PIVOT DETERMINATION, MOMENT-PRIORITIZED CONTROL ALLOCATION AND FAR DISTANCE ESTIMATION FOR ROTARY-WING UNMANNED AERIAL VEHICLES

by

Erol Kayataş

B.S., Mechanical Engineering, Boğaziçi University, 2020

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Mechanical Engineering
Boğaziçi University

2023

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my advisor, Assoc. Prof. Murat Çelik, for his invaluable assistance and direction throughout my thesis. His enthusiasm and vast knowledge were vital to my motivation during this research. In addition to his academic support, he will be my role model for the rest of my life because of the value and understanding he gives his students.

I express my gratitude to Prof. Fazıl Önder Sönmez and Assoc. Prof. Mustafa Şengör, esteemed members of my thesis committee, for their valuable time dedicated to reviewing and providing corrections during the preparation of my dissertation.

I would like to thank The Scientific and Technology Research Council of Turkey (TUBITAK) and Bogazici University Scientific Research Projects Support Fund (BAP) for financial support under the Project entitled BAP-19002.

I also want to thank my parents and family. Despite knowing nothing about my studies, they significantly contributed to my development. The most outstanding merit they gave me is the desire to raise a cheerful, wise child who will benefit humanity.

Finally, I want to express my gratitude to all my friends, especially Eray and Erkan, who gave me value and strength with their support.

ABSTRACT

PIVOT DETERMINATION, MOMENT-PRIORITIZED CONTROL ALLOCATION AND FAR DISTANCE ESTIMATION FOR ROTARY-WING UNMANNED AERIAL VEHICLES

Uncrewed Aerial Vehicles (UAVs) are becoming increasingly popular due to their low system requirements, production costs, and adaptability for various tasks. While fixed-wing UAVs are known for their capacity to carry large loads and travel great distances at high speeds, rotary-wing UAVs are distinguished by their ability to hover in midair and operate through tight spaces with quick maneuvers. Compared to their fixed-wing counterparts in military operations, their inferior utility has hindered the development of rotary-wing uncrewed aerial vehicles. However, the preparation of flying cars for transportation and an aspiration to combine rotary-wing and fixed-wing aircraft have paved the way for the technical development of these platforms. In this thesis, the rotary wing UAV technique improved by correcting faulty theories and by assigning mathematical models to phenomena that have not been previously addressed. It has been exhibited that the center of rotation is always at the zero force point since the pivot does not accelerate; prioritizing the attitude against altitude during actuator saturations improves the flight by eliminating the underlying problem, which is moment needs at large Euler angles, and a monocular camera without motion freedom is sufficient for depth estimation because the platform itself can also conduct gimbal rotations. The zero force point and the rotation center alter if a displacement restriction is exerted on the platform. A control allocation algorithm is as successful as the priority it gives angular motion. The lateral velocity of a fixed point due to camera rotation is directly proportional to depth.

ÖZET

DÖNER KANATLI İNSANSIZ HAVA ARAÇLARINDA PİVOT TESPİTİ, MOMENT ÖNCELİKLENDİRMELİ KONTROL PAYLAŞTIRIMI VE UZAK MESAFE KESTİRİMİ

İnsansız Hava Araçları (İHA); düşük sistem gereksinimleri, ucuz üretim maaliyetleri ve farklı görevlere elverişli olmalarıyla çok yaygınlaşmışlardır. Yüksek hızlarda uzak bölgelere erişebilme ve ağır yükler taşıyabilme sabit kanatlı İHA'ların karakteristik özelliğiyken; havada askıda kalabilme, dar ortamlarda ani manevralar gösterebilme de döner kanatlı İHA'lara özgü nitelikler olmuştur. Askeri eylemlerde sabit kanatlı muadilleri kadar faydalı olamamaları bu hava araçlarının gelişimini yavaşlatmıştır. Ne var ki; hava taşımacılığına uçan arabaların hazırlanması, döner kanatlı ve sabit kanatlı hava araçlarının birleştirilmek istenmesi ise bu ürünlerin günümüzde önünü açmıştır. Bu tezde de; farklı alanlardan devşirilen yanlış teorilerin düzeltilmesiyle ve önceden değinilmemiş kavramlara matematiksel model tahsis edilmesiyle döner kanat tekniği geliştirilmiştir. Görülmüştür ki; dönme merkezi her zaman sıfır kuvvet noktasıdır çünkü pivot noktası ivmelenmemektedir; eyleyici limit aşımlarında irtifadan önce duruşa öncelik vermek, bu durumun temel nedenini ortadan kaldırarak uçuşu iyileştirecektir; sıfır serbestlik dereceli sabit bir kamera ile uzak mesafe tespiti yapılabilir çünkü platform gimbal dönüşleri gerçekleştirebilir. Hareketli sisteme hareket kısıtı tanımlanırsa sıfır kuvvet noktası da, dönme merkezi de değişir. Bir kontrol paylaştırım algoritması momentlere öncelik verdiği kadar başarılıdır. Normalde sabit bir noktanın kamera dönüşüyle kazandığı yanal hız, derinlikle doğru orantılıdır.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS iii
ABSTRACT iv
ÖZET v
LIST OF FIGURES viii
LIST OF TABLES xiii
LIST OF SYMBOLS xiv
LIST OF ACRONYMS/ABBREVIATIONS xvii
1. INTRODUCTION 1
2. TECHNICAL BACKGROUND 6
2.1. Coordinate Systems 6
2.2. Rotation Philosophy 8
2.3. Forming Rotation Matrices 11
2.4. Algebraic Properties of Rotation Matrices 12
2.5. Quaternions and Hamiltonian Rotation Technique 14
2.6. Transition between Euler Angles and Quaternions 19
2.7. Mathematical Relationship between Angular Rates 20
2.8. Propeller and Thrust Mechanics 22
2.9. Aircraft Sensors 24
2.9.1. Inertial Measurement Units 25
2.9.2. Barometers 27
2.9.3. Global Positioning 29
2.9.4. Magnetometers 32
2.9.5. Range Finders 36
2.10. Estimation 38
2.11. Building the Equations of Motions for RW-UAVs 42
2.11.1. Newton's Second Law of Motion 42
2.11.2. Derivative of a Vector 43
2.11.3. Equations of Motion for Translational Subsystem 43

2.11.4. Equations of Motions for Rotational Subsystem 45
2.12. Flight Controller 47
3. ADVANCEMENTS 52
3.1. The Center of Rotation and Its Accurate Determination 52
3.2. Moment Prioritized Control Allocation 58
3.3. Estimating Far Distances using a Fixed Monocular Camera 67
4. CONCLUSION 72
4.1. FUTURE WORK 73
REFERENCES 74
APPENDIX A: A 84

LIST OF FIGURES

Figure 1.1. Drones are used in various applications using distinct modules [3]. 1

Figure 1.2. Various forms of uncrewed aerial vehicles, including hot air balloons, fixed-wing and rotary-wing aircraft, used in battles [8]. . . . 2

Figure 1.3. Turkish tactical armed fixed UAV Kızılelma [9]. 3

Figure 1.4. The use of rotary wing drones in many applications includes field scanning, local observation, and farm spraying [3]. 3

Figure 1.5. Turkish VTOL DIHA manufactured by BAYKAR [14]. 4

Figure 2.1. On a common drone, three coordinate systems are needed to specify its movements. 7

Figure 2.2. An aircraft inside rotation rings. 9

Figure 2.3. The directional convention of a rotation matrix. 9

Figure 2.4. The process for constructing a rotation matrix regarding evolving frames. 10

Figure 2.5. The procedure by which a rotation matrix is constructed across a set of fixed frames. 10

Figure 2.6. The columns of a rotation matrix have significant physical interpretations. 13

Figure 2.7. The gimbal lock situation where the $\boldsymbol{X}$ and $\boldsymbol{Z}$ axes overlap each other. 14

Figure 2.8. Complex numbers in polar coordinates are illustrated. Multiplicating a unit number by any other complex number results in the rotation of θ. 15

Figure 2.9. Complex space Cartesian coordinates representation. 17

Figure 2.10. Translations of origins and rotations of axes for the mentioned multiplications. 17

Figure 2.11. Schematic for quaternion creation and usage from angular rates. . 18

Figure 2.12. Forces and moments on the infinitesimal (dr) blade section. 22

Figure 2.13. Thrust and torque fit of 12×4.0 propeller with respect to RPM. Slopes of both figures equal to k_T and k_Q. 24

Figure 2.14. A large-sized accelerometer (left) and gyroscope (right) [34]. . . . 25

Figure 2.15. Basic schematic of MEMS accelerometer [34]. 26

Figure 2.16. Vibrating mass in MEMS gyroscope. 26

Figure 2.17. Optical Gyroscopes in operation [34]. 27

Figure 2.18. Conventional mercury barometer [40]. 28

Figure 2.19. Aneroid pressure sensor schema [40]. 28

Figure 2.20. A cross-sectional cut of MEMS capacitance pressure sensor [40]. . 29

Figure 2.21. Global positioning constellations' logos. From left to right: (a) GPS (USA), (b) GLONASS (the Soviet Union), (c) BEIDOU (China), (d) GALILEO (European Union) [46]. 30

Figure 2.22. Earth's magnetic field, poles, and declination angle [50]. These values vary by region. 32

Figure 2.23. Ideal magnetic sphere at left (a), translated magnetic field sphere due to hard iron effect at the middle (b) and shape-disturbed magnetic field sphere due to soft iron effect at right (c) [51]. 33

Figure 2.24. Calibrated magnetic field sphere over hard and soft iron effects. Normally, it is both shape-distorted and translated. Then, it is centered at the origin and transformed into a sphere [51]. 34

Figure 2.25. Working principle of a characteristic magnetometer. 35

Figure 2.26. A LiDAR-scanned depth image [55]. 36

Figure 2.27. A sonar-scanned planar depth image. It is especially used in autonomous vehicles [56]. 37

Figure 2.28. Low pass filter bode diagram showing amplitude gain in all frequency ranges considering logarithmic scale. 40

Figure 2.29. High pass filter bode diagram showing amplitude gain in all frequency ranges considering logarithmic scale. 41

Figure 2.30. Complimentary filter including low pass filter to denoise accelerometer data and high pass filter to denoise gyroscope data. 41

Figure 2.31. ArduCopter algorithm motor orientation and body-fixed coordinate system on the chassis with Γ arm length [62]. 46

Figure 2.32. Translational controller in ArduCopter algorithm. 48

Figure 2.33. Rotational Controller in ArduCopter algorithm. 49

Figure 2.34. Kp tuning procedure. Low and high Kp values have characteristic behaviors. 50

Figure 2.35. Complete descriptions of how controller gains affect step responses of a controller. 51

Figure 3.1. Free body diagram of a rotating stick in space. There are no displacement restrictions. It features concentrated masses on each side for easier formulations. 55

Figure 3.2. A drone with a lower gravity center and vertical displacement restriction due to flight controller response. The supposed scenario is shown here. The body is rotated around the center of mass. . . 56

Figure 3.3. A drone with a lower gravity center and vertical displacement restriction due to flight controller response. The carried-out scenario is demonstrated here. The body is rotated around the restriction point. 57

Figure 3.4. Kinetic diagram of a typical quadcopter. Virtual control signals are depicted separately. 59

Figure 3.5. The comparison of three different control allocations that will be tested. 61

Figure 3.6. The altitude command track performance comparison for three different control allocations. 65

Figure 3.7. The comparison of Φ angle command track performance for three different control allocations. 66

Figure 3.8. The comparison of Ψ angle command track performance for three different control allocations. 67

Figure 3.9. Distance calculation over triangulation using the dual camera. . . 68

Figure 3.10. Scenario for detecting the depth of a distant object. The algorithm does not have a prerequisite regarding the shape of the observed target. 70

LIST OF TABLES

Table 2.1. Representation of a quaternion in vectorial and Euler form. 16

Table 2.2. Representation of a quaternion's conjugate. 16

Table 2.3. Newton's Second Law of Motion in non-rotary coordinate systems. 43

Table 3.1. Simulation parameters. 64

LIST OF SYMBOLS

a_f	Cutoff frequency for filters in frequency domain
A_i	Magnetic field reading in i^{th} axis with respect to VC-NED-CS
B_i	Magnetic field reading in i^{th} axis with respect to BF-CS
$\boldsymbol{B}$	Magnetic field
c	Chord length
c_f	Cutoff constant for filters in time domain
C_D	Airfoil drag coefficient
C_L	Airfoil lift coefficient
$\boldsymbol{D}$	Drag on the specified object
$e^{\hat{\imath}\theta}$	Euler representation of a unit complex number
$\hat{e_i}$	i^{th} eigenvector
$\boldsymbol{E}$	Electric field
$\boldsymbol{F}$	Context-based force
$\boldsymbol{F_e}$	Force exerted on an electron
$G(s)$	Transfer function in the frequency domain
$\boldsymbol{g}$	Gravity
$\boldsymbol{G_W}$	Weight
h	Height of the given object
$\boldsymbol{H_G}$	Angular momentum vector
I	Inertia tensor
I_{ii}	Moments of inertia in $\hat{i}$ axis
I_{ij}	Products of inertia regarding $\hat{i}$-$\hat{j}$ axes
$\hat{i}$,$\hat{j}$,$\hat{k}$	Unit vectors in real and complex space
k_F	Propeller lift coefficient
k_Q	Propeller torque coefficient
K	System Performance Matrix
K_i	i^{th} Kalman observer gain
Kd	Derivative gain in controller

Ki	Integrator gain in controller
Kp	Proportional gain in controller
L_i	Lift on the specific object
$\boldsymbol{L}$	Rolling moment
m	Mass
$\boldsymbol{M}$	Pitching moment
$\boldsymbol{N}$	Yawing moment
$\boldsymbol{N_G}$	Geographic north
$\boldsymbol{N_M}$	Magnetic north
$\boldsymbol{p}$	Body angular rate vector in $\boldsymbol{X}$ axis
$\boldsymbol{P}$	Position vector of any object
P_a	Atmospheric pressure
$\boldsymbol{q}$	Body angular rate vector in $\boldsymbol{Y}$ axis
$\boldsymbol{Q}$	Torque on the specified object
q_i	i^{th} element of a unit quaternion
q_p	Particle charge
$\dot{q}$	A unit quaternion rate
q_u	A unit quaternion
$\bar{q}_u$	A conjugate of a unit quaternion
$\boldsymbol{r}$	Body angular rate vector in $\boldsymbol{Z}$ axis
R_1^0	Rotation matrix from BF-CS to VC-NED-CS
R_0^1	Rotation matrix from VC-NED-CS to BF-CS
R_i	Rotation around $\hat{i}$ axis
R_{ij}	Element of rotation matrix in i^{th} row j^{th} column
S	Occupied image area by the tracked object
T_P	Propellers' total thrust force
U	Virtual control matrix
U_i	i^{th} virtual control signal
U_{Mt}	Point M horizontal velocity in $pixel/s$ due to translation
U_{Mr}	Point M horizontal velocity in $pixel/s$ due to rotation
$\boldsymbol{V}$	An arbitrary direction vector

$\boldsymbol{V_{Dt}}$	Drone horizontal velocity in m/s due to translation
$\boldsymbol{W}$	Relative wind velocity
z	A complex number with real and imaginary parts
α	Angle of attack
β	Declination angle between geographic and magnetic north
Γ	Chassis arm length
Θ	Euler Angle in $\boldsymbol{Y}$ axis (Pitch) or a rotation about given line
μ	Correction factor that transforms $pixel/s$ speeds to m/s
$\boldsymbol{\nu}$	Translational velocity of the specified object
ξ_i	i^{th} sensor reading
ρ	Air density
$\boldsymbol{\varrho}$	Angular acceleration vector
Φ	Euler angle in $\boldsymbol{X}$ axis (Roll)
$\hat{\chi}_i$	i^{th} state estimation
Ψ	Euler angle in $\boldsymbol{Z}$ axis (Yaw)
$\boldsymbol{\omega_B}$	Body angular rates
ω_E	Euler angular rates
Ω	Propeller rotational velocity
Ω_{max}	Max allowable propeller velocity
Ω_{min}	Min allowable propeller velocity

LIST OF ACRONYMS/ABBREVIATIONS

2D	2 Dimensional
3D	3 Dimensional
4D	4 Dimensional
AP	Autopilot
BDS	BeiDou Navigation Satellite System
BF-CS	Body Fixed Coordinate System
CCW	Counter Clockwise
CG	Center of gravity
CW	Clockwise
dB	decibel
DCA	Direct Control Allocation
DCM	Direction Cosine Matrix
DIHA	in Turkish “Dikey IHA”
DOF	Degree of Freedom
ESC	Electronic Speed Controller
Fe-Ni	Ferrous-Nickel
FPV	First-Person View
FW-AC	Fixed Wing Aircraft
FW-UAVs	Fixed-Wing Uncrewed Aerial Vehicles
GLONASS	Global Navigation Satellite Systems
GNS	Global Navigation Systems
GNSS	Global Navigation Satellite Systems
GPS	Global Positioning Systems
IMU	Inertial Measurement Unit
I-NED-CS	Inertial North-East-Down Coordinate System
I-NED-RF	Inertial North-East-Down Reference Frame
IRNSS	Indian Regional Navigation Satellite Systems
KPS	Korean Positioning System

LiDAR	Light Detection and Ranging
MEMS	Micro Electronic Mechanical Systems
PID	Proportional Integral Derivative
PWM	Pulse Width Modulation
QZSS	Quasi-Zenith Satellite System
RNSS	Regional Navigation Satellite Systems
RPM	Revolution Per Minute
RPY	Roll Pitch and Yaw
RW-AC	Rotary Wing Aircraft
RW-UAVs	Rotary-Wing Uncrewed Aerial Vehicles
SBAS	Satellite-Based Augmentation Systems
UAVs	Uncrewed Aerial Vehicles
VC-NED-CS	Vehicle Carried North-East-Down Coordinate System
VTOL	Vertical Take-Off and Landing
WLS	Weighted Least Squares
ZFP	Zero Force Point

1. INTRODUCTION

Rotary Wing Uncrewed Aerial Vehicles (RW-UAVs) are popular flying platforms serving various fields, including entertainment, military, and medical. They can transport passengers and payloads like ammunition, food, medicine, and vaccines. In military operations, drones conduct strikes and complete duties [1]. These vehicles are very advantageous since they are simple to construct and inexpensive; also, because they are uncrewed, any damage would not impact humans. Additionally, to make drones suited for the intended activities, one may add helpful modules like cameras, radars, grapples, and weaponry, as illustrated in Figure 1.1. These platforms range in size from a few millimeters to tens of meters, and depending on the energy source, they may hover for up to several hours [2].



Figure 1.1. Drones are used in various applications using distinct modules [3].

The first uncrewed aerial vehicles, hot air balloons, debuted in Annonay, France, in 1783. It was produced to support the tourism industry within civil aviation. The first status is attributed to these balloons, notwithstanding ongoing arguments over the degree of human agency in their operation [4]. Subsequently, in the year 1849, Austrian officer Franz von Uchatius affixed explosive devices onto those platforms. The endeavor above was ineffective due to insufficient guidance and limited capacity for control technique [5]. Numerous nations used comparable military gadgets for combat operations until the culmination of the global conflicts, specifically by 1940. The progression of these vehicles may be traced back to their first development as guided missiles in 1917, followed by their subsequent transformation into fixed-wing

aircraft (FW-AC) in 1935, and more recently, the emergence of rotary-wing aircraft (RW-AC) in 2010 [6]. Figure 1.2 depicts three separate sorts of military flying vehicles used in warfare. The continuous production of military uncrewed aerial vehicles has persisted. However, as the global socio-cultural landscape has evolved, new examples of these devices have emerged in different sectors [7]. The following are the most prevalent uncrewed aerial vehicles that are extensively utilized and help humankind, along with their benefits and drawbacks.

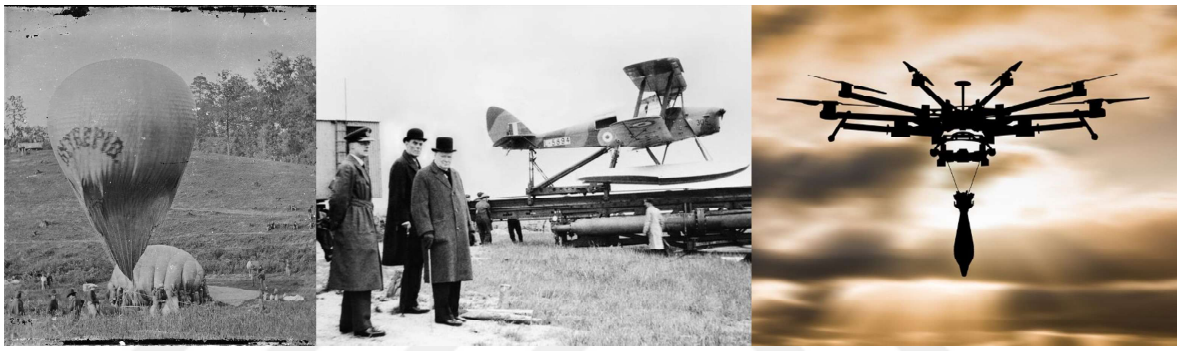


Figure 1.2. Various forms of uncrewed aerial vehicles, including hot air balloons, fixed-wing and rotary-wing aircraft, used in battles [8].

Fixed-wing uncrewed aerial vehicles (FW-UAVs) use the aerodynamic geometries of their fuselages and wings to conduct flight. Once a specific air velocity is attained, the unique configuration of the airfoil causes the airflow to be redirected in a downward direction, resulting in the generation of an upward force known as lift [4]. Figure 1.3 shows this particular example of aero geometry. Fixed-wing aircraft use engines, often internal combustion, to generate a traction force and specified relative wind. Compared to rotary-wing aircraft, these models exhibit far lower energy consumption [2]. Despite the incredible difficulty of their structural mechanics and control procedures, fixed-wing aircraft are the most extensively utilized UAVs. The fact that they can reach longer distances at more economical costs in human transportation and that war technologies are more open to development will make these aircraft the number one choice in the future, too.



Figure 1.3. Turkish tactical armed fixed UAV Kızılelma [9].

Rotary-wing uncrewed aerial vehicles (RW-UAVs) generate aeropropulsive forces by rapidly spinning their propellers at high revolutions per minute (RPM). To maintain a hovering position in air, these vehicles do not need linear velocities [10]. As mentioned earlier, the characteristic fundamentally and favorably differentiates rotary-wing drones from fixed-wing aircraft. Drones are the only viable choices when position hold is required, particularly in confined spaces [11]. Rotary-wing uncrewed aerial vehicles exhibit exceptional suitability for tasks requiring swift and agile maneuvers, including scanning forest interiors, indoor flights, and extended periods of localized field observations. Examples of rotary wing uncrewed aerial vehicles doing those specific functions are shown in Figure 1.4.

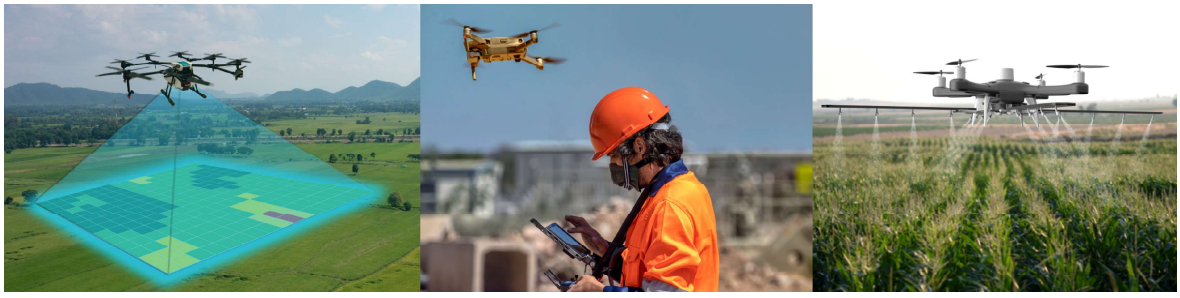


Figure 1.4. The use of rotary wing drones in many applications includes field scanning, local observation, and farm spraying [3].

Vertical Take-Off and Landing (VTOL) aircraft may be conceptualized as a hybridization of rotary and fixed-wing platforms. During the vertical takeoff and landing phases, these devices exhibit characteristics similar to rotary-wing drones [12]. During cruise flight, the behavior of these aircraft resembles that of fixed-wing aircraft, allowing them to attain greater linear speeds and power efficiency. Despite their potential, vertical takeoff and landing aircraft have encountered manufacturing and operational control challenges, resulting in limited adoption. However, it is worth noting that VTOLs have significant promise as the primary aerial vehicles anticipated for future use [13]. They can operate across extensive distances and perform well in confined area operations. The illustration in Figure 1.5 shows a Turkish VTOL DIHA that was recently designed.



Figure 1.5. Turkish VTOL DIHA manufactured by BAYKAR [14].

The uncrewed aerial vehicle family is expanding with new members via the collaborative efforts of people, commercial companies, and public enterprises. Those new arrivals include flying cars, first-person view (FPV) racing drones, and crews trained explicitly for kamikaze missions [15]. As the demands expected from these platforms get more intricate, their techniques need to undergo corresponding advancements. In the context of flying cars, for example, improved battery technology, flight control strategies, security measures, social awareness, and legal infrastructure must be established before people are transported [16]. The following three methods have been developed to improve the dynamic behavior of rotary wing uncrewed aerial vehicles, which is the objective of this thesis.

