Universidade Federal
do Rio de Janeiro
Escola Politécnica

MODELLING AND CONTROL OF A QUADROTOR VEHICLE

João Carlos Espiúca Monteiro

Trabalho de Conclusão de Curso de Bacharelado apresentado à Escola Politécnica da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Bacharel em Control and Automation Engineering.

Orientadores: Liu Hsu
Fernando Cesar Lizarralde

Rio de Janeiro
Março de 2015

This work is dedicated to my family and friends, who supported and advised me and who now share with me the joy of becoming an engineer. The satisfaction I feel upon the completion of this project is not due to the end of a chapter, but rather to the enthusiasm for the continued and relentless pursuit of the art of knowledge.

Acknowledgements

I would first like to thank my advisors, Professors Liu Hsu and Fernando Lizarralde, for all the support given in the elaboration of this dissertation, for the theoretical lessons, and for the moments of relaxation. I thank the committee members, Professors Alessandro Jacoud Peixoto and José Paulo V. S. da Cunha, for accepting to evaluate this work and for the suggestions made in order to improve it. I thank all the professors and colleagues who accompanied me throughout the years of my undergraduate studies.

I thank FAPERJ and CNPq for the funding that made the experimental development of this work possible.

Finally, but by no means less importantly, I thank my family and my closest friends, for all the love, care, and support.

Resumo do Trabalho de Conclusão de Curso apresentado à POLI/UFRJ como parte dos requisitos necessários para a obtenção do grau de Bacharel em Engenharia (B.Eng.)

MODELLING AND CONTROL OF A QUADROTOR VEHICLE

João Carlos Espiúca Monteiro

Março/2015

Orientadores: Liu Hsu

Fernando Cesar Lizarralde

Curso: Control and Automation Engineering

This dissertation addresses the control problem of quadrotor unmanned aerial vehicles (UAVs). This problem is divided into two parts: computation of the desired virtual forces vector, and mapping of this vector to propellers angular velocities (control allocation). In the first part several control strategies are presented. These are designed to accomplish different objectives and their performances are compared via simulations. Using a new approach based on optimality criteria, two control allocation methods are presented. These are required to meet the performance demand of control algorithms. To enable the results mentioned above, the dynamic model of this class of aerial vehicles is given in details and some parameters are estimated through experimental data.

Abstract of Dissertation presented to POLI/UFRJ as a partial fulfillment of the requirements for the degree of Bachelor of Engineering (B.Eng.)

MODELAGEM E CONTROLE DE UM VEÍCULO QUADRIRROTOR

João Carlos Espiúca Monteiro

March/2015

Advisors: Liu Hsu

Fernando Cesar Lizarralde

Course: Engenharia de Controle e Automação

Nesta dissertação aborda-se o problema de controle dinâmico de veículos aéreos não-tripulados (VANTs) quadrirrotores. Este problema é dividido em duas partes: cálculo do vetor de forças virtuais desejadas e mapeamento deste vetor nas velocidades de rotação das hélices (alocação de controle). Na primeira parte são apresentadas estratégias que buscam atingir diferentes objetivos e seus desempenhos são comparados por simulações. Uma nova abordagem baseada em critérios de otimalidade é utilizada para desenvolver dois métodos de alocação de controle. Estes são necessários para que a demanda de desempenho dos algoritmos de controle seja atendida. Para viabilizar a obtenção dos resultados mencionados acima, o modelo dinâmico desta classe de veículos é detalhado e alguns parâmetros são estimados a partir de dados experimentais.

Sumário

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introduction

1.1 Motivation

The popularity of unmanned aerial vehicles (UAVs) has grown significantly in recent years, along with the investment in research. Alongside this growth, a variety of applications have emerged, from civilian to military, that make use of this new technology. Among the vehicles leading this advance are those capable of vertical take-off and landing, VTOL (*Vertical Take-off and Landing*). As the name suggests, these vehicles are capable of taking off, landing and hovering vertically, i.e. without moving significantly in the horizontal plane. A special class of VTOL vehicles is that composed of multirotors, vehicles equipped with more than one propulsor. This work focuses on a particular case of this class of vehicles, composed of vehicles with four propulsors and therefore called quadrotors.

Currently, multirotors have been employed in various applications, e.g. surveillance systems, mapping and filming. Recent examples of the widespread adoption of these vehicles include their use in searching for dengue breeding sites in the cities of Santos and Limeira in the state of São Paulo, and aerial filming of carnival in the city of Rio de Janeiro. In December 2013, the American multinational Amazon announced the launch of a project called Amazon Prime Air, which seeks to deliver packages in less than 30 minutes using autonomous multirotors.

Often referred to simply as *drones*, quadrotors can be found in various sizes, from small-scale vehicles, MAVs (*Micro Aerial Vehicle*), with a wingspan on the order of ten centimeters, to larger vehicles with a wingspan on the order of one meter. The required

vehicle size is chosen according to the desired application. Usually, small quadrotors are used in indoor environments, while larger ones are more suitable for outdoor use.

Regardless of the size chosen, to ensure that these vehicles operate correctly, it is necessary to develop control algorithms that assist the pilot during teleoperated flight. In many situations, the pilot can be entirely replaced by control algorithms that must ensure the autonomous operation of the vehicle. These algorithms must guarantee robustness to parametric uncertainties, unmodeled aerodynamic effects, aerodynamic disturbances, and measurement and state estimation errors. The performance analysis of these algorithms constitutes one of the central topics addressed in this work.

1.2 Unmanned Aerial Vehicles

The concept of an unmanned aerial vehicle encompasses any and all aircraft capable of flying without passengers or crew. These vehicles may also be fully autonomous, capable of flying without pilot intervention, or remotely piloted. These characteristics make UAVs ideal for applications where the presence of human beings is dangerous, exhausting or even unfeasible.

The first recorded use of a UAV is uncertain. There are records dating back to the mid-19th century about the use of unmanned aircraft in wars and test flights. The first tests were not successful, but the history began to take a different turn in 1914, when Lawrence Sperry demonstrated to an audience of civilians and military personnel the first autopilot technology. Sperry and his assistant Emil Cachin climbed onto the wings of the aircraft, a Curtiss C-2, while flying over the audience, leaving the vehicle under the sole control of the autopilot.

The development of these vehicles has always been closely linked to military applications. During the First World War, the United States built a model called the Kettering Bug (Figure 1.1), a precursor of modern missiles, whose control system was designed by Elmer Sperry, father of Lawrence Sperry. This vehicle was still very primitive and was never used in combat. During the Second World War, the Germans developed several UAV models that were used in bombing campaigns. The most well-known of these is the V-1 (Figure 1.1), or Doodle Bug, a missile used in the extensive bombing of London in 1944. After the impact of two wars, a great effort was directed toward the research and

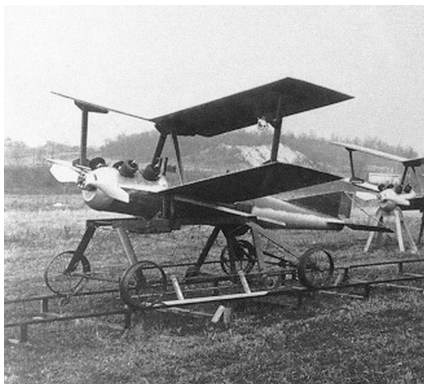


Figura 1.1: Kettering Bug[1] on the left and Doodle Bug[2] on the right.

development of these aircraft.

Since the beginning of their history, the use of UAVs has been closely tied to the military industry, but it has not been restricted to this market alone. Instead of missiles, these vehicles were refined and autonomous flight systems were developed for helicopters and, more recently, multirotors. In the last two decades, research on multirotors has grown rapidly. Currently, there are several groups involved in the study of quadrotors. In Section 1.4 these projects will be presented.

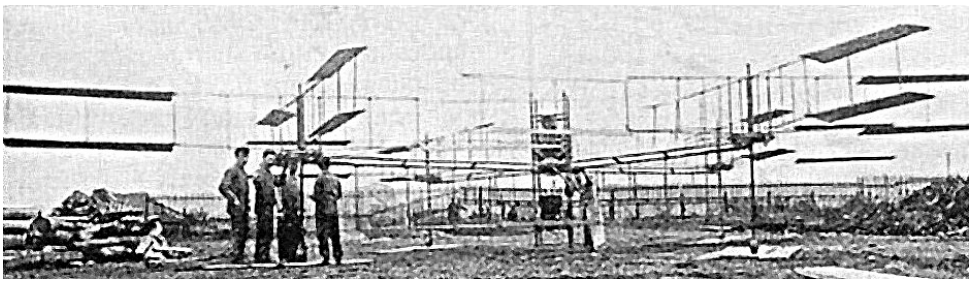
1.3 Multirotors: A Brief History

In parallel with the development of UAVs, there was also the study of helicopters and multirotors, composed of more than two propellers. Contrary to common belief, the first attempts at flight with propeller-driven vehicles were made with quadrotors. Around 1907, the French brothers Louis and Jacques Breguet, in partnership with Prof. Charles Richet, built and tested the quadrotor called the Breguet-Richet Gyroplane, or Gyroplane No 1, shown in Figure 1.2(a). Although the vehicle was unable to hover more than one meter above the ground, this is the first recorded flight of a manned VTOL vehicle.

It was not until 1924 that the French engineer Étienne Oehmichen set the first record recognized by the Fédération Aéronautique Internationale (FAI), flying a total of 360 meters with his quadrotor Oehmichen No 2 (Figure 1.2(c)). Later that same year Oehmichen flew 1 km in the same aircraft. Also in the 1920s, George de Bothezat built his quadrotor

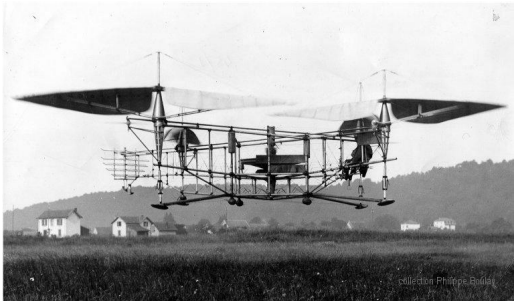
[1]Extracted from `https://upload.wikimedia.org/wikipedia/commons/thumb/3/35/Kettering_Bug.jpg/300px-Kettering_Bug.jpg`

[2]Extracted from `https://upload.wikimedia.org/wikipedia/commons/2/26/Bundesarchiv_Bild_146-1973-029A-24A,_Marschflugk%C3%B6rper_V1_vor_Start.jpg`



(a)

(b)



(c)

(d)

Figura 1.2: (a) Gyroplane No 1[3], (b) Bothezat Quadcopter[4], (c) Oehmichen No 2[5]and (d) Curtiss-Wright X-19A[6].

(Figure 1.2(b)) as part of a project for the United States Army, but the project was cancelled due to unsuccessful flights. At that time, the use of helicopters was favored, since these require a tail rotor or a second propeller to counterbalance the torque.

However, due to the inherently unstable nature of multirotors and the difficulty of commanding a large number of propulsors, helicopters gained prominence, especially during the Second World War. The only manned quadrotor to fly above ground effect was the Curtiss-Wright X-19A (Figure 1.2(d)), but even so the effort required to keep the aircraft hovering was very large, making stationary flight nearly impossible.

It was not until the mid-1990s that the development of multirotors gained new life. With the advent of microelectronics and the ability to build small, lightweight and efficient

electric motors, research on multirotors was resumed with greater intensity. Currently, there are still many challenges that must be overcome for these vehicles to gain even greater importance. Among these, the following stand out:

1. Design of onboard microelectronics for sensing and control;
2. Development of robust navigation and flight control strategies;
3. Increase in flight autonomy;
4. Reduction of vehicle structural weight;
5. Development of analytical and computational models for accurate simulation.

Much attention has been given to the second item and several control strategies capable of providing autonomous and teleoperated flight are already known. However, for all of these specifications to be met, there is still much technology to be developed.

1.4 State of the Art

Particularly since the beginning of this century, with the ever-increasing miniaturization of electronic components, batteries and motors, research in the field of quadrotors has increased considerably. Universities, researchers, companies and students have been constantly working on control strategies, trajectory tracking, reliable simulation platforms, analysis of aerodynamic effects and mechanical design metrics.

Current projects in the field of quadrotors are aimed at MAVs and small quadrotors weighing up to 4 kg. In 1999, researchers at Stanford initiated the Mesicopter project, Figure 1.3(a), which was completed in 2001. This quadrotor had dimensions of approximately two and a half centimeters. The motivation for the project was the deployment of a large number of vehicles on a planet, enabling the acquisition of an extensive amount of meteorological and atmospheric data.

Since then, interest in quadrotors has grown significantly. One of the main projects in the field was the OS4, Figure 1.3(b), from the École Polytechnique Fédérale de Lausanne (EPFL), spanning from 2004 to 2007. In this project, several control strategies

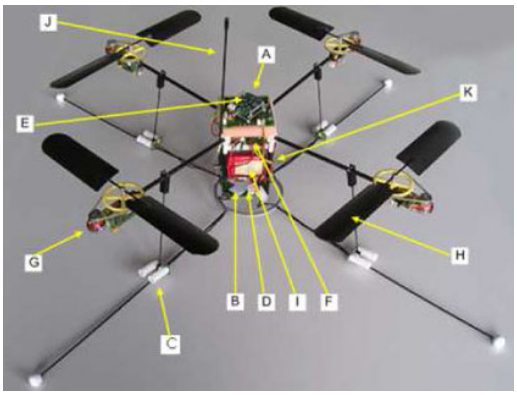
[3]Adapted from http://flyingmachines.ru/Images7/Janes/Janes13_2/ill_245b.jpg

[4]Adapted from http://api.ning.com/files/8wsCfYsVpx7tNegcbXL7RLNLLm1wQJrz3YG9gMJw*pKMUvboI37sBOkipihTLejyRlct7RzFURm9BD1J8FwbbmtBv8hVR0d3/bothezat_2.jpg

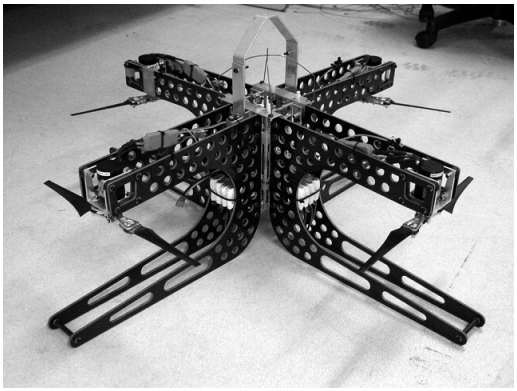
[5]Extracted from http://www.eyeondrones.com/wp-content/uploads/oehmichen-no2.jpg

[6]Adapted from http://www.diseno-art.com/images_3/Curtiss-Wright_X-19_ground.jpg

(a)



(b)



(d)

Figura 1.3: Some quadrotors developed at universities: (a) Mesicopter[7], Stanford; (b) OS4[8], EPFL; (c) Mark II X-4 Flyer[9], ANU.

were tested, such as *Backstepping* and Sliding Mode Control in [3] and Proportional Integral Derivative (PID) and Linear Quadratic Regulator (LQR) in [4]. Several inertial and aerodynamic effects are studied in [2].

Developed at the Australian National University (ANU), starting in 2004, the X-4 Flyer project made a significant contribution to the study of aerodynamic effects and provided detailed documentation of the vehicle synthesis. This study is detailed in [19] and the model developed in this work can be seen in Figure 1.3(c). Apparently, the project was completed in 2010.

Like the X-4 Flyer and the OS4, the STARMAC project also began in 2004. This project was led by Stanford University in partnership with Berkeley and consisted in the development of a quadrotor test platform. In this environment, studies were conducted on topics such as: analysis of blade flapping effects and vertical flight [10] and methods to

[7]Extracted from `http://adg.stanford.edu/mesicopter/ProgressReports/images/mesicopter.gif.gif`

[8]Adapted from `http://www.scielo.br/img/revistas/jbsmse/v34n3/a10fig01.jpg`

[9]Extracted from `unkown`

Figura 1.4: AscTec Hummingbird quadrotor by Ascending Technologies[10].

compensate for effects experienced during flight [11].

Leading the research in the field of quadrotor MAVs are the Swiss Federal Institute of Technology Zurich (ETHZ) and the University of Pennsylvania (Penn). The former has a test platform called FMA (Flying Machine Arena). In this environment, several experiments were conducted, such as: balancing an inverted pendulum [9], performing flips [14] and ball juggling [18]. The latter houses the GRASP laboratory, whose research on quadrotors has already produced results such as: trajectory tracking via a minimum *snap*[11] strategy [15], fast trajectory tracking in narrow environments [16] and coordinated group flight for load transportation [17]. Both groups use an environment monitored by the *Vicon Motion Capture System* from Vicon and AscTec Hummingbird quadrotors, Figure 1.4, from Ascending Technologies.

1.5 Quadrotor Operation

Quadrotor vehicles, also known as quadcopters, are composed of four ideally identical rotors arranged in pairs. Each pair of rotors spins in opposite directions, clockwise and counterclockwise. This configuration provides greater agility, safety and efficiency to these vehicles when compared to small helicopters [11].

Unlike helicopters, quadcopters do not have *swashplates*; therefore, all movements are generated through coordinated changes in the rotational speeds of the rotors. Figure 1.5 shows how these changes affect the attitude of these vehicles. By varying the rotational speed of the propellers simultaneously, it is possible to increase or decrease the altitude, Figure 1.5(a). Rotations around the vehicle axes also produce movement in the plane parallel to the ground. This can be achieved by increasing the speed of one rotor and proportionally reducing the speed of its pair, Figure 1.5(b). This reduction is necessary

[10]Extracted from `http://www.asctec.de/wp-content/uploads/2014/07/AscTec-hummingbird-research-drone-swarming-flight-dynamic-control-theory.jpg`

[11]The physical quantity *snap* refers to the second derivative of the acceleration of a body.

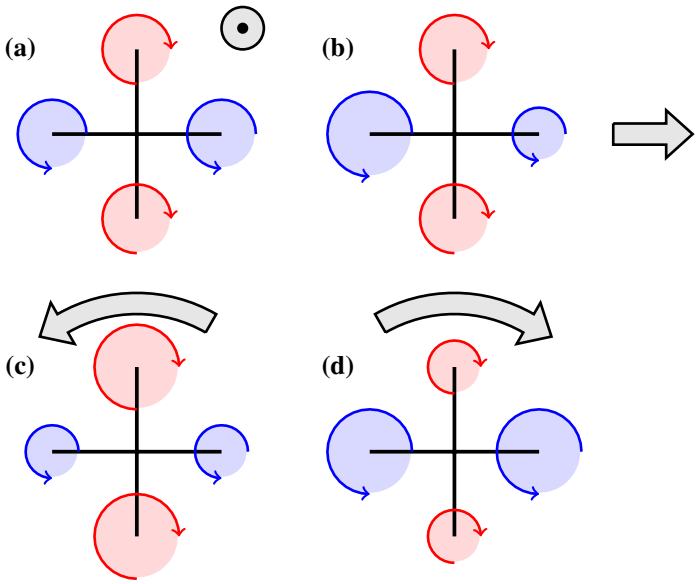


Figura 1.5: Simplified diagram of the attitude changes of a quadrotor due to changes in the rotational speed of the rotors; speed increase (a) vertical (out of the page), (b) to the right, (c) and (d) around its own axis (normal to the page).

so that the vehicle maintains its altitude. Finally, the quadrotor can also rotate around its principal axis. To achieve this movement, the rotational speed of one pair of propellers is increased and that of the other pair is proportionally decreased, Figure 1.5(c). In this way, the quadrotor rotates as mentioned, but maintains the same altitude, since there is no change in the resultant thrust.

In general, quadrotors move as described above, but there are two distinct flight configurations that can be chosen. The first is as shown in Figure 1.6(a) and is known as the (+) configuration. The second is called the (x) configuration and is illustrated in Figure 1.6(b). Generally, there are no determining factors in choosing one configuration over the other. As will be seen, these configurations only affect the way that rotor rotation is mapped into desired forces and torques.

Regarding the advantages of using quadrotors, the reduced effect of gyroscopic forces, large payload capacity and simple dynamic model stand out. Together, these factors make these vehicles relatively simple to build and control. However, perhaps the greatest disadvantage of quadrotors is the high energy consumption. This consumption means that the average flight time is in the range of ten to fifteen minutes, and may be even less when the vehicle carries extra payload, e.g. GPS and cameras.

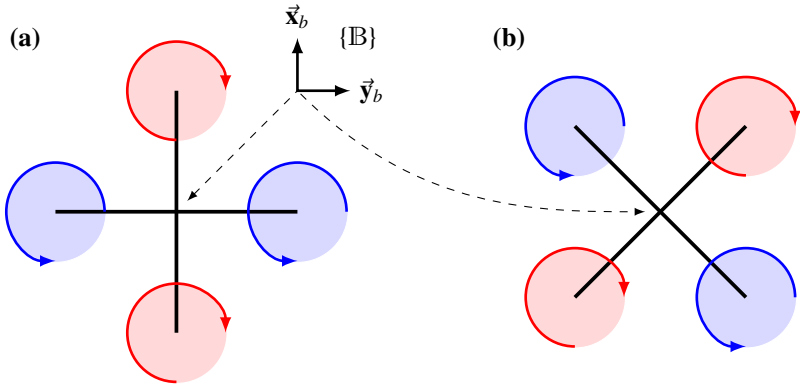


Figura 1.6: Flight configurations in (+) (a) and in (x) (b).

1.6 Contributions

The focus of this undergraduate thesis is the development and control of a small-scale unmanned quadrotor, i.e. a quadrotor MAV. The contributions of this work can be divided into three areas: dynamic modelling, high-level control and control allocation.

1.6.1 Dynamic Modelling

A detailed documentation of the dynamic model of quadrotors was obtained, resulting from the study of several publications in the field. Throughout Chapter 2 this work is presented and details are provided on how to determine all of the dynamic effects considered.

1.6.2 High-Level Control

A large part of this work was devoted to the study of control strategies suitable for use in a real quadrotor. Based on the dynamic model, some control strategies already known in the field of quadrotors were studied. The positive and negative aspects of each technique are discussed. Furthermore, two variations of the *Backstepping* control strategy and a variable-gain controller for commanding the vehicle altitude are proposed.

1.6.3 Control Allocation

As is done in the field of underwater vehicles (ROVs) and UAVs, the control strategies were designed considering a virtual force vector as the system input. In the literature, the problem of mapping this virtual vector to the real input vector is referred to as control allocation. This concept is not commonly studied in the context of quadrotors. To perform

the mapping, two methods known in the literature are studied: constrained least squares and direct allocation. A variation of the first method is presented with the purpose of reducing computational cost. In turn, the direct allocation technique is modified to reflect the behavior desired by the control loops.

1.7 Document Structure

In this chapter, a brief introduction to the class of UAVs was given. Next, a brief history of these vehicles was presented and their main characteristics were described. Research projects on quadrotor vehicles were cited, providing an idea of the state of the art in this area of research. The following paragraphs present what is covered in the upcoming chapters.

Chapter 2 presents in a clear and direct manner the process used to obtain the dynamic model of the quadrotor. First, the forces and torques acting on the vehicle are defined, and then the Newton-Euler methodology is applied to obtain the differential equations governing its behavior.

Chapter 3 addresses the control of the vehicle in an intuitive manner. The characteristics of the model are studied with the objective of providing a modular approach to the problem. Several position and orientation control strategies present in the current literature are presented. A variable-gain technique for altitude control is proposed, as well as variations of the *Backstepping* method.

Chapter 4 addresses the control allocation problem. This is a well-known problem that has its origins in the maritime and aerospace fields. However, there are no works exploring this issue in depth in the field of quadrotors. Two allocation strategies are presented and variations of them are proposed.

Chapter 5 describes the environment where the flights with the real quadrotor were conducted, as well as the experimental results obtained.

Chapter 6 presents the conclusions of this work as well as suggestions for future work.

Appendix A describes the process used to determine the noise present in the measurements of the real system.

Appendix B describes how the thrust and drag constants of the propellers of the quadrotor used in this work were obtained.

Appendix C presents the environment used to carry out the simulations described in this work, implemented in Matlab® and Simulink®. The virtual environment developed to allow a better visualization of the vehicle flight is also shown.

Capítulo 2

Dynamic Modelling

2.1 Adopted Notation

To ensure understanding of the dynamic model that will be developed in this chapter, it is necessary to define some notation conventions. Therefore, the notation adopted to represent coordinate systems, vectors and their representations, matrices, and rotation matrices is presented below.

Coordinate Systems: will be written in uppercase and enclosed in brackets, e.g. $\{\mathbb{B}\}$, and may also be called reference frames.

Vectors: will be written in bold, with an arrow above the name, e.g. $\vec{\mathbf{v}}$, and are not fixed to any reference frame. When a vector is represented in a frame, this is denoted by a superscript, e.g. $\mathbf{v^b}$ would be the vector $\vec{\mathbf{v}}$ in frame $\{\mathbb{B}\}$. In general, vectors will be written in lowercase, except for a few cases. In these cases there will still be either the frame superscript or the arrow, in order to avoid ambiguity.

Matrices: will be written in uppercase and bold, e.g. $\mathbf{M}$, so that they are not confused with the notation adopted for vector representations.

Rotation Matrices: the same notation used above for ordinary matrices, but with a left subscript and a right subscript denoting the reference frames. For example, ${}_{\mathbf{E}}\mathbf{R}_{\mathbf{B}}$ would be the rotation matrix that maps from the coordinate system $\{\mathbb{B}\}$ to the coordinate system $\{\mathbb{E}\}$.

2.2 Model Characteristics

In order for the model to reflect the flight conditions found in practice, several effects documented in the literature will be considered. The only exception will be ground effect, which will not be considered in the vehicle model. This simplification is made because this effect interferes only in landing and takeoff situations, which fall outside the scope of this work.

The following assumptions are made regarding the characteristics of both the vehicle and its components:

- Body considered rigid and symmetric around its principal axes;
- Center of gravity coincident with the geometric center of the vehicle.

In general, these approximations are valid and do not result in any loss of generality. Although the center of gravity does not coincide exactly with the geometric center, the distance between them is usually not very large and therefore the assumption is valid. As for the symmetry of the vehicle, this assumption makes the body inertia matrix diagonal, simplifying the equations that describe its dynamics.

