Development of unmanned aerial vehicles with advanced safety capabilities

Zhang, Songyuan

2017

Zhang, S. (2017). Development of unmanned aerial vehicles with advanced safety capabilities. Doctoral thesis, Nanyang Technological University, Singapore.

<http://hdl.handle.net/10356/70563>

<https://doi.org/10.32657/10356/70563>

Development of Unmanned Aerial Vehicles with Advanced Safety Capabilities

ZHANG SONGYUAN

School of Electrical and Electronic Engineering

A thesis submitted to the Nanyang Technological University

In partial fulfilment of the requirement for the degree of

Master of Engineering

2017

Acknowledgments

I would like to take this opportunity to express my sincere gratitude and appreciation to all the people who have assisted me during my master study.

First and foremost, I would like to thank my supervisor, Professor Xie Lihua, who has endlessly supported and guided me through out my research. Without his profound knowledge and countless patience, I could not come this far so smoothly.

I would like to thank Mr Quek Jian Xing and Mr Ma Xiao for their valuable technical supports. Furthermore, appreciation goes to Mr Huynh Minh Quan for his kindly assistance whenever needed.

Last but not the least, I would like to thank my husband Mr Wei Dong and my parents. Their loves and supports made me overcome all circumstances.

Contents

1	Introduction	1
1.1	Background & Motivation	1
1.2	Objectives	3
1.3	Contributions	4
1.4	Organization of The Thesis	4
2	Literature Review	6
2.1	Quadcopter Modeling, Simulation & Control	7
2.1.1	Quadcopter Modeling	7
2.1.2	Quadcopter Control	8
2.1.3	Controller Selection	9
2.2	Safety System of UAV	10
2.3	Ground Control Station Development	12
2.4	Conclusion	13
3	Quadcopter Modeling	15
3.1	Introduction	15
3.2	Preliminaries	16
3.2.1	Reference Frames	16
3.2.2	Quadcopter Kinematics	19
3.2.3	Quadcopter Dynamics	20
3.3	Parameter Identification	22
3.3.1	Thrust Coefficient	22
3.3.2	Torque Coefficient	23
3.3.3	Complete Initial Matrix	23
3.3.4	Throttle Command Analysis	24
3.3.5	Gyroscopic Forces	24

3.3.6	Full-term Moment Matrix	25
3.3.7	Equations of State	26
3.3.8	Moment of Inertia	28
3.4	Rotor Thrust Measurement and Flight Data Collection	33
3.4.1	Experiment Setup	34
3.4.2	Coefficient Analysis	35
3.5	Quadcopter Modeling	38
3.6	Conclusion	43
4	Quadcopter Control & Simulation	44
4.1	Introduction	44
4.2	Control & Simulation	45
4.2.1	Control of Quadcopter	45
4.2.2	Simulation	47
4.3	Conclusion	59
5	Ground Control Station Development	60
5.1	Introduction	60
5.2	Design Details	61
5.3	Functional Abilities of Ground Control Station	63
5.3.1	Air Vehicle Control	63
5.3.2	Post-flight Analysis	67
5.4	Conclusion	67
6	UAV Geo-fence Safety System Implementation	68
6.1	Introduction	68
6.2	Area Safety Design	69
6.2.1	Performance Zone	71
6.2.2	Soft Geo-fence	71
6.2.3	Hard Geo-fence	73

6.2.4	Safety Boundary	75
6.2.5	Return Home Route Calculation	76
6.3	Conclusion	77
7	Development of Dynamic Geo-fencing Control	78
7.1	Introduction	78
7.2	Algorithms	79
7.3	Simulation and Results	81
7.4	Conclusion	86
8	Flight Verification Experiment	87
8.1	Introduction	87
8.2	Flight Setup	89
8.3	Experimental Results	90
8.4	Discussion & Conclusion	93
9	Conclusions and Future Works	95
9.1	Conclusions	95
9.2	Future Works	96
A	Appendix	98
A.1	Safety Considerations	98
A.2	Risk Assessment Form	100
B	Publication List	102
	Bibliography	103

List of Figures

1.1	DJI crashed in White House.	2
2.1	Structures of GCS.	13
3.1	Quad VTOL Concept.	15
3.2	The inertial and body frames of a quadcopter.	17
3.3	Configuration Diagram.	17
3.4	Axis labels and rotation conventions (right-hand rotation).	18
3.5	Moment of inertia.	21
3.6	Motors Graphic.	29
3.8	ESC Graphic.	30
3.9	All axes, $J_x = \frac{1}{12}mb^2$ $J_y = \frac{1}{12}ma^2$ $J_z = \frac{1}{12}m(a^2 + b^2)$	30
3.10	Arms Graphic.	31
3.12	Central Hub Graphic.	32
3.14	Experiment Setup Overview.	34
3.15	E-logger Recorded Results shown in Excel.	35
3.16	Data analyzer used to calculate all coefficients.	36
3.17	Linear Fit of RPM vs. Throttle.	37
3.18	Linear Fit of Torque vs. RPM^2	37
3.19	Linear Fit of Thrust vs. RPM^2	38
3.20	UAV Platform.	39
3.21	Mass Moment of Inertia of UAV.	41
4.1	Control Structure Scheme.	46
4.2	Simulation Structure.	47
4.3	PID Controller used to Correct the Angle and Altitude Output.	49
4.4	Roll Controller used to Correct the Angle.	49
4.5	Pitch Controller used to Correct the Angle.	50

4.6	Yaw Controller used to Correct the Angle.	50
4.7	Tuning P Gain.	51
4.8	Tuning I Gain.	52
4.9	Tuning D Gain.	52
4.10	Configuration Chosen in Control Mixing Block.	53
4.11	Quadcopter Output Status.	54
4.12	Motor Throttle Command vs. RPM Value.	55
4.13	Quadcopter Performance in Default View.	56
4.14	Quadcopter Performance in X-Y View.	56
4.15	Quadcopter Output Status when Wind Effect Is Added.	57
4.16	Motor Throttle Command vs. RPM Value when Wind Effect Is Added.	57
4.17	Quadcopter Performance in Default View with Wind Effect.	58
4.18	Quadcopter Performance in X-Y View with Wind Effect.	59
5.1	GCS Field Setup.	61
5.2	GCS Map Display.	61
5.3	Communication-link Architecture.	62
5.4	Traditional structure of GCS design.	62
5.5	GCS Real-time Health Monitoring Window with UAVs connected.	65
5.6	UAVs 100 Formation Shape in Air.	66
5.7	UAVs 100 Formation Shape Displayed on GCS.	66
6.1	Fault Tree Analysis Diagram.	69
6.2	Calculation Illustration.	70
6.3	Geo-fence Design.	71
6.4	Hard Geo-fence Verification Simulation.	73
6.5	Air Resistance Definition in Simulation.	74
6.6	Free-fall Projection Measurement.	75
6.7	Activate Path State Machine.	76

7.1	Dynamic Soft Geo-fencing in Position Controller.	82
7.2	Quadcopter Output Status with Dynamic Soft Geo-fencing Implemented.	83
7.3	Quadcopter Performance in Default View with Dynamic Soft Geo-fence Boundary and UAV Together.	84
7.4	Quadcopter Performance in Default View with Dynamic Soft Geo-fencing Implemented.	85
7.5	Quadcopter Performance in X-Y View with Dynamic Soft Geo-fencing Implemented.	85
8.1	Tuned PID of Quadcopter F450 captured in GCS.	88
8.2	Avionics Systems of UAV.	89
8.3	UAV Flight Path With Static Geo-fencing Capability.	90
8.4	UAV Flight Path With Dynamic Geo-fencing Capability.	91
8.5	Full-path Comparison of two Scenarios.	91
8.6	Comparison of Longitude of two Scenarios.	92
8.7	Distances Comparison of Two Scenarios.	93
9.1	Safety Structure Including Dynamic Soft Geo-fence Capability.	97

List of Tables

2.1	Controller Comparisons.	9
3.1	Numerical value of identified parameters from ground experiment. . .	38
3.2	Parameter of UAV platform Dimension	40
3.3	Mass Moment of Inertia of Three Axes.	41
6.1	Flight Results with Safety System Implemented.	77
7.1	Static Soft geo-fencing vs. Dynamic Soft geo-fencing.	86
8.1	Specific Parameters of Quadcopter.	87
8.2	Summary of Tuned PID of Quadcopter F450.	88
A.1	RISK ASSESSMENT FOR AIRWORTHINESS OF UAS (Part 1) . .	100
A.2	RISK ASSESSMENT FOR AIRWORTHINESS OF UAS (Part 2) . .	101
A.3	RISK ASSESSMENT FOR THE REQUIRED TYPE OF OPERA- TIONS	101

List of Symbols

Greek Variables

$\bar{\omega}$	Angular velocity of the motor
$\bar{\omega}_i$	Speed of each motor
$\bar{\omega}_{ss}$	Expected steady-state RPM value of the motor
ϕ, θ, ψ	Euler angles
ρ	Air density
τ	External torque applied on the quadcopter's body
$\tau_\phi, \tau_\theta, \tau_\psi$	Roll, pitch and yaw torques
$\tau_{\theta gyro}, \tau_{\phi gyro}$	Relationship between gyroscopic torques and pitch, gyroscopic torques and roll action of UAV

Latin Variables

A_r	Cross-sectional area of the propeller's rotation
b	Cross-sectional area of the propeller's rotation
b_i	Bearing to the next geo-fence point (relative to North)
c_R	Conversion coefficient value from throttle % to RPM
C_T	Thrust coefficient for the motor
C_Q	Torque coefficient for the motor

d_i	Distance to the next geo-fence point
F	Force applied on the quadcopter's body
$\mathcal{F}_i$	North-East-Down inertial frame
$\mathcal{F}_b$	Body frame
$\mathcal{F}_v$	Vehicle frame
g	Gravity acceleration
r_m	Radius of the motor
J	Inertial matrix
J_A	Moment of inertia of arm
J_C	Moment of inertia of central hub
J_{ESC}	Moment of inertia of electronic speed controller (ESC)
J_M	Moment of inertia of motor
lat	Latitude of the point
lng	Longitude of the point
M	Quadcopter's total mass
$M_{\mathcal{F}_b}$	Moment matrix
P	Position state
P_m, Q_m	Rolling and pitching rates

Q	Torque force
R	Radius of earth
$R_{\mathcal{F}_b}^{\mathcal{F}_i}$	Rotation matrix maps from the initial frame to body frame
T	Thrust
$Throttle\%$	Throttle command in percentage value
$\dot{v}_{\mathcal{F}_b}$	Linear acceleration
X, Y, Z	Position coordinates in NED frame

Summary

Unmanned Aerial Vehicles (UAVs) have attracted a recurring interest in recent years. UAVs are used to tackle critical tasks in a safer and more efficient way than ground robots in many situations. Since UAVs are often operated in complex environments such as forests, mountains, urban areas, etc., in order to protect surrounding properties and people, a model based UAV control with an advanced safety system is studied.

This thesis aims to address the problems of safety for unmanned aerial vehicles. The focus of the research is to develop a practical solution to enhance the safety more effectively than traditional ways such as a pre-flight check list, etc. Hence, a novel comprehensive safety structure and a dynamic soft geo-fence generation based on model predictive control (MPC) are proposed to enhance safety. A ground control station is developed to monitor the status of UAVs to ensure flight safety.

A particular quadcopter has been developed by taking into account the characteristics of the frame and motors to enhance flight stability and maneuverability. Kinematics and dynamics equations of the quadcopter have been derived and system identification methods were applied to obtain the parameters of the model. Besides, a proportional-derivative (PD) controller was designed. Furthermore, in order to eliminate the steady state errors and to enhance the tracking performance, the controller was further improved to incorporate an integral control in the inner loop and PD controller in the outer loop. A comprehensive simulation has been carried out, which is modified and improved based on an open source simulink written by D. Hartman, K. Landis, M. Mehrer, S. Moreno, and J. Kim at Drexel University. One primary advantage of this improved simulink is the closed-loop feedback control, hence the robustness of flight can be better guaranteed. A simulator and animation program has been created to simulate and visualize the quadcopter's control

mechanisms as well as the pre-designed path flying performance.

To ensure that the UAV would stay inside the operation area and protects surroundings, an advanced safety system featured with a dynamic soft geo-fencing and a fixed hard geo-fencing have been introduced. Dynamic soft geo-fencing has been developed in the outer loop controller and used to calculate the distance between a possible future position and the hard geo-fence boundary after a fixed time duration. A dynamic soft geo-fence has been generated based on current location and the dynamics of the UAV. A model predictive control (MPC) approach was applied to design a controller for braking without breaching the hard geo-fence boundary. The performance of the dynamic soft geo-fencing control has been validated on the simulator.

All works of modeling, control and dynamic soft geo-fencing system have been validated through simulations and a series of real flight tests. Technical challenges as well as future works are highlighted.

Introduction

1.1 Background & Motivation

Autonomous robots are increasingly pivotal in our life recently. Autonomous robots have been improved vastly to the extent where they are used to tackle critical tasks as they are not only more efficient but also safer in many situations. Among them, aerial robots are more dynamic and portable as compared to ground robots in environments such as complex terrains. Hence, autonomous aerial vehicles now play an important role for applications such as data and image acquisitions, localization and tracking of targets, surveillance, geological explorations, target acquisitions, search-and-rescue, multi-Unmanned Aerial Vehicle (UAV) flockings, etc.

The recent generations of UAVs have drawn a lot of attentions with their capability of flying in an urban environment autonomously. As a result, it becomes a perfect platform for onboard sensors and computers that are able to detect and avoid obstacles, locate and navigate in complex urban environments. This kind of UAVs is necessary for both military and civilian usages. For example, having a small UAV to fly autonomously in an urban area, with the ability to spot, identify, designate, and destroy targets effectively, is an advantage for soldiers because this allows them to outmaneuver enemies and increases their safety ultimately.

On the flip side of the coin, there have been a few high-profile incidents of UAVs

recently due to their high accessibility. For instance, there was an incident at the White House where Secret Service agents found a crashed drone on the lawn shown in Figure 1.1. While follow-up news reports seemed to indicate that the owner of drone did not have any hostile intentions, the incident definitely revealed a potential vulnerability to drones at one of the most secure addresses in the world [1].

Figure 1.1: DJI crashed in White House.

In 2015, Civil Aviation Authority of Singapore came up with a new set of rules and regulations, requiring UAV operators to obtain permissions from the authority in order to fly a UAV. For a safe flight, there is a need to double check the hardwares of the UAV. The current norm or traditional method, is to check against a pre-flight check-list which is highly susceptible to human errors. Thus, the need for a better technique to inspect the condition of the UAV before and during flight is required for precise control of UAV and early-warning safety status for operators. In this study, a comprehensive experiment is conducted to get all necessary data for dynamic modeling. Furthermore, a controller is implemented in the auto-pilot board based on the modeling results. Advanced dynamic soft geo-fencing is a safety mechanism as it is usually set to be a fixed boundary to protect civilians.

In order to save the UAV as much as possible as well as protect civilians and properties, an advanced dynamic soft geo-fencing based on model predictive control (MPC) is designed and implemented to maximize the possibility that the UAV returns to a safe zone. Besides, a comprehensive safety system is designed and implemented for safer flight of the UAV. And a ground control station is developed to monitor the flight of multiple UAVs and check all necessary safety statuses in real-time.

1.2 Objectives

This research is focused on safety considerations for UAV flights. Instead of traditional ways which is to prepare a check-list to inspect the hardwares and electronic components of the UAV, a comprehensive safety system with advanced dynamic soft geo-fencing capability and a ground control station are designed and developed to not only enhance the safety of the UAV system, but also increase the recovery rates of the UAV.

Simulation experiments of UAV modeling and control are necessary in order to achieve better performance outputs in real flights. Besides, simulations have the capability of giving a pre-visual effect of the flight path and examining the functionality of dynamic soft geo-fencing system of the UAV. The first version of the safety system is introduced in Chapter 6. However, there is no state prediction in the existing geo-fencing systems, hence UAV will perform a significant delay in geo-fencing. Therefore, a model based predication mechanism is introduced in this thesis to estimate future statuses based on the UAV's dynamic model. A ground control station (GCS) is also developed to monitor statuses of the UAV to decrease the errors caused by the large amount of workload of operators. The novel part of this research is to propose a model predictive control (MPC) method that informs the UAV before reaching geo-fence boundary based on the model. Therefore, the

UAV could fly safely.

