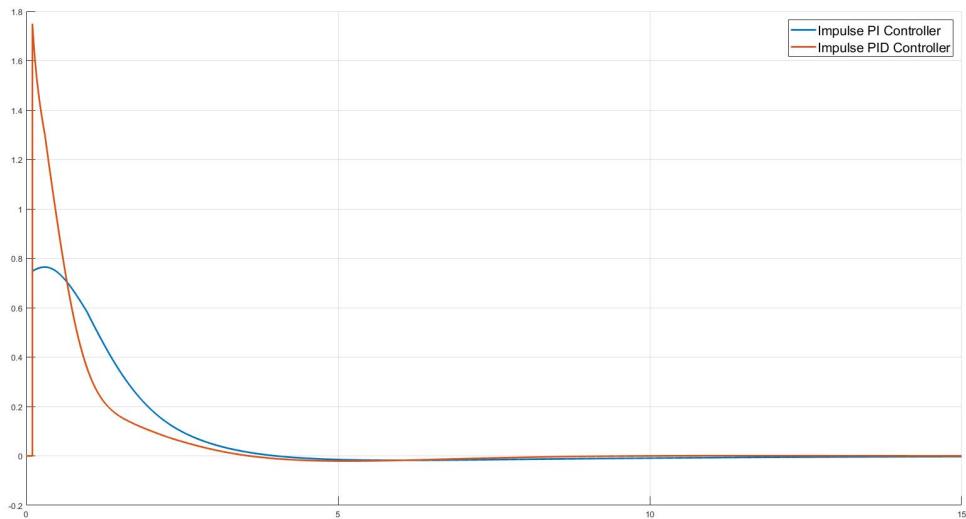


Figure 5.51: Yaw Angle Controller - Impulses

The PI controller is chosen as the final controller implementation.

Chapter 6

Flight Strategy and Obstacle Avoidance

This chapter describes the proposed and implemented flight strategy, including the obstacle avoidance technique. Now that stable flight has been achieved a more advanced flight routine can be implemented. The chapter begins by describing the proposed flight strategy for missions in a narrow corridor and confined space. After the flight strategy has been discussed, an obstacle avoidance routine is proposed and implemented. The obstacle avoidance methodology is discussed, including sensor placement and continues on to describe the mathematical modelling of the environment and the sensors for use in the simulation. Once the modelling is adequate, the avoidance controller design is described and is followed by the testing of the routine. The obstacle avoidance routine is based on proximity measurements to the aircraft's immediate surroundings.

6.1 Flight Strategy

This section describes the flight strategy proposed for flight inside a confined narrow corridor. The first step to creating a mission will be to implement a waypoint generator, allowing for mission locations to be preloaded into the craft and followed sequentially. The platform design has been previously discussed and was chosen to be a narrow, elongated design. For an aircraft of this design it is deemed beneficial for constant yaw alignment as it traverses down the corridor. The section first describes the waypoint generator proceeded by the strategy for yaw alignment.

6.1.1 Waypoint Generation

The waypoint generator is implemented as a buffer of North, East and Down position references. The references are preloaded into the drone at mission start through a ground station. Once a mission is begun the set points are fed into the relevant position controllers. Although not required, it is suggested that the first waypoint is always the starting North East location with a desired altitude reference. This allows the drone to first reach a desired height before continuing the mission.

The ground station can program an acceptable position error limit for each waypoint. Once the limit in all three axes is reached, the waypoint generator begins a timer which, when lapses, steps on to the next waypoint in the list. The ground station also has the option to move onto the next waypoint prematurely if desired.

Figure 6.1 shows a simple followed flight path set up by the waypoint generator. The blue line shown is the flown path, while the cyan dots represent the waypoints.

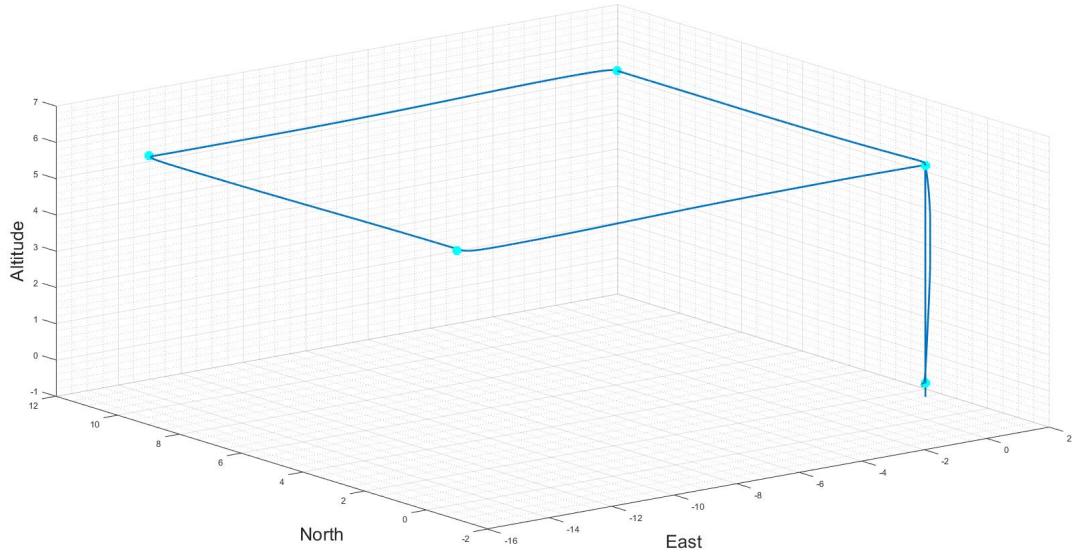


Figure 6.1: Simple Waypoint Flight

6.1.2 Yaw Alignment

The strategy for yaw alignment is to align the X-Axis of the drone's body frame with the current velocity vector. This achieves a constantly forward facing drone assisting in tight spaces as well as a more predictable sensor feedback for missions requiring additional sensors.

The yaw angle controller designed in Section 5.5.3 works on the basis of a yaw angle error. The heading alignment controller would be required to calculate a yaw error to be controlled by the yaw angle controller. Similarly to the tilt angle controller, this can be calculated using the dot and cross product between two vectors. The first vector is the current velocity in the body frame, while the second vector is the desired alignment vector. For alignment of the X-Axis the vector would be a unit vector containing only an X component. The X and Y components of the body velocity are extracted to calculate the magnitude of rotation where the axis of rotation should always be around the Z-Axis. The cross product is utilised to calculate the direction of the rotation.

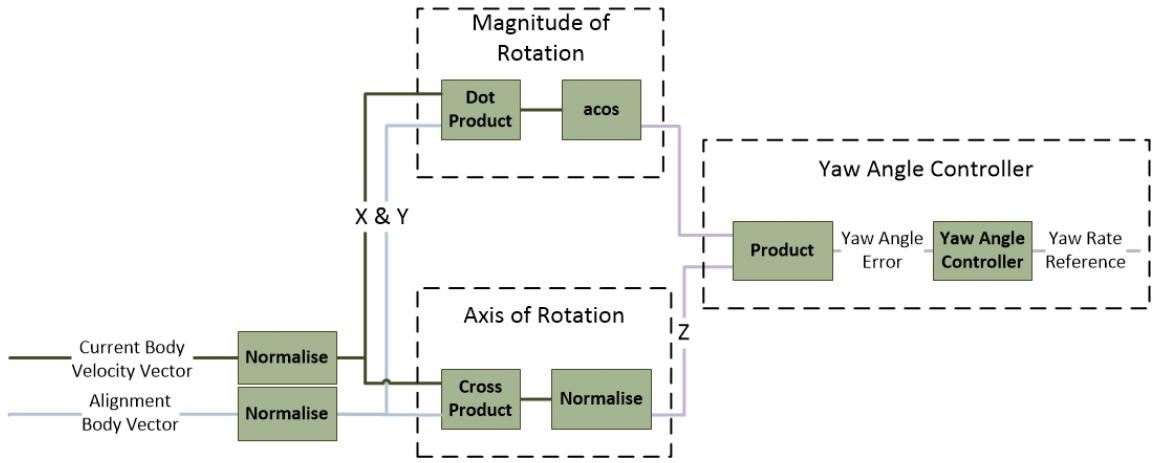


Figure 6.2: Yaw Alignment Controller

6.1.3 Yaw Alignment Discussion

At low speeds, while approaching position references, the proposed yaw alignment method can produce widely varying results. For this reason a minimum velocity magnitude is set, where below that limit the current yaw angle will be maintained. To demonstrate the effectiveness of the heading alignment, the drone is commanded to fly in a circle by commanding a North sine and East cosine velocity reference. Figure 6.3 is the output from that experiment. The coloured line represents the current heading of the craft. As seen, the craft begins facing due North and as the craft starts to fly it's path, the craft follows and aligns itself accordingly.

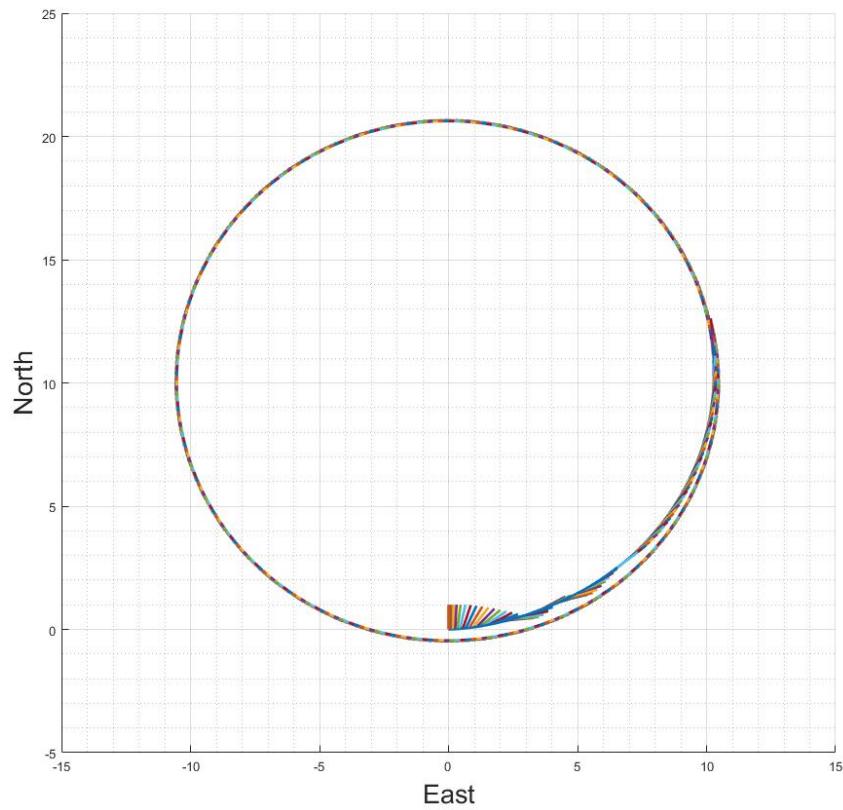


Figure 6.3: Yaw Alignment Controller

Figure 6.4 shows the calculated yaw angle error. As seen there is some initial overshoot, but the craft quickly settles onto the direct heading of the craft and maintains a close to zero yaw angle error.

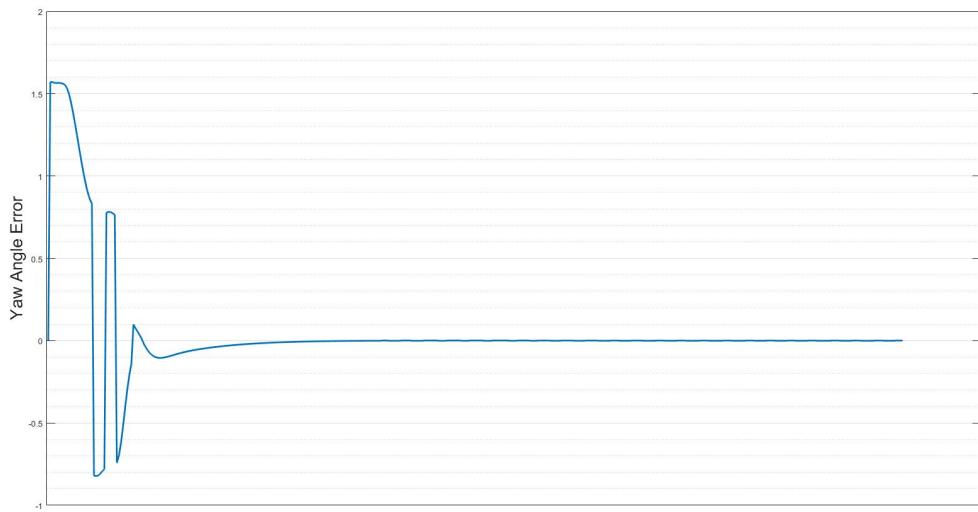


Figure 6.4: Yaw Angle Error

6.2 Obstacle Avoidance

The craft is required to navigate in an unknown environment without collision. The obstacle avoidance routine is responsible for ensuring the craft maintains a minimum set distance away from walls and other obstructions. When not in close proximity to any obstructions the obstacle avoidance routine should have a negligible effect on the craft which increases as the craft approaches a collision condition. Limited flight time dictates that the obstacle avoidance method should minimise deviation off of the desired path and allow the craft to reach the waypoints set by the waypoint generator when possible. The chosen obstacle avoidance controller should not affect the stability of the craft.