- The center of rotation and its accurate determination (Section 3.1)
- Control allocation when motors reach their physical limits (Section 3.2)
- Estimating far distances using a fixed monocular camera (Section 3.3)

The following chapter provides a technical overview of uncrewed aerial vehicles with rotary wings. While establishing this technical foundation, claims that are incomprehensible or require correction are highlighted. In the third chapter, these emphatic methods are discussed individually and presented with new solutions. Literature research is provided first in each section, followed by demonstrations of novel concepts and simulation results. A theoretical model was created to determine the center of rotation accurately. Changing motion restrictions showed how the center of rotation displaces. Control allocation with moment prioritization was simulated in SIMULINK. Different flight scenarios were conducted to verify the proposition. The theoretical infrastructure for distance/depth detection with a single fixed camera was created and presented. Concluding the findings, the thesis is finalized in the fourth chapter.

2. TECHNICAL BACKGROUND

Drones possess remarkable versatility, necessitating proficiency in several subdisciplines to analyze their dynamic model and controller [6]. Some of those concepts that need to be mastered include coordinate systems, their transformations, the aerodynamic behavior of propellers, the working principles, and estimation methods of sensors [5]. A comprehensive explanation of these fundamentals is provided before presenting the RW-UAV dynamic model and controller. Subsequently, the equations of motion are derived to establish a dynamic model for a drone, including three equations for the translational and three equations for the rotating subsystem. In the end, a cascade Proportional-Integral-Derivative (PID) controller derived from the ArduCopter algorithm is shown. In Section 2.9, where the sensors are explained, it is emphasized that the existing techniques for distance estimation are insufficient. In Section 2.11, where the equations of motion are derived, it is highlighted that the center of rotation has been detected incorrectly. In Section 2.12, where the flight controller is introduced, it is underlined that the existing control allocation techniques during actuator limit exceedances hinder the physics behind the issue.

2.1. Coordinate Systems

In essence, a minimum of three coordinate systems (Figure 2.1) is sufficient to depict the movements of uncrewed aerial vehicles completely: Inertial North-East-Down Coordinate System (I-NED-CS), Vehicle Carried North-East-Down Coordinate System (VC-NED-CS), and Body Fixed Coordinate System (BF-CS) [5]. While other coordinate sets, such as the Geodetic, Geocentric, and Earth-Centered Earth-Fixed Coordinate Systems, can be helpful when experiencing large displacements, the three coordinate systems listed above are adequate to describe the movements of drones during local surveillance (less than 5 km) [13]. However, irrespective of the specific coordinate configuration, the reference frame in which the observer is situated is the Inertial North-East-Down Reference Frame (I-NED-RF).

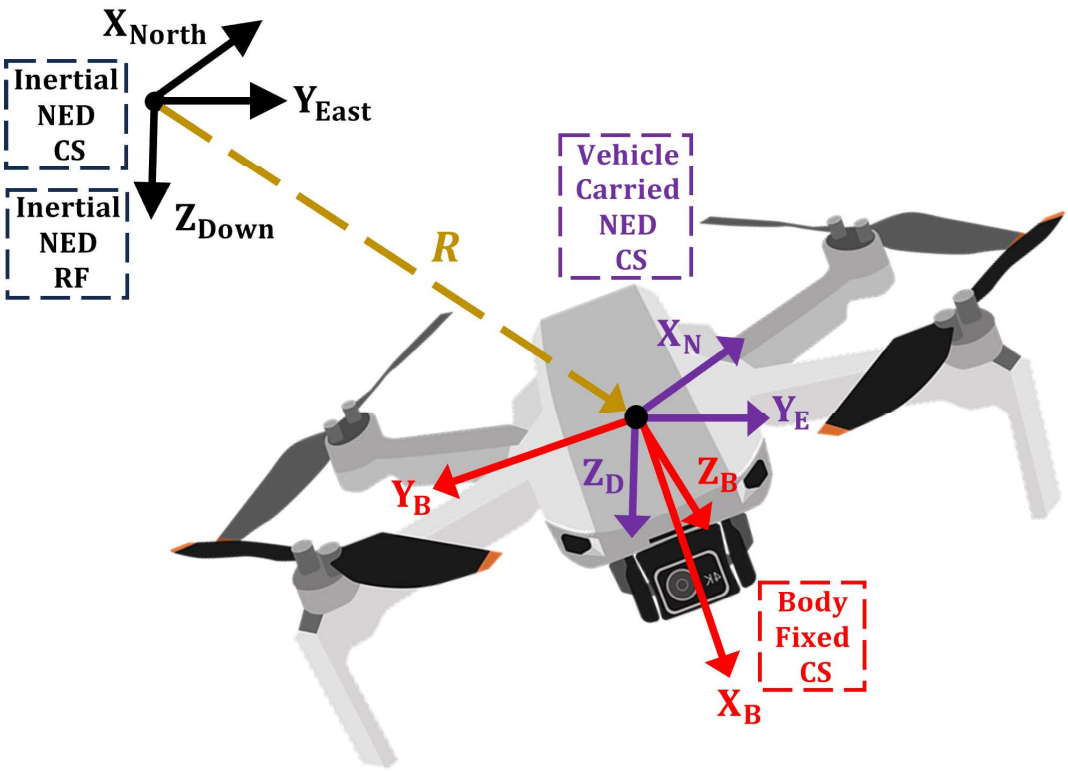


Figure 2.1. On a common drone, three coordinate systems are needed to specify its movements.

Inertial NED Coordinate System (I-NED-CS) ($X_{North}Y_{East}Z_{Down}$) is a frameset whose origin is at the point where the drone takes off (i.e., home position). The $\boldsymbol{X_{North}}$, $\boldsymbol{Y_{East}}$, and $\boldsymbol{Z_{Down}}$ axes point to North, East, and Down, respectively. This coordinate set is essential for route planning and provides traceable data about the vehicle's position concerning a home point [4].

Vehicle-Carried NED Coordinate System (VC-NED-CS) ($X_N Y_E Z_D$) shares the same direction as the I-NED-CS, but its origin always translates with the drone's center of mass [17]. This set of coordinates helps express the drone's kinematic parameters (translation and rotation; position, velocity, and acceleration) because one can project a vector from the BF-CS to the VC-NED-CS by simply applying a rotation without transformation [18]. If this axis set was not used, a $4x4$ transformation matrix would be used instead of the $3x3$ rotation matrix, and the processing load would increase.

Lastly, the Body Fixed Coordinate System (BF-CS) ($X_BY_BZ_B$) is placed at the drone's center of mass. $\boldsymbol{X_B}$, $\boldsymbol{Y_B}$, and $\boldsymbol{Z_B}$ axes point to the nose, right side, and down of the vehicle's fuselage. Because inertial sensors such as gyroscope, accelerometer, magnetometer, barometer, and visual odometry make measurements using it as a reference, the BF-CS is the most significant of the defined coordinate sets [6]. Mapping a vector from BF-CS to VC-NED-CS requires a $3x3$ rotation matrix (for three-dimensional (3D) motion) [1].

2.2. Rotation Philosophy

Rotation matrices (i.e., Direction Cosine Matrix (DCM)) are transformation operators that facilitate the projection of vectors from one coordinate system to another, specifically from/to BF-CS to/from the VC-NED-CS in this context [19]. Rotation orders, whether the successive rotations occur around the evolving or stationary frames, and rotation magnitudes are the critical concepts while building these matrices [6]. By changing the order of successive rotations (ZYX, XYZ, ZXZ, ZYZ ...), these matrices can be formed in many different types [5]. The rotation around the $\boldsymbol{Z}$ axis is known as yawing, around the $\boldsymbol{Y}$ axis is known as pitching, and rolling is known as rotation about the $\boldsymbol{X}$ axis as depicted in Figure 2.2.

The first issue is which approach/order/pattern most effectively meets dynamic expectations/analysis of selected physical systems (such as drones, aircraft, submarines, and industrial robotic arms) [1]. For example, the aircraft's tilt compensation is significant in evaluating the yaw angle from the magnetometer readings. Now that this compensation can be handled by withdrawing the pitch and roll rotations but not yaw rotation, $\boldsymbol{Z}$ rotation cannot be realized after $\boldsymbol{X}$ and $\boldsymbol{Y}$ turns [2]. In an aircraft, pitch rotation is performed more than roll rotation. Pitching is another name for nose-down and nose-up motions, and if this action occurs after the rolling, it will go out of its description due to directional change [15]. Therefore, the rotation in $\boldsymbol{Y}$ must occur before the one in $\boldsymbol{X}$. In this regard, Tait-Bryan Angles ($Z-Y-X$) is the most convenient rotation pattern for flying objects while working with evolving frames [4].

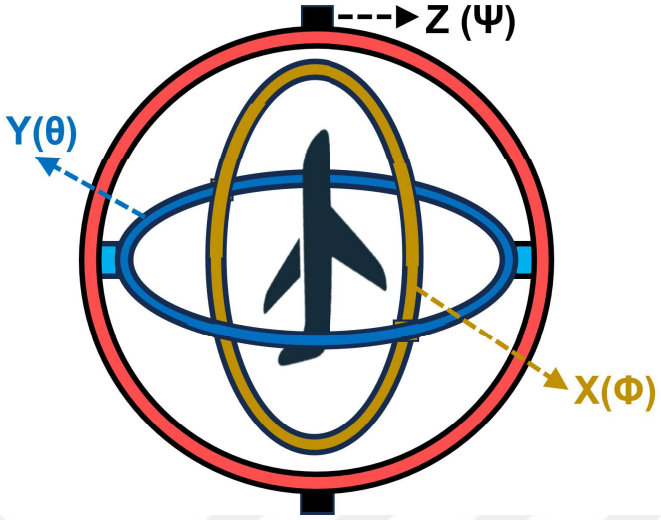


Figure 2.2. An aircraft inside rotation rings.

The second issue to be addressed is whether the rotations occur concerning the evolving or stationary frames. The rotation angles are called roll, pitch, and yaw (RPY) if all successive rotations arise relative to the first frame (VC-NED-CS in this context). If the other is true, the rotation angles are referred to as Euler Angles [13]. The rotation matrices that result from choosing between Euler Angles and roll pitch yaw angles never change. At this point, it should be noted that the rotation magnitudes inevitably vary so that both procedures yield the same rotation matrix [15]. The building procedures for these rotation matrices are provided herein through graphical representations. For more compact equations, 0 is used for the VC-NED-CS, while 1 is used for the BF-CS representation as demonstrated in Figure 2.3. Depending on which axis set the equations of motion are written in, it is of great importance to master the rotation matrix in both directions.



Figure 2.3. The directional convention of a rotation matrix.

If successive rotations are to occur according to an evolving set of coordinates (Figure 2.4), each new term is appended to the right of the multiplication.

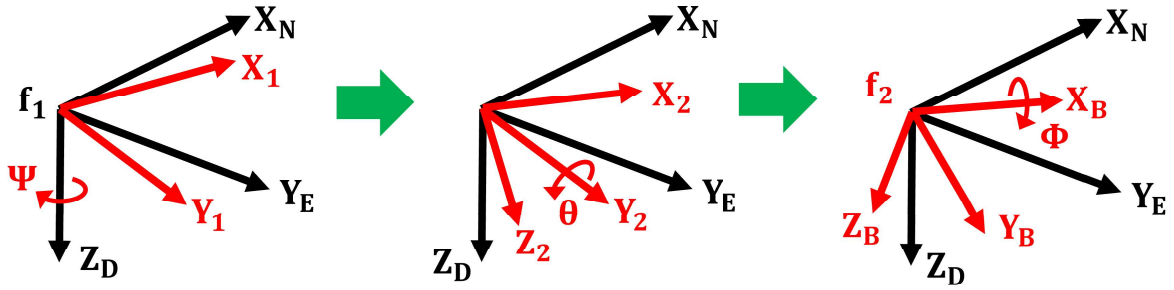


Figure 2.4. The process for constructing a rotation matrix regarding evolving frames.

Multiplying consecutive rotations produces rotation matrices as

$$R_1^0 = R_z(\Psi)R_y(\Theta)R_x(\Phi). \tag{2.1}$$

If successive rotations are to occur according to a fixed set of coordinates (Figure 2.5), each new term is appended to the left of the multiplication.

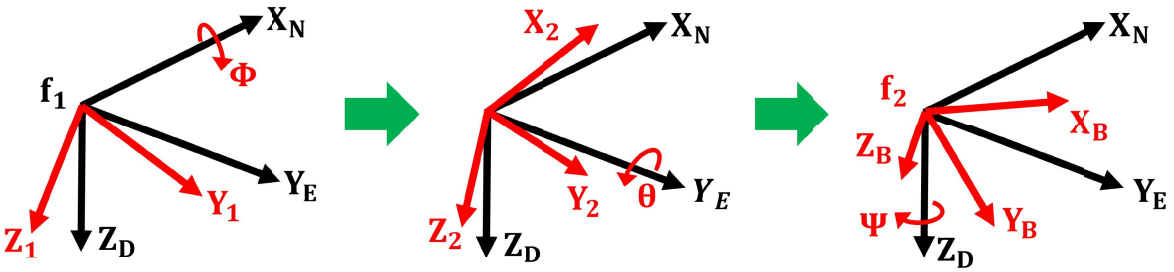


Figure 2.5. The procedure by which a rotation matrix is constructed across a set of fixed frames.

Multiplying consecutive rotations produces rotation matrices as

$$R_1^0 = R_z(\Psi)R_y(\Theta)R_x(\Phi), \tag{2.2}$$

shown in Equation (2.2). In present aerial vehicles, determining Z - Y - X angle amplitudes is facilitated by gyroscopes, specifically via quaternion rates. These gyroscopes are affixed to the drones' chassis and can measure angular velocities concerning evolving frames. Thus, Euler angles are seen as most suitable for constructing rotation matrices in the context of aerial vehicles [4].

2.3. Forming Rotation Matrices

Rotation around the $\boldsymbol{Z}$ axis is called "Baş-Yönelme" in Turkish with an angular displacement of Ψ whose limits are $\pm\pi$. Therefore, aerial vehicles can conduct full rotation about their Z axis [11]. The specific rotation matrix about the $\boldsymbol{Z}$ axis is

$$R_z(\Psi) = \begin{bmatrix} \cos\Psi & -\sin\Psi & 0 \\ \sin\Psi & \cos\Psi & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{2.3}$$

as illustrated in the Equation (2.3). The second rotation around the $\boldsymbol{Y_2}$ axis, often called pitching and known as "Dikilme" in Turkish, must be performed at an angle denoted as Θ whose limits are $\pm\pi/2$. Yet, aerial vehicles should not experience these limits ($\pm\pi/6$ at most) [11]. The specific rotation matrix about the $\boldsymbol{Y_2}$ axis is

$$R_y(\Theta) = \begin{bmatrix} \cos\Theta & 0 & \sin\Theta \\ 0 & 1 & 0 \\ -\sin\Theta & 0 & \cos\Theta \end{bmatrix}, \tag{2.4}$$

as depicted in the Equation (2.4). Final rotation around the $\boldsymbol{X_B}$ axis, commonly known as rolling and called "Yatış" in Turkish, must be executed at an angle denoted by Φ whose limits are $\pm\pi/2$. Yet, aerial vehicles should not experience these limits ($\pm\pi/6$ at most) [11]. The specific rotation matrix about the $\boldsymbol{X_B}$ axis is

$$R_x(\Phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\Phi & -\sin\Phi \\ 0 & \sin\Phi & \cos\Phi \end{bmatrix}, \tag{2.5}$$

as denoted in the Equation (2.5). A rotation matrix can be obtained through

$$R_1^0 = \begin{bmatrix} \cos\Psi & -\sin\Psi & 0 \\ \sin\Psi & \cos\Psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\Theta & 0 & \sin\Theta \\ 0 & 1 & 0 \\ -\sin\Theta & 0 & \cos\Theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\Phi & -\sin\Phi \\ 0 & \sin\Phi & \cos\Phi \end{bmatrix}, \tag{2.6}$$

by successfully applying the $Z-Y-X$ rotations [20]. The resultant rotation matrix R_1^0 that maps vectors from BF-CS to VC-NED-CS can be denoted as

$$R_1^0 = \begin{bmatrix} \cos\Theta\cos\Psi & \cos\Psi\sin\Phi\sin\Theta - \cos\Phi\sin\Psi & \sin\Phi\sin\Psi + \cos\Phi\cos\Psi\sin\Theta \\ \cos\Theta\sin\Psi & \cos\Psi\cos\Phi + \sin\Psi\sin\Phi\sin\Theta & \cos\Phi\sin\Psi\sin\Theta - \sin\Phi\cos\Psi \\ -\sin\Theta & \cos\Theta\sin\Phi & \cos\Phi\cos\Theta \end{bmatrix}. \tag{2.7}$$

In Section 2.4, the transpose of this rotation matrix can be used to project vectors from VC-NED-CS to BF-CS as well [16]. Using the trigonometric definitions of the elements

$$\begin{aligned} \Theta &= \arcsin\Big(-R_{31}\Big), \\ \Phi &= \arctan\left(\frac{R_{12}}{R_{11}}\right), \\ \Psi &= \arctan\left(\frac{R_{23}}{R_{33}}\right), \end{aligned} \tag{2.8}$$

it is also possible to calculate Euler angles from this rotation matrix as provided in Equation (2.8).

2.4. Algebraic Properties of Rotation Matrices

A vector is considered normal or normalized if its length equals 1. Orthogonal vectors are perpendicular to one another (i.e., their inner product is zero). If all vectors in a set of normalized vectors are orthogonal to each other, then the vector set is orthonormal. Algebraic properties of rotation matrices directly come from the physical meanings of their columns, as shown in Figure 2.6.

The columns of the rotation matrix denote the evolving axes of BF-CS with respect to I-NED-CS. Rows and columns of rotation matrices are orthogonal to one another; thus, rotation matrices are referred to as orthogonal matrices [6]. The algebraic specifications of rotation matrices may be listed as follows.

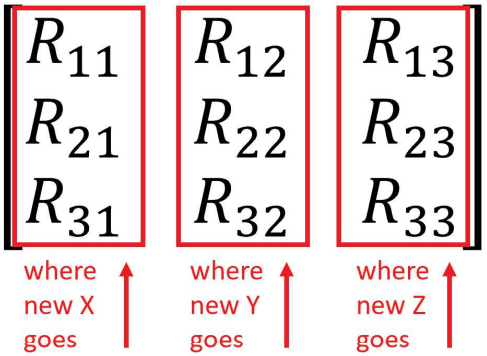


Figure 2.6. The columns of a rotation matrix have significant physical interpretations.

- The determinants of rotation matrices are equal to 1.
- The inverse and transpose of rotation matrices are always equal.
- The dot product of any row or column with itself yields a value of 1.
- The dot product of two distinct rows or columns is always zero.
- The cross product of two rows or columns yields a third row or column.

Although this rotation philosophy works well for a wide range of angles, significant shortcomings such as gimbal lock (loss of DOF) and linearity violations near $\pi/2$ appear [21]. The gimbal lock structure resembles Figure 2.7. The $\boldsymbol{X}$ and $\boldsymbol{Z}$ rings align when the angle θ equals $\pm\pi/2$. Rotating an object along the $\boldsymbol{Z}$ and $\boldsymbol{X}$ axes yields identical orientations, referred to as "loss of freedom" [2]. One can look at the same problem from the point of view of the rotation matrix and see that Φ and Ψ cannot be found independently because there are no two linearly independent equations [11].

The rotation matrix will be like

$$\begin{aligned} R_1^0 &= R_Z(\Psi)R_Y(\pi/2)R_X(\Phi), \\ &= \begin{bmatrix} 0 & \sin(\Phi+\Psi) & -\cos(\Phi+\Psi) \\ 0 & \cos(\Phi+\Psi) & \sin(\Phi+\Psi) \\ 1 & 0 & 0 \end{bmatrix}, \end{aligned} \tag{2.9}$$

the pitch angle equals $\pi/2$ [16]. Gimbal lock is an acknowledged phenomenon characterized by the inability to uniquely predict the values of Φ and Ψ angles, resulting in a

loss of freedom when the theta angle reaches $\pi/2$. One potential approach to address this issue involves using a rotation methodology grounded in quaternions organized within the realm of complex space [22].

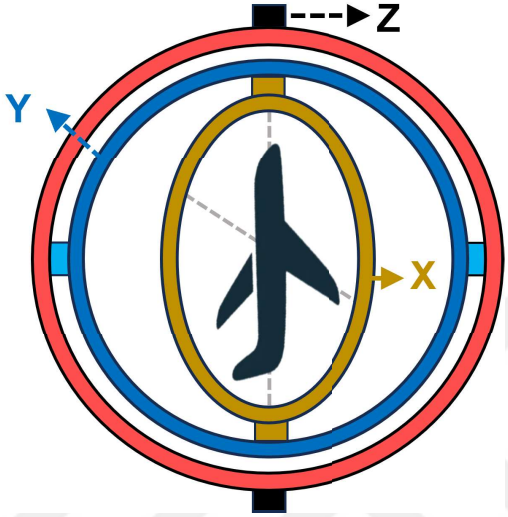


Figure 2.7. The gimbal lock situation where the $\boldsymbol{X}$ and $\boldsymbol{Z}$ axes overlap each other.

2.5. Quaternions and Hamiltonian Rotation Technique

Before discussing the quaternions, recognized as a four-dimensional (4D) extension of complex numbers, it is beneficial to recall the nature of two-dimensional complex numbers. First, the Euler representation

$$e^{i\theta} = \cos\theta + i\sin\theta, \tag{2.10}$$

must be given [23]. In this expression, θ is the argument of a complex number, and i is the complex number having its own square -1. Moreover, this number can be expressed in polar coordinates using the length and argument parameters shown in Figure 2.8.

A unit complex number is a complex number with length (norm) 1. Any complex number multiplied by the unit complex number whose argument is θ in the two-dimensional complex plane is rotated by θ [24]. This was the unique assertion that motivated Rowan Hamilton in 1843. If rotation with complex numbers is possible in

two dimensions, it should also be possible in three dimensions [25]. Thus, what should the product of $\boldsymbol{i}$ and $\boldsymbol{j}$ be equal to if one designates the third axis as $\boldsymbol{j}$? It is a well-known story that Hamilton's kids used to question him every morning about whether he could perform this multiplication. At last, he created an additional complex number $\boldsymbol{k}$. He did not use a real axis but three imaginary numbers: $\boldsymbol{i}$, $\boldsymbol{j}$, and $\boldsymbol{k}$. Algebraic relations between these complex numbers are

$$\begin{aligned} i^2 = j^2 = k^2 = ijk = -1, \\ ij = -ji = k, \\ jk = -kj = i, \\ ki = -ik = j. \end{aligned} \tag{2.11}$$

A unit quaternion conducts rotation around the embedded vector at an amount of its argument [23]. Just like 2D complex numbers do [19].

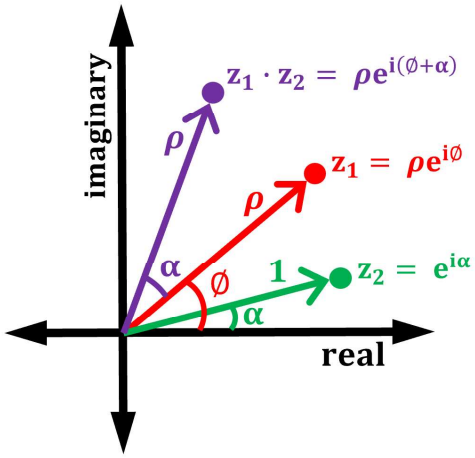


Figure 2.8. Complex numbers in polar coordinates are illustrated. Multiplicating a unit number by any other complex number results in the rotation of θ.

To represent a 2D complex number z as a 4D unit quaternion q_u, one can write

$$\begin{aligned} z &= \cos\Theta + i\sin\Theta, \\ z &= \cos\Theta + \sin\Theta(1 * i + 0 * j + 0 * k), \\ q_u &= \cos\Theta + \boldsymbol{u}\sin\Theta. \end{aligned} \tag{2.12}$$

A quaternion may be expressed as the sum of four components in an array form like

$$\begin{aligned} q_u &= \cos\Theta + \sin\Theta(1 * i + 0 * j + 0 * k), \\ &= \begin{bmatrix} q_0 & q_1 & q_2 & q_3 \end{bmatrix}. \end{aligned} \tag{2.13}$$

In this expression, Θ is the amount of rotation, $\boldsymbol{u}$ is the complex vector around which the rotation takes place, and $[q_0\ q_1\ q_2\ q_3]$ is a quaternion quartet. The real part of the quaternion is represented by q_0, whereas the imaginary part is depicted by q_1, q_2, and q_3 [23]. As 4D unit complex numbers, quaternions can be added, subtracted, multiplied, and divided. Nonetheless, they disobey the commutation law. The order of multiplication has a direct impact on the multiplication result. Furthermore, a typical quaternion, like 2D complex numbers, can be illustrated in Euler form, as shown in Table 2.1 [19].

Table 2.1. Representation of a quaternion in vectorial and Euler form.

$q_u = \cos\Theta + \boldsymbol{u} * \sin\Theta$	Vectorial Form
$q_u = e^{\boldsymbol{u}\Theta}$	Euler Form

Any quaternion's conjugate corresponds to the same real but negated vector part, which will be used for quaternion rotations. Table 2.2 can be used to visualize these definitions.

Table 2.2. Representation of a quaternion's conjugate.

$q_u = \cos\Theta + \boldsymbol{u} * \sin\Theta$ $q_u = e^{\boldsymbol{u}\Theta}$	$\bar{q}_u = \cos\Theta + -\boldsymbol{u} * \sin\Theta$ $\bar{q}_u = e^{-\boldsymbol{u}\Theta}$
A Unit Quaternion	Conjugate of the Quaternion

The next step is to build the complex space. There are three axes denoted by the letters $\boldsymbol{i}$, $\boldsymbol{j}$, and $\boldsymbol{k}$, but what about the origin? Which number should be used as the starting point? This question can be answered by writing zero for vector and rotation angle in Euler form. Then, one appears as a result, shown in Figure 2.9 [26].