As seen previously, a quadrotor flies using four rotors, and it is the differences in their rotational speeds that cause the vehicle to move. In order to describe these motions, it is necessary to define the reference frames that will be used. Figure 2.1 shows the quadrotor configuration and the location of the frames fixed to the body $\{\mathbb{B}\}$ and to the ground $\{\mathbb{E}\}$. In addition, the main forces acting on the vehicle are also represented. Note that since $\vec{\mathbf{x}_\mathbf{b}}$ and $\vec{\mathbf{y}_\mathbf{b}}$ are aligned with the structure, the quadrotor is oriented in the (+) flight configuration. Thus, rotations around these axes can be obtained by actuating only one rotor, e.g. an increase in the rotational speed of rotor four generates a rotation around $\vec{\mathbf{x}_\mathbf{b}}$.

2.3 Inertial and Aerodynamic Effects

In Section 1.4, some of the aerodynamic effects that directly interfere with the dynamics of a quadrotor were discussed. As will be shown below, there are desirable effects, which are necessary for the vehicle to fly. However, there are also effects that disturb its dynamics and may impair its performance if they are not compensated for.

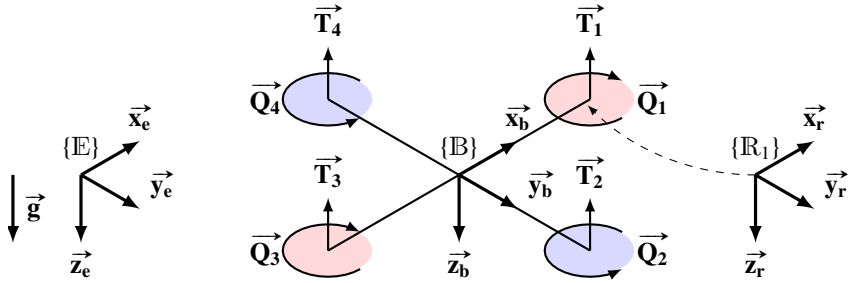


Figura 2.1: Reference frames and main forces acting on the vehicle. For simplicity, only frame $\mathbb{R}_1$ is illustrated, but there are also frames $\mathbb{R}_{2,3,4}$, fixed to the respective rotors.

This section adapts the concepts presented in [7], [20] and [10] and also addresses the action of gyroscopic precession resulting from lift dissymmetry. Not only will the aerodynamic forces be explained, but also the impact they have on the vehicle dynamics. In this way, each phenomenon will be explained in order to make explicit the equations that govern it.

Sections 2.3.1 and 2.3.5 describe in detail how propeller motion produces thrust and drag. This development is important for the construction of a reliable simulation model. With regard to vehicle control and general understanding of the model, the reader may use only the final result presented in equation (2.21) without major losses.

2.3.1 Vertical Flight

When the vehicle is in vertical flight, two main forces act on it. These are the lift force and the drag force. The first is the one that drives the vehicle upward and results from the reaction to the vertical force exerted by the rotor on the fluid. The second results from the reaction to the horizontal force exerted by the rotor on the fluid. In order to arrive at the equations that describe these forces, two different approaches will be used: momentum theory and blade element theory. The former is considerably simpler than the latter, but fails to describe more complex phenomena, e.g. *vortex ring state* and *windmill brake state*. The latter, in turn, represents a more complete analysis and makes direct use of the well-established aerodynamic theory for the study of airfoils.

Momentum Theory

Momentum theory provides a simple way to understand the forces that arise from propeller rotation, serving as a basis for more complex theories. In this section, the hover

condition will be studied, a particular case of vertical flight, and then the results will be generalized.

The basis of momentum theory consists of considering the rotor as a disk through which there is an abrupt increase in pressure, uniformly distributed. When in hover, the air column that passes through the disk can be seen as a streamtube, outside of which the airflow is not disturbed. Figure 2.2 illustrates this situation. There is a pressure reduction in the region above the disk, drawing the air above it and increasing its speed. When the air comes into contact with the disk, there is a sudden increase in pressure ΔP, causing the fluid to continue gaining speed until it once again reaches atmospheric pressure p_∞.

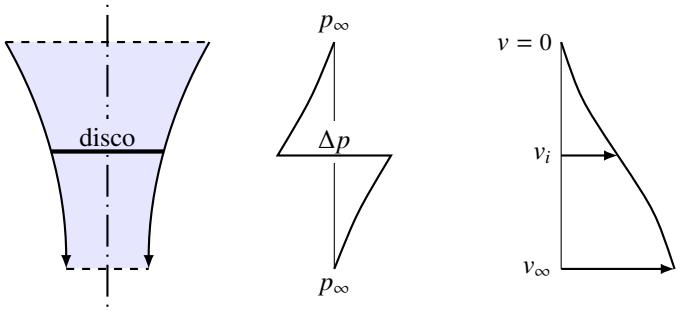


Figura 2.2: Behavior of the airflow through the disk in hover.

Bernoulli's equation states that the pressure along the flow of an incompressible fluid remains constant. Therefore, applying this relation in the region above the disk,

$$p_\infty = \frac{1}{2}\rho v_i^2 - \frac{\Delta p}{2}. \tag{2.1}$$

That is, the fluid starts at atmospheric pressure $p_\infty \in \mathbb{R}^+$ and reaches the region just above the disk with an induced velocity $v_i \in \mathbb{R}^+$ and pressure $-\Delta_p/2 \in \mathbb{R}^-$. The constant $\rho \in \mathbb{R}^+$ denotes the density of the fluid, which in this context is air. In the region below the disk, the same analysis is performed, so that

$$\frac{1}{2}\rho v_i^2 + \frac{\Delta p}{2} = p_\infty + \frac{1}{2}\rho v_\infty^2. \tag{2.2}$$

Again, the left-hand side of the equation refers to the starting region (immediately below the disk) and the right-hand side to the arrival region (at the end of the streamtube).

From the results obtained in equations (2.1) and (2.2), it is possible to eliminate the

atmospheric pressure and isolate the pressure difference, as follows

$$\Delta p = \frac{1}{2}\,\rho v_\infty^2 \tag{2.3}$$

Note that no rotation is imparted from the disk to the flow. Therefore, using conservation of momentum, the reaction thrust on the disk is known to be equal to the variation in linear momentum in the flow, that is

$$T = (\rho A v_i)\, v_\infty. \tag{2.4}$$

Where $A \in \mathbb{R}^+$ is the disk area and $\rho A v_i$ is the airflow rate passing through it. Furthermore, since pressure is the ratio between force and area,

$$\Delta p = T/A \;\;\therefore\;\; T = A\Delta p. \tag{2.5}$$

Equating equations (2.4) and (2.5) and substituting the value of the pressure variation obtained in (2.3), one has

$$v_\infty = 2v_i \tag{2.6}$$

Finally, considering that all the power P_h from the motor is converted into kinetic energy for the fluid, it follows directly from the definition of power that

$$P_h = T v_i = \frac{T^{3/2}}{\sqrt{2\rho A}}. \tag{2.7}$$

This quantity is also known as the power induced by the motor, since it represents the energy transferred from the motor to the fluid.

The procedure used here can be expanded to treat the vertical flight condition, see [20]. To do so, it is enough to consider an initial velocity $v_c \neq 0$, oriented from top to bottom, of the control volume and apply the equations of conservation of momentum, mass, and energy. Thus, it is possible to relate the power induced in hover P_h to the power induced in vertical flight P_i through

$$\frac{P_i}{P_h} = \frac{v_c}{2v_h} + \sqrt{\left(\frac{v_c}{2v_h}\right)^2 + 1}. \tag{2.8}$$

It is observed that the required power increases with the desired climb speed. That is, the greater the climb speed (*climb*), the greater the power required to spin the propellers. For small values of this climb speed, the first-order approximation

$$P_i \approx P_h + Tv_c/2 \tag{2.9}$$

may be used.

It remains to determine thrust as a function of rotor rotational speed. First, note that the power generated by the rotor is given by the product of torque and angular velocity $\tau_r\Omega$. In turn, torque is proportional to lift, with proportionality constant κ. Therefore, rotor power can be written as

$$P_i = \kappa T\Omega. \tag{2.10}$$

Equating equations (2.9) and (2.9), the lift force can be written as a function of rotor rotational speed and climb speed, according to

$$T = 2\rho A\left(\kappa\Omega - v_c/2\right)^2. \tag{2.11}$$

That is, when the vehicle is climbing and the fluid has an initial velocity $v_c > 0$, oriented from top to bottom, the thrust generated by the rotor is reduced. The opposite occurs when $v_c < 0$. Note that when the vehicle is in hover and $v_c = 0$, equation (2.11) reduces to

$$T = 2\rho A\left(\kappa\Omega\right)^2. \tag{2.12}$$

This approach does not allow an accurate analysis of vertical flight when the vehicle is descending. In this situation, the air flows from bottom to top, reducing the effort required to spin the propeller. As this velocity increases, the effects listed below emerge:

- *Vortex Ring*: formation of vortices near the blade rotation axis, which interact with the tip vortices, generating turbulent flow and loss of lift. When this occurs there is energy dissipation in the turbulent flow and it is not possible to apply the momentum theory developed in this section.

- Autorotation: a situation in which the rotor system extracts enough energy from the flow to maintain a constant descent speed.
- *Windmill Brake*: occurs when the vehicle descends at very high speeds, causing practically the entire flow to rise through the propellers; the vehicle absorbs more energy than necessary to fly, producing an undesired increase in rotor speed.

These phenomena occur in this order as the descent speed increases, i.e. as the airflow through the propeller from bottom to top increases. In particular, autorotation is an effect that can be exploited when there is motor failure, allowing a safe landing of the aircraft.

Blade Element Theory

Blade element theory treats the propeller as a set of airfoils[1] and provides a more complete approach, allowing a better understanding of aerodynamic phenomena. The entire development of this section is based on airfoil theory applied to each of the blades that make up the propeller.

Figure 2.3 shows the top view of the rotor and Figure 2.4 the profile of one of the blades. When the rotor spins, the blade sections are subjected to a vertical velocity component $v_c + v_i$ and another component tangent to the rotation Ωy. Thus, the absolute value of the resulting velocity is

$$v_\phi = \sqrt{(v_c + v_i)^2 + (\Omega y)^2}, \tag{2.13}$$

at an angle

$$\phi = \tan^{-1}\left[(v_c + v_i)/(\Omega y)\right] \approx (v_c + v_i)/(\Omega y) \tag{2.14}$$

with the disk plane. The approximation is made by means of small angles. This hypothesis implies that the denominator Ωy is significantly greater than the numerator $v_c + v_i$, i.e. the tangential velocities of the infinitesimal elements that compose the blade are significantly greater than the vertical velocity of the fluid crossing the disk plane. This is not valid near the rotor axis, but in this region the aerodynamic forces on the blades are small and do not significantly interfere with the final result.

[1] Airfoils are aerodynamic shapes capable of generating greater lift force per unit drag.

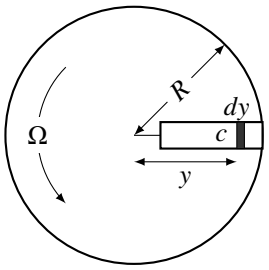


Figura 2.3: Top view of the rotor.

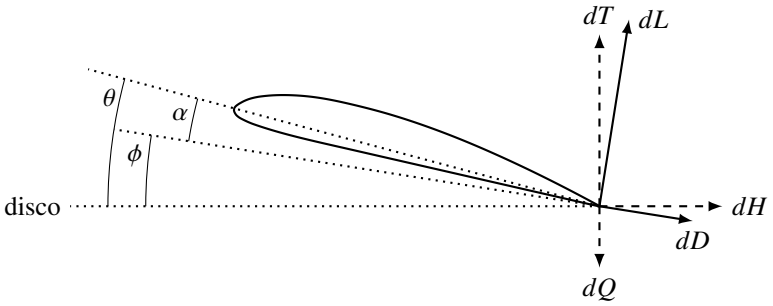


Figura 2.4: Airfoil representing the profile of the rotor blade.

In addition to the shape of the airfoil that characterizes the blade profile, its inclination relative to the disk plane is also determinant. This inclination is one of the factors that defines the angle of attack[2]. Therefore, given that θ is the blade inclination angle relative to this plane, the angle of attack is

$$\alpha = \theta - \phi. \tag{2.15}$$

With these results, it is possible to present the infinitesimal equations[3] for lift and drag, respectively,

$$\begin{aligned} dL &= (1/2)\rho\ C_L\ v_\phi^2 dA \\ dD &= (1/2)\rho\ C_D\ v_\phi^2 dA \end{aligned} \quad ; \quad dA = cdy\ . \tag{2.16}$$

Where $C_L \in \mathbb{R}$ is the lift coefficient and $C_D \in \mathbb{R}$ the drag coefficient. Projecting these forces in the direction normal to the disk and in its plane, see Figure 2.4, the infinitesimal

[2]Angle formed between the direction of the fluid flow and the airfoil chord.

[3]Infinitesimal notation is used to denote equations formed by infinitesimal quantities. It is directly related to the concept of limit and not differentiation.

thrust force dT, hub force dH, and torque around the axis dQ, will be

$$\begin{aligned} dT &= dL\cos\phi - dD\sin\phi\ , \\ dH &= dL\sin\phi + dD\cos\phi\ . \\ dQ &= ydH. \end{aligned} \tag{2.17}$$

Generally, drag force is significantly smaller than lift force, by at least one order of magnitude [7]. Furthermore, the assumption is still made that ϕ is small. Applying these assumptions to equations (2.13) and (2.17),

$$\begin{aligned} U &\approx \Omega y\ ; \\ dT &\approx dL\ ; \\ dQ &\approx dL\phi + dD\ . \end{aligned} \tag{2.18}$$

For the situations found in vertical flight, it may be considered that the hub forces acting on the two blades of each rotor cancel each other out. Therefore, integrating the relations in (2.17) and using the simplifications in (2.18), the equations for effective lift and aerodynamic torque are obtained

$$\begin{aligned} T &= N(1/2)\ \ \rho A C_T\ \ (\Omega R)^2\ , \\ Q &= N(1/2)\,\rho A C_Q R\,(\Omega R)^2\ . \end{aligned} \tag{2.19}$$

Where $C_T \in \mathbb{R}$ is the thrust coefficient, $C_Q \in \mathbb{R}$ the torque coefficient, $N \in \mathbb{R} > 0$ the number of blades, and $R \in \mathbb{R} > 0$ the blade length. These coefficients can be determined by the integrals

$$\begin{aligned} C_T &= \tfrac{1}{2}\sigma \int_0^1 C_L r^2 dr\ , \\ C_Q &= \tfrac{1}{2}\sigma \int_0^1 (\phi C_L + C_D) r^3 dr\ . \end{aligned} \tag{2.20}$$

Where $\sigma \in \mathbb{R} > 0$ is the solidity coefficient, defined by the ratio between the total blade area and the disk area $\sigma = Nc/\pi R$, and $r \in \mathbb{R} > 0$ is the normalized distance $r = y/R$. Approximate forms for writing these coefficients can be found in [7] and [20].

Finally, fixing that each propeller has only two blades, the final equations for thrust

and aerodynamic torque, widely used in the literature, are:

$$\begin{aligned} T &= \rho A \;\; C_T \;\; (\Omega R)^2 = \kappa_T \Omega^2 \,, \\ Q &= \rho A \; C_Q R \; (\Omega R)^2 = \kappa_Q \Omega^2 \,. \end{aligned} \tag{2.21}$$

2.3.2 Thrust Force

In this text, the thrust force $\vec{\mathbf{T}_\mathbf{i}}$ is defined as shown in Figure 2.1. Therefore, this force acts along the axis $\vec{\mathbf{z}_{\mathbf{r},\mathbf{i}}}$, is fixed to frame $\{\mathbb{R}_i\}$, and can be written as

$$\vec{\mathbf{T}_\mathbf{i}} = T_i \, (\vec{\mathbf{z}_{\mathbf{r},\mathbf{i}}}) \,.$$

Where the value of T_i can be seen in equations (2.21), (2.11) and (2.12), according to the required complexity.

Note that the force acts along $\vec{\mathbf{z}_{\mathbf{r},\mathbf{i}}}$ and it would be useful to represent it in the body frame. However, these frames do not necessarily coincide. In fact, when flapping occurs the propeller plane is displaced, as will be shown in Section 2.3.5. Therefore, representing the vectors in $\{\mathbb{B}\}$, the thrust equation in the body frame is given by

$$\mathbf{T}_\mathbf{i}^\mathbf{b} = T_i \, ({}_\mathbf{B}\mathbf{R}_{\mathbf{Ri}} \mathbf{e}_\mathbf{z}) \,. \tag{2.22}$$

2.3.3 Thrust Torque

As already seen, the rotors of a quadrotor are positioned at a certain distance from its center of gravity. For this reason, the thrust force, which acts along the rotor axes, produces torque components around the axes $\vec{\mathbf{x}_\mathbf{b}}$ and $\vec{\mathbf{y}_\mathbf{b}}$. If blade flapping occurs, the analysis of these torques is greatly simplified, as will be shown.

Consider then that the force $\vec{\mathbf{T}_\mathbf{i}}$ will produce a torque $\vec{\boldsymbol{\tau}_\mathbf{i}}$ and that it acts at a distance $l \in \mathbb{R} > 0$ from the center of mass, as shown in Figure 2.5. Besides this horizontal distance, also consider that the rotation plane is located at a vertical distance $h \in \mathbb{R} > 0$.

Therefore, the torques generated by each thrust force are

$$\begin{cases} \vec{\boldsymbol{\tau}_1} = [l(+\vec{\mathbf{x}_b}) + h(-\vec{\mathbf{z}_b})] \times \vec{\mathbf{T}_1} \\ \vec{\boldsymbol{\tau}_2} = [l(+\vec{\mathbf{y}_b}) + h(-\vec{\mathbf{z}_b})] \times \vec{\mathbf{T}_2} \\ \vec{\boldsymbol{\tau}_3} = [l(-\vec{\mathbf{x}_b}) + h(-\vec{\mathbf{z}_b})] \times \vec{\mathbf{T}_3} \\ \vec{\boldsymbol{\tau}_4} = [l(-\vec{\mathbf{y}_b}) + h(-\vec{\mathbf{z}_b})] \times \vec{\mathbf{T}_4} \end{cases} \tag{2.23}$$

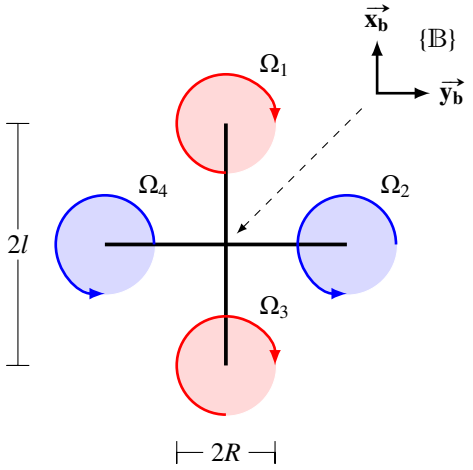


Figura 2.5: Rotor distribution and dimensions related to them.

For simplicity, it is convenient to represent the vectors above in frame {𝔹}, which follows the motion of the body. Doing so, the final form of the equations that describe the thrust torques is

$$\begin{cases} \boldsymbol{\tau}_1^{\mathbf{b}} = T_1 \left[l(+\mathbf{e_x}) + h(-\mathbf{e_z}) \right] \times ({}_{\mathbf{B}}\mathbf{R}_{\mathbf{R1}} \mathbf{e_z}) \\ \boldsymbol{\tau}_2^{\mathbf{b}} = T_2 \left[l(+\mathbf{e_y}) + h(-\mathbf{e_z}) \right] \times ({}_{\mathbf{B}}\mathbf{R}_{\mathbf{R2}} \mathbf{e_z}) \\ \boldsymbol{\tau}_3^{\mathbf{b}} = T_3 \left[l(-\mathbf{e_x}) + h(-\mathbf{e_z}) \right] \times ({}_{\mathbf{B}}\mathbf{R}_{\mathbf{R3}} \mathbf{e_z}) \\ \boldsymbol{\tau}_4^{\mathbf{b}} = T_4 \left[l(-\mathbf{e_y}) + h(-\mathbf{e_z}) \right] \times ({}_{\mathbf{B}}\mathbf{R}_{\mathbf{R4}} \mathbf{e_z}) \end{cases} \tag{2.24}$$

If flapping effects are neglected, the thrust torques take a simpler form. In this situation the matrices ${}_{\mathbf{B}}\mathbf{R}_{\mathbf{Ri}}$ become the identity, and the cross products in z are null. The sum of the thrust torque components for this case can be written as

$$\sum_{i=1}^{4} \boldsymbol{\tau}_{\mathbf{i}}^{\mathbf{b}} = l \begin{bmatrix} T_4 - T_2 \\ T_1 - T_3 \\ 0 \end{bmatrix}. \tag{2.25}$$

This form is more interesting for vehicle control and will be widely used.

2.3.4 Aerodynamic Torque

As with the thrust force, aerodynamic torque is caused by the aerodynamic forces acting on the rotor blades. Its main component results from drag forces, and for this reason many authors call it drag torque. Using equation (2.21),

$$\overrightarrow{\mathbf{Q_i}} = (-1)^i Q_i \left(\overrightarrow{\mathbf{z_{r,i}}}\right).$$

Again, since this torque acts around $\overrightarrow{\mathbf{z_{r,i}}}$, in order to represent it in $\{\mathbb{B}\}$, it is necessary to use the rotation matrix ${}_{\mathbf{B}}\mathbf{R}_{\mathbf{Ri}}$. Therefore, the aerodynamic torque represented in the body frame is

$$\mathbf{Q_i^b} = Q_i \left({}_{\mathbf{B}}\mathbf{R}_{\mathbf{Ri}}\mathbf{e_z}\right). \tag{2.26}$$

2.3.5 Lift Dissymmetry

Lift dissymmetry is a phenomenon caused by the imbalance between the lift forces acting on the propeller blades. When the vehicle is in translational flight, the advancing blades are subjected to a greater relative airspeed, while the retreating blades experience a lower speed. The greater relative speed on the advancing blade generates an increase in its lift, and the opposite occurs on the retreating blade. This concept is illustrated in Figure 2.6.

In helicopters there are techniques to mitigate the effect of lift dissymmetry. One example is the independent variation of blade pitch, with the aim of controlling the angle of attack individually and balancing the lift. Since quadrotor propellers are not articulated, this strategy cannot be implemented.

This section explains two changes in vehicle dynamics caused by lift dissymmetry. The first results from gyroscopic effects and does not impact the dynamics as strongly. The second causes more significant changes, since it changes the rotor plane and, consequently, all forces and torques generated by it.

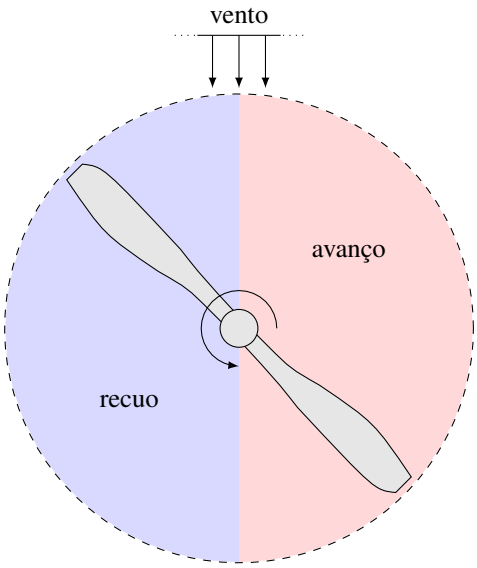


Figura 2.6: Illustration of lift dissymmetry.

Gyroscopic Precession

The difference in lift generated by the propeller blades causes a torque $\vec{\tau_d} \in \mathbb{R}^3$ to arise, see Figure 2.7. As will be shown later, this torque also generates a displacement in the propeller plane, but here the interest is focused on the gyroscopic effect it produces on the vehicle.

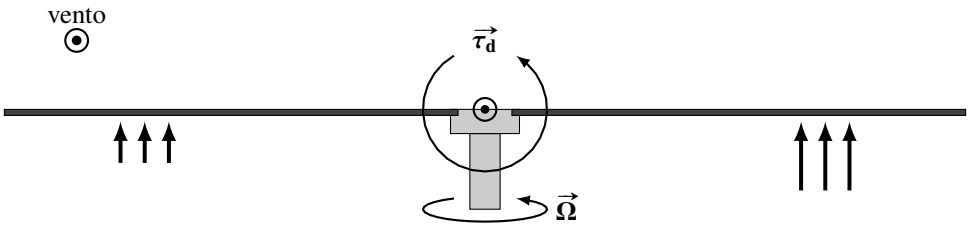


Figura 2.7: Torque on the propeller resulting from lift dissymmetry.

It follows directly from Euler's equation for the variation of angular momentum that a torque $\vec{\mathbf{T}}$ applied about an axis perpendicular to the axis of rotation, i.e. perpendicular to $\vec{\mathbf{L}}$, produces a rotation around an axis perpendicular both to $\vec{\mathbf{T}}$ and to $\vec{\mathbf{L}}$, according to

$$\vec{\mathbf{T}} = \vec{\mathbf{\Omega_p}} \times \vec{\mathbf{L}}$$

The rotational speed around this axis is called the precession speed, represented by the vector $\vec{\mathbf{\Omega_p}} \in \mathbb{R}^3$.

In a conventional helicopter, gyroscopic precession causes the nose to rise and the tail

to descend. Quadrotors, on the other hand, are composed of a set of two pairs of propellers rotating in opposite directions. For this reason, the effect of precession in these vehicles can be neglected, since one pair of propellers tends to cancel the torque exerted by the other.

In Section 2.3.7, gyroscopic effects resulting from rotor rotation will be studied and this concept will be explained in greater detail.

Blade Flapping

Lift dissymmetry causes the blades to be subjected to different forces. It is already known that this effect produces a torque on the propellers. However, this imbalance causes the propeller blades to bend, tilting the rotation plane. This effect impacts the dynamics of quadrotor vehicles more than the previous one.

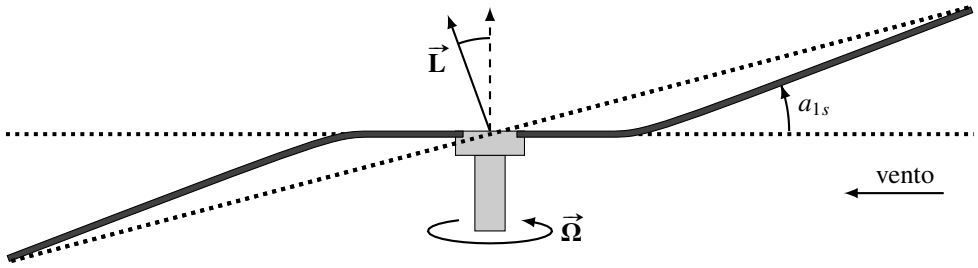


Figura 2.8: Blade flapping due to lift dissymmetry, adapted from [10].

