GLOBALLY ROBUST TRACKING CONTROL OF A QUADROTOR AERIAL VEHICLE FOR MULTI-BEHAVIOR APPLICATIONS

by

Bara Jamal Emran

B.Sc., University of Jordan, 2011

M.Sc., American University of Sharjah, 2014

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

in

THE COLLEGE OF GRADUATE STUDIES

(Mechanical Engineering)

THE UNIVERSITY OF BRITISH COLUMBIA

(Okanagan)

February 2019

The following individuals certify that they have read, and recommend to the College of Graduate Studies for acceptance, a thesis/dissertation entitled:

GLOBALLY ROBUST TRACKING CONTROL OF A QUADROTOR AERIAL VEHICLE FOR MULTI-BEHAVIOR APPLICATIONS

---

submitted by Bara Jamal Emran in partial fulfillment of the requirements of

the degree of Doctor of Philosophy.

Dr. Homayoun Najjaran, School of Engineering

Supervisor

Dr. Dwayne Tannant, School of Engineering

Supervisory Committee Member

Dr. Rudolf Seethaler, School of Engineering

Supervisory Committee Member

Dr. Liwei Wang, School of Engineering

University Examiner

Dr. William Melek, University of Waterloo

External Examiner

Abstract

This research contributes to the development of a practical controller for a quadrotor unmanned aerial vehicle (UAV) by addressing three main challenges: underactuation, model uncertainty, and actuator failure. Depending on flight conditions and the interaction with the environment, quadrotors require to operate in different flight regimes including hovering, aggressive maneuver, near-ground maneuver, and fault-tolerant flight. By realizing sensor readings and system requirements, the control system seamlessly switches between different designed schemes in real time to engage the most suitable one in the feedback loop.

To enable the full capacity of the quadrotor and perform agile maneuvers, a global attitude tracking control system is proposed. The controller is developed directly on the special Euclidean group with a region of attraction covering the entire configuration space, i.e., globally valid. The control law is based on the dynamic surface control (DSC) method to avoid “explosion of terms”, i.e., a common problem of previously reported solutions. Asymptotical convergence of tracking error is guaranteed in presence of system uncertainties and extreme disturbances without a priori knowledge of their bounds.

A position control system is proposed that outclasses the performance of existing solutions under extreme disturbances. It consists of a second order sliding mode control (SMC) system to guarantee stability of the position dynamics by generating a proper command for the attitude controller, and a switching mechanism based on multiple Lyapunov functions (MLFs) to improve tracking performance despite extreme disturbances.

A fault-tolerant tracking controller featuring fault detection and robust control capable of coping with the total failure of one or two adjacent rotors is proposed. A wind-speed sensor is used to detect actuator failure regardless of its cause. A significant novelty of this work is that a single generic control feedback strategy is adopted for both the normal flight operation and

when a fault occurs. Hence, unlike previous solutions, there is no risk of instability while switching between control laws.

Finally, the proposed attitude and position control methods have been verified on a testbed developed in this research, and the efficacy of the controller in coping with fault scenarios was proven in simulation.

Lay Summary

A practical and robust control system is developed for a generic quadrotor, which is the most commonly used type of unmanned aerial vehicles (UAVs) in civilian applications. The proposed control system significantly improves the performance of the quadrotor in controlling its attitude and position despite extreme disturbances (e.g., wind and gust), system uncertainties (e.g., addition of unknown payloads) as well as the total failure of one or two of its rotors; hence the entire control system is impressively faults tolerant. In essence, the controller adopts a smart switching mechanism that continuously evaluates the instantaneous sensor readings and flight requirements and then switches between different control strategies to maximize the quadrotor's agility and maneuverability. For verification of the proposed control system, a fully instrumented laboratory testbed with the ability to measure the quadcopter's angles as well as near-real simulations was developed and used for this research.

Preface

The entire work presented in this thesis was conducted at the Advanced Control & Intelligent Systems (ACIS) Laboratory of the University of British Columbia, Kelowna campus under the supervision of Dr. Homayoun Najjaran. I was responsible for modeling the system, designing the control algorithms, building the simulation environment and conducting the experiments with the guidance and advice of my supervisor, Dr. Najjaran.

- The motivation for considering the near-ground maneuvers was the challenges faced while carrying out a research under the supervision of Dr. Homayoun Najjaran in collaboration with Dr. Dwayne Tannant. The work involved flying in low altitudes to monitor small concentrations of fugitive methane around a landfill. The results have been published in: B. J. Emran, D. D. Tannant, and H. Najjaran, “Low-altitude aerial methane concentration mapping,” Remote Sens., vol. 9, no. 8, 2017.
- A comprehensive literature review on the control system of quadcopters leading to my in-depth understating of the outstanding problems and the state of the art has been presented in Chapter 2. This material has been disseminated for a wide audience in B. J. Emran and H. Najjaran, “A review of quadrotor: An underactuated mechanical system,” Annual Reviews in Control, Pergamon, 2018.
- The development of the laboratory testbed and its low-level control systems has been elaborated in Chapter 3. Besides the research described in this thesis, the testbed has been used extensively in other ACIS research project. In particular, one project has benefited from the testbed and published in M. K. Al-Sharman, B. J. Emran, M. A. Jaradat, H. Najjaran, R. Al-Husari, and Y. Zweiri, “Precision landing using an adaptive fuzzy multi-sensor data fusion architecture,” Appl. Soft Comput. J., vol. 69, pp. 149–164, 2018. Another research has been submitted to the International Journal of Intelligent

Unmanned Systems and is under review: K. Gupta, B.J. Emran and H. Najjaran “Vision-Based Pose-Estimation of a Multi-Rotor Unmanned Aerial Vehicle”.

- A version of the global control algorithm proposed in Chapter 4 has been submitted to ISA Transactions and is currently under review: B.J. Emran and H. Najjaran, “Global Tracking Control of Quadrotor Based on Adaptive Dynamic Surface Control”.
- An early attempt of developing the switching mechanism, presented in Chapter 5, was published in B.J. Emran and H. Najjaran, “Switching Control of Quadrotor with Adaptation Mechanism” in 2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2016.
- A version of Chapter 5 has been submitted to the ISA Transactions and is under submission: B.J. Emran and H. Najjaran, “Switching Control of Quadrotor System for Extreme Disturbances”.
- A simpler version of the adaptive mechanism proposed in Chapter 4 and 5 has been published: B.J. Emran and H. Najjaran, “Adaptive neural network control of quadrotor system under the presence of actuator constraints” in 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2017.
- A general overview of the attitude control system, proposed in Chapter 4, and position control system, proposed in chapter 5, has been submitted to the International Conference on Robotics and Automation (ICRA) and is under review: B.J. Emran and H. Najjaran, “Global Tracking Control of Quadrotor using Adaptive Dynamic Surface Control”.
- The fault estimation technique proposed in Chapter 6 has been submitted to the International Conference on Robotics and Automation (ICRA) and is under review: A. Ravishankara, B.J. Emran, M. K. Al-Sharman and H. Najjaran, “Real-time Thrust Estimation and Regulation for Multirotor Aerial Vehicle”.

- The fault recovery control system proposed in Chapter 6 has been submitted to the American Control Conference (ACC) and is under review: B.J. Emran and H. Najjaran, “Fault-tolerant Tracking Control for a Quadrotor UAV Based on Dynamic Surface Control”.

Table of Contents

List of Tables

List of Figures

List of Notations

UAV	Unmanned aerial vehicle
UMS	Underactuated mechanical system
VTOL	Vertical takeoff and landing
BC	Backstepping control
DSC	Dynamic surface control
SMC	Sliding mode control
TSMC	Terminal sliding mode control
DTSMC	Dynamic terminal sliding mode control
MIAC	Model identification adaptive control
MRAC	Model reference adaptive control
PD	Proportional–derivative
PID	Proportional-integral-derivative
LQR	Linear quadratic regulator
MLFs	Multiple Lyapunov functions
2D	Two-dimensional
3D	Three-dimensional
3 DOF	Three degree-of-freedom
6 DOF	Six degree-of-freedom
AI	Artificial intelligence
NN	Neural network
DNN	Deep neural networks
RBF-NN	Radial basis function neural network
CMAC	Cerebellar model arithmetic computer
FD	Fault diagnosis

FE	Fault estimation
FT	Fault tolerant
FTC	Fault tolerant control
KF	Kalman filter
UKF	Unscented Kalman filter
EKF	Extended Kalman filter
AFKF	Adaptive fuzzy Kalman filter
TO	Thau observer
DC	Direct current
ESC	Electrical speed controller
GUI	Graphical user interface
IMU	Inertial measurement unit
GPS	Global positioning system
HIL	Hardware-in-the-loop
OS	Operating system
ROS	Robot operating system
E-frame	Earth inertial reference frame
B-frame	Body frame of reference frame
NWU	Axes configuration in north-west-up
ZYX	Rotation axes configuration
SO(3)	Special orthogonal group
SE(3)	Special Euclidean group
COM	Center of mass
PWM	Pulse width modulation
RPM	Rotation per minute
IAE	Integral of absolute error

c_x	Trigonometric function $cos(\mathrm{x})$
s_x	Trigonometric function $sin(\mathrm{x})$
t_x	Trigonometric function $tan(\mathrm{x})$
$\cdot_\times$	Times map known also as skew matrix
$\cdot^\vee$	Vee map inverse of times map
$\Re$	Set of real numbers
Ω_i	Rotation speed of motor i in $[rad/s]$
K_b	Thrust coefficient in $[N\ s^2]$
K_d	Thrust to drag ratio
l	Distance from the rotors' center to the quadrotor's center of mass in $[m]$
f_i	Thrust force of rotor i in $[N]$
m_i	Drag moments generated by rotor i in $[N\ m]$
τ_i	Moments due to rotor i force in $[N\ m]$
ξ	linear position
V	linear velocity
Θ	Euler angles
ϕ	Roll angle
θ	Pitch angle
ψ	Yaw angle
R	Rotation matrix
ω	Angular velocity
m	Mass in $[kg]$
g	Gravitational acceleration in $[m/s\text{^}2]$
J	Inertial matrix in $[kg\ m^2]$
$sat(\cdot)$	Saturation function

I_{xx}	Inertia around x-axis in $[kg\ m^2]$
I_{yy}	Inertia around y-axis in $[kg\ m^2]$
I_{zz}	Inertia around z-axis in $[kg\ m^2]$
dt	Sampling time in $[s]$
D_ω	Lumped uncertainty of the attitude dynamics
$\widehat{D}_\omega$	Estimation of the uncertainty D_ω
$\widehat{W}_\omega$	Estimation weights of the uncertainty D_ω
$\widetilde{W}_\omega$	Estimation weights error of the uncertainty D_ω
ϵ_ω	Approximation error of the uncertainty D_ω
θ_ω	Unknown constant parameters of the attitude dynamics
$\hat{\theta}_\omega$	Estimation of the unknown parameter θ_ω
$\tilde{\theta}_\omega$	Estimation error of the unknown parameter θ_ω
R_c	Rotational command signal
R_d	Filtered signal of the rotational command signal
$\tilde{R}$	Rotation augmented error
τ_R	Time constant of the attitude low-pass filter in $[s]$
ζ_R	Augmented error of the attitude low-pass filter
s_R	Surface error of the attitude
ω_c	Angular velocity command signal
ω_d	Filtered signal of Angular velocity command signal
τ_ω	Time constant of the angular velocity low-pass filter in $[s]$
ζ_ω	Augmented error of the angular velocity low-pass filter
s_ω	Surface error of the angular velocity
D_v	Lumped uncertainty of the position dynamics
$\widehat{D}_v$	Estimation of the uncertainty D_v

$\widehat{W}_v$	Estimation disturbance weights of the uncertainty D_v
$\widetilde{W}_v$	Estimation weights error of the uncertainty D_v
ϵ_v	Approximation error of the uncertainty D_v
θ_v	Unknown constant parameters of the position dynamics
$\hat{\theta}_v$	Estimation of the unknown parameter θ_v
$\tilde{\theta}_v$	Estimation error of the unknown parameter θ_v
p_{z_d}	Desired altitude command signal
p_d	Desired position command signal
e_z	Tracking error of the altitude signal
e_p	Tracking error of the position signal
s_z	Sliding surface error of the altitude signal
s_p	Sliding surface error of the position signal
$\mathrm{W}_{\mathrm{speed}}$	Wind speed in $[\mathrm{m/s}]$
v_z^s	Free stream velocity in $[\mathrm{m/s}]$
v_z^i	Induced velocity in $[\mathrm{m/s}]$

Acknowledgements

My deepest gratitude goes to my supervisor, Dr. Homayoun Najjaran for his encouragement, guidance and support. The door to his office was always open whenever I ran into a trouble spot or had a question about my research or writing. Over the past four and half years, I have learned so much from him.

I would like to extend my gratitude to all of my colleagues at the Advanced Control & Intelligent Systems (ACIS) Laboratory for all their critiques discussions and valuable contribution to this work. Furthermore, I want to thank the UBC School of Engineering staff and faculties for their support during my years of study.

Finally, I must express my very profound gratitude to my parents and to my wife Islam for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. I also would like to thank my daughter Layla for her tolerance for my absence from home during this work. This accomplishment would not have been possible without them. Thank you.

Dedication

To my beloved wife Islam and my gorgeous daughter Layla.

Chapter 1: Introduction

During the past two decades, there has been a continuously increasing demand for deployment of UAVs in broad applications. Recently, there have been several attempts to carry out complicated tasks in various civilian applications. Some illustrative examples of this trend include a delivery system from Amazon known as Prime Air designed to increase the safety and efficiency of the package delivery system using UAVs[1]. Another futuristic example is a taxi drone by Volocopter equipped with 18 propellers and designed to fly for 30-minutes to enhance the transportation system [2]. However, multirotor vehicles have already been used in many applications where safe performance in complex and cluttered environments is required. For instance, a recent attempt of integrating a laser-based methane detector into a multi-rotor UAV to generate an efficient aerial methane concentration map of a landfill has been reported in [3]. Nevertheless, similar to any other nonlinear systems, the stability, safe operation and precise control of UAVs suffer from intrinsic complexities exasperated by underactuation, model uncertainties, actuator failure, and input constraints. The focus of this thesis is to enhance the control system of quadrotor UAV as an underactuated mechanical system and address these challenges.

1.1 Underactuated Mechanical Systems

An underactuated mechanical system (UMS) is a system that has more degrees of freedom (DOF) to be controlled than the number of independently controlled actuators exerting force or torque onto the system. A UMS offers many advantages including energy efficiency and increased operational capacity (e.g., more degrees of freedom) without the need for additional hardware. These advantages are very attractive for unmanned vehicles with limited resources and energy storage, such as aerial and underwater vehicles. Specifically, a UMS can offer a system design that costs and weighs significantly less without losing or reducing the configuration space. Although the modeling, formulation and control theory of UMS have been

studied to a great extent in the last decade, the control of a UMS in practical applications including aerial and underwater vehicles remains a nontrivial task. Very interesting reviews and books have been published on this topic to emphasize the importance and applications of the UMS [4]–[8].

In terms of the number of actuators, mechanical systems are divided into two categories: fully actuated or underactuated. Actuators are subject to different constraints. Typically, they are heavy, expensive, power hungry and require maintenance so that a fewer number of actuators is generally desirable from the system design viewpoint. Thus, a mechanical system can be designed and built as a UMS from the outset. Furthermore, even a system designed as a fully actuated system can become underactuated due to actuator failure in the runtime. Taking non-rigid dynamics and model uncertainties into account are the other reasons for considering a mechanical system as a UMS. Despite the importance of the UMS control problem in real-world applications, most available control strategies are only applicable to fully actuated mechanical systems. Specifically, those developed techniques cannot be directly used for controlling a nonlinear UMS. For example, the widely used feedback linearization technique for controlling nonlinear systems is not applicable to UMS because it cannot override and cancel the nonlinear dynamics of the UMS. This has driven researchers to expand the control strategies for UMS that are considered to be more realistic and profitable systems.

For over two decades, the UMS has continued to attract more attention in a variety of engineering disciplines, especially in robotics and aerospace. In the control community, nowadays, the UMS is considered as one of the most active research fields. Originally, the research on the UMS started by investigating the applications of non-holonomic systems. Nonholonomic systems are related to UMS since they are subjected to kinematic constraints that limit their velocity in certain directions. A famous example of such a nonholonomic system is a two-wheeled mobile robot. However, the UMS has both kinematic and dynamics constraints that for example can limit the accelerations in certain directions. Examples of the UMS range

from very simple mechanisms such as an acrobat (a single actuator two-joint pendulum) to very complex and nonlinear systems such as a humanoid robot.

1.2 Quadrotor System

In general, unmanned aerial vehicles (UAVs) have drawn a lot of attention in the last decade due to their numerous outdoor and indoor civilian applications [9]–[14]. Among different types of UAVs, quadrotors have drawn even more attention for applications in complex indoor environments due to their simplicity in design, small size, lightweight, and great dynamical maneuverability and obstacles avoidance. A typical quadrotor has six degrees of freedom and only four actuators which makes it a UMS with a degree of underactuation of two. Hence, the great advantage of the quadcopter that is its simple design comes at the cost of a complex control [15]. Specifically, the control constraints of a quadcopter complicate trajectory tracking and stabilization significantly. The main control constraints elaborated in this research are attributed to underactuation, system failure, input constraints and model uncertainties including parametric and nonparametric ones. However, similar to any highly nonlinear systems, other constraints, such as strong subsystems coupling, measurement noise, and system disturbances can also compromise the quadrotor controller performance.

Most solutions and control strategies discussed in the literature are able to accomplish their defined tasks either by restraining the capability of the quadrotor or by flying the system in a well-known and controlled environment. For instance, some solutions simplify the control problem by considering only the linear dynamics of the system. Thus, they restrain the change of the attitude angles to a small margin (<15º). Other solutions assume the system operates in perfect conditions and neglect the impact of uncertainties in the environment and system model, which can cause disturbances or lead to system failures. To achieve this, the environment is fully instrumented, and the exact pose of the UAV is given to the UAV by a precise indoor localization system, which is limited or unavailable in real-world applications. In order to control

the quadrotor using its full capabilities in an uncontrolled environment, one needs a sophisticated control system that is capable of achieving system stability and path tracking simultaneously in the presence of the mentioned challenges.

1.3 Flight Behavior

Similar to any aerial vehicles, quadrotors need to operate in different flight regimes to accomplish their tasks. Hence, the requirement of the system performance changes over time depending on the current objectives. For example, stabilizing the quadrotor to stand still in the air has different control requirements than controlling the quadrotor to track a predefined path. While the stabilizing control requires the performance to be robust and precise, the tracking requires the control law to have a wide stability range and the performance to be agile. Moreover, a fully autonomous quadrotor must also perform takeoff and landing maneuvers. However, when the vehicle is near the ground, its dynamics change dramatically. Consequently, this effect needs to be considered while developing the controller to avoid rocking and rollover behavior during the takeoff and landing. Designing a single control system to operate in multiple flight regimes is nontrivial.

The following subsections introduce the flight regimes considered in this research namely hovering, near-ground flight, aggressive maneuver, and fail-safe flight. For each flight behavior, a detailed description of its importance and performance requirements is given in addition to the traditional control strategy and algorithm.

1.3.1 Hovering Flight

In general, one of the key features of multirotor vehicles is the ability to stand still in the air, unlike fixed-wing aircraft. Hovering flight regime has important applications, for instance, it is used as a transient state to smoothen transition between different flight behaviors including transition between take-off and path following, as an idle behavior to reduce energy consumption until the system receives further commands, or even as a part of an application

procedures such as remaining motionless on top of the desired area for remote sensing and surveillance. The main objective of this flight behavior is to stabilize the system in hovering conditions; maintaining a fixed position in a 3D space with zero velocities and attitude angles. During hovering, the system's dynamics can be approximated by a linear model, which simplifies the control problem where the control objective is to maintain a fixed position. This means the control system has to be highly robust and reject external disturbances caused by for example wind and gust.

A typical procedure is to develop a simple control system with optimal control gains, such as linear quadratic regulator (LQR), which is considered to be efficient in terms of reducing energy and computational power. However, a simple LQR controller will not be sufficient to overcome all control challenges, such as system uncertainties, external disturbances, and measurement noises in order to guarantee a satisfactory performance.

1.3.2 Near-ground Flight

In general, the UAV must avoid coming into contact with the ground during the flight, though in certain applications quadrotors need to operate near the ground. For instance, during a detained visual inspection of infrastructure, the quadrotor may need to hold a close and fixed distance from the ground. Besides, a fully autonomous quadrotor is required to perform smooth takeoff and landing, which means flying at a close distance from the ground for a certain period. Such maneuvers are critical as they might lead to quadrotor's roll over and crash. The system dynamics changes dramatically near or touching the ground due to the ground effect phenomenon, which in turn affects the characteristics of the control problem that will lead to accidents if not treated properly. The main objective of this flight regime is to achieve autonomous takeoff and landing maneuvers successfully on a stationary platform, which can be horizontal or slightly tilted. The requirement of the control system is to accurately estimate and

compensate for the disturbance caused by the ground effect. In addition, the flight path must be planned so that it successfully takes off and lands.

One approach is to develop a nonlinear observer to estimate the ground effect disturbances. However, depending only on a single observer may be insufficient as the system is actively subjected to different system and control input disturbances. Using machine learning algorithms to enhance the flight path planning procedure will improve the overall maneuvers and system performance.

1.3.3 Aggressive Flight

Nowadays, quadrotors are used in various indoor and outdoor applications where they are known for their agility and maneuverability. However, they suffer from high-energy consumption and limited payload capacity. This limits their use in many applications that require relatively high flight duration and payload. In addition, UAV might need to perform an aggressive maneuver in order to avoid an obstacle or evade a collision. Thus, utilizing the full capability of quadrotors and pushing them to their limits are essential. The main requirement of this behavior is to follow a relatively aggressive path by employing the system's full capability. However, most control systems simplify the control problem by restraining the system capability. For instance, designers consider only a small operating range for attitude angles. Such simplifications are not valid for aggressive maneuvers. Flying by aggressive maneuvers excites the unmodeled dynamics of the system and increases the sensitivity of the model to uncertainties. This complicates the control problem and might lead to system instability, wandering the system away from the limited configured state space. Thus, simplified control systems will typically fail to maintain the system stability and tracking. Consequently, traditional control systems are not sufficient for aggressive maneuvers. The control system must possess robust adaptation characteristics to manage the uncertainties and compensate the disturbances excited by the aggressive maneuvers [16] to use the full potential of quadrotors.

1.3.4 Fail-safe flight

In general, physical systems are subjected to system failure due to component failure, manufacturing problems, aging or many other factors. In quadrotors, component failures can lead to a smooth and moderate situation compromising the flight condition, or an abrupt and severe situation leading to a catastrophic crash. In other words, there are different types of system failures that a quadrotor system can suffer from. One example is to lose the effectiveness of one of its actuators, i.e., rotors. In order to reduce the vulnerability of a system to faults, a health management system needs to be developed for the quadcopter. The main task for the control system in this scenario is to prevent a system failure but also avoid a crash if a fault occurs. The system control will use a health management system that is responsible for monitoring the state of the system, detecting a fault, diagnosing its nature and selecting an appropriate method to deal with it as quickly as possible.

1.4 Research Objectives

There are numerous solutions for controlling quadrotor systems in the literature. However, most control strategies are still ad-hoc and task-oriented. More precisely, they present a solution for a specific system behavior. Nevertheless, quadrotor systems have various system behaviors, such as hovering, maneuvering, takeoff, and landing. Available solutions are not suited for multi-behavior systems. In case of applying them, it could cause a disaster; crashing the system. Therefore, developing a single control system that can handle all the mentioned challenges is nontrivial. Instead, one can divide the control problem to subtasks, each of which is to address an individual challenge. Then, build a number of controllers where each one is well tuned to deal with a certain challenge. Finally, design a switching logic to select between the controllers and switch to the one that is preferable for the current needs.

The main objective of this research is to contribute to developing a practical control scheme for a quadrotor UAV for all mentioned behavior. The challenge will be on developing a

control system that is capable of achieving pre-determined flight performance for various behaviors. Here, the quadrotor flight regime is divided into four different behaviors described earlier. Depending on flight conditions and the interaction with the environment, each flight behavior has its own system performance requirements. The proposed control system will switch between differently designed control systems depending on specific conditions. The control candidate needs to achieve satisfactory performance under the presence of the challenges mentioned above. The switching between the control candidates is governed by a designed logic system known as a supervisor. The supervisor task is to use the online measurements and the current system requirements and select the most suitable control candidate to be placed in the feedback loop at each instant.

1.5 Thesis Organization

The thesis is presented in seven chapters organized as follows.

Chapter 2 focuses on reviewing several problems that complicate the process of designing trajectory tracking or stabilization of a quadcopter. The problems considered in this research include underactuation, parametric and nonparametric uncertainties in the model, system failure, and input constraints. The chapter also discusses each of these problems by addressing the challenges involved and reviews the solutions mostly concerning the application of different control strategies presented in recent literature. The chapter is organized as follows. Section 2.1 presents the underactuation challenge and the underlying internal dynamics that may raise the stability problem. Control challenges due to actuator failure in quadrotor system are introduced in Section 2.2. Section 2.3 introduces control challenges due to uncertainties in quadrotor systems including parametric and non-parametric ones. Finally, Section 2.4 addresses the control challenge due to constraints in the control signal and presents its traditional solutions.

Chapter 3 focuses on describing the typical design of quadrotor systems and their unique features. The description includes reference frames, dynamic equations, system components, and the simulation model. The chapter is organized as follows. Section 3.1 describes the basic concepts related to the structure and movements of the quadrotor system. The reference frames and the nonlinear dynamic equations are described in Section 3.2 and Section 3.3, respectively. Section 3.4 describes the typical sensors used in UAVs. Finally, Section 3.5 and Section 3.6 present the simulation model and the testbed system designed to validate the control, respectively.

Chapter 4 presents a global nonlinear tracking control system for attitude dynamics of quadrotor UAV. The proposed controller guarantees the asymptotical convergence of tracking error in the presence of underactuation, uncertainties and external disturbances. The chapter describes the development of the control system which consists of using a dynamic surface control and an adaptive mechanism. The chapter is organized as follows. Overview and problem formulations of this chapter are presented in Section 4.1 and Section 4.2, respectively. Section 4.3 describes the development of the control system in detail. Finally, Section 4.4 illustrates the features of the proposed control system using both numerical simulation and testbed.

Chapter 5 presents a novel position tracking control system for a quadrotor UAV in the presence of uncertainties and extreme external disturbances. The control method combines three techniques: a second order sliding mode control, a switching mechanism and a non-parametric adaptation mechanism. The chapter is organized as follows. Overview and problem formulations are presented in Section 5.1 and Section 5.2, respectively. The control development is presented in Section 5.3. Finally, in Section 5.4, the results are illustrated through numerical simulation tests.

Chapter 6 proposes a tolerant tracking controller for a quadrotor UAV in Cartesian space motion in spite of the total failure of one or two adjacent rotors. This integration

significantly improves the disturbance rejection and fault estimation of the thrust and rotor, respectively. This chapter is organized as follows. Section 6.1 presents an overview of the control challenge when a fault occurs in the system. The problem formulation and the description of the detection system are presented in Section 6.2 and Section 6.3, respectively. The control developments of both cases are presented in Section 6.4. Finally, the overall performance efficacy is demonstrated in Section 6.5.

Chapter 7 discusses the conclusions and future work.

Chapter 2: Literature Review

The dynamics of unmanned aerial vehicles (UAVs), similar to most highly nonlinear systems, has several characteristics that complicate the process of designing trajectory tracking or stabilization. The major problems investigated in this thesis include underactuation, parametric and nonparametric uncertainties in the model, system failure, and control input constraint. The following subsections discuss each of these characteristics by addressing their problems and reviewing the solutions concerning the application of different control strategies presented in recent literature.

This chapter is organized as follows. Section 2.1 presents the underactuation challenge and the underlying internal dynamics that may result in the instability problem. Several control techniques for solving the problem are reviewed. These include dividing the system into several subsystems, artificial intelligent algorithms, and discontinuous solutions. Control challenges due to actuator failure in quadrotor system are discussed in Section 2.2. Three solutions are discussed and addressed: redundant actuators, diagnostic controls, and fault-tolerant control. Section 2.3 introduces control challenges associated with parametric and non-parametric uncertainties in quadrotor models. Three commonly used control strategies namely robust control, adaptive control, and robust adaptive control are discussed and compared. Finally, Section 2.4 addresses the control challenge due to constraints in the control input and presents its practical solutions.

2.1 Underactuation

Quadrotors are an example of an underactuated system with six degrees of freedom and only four independent control inputs. Underactuation limits the number of system configurations that can directly be controlled. In particular, the system cannot follow unrestricted flight in full vector space due to the lack of adequate control actions in their configuration space. Hence, the dynamics model of the quadrotors is not fully linearizable. In addition, the dynamics of the

quadrotor change dramatically due to the ground effect, such as when it performs near-ground maneuvers. This adds extra challenges to the control design and requires more sophisticated control algorithms. The focus of this section is on addressing these problems and discussing various non-traditional solutions proposed in the literature, such as dividing the system into subsystems, employing artificial intelligence (AI) algorithms and the inclusion of switching schemes.

Early control systems were based on a cascade feedback strategy which involves the following steps. First, the system is divided into two subsystems including an inner-loop and an outer-loop that form the attitude dynamics and position dynamics of the UAV, respectively. Then, a feedback control is designed for each loop separately. In practice, the feedback control of the inner-loop runs with a higher frequency, around 5-10 times faster than the feedback control of outer-loop. Finally, the loops are connected using the outer-loop feedback signal to provide reference for the inner-loop. This strategy is the most used one due to its practicality and simplicity. For example, in [17] different classes of feedback control systems are presented for the cascade feedback strategy showing their implementation and differences. Another example is shown in [18] where a hybrid attitude controller was designed to track the desired trajectory globally. Although dividing the system into two subsystems has simplified the problem, one still needs to take care of the internal dynamics that may raise instability behavior. Hence, one needs non-traditional control algorithms to deal with the underactuation and underlying internal dynamics of the system. More recent attempts for controlling a quadrotor involve using backstepping control (BC) [19]. In general, the BC approach is used to overcome the coupled and cascaded dynamics of a system. The design of BC is carried out by stabilizing the system states through a step-by-step recursive process which guarantees the stability of the entire system. However, the use of BC is problematic because of the so-called “explosion of terms” problem in the control law, which may dramatically complicate the computations required depending on the size of the system. Despite this challenge, BC is currently one of the most

commonly used approaches to control quadrotor in the literature. More examples of different implementations of BC can be found through this review.

Performing a precise trajectory tracking control is a very important feature for vertical takeoff and landing (VTOL) vehicles. However, traditional control algorithms lack the capability to achieve a precise tracking control for underactuated systems due to the complexity of the problem and the presence of nonlinear dynamics. Many researches have proposed the use of artificial intelligence (AI) to achieve tracking and stabilization of underactuated systems. One of the common use of AI methods in the literature is approximation of nonlinear functions to improve the control system robustness [19]. However, there are many attempts of employing AI methods for different purposes. For instance, a self-tuning fuzzy PID controller is proposed to improve the flight stability of quadrotor in the case of undisturbed condition [20]. The latter uses of a fuzzy inference system to actively tune the PID gains. The effectiveness and stability of this algorithm were demonstrated by simulation in conjunction with the Dijkstra path planning algorithm. This approach may suffer from the lack of robustness since the performance of the resulting control system depends highly on the exact plant parameters. Furthermore, the simulation in [20] also neglects the aerodynamic and gyroscopic effects. Recently, a novel approach based on Deep Neural Networks (DNN) have been proposed in [21]. The work focuses on enhancing the quadrotor capability to follow an arbitrary and complex “hand-drawn” trajectory, precisely. The authors developed a DNN model as a generalized add-on outside the feedback control loop to enhance the maneuvering of a UAV. The proposed approach was validated experimentally and showed its performance in comparison with a traditional PID controller. In general, DNN has a superior capability of approximating nonlinear functions compared to the traditional NN. However, it requires intensive offline training using a huge amount of flight data, which may be unavailable or not transferable between different VTOL systems. Also, selecting a proper architecture for the DNN learning is very important and depends on the user’s experience.

Another approach to tackle the tracking and stabilization problem of VTOL systems is the use of discontinuous solutions. In general, discontinuous solutions are not desired since they may excite some unmodeled dynamics of the system that in turn may raise the stability problem. Nonetheless, the interest in the use of discontinuous solutions has increased lately, especially for the systems where traditional control algorithms cannot provide a solution. More insight regarding the stability of systems using discontinues solutions is given in [22], [23]. In [24], the authors have proposed a switching output feedback control for a quadrotor system modeled as a switching linear time-varying system. The control system depends on Linear Matrix Inequalities and has guaranteed the stability of the quadrotor under external disturbances. However, the proposed solution is specified to deal with network induced time-varying delays rather than a general solution. A novel switching control algorithm based on multiple Lyapunov functions (MLFs) was employed in [25] to achieve tracking of Cartesian space motion and heading angle of a quadrotor. The approach decomposed the system into three fully actuated subsystems in pitch, roll and yaw domains. In these domains, the local sub-controllers were designed using the partial feedback linearization technique. Later in [26], the local sub-controllers were enhanced by modifying their robustness using an adaptive model reference control. The proposed solution aims to integrate the MLFs technique and the adaptive mechanism to solve the underactuation and parameter uncertainty problems, simultaneously. The algorithm shows some potential in solving the underactuation problem. However, the control signals suffer from chattering effect due to the switching mechanism. A further validation using a real flight experiment is needed to show its true potential. Another attempt for using switching controller strategy was proposed in [27] for the case of large-angle rotational maneuvers. The strategy is mainly based on the attitude error model, taking the model uncertainties and external disturbances into consideration. In particular, the proposed algorithm switches between two controllers depending on the amplitude of the tracking error. That is, in case of a large tracking error, an attitude stabilization controller is selected with the objective of

driving the system into the region of attraction of the second controller. In case of a small tracking error, a more sophisticated control system is used to control the system and achieve high-accuracy tracking. Although different experiments had been carried out to validate the proposed switching controller, the stability of the overall system with the switching logic had not been proven. As a result, there may exist a switching sequence that leads the system into instability.

