PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA

MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH
UNIVERSITY OF SAAD DAHLAB BLIDA 1
INSTITUTE OF AERONAUTICS AND SPACE STUDIES

# « Quadrotor Attitude Tracking Using Nonlinear Controllers:
# - Robust Optimal Adaptive Controller
# - Geometric Adaptive Controller »

A Thesis Presented to the Department of
Construction
in Partial Fulfillment
of the Requirements
for the Degree of
Master in
Avionics Engineering

Studied and presented by:

**- ARBAOUI SABRINA**

**- BELATTAR AYA HADJER**

A Thesis Prepared under the Supervision of:

**Dr. SALIHA BENCHEIKH**

College year 2019/2020

# Acknowledgements

First of all, countless thanks to ALLAH the Creator of the Universe who endowed us with intelligence, and kept us healthy to complete this year of study.

We would like to express our gratitude to the advisor of this thesis, Dr. Saliha BENCHEIKH for entrusting us with this work and supervising it throughout its completion. We thank her for her patience, her availability and, above all, her judicious advice, which contributed to our reflection.

We would like to express our deepest appreciation to Dr.Ali Dali for his help and support in this thesis.

A big thank to the professors and the entire teaching team at the Institute of Aeronautics and Space Studies in Blida, who provided us with the tools we needed to succeed in our studies for 5 years.

To everyone who helped and motivated us during this difficult year.

THANK YOU

This brief was made possible thanks to the help of several people to whom I would like to express my gratitude, especially my lovely partner Sabrina for her patience and support.

I deeply thank all my parents who have always supported and encouraged me as academics, and especially my mother who has always supported and pushed me to give the best of myself.

I would like to express my gratitude to my friends and colleagues Marwa, Nesrine, Nassima who have provided me with moral and intellectual support throughout my process.

Finally, I greet here all those whose names I have not mentioned and who have contributed directly or indirectly to the completion of this work

Belattar Aya Hadjer

There are a number of individuals who have supported me through my aspiration of completing this thesis. I make a humble attempt to thank a few of them here. First and foremost, I would like to express my deepest gratitude to my partner Aya Hadjer “This thesis made us not only partners but also sisters for the entire life INSHALLAH”.

Overall, I want to express my warmest thanks to my parents, my super woman in this life “Mom” and my idol “Dad” for all that they gave me through my life. A big thanks to my siblings RADHIA and KHALED, their kids and to my cousins HOUDA, YAHIA, AYMEN and SOFIANE also to my best friends YASMINE and MONTACER for their unconditional love and support in all possible ways.

My most sincere thanks To all my friends, back in Ouargla and the ones that I made here in Blida, who always supported me in many different means and for sharing so many fun and stress-relieving moments, as well as the support and encouragement.

To all who helped and supported me in this life.

To the memory of my best teachers ever Mr.Mim and Mr.Baba Hamou.

Finally, I wish to express heartfelt gratitude to myself for everything.

ARBAOUI SABRINA

# Abstract:

This thesis focuses on the design and development of two independent nonlinear control methods for attitude tracking of quadrotors UAV. We first introduce the concept of the quadrotor, where the nonlinear dynamic model of quadrotor is obtained by using Newton's equations of motion along with the set of frames necessary. Then, a detailed study about each controller is presented where the first controller is a robust optimal adaptive which deals with tracking problem in presence of parametric uncertainties, actuator amplitude constraints, and unknown time-varying external disturbances. This method uses a nonlinear disturbance observer that was integrated with an adaptive control to handle external disturbance and parametric uncertainties respectively. While a PSO algorithm is proposed to tackle input constraints.  The second one presents a geometric adaptive control system with state inequality constraints for the attitude dynamics of a rigid body which was presented on $SO(3)$ manifolds avoiding undesired regions and deal with external disturbances. The control systems are designed such that the desired attitude is stabilized based on Lyapunov theory. Finally, to test and validate the proposed controllers, several simulation using Matlab code are presented with a comparison between the developed controllers in term of their dynamic performance.

**Keywords:** Quadrotor UAV, attitude tracking control, ROAC, GAC, Lyapunov theory.

# Résumé :

Cette thèse se concentre sur la conception et le développement de deux méthodes de contrôle non linéaires indépendantes pour le suivi d'attitude des drones quadrirotors. Nous introduisons d'abord le concept du quadrirotor, où le modèle dynamique non linéaire du quadrirotor est obtenu en utilisant les équations de mouvement de Newton avec les différents repères utilisés. Ensuite, une étude détaillée sur chaque contrôleur est présentée dans laquelle le premier contrôleur est un adaptatif optimal robuste qui traite le problème de suivi en présence d'incertitudes paramétriques, de contraintes d'amplitude d'actionneur et de perturbations externes variant dans le temps inconnu. Cette méthode utilise un observateur de perturbations non linéaire qui a été intégré à un contrôle adaptatif pour éliminer respectivement les perturbations externes et les incertitudes paramétriques. Alors qu'un algorithme PSO est proposé pour gérer les contraintes d'entrée. Le second présente un système de contrôle adaptatif géométrique avec des contraintes d'inégalité d'état pour la dynamique d'attitude d'un corps rigide qui a été présenté sur des variétés SO (3) évitant les régions indésirables et traitant les perturbations externes. Les systèmes de contrôle sont conçus de telle sorte que l'attitude souhaitée est stabilisée sur la base de la théorie de Lyapunov. Enfin, pour tester et valider les contrôleurs proposés, plusieurs simulations utilisant le code Matlab sont présentées afin de comparer entre les contrôleurs développés en terme de leurs performances dynamiques.

**Mots clés** : UAV quadrirotor, contrôle de suivi d’attitude, ROAC, GAC, théorie de Lyapunov.

# ملخص :

تركز هذه الأطروحة على تصميم وتطوير طريقتين مستقلتين للتحكم غير الخطي لتتبع موقف الطائرات بدون طيار الرباعية. نقدم أولاً مفهوم الرباعي، حيث يتم الحصول على النموذج الديناميكي غير الخطي للرباعي باستخدام معادلات نيوتن للحركة جنبًا إلى جنب مع مجموعة المعالم اللازمة. بعد ذلك، يتم تقديم دراسة مفصلة حول كل طريقة تحكم حيث تكون طريقة التحكم الأولى وسيلة تكيفية مثالية قوية تتعامل مع مشكلة التتبع في ظل وجود شكوك معلميه وقيود سعة المشغل والاضطرابات الخارجية غير المعروفة المتغيرة بمرور الوقت. تستخدم هذه الطريقة مراقب اضطراب غير خطي تم دمجه مع عنصر تحكم تكيفي للتعامل مع الاضطرابات الخارجية والشكوك المعلمية على التوالي. بينما تم اقتراح خوارزمية PSO لمعالجة قيود الإدخال. تمثل طرقة التحكم الثانية نظام تحكم هندسي تكيفي مع قيود عدم المساواة في الحالة لديناميكيات الموقف لجسم صلب والذي تم تقديمه على متشع(3)SO لتجنب المناطق غير المرغوب فيها والتعامل مع الاضطرابات الخارجية. تم تصميم أنظمة التحكم بحيث يتم تثبيت الموقف المطلوب بناءً على نظرية Lyapunov.

أخيرًا، لاختبار والتحقق من صحة أنظمة التحكم المقترحة، يتم تقديم محاكاة باستخدام كود Matlab للمقارنة بين أنظمة التحكم المطورة من حيث أدائها الديناميكي.

**الكلمات المفتاحية:** طائرة بدون طيار الرباعية، تتبع موقف، للتحكم غير الخطي، النموذج الديناميكي.

# TABLE OF CONTENTS

## 2. QUADROTOR DESCRIPTION

## 3. DESCRIPTION OF ATTITUDE CONTROLLERS ROAC and GAC

## 4. SIMULATION RESULTS

# LIST OF TABLES

**Page**

# LIST OF FIGURES

**Page**

## CHAPTER 1

## CHAPTER 2

## CHAPTER 3

## CHAPTER 4

# LIST OF ABREVIATIONS

| | |
|---|---|
| VTOL | Vertical Take-Off and Landing |
| CTOL | Conventional take-off and landing |
| UAV | Unmanned Arial Vehicle |
| GPS | Global Positioning System |
| DCM | Direction Cosine Matrix |
| DOF | Degrees-Of-Freedom |
| MEMS | Micro Electro Mechanical Systems |
| FPV | First Person View |
| IMU | Inertial Measurement Unit |
| LIDAR | Light Detection And Ranging |
| IR | Infrared |
| COM | Center of Mass |
| ECI | Earth Centered Inertial |
| ECEF | Earth Centered Earth-Fixed |
| PID | Proportional Integral Derivative |
| LQR | Linear Quadratic Regulator |
| LQG | Linear Quadratic Gaussian |
| MPC | Model Predictive Control |
| NDO | Non-linear Disturbance Observer |
| SMC | Sliding Mode Control |
| ROAC | Robust Optimal Adaptive Control |
| GAC | Geometric Adaptive Control |
| PSO | Particle Swarm Optimization |

# Nomenclature

| | |
|---|---|
| $\{\mathscr{B}\}$ | Body fixed  reference frame. |
| $\{I\}$ | Inertial reference frame. |
| $\{n\}$ | Navigation reference frame. |
| $F_I$ | Inertial frame. |
| $F_B$ | Body fixed frame. |
| $R_1$ | Transformation matrix from the inertial frame to the body fixed frame. |
| $R$ | Transformation matrix from the body fixed frame to the Inertial frame. |
| SO(3) | Special Orthogonal Group |
| SE (3) | Special Euclidean group |
| $\phi$ | Latitude. |
| $\lambda$ | Longitude. |
| $h$ | Height. |
| $(\varphi, \theta, \psi)$ | Euler angles (roll, pitch, yaw). |
| $I_d$ | Identity matrix. |
| $x_b$ | Vector represented in body fixed frame. |
| $\vec{x}$ | Vector represented in Inertial frame. |
| $\omega_{1,2,3,4}$ | Propeller's speed. |
| $Q$ | Quaternion. |
| $\Omega_E$ | Angular velocity expressed in the inertial reference frame. |
| $I_r$ | Inertia matrix of rigid body. |
| $I_{xx}$ | Inertia on x axis. |
| $I_{yy}$ | Inertia on y axis. |
| $I_{zz}$ | Inertia on z axis. |
| $J_r$ | Rotor Inertia. |

| | |
|---|---|
| $L$ | Arm length. |
| $g$ | Gravitational earth vector magnitude g = 9.81(m/s2). |
| $r$ | Position of center of gravity. |
| $\eta$ | Attitude. |
| $[u\ v\ w]^T$ | Linear velocities expressed in the body reference frame. |
| $[\dot{x}\ \dot{y}\ \dot{z}]^T$ | Linear velocities expressed in the inertial earth reference frame. |
| $[p\ q\ r]^T$ | Angular velocities expressed in the body reference frame. |
| $[\dot{\phi}\ \dot{\theta}\ \dot{\psi}]^T$ | Angular velocities expressed in the inertial earth reference frame. |
| $\overrightarrow{\dot{V}_E}$ | Linear acceleration expressed in the inertial earth reference frame. |
| $m$ | Mass of the quadrotor. |
| $\vec{F}$ | Quadrotor forces vector expressed in the inertial earth reference frame. |
| $\vec{\vec{u}}$ | Quadrotor torque vector expressed in the body reference frame. |
| $\vec{d}$ | External disturbance. |
| $f_i$ | Thrust force. |
| $b$ | Thrust factor. |
| $D$ | Drag factor. |
| $Q_i$ | Drag moment. |
| $x_d = [z_d\ \phi_d\ \theta_d\ \psi_d]^T$ | Desired altitude and attitude. |
| $s$ | The velocity error. |
| $\Gamma_1, \Gamma_2, \Lambda$ | Symmetric positive definite matrices. |
| $\hat{a}$ | Estimated parameters. |
| $\dot{\hat{D}}$ | Estimated disturbance derivative. |
| $\chi$ | The state vector of the exogenous system. |
| $L(x)$ | The nonlinear observer gains. |
| $\dot{e}$ | The observer error dynamic. |
| $V(x, e)$ | Lyapunov function. |
| $\Delta$ | The disturbance. |
| $\Psi$ | The configuration error. |
| $R_d$ | The desired attitude. |

| | |
|---|---|
| $e_R$ | Attitude error vector. |
| $\bar{\Delta}$ | The estimated uncertainty. |

# INTRODUCTION

## BACKGROUND

UAV technology is constantly evolving as a new innovation and a big investment where there has been a constant increase in the use of UAV over the past several decades. The UAV is a flying machine without a pilot on-board. The initial purpose for UAV was military reconnaissance, intelligence and surveillance applications. Furthermore, and because of the development in the electronics field that made a huge leap in microcomputers performance while the size and weight of the components decreases, the drone platforms got out of the military sector, and started to be introduced in the civilian applications such as the data collection, the aerial photography, the agriculture and especially in the commercial field.

Aside from further developing the current applications of UAV systems mentioned before, a large market will be explored is personal air taxi service and package delivery especially with online shopping. The latter is of particular interest to companies such as Amazon and Domino's Pizza (Desjardins, 2018), which are putting their efforts in developing the required technologies. Where according to the Drone Market Report 2020, the global UAV market will grow from $22.5 billion in 2020 to over $42.8 billion in 2025 at a CAGR (Compound annual growth rate) of 13.8% [1]. The reasons for this much interest in UAVs is due to their ability to perform those tasks, which are difficult or dangerous for humans with less cost and less investment of resources.

Due to extensive usage of the UAV, various types of UAVs are produced depending on their applications. Although there is huge variety of drone platforms. For example, there are hybrid, fixed-wing planes, single rotor helicopters, quadrotors… and each of them has advantages and drawbacks. One of the most popular UAV is the Quadrotor, because it has a number of advantages over the other UAVs due to its ability to take-off and land vertically. Furthermore, like a helicopter, the quadrotor can hover, but with its four rotors, it is capable of lifting larger payloads relative to its own weight. In addition, a small-sized quadrotor is agile, highly maneuverable, and is inherently more stable due to the four-rotor design with counter rotating props. Owing to these advantages, and a growing-range of useful applications, the quadrotor has become a popular subject for research. For that, the work developed in this thesis is focused on quadrotors.

Efficient motion control of quadrotors is still an important scientific challenge because the quadrotors are under-actuated systems, which means having fewer actuators than the degree of freedom with highly nonlinear and coupled dynamics. Furthermore, the development of effective quadrotor controllers is essential. The control techniques involved must also improve in order to provide better performance and increased versatility. Linear and nonlinear control methods are proposed in literature to achieve the control objectives. Simplistic linear control techniques were employed for computational ease and stable hover flight such as the classical PID controller, which was, applied in many approaches like Bouabdallah et al [2], [3], [4]. A modified version of the PID controller was presented in [5] also LQR controllers mentioned in [6] and [7], where Bethke et al. developed techniques using LQR to perform flight tests indoors for long duration missions. However, with better modelling techniques and faster on board computational capabilities, comprehensive nonlinear techniques to be run on real-time have become an achievable goal. Nonlinear methodologies promise to rapidly increase the performances for these systems and make them more robust. Various nonlinear control methods have been proposed in literature such as Feedback Linearization in [8], where D. Lee et al. presented two types of controllers for AQ using nonlinear techniques. These included a feedback linearization controller that involved high-order derivative terms. S. Bouabdallah et al. presented applications of control strategies using the backstepping technique in [9], [10]. In [11], a robust sliding mode controller was introduced for trajectory tracking of a quadrotor. All these control methods have been successfully controlled systems with certain and fully known dynamics.

All the control techniques suggested above require complete knowledge of the system model and model parameters. Systems that need to be controlled, such as quadrotor, suffer from uncertainty in its parameters, i.e. errors in the identified values of the parameters, which can lead to significant deterioration of the controller performance. Furthermore, unmodeled variations in system parameters such as mass or inertia during flight can cause significant stabilization errors to occur. This problem could make such systems unstable and harder to control. Therefore, the need for an accurate nonlinear model of quadrotor dynamics can be overcome by using adaptive methods that can react to and correct errors in model parameter estimates, modify parameter estimates when they change. There were many different methods used the adaptive control to deal with parametric uncertainties in literature such as in [12] where a general review of adaptive control applied to quadrotors was presented. In [13], the trajectory tracking control problem was addressed using direct and indirect model reference adaptive control. Decentralized control has been combined with adaptive nonlinear control techniques for quadrotor trajectory tracking. In [14], the quadrotor dynamic was separated into two subsystems, and decentralized adaptive controls were applied to solve the trajectory tracking problem. Similarly, a decentralized adaptive controller was proposed in [15] to stabilize the altitude and attitude of the quadrotor when model uncertainties are present. Let us notice that many of the adaptive controllers in literature present significant disadvantage, which is the external disturbances, that was not taken into account.

During the flight, the quadrotor is usually affected by external disturbances, including severe wind, rain and vibration. A realistic objective when it comes to external disturbances is to reject its effects acting on the UAV. The rejection of unknown external disturbances offers a more challenging task due to the limited available information about the disturbance. The rejection of external disturbances has been attempted in various ways where the robust control was the responsible to remain the stability to the system in the presence of disturbances. Many literatures studied the tracking control of quadrotors in the presence of parametric uncertainties and external disturbances simultaneously, for example: Bialy et al. (2013), Selfridge and Tao (2014), Zhao et al. (2015), Basri et al. (2015), Yang and Yan (2016) …

Unfortunately, the parametric uncertainties and external disturbances are not the only problems facing the quadrotors control, where the input constraints are considered as one of the reasons that reduce the performance of quadcopters by making the optimization problem extremely challenging. Methods control of input constraints was not mentioned in many approaches.

Another problem that face the control of quadrotors is the parameterization of rigid body attitude. We find different types of attitude presentations such as Euler angles, quaternions and matrix rotation. Where it is known that the Euler angles parameterization suffer from singularities, usually, the singularity is because only three parameters are used. As for the quaternions, which use four parameters to represent the attitude, they do not have singularities but they double cover the special orthogonal group what make the representation of attitude has non-unique representation and that called the ambiguities. Where for the rotation matrix that is the natural presentation of attitude is the set of orthogonal matrices whose determinant is one [16], it is a unique and global mathematical parameterization.

In addition to attitude representation, the attitude control suffers from the state constraints where there are regions that must be avoided while tracking attitude of a quadrotor. Several approaches studied the attitude control using the different attitude representations in the presence and absence of constraints.

## MOTIVATION AND OBJECTIVES

The prospected growth of the use of UAVs in the next years requires the managing of these units in a safe and controlled manner. Therefore, this thesis tries to contribute to the integration of UAVs into our daily lives, by designing and validating high performance control algorithms for quadcopters with the goal of robustly tracking attitude control.

To achieve this goal, and to deal with the challenges that arise quadrotors control, two different control approaches are developed. The parameters of the quadrotor are identified to allow the implementation of the two controllers. The study of the kinematics and dynamics will help to understand the physics of the quadrotor and its behavior. Together with the modelling, will help to determine the control algorithm structure to achieve a better stabilization. The whole system is validated and tested using Matlab code.

Many approaches studied the control of quadrotor in the presence of parametric uncertainties, external disturbances and input constraints individually. Nevertheless, there is no study before about tracking control of quadrotor in the presence of the three problems simultaneously. For that, the first objective of this thesis is to develop control algorithm based on approach [17] that deals with parametric uncertainties, external disturbances and input constraints together. The adaptive control is used to tackle with parametric uncertainties by updating an adaptive law that estimates the unknown parameters. While the robust control makes sure the closed loop control system, remain stable in the presence of disturbance using a Nonlinear Disturbances Observer. Finally, a PSO algorithm will minimize the effect of input saturation.

The second objective of this study is to develop a Geometric Adaptive Control based on approach [18]. Geometric control is concerned with the development of control systems for dynamic systems evolving on nonlinear manifolds that cannot be globally identified with Euclidean spaces. By characterizing geometric properties of nonlinear manifolds intrinsically, geometric control techniques completely avoid singularities and ambiguities that are associated with local coordinates or improper characterizations of a configuration manifold [19]. Where we will represent the attitude dynamics on $SO(3)$ and focusing on developing an adaptive attitude control that will deal with inequality constraints. Furthermore, to enable the convergence of attitude in the presence of external disturbances, an adaptive law will be update.

A Lyapunov theory will ensure the stability of the vehicle inside the flight envelope at all times in the both of approaches. A detailed study about the nature of these controllers will be presented in the next chapters of this thesis.

The final objective is to compare and evaluate the overall performance of the developed control algorithms and conclusions are drawn with respect to the effectiveness of each method in rejecting external disturbances, dealing with parametric uncertainties and input constraints.

## THESIS OUTLINE

The outline of this thesis is summarized as follows:

- In Chapter 1, general definitions and information about UAVs will be given in the first section. While in the second section, an overview about the attitude and its representation will be presented.
- In Chapter 2, more detailed information about quadrotors will be given. It also provides a general concept of quadrotor and its basic movements are explained and illustrated. In addition, the quadrotor model is further illustrated by providing some detail using the Newton-Euler method. Furthermore, the literature survey and theoretical background for the linear and nonlinear control techniques for the hovering and trajectory tracking control of the UAV will be presented finally.
- In Chapter 3, the proposed control schemes are described. This is followed by an explanation detailed of the robust optimal adaptive control and the geometric adaptive control. Where the control design of both controllers will be given in addition to their stability analysis.
- In Chapter 4, the aforementioned control schemes are validated in simulation Matlab using a more realistic model. Throughout this chapter, an explanation of the validation model differences, the parameter tuning for the tracking attitude and several simulation tests are developed. Finally, the simulation results are shown for two considered controllers.
- In the last chapter, using the results obtained in simulation, conclusions are drawn on the performance of each controller. The main points reached during the development of the thesis are presented. This thesis is finalized with remarks that could be improved in a future work.

# Chapter 1

# UAV GENERALITIES AND ATTITUDE REPRESENTATION

## 1.1 INTRODUCTION

Technology advancement has a lot of impact on the quality of life. These fast-paced technological advancements became integrated as a precision tool not only in the industry, but also in our social lives. One of this technology researches has made Unmanned Aerial Vehicles (UAVs) or simply drones, a reality which is very trending nowadays.

UAVs have been widely adopted in the military world over the last decade and the success of these military applications is increasingly driving efforts to establish UAVs in civil applications that could avoid risking human lives especially for dangerous situations such as, search and rescue at damaged disaster areas, firefighting etc. Performing these hazardous tasks without direct human interaction is highly valuable.

In aeronautics, piloting an aircraft basically means flying an aircraft, which is related to the capability to control the attitude of the vehicle. While most UAVs are remotely operated, they almost have an on board autopilot in charge of flying the aircraft. For that, for a better performance, controlling attitude poses significant challenges for the UAVs future.

The aim of this chapter is to provide an overview of drones and attitude. In the first section, a literature review about UAVs is presented firstly and then their classification and applications were discussed. In the second section, we will see different types of reference systems also some attitude presentations.

## 1.2 UAV OVERVIEW

### 1.2.1 Technology of UAVs

**U**nmanned **A**erial **V**ehicles (**UAVs**), also known as aerial drones, are an aircraft flying without a human pilot on board, whether their controlling is performed remotely by radio waves or autonomously by on board computers. They are often equipped with accessories used for surveillance and monitoring, in the form of the optoelectronic heads. The design and configuration of drones can vary according to their specifications and applications. They can be very small compared to manned air vehicles since additional space and weight considerations for human crew are not included to the air vehicle. With the latest development of Micro Electro Mechanical Systems (MEMS) technology, human crew with large weight and space can be replaced with very small electronic devices in UAVs. Therefore, for specific operations that require small size aerial vehicles, UAVs can be very effective and they can provide the attack capability for high-risk missions mostly in the military applications.

The UAVs can be used in other applications such as the surveillance, the data collections, photography, commercial and many other fields.

Drones offer less stressful environment. They are used for better decision making and able to fly in dangerous zones.

Drones vary in shape and size and they can be categorized according to weight, flight range, flight altitude, autonomy, and purpose of use, which will be detailed in the following subsection.

## 1.2.2 Classification of UAVs

There is no one standard for the classification of UAVs. Defense agencies have their own standard, and civilians have their ever-evolving loose categories for UAVs. However, we can classify them by size, range and endurance, number of propellers used inside…etc.

Here are two main types of drones from which most are developed

### 1.2.2.1 Classification of drones according to number of Propellers

We have two type of drones with different propellers: Rotary drones and Fixed Wing drones.

#### 1.2.2.1.1 Rotary drones

Rotary wing UAVs feature rotors consisting of blades that revolve around a fixed mast generating lift. The rotors generate vertical thrust by diverting the air downward. These drones can stand stable in the air and they are divided into two types:

- **Single Rotor drones** that look similar in structure and design to actual helicopters. They have one big rotor plus a small sized rotor on the tail for direction and stability. They are strong, durable and have long-lasting flight time but they have a higher complexity and can be expensive. Single rotor drone is usually used for research and surveying.

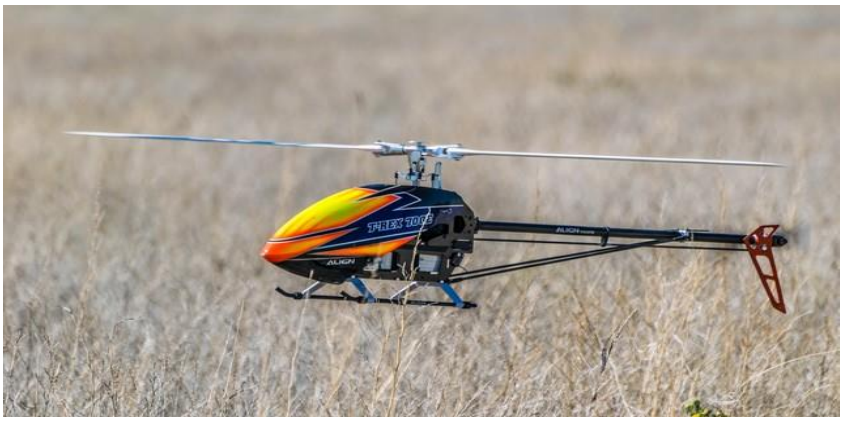

**Figure 1.1:** Single rotor UAV

- **Multirotor** that are the most popular type of drone because of their small size, ready to fly out of the box capabilities and are also the cheapest drone options. They carry several rotors on their body and can be further classified based on the number of them (tricopter, quadcopter, etc.). They are usually used in aerial photography, construction and security.

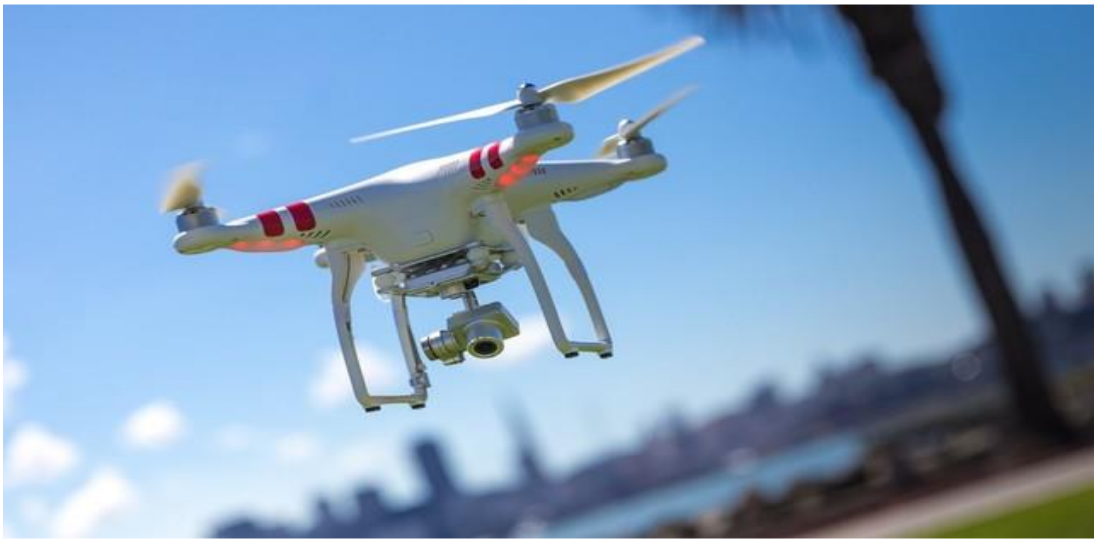

**Figure 1.2:** DJI Phantom Quadrotor

#### 1.2.2.1.2 Fixed Wing Drones

Here is entirely different category from all above. Their designs are unique as compared to commonly used multi rotor type drones. A fixed wing drones consist of one rigid wing, they are designed to look and work like an airplane. These drones are not able to stand stable in one place in air as they are not much powerful to fight against gravitational force. Fixed wing UAVs are well known in the military as they are often used when manned flight is considered too risky or difficult.