The method will be based on proximity measurements generated by sensors on board the craft. To test and validate the implementation of the obstacle avoidance controller a mathematical model of the sensor feedback needs to be created.

6.2.1 Mathematical modelling

Typical sensors used in this kind of application would be either ultrasonic proximity sensors, or in a high end application, a 360° laser range finder. This work assumes an ultrasonic sensor configuration and is modelled accordingly.

The proximity sensor placement is of critical importance to the functioning of the obstacle avoidance routine. The steps between proximity measurements are blind spots and can lead to the sensors missing an obstacle and subsequent crashing of the robot. Figure 6.5 shows the horizontal placement of sensors used in this work. There are eight sensors, all placed at 45° from each other. The vertical sensors used are placed in a simple up, down formation.

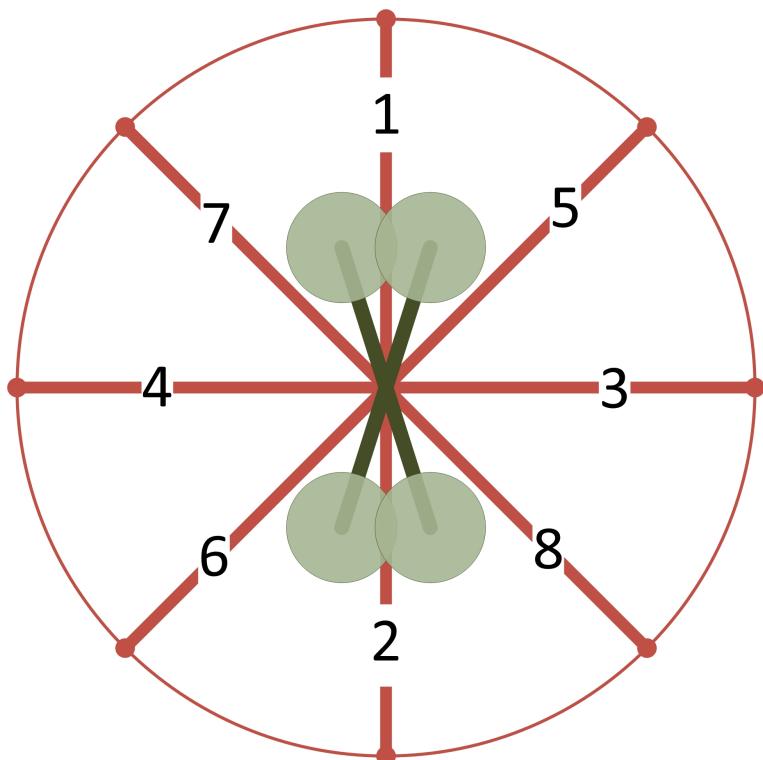


Figure 6.5: Sensor Placement for Obstacle Avoidance

The measurements from ultrasonic sensors are subject to low amplitude, high frequency noise. Appropriate low pass digital filtering can be applied to the sensor by the data collection algorithm. There is still expected variations and errors in the sensor measurement due to uneven surfaces and large angles of deflection, corrupting the data from the sensors.

Both the sensor arms and the environment are set up as a combination of straight line segments. The environment can be modelled as a complex combination of multiple line segments as shown by the outer blue lines in Figure 6.6. For simplification, each vertical wall (walls existing in the North and East plane) is considered equal for all values of Down, and each horizontal wall (the roof and floor) is considered equal for each value of North and East.

Obstacle Distance Detection

To calculate the distance measured by each sensor the model must successfully calculate the intersection of each sensor line segment, with each environment line segment. The vertical and the horizontal vectors are split up, allowing for the separate two dimensional calculation of horizontal intersections and the one dimensional calculation of vertical intersections.

The two dimensional calculation is done using a well documented line segment intersection equation. A line segment is considered as a finite portion along a line, dictated by two points existing on that line. The equations of the line segments can be represented mathematically as shown in (6.1) - (6.2). p_1 and p_2 are the end and start points of the first line segment with p_3 and p_4 belonging to the second line segment. The equations of each line segment hold true for the case of $0 \leq t \leq 1$ expressing a position along the defined segment.

$$p_1 = t(p_2 - p_1) \quad (6.1)$$

$$p_3 = t(p_4 - p_3) \quad (6.2)$$

The definition of an intersection is where these two lines are equal, breaking the equation into x and y components then creates equations (6.3) - (6.4). Where t_a and t_b are the offsets for each respective line segment.

$$x_1 + t_a(x_2 - x_1) = x_3 + t_b(x_4 - x_3) \quad (6.3)$$

$$y_1 + t_a(y_2 - y_1) = y_3 + t_b(y_4 - y_3) \quad (6.4)$$

The final step is to solve for both t_a and t_b . If both t_a and t_b exist between 0 and 1, then it is known that the line segments intersect. If either t_a or t_b exist outside of this range then the general lines intersect outside of the described segment constraints. If the lines are collinear, no intersection of the general lines, the result for t_a and t_b will be unsolvable. There might exist a possibility where the sensor line segments intersect with multiple walls, in this case the sensor would only return the closest obstacle. As mentioned, distance measurement devices struggle to return accurate results when the angle between the object and the sensor is very large. In this case the intersection is not recorded as to mimic a realistic sensor. In the case that no intersection is measured, the sensor would return its maximum range.

Figure 6.6 shows the typical result of a single obstacle detection frame. The red dots show the calculated intersection between the sensor arms and the environment. The corresponding distances are summed into a single proximity vector with the negative of that vector representing an obstacle avoidance vector shown in cyan.

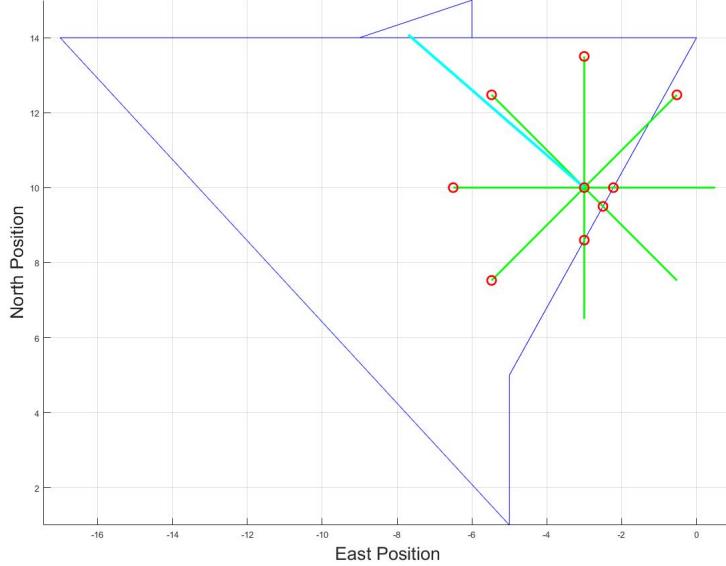


Figure 6.6: Obstacle Detection Demonstration

6.2.2 Obstacle Avoidance Methodology

The requirements for the obstacle avoidance routine are discussed above while different methodologies for collision avoidance were investigated in Section 2.6.2. This section begins by discussing these methods and how the conclusions for the chosen method were made based on the requirements for this work. The section is concluded by explaining the final implementation and it's controller.

Methodology Investigation

Three methods were explored to make an informed decision on the final implementation of the obstacle avoidance method.

The Bug algorithm will create a new path for the craft to follow based on local scans of an environment. This deviation off the set path could be useful when the craft is required to perform exploratory missions inside unknown environments, allowing the vehicle to traverse a walls contour until it finds the appropriate path again. However, the contours and new paths can lead the craft to deviate far off the desired path.

The potential field method utilises both the attractive potential of the target and the repelling forces of obstacles. The use of immediate distances from the craft limits the requirement for constant scanning and replanning of the route. This method will allow the craft to deviate off a given path while still maintaining an attractive potential to the end target. The position controller creates an attractive potential by commanding velocities proportional to distance error from a target. The obstacle avoidance routine will then need to apply it's repelling potential as a velocity command as well. The potential

field method also allows for relationships other than linear with the distance from the craft. The repelling potential could be generated from a quadratic, or even higher order polynomial, relationship with distance.

The last considered method is to utilise the benefits of a spring damper system. Using the maximum sensor distance as the resting distance, a virtual spring and damper can be created as depicted in Figure 6.7.

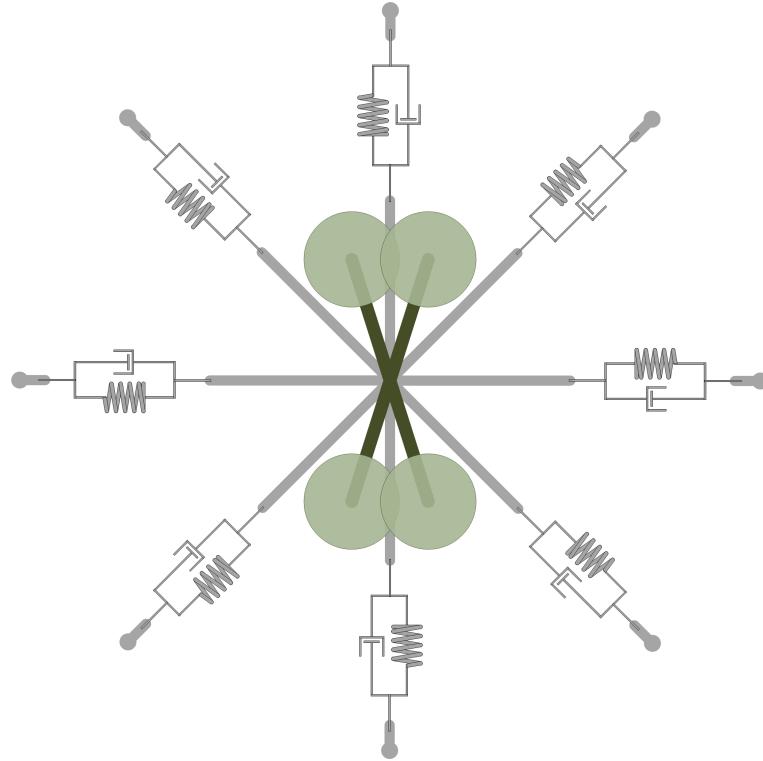


Figure 6.7: Visual descriptive aid for a virtual spring damper sensor system

The spring force can be used to push the drone away, but should not pull the vehicle towards an obstacle. The forces created by the spring should be added directly to the acceleration commands creating a virtual force. The damper is added to reduce the oscillations caused by the spring as it comes in and out of contact with the wall. The location of the spring damper in the inner control loops ensures a fast acting system, but requires a high bandwidth. The horizontal velocity controller's lag compensators also adds additional dynamics to the system that the obstacle avoidance system will need to interact with.

Each method has benefits and potential hindrances. The next section discusses combining these methods to form a successful obstacle avoidance implementation.

Methodology Implementation

This section describes the methodology behind the chosen obstacle avoidance routine. The system works on the basis of the potential field method investigated in Section 2.6.2. Figure 6.8 demonstrates how the controller feeds into the existing overall controller structure. As shown, the obstacle avoidance controller subtracts from the position controller's velocity set point to create a new velocity set point. Both set points are limited before being subtracted from one another. To ensure that the obstacle avoidance takes control

to avoid collisions, the position controller is limited to a smaller value than that of the obstacle avoidance controller.