Underactuation, uncertainties, and system coupling are not the only difficulties that complicate the design of control systems for the VTOL vehicles. Additional control complications arise when the system comes close to or is in contact with the ground, for example during takeoff and landing. Specifically, the dynamics of the vehicle changes drastically because of ground effect. This has led researchers to take additional terms into account and develop more sophisticated control algorithms for near-ground maneuvering. Early research work on landing maneuvers was simply commanded through position tracking maneuvers until the vehicle came into contact with the ground and landed. in [28], for example, the authors studied the problem of landing a quadrotor on a horizontally moving platform. The proposed control system decouples the system dynamics into inner and outer loops, to control the translational velocity and the attitude angles, respectively. However, the proposed controller did not consider uncertainties in the system, as it assumed perfect state measurements and tracking signals. A robust flight controller was designed in [29] for the quadrotor landing problem using a hybrid *automaton* approach. The proposed hybrid automaton allows for capturing the dynamics of different flight regimes that the vehicle crossovers, including free-flight and near ground maneuvers. In this work, the landing problem was broken down into three maneuvers: approaching a landing slope, sliding on the slope, and stopping. A separate trajectory generation and trajectory tracking controller was designed and applied to each maneuver. The experimental results had shown the feasibility of the proposed landing procedure under certain assumptions, e.g., knowing the angle of the landing pad. However, the main drawback of the proposed hybrid technique was that the

landing maneuver contains unconventional maneuver, namely sliding on a slope maneuver. More importantly, the technique requires that the current operational mode to be known. In particular, it requires the use of additional sensors, such as contact sensor at vehicle landing gear, to detect which part of vehicle contacts the ground to switch to a proper mode. Occasionally, VTOL vehicles need to perform near ground maneuvers to accomplish their tasks, e.g., collecting data from the surrounding environment. Hence, the vehicle will be subjected to high disturbance forces and torques generated by the vehicle vortices. To compensate for the disturbances, one approach is to estimate the external disturbances using a force and torque estimator. In [30], an estimation algorithm based on an Unscented Kalman Filter (UKF) was designed for VTOL vehicle. The proposed technique was demonstrated in an experiment to estimate aerodynamic disturbances generated by an external fan. In general, UKF surpasses Extended Kalman filter as it does not require the Jacobians of the dynamics especially for such a highly nonlinear system. For the proposed estimator to have a high performance, it still requires some parameters to be tuned manually, which might be tedious work for nonexperts. In addition, this work did not take into account system uncertainties; it assumed a perfect knowledge of the plant parameters, such as the mass and inertia matrix.

2.2 Actuator Fault

In general, a fault is something that changes the behavior of a system so that it no longer satisfies its purpose. As an example, a fault in a single system component, such as a sensor or actuator changes the performance of the overall system. This kind of failure usually happens for unexpected and unknown reasons. In order to reduce the vulnerability of a system to faults, a health management system can be employed. A health management system is an automatic inspecting system responsible for monitoring the state of the system, detecting a fault, diagnosing its nature, and selecting the appropriate action to deal with it as quickly as possible. One way to deal with a fault is to have a physical redundancy in system components,

but this might be costly, complex and impractical. Another way is by designing a reliable control system that ensures an acceptable performance despite the failure, or in other words, makes the system fault tolerant (FT). By this definition, the main purpose of the fault recovery mechanism is to maintain the system performance despite the presence of faults using the remaining non-faulty components, such as actuators and sensors [31]. Developing a control system having this feature attracted the interest of the control community for years. The purpose of this section is to formulate the fault recovery problem and review different solutions including hardware- and software-based approaches. The system faults that are taken into consideration vary from reduced efficiency to total failure of system components.

A fault tolerant control (FTC) is an integration of a health management system with the control system. It is deployed to detect and identify the nature of faults and apply a recovery method to deal with them. In general, a fault-tolerant control (FTC) system consists of two parts: fault diagnosis (FD) and control reconfiguration. The former detects and identifies the fault while the latter maintains the system performance by adapting to the faulty situation. There are different methods for implementing FD techniques. The most common method is through the use of *a prior* knowledge about the process and the interaction between various system components. This approach is known as a process-model based method. The passive FTC is used more in practice since it neither requires specific information about the fault nor sophisticated decision-making procedure to tolerate the fault, but its capacity for fault tolerance is limited. On the other hand, the active approach has a better capacity for fault tolerance since it detects and identifies the fault specifically and takes necessary condition-based actions to isolate faults of the system [32], [33]. A good review of FTC and its types can be found in [34].

System redundancy is a solution that can theoretically prevent system failure due to random causes. Yet the increased initial cost of additional hardware and system complexity limit its practicality. One approach to handle actuator redundancy is based on modular vehicle design. This technique consists of assembling a various number of standalone modules, each of

which may be capable of flying alone, in certain configurations. Hence, by increasing the number of actuated degrees of freedom of the overall vehicle, it is possible to overcome actuator failure. For example, a new class of modular aerial vehicles and its control scheme is proposed in [35].The concept is based on rigidly connecting various underactuated modules using a standard ducted-fan configuration. Thus, different dynamic behaviors and actuation scenarios can be obtained by changing the interconnection topology. The control scheme consists of control allocation module and a feedback controller. The former identifies the directions in the input space that affect the vehicle dynamics depending on the selected configuration, while the latter provides stabilization of the overall system. Later, several experimental validations of a control strategy for the proposed design are introduced in [36]. A physical prototype was realized by interconnecting two ducted-fan vehicles to demonstrate the effectiveness of the proposed methodology. The work shows a novel approach to overcome the actuator failure for different configurations in simulation, but the experimental results were limited to only one configuration with two-ducted fans. Another novel modular vehicle is the so-called Distributed Flight Array (DFA) composed of identical hexagonal-shaped modules that are cable of flying and performing a joint action [37]. In addition, each module has the ability to autonomously move on the ground and coordinate its actions with its connected peers. The main difference between the DFA and the previous modular system is its ability to adapt autonomously to arbitrary flight-feasible configurations. Later in [38], the work was validated experimentally using cascade control approach for various flight configurations. However, due to the system complexity, only hover flight condition was implemented and tested. One challenge in the modular approach is to build an accurate dynamic model of the system. A bigger challenge is to design control algorithms and estimation strategies for a system that consists of tens or maybe hundreds of individual elements interacting dynamically. This can lead to a very unexpected behavior. Thus, the major focus of the earlier work was on developing distributed and decentralized control systems that are capable of controlling such complicated systems.

System redundancy can also be achieved by simply adding extra actuators to a quadrotor frame. A hexacopter, a six-rotor UAV, is an example of vehicles with motor redundancy. Thanks to its simplicity, this approach is often the preferred solution for actuator failure. For example, a modification on the standard design of hexagon-shaped hexacopter was done in [39] by tilting the rotors towards the vehicle's vertical axis. This modification makes the hexacopter fault tolerant, since the system turns out to be controllable despite failure of a single actuator. Later, the control system of the proposed design was enhanced by considering a better design of the actuator location [40]. More precisely, the new design considered the maximum and minimum forces generated by the actuators, which made the solution more feasible for a real flight. Still, the proposed design has not been tested in an actual case of an actuator failure. Another example of motor redundancy for FT approach is proposed using a coaxial octocopter, an eight-rotor UAV, [41]. This approach proposed a complete FT architecture including error detection, fault isolation and system recovery. To stabilize the vehicle when a fault occurs, the proposed recovery algorithm compensates the loss of the failing motor by controlling its dual motor. Various flights were conducted to validate the capability of the proposed approach to maintain hovering condition while introducing artificial fault to the system's motor. However, the focus of this study was only on controlling the altitude angles while ignoring position control in the x-y plane.

The software-based approach is another way to give a system the capability to become fault tolerant without any additional hardware. It involves improving the overall control system to be tolerant to fault. An FTC involves two procedures: a fault diagnosis (FD) used to detect the occurrence of a fault, and a control recovery method used to maintain the vehicle's performance when a fault is detected. Many solutions proposed in the literature focus on FD while other solutions focus on the control recovery. However, a full solution for FTC is still an open area for research. One approach for designing an FD is based on a Thau observer (TO). Basically, the idea behind the TO is to mimic the nonlinear dynamics of a system augmented by a linear

correction term. For instance, a model-observer FD scheme based on a set of observer residuals was proposed in [42]. The proposed observer aims to detect sensor and actuator faults for a wide class of unmanned aerial vehicles. This scheme has been simulated on a quadrotor to detect sensor faults on the onboard inertial measurement unit. However, the proposed method focused only on the FD with inaccurate residuals, so it is not applicable to fault isolation and estimation. Moreover, the proposed observer is only applicable to simple models but not real processes, as it does not account for model uncertainties. An earlier work on fault estimation (FE) scheme using adaptive TO for a quadrotor is proposed in [43]. Their algorithm is demonstrated using data from a testbed as well as fault-free and fault data of real flight experiments. However, the method did not include the effect of uncertainties, disturbances, and noises, which, as a result, made the FE inaccurate and rough. A further improvement on the algorithm is proposed in [44] where the FE results have been enhanced and become more applicable to real flight data. In this work, the model uncertainties and disturbances in the real experiment were considered. The proposed adaptive scheme is based on an optimization method that can estimate the fault severity. The performance of the observer was shown using a quadrotor testbed, where 30% of effectiveness loss was imposed on one of the motors. However, the work mainly focused on FE and ignored the need of developing FTC for the fault system. Other researchers used Kalman filter and its extensions as a basis for their FE approach. Basically, Kalman Filter (KF) uses least square error optimization technique to predict the behavior of a process. However, for this filter to work it requires that the investigated process be linear. As a result, several solutions were designed to overcome this requirement. For example, Extended Kalman Filter (EKF) uses the Taylor series expansion method to linearize the nonlinear process, while Unscented Kalman Filter (UKF) uses unscented transformation for calculating the statistics of a nonlinear process. Different work were done by Zhang et al. on FE methods for quadrotors using a dual UKF [45] and a two-stage EKF [46]. The proposed techniques work efficiently to estimate the system states and parameters at the

same time. Still, the results of the FD methods were not robust for noise and have rough FD results if the transfer matrices in EKF are not sufficiently accurate. In addition, the proposed work was based only on simulation results and had not been tested in a real-flight experiment.

There are many types of faults which quadrotor systems can suffer from due to sensor and actuator failure. Here, we will mainly focus on hardware faults in actuators. Actuator faults are critical for quadrotor operation and can have different effects depending on their level. For example, losing the thrust effectiveness of one or multiple actuators is a dangerous type of fault which can lead to system instability. Different FTC strategies were proposed in the literature to deal with this challenge. For instance, an adaptive feedback technique was proposed in [47] for controlling a quadrotor with a fault recovery mechanism. Different simulations were performed to illustrate the performance of the system when the effectiveness of a single actuator was reduced by 50% for 10 seconds, and another time by a 20% to 30% effectiveness loss in two of the actuators, simultaneously. However, the technique was developed under the assumption that the quadrotor is flying at hovering condition to simplify the control design. A different approach was done recently in [48] using immersion and invariance observer to estimate the partial loss of effectiveness in actuators. The proposed FTC strategy was based on sliding mode control (SMC) and was verified, for only 20% effectiveness loss, using hardware-in-the-loop (HIL) testbed. In contrast to the former work, this work was developed based on nonlinear dynamics model and using quaternion configuration to improve system response in a wider state space and avoid singularity in control system. However, the algorithm was only developed for controlling the attitude angles and ignored the position control when the system is at fault. A good review on the existing work on FD and FTC for general UAV systems can be found in [49]. The work mainly considers techniques which target actuator and sensor faults. The work demonstrated the development and the experimentation of several FD and FTC techniques. The experiments involved studying the system stability under 70% effectiveness loss of a single actuator.

Another example of a fault which can cause a severe damage to a quadrotor system is the total loss of one or more of its actuators which is fatal to the control the system. Different FTC strategies were presented to allow the quadrotor to fly despite this loss. However, this leverage is possible under certain conditions. For example, in [50], a path following controller is presented for a quadrotor experiencing a single rotor failure. The proposed strategy controls the vehicle to maintain a constant velocity profile along the path. The control strategy was developed based on various assumptions that require validation. For example, the flight condition of the quadrotor at fault will remain close to hovering conditions. The work was only validated through simulation test using a simple dynamic model without taking aerodynamics effects into account, which may be inadequate for real flight conditions. Other control strategies were presented to allow the quadrotor to maintain its position by sacrificing the control of the heading angle. For instance, Mueller et el. [51] presented periodic solutions for a UAV experiencing a rotor failure. The proposed strategy involves rotating the vehicle freely around a defined axis fixed with respect to the body. Then, the UAV's position can be controlled by tilting this axis and varying the total produced thrust. The proposed approach was developed for failure cases including one, two or three propellers and validated experimentally for the first two cases and in simulation for the third one. However, losing the control of the heading angle may not be desirable especially for large vehicles. An emergency landing is another way of dealing with the failure of a motor. The proposed strategy involves three steps. First, the motor opposite to the failed motor is switched off to reduce its disturbance for yaw dynamics. Second, the quadrotor system is remodeled as a bi-rotor system with fixed propellers. Finally, the system is controlled to follow a pre-planned emergency landing trajectory. Using this strategy, different control solutions were proposed. For example, a PID controller was proposed in [52] and a backstepping control was proposed in [53]. This strategy drops the ability to control the yaw angle, but it presents a procedure to deal with a severe fault when it happens.

2.3 Parametric and Nonparametric Uncertainty

Most formal control methods are model-based, as they start with a mathematical model that describes the system in a selected domain of operation. However, the model may not be accurate in capturing the process dynamics resulting in model uncertainties. In particular, model uncertainties represent the differences between the model and real plant or process. There are two types of model uncertainties: parametric and nonparametric. When uncertainties occur in system parameters, they are called parametric uncertainties, such as unknown or time-varying parameters. A UAV carrying unknown payloads is a famous example of parametric uncertainty in its mass [54], [55]. However, other uncertainties such as unmodeled nonlinear dynamics and external disturbances including the effect of gust and wind are nonparametric uncertainties. In order to overcome potential model uncertainties, robust or adaptive solutions are needed to control the real plant. Those control solutions can be interpreted as an online technique capable of regulating systems with bounded uncertainties in their dynamics model [56].

In general, adaptive control has a learning feature which allows for improving its performance as adaptation progresses. On the other hand, robust control attempts to keep a consistent performance despite external disturbances and plant uncertainties. The difference between the two techniques yields different strengths and weaknesses for each. The adaptive approach excels in dealing with parametric uncertainties, uncertainties in constants, or slowly varying parameters. Adaptive control requires little or no a priori information about the unknown parameters. However, the existence of an adaptive solution for nonlinear systems generally depends on the ability for linear parameterization of the vehicle dynamics (which may not be always available). Finally, the overall control performance of an adaptive method may enhance during operation depending on how good the control system adapts to the plant uncertainties. On the other hand, robust control surpasses the adaptive approaches in dealing with nonparametric uncertainties, such as external disturbances, unmolded dynamics, and rapidly varying parameters. A robust control can exist even without full linear parameterization, but it

usually requires reasonable a priori estimates of the parameter bounds. In contrast, the overall performance of a robust controller may be low and remain the same over time, but it guarantees the stability of the overall system. The reason for the suboptimal performance of robust methods is that their main objective is to work with the model of the system with all uncertainties and find reasonable control input which may come at the cost of compromising some of the performance criteria. Besides, the uncertainty estimate is defined a priori and not updated during operation in robust control. More advanced controllers incorporate robust features into adaptive control to build robust adaptive controllers. In this type of controllers, the parametric uncertainties are reduced by parameter adaptation and nonparametric ones are handled by robust techniques [57].

Adaptive Techniques - There are many parameters in a quadrotor system that could suffer from parametric uncertainties and require the use of adaptive estimation techniques. Variation of mass, system inertia and aerodynamic coefficients are examples of such parametric uncertainties. Still, many traditional solutions proposed control systems which are not robust to parameter uncertainties, as they depend highly on the actual physical parameters of the system. For instance, an implementation of a backstepping (BC) feedback linearization technique was presented in [58]. The outdoor experiments showed a good convergence of tracking error compared to other traditional algorithms, such as the PID controller. The proposed approach depends on the exact plant parameters identified through rigorous experiments. However, such experiments require expensive tools with high accuracy which are usually not available. Thus, developing control algorithms which are robust against model uncertainties is very critical to obtain a more practical solution. This can be done through adaptive techniques. There are many proposed solutions in the literature that consider mass uncertainty only while others include the uncertainties in the inertial matrix. However, very few researchers considered uncertainties in other nonlinear parameters, such as aerodynamic coefficients. Several adaptive BC based techniques were utilized to design an adaptive controller for quadrotor with mass uncertainties.

Mainly, the BC approach is used to overcome the coupled dynamics and the underactuation problem while the adaption technique is designed to overcome the parameter uncertainties. For example, a projection-based adaptive controller is presented in [59] while a passivity-based adaptive control for mixed quadrotor-type is proposed in [60]. Both algorithms showed good compensations for mass uncertainty of the vehicle with an asymptotic tracking for the UAV's motion. The proposed approach was implemented in different applications including haptic teleoperation over the internet [60], quadrotor with serial manipulator [61], and quadrotor with tool operation [62]. However, their control law focused only on the uncertainties in mass parameter and ignored the uncertainties in angular dynamics of the vehicle and presence of the external disturbances. Other approaches included uncertainties in several parameters besides the mass. For example, an adaptive command-filtered BC control has been applied in [63] for controlling a quadrotor with uncertain parameters including mass, inertia and actuator efficiency. The proposed control law was designed for quadrotor to follow a trajectory tracking with a constrained attitude, attitude rate and velocity. The work was only validated through simulation, which may raise the question of its practicality in actual flight test. To deal with external disturbances, an adaptive integral BC approach has been proposed in [64]. The algorithm was experimented for unmodeled uncertainty and showed a good ability to reject disturbance. The proposed technique was further modified in [65] by introducing an integral-separated scheme, which is basically designed to avoid integral windup when the tracing errors are large. The scheme is modeled as a hard-switching condition which disengages an integral part of the control law if the tracking error is beyond a certain threshold. Later, an experimental test was performed in [66] using the proposed method. The results showed a satisfactory system response against disturbance in altitude; however, the attitude results were not convincing due to high measurement noises. In addition, it is worth to mention that the use of a conventional BC with integral adaptive laws is no longer applicable when the derivatives of the disturbance and the model uncertainty cannot be considered as zero. Other adaptive approaches have been

proposed to solve the parametric uncertainty. One of the most common approaches to adaptive control involves using the Model Reference Adaptive Control (MRAC) technique. For example, the design of a MRAC to control a quadrotor UAV in the presence of general uncertainties has been reported in [67]. The proposed control law is designed to be decentralized. That is, each local controller operates solely with no exchange of information between the subsystems. Although the proposed strategy has been validated experimentally to overcome uncertainties in the system model, it does not consider the uncertainty in control inputs and measurement noises. A combination of two adaptive controllers consisting of model identification adaptive control (MIAC) and MRAC for attitude stabilization and self-tuning of a quadrotor were suggested in [68]. The adaptive control is designed to overcome time-varying parameters by combining a recursive least-squares estimator with exponential forgetting technique. The effectiveness of the proposed algorithm is demonstrated in simulations and compared to another non-adaptive controller. Another combination of an adaptive controller using direct and indirect MRAC was proposed in [69]. The method demonstrated an improvement of robustness to parametric uncertainties even against the presence of losing the thrust effectiveness of one actuator. However, both methods use small angle approximation to linearize the dynamic model of the quadrotor, which raises the question of the effectiveness and performance of the proposed solutions in a wider state-space domain.

Robust Techniques - Quadrotors, like any dynamic systems, are subject to model uncertainties and external disturbances, such as wind. Dealing with disturbances and nonparametric uncertainties in system dynamics requires the control system to possess robust features. This can be achieved using various robust techniques. One approach is by augmenting a nominal controller with a robust compensator. In this case, the nominal controller is used to achieve the desired tracking, while the robust compensators are introduced to restrain the influence of system uncertainties, such as nonlinear dynamics, coupling, parametric uncertainties, and external disturbances. For example, a robust attitude control for uncertain

quadrotor was proposed and experimented in [70]. The proposed control system is linear time-invariant and consists of a proportional–derivative (PD) controller and a simple compensator. Later, the work has been extended to 6DOF motion tracking [71]. The controller was enhanced to include robustness against internal and external disturbances. Another improvement was the inclusion of a first-order compensator and a quaternion representation to avoid the singularity problem [72]. However, the experimental results show that the proposed control systems are limited to trajectory tracking and lack focus on position control. Another common approach for dealing with disturbances is by sliding mode control (SMC), which inherits robustness against disturbances under the matching conditions. In common practice, a sliding mode surface is defined as a linear combination of the position and velocity tracking errors of the system state variables. The resulting linear hyperplane defines the desired error dynamics of the state variables. Hence, it guarantees an asymptotic convergence of error dynamics to zero as time goes to infinity. However, SMC requires *a priori* knowledge of the uncertainty and disturbance bounds [73]. Several techniques have used SMC to address the position and attitude tracking control of quadrotor in presence of disturbances and model uncertainties; for example in continuous time domain [74] and in discrete time domain the authors [75]. However, the capabilities of these methods are demonstrated through simulation which may not duplicate a real experiment with the influence of actuators' limited bandwidth and measurement noises. In general, SMC suffers from the presence of chattering in its control signal. Several techniques have been proposed to eliminate this phenomenon [76]. For instance, a second order SMC to address the position and attitude tracking control for a quadrotor is proposed in [77]. Mainly, the work proposes using Routh-Hurwitz stability analysis for selecting the coefficients of the sliding manifold in a highly nonlinear relationship. However, the asymptotic convergence of sliding mode scheme is not enough for highly aggressive systems, such as quadrotor where the stabilization time is essential for its performance. To achieve finite-time convergence, different sliding mode schemes need to be used. For instance, a new scheme known as terminal sliding

mode control (TSMC) has been proposed in [78] to perform position and attitude tracking control of a quadrotor. The new scheme is based on replacing the conventional linear relation of the tracking error by a nonlinear one. The hyperplane is defined using a nonlinear term of the velocity error. This will drive the state variables to the sliding plane in a short period, which can be predicted. As a result, the error dynamics converge to zero in finite-time. However, the proposed scheme suffered from discontinuity in the switching signal which led to chattering in the control signals. This action has been addressed in [79] using dynamic terminal sliding mode control (DTSMC). Using an augmented sliding hyperplane, the improved scheme was able to guarantee a fast racking convergence and eliminate chattering caused by the switching control action. The results of the proposed scheme were validated using simulation. Still, the proposed terminal function is based on emphasizing the small tracking error to improve the convergence speed which, as a result, will have a negative impact when the feedback signal is subjected to measurement noises. Different approaches in the literature combined SMC with other techniques to enhance its features. For instance, combination of a block control technique with the super twisting SMC was demonstrated in [80]. The block control technique was used to design a smooth control law for each system's output, independently while the super-twisting SMC was implemented to improve robustness and avoid chattering. Later, the work was employed for high voltage powerline inspection applications to demonstrate its feasibility [81]. The resulting controller showed a good robustness against unmatched perturbations, such as aerodynamic forces; however, the work did not consider unmodeled uncertainties. Another combination for SMC was done in [82] to design a disturbance observer. The aim of this combination is to improve the robustness of the control system without the need of a high control gain. The work demonstrated good tracking results and robustness against external disturbances, model uncertainties and actuator failure. Still, the effects of SMC chattering on the overall tracking performance need to be investigated for real experiments. Other approaches have combined BC with SMC to improve the system stabilization and reject disturbances

simultaneously. For example, a novel robust controller for a general underactuated system based on a combination of BC and integral SMC was proposed in [83]. The proposed algorithm is aimed to stabilize the system under the presence of time-varying disturbances and uncertainties. The algorithm was demonstrated using a quadrotor system subjected to smoothly bounded disturbances and 25% parametric uncertainty. The results showed a good tracking error rapidly converged to zero with chattering-free input signal. Still, the algorithm suffers from "explosion of terms" in the control law. Furthermore, an adaptive controller to stabilize the quadrotor under input saturation was proposed in [84] which is based on the combination of BC and SMC. The proposed control design divided the problem into two nested loops: the translational outer-loop and the attitude inner-loop. The adaptive controller was introduced in the inner-loop to stabilize the attitude and counter the effect of uncertain parameters and external disturbances. In the outerloop, the input saturation problem was countered by adjusting the control parameters.

Artificial Intelligence (AI) Techniques - Several researchers investigated the integration of the artificial intelligence AI techniques into the controller to improve the overall performance using the data obtained during flight operation. The main advantage of AI techniques is its consideration of the nonparametric and nonlinear learning model. This makes it outclass most adaptive approaches, which consider only adaptation of a finite number of control parameters. For instance, the proposed algorithm in [84] was later enhanced in [85] using adaptive neural network (NN) approach. The proposed solution combined a BC and an SMC with adaptive radial basis function neural network (RBF-NN), which is used as an uncertainty observer. The aim of the proposed control was to control the attitude of a UAV in the presence of model uncertainties and external disturbances. The controller demonstrated its effectiveness to estimate lumped uncertainties without *a priori* knowledge of their bounds. However, several assumptions were made to simplify dynamic model of the UAV. In addition, the results were only demonstrated using simulation experiments which leave the concern regarding its

performance with the presence of a system delay, measurement noises and model nonlinearity unrelieved. Another interesting work was done by the Dynamic System Laboratory where they proposed a safe learning control system in [86]. In their approach, a robust control was used to guarantee the system performance while a machine learning algorithm was used to update the uncertainty model estimation. The work used a Gaussian process for online learning of the system dynamics using a set of N past observations. However, several simplifications on the model dynamics has been made including neglecting internal correlation between system dynamics and measurement noise. This was done to reduce the computational power required to calculate the inverse of the prediction matrix, which is an expensive $O(N^3)$ computation. In addition, the approach requires initialization based on experimental data obtained from a nominal control prone to failure. Finally, the proposed controller approach only stabilizes the system in the vicinity of the operating point, as it learns only the linear model.

Robust-Adaptive Techniques - Combining both adaptation and robustness techniques will yield a control system that can deal with almost all uncertainties. In general, a control system with an adaptation mechanism can drive the system to instability in the presence of small disturbances. This is due to the fact that adaptive mechanisms work better with structured and modeled uncertainties while performing poorly with unmodeled and nonparametric ones. Consequently, adaptive controls need to be enhanced with robust features to make them robust in presence of unmodeled dynamics or external disturbances. For instance, a robust adaptive control was designed in [87] for attitude tracking of a quadrotor. The proposed algorithm uses a special orthogonal group to express the attitude dynamics globally and avoids complex representation. The designed adaptive law is based on the Lyapunov approach enhanced with a robust technique to guarantee the boundedness of tracking in the presence of uncertainty in inertia matrix and external disturbances simultaneously. The experimental results showed an asymptotical tracking with a small attitude error. Still, the angular velocity tracking error was relatively large. The work was enhanced later in [88] and experimented for agile maneuvers. In

[89], a robust nonlinear composite adaptive control is used for quadrotor to follow a predefined path despite measurement noise and parameter uncertainty. The uncertainties manifested in unknown mass, system inertia, as well as thrust and drag coefficients. The algorithm combined the information obtained from tracking and parameter errors to enhance the adaption mechanism. Later, RBF-NN was introduced to enhance the robustness of the control system in the presence of unmodeled uncertainties [90]. The adaptive NN used is simple; it uses only a single hidden layer and does not require any a priori off-line training. The simulation results showed a good performance of the control system under external disturbances and actuator saturations. Still, both methods assumed the availability of the quadrotor's states while ignoring the measurement noises. A similar approach was discussed in [91] using a Cerebellar Model Arithmetic Computer (CMAC) instead of the RBF-NN as a nonlinear function approximator. The work proposed a robust technique to guide the adaptation process in order to prevent adjacent weights from drifting. The method was validated experimentally using a testbed while manipulating an external payload on the quadrotor's arms to act as an external disturbance. Still, the results showed a chattering signal on the testbed's response due to the high adjacent weights. Furthermore, a robust adaptive control was experimented in [92]. The proposed method applied a robust integral approach in the inner loop for disturbance rejection while the adaptive approach was applied to the outer loop in order to compensate for the parametric uncertainties. The adaptive method used is based on the immersion and invariance (I&I) methodology which, in contracts to the classic adaptive method, does not require the linear parameterization of the system. The results showed an asymptotic tracking performance in the presence of model uncertainties and external disturbances. However, the use of the estimated parameter in the feedback control is limited as it represents the overall uncertainties in the plant model.

2.4 Input Constraints

In general, one of the common assumptions for control systems is that they are affine in their control signal $u(t)$ appears linearly in the system dynamics. The general form of an input-affine system is represented by the following vector differential equation:

$$\dot{x} = f(x,t) + g(x,t)u(t) \tag{2-1}$$

where x represents the system state(s), f and g are linear/nonlinear functions and $u(t)$ is the control input. However, the presence of input saturation violates this assumption. This leads to a typical challenge of controlling most dynamic systems.

$$\dot{x} = f(x,t) + g(x,t)sat\big(u(t)\big) \tag{2-2}$$

where

$$sat\big(u(t)\big) = \begin{cases} u_{max} & u(t) > u_{max} \\ u(t) & u_{min} \leq u(t) \leq u_{max} \\ u_{min} & u_{min} > u(t) \end{cases} \tag{2-3}$$

Saturation is due to the working range of the actuator which is bounded to certain limits by application or physical reasons. This is known as the "input constraint" condition [93]. For example, an actuator of the quadrotor system is subjected to speed constraints, which can be operated from its lowest (turned off) to its highest (full speed) limits.

The existence of input saturation presents a significant problem for most control systems. As a result, researchers have been developing different techniques to deal with input constraints. One approach is to use an optimal linear quadratic regulator (LQR) technique and choose appropriate weights to obtain a good transient response with the presence of the input saturations [94]. Another approach uses a low-gain feedback for bounded control signal [95]. The implementation of the proposed control algorithm requires the exact model of the plant in order to guarantee the stability of the system. Still, this assumption is not always valid in real applications. Thus, one requires some adaptation mechanisms to implement the algorithm.

Input constraint problem is worse for adaptive control systems. Once the saturation is encountered, any attempt from the adaptation system to improve the control signal will fail and might lead the system to instability. In this regard, different adaptive based control systems were presented in the literature. For instance, one method suggests halting the adaptation mechanisms when the input signal gets saturated. This method works fine and guarantees reasonable control behavior; however, this will reduce the effectiveness of the adaptation mechanisms. In [96], an adaptive control system that tracks a modified reference model which takes the input saturation into account was proposed. The control algorithm is designed using the LQR technique and developed for the first order systems. Another attempt to solve the actuator's limited authority is explained in [97] and referred to as pseudo control hedging (PCH) technique. Basically, the method suggests feeding the expected system response back to the reference system when a saturation limit is activated. This approach overcomes the traditional one, as it will allow the adaptation mechanism to proceed without shutting it off.

Chapter 3: Quadrotor System

Quadrotors have been increasingly used as a preferred UAV platform for various applications due to their unique features. These features include aggressive maneuverability, vertical takeoff/landing (VTOL) ability, low cost and low maintenance. In this chapter, a description of a typical design of a quadrotor system is given. This includes reference frames, dynamic equations, system components and the simulation model. For the dynamic model, only the dynamic equations are described as the complexity of the control problem is well addressed in Chapter 2.

This chapter is organized as follows. Section 3.1 describes the basic concepts related to the structure and movements of the quadrotor system. In addition, it presents the forces and torques applied to the system. The reference frames and system states are described in Section 3.2. Section 3.3 derives the nonlinear dynamic equations of the quadrotor system in linear and rotational domain. In addition, it includes a comparison of different representation of the attitude dynamics. Section 3.4 describes the typical sensors used in UAVs. Finally, Section 3.5 and Section 3.6 present the simulation model developed by Simulink software and the testbed system designed to validate the control results in later chapters, respectively.

3.1 Basic Concepts and Movements

A typical quadrotor actuation mechanism is integrated into a multi-rotor cross-platform composed of four symmetrically configured rotors with propellers. The symmetrical platform allows for centralization of the payload and control system. The propellers' fixed-pitch allows the rotors to create a downward force while their axes of rotation are fixed and parallel to each other. However, adjacent rotors must rotate in an opposite direction to counterbalance the spinning torque of each rotor on the quadrotor and omit the need for a tail rotor. Specifically, the right and left rotors must rotate in an opposite direction to the front and rear ones. By alternating

the rotational speeds of the rotors, one can control the quadrotor to reach a feasible altitude and attitude [76].

3.1.1 Rotors

A quadrotor typically uses brushless DC-motor as its actuators. Each motor is connected to the controller through an electrical speed controller (ESC) that controls the speed of the rotor. Using a typical ESC, the rotors can spin only in one direction either clockwise or counterclockwise. In this way, the rotational speed of all motors is always considered to be positive. The rotation speed of each motor is denoted by $\Omega_i \in \Re$ in $[\mathrm{rad/s}]$, where i indicates the rotor number $i = \{1,2,3,4\}$ and selected as the front, right, rear and left rotor, respectively. The numbers of the rotors are selected in clockwise order starting from the rotor placed at the direction of the x-axis. Due to the rotation of each rotor system, different force and moments are generated.

Thrust force f_i: this force acts in the direction of the z-axis in the B-frame and directly proportional to the rotor's rotation speed Ω and defined as:

$$f_i = K_b {\Omega_i}^2 \tag{3-1}$$

where K_b is the thrust coefficient in $[Ns^2]$.

Drag moments m_i: this moment is generated by the rotation of the propeller and acts on the vehicle body in the same direction of rotation. It acts around the z-axis in the B-Frame and expressed as:

$$m_i = K_d f_i \tag{3-2}$$

where K_d is the thrust to drag ratio.

Moments due to rotors' force τ_i: this moment is generated due to the rotor thrust and the arm length of the vehicle around the center of gravity. It is expressed as:

$$\tau_i = l f_i \tag{3-3}$$

where l is the distance from the rotors' center to the quadrotor's center of mass in $[m]$.

Figure 3-1 illustrates the force and moments generated by a single rotor and their effects on the quadrotor system.

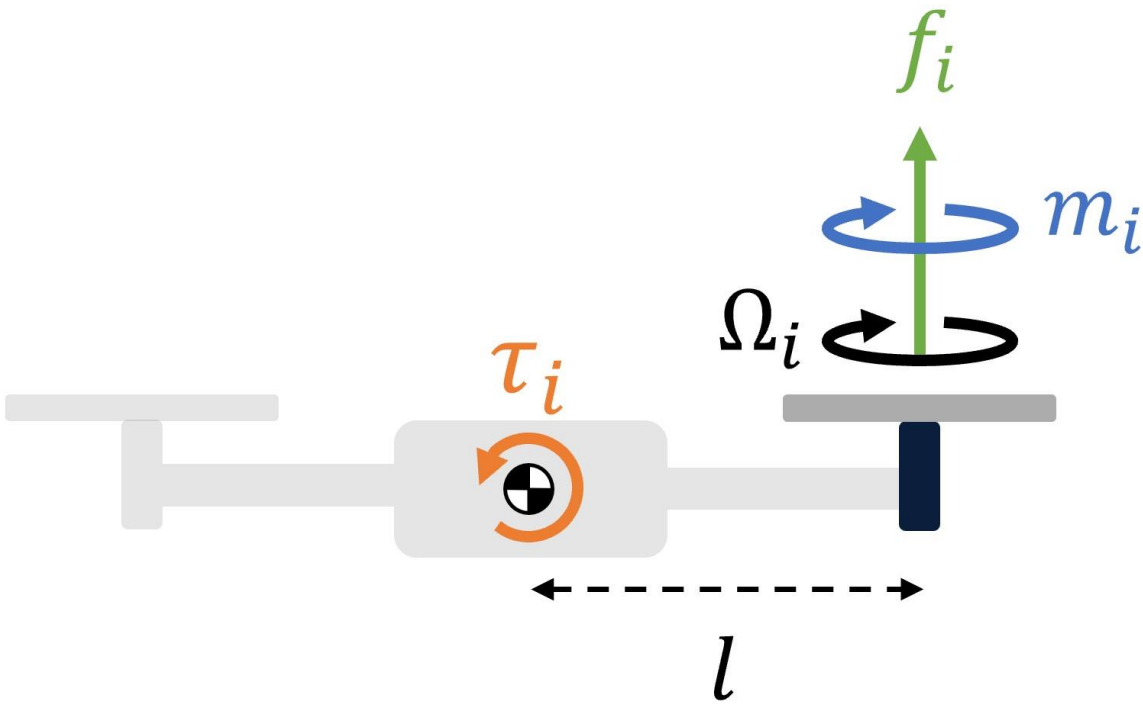

Figure 3-1: Side view of quadrotor showing thrust and moments generated by a single rotor

Since the rotor dynamics has a low influence on the resulting system behavior, it was omitted from the system dynamics [98].