1.3 Contributions

A model based dynamic geo-fencing algorithm for UAV safety system is developed and tested in both simulations and real flight situations. Firstly, a DJI F-450 10-inch quadcopter [2] is customized and modeled. Secondly, a closed-loop feedback UAV control scheme is designed for flight robustness and stability. In addition, a dynamic soft geo-fencing system is developed as the second outer loop controller to maximize the possibility to recover the UAV. Besides, a comprehensive on-board safety system is developed to handle different sensors failures before and during UAV flights. Moreover, a functional ground control station is developed for single UAV and multiple UAVs operations. Furthermore, simulations are conducted using the proposed control scheme and model. Last but not the least, a series of flight tests are accomplished to verify the results of this research.

1.4 Organization of The Thesis

This thesis is organized as follows:

Chapter 1 briefly summarizes the motivation, the objective of research as well as the contributions of the thesis.

Chapter 2 reviews the related works.

Chapter 3 describes a detailed dynamic modeling method to model UAV for simulations and controls.

Chapter 4 establishes a closed-loop feedback control simulation testbed for dynamic geo-fencing.

Chapter 5 introduces a ground control station which is designed for multiple UAVs

outdoor formation shows.

Chapter 6 discusses a safety system which is integrated in auto-pilot board.

Chapter 7 proposes a novel dynamic soft geo-fencing module as the second outer loop using model predictive controller and combining with the feedback control loop of the UAV.

Chapter 8 investigates the comprehensive safety system with dynamic soft geo-fencing capability of the UAV in both simulations and real flight tests in outdoor environments.

Chapter 9 provides conclusions of research work and recommendations of this research.

Literature Review

According to the news, there are around 675,000 consumer drones flying in the USA. And in the world scope, this number is estimated to be from two quarters to three quarters million [3]. The huge number of UAVs flying in the air may pose a lot of potential risks. If not being operated properly, the operations of unmanned aircraft may become a risk to aviation and public environment [4]. Civil Aviation Authority of Singapore requires an operator permit and/or activity permit under several circumstances and provides guidelines when flying UAV outdoor in Singapore. See Appendix A.1. Most UAV safety considerations are pre-flight check mechanisms [5] and geo-fencing system. A geo-fence is a virtual perimeter of a real-world geographic area. A geo-fence could be dynamically generated like a circle around a point location or can be a predefined set of boundaries [6]. A commercialized geo-fencing system is firstly introduced by DJI [7]. To have a more robust safety system of the UAV, a comprehensive safety system which is integrated into the autopilot of the UAV has been introduced in [8]. This is the first version of the safety system where a hierarchical three-layered safety boundaries and corresponding actions are adopted when safety system is triggered if specific sensors fail.

2.1 Quadcopter Modeling, Simulation & Control

Quadrotors as compared to conventional rotor-crafts have a simple mechanical structure and flight principle. The quadcopters use four motors in a symmetrical rigid structure which could be considered as omni-directional vehicles which remove the need of a tail rotor in conventional helicopter. The generated reactive torques are inherently canceled since the two pairs of rotors rotate in opposite directions. The complexity of the platforms is reduced by using the fixed pitch blades and mechanical vibration is decreased due to the small propeller size.

2.1.1 Quadcopter Modeling

Modeling of the UAV has been studied a lot in recent years. Mathematical modeling and simulation of an unmanned aerial vehicle are necessary for a better control of the precise movement of the quadcopter. Dynamic model of UAV is fundamental for all studies. Nevertheless, it is not an easy task to model the quadcopter due to its complex structure, non-linear dynamics and under-actuated nature. Various methods are used in practice. A dynamic model is developed using MATLAB/SIMULINK software platform in many papers such as [9]. A traditional method introduced in [10] has used differential equations of the quadcopter dynamics derived from both the Newton-Euler equations and the Euler-Lagrange equations. In [11], the popular and a standard 4-propeller quadcopter has been modeled using a non-linear modeling method. [12] has extended previous works on more complex aerodynamic effects impacting quadcopter flight in regimes beyond nominal hover conditions. In [13], the issues that arise when the UAV is deviating significantly from the hover flight regime have been investigated. Three separate aerodynamic effects including vehicular velocity, angle of attack, and airframe design are investigated as they are related to the performance of quadcopter's flight.

2.1.2 Quadcopter Control

There are many studies related to system control techniques such as proportional-integral-derivative controller (PID controller) or proportional-derivative controller (PD controller) [14, 15, 16, 17, 18], linear-quadratic regulator (LQR) controllers [19], non-linear H_∞ control [20, 21], backstepping control [22, 23, 24, 25, 26], and fuzzy logic controller [27, 28, 29].

In [24], a quaternion-based feedback control using PD controller as control scheme for attitude stabilization of UAV has been proposed. In the PD controller, the proportional actions are in terms of the quaternion vector and the two derivative actions are based on the airframe angular velocity and the quaternion velocity. In [17], both PD controller and backstepping controller have been used for attitude stabilization and hovering control. The flight control structure is not only for stabilizing the attitude of UAV but also tracking a path accurately. A structure that includes backstepping controller for stabilization and PID controller in inner and outer loop has been introduced in [16]. Moreover, a PID controller has been proposed for the pitch control of the UAV [15]. More complicated fuzzy PID controller has been developed for attitude stabilization in [14]. Besides, a method has been introduced in [20] where the dynamic model of the UAV is firstly transformed into a linear time-invariant (LTI) system and then approximated by a single-input and single-output (SISO) system. This step is developed due to the low coupling among the canals of the system. They use H_∞ theory to make controller more robust. In [21], a state-feedback H_∞ controller has been used in inner loop structure to stabilize the UAV against external disturbances. A fractional PID controller in inner loop for attitude control and backstepping in outer loop to achieve trajectory tracking has been introduced in [25]. Moreover, in [26], a backstepping controller has been used to generate the trajectory of the UAV and stabilize it in hovering attitude. Besides, a new controller called “Integral Backstepping” has been designed in [24].

<i>Control Algorithm</i>	<i>Characteristic</i>												
	Robust	Adaptive	Optimal	Intelligent	Tracking Ability	Precision	Simplicity	Disturbance rejection	Manual tuning	Unmodelled parameter handling	Fast convergence/response	Noise	Energy loss
PID/PD	1	0	0	0	1	1	2	0	2	0	1	2	0
LQR	0	2	1	0	1	0	1	1	1	0	1	1	0
Backstepping	0	2	0	0	2	1	0	2	0	1	0	0	0
H_∞	2	1	2	0	2	1	0	1	0	1	0	0	0
Fuzzy logic	1	1	1	2	1	1	1	1	1	0	1	0	0

Legend: 0 - 2 indicates the trends of characteristics performance from worst to best.

Table 2.1: Controller Comparisons.

The concept of this is to add tracking error integral terms to the normal backstepping controller to handle uncertainties in the controller. In [29], a self-tuning PID based on fuzzy logic has been proposed. UAV could achieve robustness through self-adjusting the gain of the PID controller against uncertain parameters, such as variations in payloads.

2.1.3 Controller Selection

From the discussions mentioned previously, many UAV controllers consist of an inner and outer loop structure. Differentiations and comparisons are essential to make a selection. Comparison results are based on [30, 31]. A comparison of those controllers mentioned earlier is shown in Table 2.1.

Instead of LQR, backstepping controller, etc., a classical PID controller is chosen due to its simplicity and convenience of the implementation on the autopilot and for ease in test flights. Furthermore, [32] focused on the position control where an inner-outer loop control structure for a quadcopter with state and input constraints is adopted. The outer loop generates a saturated thrust, and the reference roll and pitch angles, while the inner loop is designed to follow these reference angles using a traditional PID controller. Hence, a PID controller and PD controller are chosen for the inner and outer loops, respectively.

2.2 Safety System of UAV

Safety is a critical consideration of UAV operation, especially in outdoor flights where wind disturbances are large. Several articles focus on the safety discussions of the potential costs of crashing the UAV into the surroundings if on-board components fail. In [33], it states that safety is paramount in Singapore, especially for outdoor flights and Civil Aviation Authority of Singapore introduces laws and regulations to control UAV flying activities. In most parts of United States, these have been introduced in [34] to impose size restrictions on operating small UAVs. This research is focused on improving the safety system introduced in [8]. Corresponding actions will be activated when on-board sensors failures such as GPS signal loss, inertial measurement unit (IMU) failures. A safety system with a dynamic soft geo-fencing capability using model predictive controller (MPC) is studied.

A lot of researches have been done on using MPC as the tracking controller for high-level reference tracking [35, 36, 37, 38]. The advantages of MPC compared to other controllers such as PID and LQR are that it has the ability to anticipate future events and constraints in both states and inputs into consideration. Dynamic modeling of the UAV is required for the implementation of MPC controller. There

are three main approaches to a robust MPC:

- Min-Max MPC. The optimization in this formulation considers all possible evolutions of the disturbance. It is good for linear robust control systems, but with a high computational cost in [39] to avoid the likely feasibility problems that optimize a single control profile over all possible future disturbance realizations.
- Constraint Tightening MPC. A given margin is used to enlarge the state constraints in order to find the trajectory under any evolution of disturbance [40].
- Multi-stage MPC. In a multi-stage non-linear MPC system, a discrete scenario-tree is used to deal with the uncertainties with sets of samples. Future control inputs depend on the previous realizations of the uncertainty. A closed-loop approach is conducted to achieve better performances other than an open-loop since it will consider all measurements information [41].
- Tube MPC. A feedback controller is used in this independent nominal model of system to ensure that the actual state is converged to the nominal state [42]. Robust positively invariant (RPI) set is considered for determining the amount of the separation required between UAVs. The RPI set is a combination of all possible state deviations that may be introduced by disturbance with the feedback controller.

From the list above, the Min-Max MPC is used for dynamic soft geo-fencing system as it is the best choice for a linear robust control system after careful comparison.

2.3 Ground Control Station Development

Ground control station (GCS) is an indispensable part for the safety system of a UAV. There are various ground control stations developed for different tasks and applications, such as surveillance missions [43], voice assist GCS [44]. In [43], a ground control station has been designed as a decentralized system with a 3D realistic environment for a platform composed of multiple UAVs in surveillance missions. Speech commands used for controlling UAV have been proposed in [44]. The control commands are generated and transmitted to the UAV through the GCS.

The main functions of GCS include:

- Status monitoring. Monitor the status and movement of the UAV, such as roll, pitch and yaw angles, air speeds, flight trajectories, etc. Operators could always take appropriate actions based on the status of the UAV.
- Mission control. Path planning and flight mode mechanism are all included in mission control. Operators could choose a different control method based on the flight mode.
- Data processing. Data parsing, storage and playback could be achieved by this function.
- Debugging function. Operators could debug the parameters of the control algorithm to check the status and measurement values of sensors.

The main functions of the GCS are normally divided into four parts as shown in Figure 2.1 [45].

Figure 2.1: Structures of GCS.

2.4 Conclusion

To conclude this chapter, current research contributions of the UAV modeling, simulation and control, safety considerations and GCS development have been discussed and reviewed. Firstly, modeling methods have been reviewed to obtain a precise UAV model. Besides, different UAV controlling methods have been compared. A closed-loop feedback with a PID controller in inner and PD controller in outer loop

have been chosen on account of the feasibility and robustness. Furthermore, UAV safety considerations and MPC control have been emphasized which are useful for the safety structure designs and soft geo-fencing system implementations. Last but not the least, the existing developments of the GCS have been highlighted to make the safety structure more complete.

Quadcopter Modeling

3.1 Introduction

UAV platform without proper modeling and control tests will often cause a lot of troubles, such as instability, control difficulties, etc. A vertical take-off and landing (VTOL) quadcopter platform is studied. A VTOL is one that can hover, take off, and land vertically. This classification includes fixed-wing aircraft [46] as well as helicopters and other aircraft with powered rotors as shown in Figure 3.1, such as cyclogyros/cyclocopters and tiltrotors. In this research, we focus on building a quadcopter platform for search and rescue purpose. In order to achieve the greatest possible safety, a thoroughly modeled quadcopter platform is studied.

Figure 3.1: Quad VTOL Concept.

Aircraft flights could be described by the principles of classical mechanics. In this research, it is modeled based on individual particles, which could also be referred to as point masses. The control behavior of the UAV system can be described by a dynamic model. Maneuvering in local X, Y, and Z depends on variances of motor speeds. The dynamic model of the system can be used to develop a control strategy in such way that UAV is capable of achieving desired motions.

Kinematic and dynamics of the UAV have to be defined before deriving a full UAV mathematical model. A relationship between the UAV's position and velocity is defined by kinematic equations. The dynamic model explains the resulting acceleration forces.

3.2 Preliminaries

3.2.1 Reference Frames

Reference [47] explains the adopted coordinates and reference frames of quadcopter. Besides, the author shows the transformations among different coordinate systems.

There are three main frames of reference as shown in Figure 3.2 [47], i.e. inertial frame, body frame, and vehicle frame. Moreover, torque τ_{Mi} , force f_i and angular velocity ω_i ($i = 1, 2, 3, 4$) are generated by motor M_i accordingly.

1. Inertial frame

$\mathcal{F}_i = (\vec{x}_i, \vec{y}_i, \vec{z}_i)$ is a fixed coordinate reference with the origin located at the earth ground, for example, defined at the base station. In our system, a North-East-Down (NED) reference system is used which is x-axis towards North, y-axis towards East, and z-axis towards the center of the earth.

Figure 3.2: The inertial and body frames of a quadcopter.

2. Body frame

$\mathcal{F}_b = (\vec{x}_b, \vec{y}_b, \vec{z}_b)$, with its origin located at the center of gravity (COG) of the quadcopter. It is important to note that there are two types of quadcopter reference behaviors: $+$ type and $\times$ type, both of which are illustrated in Figure 3.3 [48].

Figure 3.3: Configuration Diagram.

The body frame reference behavior used in this research is the $\times$ type quadcopter, x -axis is along the center line of the two front rotors, and y -axis points towards the center line of two right rotors. Z -axis $\vec{z}_b = \vec{x}_b \times \vec{y}_b$, where $\times$ denotes the cross product, is facing downward of the quadcopter.

3. Vehicle frame

With the definition of $\mathcal{F}_v = (\vec{x}_v, \vec{y}_v, \vec{z}_v)$, the origin of the vehicle frame is at the Center of Gravity (COG) of the quadcopter. There are two variables $\mathcal{F}_\phi$ and $\mathcal{F}_\theta$. $\mathcal{F}_\phi$ is aligned with vehicle frame. $\mathcal{F}_\theta$ rotates about its z-axis $\vec{z}_v$ by an angle ψ so that $\vec{x}_v$ and $\vec{y}_v$ are aligned with $\vec{x}_b$ and $\vec{y}_b$, respectively. $\mathcal{F}_\theta$ is the $\mathcal{F}_\phi$ frame rotating about its y-axis, $\vec{y}_\phi$, by a pitching angle, θ , such that $\vec{x}_\phi$ and $\vec{z}_\phi$ are aligned with $\vec{x}_b$ and $\vec{z}_b$ respectively.

Both motor 1 and motor 3 spin in a counter-clockwise direction while both motor 2 and motor 4 spin in the opposite direction of its adjacent motors. The quadcopter rotation conventions are shown in Figure 3.4 [48].

Figure 3.4: Axis labels and rotation conventions (right-hand rotation).

Translation and rotation matrices are used to transform from one coordinate reference frame into another desired frame of reference. For example, the transformation from $\mathcal{F}_i$ to $\mathcal{F}_v$ provides the displacement vector from the origin of the inertial frame to the COG of the quadcopter. Also, the transformation from $\mathcal{F}_v$ to $\mathcal{F}_b$ is rotational in nature, thereby yielding the roll, pitch and yaw angles. Notation will be discussed before jumping into a detailed modeling process. The general notation is such that the rotation matrix $R_{\mathcal{F}_2}^{\mathcal{F}_1} \in R^{3 \times 3}$ mapping a given frame $\mathcal{F}_1$ to another frame $\mathcal{F}_2$ will be used in Quadrotor Kinematics and Quadrotor Dynamic Sections.