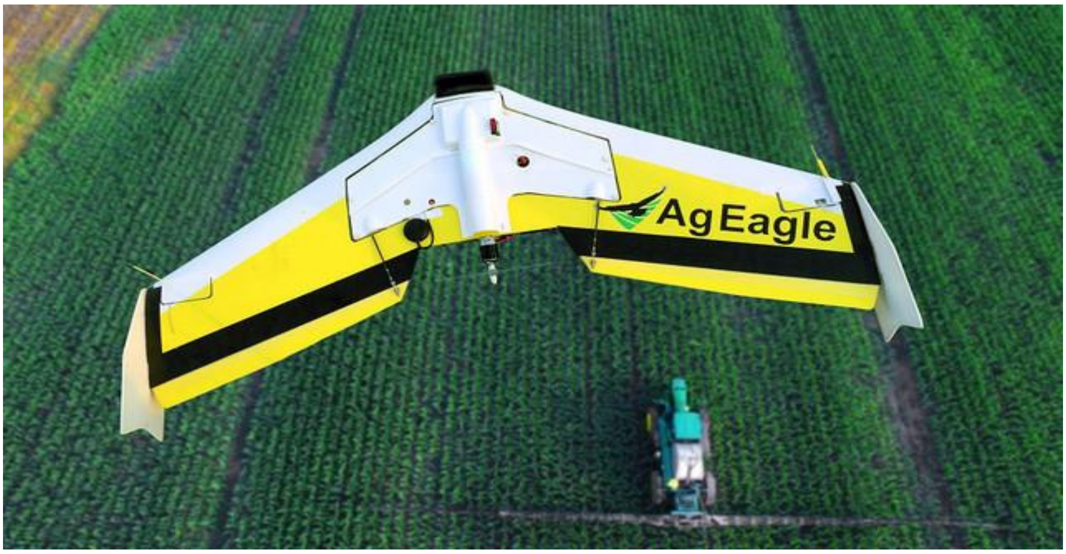

**Figure 1.3:** AgEagle RX60 Fixed wing UAV

#### 1.2.2.1.3 General comparison between fixed and rotary wing UAVs

The main advantages of fixed-wing UAVs are long endurance and high cruise speeds since they are aerodynamically more efficient. Moreover, they are relatively simpler to design compared to rotary-wing. On the other hand, since fixed-wing UAVs require runways or launch systems to take-off and land, they are disadvantageous compared to rotary wing UAVs that have **V**ertical **T**ake-**O**ff and **L**anding (**VTOL**) ability [20], [21] and [22]. Fixed-wing UAVs are also more prone to be damaged during landing compared to rotary UAVs. Summary of the comparison between fixed-wing UAVs with CTOL ability and rotary-wing UAVs with VTOL ability can be seen in Table 1.1.

**Table 1.1:** General comparison between UAV

| | Advantages | Disadvantages |
|---|---|---|
| Rotary-wings<br>VTOL UAVs | -No need for runways<br>-Hovering ability<br>-High manouverability | -Short range and endurance<br>-Limited speed and altitude flight<br>-More complex mechanism<br>-High energy consumption |
| Fixed-wings<br>CTOL UAVs | -Long range and endurance<br>-High speed and altitude flight<br>-More simple mechanism<br>-Energy efficient | -Requires runways<br>-No hovering<br>-Less maneuverability<br>-More prone to be damaged during landing |

The high maneuverability of rotary wing UAVs and the VTOL ability enables to operate in complicated and limited environments. In addition, the hovering ability that is important if operation requires staying in the air at a specific location, such as mapping, aerial photography, surveillance…, made the rotary wing UAVs very popular and the best choice for specific type of missions.

In this thesis, a quadrotor, which is also a rotary wing UAV, was used. In chapter 2, more detailed information about quadrotors will be given and discussed.

### 1.2.2.2 Classification of drones according to size

For classification according to size, we can come up with the sub-classes below.

#### 1.2.2.2.1 Very Small drones

The very small UAV class applies to UAVs with dimensions ranging from the size of a large insect to 30-50 cm long. The insect-like UAVs, with flapping or rotary wings, are a popular micro design. They are extremely small in size, very lightweight, and can be used for spying and biological warfare. Larger ones utilize conventional aircraft configuration. The choice between flapping or rotary wings is a matter of desired maneuverability. Flapping wing-based designs allow perching and landing on small surfaces.

Examples of very small UAVs are the Australian Cyber Technology CyberQuad Mini and Maxi and the US Aurora Flight Sciences Skate.

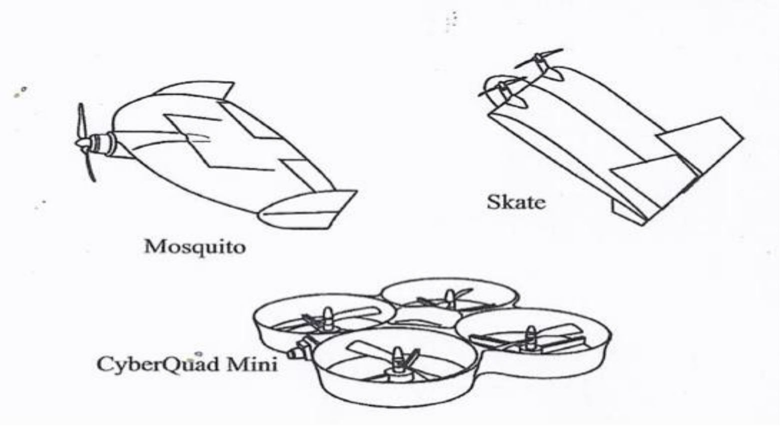

**Figure 1.4:** Very small UAVs

#### 1.2.2.2.2 Small drones

The small UAV class, also called mini-UAV, applies to UAVs that have at least one dimension greater than 50 cm and no larger than 2 meters. Many of the designs in this category are based on the fixed-wing model, and most are hand-launched by throwing them in the air.

Examples of members of this small UAV class are the Turkish Bayraktar and the US Army RQ-7 Shadow.

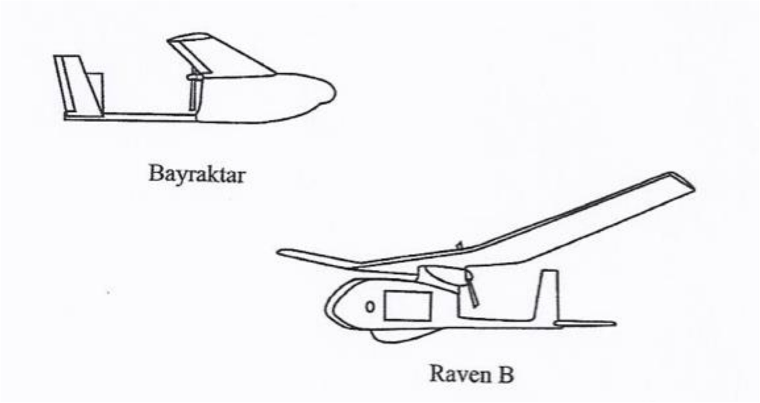

**Figure 1.5:** Small UAVs

#### 1.2.2.2.3 Medium drones

The medium UAV class applies to UAVs that are too heavy to be carried by one person but are still smaller than a light aircraft. They usually have a wingspan of about 5-10 m and can carry payloads of 100 to 200 kg.

Examples of this type are the UK Watchkeeper, the RQ-5A Hunter, and the RQ-2 Pioneer.

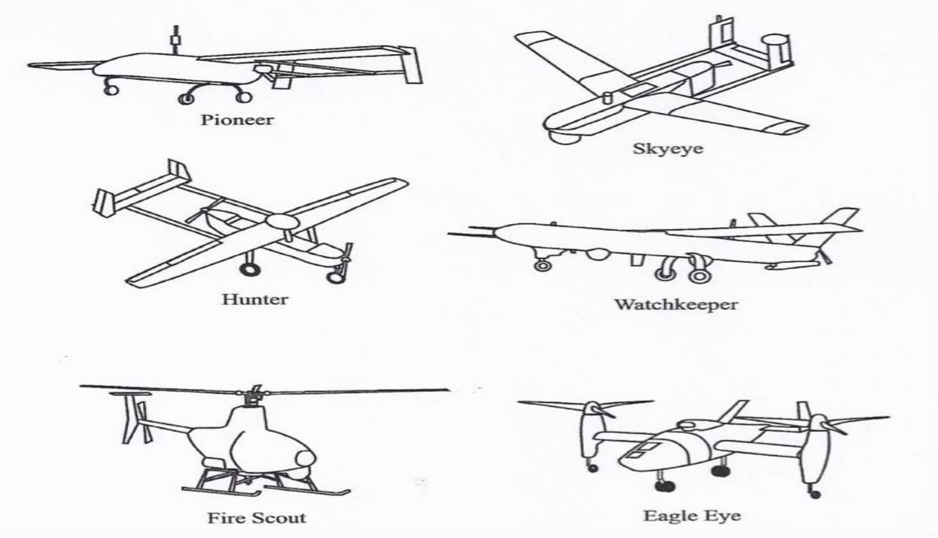

**Figure 1.6:** Medium UAVs

#### 1.2.2.2.4 Large drones

The large UAV class applies to the large UAVs that are somewhat comparable to size of aircraft. They are used mainly for combat operations by the military. Placed that cannot be covered with normal jets are usually captured with these drones. They are main device for surveillance applications. Users can also classify them further into different categories depending upon their range and flying abilities.

Examples of these large UAVs are the US General Atomics Predator A and B.

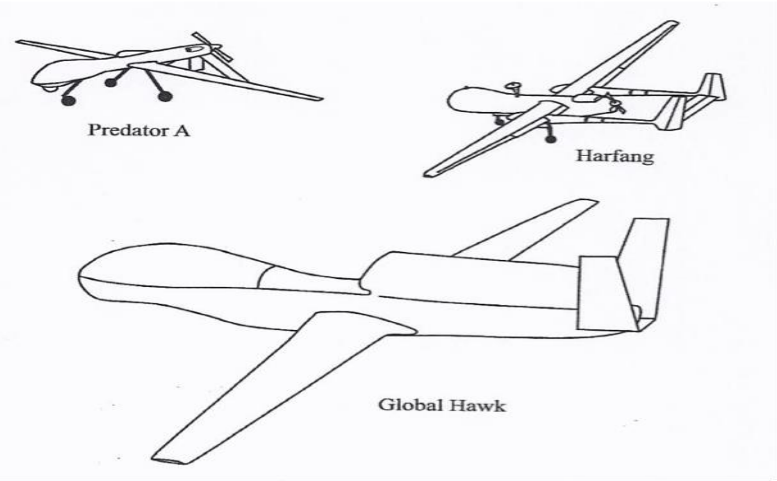

**Figure 1.7:** Large UAVs

## 1.2.3 Applications of UAVs

Mentioning all UAVs applications is a difficult task, as there are so many possibilities in civil and military fields which they both use UAVs for reconnaissance and surveillance. In this subsection, we will discuss the different applications of drones.

The two main classification for UAVs applications are the military application and the civilian application.

### 1.2.3.1 The military application

Military applications focus on weapons delivery and guided missile support as well as directing artillery and spotting enemy positions.

### 1.2.3.2 The civilian application

Today, civilian missions include various application such, some of them are listed below:

- Security awareness;
- Disaster response, including search and support to rescuers;
- Communications and broadcast, including news/sporting event coverage;
- Cargo transport;
- Spectral and thermal analysis;
- Commercial photography, aerial mapping and charting;
- Science and research;
- Critical infrastructure monitoring and inspection.

### 1.2.4 Accessories on-board

A side from the mechanical components that allow a drone to generate lift and maneuver in mid-air, drones also employ an array of sensors that constantly collect information from their surroundings. With this information, drones can maintain their positions, velocity, acceleration and other information. In this paragraph, we are quoting some sensors used in UAVs.

- **Gyroscope:** The most basic of drone sensors, a gyroscope measures the rate of rotation and helps keeping the drone balanced. Gyroscopes are devices that consist of a mounted wheel that spins on an axis that is free to move in any direction. They are used to provide stability or maintain a reference direction.
- **Accelerometer:** The accelerometer of drone works together with its gyroscope. It is used to provide the acceleration force which the drone is subjected to in all three axis X, Y and Z. This data can be used to calculate velocity, direction and even rate of change of altitude of the drone. It is also used to detect the vibration that the drone is experiencing.
- **FPV Camera:** Most of action camera lovers and shooting experts love to buy drones equipped with camera to capture classic shots at tough locations. The film making industry is utilizing them commonly for movie footage. FPV cameras are small, light and reasonably priced. They are mounted on to a drone to send real time video down to the ground using a video transmitter. The FPV camera allows you to see where the drone is flying and what it is seeing as if it had its own eyes.
- **GPS:** The most common use of GPS in UAV is navigation. It is a central component of most navigation system on a UAV. GPS is used to determine the position of the vehicle. The relative positioning and speed of the vehicle are also usually determined by the UAV GPS.

Besides gyroscopes and accelerometers, drones can also use magnetometers, pressure sensor as well as temperature sensors and more.

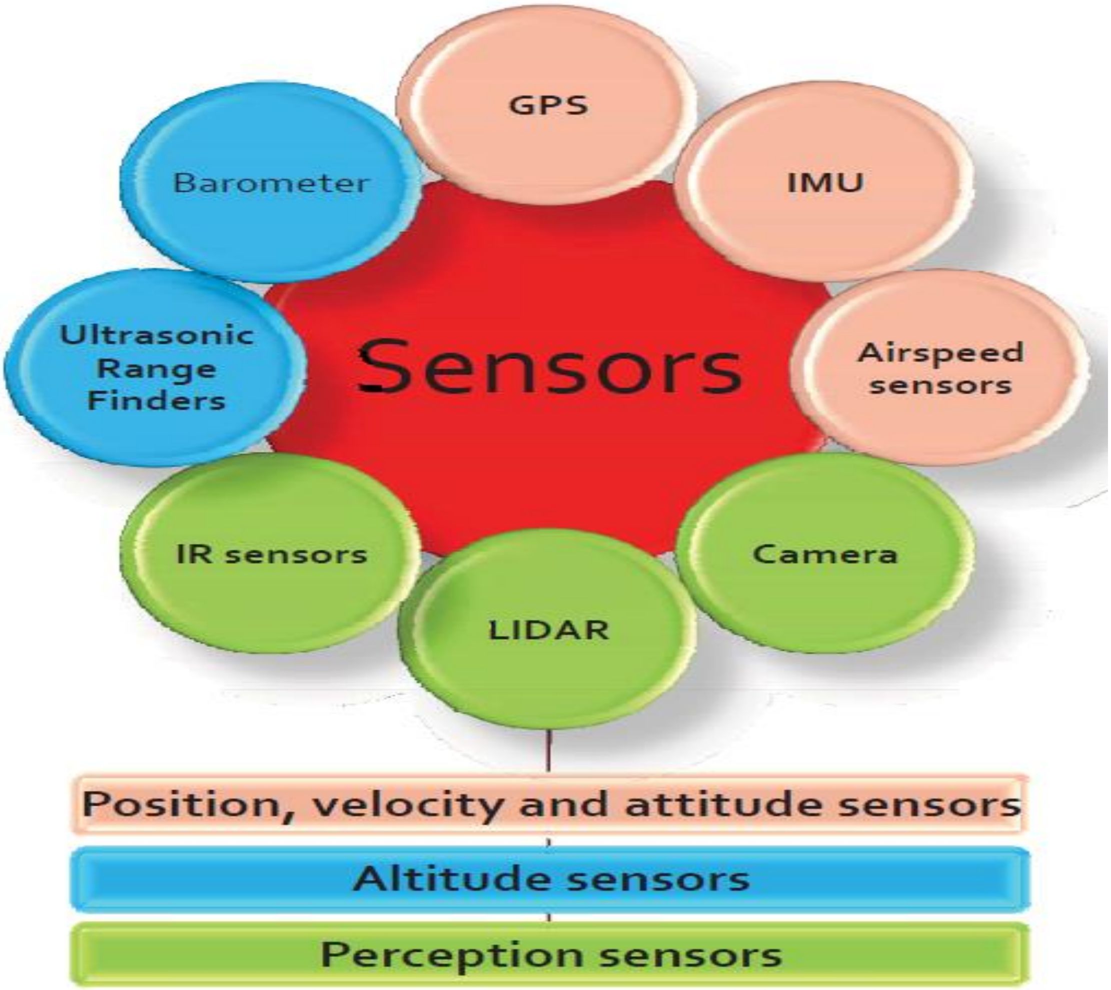

**Figure 1.8:** Various sensor types

## 1.3 PRESENTATION OF ATTITUDE

In this thesis we are interesting especially in attitude controlling for UAVs. For that, this section will introduce a presentation of attitude and mathematical preliminaries which will be used to describe the motion modelling of UAV in chapter 2.

Firstly, it is necessary to introduce the reference coordinates in which we describe the structure and the position. Then, different attitude parameterizations and motion equations are presented.

### 1.3.1 Coordinate Systems

A coordinate system is a reference system used to represent or determine the position of points or other geometric elements on a manifold. The use of a coordinate system allows problems in geometry to be translated into problems about numbers and vice versa.

There are many types of reference systems but only two reference frames will be used in this thesis, these are Earth inertial reference frame and body fixed reference frame.

### 1.3.1.1 Mobile reference system {$\mathcal{B}$}

The mobile reference system, or body fixed reference frame denoted $\mathcal{B}$, is associated to the vehicle, with its origin at the Center of Mass (COM) of the mobile as illustrated in Figure 1.9. The axes of this frame are:

- ($x-$axis): directed along the longitudinal axis oriented from the rear towards the front
- ($y-$axis): directed along the transverse axis oriented from left to right
- ($z-$axis): completes the Direct Cartesian coordinate following the rule of the right hand.

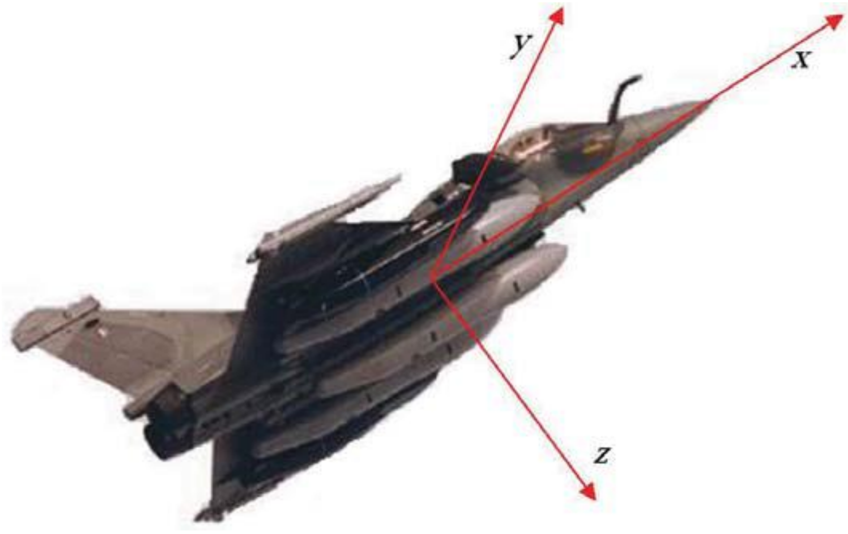

**Figure 1.9:** Mobile frame

### 1.3.1.2 The navigation reference system {n}

The navigation frame is important in navigation because it helps to know the attitude relative to the north, east, and down directions. For position and velocity, it provides a convenient set of resolving axes.

The navigation frame's origin is the point a navigation solution is sought for.

- ($x-axis$): or north (N) axis, is the projection in the plane orthogonal to the z-axis of the line from the user to the North Pole.
- ($y-axis$): By completing the orthogonal set, the y-axis always points east and is known as the east (E) axis.
- ($z-axis$): also known as the down (D) axis is defined as the normal to the surface of the reference ellipsoid, pointing roughly toward the center of the Earth.

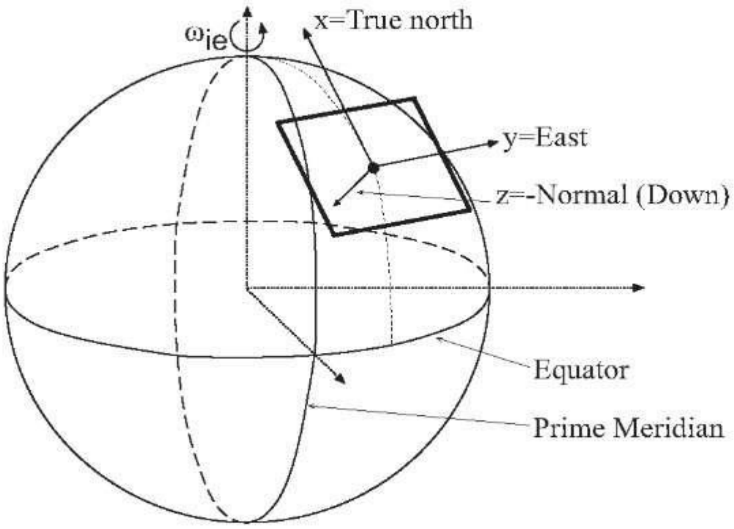

**Figure 1.10:** Navigation frame

### 1.3.1.3 The ECI (Earth Centered Inertial) reference system *{i}*

Newton’s laws are applicable in this system. The ECI does not follow the rotation of the earth and therefore do not rotate relative to the stars. The origin of this system is the center of the Earth. The corresponding coordinate system is a coordinate system with axes marked:

- ($x-axis$): to the "Vernal Equinox" (distant star).
- ($y-axis$): to complete the direct reference system.
- ($z-axis$): always points along the Earth’s axis of rotation from the center to the North Pole.

### 1.3.1.4 The ECEF (Earth Centered Earth-Fixed) reference system *{e}*

The ECEF is similar to the ECI frame, except that all axes remain fixed with respect to the Earth.

- ($x-axis$): points from the center to the intersection of the equator with the IERS reference meridian (IRM), or conventional zero meridian (CZM), which defines 0-degree longitude.
- ($y-axis$): completes the right-handed orthogonal set, pointing from the center to the intersection of the equator with the 90deg east meridian.
- ($z-axis$): always points along the Earth’s axis of rotation from the center to the North Pole.

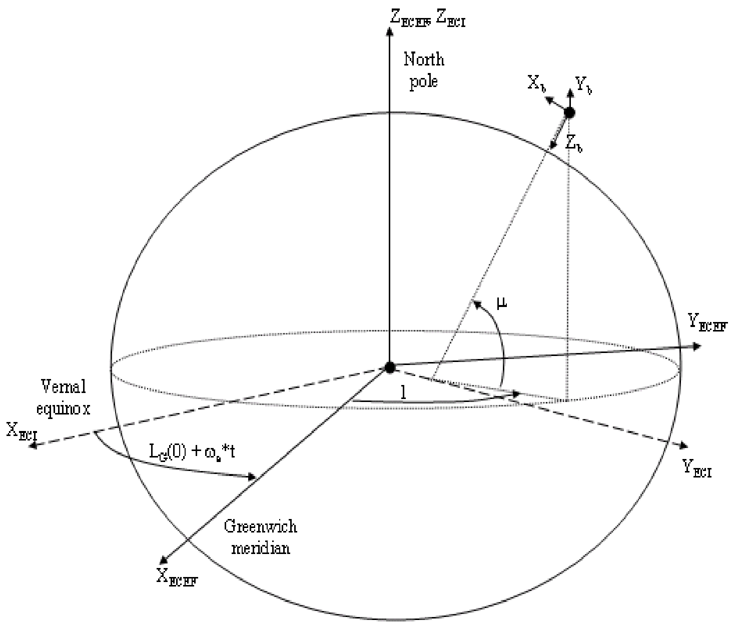

**Figure 1.11:** ECI and ECEF frames

### 1.3.1.5 Geodetic Coordinates system (The WGS-84 standard)

The "WGS 84" is a three-dimensional terrestrial reference system expressing the position in terms of latitude, longitude and altitude. These are based on a reference ellipsoid, which approximates the shape of the Earth.

- The latitude $\phi$: is the angle between the equatorial plane and the normal to the surface of the Earth (ellipsoid) at the point in question. It is zero at the equator and is counted positive for the northern hemisphere, negative for the southern hemisphere.
- The longitude $\lambda$: is the angle between the Greenwich meridian and the desired point. It is counted positively towards the East.
- The height $h$: "ellipsoidal height - not to be confused with altitude", is the difference in meters between that point and the reference ellipsoid measured normal to the ellipsoid. This value is set in a geodetic system and may differ from the altitude of several tens of meters. It should be noted that in general the satellite positioning systems provide ellipsoidal height and not an altitude.

The altitude of a point M of a topographic surface approximates the distance between the point and the reference surface known as the geoid.

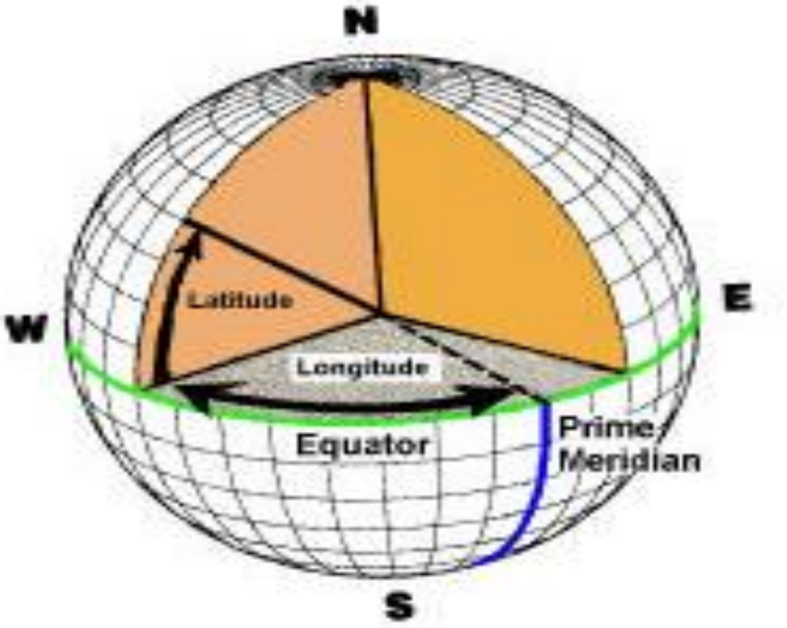

**Figure 1.12:** Latitude and longitude

### 1.3.2 Transformation Matrices

As defined previously, two reference frames will be used in the modelling of quadrotor. To find the components of a vector in both of the frames, transformation matrices have to be formulated at first. By making successive rotations around corresponding axis, one can move from one frame to another.

To relate any vector in the components of body fixed frame and Earth inertial frame, we have to make three sequence of rotations [23]. The order of rotations is important and in this thesis, the common order of roll, pitch, yaw (θ, ϕ, ψ) will be used.

By making three successive transformations, we can find the transformation matrix $R_1$ that transforms $F_I$ to $F_B$ as follows:

$$R_1 = R(\phi)R(\theta)R(\psi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c(\phi) & s(\phi) \\ 0 & -s(\phi) & c(\phi) \end{bmatrix} \begin{bmatrix} c(\theta) & 0 & -s(\theta) \\ 0 & 1 & 0 \\ s(\theta) & 0 & c(\theta) \end{bmatrix} \begin{bmatrix} c(\psi) & s(\psi) & 0 \\ -s(\psi) & c(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1.1)$$

For notational simplicity, trigonometric functions cos() and sin() are shortened as c() and s(). By extending Equation (1.1), final form of $R_1$ is obtained as follows [24]:

$$R_1 = \begin{bmatrix} c(\theta)c(\psi) & c(\theta)s(\psi) & -s(\theta) \\ s(\phi)s(\theta)c(\psi) - c(\phi)s(\psi) & s(\phi)s(\theta)s(\psi) + c(\phi)c(\psi) & s(\phi)c(\theta) \\ c(\phi)s(\theta)c(\psi) + s(\phi)s(\psi) & c(\phi)s(\theta)s(\psi) - s(\phi)c(\psi) & c(\phi)c(\theta) \end{bmatrix} \quad (1.2)$$

The rotation matrix for moving the opposite direction (from body frame to inertia frame) is given by:

$$R = R(-\psi)R(-\theta)R(-\phi)$$

Performing the multiplication, the complete rotation matrix is as follows:

$$R = \begin{bmatrix} c(\psi)c(\theta) & c(\psi)s(\theta)s(\phi) - s(\psi)c(\phi) & c(\psi)s(\theta)c(\phi) + s(\psi)s(\phi) \\ c(\theta)s(\psi) & s(\phi)s(\theta)s(\psi) + c(\phi)c(\psi) & s(\psi)s(\theta)c(\phi) - c(\psi)s(\phi) \\ -s(\theta) & c(\theta)s(\phi) & c(\phi)c(\theta) \end{bmatrix} \tag{1.3}$$

## 1.3.3 Attitude Representation

The attitude is the orientation of the body-fixed reference frame with respect to the trajectory reference frame. This orientation of a rigid body in space is often crucial, especially in aerospace applications [25].

The naturel parameterization of rigid body attitude belongs to the configuration space known as Special Orthogonal group SO (3) and is represented in most general terms as 3×3 rotation matrix.

In this subsection, we provide a description of various attitude parameterizations such as Euler angles, unit quaternions and rotation matrix.

### 1.3.3.1 Rotation Matrix representation

We call a rotation matrix or Direction Cosine Matrix (DCM), denoted $R$ every rotation of the mobile frame relative to the inertial fixed frame.