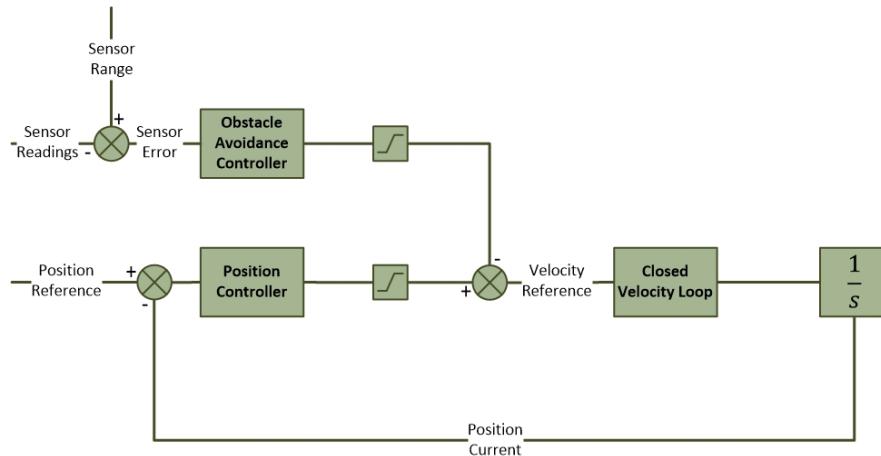


Figure 6.8: High level view of obstacle avoidance controller

The sensor values are all used and summed together to create an obstacle avoidance vector. This provides abstraction to the actual sensor placement used in practise. As long as the angle of each sensor is known a single vector can be created and used in the obstacle avoidance controller. To mimic the existing position controller structure a North, East and Down velocity command are created.

Obstacle Avoidance Controller

As shown in Figure 6.8, the obstacle avoidance controller generates a velocity setpoint reference and works in conjunction with the existing position controller. A 3-axis stable position controller has been designed in the controller chapter. The allowed gain and bandwidth of the obstacle avoidance controller can be inferred from the gain calculated for the existing controller. The sensor error is calculated by subtracting the current sensor measurement from the maximum sensor range and squaring the result. This quadratic relationship causes the obstacle avoidance controller to have a large effect when close to obstacles and less of an effect when obstacles first come into measuring distance. To limit the oscillations when approaching a wall, a derivative controller component is added. This component is calculated by taking the velocity along each sensor arm and applying a gain to each calculated velocity. It must be ensured that the derivative controller component is only active when the specific sensor is in measurement range. Figure 6.9 shows the implementation of the controller.

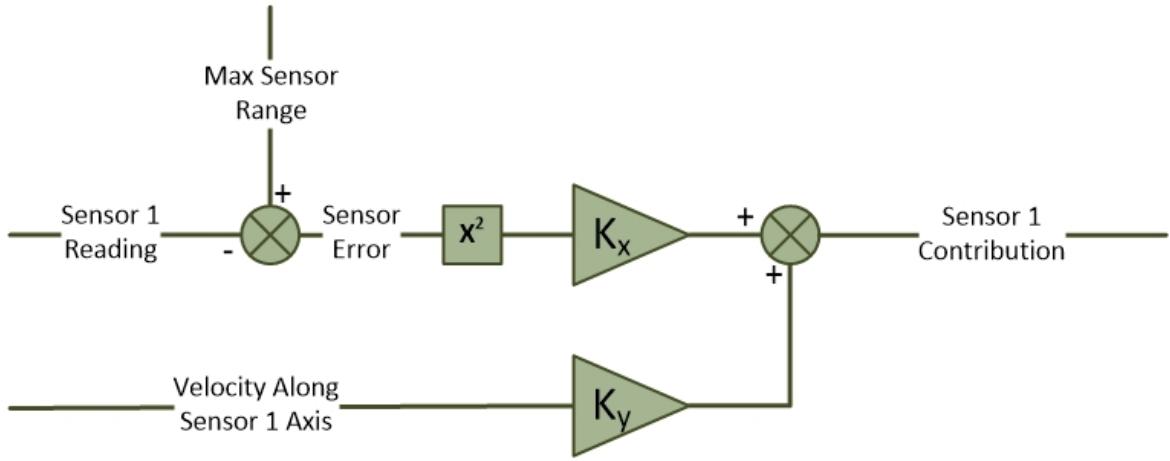


Figure 6.9: Individual Sensor Controller

The controller attempts to drive each of the combined contributions to zero. This condition is met when there are no obstacles, or when all opposing sensors measure the same result, maintaining an equilateral distance from each wall. Figure 6.10 shows how each sensor is combined. The sensor naming convention seen is specified in Figure 6.5.

A final consideration is the arm length of the sensor. The sensor maximum range should allow the craft to fly with none of the sensors active, but should also provide enough time for the system to respond. The gain of the controller should be considered at worst case, this occurs when the sensor is at a distance of close to zero, creating a sensor error equal to the square of the maximum sensor distance. To maintain stable system it is intuitive that the K_x gain of the sensor controller will decrease quadratically as the arm length increases. An arm length range of 2.5 m - 4 m was decided. Increasing the derivative gain increase the total damping in the system allowing for a higher K_x gain.

The selection of the saturations, arm length and gains will decide how close the craft will be allowed to a given obstacle.

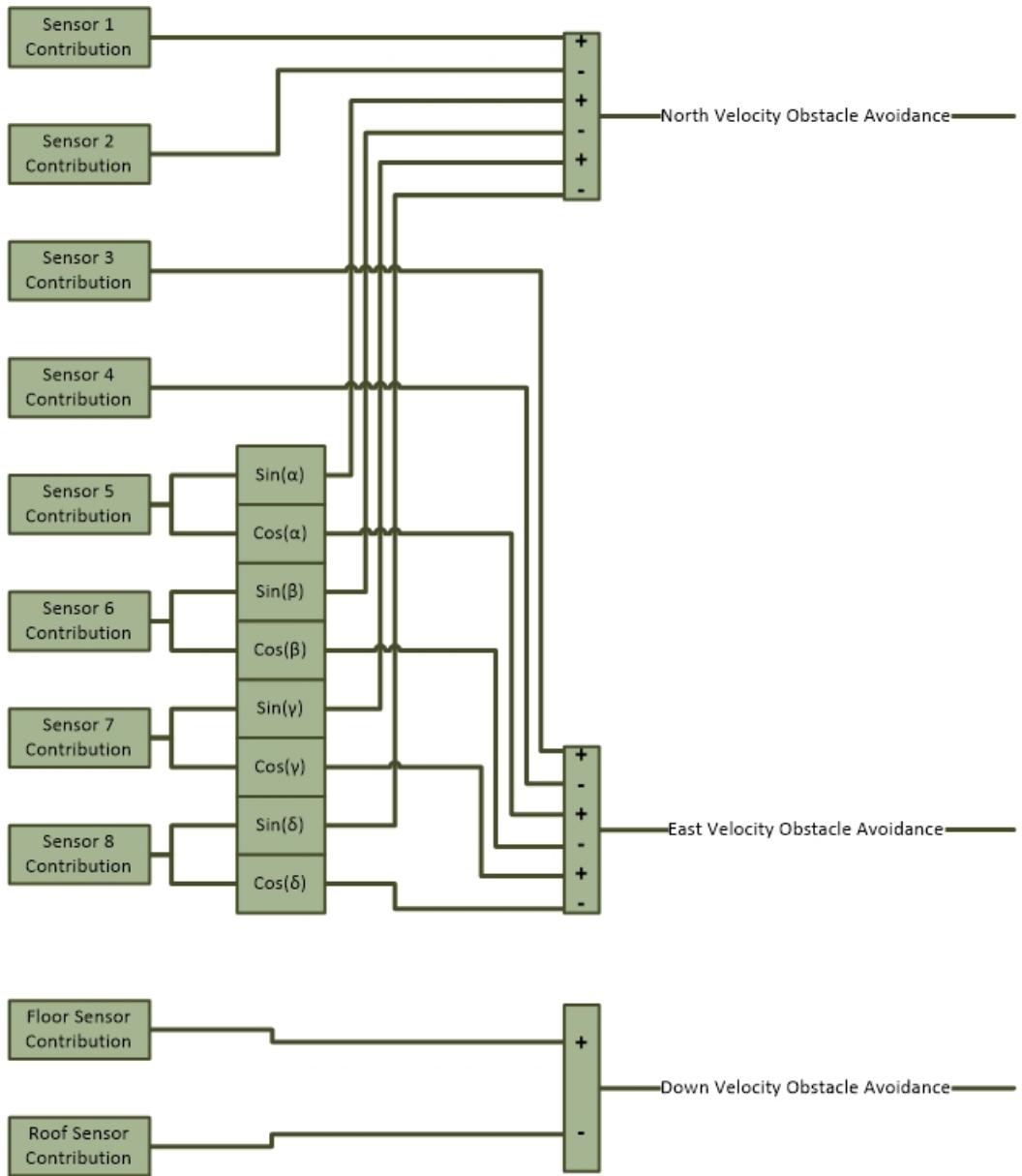


Figure 6.10: Sensor Combination

6.2.3 Obstacle Avoidance Discussion

The discussion is required to assess the feasibility and stability of the proposed obstacle avoidance routine. A simple method to assess this is by disabling the position controller and allowing the obstacle avoidance routine complete control of the aircraft's velocity reference commands. To create this scenario wall segments are placed at $-0.5m$ and $4m$ North and East, with a floor at $0m$ and roof at $-6m$ Down. The horizontal sensor range is set at $3.5m$ with the vertical sensor range set to $2m$. The craft is started at position $(0, 0, 0)$, in the bottom left corner of the four walled room.

The images shown in Figure 6.11 show the North, East and Down position respectively of the craft as it is controlled by the obstacle avoidance controller. The images on the right show the velocity set points sent by the controller to the inner velocity loop. As shown the craft is stable and moves steadily to the centre of the room with the velocity commands tending towards zero.

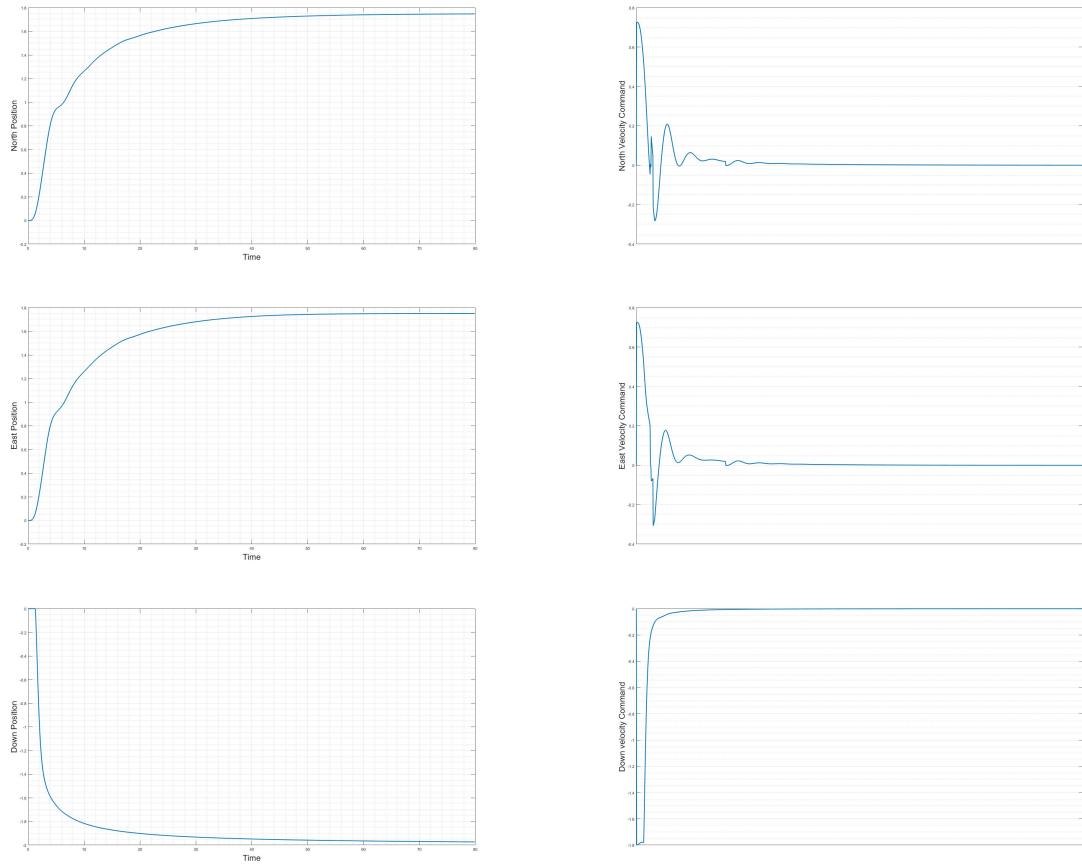


Figure 6.11: Position control using only the obstacle avoidance controller (Position (Left), Velocity Command (Right), North (Top) East (Middle), Down (Bottom))

To demonstrate the interaction of the obstacle avoidance routine with the position controller a wall is set up at 10m and the craft is commanded to go to 17m. This test will ensure that the obstacle avoidance controller works when the craft is travelling at maximum speed. Figure 6.12 shows the North position of the craft in blue, the commanded position in red and the dotted line represents the position of the wall. As seen, the craft stops within 0.5 m of the wall.