The twelve state variables and corresponding notations defined below will be used throughout this text:

- $\dot{P}_{\mathcal{F}_i}^T = [\dot{X}, \dot{Y}, -\dot{Z}]_{\mathcal{F}_i}$
- $\dot{\Omega}_{\mathcal{F}_v}^T = [\dot{\phi}, \dot{\theta}, \dot{\psi}]_{\mathcal{F}_v}$
- $\ddot{P}_{\mathcal{F}_b}^T = [\ddot{X}, \ddot{Y}, -\ddot{Z}]_{\mathcal{F}_b}$
- $\ddot{\Omega}_{\mathcal{F}_b}^T = [\ddot{\phi}, \ddot{\theta}, \ddot{\psi}]_{\mathcal{F}_b}$

3.2.2 Quadcopter Kinematics

Assume the vehicle frame of UAV is always aligned with the NED frame, with the knowledge of the inertial frame position state variables X , Y and Z which are with respect to x, y and z axis, the body frame position state variables are X , Y and Z . The translational motion relationship [47] is derived as

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ -\dot{Z} \end{bmatrix}_{\mathcal{F}_i} = \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix}_{\mathcal{F}_v} = [R_{\mathcal{F}_v}^{\mathcal{F}_b}]^T \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix}_{\mathcal{F}_b} \quad (3.1)$$

where $[R_{\mathcal{F}_v}^{\mathcal{F}_b}]^T \in R^{3 \times 3}$ is the matrix that maps frame $\mathcal{F}_b$ to frame $\mathcal{F}_v$ and is defined by

$$[R_{\mathcal{F}_v}^{\mathcal{F}_b}]^T = \begin{bmatrix} c\theta c\psi & s\phi s\theta c\psi - c\phi s\psi & c\phi s\theta c\psi + s\phi s\psi \\ c\theta s\psi & s\phi s\theta s\psi + c\phi c\psi & c\phi s\theta s\psi - s\phi c\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{bmatrix} \quad (3.2)$$

where $s\theta = \sin \theta$, $c\theta = \cos \theta$. Likewise for $s\phi$, $c\phi$, $s\psi$ and $c\psi$.

Hence, by using vehicle frame angles (ϕ, θ, ψ) and angular velocities $(\dot{\phi}, \dot{\theta}, \dot{\psi})$, the rotational motion equation can be obtained. By applying $\mathcal{F}_\phi$, $\mathcal{F}_\theta$, and $\mathcal{F}_v$ to the body frame of reference $\mathcal{F}_b$, we get:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}_{\mathcal{F}_b} = R_{\mathcal{F}_\phi}^{\mathcal{F}_b}(\phi) \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + R_{\mathcal{F}_\phi}^{\mathcal{F}_b}(\phi) R_{\mathcal{F}_\theta}^{\mathcal{F}_\phi}(\theta) \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + R_{\mathcal{F}_\phi}^{\mathcal{F}_b}(\phi) R_{\mathcal{F}_\theta}^{\mathcal{F}_\phi}(\theta) R_{\mathcal{F}_\psi}^{\mathcal{F}_\theta}(\psi) \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \quad (3.3)$$

where $R_{\mathcal{F}_\theta}^{\mathcal{F}_\phi} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\phi & s\phi \\ 0 & -s\phi & c\phi \end{bmatrix}$, $R_{\mathcal{F}_v}^{\mathcal{F}_\theta} = \begin{bmatrix} c\theta & 0 & -s\theta \\ 0 & 1 & 0 \\ s\theta & 0 & c\theta \end{bmatrix}$, and $R_{\mathcal{F}_\phi}^{\mathcal{F}_b}(\phi) = R_{\mathcal{F}_\theta}^{\mathcal{F}_\phi}(\theta) = R_{\mathcal{F}_\psi}^{\mathcal{F}_\theta}(\psi) = I$. Therefore,

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}_{\mathcal{F}_b} = \begin{bmatrix} 1 & 0 & -s\theta \\ 0 & c\phi & s\phi c\theta \\ 0 & -s\phi & c\phi c\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}_{\mathcal{F}_v} \quad (3.4)$$

It follows that

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}_{\mathcal{F}_v} = \begin{bmatrix} 1 & s\phi \tan \theta & c\phi \tan \theta \\ 0 & c\phi & -s\phi \\ 0 & s\phi \sec \theta & c\phi \sec \theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}_{\mathcal{F}_b} \quad (3.5)$$

Equations (3.1) and (3.5) represent the quadcopter's equations of motion.

3.2.3 Quadcopter Dynamics

The quadcopter dynamics are obtained through combining aerodynamics forces and blade theory. The quadcopter has four motors with propellers mounted on four arms extended at 90° to each other. Power is applied through the attached battery to create torque on the rotor shaft which in turn creates thrust for each rotor. Each of the rotors also creates torque in the blade area which affects the whole quadcopter's dynamics. An initial modeling of the quadcopter is conducted while considering limited flight envelopes, where only hovering or low velocity flight are considered during the derivation of the dynamics shown below.

To achieve this goal, the quadcopter's dynamic and kinematic equations have to be defined. The dynamic model describes how applied forces and torques result in translational and rotational accelerations. The relation between the vehicle's position and velocity is described through its kinematic equations[47].

The quadcopter is approximated as a circular sphere of radius r and mass M_0 for calculating the moment of inertia. Each motor carries a mass of m , links through arm with the length of l and connects to central sphere finally. The illustration is as shown in Figure 3.5.

Figure 3.5: Moment of inertia.

Symmetrical inertial matrix is defined by

$$J = \begin{bmatrix} J_x & 0 & 0 \\ 0 & J_y & 0 \\ 0 & 0 & J_z \end{bmatrix} \quad (3.6)$$

Due to the symmetry property of quadcopter, the matrix is diagonal and identical for both “+” and “X” configurations. The diagonal form of the matrix is convenient to further invert it for usage in the angular velocity equation.

The dynamics equation for quadcopter by applying external force is shown below.

$$\begin{bmatrix} MI_{3 \times 3} & 0 \\ 0 & I_{3 \times 3} \end{bmatrix} \begin{bmatrix} \ddot{P}_{\mathcal{F}_b} \\ \ddot{\Omega}_{\mathcal{F}_b} \end{bmatrix} + \begin{bmatrix} \dot{\Omega}_{\mathcal{F}_b} \times M \dot{P}_{\mathcal{F}_b} \\ \dot{\Omega}_{\mathcal{F}_b} \times J \dot{\Omega}_{\mathcal{F}_b} \end{bmatrix} = \begin{bmatrix} F_{\mathcal{F}_b} \\ \tau_{\mathcal{F}_b} \end{bmatrix} \quad (3.7)$$

where M is the quadcopter's total mass, $F^T = [f_x, f_y, f_z]$ and $\tau^T = [\tau_\phi, \tau_\theta, \tau_\psi]$ are

the external forces and torque vectors applied on the quadcopter's COG. The terms τ_ϕ , τ_θ , and τ_ψ are the roll, pitch and yaw torques respectively.

Thus, the translational dynamic model can be written as

$$\begin{bmatrix} \ddot{X} \\ \ddot{Y} \\ \ddot{Z} \end{bmatrix}_{\mathcal{F}_b} = \begin{bmatrix} \dot{\psi}\dot{Y} - \dot{\theta}\dot{Z} \\ \dot{\phi}\dot{Z} - \dot{\psi}\dot{X} \\ \dot{\theta}\dot{X} - \dot{\phi}\dot{Y} \end{bmatrix}_{\mathcal{F}_b} + \frac{1}{M} \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix}_{\mathcal{F}_b} \quad (3.8)$$

while the rotational model is

$$\begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = J^{-1} \left\{ \begin{bmatrix} 0 & \dot{\psi} & -\dot{\theta} \\ -\dot{\psi} & 0 & \dot{\phi} \\ \dot{\theta} & -\dot{\phi} & 0 \end{bmatrix} J \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} + \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} \right\} = \begin{bmatrix} \frac{J_y - J_z}{J_x} \dot{\theta} \dot{\psi} \\ \frac{J_z - J_x}{J_y} \dot{\phi} \dot{\psi} \\ \frac{J_x - J_y}{J_z} \dot{\theta} \dot{\phi} \end{bmatrix}_{\mathcal{F}_b} + \begin{bmatrix} \frac{1}{J_x} \tau_\phi \\ \frac{1}{J_y} \tau_\theta \\ \frac{1}{J_z} \tau_\psi \end{bmatrix}_{\mathcal{F}_b} \quad (3.9)$$

3.3 Parameter Identification

Consider all the equations in Section 3.2.1. Note that the quadcopter exhibits under-actuated dynamics where there are six DOFs but only four control inputs.

3.3.1 Thrust Coefficient

The thrusts generated by four motors, are integrated for control designs and simulations. Equation (3.10) shows the thrust, T , which is provided by a single motor and can be calculated as,

$$T = C_T \rho A_r r_m^2 \bar{\omega}^2 \quad (3.10)$$

where C_T is the thrust coefficient for the motor, ρ is the air density, A_r is the cross-sectional area of the propeller's rotation, r_m is the radius of the motor, and $\bar{\omega}$ is the

angular velocity of the motor. For flight modeling, the characterization process is simplified based on a lumped parameter approach [11], hence get Equation (3.11).

$$T = c_T \bar{\omega}^2 \quad (3.11)$$

where c_T is the lumped parameter of thrust coefficient. The thrust provided by each motor provides a force which is perpendicular to the X-Y plane of the body frame of the UAV in the positive Z direction.

3.3.2 Torque Coefficient

The torque force must be considered in order to model yaw movement using the same method of describing thrust. The related lumped parameter of torque coefficient could be described as,

$$Q = c_Q \bar{\omega}^2 \quad (3.12)$$

3.3.3 Complete Initial Matrix

With the data mentioned above, a specific matrix that describes the thrusts and torques together with Equation (3.13) and (3.14) is created. Equation (3.13) is for the “+” configuration while Equation (3.14) is for the “X” configuration.

$$\begin{bmatrix} \sum T \\ \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} c_T & c_T & c_T & c_T \\ 0 & dc_T & 0 & -dc_T \\ -dc_T & 0 & dc_T & 0 \\ -c_Q & c_Q & -c_Q & c_Q \end{bmatrix} \begin{bmatrix} \bar{\omega}_1^2 \\ \bar{\omega}_2^2 \\ \bar{\omega}_3^2 \\ \bar{\omega}_4^2 \end{bmatrix} \quad (3.13)$$

$$\begin{bmatrix} \sum T \\ \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} c_T & c_T & c_T & c_T \\ -dS_{45} & dS_{45}c_T & dS_{45}c_T & -dS_{45}c_T \\ -dS_{45}c_T & -dS_{45}c_T & dS_{45}c_T & dS_{45}c_T \\ -c_Q & c_Q & -c_Q & c_Q \end{bmatrix} \begin{bmatrix} \bar{\omega}_1^2 \\ \bar{\omega}_2^2 \\ \bar{\omega}_3^2 \\ \bar{\omega}_4^2 \end{bmatrix} \quad (3.14)$$

where d is the distance between the motors and the respectively axes of rotation. dS_{45} means $d \times \sin(45^\circ)$ which indicates the distance between the motors and the body's axes of rotation.

3.3.4 Throttle Command Analysis

Another analysis is based on the throttle command consideration. The purpose of this is that the coefficients of thrust and torque are based on the RPM of the motors but not directly the throttle command, hence it is necessary to analyze the linear expression to translate the throttle command value to the corresponding RPM value. The relation is given by,

$$\bar{\omega}_{ss} = (Throttle\%)c_R + b \quad (3.15)$$

where $\bar{\omega}_{ss}$ is the expected steady-state RPM value of the motor, $Throttle\%$ is the throttle command in percentage value, c_R is the conversion coefficient value from throttle % to RPM, and b is the y-intercept of the linear regression relationship when the throttle is at 0%.

3.3.5 Gyroscopic Forces

There is one set of forces from gyroscopic precession of the motors before arriving at the full moment matrix. They are affecting the body and are generated by the moment of inertia of each motor's rotating components J_M , the rolling and pitching rates are P_m and Q_m , and the speed of each motor is $\bar{\omega}_i$. The Equation (3.16)

and (3.17) show the relationship between gyroscopic torques and pitch, gyroscopic torques and roll action of UAV, and $\frac{\pi}{30}$ is the term corresponding to the transition from RPM to radians.

$$\tau_{\theta_{gyro}} = J_M P_m \left(\frac{\pi}{30} \right) (-\bar{\omega}_1 + \bar{\omega}_2 - \bar{\omega}_3 + \bar{\omega}_4) \quad (3.16)$$

$$\tau_{\phi_{gyro}} = J_M Q_m \left(\frac{\pi}{30} \right) (\bar{\omega}_1 - \bar{\omega}_2 + \bar{\omega}_3 - \bar{\omega}_4) \quad (3.17)$$

3.3.6 Full-term Moment Matrix

The final full-term matrix is created after considering aerodynamics, gyroscopic, thrust and torque of the quadcopter system. The Equation (3.18) and Equation (3.19) below are for “+” configuration and “X” configuration, where M is moment.

$$M_{\mathcal{F}_b} = \begin{bmatrix} dc_T \bar{\omega}_2^2 - dc_T \bar{\omega}_4^2 + J_M Q_m \left(\frac{\pi}{30} \right) (\bar{\omega}_1 - \bar{\omega}_2 + \bar{\omega}_3 - \bar{\omega}_4) \\ -dc_T \bar{\omega}_1^2 + dc_T \bar{\omega}_3^2 + J_M P_m \left(\frac{\pi}{30} \right) (-\bar{\omega}_1 + \bar{\omega}_2 - \bar{\omega}_3 + \bar{\omega}_4) \\ -c_Q \bar{\omega}_1^2 + c_Q \bar{\omega}_2^2 - c_Q \bar{\omega}_3^2 + c_Q \bar{\omega}_4^2 \end{bmatrix} \quad (3.18)$$

$$M_{\mathcal{F}_b} = \begin{bmatrix} dS_{45} c_T \bar{\omega}_2^2 - dS_{45} c_T \bar{\omega}_4^2 + J_M Q_m \left(\frac{\pi}{30} \right) (\bar{\omega}_1 - \bar{\omega}_2 + \bar{\omega}_3 - \bar{\omega}_4) \\ -dS_{45} c_T \bar{\omega}_1^2 + dS_{45} c_T \bar{\omega}_3^2 + J_M P_m \left(\frac{\pi}{30} \right) (-\bar{\omega}_1 + \bar{\omega}_2 - \bar{\omega}_3 + \bar{\omega}_4) \\ -c_Q \bar{\omega}_1^2 + c_Q \bar{\omega}_2^2 - c_Q \bar{\omega}_3^2 + c_Q \bar{\omega}_4^2 \end{bmatrix} \quad (3.19)$$

The total forces acting on the body of UAV are generated by gravity and motor thrust. By considering the lifting thrust in the positive Z direction, the force could be expressed as,

$$F_{\mathcal{F}_b} = \begin{bmatrix} 0 \\ 0 \\ c_T (\bar{\omega}_1^2 + \bar{\omega}_2^2 + \bar{\omega}_3^2 + \bar{\omega}_4^2) \end{bmatrix} \quad (3.20)$$

It should be noted that while we say there are acting aerodynamic forces, it is assumed that the static thrust and torque tests capture the elements of aerodynamics that we are interested in.

3.3.7 Equations of State

The Equation (3.21) shows the equation of angular velocity state which considers the motor's inertia, moments, and angular velocity.

$$\dot{\omega}_{\mathcal{F}_b} = (J)^{-1}[M_{\mathcal{F}_b} - \dot{\Omega}_{\mathcal{F}_b} J \omega_{\mathcal{F}_b}] = \begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} \quad (3.21)$$

where $\dot{\omega}_{\mathcal{F}_b}$ is the angular velocity across each axis in the body frame with respect to the inertial frame.