In fact, the propeller blades oscillate up and down once per revolution. Figure 2.8 shows the propeller configuration when its blades are parallel to the airflow. This is the situation of maximum blade displacement. When they are perpendicular to the flow, there is a lateral displacement $b_{1s} \in \mathbb{R}$ of the rotation plane. Figure 2.9 shows these two situations.

Besides the displacements due to lift dissymmetry, the propeller blades flex even in hover. This flexure results from a phenomenon known as coning. Coning is the result of the lift force on the propeller combined with the centrifugal inertial force. The former causes the blades to move upward. The latter exerts a force that makes the blades tend to align with the disk plane. The resultant of these two loads on the blades tilts them by an angle $a_{0s} \in \mathbb{R}$, as shown in Figure 2.10.

In helicopters the blades are heavy and the action of weight on coning must also be considered. Quadrotors, on the other hand, have light propellers, allowing their weights

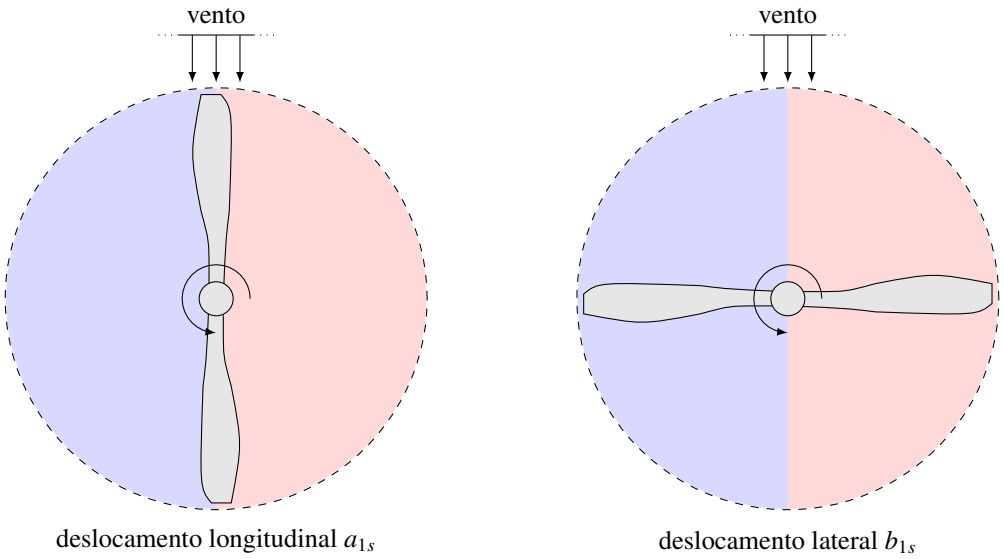


Figura 2.9: Propeller orientation in the longitudinal and lateral displacement situations.

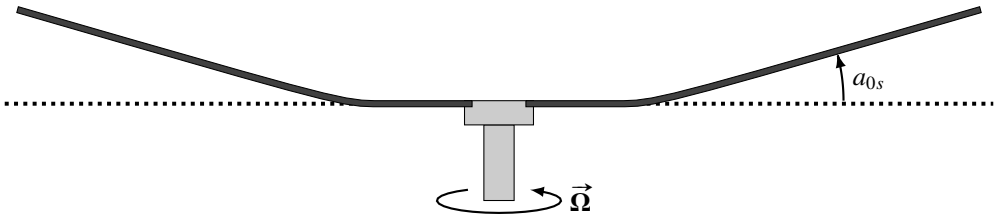


Figura 2.10: Coning due to lift and centrifugal inertial forces.

to be neglected.

Consider then the flapping angle $\beta \in \mathbb{R}$, defined as the total displacement between the horizontal plane of the vehicle and the propeller rotation plane. Using the previous information, this equation can be written as

$$\beta(\theta_r,\ \Omega) = a_{0s} - a_{1s}\cos(\theta_r) + b_{1s}\sin(\theta_r). \tag{2.27}$$

Where $\theta_r \in \mathbb{R}$ is the heading angle of the rotor, measured from the situation of maximum longitudinal displacement, see the first image in Figure 2.9.

It is evident that calculating the value of the flapping angle by (2.27) depends on knowing the values of the parameters a_{0s}, a_{1s}, and b_{1s}, in addition to the rotor rotation angle. Information on how to obtain these values is outside the scope of this work and can be found in [10] and [19].

The inclusion of the flapping effect in quadrotor dynamics must consider all the effects presented. Since coning acts in the same way on both blades, it does not contribute to the

inclination of the rotation plane. Therefore, the effective inclination of this plane will be

$$\beta_e(\theta_r,\ \Omega) = b_{1s}\sin(\theta_r) - a_{1s}\cos(\theta_r). \tag{2.28}$$

Thus, the rotation matrix ${}_{\mathbf{B}}\mathbf{R}_{\mathbf{Ri}} \in \mathrm{SO}(3)$, which maps from frame $\{\mathbb{R}_i\}$ to frame $\{\mathbb{B}\}$, where i denotes the propeller index, can be understood as

$${}_{\mathbf{B}}\mathbf{R}_{\mathbf{Ri}} = \mathbf{R}_{\mathbf{z}}(\theta_{r,i})\mathbf{R}_{\mathbf{y}}(\beta'_{e,i}). \tag{2.29}$$

Note that an auxiliary angle $\beta'_{e,i}$ was used. This occurs because it was considered that initially the body and propeller frames are aligned. Therefore, for the pair of propellers that, when viewed from above, rotates clockwise, one has $\beta'_{e,i} = \beta_{e,i}(\theta_{r,i},\ \Omega_i)$. For the pair that rotates counterclockwise, $\beta'_{e,i} = \beta_{e,i}(\text{-}\theta_{r,i},\ \Omega_i)$.

In summary, an increase in the horizontal speed of the vehicle causes the blade flapping effect to arise. This can also occur due to wind currents, since these would also generate a change in the relative airspeed seen by the rotors. For helicopters, the tilt of the disk plane due to flapping is on the order of one degree for every 10 m/s, regardless of vehicle speed [20].

2.3.6 Drag Force

Like any aerial vehicle, quadrotors are subject to drag forces resulting from the relative velocity between the vehicle and the airflow. The model of these forces is strongly related to the construction characteristics of the vehicle. In simplified form, the drag forces acting on the vehicle can be described as

$$\overrightarrow{\mathbf{F_D}} = F_d\left(-\overrightarrow{\mathbf{e}_\mathbf{v}}\right). \tag{2.30}$$

Where $\overrightarrow{\mathbf{e}_\mathbf{v}}$ is the unit vector pointing in the direction of the vehicle velocity and F_d is the value of the drag force resulting from the relative fluid velocity.

2.3.7 Rotor Dynamics

The rotors of a quadrotor are generally driven by brushless direct current (BLDC) motors, accompanied by electronic speed controllers (ESCs). These devices are responsible for

controlling motor rotational speed, and their input is given through a PWM signal, with frequency of 50 Hz. The pulse time range goes from 1 ms to 2 ms, and it is generally necessary to keep the signal at 1 ms to start the motor, as a safety measure.

Although control of motor rotational speed is performed by the ESCs, the transfer function that relates the input command to this speed is not unitary. In fact, it is possible to approximate this relation by a first-order model [19], similar to that of a common direct current motor,

$$\frac{d\Omega_i}{dt} = -\frac{1}{\tau_m}\Omega_i + \frac{K_m}{\tau_m}U_i - Q_i. \tag{2.31}$$

Where $K_m \in \mathbb{R}$ is the motor DC gain, $\tau_m \in \mathbb{R}$ the motor time constant, $U_i \in \mathbb{R}$ is the input signal, and $Q_i \in \mathbb{R}$ the motor shaft torque, see equation (2.26).

Induced Torque

The torque induced by the rotor on the vehicle can be obtained directly from Euler's equation for the variation of angular momentum. This torque represents the influence of rotor rotation on vehicle dynamics and can be divided into torque induced by the variation in rotational speed and by the Coriolis effect.

The variation of the angular momentum of a body is equal to the sum of the external torques applied to it, i.e.

$$\sum \vec{\boldsymbol{\tau}} = \frac{d}{dt}\vec{\mathbf{H}} = \frac{d}{dt}\left(\boldsymbol{\mathcal{I}}\vec{\boldsymbol{\omega}}\right). \tag{2.32}$$

Where $\boldsymbol{\mathcal{I}} \in \mathbb{R}^{3\times3}$ is the inertia matrix of the body. Since this matrix varies in a nontrivial way, it is often preferable to work in a non-inertial frame that follows the rotational motion. Applying the Coriolis theorem for differentiation in a rotating coordinate system to equation (2.32),

$$\sum \vec{\boldsymbol{\tau}} = \frac{d}{dt}\vec{\mathbf{H}} = \frac{d}{dt}\left(\boldsymbol{\mathcal{I}}\vec{\boldsymbol{\omega}}\right) + \vec{\boldsymbol{\omega}}_{\mathbf{r}} \times \left(\boldsymbol{\mathcal{I}}\vec{\boldsymbol{\omega}}\right). \tag{2.33}$$

Where $\vec{\boldsymbol{\omega}}_{\mathbf{r}}$ represents the angular velocity of the chosen frame. If this frame follows the

body frame, the final form is obtained

$$\sum \vec{\boldsymbol{\tau}} = \frac{d}{dt}\vec{\mathbf{H}} = \boldsymbol{\mathcal{I}}\,\frac{d\vec{\boldsymbol{\omega}}}{dt} + \vec{\boldsymbol{\omega}_{\mathbf{r}}} \times (\boldsymbol{\mathcal{I}}\vec{\boldsymbol{\omega}})\,. \tag{2.34}$$

Applying this concept to the rotors that compose the vehicle, it is convenient to differentiate equation (2.32) in the body frame $\{\mathbb{B}\}$. Thus, the angular velocity of the frame will be $\boldsymbol{\omega}_{\mathbf{r}} = \boldsymbol{\omega}^{\mathbf{b}}$, the rotor angular velocity $\boldsymbol{\omega} = \Omega\mathbf{e}_{\mathbf{z}}$, and the inertia matrix $\boldsymbol{\mathcal{I}} = \text{diag}\left(I_x, I_y, I_r\right)$. Note that the components $I_x \in \mathbb{R}$ and $I_y \in \mathbb{R}$ of the inertia matrix are not constant; only $I_r \in \mathbb{R}$, the inertia about the rotor axis, is constant. Thus, equation (2.33) is applied to determine the torque on the rotor, viewed in the body coordinate system,

$$\boldsymbol{\tau}^{\mathbf{b}}_{\mathbf{r,i}} + \mathbf{Q}_{\mathbf{i}} = I_r \begin{bmatrix} \Omega\omega_y^b \\ -\Omega\omega_x^b \\ d\Omega/dt \end{bmatrix}. \tag{2.35}$$

Since the rotors are fixed to the quadrotor structure, this torque will be seen on the vehicle with opposite sign. Therefore, the torque induced by the rotors on the structure will be $-\boldsymbol{\tau}^{\mathbf{b}}_{\mathbf{r}}$. For rotors 1 and 3, one has $\Omega = \Omega_i$, since both rotate clockwise, so the angular momentum will be aligned with $\vec{\mathbf{z}}_{\mathbf{r,i}}$. For rotors 2 and 4, one has $\Omega = -\Omega_i$, because they rotate counterclockwise and the angular momentum will be aligned with $-\vec{\mathbf{z}}_{\mathbf{r,i}}$.

Defining then $\Omega_R = \Omega_1 - \Omega_2 + \Omega_3 - \Omega_4$ as the residual rotor rotational speed, the torque induced on the vehicle will be

$$-\boldsymbol{\tau}^{\mathbf{b}}_{\mathbf{r}} = I_r \begin{bmatrix} -\Omega_R\omega_y^b \\ \Omega_R\omega_x^b \\ d\Omega_R/dt \end{bmatrix} + \sum \mathbf{Q}_{\mathbf{i}}. \tag{2.36}$$

2.4 Rotation Matrices

There are several possible parametrizations for orientation, e.g. Quaternions and Euler angles. Often viewed as a subgroup of Euler angles, there are also Tait-Bryan angles. These are characterized by composing rotations around the x, y, and z axes in any order. For aerial, maritime, and underwater vehicles, the most common choice is Tait-Bryan angles, and this will be the one used in this work.

In aeronautical engineering it is customary to fix the x-axis pointing toward the front of

the aircraft and the z-axis downward, with the y-axis defined to the right by the right-hand rule. Rotations around the x, y, and z axes are called *roll*, *pitch*, and *yaw*, respectively.

The parametrization used will be z-y-x, or *yaw-pitch-roll*. In order to parametrize the orientation matrix, one must then compute the rotations around each of these axes, as shown below.

$$\mathbf{R_z}(\psi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \text{rotation around the } z\text{-axis} \tag{2.37}$$

$$\mathbf{R_y}(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \qquad \text{rotation around the } y\text{-axis} \tag{2.38}$$

$$\mathbf{R_x}(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix} \qquad \text{rotation around the } x\text{-axis} \tag{2.39}$$

Therefore, the orientation matrix ${}_{\mathbf{E}}\mathbf{R}_{\mathbf{B}}(\boldsymbol{\eta}) \in \mathrm{SO}(3)$, where $\boldsymbol{\eta} \in \mathbb{R}^3$ is the vector of Tait-Bryan angles, takes the form below.

$${}_{\mathbf{E}}\mathbf{R}_{\mathbf{B}}(\boldsymbol{\eta}) = \mathbf{R_z}(\psi)\mathbf{R_y}(\theta)\mathbf{R_x}(\phi)$$

$${}_{\mathbf{E}}\mathbf{R}_{\mathbf{B}}(\boldsymbol{\eta}) = \begin{bmatrix} \mathrm{c}\psi\mathrm{c}\theta & \mathrm{c}\psi\mathrm{s}\phi\mathrm{s}\theta - \mathrm{c}\phi\mathrm{s}\psi & \mathrm{s}\phi\mathrm{s}\psi + \mathrm{c}\phi\mathrm{c}\psi\mathrm{s}\theta \\ \mathrm{c}\theta\mathrm{s}\psi & \mathrm{c}\phi\mathrm{c}\psi + \mathrm{s}\phi\mathrm{s}\psi\mathrm{s}\theta & \mathrm{c}\phi\mathrm{s}\psi\mathrm{s}\theta - \mathrm{c}\psi\mathrm{s}\phi \\ -\mathrm{s}\theta & \mathrm{c}\theta\mathrm{s}\phi & \mathrm{c}\phi\mathrm{c}\theta \end{bmatrix}^{4} \tag{2.40}$$

In Section 2.3.5, the equation of the rotation matrix that relates frames $\{\mathbb{R}_i\}$ and $\{\mathbb{B}\}$ was obtained. Using the matrix forms shown in equations (2.39) and (2.38),

$${}_{\mathbf{B}}\mathbf{R}_{\mathbf{Ri}}(\theta_r, \beta_e) = \begin{bmatrix} \mathrm{c}\theta_r\mathrm{c}\beta_e & -\mathrm{s}\theta_r & \mathrm{c}\theta_r\mathrm{s}\beta_e \\ \mathrm{s}\theta_r\mathrm{c}\beta_e & \mathrm{c}\theta_r & \mathrm{s}\theta_r\mathrm{s}\beta_e \\ -\mathrm{s}\beta_e & 0 & \mathrm{c}\beta_e \end{bmatrix}, \tag{2.41}$$

[4] In this context, the letter c is being used to denote the cosine function and s the sine function.

with subscript *i* omitted for simplicity.

2.5 Angular Velocity Transformation

Considering the axes $\{\vec{\mathbf{x}_\mathbf{b}}, \vec{\mathbf{y}_\mathbf{b}}, \vec{\mathbf{z}_\mathbf{b}}\}$ fixed in frame $\mathbb{B}$ and the *roll*, *pitch*, and *yaw* angles, it is easy to observe that these angles constitute rotations around three different principal axes. On the other hand, the angular velocity vector, $\vec{\boldsymbol{\omega}}$, can be seen as the rate of change of the angular displacement of a rigid body around a principal axis of rotation.

Therefore, in order to relate $\dot{\phi}$, $\dot{\theta}$, and $\dot{\psi}$ to $\boldsymbol{\omega}_b$, representation of the vector $\vec{\boldsymbol{\omega}}$ in frame $\mathbb{B}$, it is necessary to define two auxiliary axes $\{\vec{\mathbf{x}_1}, \vec{\mathbf{y}_1}, \vec{\mathbf{z}_1}\}$ and $\{\vec{\mathbf{x}_2}, \vec{\mathbf{y}_2}, \vec{\mathbf{z}_2}\}$, fixed in frames $\mathbb{B}_1$ and $\mathbb{B}_2$. In this way, *yaw* occurs around $\vec{\mathbf{z}_\mathbf{e}} = \vec{\mathbf{z}_1}$ and maps from $\mathbb{E}$ to $\mathbb{B}_1$; *pitch* occurs around $\vec{\mathbf{y}_1} = \vec{\mathbf{y}_2}$ and maps from $\mathbb{B}_1$ to $\mathbb{B}_2$; and *roll* occurs around $\vec{\mathbf{x}_2} = \vec{\mathbf{x}_\mathbf{b}}$, mapping from $\mathbb{B}_2$ to $\mathbb{B}$. The relation can then be established as follows:

$$
\begin{aligned}
\vec{\boldsymbol{\omega}} &= \dot{\psi}\vec{\mathbf{z}_\mathbf{e}} + \dot{\theta}\vec{\mathbf{y}_1} + \dot{\phi}\vec{\mathbf{x}_2} = \dot{\psi}\,\vec{\mathbf{z}} + \dot{\theta}\vec{\mathbf{y}_2} + \dot{\phi}\vec{\mathbf{x}_\mathbf{b}} \\
\boldsymbol{\omega}^\mathbf{b} &= \dot{\psi}\left(\mathbf{R}_\mathbf{y}\mathbf{R}_\mathbf{x}\right)^\mathrm{T}\mathbf{z} + \dot{\theta}\mathbf{R}_\mathbf{x}{}^\mathrm{T}\mathbf{y} + \dot{\phi}\mathbf{x} \\
\boldsymbol{\omega}^\mathbf{b} &= \dot{\psi}\begin{bmatrix} -\sin\theta \\ \sin\phi\cos\theta \\ \cos\phi\cos\theta \end{bmatrix} + \dot{\theta}\begin{bmatrix} 0 \\ \cos\phi \\ -\sin\phi \end{bmatrix} + \dot{\phi}\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \\
\mathbf{J}_\mathbf{r}(\boldsymbol{\eta}) &\triangleq \begin{bmatrix} -\sin\theta & 0 & 1 \\ \cos\theta\sin\phi & \cos\phi & 0 \\ \cos\phi\cos\theta & -\sin\phi & 0 \end{bmatrix} \qquad (2.42) \\
\boldsymbol{\omega}^\mathbf{b} &= \mathbf{J}_\mathbf{r}(\boldsymbol{\eta})\dot{\boldsymbol{\eta}} \qquad (2.43)
\end{aligned}
$$

Note that the matrix $\mathbf{J}_r(\boldsymbol{\eta})$ relates the rate of variation of the parametrization to angular velocity. This matrix is called the representation Jacobian.

By inverting equation (2.42), it is possible to perform the inverse mapping, i.e. write $\dot{\boldsymbol{\eta}}$ as a function of angular velocity $\boldsymbol{\omega}^\mathbf{b}$:

$$
\mathbf{J}_\mathbf{r}{}^{-1}(\boldsymbol{\eta}) = \begin{bmatrix} 0 & \sin\phi/\cos\theta & \cos\phi/\cos\theta \\ 0 & \cos\phi & -\sin\phi \\ 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \end{bmatrix} \qquad (2.44)
$$

$$\dot{\boldsymbol{\eta}} = \mathbf{J}_{\mathbf{r}}^{-1}(\boldsymbol{\eta})\boldsymbol{\omega}^{\mathbf{b}} \tag{2.45}$$

2.6 Newton-Euler Formalism

The Newton-Euler formalism is a simple way to formulate the dynamics of a rigid body. Basically, this methodology consists of combining Newton's and Euler's equations, with the goal of concisely representing the translational and rotational dynamics of a body. Another equation used in the literature to describe dynamic characteristics is Lagrange's equation. An approach based on this method can be found in [2].

With respect to the coordinate system fixed to the vehicle $\{\mathbb{B}\}$, the Newton-Euler equation can be written as

$$\begin{bmatrix} m\,\mathbf{0} \\ \mathbf{0}\;\boldsymbol{\mathcal{I}} \end{bmatrix}\begin{bmatrix} \dot{\boldsymbol{\nu}}^{\mathbf{b}} \\ \dot{\boldsymbol{\omega}}^{\mathbf{b}} \end{bmatrix} + \begin{bmatrix} \boldsymbol{\omega}^{\mathbf{b}} \times m\boldsymbol{\nu}^{\mathbf{b}} \\ \boldsymbol{\omega}^{\mathbf{b}} \times \boldsymbol{\mathcal{I}}\boldsymbol{\omega}^{\mathbf{b}} \end{bmatrix} = \begin{bmatrix} \mathbf{f}^{\mathbf{b}} \\ \boldsymbol{\tau}^{\mathbf{b}} \end{bmatrix}. \tag{2.46}$$

Throughout this chapter, the forces and torques that will be considered in the final quadrotor model were determined.

Since measurements of vehicle position are made in the inertial frame $\{\mathbb{E}\}$, the Coriolis forces disappear from the translational term. That is,

$$\begin{bmatrix} m\,\mathbf{0} \\ \mathbf{0}\;\boldsymbol{\mathcal{I}} \end{bmatrix}\begin{bmatrix} \dot{\boldsymbol{\nu}}^{\mathbf{e}} \\ \dot{\boldsymbol{\omega}}^{\mathbf{b}} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\omega}^{\mathbf{b}} \times \boldsymbol{\mathcal{I}}\boldsymbol{\omega}^{\mathbf{b}} \end{bmatrix} = \begin{bmatrix} {}_{\mathbf{E}}\mathbf{R}_{\mathbf{B}}\mathbf{f}^{\mathbf{b}} \\ \boldsymbol{\tau}^{\mathbf{b}} \end{bmatrix}. \tag{2.47}$$

It is known that the cross product by a vector $\mathbf{a} \in \mathbb{R}^3$ can be represented through the skew-symmetric matrix

$$\mathbf{S}(\mathbf{a}) = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \quad \mathbf{S}(\mathbf{a}) \in \mathrm{so}(3). \tag{2.48}$$

Therefore, using equations (2.47) and (2.48), the vehicle dynamics is described by the

equation:

$$\begin{cases} m\dot{\boldsymbol{\nu}}^{\mathbf{e}} &= {}_{\mathbf{E}}\mathbf{R}_{\mathbf{B}}\mathbf{f}^{\mathbf{b}} \\ \boldsymbol{\mathcal{I}}\dot{\boldsymbol{\omega}}^{\mathbf{b}} &= \boldsymbol{\tau}^{\mathbf{b}} - \mathbf{S}\left(\boldsymbol{\omega}^{\mathbf{b}}\right)\boldsymbol{\mathcal{I}}\boldsymbol{\omega}^{\mathbf{b}} \\ {}_{\mathbf{E}}\dot{\mathbf{R}}_{\mathbf{B}} &= \mathbf{S}\left(\boldsymbol{\omega}^{\mathbf{b}}\right){}_{\mathbf{E}}\mathbf{R}_{\mathbf{B}} \end{cases} \tag{2.49}$$

Finally, the motor model (2.31), the thrust (2.22) and drag (2.30) forces, and the thrust (2.25), aerodynamic (2.26), and rotor-induced (2.36) torques are used to write the complete equation of quadrotor dynamics:

$$\begin{cases} m\dot{\boldsymbol{\nu}}^{\mathbf{e}} &= mg\mathbf{e}_{\mathbf{z}} - {}_{\mathbf{E}}\mathbf{R}_{\mathbf{B}}\left[\left(\boldsymbol{\nu}^{\mathbf{e}}/\|\boldsymbol{\nu}^{\mathbf{e}}\|\right)F_d + \kappa_T\sum_{i=1}^{4}\left(\Omega_i^2\,{}_{\mathbf{B}}\mathbf{R}_{\mathbf{Ri}}\mathbf{e}_{\mathbf{z}}\right)\right] \\ \boldsymbol{\mathcal{I}}\dot{\boldsymbol{\omega}}^{\mathbf{b}} &= \sum_{i=1}^{4}\left(\boldsymbol{\tau}_{\mathbf{i}}^{\mathbf{b}}\right) - \kappa_Q\sum_{i=1}^{4}\left(\Omega_i\left|\Omega_i\right|{}_{\mathbf{B}}\mathbf{R}_{\mathbf{Ri}}\mathbf{e}_{\mathbf{z}}\right) - I_r\begin{bmatrix}\Omega_R\omega_y^b \\ -\Omega_R\omega_x^b \\ \dot{\Omega}_R\end{bmatrix} - \mathbf{S}\left(\boldsymbol{\omega}^{\mathbf{b}}\right)\boldsymbol{\mathcal{I}}\boldsymbol{\omega}^{\mathbf{b}} \\ {}_{\mathbf{E}}\dot{\mathbf{R}}_{\mathbf{B}} &= \mathbf{S}\left(\boldsymbol{\omega}^{\mathbf{b}}\right){}_{\mathbf{E}}\mathbf{R}_{\mathbf{B}} \\ \dot{\Omega}_i &= (K_m/\tau_m)U_i - \Omega_i/\tau_m - d\Omega_i\left|\Omega_i\right| \end{cases} \tag{2.50}$$

Note that, as will be done throughout this work, equation (2.21) was used for both aerodynamic torque and thrust force.