3.1.2 Basic Movements

As an underactuated system, a quadrotor lacks enough control signals to control all of its degrees of freedom independently. Hence, reaching any set point in space is not possible. As a result, it is important to select four variables that will make the quadrotor easier to control. Those variables could be controlled using different movements defined as throttle, roll, pitch and yaw. Controlling the movements is done by alternating the velocities of its rotors which generates torques and forces around and along the system's axes. Subsequently, the quadrotor can reach a desired altitude and attitude. Table 3.1 provides a description of each movement.

Table 3-1: Description of Quadrotor's Movements

Movement	Description	Method
Throttle	Generates a vertical force along the z-axis with respect to B-frame, which leads to raising or lowering the quadrotor.	Increasing (or decreasing) the speed of all propellers by the same amount.
Roll	Generates a torque around x-axis with respect to B-frame. This leads to the roll angle acceleration.	Increasing (or decreasing) the speed of the left propeller, and by decreasing (or increasing) the speed of the right one.
Pitch	Generates a torque around y-axis with respect to B-frame. This leads to the pitch angle acceleration.	Increasing (or decreasing) the speed of the rear propeller and by decreasing (or increasing) the speed of the front one.
Yaw	Generates a torque around z-axis with respect to B-frame. This leads to the yaw angle acceleration.	Increasing (or decreasing) both the speed of both the front and the rear propeller, while decreasing (or increasing) the speed of both the right and left ones.

Figure 3-2 illustrates a quadrotor vehicle performing a vertical movement. The direction and thickness of the curved arrows represent the rotation direction and magnitude of each rotor, respectively. The green arrows represent the thrust force generated by each rotor while the red arrows represent the resulting movement.

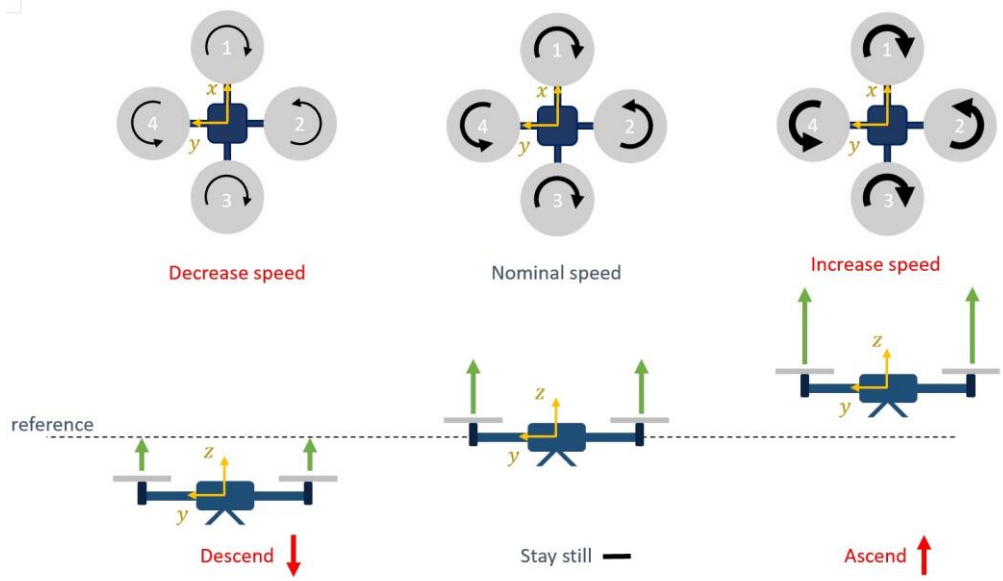

Figure 3-2: Quadrotor performing a vertical movement

Figure 3-3 shows a basic horizontal movement of a quadrotor. The green arrows represent the thrust force generated by each rotor while the red arrow represents the resultant movement.

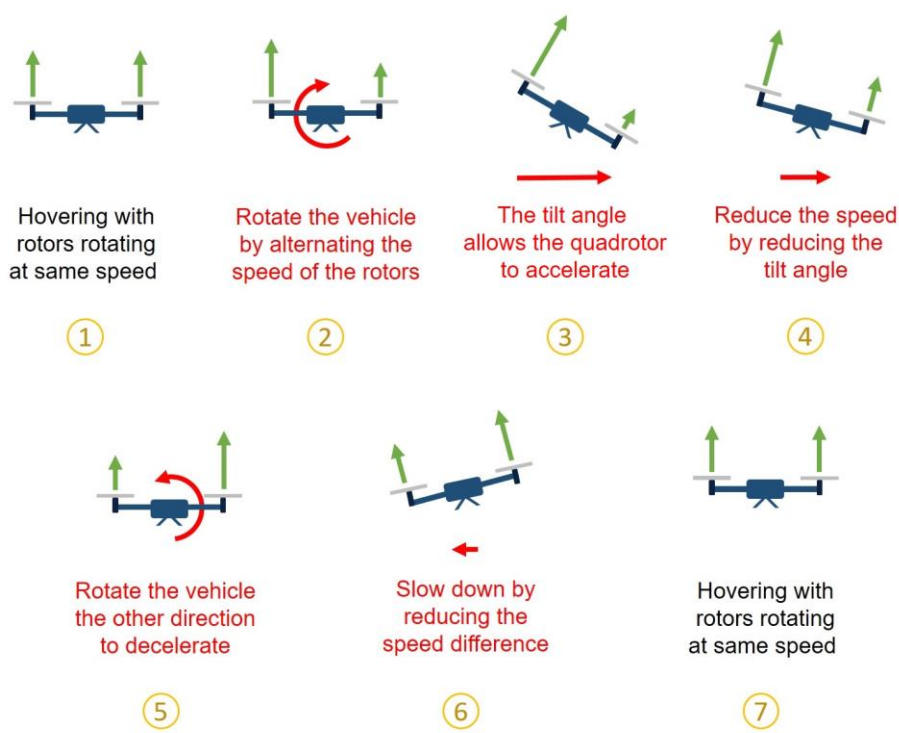

Figure 3-3: Basic horizontal movement of quadrotor vehicle

3.2 Reference Frames

In general, understanding the reference frames and rotation matrices transform states from one configuration space to another is very important in modeling and controlling mechanical systems. Having proper transformation matrices can simplify relationships. On the one hand, the equations of motion, aerodynamic forces and input torques are defined in the body frame of reference. In addition, the readings of onboard sensors including the GPS, accelerometers, rate gyros and magnetometer are obtained with respect to the body frame of reference. On the other hand, the command signals representing the desired signals are given in the earth frame of reference [99]. This section outlines the reference frames which are used to describe the position, orientation and rotation matrix.

In order to describe the motion of a 6 DOF rigid body, two reference frames are defined; earth inertial reference denoted by (E-frame) and body frame of reference denoted by (B-frame). The E-frame is an inertial right-hand reference symbolized by (O_E, x_E, y_E, z_E), where O_E is the axis origin and (x_E, y_E, z_E) are in NWU configuration with respect to the earth. Using the E-frame, the linear position of the center of gravity ξ and the ZYX Euler angles Θ are defined. The B-frame attached to the body of the quadrotor is a right-hand reference frame denoted by (O_B, x_B, y_B, z_B), where O_B is the axis origin and coincides with the center of the quadrotor's cross structure and (x_B, y_B, z_B) pointing towards the front, left and up, respectively. The linear velocity V, the angular velocity ω, the torques Γ are defined using the B-frame.

The linear position ξ is determined using the vector between the origins of the B-frame and E-frame represented in the E-frame. The Euler angles $\Theta = [\phi \quad \theta \quad \psi]^T$ representing the attitude of the quadrotor are defined by the orientation of B-frame with respect to the E-frame. A rotation matrix $R \in \mathbb{R}^{3\times3}$ is needed to map the orientation of a vector from B-frame to E-frame expressed in E-frame [100]. The rotation matrix is a special orthogonal group (SO(3)) and has the following properties:

$$R^T R = I_{3\times3}, \qquad det(R) = 1 \tag{3-4}$$

Using R, the direction of the ith axis of B-frame represented by its ith column is given by

$$b_i = Re_i, \qquad i = \{1, 2, 3\} \tag{3-5}$$

$$e_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \qquad e_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \qquad e_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{3-6}$$

Figure 3-4 illustrates quadrotor's structure and reference frames. The E-frame and B-frame are shown in black and red, respectively. Each propeller is represented by two arrows; straight and curved. The straight arrow represents the thrust force generated by the rotation of the propeller while the curved one represents the direction of rotation. The green curved arrows represent the attitude angles of the quadrotor. They are defined as the rotation of the B-Frame around E-frame with respect to the B-frame. The orange arrow represents the position vector.

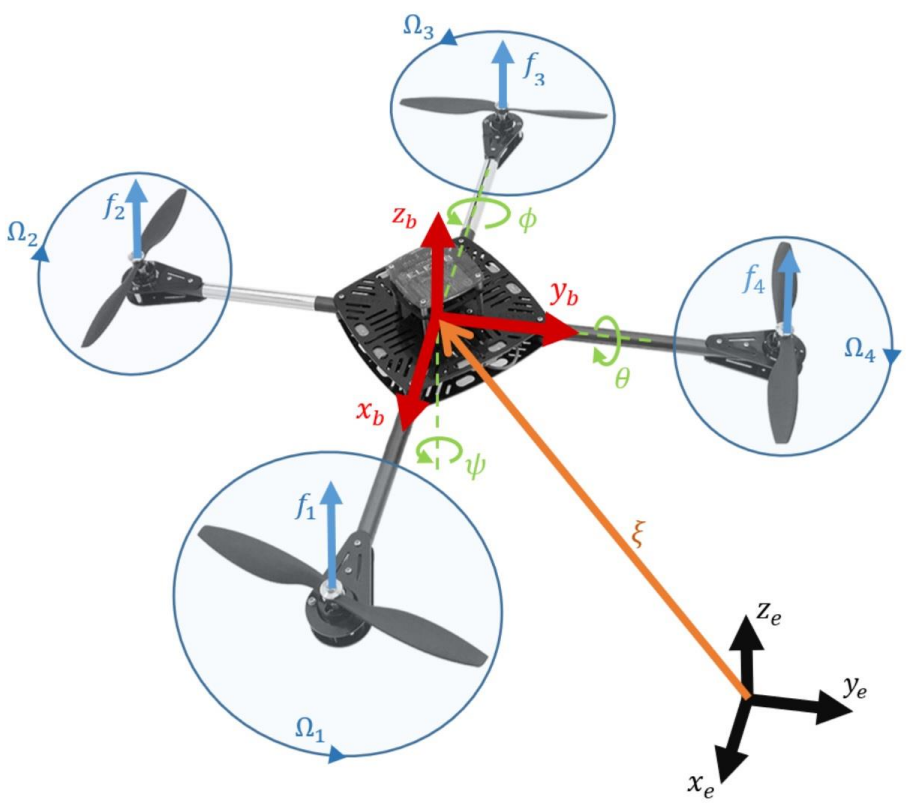

Figure 3-4: Quadrotor's structure and references frame

3.3 Newton-Euler Model

There are different models that can be used to represent the dynamic equations of a quadrotor depending on the assumptions and simplifications made to the models. As an example, taking the aerodynamics of the system into consideration or neglecting it will make a big difference in the dynamic equations. In addition, the quadrotor dynamics can be expressed in different frames, i.e., E-frame or B-frame. The desired frame to represent the dynamics is selected in terms of simplicity and assumptions. A good introduction to the dynamic model of the VTOL vehicle can be found in [101] and [102] where a common representation of the dynamics is considered. In this work, Newton-Euler formulation has been used to represent the equations of 6 DOF rigid body. The equations are represented in mixed frames which will be explained later. However, to make the body equations simpler, common assumptions have been made as follows:

1. The origin of the B-frame O_B coincides with the center of mass (COM) of the body.
2. The axes of the B-frame coincide with the body principal axes of inertia. This can be used later to simplify the inertial matrix to have a diagonal form.
3. The speeds of the propellers are controlled by high bandwidth motors.
4. The motor speed is unaffected by the quadrotor motion.
5. The overall propellers rotational inertia is much smaller than the vehicle inertia.

3.3.1 Linear Dynamics

The linear dynamics are expressed in the E-Frame. Although the forces generated by the rotors are expressed in B-Frame, the command signals are expressed easier in E-frame. From the Euler's first axioms of the Newton's second law

$$m\,\ddot{\xi} = \sum F \tag{3-7}$$

where $m \in \Re$ represents the quadrotor's mass in $[kg]$, $\xi \in \Re^3$ is the linear position wrt E-frame in $[m/s^2]$ and $\sum F \in \Re^3$ is a vector force resemble the summation of all forces action on

quadrotor's system with respect to E-frame in $[N]$. Generally speaking, there are two types of forces acting on the vehicle's body and described as follows:

1. Gravitational force F_g: The vehicle weight acts along the direction of z-axis in the E-frame and defined as:

$$F_g = -mge_3 = -\begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \tag{3-8}$$

where g is the gravitational acceleration in $[m/s^2 kg]$ and e_3 is the direction of the third axes of E-frame.

2. Total Thrust Force F_t: Due to the rotation of each rotor system, a thrust force is generated and denoted by f_i. This force acts in the direction of the z-axis in the B-frame and directly proportional to its rotation speed Ω.The total thrust acting on the body is equal to the summation of all thrust forces generated by the four rotors and defined as follows:

$$F_t = \sum_{i=1}^{4} f_i = K_b \sum_{i=1}^{4} {\Omega_i}^2 \tag{3-9}$$

Thus, the linear acceleration equation defined in (3-7) can be expressed as follows

$$m\,\ddot{\xi} = F_g + Re_3 F_t = -mge_3 + Re_3 \sum_{i=1}^{4} f_i \tag{3-10}$$

The rotation matrix R is used here to map the thrust forces from the B-frame to its equivalent in the E-frame.

3.3.2 Rotational Dynamics

The rotation dynamics are expressed in the B-Frame. This makes it easier to express all moments and desired states. From the Euler's second axioms of the Newton's second law

$$J\dot{\omega} + \omega \times J\omega = \sum M \tag{3-11}$$

where $J \in \Re^{3\times3}$ is the quadrotor's inertial matrix in $[Kg\ m^2]$, $\omega \in \Re^3$ is the angular velocity with respect to the B-frame in $[rad/s^2]$, the second term of the equation expresses the cross-coupling of the angular momentum in the system and $\sum M \in \Re^3$ represents the summation of all moments applied on the quadrotor's system with respect to the B-frame in $[Nm]$.

There are three different torques acting on the vehicle's body. They are generated directly by the rotation of the rotor systems and proportional to their rotation speed. These torques are described as follows:

1. Moment around x-axis M_x: this generated due to the rust difference between the two rotors lies along the x-axis. It is proportional to their rotational speed and the quadrotor's arm length. It is expressed as follows

$$M_x = \tau_4 - \tau_2 = K_b l\left({\Omega_4}^2 - {\Omega_2}^2\right) \tag{3-12}$$

2. Moment around y-axis M_y: this generated due to the thrust ddifference between the two rotors lies along the y-axis. It is proportional to their rotational speed and the quadrotor's arm length. It is expressed as follows

$$M_y = \tau_3 - \tau_1 = K_b l\left({\Omega_3}^2 - {\Omega_1}^2\right) \tag{3-13}$$

3. Moment around z-axis M_z: this generated due to the difference in the reaction moments caused by the rotation of the rotors. It is proportional to their rotational speed and is expressed as follows

$$M_z = m_1 - m_2 + m_3 - m_4 = K_b K_d\left({\Omega_1}^2 - {\Omega_2}^2 + {\Omega_3}^2 - {\Omega_4}^2\right) \tag{3-14}$$

The overall moments acting upon the body can be represented in a compact form as follows

$$\begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} = \begin{bmatrix} 0 & l & 0 & -l \\ -l & 0 & l & 0 \\ K_d & -K_d & K_d & -K_d \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} \tag{3-15}$$

3.3.3 Attitude Dynamics

Through the literature, there are different models that can be used to represent the attitude dynamic equations of a UAV. These different representations depend on the simplifications made to the models. One of most used representation due to its easiness to comprehend is based on Euler-angles Θ. The rotation matrix R can be expressed using the Euler angles as follows

$$R = \begin{bmatrix} c_\theta c_\psi & -c_\phi s_\psi + c_\psi s_\phi s_\theta & s_\phi s_\psi + c_\phi c_\psi s_\theta \\ c_\theta s_\psi & c_\phi c_\psi + s_\phi s_\theta s_\psi & -c_\psi s_\phi + c_\phi s_\theta s_\psi \\ -s_\theta & c_\phi s_\theta & c_\phi c_\theta \end{bmatrix} \tag{3-16}$$

where $c_x = cos(\mathrm{x})$, $s_x = sin(\mathrm{x})$ and $t_x = tan(\mathrm{x})$. Using a transfer matrix defined in [100], the relation between the Euler angles rates $\dot{\Theta}$ in the E-frame and the angular velocity ωin the B-frame is introduced as follows

$$\dot{\Theta} = T\omega \tag{3-17}$$

$$T = \begin{bmatrix} 1 & s_\phi t_\theta & c_\phi t_\theta \\ 0 & c_\phi & -s_\phi \\ 0 & \frac{s_\phi}{c_\theta} & \frac{c_\phi}{c_\theta} \end{bmatrix} \tag{3-18}$$

However, this representation suffers from singularity at $\theta = \pm\frac{\pi}{2}$ i.e., when the quadrotor is on its side. This case can occur while quadrotor is performing aggressive maneuvers, such as flipping, or reacting to severe conditions, such as system failure.

It is worth to mention that earlier works on UAV systems were interested mostly on near hovering flight, where the pitch and roll angles are small (<15°). Thus, they tend to use small angle approximation and simplify the transfer matrix, i.e. $T = I_{3\times3}$. Consequently, the relation between the Euler angle rates and angular velocity become linear i.e., $\dot{\Theta} = \omega$**.**

Recently, the focus of researchers is shifted towards using global representation of attitude dynamics that can be used for complex flight maneuvers. One way to do this is using a

special Euclidean group (SE(3)) to represent the attitude [20]. Using this representation, the attitude dynamic equation is expressed as

$$\dot{R} = R\omega_{\times} \tag{3-19}$$

The times map $\cdot_{\times}: \mathbb{R}^3 \to SO(3)$, known also as skew matrix, is defined by the condition that $a_{\times}b = a \times b$ for all $a, b \in \mathbb{R}^3$, the matrix $a_{\times}$ is given by

$$a_{\times} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \tag{3-20}$$

Where the inverse of times map is denoted by the *vee* map $\cdot^{\vee}: SO(3) \to \mathbb{R}^3$ that is $(a_{\times})^{\vee} = a$.

3.4 Sensors

In general, sensors are fundamental to identify the values of the states and to perform feedback control. Furthermore, they can be deployed in board applications, such as obstacle avoidance, auto-landing, and target following. However, some states cannot be directly measured. Thus, some techniques need to be used in order to measure their values, such as state estimation algorithms or filters like Kalman filter. The following part introduces the most common sensors used in UAV applications:

1. Gyroscope: this sensor measures the rate of rotation around a single or multiple axis. This sensor is typically used to estimate the attitude angles and stabilize the UAV system.
2. Accelerometer: this sensor measures the body acceleration of the vehicle. Furthermore, it can be used to measure the orientation of the quadrotor with respect to earth. That is, the sensor gives a direct reading for the angle, unlike the gyroscope sensor. Thus, it is usually used as a reference for the attitude angles in hovering condition; however, in motion, the body acceleration affects the accelerometer reading.

3. Magnetometer: this sensor is used to measure the heading angle of the quadrotor with respect to the true north. Hence, it is used along with the accelerometer sensor to give a full reference solution for the attitude angles.
4. IMU: this type of device consists of multiple 3-axis sensors; a 3-axis accelerometer, a 3-axis gyroscope, as well as, a 3-axis magnetometer. This device is used in order to improve the orientation measurement and estimation of the quadrotor states.
5. Range sensors: this type of sensor is mainly used for altitude measurement and control, as well as, in several applications such as obstacle avoidance. An example of this type of sensor is the ultrasonic sensor which is usually used despite its small range and low accuracy. Another example is the laser range finder which usually selected for its higher accuracy.
6. GPS: the main usage of GPS sensor is in tracking and localization of autonomous vehicle in an outdoor environment. In some applications, an integration of accelerometer measurements and GPS data are needed to get a better estimation of the position and velocity of the quadrotor system.
7. Camera: this device is commonly used on a quadrotor and usually used in advance applications. It provides a video feedback that can be used for image recognition and processing as well as obstacle avoidance and auto-landing.

3.5 Simulink Model

Through this thesis, the results of the proposed controllers are carried out through a simulation built using MATLAB SIMULINK® software. The purpose of the simulation model is to show the overall performance of the proposed controller in an ideal environment. The simulation tests were carried out assuming a small-size quadrotor with parameters as shown in Table 3-2.

Figure 3-5 and Figure 3-6 show the overall simulation and the quadrotor model carried out through SIMULINK® software, respectively.

Table 3-2: System parameters

Parameter	Description	Value	Unit
$\boldsymbol{I_{xx}}$	Inertia around x-axis	0.01	$\mathrm{kg\,m^2}$
$\boldsymbol{I_{yy}}$	Inertia around y-axis	0.01	$\mathrm{kg\,m^2}$
$\boldsymbol{I_{zz}}$	Inertia around z-axis	0.1	$\mathrm{kg\,m^2}$
$\boldsymbol{m}$	System mass	1	kg
$\mathbf{l}$	Arm length	0.5	m
$\boldsymbol{K_b}$	Thrust coefficient	5×10^{-4}	$\mathrm{N\,s^2\,rad^{-2}}$
$\boldsymbol{K_d}$	Drag to thrust ratio	0.1	m
$\boldsymbol{dt}$	Sampling time	0.001	s

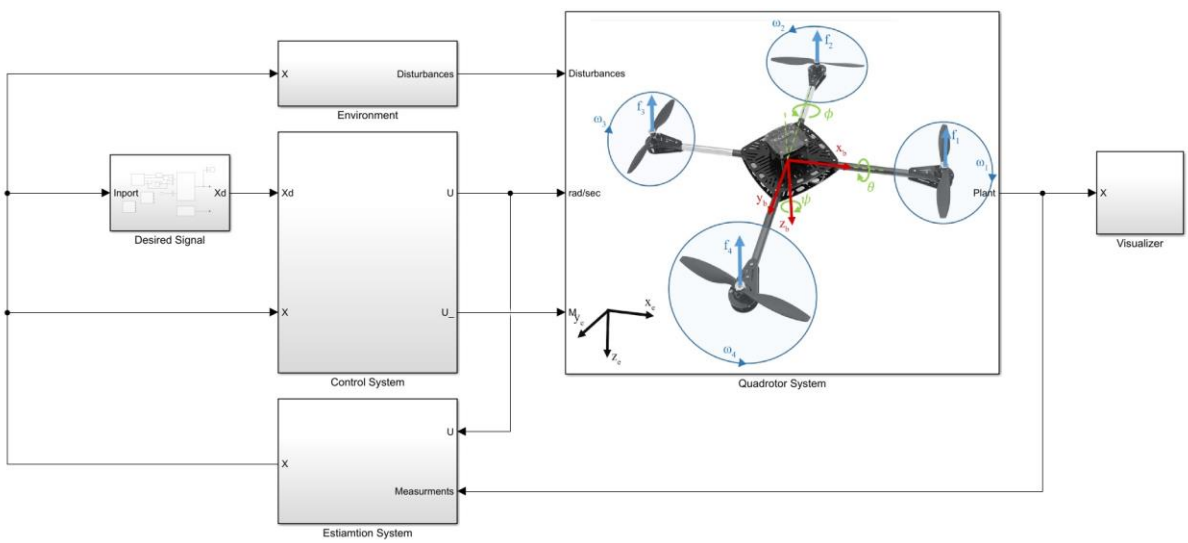

Figure 3-5: Overall simulation of the testing system carried out through Simulink

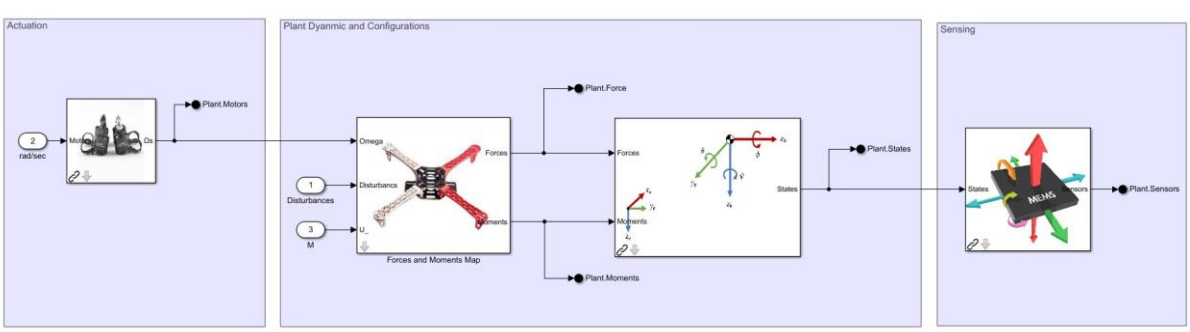

Figure 3-6: Quadrotor model carried out through Simulink

3.6 Experimental Setup

In order to experimentally validate the proposed control algorithms, a testbed system has been designed and developed. The testbed uses the Robot Operating System (ROS) environment to facilitate easy integration of the required sensors, estimation, control algorithms, and communication with the system. The testbed system consists mainly of five parts, a mechanical skeleton, sensors, actuators, a processor and a ground station. Figure 3-7 shows the testbed. The testbed is developed to resemble the angular dynamics of a quadrotor vehicle. The body frame of the quadrotor is mounted on a 3 DOF pivot joint, which allows the quadrotor to rotate about its x-, y- and z-axes. The mechanism allows the frame to have a maximum of ±35 degrees around both x- and y-axes and ±360 degrees around the z-axis. The platform supported by an aluminum base with 40 cm long and 5 cm diameter pipe. The system is mounted from the ceiling to reduce the ground effects generated by the quadrotor vortex. The testbed consists of four arms with 30 cm in length. At the end of each arm, a brushless DC motor is attached with a propeller and controlled using an Electrical Speed Controller (ESC) unit. The propeller axes of rotation are fixed and parallel to each other. All propellers have fixed-pitch blades and their airflows are generated downwards to create an upward lift. The system state variables are controlled by alternating the propellers velocities.

For the purpose of measuring the vehicle states, the testbed is equipped with various onboard sensors. In the setup, two inertial measurement unit (IMU) sensors and three joint encoders have been used. Each IMU sensor consists of 3-axis accelerometers, 3-axis gyroscopes, and 3-axis magnetometers. The reasons for using two IMU sensors are to improve the measurement accuracy and system redundancy. The joint encoders can accurately measure the platform's rotation angles which are used to obtain the orientation of the testbed with the relatively low noise level and micro degree accuracy at up to 1 kHz. In our experiments, the encoder readings are used for the purpose of estimation comparisons as they serve as the ground truth measurements.

In terms of onboard computing power, the testbed contains a Raspberry Pi 3® (RPi3) microprocessor installed at the surface of the testbed and capable of running the closed loop controller algorithm. The RPi3 is running Raspbian OS on a quad-core 64-bit ARM Cortex A53 running at 1.2 GHz. The proposed algorithms were written using C++ and standard libraries. The board is supported with an input-output extension that allows for it to control 16 PWM channels with 12-bit resolution used to the controller the ESC. A single computer, serves as the ground station, running ROS environment, which generates the command orientation signals and displays the testbed states. The onboard microprocessor is connected wirelessly to the ground station through WiFi. The communication gives the user the ability to control the testbed's orientation through a graphical user interface (GUI) and visualize the vehicle states using GAZEBO software. During this experiment, the date interchange was running at a 100 Hz.

Figure 3-7: An illustration of the testbed

Figure 3-8 shows a graphical representation of the overall architecture of the testbed system.

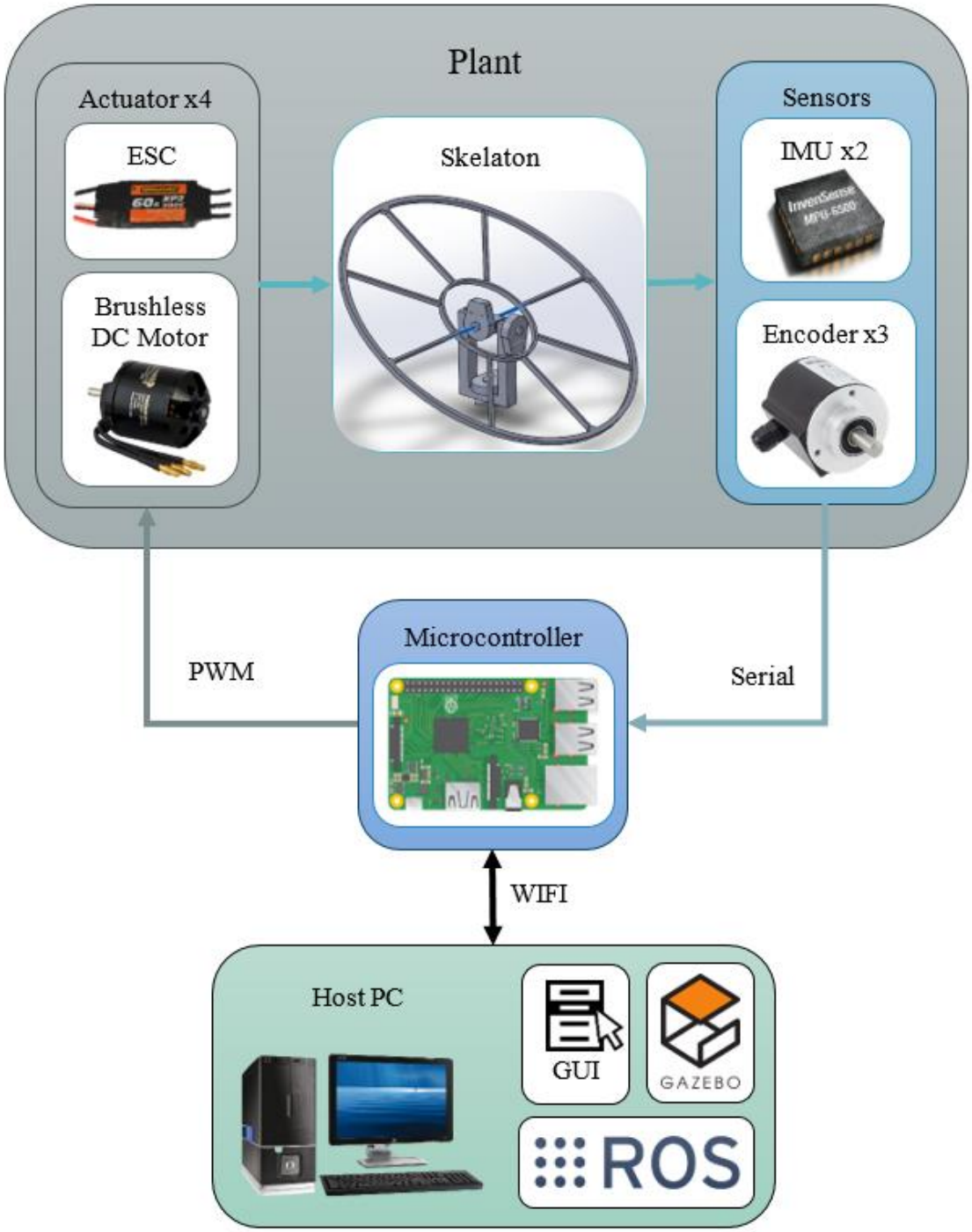

Figure 3-8: A graphical representation of the overall architecture of the experimental setup

Chapter 4: Global Attitude Tracking Control of Quadrotor

This chapter presents a global nonlinear tracking control system for the attitude dynamics of quadrotor unmanned aerial vehicles (UAVs). The proposed controller is developed directly on the special Euclidean group with a region of attraction covering the entire configuration space globally. It guarantees the asymptotic convergence of tracking error in the presence of underactuation, uncertainties and external disturbances. In particular, the control method consists of a dynamic surface control (DSC) and a non-parametric adaptation mechanism.

This chapter is organized as follows. An overview of the proposed controller is presented in Section 4.1. Section 4.2 introduces the problem formulation and the assumptions. The development of the control system is described in detail in Section 4.3. The section starts by splitting the attitude dynamics into two systems and then presents their proposed controller. Moreover, the proposed adaptation mechanism and the controller stability are presented in this section. Finally, the desirable features of the proposed control system are illustrated by both numerical simulation and experiments on a UAV testbed in Section 4.4.

4.1 Overview

In the last decade, researchers have been proposing different solutions to deal with the control challenges of quadrotor systems. More recent attempts for controlling quadrotor involve the use of backstepping control (BC) [19], [60], [103]. In general, the BC approach is used to address complexities resulting from the coupled and cascaded dynamics of a system. The BC design involves stabilizing the system states through a step-by-step recursive process that guarantees the stability of the entire system. However, the use of BC can be problematic because of “explosion of terms” in its control law. Despite this challenge, BC remains one of the commonly used approaches for quadrotor control in the literature. In this chapter, we utilize an alternative control technique called dynamic surface control (DSC). Similar to the BC method,

this technique uses a step-by-step recursive process utilizing multiple sliding surfaces to stabilize the dynamic system. However, it avoids the drawback of BC that is the “explosion of terms” when it is applied to highly complicated systems. Specifically, it does that by incorporating a series of first-order low-pass filters into the recursive process [104].

Through the literature, numerous control systems have been proposed for quadrotors with different representations of the UAV attitude [12]. In some cases, the control design is based on local coordinates for simplicity [69]. However, this restricts the ability of the control system to achieve complex maneuvers, which is critical for multirotor vehicles. In most cases, the control systems are based on Euler-angles as their attitude representation [77], [79], [90]. This allows the control system to achieve better stability and performance than that based on local coordinates. Nonetheless, the control system still suffers from singularities in their attitude representation. Furthermore, the control law will involve complicated expressions with trigonometric functions, which cannot be linearized for complex maneuvers. Other control approaches are based on quaternion for their attitude representation. Quaternion-based control systems are preferable over Euler-angle based ones, as they do not suffer from singularities in their representation [72]. Still, quaternion representations can cause ambiguity, as one attitude can be represented by two different representations. Unless this ambiguity is resolved by the control system, it will lead to undesired behavior where the quadrotor unnecessarily intends to rotate a large angle. In this chapter, the design of the control systems is based on the special Euclidean group to represent the attitude [88]. Although this representation is not as compact as the quaternion representation, it has been favored in recent work since it is a globally valid expression even for complex maneuvers while it avoids singularities and ambiguities without complex mathematical expressions.

This chapter aims to design a globally robust tracking controller of a quadrotor attitude in the presence of underactuation, uncertainties and external disturbances. The proposed controller is developed directly on the special Euclidean group with a region of attraction

covering the configuration space globally. It guarantees the asymptotical convergence of tracking error in the presence of external disturbances and uncertainties without a priori knowledge of their bounds. The stability of the proposed controller is proven within the region of interest. In particular, the control method combines three techniques: a dynamic surface control (DSC), and non-parametric adaptation mechanism. The DSC guarantees the attitude dynamics stability globally and tracking performance while avoiding the mathematical complexities associated with the highly nonlinear dynamics. The adaptation mechanism includes a radial basis function neural network (RBF-NN) to observe uncertainties without the need for prior training.