The next equation to be derived is the rotational velocity matrix based on the inertial matrix and moment matrix. It is a cross-product matrix and shown in Equation (3.22).

$$\Omega_{\mathcal{F}_b} = \begin{bmatrix} 0 & -\dot{\psi} & \dot{\theta} \\ \dot{\psi} & 0 & -\dot{\phi} \\ -\dot{\theta} & \dot{\phi} & 0 \end{bmatrix} \quad (3.22)$$

$\omega_{\mathcal{F}_b}$ as shown in Equation (3.23), describes the rotational velocity of the UAV in body frame. It is directly defined by roll, pitch and yaw rates.

$$\omega_{\mathcal{F}_b} = \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (3.23)$$

The rotation matrix that maps from the initial frame to the body frame is described in Equation (3.24).

$$R_{\mathcal{F}_b}^{\mathcal{F}_i} = \begin{bmatrix} c(\theta)c(\psi) & c(\theta)s(\psi) & -s(\theta) \\ -c(\phi)s(\psi) + s(\phi)s(\theta)c(\psi) & c(\phi)c(\psi) + s(\phi)s(\theta)s(\psi) & s(\phi)c(\theta) \\ s(\phi)s(\psi) + c(\phi)s(\theta)c(\psi) & s(\phi)c(\psi) + c(\phi)s(\theta)s(\psi) & c(\phi)c(\theta) \end{bmatrix} \quad (3.24)$$

where $s(\theta) = \sin(\theta)$, $c(\theta) = \cos(\theta)$. Likewise for $s(\phi)$, $c(\phi)$, $s(\psi)$ and $c(\psi)$. Using this rotation matrix, the quadcopter's angular velocity in the body frame related to the changes in the angle rotation shown in Equation (3.25), can be obtained.

$$\omega_{\mathcal{F}_b} = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + R_\phi \left(\begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + R_\theta \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \right) \quad (3.25)$$

where R_ϕ and R_θ are from Equation (3.24).

The velocity state equation in Equation (3.26) describes the relationship between the acceleration of the COG of the UAV, forces and torques acting on the body of frame.

$$\dot{v}_{\mathcal{F}_b} = \left(\frac{1}{M} \right) F_{\mathcal{F}_b} + g_{\mathcal{F}_b} - \Omega_{\mathcal{F}_b} \omega_{\mathcal{F}_b} = \begin{bmatrix} \dot{U} \\ \dot{V} \\ \dot{W} \end{bmatrix} \quad (3.26)$$

where $\dot{v}_{\mathcal{F}_b}$ is the linear acceleration of the COG of the UAV in the body frame with respect to the inertial frame. M is the total mass of the UAV which includes all the payloads, while $g_{\mathcal{F}_b}$ is the gravitational acceleration translated from the inertial frame to the body frame by the rotation matrix $\begin{bmatrix} R_{\mathcal{F}_b}^{\mathcal{F}_i} \end{bmatrix}$ using Equation (3.27).

$$g_{\mathcal{F}_b} = \begin{bmatrix} R_{\mathcal{F}_b}^{\mathcal{F}_i} \end{bmatrix} g_{\mathcal{F}_i} \quad (3.27)$$

Last but not the least, the position state equation is shown in Equation (3.28). It is used to establish the relationship between the linear velocity of the COG of the UAV in the inertial frame.

$$\dot{P}_{\mathcal{F}_i} = \begin{bmatrix} R_{\mathcal{F}_b}^{\mathcal{F}_i} \end{bmatrix} v_{\mathcal{F}_b} = \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} \quad (3.28)$$

3.3.8 Moment of Inertia

Moment of inertia determines how rotational velocity is affected by an applied torque. The effect is not only related to the mass of the UAV but also the distribution of the mass around the rotation axis. The moment of inertia matrix is mentioned previously in Equation (3.6). We have made an assumption that the quadrotor's structure is perfectly symmetric about x, y, and z axis, the center of mass for each arm is located at its geometric center. Since the shape is symmetrical to all axes, the mass distribution of the quadrotor to be tested is equivalent to the quadrotor which is made of several cylinder structures. Therefore, the components of the quadrotor can be considered as several cylinders.

The UAV is assumed to be perfectly symmetric about all axes and the COG is located at the geometric center of the body. Due to the symmetry of the UAV, the J_x and J_y are identical. The parallel-axis theorem is used to determine the mass moment of inertia of each part of UAV. Each part of UAV is considered as a solid cylinder. The parallel-axis equation is shown below,

$$J_{parallel-axis} = J_{COM} + mr^2 \quad (3.29)$$

where J_{COM} is the moment of inertia of each individual part of UAV about its own axis in the following discussions, which passes through its center of mass and is parallel to the axis of motion desired. m is the mass of the part, r is the perpendicular distance between the parallel axes.

3.3.8.1 Motor ($J_{x,M}$, $J_{y,M}$, and $J_{z,M}$)

Equations below are used to describe the cylinder rotating around an end diameter and cylinder rotating around a central axis.

Figure 3.6: Motors Graphic.

Cylinder rotating around an end diameter

(a) X and Y axes,
 $J_M = \frac{1}{4}mr^2 + \frac{1}{3}mh^2$

Cylinder rotating around a central axis

(b) Z axis,
 $J_M = \frac{1}{2}mr^2$

Hence, we could get the $J_{x,M}$ and $J_{y,M}$,

$$J_{x,M} = J_{y,M} = 2\left(\frac{1}{4}mr^2 + \frac{1}{3}mh^2\right) + 2\left(\frac{1}{4}mr^2 + \frac{1}{3}mh^2 + md_m^2\right) \quad (3.30)$$

In Equation (3.30), the terms in the first bracket are for motor 1 and 3, second bracket is used for motor 2 and 4. $J_{z,M}$ is described in Equation (3.31).

$$J_{z,M} = 4\left(\frac{1}{2}mr^2 + md_m^2\right) \quad (3.31)$$

Four motors are rotating around the central axis which is parallel to the Z axis of the body frame. d_m is the distance between the center of the motor and the COG of the UAV. Hence, this distance will be the same for all motors.

Figure 3.8: ESC Graphic.

3.3.8.2 Electronic Speed Controller (ESC) ($J_{x,E}$, $J_{y,E}$, and $J_{z,E}$)

Equations below are used to describe the flat plate rotating around all axes. ESC can be considered as a thin flat plate based on its design.

Flat plate rotating around X, Y, and Z axes

Figure 3.9: All axes, $J_x = \frac{1}{12}mb^2$ $J_y = \frac{1}{12}ma^2$ $J_z = \frac{1}{12}m(a^2 + b^2)$

Hence, we could get the $J_{x,E}$ and $J_{y,E}$,

$$J_{x,E} = J_{y,E} = 2\left(\frac{1}{12}ma^2\right) + 2\left(\frac{1}{12}mb^2 + md_s^2\right) \quad (3.32)$$

In Equation (3.32), the term in the first bracket are for ESC 1 and 3, second bracket is used for ESC 2 and 4. $J_{z,E}$ is described in Equation (3.33).

$$J_{z,E} = 4\left[\frac{1}{12}m(a^2 + b^2) + md_s^2\right] \quad (3.33)$$

Four ESCs are rotating around an axis that is parallel to the Z axis of the quad. d_s is the perpendicular distance between the rotation axis of ESC to COG of the UAV. Hence, this distance will be the same value for all ESCs.

3.3.8.3 Arm ($J_{x,A}$, $J_{y,A}$, and $J_{z,A}$)

Figure 3.10: Arms Graphic.

Equations below are used to describe the cylinder rotating around a central axis and rotating around an end diameter.

Cylinder rotating around a central axis

(a) X and Y axes,

$$J_A = \frac{1}{2}mr^2$$

Cylinder rotating around an end diameter

(b) X, Y and Z axes,

$$J_A = \frac{1}{4}mr^2 + \frac{1}{3}mL^2$$

Hence, we could get the $J_{x,A}$ and $J_{y,A}$,

$$J_{x,A} = J_{y,A} = 2\left(\frac{1}{2}mr^2\right) + 2\left(\frac{1}{4}mr^2 + \frac{1}{3}mL^2 + md_A^2\right) \quad (3.34)$$

In Equation (3.34), the terms in the first bracket are for arm 1 and 3, second bracket is used for arm 2 and 4. $J_{z,A}$ is described in Equation (3.35).

$$J_{z,A} = 4\left(\frac{1}{4}mr^2 + \frac{1}{3}mL^2 + md_A^2\right) \quad (3.35)$$

Four arms are rotating around an end diameter located at a distance of “ d_A ” from the Z axis of the quad, hence parallel axis theorem term is md_A^2 . Since the quadcopter is symmetrical and all arm lengths are the same, so the terms inside bracket are multiplied by 4.

3.3.8.4 Central Hub ($J_{x,H}$, $J_{y,H}$, and $J_{z,H}$)

Figure 3.12: Central Hub Graphic.

Equations below are used to describe the cylinder rotating around a central axis and rotating around an end diameter.

Cylinder rotating around a central axis

(a) X and Y axes,

$$J_H = \frac{1}{4}mr^2 + \frac{1}{12}mH^2$$

Cylinder rotating around an end diameter

(b) Z axis,

$$J_H = \frac{1}{2}mr^2$$

Hence, we could get the $J_{x,H}$ and $J_{y,H}$,

$$J_{x,H} = J_{y,H} = \frac{1}{4}mr^2 + \frac{1}{12}mH^2 \quad (3.36)$$

$J_{z,H}$ is described in Equation (3.37).

$$J_{z,H} = \frac{1}{2}mr^2 \quad (3.37)$$

The central hub is rotating around a central axis that coincides with the z axis of the UAV, therefore the distance term is 0.

3.4 Rotor Thrust Measurement and Flight Data Collection

A high precision, 3-axis load cell, is used to measure the thrust. Since only z-axis force data is required, the motor is mounted perpendicularly to the load cell. The set-up is shown in Figure 3.14. This experiment is mainly used for modeling the rotors' force dynamics. Other data such as scaled servo inputs, currents, motor rotational speeds, etc., are collected from off-the-shelf sensor collection chip e-logger.

(a) Load-cell to Measure Thrust.

(b) Ground Base Software.

(c) E-logger Setup.

Figure 3.14: Experiment Setup Overview.

3.4.1 Experiment Setup

The following assumptions are made:

- Quadcopter is mainly flying in slow and non-aggressive motions.
- Voltage supplied is constant.
- Motor is in good working condition.
- There is a good fit between a balanced propeller and a motor, which means that the spinning torque required from propeller will not let motor overheat.
- Propellers are made of carbon fiber whose pitch is relatively constant during rotation.

Figure 3.15 shows a glimpse of the data collected from the experiments conducted indoor where there is negligible wind. The experiment is replicated thrice in both of the clockwise and counter-clockwise condition so as to obtain an average reading to minimize errors. Data logging starts immediately when the motor reaches a speed of more than 200 rpm, which is a slow spinning rate because the data logging process is

Using the Data Analyzer program which is shown in Figure 3.16, all coefficients based on the characteristics of motors used, could be obtained. The data used are time, throttle, RPM, b, h, torque and L.

Figure 3.16: Data analyzer used to calculate all coefficients.

In the Figure 3.16, b is the length of test rig from the pin joint to the scale contact, h is the length of test rig from the pin joint to the center motor axis and L is the length of the rig arm from center of the motor to the scale contact.

Hence, all coefficients are generated and shown in figures below.

The equation used for conversion coefficient from throttle% to RPM and b is,

$$\bar{\omega}_{ss} = (Throttle\%)c_R + b \quad (3.15)$$

The equation used for torque coefficient is,

$$Q = c_Q \bar{\omega}^2 \quad (3.12)$$

The equation used for thrust coefficient is,

$$T = c_T \bar{\omega}^2 \quad (3.11)$$

Figure 3.17: Linear Fit of RPM vs. Throttle.

Figure 3.18: Linear Fit of Torque vs. RPM^2 .

Figure 3.19: Linear Fit of Thrust vs. RPM^2 .

All the parameters that are identified by the static experiments are listed in Table 2.2.

Description	Parameter	Value/Unit
Conversion coefficient from throttle% to RPM	C_R	84.6461/ RPM/%
Linear regression relationship from throttle% to RPM	b	848.9199/RPM
Torque coefficient	C_Q	5.0112e-09/ $N \times m/RPM^2$
Rotor thrust lift coefficient	C_T	1.9729e-07/ N/RPM^2

Table 3.1: Numerical value of identified parameters from ground experiment.

3.5 Quadcopter Modeling

After all coefficients are obtained, moments of inertia in all axes can then be calculated using MatLab based on the platform of the UAV. The platform used is DJI

F450 quadcopter, MT2212 KV1199 motors and 10×3.3 CF T-Motor propellers, a 11.1V 5300mAh Li-po battery. Full payload UAV is shown in Figure 3.20.

Figure 3.20: UAV Platform.

All measurements of the platform are shown in Table 3.2.

After all measurement data and coefficients are keyed into the program, as shown in Figure 3.21, the mass moment of inertia of all axes are calculated based on the calculation method (Equation (3.29)) introduced in Subsection 3.3.8 and shown in Table 3.3. The moments of inertia and all coefficients calculated earlier then are saved as "X" configuration in the matrix form, which will be used at a later stage.

Parameter	Value	Description
<u>Motors</u>		
m[g]	76	Mass of one motor (with propeller)
dm[cm]	22.5	Distance from the center of mass of the motor to the COG of the UAV
h[cm]	2.85	Height of the motor above the arm
r[cm]	1.375	Radius of the motor
<u>ESC</u>		
m[g]	15	Mass of one ESC
a[cm]	2.15	Width of the ESC
b[cm]	5.24	Length of the ESC
ds[cm]	13	Distance from the center of mass of ESC to the COG of the UAV
<u>Central Hub</u>		
m[g]	1092.8	Mass of the central hub include full payload
r[cm]	6.5	Radius of the central hub
H[cm]	4	Height of the central hub
<u>Arms</u>		
m[g]	5.3	Mass of one arm
r[cm]	1.5	Radius of the arm
L[cm]	18.5	Length of an arm to the point attached to hub
da[cm]	3.2	Remain length of an arm in hub

Table 3.2: Parameter of UAV platform Dimension

Figure 3.21: Mass Moment of Inertia of UAV.

Description	Parameter	Value/Unit
Moment of inertia in X axis	J_X	$0.010277 / kg \times m^2$
Moment of inertia in Y axis	J_Y	$0.010277 / kg \times m^2$
Moment of inertia in Z axis	J_Z	$0.020211 / kg \times m^2$

Table 3.3: Mass Moment of Inertia of Three Axes.

The equations used are discussed in Section 3.3.8. The general equation is,

$$J_{parallel-axis} = J_{COM} + mr^2 \quad (3.29)$$

The moment of inertia in x-axis is,

$$J_x = J_{x,M} + J_{x,E} + J_{x,A} + J_{x,H} \quad (3.38)$$

$$J_{x,M} = J_{y,M} = 2\left(\frac{1}{4}mr^2 + \frac{1}{3}mh^2\right) + 2\left(\frac{1}{4}mr^2 + \frac{1}{3}mh^2 + md_m^2\right) \quad (3.30)$$

$$J_{x,E} = J_{y,E} = 2\left(\frac{1}{12}ma^2\right) + 2\left(\frac{1}{12}mb^2 + md_s^2\right) \quad (3.32)$$

$$J_{x,A} = J_{y,A} = 2\left(\frac{1}{2}mr^2\right) + 2\left(\frac{1}{4}mr^2 + \frac{1}{3}mL^2 + md_A^2\right) \quad (3.34)$$

$$J_{x,H} = J_{y,H} = \frac{1}{4}mr^2 + \frac{1}{12}mH^2 \quad (3.36)$$

where $J_{x,M}$ is the moment of inertia of motor part in x-axis, $J_{x,E}$ is the moment of inertia of ESC part in x-axis, $J_{x,A}$ is the moment of inertia of arm part in x-axis and $J_{x,H}$ is the moment of inertia of central hub in x-axis. Moment of inertia in y-axis and z-axis use the same calculation method.