The model found in equation (2.50) considers all the forces and torques discussed earlier. If flapping and drag force are neglected, the model can be written in a simpler way, omitting only the differential equations for $\dot{\Omega}_i$.

$$\begin{cases} m\dot{\boldsymbol{\nu}}^{\mathbf{e}} &= mg\mathbf{e}_{\mathbf{z}} + \kappa_T(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2){}_{\mathbf{E}}\mathbf{R}_{\mathbf{B}}\mathbf{e}_{\mathbf{z}} \\ I_{xx}\dot{\omega}_x^b &= \omega_y^b\omega_z^b(I_{yy} - I_{zz}) + l\kappa_T(\Omega_4^2 - \Omega_2^2) - I_r\omega_y^b\Omega_R \\ I_{yy}\dot{\omega}_y^b &= \omega_z^b\omega_x^b(I_{zz} - I_{xx}) + l\kappa_T(\Omega_1^2 - \Omega_3^2) + I_r\omega_x^b\Omega_R \\ I_{zz}\dot{\omega}_z^b &= \omega_x^b\omega_y^b(I_{xx} - I_{yy}) + \kappa_Q(-\Omega_1^2 + \Omega_2^2 - \Omega_3^2 + \Omega_4^2) + I_r\dot{\Omega}_R \\ {}_{\mathbf{E}}\dot{\mathbf{R}}_{\mathbf{B}} &= \mathbf{S}(\boldsymbol{\omega}_b){}_{\mathbf{E}}\mathbf{R}_{\mathbf{B}} \end{cases} \tag{2.51}$$

It is still possible to use the orientation representation by Tait-Bryan angles, equation (2.40), and use the representation Jacobian, equations (2.44) and (2.45). Doing this,

the rotation matrix dynamics is replaced and its value can be written explicitly,

$$\begin{cases} m\dot{v}_x^e &= -(\sin\phi\sin\psi + \cos\phi\cos\psi\sin\theta)\,\kappa_T \sum_{i=1}^4 \Omega_i^2 \\ m\dot{v}_y^e &= -(\cos\phi\sin\psi\sin\theta - \cos\psi\sin\phi)\,\kappa_T \sum_{i=1}^4 \Omega_i^2 \\ m\dot{v}_z^e &= mg - (\cos\theta\cos\phi)\,\kappa_T \sum_{i=1}^4 \Omega_i^2 \\ I_{xx}\dot{\omega}_x^b &= \omega_y^b\omega_z^b(I_{yy} - I_{zz}) + l\kappa_T(\Omega_4^2 - \Omega_2^2) - I_r\omega_y^b\Omega_R \\ I_{yy}\dot{\omega}_y^b &= \omega_z^b\omega_x^b(I_{zz} - I_{xx}) + l\kappa_T(\Omega_1^2 - \Omega_3^2) + I_r\omega_x^b\Omega_R \\ I_{zz}\dot{\omega}_z^b &= \omega_x^b\omega_y^b(I_{xx} - I_{yy}) + \kappa_Q(-\Omega_1^2 + \Omega_2^2 - \Omega_3^2 + \Omega_4^2) + I_r\dot{\Omega}_R \\ \dot{\phi} &= \omega_x^b + (\sin\phi\tan\theta)\,\omega_y^b + (\cos\phi\tan\theta)\,\omega_z^b \\ \dot{\theta} &= \cos\phi\,\omega_y^b - \sin\phi\,\omega_z^b \\ \dot{\psi} &= (\sin\phi/\cos\theta)\,\omega_y^b + (\cos\phi/\cos\theta)\,\omega_z^b \end{cases} \tag{2.52}$$

Depending on the intended application, one model may be more suitable than the other. The model in equation (2.50) is the most suitable to be used in simulations, since it considers a greater number of effects found in practice. On the other hand, the model in equation (2.51) is simpler and therefore more commonly used in the development of control laws. The vehicle dynamics can be simplified even further, e.g. by neglecting less relevant dynamics and applying small-angle approximations. These simplifications are useful for some control strategies and will be shown in Chapter 3.

2.7 Conclusion

In this chapter, the dynamic model of a quadrotor vehicle was developed in detail. This work consisted of studying and compiling several texts from the literature, serving as a single source of reference for future work in the same area. The advantages and disadvantages of the models found were distinguished, allowing a simple and direct choice of the most appropriate model according to the situation. The analysis of these factors from the control perspective will be continued in Chapter 3.

Capítulo 3

Control Strategies

In Chapter 2 the vehicle's model was determined. Having knowledge of this model it is possible to propose and study control laws. In turn, these laws must be able to act on the vehicle, making it possible to perform desired tasks. As will be seen in this chapter, the problem of controlling a quadrotor can be divided into two main issues: controlling the position and controlling the orientation.

This situation highlights the cascade structure inherent to quadrotor vehicles. Discussing the problem informally, commanding the vehicle forward, i.e. along $\vec{\mathbf{x}_e}$, boils down to tilting it by a certain angle θ around $\vec{\mathbf{y}_b}$. In reality, simply tilting the vehicle around this axis is not enough when there have already been changes in the heading[1] of the vehicle. Therefore, it is evident that controlling the position of a quadrotor can be thought of as cascade control, where: the outer loop, which wants to control the position, sends tilt commands to the inner loop.

When it comes to control, it is known that parametric uncertainties can drastically influence system performance. In a quadrotor, these uncertainties are mainly related to the values of the vehicle mass, inertia matrix, motor parameters and aerodynamic constants. In particular, the body mass and the inertia matrix are easily changed if new components are added to the vehicle.

In view of these factors, this chapter aims to study control laws that can be used to command the vehicle. To facilitate obtaining these laws, the first step will consist of presenting simplifications of the model obtained in the previous chapter. With these simplifications it will be possible to choose among a set of models to design the control

[1]The vehicle's heading angle is the same as the *yaw* angle.

strategies. The problem of parametric uncertainties will be addressed considering variations in the vehicle mass. Finally, all strategies will be compared in order to determine their advantages and disadvantages.

3.1 Input Mapping

Quadrotors are multiple-input multiple-output (MIMO) systems, in which the vehicle's control inputs are the rotational speeds Ω_i of the rotors and the outputs are the position and heading of the vehicle. However, these control signals do not have an intuitive relationship with the quadrotor's dynamics. In the maritime and aerospace industry it is very common to find this situation and the solution used is to define virtual inputs that have a clearer impact on the vehicle's response. Therefore, let u_i be these auxiliary inputs, where u_1 is the force exerted on the body and $u_{2,3,4}$ the torques around the axes. The mapping $\mathbf{f_u} : \mathbb{R}^4 \mapsto \mathbb{R}^4$ that takes Ω_i to u_i is

$$\mathbf{u} = \mathbf{f_u}(\mathbf{\Omega}) = \begin{bmatrix} \kappa_T & \kappa_T & \kappa_T & \kappa_T \\ 0 & -l\kappa_T & 0 & l\kappa_T \\ l\kappa_T & 0 & -l\kappa_T & 0 \\ -\kappa_Q & \kappa_Q & -\kappa_Q & \kappa_Q \end{bmatrix} \begin{bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \end{bmatrix} = \mathbf{M\Omega} \tag{3.1}$$

and its inverse is given by:

$$\mathbf{\Omega} = \mathbf{f_u}^{-1}(\mathbf{u}) = \frac{1}{4l\kappa_T} \begin{bmatrix} l & 0 & 2 & -l\kappa_T\kappa_Q^{-1} \\ l & -2 & 0 & l\kappa_T\kappa_Q^{-1} \\ l & 0 & -2 & -l\kappa_T\kappa_Q^{-1} \\ l & 2 & 0 & l\kappa_T\kappa_Q^{-1} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \mathbf{M}^{-1}\mathbf{u}. \tag{3.2}$$

The vector $\mathbf{u} \in \mathbb{R}^4$ denotes the virtual inputs u_i, $\mathbf{\Omega} \in \mathbb{R}^4$ the squared propeller speeds Ω_i^2 and $\mathbf{M} \in \mathbb{R}^{4\times4}$ the matrix representation of the linear transformation defined by the

mapping $\mathbf{f_u}$. This mapping applied to (2.51) produces the simplified model

$$\begin{cases} m\dot{\mathbf{v}}^{\mathbf{e}} &= mg\mathbf{e_z} - u_1 {}_{\mathbf{E}}\mathbf{R_B}\mathbf{e_z} \\ I_{xx}\dot{\omega}_x^b &= \omega_y^b\omega_z^b(I_{yy} - I_{zz}) + u_2 - I_r\omega_y^b\Omega_R \\ I_{yy}\dot{\omega}_y^b &= \omega_z^b\omega_x^b(I_{zz} - I_{xx}) + u_3 + I_r\omega_x^b\Omega_R \\ I_{zz}\dot{\omega}_z^b &= \omega_x^b\omega_y^b(I_{xx} - I_{yy}) + u_4 + I_r\dot{\Omega}_R \\ {}_{\mathbf{E}}\dot{\mathbf{R}}_{\mathbf{B}} &= \mathbf{S}(\omega_b){}_{\mathbf{E}}\mathbf{R_B} \end{cases} . \tag{3.3}$$

This model makes it clear that it is possible to exert torques on the three axes of the body. However, what is desired to control is the position and heading. As the input u_1 appears in all components of the vehicle's position, it is necessary to define other control signals so that the three degrees of freedom are commanded. Considering a slave orientation loop, the vehicle control is understood as a cascade structure, illustrated in Figure 3.1.

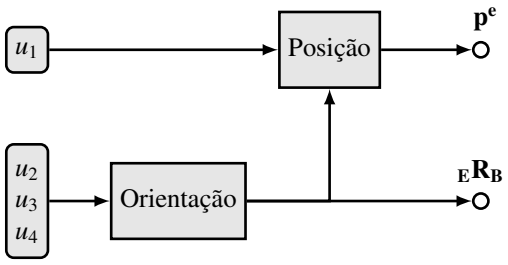


Figura 3.1: Cascade control structure of the vehicle.

It is worth noting that it is not always possible to determine the desired angular velocities of the motors through equation (3.2). This equation merely performs a linear combination of the inputs u_i to generate the squared velocities Ω_i^2. Thus, there is nothing preventing unfeasible control values from being generated, since there is saturation in the propeller speed. When this happens, a solution may consist of saturating the value of Ω_i so that it remains within an acceptable range $0 \leq \Omega_{\min} \leq \Omega_i \leq \Omega_{\max}$. By doing this, the forces u_i generated by these new angular velocities may be significantly distant from the desired ones. In Chapter 4, this problem will be formulated through concepts of control allocation, aiming to search for an optimal value of the vector $\mathbf{\Omega}$.

3.2 Simplified Models for Control

The model developed in Section 2.6 describes the differential equations that govern the dynamics of a quadrotor. However, this model is complex, making it difficult to extract control laws. In this section, changes to this model will be made. Based on these changes, some control strategies will be proposed and their performances compared by simulation. Note that simplifications are not always necessary for obtaining control algorithms, but they facilitate the process in some cases.

3.2.1 Motor Dynamics

First, it is considered that the motor dynamics are much faster than the vehicle's. In this way, the actuation of the motors can be considered instantaneous.

3.2.2 Rotor-Induced Torques

As already stated, when it comes to obtaining control strategies, less complex models may be more appropriate. This reduction in complexity can be achieved by disregarding lower-impact dynamics in the system. Changing the model (3.3), which already discards the effects resulting from blade flapping, the inertial torques induced by motor rotation, equation (2.36), are also disregarded. Thus, the body dynamics will be:

$$\begin{cases} m\dot{\boldsymbol{\nu}}^{\mathbf{e}} &= mg\mathbf{e_z} - u_{1\mathbf{E}}\mathbf{R_B}\mathbf{e_z} \\ I_{xx}\dot{\omega}_x^b &= \omega_y^b\omega_z^b(I_{yy} - I_{zz}) + u_2 \\ I_{yy}\dot{\omega}_y^b &= \omega_z^b\omega_x^b(I_{zz} - I_{xx}) + u_3 \\ I_{zz}\dot{\omega}_z^b &= \omega_x^b\omega_y^b(I_{xx} - I_{yy}) + u_4 \\ {}_{\mathbf{E}}\dot{\mathbf{R}}_{\mathbf{B}} &= \mathbf{S}(\boldsymbol{\omega}_b){}_{\mathbf{E}}\mathbf{R_B} \end{cases} . \tag{3.4}$$

3.2.3 Small Angles and Quasi-Stationary Flight

It is known that when the vehicle is close to hovering conditions, there are no major changes in the *pitch* and *roll* angles. However, the same cannot be said about the vehicle's heading, since the quadrotor remains stationary regardless of the value of this angle. These hypotheses directly impact the representation of the vehicle's orientation, allowing

the use of the simplifications below:

$$ {}_{\mathbf{E}}\mathbf{R}_{\mathbf{B}}(\boldsymbol{\eta})\mathbf{e}_{\mathbf{z}} \approx {}_{\mathbf{E}}\tilde{\mathbf{R}}_{\mathbf{B}}(\boldsymbol{\eta})\mathbf{e}_{\mathbf{z}} = \begin{bmatrix} \phi \mathrm{s}\psi + \theta \mathrm{c}\psi \\ \theta \mathrm{s}\psi - \phi \mathrm{c}\psi \\ 1 \end{bmatrix} \quad ; \quad \mathbf{J}_{\mathbf{r}}{}^{-1}(\boldsymbol{\eta}) \approx \tilde{\mathbf{J}}_{\mathbf{r}}^{-1}(\boldsymbol{\eta}) = \begin{bmatrix} 0 & \phi & 1 \\ 0 & 1 & -\phi \\ 1 & \phi\theta & \theta \end{bmatrix}. \tag{3.5} $$

Substituting equation (3.5) into (3.4), these hypotheses are included and the approximate vehicle dynamics will be

$$ \begin{cases} m \begin{bmatrix} \dot{v}_x^e \\ \dot{v}_y^e \end{bmatrix} &= u_1 \begin{bmatrix} -\cos\psi & -\sin\psi \\ -\sin\psi & \cos\psi \end{bmatrix} \begin{bmatrix} \theta \\ \phi \end{bmatrix} \\ m\dot{v}_z^e &= mg - u_1 \\ I_{xx}\dot{\omega}_x^b &= \omega_y^b \omega_z^b (I_{yy} - I_{zz}) + u_2 \\ I_{yy}\dot{\omega}_y^b &= \omega_z^b \omega_x^b (I_{zz} - I_{xx}) + u_3 \\ I_{zz}\dot{\omega}_z^b &= \omega_x^b \omega_y^b (I_{xx} - I_{yy}) + u_4 \\ \dot{\boldsymbol{\eta}} &= \tilde{\mathbf{J}}_{\mathbf{r}}^{-1} \boldsymbol{\omega}^{\mathbf{b}} \end{cases} . \tag{3.6} $$

Additionally, when the vehicle is close to hovering conditions, its angular velocity is very small, i.e. $\boldsymbol{\omega}^{\mathbf{b}} \approx \mathbf{0}$. This hypothesis makes the model even simpler, eliminating the torque components relative to gyroscopic effects, as shown in the equation below:

$$ \begin{cases} m \begin{bmatrix} \dot{v}_x^e \\ \dot{v}_y^e \end{bmatrix} &= u_1 \begin{bmatrix} -\cos\psi & -\sin\psi \\ -\sin\psi & \cos\psi \end{bmatrix} \begin{bmatrix} \theta \\ \phi \end{bmatrix} \\ m\dot{v}_z^e &= mg - u_1 \\ \boldsymbol{\mathcal{I}}\dot{\boldsymbol{\omega}}^{\mathbf{b}} &= [u_2 \; u_3 \; u_4]^T \\ \dot{\boldsymbol{\eta}} &= \boldsymbol{\omega}^{\mathbf{b}} \end{cases} . \tag{3.7} $$

Finally, still in the context of hovering flight, there remains one simplification that can be applied to equation (3.7). Since under the mentioned conditions the vehicle does not move quickly, it is assumed that the force input u_1 stays around an equilibrium point $u_1^* = mg$. With this hypothesis it becomes possible to act separately on the four degrees of freedom to be controlled, i.e. position and heading. Further defining the auxiliary inputs $\bar{u}_1 = g - u_1/m$, which acts as a *feedforward* control component, $\bar{\theta} = -(\phi \sin\psi + \theta \cos\psi)$ and $\bar{\phi} = \phi \cos\psi - \theta \sin\psi$, the simplest model that can be obtained for the quadrotor is

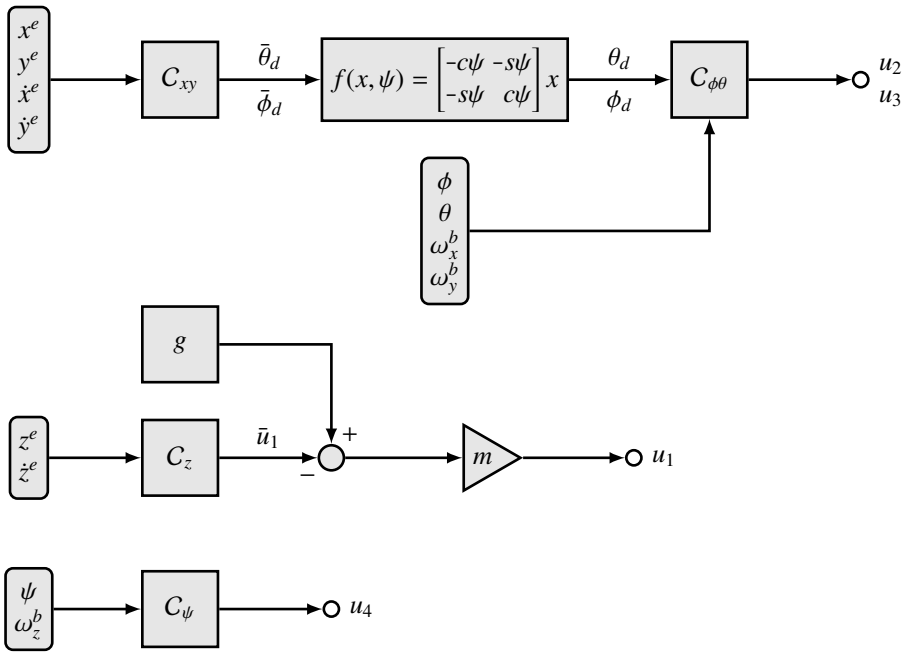


Figura 3.2: Structure of control in hover.

given by:

$$\begin{cases} \begin{bmatrix} \dot{v}_x^e \\ \dot{v}_y^e \end{bmatrix} = g \begin{bmatrix} \bar{\theta} \\ \bar{\phi} \end{bmatrix} \\ m\dot{v}_z^e = \bar{u}_1 \\ \boldsymbol{\mathcal{I}}\dot{\boldsymbol{\omega}}^{\mathbf{b}} = [u_2\ u_3\ u_4]^T \\ \dot{\boldsymbol{\eta}} = \boldsymbol{\omega}^{\mathbf{b}} \end{cases} . \tag{3.8}$$

Therefore, in hover conditions it is possible to consider a cascade control loop for position in the plane defined by $\vec{\mathbf{x}_e}$ and $\vec{\mathbf{y}_e}$, as well as individual loops for position in $\vec{\mathbf{z}_e}$ and heading. Figure 3.2 illustrates this structure.

3.3 Proportional Derivative Control

The Proportional Derivative (PD) controller represents one of the simplest and most effective control laws in the literature. This controller is composed of only two gains: proportional and derivative. The proportional gain portion acts on the current error and the derivative gain acts on a prediction of the future error.

It follows directly from the model illustrated in Figure 3.2 that the body dynamics can

be represented by a set of double integrators, whose transfer functions are:

$$\begin{cases} H_X(s) = g\bar{\Theta}/s^2 & & H_\Phi(s) = I_{xx}^{-1}U_2/s^2 \\ H_Y(s) = g\bar{\Phi}/s^2 & \text{and} & H_\Theta(s) = I_{yy}^{-1}U_3/s^2 \\ H_Z(s) = \bar{U}_1/s^2 & & H_\Psi(s) = I_{zz}^{-1}U_4/s^2 \end{cases}. \tag{3.9}$$

Since each of the loops consists of a double integrator, they can be individually controlled by PD controllers. To arrive at the control laws, consider by way of example the altitude loop and its corresponding error signal,

$$e_z = z_d^e - z^e. \tag{3.10}$$

Where the subscript d denotes the desired value of altitude. This equation can be differentiated twice to show the control signal,

$$\ddot{e}_z^e = \ddot{z}_d^e - \ddot{z}^e = \ddot{z}_d^e - \bar{u}_1. \tag{3.11}$$

Choosing the control $\bar{u}_1 = \ddot{z}_d^e + K_{\mathrm{p,z}}e_z + K_{\mathrm{d,z}}\dot{e}_z$, the error dynamics become asymptotically stable.

Therefore, defining the proportional gains $K_{\mathrm{p,i}} \in \mathbb{R} > 0$ and derivative gains $K_{\mathrm{d,i}} \in \mathbb{R} > 0$, the controllers will be, according to Figure 3.2:

$$\begin{cases} C_{xy}(t) = g^{-1}\begin{bmatrix}\ddot{x}_d^e \\ \ddot{y}_d^e\end{bmatrix} + K_{\mathrm{p,xy}}\begin{bmatrix}x_d^e(t) - x^e(t) \\ y_d^e(t) - y^e(t)\end{bmatrix} + K_{\mathrm{d,xy}}\begin{bmatrix}\dot{x}_d^e(t) - \dot{x}^e(t) \\ \dot{y}_d^e(t) - \dot{y}^e(t)\end{bmatrix} \\ C_z(t) = \ddot{z}_d^e + K_{\mathrm{p,z}}\left[z_d^e(t) - z^e(t)\right] + K_{\mathrm{d,z}}\left[\dot{z}_d^e(t) - \dot{z}^e(t)\right] \\ C_{\phi\theta}(t) = \begin{bmatrix}I_{xx}\ddot{\phi}_d \\ I_{yy}\ddot{\theta}_d\end{bmatrix} + K_{\mathrm{p},\phi\theta}\begin{bmatrix}\phi_d(t) - \phi(t) \\ \theta_d(t) - \theta(t)\end{bmatrix} + K_{\mathrm{d},\phi\theta}\begin{bmatrix}\dot{\phi}_d(t) - \omega_x^b(t) \\ \dot{\theta}_d(t) - \omega_y^b(t)\end{bmatrix} \\ C_\psi(t) = I_{zz}\ddot{\psi}_d + K_{\mathrm{p},\psi}\left[\psi_d(t) - \psi(t)\right] + K_{\mathrm{d},\psi}\left[\dot{\psi}_d - \omega_z^b(t)\right] \end{cases}. \tag{3.12}$$

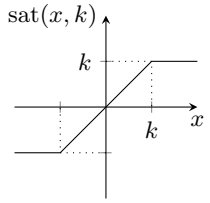
Notice that the implementation of these control laws requires prior knowledge of two derivatives of the references. With regard to the vehicle's heading, it is often possible to precompute its derivatives, as this system state is controlled directly. For example, suppose this reference is generated by a trajectory planning algorithm. In this case, it would be possible to obtain the derivatives directly from the trajectory equation. Conversely, the

pitch and *roll* angles generally have their references defined by the position control loop, see Figure 3.2. This makes it more difficult to determine the value of their derivatives.

To circumvent this problem and define a more autonomous controller, in the sense of reducing the interference of other factors upon it, two solutions will be presented. The first consists of using approximate derivatives of the references. These signals also pass through saturators, which establish the maximum and minimum values allowed for the speed and acceleration of the reference. The second is based on the reference model concept. In this solution the reference is filtered, allowing the values of the derivatives to be obtained. These two solutions will be illustrated for an arbitrary reference $r(t)$.

3.3.1 Approximate Derivative with Saturation

As previously stated, this solution makes use of approximate differentiators and saturators to obtain approximations of the input signals. In this context, a saturator is defined as the function sat : $\mathbb{R} \rightarrow \mathbb{R}$, such that



$$\text{sat(x, k)} = \begin{cases} k & \text{if} \quad x > k \\ x & \text{if} \quad -k \leq x \leq k \\ -k & \text{if} \quad x < -k \end{cases} \tag{3.13}$$

Including saturation in the calculation of the approximate derivative $\dot{\tilde{r}}(t)$ of a reference $r(t)$, its equation will be

$$\dot{\tilde{r}}(t) = \text{sat}\left(\mathcal{L}^{-1}\left\{\frac{s}{\tau_d s + 1}\, r(s)\right\},\ \dot{r}_{\max}\right). \tag{3.14}$$

Where $\dot{r}_{\max} \in \mathbb{R} > 0$ is the maximum allowed speed, $\tau_d \in \mathbb{R} > 0$ the time constant of the approximate differentiator and $\mathcal{L}$ the operator denoting the Laplace Transform.

3.3.2 Reference Model

The reference model concept is based on filtering the input signal. Using an n-th order filter it is possible to obtain up to the n-th derivative of the filtered signal. Note that the derivative is of the filtered signal. Therefore, one must choose the filter parameters so that

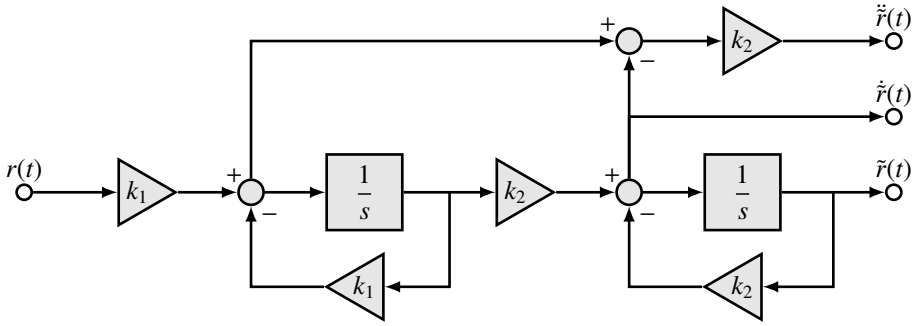


Figura 3.3: Block diagram of a realization for (3.16).

its response has enough bandwidth to reproduce the input signal.

The construction of an n-th order filter to be used as a reference model basically comes down to composing n first-order filters in series. Each of the filters has a transfer function

$$H_i(s) = \frac{k_i}{s + k_i}, \tag{3.15}$$

where $k_i \in \mathbb{R} > 0$ determines the bandwidth of the i-th filter. More precisely $-k_i$ is the pole location related to this filter. Since a second-order filter is desired, the transfer function is

$$H(s) = H_1(s)H_2(s) = \left(\frac{k_1}{s + k_1}\right)\left(\frac{k_2}{s + k_2}\right) = \frac{k_1 k_2}{s^2 + s(k_1 + k_2) + k_1 k_2} \tag{3.16}$$

The realization with integrators and the measurement of the signals $\tilde{r}(t)$, $\dot{\tilde{r}}(t)$ and $\ddot{\tilde{r}}(t)$ can be seen in the block diagram of Figure 3.3.