Rotation matrices form a group under the operation of matrix multiplication called the Special Orthogonal Group $SO(3) \in \mathbb{R}^{3\times 3}$. The abbreviation $SO$ refers to the properties of rotation matrices:

$$SO(3) = \{R \in \mathbb{R}^{3\times 3} | R^T R = RR^T = I_d, \det(R) = 1\} \tag{1.4}$$

Since it uses nine numbers to represent three angular degrees of freedom, there are six independent constraints on the matrix elements. Each column (and row) is unit vector, which gives us three constraints and the columns (and rows) are orthogonal to each other, yielding another three constraints. The translation and rotation together are represented as Special Euclidean group SE (3) [26].

To represent vector components in another frame, we need a rotation matrix. For example, if we have a vector $\vec{x}$ in earth inertial frame and we want to represent it in body-fixed frame, we write:

$$x_b = R^T x \tag{1.5}$$

with R is the matrix obtained in equation (1.2).

The kinematic equation of rigid body rotation described by matrix R is given by [26]:

$$\dot{R} = RS(\omega) \tag{1.6}$$

where S(.) is anti-symmetric matrix defined by:

$$S(x) = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix} \; and\; x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \tag{1.7}$$

### 1.3.3.2 Euler Angle parameterization

In Euler angle representation, the rotation from the inertial frame to the body frame is formed by three successive rotations using the right hand rule.

A rotation around the axis $e_3^i$ with an angle $\psi$ called "yaw angle", which transforms the basis $\{e_1^i, e_2^i, e_3^i\}$ to $\{\, e_1^{i'}, e_2^{i'}, e_3^{i'} \}$ gives the corresponding rotation matrix below:

$$\mathrm{R}_{\boldsymbol{z}} = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

A rotation around the axis $e_2^{i'}$ with an angle $\theta$ called "pitch angle", which transforms the basis $\{\, e_1^{i'}, e_2^{i'}, e_3^{i'} \}$ to $\{\, e_1^{i''}, e_2^{i''}, e_3^{i''} \}$ gives the following corresponding rotation matrix:

$$\mathrm{R}_{\boldsymbol{y}} = \begin{bmatrix} \cos(\theta) & 0 & \sin(\mathrm{\theta}) \\ 0 & 1 & 0 \\ -\sin(\mathrm{\theta}) & 0 & \cos(\mathrm{\theta}) \end{bmatrix}$$

A rotation around the axis $e_1^{i''}$ with an angle $\phi$ called "roll angle", which transforms the basis $\{\, e_1^{i''}, e_2^{i''}, e_3^{i''} \}$ to $\{\, e_1^b, e_2^b, e_3^b\}$ gives the following corresponding rotation matrix:

$$\mathrm{R}_{\boldsymbol{x}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi) & \cos(\phi) \end{bmatrix}$$

The total rotation $R$ ($\psi$, $\theta$, $\phi$), which transforms from the body frame to the inertial frame is the rotation matrix given in equation (1.2).

Euler angles ($\phi$, $\theta$, $\psi$) can be derived directly from the integration of the angular velocity provided by gyroscopes. Either $\omega \in \mathbb{R}^3$ the angular velocity vector expressed in the body frame, the equation linking this velocity vector to Euler angles is given as follows [26]:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \frac{1}{\cos\theta} \begin{bmatrix} \cos\theta & \sin\phi \sin\theta & \cos\phi \sin\theta \\ 0 & \cos\phi \cos\theta & -\sin\phi \cos\theta \\ 0 & \sin\phi & \cos\phi \end{bmatrix} \omega \tag{1.8}$$

From equation (1.8), at Pitch angle of $\frac{\pi}{2}$, it is unable to differentiate between Yaw and Roll degrees of freedom that is mean, Euler angles suffer from singularity.

#### 1.3.3.3 Quaternion parameterization

Generally, the Euler axis-angle attitude representation is not trivial for the mathematical manipulation point of view. For this and to give another global parameterization using only four parameters (against nine in rotation matrix parameterization), Euler extend his theorem of angle-axis representation by introducing a rotation around a unit vector $a$ considered as imaginary complex part with an angle $\theta$ considered as scalar part, which gives:

$$Q = e^{\frac{\theta}{2}(a_x i + a_y j + a_z k)} = \cos(\frac{\theta}{2}) + \sin\left(\frac{\theta}{2}\right)\left(a_x i + a_y j + a_z k\right) \tag{1.9}$$

where $i^2 = j^2 = k^2 = -1$ is a generalization of complex numbers.
We can note:

$$Q = \begin{bmatrix} q_0 \\ q \end{bmatrix} = \begin{bmatrix} \cos(\frac{\theta}{2}) \\ \sin(\frac{\theta}{2})a \end{bmatrix} \tag{1.10}$$

with $q_0 \in \mathbb{R}$ and $q = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} \in \mathbb{R}^3$.

This notation conducts us to the fact that in general $Q \in \mathbb{R}^4$, but since $a$ is a unit vector, therefore:

$$\|Q\| = Q^T Q = q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1 \tag{1.11}$$

which means that the set of unit quaternions define the unit sphere $S^3$ such that:

$$S^3 = \{Q \in \mathbb{R}^4 | Q^T Q = 1\}$$

We can cite some properties of quaternions as follows:

- The multiplication of two quaternions $P = (p_0, p)$ and $Q = (q_0, q)$, denoted by "$\odot$", is a quaternion given by:

$$P \odot Q = \begin{bmatrix} p_0 q_0 - p^T q \\ p_0 q + q_0 p + p \times q \end{bmatrix} \tag{1.12}$$

- The inverse of a quaternion $Q = (q_0, q)$ is also a quaternion defined by $Q^{-1} = (q_0, -q)$.

The rotation matrix of (1.3) can be rewritten as a function of the quaternion as follows:

$$R(q_0, q) = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_1 q_3 - q_0 q_2) \\ 2(q_1 q_1 - q_1 q_1) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 - q_0 q_1) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_2 q_3 - q_0 q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \tag{1.13}$$

which gives us:

$$R(q_0, q) = I_{3\times 3} + 2S(q)^2 + 2q_0 S(q) \tag{1.14}$$

Euler angles can also be deduced from the quaternion and the rotation matrix as follows:

$$\begin{cases} \phi = atan2(R_{32}, R_{33}) = atan2[2(q_2 q_3 + q_0 q_1); 1 - 2(q_1^2 + q_2^2)] \\ \theta = -\operatorname{asin}(R_{31}) = -\operatorname{asin}[2(q_1 q_3 - q_0 q_2)] \\ \psi = atan2(R_{21}, R_{11}) = atan2[2(q_1 q_2 + q_0 q_3); 1 - 2(q_2^2 + q_3^2)] \end{cases} \tag{1.15}$$

where $atan2$ is the inverse tangent function in all four quadrants.

## 1.3.4 Problems of attitude representations

### 1.3.4.1 Problem with rotation matrix

The unique problem of rotation matrix parameterization is that since the translation and rotation constraints are represented together, the 6 constraints are larger compared to all other parameterizations mentioned. They are therefore computationally more expensive than they them. In the other hand, these have the advantage that they have no singularities or ambiguities such as double cover in attitude space in their representation as the rotation matrix is uniquely determined for a given configuration [26].

### 1.3.4.2 Problem with Euler angles

They exhibit a phenomenon known as Gimbal Lock, which results in a singularity in the attitude representation. Note that the Euler angles are relative sequential rotations. For example, in a (123) rotation, a single roll angle will change the orientation of pitch and yaw axis but in case of a single pitch angle rotation, the roll axis remains intact while yaw axis changes its orientation with the pitch rotation. When the pitch is 90°, the roll and yaw axis becomes the same and system can rotate only about 2 axis in space at that particular instant. Therefore, when the axis of two out of the three gimbals are driven parallel to each other in a configuration, there is a loss of degree of freedom and the dimension of the attitude space reduces to 2. The gimbal is still free to move, however, in order to move along the third missing axis in this configuration, the body will have to move simultaneously along two gimbals axis and may exhibit unfamiliar motions when such a situation is encountered physically.

### 1.3.4.3 Problem with Quaternions

- Ambiguities in representing an attitude because there are two quaternions. The constraint of unit modulus of quaternions restricts the quaternions to a sphere of unit radius in 4 dimensions, known as the three-sphere which is the set of unit-vectors in $\mathbb{R}^4$. This three sphere double covers the attitude configuration of the special orthogonal group, SO (3).

- Single physical attitude of a rigid body may yield two different control inputs, which causes inconsistency in the resulting control system, a specific choice between two quaternions generates discontinuity that makes the resulting control system sensitive to noise and disturbances.

  ➢ The following table summarize the properties of the three representations of attitude studied above by representing the globality and uniqueness properties:

**Table 1.2:** Global and unique properties of attitude parametrization

| Attitude parametrization | Globality | Uniqueness |
|---|---|---|
| Rotation matrices | Yes | Yes |
| Quaternions | Yes | No |
| Euler angles | No | No |

## 1.3.5 Attitude kinematics and dynamics

### 1.3.5.1 Attitude kinematics

The study of attitude kinematics is based on the time derivative of a vector in a rotating coordinate system.

Let $e_1^{bi}, e_2^{bi}, e_3^{bi} \in \mathbb{R}^3$ be the principal axis of the body fixed frame expressed in the inertial reference frame, then :

$$R = \begin{bmatrix} e_1^{bi} & e_2^{bi} & e_3^{bi} \end{bmatrix} \quad (1.16)$$

where $e_1^{bi}, e_2^{bi}, e_3^{bi}$ are column vectors forming the columns of $R$.

The derivative of the principal axis expressed in the inertial reference frame is given as:

$$\frac{d}{dt}\left(e_1^{bi}(t)\right) = S\big(\Omega_E(t)\big)e_1^{bi}(t) \quad (1.17)$$

$$\frac{d}{dt}\left(e_2^{bi}(t)\right) = S\big(\Omega_E(t)\big)e_2^{bi}(t) \quad (1.18)$$

$$\frac{d}{dt}\left(e_3^{bi}(t)\right) = S\big(\Omega_E(t)\big)e_3^{bi}(t) \quad (1.19)$$

where $\Omega_E$ is the vector of angular velocity expressed in the inertial reference frame and $S\big(\Omega_E(t)\big)$ is the skew symmetric matrix defined by equation (1.7)

**Attitude kinematics on $SO(3)$**

To get attitude kinematics using rotation matrix as parametrization of attitude, it suffices to calculate the expression of the derivative of R using the elementary definition (1.16) which gives:

$$\dot{R}(t) = \begin{bmatrix} \dot{e}_1^{bi}(t) & \dot{e}_2^{bi}(t) & \dot{e}_3^{bi}(t) \end{bmatrix}$$

$$\dot{R}(t) = S(\Omega_E(t))\begin{bmatrix} e_1^{bi} & e_2^{bi} & e_3^{bi} \end{bmatrix} \tag{1.20}$$

Using the fact that

$$\Omega_E(t) = R(t)\Omega_B(t) \tag{1.21}$$

with $\Omega_B$ is the vector of angular velocity expressed in the body reference frame. In addition to the property of skew symmetric matrix:

$$S(Rx) = RS(x)R^T \tag{1.22}$$

which makes the equation (1.20) as:

$$\dot{R}(t) = R(t)S(\Omega_B(t)) \tag{1.23}$$

### 1.3.5.2 Attitude dynamics

Consider a rigid-body moving in 3D space. The angular momentum can be expressed in the inertial reference frame as:

$$L(t) = R(t)I_r\Omega_B(t) \tag{1.24}$$

where $I_r$ is the moment of inertia or inertia matrix of rigid body expressed in the body reference frame.

Using the Newton's law of motion, one can get:

$$\frac{d}{dt}(L(t)) = R(t)\bar{u}(t) \tag{1.25}$$

With:

$\vec{\bar{u}}$ : is the quadrotor torque vector expressed in the body reference frame.

Using equation (1.24) and (1.25), the simplified attitude dynamics can be expressed as:

$$I_r\dot{\Omega}_B(t) = -S(\Omega_B(t))I_r\Omega_B(t) + \bar{u}(t) \tag{1.26}$$

## 1.4 CONCLUSION

In this chapter, we have concluded the preliminary notions about UAVs technology and the attitude parameterizations as an introduction in order to achieve the aim of developing a quadrotor model, which is necessary in this thesis. Quadrotor is a small rotary-wing UAV with vertical take-off and landing (VTOL) ability. The main difference of quadrotors compared to other conventional VTOL UAVs such as helicopters is using fixed-pitch propellers to control the quadrotor instead of variable-pitch propellers, which are generally used in helicopters. As the name implies, fixed pitch-propeller systems could not change the pitch angle of the propeller, instead, control problem can be handled by changing the angular velocities of each propeller properly. According to that, we conclude that, the basic difference and advantage of quadrotors compared to other VTOL UAVs with variable-pitch propellers is the removal of complex mechanical transmission mechanisms that complicates both structural and aerodynamic design. This advantage made them one of the most popular platforms for UAV researches.

Due to its versatility, availability, and mechanical characteristics, a quadrotor was chosen in this thesis for developing control strategies.

Detailed modeling and control techniques for quadrotors will be described in the next chapter.

# Chapter 2

# QUADROTOR DESCRIPTION

## 2.1 INTRODUCTION

At the beginning of the 21st century, the evolution of aerial robotics has allowed a wide range of applications fields. One of the most known aerial robot is the Quadrotor or Quadcopter. A quadrotor is a rotorcraft capable of hover, forward flight, and VTOL and is emerging as a fundamental research and application platform at present with flexibility, adaptability, and ease of construction. They are equipped with a set of sensors allowing the measurements of the position, linear and angular velocities and other physical quantities used to estimate crucial information such as attitude. These data are required to control the motion of the quadrotors.

Since a quadrotor is basically considered an unstable system with the characteristics of dynamics such as being intensively nonlinear, multivariable, strongly coupled, and under-actuated, a precise and practical model is critical to control the vehicle which seems to be simple to operate. As a rotorcraft, the complicated aerodynamic effects of the rotors mainly dominate the dynamics of a quadrotor. Many controllers have been presented to overcome this aerodynamic complexity of the control [27].

This chapter provides a presentation of quadrotor UAVs firstly. Secondly, it gives a tutorial of the platform configuration, comprehensive nonlinear model, and dynamic model identification for a quadrotor which will be simplified to use in the derivation of control laws that are explained in Chapter 3. Finally, various control methods that have been proposed for quadrotors controlling were presented.

## 2.2 QUADROTOR PRESENTATION

### 2.2.1 Quadrotor definition

A quadrotor is a rotary wing UAV consisting of four rotors located at the ends of cross structure. Its four rotor based propulsion system provides higher payload capacity and maneuverability. By varying the speeds of each rotor, the flight of the quadrotor is controlled. This type of UAV possess certain essential characteristics, which highlight their potential for use in search and rescue applications. Characteristics that provide a clear advantage over other flying UAVs include their Vertical Takeoff and Landing (VTOL) and hovering capability, as well as their ability to make slow precise movements.

Rotors driven with motors give the dominating forces and moments acting on the quadrotor. According to the orientation of the blades, relative to the body coordinate system, there are two basic types of quadrotor configurations: Plus and Cross configurations.

In the Plus configuration, a pair of blades, spinning in the same clockwise or counter-clockwise direction, are fabricated on $x$ and $y$ coordinates of the body frame coordinate system. On the contrary, a different Cross configuration is adopted by some other quadrotors, in which there is no rotor at the front or the rear but instead two rotors are on the right side and two others on the left.

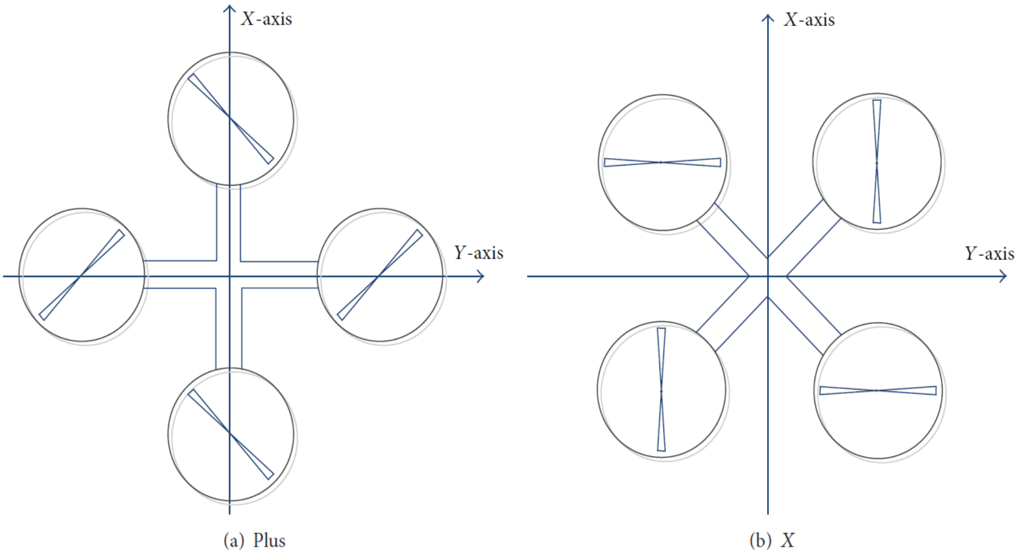

**Figure 2.1:** Plus and Cross configurations

In contrast with the Plus configuration, for the same desired motion, the Cross-style provides higher momentum that can increase the maneuverability performance as each move requires all four blades to vary their rotation speed.

## 2.2.2 Basic concepts & movements of the quadrotor

The quadrotor has 6-DOF and four actuators. Each motor is connected to a propeller and all the propellers' axes of rotation are fixed and parallel to each other. In addition, all the propellers have fixed-pitch blades and their airflow goes downwards to get an upward lift.

To balance the quadrotor and remove the need for a tail rotor, an opposite pair's directions have been configured. The left and the right propellers rotate clockwise, while the front and the rear ones rotate counter-clockwise with the same speed.

The space motion of the rigid body quadrotor can be divided into two parts:

- The barycenter movement: three barycenter movements that correspond with the three translations. These movements define the position of the quadrotor.

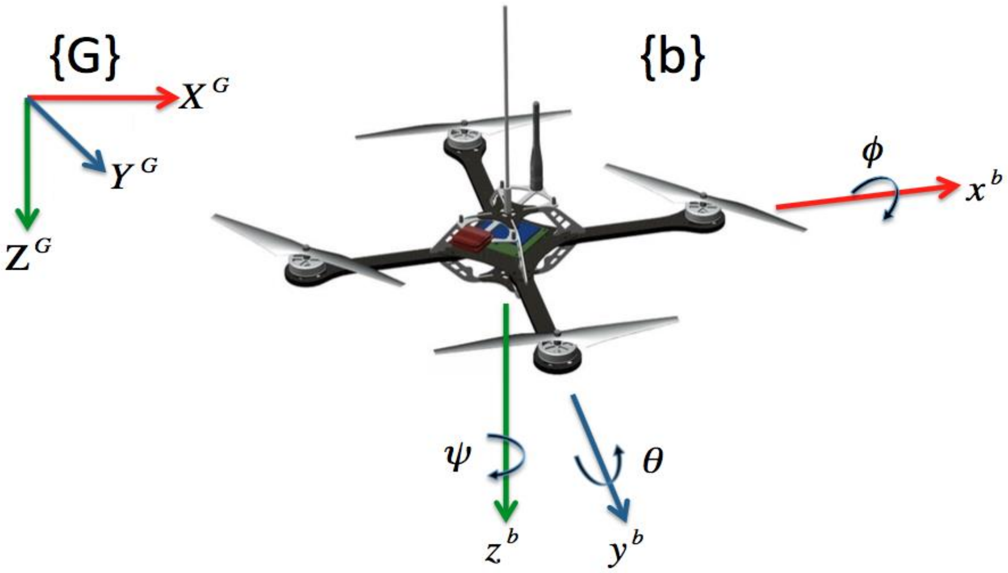

**Figure 2.2:** Structure model of quadrotor with inertial frame and body frame

- Movement around the barycenter: three angular motions that correspond with the three rotation motions along the axes. This movement defines the attitude of the quadrotor.

This leads to six different degrees of freedom, whose control can be implemented by adjusting the rotational speed of the different motors. Those variables could be controlled using different movements, which allow the quadrotor to reach a desired altitude and attitude. Those movements are defined as throttle, roll, pitch and yaw. Follows is a description of each movement.

### 2.2.2.1 Throttle movement of quadrotor

Throttle movement is achieved by increasing (or decreasing) the speeds of all propellers by the same value. This generates a vertical force along z-axis in B-frame, which leads to raising or lowering the quadrotor.

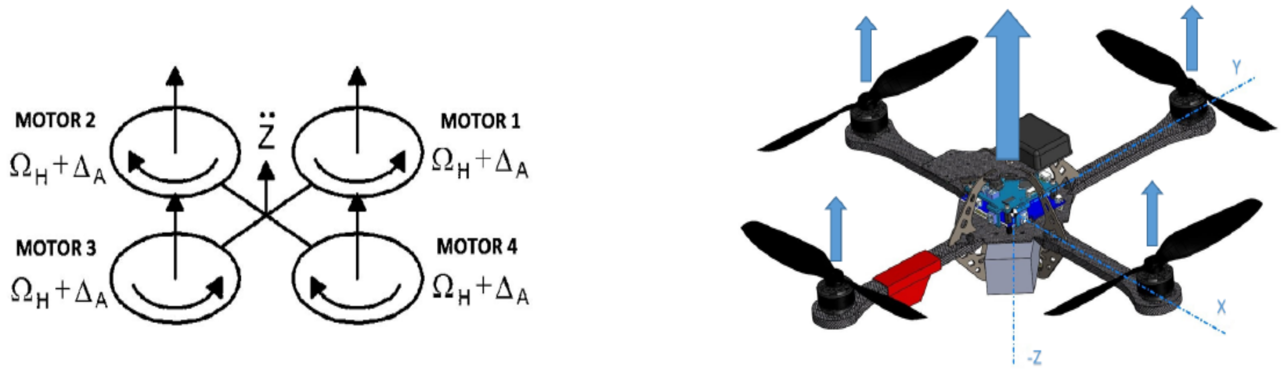

**Figure 2.3:** Throttle movement of quadrotor

### 2.2.2.2 Roll movement of quadrotor

Roll movement is achieved by increasing (or decreasing) the left propeller’s speed and by decreasing (or increasing) the right one. This difference in speed generates a torque in x-axis in B-frame, which makes the quadrotor turn and leads to roll angle acceleration.

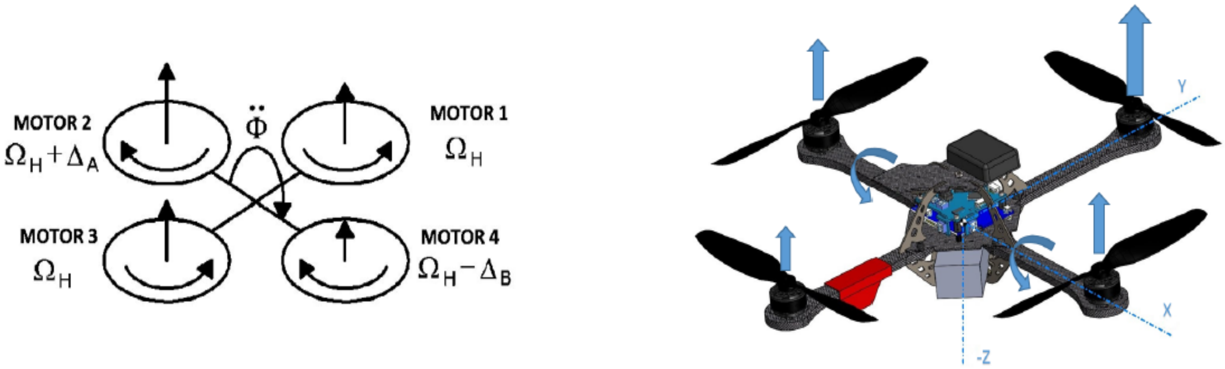

**Figure 2.4:** Roll movement of quadrotor

### 2.2.2.3 Pitch movement of quadrotor

This command is similar to the roll one and is provided by increasing (or decreasing) the rear propeller speed and by decreasing (or increasing) the front one. This lead to a torque with respect to the Y-axis in body frame that makes the quadrotor turn.

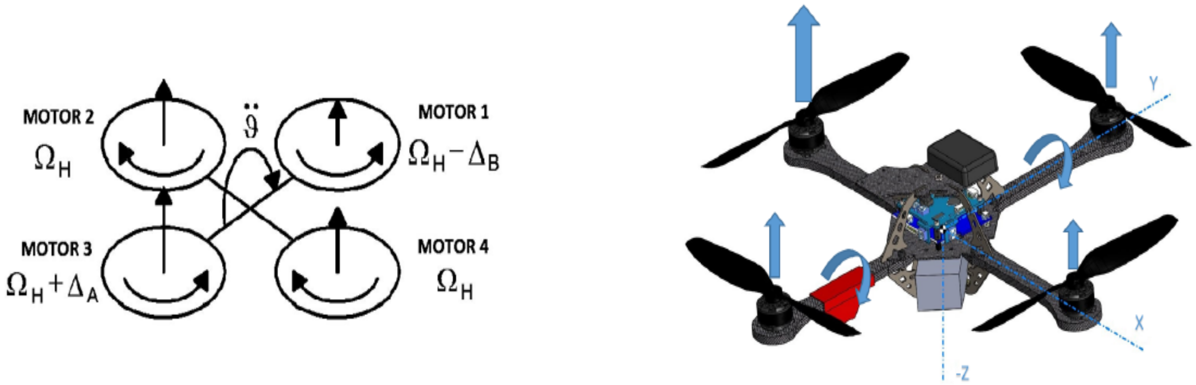

**Figure 2.5:** Pitch movement of quadrotor

### 2.2.2.4 Yaw movement of quadrotor

This command is provided by increasing (or decreasing) the front-rear propellers speed and by decreasing (or increasing) that of the left-right couple. It leads to a torque with respect the Z-axis in body frame, which makes the quadrotor turn. The yaw movement is generated thanks to the fact that the left-right propellers rotate clockwise while the front-rear ones rotate counterclockwise.

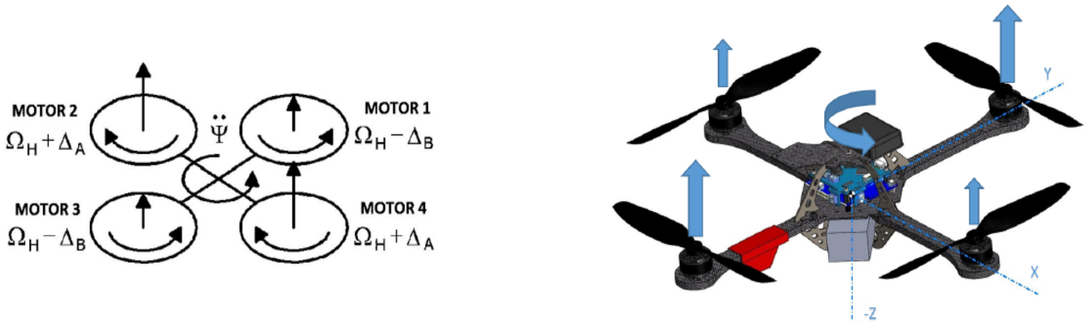

**Figure 2.6:** Yaw movement of quadrotor

## 2.3 QUADROTOR MATHEMATICAL MODEL

The first step towards designing a controller for quadrotor is to state a mathematical model to work with. This section presents mathematical models for the quadrotor UAV system where 6-DOF rigid body kinematics and dynamics are applied to it using Newton-Euler equations.

The equations of motion are derived by using of two coordinate frames which were defined previously, the body-fixed and earth-fixed frame, and they are more conveniently formulated in the body-fixed frame. To make the body equations simpler, two assumptions have been made:

- The origin of the body-fixed frame is coincident with the center of mass (COM) of the body.
- The axes of the B-frame coincide with the body principal axes of inertia to have a diagonal Inertia matrix.

The position of the center of gravity of the quadrotor in inertial frame is represented by:

$$r = [x\ y\ z]^T \tag{2.1}$$

where the attitude is represented by:

$$\eta = [\phi\ \theta\ \psi]^T \tag{2.2}$$

In this work, we interest in developing a quadrotor controller that avoid problems related to attitude tracking such as parametric uncertainties, external disturbances… For that, we take in consideration the presence of external disturbances in the modelling of quadrotor dynamic model.

### 2.3.1 Kinematics

In order to modelling a quadrotor UAV, we need to convert translational and rotational velocities from the vehicle body frame to the inertial reference frame. Kinematics of translation and rotation can be applied to achieve this conversion by using Euler angles, which are used to convert from the inertial earth fixed reference frame to the body fixed reference frame using the rotation sequence $\psi$-$\theta$-$\phi$.