The equations of final model that will be used in control system are,

1. Relationship between gyroscopic torques and pitch, gyroscopic torques and roll action of UAV. J_M is the moment of inertia of UAV, P_m and Q_m are the rolling and pitching rates, and the speed of each motor is $\bar{\omega}_i$.

$$\tau_{\theta_{gyro}} = J_M P_m \left(\frac{\pi}{30}\right) (-\bar{\omega}_1 + \bar{\omega}_2 - \bar{\omega}_3 + \bar{\omega}_4) \quad (3.16)$$

$$\tau_{\phi_{gyro}} = J_M Q_m \left(\frac{\pi}{30}\right) (\bar{\omega}_1 - \bar{\omega}_2 + \bar{\omega}_3 - \bar{\omega}_4) \quad (3.17)$$

2. The total force acting on UAV body, where C_T is the rotor thrust lift coefficient, is given by.

$$F_{\mathcal{F}_b} = \begin{bmatrix} 0 \\ 0 \\ c_T(\bar{\omega}_1^2 + \bar{\omega}_2^2 + \bar{\omega}_3^2 + \bar{\omega}_4^2) \end{bmatrix} \quad (3.20)$$

3. The angular velocity is related to the motor's inertia, moments, and angular velocity.

$$\dot{\omega}_{\mathcal{F}_b} = (J)^{-1} [M_{\mathcal{F}_b} - \dot{\Omega}_{\mathcal{F}_b} J \omega_{\mathcal{F}_b}] = \begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} \quad (3.21)$$

4. The rotational velocity matrix based on the inertial matrix and moment matrix is given by.

$$\Omega_{\mathcal{F}_b} = \begin{bmatrix} 0 & -\dot{\psi} & \dot{\theta} \\ \dot{\psi} & 0 & -\dot{\phi} \\ -\dot{\theta} & \dot{\phi} & 0 \end{bmatrix} \quad (3.22)$$

5. The velocity state describes the relationship between the acceleration of the COG of the UAV and forces and torques acting on the body of frame.

$$\dot{v}_{\mathcal{F}_b} = \left(\frac{1}{M}\right)F_{\mathcal{F}_b} + g_{\mathcal{F}_b} - \Omega_{\mathcal{F}_b}\omega_{\mathcal{F}_b} = \begin{bmatrix} \dot{U} \\ \dot{V} \\ \dot{W} \end{bmatrix} \quad (3.26)$$

6. The position state establishes the relationship between the linear velocity of the COG of the UAV in the inertial frame.

$$\dot{P}_{\mathcal{F}_i} = \begin{bmatrix} R_{\mathcal{F}_b}^{\mathcal{F}_i} \end{bmatrix} v_{\mathcal{F}_b} = \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} \quad (3.28)$$

3.6 Conclusion

In this research, the conversion coefficients from throttle percentage commands to RPM values, torque coefficients and thrust coefficients of the UAV, are derived. The mass moments of inertia in all axes are calculated based on all parts of UAV after obtaining all the coefficients in such a way that all the necessary parts of UAV are measured. The calculation results of motor test data and mass moments of inertia shown in Figure 3.21 are saved in a matrix and then used in simulation Quadcopter Control Mixing function. The Equations (3.15), (3.16), (3.17), (3.18), (3.19), (3.20), (3.21), (3.22), (3.24), (3.26) and Equation (3.28) are directly used in simulink Quadcopter Dynamic S-function for simulations and testings (Chapter 4).

Quadcopter Control & Simulation

4.1 Introduction

UAV control safety could be achieved with a flight controller. This chapter details the development of a flight controller and simulation procedures for the quadcopter. The quadcopter modeling matrix containing motor test data and moments of inertia of all the axes is obtained in Chapter 3. Based on the dynamics and kinematics derived in the previous chapter, a flight controller is designed. Furthermore, simulations are conducted to examine modeling results and the designed controller.

The control system in the UAV is a proportional-integral-derivative controller (PID controller). It is commonly used in industry control systems since it only relies on the measurement process variables and it is quite popular in UAV autopilot control systems. The current control action calculates an error value as the different baseline between a desired set-point and a measured process variable. PID controller minimizes the error over time by adjusting a control variable. Defining $u(t)$ as the controller output, the form of the PID algorithm is shown in Equation (4.1).

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt} \quad (4.1)$$

where K_p , K_i , K_d are all non-negative coefficients for the proportional, integral, and derivative terms.

Since PID controller only relies on the measurement process variables, it can handle a lot of specific processes required by tuning the above mentioned three parameters such as the UAV's control systems [49, 50].

4.2 Control & Simulation

The open source MatLab simulink program [51], is modified to fit into the system. The control system is modified and the following safety system is further improved based on the following criteria: stability of the UAV, control performance and difficulty of implementation in an autopilot board.

The challenges of the control and safety system design are:

- Path generation is necessary to give a clear simulation result and check the preciseness of the dynamic control of the quadcopter.
- Animation is needed to give a better view of safety system's performance.

In the existing research, a lot of works have been done on quadcopter modeling, simulation and control, but less has been done for external affections, such as [52, 53, 54]. Hence, in the simulation and control of the UAV, wind effects are considered based on the average wind speed in Singapore. This section details the simulation results of the control.

4.2.1 Control of Quadcopter

Control system of UAV follows the procedure shown in Figure 4.1.

Figure 4.1: Control Structure Scheme.

Path planning command (desired position command), current position and attitude of the quadcopter are fed into outer loop control which is the position controller using proportional-derivative controller (PD controller) to get an attitude command. The inner loop control, on the other hand, will then get the attitude command, current position and attitude of the quadcopter as its inputs in order to correct the roll, pitch, yaw angles and altitude commands for the next stage. Control mixing block in the simulink will generate the four motors' commands based on the quadcopter configuration which is a "X" configuration in our experiment. The next block is the dynamics block with the addition of disturbances such as wind effects. And this block also considers the testbed dynamic modeling in order to get precise control outputs. The whole process is a closed-loop [55] feedback control which corrects the attitude of the quadcopter in real time.

Remarks:

- Dynamics modeling results are used for the quadcopter's estimator system.
- Controllers in outer loop control and inner loop control are developed based on the reference [32].

4.2.2 Simulation

This subsection presents the simulation results, based on different PID controllers. The desired path is a diamond shape with an area size of $100m \times 100m$. The yaw angle is fixed to be 0 degree which is the same heading as the autopilot board's pre-set yaw angle. This path is used as the input to the position controller as a time-series based position command. The simulink is shown in Figure 4.2.

Figure 4.2: Simulation Structure.

The initial condition is preloaded into the structure where quadcopter is booted up. Hence components in the initial matrix are all 0. This initial condition will be considered in DynamicSFunction in the Quadcopter Dynamics block first. The initial condition used in simulink could be described in Equation (4.2).

$$init = [\dot{\phi}, \dot{\theta}, \dot{\psi}, \phi, \theta, \psi, U, V, W, X, Y, Z] \quad (4.2)$$

where $\dot{\phi}$, $\dot{\theta}$ and $\dot{\psi}$ represent the angular velocity, ϕ , θ , ψ represent the angle, U, V and W are velocity, X, Y and Z are the current position of quadcopter. All variables

are relative to the earth frame.

Before the simulation begins, initial condition matrix, path planning matrix and modeling matrix have to be loaded in the workspace.

The entire control structure in the quadcopter is a closed-loop feed back control. Hence, with the path command inputs and outputs of the dynamic block, roll, pitch angles are calculated again according to the current velocity and desired velocity in X-Y axes. The outputs of the position controller are the roll, pitch command and altitude command. Linear velocity constraints of the quadcopter are:

$$|v| \leq \begin{bmatrix} V_{xmax} & V_{ymax} & V_{zmax} \end{bmatrix}^T = \begin{bmatrix} 5 & 5 & 6 \end{bmatrix}^T m/s$$

Altitude command is the input to the third block which is the attitude block together with the outputs of the dynamic block. Output of the dynamic block is in matrix form shown in Equation (4.3). This is where the PID controller enters the calculation to correct angles of the quadcopter shown in Figure 4.3. The maximum angles in the three axes are limited within 60° .

$$Output = \begin{bmatrix} \dot{\phi}, \dot{\theta}, \dot{\psi}, \phi, \theta, \psi, U, V, W, X, Y, Z \end{bmatrix} \quad (4.3)$$

where $\dot{\phi}$, $\dot{\theta}$ and $\dot{\psi}$ represent the angular velocities, ϕ , θ , ψ represent the angles, U, V and W are velocities, X, Y and Z are the current position of the quadcopter. All variables are relative to the earth frame and current status of the quadcopter.

Figure 4.3: PID Controller used to Correct the Angle and Altitude Output.

In the Figure 4.3, the state inputs P, Q and R are the same as $\dot{\phi}$, $\dot{\theta}$ and $\dot{\psi}$ representing the angular velocities, Phi, Theta and Psi are the same as ϕ , θ , ψ representing the angles. Detailed controllers design under the mask are shown in Figure 4.4, 4.5, and 4.6.

Figure 4.4: Roll Controller used to Correct the Angle.

Figure 4.5: Pitch Controller used to Correct the Angle.

Figure 4.6: Yaw Controller used to Correct the Angle.

In order to obtain the best PID controller gains, a simulation is conducted and shown in Figure 4.7, 4.8, and 4.9. Since the simulation platform is symmetrical to body x and y axes, the controller gains for roll and pitch are the same. Firstly, I and D gains are set to zero, P gain is increased until the output is with a steady oscillation. Secondly, D gain is increased until the system is critically damped. Then repeat the last two steps until no oscillation is observed. We record P and D gains respectively. Lastly, I gain is increased until the response achieves the desired performance. In this research, the ideal properties of the UAV's controller include zero steady state error and relatively fast responses. As shown in Figure 4.7, $P = 1$

is abandoned because it has a large overshoot of angle. Conversely, $P = 3$ is not adopted due to a slower response than the condition in $P = 2$. In Figure 4.8, $I = 1.1$ has the better reference tracking than $I = 1.5$ and a faster transient response than $I = 0.5$. Similarly, in D gain comparison simulation, $D = 1.2$ is chosen using the same criteria. Eventually, the PID controller gains are finalized.

- P gain: 2
- I gain: 1.1
- D gain: 1.2

Figure 4.7: Tuning P Gain.

Figure 4.8: Tuning I Gain.

Figure 4.9: Tuning D Gain.

All correction values of the angles for the respective three axes and altitude correction data are calculated to generate the commands for each one of the four motors,

which is based on the quadcopter's configuration shown in the fourth block in Figure 4.2. A “X” configuration is chosen in our case which is shown in Figure 4.10.

Figure 4.10: Configuration Chosen in Control Mixing Block.

The functionalities of the dynamics blocks are:

- Motor commands are converted from percentage of throttle to RPM based on the motor dynamic modeling results.
- Disturbances (Wind effect) are considered to get the current status of the quadcopter. Wind in Singapore is generally light, with the average speed of $2.5m/s$ unless extreme conditions happen [56].

The results of the control with no disturbance are shown in Figure 4.11. The dot line indicates the desired value and the solid line is the actual value and the results show that with the help of dynamic modeling, the quadcopter can be controlled better.

Figure 4.11: Quadcopter Output Status.

The dot lines in three Angles figures in Figure 4.11 are the desired angles, the solid lines are the actual angles during the flight. The dot lines in three Positions figures are the desired positions in each axis and the solid lines are simulated flight performance of the quadcopter. The quadcopter follows the desired angles and positions.

Figure 4.12: Motor Throttle Command vs. RPM Value.

The work of converting throttle percentage commands to RPM, which is inside the dynamics block, can be seen in Figure 4.12. The blue line shows the motor's throttle command in percentage form while the green color line reflects the motor's speed in RPM. During the UAV's take-off period, more lift is generated, hence RPM of the motor is larger than moving period.

With the help of animation, the performance of the UAV could be clearly seen in Figure 4.13 and 4.14. The animation has the ability to check the UAV's performances and the attitude changes during the flights in three axes.

In default view, the UAV takes-off from the ground and follows the desired path during the flight.

Figure 4.13: Quadcopter Performance in Default View.

In X-Y view, the desired diamond path could be examined. The quadcopter has slight drifts during the flight.

Figure 4.14: Quadcopter Performance in X-Y View.

The improved simulink with the added in wind disturbance function could check the UAV's performance influenced by the wind effect. With the wind disturbances

at a velocity of 2.5m/s in both X and Y axes, the results are different compared to the previous ones. They are shown in Figure 4.15 and Figure 4.16.

Figure 4.15: Quadcopter Output Status when Wind Effect Is Added.

Figure 4.16: Motor Throttle Command vs. RPM Value when Wind Effect Is Added.

In Figure 4.15, the errors between the actual angles and the desired angles, the actual positions during flight and the desired position are larger than those when there is no wind (Figure 4.11).

In Figure 4.16, the RPM value is displayed in orange color. The motor throttle command is kept at 40% during flight when wind effect is added, and speeds of four motors are around 4500RPM. In the previous discussion, the motor throttle command is less than 40% during flight without wind effect added, and speeds of four motors are around 4000RPM. Hence, the wind function in simulink is useful to double check the quadcopter's performance.

The performance overview is shown in Figure 4.17 and Figure 4.18. From figures, it shows that the control results are affected by disturbances. The error between the desired position and the actual position is larger than the error when there is no wind. The overall performance is affected as well.

Figure 4.17: Quadcopter Performance in Default View with Wind Effect.

Figure 4.18: Quadcopter Performance in X-Y View with Wind Effect.

4.3 Conclusion

In this chapter, a simulation study is carried out to set proper PID controller gains and test the effects of external disturbances such as wind. Linking to the modeling of quadcopter in Chapter 3, the simulation is capable of estimating the UAV's status. Closed loop feedback control is used in the simulation process as well as the autopilot board. The PD controller is used for outer loop position control and PID controller is used for inner loop attitude control, in such way that we are able to tune these controllers flexibly during real flights.

Ground Control Station Development

A functional ground control station is necessary in order to reduce the safety risks caused by human factors. The workload of the UAV operators will grow exponentially with the increasing numbers of UAVs to be controlled. Ground control station (GCS) assists the operator in the challenging tasks of managing a system with multiple UAVs, thus reducing the dangers caused by human errors[43].

The research considers the safety issues of the UAV during flight. Hence, a ground control station with the capabilities to monitor the UAV in real time is necessary. In order to assist the ground operator in controlling multiple UAVs, a customized ground control station with the capability of displaying the status of multiple UAVs and sending commands according to the specified missions, is developed. MicroHard N2420 2.4GHz device is used to establish the communication links between the ground control station and the UAVs.

5.1 Introduction

To monitor the flight status of the UAV in real time, all necessary data should be sent back to the Ground Control Station(GCS), which is a rugged and semi-waterproof

Panasonic Toughbook. To ensure that data are secured, a 128-bit Advanced Encryption Standard(AES)[57] data encryption is added into the data communication link. The data set is defined in both the air and the ground side of the communication modules. In addition, historical flight trajectories of all the UAVs are displayed on the 2D GoogleMap as shown in Figure 5.2. The GCS operator can have an overview of the relative positions of all the UAVs at the same time via the external LCD monitor. A customized Micro Air Vehicle Link (Mavlink) protocol is used as the communication protocol between the GCS and UAVs, and between the high-level board and autopilot board. The advantages of the MAVLink are light weight and its header-only message marshaling style. The original MAVLINK protocol[58] is designed and implemented in early 2009 by Lorenz Meier under GNU Lesser General Public License (LGPL).

Figure 5.1: GCS Field Setup.

Figure 5.2: GCS Map Display.

5.2 Design Details

When a UAV performs fully automated flights, the remote-control command is sent from the GCS to the UAV in order to perform such actions as taking-off, path start, etc. The architecture of the communication link between the UAV and the GCS can be seen in Figure 5.3.

Figure 5.3: Communication-link Architecture.

The design of the GCS follows the traditional structure of the GCS software in Figure 5.4. The flow of data and information has been introduced in this structure. The dedicated GCS has the ability to support various communication methods, such as WiFi, serial-port, UART at a different baud rate, etc.

Figure 5.4: Traditional structure of GCS design.

5.3 Functional Abilities of Ground Control Station

The GCS for multiple UAVs has certain functionalities is discussed below.

5.3.1 Air Vehicle Control

This function is used to monitor and effectively control UAVs during the whole flight process. It allows the GCS with the capability to choose a suitable data link to communicate with the UAV and display the localisation data of each UAV. The numbers of UAVs could be set and modified based on the situation.

1. Main Localization Data

GPS data with projected X, Y, Z coordinates and satellite numbers are shown in each UAV window. UAV's heading based on high level on-board compass is shown on the window as well.

2. Battery Voltage

Battery Voltage is displayed on the screen. If the voltage is lower than the pre-set threshold value, the on-board system will take fail-safe action for safety landing.

3. High Level Flight Computer Modes

High Level Flight Computer Modes are Take-off, Hold, Path, Land, Return Home, and Idle. These modes are mainly used for formation path performances.

4. Low Level Autopilot Modes

These modes include arm, geo-fence, fail-safe, high level faulty flag, battery condition, GPS, and gyroscopic reading flags. These information are integrated and encoded into 16-bit integer in order to decrease the data payload.

5. Version Control

This mode shows low and high level computer firmware git-commit numbers. A tag is attached to the commit numbers when the repository is stable. GCS will detect it if firmware is not in a clean status, i.e. the firmware has some changes after the stable version.

The GCS provides the operator with a quick overview of the UAV statuses via the color of the boxes. The color differences could be seen in Figure 5.5.