4.2 Problem Formulation

In general, quadrotor systems suffer from uncertainties in their parameters and models which may cause inaccuracy or instability for the control system. Thus, the control systems require having some robustness to overcome the effect of uncertainties and adaptation mechanism to improve the system response over time. One can consider a nonlinear system described by (3-11) and (3-19) and rewrite it as follows

$$\dot{R} = F_R(R, \omega) \tag{4-1}$$

$$\dot{\omega} = F_\omega(\omega) + \theta_\omega u_\omega + D_\omega \tag{4-2}$$

where

$$F_R(R, \omega) = R\omega_\times \tag{4-3}$$

$$J^{-1}(d_\omega - \omega \times J\omega) = F_\omega + D_\omega \tag{4-4}$$

$$F_\omega(\omega) = -J_0^{-1}\,(\omega \times J_0\omega\,) \tag{4-5}$$

$$D_\omega = \triangle\, F_\omega + J^{-1}d_\omega \tag{4-6}$$

$$\theta_\omega = J^{-1}\begin{bmatrix} K_b & 0 & 0 \\ 0 & K_b & 0 \\ 0 & 0 & K_d \end{bmatrix} \tag{4-7}$$

$$u = \begin{bmatrix} u_p \\ u_q \\ u_r \end{bmatrix} = \begin{bmatrix} 0 & -l & 0 & l \\ -l & 0 & l & 0 \\ -1 & 1 & -1 & 1 \end{bmatrix} \begin{bmatrix} {\Omega_1}^2 \\ {\Omega_2}^2 \\ {\Omega_3}^2 \\ {\Omega_4}^2 \end{bmatrix} \tag{4-8}$$

where $u(t) \in \mathbb{R}^3$ is the input vector, $y(t) = R \in \mathbb{R}^{3\times3}$ is the output, $x(t) = [R,\ \omega]^T \in \mathbb{R}^{12}$ is the state vector $F_\omega \in \mathbb{R}^3$ are the state function, $\theta_\omega \in \mathbb{R}^{3\times3}$ is the system uncertainty and $D_\omega \in \mathbb{R}^3$ is the lumped uncertainty in the system model including unmodeled dynamics and external disturbances. This chapter aims to design a robust motion controller based on an adaptive neural network for a 6-DOF quadrotor to follow a designed path in presence of the system underactuation, parameter uncertainties, model uncertainties and disturbances under the following assumptions:

1. All of the state variables, $x(t)$, are measurable.
2. θ_ω represents unknown constant parameters with known sign and lower threshold $\bar{\theta}_\omega$.
3. Upper bounds for the vehicle uncertainties $D_\omega(t)$ are assumed $\|D_\omega(t)\| \leq D_{\omega_{max}}$. This assumption can be satisfied by a nonlinear model with a good level of accuracy.
4. Under the above assumptions, the system stability can be guaranteed using the proposed control systems.

4.3 Control Development

The attitude dynamics is a fully actuated system, it has 3 DOF and 3 input signals i.e., $u_\omega \in \mathbb{R}^3$. The objective of the control system is to guarantee the convergence of the attitude, represented by a rotation matrix R, toward the rotational command signal R_c. Furthermore, the control system has to guarantee system stability and tracking performance in the presence of uncertainties and disturbances. This will be accomplished using DSC technique. Figure 4-1 illustrates the block diagram of the attitude control system. The attitude control is divided into two steps: angle control and angular rate control.

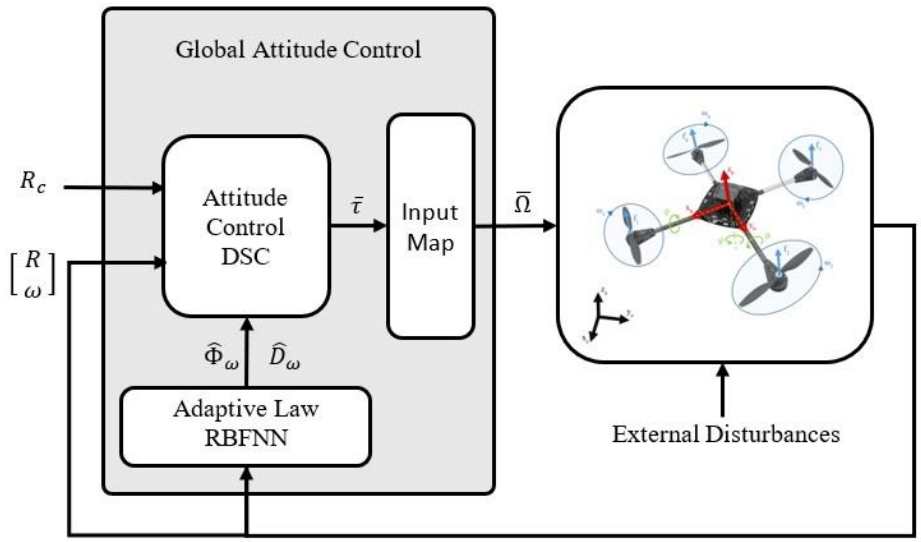

Figure 4-1: Block diagram of the attitude control system.

4.3.1 Attitude Tracking Control

To insure the smoothness of the attitude command tracking signal R_c designed by the user or generated by the outer loop i.e., position control, let us introduce a first-order low-pass filter as

$$\tau_R \dot{R}_d + R_d = R_c \tag{4-9}$$

where τ_R represents the time constant of the filter which will be determined later. $R_d \in \mathrm{SO}(3)$ is the filtered signal of the command attitude. The augmented error can be defined as

$$\zeta_R = R_d - R_c \tag{4-10}$$

Proposition 1: For a given desired attitude tracking R_d and the current attitude R, the first error surface vector for attitude dynamics is defined as

$$s_R = \frac{1}{2}\left(\tilde{R} - \tilde{R}^T\right)^\vee \tag{4-11}$$

where the $\tilde{R}$ represents the rotation augmented error and is defined as

$$\tilde{R} = {R_d}^T R \tag{4-12}$$

and its derivative is calculated as

$$\dot{\tilde{R}} = \dot{R}_d{}^T R + R_d{}^T \dot{R} = \dot{R}_d{}^T R + R_d{}^T R \omega_\times \tag{4-13}$$

The time derivative of the first surface error is calculated using (4-1) as

$$\dot{s}_R = \frac{1}{2}\left(\dot{\tilde{R}} - \dot{\tilde{R}}^T\right)^\vee \tag{4-14}$$

This equation can be further simplified as

$$\dot{s}_R = \Psi_1 + \Psi_2 \omega \tag{4-15}$$

where

$$\Psi_1 = \frac{1}{2}\left(\dot{R}_d{}^T R - R^T \dot{R}_d\right)^\vee \tag{4-16}$$

$$\Psi_2 = \frac{1}{2}\left(I_{3\times3} trace\left(\tilde{R}\right) - \left(\tilde{R}\right)\right)^\vee \tag{4-17}$$

It is worth to mention that Ψ_2 is a nonlinear function of the augmented rotation error and the function inverse is defined globally. At this step, the objective is to ensure the stability of the first sliding surface that is to drive $s_R \to 0$ as $t \to \infty$. This is done by designing a suitable forcing term for the angular velocity defined as:

$$\omega_c = \Psi_2^{-1}(-\Psi_1 - K_R S_R) \tag{4-18}$$

where K_R is a positive diagonal matrix which will be defined later. When the angular velocity converges to the forcing term that is $\omega \to \omega_c$, the dynamics of the error surface becomes

$$\dot{s}_R = -K_R S_R \tag{4-19}$$

which as a result guarantees the convergence of the first error surface s_R.

4.3.2 Angular Velocity Tracking Control

Similarly, we can choose the second first-order low-pass filter for the command angular velocity as follows

$$\tau_\omega \dot{\omega}_d + \omega_d = \omega_c \tag{4-20}$$

and its augmented angular velocity error

$$\zeta_\omega = \omega_c - \omega_d \tag{4-21}$$

where τ_ω represents the time constant of the filter and will be determined later, and $\omega_d \in \mathbb{R}^3$ is the desired angular velocity.

Proposition 2: For a given desired angular velocity ω_d and the current angular velocity ω, we define the second error surface vector as follows

$$s_\omega = \omega - \omega_d \tag{4-22}$$

Its time derivative is calculated as,

$$\dot{s}_\omega = \dot{\omega} - \dot{\omega}_d \tag{4-23}$$

$$\dot{s}_\omega = F_\omega + \theta_\omega\, u_\omega + D_\omega - \dot{\omega}_d \tag{4-24}$$

At this step, the objective is to ensure the stability of the second error surface that is to drive $s_\omega \to 0$ as $t \to \infty$. This is done by designing a suitable input signal u_ω. The estimation of the parameter θ_ω and its error can be defined as

$$\tilde{\theta}_\omega = \theta_\omega - \hat{\theta}_\omega \tag{4-25}$$

where $\hat{\theta}_\omega$ is a diagonal matrix representing the estimation of the unknown parameter θ_ω and $\tilde{\theta}_\omega$ is the estimation error. Using this relation, we can now define the control signal as

$$u_\omega = \hat{\theta}_\omega^{-1}(q_\omega) \tag{4-26}$$

$$q_\omega = \dot{\omega}_d - F_\omega - \widehat{D}_\omega - K_\omega s_\omega$$

where K_ω is a positive diagonal matrix which will be defined later, and $\widehat{D}_\omega \in \mathbb{R}^3$ is the disturbance estimation defined using the RBF-NN as

$$\widehat{D}_\omega = \hat{f}(s_w) = \widehat{W}_\omega{}^T h_\omega(s_w) \tag{4-27}$$

where $\widehat{W}_\omega$ is the estimated disturbance weights. Substituting the input signal u_ω into the dynamics of the second error surface defined in (4-24) yields

$$\dot{s}_\omega = -K_\omega s_\omega + \widetilde{W}_\omega{}^T h_\omega + \epsilon_\omega + \tilde{\theta}_\omega u_\omega \tag{4-28}$$

$$\widetilde{D}_\omega = D_\omega - \widehat{D}_\omega = \widetilde{W}_\omega{}^T h_\omega + \epsilon_\omega \tag{4-29}$$

where $\widetilde{W}_\omega$ is the error in the estimated weights and their desired values of the disturbance signal. The approximation error ϵ_ω is limited and sufficiently small. As the estimation error signal

of $\widetilde{W}_\omega$ and $\tilde{\theta}_\omega$ approaches zero, the angular velocity error surface will also approach zero, *i.e.*$s_\omega \to 0$.

4.3.3 Adaptive Neural Network

In this work, radial basis functions neural network (RBF-NN) technique is used to approximate the uncertainty in the nonlinear function $f(x)$ [105]. In short, the RBF-NN algorithm is defined as

$$h_j = e^{\frac{\|x-c_j\|^2}{2{b_j}^2}} \tag{4-30}$$

$$f(x) = {W^*}^T h(x) + \epsilon \tag{4-31}$$

where x represents the inputs of the network and selected separately for each controller, $j$denotes the number of hidden layer nodes in the neural network, h define the output of the Gaussian function, the weights W^* represent the ideal neural network weights, the epsilon ϵ represents the approximation error and the output value of the NN is defined by f. In order to estimate the nonlinear function $f(x)$, we use estimated weights instead of the ideal ones

$$\hat{f}(x) = \widehat{W}^T h(x) \tag{4-32}$$

$$\widetilde{W} = W - \widehat{W} \tag{4-33}$$

where $\hat{f}$ and $\widehat{\mathrm{W}}$ represent the estimated disturbance and weights, respectively, $\widetilde{\mathrm{W}}$ defined as the error between the estimated weights and their desired values, that is

$$f - \hat{f} = \widetilde{W}^T h + \epsilon \tag{4-34}$$

4.3.4 Error Dynamics

Once the design procedure is applied, the closed-loop error dynamics needs to be derived for stability analysis. Using the definition of the error surfaces, forcing term and input signals, the nonlinear system equations for the rotation dynamics can be described as the error equations of DSC.

$$\dot{s}_R = -\dot{R}_d{}^T R + \Psi_2\left(\omega_{c\times} - s_{\omega\times} + \zeta_{\omega\times}\right) \tag{4-35}$$

$$\dot{s}_R = -K_R s_R + \Psi_2(s_\omega - \zeta_\omega) \tag{4-36}$$

and for the angular velocity as

$$\dot{s}_\omega = -K_\omega s_\omega - \widetilde{W}_\omega{}^T h_\omega - \epsilon_\omega - \tilde{\theta}_\omega u_\omega \tag{4-37}$$

The dynamics of the augmented errors defined in (4-10) and (4-21) affects the overall error dynamics of the closed-loop system. By assuming fast responses of the first-order low-pass filters, that is faster than the error surfaces and system dynamics, one can simply ignore their effects when designing the control system. Thus, the dynamics of the error surfaces becomes

$$\dot{s}_R \approx -K_R s_R + \Psi_2 s_\omega \tag{4-38}$$

$$\dot{s}_\omega = -K_\omega s_\omega - \widetilde{W}_\omega{}^T h_\omega - \epsilon_\omega - \tilde{\theta}_\omega u_\omega \tag{4-39}$$

Now, using the following Lyapunov function

$$V = \frac{1}{2}(s_R^T s_R + s_\omega^T s_\omega + \tilde{\theta}_\omega{}^T \Gamma_\omega^{-1}\tilde{\theta}_\omega + \widetilde{W}_\omega^T \gamma_\omega^{-1}\widetilde{W}_\omega)\dot{s}_R = \Psi_1 + \Psi_2\omega \tag{4-40}$$

where Γ_ω and $\gamma_\omega$are positive matrices. Taking the derivative of the Lyapunov function yields

$$\begin{aligned}
\dot{V} &= s_R\left(-K_R s_R + R_d{}^T R s_\omega\right) + s_\omega\left(-K_\omega s_\omega + \widetilde{W}_\omega{}^T h_\omega + \epsilon_\omega + \tilde{\theta}_\omega u_\omega\right) - \tilde{\theta}_\omega{}^T \Gamma_\omega^{-1}\dot{\hat{\theta}}_\omega \\
&\quad - \widetilde{W}_\omega^T \gamma_\omega^{-1}\dot{\hat{W}}_\omega \\
\dot{V} &= -K_R\|s_R\|^2 - K_\omega\|s_\omega\|^2 + s_R R_d{}^T R s_\omega - s_\omega\epsilon_\omega + \widetilde{W}_\omega{}^T h_\omega s_\omega^T + \tilde{\theta}_\omega u_\omega s_\omega^T \\
&\quad - \tilde{\theta}_\omega{}^T \Gamma_\omega^{-1}\dot{\hat{\theta}}_\omega - \widetilde{W}_\omega^T \gamma_\omega^{-1}\dot{\hat{W}}_\omega \\
\dot{V} &= -K_R\|s_R\|^2 - K_\omega\|s_\omega\|^2 - s_\omega\epsilon_\omega + s_R R_d{}^T R s_\omega + \widetilde{W}_\omega{}^T\left(h_\omega s_\omega^T - \gamma_\omega^{-1}\dot{\hat{W}}_\omega\right) \\
&\quad + \tilde{\theta}_\omega\left(u_\omega s_\omega^T - \Gamma_\omega^{-1}\dot{\hat{\theta}}_\omega\right)
\end{aligned} \tag{4-41}$$

In order to guarantee the stability and avoid the singularity in the control signal, we can choose the following adaptation function

$$\dot{\hat{W}}_\omega = \gamma_\omega\, h_\omega s_\omega^T \tag{4-42}$$

$$\alpha_\omega = \Gamma_\omega u_\omega s_\omega^T \tag{4-43}$$

$$\dot{\hat{\theta}}_\omega = \begin{cases} -\eta_\omega \alpha_\omega & a_\omega > 0 \\ -\eta_\omega \alpha_\omega & \alpha_\omega \leq 0 \;\&\; \hat{\theta}_\omega > \bar{\theta}_\omega \\ -\eta_\omega & \alpha_\omega \leq 0 \;\&\; \hat{\theta}_\omega \leq \bar{\theta}_\omega \end{cases} \tag{4-44}$$

where $\bar{\theta}_\omega$ represent the lower limit of the estimated parameter and $\hat{\theta}_\omega(0) > \bar{\theta}_\omega$. Then the derivative becomes

$$\dot{V} = -K_R \|s_R\|^2 - K_\omega \|s_\omega\|^2 + s_R {R_d}^T R s_\omega - s_\omega \epsilon_\omega \tag{4-45}$$

$$\dot{V} \leq K_R \|s_R\|^2 - K_\omega \|s_\omega\|^2 - s_\omega \epsilon_\omega \tag{4-46}$$

Using the RBF, the approximation error ϵ_ω is limited and sufficiently small. Therefore, $(\dot{V} < 0)$ and it guarantees exponential convergence for the error surface s_R and s_ω which, as a result, guarantees the convergence of tracking error $R \to R_d$.

4.4 Results

This section demonstrates the results of the proposed controller through a simulation software and experimentally using a testbed. The purpose of the simulation is to show the overall performance of the proposed controller in an ideal environment, while the experimental experiments show the feasibility of the proposed controller in real-life implementation.

4.4.1 Numerical Simulation

The simulation shows that the performance of the proposed controller can exceed these approaches in terms of accuracy and convergence rate under the presence of uncertainties and disturbances. The simulation tests were carried out for a small-size quadrotor with designed parameters as shown in Chapter 3. Table 4-1 shows the control parameter used in the simulation tests.

The simulated tests illustrate the response of the proposed controller under different conditions including the presence of unmodeled dynamics, parameter uncertainties and external disturbance. The desired attitude trajectory is defined as a train of pulses. In addition, the

quadrotor will be subjected to external disturbances i.e., wind and gust along all its axes. The required performance of the system includes showing a good error convergence with a proper estimation and handling of the system uncertainties, as well as, a strong disturbance rejection while following a specified trajectory.

Table 4-1: Attitude control parameters

Parameter	Value	Unit
$\boldsymbol{\tau_R}$	40	s
$\boldsymbol{\tau_\omega}$	50	s
$\boldsymbol{K_R}$	$diag(13,13,9)$	-
$\boldsymbol{K_\omega}$	$diag(15,15,10)$	-
$\boldsymbol{\gamma_\omega}$	0.1	-
$\boldsymbol{\Gamma_\omega}$	0.01	-

Figure 4-2 and Figure 4-3 show the Euler angles and rotation matrix responses of the system, respectively. In these figures, the blue line represents the actual response while the green line represents the command signal. It is clearly seen that the control system was able to track the command signals efficiency despite their high amplitude, i.e., around $\pm\pi$. Figure 4-4 illustrates the angular rate response of the system. It shows that the angular amplitude is acceptable and physically achievable below $\pm 20\ [rad/s]$.

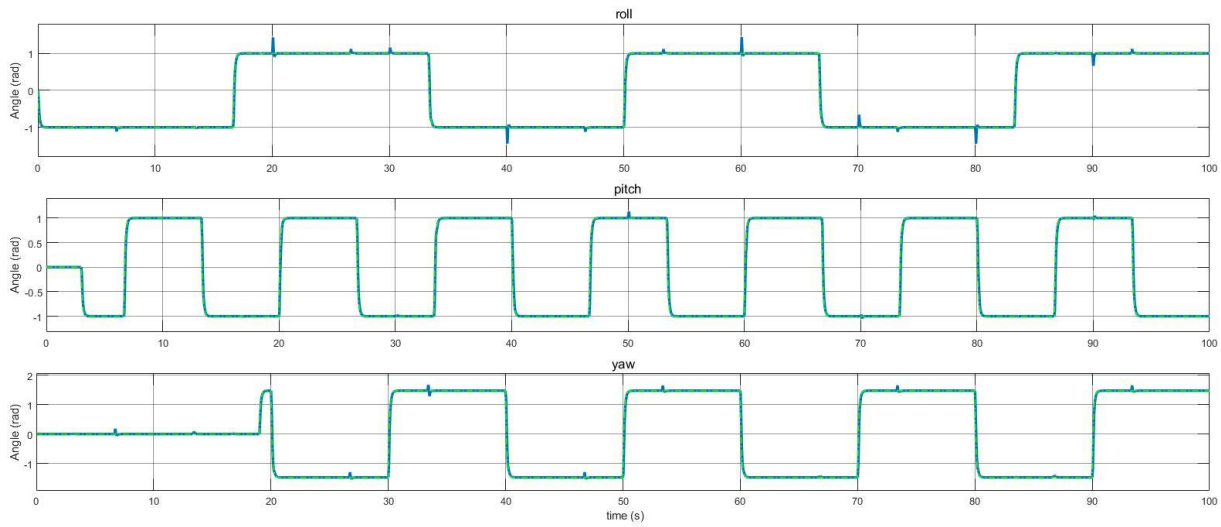

Figure 4-2: Euler angles response, where blue is the actual and green is the command

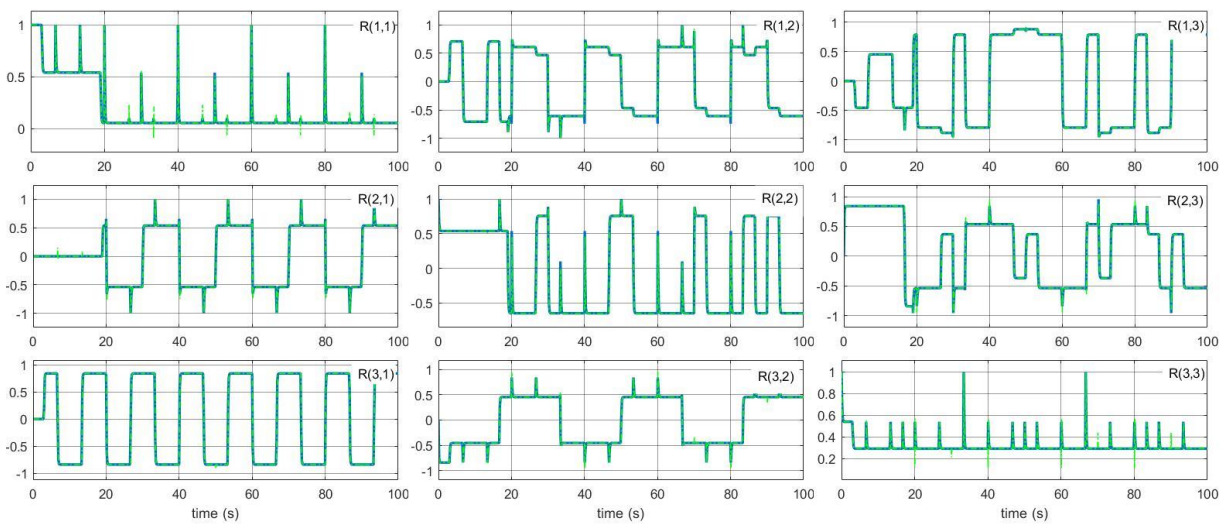

Figure 4-3: Rotation matrix response, where blue is the actual and green is the command

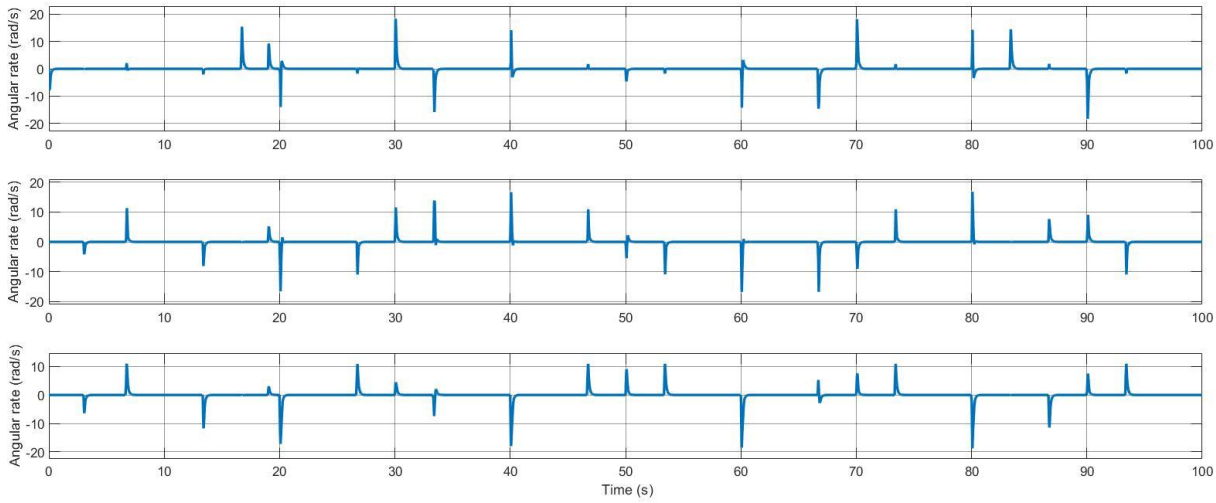

Figure 4-4: Angular rate response around x-, y- and z-axis.

Detailed navigational errors between the actual and desired trajectory in Euler angles and angular rate are shown in Figure 4-5 and Figure 4-6, respectively. The results show that the proposed control system is capable of eliminating the tracking error despite the presence of the mentioned challenges.

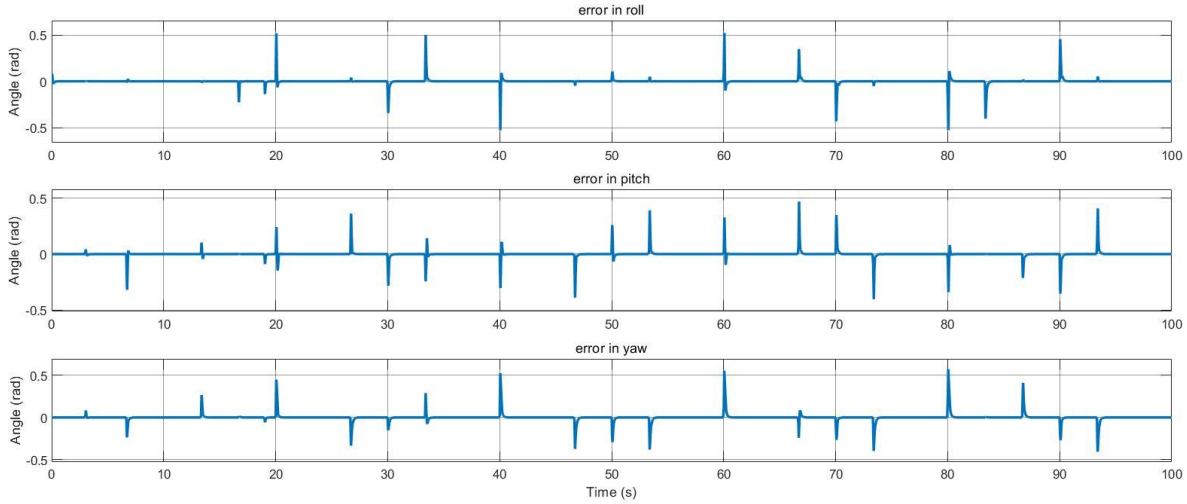

Figure 4-5: Tracking error of the Euler angles in roll, pitch and yaw domain

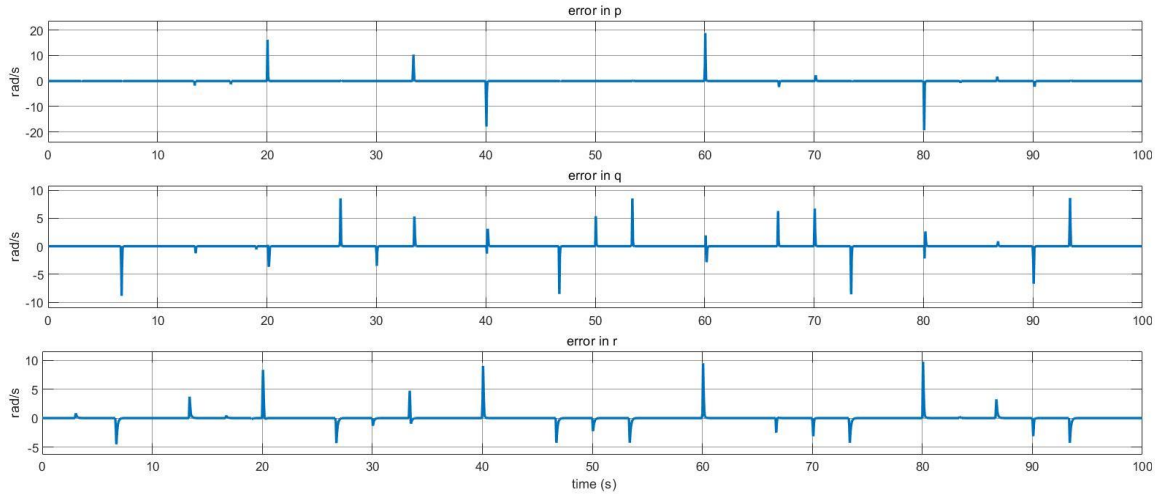

Figure 4-6: Tracking error of the angular rate around x-, y- and z-axis.

In order to show the efficiency of the proposed control algorithm, several comparison tests have been carried out. Two algorithms have been selected for the comparison; a PID-based technique, similar to the one typically used in the off-the-shelf controller, and the control algorithm proposed by T. Lee in [106]. The comparison tests are carried out by generating a train of pulses as a command signal and then calculating the integral of absolute error (IAE) for each algorithm. Two sets of command signals are generated; one with a range between $\pm\frac{\pi}{3}$ and the other between $\pm\frac{2\pi}{3}$. For each set, the tests were carried out with and without applying external disturbances. The results of the comparison tests are shown in Table 4-2. In this table, lower numbers of IAE indicate better performance and smaller tracking error.

Table 4-2: Attitude comparison tests

Method	**Roll command range: $\left[\frac{-\pi}{3}, \frac{\pi}{3}\right]$**		**Roll command range: $\left[\frac{-2\pi}{3}, \frac{2\pi}{3}\right]$**	
	Without disturbance	With disturbance	Without disturbance	With disturbance
PID	5400	6900	12323	18230
T. Lee [106]	2320	2711	2497	2912
Proposed	482	650	645	946

The results show that the proposed control method outperforms the other methods in all tests.

4.4.2 Experimental Test

In the experimental test, the control algorithm has been implemented using C++ language. All control process and measurements were performed onboard. In this experiment, the desired command signal in roll and yaw domains was selected as a pulse signal. The initial values of the NN parameters have been selected to be random values, as the NN doesn't require any prior training.

Figure 4-7 shows the output response of the setup for the roll and yaw angles. The result shows a good convergence for the yaw angle and acceptable convergence for the roll angle.

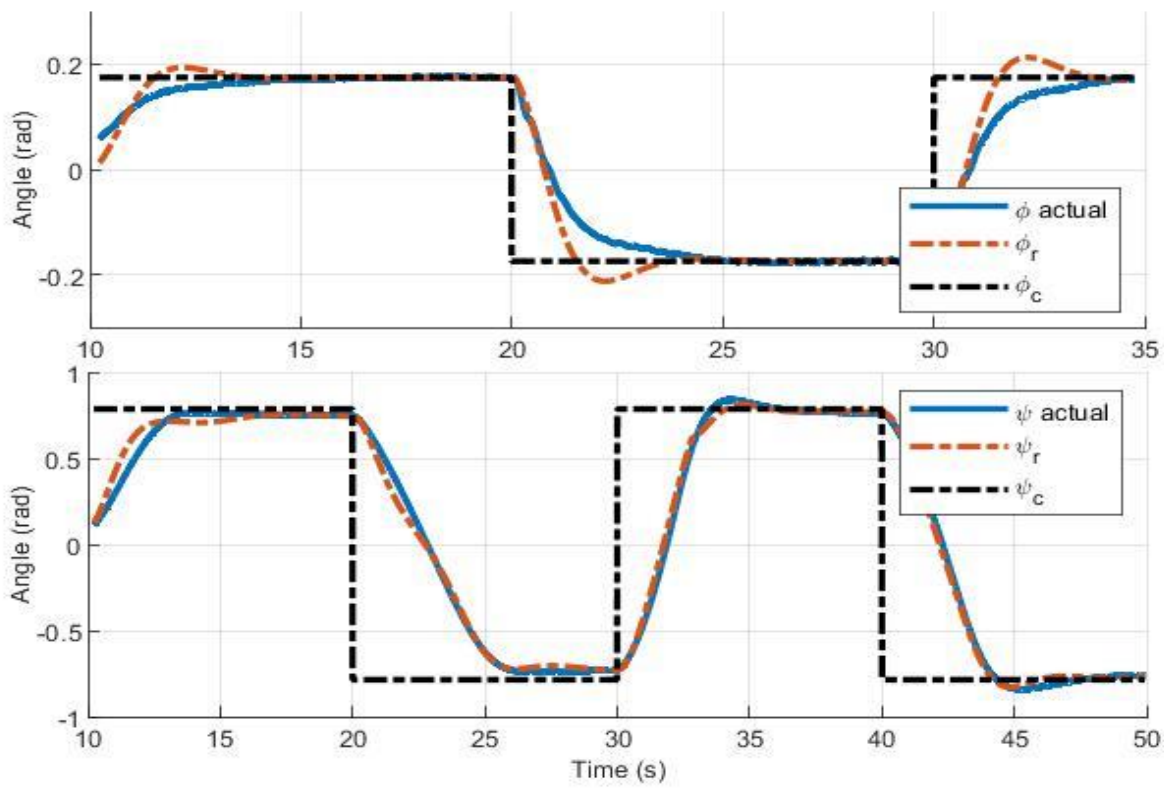

Figure 4-7: Command, Actual and reference trajectory for roll and yaw

Figure 4-8 illustrates the tracking errors for the roll and yaw angles. The error signal reached zero within an acceptable time (less than 5 seconds). Figure 4-9 shows the input signal generated by the proposed control system for the roll and yaw angles. Saturation limits were applied to the input signals of $\pm 2\ [Nm]$.