### 2.3.1.1 Translational velocities

To convert the translational velocities, we will use the rotation matrix that was obtained in the previous chapter to convert from E-frame to B-frame:

$$R_1 = \begin{bmatrix} c(\theta)c(\psi) & c(\theta)s(\psi) & -s(\theta) \\ s(\phi)s(\theta)c(\psi) - c(\phi)s(\psi) & s(\phi)s(\theta)s(\psi) + c(\phi)c(\psi) & s(\phi)c(\theta) \\ c(\phi)s(\theta)c(\psi) + s(\phi)s(\psi) & c(\phi)s(\theta)s(\psi) - s(\phi)c(\psi) & c(\phi)c(\theta) \end{bmatrix} \tag{2.3}$$

Let us $V_B$ and $V_E$ be the components of the translational velocity of quadrotor in frames B-frame and E-frame, respectively. Then by using transformation matrix obtained in Equation (2.3), one can obtain $V_B$ as follows:

$$V_B = R_1 V_E \tag{2.4}$$

It is important to note that transformation matrix $R_1$ is nonsingular. Therefore, inverse of $R_1$ exists. By multiplying both sides of Equation (2.4) with $R_1^{-1}$, one can also obtain:

$$V_E = R_1^{-1} V_B \tag{2.5}$$

where $R_1^{-1} = R$ obtained in equation (1.3) from the previous chapter. So $V_E$ can be converted to velocities in the inertial earth reference frame as follows:

$$V_E = R V_B \tag{2.6}$$

By differentiating Equation (2.1) with respect to time and using equation (2.6), one obtains:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} c(\psi)c(\theta) & c(\psi)s(\theta)s(\phi) - s(\psi)c(\phi) & c(\psi)s(\theta)c(\phi) + s(\psi)s(\phi) \\ c(\theta)s(\psi) & s(\phi)s(\theta)s(\psi) + c(\phi)c(\psi) & s(\psi)s(\theta)c(\phi) - c(\psi)s(\phi) \\ -s(\theta) & c(\theta)s(\phi) & c(\phi)c(\theta) \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{2.7}$$

where

$V_B = [u \; v \; w]^T \in \mathbb{R}^3$are the linear velocities expressed in the body reference frame.

$V_E = [\dot{x} \; \dot{y} \; \dot{z}]^T \in \mathbb{R}^3$are the linear velocities expressed in the inertial earth reference frame.

### 2.3.1.2 Rotational velocities

Rotational velocities in the body reference frame can be converted to the inertial reference frame using a similar approach as for the translational velocities with the consideration of the rotation about the appropriate vector in each intermediate frame [31]:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\phi \tan\theta & \cos\phi \tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi \sec\theta & \cos\phi \sec\theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{2.8}$$

where

$\vec{\Omega}_B = [p\ q\ r]^T \in \mathbb{R}^3$ are the angular velocities expressed in the body reference frame.

$\vec{\Omega}_E = [\dot{\phi}\ \dot{\theta}\ \dot{\psi}]^T \in \mathbb{R}^3$ are the angular velocities expressed in the inertial earth reference frame.

### 2.3.2 Dynamics

Conservation of linear and angular momentum are applied to derive the rigid body dynamics of the quadrotor UAV system. A 6 DOF rigid body under external forces and moments applied to the center of gravity in the presence of external disturbances written in the body reference frame can be expressed in Newton-Euler form as:

$$m\overrightarrow{\dot{V}_E} = \vec{F} + m\overrightarrow{d_1} \tag{2.9}$$

$$I_r\overrightarrow{\dot{\Omega}_B} + \vec{\Omega}_B \times I_r\vec{\Omega}_B = \vec{\vec{u}} + \vec{d} \tag{2.10}$$

where

$\vec{\Omega}_B$: is the angular velocities expressed in the body reference frame,

$\overrightarrow{\dot{V}_E}$: is the linear acceleration expressed in the inertial earth reference frame,

$m$: is the mass of the quadrotor,

$\vec{F}$: is the quadrotor forces vector expressed in the inertial earth reference frame,

$\vec{\vec{u}}$ : is the quadrotor torque vector expressed in the body reference frame,

$\overrightarrow{d_1}$ : is the external disturbance where:

$$\vec{d}_1 = \begin{bmatrix} 0 \\ 0 \\ d_1 \end{bmatrix} \tag{2.11}$$

$\vec{d} = [d_2\ d_3\ d_4]^T$ is the external disturbance with unknown bound generated by an exogenous system,

$I_r \in \mathbb{R}^{3\times 3}$ : is the inertia matrix of the quadrotor where symmetry is assumed in all axes resulting in all off-diagonal values to be zero:

$$I_r = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \tag{2.12}$$

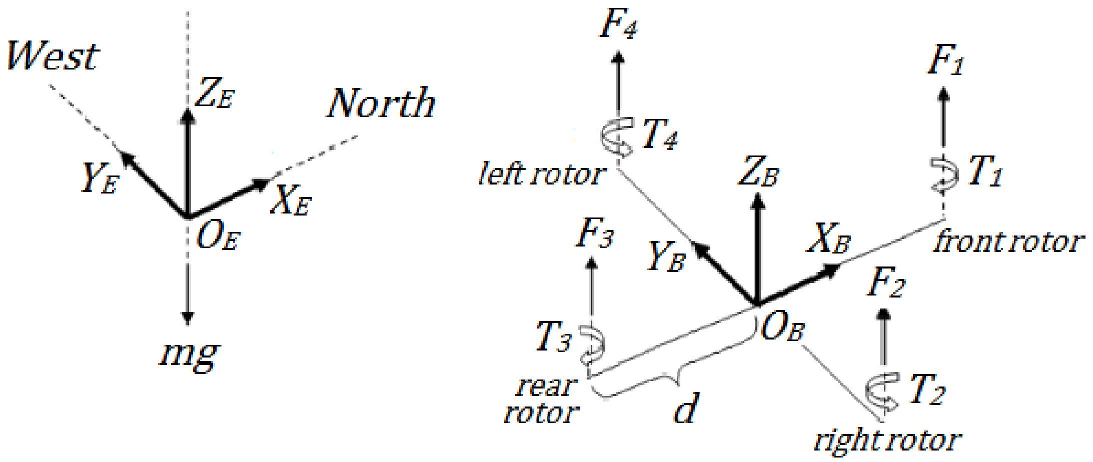

**Figure 2.7:** Rotor configuration, force and torque generated by each propeller of the quadrotor, reference frames FB and FE

### 2.3.2.1 Forces

In quadrotors, there are two main forces acting on its body:

- **Thrust force** which is the summation of all forces generated by the four motors. All these forces are proportional to the propellers' speed square. Moreover, they are acting only along z-axis of the B-frame and could be calculated using the following equation:

$$f_i = b\omega_i^2 \tag{2.13}$$

$$u_1 = f_1 + f_2 + f_3 + f_4 = b(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \tag{2.14}$$

where

$b$: is the thrust factor

$f_i$: is the thrust produced by rotors with $i = 1, 2, 3, 4$

$\omega_i$: is the angular speed of the $i_{th}$ rotor

$u_1$: is the total thrust in B-frame.

- **Gravitational Force** which is generated by the gravitational acceleration acting along the z-axis expressed in inertial earth reference frame. It is giving by:

$$P = mg \tag{2.15}$$

Combining all forces acting on the quadrotor body expressed in the inertial earth reference frame yields:

$$\vec{F} = Ru_1 + m\vec{g} \tag{2.16}$$

$$F = \begin{bmatrix} s(\psi)s(\phi) + c(\psi)s(\theta)c(\phi) \\ -c(\psi)s(\phi) + s(\psi)s(\theta)c(\phi) \\ c(\theta)c(\phi) \end{bmatrix} u_1 - \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \tag{2.17}$$

### 2.3.2.2 Torques

Two main sources generate torques in a quadrotor system:

- Torques generated directly by the main movement inputs. They are proportional to the speed of the propellers and acting on the body frame.

$$\begin{bmatrix} u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} bL(\omega_2^2 - \omega_4^2) \\ bL(\omega_3^2 - \omega_1^2) \\ -Q_1 + Q_2 - Q_3 + Q_4 \end{bmatrix}$$

(2.18)

where

$Q_i = d\omega_i^2$ is the drag moment of the $i^{th}$ rotors and $d$ is the drag factor and $L$ is the distance between the center of a propeller and the center of mass of quadrotor.

- **Gyroscopic Torque**: Since two of the propellers are rotating clockwise and the other two rotating counterclockwise, an overall imbalance will happen when the algebraic sum of the rotor speeds is not equal to zero. This imbalance will cause a gyroscopic effect. Furthermore, this effect is proportional to the roll and pitch rates. The following equation defines the overall propeller's speed:

$$\Omega_r = -\omega_1 + \omega_2 - \omega_3 + \omega_4 \tag{2.19}$$

Combining all torques acting on quadrotor body expressed in the body reference frame yields:

$$\bar{u} = -J_r\Omega_r \begin{bmatrix} q \\ -p \\ 0 \end{bmatrix} + \begin{bmatrix} u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} -J_r\Omega_r q + u_2 \\ J_r\Omega_r p + u_3 \\ u_4 \end{bmatrix} \tag{2.20}$$

with $J_r$ the total rotational moment of inertia of propeller.

In order to simplify the calculation of angular velocity of rotors, equation (2.14) with equation (2.18) become:

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} b & b & b & b \\ 0 & -bL & 0 & bL \\ -bL & 0 & bL & 0 \\ -d & d & -d & d \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} \tag{2.21}$$

### 2.3.2.3 The dynamic equations

The dynamical equations of quadrotor UAV derived from equations (2.9) and (2.10) can be written in following forms below.

**For Translational Motion**, and by substituting equation (2.11) and equation (2.17) into equation (2.9), the nonlinear dynamic model is given as follows:

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = -\begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} + R\begin{bmatrix} 0 \\ 0 \\ {}^{u_1}\!/_{m} \end{bmatrix} + d_1\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{2.22}$$

**For Rotational Motion**, and by substituting equations (2.12) and (2.20) into equation (2.10), the following equation of the nonlinear dynamical model can be obtained as:

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} (I_{yy} - I_{zz})qr/I_{xx} \\ (I_{zz} - I_{xx})pr/I_{yy} \\ (I_{xx} - I_{yy})pq/I_{zz} \end{bmatrix} + \bar{u}\begin{bmatrix} 1/I_{xx} \\ 1/I_{yy} \\ 1/I_{zz} \end{bmatrix} + \begin{bmatrix} d_2 \\ d_3 \\ d_4 \end{bmatrix} \tag{2.23}$$

The relation between body angular velocities and rate of change of Euler angles is given by:

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} , \qquad \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} \tag{2.24}$$

Substituting the relations defined in equation (2.24) into equation (2.23), the simplified nonlinear dynamic model for rotational motion is obtained as follows:

$$\begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} (I_{yy} - I_{zz})\dot{\theta}\dot{\psi}/I_{xx} \\ (I_{zz} - I_{xx})\dot{\psi}\dot{\phi}/I_{yy} \\ (I_{xx} - I_{yy})\dot{\phi}\dot{\theta}/I_{zz} \end{bmatrix} + \bar{u}\begin{bmatrix} 1/I_{xx} \\ 1/I_{yy} \\ 1/I_{zz} \end{bmatrix} + \begin{bmatrix} d_2 \\ d_3 \\ d_4 \end{bmatrix} \tag{2.25}$$

The final simplified form of the dynamic model of quadrotor is given as:

$$\begin{cases} \ddot{x} = (\cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi)\frac{u_1}{m} \\ \ddot{y} = (\cos\phi \sin\theta \cos\psi - \sin\phi \sin\psi)\frac{u_1}{m} \\ \ddot{z} = -g + \cos\phi \cos\theta \frac{u_1}{m} + d_1 \\ \quad \ddot{\phi} = \dot{\theta}\dot{\psi}\frac{I_{yy}-I_{zz}}{I_{xx}} - \frac{J_r}{I_{xx}}\dot{\theta}\Omega_r + \frac{u_2}{I_{xx}} + d_2 \\ \quad \ddot{\theta} = \dot{\psi}\dot{\phi}\frac{I_{zz}-I_{xx}}{I_{yy}} + \frac{J_r}{I_{yy}}\dot{\phi}\Omega_r + \frac{u_3}{I_{yy}} + d_3 \\ \quad \ddot{\psi} = \dot{\phi}\dot{\theta}\frac{I_{xx}-I_{yy}}{I_{zz}} + \frac{u_4}{I_{zz}} + d_4 \end{cases} \tag{2.26}$$

where

$\ddot{x}, \ddot{y}, \ddot{z}$ : are the linear accelerations around X, Y and Z-axis respectively.

$\ddot{\phi}, \ddot{\theta}, \ddot{\psi}$ : are the angular accelerations around X, Y and Z-axis respectively.

This dynamic model will be used in the next chapter in order to design controllers for quadrotors attitude tracking using two different methods.

## 2.4 CONTROL TECHNIQUES

The state of the art in quadrotor control has suffered a drastic change in the last few years. The number of projects tackling this problem, in different research fields, has considerably increased. Numerous approaches can be taken when designing stable control laws for quadrotor UAVs. In general, these approaches can be divided into linear control, nonlinear control and intelligent control as shown in figure 2.8. In this section, an overview of the most important developed techniques applied to quadrotors UAVs will be presented.

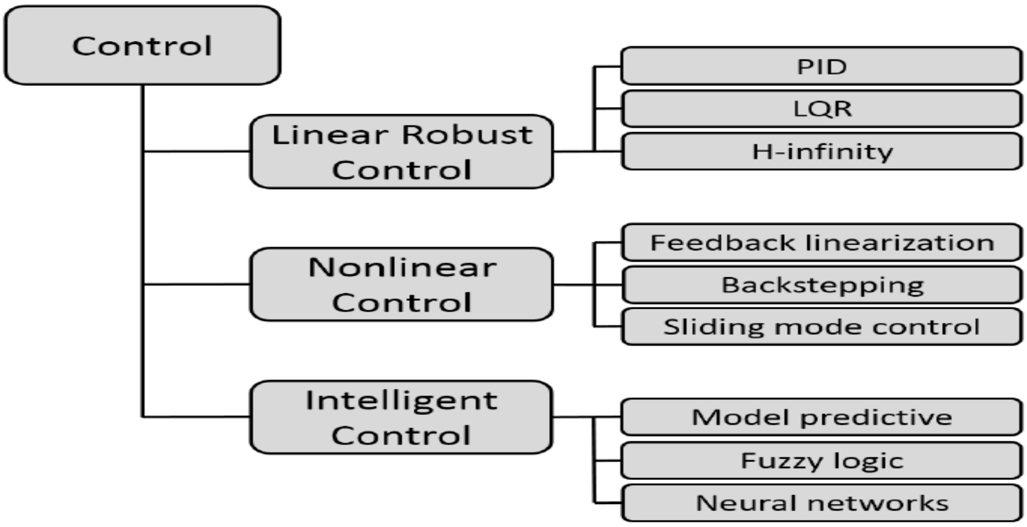

**Figure 2.8:** Categorization of controllers

### 2.4.4 Linear Robust Controllers

Early in quadrotor development, it was found that linear controllers were sufficient to obtain stable flight. However, with the evolution of controlling it seemed that the application of linear controllers to nonlinear systems such as UAVs would not result in a robust response due to their inherent design through linearization. We examine several of these control techniques to include, a Proportional Integral Derivative (PID) controller, Linear Quadratic Controllers, and $H_\infty$ controller.

### 2.4.1.1 Proportional Integral Derivative Controller

The PID controller (or three-term controller) is one of the most popular controllers due to its simplicity. PID controllers are considered as a classical approach in control theory.

A PID controller is a control loop mechanism employing feedback that is widely used in industrial control systems and a variety of other applications requiring continuously modulated control. A PID controller continuously calculates an error value e(t) as the difference between a desired set point (SP=r(t)) and a measured process variable (PV=y(t)) and applies a correction based on proportional, integral, and derivative terms (denoted P, I, and D respectively).

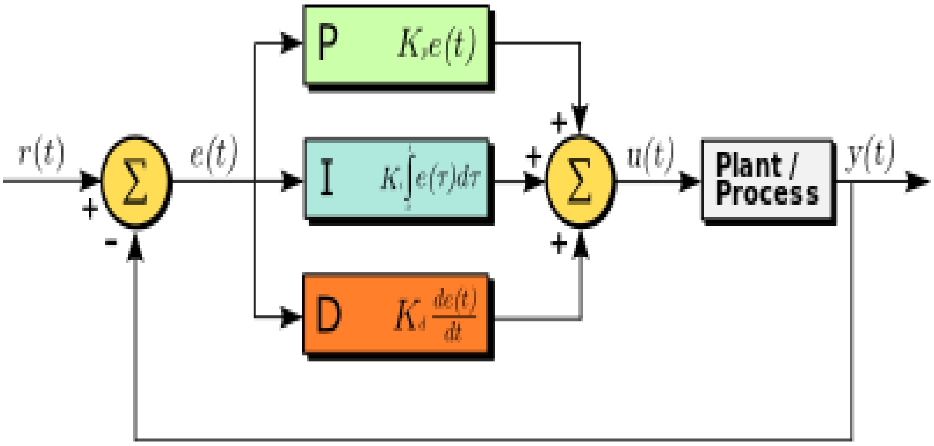

**Figure 2.9:** Block diagram of PID controller

The block diagram of figure 2.9 shows the principles of how these terms are generated and applied. The controller attempts to minimize the error over time by adjustment of a control variable u(t), such as the opening of a control valve, to a new value determined by a weighted sum of the control terms .

The overall control function is:

$$\mathrm{u(t)} = K_p\ \mathrm{e(t)} + K_i \int_0^t \mathrm{e(t')}\ \mathrm{dt'} + K_d\ \frac{\mathrm{de(t)}}{\mathrm{dt}} \qquad (2.27)$$

where $K_p$ , $K_i$ and $K_d$ , all non-negative, denote the coefficients for the proportional integral and derivative terms respectively ( denoted P, I, and D).

### 2.4.1.2 Linear Quadratic Controller

There are two types of Linear Quadratic Controllers, namely: Linear Quadratic Regulator (LQR) controllers and Linear Quadratic Gaussian (LQG) Controllers for quadrotor control. In the former, the system is optimized based on a cost function and minimum cost by weighting factors supplied by the user. In the latter, the LQG controller is a combination of a Kalman type filter with a linear-quadratic regulator LQR.

### 2.4.1.3 $H_\infty$ Controller

For a system with external disturbances and model uncertainties, the $H_\infty$ controller can be a good choice for linear control of quadrotors. The quadrotor system is routinely affected by wind gusts and model uncertainties. Thus, some researchers have applied the $H_\infty$ controller into the quadrotor system to make the system more robust to external disturbances [28]. $H_\infty$ control is concerned with assuring overall stability for the closed-loop system by applying optimization in the frequency domain. This is achieved by finding a feedback gain that will minimize the maximum response for the closed-loop system in the frequency domain together with assuring closed-loop stability [32]. Using the H1 technique, the formulation derives a controller using the Riccati equations there by solving an optimization problem to control the quadrotor. To do this the Linear Matrix Inequality (LMI), approach has typically been applied for solutions of the Riccati equation. On the other hand, the nonlinear H1 controller is generally obtained by Hamilton-Jacobi equations that can replace the linear Riccati assumptions.

## 2.4.5 Nonlinear Controllers

This subsection will emphasize nonlinear control design, which is focused on altering the stability of nonlinear systems specifically. Since the quadrotor system has four inputs and six degrees of freedom, it considered as a nonlinear under-actuated system. Therefore, to get better performance, a nonlinear controller is warranted and since the control of quadrotors has been looked at for a number of years now, there are a large body of papers developing nonlinear control theory for quadrotors. These approaches include feedback linearization, backstepping control techniques, and sliding mode control that will be presented next.

### 2.4.2.1 Feedback Linearization

One of the more common approaches in nonlinear control is feedback linearization. Using this method, the nonlinear system is transformed into an equivalent linear system. Then using the linear systems, similarity transformation was used to produce a nonsingular matrix. This is a form of diffeomorphism and can be used to transform the state variables of the nonlinear system into a linear system. Then a standard linear control theory can be applied to the system and subsequently the solution from the linearized system is converted back into the nonlinear system [28].

A generic nonlinear system can be characterized as [33]:

$$\dot{\vec{x}} = \vec{f}(x) + g(x)\vec{u} \tag{2.28}$$

where

$\vec{x} \in \mathbb{R}^n$ is the state vector, $\vec{u} \in \mathbb{R}^m$ are the inputs to the system, $\vec{f}(x) \in \mathbb{R}^n$ represents the nonlinear system dynamics and $g(x) \in \mathbb{R}^{n\times m}$ is the nonlinear input matrix.

Given that $(x)$ is invertible, feedback linearization can be applied to this nonlinear system by inverting Equation (2.28) as:

$$\vec{u}(x) = g^{-1}(x)\left[\vec{U}(x) - \vec{f}(x)\right] \tag{2.29}$$

$\vec{U}(x) \in \mathbb{R}^n$ is a virtual controller which can be designed using classic control techniques to guarantee desirable stability dynamics. Most often, the virtual control signal is generated using a simple linear controller that assures the desirable dynamics.

### 2.4.2.2 Backstepping

Backstepping control is a popular and effective approach to stabilize nonlinear systems. In a similar fashion as feedback linearization, backstepping control applies (multiple) feedback loops. The technique is constructed from subsystems that can be stabilized using other methods. The process starts with a known-stable system and “back out” new controllers that progressively stabilize each of the subsystems. The process completes when the final control is achieved. Instead of minimizing a cost function, like is performed in H∞ loop shaping and LQR control, a virtual control input is generated using a Lyapunov approach such that the closed-loop system exhibits desired dynamic characteristics [29]. When applied in a cascade architecture, emphasis must be placed on selecting an appropriate Lyapunov equation for each loop to ensure the output of the outer loop, which acts as a virtual control input to the inner loop, assures favorable transient and steady-state characteristics of the closed-loop system.

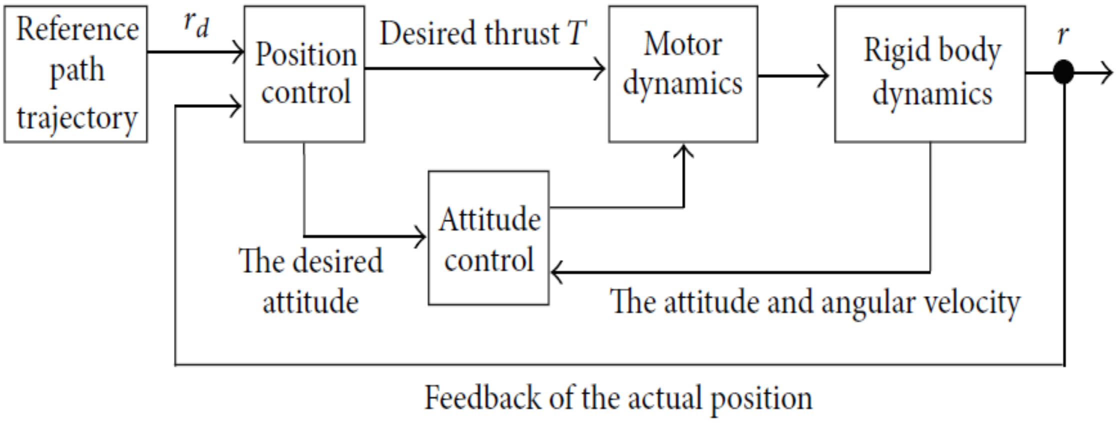

**Figure 2.10:** Architecture of cascade backstepping control applied to quadrotor UAV

### 2.4.2.3 Sliding Mode Controller

Sliding Mode Controller (SMC) is a nonlinear control method that modifies the system using a discontinuous control signal thereby forcing the system to move within the system's normal behavior [30]. The control law is not continuous in time and it switches from one state to another based on position in state space. While there are model uncertainties and external disturbances, this control technique guides the system to the sliding surface. The sliding surface is located between the control structures, so that the control law has to switch from one structure to another one. Hence, SMC technique can be classified as one of the variable structure control method. This characteristic of the control law and delay in control switching is a disadvantage when using SMC causing some chattering behavior. To reduce the chattering, some ideas have been suggested. The first idea is to design the switching components control law in the continuous control one for reducing the amplitude of chattering. The second idea is to approximate the signum function by the saturation function that has a high slope. To design the sliding mode controller, a designer defines a sliding surface, and then designs the controller for the reaching phase where the system stays on the sliding surface. In the reaching phase, the controller can be proposed by Lyapunov theory that assures stable conditions on the sliding surface in finite time.

## 2.4.6 Intelligent Controller

Control systems based on biological inspired intelligent systems have become increasingly popular due to the potential to mimic human body functions that have been optimized through years of evolution [30]. These systems can be characterized by their ability to learn from environmental information that will increase their robustness, accuracy and intelligence. Unique characteristic of an intelligent control is that it covers a very wide range of uncertainty compared with other control strategies. This reason led to development of control strategies such as mode predictive, fuzzy logic, and neural network controllers. The following subsection deal with these controllers and application to the quadrotor.

### 2.4.3.1 Model Predictive Controller

By increasing the required coverage of uncertainty, the control strategies need to predict the future behavior of the system and generate the future control input for optimizing a cost function. The model predictive controller (MPC) is categorized as an advanced process control method that is used for maintaining the output at the operational conditions and set points. The MPC strategy is particularly suited for problems with constraints on input, output and states, and varying objectives and limits on variables. Although the MPC needs a precise prediction model and full-state estimation, it has the advantage that it can enforce constraints on inputs and outputs, and its systematic design is easy to maintain.

### 2.4.3.2 Fuzzy Logic

Fuzzy logic control is a heuristic approach that easily embeds the knowledge and key elements of human thinking in the design of nonlinear controllers. Qualitative and heuristic considerations, which cannot be handled by conventional control theory, can be used for control purposes in a systematic form, applying fuzzy control concepts. Fuzzy logic control does not need to accurate mathematical model, can work with imprecise inputs, can handle nonlinearity and can present disturbance insensitivity greater than the most nonlinear controllers. Fuzzy logic controllers usually outperform other controllers in complex, nonlinear, or undefined systems for which a good practical knowledge exists.

Fuzzy logic controllers are based on fuzzy sets, that is, classes of objects in which the transition from membership to non-membership is smooth rather than abrupt. Therefore, boundaries of fuzzy sets can be vague and ambiguous, making them useful for approximation models:

1. A Rule-Base (a set of If-Then rules) holds the knowledge, in the form of a set of rules, of how to achieve the best control result.
2. An Inference Mechanism evaluates control rules related with the current time and then decide what the best input to control the plant.
3. A Fuzzification interface simply modifies the control inputs into information that the inference mechanism can easily utilize to compare to the rules in the rule-base.
4. A Defuzzification interface converts the results of the inference mechanism into actual inputs for the plant.

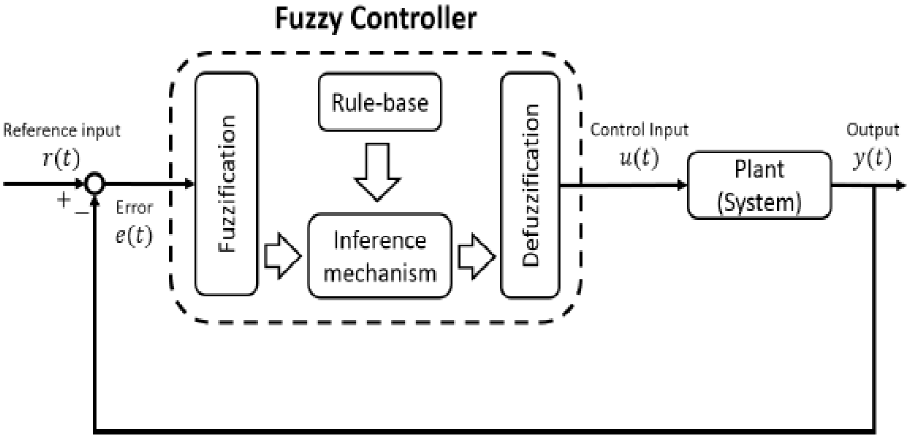

**Figure 2.11:** Block diagram of simple fuzzy logic controller

### 2.4.3.3 Neural Network

A neural network is a series of algorithms that endeavors to recognize underlying relationships in a set of data through a process that mimics the way the human brain operates. In this sense, neural networks refer to systems of neurons, either organic or artificial in nature. Neural network generates the best possible result without needing to redesign the output criteria. The NN control strategy has been used for a design of a nonlinear dynamic system with uncertain nonlinear terms and system errors. The objective of this control strategy is to find the weights for achieving a desired input and output. Through this process termed as training the network, the system obtains a control law overcoming a wide range of uncertainty.