- Grey - the UAV is not in operation.
- Yellow - the UAV is in operation.
- Red - Warning indicator that a safe mode has been triggered upon detecting errors, such as a weak global positioning system (GPS) signal, low battery voltage, large amount of tracking errors, etc.

Details of the UAV's status are displayed in each box. And this could be customized based on the mission's requirements.

- UAV ID
- Position x, y, z (in meters)
- Heading (in degree)
- Number of GPS satellites

- GPS time: hour | minutes | second
- On-board battery reading
- Low level auto pilot mode: normal | return home
- High level auto pilot mode: manual | auto
- Error message

Figure 5.5: GCS Real-time Health Monitoring Window with UAVs connected.

The path of each respective UAV is displayed on the 2D map as seen in Figure 5.6 and Figure 5.7. All the geo-fence boundaries are displayed in the map to provide the operator with visual cues regarding the safety status of each UAV. Heading

information of each UAV is indicated using a small red dot. Each UAV shape is indicated using a yellow circle.

Figure 5.6: UAVs 100 Formation Shape in Air.

Figure 5.7: UAVs 100 Formation Shape Displayed on GCS.

5.3.2 Post-flight Analysis

The GCS supports post flight analysis such as the coefficients of estimated position versus current position, estimated velocity versus current velocity, etc. These data could only be collected from the log files in the SD card of each UAV.

5.4 Conclusion

The ground control station is developed to display and monitor multiple UAVs' data and map information. It is implemented with static soft geo-fencing system and used during the multiple UAVs formation shows to be introduced in Chapter 6. Customized Mavlink protocol is used as the communication protocol between the GCS and UAVs. The possibilities of safety failures are decreased with the GCS in our formation flights. The post flight analysis will be improved to enable real-time analysis without using an SD card in the future.

UAV Geo-fence Safety System Implementation

6.1 Introduction

A safety system which is meant to be implemented for the multiple UAVs formation shows, aims to protect civilians and properties. There are two parts of safety consideration:

- UAV Safety implementation:

We focus on safety features such as UAV airworthiness and stability. We assume that the UAV has accomplished its safety inspection such as checks on wire connection, sensor calibration etc. Different safety action modes are developed based on several safety triggering conditions.

- Area safety implementation:

Several geo-fence[\[59\]](#) polygons are predefined to work with UAV safety implementation. These geo-fences are designed to protect necessary assets and people when the UAV performance is in progress. Three layers of geo-fencing polygons are designed to maximize the safe possibilities. They are soft geo-fence, hard geo-fence and maximum safety boundary.

To operate UAV outdoor, we must not influence main civil and military air traffic. It is necessary to report to local air traffic management authority and design a safety geographical boundary. Another safety concern is UAV reliability, this includes avionics stability, program robustness, anti-collision reliability, and etc. A fixed geographic volume is used to examine the performance of the design of such a safety system. The designed procedure is based on the risk assessment form as shown in Appendix A.2. The detailed fault tree analysis is illustrated in Figure 6.1.

Figure 6.1: Fault Tree Analysis Diagram.

6.2 Area Safety Design

A program based on pre-designed boundary shape is used to generate three boundaries automatically. When the GPS coordinates in radians of the performance centre are provided and defined as $i = 1$, the program can calculate the final geo-fence

boundary coordinates to form a closed contour by using its bearing and distance to next boundary point. The equations used are Eq. (6.1), (6.2), (6.3) [60].

$$d_{div_r} = d_i / R \quad (6.1)$$

$$lat(i+1) = \arccos(\sin(lat(i)) \times \cos(d_{div_r}) + \cos(lat(i)) \times \sin(d_{div_r}) \times \cos(b_i)) \quad (6.2)$$

$$lng(i+1) = lng(i) + \arctan 2(\sin(b) \times \sin(d_{div_r}) \times \cos(lat(i)), \cos(d_{div_r}) - \sin(lat(i)) \times \sin(lat(i+1))) \quad (6.3)$$

where i is numeric sequence of boundary coordinates, d_i is the distance to the next $(i+1)$ geo-fence point, R is the radius of earth, b_i is the bearing to the next geo-fence point (relative to North), lat and lng are the latitude and longitude of the point. Figure 6.2 illustrates the equations above. A is the i point, B is the $i+1$ point. With the distance d_i and bearing b_i from point A to point B, the coordinates of A, and the radius of earth, coordinates of B can be calculated.

Figure 6.2: Calculation Illustration.

6.2.1 Performance Zone

The performance zone is shown as the blue box in the Figure 6.3 and its dimension is $80\text{m (L)} \times 80\text{m (W)} \times 60\text{m (H)}$. This zone is the normal UAV operating zone during performance. The 60 meters height is based on the rules and regulations of Civil Aviation Authority of Singapore (CAAS).

Figure 6.3: Geo-fence Design.

6.2.2 Soft Geo-fence

The soft geo-fence boundary, shown as the green box in the Figure 6.3, allows the formation to operate smoothly without interruption when failures occur. It is also served as the first layer of protection to contain the performance within the area boundary. It has a fixed dimension of: $100\text{m (W)} \times 100\text{m (L)} \times 70\text{m (H)}$. If the UAV low level autopilot GPS detects the breaching of the boundary, return to safety exit

mode will be triggered as shown in Figure 6.1. UAV will return to the launch site following the predefined unique route.

To calculate the distance between the performance zone and soft geo-fence (20m), we need to know what will happen when the UAV breaches the soft geo-fence boundary. The program handles geo-fence calculations at 10 Hz. And the geo-fence algorithm confirms the triggering event after 5 GPS counts of breaching. That means, Return Home mode will be triggered $0.1s \times 5 = 0.5s$ after the UAV crosses the soft geo-fence. With the 2m/s UAV ground speed, and maximum 6m/s wind speed (same direction with UAV), and the mass of UAV is 4kg, also considering the surface area of the UAV, the induced force caused by the wind is estimated to be 2N to 4N. This will give the worst case constant acceleration of $a = F/m = 4N/4kg = 1m/s^2$. Hence, we can estimate the maximum velocity of the UAV after 0.5s with the velocity equation.

$$V = V_0 + at = 2 + 1 \times 0.5 = 2.5m/s \quad (6.4)$$

Assuming the UAV is flying at 2.5m/s, the UAV would have traveled $2.5m/s \times 0.5s = 1.25m$ during this period. After which the UAV will attempt to decelerate at a rate of $1.5m/s^2$ to reach a hovering position. Based on the Equation (6.5), the UAV will take 3.75s to stop.

$$t = (V_{final} - V_{initial}) \times a = (0 - 2.5) \times (-1.5) = 3.75s \quad (6.5)$$

The distance traveled during this deceleration period is:

$$d = (V_{final}^2 - V_{initial}^2)/2a = (0 - 2.5^2)/(2 \times 1.5) = 2.083m \quad (6.6)$$

The total distance that the UAV has traveled is:

Total distance = Distance traveled prior to detecting soft geo-fence + Distance traveled during deceleration = $1.25m + 2.083m = 3.33m$. Therefore, the soft geo-fence boundary needs to be at least 3.33m away from the hard geo-fence boundary. The distance between the soft geo-fence and the hard geo-fence is $((120 - 100))/2 = 10m > 3.33m$.

6.2.3 Hard Geo-fence

To ensure that the UAVs never cross the safety boundary, a hard geo-fence is designed. The UAV flight terminates upon crossing this geo-fence. This hard geo-fence is defined as 120m (W) $\times$ 120m (L) $\times$ 80m (H). A simulation has been conducted for verifying the geo-fence setup. Software-in-the-loop for PX4 firmware is an open source simulation tool to verify the performance of the aerial vehicle. As shown in the Figure 6.4, top left is the message window for user to observe UAV status such as Arm, Height, Mode, etc. The tool also generates a real location map. The window shown in the bottom left is the command terminal where command is sent.

Figure 6.4: Hard Geo-fence Verification Simulation.

Our experiment is designed as below. Firstly, set wind speed as 6m/s, and wind direction is the same as UAV velocity pointing south. The parameters of wind speed and direction are shown in the terminal as SIM_WIND_SPD and SIM_WIND_DIR

respectively. Simulation wind speed frame is the same as the UAV body frame. Hence 0 *degree* is the forward direction of the UAV in the map. Secondly, UAV takes off to 80 *meters* above the ground. When UAV reaches 80 *meters*, velocity command is set to 2*m/s* forward direction along the UAV body frame. The UAV is initialized as pointing south in earth frame. Thirdly, cut throttle by setting UAV mode to STABLIZED. UAV will free fall against with air resistance. The acceleration of air resistance is defined as linearized relationship to UAV speed.

$$a_{airresistance} = -\frac{v_{current}}{v_{terminal}}g \quad (6.7)$$

where $v_{current}$ is current UAV velocity, g is gravity acceleration $9.81m/s^2$, $v_{terminal}$ is the terminal velocity defined as

$$v_{terminal} = \sqrt{\frac{mg}{k}} = 15m/s \quad (6.8)$$

with $k = 0.174$ defined as a constant in the simulation system. Figure 6.5 shows the algorithm implemented in simulation program.

```

194  if (terminal_velocity > 0) {
195      // air resistance
196      Vector3f air_resistance = -aircraft.get_velocity_air_ef() * (GRAVITY_MSS/
terminal_velocity);
197      body_accel += aircraft.get_dcm().transposed() * air_resistance;
198  }
```

Figure 6.5: Air Resistance Definition in Simulation.

After conducting the experiment, the result of experiment is shown in Figure 6.6.

Figure 6.6: Free-fall Projection Measurement.

By measuring the distance between cut-throttle-point and crash-point, the result of free-fall with wind speed of $6m/s$ is verified to be $24.5m < 25m$. Our safety boundary to the hard geo-fence boundary is $(170 - 120)/2 = 25m$.

6.2.4 Safety Boundary

In the flying area, safety boundary dimension is defined as $170m (L) \times 170m (W) \times 80m (H)$ geo-cubic. This is the largest area which could be covered by the UAV shown in red color in Figure 6.3.

6.2.5 Return Home Route Calculation

When the respective UAV has its Return Home Action triggered, its aerial system will calculate the shortest evacuation path to the performance region boundary, from where it currently is. Every evacuation path is a unique exit route. Each aerial system requires its own GPS coordinates in order to calculate the shortest evacuation path. The respective UAV will then enter its safety return home path once it reaches the performance boundary. A reasonable traffic management is considered when UAV enters the safety return home route.

A centralized control method is adopted for which all control messages are sent out by the GCS, thus there is no inter-communication between UAVs. It is difficult to design an on-line safety return home route with collision avoidance feature.

Figure 6.7: Activate Path State Machine.

A state machine, illustrating the specific path used during performance, is shown in Figure 6.7. RH, SF, HF, SL mean return home flag, soft geo-fencing flag, hard geo-fencing flag, and safety landing flag respectively. When Return Home is activated, the divergence path will be entered first, whereby the UAV will fly in a diverging movement from performance center to the boundary as fast as possible. The next step depends on the respective UAV's GPS location. If the UAV has its GPS location out of the soft geo-fence, return home path will be loaded into mission. If the hard

geo-fence is breached ($HF = 1$), flight termination will be activated. If GPS signal is lost ($SL = 1$), the return home route operation will be meaningless. The safety landing path is entered immediately.

6.3 Conclusion

The safety system is tested during the 16 UAVs flight demonstrations: Future of Us at Gardens by the Bay. The results show that the system significantly decreases the flight failures. With the static geo-fencing system implemented, there is no more UAV flying out of the safety boundary. Table 6.1 shows a few incidents that the safety system helped, thus increasing the overall safety level of the multi-UAVs systems. It allows us to debug after a fail-safe action is triggered and track the log files to know what exactly happened.

Date	Safety Action Triggered	Condition	Reasons
15 Apr 2015	Return Home	High Level Autopilot Faulty Flag	High Level Autopilot Lockup
6 May 2015	Flight Termination	Low Level Autopilot Lockup	Power Interface Module Lost Power
20 Oct 2015	Safety Landing	Attitude $> \pm 45^\circ$	Strong Wind
2 Nov 2015	Return Home	Soft Geo-fence Violence	Wrong Formation Path Design
23 Dec 2015	Safety Landing	Battery Critical Low	Battery Broken
7 Jan 2016	Return Home	RC Loss	RC Antenna Improper Installation

Table 6.1: Flight Results with Safety System Implemented.

The safety system has to be improved by incorporating a dynamic soft geo-fencing system to maximize the possibilities of UAVs' survival.

Development of Dynamic Geo-fencing Control

7.1 Introduction

A comprehensive static geo-fencing system mentioned in Chapter 6 has been developed for the 16 quadcopters formation shows. The system is built into the autopilot board and is to be initialized whenever the UAV is powered on. Safety action will be triggered if certain condition is met which is detailed in Chapter 6.1. Since all the geo-fences are fixed and the system has no ability to conduct the future state prediction of the UAV, there may be a possibility that the UAV will be lost if it crosses the boundary because of wind or other issues. Hence, the dynamic soft geo-fencing is a new concept in order to maximize the possibility to get UAV back before it breaches the hard geo-fence.

The proposed advanced dynamic geo-fencing scheme is a hierarchical method consisting of a model predictive controller (MPC). This controller has a top-layer and a bottom-layer feedback loop. It is developed as a second outer loop for safety actions, and first outer loop is the position loop. This is a further development of UAV control introduced in Chapter 4.

MPC control has been used in the industries, such as chemical plants and oil re-

fineries since the 1980s, power system balancing models [61]. It relies on dynamic model of the process and the advantage is that it allows the current time slot to be optimized, while keeping future time slots in account. MPC has the ability to anticipate future events. In this chapter, the dynamic geo-fencing is discussed in detail.

This is the first time that safety system in autopilot board is proposed in UAV research area. The framework is designed such that it is convenient for implementation in autopilot board.

7.2 Algorithms

As discussed in Chapter 3, the position state Equation (3.28) is used to establish the linear velocity of the COG of the UAV.

$$\dot{P}_{\mathcal{F}_i} = \left[R_{\mathcal{F}_b}^{\mathcal{F}_i} \right] v_{\mathcal{F}_b} = \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} \quad (7.1)$$

Some conditions are listed before the derivations of algorithms.

1. UAV is moving slowly such that the altitude is not changed when UAV is flying in horizontal plane.
2. The shape of UAV is symmetrical in both x and y axes.

Therefore, an assumption is made that the UAV is able to be treated as point mass in this research.

MPC based controller is to be designed in order to deal with the whole process of the dynamic soft geo-fencing system. UAV is assumed to fly in X-Y surface only through

a sequence of points defined as $(X(t), Y(t)), t \geq 0$. There are several assumptions:

- UAV exactly follows the predefined path $(X(t), Y(t)), t \geq 0$.
- UAV acceleration is limited by pitch, roll rates in the system which are defined as a_{max} .

Hence, according to the assumptions, the system has the following constraint,

$$\sqrt{\dot{v}_x^2 + \dot{v}_y^2} \leq a_{max} \quad (7.2)$$

where v_x and v_y are the velocities to the respected axis.

$$X + Y < \frac{\sqrt{2}}{2} a_{max} t_o^2 - (v_x + v_y) t_o \quad (7.3)$$

where X, Y are the current position of UAV. t_o is a set time which is decided by the radius of the soft geo-fence boundary.

The MPC program will be activated when the predicted location of UAV reaches or crosses the hard geo-fence. The UAV model is discretized with the sampling time t ($t = t_o$). Assume that a desired set-point is provided over a finite time horizon of N steps at each sampling time t , with $N \in \mathbb{Z}^+$, in terms of a sequence of set-points on the fixed hard geo-fence boundary (X_i^s, Y_i^s) and $Y_i^s = -X_i^s$, with $i = t+1, \dots, t+N$. Consider the cost function shown in Equation 7.4.

$$J = \sum_{i=t+1}^{t+N} \alpha_1 [(X_i - X_i^s)^2 + (Y_i - Y_i^s)^2] + \sum_{i=t}^{t+N-1} [\alpha_2 (v_{xi} - v_{x_{desi}})^2 + \alpha_3 (v_{yi} - v_{y_{desi}}^2)] \quad (7.4)$$

where $v_{x_{desi}}$ and $v_{y_{desi}}$ are the desired velocity of UAV, α_1 , α_2 and α_3 are tuning parameters.