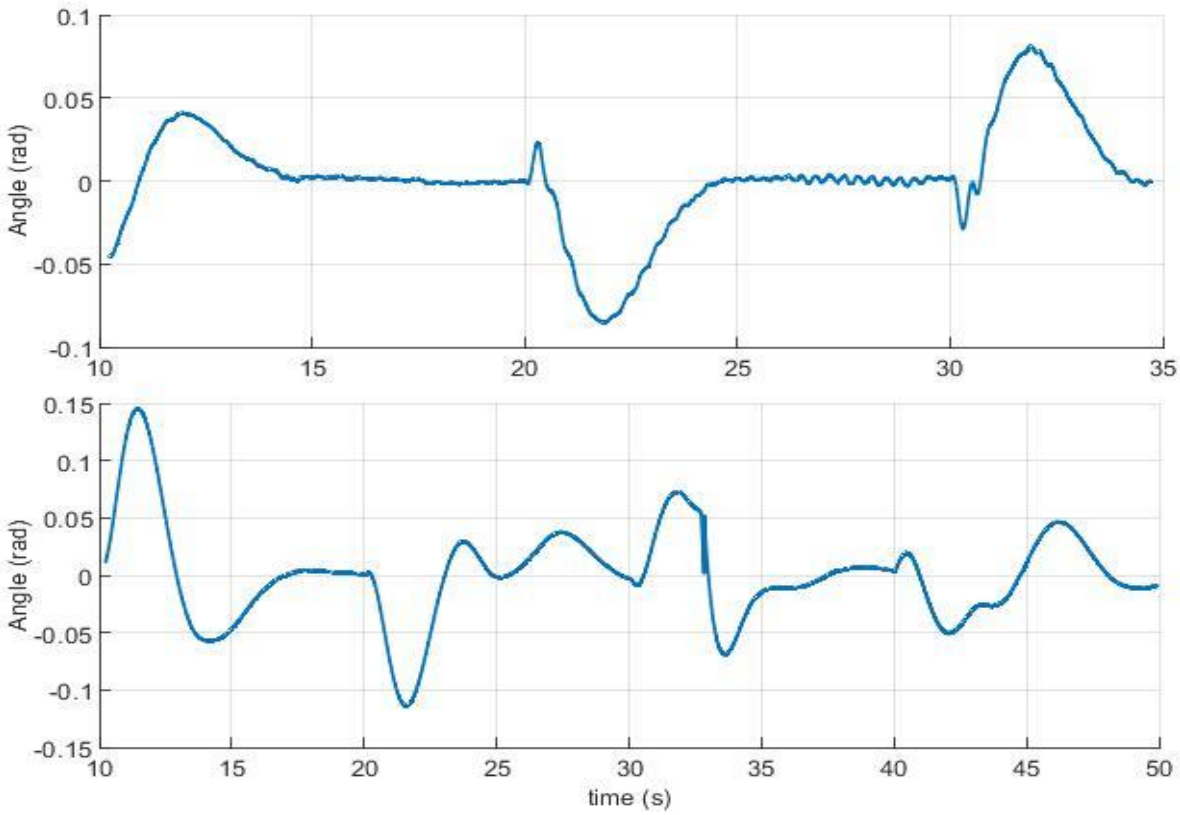

Figure 4-8: Error between the command signal and actual signal, for roll and yaw angles

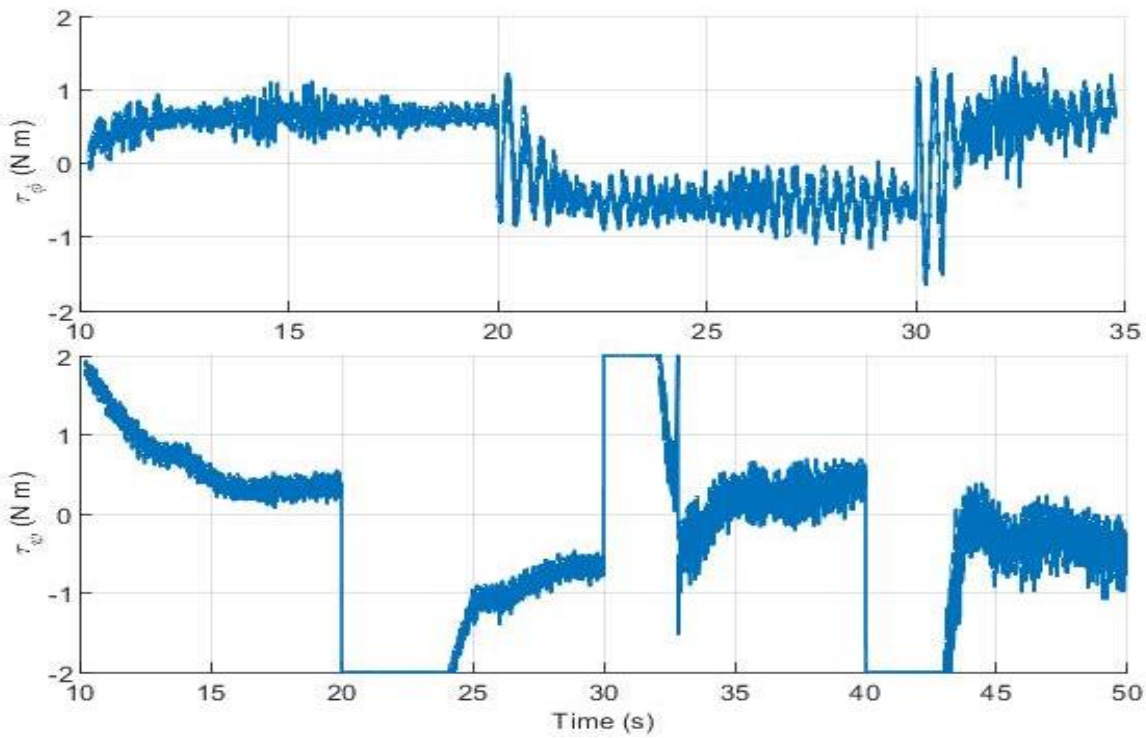

Figure 4-9: Input signal and saturation limit.

4.5 Summary

A robust adaptive attitude control system is considered for small-scale quadrotors to follow the desired path despite the system's uncertainties and external disturbances. The proposed attitude controller is based on dynamic surface control and developed directly on the special Euclidean group with a region of attraction covering the configuration space globally. In addition, the proposed controller is designed with an adaptive radial basis function neural network with a single hidden layer to overcome the system's uncertainties and estimate its bounded unknown parameters. This will allow the controller to maintain both dynamics stability and tracking performance while avoiding the singularity associated with orientation representations. The performance of the proposed controller was compared to two other approaches; a traditional PID control used typically in off-the-shelf solutions and a state-of-the-art solution proposed by T. Lee in [106]. The results show that the proposed control outperforms those two approaches in terms of both accuracy and convergence rate. The stability of the

control system is proven using a Lyapunov function within the region of interest and its performance has been validated by both simulation and experiments.

Chapter 5: Linear Dynamics Control in Presence of Extreme External Disturbances

This chapter presents a novel position tracking control system for a quadrotor unmanned aerial vehicle (UAV) in the presence of uncertainties and extreme external disturbances. In particular, the control method combines three techniques: a second order sliding mode control (SMC), a switching mechanism and a non-parametric adaptation technique. The objective of the position controller is to generate a proper attitude command signal for the attitude controller. It uses the switching mechanism and the adaptation technique to enhance the vehicle tracking performance in the presence of the mentioned challenges.

This chapter is organized as follows. The chapter starts by presenting an overview and the selected feedback strategy in Section 5.1. Section 5.2 describes the problem formulation and the assumptions considered in this chapter. The control development is presented in Section 5.3. The section introduces three different control systems namely hovering control, translation control and translation control with high disturbances. The development control of each control system is presented, as well as, its stability analysis using Lyapunov theory. Finally, in Section 5.4, the results are illustrated through numerical simulation tests.

5.1 Overview

Quadrotor systems lack adequate control actions to control their entire configuration space directly due to underactuation. One method to control all 6 DOF is based on a cascade feedback strategy using a dynamic inversion technique [107]. This is done by dividing the system into two subsystems: an inner-loop that forms the attitude dynamics and an outer-loop that forms the position dynamics of the UAV. Then, a feedback control is designed for each loop separately. Lastly, the two loops are connected using the outer-loop feedback signal to provide an attitude reference for the inner-loop system. The overall goal of the control system is to achieve system stability and tracking for the desired command signals selected as a position in

3 DOF $[x_d, y_d, z_d]$ and heading angle ψ_d. This chapter focuses on the outer-loop dynamics, the position dynamics, and its control system, while the control system of the inner-loop dynamics has been investigated in Chapter 4.

5.2 Problem Formulation

In general, quadrotor systems suffer from uncertainties in their parameters and models which may cause inaccuracy or instability for the control system. Thus, the control systems require some robustness features to overcome the uncertainties effects, as well as, an adaptation mechanism to improve the system response with time. A nonlinear system described earlier in (3-10) can be modeled as follows

$$\dot{p} = v \tag{5-1}$$

$$\dot{v} = F_v + \theta_v G_v u_v + D_v \tag{5-2}$$

where

$$F_v = -g e_3 \tag{5-3}$$

$$G_v = R e_3 \tag{5-4}$$

$$\theta_v = \frac{K_b}{m} \tag{5-5}$$

$$u_v = \sum_{i=1}^{4} {\Omega_i}^2 \tag{5-6}$$

where $u_v(t) \in \mathbb{R}$ is the input vector, $y(t) \in \mathbb{R} \times SO(3)$ is the output, $x(t) = [P^T,\ v^T]^T \in \mathbb{R}^6$ is the state vector, $F_v \in \mathbb{R}^3$ is the state function, $\theta_v \in \mathbb{R}$ is the parameter uncertainties and $D_v \in \mathbb{R}^3$ is the lumped uncertainties in the system model associated with unmodeled dynamics and external disturbances.

Now, the aim is to design a robust position controller based on an adaptive neural network for a 6 DOF quadrotor to follow a designed path in presence of system underactuation, parameter uncertainties, model uncertainties and disturbances under the following assumptions:

1. All of the state variables, $x(t)$, are measurable.
2. θ_v is unknown constant parameter with known sign and lower threshold $\bar{\theta}_v$.
3. Upper bounds for the vehicle uncertainties $D_v(t)$ is assumed so that $\|D_v(t)\| \leq D_{v_{max}}$. This assumption can be achieved using a nonlinear model with a good level of accuracy.
4. With the above assumptions, the system stability can be guaranteed using the proposed control systems.

5.3 Control Development

The outer-loop dynamics, given by (5-1) and (5-2), include the altitude motion in, z-axis and internal dynamics of the translation motion in, x- and y-axis. This has been introduced using the feedback linearization technique on the quadrotor system. Still, one must make sure that the internal dynamics are stable in order to guarantee the stability of the overall system. The position flight control is divided into two controllers, altitude control and position control.

5.3.1 Altitude Control

The purpose of this section is to develop a control system for the altitude motion in the z-axis. The dynamics of the z-axis can be expressed as

$$\dot{p}_z = v_z \tag{5-7}$$

$$\dot{v}_z = F_z + \theta_v \, R_{3,3} \, u_v + D_z \tag{5-8}$$

where the z-subscript indicates the third element of a corresponding vector, $R_{3,3}$ is the third element of the third column of the current attitude rotation matrix. In order to design a proper tracking of the desired altitude command signal p_{z_d}, the value of the input signal u_v must be calculated. First, under the assumption of a smooth desired position tracking command signal $p_{z_d} \in \mathbb{R}$, a tracking error signal for altitude is given by,

$$e_z = p_z - p_{z_d} \tag{5-9}$$

$$\dot{e}_z = v_z - \dot{p}_{z_d} \tag{5-10}$$

Then, the sliding surface for the sliding mode control of uncertain nonlinear systems can be defined as

$$s_z = \dot{e}_z + \lambda_z\, e_z \tag{5-11}$$

where λ_z is a design positive constant matrix that will be defined later. The dynamics of the sliding surface using the nonlinear system defined in (5-7) yields

$$\dot{s}_z = \dot{v}_z - \ddot{p}_{z_d} + \lambda_z\, e_z \tag{5-12}$$

Define the estimation of the parameter θ_v and its error as

$$\tilde{\theta}_v = \theta_v - \hat{\theta}_v \tag{5-13}$$

where $\hat{\theta}_v \in \mathbb{R}$ is the estimation of the unknown parameter θ_v and $\tilde{\theta}_v$ is the estimation error. We can now define the input signal u_v as follows

$$\bar{u}_v = \left(R_{3,3}\hat{\theta}_v\right)^{-1} q_z \tag{5-14}$$

$$q_z = \dot{v}_z - \widehat{D}_z - F_z - \lambda_z \dot{e}_z - K_z s_z \tag{5-15}$$

where K_z is a positive constant and selected later with λ_z to represent a stable Hurwitz polynomial. $\widehat{D}_z \in \mathbb{R}$ denotes the estimate of the disturbance signal D_z defined using the RBF-NN as

$$\widehat{D}_z = \hat{f}(e_z, \dot{e}_z) = \widehat{W}_z^{\,T} h_z(s_z) \tag{5-16}$$

where $\widehat{W}_z$ is the estimated disturbance weights. By substituting the input signal into (5-11) the dynamics of the sliding surface can be written as

$$\dot{s}_z = -K_z s_z - \widetilde{W}_z^T h_z + \epsilon_z - \tilde{\theta}_v\, R_{3,3}\, q_z \tag{5-17}$$

$$\widetilde{W}_{v_z} = W_{v_z} - \widehat{W}_{v_z} \tag{5-18}$$

where $\widetilde{W}_z$ is the estimation error of the disturbance signal. With $K_z > 0$ and $\widetilde{W}_z,\ \tilde{\theta}_v \to 0\ as\ t \to \infty$, then $s_z \to \epsilon_z$.

Proof: using the following Lyapunov function:

$$V_z = \frac{1}{2} s_z^2 + \frac{1}{2\gamma_z} \widetilde{W}_z^{\,2} + \frac{1}{2\eta_z} \hat{\theta}_z^{\,2} \tag{5-19}$$

where γ_z and η_z are positive constants representing the learning rate of the estimated weights and parameters, respectively. Taking the derivative of the Lyapunov function yields

$$\dot{V}_z = -K_z\, s_z^2 + \epsilon_z\, s_z + \widetilde{W}_z^{\,T}\left(s_z h_z - \frac{1}{\gamma_z}\dot{\hat{W}}_z\right) - \hat{\theta}_v\left(R_{3,3}\, s_z\, q_z + \frac{1}{\eta_z}\dot{\hat{\theta}}_v\right) \tag{5-20}$$

In order to guarantee the stability and avoid the singularity in the control signal, we can choose the following adaption function

$$\dot{\hat{W}}_z = \gamma_z\, s_z\, h_z \tag{5-21}$$

$$\alpha_z = R_{3,3}\, s_z\, q_z \tag{5-22}$$

$$\dot{\hat{\theta}}_v = \begin{cases} -\eta_z\, \alpha_z & \alpha_z > 0 \\ -\eta_z\, \alpha_z & \alpha_z \leq 0\ \&\ \hat{\theta}_v > \bar{\theta}_v \\ -\eta_z & \alpha_z \leq 0\ \&\ \hat{\theta}_v \leq \bar{\theta}_v \end{cases} \tag{5-23}$$

where $\bar{\theta}_v$ represent the lower limit of the estimated parameter and $\hat{\theta}_v(0) > \bar{\theta}_v$. Then the derivative of the Lyapunov function becomes

$$\dot{V}_z = -K_{p_z} s_z^2 + \epsilon_z\, s_z \tag{5-24}$$

Using the RBF, the approximation error ϵ is limited and sufficiently small. Therefore, $\dot{V}_z < 0$ and gives exponential convergence for the filter error s_z which, as a result, guarantees the convergence of tracking error e_z. Using the Barbalat's extension so does $\dot{e}_z$. Hence the altitude tracking is achieved.

This altitude control can then be used along with the attitude control which was proposed earlier in Chapter 4 as a hover control for UAV. In this case, the attitude control system used as a regulator, i.e., the command signal $R_c$received by the attitude control is set to an identity matrix. Figure 5-1 illustrates the block diagram of the hover control system.

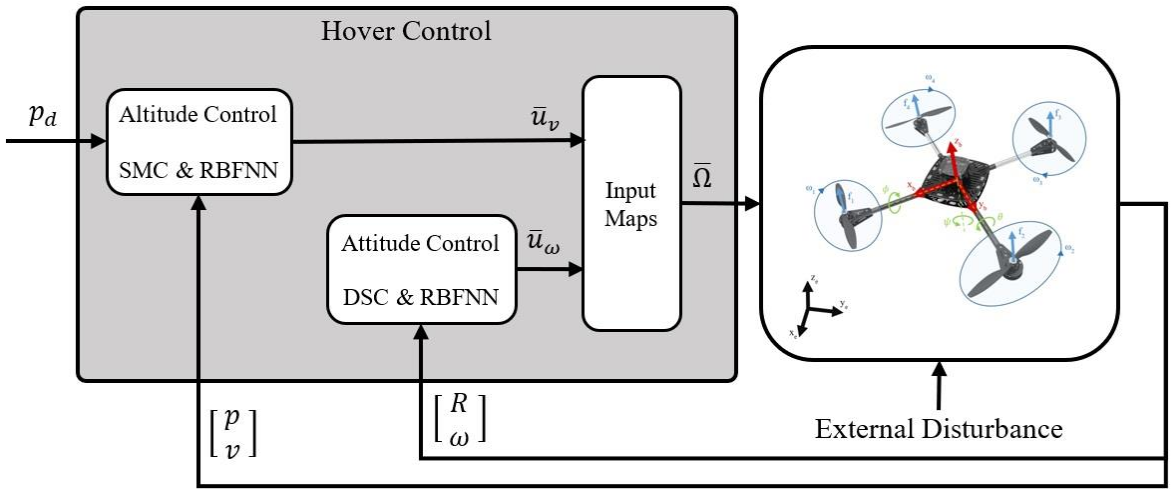

Figure 5-1: Block diagram of the hover control system

5.3.2 Position Control

The goal of the position control is to generate a proper attitude command R_c for the attitude controller to track. The command signal is developed in the $\mathrm{SO}(3)$, therefore it avoids singularities of Euler angles and unwinding quaternions.

Under the assumption of a smooth desired position tracking command signal, $p_d(t) = \left[p_{x_d}, p_{y_d}, p_{z_d}\right]^T \in \mathbb{R}^3$. The tracking error signal for position states can be defined as

$$e_p = \begin{bmatrix} e_x \\ e_y \\ e_z \end{bmatrix} = \begin{bmatrix} p_x - p_{x_d} \\ p_y - p_{y_d} \\ p_z - p_{z_d} \end{bmatrix} \tag{5-25}$$

$$e_v = \dot{e}_p = \begin{bmatrix} \dot{e}_x \\ \dot{e}_y \\ \dot{e}_z \end{bmatrix} = \begin{bmatrix} v_x - \dot{p}_{x_d} \\ v_y - \dot{p}_{y_d} \\ v_z - \dot{p}_{z_d} \end{bmatrix} \tag{5-26}$$

Thus, the sliding surface for the sliding mode control of uncertain nonlinear systems can be defined as

$$s_p = \begin{bmatrix} s_x \\ s_y \\ s_z \end{bmatrix} = e_v + \lambda_p \, e_p \tag{5-27}$$

where λ_p is a design positive diagonal matrix defined later. The nonlinear dynamics of the sliding surface in (5-27) can be written as

$$\dot{s}_p = (\dot{v} - \dot{v}_d) + \lambda_p \, e_v \tag{5-28}$$

$$\dot{s}_p = (F_v + \theta_v G_v u_v + D_v - \dot{v}_d) + \lambda_p e_v \tag{5-29}$$

Using the relation between the rotation matrix and sliding surface dynamics, one can utilize the sliding error to construct the direction of the third axis of the command attitude signal as follows

$$q_p = \dot{v}_d - \widehat{D}_v - F_v - \lambda_p e_v - K_p s_p \tag{5-30}$$

$$R_c e_3 = b_{3_c} = \frac{q_p}{\|q_p\|} \tag{5-31}$$

where K_p is a positive diagonal matrix and selected later with λ_p to represent a stable Hurwitz polynomial. $\widehat{D}_v \in \mathbb{R}^3$ denotes an estimate of the disturbance signal D_v defined using the RBF-NN as

$$\widehat{D}_v = \hat{f}(e_v, \dot{e}_v) = \widehat{W}_v^{\,T} h_v(s_v) \tag{5-32}$$

where $\widehat{W}_v$ is the estimated disturbance weights. Similar to (5-21), we can generalize the adaption law of the disturbance weights ($\widehat{W}_v$) for all position dynamics as

$$\dot{\widehat{W}}_v = \gamma_v \, s_z \, h_v \tag{5-33}$$

The component of the unit vector b_{3_c} is constructed to resemble the weight of the correction term q_{p_i} of each translation axis. The direction of the first axis of the command attitude signal can be chosen to specify the desired heading direction of the quadrotor in the horizontal plane. This is achieved using the command heading angle ψ_c as follows

$$b_{1_{ideal}} = \begin{bmatrix} cos(\psi_c) & -sin(\psi_c) & 0 \\ sin(\psi_c) & cos(\psi_c) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \tag{5-34}$$

However, one needs to ensure that the attitude command signal (R_c) is defined properly that is $R_c \in SO(3)$. In practice, the direction of the first column of the matrix in (5-33) needs to be orthogonal to the third column. Thus, the projection of the first axis on the third axis is given by,

$$b_{1_c} = Proj_{b_{3_c}}\left(b_{1_{ideal}}\right) \tag{5-35}$$

$$b_{1_c} = b_{1_{ideal}} - \left(b_{1_{ideal}}^T b_{3_c}\right)\frac{b_{3_c}}{\left\|b_{3_c}\right\|} \tag{5-36}$$

It is worth to mention that, using the projection function, the first axis of the attitude command signal as $t \to \infty$ converges to the projection of b_{1_c} not $b_{1_{ideal}}$. Then the full construction of the attitude command signal is found as

$$R_c = \left[b_{1_c}, b_{3_c} \times b_{1_c}, b_{3_c}\right] \tag{5-37}$$

Then, the position control can then be used along with the attitude and altitude control systems as a 6 DOF control for UAV. In this case, the attitude control system acts as a tracking controller for the command signal generated by the position controller. Figure 5-2 illustrates the block diagram of the hover control system.

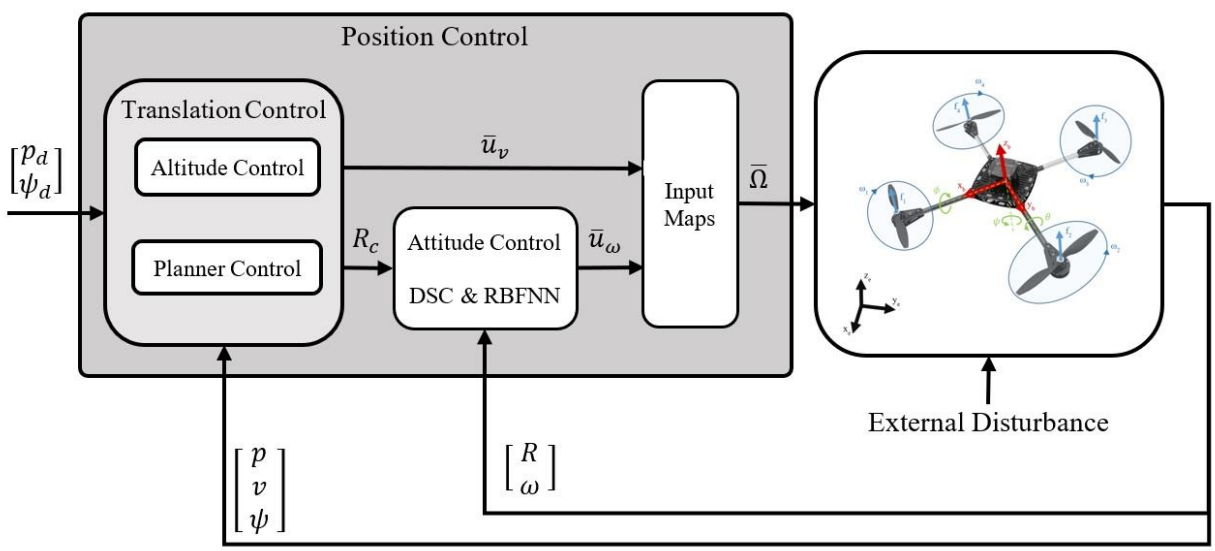

Figure 5-2: Block diagram of a 6 DOF controller

5.3.3 Position Control with Extreme Disturbances

The desired rotational matrix proposed in the previous section works efficiently when the system is subjected to relatively low disturbances. However, when one of the system's axis is subjected to a higher disturbance than the others, the control system will not be able to reject the disturbances, effectively. In other words, the position control signals will appear in the components of the third vector of the rotational matrix, defined in (5-30). Due to the normalization of these components, this vector must be biased towards the axis with the higher value, i.e. higher control signal. Thus, the components along the other axes will be attenuated. Furthermore, the thrust input signal only depends on the z-axis control signal and is not affected by the error of the x- or y-axis. As a result, the system will be able to eliminate the tracking error of the dominant axis, the z-axis, and ignore the tracking error of the other axes. In this thesis, a novel approach is proposed to address this challenge.

The enhanced position controller is developed using a switching mechanism based on the Multiple Lyapunov Function (MLF) method. The procedure involves the following. First, the system is divided into three distinct domains where each domain resembles a single axis, i.e. $i = \{x, y, z\}$. Second, the attitude control and surface error for each domain is calculated and subsequently the one with the highest value as the dominant domain is selected in the following order:

1. X-axis: the third axis of the command attitude signal can be redefined as

$$\begin{bmatrix} q_{p_x} \\ q_{p_y} \\ q_{p_z} \end{bmatrix} = \begin{bmatrix} \dot{v}_{d_x} - \widehat{D}_{v_x} - F_{v_x} - \lambda_{p_x} e_{v_x} - K_{p_x} s_{p_x} \\ 0 \\ -F_{v_z} \end{bmatrix} \tag{5-38}$$

$$R_{c_x} e_3 = b_{3_c} = \frac{q_p}{\|q_p\|} \tag{5-39}$$

$$V_x = \frac{1}{2} s_{p_x}^2 \tag{5-40}$$

This rotation matrix will stabilize the system in the x-axis while the third component is used to keep the rotation matrix point upward.

2. Y-axis: the third axis of the command attitude signal can be redefined as follow

$$\begin{bmatrix} q_{p_x} \\ q_{p_y} \\ q_{p_z} \end{bmatrix} = \begin{bmatrix} 0 \\ \dot{v}_{d_y} - \widehat{D}_{v_y} - F_{v_y} - \lambda_{p_y} e_{v_y} - K_{p_y} s_{p_y} \\ -F_{v_z} \end{bmatrix} \tag{5-41}$$

$$R_{c_y} e_3 = b_{3_c} = \frac{q_p}{\|q_p\|} \tag{5-42}$$

$$V_y = \frac{1}{2} s_{p_y}^2 \tag{5-43}$$

Similarly, this rotation matrix will stabilize the system in the y-axis while the third component is used to keep the rotation matrix point upward.

3. Y-axis: the third axis of the command attitude signal can be redefined as follow

$$\begin{bmatrix} q_{p_x} \\ q_{p_y} \\ q_{p_z} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \dot{v}_{d_z} - \widehat{D}_{v_z} - F_{v_z} - \lambda_{p_z} e_{v_z} - K_{p_z} s_{p_z} \end{bmatrix} \tag{5-44}$$

$$R_{c_z} e_3 = b_{3_c} = \frac{q_p}{\|q_p\|} \tag{5-45}$$

$$V_z = \frac{1}{2} s_{p_z}^2 \tag{5-46}$$

This rotation matrix focuses only on stabilizing the z-axis.

Finally, a switching controller law using multiple Lyapunov functions that keep the overall system stable is applied.

Theorem 1: Let the pair (R_{c_i}, V_i) with $i = x, y, z$ define the subsystem controllers and corresponding Lyapunov functions. Then, the control law

$$R_c = \{R_{c_i} : V_i = max\{V_x, V_y, V_z\}\} \tag{5-47}$$

guarantees the tracking of the quadrotor systems.

Proof: Let us consider

$$V_\sigma = max\{V_x, V_y, V_z\} \tag{5-48}$$

This is a bounded function, since

$$a_l^i \|x\| < V_i \leq a_u^i \|x\| \tag{5-49}$$

with $i = x, y, z$ then

$$\underline{a}\|x\| < V_\sigma \leq \overline{a}\|x\| \tag{5-50}$$

with

$$\underline{a} = max\{a_l^i\}, \qquad \overline{a} = min\{a_u^i\} \;\; \forall i \tag{5-51}$$

Thus, V_i is a piecewise decreasing function in the intervals defined by $\sigma(t)$. We have that at each interval, the selected control R_{c_i} ensures the exponential decay of V_i. Due to the continuity of V_i for $i = x, y, z$ the function satisfies $V_{\sigma(t_{i+1})} < V_{\sigma_{(ti)}}$ since V_i's cannot jump at the time of switching instants. Therefore, V_σ is a Lyapunov-like function. All $s_i(t)$'s goes to zero exponentially. Hence, the tracking is satisfied in all three subspaces. This concludes the proof.

The overall block diagram of the proposed controller is shown in Figure 5-3.

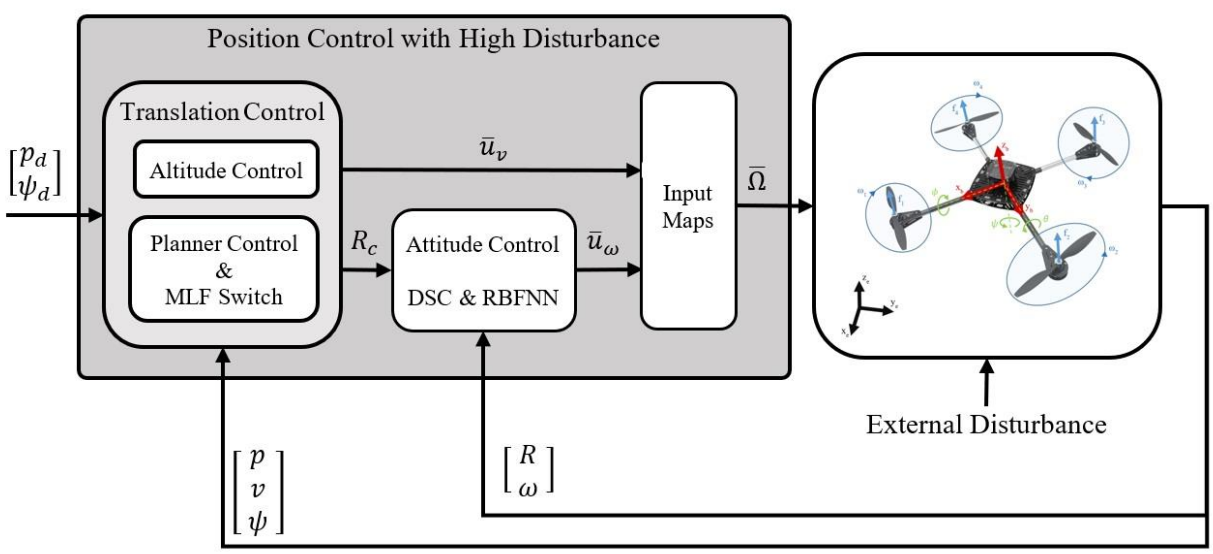

Figure 5-3: Block diagram of 6 DOF controller with high disturbance rejection

5.4 Results

In this section, the results of the proposed controllers through a simulation software are demonstrated. The purpose of the simulation is to show the satisfactory overall performance of the proposed controller in an ideal environment. The simulation shows that the performance of the proposed controller can exceed that of traditional approaches in terms of accuracy and convergence rate under the presence of uncertainties and disturbances. The simulation tests were carried out using the control parameters shown in Table 5-1.

Table 5-1: Position control parameters

Parameter	Value	Unit
$\boldsymbol{\lambda_z}$	10	-
$\boldsymbol{k_z}$	8	-
$\boldsymbol{\lambda_p}$	$diag(5,5,10)$	-
$\boldsymbol{k_p}$	$diag(4,4,8)$	-
$\boldsymbol{k_\omega}$	$diag(15,15,10)$	-
$\boldsymbol{\gamma_z}$	0.1	-
$\boldsymbol{\eta_z}$	0.1	-
$\boldsymbol{\gamma_v}$	0.01	-

The simulation tests illustrate the response of the proposed controller under different conditions including the presence of unmodeled dynamics, parameter uncertainties and external disturbance. The desired trajectory is defined as a set of three flight behaviors: vertical take-off, hovering in XY-plane, and a vertical landing. At the vertical take-off behavior, the system is commanded to start rising until it reaches a certain altitude with a velocity of 1 m/s while the system is subjected to uncertainties in its parameters and dynamics. The required performance of the system for this stage includes showing a good error convergence with a proper estimation and handling of the system uncertainties. At the hovering stage, the system will be subjected to external disturbances, i.e., wind gust, on the z-axis. The performance of the system is required

to show a good disturbance rejection while keeping the specified altitude. This includes demonstrating a good error convergence and an adequate estimation of the external disturbances. Finally, for a landing test, the quadrotor is commanded to start descending from a specific altitude until it reaches the ground with a slower slope, 0.5 m/s. The quadrotor needs to land safely and precisely. Figure 5-4 shows the 3D path of the actual and desired trajectories of the proposed system.

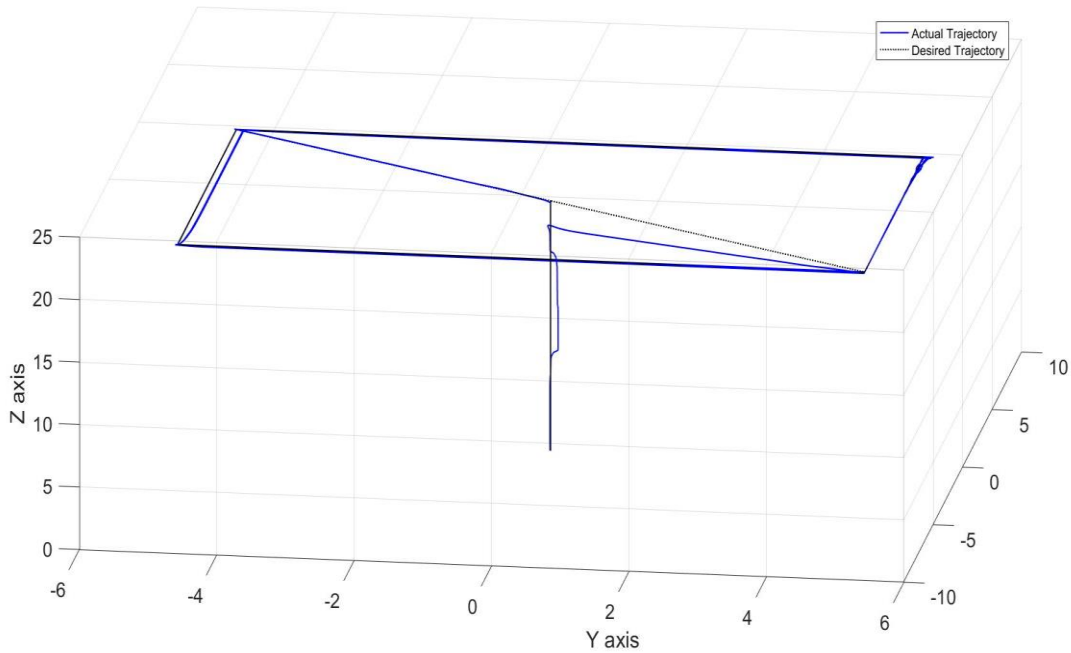

Figure 5-4: Desired and actual trajectories

Figure 5-5 shows the command, desired and actual signals of the linear dynamics. Figure 5-6 shows the corresponding responses of Euler-angle. The results show a precise tracking of the command signals. Due to the uncertainties in the system model, the adaptation mechanism, adopted by the proposed control system, enhances the system behavior continuously. The quadrotor was able to perform all required behaviors despite the considered challenges.