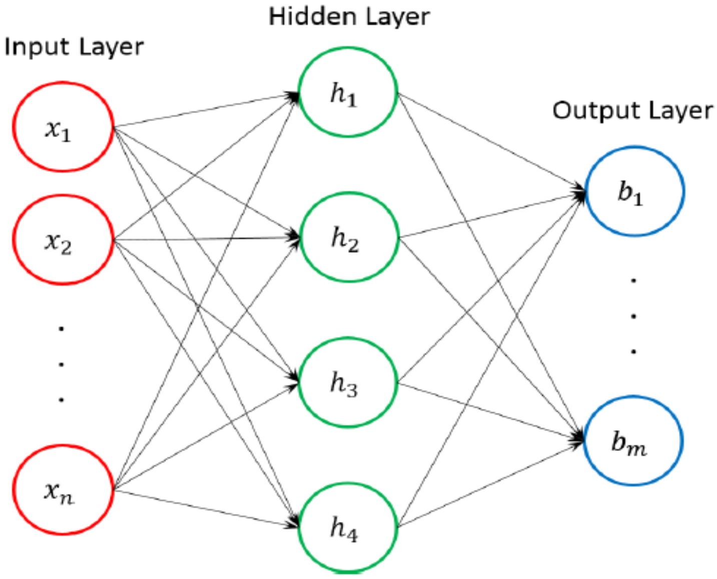

**Figure 2.12:** An example of neural network system

## 2.5 CONCLUSION

In this chapter, qualitative introduction on the principles of working of a quadrotor was discussed, in addition to the mathematical model, which was presented. The full nonlinear equations of the dynamic model have been obtained by classical Euler-Newtonian mechanics with the knowledge of different force imposed on, especially aerodynamic function, as a base and preliminary for the next control and simulation work. Then we studied many different controllers that may be used by quadrotors in a variety of scenarios, where we have shown that there are many previous works have demonstrated that it is possible to control the quadrotor using linear control techniques by linearizing the dynamics around an operating point, usually chosen to be the hover. However, a wider flight envelope and better performances can be achieved by using nonlinear control techniques that consider a more general form of the dynamics of the vehicle in all flight zones. As the number of applications grow, the need for imaginative new control strategies that build upon older ones will undoubtedly grow for better performance. For that, two new adaptive control strategies were proposed to make a quadrotor stably and robustly track a desired attitude under the influences of many constraints as well as input uncertainties and external disturbances. These two adaptive controllers will be presented and detailed in the next chapter.

# Chapter 3

# DESCRIPTION OF ATTITUDE CONTROLLERS ROAC and GAC

## 3.1 INTRODUCTION

In practice, there are various technical challenges in the control of a quadrotor UAV that is subjected to unknown external disturbances, model uncertainties and input constraints. The quadrotor is always subject to different uncertainties, i.e. the quadrotor motions cannot always be described by their exact dynamics.

The unknown disturbances in practical aerospace environments include wind gust, noises, …etc. The model uncertainties of a quadrotor UAV are usually induced by the imprecise hydrodynamic coefficients which arise in the mathematical model of the quadrotor, while the input constraints are caused by the effect of input saturation. Unfortunately, only a few types of research have addressed disturbances, parametric uncertainties and input constraints in simulations or experiments simultaneously.

To tackle these technical issues that were mentioned previously, two methods for tracking attitude control of quadrotors will be introduced in this chapter.

The two adaptive control strategies are based on Lyapunov theory in order to design effective controllers. The first strategy, Robust Optimal Adaptive Control ROAC developed by *M. Navabi* [17], is a composition of two controllers that used Lyapunov based technique optimized by PSO algorithm for controlling nonlinear system in addition to a Disturbance Observer Based Control. For the second strategy, a Geometric Adaptive Control GAC developed by *Shankar Kulumani* [18], is used to stabilize the desired attitude while the controlled attitude avoids undesired regions defined by an inequality constraint. For disturbances rejection, an adaptive update law is added for attitude stabilization.

This chapter is organized as follow. In Section 3.2, the first controller approach ROAC is detailed. The design of ROAC with the stability analysis are given in the same section. The second controller GAC will be presented and detailed in section 3.3. The chapter is ended by conclusion that summarizes the whole chapter.

## 3.2 ROBUST OPTIMAL ADAPTIVE CONTROLLER

In this section, a nonlinear robust optimal adaptive controller ROAC will be studied based on the approach described in [17]. The aim of the controller is to solve attitude tracking control problem in presence of parametric uncertainties, exogenous disturbance, and input constraints.

The procedure proposed consists of two stages. In the first stage, the controller is designed under the assumption that there is no disturbance or the disturbance is measurable. In the second stage, a nonlinear disturbance observer is designed and then integrated with the previous designed controller.

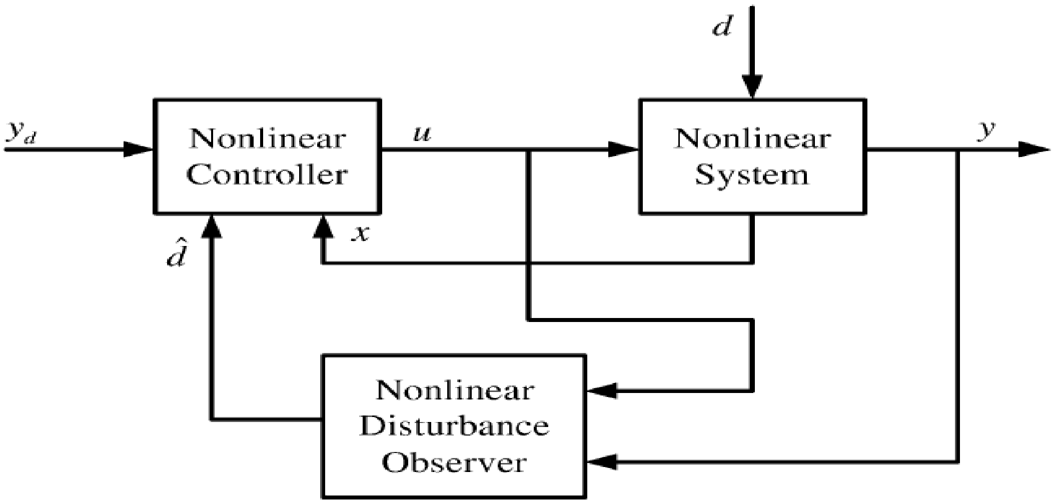

**Figure 3.1**: Structure of nonlinear system controller ROAC

The ROAC control method can be explained in three steps. The first step is the adaptive controller. In this step, an adaptive control law is used to tackle and stabilize the problem of nonlinear and parametric uncertainties of quadrotor. The second step is about using a PSO algorithm, which was combined with the adaptive controller to obtain an optimal adaptive controller that regulate controller parameter offline. In the third step, to improve the robustness of system when there are unknown external disturbances, a nonlinear disturbance observer was added.

Before explaining these steps, the dynamic model of the quadrotor UAV obtained in the previous chapter can be conveniently viewed as a system composed of two subsystems, the position subsystem and the rotational subsystem. Where it is shown that the control inputs appear in altitude and attitude, in such a way that the translation subsystem (x and y) depends on the angle subsystem, but the angle subsystem is independent from the translation one. For that, we can rewrite the equation (2.26) by dividing the simplified dynamic model obtained previously into two subsystems. The first, $\prod_1$ (under-actuated subsystem), comprises linear translations in *x* and *y* directions, and the second one, $\prod_2$ (fully actuated sub-system), contains the dynamics of altitude and attitude. The subsystems $\prod_1$ and $\prod_2$ are as follow:

$$\prod_1 = \begin{cases} \ddot{x} = (\cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi)\frac{u_1}{m} \\ \ddot{y} = (\cos\phi\sin\theta\cos\psi - \sin\phi\sin\psi)\frac{u_1}{m} \end{cases} \tag{3.1}$$

$$\prod_2 = \begin{cases} \ddot{z} = -g + \cos\phi\cos\theta\frac{u_1}{m} + d_1 \\ \ddot{\phi} = \dot{\theta}\dot{\psi}\frac{I_{yy}-I_{zz}}{I_{xx}} - \frac{J_r}{I_{xx}}\dot{\theta}\Omega_r + \frac{u_2}{I_{xx}} + d_2 \\ \ddot{\theta} = \dot{\psi}\dot{\phi}\frac{I_{zz}-I_{xx}}{I_{yy}} + \frac{J_r}{I_{yy}}\dot{\phi}\Omega_r + \frac{u_3}{I_{yy}} + d_3 \\ \ddot{\psi} = \dot{\phi}\dot{\theta}\frac{I_{xx}-I_{yy}}{I_{zz}} + \frac{u_4}{I_{zz}} + d_4 \end{cases} \tag{3.2}$$

Therefore, for the design of controller, it suffices to extract appropriate outputs from subsystem $\prod_2$.

## 3.2.1 Control design

A control system is a system that maintain equilibrium determined by an input set point. By comparing the values that define the overall state or orientation of a system to the desired values. The objective of our controller is forcing the states of subsystem $\prod_2$ to track desired altitude and attitude $x_d = [z_d\ \phi_d\ \theta_d\ \psi_d]^T$ despite the presence of parametric uncertainties, input constraints and external disturbance. To guarantee the tracking performance, this approach is based on Lyapunov theory.

### 3.2.1.1 Adaptive Trajectory Tracking control design

In this part, a control law is obtained to deal with parametric uncertainties problem in the first step. In the second step, we developed a new performance index for evaluating the performance of adaptive controller.

The nonlinear dynamic model associated with fully actuated subsystem $\prod_2$ of the quadrotor, obtained in equation (3.2), can be rewritten in the following form:

$$H(x)\ddot{x} + C(x,\dot{x})\dot{x} + g(x) = U + D \tag{3.3}$$

where

$x = [z\ \phi\ \theta\ \psi]^T$ denotes state vector of system

$U = [u_1\ \ u_2\ u_3\ u_4]$ is the control input vector

D represents measurable disturbance originated from $\Omega_r$ with unknown bound

$H$(x) is the system inertia matrix, $C(x,\dot{x})$ contains centrifugal and Coriolis torques, and $g$(x) indicates the vector of gravitational torques. They are presented as follows:

$$H(x) = \begin{vmatrix} \frac{m}{\cos\phi\cos\theta} & 0 & 0 & 0 \\ 0 & I_{xx} & 0 & 0 \\ 0 & 0 & I_{yy} & 0 \\ 0 & 0 & 0 & I_{zz} \end{vmatrix} \tag{3.4}$$

$$C(x,\dot{x}) = \begin{vmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -\dot{\psi}a_1 + J_r\Omega_r & 0 \\ 0 & -\dot{\psi}a_2 - J_r\Omega_r & 0 & 0 \\ 0 & 0 & -\dot{\psi}a_3 & 0 \end{vmatrix} \tag{3.5}$$

$$g(x) = \begin{vmatrix} \frac{mg}{\cos\phi\cos\theta} \\ 0 \\ 0 \\ 0 \end{vmatrix} \tag{3.6}$$

The terms $a_1, a_2\ and\ a_3$ are defined as:

$a_1 = I_{yy} - I_{zz}$ , $a_2 = I_{zz} - I_{xx}$ , $a_3 = I_{xx} - I_{yy}$

It is very complicated to obtain the control laws to achieve the desired trajectory $x_d$ due to parametric uncertainties problem. For that, we choose a Lyapunov function, which is positive definite itself and has negative semi-definite first order derivative with respect to time. The Lyapunov function is chosen as:

$$\begin{cases} V(t) = \frac{1}{2}[s^T H s + \tilde{a}\Gamma_1^{-1}\tilde{a} + \tilde{D}^T\Gamma_2^{-1}\tilde{D} \\ \tilde{a} = \hat{a} - a, a = \left[m\, I_{xx}\, I_{yy}\, I_{zz}\right]^T,\ \hat{a} = [\hat{m}\, \hat{I}_{xx}\, \hat{I}_{yy}\, \hat{I}_{zz}]^T \\ \tilde{D} = D - \hat{D} \end{cases} \tag{3.7}$$

with $a$, $\hat{a}$ unknown parameter and its estimated; $D$, $\hat{D}$ unknown disturbances and its estimated; $\tilde{a}, \tilde{D}$ estimation errors; and $\Gamma_1, \Gamma_2$ symmetric positive definite matrices.

We define the velocity error $s$ as:

$$s = \dot{\tilde{x}} - \Lambda\tilde{x} = \dot{x} - \dot{x}_r \tag{3.8}$$

where $\tilde{x} = x - x_d$ is tracking error and Λ is a symmetric positive definite matrix.

For obtaining control and adaption laws, the new variable is defined as: $\dot{x}_r = \dot{x}_d - \Lambda\tilde{x}$

The first order derivative of $V(t)$ with respect to time is found as:

$$\dot{V}(t) = s^T H\dot{s} + \frac{1}{2}s^T\dot{H}s + \tilde{a}^T\Gamma_1^{-1}\dot{\hat{a}} - \tilde{D}^T\Gamma_2^{-1}\dot{\hat{D}} \tag{3.9}$$

By using the derivative of equation (3.8), equation (3.9) becomes:

$$\dot{V}(t) = s^T(H\ddot{x} - H\ddot{x}_r) + \frac{1}{2}s^T\dot{H}s + \tilde{a}^T\Gamma_1^{-1}\dot{\hat{a}} - \widetilde{D}^T\Gamma_2^{-1}\dot{\hat{D}} \qquad (3.10)$$

Using equation (3.3), the following expression is calculated:

$$\dot{V}(t) = s^T(U + D - C\dot{x} - g - H\ddot{x}_r) + \frac{1}{2}s^T\dot{H}s + \tilde{a}^T\Gamma_1^{-1}\dot{\hat{a}} - \widetilde{D}^T\Gamma_2^{-1}\dot{\hat{D}} \qquad (3.11)$$

We can replace $(\dot{x})$ by $(s + \dot{x}_r)$, then equation (3.11) becomes:

$$\dot{V}(t) = s^T(U + D - C\dot{x}_r - g - H\ddot{x}_r) + s^T\left(\dot{H} - 2C\right) + \tilde{a}^T\Gamma_1^{-1}\dot{\hat{a}} - \widetilde{D}^T\Gamma_2^{-1}\dot{\hat{D}} \qquad (3.12)$$

The quadratic function associated with a skew-matrix is zero and suppose that $H(x), C(x,\dot{x})\dot{x}\ and\ g(x)$ linearly depend on unknown parameters of system with the design of controller is:

$$U = Y\hat{a} - \widehat{D} - K_D s \qquad (3.13)$$

The derivative of Lyapunov function becomes:

$$\dot{V}(t) = s^T Y\tilde{a} + s^T\widetilde{D} - s^T K_D s + \tilde{a}^T\Gamma_1^{-1}\dot{\hat{a}} - \widetilde{D}^T\Gamma_2^{-1}\dot{\hat{D}} \qquad (3.14)$$

The adaption law and disturbance estimation are chosen as:

$$\dot{\hat{a}} = -\Gamma_1 Y^T s \qquad (3.15)$$

$$\dot{\hat{D}} = \Gamma_2 s \qquad (3.16)$$

The final form of $\dot{V}(t)$, which is negative semi-definite, is obtained as:

$$\dot{V}(t) = -s^T K_D s \leq 0 \qquad (3.17)$$

According to Lyapunov theory and Barbalat's lemma, the errors tracking converge to zero and the stability is guarantee.

The controller design depends on $\Lambda$ , $\Gamma_1$ and $K_D$. To determine the previous parameters, a new performance index minimized is developed.

The performance index $J(\Lambda, \Gamma_1, K_D)$ is a summation of three terms:

- The first term deals with control inputs constraints by minimizing summation of integral of violated control law (SIVCL) for each control input:

$$IVCL_i = \begin{cases} \int_{t=0}^{T} a_i\left|u_i(t) - u_i^{lim}\right|, a_i > 0 \quad if \quad u_i(t) < u_i^{min} or\ u_i(t) > u_i^{max} \\ 0 \qquad\qquad if \qquad u_i^{min} \leq u_i(t) \leq u_i^{max} \end{cases} \qquad (3.18)$$

$$u_i^{lim} = \begin{cases} u_i^{ma} & u_i(t) > u_i^{max} \\ u_i^{min} & u_i(t) < u_i^{min} \end{cases} \tag{3.19}$$

$$SIVCL = \sum_{i=1}^{n} IVCL_i \tag{3.20}$$

where $i = 1, \dots, n$ is the number of actuators.

T is simulation time span.

$u_i$ is control law for the $i^{th}$ actuators.

$u_i^{max}$ *and* $u_i^{min}$ are upper and lower bounds of the $i^{th}$ actuators.

$a_i$ is weighting factor.

We want to find the fastest and optimal response, which is the desired response when SIVCL equal to zero.

- The second term considers time response of closed-loop system to achieve the desired response by using Summation of Settling Time (SST) of all outputs:

$$SST = \sum_{i=1}^{n} b_i\, St_i, \;\; b_i > 0 \tag{3.21}$$

where $b_i$ is weighting factor and $St_i$ is settling time of the $i^{th}$ outputs which is the necessary time to become a response steady.

- The third term exhibits total energy consumption using Summation of Lower Riemann Sum (SLRS) of control laws:

$$SLRS = \sum_{i=1}^{n} c_i \int_{t=0}^{T} u_i\, dt \tag{3.22}$$

where $c_i$ is weighting factor.

The importance of each term and magnitude of weighting coefficients in performance index are determined by designer requirements.

To satisfy the control system objectives, the performance index is written as:

$$J(\Lambda, \Gamma_1, K_D) = SIVCL + SST + SLRS \tag{3.23}$$

In the next step, the performance index will minimize using PSO algorithm.

### 3.2.1.2 Adaptive Control Parameters Optimization

In this part, a Particle Swarm Optimization (PSO) algorithm was used to determine adaptive control parameters in order to satisfy the problem of input constraints by determining the best settling time under constraints on control law.

A PSO algorithm is a stochastic optimization technique. It is inspired by the swarm behavior of birds flocking, and utilizes this behavior to guide the particles to search for globally optimal solutions.

In PSO, a population of particles is spread randomly throughout the search space. The particles are assumed to be flying in the search space. The velocity and position of each particle is updated iteratively based on personal and social experiences. Each particle possesses a local memory in which the best so far achieved experience is stored. Also a global memory keeps the best solution found so far. The sizes of both memories are restricted to one. The local memory represents the personal experience of the particle and the global memory represents the social experience of the swarm. The balance between the effect of the personal and social experiences are maintained using randomized correction coefficients.

The following figure represents a flowchart of the PSO algorithm.

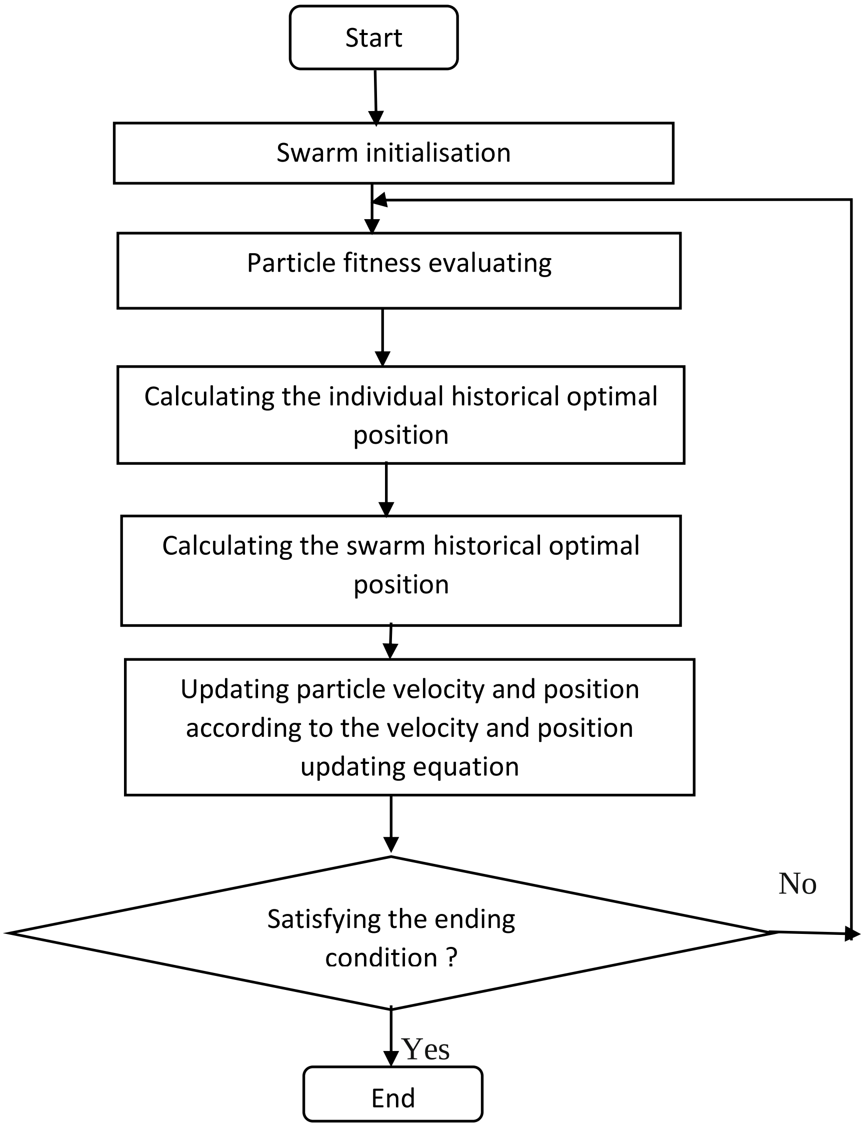
Start
Swarm initialisation
Particle fitness evaluating
Calculating the individual historical optimal position
Calculating the swarm historical optimal position
Updating particle velocity and position according to the velocity and position updating equation
Satisfying the ending condition ?
No
Yes
End

**Figure 3.2:** Flowchart of PSO algorithm

By tracking these steps, we will simulate a PSO algorithm using Matlab code to minimize the parameters of the controlled system in order to achieve to the control objectives in the next chapter.

### 3.2.1.3 Nonlinear Disturbance Observer Design

In the third step, to improve disturbance attenuation ability of nonlinear controllers, a nonlinear disturbance observer NDO will be used.

A nonlinear disturbance observer NDO is designed to deduce external disturbances and then to compensate for the influence of the disturbances using proper feedback.

The design of the controller is separated from the design of the disturbance observer. For that, the disturbance observer will be integrated with the controller by replacing the disturbance in the control law with its estimation yielded by the disturbance observer.

The mathematical model of a quadrotor is described by:

$$\dot{x} = f(x) + g_1(x)u + g_2(x)d \tag{3.24}$$

where

$$\begin{cases} x = [x_1\ x_2\ x_3\ x_4\ x_5\ x_6\ x_7\ x_8]^T = [z\ \dot{z}\ \phi\ \dot{\phi}\ \theta\ \dot{\theta}\ \psi\ \ \dot{\psi}]^T \ \text{ is the state vector} \\ u = [u_1\ u_2\ u_3\ u_4]^T \qquad \text{is input vector} \\ d = [d_1\ d_2\ d_3\ d_4]^T \qquad\quad \text{is external disturbance vector} \end{cases} \tag{3.25}$$

It is assumed that $f(x), g_1(x), g_2(x)$ are smooth nonlinear functions in terms of x.
Using equation (3.2) and (3.25), we obtain:

$$f(x) = \begin{bmatrix} x_2 \\ -g \\ x_4 \\ I_1x_6x_8 \\ x_6 \\ I_2x_4x_8 \\ x_8 \\ I_3x_4x_6 \end{bmatrix} \quad g_2\ (x) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},\ g_1(x) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \frac{m}{\cos x_3 \cos x_5} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \frac{1}{I_{xx}} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{I_{yy}} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{I_{zz}} \end{bmatrix} \tag{3.26}$$

We supposed that the disturbance $d$ is generated by a linear exogenous system, then the model of external disturbance can be represented as:

$$\begin{cases} \dot{\chi} = A\chi \\ d = C\chi \end{cases} \tag{3.27}$$

where $\chi = [\chi_1\ \chi_2\ \chi_3\ \chi_4\ \chi_5\ \chi_6\ \chi_7\ \chi_8]^T$ is the state vector of the exogenous system,

$A \in \mathbb{R}^{8\times8}$ and $C \in \mathbb{R}^{4\times8}$are given known matrices.

In NDO designing, it assumed there are four different disturbance sources:

$$\begin{cases} d_1 = 2\sin(2t+1) \\ d_2 = 0.25\sin(3t+1) \\ d_3 = 0.05\sin(2t+1) \\ d_4 = 0.05\sin(t) \end{cases} \tag{3.28}$$

To estimate the unknown disturbance d, a basic disturbance observer is designed as:

$$\dot{\hat{\chi}} = A\chi + L(x)g_2(x)e_d \tag{3.29}$$

with $e_d = d - \hat{d}$ and $L(x)$ is the nonlinear gain function of the observer.

Using $e_d$ with equations (3.24) and (3.29), the nonlinear observer becomes:

$$\dot{\hat{\chi}} = A\chi + L(x)(\dot{x} - f(x) - g_1(x)u - g_2(x)\hat{d}) \tag{3.30}$$

However, the above disturbance observer cannot be implemented since the derivative of the state is required (sensors problem). A new nonlinear disturbance observer is then proposed after defining a following auxiliary variable:

$$\kappa = \hat{\chi} - \varsigma(\mathrm{x}) \tag{3.31}$$

$\varsigma$(x) is a nonlinear function to be designed. The nonlinear observer gain L(x) is then determined by: $L(x) = \frac{\partial\varsigma(x)}{\partial x}$

The nonlinear observer is obtained by deriving $\kappa$ as:

$$\dot{\kappa} = (A - L(x)g_2(x)C)\kappa + A\varsigma(x) - L(x)\big(f(x) + g_1(x)u - g_2(x)C\varsigma(x)\big) \tag{3.32}$$

Consider $e = \chi - \hat{\chi}$ is the estimation error. The disturbance observer can exponentially track the disturbance if the nonlinear gain function L(x) is chosen such that the error dynamics is globally exponentially:

$$\dot{e} = (A - L(x)g_2C)e \tag{3.33}$$

where $(A - L(x)g_2C)$ is considered as Hurwits matrix with :

$$\varsigma(x) = [L_{12}x_2\ 0\ L_{34}x_4\ 0\ L_{56}x_6\ 0\ L_{78}x_8\ 0]^T \tag{3.34}$$

### 3.2.2 Stability analysis

In order to verify and ensure the stability of the closed-loop performance of the system, M.*Navabi* has used a Lyapunov theory to performed stability analysis.
The composite controller that compensate the external disturbance is:

$$U_c = U + \gamma d \tag{3.35}$$

Using the composite controller in the mathematical model of quadrotor equation (3.24) yields:

$$\dot{x} = f(x) + g_1(x)U + g_1(x)\gamma d + g_2(x)d \tag{3.36}$$

Satisfying the condition of disturbance rejection

$$g_1(x)\gamma = -g_2(x), \tag{3.37}$$

the closed-loop dynamic of quadrotor becomes:

$$\dot{x} = f(x) + g_1(x)U \tag{3.38}$$

which shows that the control law $U = Y\hat{a} - \widehat{D} - K_D s$ is the responsible about global stability of the system and convergence of tracking.

For more robustness, the disturbance $d$ in the control law is replaced with its estimation $\hat{d}$ yielded by the disturbance observer equation (3.33), what makes the closed-loop system can rewrite as:

$$\dot{x} = f(x) + g_1(x)U + g_2(x)\left(d - \hat{d}\right) \tag{3.39}$$

For the stability analysis, the composite system of closed-loop and observer error dynamics is used:

$$\begin{cases} \dot{\mathrm{x}} = \mathrm{f(x)} + \mathrm{g_1(x)U} + \mathrm{g_2(x)Ce} \\ \dot{\mathrm{e}} = (\mathrm{A} - \mathrm{L(x)g_2(x)C})\mathrm{e} \end{cases} \tag{3.40}$$

The Lyapunov function of the system above is given by:

$$V(x,e) = V_{controller} + V_{observer} \tag{3.41}$$

where

$$\begin{cases} V_{controller} = \frac{1}{2}\left[s^T H s + \tilde{a}^T \Gamma_1^{-1} \tilde{a} + \widetilde{D}^T \Gamma_2^{-1} \widetilde{D}\right] \\ V_{observer} = e^T P e \quad \mathit{with\ P\ a\ positive\ definite\ matrix} \end{cases} \tag{3.42}$$

Deriving equation (3.41):

$$\dot{V}(x,e) = \frac{\partial V_{controller}}{\partial x}(f + g_1(x)U) + \frac{\partial V_{controller}}{\partial x} g_2(x) + 2Pe^T(A - L(x)g_2(x)C)e \tag{3.43}$$

where

$$\begin{cases} \frac{\partial V_{controller}}{\partial x}(f + g_1(x)U) = -s^T K_D s \le 0 < \delta_1\|x\| \text{ with } \delta_1 \text{is a small positive scalar} \\ 2Pe^T(A - L(x)g_2(x)C)e < -\delta e^T e \text{ with } \delta \text{ is a small positive scalar} \end{cases} \tag{3.44}$$

The final form of Lyapunov derivative function is:

$$\dot{V}(x,e) < -\delta_1\|x\| + \frac{\partial V_{controller}}{\partial x} g_2(x)Ce - \delta e^T e \tag{3.45}$$

As a result according to Lyapunov first derivative and a method described in [34], for every state and observer error, the system is stable and they converge to origin as $t \to \infty$.

## 3.3 GEOMETRIC ADAPTIVE CONTROLLER

This second section presents nonlinear geometric adaptive control GAC, developed by *Shankar Kulumani* in [18], for a quadrotor unmanned aerial vehicle under the influence of uncertainties. Assuming that there exist unstructured disturbances in the translational dynamics and the attitude dynamics, a geometric nonlinear controller is developed to follow an attitude tracking command. In particular, a new form of an adaptive control term is proposed to guarantee asymptotical convergence of tracking error variables when there exist uncertainties at the dynamics of quadrotors where the disturbances are considered arbitrary without any simplification. The corresponding stability properties are analyzed mathematically, and it is verified by several simulations in the next chapter.