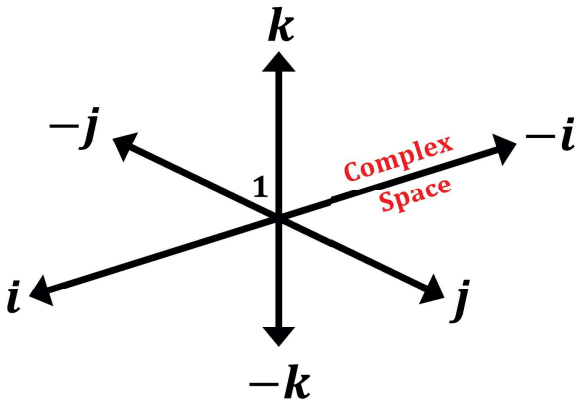


Figure 2.9. Complex space Cartesian coordinates representation.

Because the origin is 1 rather than 0, multiplications have complicated meanings known as rotation and translation of complex space. For example, the left multiplication of $\boldsymbol{i}$ with the axes rotates the space CCW and shifts the origin to the positive $\boldsymbol{X}$ side. The right multiplication of $\boldsymbol{i}$ with the axis causes the system to turn clockwise while translating the origin to the positive $\boldsymbol{X}$ side [23]. Once more, the right multiplication of $-\boldsymbol{i}$ rotates the system CCW and shifts the origin to the opposing side. Rotations must be actualized without translation. Otherwise, the complex space will change shape, the center of the unit circle will move away, and the results will not be accurate. The required multiplications are depicted in Figure 2.10 [19].

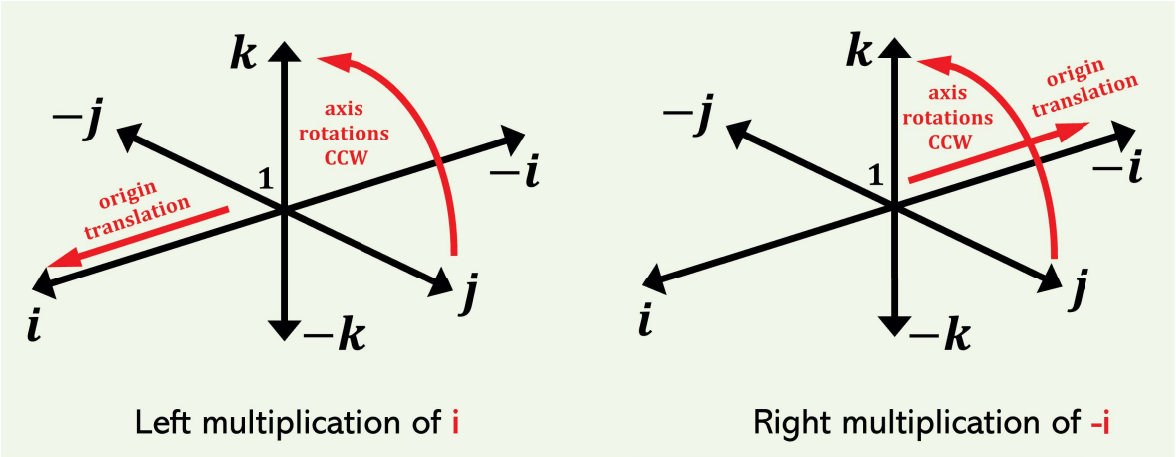


Figure 2.10. Translations of origins and rotations of axes for the mentioned multiplications.

If one combines the two rotations exhibited in Figure 2.10, the stated zero translation condition for pure rotation can be satisfied. Remember that complex numbers in these cases are conjugate and have the same amount of rotation, which must be halved when constructing the multiplication [26].

Theorem 2.1. *Assume that* $\boldsymbol{V}$ *is a vector rotated in complex space around the direction* $\boldsymbol{U}$ *and the quantity of* Θ*. The rotated vector* $\boldsymbol{V}$ *can be written as*

$$R_{\boldsymbol{u},\Theta}(V) = e^{\boldsymbol{u}\Theta/2} \otimes V \otimes e^{-\boldsymbol{u}\Theta/2}.$$

So, how are these quaternions produced and used in robotic systems? When gyroscope values (angular rates) are collected in each cycle, it is necessary to understand how to transform them into quaternion data. The quaternion rate

$$\dot{q}_u = \frac{1}{2}\omega \otimes q_u, \tag{2.14}$$

where $\dot{q}_u$ denotes the actual quaternion rate, and ω is the body angular velocity, which is the crucial concept here. As given in Equation (2.14), this setup can be built in SIMULINK as given in Figure 2.11.

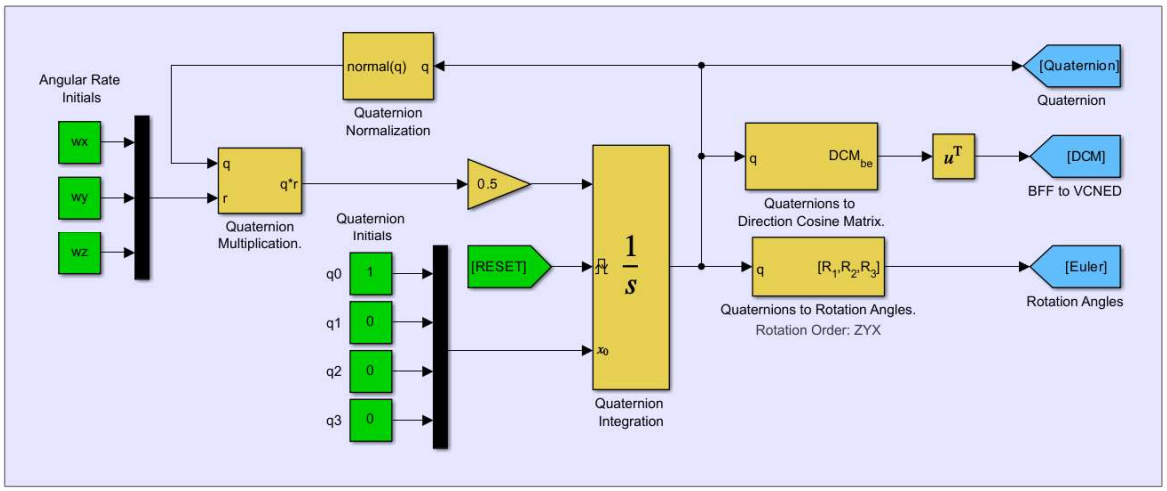


Figure 2.11. Schematic for quaternion creation and usage from angular rates.

This model can be used to retrieve quaternions for each time step. Then, functions like "*quat2eul*" and "*quat2dcm*" can be used to generate more data [27]. Another critical difficulty is figuring out how to use quaternions to remove the gimbal lock

when the pitch angle is $\pi/2$ radians. The only way to avoid gimbal lock is to employ a quaternion-based attitude controller [23]. The numerical singularity in the rotation matrix appears again when returning to Euler Angles from quaternions [19]. Regardless of whether quaternions are utilized in the state estimation phases, a linear dependent equation system is always produced at $\pi/2$ radians perpendicular moments. Although this circumstance makes little sense for conventional drone flights, it is a significant issue in drones with unique flight modes such as somersaults and dive modes.

2.6. Transition between Euler Angles and Quaternions

Euler angles and quaternions are two distinct approaches employed to characterize the attitude of an aircraft. These procedures fundamentally differ from one another in their fundamental principles. Quaternions consistently adhere to the VC-NED-CS coordinate system, but Euler angles do not belong to any coordinate set that satisfies the right-hand rule. While Euler angles define rotation around 3 different vectors whose definitions are always the same, quaternions provide rotation around a vector whose description constantly changes. For these reasons, a direct transition between quaternions and Euler angles cannot be achieved. However, it is known that the columns of the rotation matrix are the projections of the BF-CS axes on VC-NED-CS (see Figure 2.6). In other words, BF-CS can be created by rotating the VC-NED-CS axis vectors as

$$DCM = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 + q_0q_3) & 2(q_1q_3 - q_0q_2) \\ 2(q_1q_2 - q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 + q_0q_1) \\ 2(q_1q_3 + q_0q_2) & 2(q_2q_3 - q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix}, \tag{2.15}$$

where q_i is the i^{th} term of the quaternion. These new BF-CS axes definitions, which can be created with quaternion rotation, can be obtained by writing the columns of the rotation matrix (or DCM). Switching from DCM to Euler angles with element-wise equations is also possible (see Equation (2.8)). To reduce the processing load, the angular velocities coming from the gyroscope are converted into quaternion rates and quaternions. Therefore, returning from quaternion to DCM and then Euler angles is

crucial to keep the controller working. Besides, to build a corresponding quaternion from the Euler Angles

$$q = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} \cos\frac{\Phi}{2}\cos\frac{\Theta}{2}\cos\frac{\Psi}{2} + \sin\frac{\Phi}{2}\sin\frac{\Theta}{2}\sin\frac{\Psi}{2} \\ \sin\frac{\Phi}{2}\cos\frac{\Theta}{2}\cos\frac{\Psi}{2} - \cos\frac{\Phi}{2}\sin\frac{\Theta}{2}\sin\frac{\Psi}{2} \\ \cos\frac{\Phi}{2}\sin\frac{\Theta}{2}\cos\frac{\Psi}{2} + \sin\frac{\Phi}{2}\cos\frac{\Theta}{2}\sin\frac{\Psi}{2} \\ \cos\frac{\Phi}{2}\cos\frac{\Theta}{2}\sin\frac{\Psi}{2} - \sin\frac{\Phi}{2}\sin\frac{\Theta}{2}\cos\frac{\Psi}{2} \end{bmatrix}, \tag{2.16}$$

can be used. In the present context, it is essential to acknowledge that the construction of DCM is unavoidable in getting Euler angles from quaternions. Euler angles can also be obtained from quaternions.

2.7. Mathematical Relationship between Angular Rates

The Euler angular rates refer to the time variation of the Euler angles, which are used to define the spatial orientation of an object inside a three-dimensional environment [28]. Without referring to a particular orientation representation such as Euler angles, body angular rates describe how fast the UAV rotates around its body axes as

$$\boldsymbol{\omega_E} = \begin{bmatrix} \dot{\Phi} \\ \dot{\Theta} \\ \dot{\Psi} \end{bmatrix} \qquad \boldsymbol{\omega_B} = \begin{bmatrix} p \\ q \\ r \end{bmatrix}. \tag{2.17}$$

Euler angular rates ($\boldsymbol{\omega_E}$) are associated with changes in orientation angles, while body angular rates ($\boldsymbol{\omega_B}$) are related to rotation rates about the UAV's body axes. Therefore, assuming that Euler and body angular rates are equivalent in rotary-wing UAVs is incorrect. This is a mistake made by many working groups. Furthermore, Euler angular rates are not associated with any right-handed coordinate system [25]. On the other hand, platform-mounted gyroscopes measure angular velocities relative to right-handed BF-CS axes. No one can relate Euler angular rates to body angular rates using a rotation matrix [4]. However, a straightforward mathematical connection

can be established between the two as

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} \dot{\Phi} \\ 0 \\ 0 \end{bmatrix} + R_X \begin{bmatrix} 0 \\ \dot{\Theta} \\ 0 \end{bmatrix} + R_X R_Y \begin{bmatrix} 0 \\ 0 \\ \dot{\Psi} \end{bmatrix}. \tag{2.18}$$

The following procedure works when transforming gyroscope-measured body angular rates to Euler angular rates as given in Equation (2.18).

(i) Since the two coordinate sets are not differentiated from each other yet, the body's angular velocity about $\boldsymbol{X}$ axis (p) equals the $\dot{\Phi}$.

(ii) Only $\boldsymbol{X}$ rotation should be applied during the transition from body $\boldsymbol{Y}$ axis angular velocity (q) to $\dot{\Theta}$ since it was just rotated along the $\boldsymbol{X}$ axis.

(iii) $\boldsymbol{X}$ and $\boldsymbol{Y}$ rotation must be performed while transforming from body $\boldsymbol{Z}$ angular velocity (r) to $\dot{\Psi}$ because two rotations have already occured [29].

The relation matrix between body angular rates and Euler angular rates is

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin\Theta \\ 0 & \cos\Phi & \sin\Phi\cos\Theta \\ 0 & -\sin\Phi & \cos\Phi\cos\Theta \end{bmatrix} \begin{bmatrix} \dot{\Phi} \\ \dot{\Theta} \\ \dot{\Psi} \end{bmatrix}. \tag{2.19}$$

The inverse of rotation matrices is commonly referred to as their transpose. However, the relation matrix in this case does not fit the criteria for becoming a rotation matrix. A matrix inversion procedure can convert body angular rates to Euler rates. This conversion is highly advantageous as the gyroscope readings always provide the body angular velocities, and the Euler angular rates obtained from the relationship matrix serve as feedback to the controller loop [30]. The inverse of the given relation matrix is as

$$\begin{bmatrix} \dot{\Phi} \\ \dot{\Theta} \\ \dot{\Psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\Phi\tan\Theta & \cos\Phi\tan\Theta \\ 0 & \cos\Phi & -\sin\Phi \\ 0 & \sin\Phi\sec\Theta & \cos\Phi\sec\Theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix}. \tag{2.20}$$

It encases two critical problems that might happen near pitch angles of $\pi/2$ degrees.

- $\tan\Theta$ and $\sec\Theta$ are indefinite or infinite.
- Φ cannot be found through inverse trigonometric operations.

It is known that the pitch angle in a somersault or diving mode is $\pi/2$ radians or very close to it. Then, the controller becomes unresponsive at these moments because the Euler angles and their rates cannot be calculated.

2.8. Propeller and Thrust Mechanics

Propellers are the main actors that produce lift by re-directing the air flows downwards and inducing a pressure difference between their upper and lower surfaces [17]. Using Blade Element Momentum Theory, lift and drag forces on a propeller blade can be formulated by taking airflow velocities as variables. Rotary wing blades can be considered a combination of multiple airfoils whose angles of attack and chord lengths vary with respect to radial distance [31]. Like fixed wings, propellers generate lift and drag forces as air flows over them. Radius-varying geometry requires infinitesimal free-body diagram (Figure 2.12) analysis to model these blades.

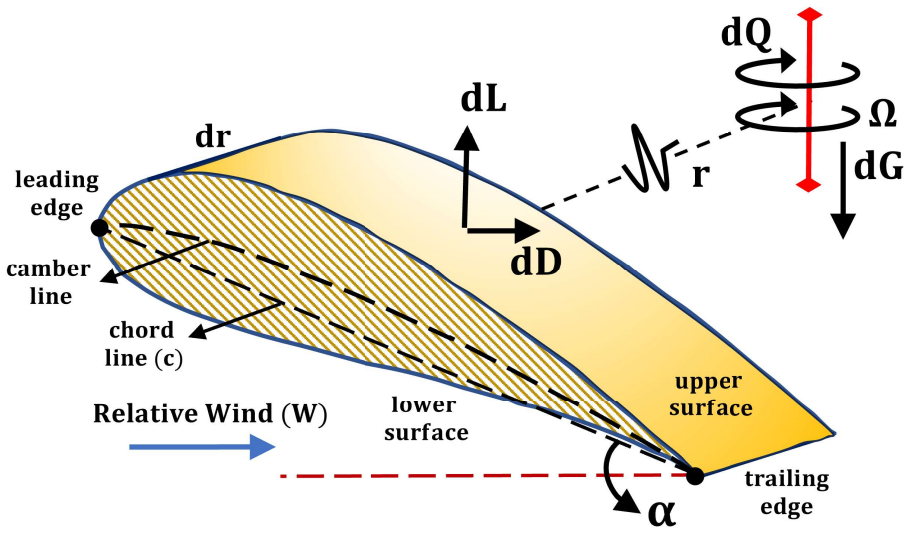


Figure 2.12. Forces and moments on the infinitesimal (dr) blade section.

In the schematic, dG, Ω, and α are differential weight, propeller rotational velocity, and angle of attack. They gain vectorial meaning with this statement: Aerodynamic drag forces always emerge in the direction of wind axes while the lift is perpendicular to it. Moreover, drag force creates a torque on the rotor shaft after multiplying it with the distance to the rotation axes. The differential lift (dL_i), drag (dD), and torque (dQ) may be formulated as

$$\begin{aligned} dL_i &= \frac{1}{2}\rho\Omega^2 r^2 c(r) dr C_L(\alpha), \\ dD &= \frac{1}{2}\rho\Omega^2 r^2 c(r) dr C_D(\alpha), \\ dQ &= r * dD, \end{aligned} \tag{2.21}$$

where c is the chord length determined by radius, ρ is the density of the air, and α is the angle of attack depending on the airfoil's geometry. C_L and C_D are the lift and drag coefficients conditioned by the angle of attack. Airflow passing through the airfoils also creates a pitching moment. However, this moment is negligibly small [32]. After taking the definite integral over the blade radii, the most simplified form of propeller lift and torque can be expressed as

$$\begin{aligned} L_i &= k_F \Omega^2, \\ Q &= k_Q \Omega^2, \end{aligned} \tag{2.22}$$

where k_F and k_Q are thrust and torque coefficients greater than zero. The crucial assumption is that the relative wind velocity occurs only when the propeller rotates. The angle of attack changes at each element if there is an external wind, meaning that the lift and torque of the propellers change. This effect shows itself strongly in coaxial propeller systems. However, a mathematical model for these coaxial propellers has not been made yet. The discrepancy is added to future work.

In Section 3.2, a mathematical model for a quad-copter to test and compare the control allocation algorithms are built in SIMULINK. Both platform and propeller parameters are realistically entered into the model to assess the limit violations correctly. The selected propeller for the model is a 12×4.0 two-bladed propeller, and its k_F and k_Q coefficients are calculated from the experimental data (Figure 2.13) [33]. While the

number 12 gives the diameter of the propeller from tip to tip, 4.0 is the pitch of the propeller, that is, the screw step that indicates how far it goes with a revolution. The units of both lengths are inches, but since SI units are used in modeling, they have been converted to meters.

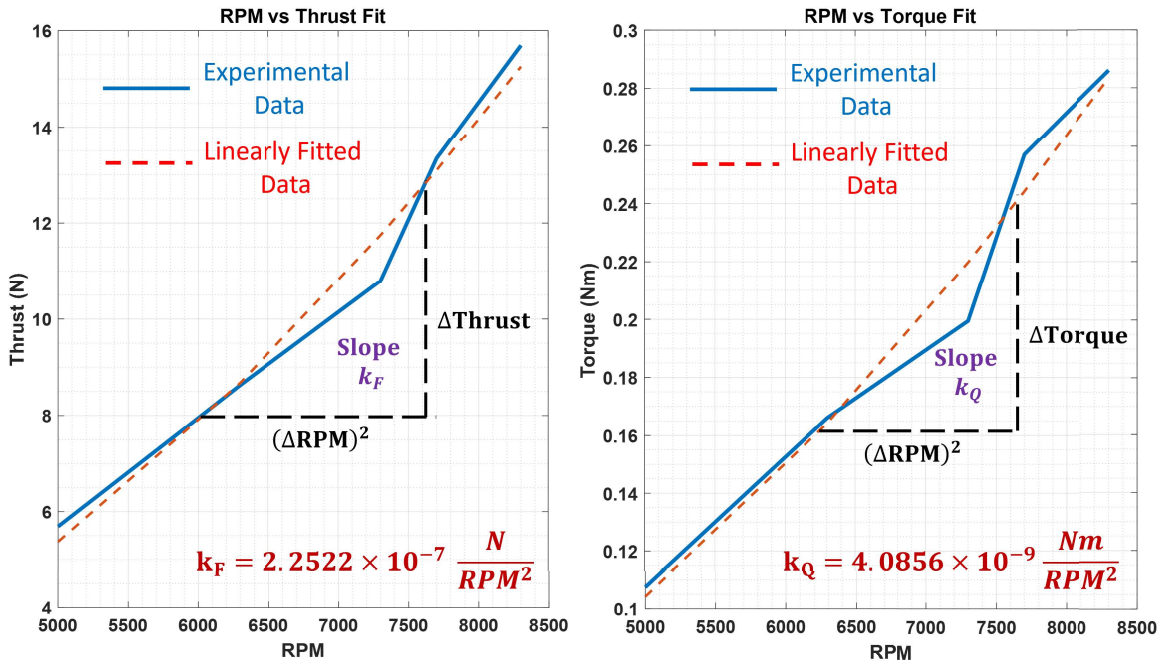


Figure 2.13. Thrust and torque fit of 12×4.0 propeller with respect to RPM. Slopes of both figures equal to k_T and k_Q.

2.9. Aircraft Sensors

Sensors play a significant role in observing the aircraft's actualized kinematics. Monitoring several parameters, including acceleration, velocity, location, altitude, temperature, pressure, and humidity, is essential to maintain precise navigation and to operate the control surfaces. The following section provides an overview of these sensors and their operational principles. Proficiency in sensor mechanics is crucial for accurately assessing potential issues on the aerial platform. In addition to mechanical operation, mastering estimation algorithms is essential to benefit from these sensors most. Even the most minor vibrations disrupt their operation since these hardware are obtained using MEMS production techniques.

2.9.1. Inertial Measurement Units

Inertial Measurement Units (IMUs) include accelerometers and gyroscopes, which quantify the linear accelerations and angular velocities respectively [24]. Although larger sizes of them have simple methods such as deflecting springs due to inertial force and angular momentum conservation (Figure 2.14), MEMS inertial sensors typically rely on capacitance change, a very susceptible and specialized method [6]. As a result, these components find wide use in aircraft with their fast reaction capabilities.

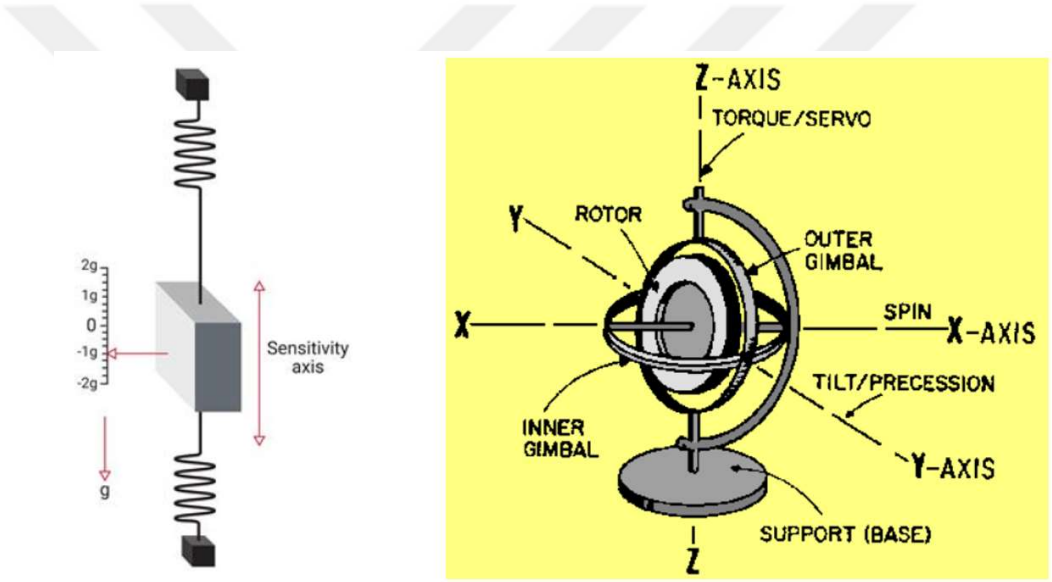


Figure 2.14. A large-sized accelerometer (left) and gyroscope (right) [34].

MEMS accelerometers like the one in Figure 2.15 consist of membranes that can precisely measure capacitive changes and a spring-supported mass traveling between these membranes. The idea is when the platform to which the sensor is attached accelerates, the central mass is displaced, and so is the capacitance. One may convert this deviation into acceleration with a simple coefficient [21]. Hence, accelerometers detect linear accelerations [35]. As mentioned before, minimal capacitive values that change with this membrane movement also change with vibration. Then, estimation algorithms, such as the Kalman filter, particle filter, and low and high pass filters, are needed to detect unreal measurements.

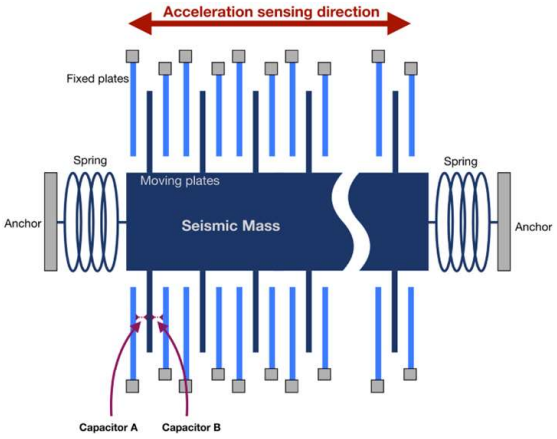


Figure 2.15. Basic schematic of MEMS accelerometer [34].

MEMS gyroscopes, also called MEMS angular rate sensors, are high-tech sensors that capture angular velocities. According to how they work, MEMS gyroscopes can be broken down into two main groups: Coriolis and optical gyroscopes. Coriolis gyroscopes are easy to find and offer different levels of accuracy, while optic gyroscopes are often very rigorous and too expensive for use in ordinary vehicles [36]. MEMS gyroscopes use the Coriolis force to measure rotational speeds; that is where their name comes from. Inside this hardware, an oscillating mass moves back and forth in the $\boldsymbol{X}$ and $\boldsymbol{Y}$ axes, corresponding to continuous translational velocity (Figure 2.16). If one gives an angular momentum to the system that the gyroscope is attached to, the mass will be displaced due to Coriolis force. With a constant, the change in capacitance caused by this movement can be converted into angular speed [37]. MEMS IMU sensor calibration requires using exact setups at least once [36].