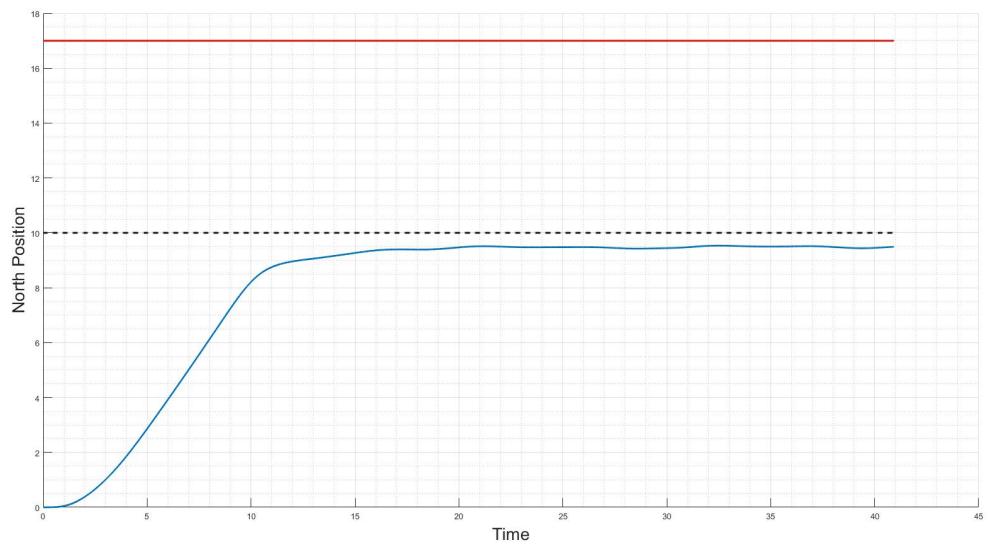


Figure 6.12: Sensor Combination

Although these examples show the method as an effective means of obstacle avoidance, this method has limitations that are explored in the proceeding flight tests chapter.

Chapter 7

Simulated Flight Tests

This chapter sets out to verify the performance of the proposed controller and flight strategies laid out in the previous sections of this report. The system identification chapter has outlined the data used as well as the configuration of the simulation. These parameters are used in conjunction with the controllers and flight strategy designed in previous chapters.

The first section of this chapter specifies what the tests aim to demonstrate and is proceeded by the methodology followed to do so. Each simulated test is designed to showcase an aspect of the design. The results from the simulation test runs are presented and finally the feasibility and performance of the craft are discussed.

7.1 title

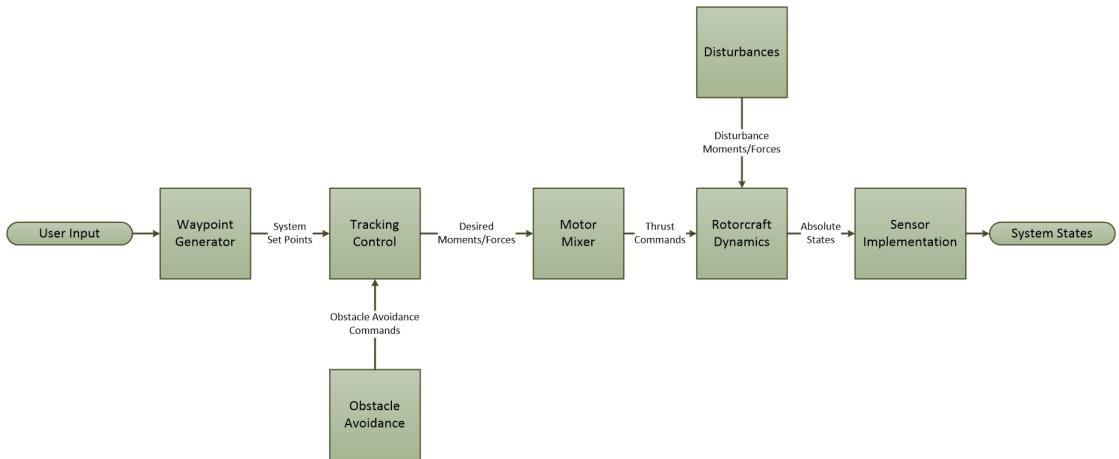


Figure 7.1: High level view of simulation set up.

7.2 Aims and Objectives

The objective of this work was to design an aerial vehicle that is capable of flight inside a confined space and narrow corridor. The first goal is to create a craft that is stable in the presence of system and sensor noise and impurities. The craft must be shown to perform this task in the presence of disturbances. Once the craft is shown to produce stable results in the presence of disturbances using the designed flight controllers, the

obstacle avoidance system needs to be tested. The first test must prove that in a simple environment with disturbances the system remains stable and ensures there is no collision. The next test must be to evaluate the drone navigating in a simple environment with the assistance of the obstacle avoidance controller. The difference with these tests will be to prove that the obstacle avoidance routine will navigate around obstacles and not simply just avoid them. After the drone can successfully navigate a simple environment, a more complex environment needs to be tested creating the needs for more complex manoeuvres. Finally the limitations in the proposed method need to be shown, an environment where the craft will not be able to complete its desired mission must be shown to understand where improvements and future work can be aimed.

7.3 Testing Methodology and Results

This section describes the method and results used to achieve the testing aims outlined above. A multitude of test scenarios are generated, all with a specific purpose to achieve one of the testing aims described above. The testing is structured to follow a logical pattern of validations.

7.3.1 Fully Integrated and Controlled System

The objective of the first test is to validate the designed controllers in the presence of a large disturbance. The disturbance is simulated as a 5 ± 1 m/s wind flowing at $45^\circ \pm 10^\circ$, measured off the North axis in a counter clockwise direction. Figures 7.2 and 7.3 are basic step responses with the above mentioned disturbance. The angle of the wind is in the same direction as the East position reference and the opposite direction to the North position reference. The green line in both images depicts the undisturbed system, with the blue line showing the disturbed system.

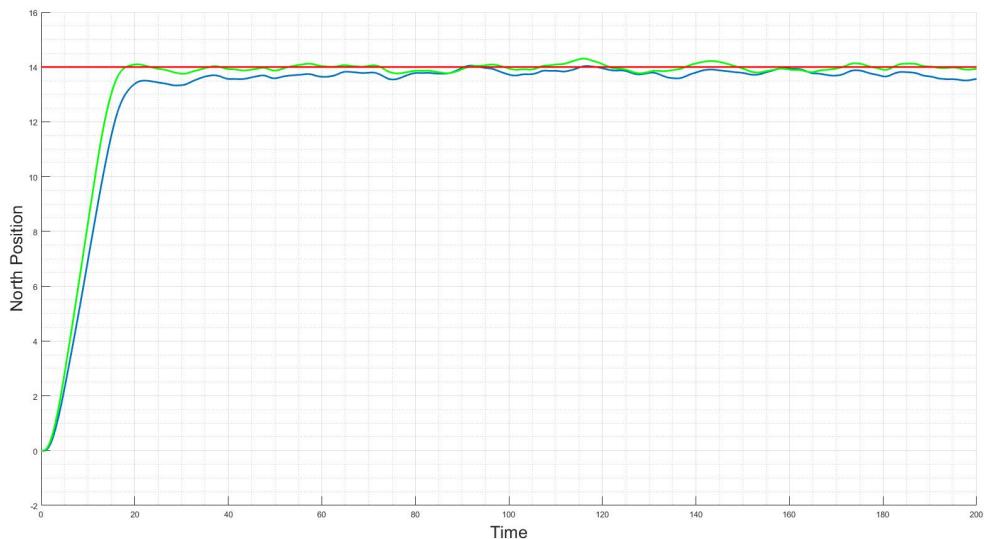


Figure 7.2: Step response with disturbance - north position plot

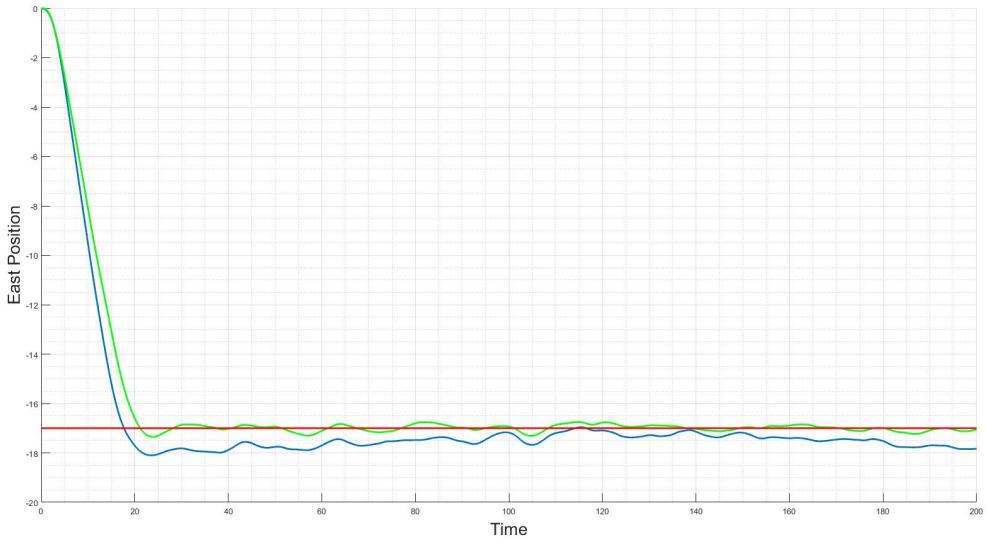


Figure 7.3: Step response with disturbance - east position plot

The wind causes the East controller to overshoot and sit around 1 m off the desired setpoint. The North controller is expected to fly into the wind and as expected exhibits a slower rise time to the non disturbed system.

Next, the waypoint generator is loaded with five waypoints, commanding the drone to first reach a set height and then fly in a rectangle ending at the start point. The large wind is angled to push the drone South and West and is present from the beginning of the simulation. As the craft reaches it's set height the craft is commanded to maintain in it's current North and East position. The North and East controllers will have a small position error until the wind forces the drone off of the set point. An initial offset is thus expected before the velocity controllers are given larger setpoint to follow. Figures 7.4 and 7.5 show the North and East positions respectively.

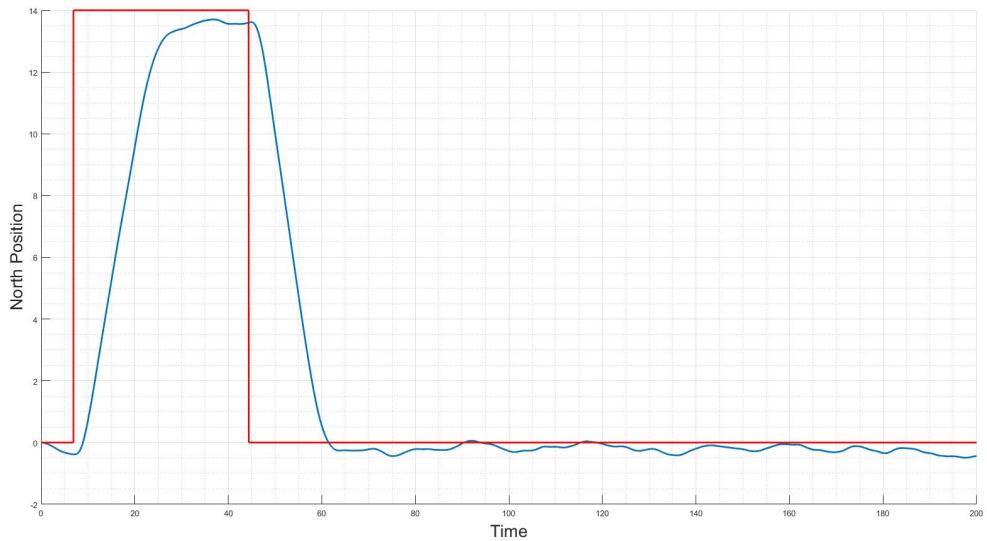


Figure 7.4: Waypoint flight with disturbance - north position plot

The North position is initially offset by the wind and is commanded to fly against the wind before returning South with the wind pushing it towards the setpoint. The North position reaches within 0.5 m in both cases. The seemingly linear portion of the curve entails that the disturbance is causing the drone to hit the upper saturation limit of the velocity command.