Another way to obtain the desired signals is through a state-space representation. Thus, consider a realization of the second-order filter transfer function $H(s)$ in the observable canonical form,

$$\begin{cases} \dot{x}(t) = \begin{bmatrix} -(k_1 + k_2) & 1 \\ -k_1 k_2 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ k_1 k_2 \end{bmatrix} r(t) \\ \tilde{r}(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} x(t) \end{cases}. \tag{3.17}$$

Through a rearrangement of the variables, it follows directly from this equation that the

values of the approximate derivatives can be written as:

$$\begin{cases} \tilde{r}(t) = x_1(t) \\ \dot{\tilde{r}}(t) = -(k_1 + k_2)x_1(t) + x_2(t) \\ \ddot{\tilde{r}}(t) = -(k_1 + k_2)\dot{\tilde{r}}(t) + k_1 k_2 \left[r(t) - x_1(t)\right] \end{cases} . \tag{3.18}$$

For all strategies that follow, it was chosen to use the reference model method. In addition to constituting a more elegant solution, this method is more stable and less sensitive to high-frequency noise, inherent to measurements made in the real world.

3.4 Proportional Integral Derivative Control

The Proportional Integral Derivative (PID) controller represents a more general situation of the PD controller. The integral control portion, as the name suggests, acts on the integral of the error. This action combats steady-state errors, disturbances and parametric uncertainties. In contrast, the addition of the integral action adds a pole to the system, contributing to oscillations occurring.

In general, the integral portion is only added if necessary, and it is no different for quadrotors. In this work there is interest in analyzing the effect of parametric uncertainties on the closed-loop dynamics of the vehicle. In particular, variations caused by uncertainty in the mass of the vehicle.

When there is uncertainty in the quadrotor mass, the *feedforward* control portion is not able to nullify the effect of gravity, leading to steady-state error. This effect can be treated as an input disturbance to the system. Using model (3.8) and the PD controller (3.12) it is possible to determine the steady-state error relating to this disturbance. Thus, using the Final Value Theorem (FVT),

$$e_z(\infty) = \lim_{s\to 0} s\mathcal{L}\left\{e_z(t)\right\} = s\,\frac{\bar{m}g}{s\left(s^2/m - k_d s - k_p\right)} = -\frac{\bar{m}g}{k_p}, \tag{3.19}$$

where $\bar{m} = m - \tilde{m}$ is the difference between the actual mass and the one used in the *feedforward* portion of the altitude control. Then adding an integral control portion, it is

verified again by the FVT that the disturbance is completely nullified in steady state,

$$e_z(\infty) = \lim_{s\to 0} s\mathcal{L}\{e_z(t)\} = s\,\frac{\bar{m}g}{s\left(s^2/m - k_d s - k_p - k_i/s\right)} = 0. \tag{3.20}$$

Therefore, the addition of the integral action in the control becomes interesting, especially in the altitude loop. Later on the effect of the integral action on the input disturbance generated by the mass uncertainty will be addressed in detail.

3.5 PD Control with Variable Gain for Altitude

The variable gain strategy is characterized by changing one or more gains of the controller according to the error. This variation is given by an arbitrary function, making the controller nonlinear. Intuitively, it is expected that: if gains that displace, at a given instant, the poles to the right half-plane are not generated, the system will be stable. To assert the stability of the system a formal proof is necessary, and it is not necessarily required to restrict the poles to the left half-plane, but the above interpretation facilitates understanding.

It was shown that the altitude of the quadrotor can be commanded through a simple PD controller. This strategy is composed of two gains: proportional and derivative, $K_{\mathrm{p,z}}$ and $K_{\mathrm{d,z}}$ respectively. Rewriting equation (3.12) and defining

$$e_z = z_d^e - z^e \;\;\therefore\;\; \dot{e}_z = \dot{z}_d^e - \dot{z}^e \tag{3.21}$$

it is possible to factor out one of the gains and define the following structure:

$$C_z(t) = K_{\mathrm{p,z}}\left(e_z + K_{\mathrm{d,z}}\dot{e}_z\right) \tag{3.22}$$

Thus, the gain $K_{\mathrm{d,z}}$ will be fixed and a function $f_k : \mathbb{R} \mapsto [k_1, \infty)$, where $k_1 \in \mathbb{R} > 0$, always positive will be defined for the other. This approach is consistent with the idea of ensuring that the poles are restricted to the left half-plane.

This function maps the altitude error to the proportional gain, increasing it as the error

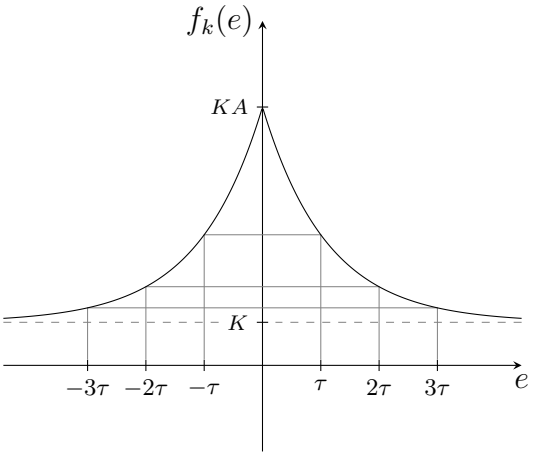


Figura 3.4: Graph of the gain function 3.23.

decreases. The choice was made for the function

$$f_k(e) = K\left[A\exp\left(-\tau|e|\right) + 1\right], \tag{3.23}$$

where $K \in \mathbb{R} > 0$ and $A \in \mathbb{R} > 0$ determine the gain amplitude and $\tau \in \mathbb{R} > 0$ the speed of gain variation, see Figure 3.4. Note that in this case $f_k : \mathbb{R} \mapsto [k_1, k_2]$, but the stability proof will be conducted for the more general case, with $k_2 \to \infty$.

The proof of global stability for this strategy can be done using the circle criterion [13]. To this end, two definitions and a corollary will be presented to be used in the deduction.

Definition 3.5.1. *A rational function $G(s)$, with $s \in \mathbb{C}$ is said to be positive real (PR) if and only if the following properties are satisfied:*

1. *all poles of $G(s)$ have* $\mathrm{Re}\{s\} < 0$,
2. $\mathrm{Re}\{G(j\omega)\} \geq 0,\ \forall\ \omega \in [0, \infty)$.

Definition 3.5.2. *Given that $G(s)$ is not identically zero for all s, then $G(s)$ is said to be strictly positive real (SPR) if and only if $\exists\epsilon$ such that $G(s-\epsilon)$ is PR.*

Corollary 3.5.3. *If a system $G(s)$ is fed back by a function $\psi(.) \in [k_1, \infty)$, then it follows directly from the circle criterion that it will be globally exponentially stable if and only if $G(s)[1 + k_1 G(s)]^{-1}$ is SPR.*

Proof. Note that the error dynamics can be written according to the block diagram of Figure 3.5. Since the plant in question is a double integrator, i.e. $G(s) = 1/s^2$, we have

$$\bar{G}(s) = \frac{1 + k_d s}{s^2} \ \text{ and } \ \bar{G}(s)[1 + k_1\bar{G}(s)]^{-1} = \frac{k_d s + 1}{s^2 + k_1 k_d s + k_1}. \tag{3.24}$$

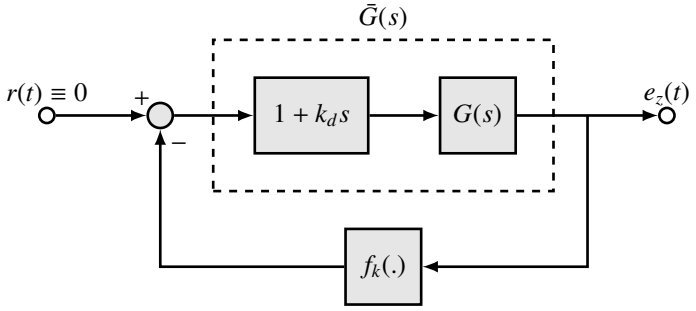


Figura 3.5: Block diagram of the PD control with variable gain.

Through Definition 3.5.2 it can be easily verified that $\bar{G}(s)[1 + k_1\bar{G}(s)]^{-1}$ is SPR. Therefore, by Corollary 3.5.3 it is concluded that the origin of the system will be globally exponentially stable. □

To illustrate the stability of the system, consider the phase portrait of Figure 3.6 obtained for $f_k(e) = 40\,[2.2\exp(-2|e|) + 1]$ and $k_d = 0.23$. Through this diagram it is possible to observe several trajectories converging to the origin from different initial conditions.

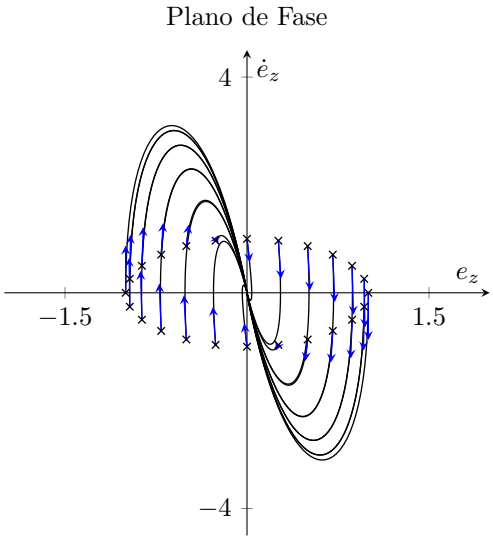


Figura 3.6: Phase portrait of the double integrator controlled by the PD controller with variable gain.

Similar to PID control, this strategy aims to reduce the steady-state error resulting from input disturbances. Despite not completely nullifying the error, variable gain control also reduces the settling time, providing an advantage over the other. This issue will be raised again in Section 3.8.2, where the PD, PID and PD with variable gain strategies will be compared in view of uncertainties in the measurement of the vehicle mass.

3.6 Lyapunov Control for Attitude

This control strategy makes direct use of Lyapunov's theory to extract a control law for the vehicle's orientation. The development that will be shown is based on equation (2.51) and ignores only the effect of the torque induced by the motors around $\vec{\mathbf{z}_\mathbf{b}}$.

Let $\mathbf{x}_{\boldsymbol{\alpha}} \in \mathbb{R}^6$ be the state vector that contains only the terms referring to the body's orientation. This vector can be defined as

$$\mathbf{x}_{\boldsymbol{\alpha}} = \begin{bmatrix} \phi & \theta & \psi & \omega_x^b & \omega_y^b & \omega_z^b \end{bmatrix}^{\mathrm{T}}. \tag{3.25}$$

Therefore, its derivative $\mathbf{f}_{\boldsymbol{\alpha}}(\mathbf{x}_{\boldsymbol{\alpha}}, \mathbf{u}) = \dot{\mathbf{x}}_{\boldsymbol{\alpha}}$ will be

$$\mathbf{f}_{\boldsymbol{\alpha}}(\mathbf{x}_{\boldsymbol{\alpha}}, \mathbf{u}) = \begin{bmatrix} [x_4 \ x_5 \ x_6]^{\mathrm{T}} \\ (I_{yy} - I_{zz})x_5x_6/I_{xx} - I_r x_5 \Omega_R / I_{xx} + u_2/I_{xx} \\ (I_{zz} - I_{xx})x_6x_4/I_{yy} + I_r x_4 \Omega_R / I_{yy} + u_3/I_{yy} \\ (I_{xx} - I_{yy})x_4x_5/I_{zz} + u_4/I_{zz} \end{bmatrix}. \tag{3.26}$$

Assuming that it is desired to stabilize the angles in a certain configuration $\mathbf{x}_{\boldsymbol{\alpha}\mathbf{d}} = \begin{bmatrix} \phi_d \ \theta_d \ \psi_d \ 0 \ 0 \ 0 \end{bmatrix}^{\mathrm{T}}$, the potential function is defined

$$V(\mathbf{x}_{\boldsymbol{\alpha}}) = \frac{1}{2}\left[\sum_{i=1}^{3}(x_{id} - x_i)^2 + \sum_{i=4}^{6} x_i^2\right]. \tag{3.27}$$

Using the hypothesis that the body is symmetric, $I_{xx} = I_{yy}$, the time derivative of the potential function $\dot{V}(\mathbf{x}_{\boldsymbol{\alpha}}) = (\nabla V)^{\mathrm{T}} \mathbf{f}_{\boldsymbol{\alpha}}$ will be

$$\dot{V}(\mathbf{x}_{\boldsymbol{\alpha}}) = (x_1 - x_{1d})x_4 + (x_2 - x_{2d})x_5 + (x_3 - x_{3d})x_6 + \frac{x_4 u_2}{I_{xx}} + \frac{x_5 u_3}{I_{yy}} + \frac{x_6 u_4}{I_{zz}}. \tag{3.28}$$

Therefore, inputs are chosen

$$\begin{cases} u_2 &= I_{xx}(\phi_d - \phi) - k_{\mathrm{d}1}\omega_x^b \\ u_3 &= I_{yy}(\theta_d - \theta) - k_{\mathrm{d}2}\omega_y^b \\ u_4 &= I_{zz}(\psi_d - \psi) - k_{\mathrm{d}3}\omega_z^b \end{cases}, \tag{3.29}$$

so that the function $\dot{V}(\mathbf{x}_\alpha)$ becomes negative semi-definite for gains $k_{\mathrm{di}} \in \mathbb{R} > 0$.

$$\dot{V}(\mathbf{x}_\alpha) = -\left(\frac{k_{\mathrm{d1}} x_4^2}{I_{xx}} + \frac{k_{\mathrm{d2}} x_5^2}{I_{yy}} + \frac{k_{\mathrm{d3}} x_6^2}{I_{zz}}\right) \tag{3.30}$$

Therefore, the chosen function is indeed a Lyapunov Function. It is also possible to show that this control choice produces global stability through La Salle's theorem [2].

Note that the control found is also PD. The only difference between this and the one defined in (3.12) is the presence of an *offset* in the proportional gains. This control is only shown to corroborate the validity of the PD control, even in regions that exceed the previously established approximations.

3.7 *Integral Backstepping* Control for Attitude

The strategies previously shown are relatively simple and produce equally simple control laws. Both aim to stabilize the vehicle in desired positions and orientations. This is the first strategy that aims to track variations in desired posture.

Integral Backstepping control combines the characteristics of a proportional integral derivative (PID) control with those of a *Backstepping* control. The portion referring to PID introduces an integral portion into the control, aiming to correct steady-state errors and make the control more robust to disturbances. *Backstepping*, on the other hand, consists of recursively using one state as the input to the other. The stability proof of this control is done by Lyapunov theory [2].

First, the hypothesis that $\dot{\boldsymbol{\eta}} = \boldsymbol{\omega}^{\mathbf{b}}$ will be made. The procedures to generate the control laws are very similar, therefore only the step-by-step to determine the *roll* angle control will be exposed. First, the error is defined

$$e_1 = \phi_d - \phi \tag{3.31}$$

and its dynamics is calculated:

$$\dot{e}_1 = \dot{\phi}_d - \omega_x^b. \tag{3.32}$$

Assuming that the angular velocity component ω_x^b can be used as control, its desired value

is given by

$$\omega_{xd}^{b} = \dot{\phi}_d + k_1 e_1 + \lambda_1 \bar{e}_1 \tag{3.33}$$

to stabilize the dynamics of e_1, where $k_1 \in \mathbb{R} > 0$ and $\lambda_1 \in \mathbb{R} \geq 0$ are gains and $\bar{e}_1 = \int_{t_0}^{t} e_1(\tau) d\tau$. However, there is still the angular velocity error

$$e_2 = \omega_{xd}^{b} - \omega_{x}^{b} \tag{3.34}$$

with dynamics

$$\dot{e}_2 = \dot{\omega}_{xd}^{b} - \dot{\omega}_{x}^{b} = \ddot{\phi}_d + k_1 \dot{e}_1 + \lambda_1 e_1 - \dot{\omega}_{x}^{b}. \tag{3.35}$$

Substituting (3.34) into (3.32),

$$\dot{e}_1 = e_2 - k_1 e_1 - \lambda_1 \bar{e}_1. \tag{3.36}$$

Now applying equations (3.36) and the dynamics of ω_x^b, equation (3.3), the final form of the dynamics of e_2 is reached, which includes the control input u_2.

$$\dot{e}_2 = \ddot{\phi}_d + k_1 \left(e_2 - k_1 e_1 - \lambda_1 \bar{e}_1\right) + \lambda_1 e_1 - \omega_y^b \omega_z^b \frac{(I_{yy} - I_{zz})}{I_{xx}} - \frac{u_2}{I_{xx}} \,. \tag{3.37}$$

The desired dynamics for the angular velocity error e_2 is

$$\dot{e}_2 = -k_2 e_2 - e_1, \tag{3.38}$$

where $k_2 \in \mathbb{R} > 0$ is a constant that defines the speed at which this error goes to zero. In order for (3.37) to become like (3.38), it is necessary to define the control

$$u_2 = I_{xx} \left[e_1 (1 - k_1^2 + \lambda_1) + e_2 (k_2 + k_1) - k_1 \lambda_1 \bar{e}_1 + \ddot{\phi}_d \right] - \omega_y^b \omega_z^b (I_{yy} - I_{zz}) \tag{3.39}$$

The same procedure is adopted to determine the other control inputs, defining the error in the measure of a state and its variation. Doing this, the remaining control laws

that define the *Integral Backstepping* algorithm are given by

$$\begin{cases} u_3 = I_{yy}\left[e_3(1-k_3^2+\lambda_2)+e_4(k_4+k_3)-k_3\lambda_2\bar{e}_1+\ddot{\theta}_d\right]-\omega_z^b\omega_x^b(I_{zz}-I_{xx}) \\ u_4 = I_{zz}\left[e_5(1-k_5^2+\lambda_3)+e_6(k_6+k_5)-k_5\lambda_3\bar{e}_1+\ddot{\psi}_d\right]-\omega_x^b\omega_y^b(I_{xx}-I_{yy}) \end{cases} \tag{3.40}$$

Where $e_3 = \theta_d - \theta$, $e_4 = \omega_{yd}^b - \omega_y$, $e_5 = \psi_d - \psi$ and $e_6 = \omega_{zd}^b - \omega_z$. Note that this control is identical to PID, minus a term referring to gyroscopic effects. This new term can be interpreted as a feedback linearization portion.

3.8 Simulation Results

In this chapter, a few different strategies for the control of a quadrotor were presented. These strategies can be divided into two groups:

Position Control

1. PID Control with Altitude *Feedforward*
2. PD Control with Variable Gain and Altitude *Feedforward*

Attitude Control

1. PD Control
2. Lyapunov Control
3. *Integral Backstepping* Control

In this section, the results obtained by simulation for each of these control laws will be shown. The objective of these simulations is to evaluate the stability and performance of each control law in closed loop. As will be broadly discussed in Appendix C, the simulation platform was entirely developed in Matlab® and Simulink®.

All simulations were performed using the parameters of the Parrot®Ar.Drone quadrotor. These data are found in Table 3.1.

The thrust κ_T and drag κ_Q constants were estimated and this process is described in Appendix B.

For the simulation to be more faithful to what is found in practice, all tests were run considering the presence of white noise in the system measurements. To size these noise components, flight information from the real system was obtained. The procedure to carry out the sizing is described in Appendix A and the results arranged in Table 3.2.

Tabela 3.1: Parameters of the Parrot® Ar.Drone quadrotor.

Description	Parameter	Value	Unit
Total mass of the vehicle	m	0.890	kg
Inertia about $\vec{\mathbf{x}_\mathbf{b}}$	I_{xx}	9.57	10^{-3}kg.m^2
Inertia about $\vec{\mathbf{y}_\mathbf{b}}$	I_{yy}	18.57	10^{-3}kg.m^2
Inertia about $\vec{\mathbf{z}_\mathbf{b}}$	I_{zz}	25.55	10^{-3}kg.m^2
Arm length	l	0.152	m
Propeller radius	R	0.146	m
Thrust constant	κ_T	8.7	10^{-5}N.s^2
Drag constant	κ_Q	1.4	10^{-5}N.m.s^2

Tabela 3.2: Estimate of white noise parameters observed in the system.

Measurement	Mean	Standard Deviation	Unit
Position	0	0.04	mm
Roll	0	0.01	deg
Pitch	0	0.01	deg
Yaw	0	0.01	deg
Linear Velocity	0	2.00	mm/s
Angular Velocity	0	0.60	deg/s

In addition to the presence of noise, constraints on the maximum and minimum rotation speeds of the propellers were imposed. These limits are imposed through the input u_1, which represents the force exerted along $\vec{\mathbf{z}_\mathbf{b}}$. In this way, a minimum force of 0 N and a maximum force capable of generating 2g of acceleration was considered.

3.8.1 Attitude Control

Here the results obtained by the vehicle's attitude control loops will be presented. For each strategy the control performance will be tested using the following references: step, ramp and sinusoidal. For the sinusoidal reference the frequency response[2] of the system will be calculated for a frequency range $\omega \in [0.1\ 200]$ rad/s, with an amplitude of 10 deg. As for the step and ramp references, amplitudes of 30 deg and 10 deg/s will be used for all angles. These values are found in Table 3.3.

Due to the vehicle's symmetry, the *roll* and *pitch* responses are identical. Therefore, only the simulation results regarding the *pitch* and *yaw* angles will be shown, with the response of the *roll* angle being implicit. When this does not occur, adjustments are made

[2]This frequency response is obtained approximately, applying sinusoidal inputs and measuring the output value after a predetermined number of periods.

Tabela 3.3: Parameters of the attitude references.

Signal	$A\sin(\omega t)$		At	$A\mathbb{1}(t)$
	A (deg)	ω (rad/s)	A (deg/s)	A (deg)
Roll	10	[0.1 200]	10	30
Pitch	10	[0.1 200]	10	30
Yaw	10	[0.1 200]	10	30

to the gains so that the two responses match.

PD Controller

The PD controller was the first to be developed in this text. This is the simplest control strategy to be implemented on the vehicle capable of enabling hover flight. Several performance tests of this controller were carried out, as described above, Table 3.3. These tests were done on the simulation platform developed in Matlab® and Simulink®, which is described in Appendix C.

After several simulations, the controller parameters were tuned, Table 3.4. With these values for the gains, the step responses, Figures 3.7 and 3.8, ramp responses, Figures 3.9 and 3.10 and the frequency response, Figure 3.11 were obtained.

Tabela 3.4: Gains used in the PD controller.

Loop	K_p	K_d
Pitch	30	3
Yaw	0.13	0.13

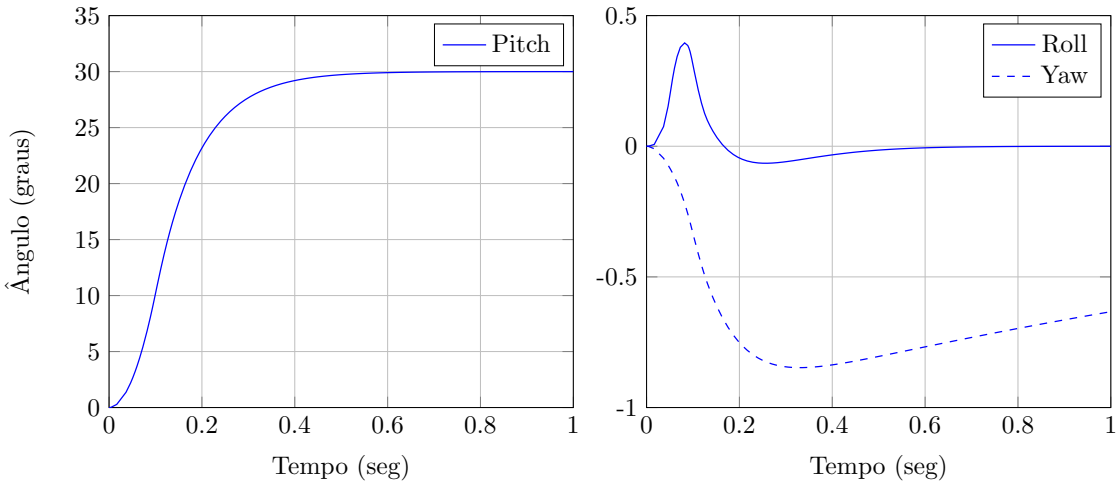


Figura 3.7: *Pitch* step response of the Proportional Derivative controller.

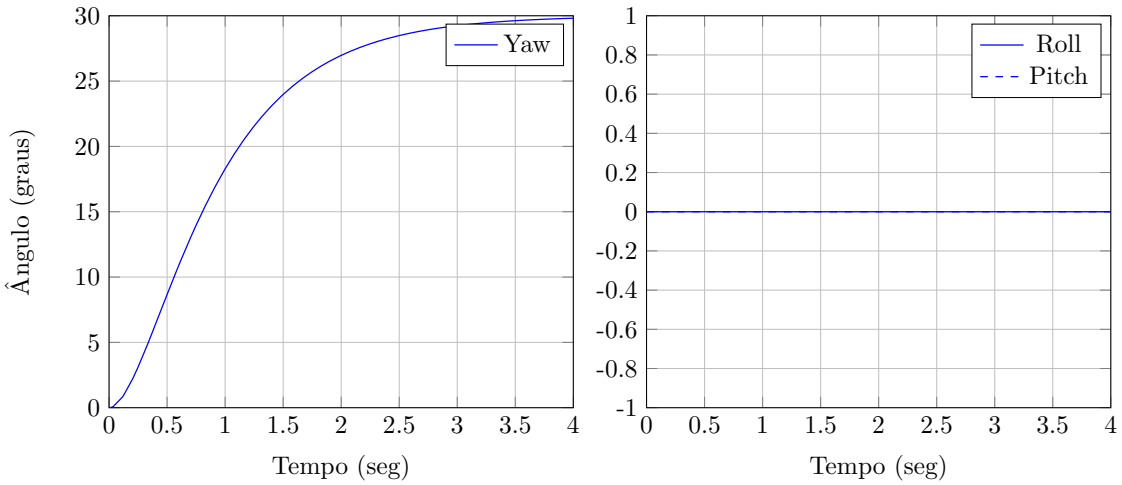


Figura 3.8: *Yaw* step response of the Proportional Derivative controller.