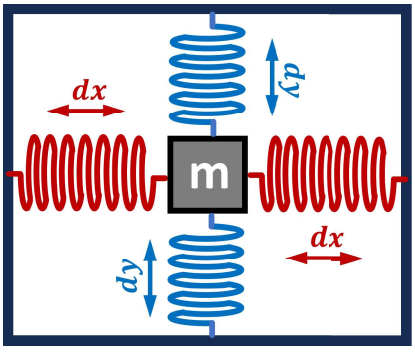


Figure 2.16. Vibrating mass in MEMS gyroscope.

Optical gyroscopes are sensitive and advanced hardware because they work with light. In this equipment, there are two channels that the rays must follow. Angular velocity calculation is made by examining the kinematics of these rays along their tracks [38]. When there is no rotational motion, these rays travel their routes simultaneously. When angular velocity is present, the rays reach their targets at different times due to the effect of rotation on the relative speeds of the beams (Figure 2.17). This phenomenon, known as the Sagnac Effect, is the operating technique of optical gyroscopes [38].

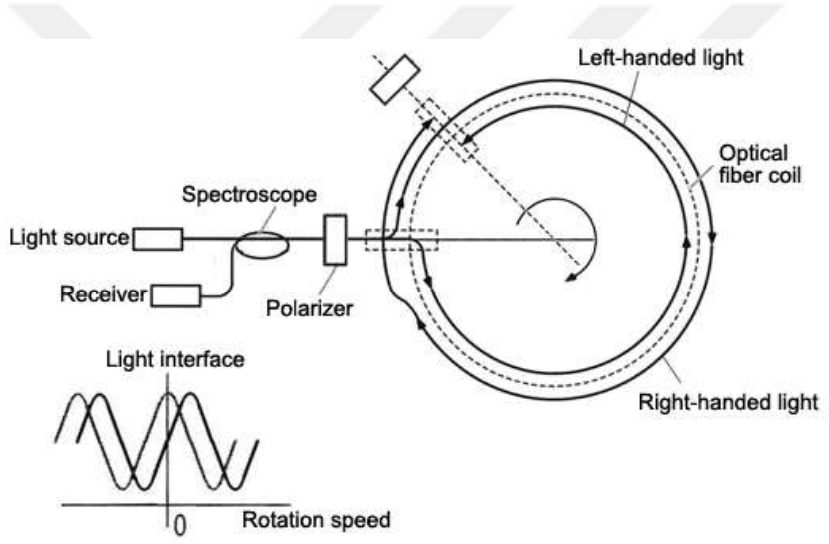


Figure 2.17. Optical Gyroscopes in operation [34].

2.9.2. Barometers

In the past, barometers were commonly constructed using liquid substances such as mercury and colored alcohol. The alteration in pressure induces a corresponding change in the liquid level within the column, which a straightforward linear function can elucidate [39]. The Torricelli assembly for evaluating atmospheric pressure was the first of its kind. In 1643, Evangelista Torricelli created the procedure given in Figure 2.18. The fundamental principle here is that the pressure at the bottom of the vacuumed column must be equal to atmospheric pressure as

$$P_a = \rho h g. \tag{2.23}$$

In this context, P_a is the atmospheric pressure, ρ is the liquid density inside the evacuated column, h is the column height, and g is the specific gravity [40].

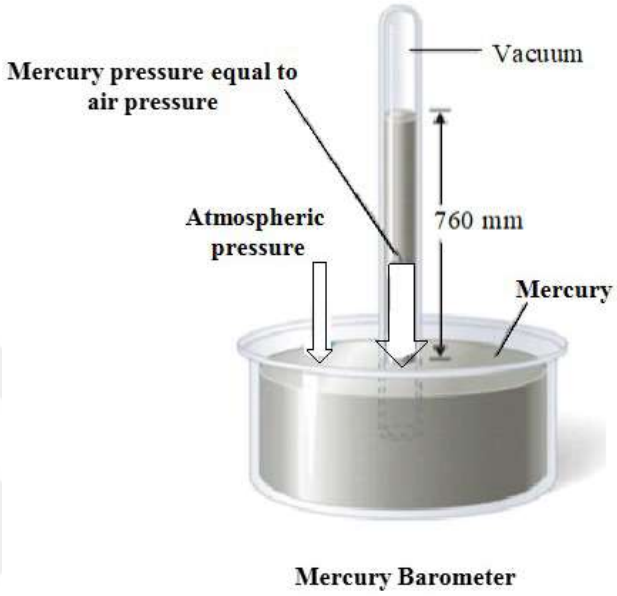


Figure 2.18. Conventional mercury barometer [40].

A few years later, in 1844, Lucien Vidi created aneroid pressure sensors better suited for mechanical systems (Figure 2.19). These structures have a moving indicator bar on the pressure scale. Small changes in the pressure diaphragm are transmitted to the lever incrementally [41].

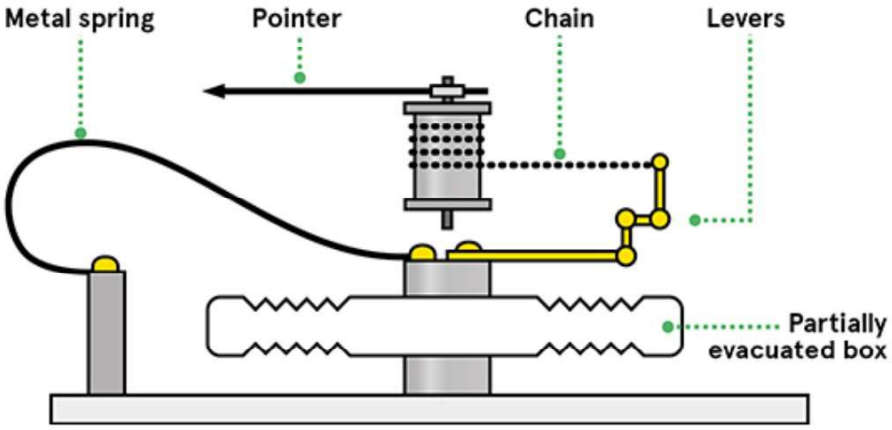


Figure 2.19. Aneroid pressure sensor schema [40].

MEMS barometers are remarkable and outstanding new hardware for measuring atmospheric pressure and altitude. These sensors are also used in weather forecasts and smartphone applications [42]. Like aneroid pressure sensors, MEMS pressure sensors create capacitive changes by stretching their tiny diaphragms. The changes in pressure have a direct effect on how the diaphragm deforms. A coefficient can show a link between capacitive change and stress [40].

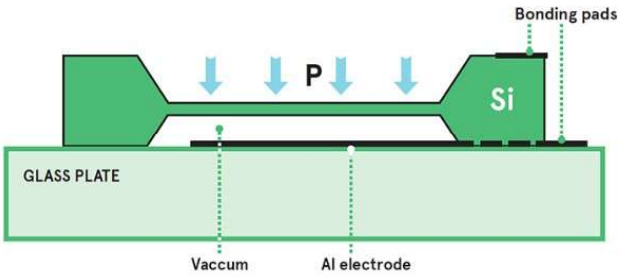


Figure 2.20. A cross-sectional cut of MEMS capacitance pressure sensor [40].

Now that MEMS piezo-resistive strain gauge sensors, as depicted in Figure 2.20, work with capacitance changes, i.e., the distance between capacitive plates, they are highly susceptible to oscillations. When mounting these devices to the chassis and developing sensor fusion algorithms, one must pay close attention to them [41].

2.9.3. Global Positioning

When visibility is restricted, and an aircraft is required to operate at far distances, it is necessary to employ supplementary navigation techniques. Global Navigation Systems (GNS) are the way to go. Since the beginning of the 19th century, countries have set up satellite constellations to find where their aerial platforms are around the world [43]. Historical evidence demonstrates that the initial undertakings were pursued with the explicit aim of achieving military objectives. Furthermore, assistance was also extended to facilitate their reintegration into civilian society. The United States of America made initial progress in this domain by implementing NAVSTAR GPS in 1978. Following this, the GLONASS system was established by the Soviet Union in

1982, succeeded by China's BEIDOU system in 2000, and the GALILEO system by the European Union in 2005 [44]. The Galileo satellite navigation system was developed by a consortium within the European Union, distinct from previous designs, with a primary focus on servicing civilian purposes. Being the most recent, Galileo has the utmost precision [45]. Figure 2.21 shows the logos of these navigation systems.



Figure 2.21. Global positioning constellations' logos. From left to right: (a) GPS (USA), (b) GLONASS (the Soviet Union), (c) BEIDOU (China), (d) GALILEO (European Union) [46].

These navigations achieve a complete orbit in around 11 hours and 58 minutes. Therefore, they complete two revolutions about the Earth in over 24 hours. "Ground Track" is the term used to refer to satellites' trajectories on the earth's surface [47]. Making the ground track work better is possible by creating six orbitals, each with an equal and gradually rising slope toward the equator. Four satellites are set up in a certain way inside each orbit. For example, the Global Positioning System (GPS) constellation comprises 24 primary and 8 supplementary satellites at 20200 kilometers. The GPS signal can provide the following information to the receiver it reaches [44].

- Receiver distance to the satellite
- Declination angle
- Satellite's clock
- Satellite's range
- Satellite's velocity
- Satellite's health

Trilateration uses four satellite distance measurements to find the location. Indeed, three satellite distances and an equation representing Earth's surface can simplify position estimation to a point. There must be a fourth satellite so that time can be corrected. Still, several things could go wrong with the electromagnetic waves sent from satellites to people worldwide. These problems could cause delays, weaken the signals, or even send wrong information to the receivers [46]. Chief among these are listed below.

- Atmospheric effect
- Multi-path effect
- Ephemerisis effect
- Satellite's positional geometry effect
- Satellite's clock drift

Electromagnetic waves sent by satellites may slow down because of the atmosphere and clouds, which causes them to arrive at their destination later than planned. Even a short delay, such as a few nanoseconds, can cause a position deviation of tens of meters (Atmospheric effect). Signals from satellites change direction and intensity after being reflected from geographical elements. Receivers may assume these changes are natural and misjudge them (Multi-path effect) [44]. When the satellites sending signals are close to each other, the compensation rate for errors decreases. Because the sets of linear equations that need to be solved reduce the degrees of independence. Sensitivity increases when the satellites are positioned on opposite sides and at considerable distances (Satellite's positional geometry effect). Dilution of information transmitted via satellite signals may result from the gravitational phenomena occurring in the space-time fabric and the relativistic effects experienced by time (Ephemerisis effect). When measured from Earth, satellites have a 45-microsecond daily delay. Ground control systems fix this delay (Satellite's clock drift) [47].

In addition to the global constellations, many governments have developed regional navigation satellite systems (RNSS). They are specifically designed to maximize sensitivity inside their territories. The Indian Regional Navigation Satellite System

(IRNSS), the Quasi-Zenith Satellite System (QZSS) of Japan, and the Korean Positioning System (KPS) are the prominent examples of them [46]. In conjunction with the use of more satellites, ground control stations have the potential to augment and intensify radio transmissions. Thus, satellite-based augmentation systems (SBAS) are used for advanced purposes such as driverless vehicles and mechanized agriculture [45].

2.9.4. Magnetometers

A magnetometer is a compass sensor that detects the angle formed between the vehicle's $\boldsymbol{X}$ axis and geographic north [48]. It is essential to be aware that the geographic and magnetic north poles exhibit distinct locations concerning the meridians (Figure 2.22). The declination angle, denoting the angular difference between the geographic and magnetic north poles, measures 6 degrees in Istanbul [49]. Magnetometers quantify the Earth's magnetic field, sometimes called magnetic dipole, flux, or strength. The magnitude of this magnetic field ranges from 0.25 to 0.65 Gauss, contingent upon the geographical locations on the Earth's surface [50].

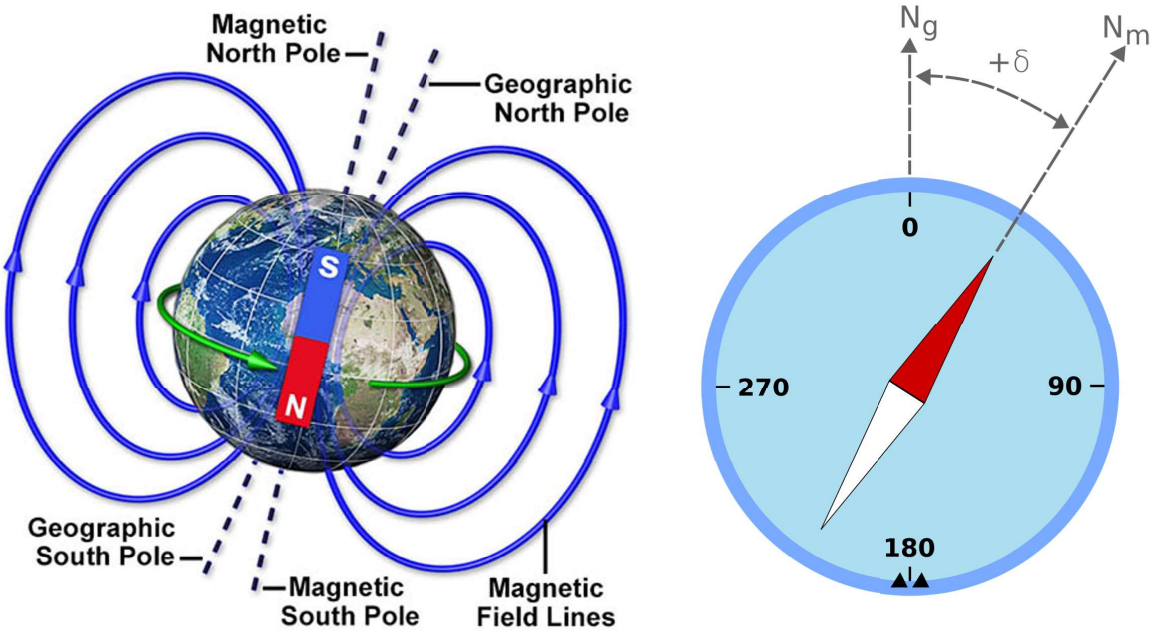


Figure 2.22. Earth's magnetic field, poles, and declination angle [50]. These values vary by region.

Beneath the lithosphere are accumulations of molten metallic substances inside the Earth, mainly composed of Iron and Nickel (Fe-Ni) having plenty of valence electrons [51]. Convective fluid motions are initiated by temperature variations extending from the Earth's core to its crust. The large-scale movements of these molten metals result in the production of stable magnetic fields via the migration of their unbound electrons. The Earth's magnetic field lines extend in three dimensions from the southern pole to the northern pole and beyond into space [52]. However, paramagnetic materials may affect line orientations and intensities, distorting magnetic field measurements. This can be examined under two headings: Effects of Hard and Soft Iron [49].

If the angular position of a magnetometer rotated in 3 axes and the magnetic field measurement were to be graphed, a perfect sphere would have to be obtained as depicted in Figure 2.23 (a). If the same graph is drawn next to a source emitting an active magnetic field, it will be seen that the sphere does not change shape but is shifted as exhibited in Figure 2.23 (b). This is known as the hard iron effect. If the same measurements and drawings are made next to paramagnetic materials, it will be seen that the sphere does not translate but becomes elliptical as in Figure 2.23 (c). This is known as the soft iron effect.

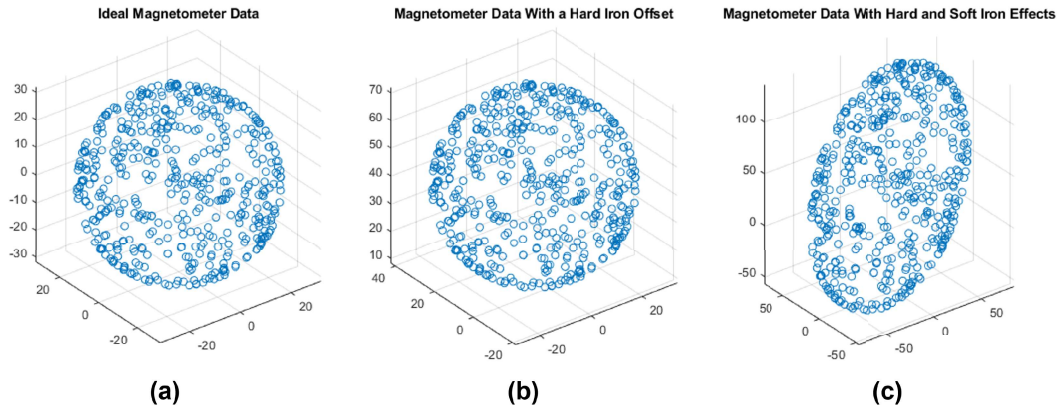


Figure 2.23. Ideal magnetic sphere at left (a), translated magnetic field sphere due to hard iron effect at the middle (b) and shape-disturbed magnetic field sphere due to soft iron effect at right (c) [51].

Irrespective of the orientation, the magnetic flux intensity is expected to stay constant when the magnetometers undergo rotation in all directions, forming a spherical shape [49]. However, due to the effect of hard and soft iron, the magnetic flux intensity is measured differently in changing directions. Thus, a perfect sphere cannot be obtained [50]. Estimating the ideal sphere with the available data and finding correction factors for measurements outside the sphere surface is known as the magnetometer calibration process [51].

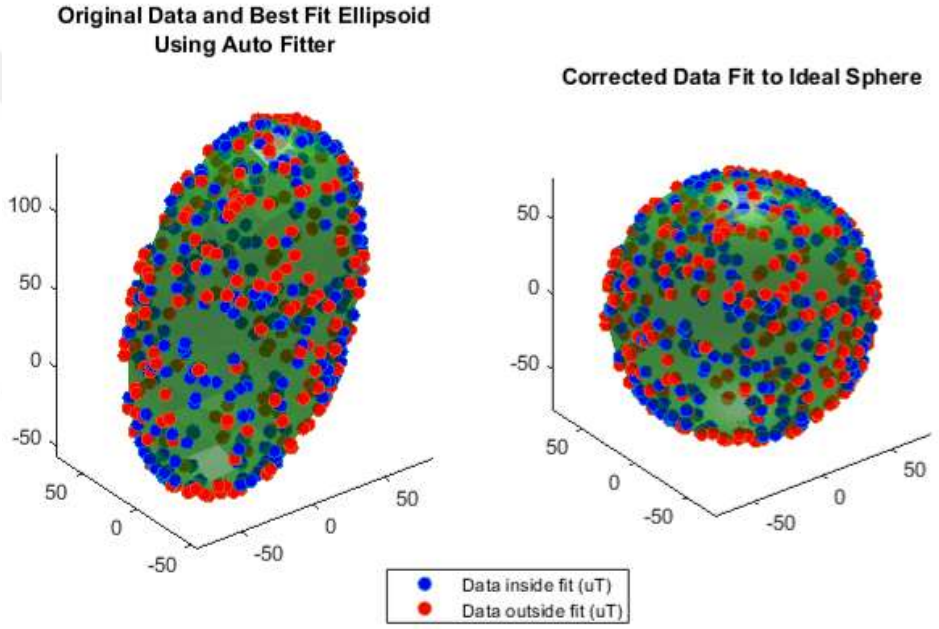


Figure 2.24. Calibrated magnetic field sphere over hard and soft iron effects. Normally, it is both shape-distorted and translated. Then, it is centered at the origin and transformed into a sphere [51].

Unlike their traditional counterparts, which work by sensing the current induced in a coil exposed to magnetic flux, modern MEMS magnetometers work directly using the Lorentz Force and Hall Effect [53]. If a perpendicular magnetic field is applied to a plate with a current passing through it, the Lorentz force occurs, which polarizes the charged particles. The voltage difference resulting from polarization is called the Hall Effect. If a moving charged particle is exposed to a magnetic field perpendicular to its velocity, it will experience both the electric field force ($q_p\boldsymbol{E}$) and Lorentz Force

$(q_p\boldsymbol{\nu} \times \boldsymbol{B})$ as

$$\boldsymbol{F_e} = q_p\boldsymbol{E} + q_p\boldsymbol{\nu} \times \boldsymbol{B}. \tag{2.24}$$

In this context, q_p is particle charge, $\boldsymbol{\nu}$ is electron drift velocity, and $\boldsymbol{B}$ is plate-perpendicular magnetic flux. When the magnetic field acts on the current-carrying plates in MEMS magnetometers, the Lorentz Force causes the charges to polarize (Hall Effect) and voltage differences to occur [35]. The Lorentz force, Hall Effect, and plate edge voltage difference are proportional to the drift velocities of electrons, magnetic flux, and electrons' charges. Thus, magnetometer-derived heading rotation assessment requires careful processing of these three variables [31]. Figure 2.25 demonstrates the typical working schema of the MEMS magnetometer.

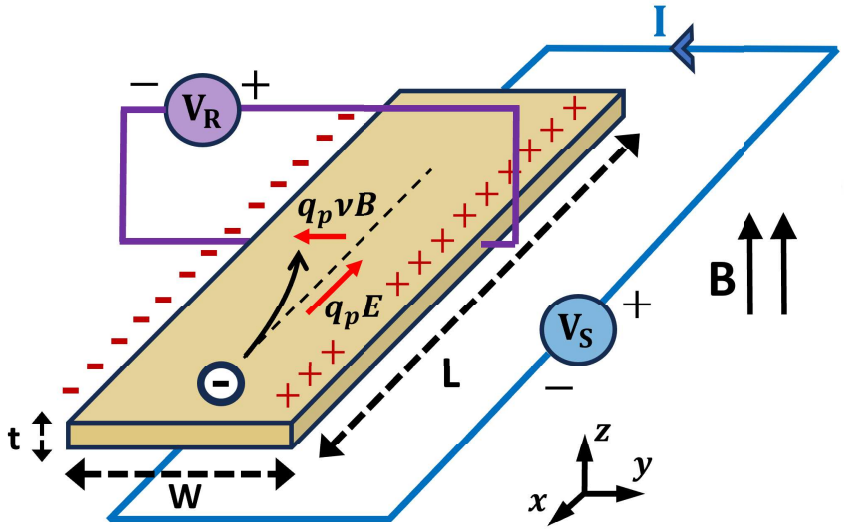


Figure 2.25. Working principle of a characteristic magnetometer.

A crucial issue must be declared here: in addition to magnetic strength and particle charges, electron drift velocities determine the Lorentz Force, Hall effect, and the magnetometers' readings [20]. Since the electron net velocity is the sum of the drift speed due to the electric field and the translational speed due to platform motion, it cannot be known with certainty in moving platforms such as uncrewed aerial vehicles or rockets [54]. Thus, using magnetometers on moving platforms causes some failures. Future work is suggested for this issue.

2.9.5. Range Finders

RW-UAVs must be able to figure out how far away they are from their surroundings, which is needed to determine the altitude and avoid objects. LiDAR and ultrasonic radars (sonar) are the primary sensors employed for depth estimation.

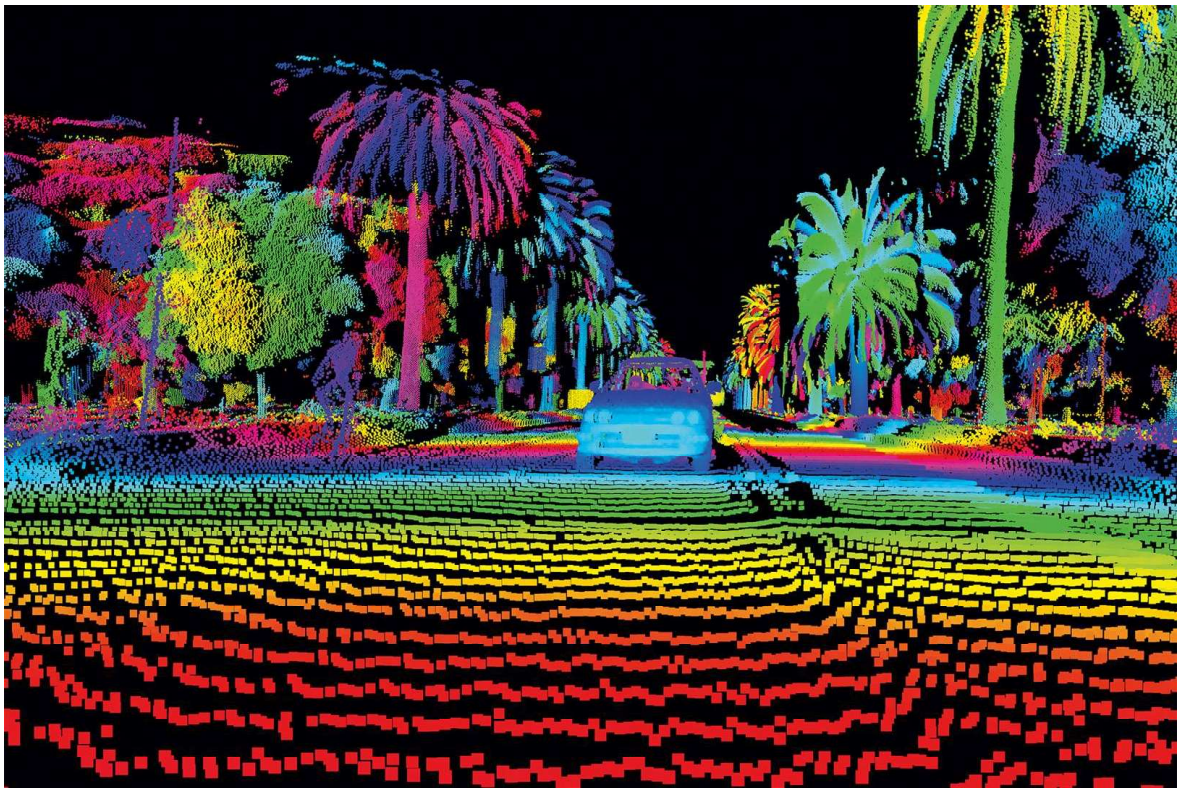


Figure 2.26. A LiDAR-scanned depth image [55].