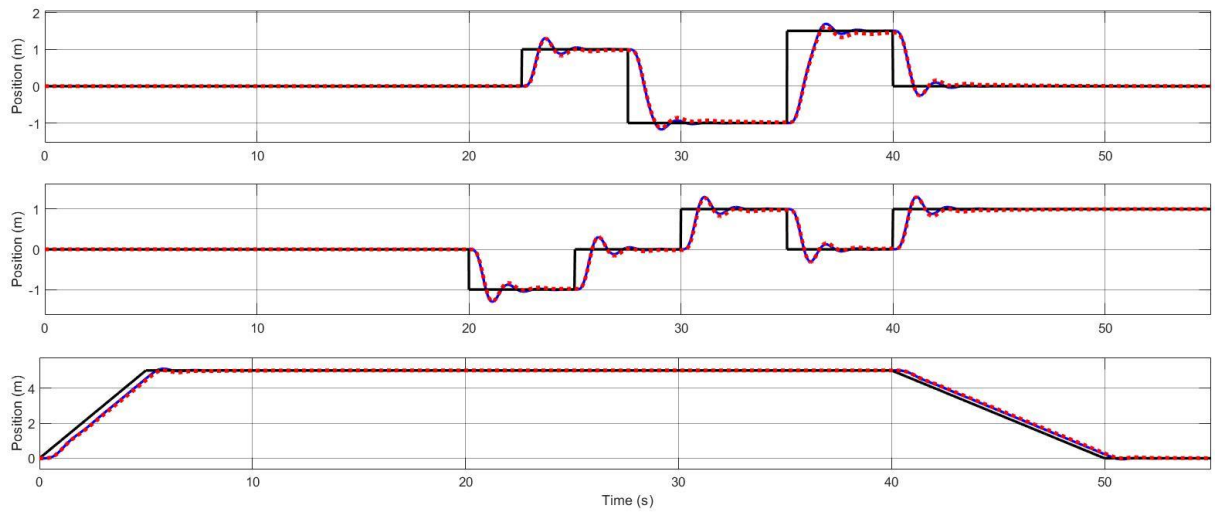

Figure 5-5: Position command and response in x-, y-, z-axis; command signal is in black, the desired signal is in red and actual signal is in blue.

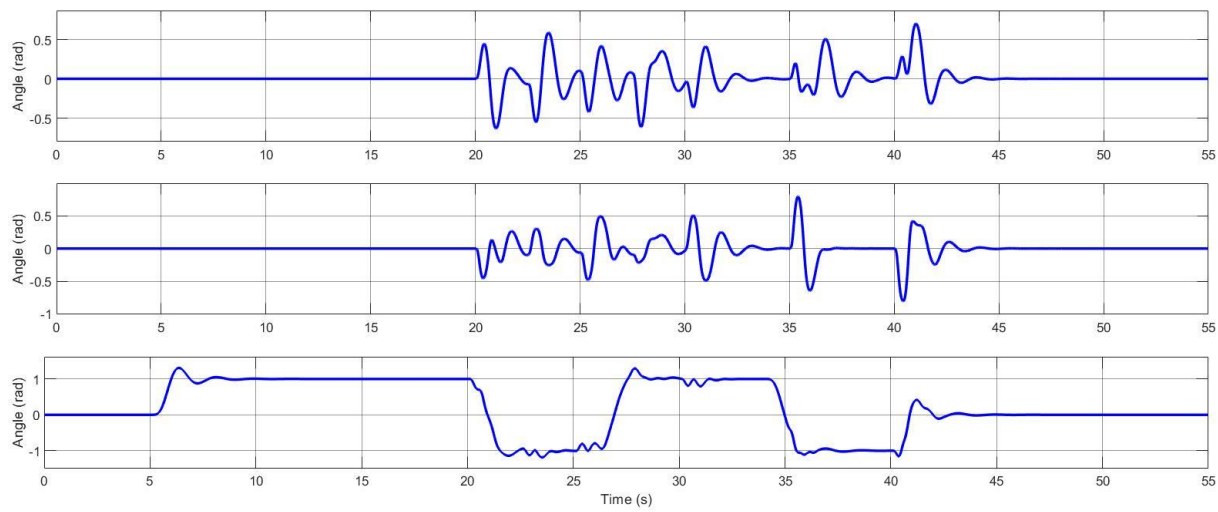

Figure 5-6: Euler-angle response in roll, pitch and yaw domain

Detailed navigational errors between the actual and desired trajectory in position and Euler angles are shown in Figure 5-7 and Figure 5-8, respectively. It is clearly seen that the proposed control system was able to achieve a precise tracking of the command signal, i.e., the tracking error converges to zero, despite the presence of the model uncertainties and external disturbances.

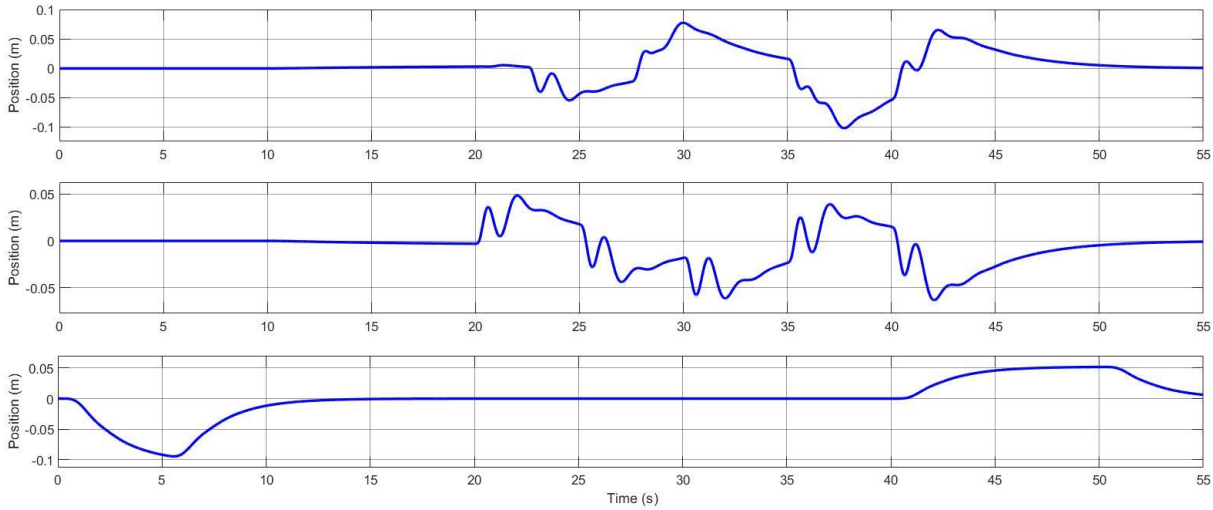

Figure 5-7: Tracing error of the position signals in x-, y-, and z-axis.

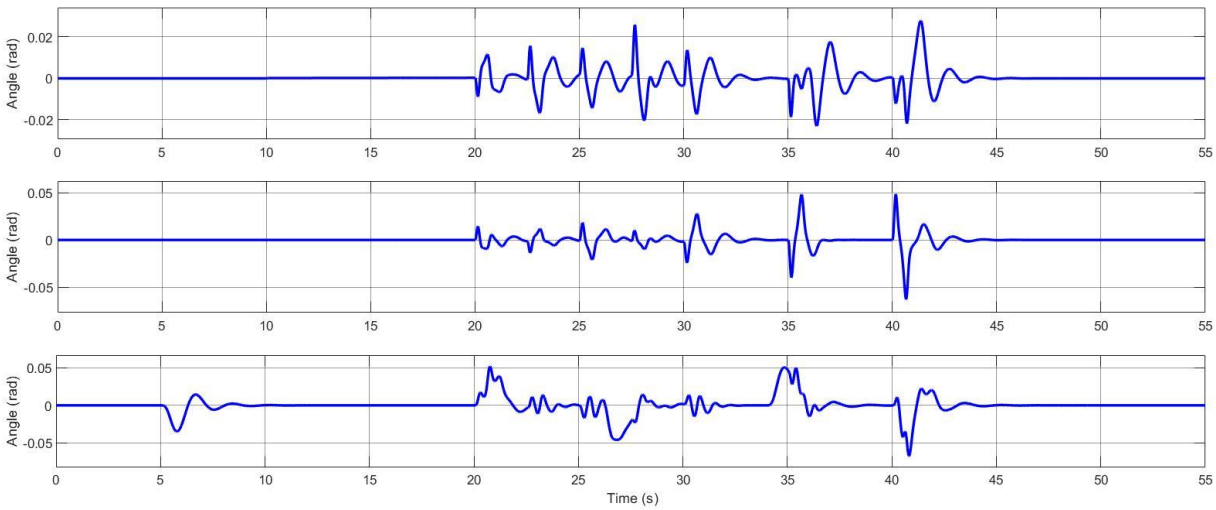

Figure 5-8: Tracing error of the attitude signals in roll, pitch and yaw domain

A comparison test has been carried out to show the efficiency of the proposed control algorithm. The test is carried out with and without applying the proposed switching mechanism to the position controller while subjecting the vehicle to external disturbances of 4N in its x and y-axes. The comparison tests are carried out by generating a train of pulses as command signals and then calculating the integral of absolute error (IAE) for each method. The results of the comparison test are shown in Table 5-2. In this table, lower numbers of IAE indicate better performance and smaller tracking error.

Table 5-2: Position comparison test

Axis	Without Switching Mechanism	With Switching Mechanism
x-axis	6733	1957
y-axis	3081	1041
z-axis	123.7	123.6

The results show that adding the proposed switching mechanism to the position controller enhanced the overall tracking performance in the x-y plane.

5.5 Summary

In this chapter, a novel position controller is designed for quadrotor UAV in the presence of uncertainties and extreme external disturbances. In particular, the control method combines three techniques: a second order sliding mode control (SMC), a switching mechanism and a non-parametric adaptation mechanism. The SMC is used to guarantee the stability of the position dynamics by generating a proper attitude command for the attitude controller. The switching mechanism is based on multiple Lyapunov functions (MLFs) and enabled in the presence of extreme disturbances to improve the tracking performance. The adaptation mechanism includes a radial basis function neural network (RBF-NN) to observe uncertainties without the need for prior training. The result showed that the proposed controller managed to improve the tracking performance of the translation dynamics in the presence of the mentioned challenges.

Chapter 6: Fault-Tolerant Control

A fault-tolerant tracking controller is proposed for a quadrotor unmanned aerial vehicle in the Cartesian space in spite of the total failure of one or two adjacent rotors. Depending on their severity, actuator faults can have detrimental effects on quadrotor operation and lead to system instability and catastrophic crash if not tackled properly. The proposed schemes consist of a real-time fault estimation and robust control strategy. The estimation scheme integrates a wind sensor mounted onboard to measure the wind speed and hence the thrust generated by the rotors directly. The wind measurement is then used as fault detection. The robust control adopts a single control feedback strategy for both the normal flight operation, as well as, when a fault occurs; hence there is no risk of instability while switching between control laws. This integration significantly improves the disturbance rejection and fault estimation of the thrust and rotor, respectively. When a fault occurs, the controller changes the input mapping relationship to allow the control system to achieve tracking to any point in the Cartesian space without the need to switch the control law. This is achieved while losing the ability to control the yaw angle, which is justifiable under emergency circumstances.

This chapter is organized as follows. Section 6.1 presents an overview of the control challenges when a fault occurs in the system and the different solutions available in the literature. The problem formulation is presented in detail in Section 6.2. This is followed by a description of the detection system in Section 6.3. Different experiments were performed to validate the efficiency of the proposed sensor. The control developments of both cases are presented in Section 6.4. Furthermore, the section proves the stability of the proposed control system using the Lyapunov stability analysis. Finally, the overall performance efficacy is demonstrated through numerical simulations for different fault cases in Section 6.5.

6.1 Overview

In general, quadrotors often need to operate in critical or hazardous environments where safety has the highest priority [3], [10]. However, quadrotors, similar to any other systems, are naturally subjected to failure of their components which can lead to a moderate situation, such as terminating the current flight operation or an extreme situation, such as a crash. Failure usually happens for unexpected and unknown reasons. In order to reduce the vulnerability of a system to failure and maintain the system performance using the remaining non-faulty components, one needs to employ fault recovery mechanisms [31]. Developing this type of controller has been of a great interest to the control community. Different approaches were reviewed in Chapter 2 to deal with actuator faults. These approaches vary from hardware approaches, using modular design [6], [7], or simple physical redundancy in the system components[40], [41], to software approaches [48][47][47][47][47][47][47]-[49]. In case of a severe fault, a common strategy involved the procedure of dropping the ability to control the yaw angle [51], while other strategies proposed emergency landing [52], [53].

Typically, rotor control systems in UAV are operated in open control loops. That is, the flight controller (FC) sends the desired speed value to the ESC that applies a particular voltage to the motor based on previously calibrated settings. The ESC uses the feedback signals of the rotor current and voltage to maintain its desired speed. In this method; however, the thrust generated by the propeller spinning is unknown to the FC. Thus, there has been an enormous strive to develop methods to obtain closed-loop control systems for rotor systems. The methods range from using different sensors to thrust modeling and estimation. Different sensors were used as indirect measurement of the thrust, such as voltage sensors [108], fabricated pitot-tube [109] and unidirectional pitot-tube sensor [110]. Other approaches are model-based such as the use of nonlinear models in [111] and iterative calculation in [112]. A combination of these two approaches was proposed lately in [113]. Still these approaches lack a direct measurement of the thrust force and require a significant amount of calibration for rotor systems that complicate

the practical use of these systems. In this research, a type of wind sensor that can efficiently measure the generated thrust of a propeller is used to detect the thrust fault in the rotor systems.

6.2 Problem Formulation

In general, it is possible for a fault to occur in the components of the vehicle suddenly during the flight and lead to a change in the vehicle behavior. That is, in case that a failure occurs for the ith rotor of the quadrotor, the control signal applied to this rotor will be affected as well. One of the common types of rotor failure is known as “loss of effectiveness”. This type of fault can happen in different degrees starting from a small loss in rotor capability to a severe loss of power causing a total rotor failure. In the latter case, the rotor will stop responding to any control action applied and unless the system is treated, it will drive the system to instability and crash.

In fault-free, the evolution of the state variables is governed by the dynamic equations described in Chapter 3. The equations are rewritten as follows

$$\dot{p} = v \tag{6-1}$$

$$\dot{v} = F_v + \theta_v G_v u_v + D_v \tag{6-2}$$

$$\dot{R} = F_R(R, \omega) \tag{6-3}$$

$$\dot{\omega} = F_\omega(\omega) + \theta_\omega u_\omega + D_\omega \tag{6-4}$$

$$u_v = \sum_{i=1}^{4} \Omega_i^2 \tag{6-5}$$

$$u_\omega = \begin{bmatrix} u_{\omega_p} \\ u_{\omega_q} \\ u_{\omega_r} \end{bmatrix} = \begin{bmatrix} 0 & -l & 0 & l \\ -l & 0 & l & 0 \\ 1 & -1 & 1 & -1 \end{bmatrix} \begin{bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \end{bmatrix} \tag{6-6}$$

where $u_v(t) \in \mathbb{R}$is the input vector, $y(t) = [P^T, \psi] \in \mathbb{R}^4$ is the output vector,$x \in \Re^9 \times \mathrm{SE}(3)$ is the state vector,$F_v \in \mathbb{R}^3$ and $F_\omega \in \mathbb{R}^3$ are the state functions, $\theta_v \in \mathbb{R}$ and $\theta_\omega \in \mathbb{R}^{3\times3}$ are the

parameter uncertainties and $D_v \in \mathbb{R}^3$ and $D_\omega \in \mathbb{R}^3$ are the lumped uncertainties in the system model including unmodeled dynamics and external disturbances. The angular speed of each rotor is bounded by minimum and maximum values i.e., $\Omega_i = [\Omega_{min}, \Omega_{max}]$. When one of the system's rotors is at complete fault, its rotation speed in all directions will be zero, i.e. $\Omega_j = f_j = 0$.

In this thesis, two cases are considered; total failure in a single rotor and total failure in two opposing rotors. These cases are distinguished by the health status of the system rotors. The description of each case is presented in the following subsections.

6.2.1 Single-Rotor Failure

One of the considered cases in this research is the single rotor failure case. In this case, the health status of a single rotor becomes "faulty" which will alter the vehicle's dynamics dramatically. Nevertheless, the system needs to maintain its flight operation and avoid a crash. Due to the rotor failure the force and moments map is different. Thus, the input signal loses one degree of freedom and becomes $u \in \Re^3$. Without loss of generality, the fourth motor (rotor 4) is considered to experience complete failure, i.e., $f_4 = \Omega_4 = 0$. The input map defined in (6-5) and (6-6) becomes

$$u_v = \sum_{i=1}^{3} \Omega_i^2 \tag{6-7}$$

$$u_\omega = \begin{bmatrix} u_{\omega_p} \\ u_{\omega_q} \\ u_{\omega_r} \end{bmatrix} = \begin{bmatrix} 0 & l & 0 & 0 \\ -l & 0 & l & 0 \\ 1 & -1 & 1 & 0 \end{bmatrix} \begin{bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \end{bmatrix} \tag{6-8}$$

Remark 1: in case of failure of a different rotor, the corresponding column in the input map will be eliminated.

6.2.2 Two-Rotor Failure

Similar to the single-rotor failure scenario, the two-rotor failure case is defined to represent the change in the dynamics of the quadrotor system. The control system; nevertheless, needs to maintain its flight operation and avoid crashing in the same manner. With a total failure in two rotors, the input signal will be defined as $u \in \Re^2$. Hence, the system will lose its effectiveness in control on one of its control motions, i.e. u_{ω_p} or u_{ω_q} command. Without loss of generality, it is considered that rotors 4 and 2 (the opposite rotors) are faulty, i.e. $\Omega_2 = \Omega_4 = 0$. This will change the force and moment map defined earlier. The new input map will become

$$u_v = (\Omega_1^2 + \Omega_3^2) \tag{6-9}$$

$$u_\omega = \begin{bmatrix} u_{\omega_p} \\ u_{\omega_q} \\ u_{\omega_r} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ -l & 0 & l & 0 \\ 1 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \end{bmatrix} \tag{6-10}$$

6.2.3 Objectives

This chapter aims to design a fault detection and fault recovery control. The solution involves some modifications to the attitude and position controllers defined in Chapter 4 and Chapter 5, respectively. The resultant control systems are to follow a designed path in presence of system underactuation, parameter uncertainties, actuator faults, model uncertainties and disturbances under the following assumptions:

1. All of the state variables, $x(t)$, are measurable.
2. θ_v and θ_ω are unknown constant parameters with known sign and lower threshold $\bar{\theta}_v$ and $\bar{\theta}_\omega$, respectively.

3. Upper bounds for the vehicle uncertainties $D_v(t)$ and D_ω are assumed, i.e., $\|D_v(t)\| \leq D_{v_{max}}$ and $\|D_\omega(t)\| \leq D_{\omega_{max}}$, respectively. This assumption can be achieved by using a nonlinear model with a good level of accuracy.
4. With the above assumptions, the system stability can be guaranteed using the proposed control systems.

6.3 Thrust Detection

In order to detect the fault occurrence in a rotor system, a detection of the thrust force is required. In this work, a novel thrust detection system is developed. It consists of a wind sensor and an adaptive Kalman filter. The wind sensor is selected to design a close feedback loop and detect the thrust value, and the Kalman filter is used to enhance the measurement signals. However, several assumptions are made to simplify the wind flow relation. Mainly, the horizontal components of the wind velocities and UAV's velocities are ignored. These assumptions are justified, as the focus of the thrust detection is only to detect the occurrence of a fault in the rotor system, i.e., when a rotor is at complete failure, and the precise measurement of a value for the wind speed is unnecessary. The wind flow theory, wind sensor and the Kalman filter are introduced in the following subsections.

6.3.1 Wind Flow Theory and Wind Sensor

The resultant wind velocity is the sum of two velocities: the velocity of the wind approaching the propellers denoted by v_z^s, known as the free stream velocity, and the propellers' induced velocity denoted by v_z^i. The wind sensor measures the following value:

$$v_{total} = v_z^i + v_z^s \tag{6-11}$$

In a relatively low wind disturbance, one can assume that $v_z^s = 0$. In addition, the relation between the thrust and wind can be simplified under the assumption of a linear system. Thus, the relation between the thrust force and momentum $f = \partial \mathrm{P}/\partial \mathrm{x}$, can be rewritten as

$$f = \frac{m_{air}}{\Delta t}(v_{total} - v_z^s) \tag{6-12}$$

where $\mathrm{m_{air}}$ is the mass of the air flowing through the propeller.

It is worth mentioning that the closer the stream velocity is to the induced velocity, the lower the thrust and efficiency of the propeller are. This can be seen as the drop of thrust midflight when wind buffets a UAV. This can be expressed as

$$\bar{v}_{total} = \gamma v_z^i + v_z^s \tag{6-13}$$

$$\bar{f} \leq f \tag{6-14}$$

In this research, the ground truth is measured using a thrust stand obtained from RCBenchmark, RCBenchmark Thrust stand and Dynamometer Series 1580 [114]. The thrust stand incorporates a micro-controller board along with 3-axis strain gauge to measure the forces and the torque derived by the propeller. The board also collects the rotor voltage and current data, and the PWM signal of the ESC's which allows measuring the RPM and power drawn by the rotor. The thrust stand is connected to a computer and a GUI which allows the user to control the board and collect and visualize the data. The testbed collects data at a sampling frequency of up to 1kHz but averages the data to eliminate noise and outliers. As a result, the effective sampling frequency is 40Hz. Figure 6-1 shows the experimental setup that consists of the thrust stand, rotor and wind sensor.

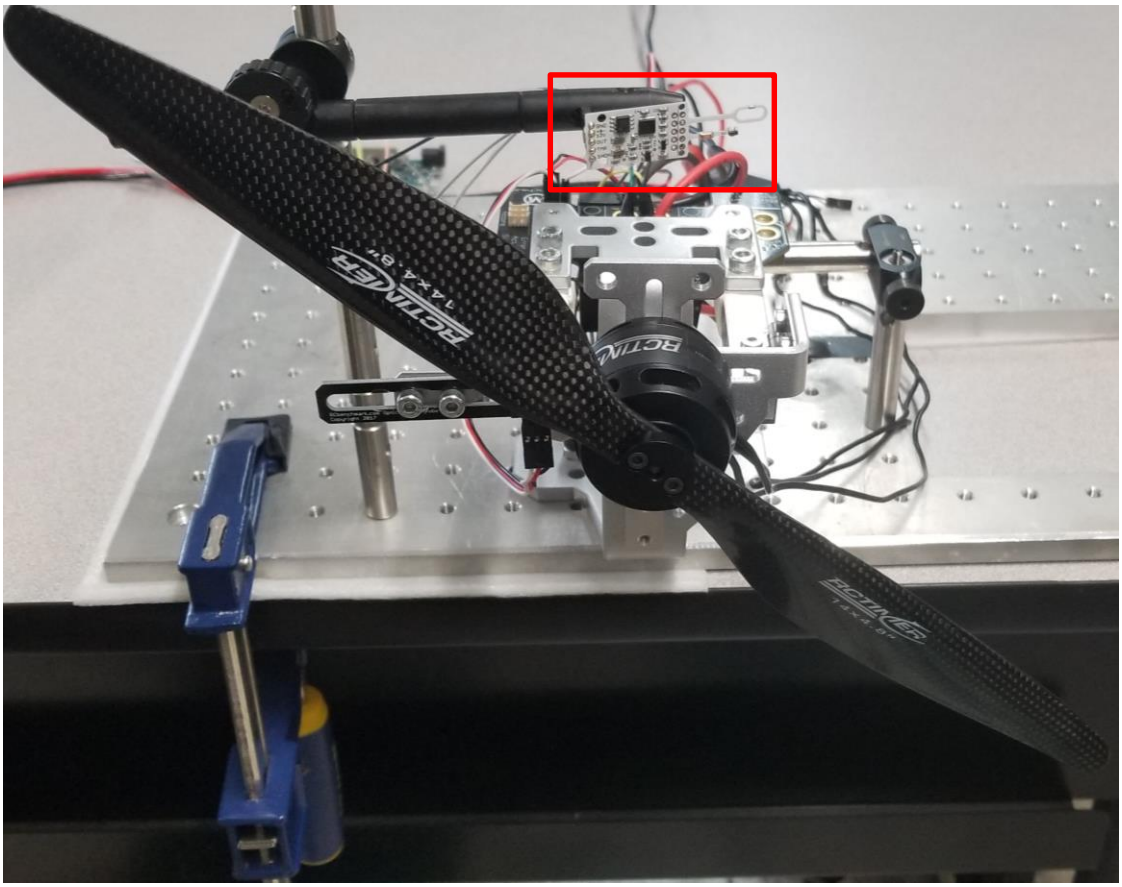

Figure 6-1: Thrust stand and wind sensor setup, where the wind sensor appear in the red box

The selected wind sensor is the Modern Device-Wind Sensor Rev. P [115]. This device is a hot wire anemometer that is designed to measure wind from any direction. Hot wire anemometers work by passing a constant current through a thin wire, which heats up due to the current. As the wind flows past the wire, the wire will be cooled down resulting in a drop in the measured voltage [116]. This system allows measuring the resultant wind velocity downstream of the rotor system. A series of tests were performed to understand the limitations and characterization of the wind sensor, including measurement lag and sensor response. Measurement lag is the time it takes the sensor to respond to a thrust change. To demonstrate the sensor response and lag, a bump test was conducted that is a large change in RPM was applied and the sensor response was plotted. The result of the bump test is shown in Figure 6-2.

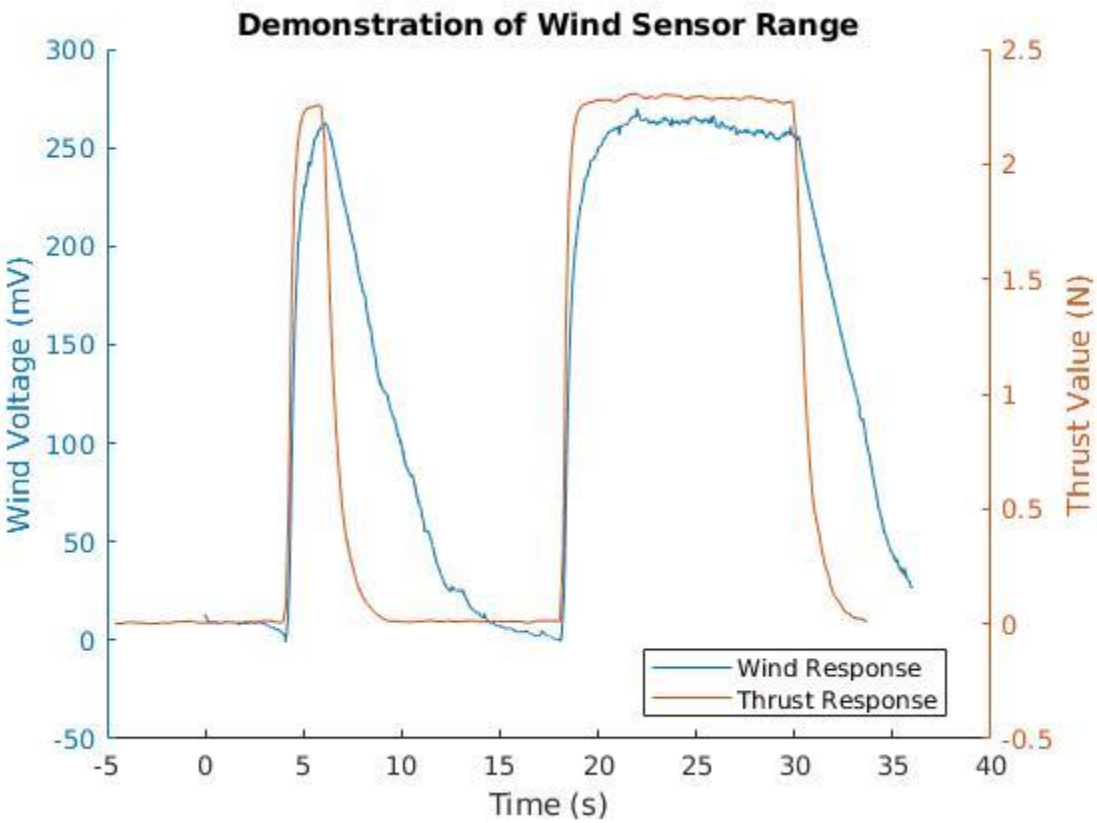

Figure 6-2: Wind sensor response and dynamics

From the sensor's documentation, the wind speed measured by the sensor is given by:

$$W_{speed} = \alpha \frac{\Delta V}{temp^{0.115157}} \tag{6-15}$$

where W_{speed} is the wind speed in $[m/s]$, α is a constant rotation given by the document, ΔV is the difference between the measured voltage and calibrated voltage in $[Volt]$ when there is no wind and $temp$ is the current temperature measured by the sensor in $[°C]$.

6.3.2 Adaptive Fuzzy Kalman Filter

In general, to avoid a divergence in the filtered data, accurate *a priori* information of the measurement and process noise matrices are required [117], [118]. With imperfect *a priori* information, a different procedure is needed, e.g., the use of adaptation technique through the estimation filter. Several adaptive estimation techniques were proposed for adapting the Kalman filter using R or Q matrices in [119]–[121]. In this research, an adaptive fuzzy Kalman filter

(AFKF) has been used to improve the performance of the conventional Kalman filter. The filter is used to enhance the measurement noise rejection and the navigational states estimation. The filter incorporates fuzzy logic rulesto adjust the variances of the R and Q matrices. These rules are adopted from previous work [122], [123]. The recurrent adjustment process of the variances of the statistical matrices R_k is designed based on an IAE approach. This approach is based on a covariance-matching algorithm which examines the degree of the mismatch (DoM) between the actual covariance v_{k+1} of the residual with its theoretical value S_{k+1}

$$DoM = S_{k+1} - C_{k+1} \tag{6-16}$$

After calculating DoM, a fuzzy inference system has been tuned, similar to the one proposed in [122], to compute the adjustment of the measurement noise matrix $\mathrm{R_{k+1}}$. Theoretically, any change in $\mathrm{R_k}$ leads to a change in the covariance matrix $\mathrm{S_{k+1}}$, accordingly. Hence, the DoM can be reduced by adjusting the value of $\mathrm{R_{k+1}}$ at each state estimation cycle

$$R_{k+1} = R_k + \delta R_{k+1} \tag{6-17}$$

Figure 6-3 demonstrates the thrust estimation using the conventional Kalman filter (KF) and the AFKF estimation techniques. It is apparent that the AFKF has exhibited better performance in terms of rejecting the noise associated with the measurement compared to the traditional KF. Figure 6-4 presents the estimation errors of the position states. The AFKF shows lower magnitudes of estimation error compared to those of the conventional KF.

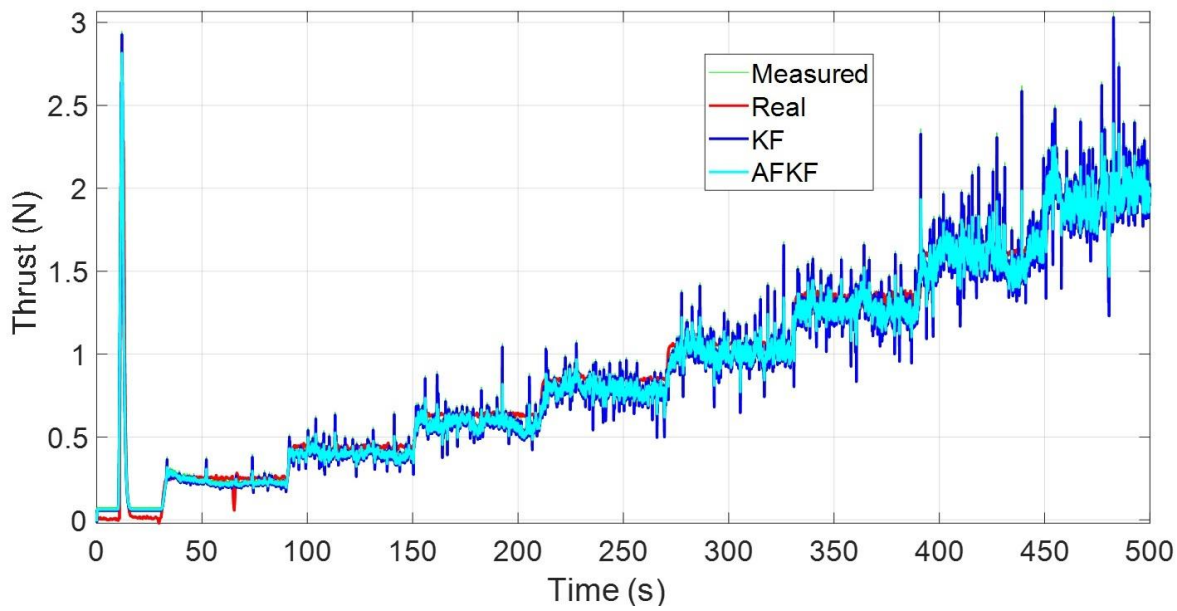

Figure 6-3: Thrust state estimation

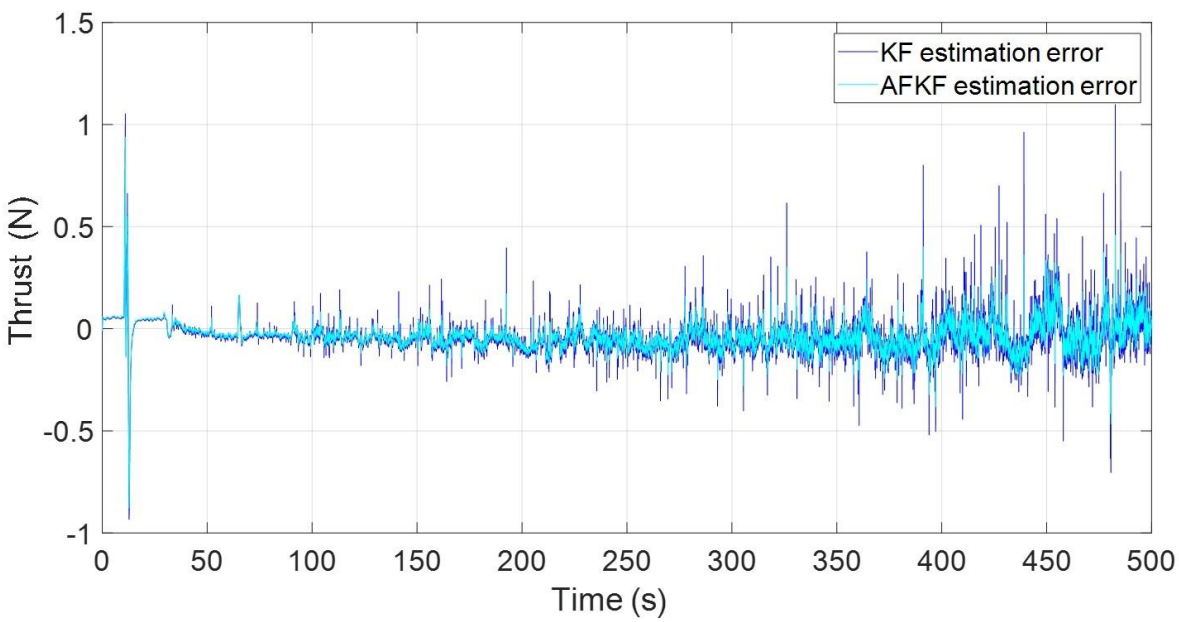

Figure 6-4: Thrust estimation error

6.4 Control Development

The objective of the proposed control system is to achieve system stability and tracking for the desired command signals in Cartesian space $[x_d, y_d, z_d]$ and heading angle ψ_d despite having the failure of one or two opposite rotors. Due to the underactuated nature of the system,

a cascade feedback strategy is adopted here similar to the one developed in Chapter 5. Figure 6-5 illustrates the overall control architecture proposed in this chapter.