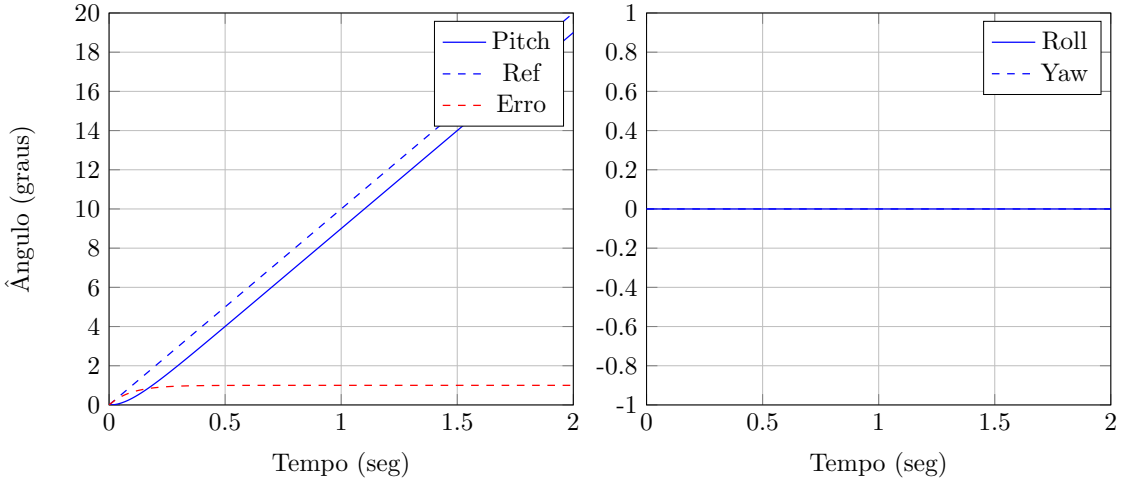


Figura 3.9: *Pitch* ramp response of the Proportional Derivative controller.

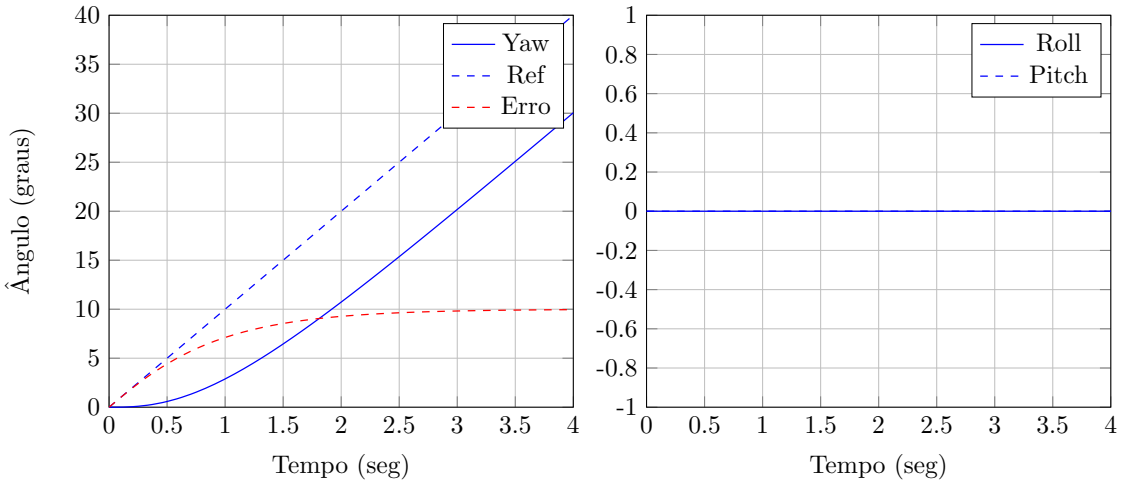


Figura 3.10: *Yaw* ramp response of the Proportional Derivative controller.

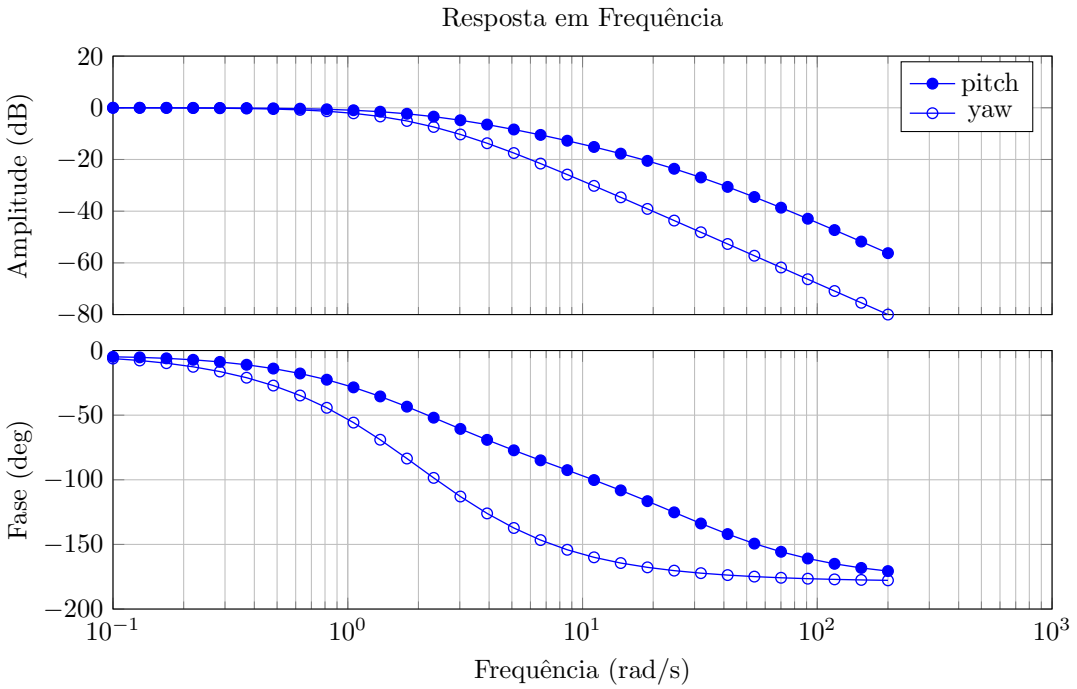


Figura 3.11: Frequency response of the attitude Proportional Derivative controller.

The choice of gains was made aiming to obtain a response without overshoot and with a low settling time. This idea was applied in both loops. As the *yaw* loop is naturally slower, its settling time specification was greater, 3 seconds, against 0.5 seconds in the other loops.

It is observed that the gains of the *yaw* control loop are significantly lower than those of the *pitch* loop. This is due to the fact that it is difficult to meet fast specifications in the former. This difficulty stems from the relationship between the thrust κ_T and drag κ_Q constants, since the first is usually at least an order of magnitude lower than the second. Thus, from equation 3.1, we see that it is easier to generate torques around the *roll* and

pitch axes than around the *yaw* axis.

Lyapunov Controller

It was demonstrated that the controller obtained through the direct application of Lyapunov's stability theory takes the form of a PD. However, the choice of proportional gains is restricted to the vehicle's moment of inertia.

Therefore, making only the adjustment of the derivative gain parameters, their values are given by Table 3.5.

Tabela 3.5: Gains used in the attitude Lyapunov controller.

Loop	K_p	K_d
Pitch	I_{yy}	$-I_{yy}/2$
Yaw	I_{zz}	$-I_{zz}$

Due to the low gains found with this controller, the system response becomes very slow, with a bandwidth smaller than that of the PD controller. Therefore, the responses found for the gains of Table 3.5 will be omitted.

Integral Backstepping Controller

Finally, the *Integral Backstepping* controller was the last to be tested. The choice of gains used in the *Integral Backstepping* controller was made iteratively, seeking to meet the same specifications for rise time, settling time and overshoot, imposed on the PD controller. Thus, the gains shown in Table 3.6 were found.

Tabela 3.6: Gains used in the attitude *Integral Backstepping* controller.

Loop	k_1	k_2	λ
Pitch	20	20	1
Yaw	5.5	4	0.1

Again it is possible to observe the difference between the gains used in the *pitch* loop and in the *yaw* loop. Analyzing the responses obtained for the step responses, Figures 3.12 and 3.13, ramp responses, Figures 3.14 and 3.15 and the frequency response, Figure 3.16, the better performance of the *Integral Backstepping* controller when compared to the PD becomes evident. One observes a lower steady-state error, lower error in the angles that are being kept at zero, and greater bandwidth (approximately one decade).

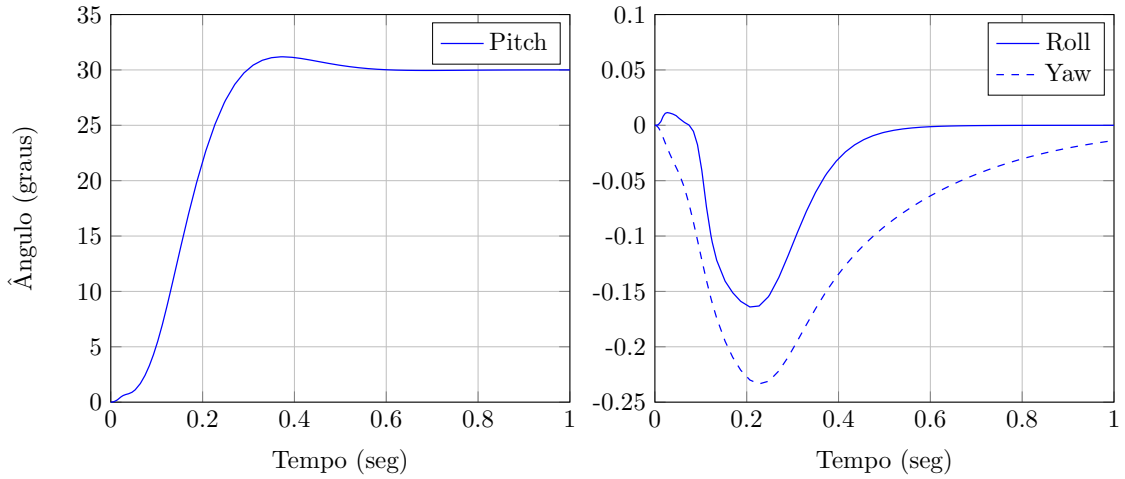


Figura 3.12: *Pitch* step response of the *Integral Backstepping* controller.

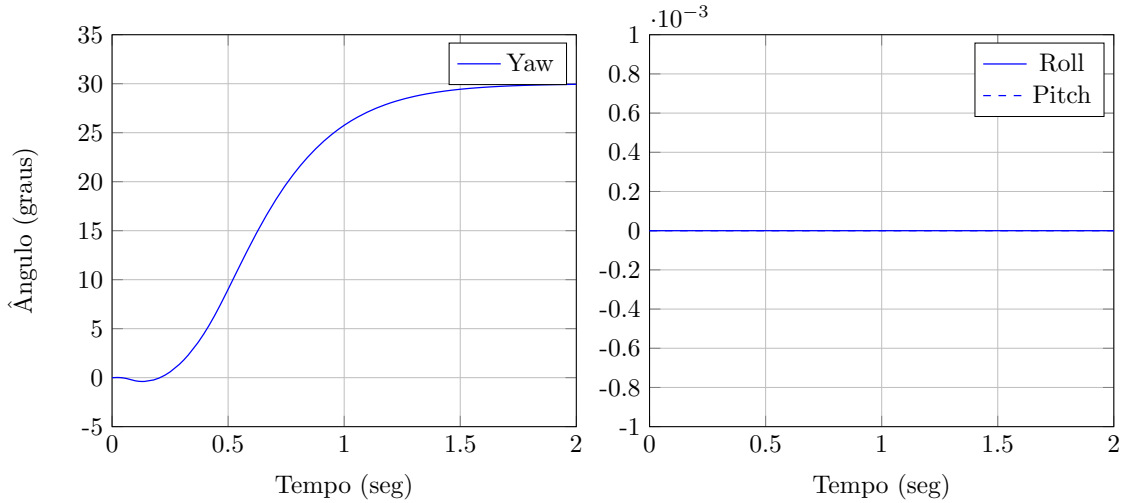


Figura 3.13: *Yaw* step response of the *Integral Backstepping* controller.

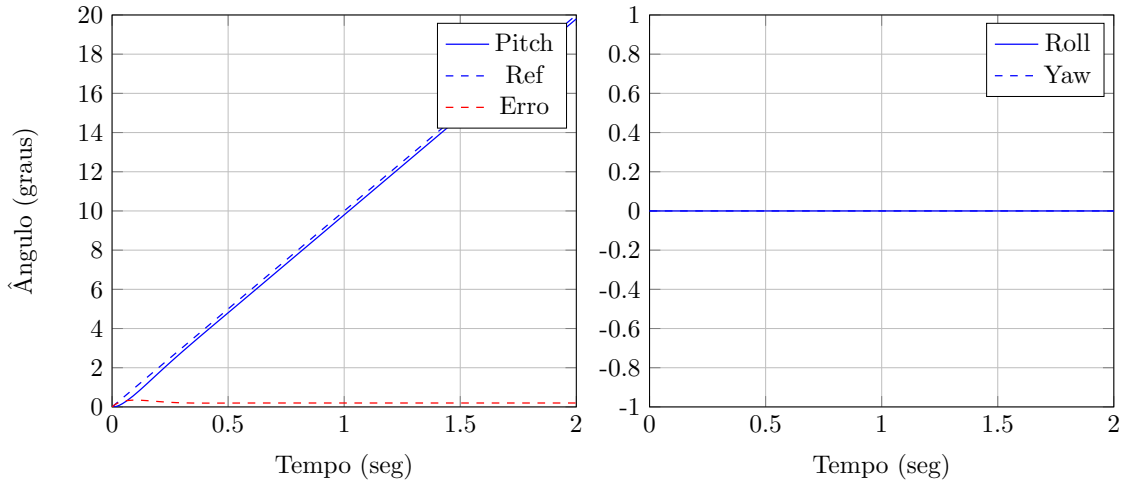


Figura 3.14: *Pitch* ramp response of the *Integral Backstepping* controller.

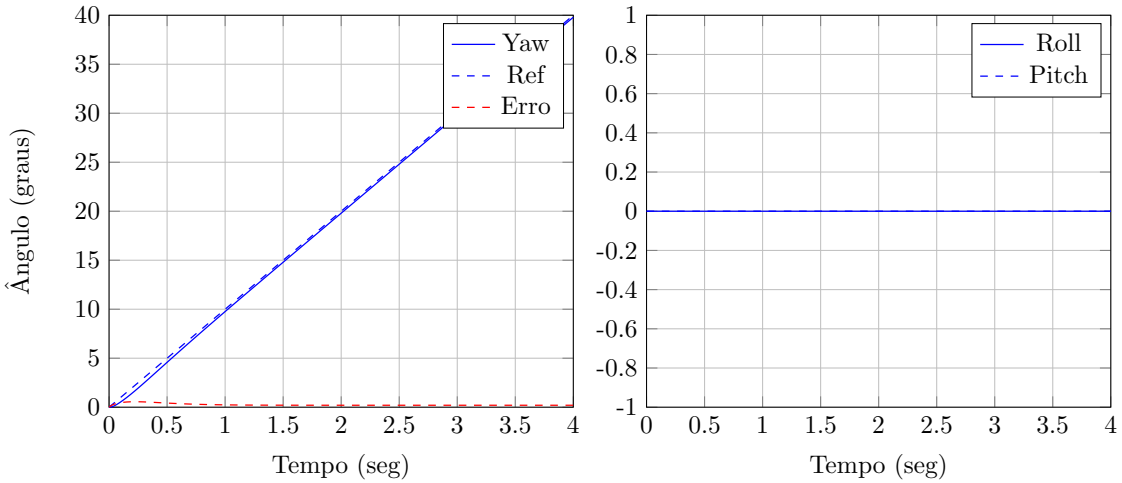


Figura 3.15: *Yaw* ramp response of the *Integral Backstepping* controller.

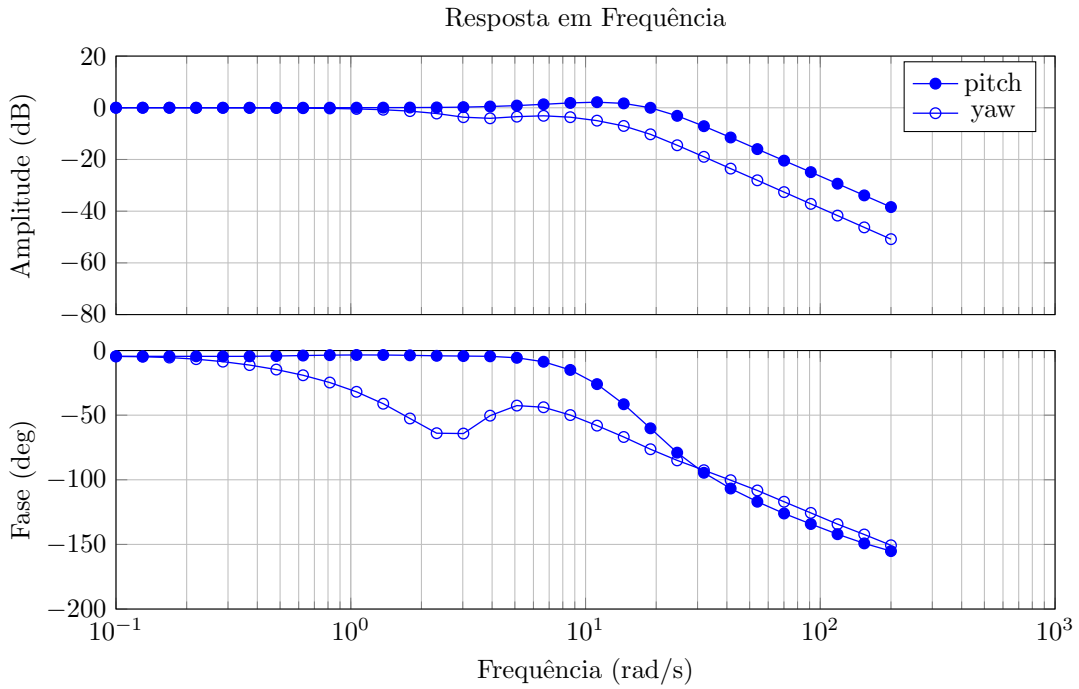


Figura 3.16: Frequency response of the attitude *Integral Backstepping* controller.

Conclusion

With the results obtained by the simulation, it is possible to make a decision regarding the most appropriate strategy to command the vehicle's orientation. Obviously, the choice depends directly on the situation and the operation desired to execute. Even presenting a lower performance than the *Integral Backstepping* controller, the PD controller has a simpler structure, easy implementation and low computational cost. Therefore, it is more suitable for situations close to hovering flight, where the trajectories are slow.

In contrast, the *Integral Backstepping* controller is more complex, depends on the im-

plementation of filters to use the reference model technique and requires a better knowledge of the vehicle's parameters to enable feedback linearization. This high complexity allows this controller to present a performance superior to the PD, making it more indicated for trajectory tracking situations, or simply when a faster vehicle response is desired.

3.8.2 Position Control

Here the results obtained through the implementation of PID control and PD control with variable gain will be presented. The tests that were carried out to evaluate the closed-loop response of the quadrotor follow the same line as those done for orientation. Thus, step, ramp and sinusoidal references for the position were stipulated, according to Table 3.7.

Tabela 3.7: Parameters of the position references.

Signal	$A\sin(\omega t)$		At	$A\mathbb{1}(t)$
	A (m)	ω (rad/s)	A (m/s)	A (m)
Position-x	1	[0.1 200]	2	1
Position-y	1	[0.1 200]	2	1
Position-z	1	[0.1 200]	2	2

Finally, the test for the altitude step will be redone, now considering a 50% uncertainty in the vehicle mass value.

The performance of any position control strategy depends directly on how the orientation is being commanded. In the previous section it was concluded that the *Integral Backstepping* controller is the one that presents the best performance among those tested. Thus, this was the controller elected to command the orientation loop.

Similar to what was done for the vehicle's attitude, only the tests relating to the vehicle's position in $\vec{\mathbf{x}_e}$ and $\vec{\mathbf{z}_e}$ will be shown, leaving implicit the response along $\vec{\mathbf{y}_e}$.

PID Controller with Altitude *Feedforward*

The controller was tuned to obtain responses with overshoots of less than 20% and 5% for position and altitude, and settling times of 4 seconds for planar position and 1 second for altitude. This tuning led to the gains in Table 3.8.

Note that the integral portion of the control was not considered. The integral component is only interesting to combat uncertainties in the vehicle's mass and, as will be seen in Section 3.8.2, its action is very slow, not justifying its permanence in the control.

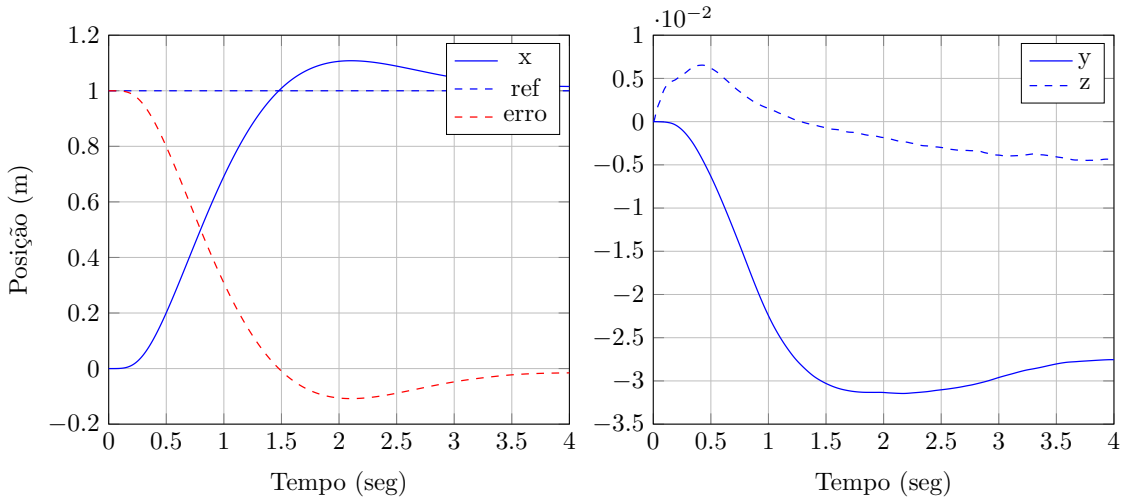


Figura 3.17: Step response in $\mathbf{x_e}$ of the PID controller.

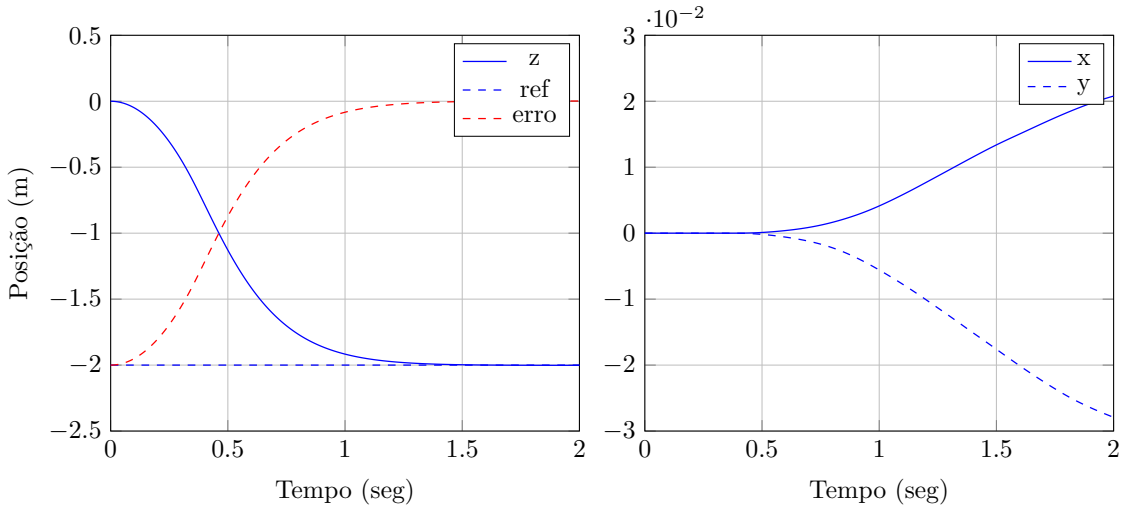


Figura 3.18: Altitude step response of the PID controller.

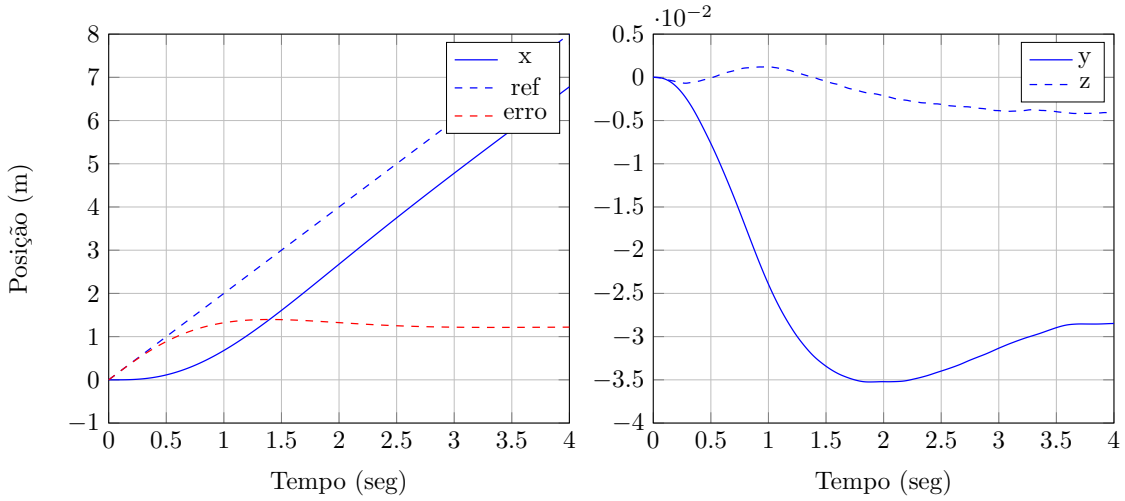


Figura 3.19: Ramp response in $\mathbf{x_e}$ of the PID controller.

Tabela 3.8: Gains used in the position PID controllers.

Loop	K_p	K_d	K_i
Planar Position	0.4	0.25	0
Altitude	-30	10	0

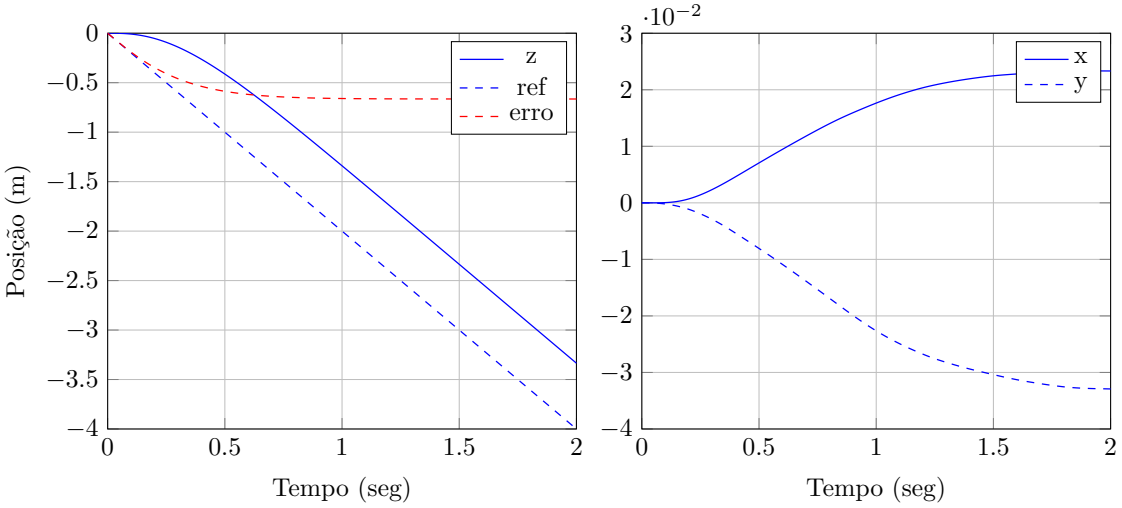


Figura 3.20: Altitude ramp response of the PID controller.