LiDAR, an acronym for Light Detection and Ranging, is a remote sensing technique that employs laser light to determine distances (LiDAR scanned depth image in Figure 2.26). The fundamental principle underlying LiDAR is relatively straightforward [55].

- Laser Emission: A LiDAR system generates laser pulses in the direction of the designated target area. Typically, laser pulses manifest as brief episodes of light emission.

- Reflection: When the laser pulse encounters an item obstructing its trajectory, it undergoes reflection and returns toward the LiDAR sensor.
- Time-of-Flight Measurement: The duration of the laser pulse's journey to and from an object is measured by the LiDAR system. This is called time-of-flight.
- Calculation of Distance: The LiDAR system uses the constant speed of light to determine the distance to the object. This is achieved by multiplying the flight time by the speed of light and dividing the result by two, as the laser pulse must traverse the distance to the target.

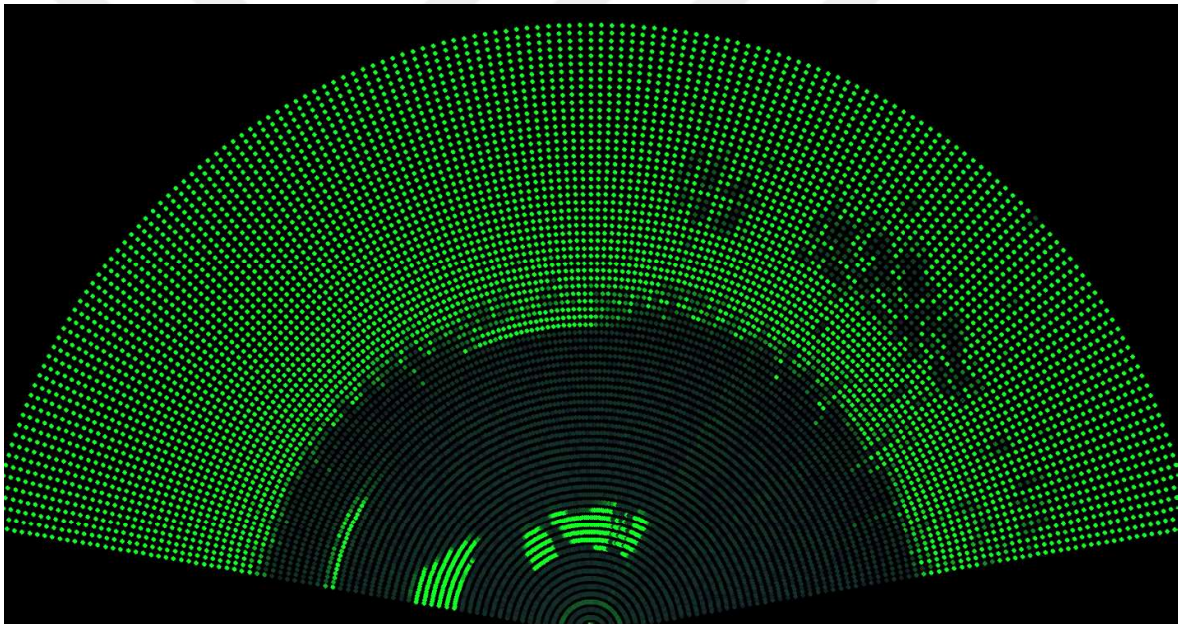


Figure 2.27. A sonar-scanned planar depth image. It is especially used in autonomous vehicles [56].

Ultrasonic radar, commonly known as ultrasonic sensors or sonar, employs sound waves to determine distance (Sonar scanned depth image in Figure 2.27). The following concisely elucidates the operational mechanisms involved [56].

- Sound Wave Emission: An ultrasonic sensor is designed to release a burst of high-frequency sound waves, commonly called ultrasonic waves, in the direction of the intended target region. The sound waves frequently exceed the audible range for

humans, commonly falling within the ultrasonic frequency spectrum.

- Reflection: When sound waves come into contact with an object, they reflect from its surface and propagate toward the sensor.
- Time-of-Flight Measurement: The sensor quantifies the duration required for the sound waves emitted to traverse the distance to the object and subsequently return. The measurement above is commonly referred to as time-of-flight.
- Calculation of Distance: The sensor determines the distance to the objects by multiplying the time-of-flight by the speed of sound and dividing by two (because sound waves must travel to and from the object).
- Distance Output: The sensor outputs the calculated distance. Ultrasonic sensors may also report signal strength, which can be used to infer surface qualities.

LiDAR and sonar sensors can provide object avoidance and local positioning by measuring distances closer than 1.5 meters. However, since they cannot measure longer distances, they cannot be used in target detection, tracking, and route planning. Although long distances can be calculated with the dual use of visible light cameras (stereovision), the resulting high production and computational costs have made this technique not preferred. As a part of this thesis, a new method is introduced, performing far distance estimation with a single visible light camera fixedly mounted on the flying platform chassis.

2.10. Estimation

Like the flight controller and the physical platform necessity, the state estimation algorithm is essential for drones to become operable. The flight controller has many feedback blocks, so these values must be obtained correctly to generate control commands. An inaccuracy or delay in the kinematic state feedback will directly affect the flight success [24]. Although it seems very simple to observe the system states with the appropriate sensors, on the contrary, it is tough to get these measurements correctly due to nondeterministic effects. Determinism is a state in which everything happening occurs for a connected reason [21]. With changes in the known cause, precisely how the

result will appear is predicted. However, cases that cannot be attributed to any available reason or technique are observed during sensor operations. These may be non-zero kinematic readings when the platform is stationary, varying readings over time for the same states, and values measured beyond physical limits, albeit instantaneously [35]. Nevertheless, nothing can happen without rational reasons to bring it to reality.

Nondeterminism is not a process detached from reality. The fact that the reasons governing some operations (especially in micro levels of semi-conductors) are too complex and quite sophisticated to be modeled has created an escape point called "Stochastic". In cases where the reasons are not as important as the results themselves, reasons are put in a black box. Then statistical relationships are established between inputs and outputs rather than a cause and effect basis [57]. This is the situation observed during the operation of sensors with physical dimensions at the micro level [22]. Consequently, state estimation algorithms come into play, and they make robust predictions by using dynamic models of the system, sensor readings, and built-in filters.

A low-pass filter for accelerometers attenuates or reduces high-frequency components in the output signal while letting low-frequency components pass through. The technique smooths out quick changes in the accelerometer data, highlighting slower variations or acceleration trends. This is helpful in applications that aim to capture fundamental motion or lower-frequency components while limiting noise or high-frequency interference [58]. The basic idea behind it is that the present measurement's amount of change is not added to the last known state if it is more than the allowed amount of change (cut-off frequency) per unit time. It is passed as is if lower. The Euler exponential function is used to conduct the mentioned operation as

$$\begin{aligned} \xi_i &= (1 - e^{-c_f \Delta\xi})\xi_{i-1} + \Delta\xi e^{-c_f \Delta\xi} \text{: time domain,} \\ \frac{\xi_i}{\xi_{i-1}} &= G(s) = \frac{a_f}{s + a_f} \text{: frequency domain,} \end{aligned} \tag{2.25}$$

where ξ_i is the i^{th} sensor reading, c_f and a_f are the cutoff frequencies, $G(s)$ is the transfer function. A better explanation may be given through the Bode plot of $G(s)$

in Figure 2.28. It is shown that all signals pass up to the cut-off frequency, which is -3dB. Beyond the cut-off frequency, all signals are exponentially attenuated with a fast reduction rate of -20dB/decade.

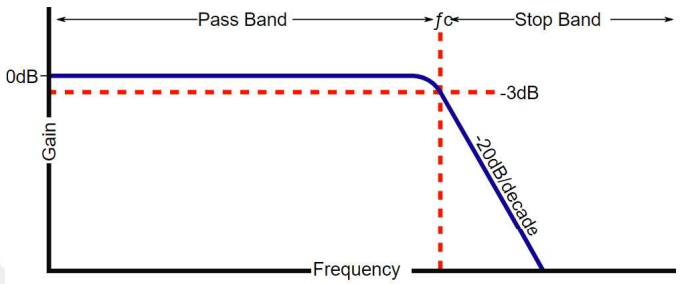


Figure 2.28. Low pass filter bode diagram showing amplitude gain in all frequency ranges considering logarithmic scale.

A high-pass gyroscope filter eliminates low-frequency components from the output signal while letting high-frequency components pass through. High-pass filters reduce lagging changes or drift, emphasizing rapid fluctuations or dynamic angular motion. This is important in applications that want to capture quick orientation changes while minimizing sluggish drift. The objective is not to add the current measurement to the previous known state if its difference is smaller than the allowable change (cut-off frequency) per unit time. The Euler exponential function is utilized as

$$\begin{aligned} \xi_i &= e^{-c_f \Delta\xi}\xi_{i-1} + \Delta\xi(1 - e^{-c_f \Delta\xi}) \text{: time domain,} \\ \frac{\xi_i}{\xi_{i-1}} &= G(s) = \frac{s}{s + a_f} \text{: frequency domain,} \end{aligned} \tag{2.26}$$

where ξ_i is the i^{th} sensor reading, c_f and a_f are the cutoff frequencies, and $G(s)$ is the transfer function. Better representation may be given through the Bode plot of $G(s)$ in Figure 2.29. It is shown that all signals attenuated up to the cut-off frequency in a fast reduction of -20dB amplitude gain. Beyond the cut-off frequency, all signals are directly passed. Removing both low-frequency and high-frequency waves from measurement can be done most accurately with Fourier Series. However, for Fourier transforms, all data must be available. Fourier transform cannot be applied to an instantaneous measurement whose rest is unknown.

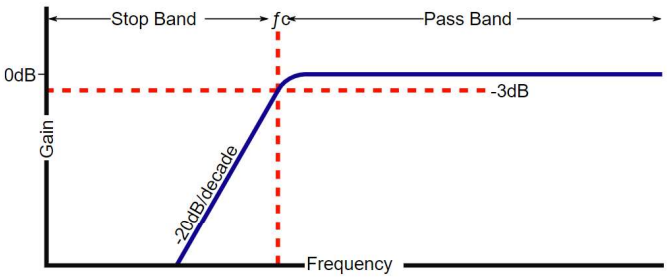


Figure 2.29. High pass filter bode diagram showing amplitude gain in all frequency ranges considering logarithmic scale.

Accelerometers are good at detecting slow changes in motion, although they may deviate instantaneously. Gyroscopes, on the other hand, are excellent at detecting quick changes but can accumulate errors over time. The complementary filter tries to produce an accurate and drift-free estimation of the system's orientation by combining low-pass filtered accelerometer data with high-pass filtered gyroscope data (see Figure 2.30).

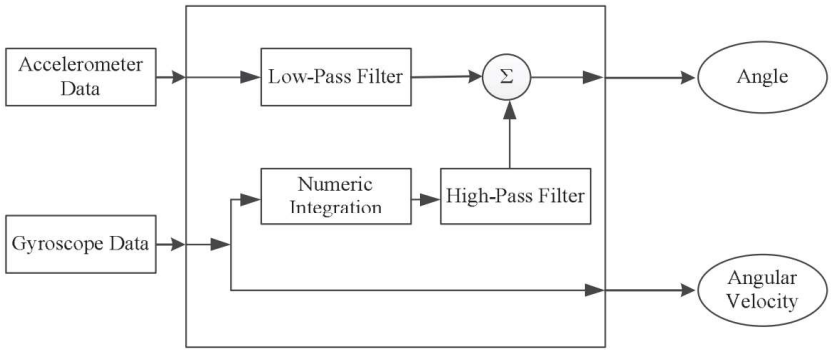


Figure 2.30. Complimentary filter including low pass filter to denoise accelerometer data and high pass filter to denoise gyroscope data.

The Kalman Filter is a sophisticated recursive mathematical technique employed to estimate the state of a linear dynamic system based on a sequence of noisy measurements. The development of this technique can be attributed to Rudolf E. Kálmán, and it has been widely utilized across several disciplines. The Kalman Filter updates and refines its system state estimates depending on the newest data and the dynamic model. It has two primary steps: prediction and update. During the prediction step,

the Kalman Filter anticipates the system's subsequent state by considering both the present state and the system's dynamics as

$$\hat{\chi}_i = K_i * \xi_i + (1 - K_i) * \hat{\chi}_{i-1}, \tag{2.27}$$

where $\hat{\chi}_i$ is the i^{th} estimation, K_i is the i^{th} Kalman gain and ξ_i is the i^{th} sensor measurement. After this step, it also provides a forecast of the covariance of the estimation error by considering the uncertainty in both the dynamics of the system and the preceding estimation. The filter compares the expected state against sensor measurements in the update step. Then, it weights the projected state and measurement depending on its reliability (Equation (2.27)). Finally, it estimates the revised state estimate and covariance to improve system state estimation.

2.11. Building the Equations of Motions for RW-UAVs

There are six degrees of freedom for a rigid body in space: three for translation and three for rotation. The crucial question is how many degrees of freedom can be independently controllable [18]. To achieve complete control over all degrees of freedom without experiencing coupling effects, drones must possess a minimum of three thrusters aligned to orthonormal axes [28]. Hence, conventional drones are referred to as "under-actuated" due to their thrust vectors being entirely directed along the $\boldsymbol{Z}$ axis. Three rotations and the $\boldsymbol{Z}$ axis translation are controllable independently, whereas the $\boldsymbol{X}$ and $\boldsymbol{Y}$ axis translations depend on those four motions [30]. Before formulating the equations of motion, it is necessary to present two fundamental concepts: Newton's Second Law of Motion and the Derivative of a Vector.

2.11.1. Newton's Second Law of Motion

The rate of change of an object's momentum concerning time is directly proportional to the applied force [59]. Non-zero net force and moment generate translational and rotational acceleration inversely proportional to mass and mass moment of inertia, as shown in Table 2.3 below. In the context of aerial vehicles, it is postulated that

both mass and inertia remain constant over time [15].

Table 2.3. Newton's Second Law of Motion in non-rotary coordinate systems.

$\boldsymbol{F} = \frac{d(m\boldsymbol{V})}{dt} = m\frac{d\boldsymbol{V}}{dt} = m\boldsymbol{a}$	$\boldsymbol{Q} = \frac{d(I\boldsymbol{\omega})}{dt} = I\frac{d\boldsymbol{\omega}}{dt} = I\boldsymbol{\varrho}$
Translational Subsystem	Rotational Subsystem

2.11.2. Derivative of a Vector

A derivative refers to the instantaneous variations of any characteristic with respect to time. The existence of both magnitude and direction in a vector implies that it can be differentiated through changes in any of two [17]. The definition of the derivative operation for a vector $\boldsymbol{A}$ representing any kinematic characteristic in a rotating frame is given by

$$\frac{d\boldsymbol{A}}{dt} = \dot{A}\hat{e}_A + \boldsymbol{\omega} \times \boldsymbol{A}, \tag{2.28}$$

where $\hat{e}_A$ is a constant direction vector of $\boldsymbol{A}$ and $\boldsymbol{\omega}$ is the rotational velocity of the same vector [31]. The first term in the derivative operation above is a constant direction vector that emerges when the magnitude changes. The second term is a constant magnitude vector that appears when the direction changes. Drones are subject to forces, and moments originate in VC-NED-CS and BF-CS; that is why the equations of motion can be expressed in either of them [20]. Since sensors are mounted to the chassis and measurements are performed relative to BF-CS, equations of motion are constructed concerning it [54]. The equation of motion for the rotational subsystem is dynamic moment balance, whereas it is dynamic force balance for the translational subsystem.

2.11.3. Equations of Motion for Translational Subsystem

Taking BF-CS as the reference coordinate system and the I-NED-RF as the reference frame, the equation of motion for the translation subsystem can be written

as

$$m(\dot{V}\hat{e}_V + \boldsymbol{\omega} \times \boldsymbol{V}) = R_0^1 \boldsymbol{G_W} + \boldsymbol{T_P} + \boldsymbol{D_W} + R_0^1 \boldsymbol{F_C}, \tag{2.29}$$

where m represents the mass of the aerial vehicle, $\boldsymbol{V}$ denotes the translational velocity, $\hat{e}_V$ stems for unit direction vector of $\boldsymbol{V}$ and $\boldsymbol{\omega}$ is the rotational velocity of body [17]. R_0^1 depicts the rotation matrix that maps a vector from VC-NED-CS to the BF-CS, and $\boldsymbol{G_W}$ represents the weight of the vehicle. $\boldsymbol{T_P}$ signifies the propellers' total thrust force directed in the $\boldsymbol{Z}$ direction, and $\boldsymbol{D_B}$ represents the drag force on the body resulting from the relative wind speed. Lastly, $\boldsymbol{F_C}$ denotes the Coriolis force acting on the drone's translating body due to the Earth's angular velocity. Note that the left side of the equation represents the derivative of a velocity vector defined in BF-CS [31]. Propeller thrust $\boldsymbol{T_P}$, wind drag $\boldsymbol{D_W}$, and Coriolis forces $\boldsymbol{F_C}$ are defined below.

The thrust generated by a propeller can be expressed as the product of its square of angular velocity Ω^2 and a thrust coefficient k_F, just like given in Section 2.8. The total thrust force of a drone, where i represents the number of rotors can be given as

$$\boldsymbol{T_P} = \begin{bmatrix} 0 \\ 0 \\ -k_F \sum_{all} \Omega_i^2 \end{bmatrix}. \tag{2.30}$$

Multiplying the dynamic pressure P with the normal surface area exposed to the flow and the drag coefficient yields the drag force acting on the drone fuselage as

$$\boldsymbol{D_W} = \begin{bmatrix} \frac{1}{2}\rho V_X^2 S_X C_{DX} \\ \frac{1}{2}\rho V_Y^2 S_Y C_{DY} \\ \frac{1}{2}\rho V_Z^2 S_Z C_{DZ} \end{bmatrix}, \tag{2.31}$$

where ρ is the air density, C_{Di} is the drag coefficient in the $\boldsymbol{i}$ direction, and S_i is the block area normal to airflow [4]. When drones possess linear velocity, they experience the Coriolis force due to the Earth's rotation [54]. Coriolis force is insignificant for drone movements because Earth's rotational velocity Ω_E is too modest (7.29E-5 rad/s) [60].

Finally, one can write the equation of motion for drones as

$$m\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = R_0^1 \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -k_F \sum_{all} \Omega_i^2 \end{bmatrix} + \begin{bmatrix} \frac{1}{2}\rho V_X^2 S_X C_{DX} \\ \frac{1}{2}\rho V_Y^2 S_Y C_{DY} \\ \frac{1}{2}\rho V_Z^2 S_Z C_{DZ} \end{bmatrix} - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} mV_X \\ mV_Y \\ mV_Z \end{bmatrix}, \quad (2.32)$$

where p, q, and r rotational velocities of the body. After gathering kinematic properties by referencing BF-CS, they will transformed to VC-NED-CS to make trajectory planning.

2.11.4. Equations of Motions for Rotational Subsystem

Now that the moments causing rotational motion are defined in BF-CS, the equations of motion are written concerning it so that any rotation matrix does not appear in the equations [61]. One can show the dynamic moment equilibrium of moments as the equation of motion for the rotational subsystem below.

$$\boldsymbol{I}\dot{\boldsymbol{\omega}}_{\boldsymbol{b}} + \boldsymbol{\omega}_{\boldsymbol{b}} \times (\boldsymbol{I}\boldsymbol{\omega}_{\boldsymbol{b}}) = \boldsymbol{Q} + \boldsymbol{Q}_{\boldsymbol{p}}, \quad (2.33)$$

where $\boldsymbol{I}$ is the rotational inertia tensor of the drone, $\boldsymbol{\omega_b}$ is the angular velocity vector of the body, $\boldsymbol{Q}$ is the moment vector that thrust forces induce concerning the center of gravity, and $\boldsymbol{Q_p}$ is the gyroscopic torques of rotating propellers during rotations [60]. Since these rotor gyroscopic torques are very small compared to the thrust force-induced moments, they do not need to be considered. Dynamic moment comes from the derivative of the angular momentum vector (left side of the Equation (2.33)) [18]. Before representing the thrust force moment vector $\boldsymbol{Q}$, to clarify the upcoming direction assignments, Figure 2.31 should be given.

Note that moments for the $\boldsymbol{X}$, $\boldsymbol{Y}$, and $\boldsymbol{Z}$ axes are named rolling, pitching, and yawing moments [61]. Then, one can represent the moment vector $\boldsymbol{Q}$ as

$$\boldsymbol{Q} = \begin{bmatrix} U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} -k_F\Omega_1^2\Gamma + k_F\Omega_2^2\Gamma + k_F\Omega_3^2\Gamma - k_F\Omega_4^2\Gamma \\ k_F\Omega_1^2\Gamma - k_F\Omega_2^2\Gamma + k_F\Omega_3^2\Gamma - k_F\Omega_4^2\Gamma \\ k_Q\Omega_1^2 + k_Q\Omega_2^2 - k_Q\Omega_3^2 - k_Q\Omega_4^2 \end{bmatrix}, \quad (2.34)$$

where U_i is the i^{th} virtual control signal. After the elaborations listed, the final equation of motion for the rotational subsystem will be

$$I_{3x3}\begin{bmatrix}\dot{p}\\ \dot{q}\\ \dot{r}\end{bmatrix} = \begin{bmatrix}-k_F\Omega_1^2\Gamma + k_F\Omega_2^2\Gamma + k_F\Omega_3^2\Gamma - k_F\Omega_4^2\Gamma\\ k_F\Omega_1^2\Gamma - k_F\Omega_2^2\Gamma + k_F\Omega_3^2\Gamma - k_F\Omega_4^2\Gamma\\ k_Q\Omega_1^2 + k_Q\Omega_2^2 - k_Q\Omega_3^2 - k_Q\Omega_4^2\end{bmatrix} - \begin{bmatrix}p\\ q\\ r\end{bmatrix} \times \left(I_{3x3}\begin{bmatrix}p\\ q\\ r\end{bmatrix}\right). \quad (2.35)$$

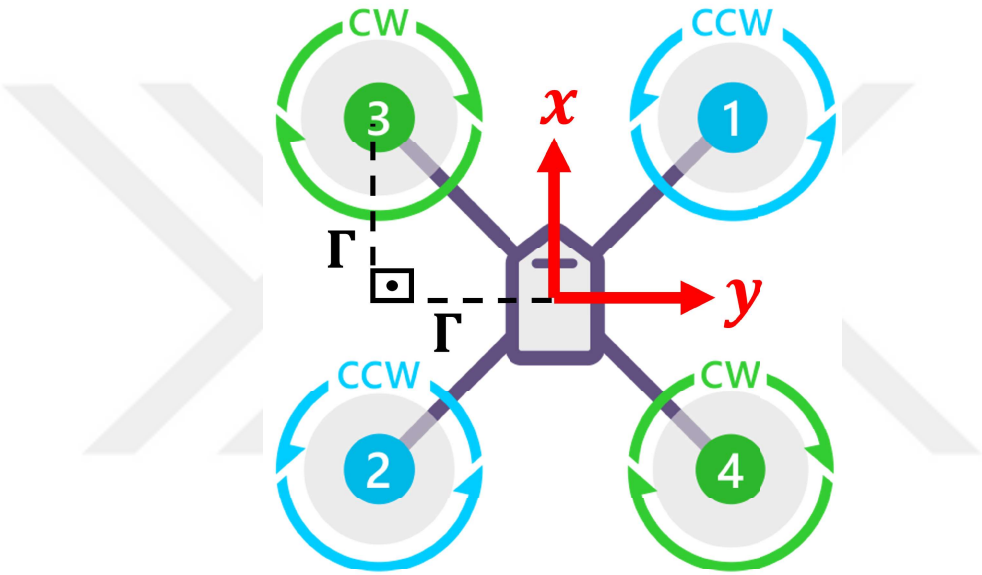


Figure 2.31. ArduCopter algorithm motor orientation and body-fixed coordinate system on the chassis with Γ arm length [62].

A significant concern with rotational motion is where the centers of rotation (pivot) of drone bodies are. The significance of pivots is more evident in large-scale and heterogeneous complex structures rather than in small-sized drones. Determining the point at which the moments induce rotation in the fuselage is of utmost significance in the context of any aircraft. Opposed to conventional assertions, designating this particular position as the center of mass in the rotary wing concept is a superficial and impractical presumption. As a part of the thesis, a fundamental methodology is developed to determine the center of rotation concerning flight conditions and controller interceptions. It was also stated what anomalies would occur if the center of rotation was determined incorrectly without considering the flight conditions.

2.12. Flight Controller

The flight controller is the primary algorithm responsible for the motion of platforms. Once a robust physical system with appropriate actuators has been constructed, a controller must operate this mechanism with high precision and frequency [63]. Flight controller refers to the algorithm that monitors the kinematic states of the system and sends the correct power signals to the actuators to produce the desired movements by assessing these states [64]. The estimation algorithm is responsible for determining the kinematic states of the system by using sensor input. On the other hand, the mission planner generates the target states. The difference between the current and reference states is then sent to the flight controller as an error, eventually translating into appropriate actuator power signals. The frequency of the control loop and the level of sensor precision directly influence the degree of success achieved in the flight [13]. The standard PID technique is often used to control RW-UAVs because of its simplicity and clear structural characteristics [18]. This section primarily concentrates on the generation of actuator signals via the use of state errors. Numerous controllers have been documented in the literature; however, this discussion focuses on the renowned ArduCopter controller. Thanks to its widespread adoption among individuals, the extensive historical development of collective intelligence, and its efficacy as evidenced by the record-breaking duration of flying hours [65]. Despite the insufficient studies for the ArduCopter controller flow, it is extracted via the flight code analysis. Figure 2.32 shows the flow chart depicting the translational controller.