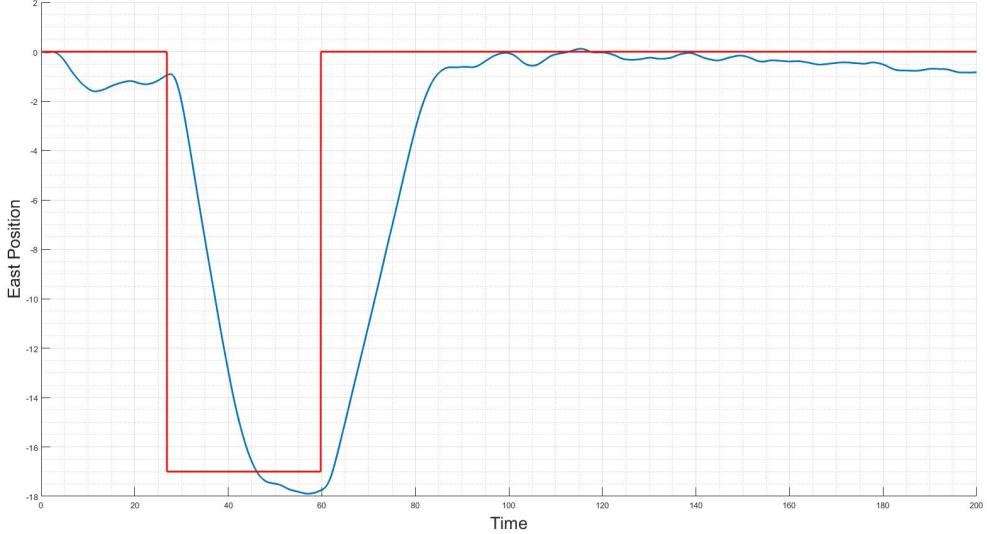


Figure 7.5: Waypoint flight with disturbance - east position plot

The East position has more difficulty handling the disturbance. The non symmetry of the craft causes a larger surface area in that plane, which leads to a larger drag caused by wind disturbances. The East position is initially pushed a maximum of 1.5 m off of its setpoint and settles to just less than a metre off the final desired position.

The above tests show the craft to be stable under simulated disturbances. The craft has more difficulty rejecting disturbances caused by wind in the Y-Body frame of the craft and can be reduced by angling the craft towards the wind during flight.

7.3.2 Basic Obstacle Avoidance Routine

The next test is designed to evaluate the effectiveness of the obstacle avoidance routine in the presence of a disturbance. The Y-Body Axis was seen to be less resilient towards disturbances and will be tested for this purpose. The test is designed to see if the craft can maintain a set distance away from a wall while the tracking system and disturbance push it towards the wall.

To accomplish a wall was placed along 1 m East and another wall at -3 m East as shown by the red lines in Figure 7.6. A constant 10 km/h wind was applied to the craft facing due East, pushing the craft into the wall. The Cyan Dot in the image represents the waypoint at (15, 0). There are two images in Figure 7.6, the image on the left shows the flown flight path, while the image on the right adds the obstacle avoidance vector generated by the obstacle avoidance controller.

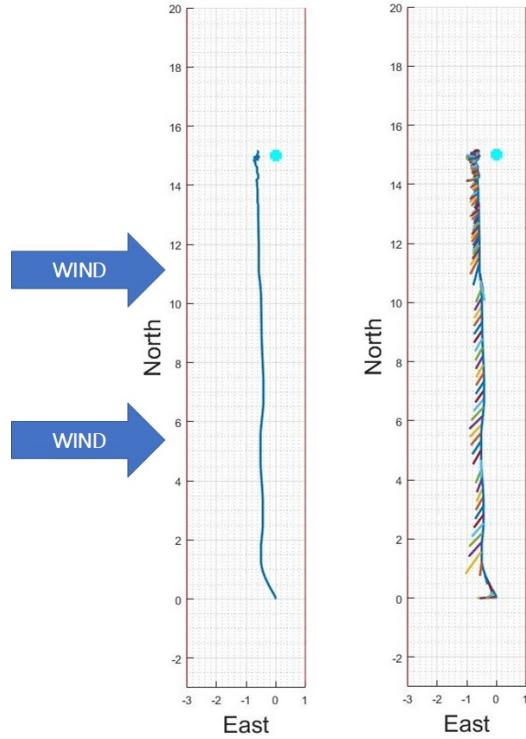


Figure 7.6: Corridor flight with disturbance - showing flight path (left) and obstacle avoidance vector (right)

The drone is commanded to position 0 East and the wind is attempting to force the Drone directly East as well. The obstacle avoidance vector successfully steers the craft away from the East wall and maintains an average distance of 0.62 m off of the wall with a standard deviation of 0.09 m. The obstacle avoidance vector shown in the coloured line of the right image shows the direction in which the drone is being pushed by the obstacle avoidance controller. The proximity to the East wall pushes the drone in a Westward direction, with a South facing component. The derivative portion of the controller creates the South facing vector as it moves in a Northerly direction.

Figures 7.7, 7.8 and 7.9 show the same flight with each position plotted against time. The green line shows the path flown in a scenario where the obstacle avoidance controller is deactivated. The red line is the set point with the blue line being the flown path with obstacle avoidance activated. The black dotted line in the East plot represent the wall at 1 m.

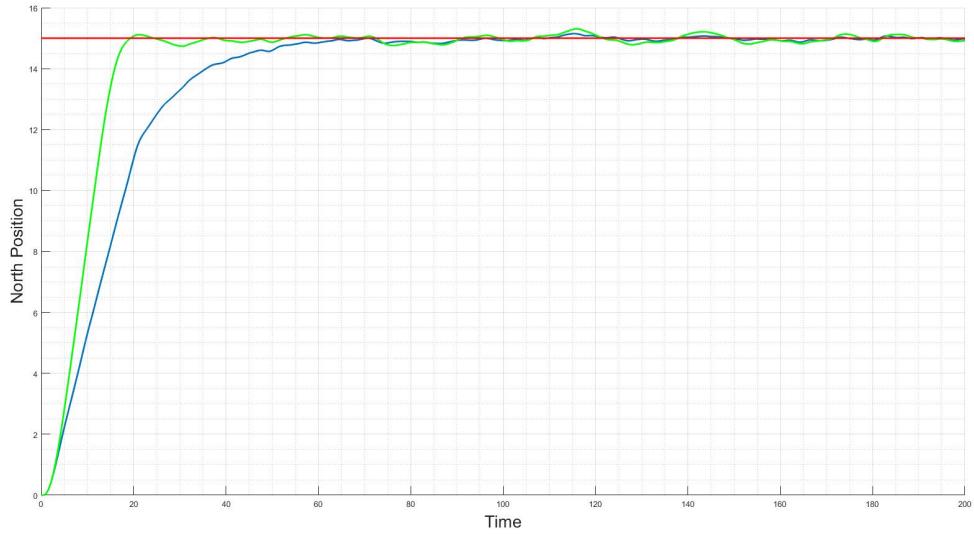


Figure 7.7: Corridor flight with disturbance - north position

The obstacle avoidance controller slows down the North settling time of the craft. The obstacle avoidance controller derivative component will increase damping in the system as seen by the reduced overshoot.

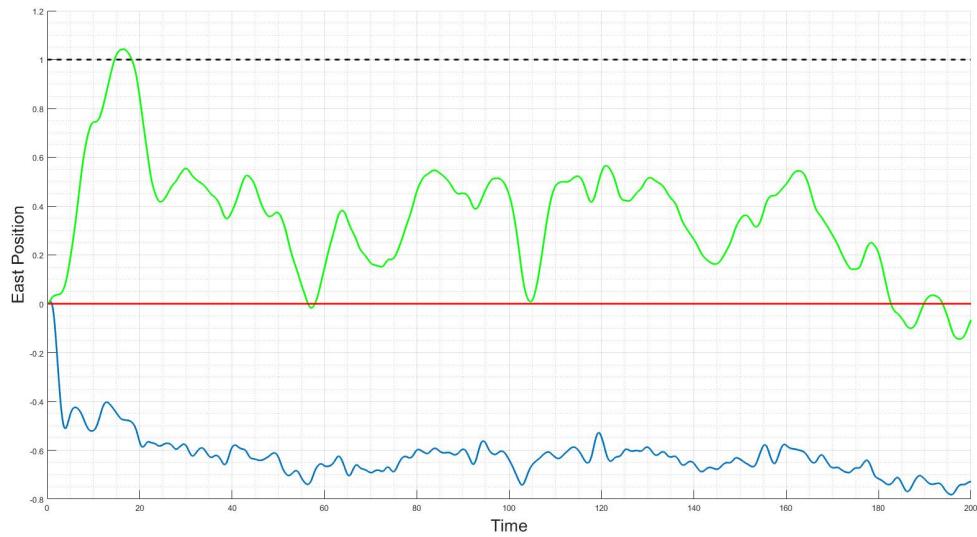


Figure 7.8: Corridor flight with disturbance - east position

As expected from the previous test, without the obstacle avoidance controller the drone will collide with the wall at 1 m. The obstacle avoidance controller pushes the craft off of it's setpoint to ensure it maintains a set distance form the wall. The increased damping in the obstacle avoidance controller also reduces the oscillations seen by the craft.

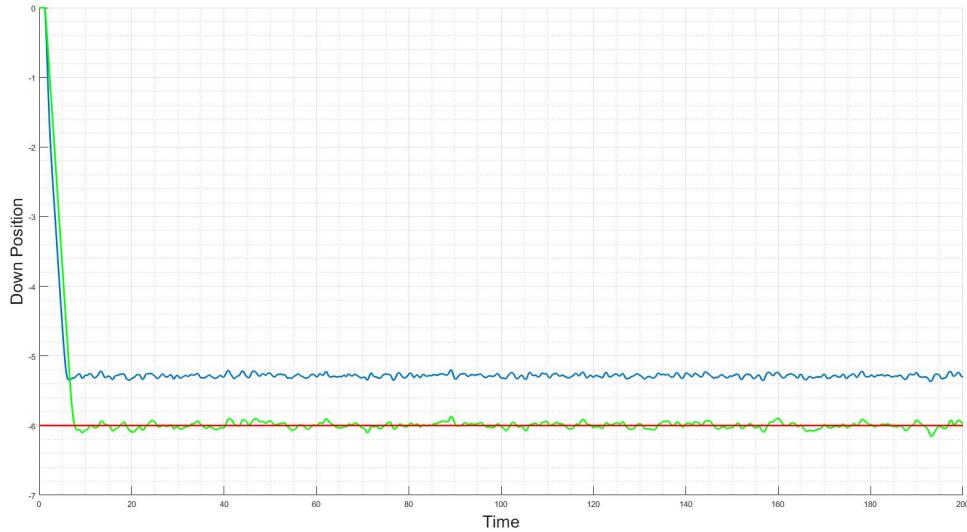


Figure 7.9: Corridor flight with disturbance - down position

The horizontal obstacle avoidance has been the main focus this far. In this scenario a roof was placed at 6 m with a floor at 0 m. The craft settles an average distance of 0.71 m from the roof with a standard deviation of 0.03 m.

The obstacle avoidance system has shown it's effectiveness by avoiding a collision in the presence of a disturbance. The routine successfully kept the drone away from the walls by maintaining a set distance. The routine also ensures that the craft flies slowly and steady when in the presence of obstacles.

7.3.3 Obstacle Avoidance Navigation

The next series of tests aims to test the capabilities of the obstacle avoidance routine and it's ability to navigate through a simple terrain. To test this different test scenarios have been configured. Each environment is in a sawtooth shape which traverses from South to North. The drone will be given only a North reference with the East position controller disabled. The obstacle avoidance routine will be required to ensure the drone does not collide with any of the walls, as well as find a path through the corridor to the desired North position. The test will be run three times. The first test will be in a wide corridor where the obstacle avoidance sensors will not always be activated. The second test will be a narrow corridor which will show how the craft behaves if it is constantly in close proximity to a wall. The third test will show the effect the yaw alignment algorithm, has on the system and vice versa.

Wide Corridor

The first test requires a large corridor to be used and are represented by the red lines in Figure 7.10. The vehicle is commanded to fly to a North position of 34 m as shown by the cyan line. The flown path is shown in blue with the obstacle avoidance vector shown in the coloured line. The East position control is completely dictated by the obstacle avoidance routine.