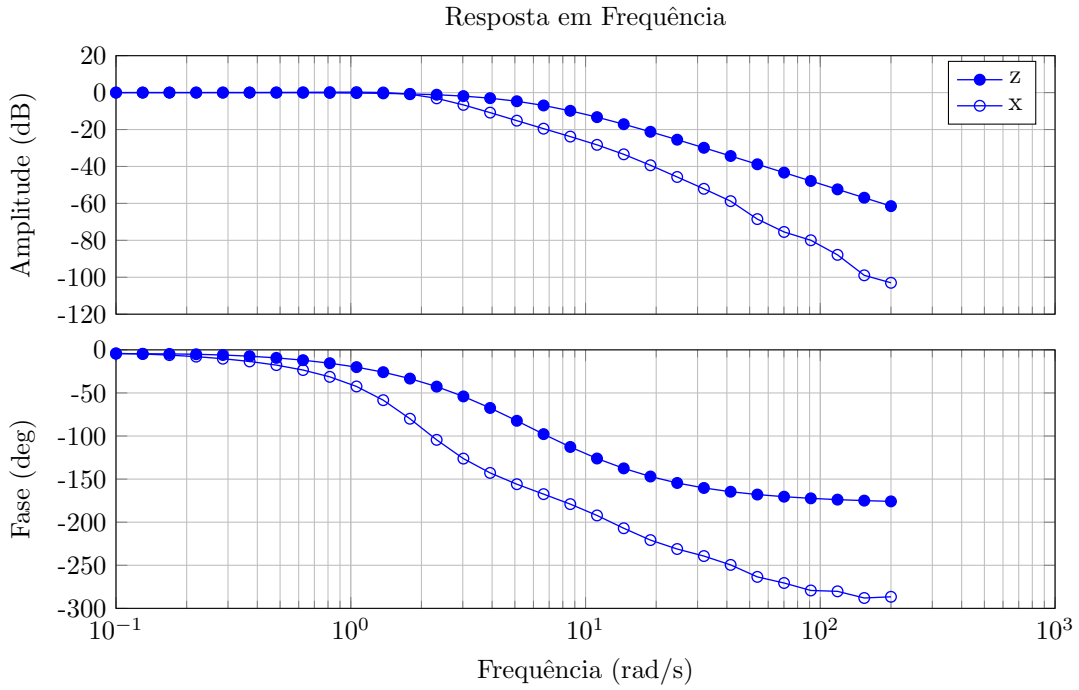


Figura 3.21: Frequency response of the position PID controller.

The magnitude difference between the position and altitude gains is mainly due to the units of each loop. The loop that controls the quadrotor's position in the plane generates angle references for the *roll* and *pitch* loops. On the other hand, the altitude control loop produces forces in $\vec{\mathbf{z}}_\mathbf{b}$, referring to the control signal u_1.

The simulations executed with this tuning generated the step responses, Figures 3.17 and 3.18, ramp responses, Figures 3.19 and 3.20 and the frequency response, Figure 3.21.

PD Controller with Variable Gain and Altitude *Feedforward*

This controller aims to improve the response of the classic Proportional Derivative. The basic idea is to increase the gain as the error approaches zero, reducing steady-state error and increasing bandwidth. Thus, the chosen parameters implement equation (3.23) as follows:

$$f_k(e_z) = -30\left[2.2\exp\left(-2|e_z|\right) + 1\right]. \tag{3.41}$$

The idea of this strategy is to improve the performance of the altitude loop when compared to the classic PD controller. Therefore, only the responses relating to the vehicle's altitude will be shown.

A better performance of this strategy can be noted when compared to the classic PID. Analyzing Figures 3.22 and 3.23 it is possible to notice a lower steady-state error and lower rise and settling times.

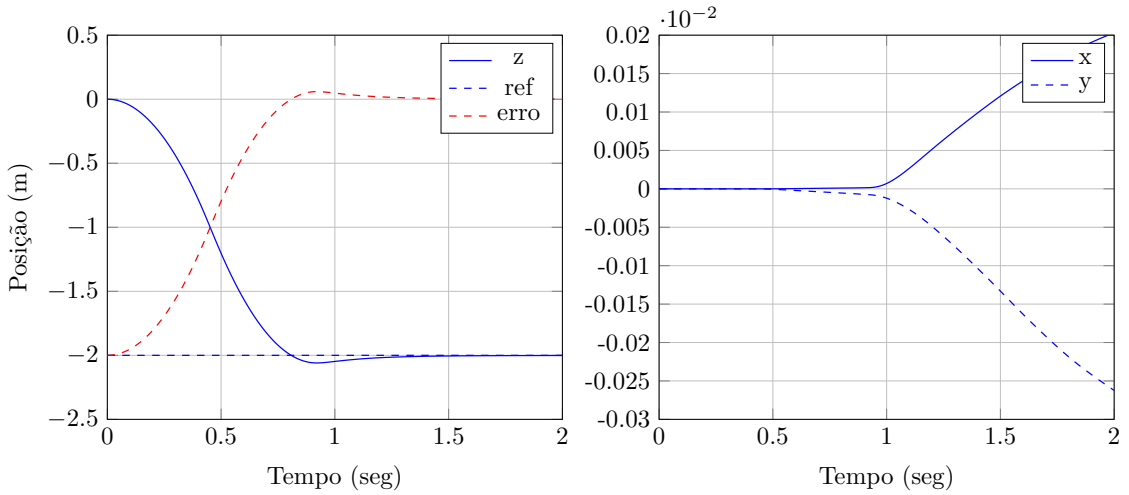


Figura 3.22: Altitude step response of the Proportional Derivative Controller with Variable Gain.

Vehicle Mass Uncertainty Analysis

When there is uncertainty in the vehicle's mass measurement, the *feedforward* control portion is not able to cancel the action of gravity. The effect of this residual force can be

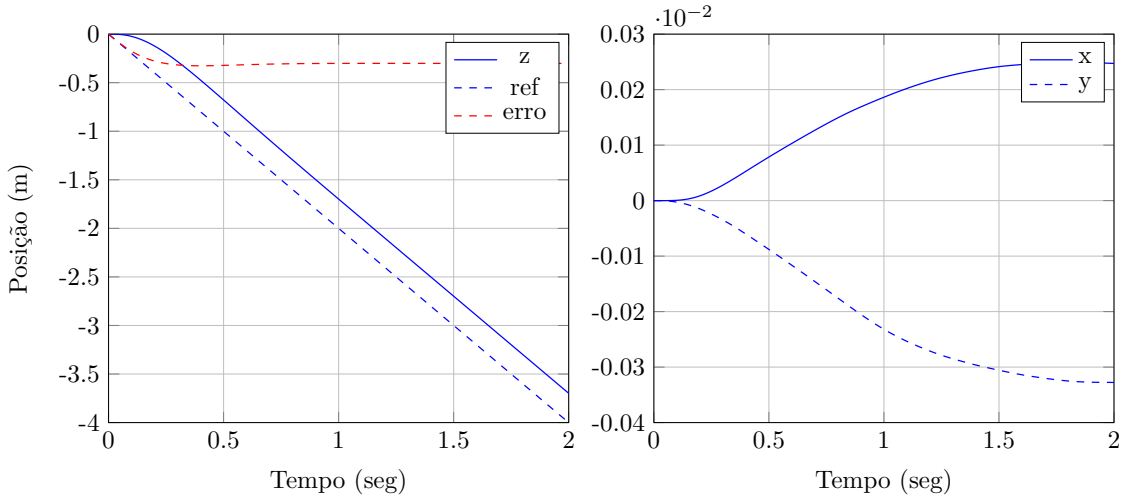


Figura 3.23: Altitude ramp response of the Proportional Derivative Controller with Variable Gain.

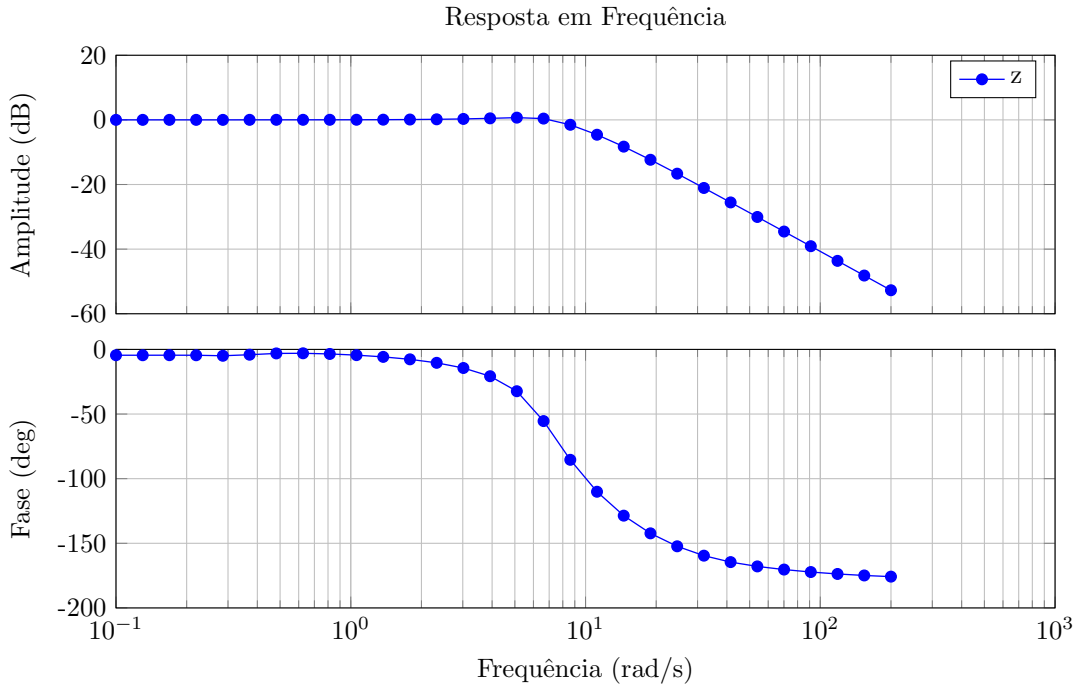


Figura 3.24: Frequency response of the Proportional Derivative Controller with Variable Gain.

understood as an input disturbance. If not compensated, this disturbance causes steady-state error. To test the performance of the control strategies, a 50% addition to the vehicle's actual mass will be considered. Thus, according to equation (3.19), we will have $\bar{m} = m/2$.

The steady-state error found when the PD controller is used with parameters given by Table 3.8 can be determined by equation (3.19). Its value will be $|e_z(\infty)| = 0.146$ m and can be verified in Figure 3.25. To compensate for this disturbance, it would be

possible to add an integral control portion. However, it is not possible to reach the desired specification in this case, which invalidates this solution. Figure 3.25 shows the result of the PID controller with integral gain $K_i = -50$, producing an overshoot of 30%.

To solve this problem, PD control with variable gain can be used. By increasing the gain as the error decreases, the static gain of the system also rises. When it is in static equilibrium, the sum of the *feedforward* action and the force generated by the control must be equal in absolute value to the vehicle's weight. That is, the minimum error will be

$$e_z(\infty)_{\min} = -\frac{\bar{m}g}{\max\{f_k(e_z)\}}. \tag{3.42}$$

Using the function $f_k(e_z)$ defined in (3.41) the minimum error will be $|e_z(\infty)_{\min}| = 0.046$ m. Figure 3.25 shows that the steady-state error value was $e_z(\infty) = 0.049$, very close to the minimum.

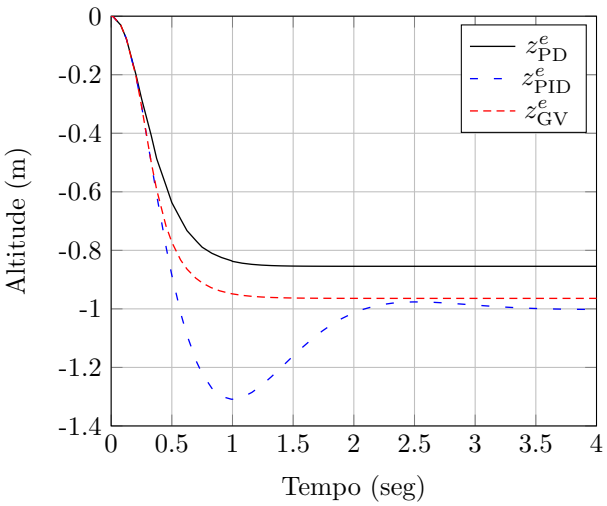


Figura 3.25: Response of PD, PID and PD with variable gain (VG) controllers to a 1 m step input in altitude and 50% uncertainty in mass.

Conclusion

With the results obtained for each of the control strategies, it is concluded that the PD control with variable gain was the one that obtained the best performance. This strategy was capable of meeting the specifications and was the one that presented the greatest bandwidth. Furthermore, the steady-state error resulting from uncertainty in the vehicle mass was reduced in a satisfactory manner, without this hindering system performance.

Although the PID controller is able to zero this error, it does so in exchange for a high overshoot. The addition of the integral action also requires attention to the need to implement integral *windup* strategies, adding more complexity to the control. Therefore, this solution becomes unfeasible in the sense of rejecting the considered input disturbance.

3.9 Teleoperation

Several quadrotor applications make use of teleoperation, i.e. a pilot is responsible for closing the position and heading loops of the vehicle. In these cases the *pitch* and *roll* loops are not changed, but adaptations to the position and heading loops are required. These changes are nothing more than simplifications that transfer part of the system's feedback to the operator. Usually, it is assumed that the position and heading references start to be given by velocity commands. As this approach is simplifying and is contained in the laws shown in this chapter, only a mention to its existence is made.

3.10 Conclusion

Based on what was seen in this chapter, there is enough information to decide regarding the most adequate control strategies to command the vehicle. As expected, the tested strategies have positive and negative points, which must be weighed in the final decision.

According to the obtained results it is possible to affirm that the most robust set of strategies is the one formed by:

- *Integral Backstepping* for orientation;
- PD for planar-xy position;
- PD with variable gain and *feedforward* for altitude

Together these strategies provide the best possible response among those tested in this work. This control set is more suitable for harsher flight conditions.

As for implementation cost, the variable gain and *Integral Backstepping* methods are considerably more complex than the classic PD. Concerning the altitude of the quadrotor, this control presents a greater error when there is uncertainty in the vehicle mass measurement. However, this error can be insignificant, especially when the mass does not

vary and its value is known a priori. As for orientation, a classic PD controller has good performance in conditions close to hover flight. Taking these factors into consideration, another possible set of strategies is given by:

- PD for orientation;
- PD for planar-xy position;
- PD with *feedforward* for altitude.

This set entirely formed by PD controllers is extremely simple and meets the needs found in slow flight conditions.

Capítulo 4

Control Allocation

4.1 Introduction

Determining the propeller speeds that produce the input vector $\mathbf{u}$ may not be possible through direct use of equation (3.2). This occurs mainly when negative values for the desired quadratic speeds are requested by the control loops. This phenomenon creates the need to formulate a new way of determining the quadratic speed vector $\mathbf{\Omega}$.

In the literature, the problem of determining the mapping between the desired force inputs and the actuator inputs of a system is called control allocation. This approach originated in the maritime and aerospace industries, but is currently applied in several other areas, such as robotics and the automotive industry. As observed in Chapter 3, the control allocation method allows a modular treatment of the control problem. This treatment enables the development of high-level strategies that use force vectors as control signals.

There are several techniques aimed at solving the control allocation problem. In [12] a survey of approaches used in various industries is carried out and references to important contributions in each area are cited. In [8] the problem is treated from the perspective of aircraft control, which is close to what is intended in this work. In this chapter, two well-known approaches will be studied: constrained least squares and direct allocation. A variation of the direct allocation strategy will also be proposed, with the objective of better adapting the method to the flight dynamics of a quadrotor vehicle.

4.2 Problem Definition

When dealing with multirotor vehicles, in the specific context of this work those with four propulsors (quadrotors), the control allocation problem reduces to determining the propeller speeds. These speeds must be such that the desired vertical force and *roll*, *pitch*, and *yaw* torques are achieved. In Section 3.1 the mapping from the propulsor rotation speeds to the force vector was established. For convenience, this equation is repeated below:

$$\mathbf{u} = \begin{bmatrix} \kappa_T & \kappa_T & \kappa_T & \kappa_T \\ 0 & -l\kappa_T & 0 & l\kappa_T \\ l\kappa_T & 0 & -l\kappa_T & 0 \\ -\kappa_Q & \kappa_Q & -\kappa_Q & \kappa_Q \end{bmatrix} \begin{bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \end{bmatrix} = \mathbf{M}\boldsymbol{\Omega}. \tag{4.1}$$

The control algorithms assume that the vehicle is controlled by the vector $\mathbf{u}$, therefore, one must determine $\boldsymbol{\Omega}$ that produces this vector. The matrix $\mathbf{M}$ has full rank and is invertible, so it would apparently always be possible to perform the inverse mapping. However, there are boundary constraints on the propeller rotation speed values,

$$0 \leq \Omega_{\min} \leq \Omega_i \leq \Omega_{\max}, \tag{4.2}$$

so that the virtual force vector is restricted to a feasible region.

One then defines $\mathbb{W}$ as the feasible set of control vectors $\boldsymbol{\Omega}$, determined by equation (4.2). This domain is mapped through equation (4.1) to form the set $\mathbb{U}$, which contains the feasible values of the virtual force vectors $\mathbf{u}$. Note that since $\mathbb{W}$ is polyhedral, more precisely a hypercube, the set $\mathbb{U}$ through the mapping is also polyhedral.

Therefore, given a desired virtual force vector, if $\mathbf{u} \in \mathbb{U}$, then the speed vector is readily obtained. Otherwise, it is necessary to employ some strategy to reposition the force vector so that its new value belongs to the feasible region. Since the problem is subject to boundary constraints, it is known that when $\mathbf{u}$ is not feasible, the best value that approximates this vector lies on the boundary of $\mathbb{U}$. The methods to be presented make use of this fact to optimize the value of the force vector according to some optimality criterion.

4.3 Constrained Least Squares

Traditionally, the approach that aims to minimize the quadratic error between the force vectors is treated as a sequential two-stage least squares problem. This occurs because in more general cases there are more actuators than degrees of freedom, so that the mapping between the desired forces and the actuator signals is not bijective. That is, there is more than one actuator signal configuration for the same virtual force vector. Fortunately, in the case of quadrotor vehicles this mapping is one-to-one, so that only a single least squares problem needs to be solved. If this were not the case, after finding the optimal solutions, a possible tiebreaker criterion would be to choose the one closest to the current actuator signal.

Therefore, the problem can be formulated as follows. Let $\mathbf{u_d}$ be the desired input vector, $\mathbf{\Omega_d} = \mathbf{M}^{-1}\mathbf{u_d}$ the desired quadratic speed vector, and $\Omega_{\text{min}} \geq 0$ the minimum propeller rotation speed. The optimization consists of:

$$\begin{aligned} \text{minimize} : f(\mathbf{\Omega}) &= \|\mathbf{u} - \mathbf{u_d}\|_2^2, \\ \text{s.t.} : \mathbf{g}(\mathbf{\Omega}) &= \mathbf{\Omega} - \Omega_{\text{min}}^2 \mathbf{1} \geq \mathbf{0}. \end{aligned} \tag{4.3}$$

Where $f : \mathbb{R}^4 \to \mathbb{R}$ is the objective function that aims to minimize the distance between the generated and desired forces and $\mathbf{g} : \mathbb{R}^4 \to \mathbb{R}^4$ a vector containing four constraints on the minimum propeller speeds. By the definition of the Euclidean norm,

$$f(\mathbf{\Omega}) = \mathbf{\Omega}^{\text{T}}\mathbf{M}^{\text{T}}\mathbf{M}\mathbf{\Omega} - 2\mathbf{\Omega}^{\text{T}}\mathbf{M}^{\text{T}}\mathbf{u_d} + \mathbf{u_d}^{\text{T}}\mathbf{u_d}. \tag{4.4}$$

Minimizing the problem through the method of Lagrange Multipliers consists of defining the Lagrangian $L(\mathbf{\Omega}) \triangleq f(\mathbf{\Omega}) - \boldsymbol{\lambda}^{\text{T}}\mathbf{g}(\mathbf{\Omega})$ and finding the solutions that satisfy the equations below.

$$\begin{cases} \dfrac{\partial L}{\partial \mathbf{\Omega}} = \dfrac{\partial f}{\partial \mathbf{\Omega}} - \dfrac{\partial \mathbf{g}}{\partial \mathbf{\Omega}}\boldsymbol{\lambda} = 0 \\ \mathbf{g}(\mathbf{\Omega}) \qquad\qquad\qquad\quad \geq \mathbf{0} \end{cases}, \quad \boldsymbol{\lambda} \triangleq \begin{bmatrix} \lambda_1 & \lambda_2 & \lambda_3 & \lambda_4 \end{bmatrix}^{\text{T}}. \tag{4.5}$$

Computing the partial derivatives, the optimal solutions must satisfy the necessary condi-

tions

$$\frac{\partial L}{\partial \mathbf{\Omega}} = \mathbf{M}^{\mathrm{T}} \left(\mathbf{M}\mathbf{\Omega} - \mathbf{u_d} \right) - \boldsymbol{\lambda} \quad \longrightarrow \quad \begin{cases} \mathbf{\Omega} & = \left(\mathbf{M}^{\mathrm{T}}\mathbf{M} \right)^{-1} \boldsymbol{\lambda} + \mathbf{\Omega_d} \\ \mathbf{g}(\mathbf{\Omega}) & \geq \mathbf{0} \end{cases} . \tag{4.6}$$

This system of equations has 16 solutions. These solutions are found by choosing a set of constraints to ignore $\lambda_j = 0$ and another set to enforce $g_i(\mathbf{\Omega}) = 0$. Defining then the matrix $\mathbf{R} \in \{0, 1\}^{n_i \times 4}$, where n_i is the number of constraints to be enforced, whose rows select the constraints $g_i(\mathbf{\Omega})$ from the constraint vector, the solutions arise from the linear system below:

$$\mathbf{R} \left(\mathbf{M}^{\mathrm{T}}\mathbf{M} \right)^{-1} \mathbf{R}^{\mathrm{T}}\mathbf{x} = \mathbf{R} \left(\Omega_{\min}\mathbf{1} - \mathbf{\Omega_d} \right), \tag{4.7}$$

so that

$$\boldsymbol{\lambda} = \mathbf{R}^{\mathrm{T}}\mathbf{x}. \tag{4.8}$$

Since the matrix $\mathbf{M}$ is known, it is possible to pre-compute the inverse of the matrix multiplying $\mathbf{x}$ on the left in (4.7). Note that this matrix has dimension $n_i \times n_i$ and rank n_i[1] and is therefore non-singular. Hence, it is always possible to determine the solution $\mathbf{x}$ of the problem.

To prove that these solutions are indeed global minima of the Lagrangian with fixed $\boldsymbol{\lambda}$, it is still necessary to show that

$$\frac{\partial^2 L}{\partial \mathbf{\Omega}^2} = \mathbf{M}^{\mathrm{T}}\mathbf{M} \tag{4.9}$$

is positive definite. By definition, since this matrix is symmetric, being positive definite is equivalent to saying that

$$\mathbf{v}^{\mathrm{T}} \left(\frac{\partial^2 L}{\partial \mathbf{\Omega}^2} \right) \mathbf{v} = \|\mathbf{M}\mathbf{v}\|_2^2 > 0, \quad \forall \mathbf{v} \neq \mathbf{0}. \tag{4.10}$$

Since the matrix $\mathbf{M}$ is non-singular, this equation holds. This concludes the proof.

[1] This conclusion can be obtained through the inequality $\rho(A) + \rho(B) - n \leq \rho(AB) \leq \min\{\rho(A), \rho(B)\}$ and using the fact that $\mathbf{M}$ has full rank and, therefore, $\left(\mathbf{M}^{\mathrm{T}}\mathbf{M} \right)^{-1}$ does as well.

4.3.1 Generalization of the Method

The objective function (4.3) and (4.4) to be optimized considers equal weights for all elements of the input vector. That is, no normalization or weighting is applied to them. It is generally of interest to weight the components of the force vector differently, whether to compensate for the difference in units or to give more importance to a particular measure.

To assign different weights to the inputs u_i, one defines the positive definite matrix $\mathbf{W} \in \mathbb{R}^{4\times4}$. Adding this weight matrix to the problem, the objective function is rewritten as follows:

$$f(\mathbf{\Omega}) = \|\mathbf{W}(\mathbf{u} - \mathbf{u_d})\|_2^2 . \tag{4.11}$$

The solution of this new problem is obtained by using a matrix $\overline{\mathbf{M}} = \mathbf{WM}$ in place of the matrix $\mathbf{M}$. With this change, all previous results are preserved.

In practice, the real vehicle also has saturations related to the maximum propeller rotation speed $\Omega_{\text{max}} \geq \Omega_{\text{min}}$. In this case, four new constraints are added, resulting in 81 candidate solutions. These candidate solutions are found as in the previous section, requiring only that the maximum and minimum constraints for the same propeller are not satisfied simultaneously.

It becomes evident that considering all possible solutions to determine the optimal control is not adequate. If this were done, 1620 additions and 1755 multiplications would be required, consuming too much processing time and making the implementation infeasible.

To circumvent this problem, a simple iterative method can be employed. This method consists of saturating at each iteration one of the values that exceeds the boundary and computing the solution in this situation. The procedure is repeated until a valid solution is found. This algorithm converges in at most four iterations, requiring at most (at least) 80 (28) additions and 85 (32) multiplications. Other more advanced methods can be found in [8].