Therefore, the first step in control design is to propose a configuration error function on $SO(3)$ with a logarithmic barrier function to avoid constrained regions with the advantage to handle an arbitrary number of constrained regions without modification. Then, an adaptive update law will be developed to tackle the external disturbances.

### 3.3.1 Attitude dynamics and inequality constraints

Rigid body attitude control is an important problem for aerospace vehicles because of problems associated with its different parameterization. In this study, singularities and ambiguities of local parameterization such as Euler angles and quaternion are completely avoided using the rotation matrix as a parametrization of the attitude to generate agile maneuvers in a uniform way. The control of a trajectory tracking problem requires state feedback to define tracking errors between the current states and the desired states. Since the closed loop system dynamics evolve on nonlinear manifolds, which describe the configuration space of the quadrotor attitude $R \in SO(3)$ , error functions are defined on the same manifolds.

The nonlinear manifold is a topological space that locally resembles Euclidean space near each point. More precisely, an n-manifold is a topological space with the property that each point has a neighborhood that is homeomorphic to the Euclidean space of dimension n. The concept of a manifold is central to many parts of geometry and modern mathematical physics because it allows complicated structures to be described and understood in terms of the simpler local topological properties of Euclidean space.

In order to design an attitude controller, we will first define the attitude dynamics in SO(3) group and the inequality constraints.

### 3.3.1.1 Attitude Dynamics in SO(3)

The attitude dynamic equations of a rigid body are given by, [35]:

$$I_r\dot{\Omega}_B + \Omega_B \times I_r\Omega_B = u + W(R, \Omega_B)\Delta \tag{3.46}$$

$$\dot{R} = R\widehat{\Omega}_B \tag{3.47}$$

where $I_r \in \mathbb{R}^{3\times3}$ is the inertia matrix, $\Omega_B \in \mathbb{R}^3$is the angular velocity expressed with respect to the body fixed frame, and $R \in SO(3)$ is a rotation matrix that represents the transformation of the representation of a vector from a fixed frame to the inertial reference frame. The control moment is denoted by $u \in \mathbb{R}^3$ and it is expressed with respect to the body fixed frame. It assume that the external disturbance is expressed by $W(R, \Omega_B)\Delta$, where $W(R, \Omega_B)\colon SO(3) \times \mathbb{R}^3 \to \mathbb{R}^{3\times p}$ is a known function of the attitude and the angular velocity. The disturbance is represented by $\Delta\in \mathbb{R}^{p}$ and is unknown but fixed uncertain parameter. Furthermore, $W(R, \Omega_B)$ and $\Delta$ are bounded by

$$\|W\| \leq B_W \quad , \ \|\Delta\| \leq B_\Delta \tag{3.48}$$

This form of uncertainty enters the system dynamics through the input channel and as a result is referred to as a matched uncertainty. While this form of uncertainty is easier than the unmatched variety many physically realizable disturbances may be modeled in this manner.

The hat map $\wedge\colon \mathbb{R}^3 \to \mathrm{SO}(3)$ represents the transformation of a vector in $\mathbb{R}^3$ to a $3 \times 3$ skew-symetric matrix such that $\hat{x}y = x \times y$ for any $x, y \in \mathbb{R}^3$ [36]. It replaces S(.) in equation (1.6). The inverse of the hat map is the vee map $\vee\colon \mathbb{R}^3 \to \mathrm{SO}(3)$. The properties of the hat map are:

$$x.\hat{y}z = y.\hat{z}x \ , \hat{x}\hat{y}z = (x.z)y - (x.y)z \tag{3.49}$$

$$\widehat{x \times y} = \hat{x}\hat{y} - \hat{y}\hat{x} = yx^T(A - A^T)^\vee \tag{3.50}$$

$$tr[A\hat{x}] = \frac{1}{2}tr[\hat{x}(A - A^T)] = -x^T(A - A^T)^\vee \tag{3.51}$$

$$\hat{x}A + A^T\hat{x} = (\{tr[A]I_{3\times3} - A\}x)^\wedge \tag{3.52}$$

$$R\hat{x}R^T = (Rx)^\wedge \ , \quad R(x \times y) = Rx \times Ry \tag{3.53}$$

for any $x, y, z \in \mathbb{R}^3$, $A \in \mathbb{R}^{3\times3}$ and $R \in SO(3)$ . In this study, the maximum eigenvalue and the minimum eigenvalue of J are denoted by $\lambda_M \ ,\lambda_m$ respectively . The Frobenius norm of the matrix A is denoted by $\|A\| \leq \|A\|_f = \sqrt{tr(A^TA)} \leq \sqrt{rank(A)}\|A\|$

### 3.3.1.2 State Inequality Constraint

The two-sphere is the manifold of unit-vectors in $\mathbb{R}^3$ such that $S^2 = \{q \in \mathbb{R}^3 | \,\|q\| = 1\}$. We define $r \in S^2$ to be a unit vector from the mass center of the rigid body along a certain direction and it is represented with respect to the body-fixed frame. Let $v \in S^2$ is defined to be a unit vector from the mass center of the rigid body toward an undesired pointing direction and represented in the inertial reference frame.

It is further assumed that optical sensor has a strict non-exposure constraint with respect to the celestial object. This hard constraint is formulated as:

$$\mathrm{r}^{\mathrm{T}}\mathrm{R}^{\mathrm{T}}\mathrm{v} \leq \cos\theta \tag{3.54}$$

where $0° \leq \theta \leq 90°$ is the required minimum angular separation between r and $\mathrm{R}^{\mathrm{T}}\mathrm{v}$.

The objective here is to determine a control input that insure the stabilization of the system from an initial attitude $\mathrm{R}_0$ to a desired attitude $\mathrm{R}_\mathrm{d}$ while satisfying (3.54)

## 3.3.2 Attitude Control on $SO(3)$ with Inequality Constraints

In order to achieve the aim of this study which is designing a geometric adaptive control that ensure the stabilization of the system where the disturbances are considered, the author first, selects a proper configuration error function that is positive definite function which measures the error between the current configuration and the desired configuration. Then he defines a configuration error vector and a velocity error vector in the tangent space through the derivative of [36]. A rigorous Lyapunov analysis is presented to establish stability properties without any timescale separation assumption. To handle the attitude inequality constraint, the author proposes a new attitude configuration error function. More explicitly, he extends the trace form used in [36], for attitude control on with the addition of a logarithmic barrier $SO(3)$ function. Based on the proposed configuration error function, nonlinear geometric attitude controllers are constructed. A smooth control system is first developed assuming that there is no disturbance, and then it is extended to include an adaptive update law for stabilization in the presence of unknown disturbances. The proposed attitude configuration error function and error dynamic are summarized as follows.

- **Attitude error function**

Recall that R is the rotation matrix to describe the quadrotor attitude, and $\mathrm{R}_\mathrm{d}$ is the desired rotation matrix. To describe the relative rotation from the body frame to the desired frame, an attitude error function $\Psi\colon SO(3) \to \mathbb{R}^3$, an attitude error vector $\mathrm{e}_\mathrm{R} \in \mathbb{R}^3$ and an angular velocity $\mathrm{e}_\Omega \in \mathbb{R}^3$ , are defined as follows :

$$\Psi(\mathrm{R}) = \mathrm{B(R)A(R)} \tag{3.55}$$

$$\mathrm{e_R} = \mathrm{e_{R_A}B(R) + A(R)e_{R_B}} \tag{3.56}$$

$$\mathrm{e}_\Omega = \Omega \tag{3.57}$$

with

$$\mathrm{A(R)} = \frac{1}{2}\,\mathrm{tr[G(I - R_d^T R)]} \tag{3.58}$$

$$\mathrm{B(R)} = 1 - \frac{1}{\alpha}\ln\left(\frac{\cos\theta - \mathrm{r^T R^T v}}{1 + \cos\theta}\right) \tag{3.59}$$

$$\mathrm{e_{R_A}} = \frac{1}{2}\mathrm{(GR_d^T - R^T R_d G)}^\vee \tag{3.60}$$

$$\mathrm{e_{R_B}} = \frac{\mathrm{(R^T v)}^\vee \mathrm{r}}{\alpha(\mathrm{r^T R^T v} - \cos\theta)} \tag{3.61}$$

where $\alpha \in \mathbb{R}$ is a positive constant and $\mathrm{G} \in \mathbb{R}^{3\times3}$ is a diagonal matrix.

Then, the following properties hold:

**(i)** It's noted that equation (3.58) is positive definite function about $\mathrm{R_d}$ . We have $0° \leq \theta \leq 90°$ which is the constraint angle , so that $0 \leq \cos\theta$ , then we have the term $\mathrm{r^T R^T v}$ represents the cosine of the angle between the body fixed vector and the inertial vector v , it follows that

$$0 \leq \frac{\cos\theta - \mathrm{r^T R^T v}}{1 + \cos\theta} \leq 1$$

for all $\in \mathrm{SO}(3)$. Then its negative logarithm is always positive and $1 < \mathrm{B}$.

As a result, the error function $\Psi = \mathrm{AB}$ , that is composed of two positive terms, is always positive definite and it is minimized at $\mathrm{R} = \mathrm{R_d}$

**(ii)** The variation of A(R) with respect to a variation of $\delta\mathrm{R} = \mathrm{R}\hat{\eta}$ for $\eta \in \mathbb{R}^3$ is given by :

$$\mathrm{D_R A}.\,\delta\mathrm{R} = \eta.\,\mathrm{e_{R_A}} \tag{3.62}$$

Using the several property of the hat map that was given in equation (3.51) and the variation of equation (3.58), which was taken with respect to $\delta\mathrm{R} = \mathrm{R}\hat{\eta}$, equation (3.62) becomes:

$$\mathrm{D_R A}.\,\delta\mathrm{R} = \eta.\frac{1}{2}\mathrm{(GR_d^T - R^T R_d G)}^\vee \tag{3.63}$$

**(iii)** The variation of B(R) with respect to a variation of $\delta\mathrm{R} = \mathrm{R}\hat{\eta}$ for $\eta \in \mathbb{R}^3$ is given by :

$$\mathrm{D_R B}.\,\delta\mathrm{R} = \eta.\,\mathrm{e_{R_B}} \tag{3.64}$$

Using the scalar triple product from equation (3.49) with the variation of equation (3.59), the final form of equation (3.64) is given as:

$$\mathrm{D_R B.\,\delta R = \eta.\frac{(R^T v)^{\vee} r}{\alpha(r^T R^T v - \cos\theta)}} \tag{3.65}$$

**(iv)** There are four critical points of $\mathrm{e_{R_A}}$ , the desired attitude $\mathrm{R_d}$ as well as rotations about each body fixed axis by 180° and the repulsive error vector $\mathrm{e_{R_B}}$ is zero only when the numerator $\mathrm{(R^T v)^{\vee} r = 0}$ . This condition only occurs if the desired attitude results in the body fixed vector r becoming aligned with $\mathrm{R^T v}$ while simultaneously satisfying $\{\mathrm{R_d}\} \cup \{\mathrm{R_d}\ \exp(\pi\hat{s})\}$ for s∈{ $\mathrm{e_1, e_2, e_3}$}.

**(v)** Since it assumes the system will not operate in violation of the constraints, the addition of the barrier function does not add additional critical points to the control system. The desired equilibrium is $\mathrm{e_R = 0}$ and A=0. Therefore, An upper bound of $\|\mathrm{e_{R_A}}\|$ is given as :

$$\|\mathrm{e_{R_A}}\|^2 \leq \frac{\mathrm{A(R)}}{\mathrm{b_1}} \tag{3.66}$$

where the constant $\mathrm{b_1}$ is given by $\mathrm{b_1 = \frac{h_1}{h_2+h_3}}$ for :

$$\begin{cases} h_1 = \min\{g_1 + g_2,\ \ g_2 + g_3,\ \ g_3 + g_1\} \\ h_2 = \min\{(g_1 - g_2)^2, (g_2 - g_3)^2, (g_3 - g_1)^2\} \\ h_3 = \min\{(g_1 + g_2)^2, (g_2 + g_3)^2, (g_3 + g_1)^2\} \end{cases}$$

- **Error dynamic**

In order to find the attitude error dynamics for $\Psi$, some error dynamics of the system will be defined.

Firstly, using the kinematics from equation (3.47) and noting that $\dot{R}_d = 0$, the time derivative of $R_d^T R$ is given as:

$$\frac{d(R_d^T R)}{dt} = R^T\,R\hat{e}_{\Omega}$$

(3.67)

By substituting equation (3.67) in equation (3.58), the time derivative of $A(R)$ is given as:

$$\frac{d(A(R))}{dt} = -\frac{1}{2}\ tr[GR_d^T R\hat{e}_{\Omega}] \tag{3.68}$$

Applying the hat map property from equation (3. 51), equation (3.68) becomes:

$$\mathrm{\frac{d}{dt}(A(R)) = e_{R_A}.\,e_{\Omega}} \tag{3.69}$$

The time derivative of the repulsive error function is given by:

$$\frac{d(B(R))}{dt} = \frac{r^T(\hat{\Omega}_B R^T)v}{\alpha(r^T R^T v - \cos\theta)} \tag{3.70}$$

Using the scalar triple product that was given by equation (3.59), the equation (3.70) can be rewritten as:

$$\frac{d}{dt}(B(R)) = e_{R_B}.e_\Omega \tag{3.71}$$

By deriving equation (3.60) and using the hat map property given in equation (3.52), the derivative of attractive attitude error vector, $e_{R_A}$, is given by :

$$\frac{d\left(e_{R_A}\right)}{dt} = \frac{1}{2}(\hat{e}_\Omega R^T R_d G + (R^T R_d G)^T \hat{e}_\Omega)^\vee = \mathrm{E}(\mathrm{R}, \mathrm{R_d})\mathrm{e}_\Omega \tag{3.72}$$

where $E(R, R_d)$ is a matrix $\in \mathbb{R}^{3\times3}$ given by:

$$E(R, R_d) = \frac{1}{2}(tr[R^T R_d G]I - R^T R_d G) \tag{3.73}$$

To find the time derivative of the repulsive attitude error vector $e_{R_B}$, we first derive equation (3.61):

$$\frac{d(e_{R_B})}{dt} = a\Omega_B v^T Rr - aR^T v\Omega_B^T r + bR^T\hat{v}Rr \tag{3.74}$$

with $a \in \mathbb{R}$ and $b \in \mathbb{R}$ given by:

$$a = [\alpha(r^T R^T v - \cos\theta)]^{-1} \quad , \quad b = \frac{r^T\hat{\Omega}_B R^T v}{\alpha(r^T R^T v - \cos\theta)^2}$$

Then, using the scalar triple product from equation (3.49) as $r.\Omega_B \times (R^T v) = (R^T v).r \times \Omega_B$, equation (3.74) becomes:

$$\frac{\mathrm{d}}{\mathrm{dt}}(\mathrm{e}_{\mathrm{R_B}}) = \mathrm{F(R)}\mathrm{e}_\Omega \tag{3.75}$$

where $F(R)$ is a matrix $\in \mathbb{R}^{3\times3}$ given as follows:

$$F(R) = \frac{1}{\alpha(r^T R^T v - \cos\theta)}[(v^T Rr)I - R^T vr^T + \frac{R^T\hat{v}Rrv^T R\hat{r}}{(r^T R^T v - \cos\theta)}] \tag{3.76}$$

Deriving the configuration error function gives:

$$\frac{d(\Psi)}{dt} = \dot{A}B + A\dot{B} \tag{3.77}$$

By substituting equations (3.58), (3.59), (3.69) and (3.71) into equation (3.77) with the use of equation (3.56), the final form of equation (3.77) is given as follows:

$$\frac{\mathrm{d}}{\mathrm{dt}}(\Psi) = \mathrm{e_R}.\mathrm{e}_\Omega \tag{3.78}$$

The time derivatives of $\mathrm{e_R}$, $\mathrm{e}_\Omega$ are respectively:

$$\frac{\mathrm{d}}{\mathrm{dt}}(\mathrm{e_R}) = \dot{e}_{R_A}\mathrm{B(R)} + \mathrm{e_{R_A}}\dot{\mathrm{B}}(\mathrm{R}) + \dot{\mathrm{A}}(R)\mathrm{e_{R_B}} + \mathrm{A(R)}\dot{e}_{R_B} \tag{3.79}$$

$$\frac{d}{dt}(e_\Omega) = {I_r}^{-1}(-\Omega_B \times J\Omega_B + u + W(R, \Omega_B)\Delta) \tag{3.80}$$

### 3.3.2.1 Attitude controller without disturbance

In this subsection, a nonlinear geometric controller for the stabilization of a rigid body is going to be introduced, where we assume that there is no disturbances, i.e. $\Delta = 0$.

For a given desired attitude command $(R_d, \Omega_d = 0)$, that satisfies the condition in equation (3.54) and positive constraints $k_R, k_\Omega \in \mathbb{R}$, a control input $u \in \mathbb{R}^3$ is defined as follows:

$$u = -k_R e_R - k_\Omega e_\Omega + \Omega_B \times I_r \Omega_B \tag{3.81}$$

Consider the following Lyapunov function:

$$V = \frac{1}{2} e_\Omega . I_r e_\Omega + k_R(R, R_d) \tag{3.82}$$

From the property **(i)** above, $V \geq 0$ . Using equations (3.78) and (3.80) with $\Delta = 0$, the time derivative of V is given by:

$$\dot{V} = -k_\Omega \|e_\Omega\|^2 \tag{3.83}$$

Since V is positive definite and $\dot{V}$ is negative semi-definite, the zero equilibrium point $e_R, e_\Omega$ is stable in the sense of Lyapunov. This also implies $\lim\limits_{t\to\infty} \|e_\Omega\| = 0$ and $\|e_R\|$ is uniformly bounded, as the Lyapunov function is non-increasing. From equations (3.72) and (3.75), $\lim\limits_{t\to\infty} \dot{e}_R = 0$ . One can show that $\|\ddot{e}_R\|$ is bounded. From Barbalat's Lemma, it follows $\lim\limits_{t\to\infty} \dot{e}_R = 0$ . Therefore, the zero equilibrium of the attitude error is asymptotically stable.

Furthermore, since $\dot{V} \leq 0$ , then the Lyapunov function is uniformly bounded which implies:

$$\Psi(R(t)) \leq V(t) \leq V(0)$$

In addition, the logarithmic term in (3.59) ensures $\Psi(R) \to \infty$ as $r^T R^T v \to \cos\theta$ . Therefore, the inequality constraint is always satisfied given that the desired equilibrium lies in feasible set. Then is asymptotically stable, and the inequality constraint is satisfied.

### 3.3.2.2 Adaptive Control

In this case, an adaptive controller for attitude control in the presence of external disturbance $\Delta$, which is fixe but unknown, will be detailed to asymptotically stabilize the system to a desired attitude.

First, it's necessary to show that $\dot{\boldsymbol{e}}_{\boldsymbol{R}}$ is bounded. For this, consider a domain D about the desired attitude defined as:

$$D = \{R \in SO(3) | \Psi < \psi < h_1, r^T R^T v < \beta < \cos\theta\} \tag{3.84}$$

Then the following statements hold:

**(i)** The selected domain ensues that the configuration error function is bounded $\Psi < \psi$. This implies that both A(R) and B(R) are bounded by constants $c_A c_B < \psi < h_1$ . Furthermore, since $\|B\| > 1$ this ensures that $c_A, c_B < \psi$, which defines the upper bounds of A(R) and B(R) by:

$$A(R) < c_A \quad , \quad B(R) < c_B \tag{3.85}$$

**(ii)** Upper bounds of $E(R, R_d)$ and F(R) are given by using the Frobenius norm . The Frobenius norm $\|E\|_F$ is given as :

$$\|E\|_F^2 \leq \frac{1}{4}(tr[G^2] + tr[G^2]) \leq \frac{1}{2} tr[G]^2 \tag{3.86}$$

since $\|E\| \leq \|E\|_F$ , which gives:

$$\|E\| \leq \frac{1}{\sqrt{2}} tr[G] \tag{3.87}$$

The Frobenius norm $\|F\|_F$ is:

$$\|F\|_F = \frac{1}{\alpha^2 (r^T R^T v - \cos\theta)^2} \left[tr[a^T a] - 2tr[a^T b] + 2tr[a^T c] + tr[b^T b] - 2tr[b^T c] + tr[c^T c]\right] \tag{3.88}$$

where the terms a, b and c are given by:

$$a = r^T R r I \ , \ b = R^T v r^T \ , \quad c = \frac{R^T \hat{v} R r v^T R \hat{r}}{r^T R^T v - \cos\theta}$$

which gives the upper bound as follows:

$$\|F\| \leq \frac{(\beta^2 + 1)(\beta - \cos\theta)^2 + 1 + \beta^2(\beta^2 - 2)}{\alpha^2 (\beta - \cos\theta)^4} \tag{3.89}$$

**(iii)** Upper bounds of the attitude error vectors $e_{R_A}$ and $e_{R_B}$ are given as:

$$\|e_{R_A}\| \leq \sqrt{\frac{\psi}{b_1}} \tag{3.90}$$

$$\|e_{R_B}\| \leq \frac{\sin\theta}{\alpha(\cos\theta - \beta)} \tag{3.91}$$

These results are combined to yield a maximum upper bound of the time derivative of the attitude error vector $\dot{e}_R$ as:

$$\|\dot{e}_R\| \leq H\|e_\Omega\| \quad (3.92)$$

with $H \in \mathbb{R}$ is defined as:

$$H = \|B\|\|E\| + 2\|e_{R_A}\|\|e_{R_B}\| + \|A\|\|F\| \quad (3.93)$$

## Adaptive Attitude Control

Given a desired attitude command $(R_d, \Omega_d = 0)$ , and positive constraints $k_R, k_\Omega, k_\Delta, c \in \mathbb{R}$., a control input $u \in \mathbb{R}^3$ and an adaptive update law for the estimated uncertainty $\bar{\Delta}$ are defined as follows:

$$u = -k_R e_R - k_\Omega e_\Omega + \Omega_B \times I_r \Omega_B - W\bar{\Delta} \quad (3.94)$$

$$\dot{\bar{\Delta}} = k_\Delta W^T(e_\Omega + ce_R) \quad (3.95)$$

Consider the Lyapunov function V given by:

$$V = \frac{1}{2} e_\Omega . I_r e_\Omega + k_R \Psi + cI_r e_\Omega . e_R + \frac{1}{2k_\Delta} e_\Delta . e_\Delta \quad (3.96)$$

Over the domain D in equation (3.84), the Lyapunov function is bounded in D by

$$\mathrm{V} \leq \mathrm{z^T W z} \quad (3.97)$$

where $\mathrm{e}_\Delta = \Delta - \bar{\Delta}$ , $\mathrm{z} = [\|\mathrm{e_R}\|, \|\mathrm{e}_\Omega\|, \|\mathrm{e}_\Delta\|]^\mathrm{T} \in \mathbb{R}^3$ and the matrix $\mathrm{W} \in \mathbb{R}^{3\times3}$ is given by:

$$\mathrm{W} = \begin{bmatrix} \mathrm{k_R}\psi & \frac{1}{2}\mathrm{c}\lambda_\mathrm{M} & 0 \\ \frac{1}{2}\mathrm{c}\lambda_\mathrm{M} & \frac{1}{2}\lambda_\mathrm{M} & 0 \\ 0 & 0 & \frac{1}{2\mathrm{k}_\Delta} \end{bmatrix}$$

The time derivative of V with the control inputs (3.94) is given by:

$$\dot{\mathrm{V}} = -\mathrm{k}_\Omega \mathrm{e}_\Omega^\mathrm{T} \mathrm{e}_\Omega + (\mathrm{e}_\Omega + \mathrm{ce_R})^\mathrm{T} \mathrm{We}_\Delta - \mathrm{k_R ce_R^T e_R} - \mathrm{k}_\Omega \mathrm{ce_R^T e}_\Omega + cI_r \mathrm{e}_\Omega^\mathrm{T} \dot{\mathrm{e}}_\mathrm{R} - \frac{1}{\mathrm{k}_\Delta} \mathrm{e}_\Delta^\mathrm{T} \dot{\bar{\Delta}} \quad (3.98)$$

where $\dot{\mathrm{e}}_\Delta = -\dot{\bar{\Delta}}$ is used. The terms linearly dependent on $\mathrm{e}_\Delta$ are combined with (3.95) to yield

$$\mathrm{e}_\Delta^\mathrm{T}(\mathrm{W^T}(\mathrm{e}_\Omega + \mathrm{ce_R}) - \frac{1}{\mathrm{k}_\Delta} \dot{\bar{\Delta}} = 0 \quad (3.99)$$

An upper bound on $\dot{\mathrm{V}}$ is written as:

$$\dot{\mathrm{V}} \leq -\xi^\mathrm{T} \mathrm{M} \xi \quad (3.100)$$

where $\xi = [\|\mathrm{e_R}\|, \|\mathrm{e}_\Omega\|] \in \mathbb{R}^2$, and the matrix $\mathrm{M} \in \mathbb{R}^{2\times2}$ is:

$$\mathrm{M} = \begin{bmatrix} \mathrm{k_R c} & \frac{\mathrm{k_\Omega c}}{2} \\ \frac{\mathrm{k_\Omega c}}{2} & \mathrm{k_\Omega - c\lambda_M H} \end{bmatrix} \tag{3.101}$$

To satisfy that the matrix M is positive definite, c is chosen as:

$$0 < c < \frac{4k_R k_\Omega}{k_\Omega^2 + 4k - R\lambda_M H} \tag{3.102}$$

This implies that $\dot{\mathrm{V}}$ is negative semidefinite and $\lim_{t \to \infty} \xi = 0$ . As the Lyapunov function is non-increasing, z is uniformly bounded. The zero equilibrium of the error vectors is stable in the sense of Lyapunov. Furthermore, $e_R, e_\Omega \to 0$ as $t \to \infty$ , and $\bar{\Delta}$ is uniformly bounded.

## 3.4 CONCLUSION

This chapter presents detailed studies about two nonlinear strategies of tracking attitude control of UAV quadrotors, where a robust optimal adaptive control method was used in the first strategy to deal with parametric uncertainties, external disturbances and input constraints simultaneously, and a geometric adaptive control of attitude dynamics on $SO(3)$ with state inequality constraints in the second strategy. In the first controller, the parametric uncertainties have been compensated using the adaptive nonlinear control method based on Lyapunov stability arguments, that has been widely used as a suitable choice. While a nonlinear disturbance observer was used to deduce the external disturbances and finally the inputs of system were optimized using a PSO algorithm. For the second controller, the proposed control system is developed on $SO(3)$ and avoids the singularities of Euler angles parameters while incorporating state inequality constraints. In addition, the unwinding and double coverage ambiguity of quaternions are also completely avoided. The control system handles uncertain disturbances while avoiding constrained regions using an update adaptive law. The validity of the developed controllers will be confirmed by several simulations in the next chapter.

# Chapter 4

# SIMULATION RESULTS

## 4.1 INTRODUCTION

To verify the control methods that have been proposed and developed in chapter 3, several simulations using MATLAB/CODE have been conducted in this chapter. The performances of the proposed methods are evaluated on a high fidelity model of the quadrotor where nonlinear dynamics, external disturbances, and parametric uncertainties are taken into account along with the inputs constraints. The simulations are divided into three sections in accordance with chapter 3, where section 4.2 holds the simulation results of the Robust Optimal Adaptive Controller, and section 4.3 presents the simulation results of the Geometric Adaptive Controller. While the last section provides a comparison analysis between the performances of the nonlinear geometric adaptive controller and the robust optimal adaptive controller, where the controllers are tested on their ability to track a desired attitude of a quadrotors.

## 4.2 SIMULATION RESULTS OF ROBUST OPTIMAL ADAPTIVE CONTROLLER

In this section, simulation results are presented to show the efficiency of the robust optimal adaptive control in the presence of external disturbances, parametric uncertainties and input constraints, in which the developed mathematical models in Chapter 2 and control architectures in Chapter 3 are numerically solved.

The quadrotor parameters that were used in simulations are the same used in [37] and they are given in Table 4.1 as follows

**Table 4.1:** Parameters of Quadrotor

| Parameters | Description | Value | Unit |
|---|---|---|---|
| g | Gravity | 9.8° | $m.s^{-2}$ |
| m | Mass | 0.650 | $kg$ |
| $\mathrm{I}_{xx}$ | Inertia on $x$ axis | 7.5e-3 | $kg.m^2$ |
| $\mathrm{I}_{yy}$ | Inertia on $y$ axis | 7.5e-3 | $kg.m^2$ |
| $\mathrm{I}_{zz}$ | Inertia on $z$ axis | 1.3e-2 | $kg.m^2$ |
| $\mathrm{J}_r$ | Rotor Inertia | 6e-5 | $kg.m^2$ |
| L | Arm length | 0.23 | $m$ |
| b | Thrust coefficient | 3.13e-5 | $N.s^2$ |
| d | Drag coefficient | 7.5e-7 | $N.m.s^2$ |

We will use the same quadrotor configuration for all subsequent simulations, including those equipped with nonlinear geometric adaptive controller.