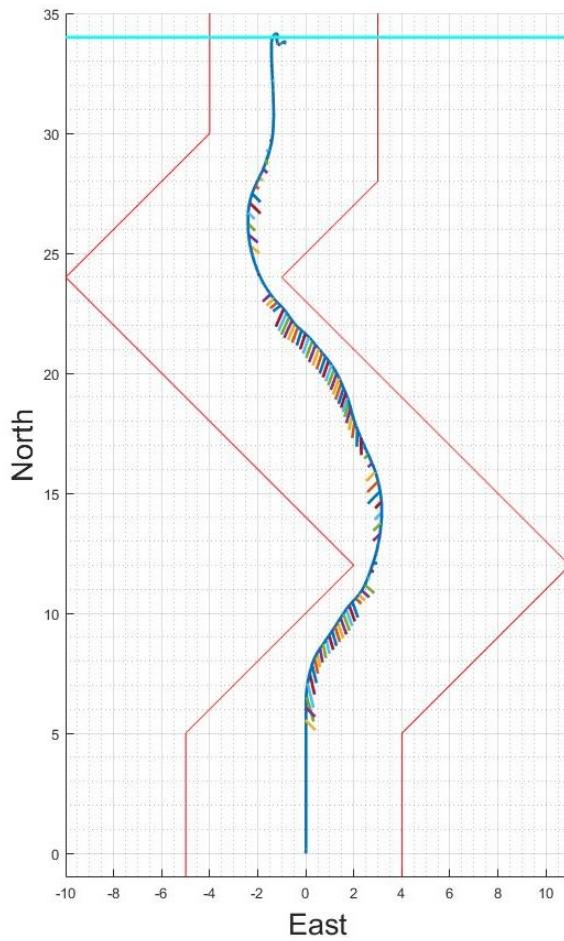


Figure 7.10: Navigated flight in a wide corridor

The craft maintains a minimum distance of 0.5 m away from the wall and flies along an efficient path to reach the goal, only deferring from the straight line path when necessary to avoid a wall. Although a successful flight, the image does show the importance of the density for sensor placement. The sharp corners created by the wall only activate one of the sensors which creates a small obstacle avoidance vector. Having a more dense sensor placement will nudge the drone further away from the dangerous corners.

Narrow Corridor

The next tests showcases the drone's ability to fly in a confined space where the obstacle avoidance vector is always activated from multiple sides. The narrow corridor is shown in Figure 7.11, with the drone requested to reach 17m North. Once again the East position of the craft is completely dictated by the obstacle avoidance vector. The coloured line also once again demonstrates the current obstacle avoidance vector which is commanding velocities.

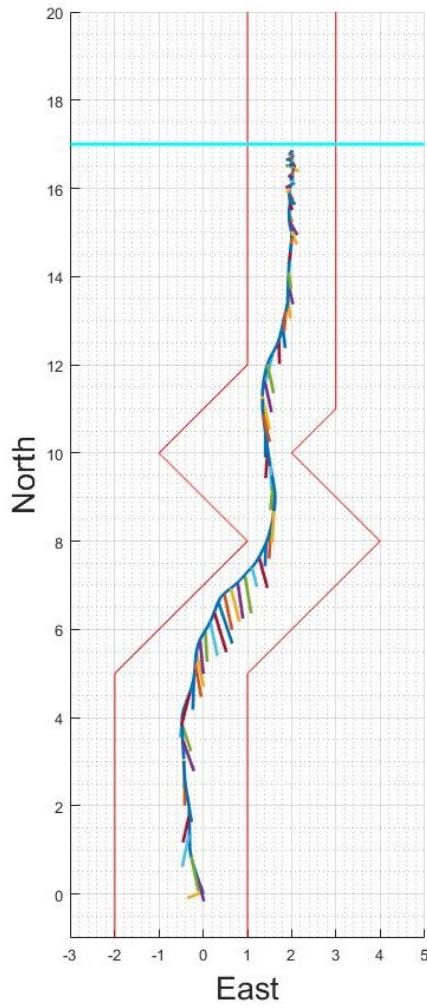


Figure 7.11: Navigated flight in a narrow corridor

The drone remains a minimum of 0.5 m away from the walls. This test demonstrates the importance of the derivative action of the obstacle avoidance vector, specifically in a confined space. Although all the sensors are active due to close proximity to multiple walls, the walls which the drone is approaching take precedence and the obstacle avoidance controller forces large action. The walls which the drone is not approaching have limited interference for the flight path.

Yaw Alignment in a Confined Space

The next test is designed to evaluate the operation of both the obstacle avoidance and the yaw alignment strategies when used in conjunction with one another. The wide corridor test is run again, this time with the yaw alignment routine activated. Figures 7.12, 7.13 and 7.14 represent the results from those flights. The first image shows only the flown path with the second image including the heading vector of the craft.

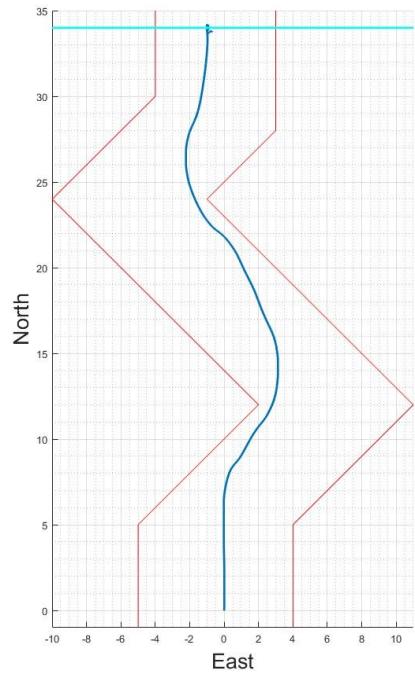


Figure 7.12: Navigated flight in wide corridor with yaw alignment

The path flown closely resembles that of the original path without yaw alignment. The craft does however get closer to the sharp corners of the wall, the yawing craft will affect the response of the obstacle avoidance controller due to the changing sensor readings. A tight sensor placement density will reduce the effect this has on the craft.

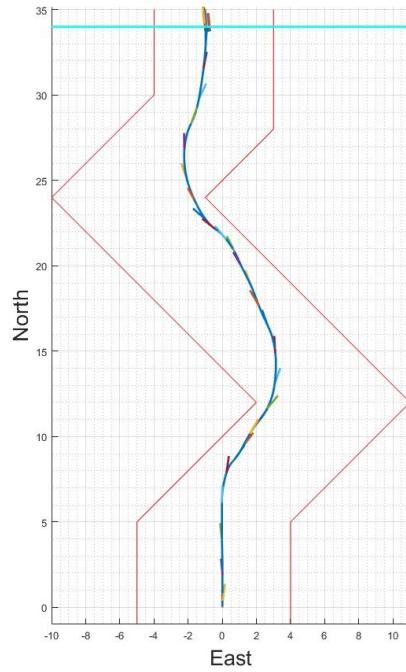


Figure 7.13: Navigated flight in wide corridor with yaw alignment showing current heading of craft

Figure 7.13 overlays the current heading of the craft on the flight path. The yaw alignment is shown to work reasonably well but can be quantified better using Figure 7.14 which shows the yaw error throughout the flight.

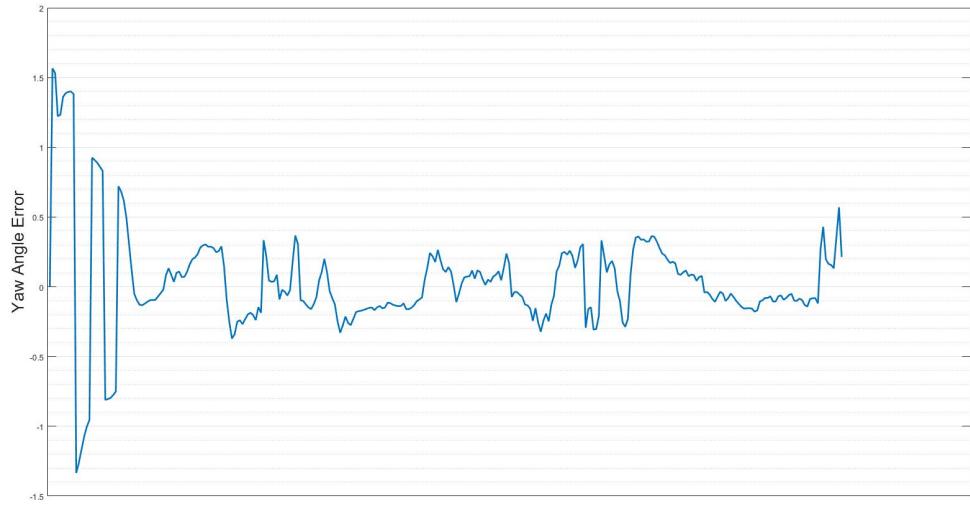


Figure 7.14: Yaw error of vehicle during navigated flight in wide corridor with yaw alignment

The yaw alignment routine runs at a slower pace to that of the velocity controller and hence has large error when there is a sudden change of direction. The average yaw

alignment error for the flight is only 1.97° with a large standard deviation of 21.53° . The constant changes of the craft's velocity leads to varying alignment reference which can cause to a large deviation in error measurements. In some mission cases this might be unacceptable and the craft's current velocity should not be used as the alignment vector, but rather a set heading which can be varied.

7.3.4 Generic Mission Flight

Now that each component of the craft's control system has been tested and verified a generic example of flight test can be created. This group of tests is run in a simulated environment that contains a combination of open spaces, narrow corridors and unexpected obstacles. The waypoint generator is loaded with 9 waypoints loosely placed around the environment. The waypoints are placed to designed to make the craft explore the entire environment with the need for avoiding obstacles and flying down corridors un assisted. The final waypoint is placed in an unavailable location to assess the craft's behaviour. The test is run both with and without the yaw alignment routine enabled.

The images seen in Figure 7.15 show both runs with and without their proximity vectors. The waypoints are shown in cyan, starting at $(0, 0)$ and moving in a counter clockwise direction. The last waypoint is placed inside the wall where the drone cannot fly.

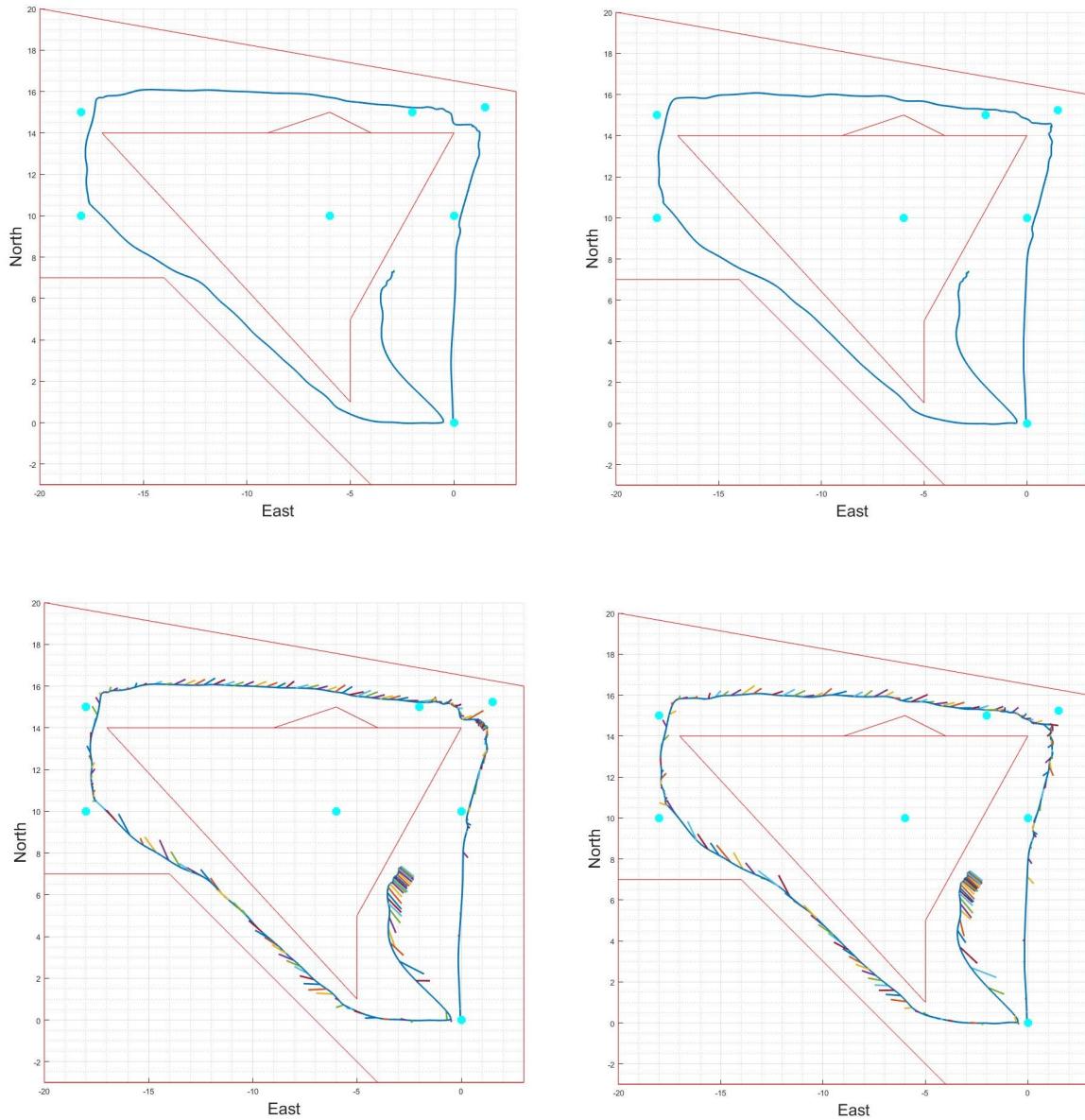


Figure 7.15: Generic mission flight in a simulated environment with loosely placed waypoints. (Left - no yaw alignment, Right - yaw alignment)

The craft does not hit any obstacles in both runs. A more detailed flight path or a more dense sensor placement would assist in ensuring the craft keeps a further proximity from the sharp corners.