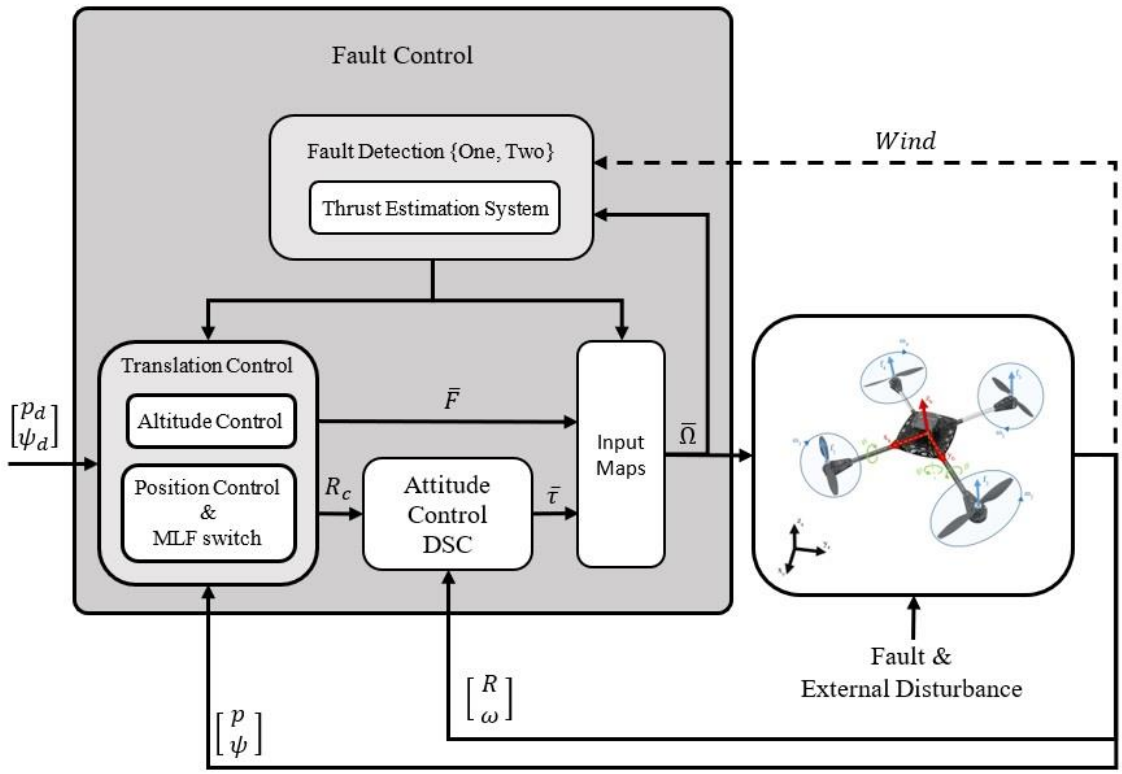

Figure 6-5: Block Diagram of the fault tolerant control system.

The outer-loop assigns a command rotation signal R_c and altitude command signal $\bar{u}_v$ using the desired position p_d and heading angle ψ_d. If the system is at fault, the translation control will drop the control of the heading angle and use the current heading angle ψ to generate a proper rotation command signal R_c. In case of two motor failure the translation control will use the MLF mechanism to select a proper planner signal to track. The inner-loop control the attiude dynamics and generate the torque signal $\bar{u}_\omega$. The input map is selected depending on the number and order of faulty rotors (i.e., one or any two opposite rotors).

6.4.1 Fault-Free Control

Let us start by introducing the fault-free control system. The control system is used to fly the system to follow the desired command signal in the absence of any fault in the vehicle rotors.The position dynamics are given by (6-7) and (6-8). The goal of the position control is to generate a proper attitude command R_c for the attitude controller. The command signal is developed in the SO (3), hence it avoids singularities of Euler angles and unwinding quaternions. The control system is based on the one developed earlier in Chapter 5.

Using the equations (5-14) and (5-15), the input signal u_v is expressed as follows

$$\bar{u}_v = \left(R_{3,3}\hat{\theta}_v\right)^{-1} q_z \tag{6-18}$$

$$q_z = \dot{v}_z - F_z - \lambda_z \dot{e}_z - K_z s_z \tag{6-19}$$

Then, the rotation matrix is utilized similar to (5-30)-(5-35). Thus,

$$q_p = \begin{bmatrix} q_{p_x} \\ q_{p_y} \\ q_{p_z} \end{bmatrix} = \dot{v}_d - \widehat{D}_v - F_v - \lambda_p e_v - K_p s_p \tag{6-20}$$

$$b_{3_c} = \frac{q_p}{\|q_p\|} \tag{6-21}$$

$$R_c = \left[b_{1_c}, b_{3_c} \times b_{1_c}, b_{3_c}\right] \tag{6-22}$$

where b_{1_c} is function of ψ_c and orthogonal to b_{3_c}. The attitude control is adopted from the result of Chapter 4 expressed in equations (4-26) and (4-27). Using the DSC technique, the control signal is defined as follows

$$\bar{u}_\omega = \hat{\theta}_\omega^{-1}(q_\omega) \tag{6-23}$$

$$q_\omega = \dot{\omega}_d - F_\omega - \widehat{D}_\omega - K_\omega s_\omega \tag{6-24}$$

where $\bar{u}_\omega$ vector is the input control signal for the attitude dynamics.

6.4.2 Single-Rotor Failure

With three propellers remaining and $f_4 = 0$, it will be harder for the quadrotor to control its six degrees of freedom. As the stability of the vehicle does not depend on the heading angle,

the control strategy will relinquish the control of ψ in order to control the other attitude angles, i.e. $\bar{u}_{\omega_r} = 0$. Hence, the quadrotor will start spinning about its primary axis and tilting its axis to control the translational movement. The developed control system will be able to stabilize the vehicle without the need to change the control structure. That is, the solution is to keep using the same control system presented in the fault-free case but using a modified input map. Nevertheless, one needs to replace the heading angle used in (6-22) to current heading angle, i.e. $\psi_c = \psi$. This is done to construct the command rotation matrix R_c without controlling the heading angle.

As discussed earlier, the input map of the vehicle changes when a fault happens as shown in (6-7) and (6-8). In the case of a single rotor failure, the rotor command signals can be obtained using the following relation

$$\begin{bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 1 & 0 & -2/l \\ 2 & l & 0 \\ 1 & 0 & 2/l \end{bmatrix} \begin{bmatrix} \bar{u}_v \\ \bar{u}_{\omega_p} \\ \bar{u}_{\omega_q} \end{bmatrix} \tag{6-25}$$

Although the system omits the control of the yaw angle $\bar{u}_{\omega_r} = 0$, the resultant torque around the z-axis will not be zero as seen in (9):

$$\tau_r = K_d(f_1 - f_2 + f_3) \tag{6-26}$$

This means that the system will spin around its z-axis, as expected. Using this will allow the system to control its translation motion and track the desired position command precisely

6.4.3 Two-Rotor Failure

With two propellers remaining and $f_4 = f_2 = 0$, the system will relinquish the motion signal around the x-axis, i.e. $u_{\omega_p} = 0$. However, in this case, omitting the control of heading angle alone will not be enough to control the translational motion. Thus, the developed control system will apply a change on the position control using a switching mechanism based on Multiple Lyapunov Functions (MLFs). To control the planner dynamics using one input signal

u_{ω_r}, the switching control will allow the system to switch between two degrees of freedom, i.e., x-axis and y-axis, and choose the one with the highest error adopting the following Lyapunov functions

$$V_x = \frac{1}{2} s_{p_x}^2 \tag{6-27}$$

$$V_y = \frac{1}{2} s_{p_y}^2 \tag{6-28}$$

The domain with the highest error will be selected as the active domain, while the control signal defined in (6-20) for the other domain will be replaced by zero. That is

$$\begin{aligned} V_x \geq V_y \rightarrow q_{p_y} = 0 \\ V_x < V_y \rightarrow q_{p_x} = 0 \end{aligned} \tag{6-29}$$

Thus, the remaining control input u_{ω_q} will be able to stabilize the system without the need to change the attitude control structure. Nevertheless, as the command rotation R_c defined in (6-22) will be filtered by a first-order low-pass filter as seen in Chapter 4, the system will not suffer from chattering in its control input.

As discussed earlier, the input map of the vehicle shown in (6-11) and (6-12) changes when a fault happens. In the case of the two-rotor failure, the rotor command signals can be obtained using the following relation

$$\begin{bmatrix} \Omega_1^2 \\ \Omega_3^2 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & -1/l \\ 1 & 1/l \end{bmatrix} \begin{bmatrix} \bar{u}_v \\ \bar{u}_{\omega_q} \end{bmatrix} \tag{6-30}$$

Again, the system omits the control of the yaw angle $\bar{u}_{\omega_r} = 0$ though the resultant torque around the z-axis will not be zero

$$\tau_r = K_d (f_1 + f_3) \tag{6-31}$$

which means the system will again spin around its z-axis.

6.5 Results

The simulation results describe the overall performance of the proposed controller in terms of accuracy and convergence rate in presence of external disturbances and different fault cases. The simulations are carried out by MATLAB SIMULINK®.

Table 6-1 shows the control parameters that are used in the simulation. As the control system employs a cascade structure, the control parameters are selected to have a faster attitude control loop than the position control loop. In addition, the gains for the first-order filters are selected to have a faster response than the system dynamics to reduce their effects on the overall dynamics.

Table 6-1: Fault tolerant control parameters

Parameter	Value	Unit
$\boldsymbol{\lambda_z}$	3	-
$\boldsymbol{k_z}$	1	-
$\boldsymbol{\lambda_p}$	5	-
$\boldsymbol{k_p}$	4	-
$\boldsymbol{k_\omega}$	10	-

6.5.1 Single-Rotor Failure

Similar to the simulation test done in Chapter 5, the desired trajectory is defined as a set of four flight behaviors: vertical take-off, hovering in XY-plane, tracking, and vertical landing. However, in this case, the system will assume to have a total failure of rotor number 4 from the beginning of the simulation. Hence, the control system will drop its ability to control the heading angle and instead control the remaining DOF. The system is required to demonstrate a proper error convergence and an adequate rejection of the external disturbances.

Figure 6-6 shows the position response of the system for the case of single-rotor failure in x-, y- and z-axis. The control system was able to achieve a proper tracking of the position command signals despite the presence of the external disturbances and actuator-failure. Figure

6-7 and Figure 6-8 show the angular rate and Euler angle responses of the system, respectively. As mentioned, the proposed controller adopts the strategy of dropping the control of the yaw angle to control the rest of the system dynamics. As a result, the angular rate around the z-axis shown a constant value of 0.1 [rad/s]. This rate is defined by many factors including motor coefficient, rotors speed, external disturbances and friction.

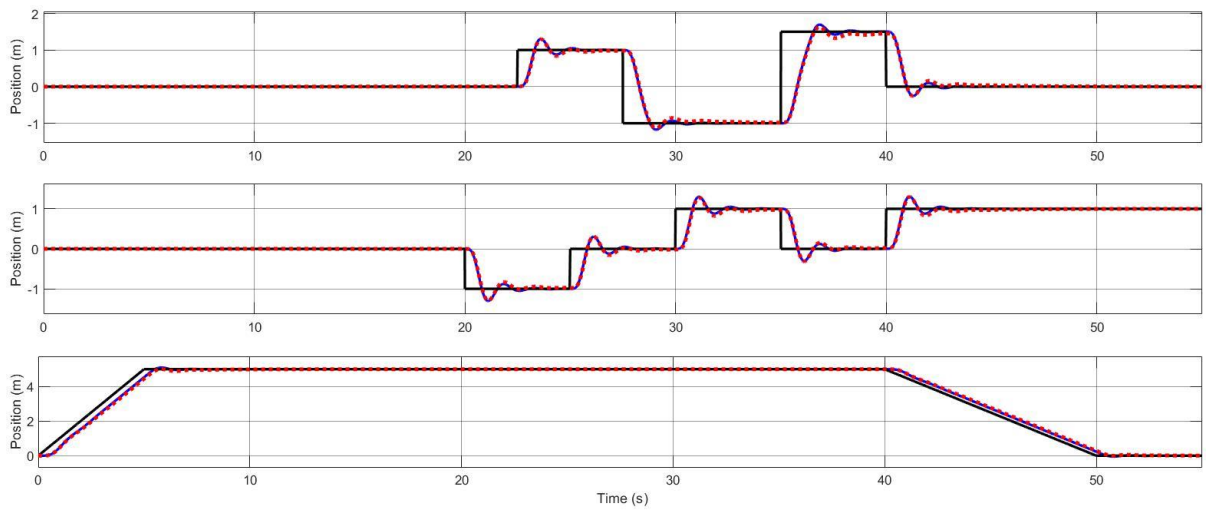

Figure 6-6: Position signals response for the case of a single-rotor failure. The black signal represents the command, the red line represents the desired while the blue represents the actual position signal

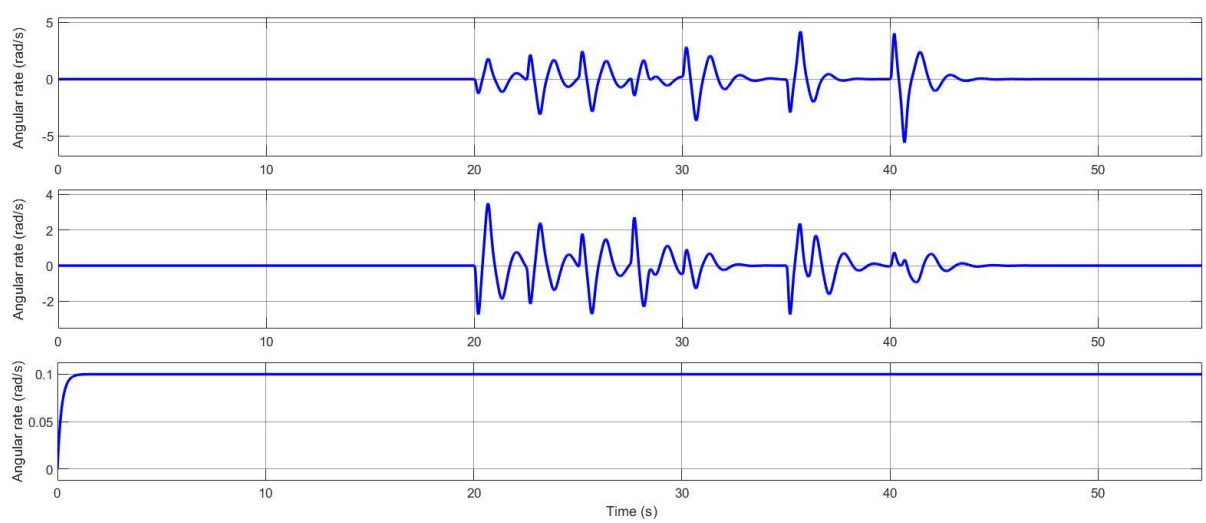

Figure 6-7: Angular rate response for the case of a single rotor failure

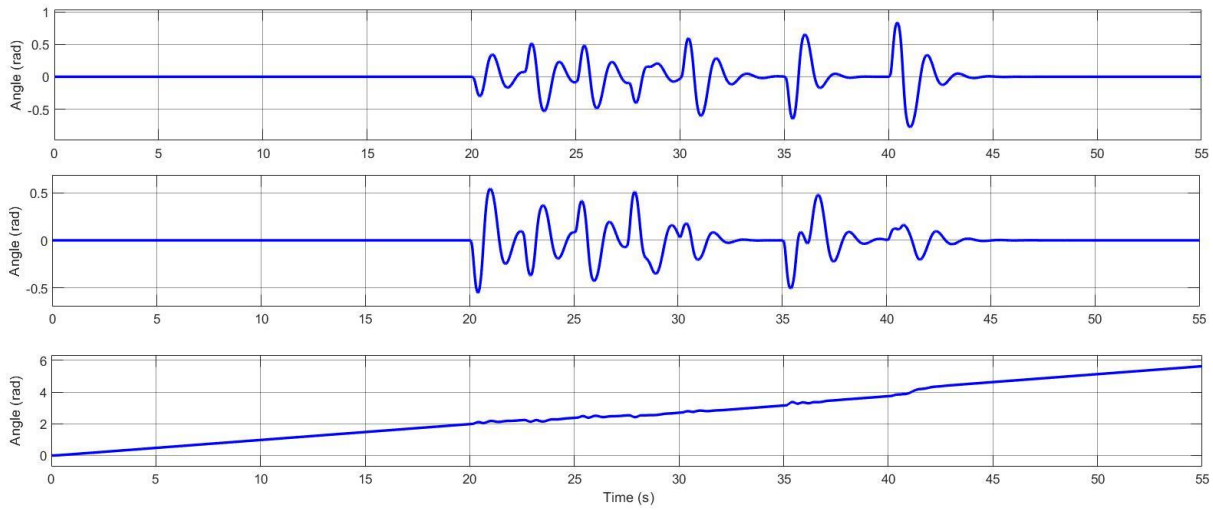

Figure 6-8: Euler angles response for the case of a single rotor failure

The tracking error of the position and Euler angles for the case of single-rotor failure are illustrated in Figure 6-9 and Figure 6-10, respectively. The tracking errors show the capability of the proposed controller to track the command position accurately. The system was able to control its roll and pitch angles, precisely to track the corresponding attitude command despite the presence of external disturbances and failure of actuator number four.

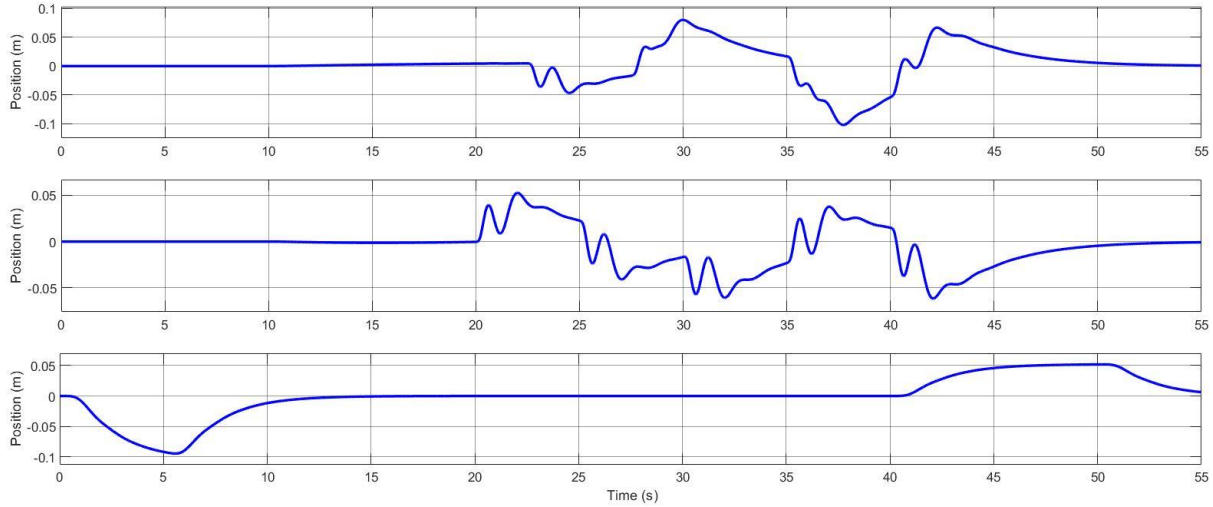

Figure 6-9: Tracking error of the position signals for the case of a single rotor failure

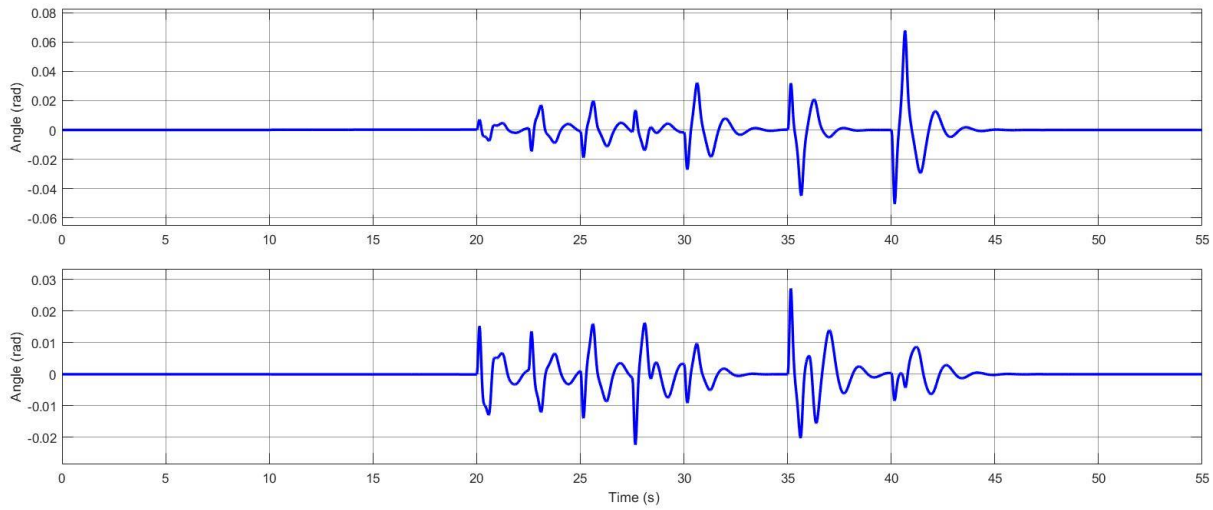

Figure 6-10: Tracking error of roll and pitch angles for the case of a single rotor failure

6.5.2 Two-Rotor Failure

In this case, the vehicle will assume to have a fault in two of its rotors, i.e. $f_2 = f_4 = 0$. Thus, the system will lose its ability to control the roll domain directly, i.e. $\tau_r = 0$. The required performance of the vehicle is to maintain its position despite the presence of fault and external disturbances. The control system will deploy the switching mechanism to give the vehicle the ability to control its planner dynamics using only one input motion, i.e. τ_q.

Figure 6-11 illustrates the response of the position signals in x-, y- and z-axis. It shows a proper tracking to the position command signal. For the planner dynamics, the proposed control system was able to sustain the horizontal position within a small margin, i.e. ± 0.6 $[m]$. The position response shows a slow oscillatory behavior with a low frequency due to the switching mechanism adopted by the control system. Figure 6-12 shows the angular rate response of the quadrotor system. Due to the fact that control system dropped controlling the yaw angle, the system is rotating around it z-axis with a constant speed of around 2 [rad/s]. Figure 6-13 shows the Euler angles response in roll, pitch and yaw angles, respectively. The roll and pitch angles were able to track the rotational command signal R_c generated by the switching mechanism.

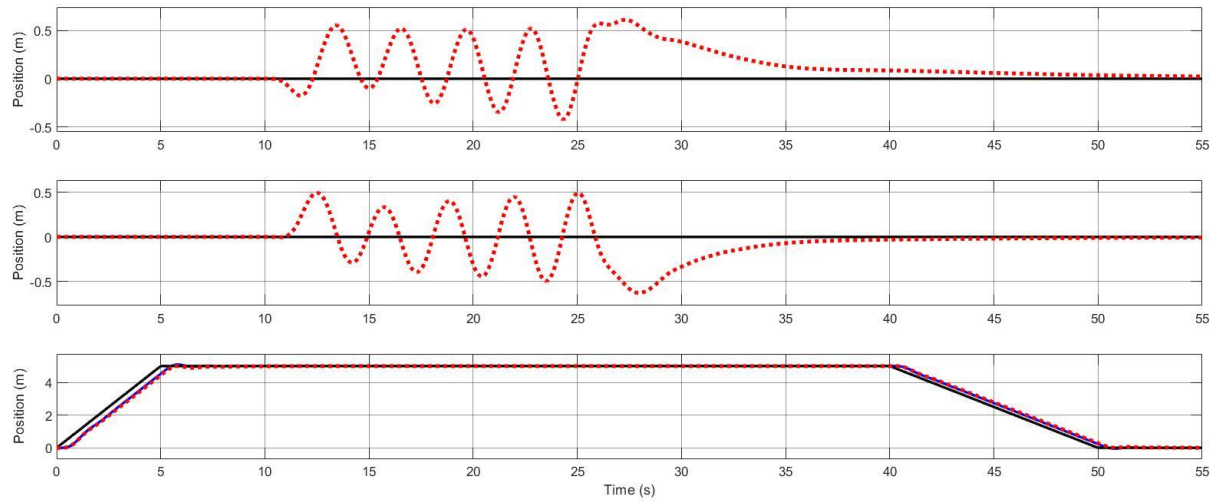

Figure 6-11: Position signals response for the case of two-rotors failure. The black signal represents the command, the red line represents the desired while the blue represents the actual position signal

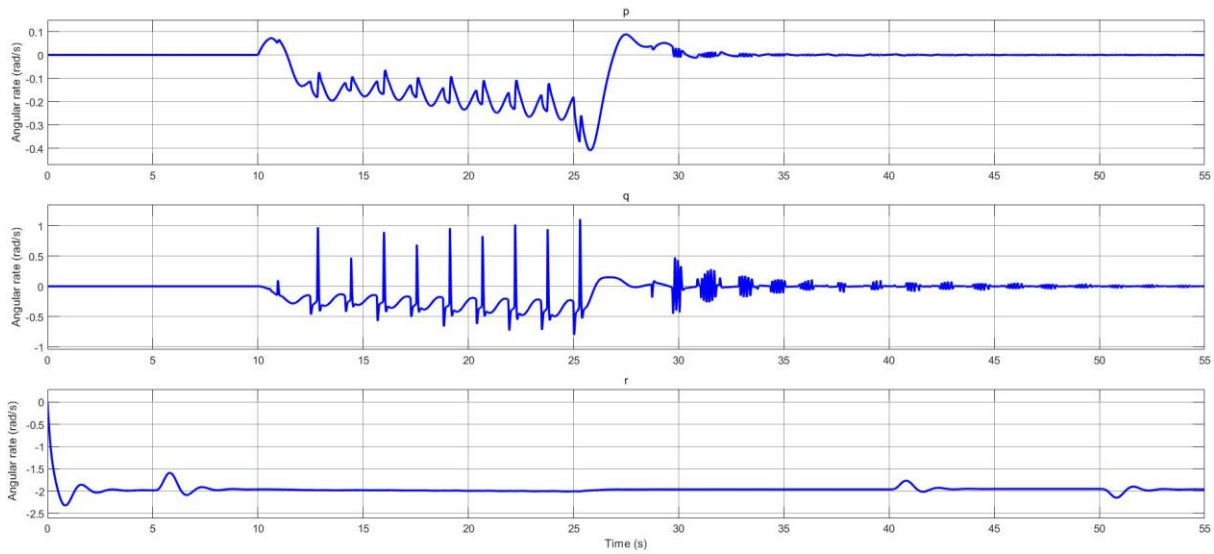

Figure 6-12: Angular rate response for the case of two-rotors failure

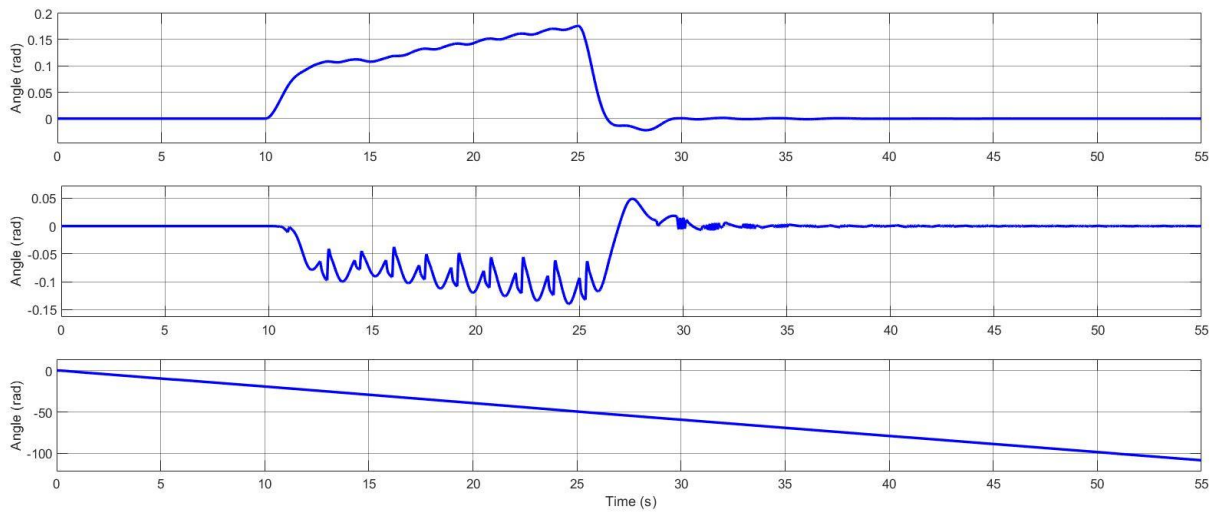

Figure 6-13: Euler angles response for the case of two-rotors failure

The tracking error of the position and Euler angles are shown in Figure 6-14 and Figure 6-15, respectively. It is clearly seen that the proposed controller was able to effectively track the altitude command and maintain a small error in the horizontal axis. In addition, the figures show the controller capability of controlling both the roll and pitch angles despite the loss of two rotors.

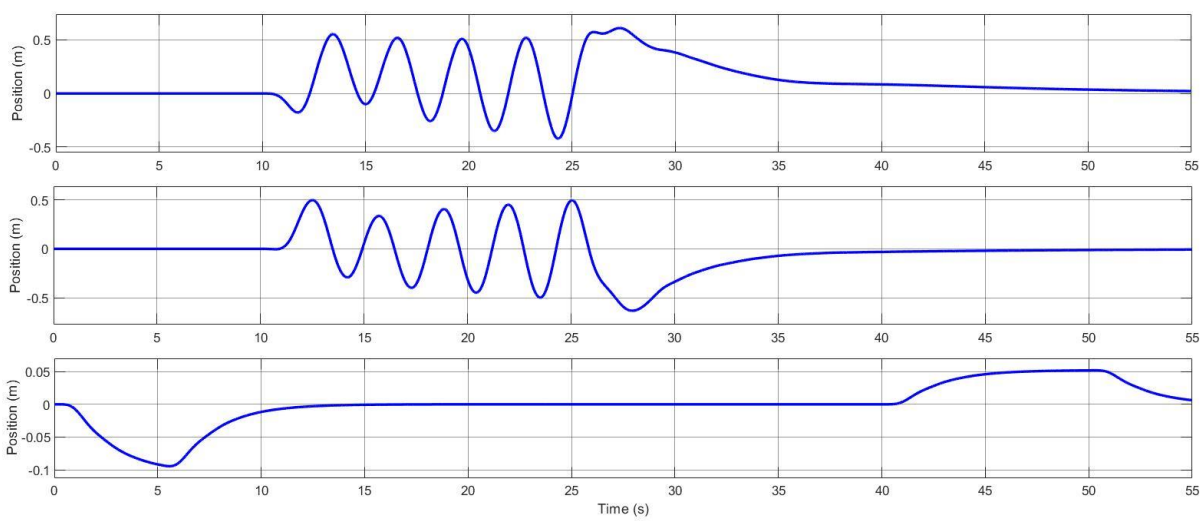

Figure 6-14: Tracking error of the position signals for the case of two-rotors failure

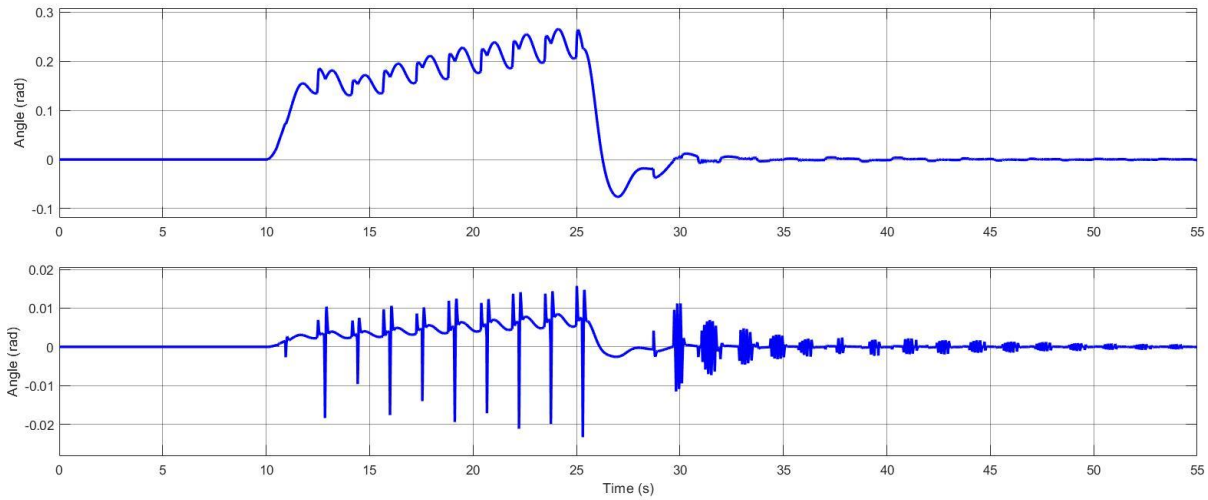

Figure 6-15: Tracking error of the roll and pitch angles for the case of two-rotors failure

6.6 Summary

In this chapter, a fault-tolerant tracking scheme is proposed for a quadrotor UAV in spite of the total failure of one or two adjacent rotors. The proposed scheme consists of two techniques; a fault estimation and a control strategy. On one hand, the fault estimation approach is designed to sense and estimate the thrust produced by the rotor's fixed-pitch blade directly. It consists of integrating a wind sensor, used to measure the wind speed generated by the rotors, and an adaptive Kalman filter, used to reduce the measurement noises. It has been proven that the proposed integration significantly improved disturbance rejection and fault estimation. On the other hand, the proposed control strategy is a generic control feedback that is responsible of stabilizing the control system and achieving path tracking in the presence of actuator failures. In particular, the control strategy adopts a single robust control system that only changes the input mapping when a fault occurs. In construct to the-state-of-the-art solutions, the proposed strategy eliminates the risk of instability which occurs when switching the control system. The proposed scheme has been validated with simulation in case of total failure of one or two adjacent rotors.

Chapter 7: Conclusions

This thesis focused on the development of a practical control scheme for a quadrotor unmanned aerial vehicle (UAV) as an underactuated mechanical system (UMS). The thesis focused on four control challenges namely; underactuation, model uncertainty, extreme external disturbances, and actuator failures. These challenges have been elaborated indifferent flight regimes of the quadrotor while interacting with its environment. These regimes include hovering, aggressive maneuver, near-ground maneuver, and fault-tolerant flight. Each of these flight regimes has its own flight conditions and system performance requirements. Based on a comprehensive literature review, it was concluded that most available control strategies are ad-hoc solutions, as they present a solution for a particular system behavior. Applying these solutions in multi-behavior flight could cause the system to crash. Therefore, a robust tracking switching controller with a region of attraction that covers the entire configuration space was proposed as a stable, versatile and practical control scheme. The proposed control system switches between different designed schemes in real time and engages the most suitable one in the feedback loop to meet the performance requirement of each flight regime. The switching is governed by a carefully designed logic system, known as the control supervisor, based on the sensors readings and current system requirements. The validity of the proposed control scheme was demonstrated by simulation tests and experiments that were carried out on a testbed developed in this research.