### 4.2.1 Results for Adaptive trajectory tracking

The first simulation deals with parametric uncertainties where we tried to detect the effects of time-varying parameters on tracking performance. The initial conditions are set to:

- Translational position: $[0\ 0\ 0]^T$ .
- Angular rates: $[10\ 10\ 10]^T$.
- The mass and moments of inertia are increased every 10 seconds with 100% of uncertainty.
- The desired trajectory is associated with the sinusoidal reference signal:

  $r_d = 10\sin(0.5t)$

The simulation results are shown in Figure 4.1 and Figure 4.2. It can be seen from Figure 4.1 that the proposed control technique provided very accurate tracking of the attitude reference trajectory in very fast and precisely way. While the estimation of quadrotor parameters response has been shown in Figure 4.2. From the estimation plot, it can be noticed that the proposed method can estimate unknown parameters quickly and accurately, in the same time it fixes them bounded.

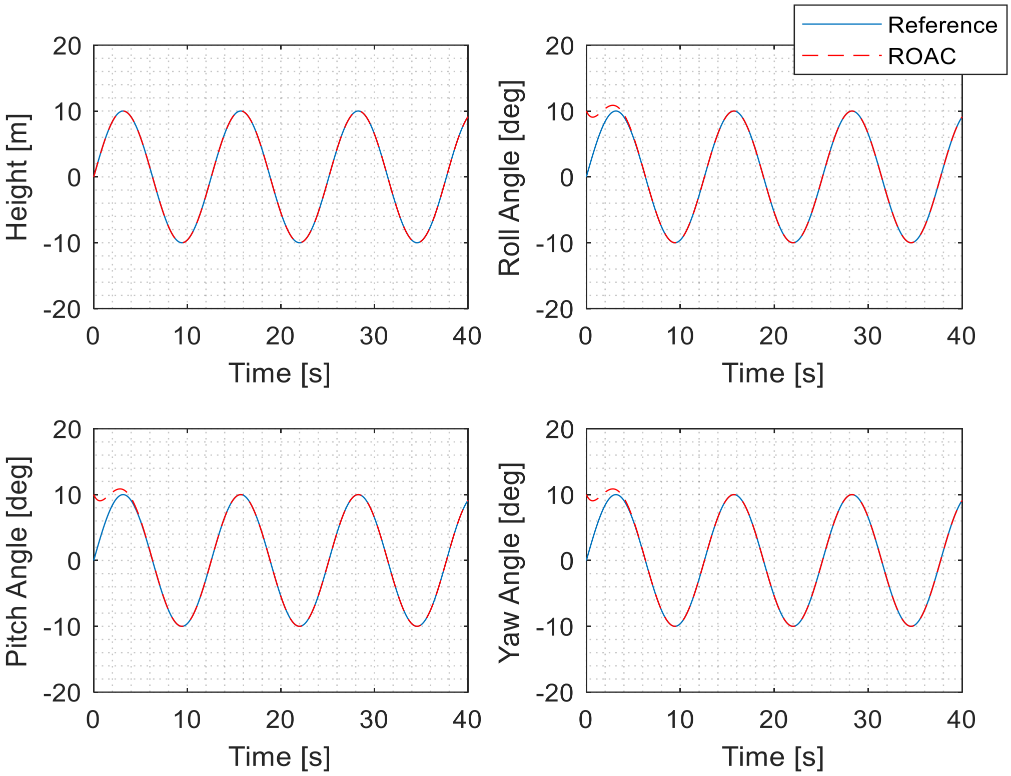

**Figure 4.1**: Altitude and attitude control with a sinusoidal desired output in presence of uncertainty but without external disturbance.

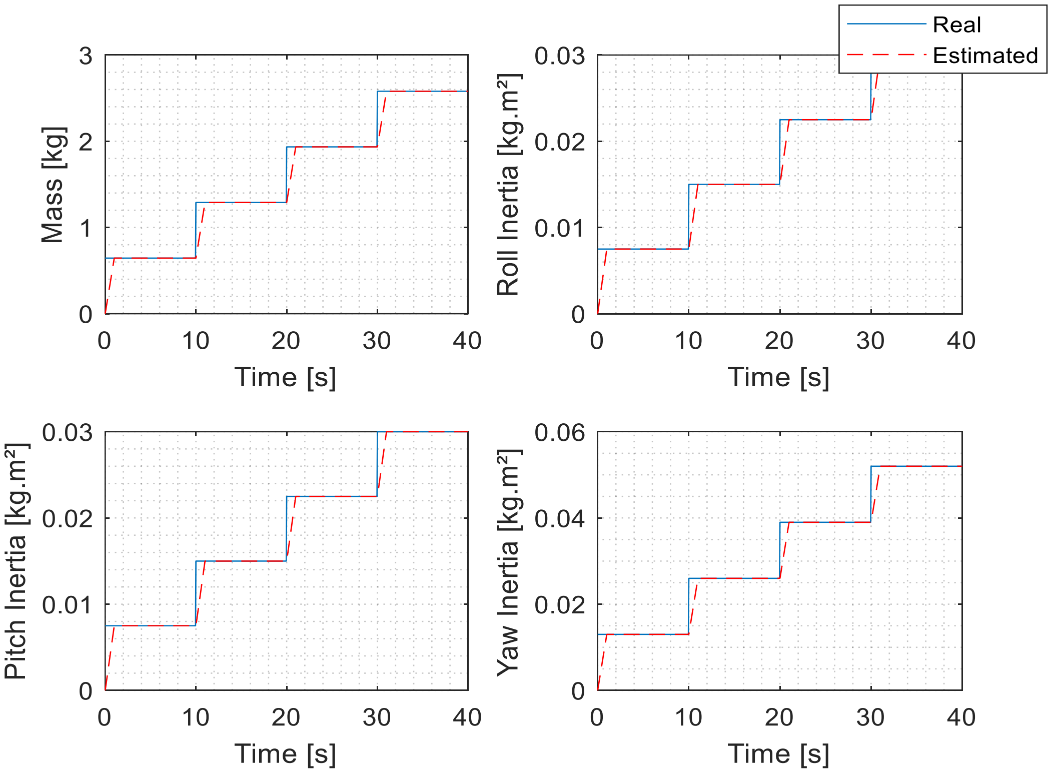

**Figure 4.2**: Estimation of time-varying unknown parameters of quadrotor.

### 4.2.2 Results for trajectory tracking using Nonlinear Disturbance Observer

This subsection evaluates the performance of the nonlinear disturbance observer based controller developed before in order to verify its robustness issue against external disturbance. For the purpose of the simulation, the external disturbances imposed on the system for 7.5 seconds and their vectors are given by the functions:

$$\begin{cases} d_1 = 2\sin(2t+1) \\ d_2 = 0.25\sin(3t+1) \\ d_3 = 0.05\sin(2t+1) \\ d_4 = 0.05\sin(t) \end{cases}$$

The initial value of the exogenous system is taken as:

$$\chi_0 = [2\sin 1\ \ 4\cos 1\ \ 0.25\sin 1\ \ 0.75\cos 1\ \ 0.05\sin 1\ \ 0.1\sin 1\ \ 0\ 0.05]^T$$

The plots of the estimated disturbances are presented in Figure 4.3, where it can be seen that the nonlinear disturbance observer can estimate the unknown disturbances in close vicinity of the real disturbances, while the magnitude of observer gains are the responsible of the convergence rate. For that, a faster convergence depends on large gains without any modification needs on the controller.

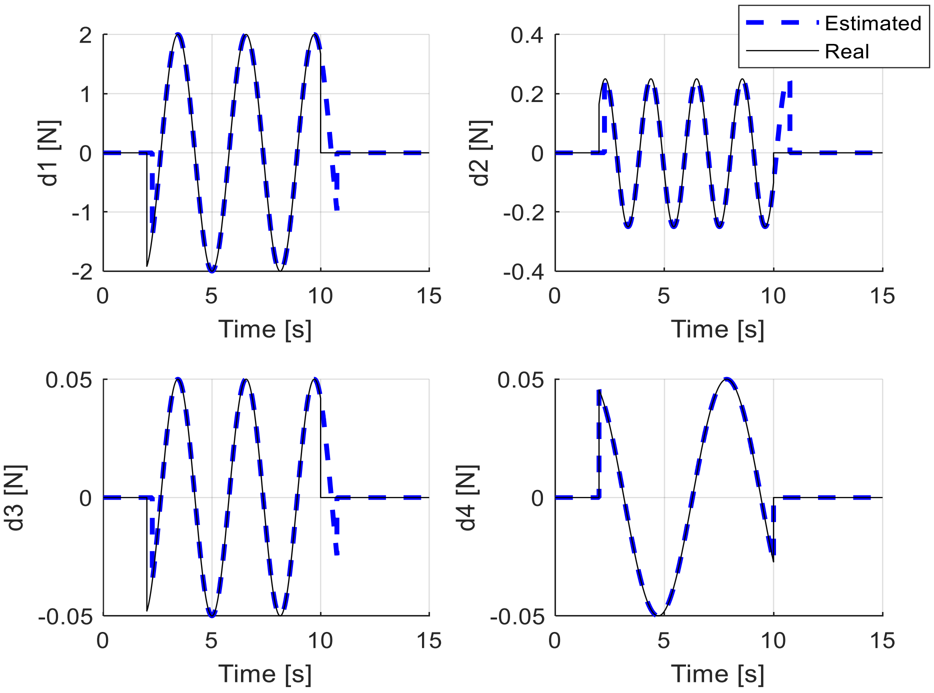

**Figure 4.3:** Disturbance estimated by nonlinear disturbance observers.

Figures 4.4 and 4.5 show the effect of the external disturbances acting on the stabilization of quadrotor during the fly, where they provide a comparison between the stabilization of system with and without using the nonlinear disturbance observer on the altitude, attitude and translational trajectories respectively.

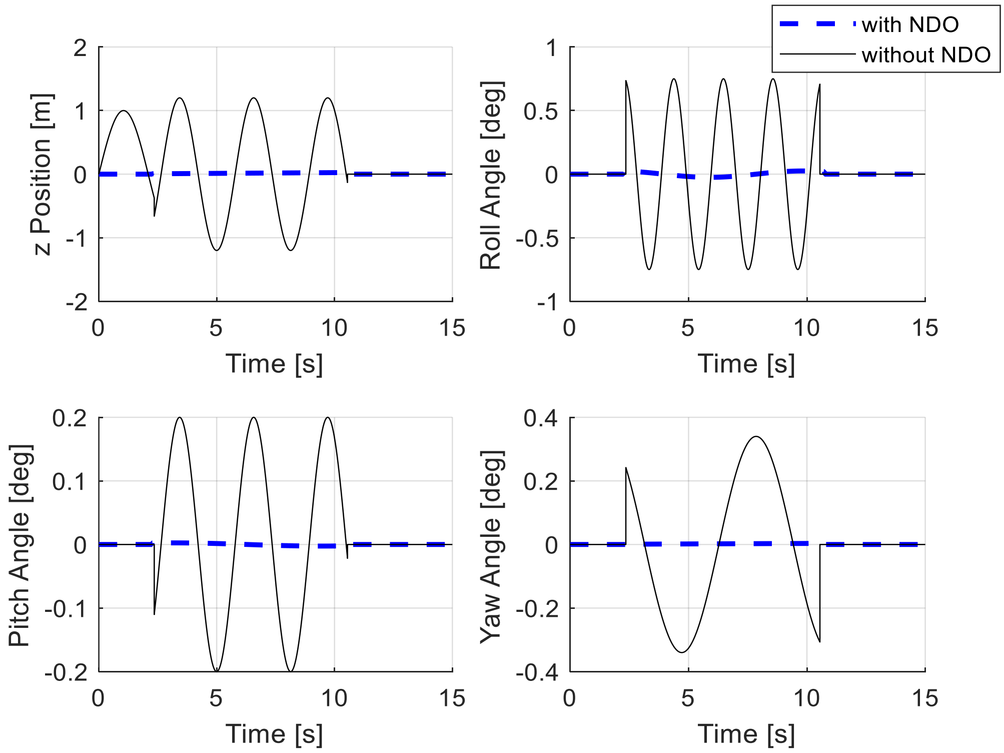

**Figure 4.4:** Altitude and attitude stabilization with NDO

From Figure 4.4, we can see that the external disturbances effect badly on the stability performance of the system with oscillatory response in altitude and attitude, while the integrated NDO with the adaptive controller showed better performance against external disturbances and parametric uncertainties with a guarantee of stable flight.

The effects of external disturbance on quadrotor position are depicted in Figure 4.5. As the translational motion of the quadrotor relies on the attitude angles, the consequences of acquiring the desired bounded and smooth attitude angles are reflected in the position plots in the form of deviation. From position plots, it can be inferred that the proposed composite controller showed better stiffness against the disturbances by retaining the position in the close vicinity of the desired values with less deviation from origin.

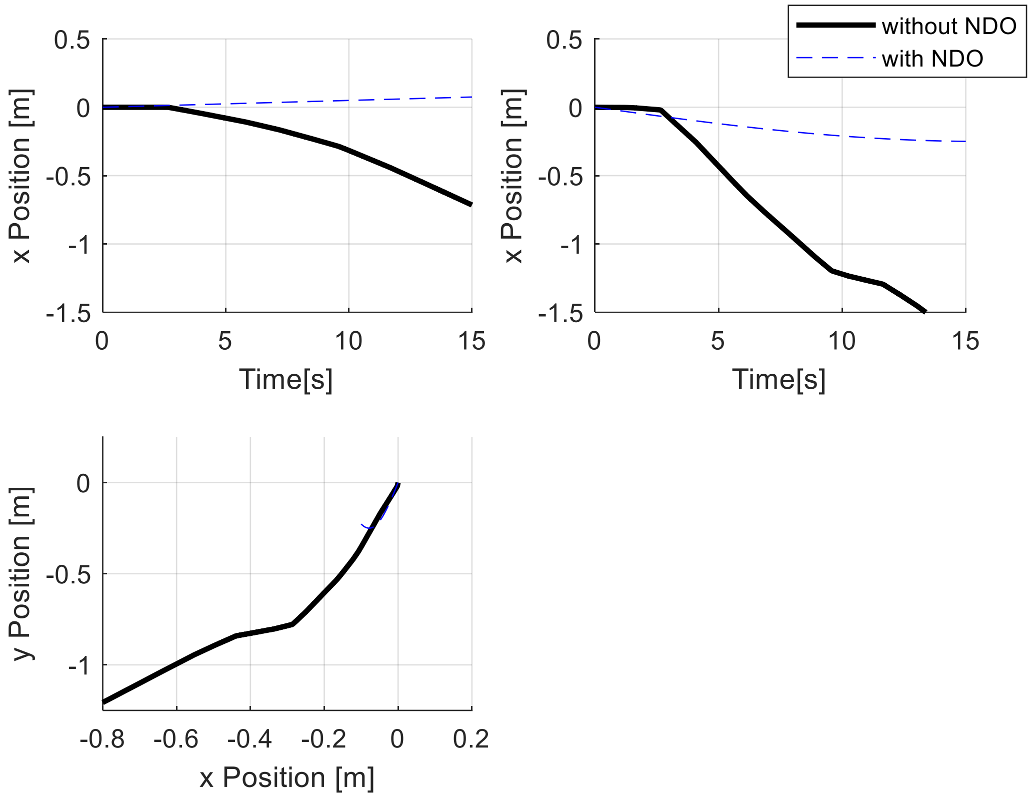

**Figure 4.5**: Position of quadrotor in hovering, while the system is subject to external disturbances.

### 4.2.3 Results of PSO algorithm

In order to reach the control objectives, this subsection we will introduce the Particle Swarm Optimization (PSO) algorithm to our system where the quadrotor should reach desired set point in presence of control input constraints along with parametric uncertainties and external disturbances. The PSO algorithm is utilized to tune the controller by minimizing the presented cost function. The initial conditions are set to: $x_0 = [0\ 0\ 10°\ 0\ 10°\ 0\ 10°\ 0]^T$, while the final set points are 1 and 0 for the final altitude and attitude respectively.

The simulation results of variation of the global best cost versus the Number of Function Evaluation (NFE) in the PSO algorithm is presented in Figure 4.6. While Table 4.2 presents the best solutions founded by the PSO algorithm for different weighting factors.

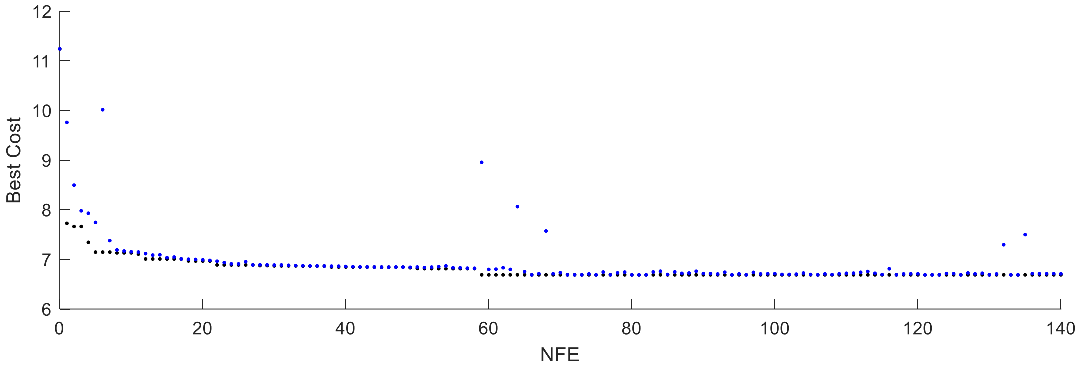

**Figure 4.6**: Global best cost

**Table 4.2**: Best solution obtained by the PSO algorithm with different weighting factor values.

| Weighting Factor | $\Lambda$ | $\Gamma_1$ | $K_D$ | Best Cost |
|---|---|---|---|---|
| a1=a2=1<br>b1=b2=1<br>c1=c2=1 | 1.5364 | 1.2805 | 1.2092 | 6.68853 |
| a1=a2=2<br>b1=b2=1<br>c1=c2=1 | 3.6270 | 2.7571 | 1.3401 | 7.76627 |
| a1=a2=1<br>b1=b2=2<br>c1=c2=1 | 1.9190 | 4.8436 | 1.7065 | 9.70998 |
| a1=a2=1<br>b1=b2=1<br>c1=c2=2 | 7.9660 | 3.9880 | 1.8311 | 10.86427 |

From the Table 4.2, we can see that the best solution for our system is the solution obtained for a1=a2=1, b1=b2=1 and c1=c2=1.

Figures 4.7 and 4.8 provide results of outputs tracking and it was shown that the outputs properly reach desired set points, and the control inputs are continuous and limited as desired.

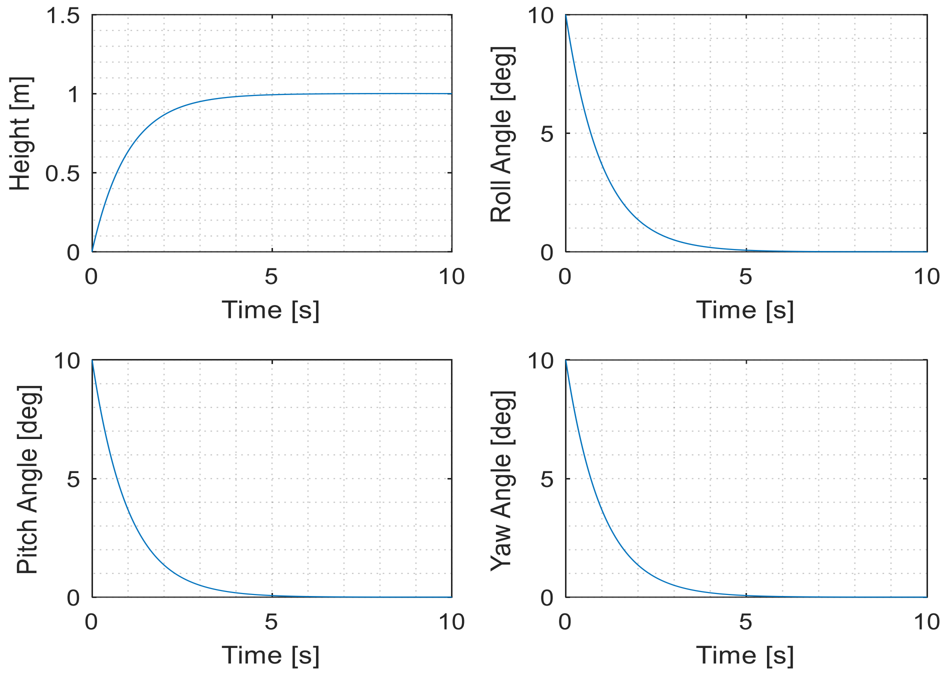

**Figure 4.7:** Simulation results of the set point tracking

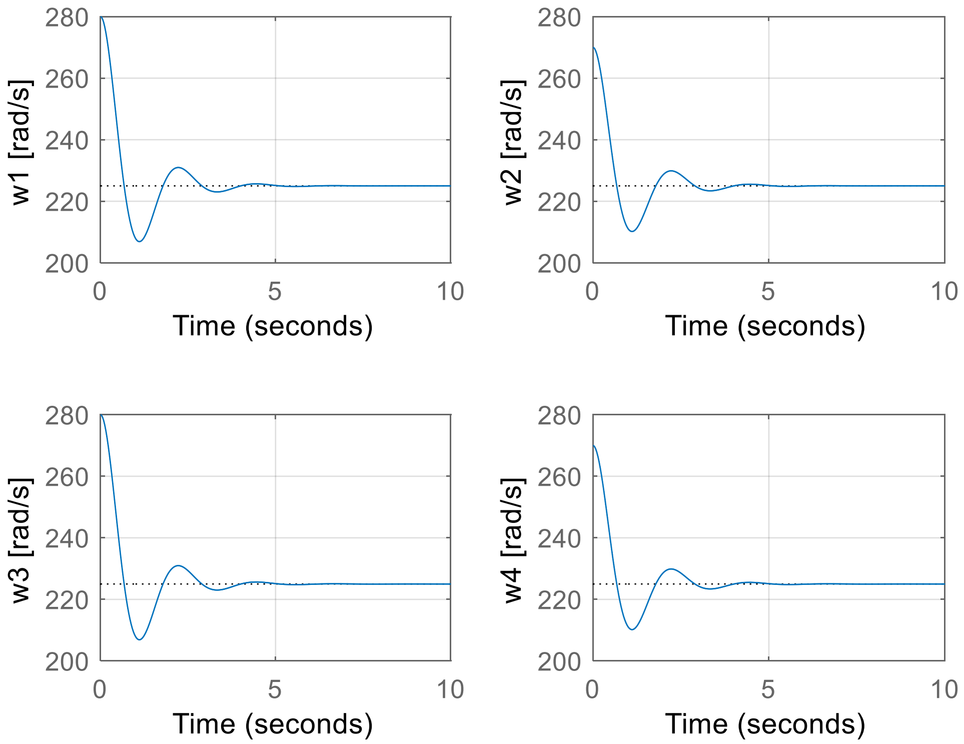

**Figure 4.8:** Angular speed of rotors

## 4.3 SIMULATION RESULTS OF GEOMETRIC ADAPTIVE CONTROLLER

In this section, the simulation of attitude tracking by the geometric adaptive nonlinear control is given to validate the performance of this controller against the external disturbances and inequality constraints.

Where a set of proposed scenarios are presented to test the different control variations. The model parameters for the quadrotor that were used in these simulations are the same as the parameters used with the robust optimal adaptive controller as mentioned before, in order to facilitate the comparison between the two controllers in the next section, while the control system parameters are chosen as:

- $G = diag[0.9,1.1,1.0]$ $\alpha = 15$ $c = 1.0$
- Attitude error gain ($k_R$): 0.4
- Angular velocity error gain ($k_\Omega$): 0.296
- Disturbance error gain ($k_\Delta$): 0.5
- A body fixed sensor: $r = [1,0,0]$
- The multiple inequality constraints are defined in Table 4.3 as:

**Table 4.3**: Multiple inequality constraints

| Constraint Vector ($v$) | Angle ($\theta$) |
|---|---|
| $[0.174, -0.934, -0.034]^T$ | 40° |
| $[0, 0.7071, 0.7071]^T$ | 40° |
| $[-0.853, 0.436, -0.286]^T$ | 40° |
| $[-0.122, -0.140, -0.983]^T$ | 20° |

- $\Delta = [2\sin(2t+1)\ \ 0.25\sin(3t+1)\ \ 0.05\sin(2t+1)]^T$
- The initial conditions are set to:

$R_0 = \exp\left(225° \times \frac{\pi}{180}\hat{e}_3\right)$ $\Omega_0 = 0$

### 4.3.1 Configuration error function visualization

As described in Chapter 3, the stability of the nonlinear geometric controller is evaluated by analyzing the error functions to check whether the desired attitude is asymptotically stabilized. Therefore, we firstly visualize the attitude error function on $SO(3)$ using a spherical coordinate representation in terms of a latitude and longitude. Figures 4.9 to 4.11 provide a visualization of the attitude error function potential in addition to its composites. Where, Figure 4.9. represents the attractive error function $A(R)$, from it can be seen that the desired attitude lies at the minimum of $A(R)$. While Figure 4.10 represents the repulsive error function $B(R)$, where it shows that as the boundary of the constraint is neared, the barrier term increases and this was caused by the state inequality constraints which was incorporated by applying a logarithmic barrier term. The logarithmic barrier quickly decays away from the constraint boundary in addition to the positive constant $\alpha$ which serves to shape the barrier function. As $\alpha$ is increased the impact of $B(R)$ is reduced away from the constraint boundary.

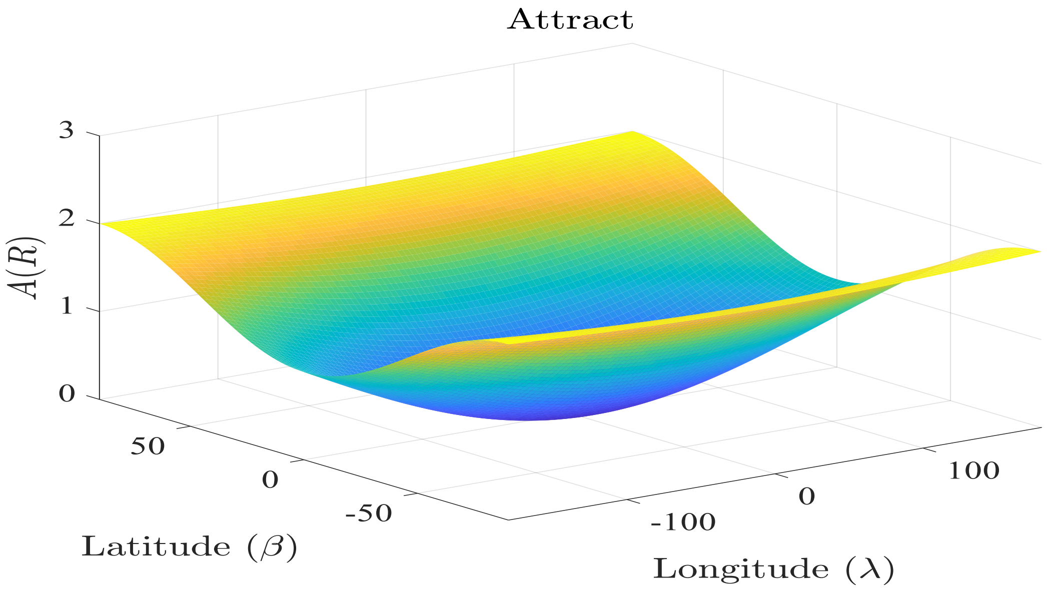

**Figure 4.9**: Attractive $A(R)$ error function visualization

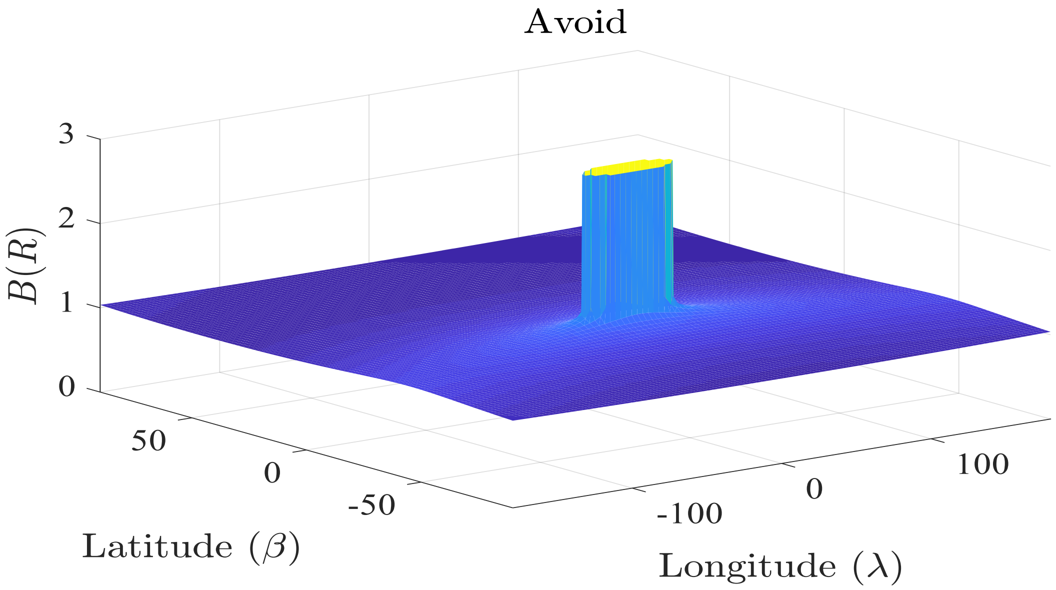

**Figure 4.10:** Repulsive $B(R)$ error function visualization

In order to visualize the attitude error function, we will represent the superposition of the attractive and repulsive functions in Figure 4.11. From this figure we can define our control system such that the attitude trajectory follows the negative gradient of $\Psi$ toward the minimum at $R = R_d$, while avoiding the constrained region.