A better analysis of the separate runs can be made while observing the North and East positions relative to time as shown in Figures 7.16 and 7.17. The green line shows the position without yaw alignment, while the blue line shows the position when yaw alignment is disabled. The average difference between the position along the North axis is 0.12 m with a standard deviation of 0.79 m. The East axis boasts a mean difference of only 3 cm with a standard deviation of 0.6 m.

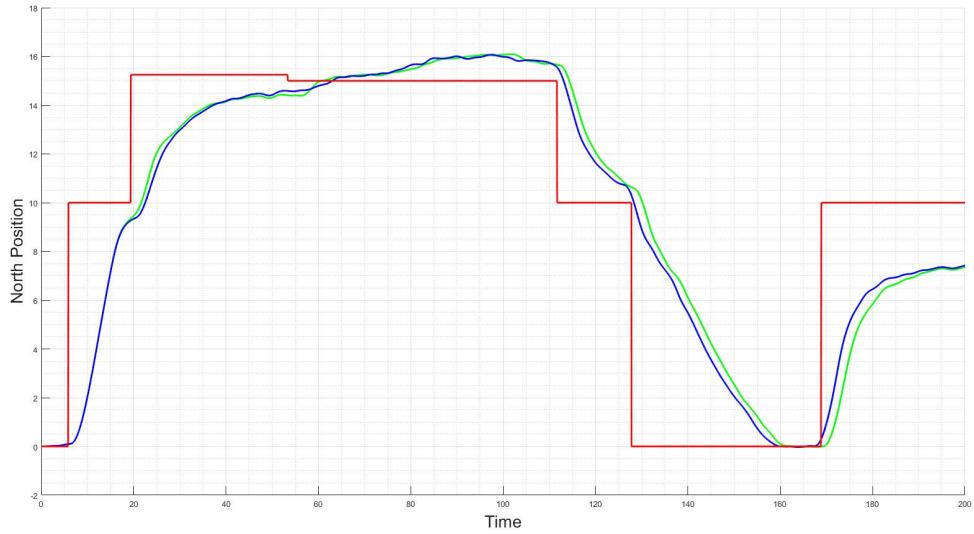


Figure 7.16: North position plot of a generic flight test. Showing the result both with and without yaw alignment.

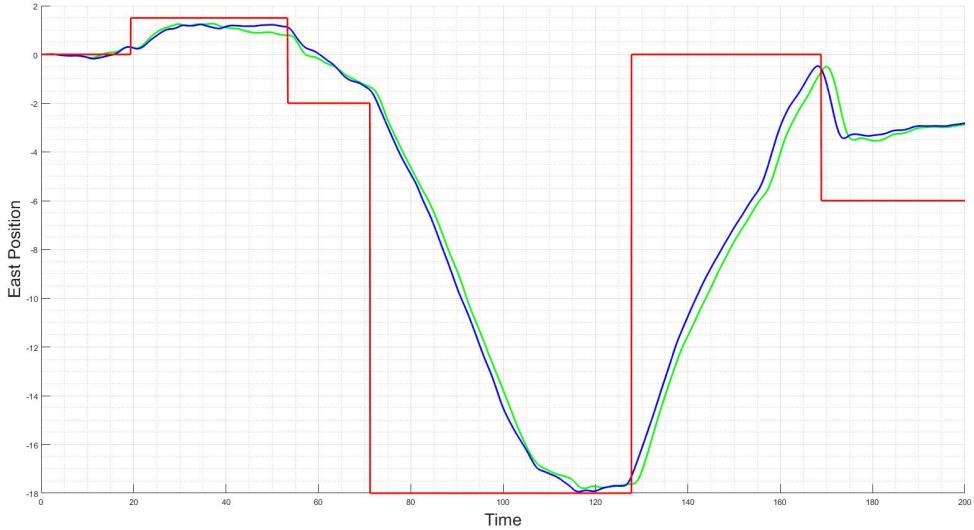


Figure 7.17: East position plot of a generic flight test. Showing the result both with and without yaw alignment.

To properly assess the yaw alignment strategy in this scenario Figures 7.18 and 7.19 are presented. Figure 7.18 overlays the current heading of the craft onto the flown flight path.

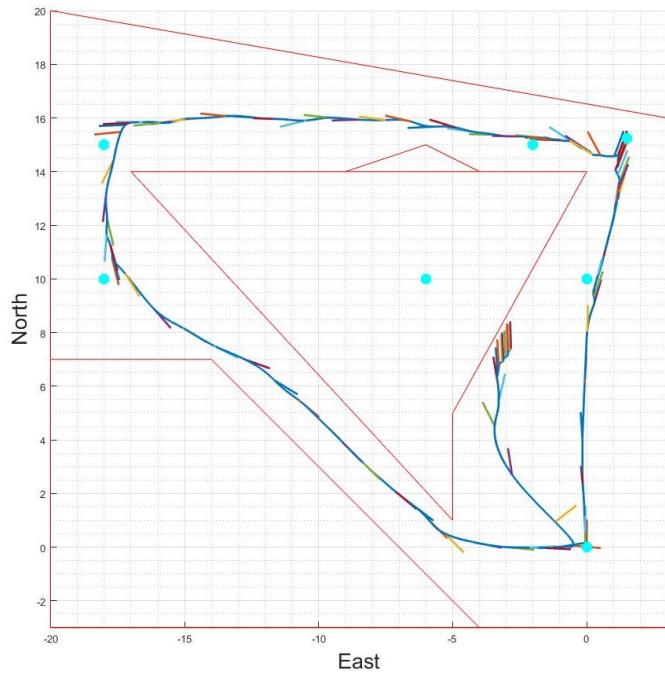


Figure 7.18: Yaw alignment plot of a generic flight test while utilising the heading alignment controller.

When the drone is constantly changing direction the yaw alignment has some difficulty maintaining the heading alignment. When there are longer stretches of straight path the yaw alignment has the time to reach the desired heading and maintain the yaw error within 3.56° with a standard deviation of 5.49° . Figure 7.19 gives a more empirical view of the yaw alignment by isolating the yaw error calculated.

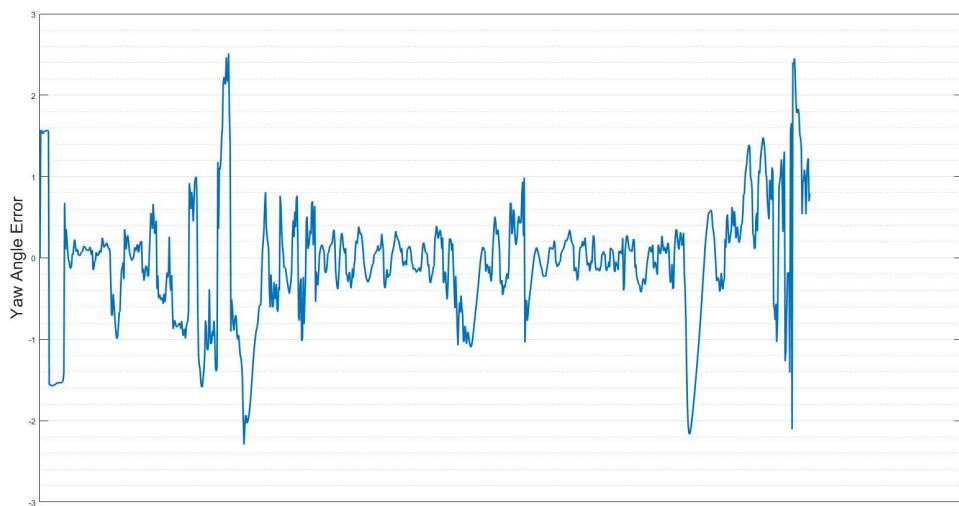


Figure 7.19: Yaw error plot of a generic flight test while utilising the heading alignment controller.

7.3.5 Limitations of the Design

The last set of tests are designed to show some of the limitations of the obstacle avoidance technique as a navigation algorithm. Situations exist where the obstacle avoidance routine will cause the drone to stall and not continue on it's mission. Two scenarios have been designed. The first scenario is a basic corner shown in Figure 7.20.

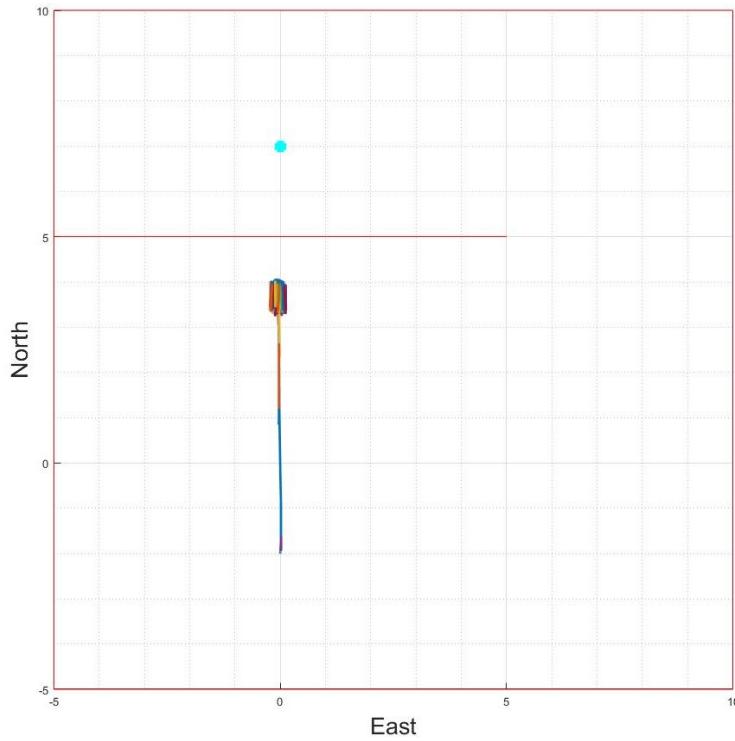


Figure 7.20: Limitations of the obstacle avoidance routine as a navigation algorithm. Straight wall in a wide space.

As the craft approaches the wall at 5 m North it gets forced to stop by the obstacle avoidance vector. There is no additional information being fed to the the system that will inform it to continue on it's path. In this situation the obstacle avoidance works appropriately to avoid a collision. Figure 7.21 shows a situation where the drone will not collide with a wall, but the drone obstacle avoidance routine halts the mission unnecessarily.