At each sampling time $t = kTn_t$, $k \in \mathbb{Z}^+$, the optimization problem could be solved:

$$\begin{aligned}
 \min(J) \quad \text{subj.to} \quad & \dot{P}_{\mathcal{F}_i} = [R_{\mathcal{F}_b}^{\mathcal{F}_i}] v_{\mathcal{F}_b} = \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} \\
 & \sqrt{\dot{v}_x^2 + \dot{v}_y^2} \leq a_{max}, \\
 & X + Y < \frac{\sqrt{2}}{2} a_{max} t_o^2 - (v_x + v_y) t_o, \\
 & \|a_{k,t}\| \leq a_{max}, \quad k = t + 1, \dots, N.
 \end{aligned} \tag{7.5}$$

When solutions of X_t^* , Y_t^* , $v_{x_{desi}}$ and $v_{y_{desi}}$ are found, values are sent to the position controller as next targeted set-point with the desired velocity. The optimal solutions of X_t^* , Y_t^* sent to the position controller will select the ones that fall on the hard geo-fence boundary. The state information of the UAV system and the hard geo-fence boundary coordinates are needed, i.e. UAV current position $(X(t), Y(t))$, the hard geo-fence boundary coordinates (X_i^s, Y_i^s) and $Y_i^s = -X_i^s$, with $i = t + 1, \dots, t + N$. The desired velocities and accelerations in all axes are under the maximum values without out of control.

Control objectives of this MPC design is a *Constrained Control*:

1. Preventing UAV from breaching the hard geo-fence.
2. Keep the position of UAV within the hard geo-fence boundary.
3. Decrease the velocity of UAV within the velocity limitation.

7.3 Simulation and Results

The initial condition of UAV, path planning and quadcopter model, remain the same as the Section 4.2.2. The dynamic soft geo-fencing system is developed in the second block which is position controller of Figure 4.2. The idea of the dynamic soft

geo-fencing system can be described as:

- Soft geo-fence is generated based on current velocity, acceleration of quadcopter.
- The system continuously calculates the distance between the fixed hard geo-fence and the edge of the soft geo-fence boundary.
- When breaching the hard geo-fence in time T which is set to 2 seconds, it generates a new set of position command and desired velocity command in order to stop UAV.

The design structure is shown in Figure 7.1.

Figure 7.1: Dynamic Soft Geo-fencing in Position Controller.

Both Activate Geofence Stop block and Check Activation Geofence Stop block will

be activated when the breaching flag equals one. It overwrites the next path commands on the hard geo-fence boundary. The Check Activation Geofence Stop function is used to inspect if disturbances exist. Thus UAV tries to hover on the boundary without drifting away. Both the outputs of the quadcopter statuses and path commands go to geo-fence calculation block, where the soft geo-fence boundary based on current status of the quadcopter and the distance between the soft and hard geo-fence are calculated. If breaching is detected, the activated flag will be set to zero and this flag will trigger the Activate Geofence Stop block and Check Activation Geofence Stop block. The MPC is developed in MPC block.

The path is the same diamond shape as mentioned in Chapter 4. The hard geo-fence is set as $y = -x$ line excluding the original point. The data could be seen in Figure 7.2.

Figure 7.2: Quadcopter Output Status with Dynamic Soft Geo-fencing Implemented.

From the position graphs, as shown in the Figure 7.2, the position after time 27s remains unchanged. This shows that quadcopter detects the hard geo-fence and stops.

The Figure 7.3 shows the current position of the quadcopter and the soft geo-fence boundary together. In Figure 7.3, the black dot indicates the position of the UAV. The black circle is the generated soft geo-fence boundary of UAV. Soft geo-fence boundary is continuously generated according to the current position and status of the UAV.

Figure 7.3: Quadcopter Performance in Default View with Dynamic Soft Geo-fence Boundary and UAV Together.

The performance overview is shown in Figure 7.4 and 7.5. The color plane is used to indicate the hard geo-fence $y = -x$ boundary.

Figure 7.4: Quadcopter Performance in Default View with Dynamic Soft Geo-fencing Implemented.

Figure 7.5: Quadcopter Performance in X-Y View with Dynamic Soft Geo-fencing Implemented.

From the performance animation in both views, there are overshoots, i.e. the quadcopter breached the hard geo-fence. This may be caused by control part of UAV being not as precise or due to wind disturbance.

Comparison between static soft geo-fencing and dynamic soft geo-fencing,

	Static Soft geo-fencing	Dynamic Soft geo-fencing
Pre-desgin Geofence Shape Needed	Yes	No
Measure Average Wind Velocity	Yes	Yes
More Calculation Needed	Yes	No
Assume UAV Velocity	Yes	No
State Prediction Capability	No	Yes

Table 7.1: Static Soft geo-fencing vs. Dynamic Soft geo-fencing.

The static soft geo-fencing needs to do more calculations based on the pre-design geo-fence shape, wind velocities and the UAV maximum velocity. But the dynamic soft geo-fencing algorithm need not to pre-design of the geo-fence boundary since it has the position prediction capability based on UAV's current velocity and acceleration. Hence, the dynamic soft geo-fencing has more advantages compared to the static soft geo-fencing.

7.4 Conclusion

In this chapter, a dynamic soft geo-fencing safety method is proposed to increase the safety of UAVs and their surrounding properties and humans. The MPC is used to achieve the objective. Simulation results demonstrate the effectiveness of the proposed method. Real flight comparison between the static geo-fencing and the dynamic geo-fencing is to be introduced in Chapter 8.

Flight Verification Experiment

8.1 Introduction

Even the most reliable system has to be tested frequently to ensure ultimate safety. In this chapter, UAV flight test set-up, procedure and test results are discussed and analyzed. The quadcopter as shown in Figure 3.20, is used as the testbed and its specification shown in Table 8.1.

Description	Value/Unit
Maximum horizontal velocity	3 m/s
Maximum vertical velocity	1 m/s
Inner loop update rate	400 Hz
Outer loop update rate	50 Hz
SD card log rate	10 Hz
GPS accuracy	3 m
Mass of UAV	1.261kg
Flight time	20 minutes

Table 8.1: Specific Parameters of Quadcopter.

The proposed PID controller is tested on a quadcopter and is fine tuned during the actual flight. The final PID values are captured in GCS as shown in Figure 8.1.

And summary of final PID values is displayed in Table 8.2.

Figure 8.1: Tuned PID of Quadcopter F450 captured in GCS.

Description	Value
MPC_XY_P (Multiple Copter Position Controller P Gain)	1
MPC_XY_VEL_D (Multiple Copter Horizontal Velocity Controller D Gain)	0.005
MPC_XY_VEL_I (Multiple Copter Horizontal Velocity Controller I Gain)	0.005
MPC_XY_VEL_P (Multiple Copter Horizontal Velocity Controller P Gain)	0.1
MPC_Z_P (Multiple Copter Altitude Controller P Gain)	1
MPC_Z_VEL_D (Multiple Copter Altitude Velocity Controller D Gain)	0
MPC_Z_VEL_I (Multiple Copter Altitude Velocity Controller I Gain)	0.02
MPC_Z_VEL_P (Multiple Copter Altitude Velocity Controller P Gain)	0.1

Table 8.2: Summary of Tuned PID of Quadcopter F450.

A program is developed to record the GPS signals data into an SD card, which can be used during ground analysis.

8.2 Flight Setup

The overview of the avionic structured is shown in Figure 8.2. It is assembled using common off-the-shelf items.

Figure 8.2: Avionics Systems of UAV.

The autopilot that is used and placed on the quadcopter, is an ARM Cortex M4 processor Pixhawk board. The attitude and position control are running in $400Hz$ and $50Hz$, respectively. Dynamic geo-fencing system is programmed in the autopilot board and simple path is uploaded into the autopilot as well. Ground control computer is used to monitor both the status of the UAV and its flight path. For single UAV flight, QGroundControl GCS [62] software is used. It allows for ease of customization to fit the test requirements.

8.3 Experimental Results

Two scenarios are developed to test the dynamic soft geo-fencing safety system.

- Fly the path with the fixed hard geo-fence boundary only.
- Fly the path with the dynamic soft geo-fencing safety system implemented and with the fixed hard geo-fence boundary.

The quadcopter flies in autonomous mode in both scenarios, with the same start point that is predefined in the program. The flight positions are recorded and a comparison is made for two scenarios. The altitude is set to $10m$ height above ground. The videos of the actual flights are recorded with a mobile phone and the screen shots of the videos are shown in figures below. The orange color square indicates the hard geo-fence boundary, the yellow circle shows the position of UAV.

Figure 8.3: UAV Flight Path With Static Geo-fencing Capability.

Figure 8.4: UAV Flight Path With Dynamic Geo-fencing Capability.

The data logged in SD card are analyzed. The analysis of the complete flight trajectories from the log files of the SD card, is shown in the Figure 8.5.

Figure 8.5: Full-path Comparison of two Scenarios.

The yellow line indicates the hard geo-fence boundary. The brown line shows the UAV flight path with the static geo-fencing algorithm while the blue line is the flight

path with dynamic geo-fencing algorithm. The results show that although both the algorithms have overshoots, the amount of overshoot of the dynamic geo-fencing is significantly smaller than the static geo-fencing. Hence, the use of the dynamic soft geo-fencing algorithm is better and the overshoot might be due to wind or inaccuracy of GPS signals.

The comparison of longitude of two scenarios are shown in Figure 8.6. The brown color shows the flight data of the UAV with static geo-fencing algorithm and the blue color indicates the flight data of UAV with dynamic geo-fencing algorithm. The straight line is the hard geo-fence boundary which is defined as a constant longitude.

Figure 8.6: Comparison of Longitude of two Scenarios.

The starting positions of two scenarios as shown in the Figure 8.7 are different, because the logging starts at the different positions.

The distance differences of two scenarios could be seen in Figure 8.7.

Figure 8.7: Distances Comparison of Two Scenarios.

Both lines reflect the distance between the UAV and the hard geo-fence boundary. The black line shows the result when the dynamic geo-fencing algorithm is used while the red line serves as a benchmark where there is static geo-fencing algorithm. The time line in the figure is meaningless, because the only critical data is the distance between two lines when UAV reaches the final state. The distance axis indicates the distance between the hard geo-fence boundary and the position of the UAV. From the figure above, the dynamic geo-fencing algorithm gives the closer distance difference of the final state of the UAV than the static geo-fencing. Hence, the result shows that the dynamic soft geo-fencing algorithm is useful.

8.4 Discussion & Conclusion

There exist overshoots on the quadcopter with the dynamic soft geo-fencing implemented. The quadcopter tries to compensate the wind in order to reach the point on

the hard geo-fence. The results of several flight tests demonstrate the effectiveness of the proposed dynamic geo-fencing method which is implemented in the autopilot board. In this way, errors of the system can be found during the numerous flight tests to further increase the safety of the UAV platform. The controller that has ability to auto-adjust the PID has to be implemented in the autopilot board in the future to compensate overshoot error.

Conclusions and Future Works

9.1 Conclusions

This thesis presents novel contributions in developing a comprehensive safety system with dynamic soft geo-fencing capability to increase the safety during UAV flight.

Firstly, a detailed dynamic modeling is conducted to model a specific quadcopter for this research.

Secondly, a simulation method of control is thoroughly introduced. A path planning program is implemented to test the performance of quadcopter in Simulink. A PD controller is used in the outer loop while a PID controller is used in the inner loop. An animation software is also further modified to check the control performance of quadcopter when it flies a fixed path.

Thirdly, an exhaustive safety system is developed to check all the possible failures of UAV during flight. All four motors are checked for their rotation conditions once the UAV is powered on. Sensors onboard, such as IMU, GPS quality, communication link between UAV and transmitter, etc. are checked continuously. If a triggering condition is satisfied, necessary actions are conducted to protect property and civilians. The fault tree analysis are addressed in detail as shown in Chapter 6. A three layers geo-fence boundaries is designed and tested during the 16 UAV formation

show. Moreover, a GCS is developed to monitor the real time status for multiple UAVs.

Last but not the least, a MPC is designed for dynamic soft geo-fencing to predict the possible future location based on the acceleration, speed of quadcopter and the distance between fixed hard geo-fence and predicted soft geo-fence. The position setpoint is changed if a breaching is detected. Then the performance have been examined. The flight experiments have demonstrated the effectiveness of the proposed safety system.

9.2 Future Works

The flight experiment results presented in Chapter 8 show the existence of an overshoot. Hence, a controller that has ability to auto-adjust PID are to be added into the autopilot board to compensate the overshoot error.

Although MPC is used in the dynamic soft geo-fencing to stop the UAV at the precise position on the hard geo-fence boundary, the need to have the UAV to adjust its attitude controller automatically in actual flight is preferred. This is because it allows the UAV to get back to its correct path location instead of stopping at the hard geo-fence boundary. Hence, the system shown in Figure 9.1 that has ability to auto-adjust PID value is going to be studied in order to auto-adjust the controller in real time.

Last but not the least, additional functions have to be added in the GCS for multiple UAVs monitoring as well, such as real-time image stitching, dynamic soft geo-fencing real-time generation. Communication protocol has to be further improved to make it more light weight to achieve less transmission time.

Figure 9.1: Safety Structure Including Dynamic Soft Geo-fence Capability.

Appendix

A.1 Safety Considerations

Here's what you should observe when flying unmanned aircraft for recreation under conditions that do not require a permit. Things we should do:

- Know the characteristics of the aircraft and how to fly it safely.
- Ensure that the aircraft is safe for flight before you operate it.
- Fly only in good visibility and weather conditions.
- Keep your aircraft within your sight at all times.
- Ensure the operation of transmitting devices of the unmanned aircraft system complies with IDA requirement.
- Keep a sufficient distance from people, property and other aircraft (manned or unmanned).

Things we should avoid:

- Don't fly the aircraft over any crowd.
- Don't fly an aircraft weighing more than 7kg (in total).
- Don't suspend, carry or attach any item to the aircraft, unless it is manufactured to hold the item.
- Don't carry hazardous substances using the aircraft.
- Don't drop or discharge any item or substance from the aircraft.
- Don't fly where you may interfere with emergency service provider; or over moving vehicles where you can endanger or distract drivers.
- Don't fly the aircraft over or within restricted, prohibited or danger areas, including security-sensitive locations.
- Don't fly within 5km of any airport/military airbase, or higher than 200 feet.