This chapter is organized as follows. Section 7.1 presents a summary of the research activities and conclusions of the overall work. The key research contributions and their importance are summarized in Section 7.2. Finally, Section 7.3 points to possible future works that can lead to improving the performance of the proposed control scheme.

7.1 Summary

Due to their unique features, there has been an increasing demand for the use of quadrotors in broad civilian applications, especially in indoor applications where the environment is defined to be complex and cluttered. In this environment, the flight controller needs to employ the quadrotors' full capabilities in order to meet the required flight performances and perform accurate path tracking. Perhaps, the greatest advantage of the quadrotor is its simple structural and mechanical design, but this advantage comes at the cost of a complex control system. Specifically, the control constraints of a quadcopter complicate trajectory tracking and stabilization significantly. The main control constraints elaborated in this research are attributed to underactuation, system failure, input constraints and model uncertainties including parametric and nonparametric ones. Moreover, fully autonomous quadrotors are required to operate in different flight regimes, such as hovering, aggressive maneuver, near-ground maneuver, and fault-tolerant flight to accomplish their tasks. Each of these flight regimes has its own flight constraints and system performance requirements. In order to control the quadrotor using its full capabilities in an uncontrolled environment, one needs a sophisticated control system that is capable of achieving system stability and path tracking simultaneously in the presence of the mentioned challenges. The traditional schemes of solving these challenges involve developing separate flight controllers and switching between them depending on the flight task. Although this scheme simplifies the control problem and the design of the control law, it suffers from the complexity in proving the stability of the overall system and the appearance of chattering in the control signals due to the switching mechanism. In this thesis, the proposed control scheme is based on the idea of splitting the attitude dynamics and translation dynamics, and then controlling each one separately using a different technique. Generally speaking, controlling the altitude dynamics along with the attitude dynamics guarantees the stability of the system in all system behavior despite the presence of the challenges mentioned earlier. However, the attitude dynamics are considered the more

critical dynamics. This is due to the fact that the translational dynamics of the quadrotor directly depend on the attitude but not the other way around. Hence, the attitude dynamics need to be controlled accurately and precisely with a higher response rate in comparison to the translational dynamics. Thus, one needs to avoid chattering in the attitude command signals as this will greatly contribute to exciting the system's highly nonlinear dynamics and, as a result, complicate the stability and tracking problems. In contrast, translation dynamics can be controlled in a lower frequency where the influence of the chattering in the command signals is attenuated on the system performance. The proposed control scheme uses this approach in order to control the quadrotor in different flight regimes. The development of the proposed controller is described in the following three steps.

The first step focuses on developing a globally robust attitude controller for three degrees of freedom UAV. The controller is designed with several novel features that allow the control system to perform better than the existing approaches investigated in the literature review. For start, the structure of the controller is developed using dynamics surface control (DSC) technique. This allows to simplify the control law and avoid an "explosion of terms" associated with the traditional backstepping techniques. In addition, the control system is developed directly on the special Euclidean group with a region of attraction covering the configuration space globally. This allows the quadrotor to perform well in complex and aggressive maneuvers. Furthermore, the robustness feature of the controller allows controlling the quadrotor's attitude despite the presence of disturbances, model uncertainties, as well as, actuator failures. Moreover, as the attitude controller is enabled through the different flight regimes only by changing the mapping of the input without switching the control scheme, there is no risk of instability and chattering in the control signals. Thus, the proposed attitude controller can achieve precise path tracking and meet the required performance of different flight behaviors.

The second step focuses on developing a novel robust position controller capable of controlling the system precisely and eliminating the tracking errors more rapidly than the existing techniques. In particular, the control method combines three techniques: a second order sliding mode control (SMC), a switching mechanism and a non-parametric adaptation mechanism. The SMC is used to guarantee the stability of the position dynamics by generating a proper attitude command for the attitude controller. The switching mechanism is based on multiple Lyapunov functions (MLFs) and enabled in the presence of extreme disturbances to improve the tracking performance. In the case of low disturbances, the controller works in eliminating the tracking error of all translational axes. However, in the case of extreme disturbances, the switching mechanism selects the translational axis with the highest tracking error as the dominant axis and generates a suitable rotation command signal to reduce its tracking error. This technique allows the quadrotor to counteract high disturbances applied to the system, such as the disturbances generated due to the ground effects while, for example, the UAV is performing near ground maneuvers. Finally, the adaptation mechanism includes a radial basis function neural network (RBF-NN) to observe uncertainties without the need for prior training. As a result of using the proposed controller, the result showed a significant improvement in the asymptotical convergence of the tracking error in the presence of external disturbances and uncertainties without a priori knowledge of their bounds.

The third step involves designing a novel real-time fault estimation to detect and diagnose the cause of the occurrence of a fault in a rotor system. In a typical quadrotor system, the rotor systems are operated in open control loops. That is, the thrust generated by the propeller spinning is unknown to the flight controller. In this thesis, a real-time thrust monitoring system is developed to estimate the generated thrust directly. The fault estimation consists of a wind sensor and an adaptive estimation algorithm. The wind sensor is mounted onboard to measure the wind speed generated by the rotor directly. The signal generated by the fault estimator is then used by the supervisor to indicate the health status of the rotors. When a fault

occurs, the supervisor senses the fault and changes the input mapping relationship in the attitude controller. This allows the proposed control system to achieve tracking to any point in the Cartesian space without the need to switch the control law. This is achieved while losing the ability to control the yaw angle, which is justifiable under emergency circumstances.

7.2 Key Contributions

The following outlines the key contributions of this thesis.

- The proposed control system is developed directly on the nonlinear configuration manifold to enable robust tracking with a region of attraction that covers the entire configuration space. In addition, it is developed for the full 6 DOF dynamic model of a small-scale quadrotor without any simplification or assumptions except omitting the motor dynamics.
- The proposed attitude controller is based on dynamic surface control and developed directly on the special Euclidean group with a region of attraction covering the configuration space globally. This will allow the controller to maintain both dynamics stability and tracking performance while avoiding the singularity associated with orientation representations.
- A novel position controller combines a second order sliding mode controller and a switching mechanism based on multiple Lyapunov functions. The proposed controller improves the tracking performance of the translation dynamics in the presence of high disturbances.
- The proposed solution adopts a single control feedback strategy for both the normal flight operation, as well as, when a fault occurs. Hence, there is no risk of instability while switching between control laws. When a fault occurs, the controller changes the input mapping relationship to allow the control system to achieve tracking to any point in the Cartesian space without the need to switch the control law.

- A novel real-time approach to sense and estimate the thrust produced by rotor's fixed-pitch blade is proposed. The approach consists of integrating a wind sensor mounted onboard to measure the wind speed generated by the rotor directly. The integration significantly improves the disturbance rejection and fault estimation of the thrust and rotor, respectively.
- Finally, the proposed work is validated with simulation for agile maneuver and further examined using a testbed subjected to high disturbances. The testbed has 3 DOF for the attitude dynamics and another 3 DOF implemented in simulation for the translational dynamics since the testbed rotates around three axes, but its position is fixed. The performance of the proposed design is compared with that of a traditional PID control, which is commonly used in many research and commercial vehicles. The numerical simulations are carried out using MATLAB SIMULINK®. The results show that the proposed control approach outperforms similar approaches in terms of both accuracy and convergence rate.

7.3 Future Work

It has been demonstrated that the proposed control scheme surpasses other studies in terms of performances while covering the entire configuration space and avoiding the "explosion of terms". The performance of the resultant control system has been validated by both simulation and experiments using 3 DOF testbed. How the control system performs in real-flight experiments will be the topic of further studies. In an uncontrolled environment, the system may be subjected to unexpected and unknown disturbances. Thus, one future direction can be to validate the proposed algorithm against the unmodeled dynamics in 6 DOF. Even though the attitude dynamics have more influence on the stability of the system in comparison with the translation dynamics, real-flight validation tests are still required. Another future direction can be in terms of simplifying the procedure of identifying the control parameters. Although the

proposed control system uses adaptive techniques to compensate for system uncertainties, several tests are still required in order to find suitable control parameters. One solution to that could be using model reference adaptive control in conjunction with the dynamic surface control in the attitude controller. This will contribute to having a consistent system behavior regardless of the actual system parameters with less test. Another possible future direction is validating the fault-tolerant behavior of the proposed scheme in real flights. This was inapplicable to validate due to the physical design of the testbed. This is because the fault-tolerant behavior requires the quadrotor to spin infinitely around its z-axis. Unfortunately, this is not possible with the current design of the testbed.

Having a closed-loop to control the thrust generated by the rotor system allows for a better flight performance. Hence, developing a real-time thrust estimation will contribute to robust and agile maneuvers. Applications for this level of performance include a better landing algorithm, proximity detection for obstacles; such as, walls, better hovering control, and detecting the loss of effectiveness of a rotor system. These applications would all benefit greatly from having a direct measurement of the thrust rather than an estimation, such as the typical procedure. One future direction can be in applying the developed real-time estimation system towards these problems and find a solution that fits the criteria of having a direct measurement of the thrust. Another future direction can be in improving the used estimation algorithm. This includes increasing the measurement range of the wind sensor and improving the response of the Kalman filter in the presence of a wind disturbance. Another future direction can be toward modifying the static nature of the experiments. The estimation algorithm developed in this thesis focused on a fixed and controlled environment with only one motor tested. In real situations, the behavior of the estimation algorithm can be different. Thus, future work can include integrating the wind sensors on multiple rotors when flying the vehicle and testing the response of the algorithm in a real environment.

Bibliography

[1] Amazon, “Prim Air,” 2018. [Online]. Available: https://www.amazon.com/Amazon-Prime-Air/b?ie=UTF8&node=8037720011.

[2] Volocopter, “volocopter,” 2017. [Online]. Available: https://www.volocopter.com/en/.

[3] B. J. Emran, D. D. Tannant, and H. Najjaran, “Low-altitude aerial methane concentration mapping,” *Remote Sens.*, vol. 9, no. 8, 2017.

[4] M. W. Spong, “Underactuated Mechanical Systems,” *Lect. notes Control Inf. Sci. Control Probl. Robot. Autom.*, pp. 135–150, 1998.

[5] Z. Jiang, “Controlling Underactuated Mechanical Systems : A Review and Open Problems,” in *Advances in the Theory of Control, Signals and Systems with Physical Modeling*, 2011, pp. 77–88.

[6] Y. Liu and H. Yu, “A survey of underactuated mechanical systems,” *IET Control Theory Appl.*, vol. 7, no. 7, pp. 921–935, 2013.

[7] X. Xin and Y. Liu, *Control Design and Analysis for Underactuated Robotic Systems*. Springer, 2014.

[8] R. Tedrake, “Underactuated Robotics: Algorithms for Walking, Running, Swimming, Flying, and Manipulation (Course Notes for MIT 6.832).”

[9] S. Gupte, P. I. T. Mohandas, and J. M. Conrad, “A survey of quadrotor unmanned aerial vehicles,” in *Conference Proceedings - IEEE SOUTHEASTCON*, 2012.

[10] K. Máthé and L. Buşoniu, “Vision and Control for UAVs: A Survey of General Methods andof Inexpensive Platforms for Infrastructure Inspection,” *Sensors*, vol. 15, no. 7, pp. 14887–14916, 2015.

[11] G. Heredia *et al.*, “Control of a multirotor outdoor aerial manipulator,” in *IEEE International Conference on Intelligent Robots and Systems*, 2014, pp. 3417–3422.

[12] R. Mahony, V. Kumar, and P. Corke, “Multirotor Aerial Vehicles: Modeling, Estimation,

and Control of Quadrotor," *IEEE Robot. {&} Autom. Mag.*, vol. 19, no. 3, pp. 20–32, 2012.

[13] A. Symington, R. De Nardi, S. Julier, and S. Hailes, "Simulating quadrotor UAVs in outdoor scenarios," in *IEEE International Conference on Intelligent Robots and Systems*, 2014, pp. 3382–3388.

[14] G. Darivianakis, K. Alexis, M. Burri, and R. Siegwart, "Hybrid Predictive Control for Aerial Robotic Physical Interaction towards Inspection Operations," pp. 53–58, 2014.

[15] B. J. Emran and H. Najjaran, "A review of quadrotor: An underactuated mechanical system," *Annu. Rev. Control*, Oct. 2018.

[16] B. J. Emran and H. Najjaran, "Robust Adaptive Control for Quadrotor with Bounded Gain Forgetting," in *Canadian Society for Mechanical Engineering (CSME)*, 2016.

[17] A. Roza and M. Maggiore, "A class of position controllers for underactuated VTOL vehicles," *IEEE Trans. Automat. Contr.*, vol. 59, no. 9, pp. 2580–2585, 2014.

[18] R. Naldi, M. Furci, R. G. Sanfelice, and L. Marconi, "Global trajectory tracking for underactuated VTOL aerial vehicles using a cascade control paradigm," in *IEEE Conference on Decision and Control (CDC)*, 2013, pp. 4212–4217.

[19] A. Das, F. Lewis, and K. Subbarao, "Backstepping approach for controlling a quadrotor using lagrange form dynamics," *J. Intell. Robot. Syst. Theory Appl.*, vol. 56, no. 1–2, pp. 127–151, 2009.

[20] D. Gautam and C. Ha, "Control of a quadrotor using a smart self-tuning fuzzy PID controller," *Int. J. Adv. Robot. Syst.*, vol. 10, pp. 1–9, 2013.

[21] Q. Li, J. Qian, Z. Zhu, X. Bao, M. K. Helwa, and A. P. Schoellig, "Deep neural networks for improved, impromptu trajectory tracking of quadrotors," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 5183–5189.

[22] R. A. Decarlo, M. S. Branicky, S. Pettersson, and B. Lennartson, "Perspectives and results on the stability and stabilizability of hybrid systems," *Proc. IEEE*, vol. 88, no. 7, pp. 1069–1082, 2000.

[23] M. S. Branicky, “Multiple Lyapunov functions and other analysis tools for switched and hybrid systems,” *IEEE Trans. Automat. Contr.*, vol. 43, no. 4, pp. 475–482, 1998.

[24] G. Nikolakopoulos and K. Alexis, “Switching networked attitude control of an unmanned quadrotor,” *Int. J. Control. Autom. Syst.*, vol. 11, no. 2, pp. 389–397, 2013.

[25] A. Yesildirek and B. Imran, “Nonlinear Control of Quadrotor Using Multi Lyapunov Functions,” in *American Control Conference (ACC)*, 2014.

[26] B. J. Emran and H. Najjaran, “Switching control of quadrotor with adaptation mechanism,” in *2016 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2016 - Conference Proceedings*, 2017, pp. 4872–4877.

[27] L. Wang and J. Su, “Switching control of attitude tracking on a quadrotor UAV for large-angle rotational maneuvers,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 2907–2912.

[28] H. Voos and H. Bou-Ammar, “Nonlinear tracking and landing controller for quadrotor aerial robots,” in *IEEE International Conference on Control Applications*, 2010, pp. 2136–2141.

[29] D. Cabecinhas, R. Naldi, C. Silvestre, R. Cunha, and L. Marconi, “Robust landing and sliding maneuver hybrid controller for a quadrotor vehicle,” *IEEE Trans. Control Syst. Technol.*, vol. 24, no. 2, pp. 400–412, 2016.

[30] C. D. McKinnon and A. P. Schoellig, “Unscented External Force and Torque Estimation for Quadrotors,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 5651–5657.

[31] M. Blanke, J. Lunze, M. Kinnaert, M. Staroswiecki, and J. Schröder, *Diagnosis and fault-tolerant control*, 3rd ed. 20. Springer, 2006.

[32] E. Sobhani-Tehrani and K. Khorasani, *Fault diagnosis of nonlinear systems using a hybrid approach*. Springer, 2009.

[33] Y. Zhang and J. Jiang, “Bibliographical review on reconfigurable fault-tolerant control

systems," *Annu. Rev. Control*, vol. 32, no. 2, pp. 229–252, 2008.

[34] C. Rabbath and N. Lechevin, *Safety and Reliability in Cooperating Unmanned Aerial Systems*. New Jersey: World Scientific, 2010.

[35] F. Forte, R. Naldi, A. Serrani, and L. Marconi, "Control of modular aerial robots: Combining under- and fully-actuated behaviors," in *Proceedings of the IEEE Conference on Decision and Control*, 2012, pp. 1160–1165.

[36] R. Naldi, F. Forte, A. Serrani, and L. Marconi, "Modeling and Control of a Class of Modular Aerial Robots Combining under Actuated and Fully Actuated Behavior," *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 5, pp. 1869–1885, 2015.

[37] R. Oung, "The Distributed Flight Array," ETH ZURICH, 2013.

[38] R. Oung and R. D'Andrea, "The Distributed Flight Array: Design, implementation, and analysis of a modular vertical take-off and landing vehicle," *Int. J. Rob. Res.*, vol. 33, no. 3, pp. 375–400, 2014.

[39] J. I. Giribet, R. S. Sánchez-Peña, and A. S. Ghersin, "Analysis and design of a tilted rotor hexacopter for fault tolerance," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 52, no. 4, pp. 1555–1567, 2016.

[40] C. D. Pose, J. I. Giribet, and A. S. Ghersin, "Hexacopter fault tolerant actuator allocation analysis for optimal thrust," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2017, pp. 663–671.

[41] M. Saied *et al.*, "Fault Diagnosis and Fault-Tolerant Control Strategy for Rotor Failure in an Octorotor," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 5266–5271.

[42] A. Freddi, S. Longhi, and A. Monteriù, "A diagnostic thau observer for a class of unmanned vehicles," *J. Intell. Robot. Syst. Theory Appl.*, vol. 67, no. 1, pp. 61–73, 2012.

[43] C. Zhaohui, H. Noura, T. B. Susilo, and Y. Al Younes, "Engineering implementation on fault diagnosis for quadrotors based on nonlinear observer," in *IEEE Chinese Control and*

Decision Conference (CCDC), 2013, no. 25th, pp. 2971–2975.

[44] Z. Cen, H. Noura, T. B. Susilo, and Y. Al Younes, “Robust Fault Diagnosis for quadrotor uavs using Adaptive Thau observer,” *J. Intell. Robot. Syst. Theory Appl.*, vol. 73, no. 1–4, pp. 573–588, 2014.

[45] L. Ma and Y. Zhang, “DUKF-based GTM UAV fault detection and diagnosis with nonlinear and LPV models,” in *IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA)*, 2010, pp. 375–380.

[46] M. H. Amoozgar, A. Chamseddine, and Y. Zhang, “Experimental Test of a Two-Stage Kalman Filter for Actuator Fault Detection and Diagnosis of an Unmanned Quadrotor Helicopter,” *J. Intell. Robot. Syst.*, vol. 70, no. 1–4, pp. 107–117, 2013.

[47] M. Ranjbaran and K. Khorasani, “Fault recovery of an under-actuated quadrotor aerial vehicle,” in *IEEE Conference on Decision and Control (CDC)*, 2010, pp. 4385–4392.

[48] W. Hao and B. Xian, “Nonlinear adaptive fault-tolerant control for a quadrotor UAV based on immersion and invariance methodology,” *Nonlinear Dyn.*, vol. 90, no. 4, pp. 2813–2826, 2017.

[49] Y. M. Zhang *et al.*, “Development of advanced FDD and FTC techniques with application to an unmanned quadrotor helicopter testbed,” *J. Franklin Inst.*, vol. 350, no. 9, pp. 2396–2422, 2013.

[50] A. Akhtar, S. L. Waslander, and C. Nielsen, “Fault tolerant path following for a quadrotor,” in *IEEE Conference on Decision and Control (CDC)*, 2013, pp. 847–852.

[51] M. W. Mueller and R. D’Andrea, “Relaxed hover solutions for multicopters: Application to algorithmic redundancy and novel vehicles,” *Int. J. Rob. Res.*, vol. 35, no. 8, pp. 873–889, 2016.

[52] V. Lippiello, F. Ruggiero, and D. Serra, “Emergency landing for a quadrotor in case of a propeller failure: A PID approach,” in *IEEE International Conference on Intelligent Robots and Systems (ICRA)*, 2014, no. August 2015, pp. 4782–4788.

[53] V. Lippiello, F. Ruggiero, and D. Serra, “Emergency landing for a quadrotor in case of a propeller failure: A backstepping approach,” in *IEEE International Conference on Intelligent Robots and Systems (ICRA)*, 2014, pp. 4782–4788.

[54] B. J. Emran, J. Dias, L. Seneviratne, and G. Cai, “Robust adaptive control design for quadcopter payload add and drop applications,” *Chinese Control Conf. CCC*, vol. 2015–Septe, pp. 3252–3257, 2015.

[55] B. Min, J. Hong, and E. Matson, “Adaptive robust control (ARC) for an altitude control of a quadrotor type UAV carrying an unknown payloads,” in *International Conference on Control, Automation and Systems*, 2011, pp. 1147–1151.

[56] E. Lavretsky and K. A. Wise, *Robust and Adaptive Control: with Aerospace Applications*. New York;London: Springer, 2013.

[57] J.-J. E. Slotine and W. Li, *Applied Nonlinear Control*. Prentice-Hall, 1991.

[58] Y. C. Choi and H. S. Ahn, “Nonlinear control of quadrotor for point tracking: Actual implementation and experimental tests,” *IEEE/ASME Trans. Mechatronics*, vol. 20, no. 3, pp. 1179–1192, 2015.

[59] Mu Huang, Bin Xian, Chen Diao, Kaiyan Yang, and Yu Feng, “Adaptive tracking control of underactuated quadrotor unmanned aerial vehicles via backstepping,” in *IEEE American Control Conference (ACC)*, 2010, pp. 2076–2081.

[60] C. Ha, Z. Zuo, F. B. Choi, and D. Lee, “Passivity-based adaptive backstepping control of,” *Rob. Auton. Syst.*, vol. 62, no. 9, pp. 1305–1315, 2014.

[61] H. Yang and D. Lee, “Dynamics and Control of Quadrotor with Robotic Manipulator,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 5544–5549.

[62] H. N. Nguyen, C. Ha, and D. Lee, “Mechanics, control and internal dynamics of quadrotor tool operation,” *Automatica*, vol. 61, pp. 289–301, 2015.

[63] I. H. Choi and H. C. Bang, “Adaptive command filtered backstepping tracking controller

design for quadrotor unmanned aerial vehicle," *Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng.*, vol. 226, no. 5, pp. 483–497, 2012.

[64] Y. Al Younes, A. Drak, H. Noura, A. Rabhi, and A. El Hajjaji, "Quadrotor Position Control Using Cascaded Adaptive Integral Backstepping Controllers," *Appl. Mech. Mater.*, vol. 565, pp. 98–106, 2014.

[65] Z. Fang and W. Gao, "Adaptive integral backstepping control of a Micro-Quadrotor," in *Proceedings of the 2nd International Conference on Intelligent Control and Information Processing, ICICIP 2011*, 2011, no. PART 2, pp. 910–915.

[66] Z. Fang and W. Gao, "Adaptive backstepping control of an indoor micro-quadrotor," *Res. J. Appl. Sci. Eng. Technol.*, vol. 4, no. 21, pp. 4216–4226, 2012.

[67] M. Mohammadi and A. M. Shahri, "Adaptive nonlinear stabilization control for a Quadrotor UAV: Theory, simulation and experimentation," *J. Intell. Robot. Syst. Theory Appl.*, vol. 72, no. 1, pp. 105–122, 2013.

[68] M. Schreier, "Modeling and adaptive control of a quadrotor," in *IEEE International Conference on Mechatronics and Automation*, 2012, pp. 383–390.

[69] Z. T. Dydek, A. M. Annaswamy, and E. Lavretsky, "Adaptive control of quadrotor UAVs: A design trade study with flight evaluations," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 4, pp. 1400–1406, 2013.

[70] H. Liu, Y. Bai, G. Lu, and Y. Zhong, "Robust attitude control of uncertain quadrotors," *IET Control Theory Appl.*, vol. 7, no. 11, pp. 1583–1589, 2013.

[71] H. Liu, J. Xi, and Y. Zhong, "Robust motion control of quadrotors," *J. Franklin Inst.*, vol. 351, no. 12, pp. 5494–5510, 2014.

[72] H. Liu, X. Wang, and Y. Zhong, "Quaternion-based robust attitude control for uncertain robotic quadrotors," in *IEEE Transactions on Industrial Informatics*, 2015, vol. 11, no. 2, pp. 406–415.

[73] Y. Shtessel, C. Edwards, L. Fridman, and A. Levant, *Sliding mode control and*

observation. 2014.

[74] D. Lee, H. J. Kim, and S. Sastry, “Feedback linearization vs. adaptive sliding mode control for a quadrotor helicopter,” *Int. J. Control. Autom. Syst.*, vol. 7, no. 3, pp. 419–428, 2009.

[75] J.-J. Xiong and G. Zhang, “Discrete-time sliding mode control for a quadrotor UAV,” *Opt. - Int. J. Light Electron Opt.*, vol. 127, no. 8, pp. 3718–3722, 2016.

[76] G. Bartolini, A. Ferrara, and E. Usai, “Chattering avoidance by second-order sliding mode control,” *IEEE Trans. Automat. Contr.*, vol. 43, no. 2, pp. 241–246, 1998.

[77] E.-H. Zheng, J.-J. Xiong, and J.-L. Luo, “Second order sliding mode control for a quadrotor UAV,” *ISA Trans.*, vol. 53, no. 4, pp. 1350–1356, 2014.

[78] J. J. Xiong and E. H. Zheng, “Position and attitude tracking control for a quadrotor UAV,” *ISA Trans.*, vol. 53, no. 3, pp. 725–731, 2014.

[79] J.-J. Xiong and G.-B. Zhang, “Global fast dynamic terminal sliding mode control for a quadrotor UAV,” *ISA Trans.*, vol. 66, pp. 233–240, 2017.

[80] L. Luque-Vega, B. Castillo-Toledo, and A. G. Loukianov, “Robust block second order sliding mode control for a quadrotor,” *J. Franklin Inst.*, vol. 349, no. 2, pp. 719–739, 2012.

[81] L. F. Luque-Vega, B. Castillo-Toledo, A. Loukianov, and L. E. Gonzalez-Jimenez, “Power line inspection via an unmanned aerial system based on the quadrotor helicopter,” in *Proceedings of the Mediterranean Electrotechnical Conference - MELECON*, 2014, pp. 393–397.

[82] L. Besnard, Y. B. Shtessel, and B. Landrum, “Quadrotor vehicle control via sliding mode controller driven by sliding mode disturbance observer,” *J. Franklin Inst.*, vol. 349, no. 2, pp. 658–684, 2012.

[83] H. Ramirez-Rodriguez, V. Parra-Vega, A. Sanchez-Orta, and O. Garcia-Salazar, “Robust backstepping control based on integral sliding modes for tracking of quadrotors,” *J. Intell. Robot. Syst. Theory Appl.*, vol. 73, no. 1–4, pp. 51–66, 2014.

[84] X. Gong, Y. Bai, Z. Hou, C. Zhao, Y. Tian, and Q. Sun, “Backstepping sliding mode tracking control of quad-rotor under input saturation,” *Int. J. Intell. Comput. Cybern.*, vol. 5, no. 4, pp. 515–532, 2012.

[85] C. Peng, Y. Bai, X. Gong, Q. Gao, C. Zhao, and Y. Tian, “Modeling and Robust Backstepping Sliding Mode Control with Adaptive RBFNN for a Novel,” *IEEE/CAA J. Autom. Sin.*, vol. 2, no. 1, pp. 56–64, 2015.

[86] F. Berkenkamp and A. P. Schoellig, “Safe and robust learning control with Gaussian processes,” in *European Control Conference (ECC)*, 2015, pp. 2496–2501.

[87] T. Lee, “Robust Adaptive Attitude Tracking on SO (3) With an Application to a Quadrotor UAV,” *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 5, pp. 1924–1930, 2013.

[88] F. A. Goodarzi, D. Lee, and T. Lee, “Geometric Adaptive Tracking Control of a Quadrotor UAV on SE(3) for Agile Maneuvers,” *J. Dyn. Syst. Meas. Control*, vol. 137, no. 9, p. 091007, 2014.

[89] B. J. Emran and A. Yesildirek, “Robust Nonlinear Composite Adaptive Control of Quadrotor,” *Int. J. Digit. Inf. Wirel. Commun.*, vol. 4, no. 2, pp. 213–225, 2014.

[90] B. J. Emran and H. Najjaran, “Adaptive neural network control of quadrotor system under the presence of actuator constraints,” in *2017 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2017*, 2017, vol. 2017–Janua.

[91] C. Nicol, C. J. B. MacNab, and A. Ramirez-Serrano, “Robust adaptive control of a quadrotor helicopter,” *Mechatronics*, vol. 21, no. 6, pp. 927–938, 2011.

[92] B. Zhao, B. Xian, Y. Zhang, and X. Zhang, “Nonlinear Robust Adaptive Tracking Control of a Quadrotor UAV Via Immersion and Invariance Methodology,” *IEEE Trans. Ind. Electron.*, vol. 62, no. 5, pp. 2891–2902, 2015.

[93] A. H. Glattfelder and W. Schaufelberger, *Control Systems with Input and Output Constraints*, 1es ed. New York;London: Springer, 2003.

[94] L. Luo and D. E. Miller, “Near optimal lqr performance for uncertain first order systems,”

Int. J. Adapt. Control Signal Process., vol. 18, no. 4, pp. 349–368, 2004.

[95] B. Zhou, Z. Lin, and G. R. Duan, “L∞ and L2 low-gain feedback: Their properties, characterizations and applications in constrained control,” *IEEE Trans. Automat. Contr.*, vol. 56, no. 5, pp. 1030–1045, 2011.

[96] B. S. Park, J. Y. Lee, J. B. Park, and Y. H. Choi, “Adaptive control for input-constrained linear systems,” *Int. J. Control. Autom. Syst.*, vol. 10, no. 5, pp. 890–896, 2012.

[97] B. L. Stevens, F. L. Lewis, and E. N. Johnson, “Adaptive Control With Application To Miniature,” in *Aircraft Control and Simulation: Dynamics, Controls Design, and Autonomous Systems,* Third Edit., John Wiley & Sons, Inc, 2015.

[98] M. W. Mueller and R. D’andrea, “Stability and control of a quadrocopter despite the complete loss of one, two, or three propellers.”

[99] A. Puri, “A Survey of Unmanned Aerial Vehicles (UAV) for Traffic Surveillance,” *Tech. Pap.*, pp. 1–29, 2005.

[100] J. E. Gentle, *Matrix Algebra Theory, Computations, and Applications in Statistics*. Springer, 2007.

[101] M.-D. Hua, T. Hamel, P. Morin, and C. Samson, “Introduction to feedback control of underactuated VTOLvehicles: A review of basic control design ideas and principles,” *IEEE Control Syst.*, vol. 33, no. 1, pp. 61–75, 2013.

[102] X. Zhang, X. Li, K. Wang, and Y. Lu, “A survey of modelling and identification of quadrotor robot,” *Abstract and Applied Analysis*, vol. 2014. pp. 1–16, 2014.

[103] Y. Choi, H. A.-I. transactions on mechatronics, and undefined 2015, “Nonlinear control of quadrotor for point tracking: Actual implementation and experimental tests,” *ieeexplore.ieee.org*.

[104] B. Song and K. Hedrick, *Dynamic Surface Control of Uncertain Nonlinear Systems, An LMI Approach*. 2011.

[105] J. Liu, *Radial Basis Function (RBF) Neural Network Control for Mechanical Systems*

Design, Analysis and Matlab Simulation, vol. 1. SpringerLink ebooks, 2013.

[106] F. a. Goodarzi, D. Lee, and T. Lee, “Geometric Adaptive Tracking Control of a Quadrotor Unmanned Aerial Vehicle on SE(3) for Agile Maneuvers,” *J. Dyn. Syst. Meas. Control*, vol. 137, no. 9, p. 091007, 2015.

[107] A. Das, F. Lewis, and K. Subbarao, “Sliding Mode Approach to Control Quadrotor Using Dynamic Inversion,” in *Intechopen.Com*, InTech, 2006.

[108] A. U. Awan, J. Park, and H. J. Kim, “Thrust estimation of quadrotor UAV using adaptive observer,” *2011 11th Int. Conf. Control Autom. Syst.*, pp. 131–136, 2011.

[109] D. W. Yeo, N. Sydney, D. A. Paley, and D. Sofge, “Onboard Flow Sensing for Downwash Detection and Avoidance with a Small Quadrotor Helicopter.”

[110] B. Arain and F. Kendoul, “Real-Time Wind Speed Estimation and Compensation for Improved Flight,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 50, no. 2, pp. 1599–1606, 2014.

[111] N. Sydney, B. Smyth, and D. A. Paley, “Dynamic control of autonomous Quadrotor flight in an estimated wind field,” *Proc. IEEE Conf. Decis. Control*, pp. 3609–3616, 2013.

[112] M. Bangura and R. Mahony, “Nonlinear dynamic modeling for high performance control of a quadrotor,” in *Australasian Conference on Robotics and Automation, ACRA*, 2012.

[113] M. Bangura and R. Mahony, “Thrust Control for Multirotor Aerial Vehicles,” *IEEE Trans. Robot.*, vol. 33, no. 2, pp. 390–405, 2017.

[114] RCbenchmark, “Series 1580 - Thrust Stand and Dynamometer | RCbenchmark,” 2018. [Online]. Available: https://www.rcbenchmark.com/dynamometer-series-1580/. [Accessed: 26-Oct-2018].

[115] Modern Device, “Wind Sensor Rev. P - Low Cost Anemometer | Modern Device,” 2018. [Online]. Available: https://moderndevice.com/product/wind-sensor-rev-p/. [Accessed: 26-Oct-2018].

[116] S. Corrsin, “Extended applications of the hotwire anemometer. Investigations of the flow

in round, turbulent jets," California Institute of Technology.

[117] A.H.Mohamed and K. P. Schwarz, "Adaptive Kalman Filtering for INS/GPS," *J. Geod.*, vol. 73, no. 4, pp. 193–203, 1999.

[118] P. J. Escamilla-Ambrosio and N. Mort, "Development of a fuzzy logic-based adaptive Kalman filter," in *Control Conference (ECC), 2001 European*, 2001, pp. 1768–1773.

[119] K. D. Sebesta and N. Boizot, "A real-time adaptive high-gain EKF, applied to a quadcopter inertial navigation system," *IEEE Trans. Ind. Electron.*, vol. 61, no. 1, pp. 495–503, 2014.

[120] A. Makni, H. Fourati, and A. Y. Kibangou, "Energy-Aware Adaptive Attitude Estimation under External Acceleration for Pedestrian Navigation," *IEEE/ASME Trans. Mechatronics*, vol. 21, no. 3, pp. 1366–1375, 2016.

[121] C. M. Lin and C. S. Hsueh, "Adaptive EKF-CMAC-based multisensor data fusion for maneuvering target," *IEEE Trans. Instrum. Meas.*, vol. 62, no. 7, pp. 2058–2066, 2013.

[122] M. K. Al-Sharman, M. A. Jaradat, and M. F. Abdel-Hafez, "Intelligent attitude and flapping angles estimation of flybarless helicopters under near-hover conditions," *J. Franklin Inst.*, vol. 335, no. 14, pp. 6830–6858, 2018.

[123] M. K. Al-Sharman, B. J. Emran, M. A. Jaradat, H. Najjaran, R. Al-Husari, and Y. Zweiri, "Precision landing using an adaptive fuzzy multi-sensor data fusion architecture," *Appl. Soft Comput. J.*, vol. 69, 2018.