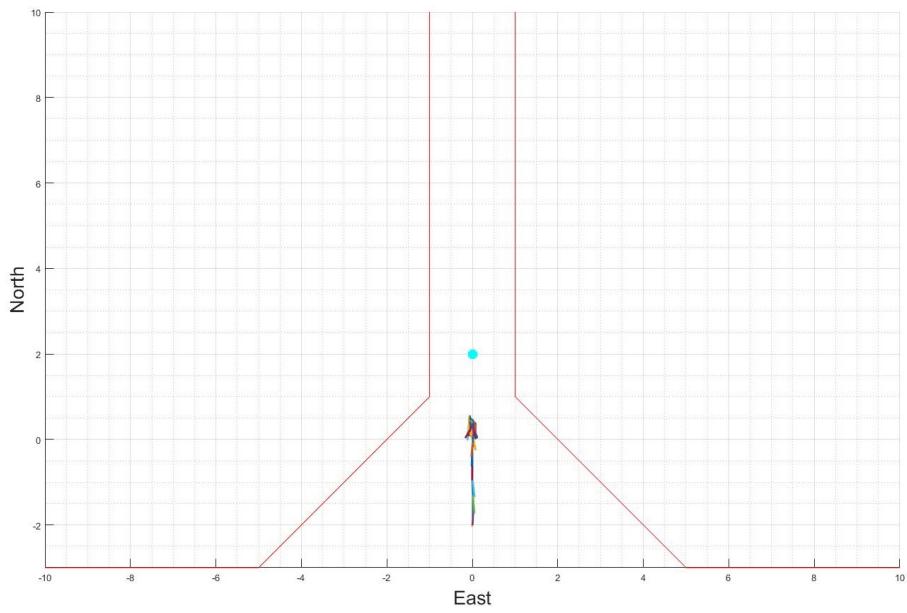


Figure 7.21: Limitations of the obstacle avoidance routine as a navigation algorithm. Narrow corridor proceeding a wide open space.

The narrow corridor is wide enough for the drone to fit in, but the two front facing angled sensors pick up a disturbance and halt the drone from completing it's mission. Both of these situations call for a higher route planning algorithm, or more intelligently placed waypoints.

Chapter 8

Conclusions and Recommendations

The final chapter concludes on the work done in this project. It begins by summarising the conclusions and discussions had during the course of this work and finalises by recommending future areas that should be focused on to achieve the goal flight inside a confined environment.

8.1 Summary and Conclusions

The thesis successfully designed and simulated a flight strategy capable of obstacle avoidance and basic navigation inside a confined environment. The strategy proposed was tested through simulation to maintain a distance of 0.5 m away from walls and obstacles while maintaining stable flight. The problem was solved by choosing a craft design to help fly in a narrow corridor and designing a set of controllers to maintain stable flight with an over arching flight strategy.

The craft design of the vehicle was accomplished through analysis of conventional rotor wing configurations and flight theory. The vehicle required high payload capabilities for additional sensor packs and larger power sources to increase flight time. This was accomplished by choosing a design that optimised thrust capabilities in a narrow space. The design varies from a traditional quadcopter by having a 20% overlap of the front and rear rotor sets.

A three tiered controller system was designed to control the proposed platform in six degrees of freedom. The three tiers were broken into an altitude, horizontal and heading controller. All three systems were shown to be capable of rejecting disturbances and providing stable control. The altitude control system controls the height of the craft by commanding a climb rate which in turn controls the acceleration of the craft in line with the body Z-Axis. The horizontal controller is responsible for controlling the North and East position and velocity of the craft. This was accomplished by relating the North and East accelerations to relative pitch and roll angles for the craft which in turn command the pitch and roll angular rates. The heading controller is responsible for controlling the yaw angle of the craft by commanding a yaw rate. Each controller fed their setpoints into a motor mixer which created the correct thrust outputs for each motor.

An investigation into existing collision avoidance techniques led to the successful generation of a proximity based obstacle avoidance routine. The method chosen requires a proximity measurement relative to the craft in the X, Y and Z-Body Axis and utilises the potential field method of obstacle avoidance. This allowed the craft to avoid obstacles by maintaining a set distance from obstructions in all three axes.

To enable autonomous flight a waypoint generator was created which enables the aircraft to automatically step between position set points. The environment and the design of craft required an additional flight strategy which aligns the heading of the craft with it's current direction. This ensured that the craft's longer axis is always in the direction of flight minimising drag and proximity to narrow corridors.

To validate the controller scheme and flight strategy, accurate mathematical modelling of the craft and disturbances was required. This was accomplished through a system identification process including real world measurements and collection of data from proposed sensors. The disturbances and the system were modelled using Matlab and Simulink.

The simulation showed that the proposed flight strategy and controllers could be used for navigation in a confined environment. The platform was designed to ensure sufficient thrust capabilities for a larger power source and additional sensor payload making it suitable for expansion into industrial applications.

8.2 Recommendations

The following recommendations are proposed to improve the viability of the system as an autonomous platform and expand the work to create a real world implementation.

- The proposed obstacle avoidance routine has proven to be capable of providing navigation in some environments. The limitations of the design can be assisted by a higher level route planning algorithm which utilises sensor information to create more autonomy for missions.
- This work used simulation to prove the effectiveness of the system. To finalise the validation of the proposed system real world flight tests should be conducted using the proposed platform construction. It is recommended that prior to any implementation of the flight strategy or obstacle avoidance the mechanical construction is verified to be robust to ensure a good flight set up limiting risk during flight testing.
- Additional flight modes should be created to allow the pilot control of the more inner loops. This work creates a waypoint generator that feeds velocity commands. Situations exist, specifically during initial testing, that require the pilot is granted control of the craft's flight routine.
- Research has been done and reviewed to show the effects and subsequent disturbances of flight near walls. A more detailed measurement of these disturbances for the proposed craft should be done to ensure the system is capable of rejecting them substantially.
- A robust state estimator would reduce noise and error on the measurements and allow for the implementation of a disturbance observer based control algorithm. Such an algorithm could assist with successful rejection of larger disturbances while limiting the effect on the tracking control.

Bibliography

- [1] F.A. Association. *Rotorcraft Flying Handbook*. FAA Handbooks Series. Aviation Supplies & Academics, Incorporated, 2001.
- [2] Andrew J. Barry, Helen Oleynikova, Dominik Honegger, Marc Pollefeyt, and Russ Tedrake. Fast Onboard Stereo Vision for UAVs. *IROS Workshop*, page 7, 2015.
- [3] John H Blakelock. *Automatic Control of Aircraft and Missles*. 1991.
- [4] Felipe Bohorquez, Paul Samuel, Jayant Sirohi, Darryll Pines, Lael Rudd, and Ron Perel. Design, Analysis and Hover Performance of a Rotary Wing Micro Air Vehicle. *Journal of the American Helicopter Society*, 48(2):80, 2003.
- [5] Adrien Briod, Dario Floreano, Przemyslaw Kornatowski, and Jean-Chirstophe Zufery. A Collision-resilient Flying Robot. *Journal of Field Robotics*, 31(4):496–509, 2014.
- [6] Adrien Briod, Adam Klaptocz, Jean-christophe Zufferey, and Dario Floreano. The AirBurr : A Flying Robot That Can Exploit Collisions. pages 569–574, 2012.
- [7] Yihua Cao, Dong Li, Qiang Zhang, and Hang Bian. Recent Development of Rotorcraft Configuration. *Recent Patents on Engineering*, 1(1):49–70, 2007.
- [8] Lozano R Dzul AE Castillo, P. *Modelling and control of mini-flying machines*.
- [9] Pedro Castillo, Alejandro Dzul, and Rogelio Lozano. Real-time stabilization and tracking of a four-rotor mini rotorcraft. *IEEE Transactions on Control Systems Technology*, 12(4):510–516, 2004.
- [10] Federal Aviation Administration. Helicopter Components , Sections , and Systems. In *Helicopter Instructor’s Handbook*, chapter Chapter 5, page 183. US Department of Transportation, Oklahoma, 2012.
- [11] Kenneth Warren Flanigan. Gas Powered Tip-Jet-Driven Tilt-Rotor Compound VTOL Aircraft, 2006.
- [12] Jeremy Green, A. Marnewick, and J. H.C. Pretorius. Prototyping during the requirements elicitation process in the development of an underground unmanned aerial system. *Proceedings of the 2015 Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference, PRASA-RobMech 2015*, pages 60–65, 2015.
- [13] K Großekatthöfer and Z Yoon. Introduction into quaternions for spacecraft attitude representation. *TU Berlin*, pages 1–16, 2012.

- [14] Lee N Hager. Drive System for a Variable Diameter Tilt Rotor, 2000.
- [15] Gabriel M Hoffmann, Claire J Tomlin, Haomiao Huang, and Steven L Waslander. Quadrotor Helicopter Flight Dynamics and Control : Theory and Experiment . *The American Institute of Aeronautics and Astronautics*, (August):1–20, 2007.
- [16] Jonathan P How, Thesis Supervisor, and Eytan H Modiano. Design and Control of an Autonomous Variable-Pitch Quadrotor Helicopter. 2012.
- [17] Wayne Johnson. *Helicopter Theory*. Courier Corporation, Stanford, 1994.
- [18] Adam Klaptocz, Adrien Briod, Ludovic Daler, Jean-christophe Zufferey, and Dario Floreano. Euler Spring Collision Protection for Flying Robots. pages 1886–1892, 2013.
- [19] Adam Klaptocz, Adrien Briod, Jean-christophe Zufferey, and Dario Floreano. An Indoor Flying Platform with Collision Robustness and Self-Recovery. pages 3349–3354, 2010.
- [20] Daewon Lee, Asad Awan, Suseong Kim, and H. Jin Kim. Adaptive Control for a VTOL UAV Operating Near a Wall, 2012.
- [21] J. Gordon Leishman. *Principles of Helicopter Aerodynamics*. Cambridge Aerospace Series. Cambridge University Press, 2nd edition edition, 2006.
- [22] Andrew H Logan and Richard E Moore. Helicopter Antitorque System Using Circulation Control, 1980.
- [23] Rogelio Lozano Claude Luis Rodolfo Garca Carrillo, Alejandro Enrique Dzul Lpez. *Modeling the Quad-Rotor Mini-Rotorcraft*. 2013.
- [24] Vladimir Lumelsky and Tim Skewis. Incorporating Range Sensing in the Robot Navigation Function. *IEEE Transactions on Systems, Man and Cybernetics*, 20(5):1058–1069, 1990.
- [25] Teppo Luukkonen. Modelling and Control of Quadcopter. *Journal of the American Society for Mass Spectrometry*, 22(7):1134–45, 2011.
- [26] Malloy Aeronautics. Drone 3 MA Hoverbike.
- [27] Daniel Mellinger, Nathan Michael, and Vijay Kumar. Trajectory generation and control for precise aggressive maneuvers with quadrotors. *Springer Tracts in Advanced Robotics*, 79:361–373, 2014.
- [28] Paul Möller. *Automated Landing of a Quadrotor Unmanned Aerial Vehicle on a Translating Platform*. PhD thesis, Stellenbosch, 2015.
- [29] Yogianandh Naidoo, R Stopforth, and Glen Bright. Rotor Aerodynamic Analysis of a Quadrotor for Thrust Critical Applications. Technical Report November, University of KwaZulu Natal, Durban, 2011.
- [30] NASA. Reynold's number.

- [31] Paul Y Oh, Michael Joyce, and Justin Gallagher. Designing an Aerial Robot for Hover-and-Stare Surveillance. *IEEE Advanced Robotics*, (12):303–308, 2005.
- [32] Paul Pounds, Robert Mahony, and Peter Corke. Modelling and Control of a Quad-Rotor Robot. *East*, 4:44, 2006.
- [33] Marc Raibert, Kevin Blankespoor, Gabriel Nelson, and Rob Playter. BigDog , the Rough-Terrain Quadruped Robot. *The international federation of automatic control*, 17:6–9, 2008.
- [34] Maria Isabel Ribeiro. Obstacle avoidance. *Instituto de Sistemas e Robótica, Instituto Superior Técnico*, page 1, 2005.
- [35] David Conal Robinson. *Modelling and Estimation of Aerodynamic Disturbances Acting on a Hovering Micro Helicopter in Close Proximity to Planar Surfaces*. PhD thesis, Monash, 2016.
- [36] D.C. Robinson, Hoam Chung, and K. Ryan. Computational investigation of micro rotorcraft near-wall hovering aerodynamics. In *Unmanned Aircraft Systems (ICUAS), 2014 International Conference on*, pages 1055–1063, May 2014.
- [37] Wicks C Wilson R Rorrer G Welty, J. *Fundamentals of Momentum, Heat and Mass Transfer*.
- [38] La Young, Ew Aiken, Jl Johnson, J Andrews, J Klem, and R Demblewski. New concepts and perspectives on micro-rotorcraft and small autonomous rotary-wing vehicles. Technical report, US Army Aviation, 2002.