The controller starts by just computing the positional errors. The reference values are obtained from the mission planner: the radio controller, a ground station, or an autonomous flight route. On the other hand, the actual locations are gathered using the state estimate process [66]. To mitigate aggressive signals, a saturation operation is added to confine the calculated errors within a defined range between the states of “Leash Down” and “Leash Up” [18]. Subsequently, these values are sent to the square root controller to convert into target velocities. The square root controller is similar to the PID technique, as it amplifies positional errors, although with the

additional step of taking the square root of those. Experimental results have shown that the settling time is reduced compared to a standard proportional-integral-derivative (PID) block [7]. However, in the wake of very aggressive control signals, the newly produced velocity commands should be limited by "Speed Up" and "Speed Down" values, resulting in saturation. The saturation limits for both the position and speed commands are adjustable in the flight code [67]. Further, it is possible to provide or include arbitrary velocity desires since flight modes may need direct intervention in velocity, in addition to indirect velocity targets [68].

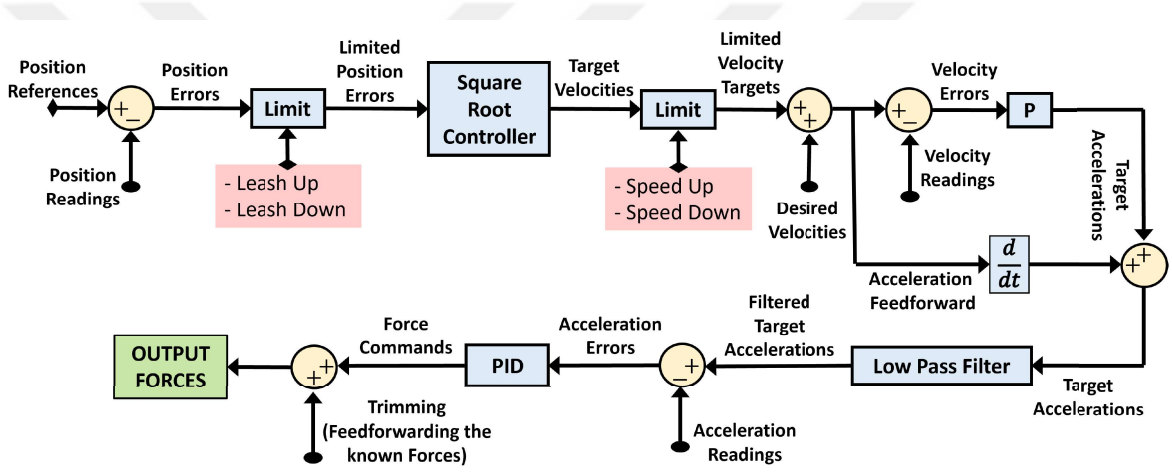


Figure 2.32. Translational controller in ArduCopter algorithm.

Next, the velocity feedback obtained from the sensors is subtracted from the target velocities to determine the velocity errors. The "P block" employs a multiplication operation to create target accelerations based on the velocity errors [28]. To reduce settling time, target accelerations should be supplied forward by derivatives of target velocity [69]. A low-pass filter is used to smooth the updated acceleration targets because the dynamic system already has a natural frequency that can change from 0 to 10 Hz (for RW-UAVs). It is best not to feed the system because it cannot respond to those fast calls [57]. Furthermore, without this filter, the generation of high-frequency directives would result in the motors attempting to draw excessive currents, leading to a decrease in power economy and compromised physical stability [63]. In the meantime, the platform's linear accelerations must be subtracted from the target accelerations to calculate the acceleration errors. Then, this error is transmitted to the PID controller

block to generate the necessary forces along three orthogonal axes [7]. To lessen the controller load and enhance responsiveness, it is recommended to include a feed-forward mechanism into the system by transmitting the permanent forces operating on the platform as trimming inputs. This approach effectively prevents integral accumulation [66]. Such permanent forces may be exemplified as wind forces and weight.

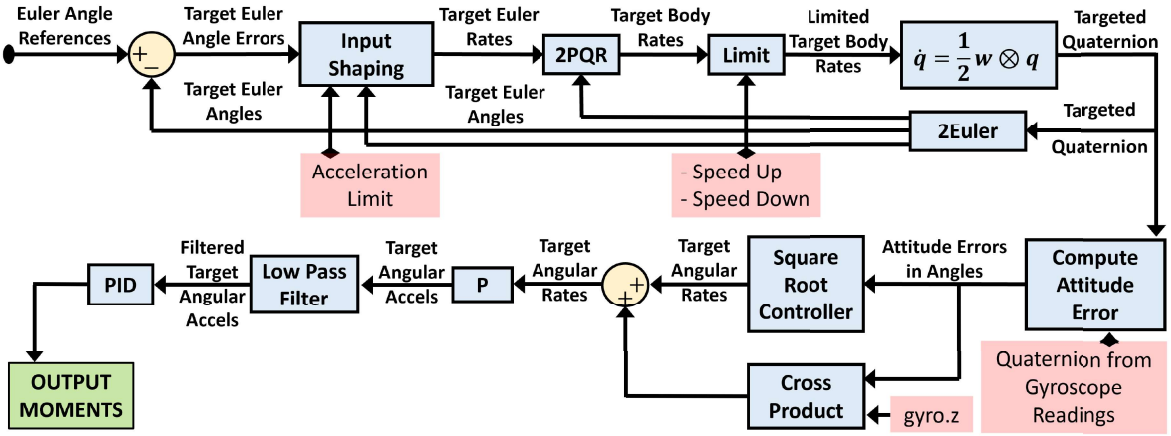


Figure 2.33. Rotational Controller in ArduCopter algorithm.

Calculating the angular position errors, as illustrated in Figure 2.33, is the first step for the rotational controller. However, different than usual, the state error in the rotational controller is identified via a somewhat intricate procedure. The information about the angular state is held in the quaternion, and the angular defects are obtained by subtracting the quaternions [22]. Due to this rationale, inside the scheme, the first step involves constructing the desired quaternion based on Euler angle commands. The present state quaternion is derived from the angular velocities from the gyroscope measurements. Following the target Euler angle errors, input shaping and saturations are undertaken to saturate and reshape such mismatches effectively [29].

The input shaping block outputs Target Euler Rates specified in the VC-NED-CS. The conversion of these rates to [p q r] is completed as stated in Section 2.7 [68]. Subsequently, the above-mentioned desired quaternion will be introduced to the controller. The evaluation of the attitude error may be accomplished by subtracting the current quaternion from the desired quaternion [13]. The angular position error vector

undergoes an instantaneous transformation into body angular errors described in BF-CS. The angular errors are sent towards the square root controller, which generates target body angular rates [24]. Like the translational controller, the desired angular accelerations are subjected to a low-pass filter. Following the implementation of the Proportional-Integral-Derivative (PID) block, three moments are acquired along three perpendicular axes that form an orthonormal basis [21]. It should be noted that the inertia tensor is not included here. The ArduCopter system incorporates compensation for these values inside its control allocation or motor mixing process. The best controller configurations may be determined by analyzing motion graphics to improve precision and efficiency [28]. Assume that the rise time required for the system to reach 97 percent of the command becomes excessively long, and the system exhibits clumsiness during sudden movements. In that case, augmenting the proportionate gain to enhance its aggressiveness is advisable, Figure 2.34.

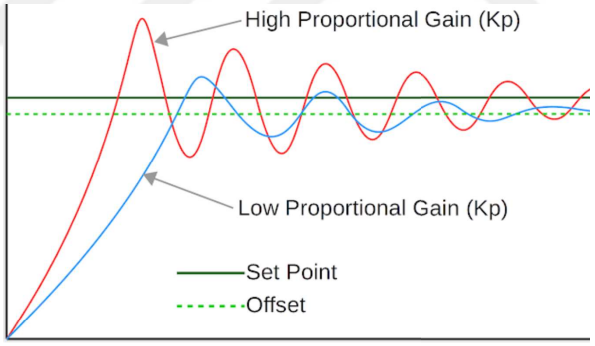


Figure 2.34. Kp tuning procedure. Low and high Kp values have characteristic behaviors.

Increasing the proportional gain in a controller system also accelerates the response time of the loop by providing instantaneous reactions. In this scenario, augmenting the loop frequency and the derivative gain is crucial to prevent overshoots [69]. An excessively high value of derivative gain has the potential to cause oscillations characterized by increased frequencies and decreased amplitudes. By utilizing the method of integral accumulation, it becomes feasible to counterbalance the influence of these steady external factors. Hence, the behavior of steady-state error directly indicates the

accuracy of integral gain. Effects of controller gains are tabulated in Figure 2.35. If the forces and moments mentioned above are meant to have a continuous and lasting effect, it is necessary to counterbalance them through trimming. Trimming acts as a feed-forward reflex mechanism [30]. Aggressive reactions to random external perturbations need integral accumulation around zero. Rapid integral accumulation changes on the positive and negative sides ease fast reactions. Nevertheless, due to the long-term perturbations, integral accumulation may increase for extended periods. For this kind of issue, anti-windup techniques improve integration [63].

Parameter Increase	Rise Time	Overshoot	Settling Time	Steady-State Error
Kp	↓	↑	Small Change	↓
Ki	↓	↑	↑	Great Reduce
Kd	Small Change	↓	↓	Small Change

Figure 2.35. Complete descriptions of how controller gains affect step responses of a controller.

In addition to PID algorithm design and gains, flight health depends on controller loop frequency, estimation architecture, PWM generation and sending process, ESC drive mode, processor clock speed, vibration severity, low and high pass filters' cut-off frequencies, actuator dynamics, possible process delays, and sensor qualities [7]. The flight controller produces virtual control signals (thrust, roll moment, pitch moment, yaw moment) upon completing a control cycle. The angular speed of each rotor and the propeller is determined based on these virtual control signals. If calculated revolutions exceed the rotor limits, they cannot be achieved. Realized virtual control signals will eventually differ if actuator behavior is changed. As a part of this thesis, a control allocation algorithm that solves the actuator saturations of quad-copters based on the underlying physical cause is introduced. Simulation tests of this technique are carried out and compared to traditional methods.

3. ADVANCEMENTS

The assertion that the center of mass is the pivot point for the objects in air has been erroneously integrated into the rotary wing UAV concept. The study here has shown that the center of rotation is the zero force point. Displacement constraints change both the zero force point and the center of rotation. Approaching the control allocation mathematically obscured the physics behind actuator saturations and prevented reaching the optimal solution. Moment needs at high Euler angles are the leading cause of actuator limit excesses. These moment requirements at angular position limits are only for attitude neutralization. Therefore, prioritizing moments in actuator saturations can solve the underlying cause. Considering the opportunities offered by rotary wing platforms (such as gimbal-like rotations and translations), it is possible to detect the depth of environmental objects with a fixed single camera. Depth detection with a single camera set to the aircraft was deemed extravagant, and establishing a technique for this was avoided. However, studies here have resulted in a handy and straightforward method based on vector derivatives. These three novel techniques are described in detail below.

3.1. The Center of Rotation and Its Accurate Determination

The center of mass is generally accepted as the rotation pivot of aerial vehicles [4]. Many working groups use this information without knowing the physics on which it is based and without questioning whether it is suitable for their concepts. First, why the center of mass is the center of rotation is explained, and then under which conditions it can change is given.

The center of mass and the principal axes where rotation occurs can be found by diagonalizing the inertia tensor [70]. The eigenvectors ($\hat{e}_1$, $\hat{e}_2$, $\hat{e}_3$) of the inertia matrix, which can be built around any coordinate system, are the principal $\boldsymbol{X}$, $\boldsymbol{Y}$, and $\boldsymbol{Z}$ axes in which mass is distributed evenly. Then, the eigenvalues (λ_1, λ_2, λ_3) are the

moments of inertia (I_{xx}, I_{yy}, I_{zz}) of the aircraft, while the products of inertia (I_{xy}, I_{xz}, I_{yz}) are mostly zero due to symmetric orientations. Eigenvectors can be determined by expressing the angular momentum vector ($\boldsymbol{H_G}$) as an eigenvalue problem

$$\begin{gathered}\boldsymbol{H_G} = \begin{bmatrix} H_{Gx} \\ H_{Gy} \\ H_{Gz} \end{bmatrix} = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{yx} & I_{yy} & -I_{yz} \\ -I_{zx} & -I_{zy} & I_{zz} \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}, \\ \begin{bmatrix} (I_{xx} - \lambda_i) & -I_{xy} & -I_{xz} \\ -I_{yx} & (I_{yy} - \lambda_i) & -I_{yz} \\ -I_{zx} & -I_{zy} & (I_{zz} - \lambda_i) \end{bmatrix} \begin{bmatrix} e_{ix} \\ e_{iy} \\ e_{iz} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix},\end{gathered} \tag{3.1}$$

where λ_i is the i^{th} eigenvalue and $\hat{e}_i$ is the corresponding eigenvector. No matter how the inertia matrix is written, eigenvalues and eigenvectors never change. With a diagonalization study, the platform's principal axes and center of mass can be easily found [58]. The real question is, under which conditions is the center of rotation for aircraft the center of mass? Contrary to the unquestioned acceptance of the literature, rotary-wing drones need to judge this phenomenon on their own. Despite several studies on center of mass estimation, there is little research about the pivot and center of mass correlation. Without providing any technical rationale, every study unequivocally asserts that the center of mass is the pivot point for rigid objects in space or air [60] [71]. Nevertheless, the lack of causality hinders comprehension and the ability to enhance the situation.

Alexander et al. [72] investigated the rotational dynamics of a rigid body containing movable objects. The Amontons-Coulomb Law was employed in modeling the mobile mass, analogous to the fuel in fixed-wing aircraft. The non-zero velocities of the moving mass around the body result in instantaneous rotations. It has been said that the center of mass, which changes with the displacement of the moving object, also changes its center of rotation. Then, the controller leaves the successful operating range to which it is tuned and begins to drive the aircraft poorly. The final view is that the center of rotation is the center of mass, but it should be noted that the aircraft examined here is a fixed-wing aircraft, not a rotary wing.

Manshadi et al. [71] endeavored to estimate the mass centroid of an aircraft solely using kinematic measures. A simulation environment was built utilizing a rotary wing aerial vehicle's 6 DOF (degree of freedom) dynamic model, acknowledging that the gyroscope and accelerometer exhibit exact accuracy. Then, the authors proposed using stochastic processes like the Kalman filter and Monte Carlo analysis to enhance real-life IMU estimations. Since the kinematic parameters of the 6 DOF dynamic model can be readily correlated with the readings from the accelerometer and gyroscope, the identification of the center of rotation is straightforward. The pivot point is again the center of mass, but the system moves without motion constraints.

From another point of view, "The Drone Pendulum Fallacy" posits that the system can achieve stability through natural pendulum motion, provided that the center of mass of the rigid body is significantly lower than the actuators [73]. The crucial mistake in this approach is to claim that the center of rotation is somehow in the middle of the thrust plane. In this way, lowering the center of mass will cause natural pendulum motion and increase system stability. However, just as the theory's assumption was wrong, so was the theory itself [4].

Two guiding questions for correctly determining the center of rotation of rotary wing drones are as follows:

(i) Where is the center of rotation of solid objects subject to moments in space when there are no restrictions on displacements (see Figure 3.1)? Why?
(ii) Because the flight controller may impose displacement restrictions, can drones change their center of rotation while they are in air?

Translation in which the acceleration is perpendicular to the velocity vector and the magnitude of both velocity and acceleration does not change is called rotation [4]. In the interim, the position traces a circular path and sweeps the angular displacement of 2π radians over time, denoted as t. The rotational motion's momentum, power, and energy can also be expressed through translational motion. Moreover, the forces on

the object in rotational motion should create acceleration perpendicular to the velocity. The velocity and acceleration should not change, and the net force should be zero at the rotation point since it is not accelerating. Based on these deductions, the first query can be resolved because the rotation point has no feature other than being the "Zero Force Point (ZFP)".

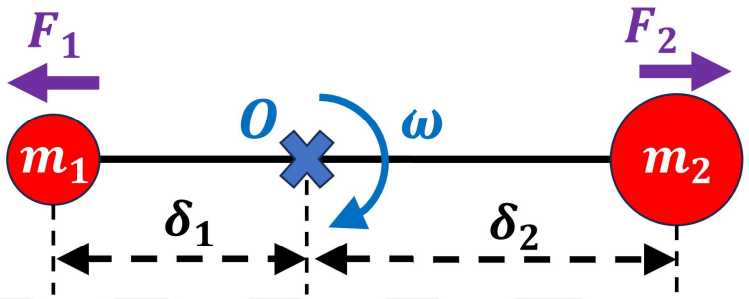


Figure 3.1. Free body diagram of a rotating stick in space. There are no displacement restrictions. It features concentrated masses on each side for easier formulations.

Figure 3.1 depicts a stick with a mass $m_1 + m_2$ rotating in space without motion restrictions. To make the modeling more comprehensible, distributed loads at the left and right of the rotation point are concentrated on their sides as m_1 and m_2. O represents the arbitrarily selected center of rotation, ω is the angular velocity, and δ_1 and δ_2 depicts the masses' distances to the pivot. In space, if a free rod without displacement constraints is exposed to a moment with no friction or damping, it will begin to rotate. For the "O" point to be "ZFP," the sum of the centripetal forces written here must be zero. The summation of centripetal forces and the required condition for the net force to be zero is

$$\begin{aligned} m_1\omega^2\delta_1 - m_2\omega^2\delta_2 = 0, \\ \frac{\delta_1}{\delta_2} = \frac{m_2}{m_1}. \end{aligned} \tag{3.2}$$

The required condition is the product of the masses, and their respective distances to the center of rotation must be equal. This definition is also the description of the center of mass. Thus, under moment actuation, rigid objects in space rotate around their center of mass (answer to the first question).

Theorem 3.1. *The center of mass is an object's zero force point when it rotates freely in air or space. However, displacement restrictions might alter the ZFP.*

Proposition 3.2. *When objects in the air can limit their movement to a specific plane or point, they change their center of rotation.*

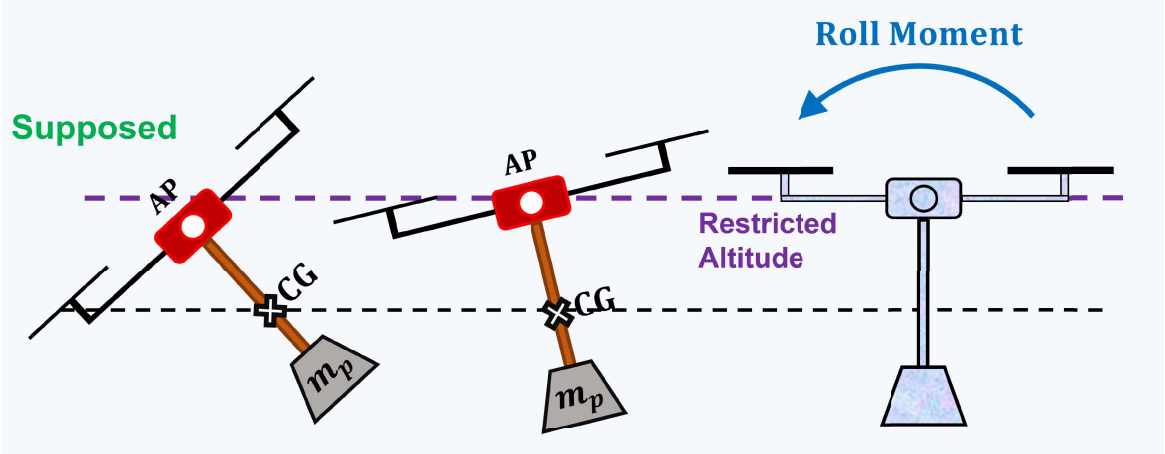


Figure 3.2. A drone with a lower gravity center and vertical displacement restriction due to flight controller response. The supposed scenario is shown here. The body is rotated around the center of mass.

Extra payload is represented by m_p, "AP" denotes the autopilot location, and "CG" is the center of gravity. While commanding a moment, the flight controller does not wish to read any translational change. This occurs if the center of mass known to the flight controller changes by adding an unexpected load to the platform. Because IMU sensors are not placed at the center of rotation, they detect rotational and translational motion during pure rotations. After that, the flight controller issues directives to revert the undesirable vertical translations. Propellers will be ordered to produce the desired torque and stop vertical deflection. This creates the vertically constrained horizontal slider as shown in Figures 3.2 and 3.3. Before these foundations, it is supposed that the center of rotation will be CG, as shown in Figure 3.2. However, with the horizontal slider displacement restriction, it can be seen that the rotation will be around the AP point, as in Figure 3.3.

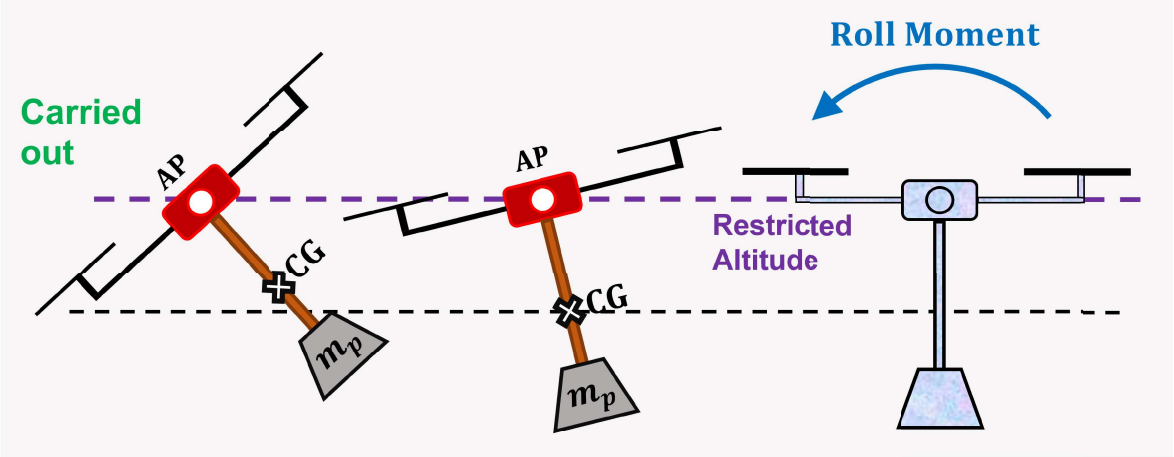


Figure 3.3. A drone with a lower gravity center and vertical displacement restriction due to flight controller response. The carried-out scenario is demonstrated here. The body is rotated around the restriction point.

Corollary 3.3. *The zero force point, i.e., the rotation pivot, changes when the flight controller applies movement restrictions to the RW-UAVs.*

Contrary to expectations, the fact that the pivot is AP rather than CG brings about significant anomalies, as listed below.

- Unexpected translational motions
- Increased inertia
- Swing movement and resonance caused by ill-suited control coefficients
- Motors operating at the limit due to increased state errors

The solution is to ensure that the flight controller receives correct sensor feedback after the center of mass is reported via offset. Otherwise, the expected kinematic situations after the command will be very different from the real ones, and the controller will cause instability. Contrary to the unquestioning acceptance of many working groups, the center of rotation is always ZFP, not the center of mass. The center of rotation may shift due to displacement limitations the flight controller sets. Thus, it is essential to get ready ahead for this possibility.

3.2. Moment Prioritized Control Allocation

Numerous research in the literature focus on the control allocation of RW-UAVs to address actuator loss [74] [75] [76], mitigate rotors' limit excesses [77] [78], and ensure optimized operating conditions within overactuated systems (such as hexacopters, octocopters, etc.) [79] [80]. Traditional quad-copters possess four degrees of freedom that are independently controllable, as seen in Figure 3.4. Conversely, some examples of these vehicles may own several rotors and degrees of freedom that deviate from the standard configuration. Considering all, a control allocation algorithm may address at least one of three fundamental issues [69].

- Modifying the revolutions to ensure they fall within the physical limits
- Conscious weighting of rotors in over-actuated systems
- Redistribution of the control signals in case of a rotor failure

Given that quad-copters have four control mapping equations, each corresponding to an unknown rotor speed, any saturation would inevitably suppress control signals. The study here and the control allocation studies in the literature adopt the most strategic manner while re-constructing control commands of saturated quad-copter rotors. Applying enough cut / adding to all rotors individually to eliminate overshoots / undershoots without considering the direction of the control signals to prioritize thrust (Force Prioritized Control Allocation (FPCA)) [77] and re-shaping the control signals by giving relative importance to control signals (Partial Control Allocation (PCA)) [80] are the two most dominant techniques embraced in the literature (see Figure 3.5). However, these approaches failed to detect the physical principles that caused the actuators to reach their limits. Moreover, both methods have features that may not be desired when integrated into the flight controller (such as loops that have uncertain duration). Quad-rotors are designed to have thrust at least twice their weight. So how can limit exceeds be observed? Getting to the root of the problem with the study presented here has led to the emergence of a much more robust technique with a more straightforward implementation.