A.2 Risk Assessment Form

Risk Assessment Form for UAS						
S/N	Hazard	Phase(s) of Flight	Consequences/(s)	Causal Factor/(s)	Risk Level	Control/ Recovery Measures
RISK ASSESSMENT FOR AIRWORTHINESS OF UAS						
1	Structure Broken: -Quadcopter Arm, -Propeller etc. -Avionics Mounting. -Wing Break.	Takeoff, Mid-flight, Approach	-Drone crashes, -Flight Termination	-Improper installation of drone frame. -Have potential break points in structure.	4*1=4 Low A*B=C A is severity B is likelihood C is risk level	-FTM(Flight termination,mode) activated -Hardware team ensures that hardware structure was installed properly -Pilot team log every potential damage even it is not damaged apparently
2	Throttle Power system Spoiled: -Motor/ESC loss power or overheat	Takeoff, Mid-flight, Approach	-Flight Termination Mode	-Environment temperature too high -Current load too high due to motor overload and operate too long	4*1 = 4 Low	-FTM Activated -Pilot team monitor the flying time of the UAV
3	Main power Loss: -Li-Po Battery reading less than 3.4V per cell	Take-off, Mid-flight, Approach	-Flight Termination Mode	-Unbalanced charging let voltage drop faster	4*1 = 4 Low	-FTM Activated - Hardware team store Li-Po battery properly - Base monitor drone voltage
4	Electrical Management System Board Power loss	Takeoff, Mid-flight, Approach	-Drone Crashes, -Flight Termination	Power cable loose	4*1 = 4 Low	-FTM Activated -Avionics team ensures quality of each board
5	Autopilot, -program crash, -board lockup, -board loss power	Takeoff, Mid-flight, Approach	-Flight Termination	-Program has potential bug, -Board power supply cable loose	4*1 = 4 Low	-FTM Activated -Autopilot team tries to ensure program operate well with no bugs -Pilot team ensures cable connect properly before flight
6	sensor track error: -Gyroscope, Magnetometer, Accelerometer, Barometer, -GPS loss	Takeoff, Mid-flight, Approach	-Drone land on undesirable land. - Landing	-IMU has interference, -IMU board fail	3*2=6 Medium	-Land Activated -Pilot team ensures no interference within 300m of flight zone, -Avionics team ensures the quality of autopilot IMU board
7	IMU board stuck	Takeoff, Mid-flight, Approach	-Flight termination	-IMU board fail	4*1=4 Low	-FTM Activated -Avionics team ensures the quality of low level autopilot IMU board

Table A.1: RISK ASSESSMENT FOR AIRWORTHINESS OF UAS (Part 1)

Risk Assessment Form for UAS						
S/N	Hazard	Phase(s) of Flight	Consequences/(s)	Causal Factor/(s)	Risk Level	Control/ Recovery Measures
RISK ASSESSMENT, FOR AIRWORTHINESS OF UAS						
8	Telemetry Datalink Loss	Takeoff, Mid-flight, Approach	-May not display drone status on GCS has potential danger if any fail happen, -Return Mode	-Power supply loss, -Antenna not fix well -EMI	1*2=2 Low	-Return Activated -Avionics team ensures the quality of soldering port of telemetry power and antenna port -Pilot team ensures no interference within 300m of flight zone
9	Remote Controller Link Loss	Takeoff, Mid-flight, Approach	-Return	-cable not connect well, -EMI	2*2=4 Low	-Return Activated -Avionics team ensures the contact of RC cables -Pilot team ensures cable connect properly before flight

Table A.2: RISK ASSESSMENT FOR AIRWORTHINESS OF UAS (Part 2)

S/N	Hazard	Phase(s) of Flight	Consequences/(s)	Causal Factor/(s)	Risk Level	Control/ Recovery Measures
RISK ASSESSMENT FOR, THE REQUIRED TYPE OF OPERATIONS						
1	Fly near to the safety boundary	Takeoff, Mid-flight, Approach	-Potential danger to, cross safety boundary, (Soft geo-fencing Adaptive Mode) (Hard geo-fencing Flight termination)	-GPS loss -Strong Wind	2*2=4 Low 4*1=4 Low	-Pilot team ensures weather condition suitable for flight -Pilot team ensures, no interference within 300m of flight zone -Autopilot team will design geo-fencing algorithm to prevent cross -FTM/Adaptive Mode Activated
RISK ASSESSMENT FOR, THE AREA OF OPERATIONS						
1	Built-up areas in the vicinity of the area of operations	Mid-flight	Drone collides with, building or trees and crash lands on nearby people and/or buildings Possible collide to trees	-Mission Planner need to simulate waypoints before load into autopilot -Mission waypoints not consider potential dangerous building in the flight zone	4*1=4 Low	-Mission planner team ensures no potential obstacles in flight paths -Pilot Team ensures no people inside geo-fence boundary always inspect UAV status in air -Base Team ensure UAV is in normal status

Table A.3: RISK ASSESSMENT FOR THE REQUIRED TYPE OF OPERATIONS

Publication List

- Published papers

1. S. Zhang, D. Wei, M. Q. Huynh, J. X. Quek, X. Ma, and L. Xie, “Model Predictive Control Based Dynamic Geofence System for Unmanned Aerial Vehicles,” in *AIAA Infotech@ Aerospace*, p. 0675, 2017.
2. D. Wei, S. Zhang, L. Xie, X. Dong, and K. Z. Y. Ang, “Comprehensive Safety System Design and Development for Unmanned Aerial Vehicles Formation Flight System,” in *AIAA Infotech@ Aerospace*, p. 0249, 2016.

Bibliography

- [1] D. Barry, “Drones: Safety Considerations and Potential Liability Concerns.” <http://blog.willis.com/2015/12/drones-safety-considerations-and-potential-liability-concerns-definitely-more-than-just-a-toy/>, 29.13.2015.
- [2] G. Hoffmann, D. G. Rajnarayan, S. L. Waslander, D. Dostal, J. S. Jang, and C. J. Tomlin, “The stanford testbed of autonomous rotorcraft for multi agent control (STARMAC),” in *Digital Avionics Systems Conference, 2004. DASC 04. The 23rd*, vol. 2, pp. 12–E, IEEE, 2004.
- [3] “How many consumer drones, toys aside, exist in the USA as of Jan, 2016.” <http://www.droneflyers.com/2016/01/how-many-consumer-drones-toys-aside-exist-in-the-usa-as-of-jan-2016/>.
- [4] “Flying of Unmanned Aircraft.” <https://www.caas.gov.sg/caas/en/ANS/unmanned-aircraft.html/>.
- [5] “UAV Safety Checklist.” <http://air-vid.com/wp/uav-safety-checklist/>.
- [6] “Geo-fence.” <https://en.wikipedia.org/wiki/Geo-fence/>.
- [7] “DJI Introduces New Geofencing System for its Drones.” <https://www.dji.com/newsroom/news/dji-fly-safe-system>.
- [8] D. Wei, S. Zhang, L. Xie, X. Dong, and K. Z. Y. Ang, “Comprehensive Safety System Design and Development for Unmanned Aerial Vehicles Formation Flight System,” in *AIAA Infotech@ Aerospace*, p. 0249, 2016.

-
- [9] S. Khatoon, M. Shahid, H. Chaudhary, *et al.*, “Dynamic modeling and stabilization of quadrotor using PID controller,” in *Advances in Computing, Communications and Informatics (ICACCI, 2014 International Conference on)*, pp. 746–750, IEEE, 2014.
- [10] T. Luukkonen, “Modelling and control of quadcopter,” *Independent research project in applied mathematics, Espoo*, 2011.
- [11] S. K. Phang, C. Cai, B. M. Chen, and T. H. Lee, “Design and mathematical modeling of a 4-standard-propeller (4SP) quadrotor,” in *Intelligent Control and Automation (WCICA), 2012 10th World Congress on*, pp. 3270–3275, IEEE, 2012.
- [12] H. Huang, G. M. Hoffmann, S. L. Waslander, and C. J. Tomlin, “Aerodynamics and control of autonomous quadrotor helicopters in aggressive maneuvering,” in *Robotics and Automation, 2009. ICRA’09. IEEE International Conference on*, pp. 3277–3282, IEEE, 2009.
- [13] G. M. Hoffmann, H. Huang, S. L. Wasl, and E. C. J. Tomlin, “Quadrotor helicopter flight dynamics and control: Theory and experiment,” in *In Proc. of the AIAA Guidance, Navigation, and Control Conference*, 2007.
- [14] S. F. Ahmed, K. Kushsairy, M. I. A. Bakar, D. Hazry, and M. K. Joyo, “Attitude stabilization of Quad-rotor (UAV) system using Fuzzy PID controller (an experimental test),” in *Computing Technology and Information Management (ICCTIM), 2015 Second International Conference on*, pp. 99–104, IEEE, 2015.
- [15] D. Misra, S. Bhattacharjee, B. Goswami, and R. Ghosh, “Guaranteed performance PID controller for UAV pitch control,” in *2016 IEEE First International Conference on Control, Measurement and Instrumentation (CMI)*, pp. 172–176, IEEE, 2016.

- [16] Z. Zuo, "Trajectory tracking control design with command-filtered compensation for a quadrotor," *Control Theory & Applications, IET*, vol. 4, no. 11, pp. 2343–2355, 2010.
- [17] İ. Dikmen, A. Arısoy, and H. Temeltaş, "Attitude control of a quadrotor," in *Recent Advances in Space Technologies, 2009. RAST'09. 4th International Conference on*, pp. 722–727, IEEE, 2009.
- [18] A. Tayebi and S. McGilvray, "Attitude stabilization of a four-rotor aerial robot," in *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, vol. 2, pp. 1216–1221, IEEE, 2004.
- [19] S. Bouabdallah, A. Noth, and R. Siegwart, "Pid vs LQ control techniques applied to an indoor micro quadrotor," in *Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, vol. 3, pp. 2451–2456, IEEE, 2004.
- [20] A. Torres Gaitan and Y. Bolea, "Modeling and robust attitude control of a quadrotor system," in *Electrical Engineering, Computing Science and Automatic Control (CCE), 2013 10th International Conference on*, pp. 7–12, IEEE, 2013.
- [21] X. Wang, G. Lu, and Y. Zhong, "Robust H_∞ flight control of small-scale unmanned helicopter for WCICA 2014 proceedings published by IEEE," in *Intelligent Control and Automation (WCICA), 2014 11th World Congress on*, pp. 639–644, IEEE, 2014.
- [22] T. Madani and A. Benallegue, "Backstepping control for a quadrotor helicopter," in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pp. 3255–3260, IEEE, 2006.
- [23] P. De Monte and B. Lohmann, "Trajectory tracking control for a quadrotor

- helicopter based on backstepping using a decoupling quaternion parametrization,” in *Control & Automation (MED), 2013 21st Mediterranean Conference on*, pp. 507–512, IEEE, 2013.
- [24] L. Tan, L. Lu, and G. Jin, “Attitude stabilization control of a quadrotor helicopter using integral backstepping,” in *Automatic Control and Artificial Intelligence (ACAI 2012), International Conference on*, pp. 573–577, IET, 2012.
- [25] F. Jiahe and L. Rui, “Fractional PID and backstepping control for a small quadrotor helicopter,” in *Control Conference (CCC), 2015 34th Chinese*, pp. 5701–5706, IEEE, 2015.
- [26] K. M. Zemalache, L. Beji, and H. Marref, “Control of an under-actuated system: application a four rotors rotorcraft,” in *Robotics and Biomimetics (ROBIO). 2005 IEEE International Conference on*, pp. 404–409, IEEE, 2005.
- [27] L. Doitsidis, K. P. Valavanis, N. C. Tsourveloudis, and M. Kontitsis, “A framework for fuzzy logic based UAV navigation and control,” in *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, vol. 4, pp. 4041–4046, IEEE, 2004.
- [28] S. K. Hong, “Fuzzy logic based closed-loop strapdown attitude system for unmanned aerial vehicle (UAV),” *Sensors and Actuators A: Physical*, vol. 107, no. 2, pp. 109–118, 2003.
- [29] T. Sangyam, P. Laohapiengsak, W. Chongcharoen, and I. Nilkhamhang, “Path tracking of UAV using self-tuning PID controller based on fuzzy logic,” in *SICE Annual Conference 2010, Proceedings of*, pp. 1265–1269, IEEE, 2010.
- [30] L. M. Argentim, W. C. Rezende, P. E. Santos, and R. A. Aguiar, “PID, LQR and LQR-PID on a quadcopter platform,” in *Informatics, Electronics & Vision*

- (ICIEV), *2013 International Conference on*, pp. 1–6, IEEE, 2013.
- [31] A. Zulu, S. John, *et al.*, “A Review of Control Algorithms for Autonomous Quadrotors,” *Open Journal of Applied Sciences*, vol. 4, no. 14, p. 547, 2014.
- [32] N. Cao and A. F. Lynch, “Inner-Outer Loop Control for Quadrotor UAVs with Input and State Constraints,” *IEEE Transactions on Control Systems Technology*, vol. 24, no. 5, pp. 1797–1804, 2016.
- [33] L. M. Kok, “Singapore to introduce drone law: 5 things about these flying machines.” <http://www.straitstimes.com/singapore/singapore-to-introduce-drone-law-5-things-about-these-flying-machines/>, 2015.
- [34] R. E. Weibel and R. J. Hansman, “Safety considerations for operation of unmanned aerial vehicles in the national airspace system,” 2006.
- [35] Y. Kang and J. K. Hedrick, “Linear tracking for a fixed-wing UAV using nonlinear model predictive control,” *Control Systems Technology, IEEE Transactions on*, vol. 17, no. 5, pp. 1202–1210, 2009.
- [36] A. Raemaekers, “Design of a model predictive controller to control UAVs,” *Traineeship report, Technische Universiteit Eindhoven, Netherlands*, 2007.
- [37] E. Joelianto, E. Maryami Sumarjono, A. Budiyo, and D. Retnaning Penggalih, “Model predictive control for autonomous unmanned helicopters,” *Aircraft Engineering and Aerospace Technology*, vol. 83, no. 6, pp. 375–387, 2011.
- [38] B. B. Stovner, “Model Predictive Control in UAV trajectory planning and gimbal control,” 2014.
- [39] P. Scokaert and D. Mayne, “Min-max feedback model predictive control for

- constrained linear systems,” *Automatic Control, IEEE Transactions on*, vol. 43, no. 8, pp. 1136–1142, 1998.
- [40] A. Richards and J. How, “Robust stable model predictive control with constraint tightening,” in *American Control Conference, 2006*, pp. 6–pp, IEEE, 2006.
- [41] S. Lucia, R. Paulen, and S. Engell, “Multi-stage nonlinear model predictive control with verified robust constraint satisfaction,” in *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, pp. 2816–2821, IEEE, 2014.
- [42] W. Langson, I. Chrysoschoos, S. Raković, and D. Q. Mayne, “Robust model predictive control using tubes,” *Automatica*, vol. 40, no. 1, pp. 125–133, 2004.
- [43] D. Perez-Rodriguez, I. Maza, F. Caballero, D. Scarlatti, E. Casado, and A. Ollero, “A Ground Control Station for a Multi-UAV Surveillance System: Design and Validation in Field Experiments,” *Journal of Intelligent & Robotic Systems*, vol. 69, no. 1-4, pp. 119–130, 2013.
- [44] H. Lokesha, S. Veena, J. A. Rathod, *et al.*, “Design of speech based ground control station for controlling the micro air vehicles,” in *2015 Fifth International Conference on Advances in Computing and Communications (ICACC)*, pp. 451–454, IEEE, 2015.
- [45] X. Ding and X. Wang, “Design and realization of ground control station for multi-propeller multifunction aerial robot,” in *Mechatronics and Automation (ICMA), 2014 IEEE International Conference on*, pp. 227–232, IEEE, 2014.
- [46] I. T. Gaillimore and K. D. Driver, “Vertical Takeoff and Landing (“VTOL”) Aircraft,” Mar. 15 2013. US Patent App. 13/843,163.
- [47] S. A. Raza, “Design and control of a quadrotor unmanned aerial vehicle,” *University of Ottawa (Canada) Theses*, 2010.

- [48] “Quadcopter Dynamic Modeling and Simulation.” <https://github.com/dch33/Quad-Sim/>.
- [49] T. Bresciani, “Modelling, identification and control of a quadrotor helicopter,” *MSc Theses*, 2008.
- [50] P. E. Pounds, D. R. Bersak, and A. M. Dollar, “Stability of small-scale UAV helicopters and quadrotors with added payload mass under PID control,” *Autonomous Robots*, vol. 33, no. 1-2, pp. 129–142, 2012.
- [51] “Quadcopter Dynamic Modeling and Simulation.” <https://github.com/dch33/Quad-Sim/>.
- [52] G. Cai, B. M. Chen, K. Peng, M. Dong, and T. H. Lee, “Modeling and control system design for a UAV helicopter,” in *Control and Automation, 2006. MED’06. 14th Mediterranean Conference on*, pp. 1–6, IEEE, 2006.
- [53] M. Ryll, H. H. Bühlhoff, and P. R. Giordano, “Modeling and control of a quadrotor UAV with tilting propellers,” in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pp. 4606–4613, IEEE, 2012.
- [54] B. Erginer and E. Altuğ, “Modeling and PD control of a quadrotor VTOL vehicle,” in *Intelligent Vehicles Symposium, 2007 IEEE*, pp. 894–899, IEEE, 2007.
- [55] V. Grabe, H. H. Bühlhoff, and P. R. Giordano, “On-board velocity estimation and closed-loop control of a quadrotor UAV based on optical flow,” in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pp. 491–497, IEEE, 2012.
- [56] “Climate of Singapore.” www.weather.gov.sg/climate-climate-of-singapore/.

-
- [57] N.-F. Standard, “Announcing the advanced encryption standard (aes),” *Federal Information Processing Standards Publication*, vol. 197, pp. 1–51, 2001.
- [58] “MAVLink Micro Air Vehicle Communication Protocol.” <http://qgroundcontrol.org/mavlink/start/>.
- [59] J. P. Munson and V. K. Gupta, “Location-based notification as a general-purpose service,” in *Proceedings of the 2nd international workshop on Mobile commerce*, pp. 40–44, ACM, 2002.
- [60] F. Ivis, “Calculating geographic distance: concepts and methods,” in *Proceedings of the 19th Conference of Northeast SAS User Group*, 2006.
- [61] M. Arnold and G. Andersson, “Model predictive control of energy storage including uncertain forecasts,” in *Power Systems Computation Conference (PSCC), Stockholm, Sweden*, 2011.
- [62] “QGroundControl GCS.” <http://qgroundcontrol.io/>.