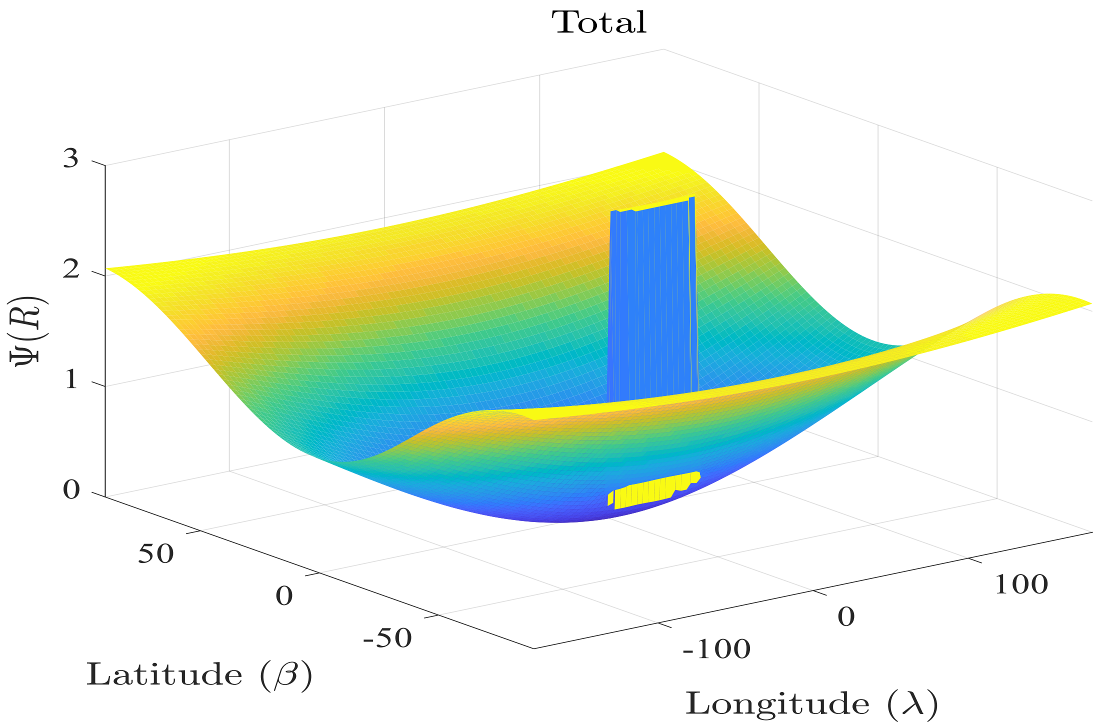

**Figure 4.11:** Configuration error function visualization

### 4.3.2 Results of Attitude control without Disturbance

In the first scenario of simulation for attitude control and stabilization, we assume that there are no disturbances. Where Figures 4.12 and 4.13 show the simulation results without using the adaptive update law.

From Figure 4.12, we can see that the system does not achieve to zero steady state error which means the system is not stable. The same thing for the configuration error function that was presented in Figure 4.13, which couldn’t converge to zero while there exist steady state errors.

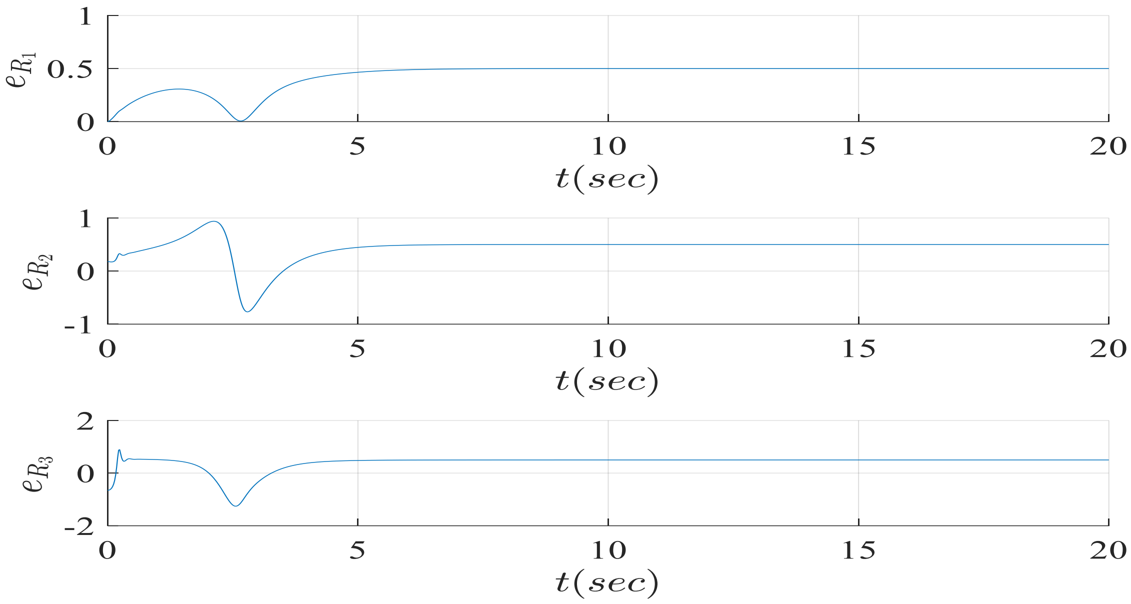

**Figure 4.12:** Attitude error vector $e_R$

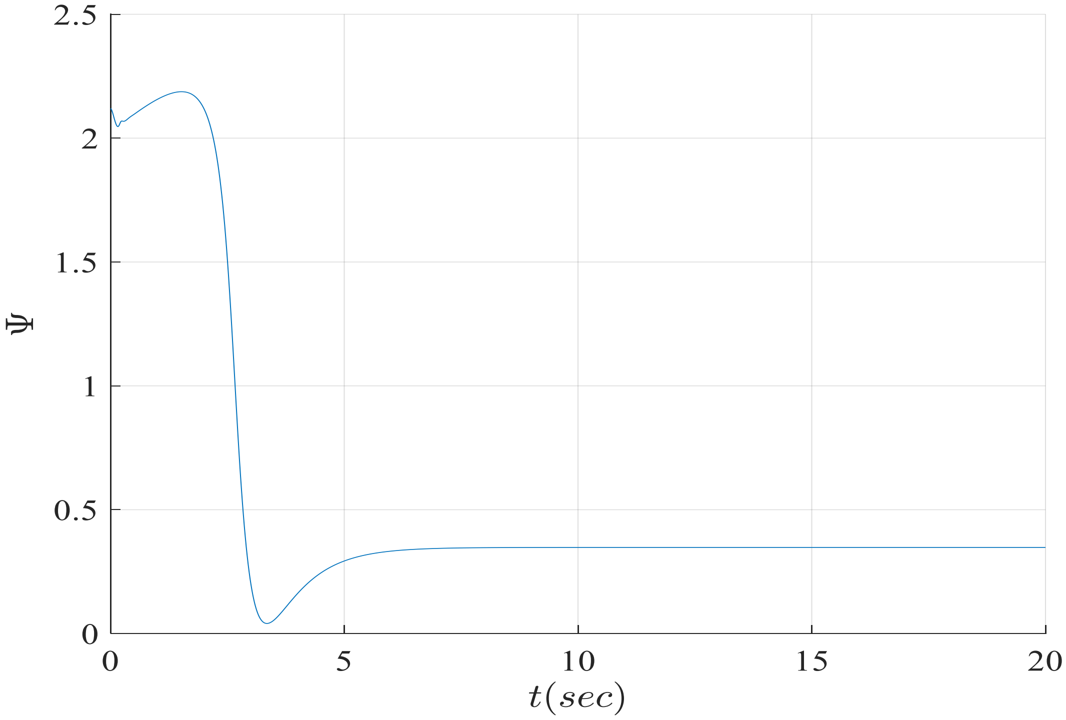

**Figure 4.13:** Configuration Error $\Psi$ without adaptive law

### 4.3.3 Results of Adaptive Attitude Control

In the second scenario of simulation for attitude control, we will use the adaptive update law developed in Chapter 3 in order to handle the uncertain disturbances while avoiding constrained regions. Where the simulation results are presented in Figures 4.14 to 4.17.

Figure 4.14 presents the configuration error function after applying the adaptive update law, where it was shown that the error function converged to zero this time, that means that the adaptive control can asymptotically stabilize the system to a desired attitude while ensuring that state constraints are satisfied. The angle to constraints are presents in Figure 4.15 where we can see that he angle $\arccos(r^T R^T v_i)$ betweene the body fixed sensor and each constraint is satisfied for the entire maneuver .

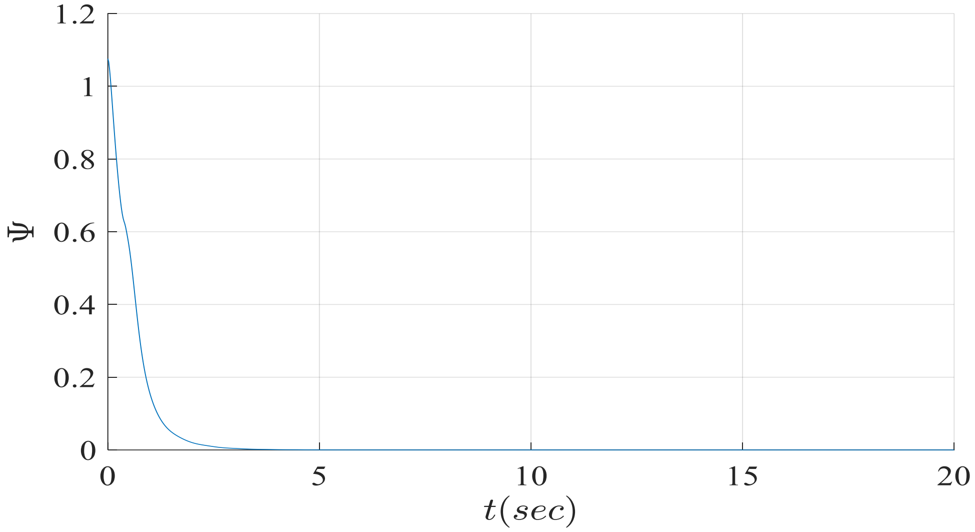

**Figure 4.14:** Configuration Error $\Psi$ with adaptive law

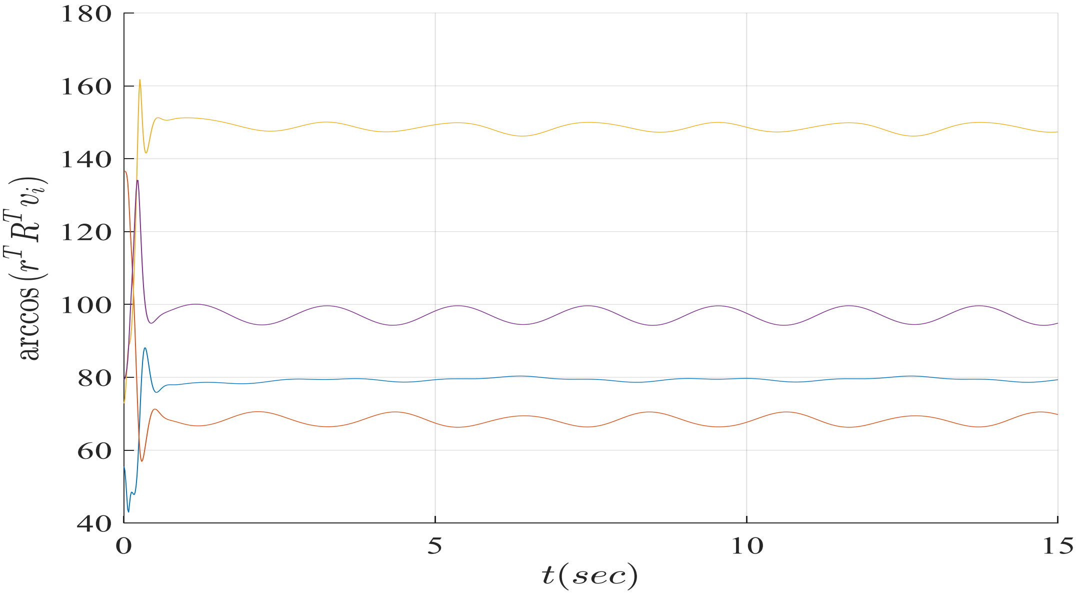

**Figure 4.15:** Angle to constraints

Figure 4.16 shows the efficiency of the adaptive control against unknown disturbances. Where the adaptive control estimates the real disturbance while fixed them bounded and that enables the system to stabilize to a desired attitude.

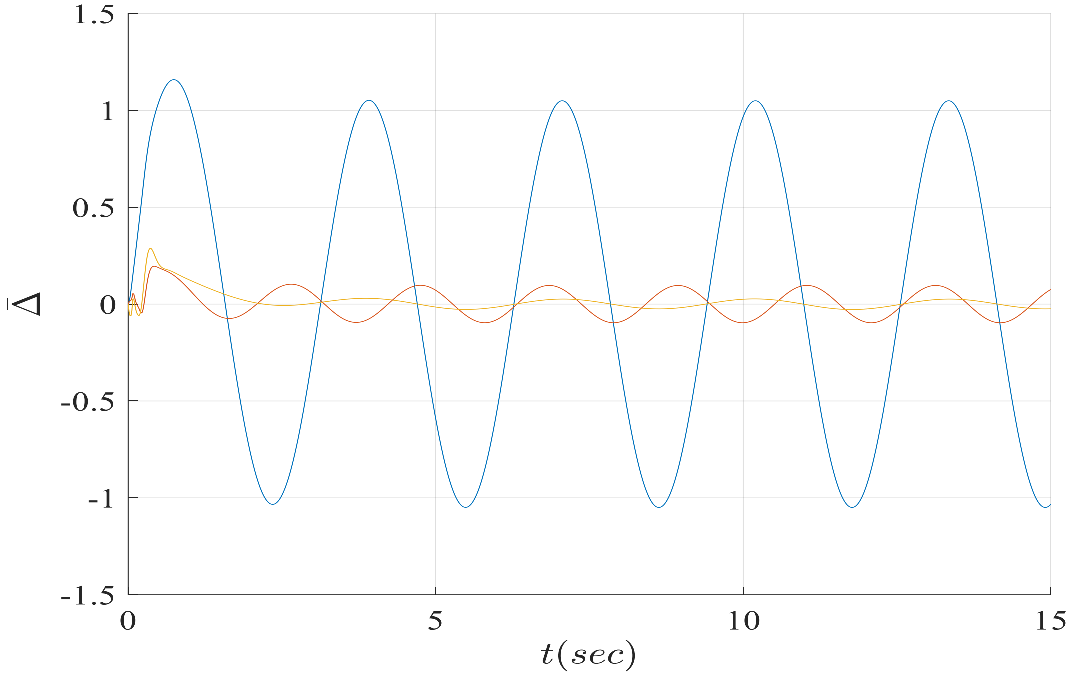

**Figure 4.16:** Disturbance estimate $\bar{\Delta}$

The path of the body fixed sensor in the inertia frame was presentd in Figure 4.17. The system avoids the constrained region illustrated by the circular cone in Figure 4.17 while the initial and the final attitude were represented with a green cirle and a green X respectively, by rotating around the boundary of the constraint. This verifies that the proposed control system exhibits the desired performance.

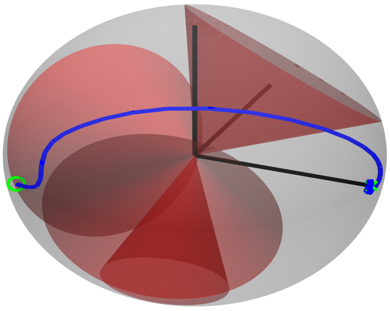

**Figure 4.17:** Attitude trajectory

## 4.4 COMPARISON ANALYSIS

In this subsection, simulation results of Robust optimal adaptive controller will be compared to the Geometric adaptive controller using the same initial and final conditions sets. Where we will compare the tracking errors between the actual and desired trajectory. The simulation results of this comparison is presented in Figure 4.18.

From Figure 4.18, it is observed that the geometric adaptive controller converges faster than the robust optimal adaptive controller towards the 0 for the attitude and 1 for the trajectory. While the robust optimal adaptive controller takes 4s to converge to the desired set points.

For more precisely comparison results between the aforementioned controllers, Table 4.4 presents the indices that allows to a better analysis which are: the response time and the overshoot.

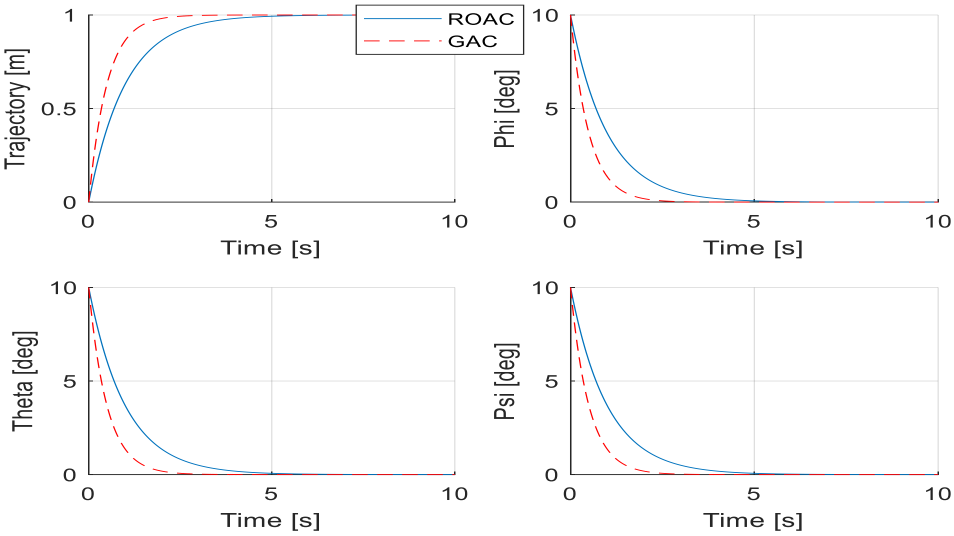

**Figure 4.18:** Comparison between ROAC and GAC controllers

**Table 4.4:** Simulation results comparison

| | Response time | Overshoot | Run time |
|---|---|---|---|
| GAC | 1 s | 0% | 6 s |
| ROAC | 4 s | 0% | 9 s |

## 4.5 CONCLUSION

This chapter presents simulation results for the robust optimal adaptive and geometric adaptive controllers developed and studied in the previous chapter. Where each controller was tested and evaluated via different scenarios of simulation in order to verify its efficiency in trajectory tracking. The first controller provides a good altitude and attitude tracking in the presence of parametric uncertainties using an adaptive controller which estimate the desired parameters, while the integrated nonlinear disturbance observer estimates the fixed and unknown disturbances. Furthermore, in order to deal with input constraints, we used a PSO algorithm. The second controller showed more efficiency results in trajectory tracking while avoiding inequality constraints regions and better performance in estimating the unknown disturbances by fixing them bounded using an adaptive update law.

Finally, we conclude that the nonlinear geometric controller offers greater tracking capability and robustness to external disturbances and inequality constraints. It is also fast to respond to errors and damp out oscillatory responses.

# CONCLUSIONS

Autonomous robotic systems have been developing in an accelerating pace in recent years, what started as an interesting concept, exclusive to only a few institutions, has become a global trend all over the world especially the aerial vehicles namely Drones. The most popular and advantageous aerial vehicle is the Quadrotor UAV because of its VTOL concept especially for specific type of missions/operations. Although quadrotor have many advantageous properties, it has a highly nonlinear, coupled and under-actuated dynamics. Therefore, control of the vehicle is not straightforward and many researchers interested in designing and verifying control methods for quadrotors.

In this thesis, attitude tracking of a quadrotor is obtained by using two independent control strategies called as “Robust Optimal Adaptive Controller” and “Geometric Adaptive Controller”. Throughout the thesis, first, dynamic model of the quadrotor is derived by using Newton’s equations of motion. Then, each control methods are studied and developed in order to get the final control design. Finally, the validation of control systems is obtained by simulations in MATLAB/Code and control systems are compared to each other.

The first controller is addressed to attitude stabilization and tracking problem of quadrotor in the presence of parametric uncertainties, external disturbances and input constraints. Based on numerical results, robust optimal adaptive controller has the ability to stabilize nonlinear dynamic system of quadrotor, force the states to follow desired reference signals, and find optimal solution for the tracking problem. Where the developed adaptive update law deals with parametric uncertainties by estimating the real parameters in very accurate way. While to reduce the effect of input saturation of the uncertain quadrotor model, the PSO algorithm showed that it can achieved to the optimal adaptive controller parameters quickly and efficiently. The nonlinear disturbance observer integrated with the optimal adaptive controller provides a high accurate attenuation to the external disturbances.

The second controller is a developed geometric adaptive control system which incorporates state inequality constraints on SO(3). The presented control system is developed directly on SO(3) and it avoids singularities and ambiguities that are inherent to attitude parameterizations. In order to avoid the constrained regions, the attitude configuration error is augmented with a barrier function. While to cancel the effects of uncertainties, an adaptive control law is proposed. The numerical results showed that it is straightforward to incorporate an arbitrary amount of large constraints. Another feature of this control is that it is computed autonomously on-board the UAV.

According to the comparison results, it could be stated that, the geometric adaptive controller could track the attitude trajectory with high accuracy compared to robust optimal adaptive controller. Where the presented geometric adaptive controller is simple, efficient and ideal for hardware implementation on embedded systems.

Although simulations give good and motivating results, to achieve a complete validation, controllers have to be verified by real time experiments. Therefore, experimental validation in a real environment of obtained control algorithms could be performed as a future work. In addition, dynamic model of the quadrotor can be detailed to obtain more realistic and accurate results and performance of controllers could be increased by making some modifications.

# BIBLIOGRAPHY

[1] "The Drone market size 2020-2025 5 key take aways", Drone Industry Insights.com.

[2] S. Bouabdallah, P. Murrieri, and R. Siegwart, "Design and control of an indoor micro quadrotor," in Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on, vol. 5. IEEE, 2004, pp. 4393–4398.

[3] S. Bouabdallah, A. Noth, and R. Siegwart, "Pid vs lq control techniques applied to an indoor micro quadrotor," in Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on, vol. 3. IEEE, 2004, pp. 2451–2456.

[4] S. Bouabdallah, P. Murrieri, and R. Siegwart, "Towards autonomous indoor micro vtol," Autonomous robots, vol. 18, no. 2, pp. 171–183, 2005.

[5] E. Paiva, J. Soto, J. Salinas, and W. Ipanaqué, "Modeling, simulation and implementation of a modified PID controller for stabilizing a quadcopter" in Proc. IEEE Int. Conf. Automatica (ICA-ACCA), Curico, Chile, Oct. 2016, pp. 1_6.

[6] M. Valenti, B. Bethke, G. Fiore, J. P. How, and E. Feron, "Indoor multivehicle flight testbed for fault detection, isolation, and recovery," in Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, Keystone, CO, vol. 63, 2006, p. 64

[7] J. P. How, B. Bethke, A. Frank, D. Dale, and J. Vian, "Real-time indoor autonomous vehicle test environment," IEEE control systems, vol. 28, no. 2, pp. 51–64, 2008.

[8] D. Lee, H. J. Kim, and S. Sastry, "Feedback linearization vs. adaptive sliding mode control for a quadrotor helicopter," International Journal of control, Automation and systems, vol. 7, no. 3, pp. 419–428, 2009.

[9] S. Bouabdallah and R. Siegwart, "Full control of a quadrotor," in 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 2007, pp. 153–158.

[10] S. Bouabdallah and R. Siegwart, "Backstepping and sliding-mode techniques applied to an indoor micro quadrotor," in Proceedings of the 2005 IEEE international conference on robotics and automation. IEEE, 2005, pp. 2247–2252.

[11] N. Fethalla, M. Saad, H. Michalska, and J. Ghommam, "Robust observer-based dynamic sliding mode controller for a quadrotor UAV," IEEE Access, vol. 6, pp. 45846_45859, 2018.

[12] H. Mo and G. Farid, "Nonlinear and adaptive intelligent control techniques for quadrotor UAV_A survey," Asian J. Control, vol. 21, no. 3, pp. 1_20, 2019.

[13] 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, Jul. 2013.

[14] M. Mohammadi and A. M. Shahri, “Modelling and decentralized adaptive tracking control of a quadrotor UAV,” in Proc. 1st RSI/ISM Int. Conf. Robot. Mechatronics (ICRoM), Feb. 2013, pp. 293_300.

[15] M. Mohammadi and A. M. Shahri, “Adaptive nonlinear stabilization control for a quadrotor UAV: Theory, simulation and experimentation,” J. Intell. Robotic Syst., vol. 72, no. 1, pp. 105_122, 2013.

[16] N.A. Chaturvedi, A.K. Sanyal, and N.H. McClamroch. “Rigid-body attitude control”. IEEE Control Systems Magazine, 31(3):30–51, Jun. 2011. ISSN 1066-033X.

[17] Navabi, M., Mirzaei, H. “Robust optimal adaptive control for quadrotors”, Latin American Journal of Solids and Structures 14 (2017) 1040-1063.

[18] Shankar Kulumani, Christopher Poole, and Taeyoung Lee “Geometric Adaptive Control of Attitude Dynamics on SO(3) with State Inequality Constraints” (2016).

[19] Taeyoung Lee, “Robust Adaptive Geometric Tracking Controls on SO(3) with an Application to the Attitude Dynamics of a Quadrotor UAV”, CMMI-1029551 (2011).

[20] K. Nonami, M. Kartidjo, K. J. Yoon and A. Budiyono, Autonomous Control Systems and Vehicles Intelligent Unmanned Systems, Springer, Japan, 2013.

[21] R. Austin, “Unmanned Aircraft Systems UAVS Design, Development and Deployment,” John Wiley & Sons Ltd, Great Britain, 2010.

[22] R. K. Barnhart, S. B. Hottman, D. M. Marshall, E. Shappee, “Introduction to Unmanned Aircraft Systems,” CRC Press Taylor & Francis Group, USA, 2012.

[23] D. McLean, “Automatic Flight Control Systems,” Prentice Hall International (UK) Ltd, Cambridge, Great Britain, 1990.

[24] B. Etkin, “Dynamics of Atmospheric Flight, Dover Publications,” New York, USA, 2005.

[25] M.D. Shuster. “A survey of attitude representations”. The Journal of the aeronautical science, 41(4):439–517, October-December 1993.

[26] Hardik Parwana, Mangal Kothari “Quaternions and Attitude Representation” (2017).

[27] D. Mellinger and V. Kumar, “Minimum snap trajectory generation and control for quadrotors,” in Proceedings of the IEEE International Conference on Robotics and Automation (ICRA ’11), pp. 2520–2525, IEEE, Shanghai, China, May 2011.

[28] Jinho Kim, Student Member, IEEE, S. Andrew Gadsden, Member, IEEE, and Stephen A. Wilkerson, Member, IEEE "A Comprehensive Survey of Control Strategies for Autonomous Quadrotors" (2020)

[29] Madani & Benallegue, "Backstepping Control for a Quadrotor Helicopter" 1-4244-0259 X/06/S20.00 2006 IEEE

[30] V. Utkin, "Variable structure systems with sliding modes," IEEE Transactions on Automatic control, vol. 22, no. 2, pp. 212–222, 1977.

[31] Beard, Randal W., McLain, Timothy W "Small Unmanned Aircraft: Theory and Practice" (February 26, 2012) Hardcover.

[32] Kwakernaak, 1993 "Robust control and $H_\infty$-optimization—Tutorial paper".

[33] D. Ito, J. Georgie, J.Valasek, and D.T. Ward. "Reentry vehicle flight control design guidelines: Dynamic inversion". Technical Report NASA/TP-2002–210771, NASA, 2002.

[34] Alberto Isidori "Nonlinear Control Systems II" , Springer, London, 1999.

[35] P. Hughes, "Spacecraft Attitude Dynamics". Dover Publications, 2004.

[36] F. Bullo and A. D. Lewis, "Geometric Control of Mechanical Systems", ser. Texts in Applied Mathematics. New York-Heidelberg-Berlin: Springer Verlag, 2004, vol. 49.

[37] Bouabdallah and Siegwart (2006) "Towards Intelligent Miniature Flying Robots".