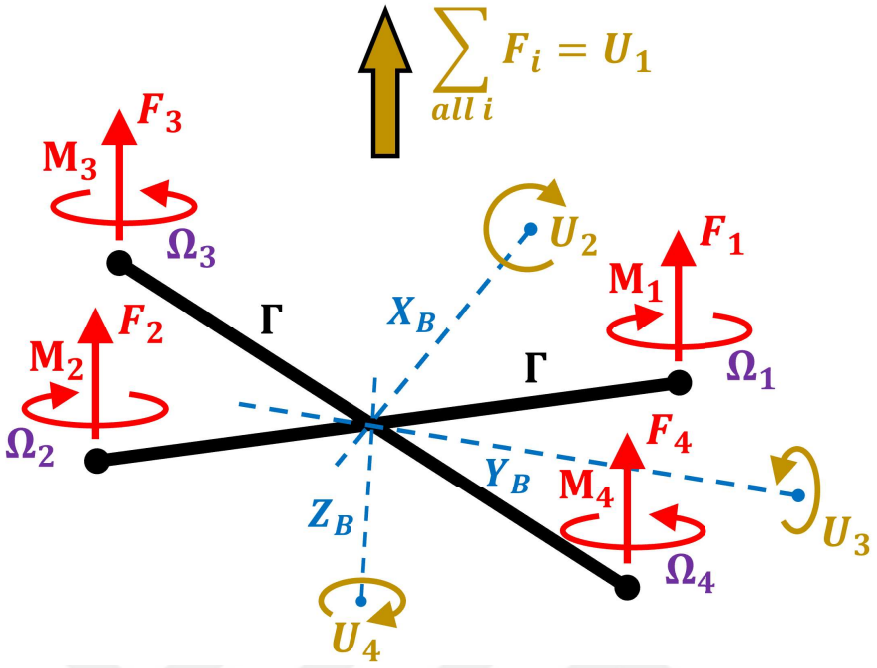


Figure 3.4. Kinetic diagram of a typical quadcopter. Virtual control signals are depicted separately.

Total thrust force (U_1) and three moments (U_2 , U_3 and U_4) are illustrated in Figure 3.4. Their further mathematical representations are given as

$$U = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} \text{thrust} \\ \text{roll moment} \\ \text{pitch moment} \\ \text{yaw moment} \end{bmatrix} = \begin{bmatrix} F_1 + F_2 + F_3 + F_4 \\ -F_1\Gamma + F_2\Gamma + F_3\Gamma - F_4\Gamma \\ F_1\Gamma - F_2\Gamma + F_3\Gamma - F_4\Gamma \\ M_1 + M_2 - M_3 - M_4 \end{bmatrix}, \tag{3.3}$$

where Γ is the moment arm of the propeller thrust forces (F_i) and M_i is the shaft torque of i^{th} rotor. They are fully controllable through propellers' revolutions [13]. The eventual control mapping equation, including four linear independent equations, is (referencing Figure 3.4)

$$\begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} k_F & k_F & k_F & k_F \\ -k_F\Gamma & k_F\Gamma & k_F\Gamma & -k_F\Gamma \\ k_F\Gamma & -k_F\Gamma & k_F\Gamma & -k_F\Gamma \\ k_Q & k_Q & -k_Q & -k_Q \end{bmatrix} \begin{bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \end{bmatrix}. \tag{3.4}$$

After each control cycle, the U matrix is multiplied by the inverted system performance matrix (K^{-1}) to give the square of the rotor revolutions' matrix (Ω^2). To find unknown rotor revolutions, the process introduced in Equation (3.4) is usually reversed as

$$\begin{bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \end{bmatrix} = \begin{bmatrix} k_F & k_F & k_F & k_F \\ -k_F\Gamma & k_F\Gamma & k_F\Gamma & -k_F\Gamma \\ k_F\Gamma & -k_F\Gamma & k_F\Gamma & -k_F\Gamma \\ k_Q & k_Q & -k_Q & -k_Q \end{bmatrix}^{-1} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix}. \tag{3.5}$$

Mathematical functions or equations always provide numerical outputs referred to as "results." However, the user must ensure that these numerical values are physically valid. The fundamental theorem of algebra posits that a polynomial of degree n has precisely n roots. They may exist as positive or negative and manifest as real or imaginary values. However, a rotor cannot undergo negative or complex revolutions. A straight linear equation solution method may provide unattainable rotor outcomes in this context.

The control allocation algorithm developed here aims to re-construct control signals when they exceed the actuator limits. The first step should be to find the line of behavior that leads to limit violations. Any approach formulated without comprehending the inherent characteristics of the motion in which the actuators are saturated would merely serve as a mathematical dodge. There are six basic movements in rotary wing drones, where actuators operate differently.

- Take-Off
- Hovering
- Climbing and descending
- Creating a tilted attitude
- Staying tilted to translate
- Retracting tilted attitude

Considering these platforms are designed to have a thrust at least twice their weight, the actuators cannot get saturated during take-off, climbing, and descending. Even though 30-degree roll and pitch angles are attained, 75 percent of the thrust balances weight. Thus, creating a tilted attitude and staying tilted to translate does not require actuators to push their limits. However, if angular acceleration is requested at high Euler angles, limit excesses are observed while retracting the tilted attitude. The moments asked at high Euler Angles are associated with the occurrence of limit violations. However, the over-limit situation will disappear as soon as the moment needs are cleared and the attitude is neutralized. Since the moment needs are for very short periods, and the angular subsystem is very fast compared to the translational motion, sacrificing force will result in minimal losses [4]. Therefore, prioritizing moment commands to refine the attitude in limit saturations will be the fastest solution to eliminate the root cause. Simulation tests are used to compare the Moment Prioritized Control Allocation (MPCA), Force Prioritized Control Allocation (FPCA), and Partial Control Allocation (PCA). How much altitude is lost when thrust is waived has also been shown. Figure 3.5 indicates how these three control allocation methods change raw commands (R_i, i^{th} revolution) by weighting them relatively. No matter the command vector direction, FPCA maintains thrust. PCA prioritizes thrust while preserving the command vector direction. MPCA, the new approach, maintains command vector direction by fully prioritizing the moments.

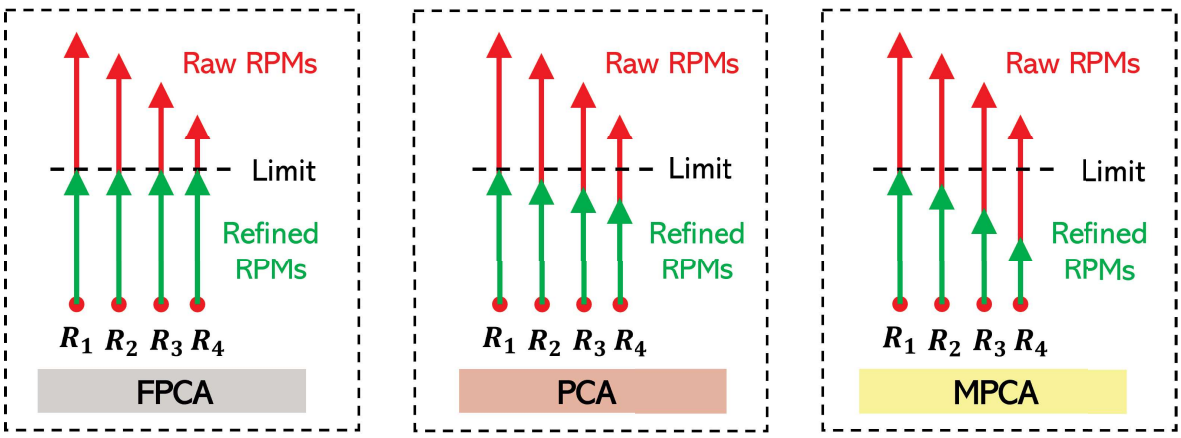


Figure 3.5. The comparison of three different control allocations that will be tested.

The algorithm developed here assumes that the propellers have only angular velocity limits and that their angular acceleration is not restricted. Since rotor dynamics are much faster than platform body dynamics, rotor dynamics were ignored [69]. Commanded angular velocities were assumed to be achieved immediately. The algorithm for Moment Prioritized Control Allocation (MPCA) is outlined below. Algebraic operations are calculated instantly in each controller cycle, with no undefined interval or "for loop."

Theorem 3.4. *Quad-copters are designed to have a thrust force of at least twice their weight. If motors must perform above their maximum capacity, it is because of the moment the command is given when the platform is inclined at large Euler angles.*

Proposition 3.5. *If the moment command still comes when the platform is tilted to the limits and has high Euler angles, this is only to bring the angular position to neutral. Since the root cause of saturation disappears when the neutral position is reached, moments should be given priority in cases of limit excesses.*

According to the proposition, when the results of Equation (3.5) surpass the rotor limits, the virtual command signals are re-constructed by prioritizing the moments. The implementation steps of the new control allocation based on this rationale are as follows.

(i) Initially, the first solution should be found over Equation (3.5) and checked whether there is any excess

$$\forall \Omega_i : \Omega_{min} < \Omega_i < \Omega_{max}. \tag{3.6}$$

(ii) If there is an upper limit violation, all values are subjected to a cutoff equal to the surplus of the highest, exceeding

$$\Omega^2 = \Omega^2 - (\max(\Omega^2) - \Omega^2_{max}). \tag{3.7}$$

(iii) If the lower limit is exceeded, an addition large enough to bring the lowest value up to the lower limit is applied to all values

$$\Omega^2 = \Omega^2 + (\Omega^2_{min} - \min(\Omega^2)). \tag{3.8}$$

(iv) Since the control mapping equations are linear, increasing or decreasing the revolutions altogether does not affect moments. However, these changes may reduce thrust force, which can be quickly resolved after the actuators' saturation removal. Post process on Ω^2 leads revolutions to exceed the predetermined range, even if they lie within limits initially. Therefore, the upper and lower boundary corrections need a final individual adjustment as

$$\begin{aligned} \Omega_i^2 &= \Omega_i^2 - (\Omega_i^2 - \Omega_{max}^2 + |\Omega_i^2 - \Omega_{max}^2|) * 0.5, \\ \Omega_i^2 &= \Omega_i^2 + (\Omega_{min}^2 - \Omega_i^2 + |\Omega_{min}^2 - \Omega_i^2|) * 0.5. \end{aligned} \tag{3.9}$$

In contrast to the control allocation methods presented in the literature, which pose challenges in implementation, the algorithm proposed herein offers a straightforward approach supported by robust physics. Moment Prioritized Control Allocation (MPCA) is achieved by applying the provided equations sequentially. The method has high suitability for use in a flight controller due to its absence of loops, elimination of potential zero division, and avoidance of square root operations on negative values.

The motion graphs below illustrate the relative effectiveness of the MPCA algorithm in comparison to the FPCA and PCA algorithms that dominate the literature [77] [81]. During the creation of these visuals, a dynamic model was developed in SIMULINK for a quad-copter with physical parameters specified in Table 3.1. Off-diagonal terms of inertia tensor are all zero due to symmetric structure. The RPM operating range of the selected rotor has been narrowed by 25 percent to make it easier to see limit exceedances. The PCA algorithm prioritized thrust three times over moments when overshoot was present, but moments were not zeroed. The controller, which consists of nested PID blocks, has been tuned manually. Coefficients from the inner angular velocity cycle to the linear position cycle were entered by trial and error method, and the values that followed the most appropriate route were entered. Attitude is controlled in three axes together with the $\boldsymbol{Z}$ position. Movements in the $\boldsymbol{X}$ and $\boldsymbol{Y}$ axes were not separately controlled. Movements in these axes are converted into angular position and thrust requirements.

Table 3.1. Simulation parameters.

Mass	3.5 kg
Inertia Tensor	$\begin{bmatrix} 0.62 & 0 & 0 \\ 0 & 0.62 & 0 \\ 0 & 0 & 0.65 \end{bmatrix}$ kgm^2
Selected Motor and Propeller	U3 KV700 and 12x4.0 CF
k_F	2.2522E-7 $N/(RPM)^2$
k_Q	4.0856E-9 $Nm/(RPM)^2$
Chassis Arm Length	0.35 m
RPM_{max}	8000 revolution per minute
RPM_{min}	4000 revolution per minute
Flight Controller	Cascaded PID
Attitude Compensation for Thrust	Available
Tested Control Allocations	MPCA, PCA, FPCA
SIMULINK Solver	ode23 (Bogacki-Shampine) 400 Hz

The flight path has been specially designed to compare the control allocation algorithms and make accurate analyses within themselves.

(i) Between $0 - 10$ seconds, the quad-copter takes off. No angular motion is commanded.

(ii) Between $10 - 20$ seconds, aircraft keeps altitude stationary at 30 meter. No angular motion is commanded.

(iii) Between $20 - 42$ seconds, the altitude commands oscillate with the ± 10 meter amplitudes. No angular motion is commanded.

(iv) Between $42 - 80$ seconds, ± 30 degrees angular motions are conducted in three axes: roll, pitch, and yaw. No altitude change is requested.

(v) Between $80 - 120$ seconds, combined altitude and attitude maneuvers are given.

The actuators were not saturated, and the aircraft never diverted from its $\boldsymbol{Z}$ path during takeoff, referencing Figure 3.6. Similarly, there are no limit violations or

deviance from the planned trajectory when performing hover and altitude maneuvers. Even though the altitude remains constant at the fourth stage, when commands for angular movement are given, limit exceedances happen at large Euler angles, which leads to the start of route deviations. In the "Attitude Change" region, FPCA is seen to lose control and go off the path (fall), while MPCA and PCA algorithms continue the flight safely, although they sacrifice altitude. Upon analyzing the combined maneuver region, it becomes apparent that the MPCA approach outperforms the PCA technique. According to post-flight analysis, the MPCA algorithm exceeded revolution limits by 20 percent of the flight, while PCA exceeded them by 25 percent and FPCA by 30 percent. When the altitude loss was examined, it came out that the MPCA algorithm had a maximum height loss of 5 meters, which was resolved in 5 seconds.

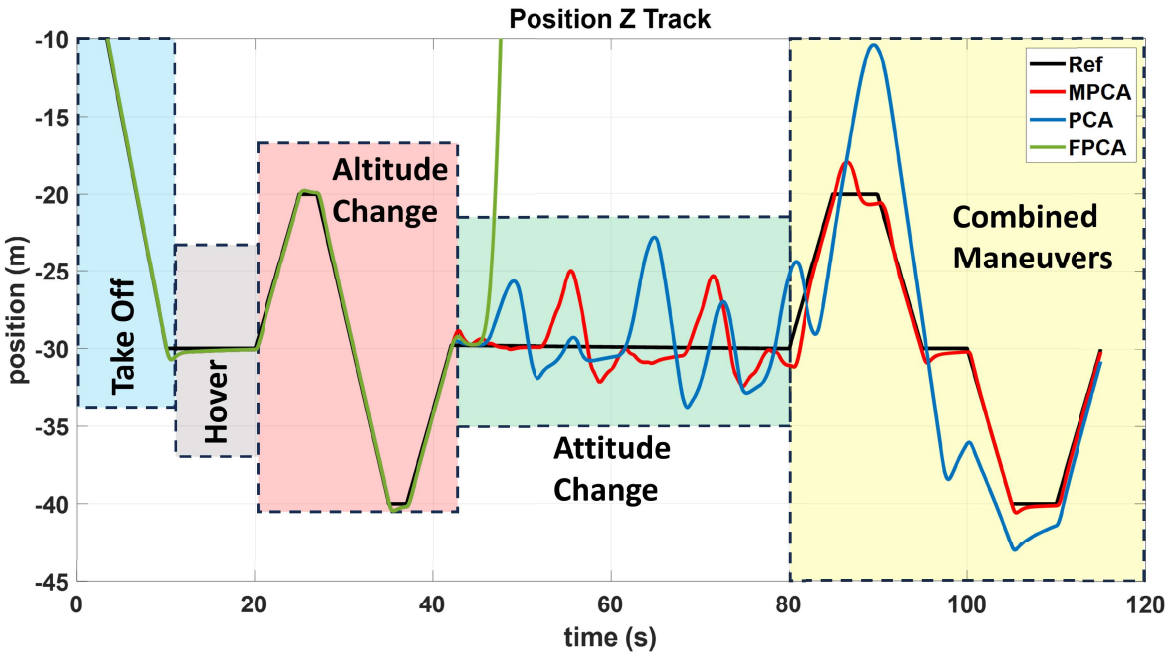


Figure 3.6. The altitude command track performance comparison for three different control allocations.

No limit violations are seen in the first three stages of large-altitude maneuvers. Nevertheless, route deviations are visible in the "Attitude Change" region due to actuator saturations, referencing Figure 3.7. FPCA oscillates and loses control immediately after commands are reached, while MPCA and PCA algorithms consistently maintain

stability. The overshoot examination again shows that MPCA works better than PCA. As an essential point to remember, FPCA does not care about moments, while PCA 0.3 times reduces their intensity, and MPCA directly prioritizes them. One may establish a relationship between a control allocation's success and the importance attached to moments. Since the system is identical in the $\boldsymbol{X}$ and $\boldsymbol{Y}$ axes, Θ track performance is not given.

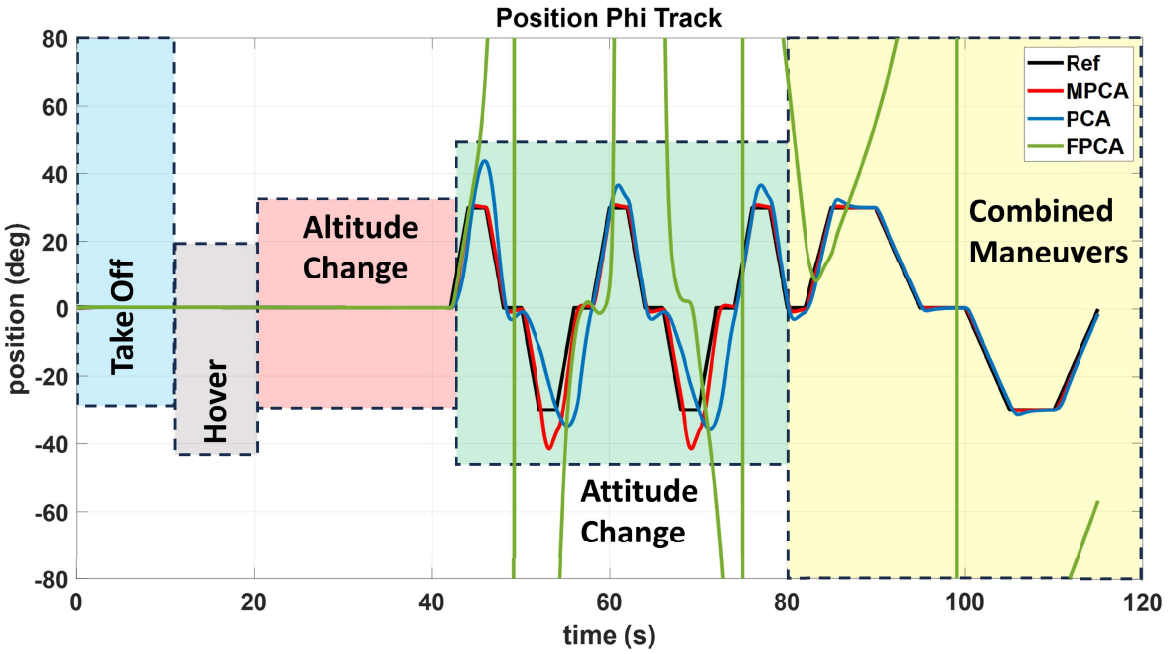


Figure 3.7. The comparison of Φ angle command track performance for three different control allocations.

Because the motors are located quite far from the center of rotation, inertia is more prominent on the $\boldsymbol{Z}$ axis, unlike the angular movement on the $\boldsymbol{X}$ and $\boldsymbol{Y}$ axes. For this reason, the natural frequency of the angular motion in the $\boldsymbol{Z}$ axis is lower and is driven by high controller gains. While the behavior of the control allocation algorithms has not changed, the severities of the upper exceedances have changed. MPCA algorithm is more successful than the Φ angle track, referencing Figure 3.8.

Corollary 3.6. *Moment-prioritized control allocation allows flights to be executed optimally by promptly addressing the primary cause of limit excesses, which is the moment needs given at large Euler Angles.*

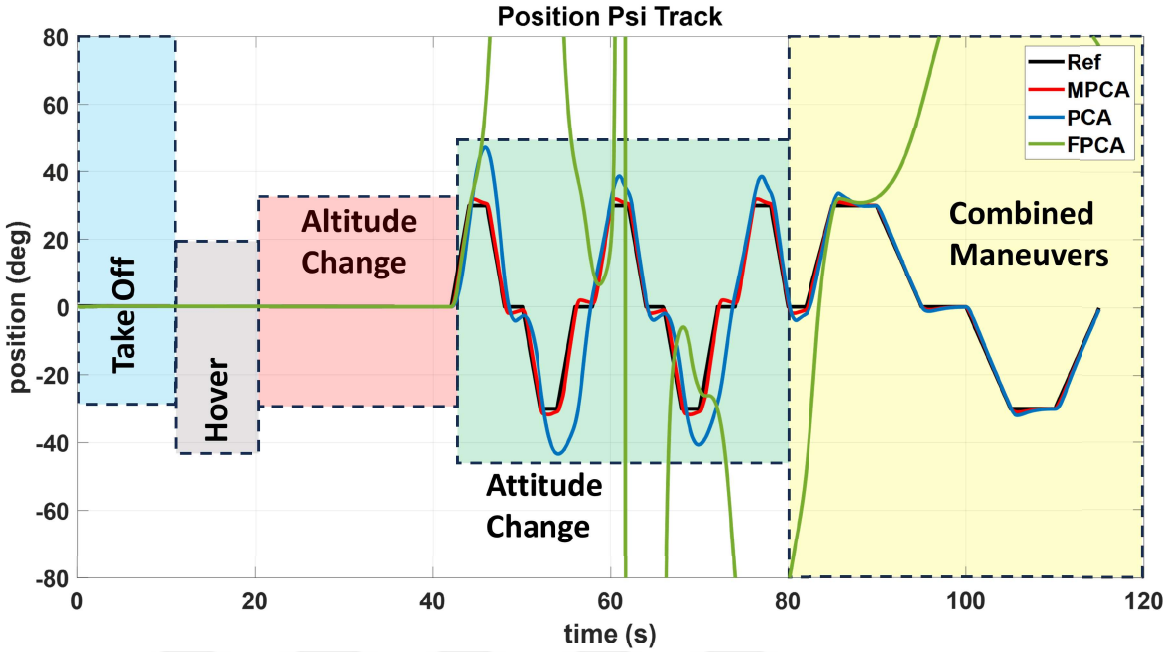


Figure 3.8. The comparison of Ψ angle command track performance for three different control allocations.

As a supplementary improvement, concentrating mass at the center reduces inertia, so moment levels can also drop. However, propeller revolution deviations cause high-frequency angular oscillations when inertia is low. In this case, the best strategy is to increase the inertia to a specific limit (based on a dynamic system) and use the MPCA algorithm.

3.3. Estimating Far Distances using a Fixed Monocular Camera

During surveillance, indoor flight, and object avoidance, drones must detect the distance between surroundings. Range finders such as LiDARs, ultrasonic radars, and visible light cameras can be used to determine depths. Concerning lower establishment price, computational load, and increased working range, visible light cameras are the best candidates. Depth detection with these cameras has been widely studied in the literature [82] [83] [84]. Nevertheless, these studies have been limited to creating a depth map of the immediate environment. Long distances (bigger than 1 km) must also be estimated for route planning, target detection, and aircraft tracking. However,

costly methods (such as advanced LiDARs and sonars) are not preferred for drones, which are popular with their cheap production. There are two main concepts while measuring distance with visible light cameras:

- Single camera, i.e., monocular vision
- Double camera, i.e., stereo vision

The stereo-vision approach achieves a high level of accuracy in determining depth. However, it has significant costs in terms of both mechanical and computational. Installation sharpness of the physical system, calibration equivalence of two cameras, rectification of the photographs taken, correspondence of the same points in different pictures, and reconstruction of the seen environment are severe processing burdens [85] [86]. Lipnickas et al. [82] presented a study showing how to measure distance through stereovision and create a depth map of the immediate environment. In essence, two cameras and a target point create a triangle whose base angles are known since they are the pan angles of the cameras. Gathering distance from stereovision and accompanying triangles has a unique name called "Triangulation" (Figure 3.9).

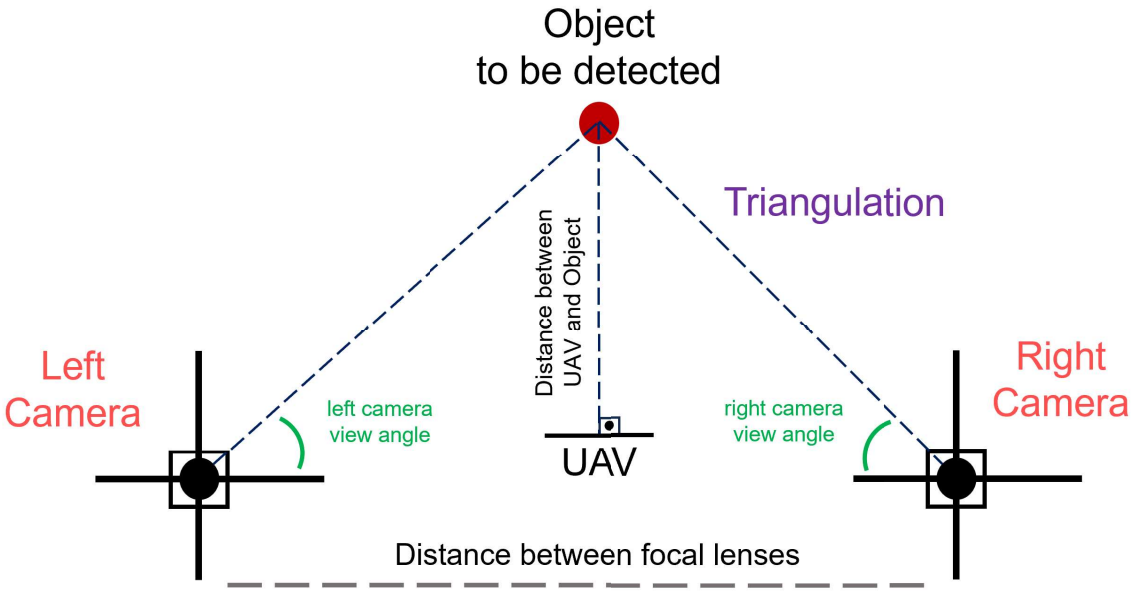


Figure 3.9. Distance calculation over triangulation using the dual camera.

Monocular vision provides distance information by knowing any length over the object in advance or by examining the pixel speed of the target object as the camera rotates. Sabreddini et al. [87], and Chun [88] demonstrated two methods to obtain depth using a monocular camera. Sadreddini and his colleagues [87] used a previously known edge length to estimate the distance, while Chun and his colleagues [88] used a rotating mirror.

Having prior knowledge of the length of an object whose depth needs to be assessed and using revolving mirrors on rotary wing platforms are not practical. This thesis also presents a novel method for accurately measuring long distances using a single camera rigidly fastened to the drone chassis that can be conceptualized as in Figure 3.10. Increased working range, reduced establishment price, no need for prior distance information or movable parts, and lower computational burden are the advantages of this method.

Theorem 3.7. *A fixed point may have translational velocity in the image due to the translation or rotation of the camera, thanks to the vector derivative.*

Proposition 3.8. *The depth can be resolved by providing the camera with known translational and rotational velocities and analyzing the separate speeds of the indicated point in the image.*

By referencing Figure 3.10, S is the area occupied by the object in the picture; A and B are the two furthest points of this area; M is the midpoint of the edge drawn with points A and B; $\boldsymbol{P}$ is the position vector drawn from the camera to the object. The $\boldsymbol{P}$ is assumed to be unchanged when the length AB or the area S is almost intact. To conduct the kinematic analysis of aircraft, the reference frame in which the observer is located and the reference coordinate set signifying the principal directions must be defined. Then, the algorithm steps follow.

- Reference Frame: Inertial NED Reference Frame
- Reference Coordinate Set: Body Fixed Coordinate System ($X_BY_BZ_B$)

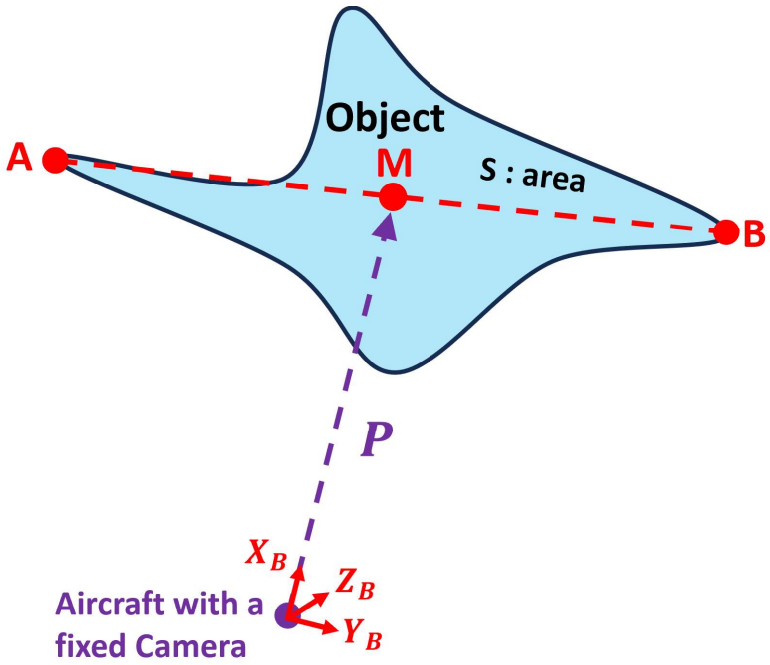


Figure 3.10. Scenario for detecting the depth of a distant object. The algorithm does not have a prerequisite regarding the shape of the observed target.

(i) The first step of the algorithm is completed when the object enters the camera's field of view and generates a point within a pixel region that is nearly stationary, irrespective of the object's posture, shape, or rotational motion.

(ii) Drone displaces through the $\boldsymbol{Y_F}$ with a known translational velocity $\boldsymbol{V_{Dt}}$ quickly (0.5 s). The pixel velocity of point M ($\boldsymbol{U_{Mt}}$) can be related to platform translational velocity ($\boldsymbol{V_{Dt}}$) with a correction factor as

$$\boldsymbol{V_{Dt}} = \mu \boldsymbol{U_{Mt}}. \tag{3.10}$$

Hence, any pixel rate can be turned to real velocity using the μ correction factor, assuming the distance remains constant.

(iii) The lateral motion of the same M point in the image (again in a short period of 0.5 seconds) can also be achieved by the drone yaw rotation. The yawing rate of the platform (r) and the lateral pixel velocity of point M ($\boldsymbol{U_{Mr}}$) due to rotation are known from the gyroscope and image process. With the correction (μ), the lateral velocity of point M ($\boldsymbol{V_{Mr}}$) can also be converted to m/s units as

$$\boldsymbol{V_{Mr}} = \mu \boldsymbol{U_{Mr}}. \tag{3.11}$$

(iv) The fact that point M has lateral velocity as the camera rotates is mainly due to vector derivative (see the Section 2.11.2). Assume that $\boldsymbol{P}$ depicts the position vector of point M, the rotational velocity of the camera (r) and lateral velocity of point M ($\boldsymbol{V_{Mr}}$) are known as

$$\begin{bmatrix} 0 \\ V_{Mr} \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ r \end{bmatrix} \times \begin{bmatrix} P \\ 0 \\ 0 \end{bmatrix}. \tag{3.12}$$

(v) Then, one can write the finalized relation as

$$\begin{aligned} P &= \mu \frac{V_{Mr}}{r}, \\ P &= \mu \frac{\mu U_{Mr}}{r}. \end{aligned} \tag{3.13}$$

Corollary 3.9. *The lateral velocities of a fixed point in the image with camera translation and rotation provide depth detection. The vector derivative expresses the velocity of a fixed point with camera rotation. This velocity is directly proportional to the depth in between.*

4. CONCLUSION

The integration of rotary-wing uncrewed aerial vehicles into various aspects of society is steadily increasing, with a notable impact on transportation and entertainment. Nevertheless, military-oriented aviation and the lack of progress in battery technology impeded the advancement of these vehicles' techniques. However, the desire to integrate rotary wing platforms such as flying cars and VTOL into daily life has made it inevitable to improve the dynamic performance of these vehicles. This thesis also developed three techniques to improve these vehicles' dynamic behaviors. A comprehensive evaluation of the center of rotation revealed that the pivot is always the zero force point since it is not accelerated. On the other hand, the displacement restrictions defined in the system can change the zero force point as well as the rotation center. If the flight controller is not notified of the final pivot point, some anomalies may result in flight failure. The flight controller completes a mathematical loop and produces revolutions for the rotors. The control commands will deviate if these values exceed the actuators' physical limits. In the meantime, continuing the flight by prioritizing the moments will resolve the limit exceedance situation as soon as possible and minimize command loss. Additionally, simulation results have shown that not prioritizing attitude can lead to sudden crashes. Using the vector derivative formulation, a single fixed camera can determine the far distances of environmental objects. The lateral velocity of a fixed point in the rotating camera image is directly proportional to the depth in between. The exact equations derived for this distance determination are presented in the report.

4.1. FUTURE WORK

The techniques adopted in rotary-wing uncrewed aerial vehicles still lag behind their ideal point. Thoroughly criticizing established foundations, learning about new subjects, and changing outdated and incorrect practices can all lead to further technical advancement. Various challenges need to be addressed. Some of the most noteworthy ones are listed here.

- Detailed mathematical model of CCW and CW rotating coaxial propulsion systems
- Disturbed Hall effect of magnetometer during translational motion
- Zero stiffness mechanism for the flight controller to be protected from vibrations
- Rotating actuator plane instead of the whole body to increase aggressiveness by inertia reduction
- Adjustable dihedral angle for increased aggressiveness

Our research on these topics will also continue. Nonetheless, extra efforts will be greatly appreciated.

REFERENCES

1. Kim, J., S. A. Gadsden and S. A. Wilkerson, "A comprehensive survey of control strategies for autonomous quadrotors", *Canadian Journal of Electrical and Computer Engineering*, Vol. 43, No. 1, pp. 3–16, 2019.

2. Raza, Syed Ali, and Gueaieb, Wail, "Intelligent flight control of an autonomous quadrotor", *Motion Control*, Vol. 1, pp. 245–264, 2010.

3. Team, J. R., "Different Types of Drones", `https://www.jouav.com`, accessed on August 27, 2023.

4. Gudmundsson, S., *General Aviation Aircraft Design: Applied Methods and Procedures*, Butterworth-Heinemann, 2013.

5. Idrissi, M., M. Salami and F. Annaz, "A review of quadrotor unmanned aerial vehicles: applications, architectural design and control algorithms", *Journal of Intelligent & Robotic Systems*, Vol. 104, No. 2, p. 22, 2022.

6. Alaimo, A., V. Artale, C. Milazzo, A. Ricciardello and L. Trefiletti, "Mathematical modeling and control of a hexacopter", *International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 1043–1050, IEEE, 2013.

7. Fumagalli, M., R. Naldi, A. Macchelli, R. Carloni, S. Stramigioli and L. Marconi, "Modeling and control of a flying robot for contact inspection", *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3532–3537, IEEE, 2012.

8. World, I., "Drones' History", `https://informativeworld2020.blogspot.com`, accessed on January 11, 2022.

9. TECH, B., "Kizilelma", `https://baykartech.com`, accessed on June 19, 2023.

10. Holda, C., B. Ghalamchi and M. W. Mueller, "Tilting multicopter rotors for increased power efficiency and yaw authority", *International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 143–148, IEEE, 2018.

11. Kayatas, E. and M. Çelik, "The Effects of Propellers Tilted Inward On Hover Rigidity In Rotary Wing UAVs", *European Journal of Science and Technology*, Vol. 38, pp. 122–129, 2022.

12. Aviators, A., "VTOLs Explained", `https://aviationforaviators.com/`, accessed on July 09, 2021.

13. Cazaurang, F., K. Cohen and M. Kumar, *Multi-rotor Platform Based UAV Systems*, Elsevier, 2020.

14. TECH, B., "Baykar IHA", `https://baykartech.com/`, accessed on April 12, 2022.

15. Zulu, A. and S. John, "A review of control algorithms for autonomous quadrotors", *Open Journal of Applied Sciences*, Vol. 14, No. 2, pp. 547–556, 2016.

16. Kayatas, E. and M. Celik, "The effects of dihedral angle on hover rigidity and planar maneuvers in rotary-wing UAVs", *Aircraft Engineering and Aerospace Technology*, Vol. 95, No. 7, pp. 1029–1036, 2023.

17. Chovancová, A., T. Fico, L. Chovanec and P. Hubinsk, "Mathematical modeling and parameter identification of quadrotor", *Procedia Engineering*, Vol. 96, pp. 172–181, 2014.

18. Abdelkhalek, M. A., M. S. El-Demerdash, A. A. El-Tahan and T. N. Dief, "Attitude stability of quadcopter using classic control with angular acceleration", *International Journal of Computer Science and Information Technology and Security*, Vol. 5, No. 4, pp. 325–331, 2015.

19. Deavours, C. A., "The quaternion calculus", *The American Mathematical Monthly*,

Vol. 80, No. 9, pp. 995–1008, 1973.

20. Hamandi, M., F. Usai, Q. Sablé, N. Staub, M. Tognon and A. Franchi, "Design of multirotor aerial vehicles: A taxonomy based on input allocation", *The International Journal of Robotics Research*, Vol. 40, No. 8-9, pp. 1015–1044, 2021.

21. Buchholz, J. M., *Multirotor UAS Sense and Avoid with Sensor Fusion*, Ph.D. Thesis, Embry-Riddle Aeronautical University, 2019.

22. Grzes, M., M. Slowik and Z. Gosiewski, "Multirotor UAV sensor fusion for precision landing", *Aircraft Engineering and Aerospace Technology*, Vol. 91, No. 2, pp. 241–248, 2018.

23. Chou, J. C., "Quaternion kinematic and dynamic differential equations", *IEEE Transactions on Robotics and Automation*, Vol. 8, No. 1, pp. 53–64, 1992.

24. Liu, B., J. Xu, B. Fu, Y. Hao and T. An, "A covariance shaping filtering method for tightly-coupled MIMU/GNSS of UAV", *Aircraft Engineering and Aerospace Technology*, Vol. 91, No. 10, pp. 1257–1267, 2019.

25. Hart, J. C., G. K. Francis and L. H. Kauffman, "Visualizing quaternion rotation", *ACM Transactions on Graphics (TOG)*, Vol. 13, No. 3, pp. 256–276, 1994.

26. Voight, J., *Quaternion algebras*, Springer Nature, Hanover, 2021.

27. MATHWORKS, "Quaternions", `https://www.mathworks.com`, accessed on November 12, 2023.

28. Ducard, G. J. and M.-D. Hua, "Discussion and practical aspects on control allocation for a multi-rotor helicopter", *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. 38, pp. 95–100, 2012.

29. Minh, T. B., H. Vo and L. T. Hua, "Development of a novel V-frame octocopter: de-

sign, kinematic analysis, and simulation using PID controllers with Ziegler Nichols tuning method", *International Journal of Intelligent Unmanned Systems*, 2023.

30. Kotarski, D., J. Kasać and G. Dekoulis, "Generalized control allocation scheme for multirotor type of UAVs", *Drones—Applications; Dekoulis, G., Ed.; IntechOpen: London, UK*, pp. 43–58, 2018.

31. Dharmawan, A., A. Ashari and A. E. Putra, "Mathematical modeling of translation and rotation movement in quad tiltrotor", *International Journal on Advanced Science, Engineering and Information Technology*, Vol. 7, No. 3, pp. 1104–1113, 2017.

32. Klimczyk, W. A., "Aerodynamic design and optimization of propellers for multirotor", *Aircraft Engineering and Aerospace Technology*, Vol. 94, No. 1, pp. 21–30, 2022.

33. Motor, T., "U3", `https://store.tmotor.com/`, accessed on October 11, 2022.

34. Neubrex Co., L., "Optic Gyroscopes", `https://www.neubrex.com/`, accessed on May 25, 2021.

35. Chen, D. and G. X. Gao, "Probabilistic graphical fusion of LiDAR, GPS, and 3D building maps for urban UAV navigation", *Navigation*, Vol. 66, No. 1, pp. 151–168, 2019.

36. Ishlinskii, A. I., "Orientation, Gyroscopes, and Inertial Navigation", *Moscow Izdatel Nauka*, 1976.

37. Xie, H. and G. K. Fedder, "Integrated microelectromechanical gyroscopes", *Journal of Aerospace Engineering*, Vol. 16, No. 2, pp. 65–75, 2003.

38. Post, E. J., "Sagnac effect", *Reviews of Modern Physics*, Vol. 39, No. 2, p. 475, 1967.

39. Park, J., D. Lee and C. Park, "Implementation of a vehicle navigation system using GNSS, ins, odometer, and barometer", *Journal of positioning, navigation, and Timing*, Vol. 4, No. 3, pp. 141–150, 2015.

40. Inc., H. I., "Barometer", `https://aerospace.honeywell.com`, accessed on March 05, 2023.

41. Zhang, Jieying and Edwan, Ezzaldeen and Zhou, Junchuan, and Chai, Wennan and Loffeld, Otmar, "Performance investigation of barometer aided GPS/MEMS-IMU integration", *IEEE/ION Position, Location and Navigation Symposium*, pp. 598–604, IEEE, 2012.

42. Lee, Y.-C., M.-L. Hsieh, P.-S. Lin, C.-H. Yang, S.-K. Yeh, T. T. Do and W. Fang, "CMOS-MEMS technologies for the applications of environment sensors and environment sensing hubs", *Journal of Micromechanics and Microengineering*, Vol. 31, No. 7, p. 074004, 2021.

43. Mader, G. L., "GPS antenna calibration at the National Geodetic Survey", *GPS Solutions*, Vol. 3, pp. 50–58, 1999.

44. Takasu, T. and A. Yasuda, "Evaluation of RTK-GPS performance with low-cost single-frequency GPS receivers", *International Symposium on GPS/GNSS*, pp. 852–861, 2008.

45. Agency, E. S., "GNSS-GALILEO", `https://galileognss.eu/`, accessed on October 09, 2023.

46. NOAA, "GNSS-GPS", `https://www.gps.gov/`, accessed on October 09, 2023.

47. Aughe, J., Robert, "Applications of GPS techniques to sports", *International Journal of Sports Physiology and Performance*, Vol. 6, No. 3, pp. 295–310, 2011.

48. Primdahl, F., "The fluxgate magnetometer", *Journal of Physics E: Scientific In-*

struments, Vol. 12, No. 4, p. 241, 1979.

49. Team, M.-D., "Magnetic Declination", `https://www.magneticdeclination.com/`, accessed on April 22, 2021.

50. Inc., H. I., "Declination Angle", `https://aerospace.honeywell.com`, accessed on April 25, 2021.

51. The MathWorks, I., "MAGCAL", `https://www.mathworks.com`, accessed on February 11, 2022.

52. The MathWorks, I., "MAGCALIB", `https://www.mathworks.com/`, accessed on February 11, 2022.

53. Yang, X., L. Mejias Alvarez and M. Garratt, "Multi-sensor data fusion for UAV navigation during landing operations", *Australasian Conference on Robotics and Automation (ACRA 2011)*, pp. 1–10, Australian Robotics & Automation Association, 2011.

54. Kotarski, D., M. Piljek, M. Tevčić and V. Vyroubal, "Mathematical modeling and dynamics analysis of flat multirotor configurations", *WSEAS transactions on systems*, Vol. 16, pp. 47–52, 2017.

55. World, G., "LiDAR", `https://www.geospatialworld.net`, accessed on November 18, 2023.

56. Exchange, S., "Sonar", `https://electronics.stackexchange.com`, accessed on November 18, 2023.

57. Du, Hao and Wang, Wei and Xu, Chaowen and Xiao, Ran, and Sun, Changyin, "Real-time onboard 3D state estimation of an unmanned aerial vehicle in multi-environments using multi-sensor data fusion", *Sensors*, Vol. 20, No. 3, p. 919, 2020.

58. Haibo Wang, X. C., "Low-level control with actuator dynamics for multirotor UAVs", *Robotic Intelligence and Automation*, Vol. 43, No. 3, pp. 290–300, 2023.

59. De Martini, D., G. V. Gramazio, A. Bertini, C. Rottenbacher and T. Facchinetti, "Design and modeling of a quadcopter with double axis tilting rotors", *Unmanned Systems*, Vol. 5, No. 03, pp. 169–180, 2017.

60. Piljek, P., D. Kotarski and M. Krznar, "Method for characterization of a multirotor UAV electric propulsion system", *Applied Sciences*, Vol. 10, No. 22, p. 8229, 2020.

61. Rinaldi, F., A. Gargioli and F. Quagliotti, "PID and LQ regulation of a multirotor attitude: Mathematical modeling, simulations, and experimental results", *Journal of Intelligent & Robotic Systems*, Vol. 73, pp. 33–50, 2014.

62. Team, A. D., "ArduPilot Codebase", `https://ardupilot.org/dev`, accessed on October 20, 2022.

63. Benić, Z., P. Piljek and D. Kotarski, "Mathematical modeling of unmanned aerial vehicles with four rotors", *Interdisciplinary Description of Complex Systems: INDECS*, Vol. 14, No. 1, pp. 88–100, 2016.

64. Orbea, D., J. Moposita, W. G. Aguilar, M. Paredes, G. León and A. Jara-Olmedo, "Math model of UAV multi-rotor prototype with fixed wing aerodynamic structure for a flight simulator", *Augmented Reality, Virtual Reality, and Computer Graphics: 4th International Conference, AVR, Ugento, Italy*, pp. 199–211, 2017.

65. Hosseini, A., H. Karimi, P. Zarafshan, J. Massah and Y. Parandian, "Modeling and control of an octorotor flying robot using the software in a loop", *4th International Conference on Control, Instrumentation, and Automation (ICCIA)*, pp. 52–57, IEEE, 2016.

66. Kim, H., H. S. Jeong, K. T. Chong and D. J. Lee, "Dynamic modeling and control techniques for multi-rotor flying robots", *Transactions of the Korean Society of*

Mechanical Engineers A, Vol. 38, No. 2, pp. 137–148, 2014.

67. Team, A. D., "ESCs and Motors", `https://ardupilot.org`, accessed on May 25, 2021.

68. Moussid, M., A. Sayouti and H. Medromi, "Dynamic modeling and control of a hexarotor using linear and nonlinear methods", *International Journal of Applied Information Systems*, Vol. 9, No. 5, pp. 9–17, 2015.

69. Härkegård, O. and S. T. Glad, "Resolving actuator redundancy—optimal control vs. control allocation", *Automatica*, Vol. 41, No. 1, pp. 137–144, 2005.

70. Center, G. R., "Aircraft Rotations", `https://www1.grc.nasa.gov`, accessed on February 22, 2022.

71. Manshadi, A. and F. Saghafi, "In-Flight Estimation of Time-Varying Aircraft Center of Gravity Position Based on Kinematics Approach", *Journal of Aircraft*, Vol. 55, No. 5, pp. 2037–2049, 2018.

72. Sakharov, A. V., "Rotation of the body with movable internal masses around the center of mass on a rough plane", *Regular and Chaotic Dynamics*, Vol. 20, pp. 428–440, 2015.

73. Allain, R., "Rocket Pendulum", `https://www.wired.com/story`, accessed on October 22, 2022.

74. Argha, A., S. W. Su and B. G. Celler, "Control allocation-based fault tolerant control", *Automatica*, Vol. 103, pp. 408–417, 2019.

75. Marks, A., J. F. Whidborne and I. Yamamoto, "Control allocation for fault tolerant control of a VTOL octorotor", *UKACC International Conference on Control*, pp. 357–362, IEEE, 2012.

76. Zhou, Q.-L., Y. Zhang, C.-A. Rabbath and D. Theilliol, "Design of feedback linearization control and reconfigurable control allocation with application to a quadrotor UAV", *Conference on Control and Fault-Tolerant Systems (SysTol)*, pp. 371–376, IEEE, 2010.

77. Monteiro, J. C., F. Lizarralde and L. Hsu, "Optimal control allocation of quadrotor UAVs subject to actuator constraints", *American Control Conference (ACC)*, pp. 500–505, IEEE, 2016.

78. Kirchengast, M., M. Steinberger and M. Horn, "Control allocation under actuator saturation: An experimental evaluation", *IFAC-PapersOnLine*, Vol. 51, No. 25, pp. 48–54, 2018.

79. Vepa, R., "Control allocation in an over-actuated octocopter drone", *International Symposium of Asian Control Association on Intelligent Robotics and Industrial Automation (IRIA)*, pp. 7–12, IEEE, 2021.

80. Johansen, Tor A, and Fossen, Thor I, "Control allocation—A survey", *Automatica*, Vol. 49, No. 5, pp. 1087–1103, 2013.

81. Smeur, E., D. Höppener and C. De Wagter, "Prioritized control allocation for quadrotors subject to saturation", *International Micro Air Vehicle Conference and Flight Competition*, pp. 37–43, Delft University of Technology, 2017.

82. Lipnickas, A. and A. Knyš, "A stereovision system for 3-D perception", *Elektronika ir Elektrotechnika*, Vol. 91, No. 3, pp. 99–102, 2009.

83. Faugeras, O., *Three-dimensional computer vision: a geometric viewpoint*, MIT press, 1993.

84. Serafinavičius, P., S. Sajauskas and G. Daunys, "Evaluation of hand pointing system based on 3-D computer vision", *Elektronika and Elektrotechnika*, Vol. 88, No. 8, pp. 95–98, 2008.

85. Adil, E., M. Mikou and A. Mouhsen, "A novel algorithm for distance measurement using stereo camera", *CAAI Transactions on Intelligence Technology*, Vol. 7, No. 2, pp. 177–186, 2022.

86. Solak, S. and E. D. Bolat, "Distance estimation using stereo vision for indoor mobile robot applications", *9th International Conference on Electrical and Electronics Engineering (ELECO)*, pp. 685–688, IEEE, 2015.

87. Sadreddini, Z., T. Çavdar and H. B. Jond, "A distance measurement method using a single camera for indoor environments", *39th International Conference on Telecommunications and Signal Processing (TSP)*, pp. 462–465, IEEE, 2016.

88. Chun-Shin, L., S. Jaehong and C. Heesung, "Distance measurement using a single camera with a rotating mirror", *International Journal of Control, Automation, and Systems*, Vol. 3, No. 4, pp. 542–551, 2005.

89. Moussid, M., A. Sayouti and H. Medromi, "Dynamic modeling and control of a hexarotor using linear and nonlinear methods", *International Journal of Applied Information Systems*, Vol. 9, No. 5, pp. 9–17, 2015.

90. Gui, P., L. Tang and S. Mukhopadhyay, "MEMS-based IMU for tilting measurement: Comparison of complementary and Kalman filter-based data fusion", *10th Conference on Industrial Electronics and Applications (ICIEA)*, pp. 2004–2009, IEEE, 2015.

91. Team, A. D., "TECS and L1", `https://ardupilot.org`, accessed on July 07, 2023.

92. Parts, A., "Filters' Bodes", `https://www.amplifiedparts.com`, accessed on November 15, 2023.

APPENDIX A: Quoted Figures Used in the Thesis

- While quoting the figures in the thesis, which are protected by public license and open license and are open to the use of all individuals in interactive environments, "we have acted in accordance with the restrictions required by the license."
- The figures that emerged within the scope of this thesis study and whose copyright was transferred to the publishing house were used in the thesis book by the "publishing policy valid for re-using texts and graphics produced by the author" on the publishing house's website.