4.4 Direct Allocation

The direct allocation method seeks to rescale the control vector such that it lies on the boundary of the region formed by the constraints. Naturally, since rescaling the control vector preserves its direction, the solution found will only be slower than the desired one. Therefore, unlike the least squares method, in addition to determining an optimal solution, this method guarantees that no unpredictable movements will be generated.

Let $\mathbb{U}$ be the feasible set of control vectors $\mathbf{u}$, $\mathbb{W}$ the feasible set of control vectors $\mathbf{\Omega}$, and $\mathbf{u_d}$ the desired force vector, the method can be described as follows:

1. If $\mathbf{u_d} \in \mathbb{U}$, then set $\mathbf{\Omega} = \mathbf{M}^{-1}\mathbf{u_d}$ and the value is already determined.
2. Otherwise, find $\alpha \in \mathbb{R}$ such that $\bar{\mathbf{u}}_\mathbf{d} = \alpha\mathbf{u_d}$ lies on the nearest boundary of the feasible region $\mathbb{U}$.
3. With the value determined in 2, set $\mathbf{\Omega} = \mathbf{M}^{-1}\bar{\mathbf{u}}_\mathbf{d}$.

As mentioned, this approach preserves the direction and sense of the control vector. Furthermore, it does so by minimizing the distance between the generated vector and the desired one when the problem is subject to this direction constraint. This treatment has the advantage of preserving the characteristics of the desired control. However, when no constraint is imposed on the direction of the control vector, there is no guarantee that this solution will be optimal.

4.4.1 Method Variation

As defined, the direct allocation method rescales the entire virtual force vector so that it lies on the boundary of the feasible region. The choice of this approach is not necessarily the most adequate, and it is possible to study variations of this method. Therefore, two alternatives to the direct allocation technique are proposed, both of which preserve the direction of a three-dimensional subvector of the force vector.

First, one must define the topology of the feasible region $\mathbb{U}$. More precisely, how the

hypercube $\mathbb{W}$ defined by (4.2) is mapped into $\mathbb{U}$. Applying the inverse mapping

$$\boldsymbol{\Omega} = \frac{1}{4l\kappa_T} \begin{bmatrix} l & 0 & 2 & -l\kappa_T\kappa_Q^{-1} \\ l & -2 & 0 & l\kappa_T\kappa_Q^{-1} \\ l & 0 & -2 & -l\kappa_T\kappa_Q^{-1} \\ l & 2 & 0 & l\kappa_T\kappa_Q^{-1} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \mathbf{M}^{-1}\mathbf{u} \tag{4.12}$$

and the known propulsor constraints (4.2), one obtains the inequalities below that define $\mathbb{U}$:

$$\begin{aligned} 4\Omega_{\text{min}}^2 \;&\leq\; \kappa_T^{-1}u_1 \pm 2(l\kappa_T)^{-1}u_2 + \kappa_Q^{-1}u_4 \;\leq\; 4\Omega_{\text{max}}^2 \\ 4\Omega_{\text{min}}^2 \;&\leq\; \kappa_T^{-1}u_1 \pm 2(l\kappa_T)^{-1}u_3 - \kappa_Q^{-1}u_4 \;\leq\; 4\Omega_{\text{max}}^2 \end{aligned}. \tag{4.13}$$

It becomes evident that this region is characterized by a set of hyperplanes in $\mathbb{R}^4$. It is also possible to define a change of variables to make this fact more evident. Therefore, consider the auxiliary variables

$$\begin{aligned} \tilde{u}_1 &= 2\kappa_T^{-1}u_1 \\ \tilde{u}_2 &= 2(l\kappa_T)^{-1}u_2 \\ \tilde{u}_3 &= 2(l\kappa_T)^{-1}u_3 \\ \tilde{u}_4 &= 2\kappa_Q^{-1}u_4 \end{aligned} \quad \text{and} \quad \begin{aligned} \tilde{\Omega}_{\text{min}}^2 &= 4\Omega_{\text{min}}^2 \\ \tilde{\Omega}_{\text{max}}^2 &= 4\Omega_{\text{max}}^2 \end{aligned}. \tag{4.14}$$

Using these variables in (4.13), the feasible region of virtual forces is defined by the inequalities

$$\begin{aligned} \tilde{\Omega}_{\text{min}}^2 \;&\leq\; \tilde{u}_1/2 \pm \tilde{u}_2 + \tilde{u}_4/2 \;\leq\; \tilde{\Omega}_{\text{max}}^2 \\ \tilde{\Omega}_{\text{min}}^2 \;&\leq\; \tilde{u}_1/2 \pm \tilde{u}_3 - \tilde{u}_4/2 \;\leq\; \tilde{\Omega}_{\text{max}}^2 \end{aligned}. \tag{4.15}$$

It remains now to define the two subvectors of $\mathbf{u_d}$ for which one wishes to preserve the desired direction. A natural choice is the one that selects the last three components, responsible for the *roll*, *pitch*, and *yaw* torques. In this way, there is a guarantee that the torque direction will not be altered and, therefore, the vehicle will rotate about the same axis originally defined. Another possible choice arises from assigning less importance to the *yaw* of the vehicle. In this case, a subvector with the first three components of $\mathbf{u_d}$ is chosen. The feasible regions in $\mathbb{R}^3$ defined by these strategies are shown in Figure 4.1. In

what follows, these two cases will be studied in greater detail.

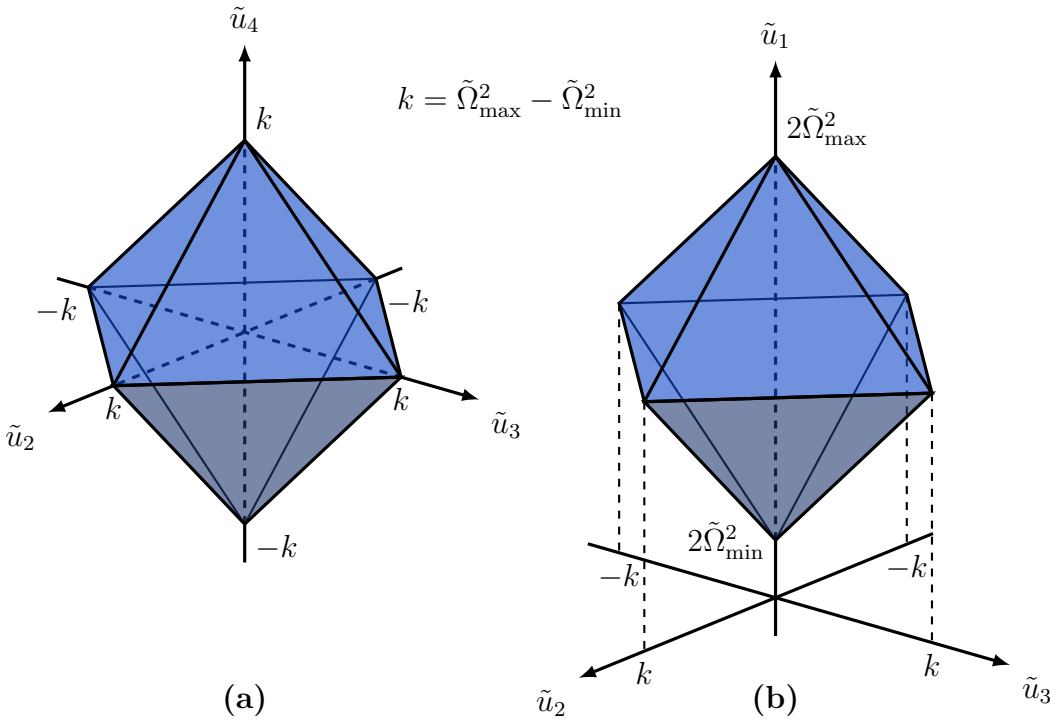


Figura 4.1: Octahedra defining the feasible regions when (a) u_1 is kept free and (b) u_4 is kept free.

Free Vertical Force

In this case, the vertical force exerted on the vehicle, represented by the component u_1 of the virtual force vector, is kept free. That is, one wishes to preserve the direction only of the torque components $\mathbf{u_\tau}^{\mathrm{T}} = [u_2\ u_3\ u_4]$. Subtracting the two inequalities computed in (4.15), the force component is removed, leaving only four inequalities, shown below in compact form:

$$\tilde{\Omega}^2_{\min} - \tilde{\Omega}^2_{\max} \leq \tilde{u}_4 \pm \tilde{u}_3 \pm \tilde{u}_2 \leq \tilde{\Omega}^2_{\max} - \tilde{\Omega}^2_{\min}. \tag{4.16}$$

Each of these eight planes is defined by a normal vector $\mathbf{n} \in \mathbb{R}^3$ and a point $\mathbf{p_0} \in \mathbb{R}^3$ contained in them. Note that every two planes share the same normal vector, i.e., they are parallel. Therefore, the eight planes can be defined by four normal vectors and two points:

$$\mathbf{n}^{\mathrm{T}} = \begin{bmatrix} \pm 1 & \pm 1 & 1 \end{bmatrix} \quad \text{and} \quad \mathbf{p_0}^{\mathrm{T}} = \begin{bmatrix} 0 & 0 & \pm(\tilde{\Omega}^2_{\max} - \tilde{\Omega}^2_{\min}) \end{bmatrix}. \tag{4.17}$$

With the plane equations, the intersection of the torque vector $\mathbf{u}_\tau$ with them can be computed. Geometrically, one can conclude that the scaling constant $\alpha \in \mathbb{R}$ that makes this vector touch the planes is:

$$\alpha = \frac{\mathbf{n} \cdot \mathbf{p_0}}{\mathbf{n} \cdot \bar{\mathbf{u}}} = \pm \frac{\tilde{\Omega}^2_{\max} - \tilde{\Omega}^2_{\min}}{\tilde{u}_4 \pm \tilde{u}_3 \pm \tilde{u}_2}. \tag{4.18}$$

Since the torque vector must not change its sense and one wishes to find an intersection on the boundary of the region shown in Figure 4.1, then

$$\alpha = \frac{\tilde{\Omega}^2_{\max} - \tilde{\Omega}^2_{\min}}{|\tilde{u}_2| + |\tilde{u}_3| + |\tilde{u}_4|} = \frac{2\left(\Omega^2_{\max} - \Omega^2_{\min}\right)}{(l\kappa_T)^{-1}\left(|u_2| + |u_3|\right) + \kappa_Q^{-1}|u_4|}. \tag{4.19}$$

With the value of α determined, the torque components of the control vector are rescaled according to $\alpha\mathbf{u}_\tau$. It remains only to determine the force component. To this end, it is necessary to return to (4.15), from which the following result can be extracted:

$$\begin{aligned} \tilde{u}_1 &\geq 2\tilde{\Omega}^2_{\min} + \max\left\{2|\tilde{u}_2| - \tilde{u}_4,\ 2|\tilde{u}_3| + \tilde{u}_4\right\} \\ \tilde{u}_1 &\leq 2\tilde{\Omega}^2_{\max} + \min\left\{2|\tilde{u}_2| - \tilde{u}_4,\ 2|\tilde{u}_3| + \tilde{u}_4\right\} \end{aligned}. \tag{4.20}$$

If the value of the force component satisfies these inequalities, its value is kept. Otherwise, its value is saturated to equal that of the violated boundary.

Free Yaw Torque

Another possible choice would be to allow the sense of the vehicle's yaw torque to be altered. In this case, the subvector for which one wishes to preserve the direction and sense would be composed of the first three components of $\mathbf{u}$. There are three problems with this approach:

1. The physical meaning of preserving the sense of a vector composed of torques and vertical force is not clear;
2. The sense of the yaw torque can be altered, introducing a non-minimum phase behavior in the yaw response of the system;
3. The problem may have no solution, since the feasible region does not contain the origin, see Figure 4.1.

In view mainly of the third item, this case cannot be classified as a direct allocation problem, because in order for a solution to be found it may be necessary to alter the direction of the chosen subvector. Therefore, this case is disregarded in this context.

4.5 Conclusion

In this chapter, the concept of control allocation applied to quadrotor vehicles was presented. This notion is usually omitted in the current literature on this class of vehicles. Two distinct approaches aimed at solving the allocation problem were proposed. The first was based on minimizing an objective function defined by the distance between a new control vector and the desired one. The second was derived from the definition of direct allocation, designed to guarantee the direction and sense of the desired torque.

The parallel between the objective of the second approach and the desired behavior of the virtual force vector is evident. Furthermore, the least squares solution requires a higher computational effort and its implementation is more complex. Therefore, direct allocation with free vertical force was chosen as the control allocation strategy. All results in Chapter 3 were obtained using this method.

Capítulo 5

Experimental Results

In order to validate the theoretical results, several experiments were conducted with a real quadrotor. Due to practical limitations, it was not possible to validate the more advanced strategies proposed in this work, so the implementation was restricted to PD control strategies.

The results shown in this section were obtained at the Electrical Engineering Laboratory (LEE) of the State University of Rio de Janeiro (UERJ). All measurements were taken using the Vicon motion capture system, Figure 5.1(b), at a sampling rate of 50 Hz. The vehicle used was an AR.Drone by Parrot. In Figure 5.1(a) one can observe this vehicle on the ground and the available flight area. In [5] and [6] the authors clearly describe the integration process of the systems involved and implement a digital control strategy similar to the one that will be presented.

(a) (b)

Figura 5.1: AR.Drone quadrotor by Parrot and available flight area (a); one of the four Vicon cameras used for capturing the vehicle's pose (b).

Since it was not possible to access the vehicle's inner attitude control loop, the implementation of the control strategies studied in this work was compromised. The communication system with the quadrotor accepts four control signals:

1. pitch inclination;
2. roll inclination;
3. climb velocity;
4. yaw velocity.

Therefore, a PD controller was implemented for position in the horizontal plane. For the other loops, a proportional-only controller was designed. The control laws were designed according to (3.12) and the parameters in Table 3.1. The control discretization was performed using the backward Euler approximation,

$$\frac{dx(t)}{dt} \approx \frac{x(t) - x(t-h)}{h}. \tag{5.1}$$

It is also worth noting that the measurement processing, followed by the transmission of the control signals, was kept in synchrony with the arrival of data sent by the Vicon capture system. This synchrony is important so that the system is approximately time-invariant with respect to digitization [1].

After several tests, the position and yaw control loops were tuned according to Table 5.1. To test the control performance, several step references were sent to each loop and the vehicle's response was observed. The flight results can be seen in Figure 5.2.

Tabela 5.1: Gains used in the PID position controllers.

Loop	K_p	K_d
Planar Position	0.4	0.25
Altitude	0.8	0
Yaw	0.64	0

It is worth noting that the planar position control loop exhibited a similar error in x and y, approximately 5 cm. The apparent difference in the plots is only due to the difference in scale. In turn, the altitude control loop exhibited a small steady-state error of approximately 2 cm. The yaw loop was able to completely eliminate the error, presenting a smoother response than the others.

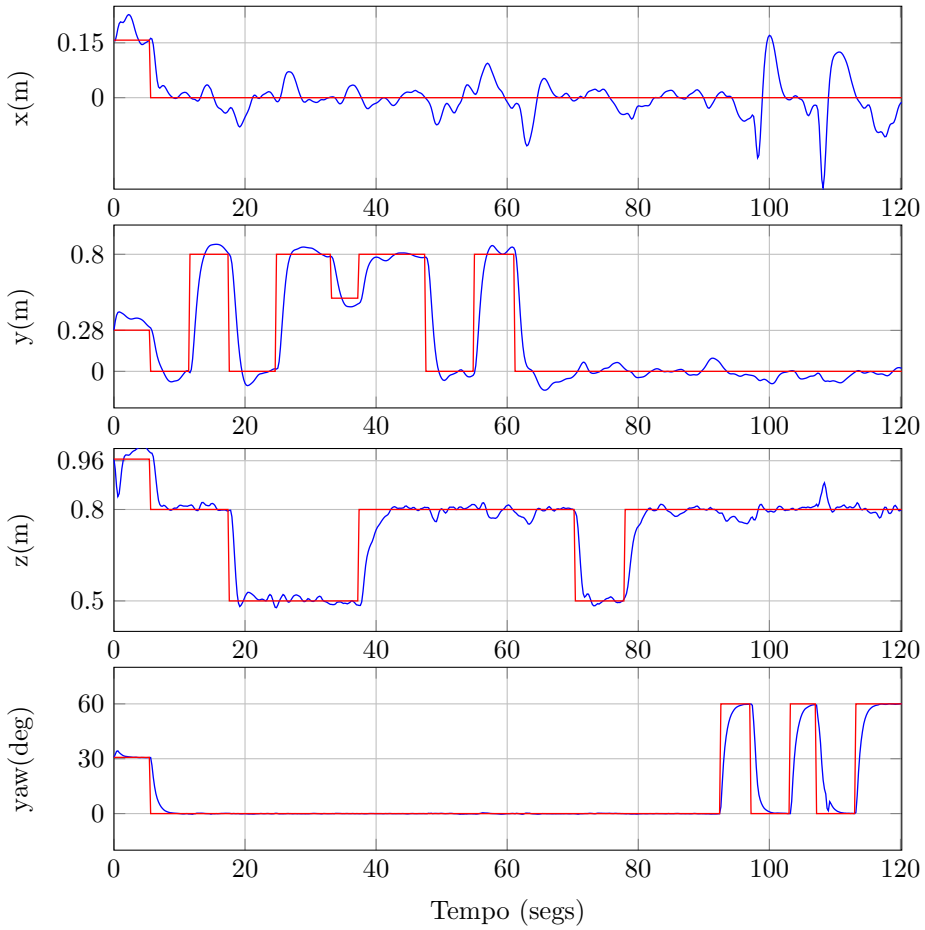


Figura 5.2: Flight results using the gains from Table 5.1.

Since the yaw and altitude control is done by velocity, a first-order response is expected. For yaw this indeed occurs, but not for altitude, which does not exhibit as clear a first-order response. This is due to a slower response of the velocity controller internal to the vehicle.

Overall, the responses were close to those of the theoretical model. Therefore, it is concluded that the model is valid and satisfactory for the design of the control strategies that were experimentally implemented.

Capítulo 6

Conclusion and Future Work

In this work the problem of dynamic modelling and control of quadrotor vehicles was addressed. A model that considers several aerodynamic and inertial effects was derived and some control strategies were designed to control this model. It was shown that these vehicles possess cascade structures, where the inner loop is responsible for attitude control and the outer loop for the vehicle's position control. Together these loops control the vehicle's pose.

The control problem was studied in detail, enabling an intuitive analysis of the control laws used in quadrotor vehicles. Furthermore, an altitude control strategy with variable gain was presented, aiming to reduce the steady-state error due to mass uncertainty. The global asymptotic stability of this law was proven using the Circle Criterion.

In accordance with what is done in the UAV literature, a virtual control vector was considered. Assuming this vector as the system input, the control strategies were designed. Subsequently, it was shown that it is not always possible to obtain the motor speeds from this virtual vector and a new treatment based on control allocation was presented. Two solutions for obtaining this mapping were shown. The first was based on an optimality criterion using constrained least squares. The second solution employed the concept of direct allocation to maintain the direction of the torque desired by the controller, possibly at the cost of a reduction in vertical force.

6.1 Future Work

The following research topics are proposed to continue this work:

1. **Application of the variable gain strategy to other control laws:**

   In Chapter 3 it was shown that the variable gain strategy applied to PD altitude control is capable of producing a faster response. Moreover, this controller also reduces errors due to parametric uncertainties. It is thus proposed to add this variable gain concept to other control strategies, possibly allowing an improvement in initial performance.

2. **Study of the performance of an adaptive controller for altitude and attitude:**

   The motivation for using an adaptive control strategy is the reduction of the controller's sensitivity to disturbances. Using this controller it would be possible to adapt several model parameters. It is expected that an adaptive approach would be capable of producing a control law more robust to uncertainties and disturbances.

3. **Use of the Kalman Filter to estimate states:**

   Another interesting research topic would be the use of the Kalman Filter to estimate the states of the vehicle's dynamic model. The results should then be compared to those obtained when using: (i) traditional deterministic observers and (ii) approximate differentiators for obtaining velocities.

4. **Obtaining more conclusive experimental results:**

   More detailed experimental tests of the control strategies obtained in this work can still be carried out. These would allow a comparison between the theoretical and practical performances of the proposed laws, in addition to providing feedback that enables their improvement.

Apêndice A

Noise Dimensioning

In Chapter 3 the presence of white noise in the pose, linear velocity, and angular velocity measurements of the vehicle was considered. In this appendix the procedure used to dimension these signals will be described.

The environment used during the experiments is briefly described in Chapter 5. For a better understanding of the procedure that will be described, reading that chapter is suggested.

A.1 Procedure

The vehicle was placed at the center of the camera capture area and was kept stationary for approximately 14 seconds. The capture data were saved and can be seen in Figures A.1 and A.2.

Computing the mean and standard deviation from the data, the values shown in Table 3.2 were obtained. The velocity values were obtained assuming zero initial conditions and the first-order approximation

$$\dot{f}(k) \approx \frac{f(k) - f(k-1)}{h} \tag{A.1}$$

for the derivative computation. In the experiment conducted $h = 0.02$, since the sampling rate was 50 Hz.

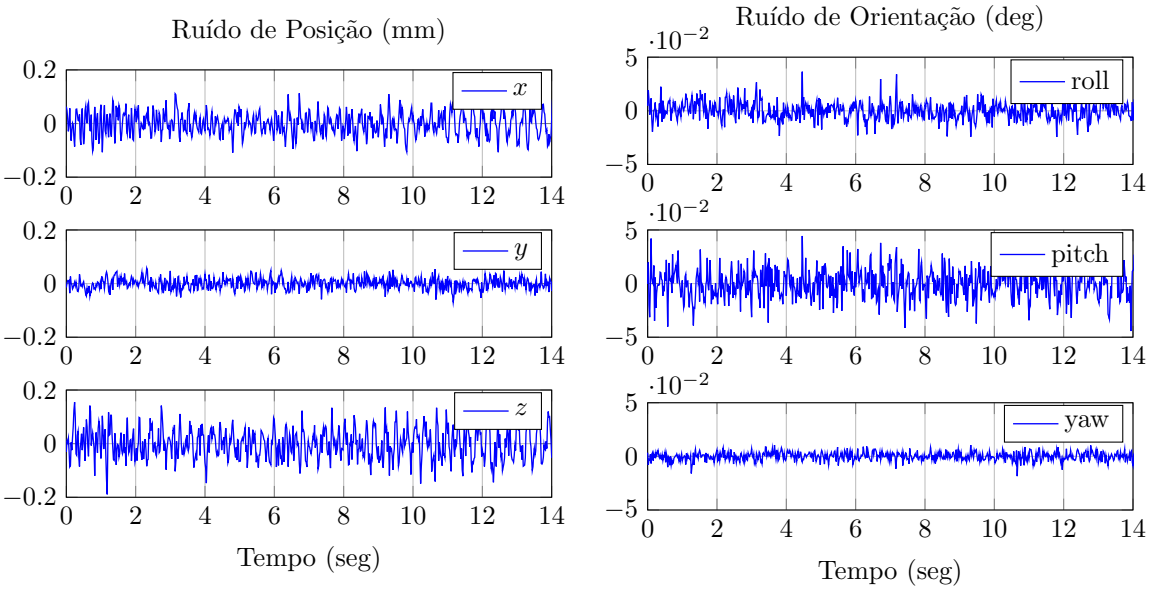


Figura A.1: Pose capture data of the vehicle resting on the ground.

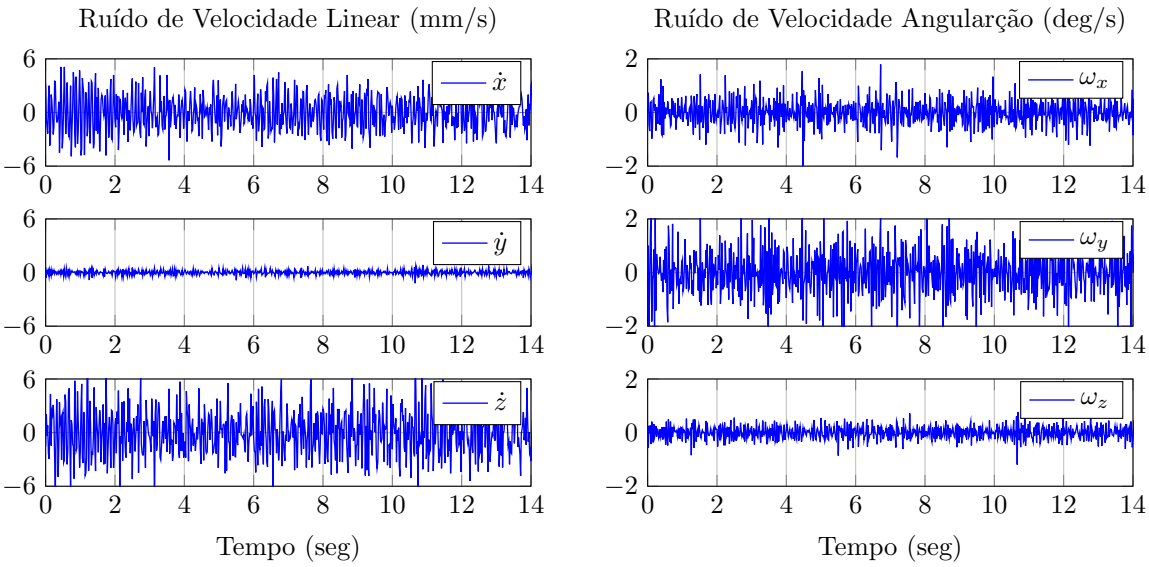


Figura A.2: Velocity data derived from the pose capture of the vehicle resting on the ground.

Apêndice B

Estimation of Thrust and Drag Constants

Since closed-loop control was not implemented in practice, the estimation of the torque and drag constants was performed in open loop. The AR.Drone quadrotor already has an internal attitude loop, making it easy to fly. The vehicle accepts roll and pitch inclination commands, as well as yaw rotational velocity and altitude linear velocity commands. The control signal reported by the vehicle is the PWM signal, in the range of 0 to 255, sent by the vehicle to its motors. Since this is approximately proportional to the propeller rotation speed, it will be considered that the propellers spin at speeds limited by the same PWM range.

Two experiments were conducted. In the first, an attempt was made to maintain the vehicle in stationary hover with constant heading. In the second experiment, only the heading was allowed to vary. The flight results are shown in Figure B.1. The procedure for extracting the parameters from the results is described for each case.

B.1 Estimation of the Thrust Constant

When the quadrotor is in stationary hover, it follows directly from the force equilibrium of the system that the resultant vertical force produced by the motors must equal the vehicle's weight. Therefore, using the model obtained in equation (3.6), the assumption that there is

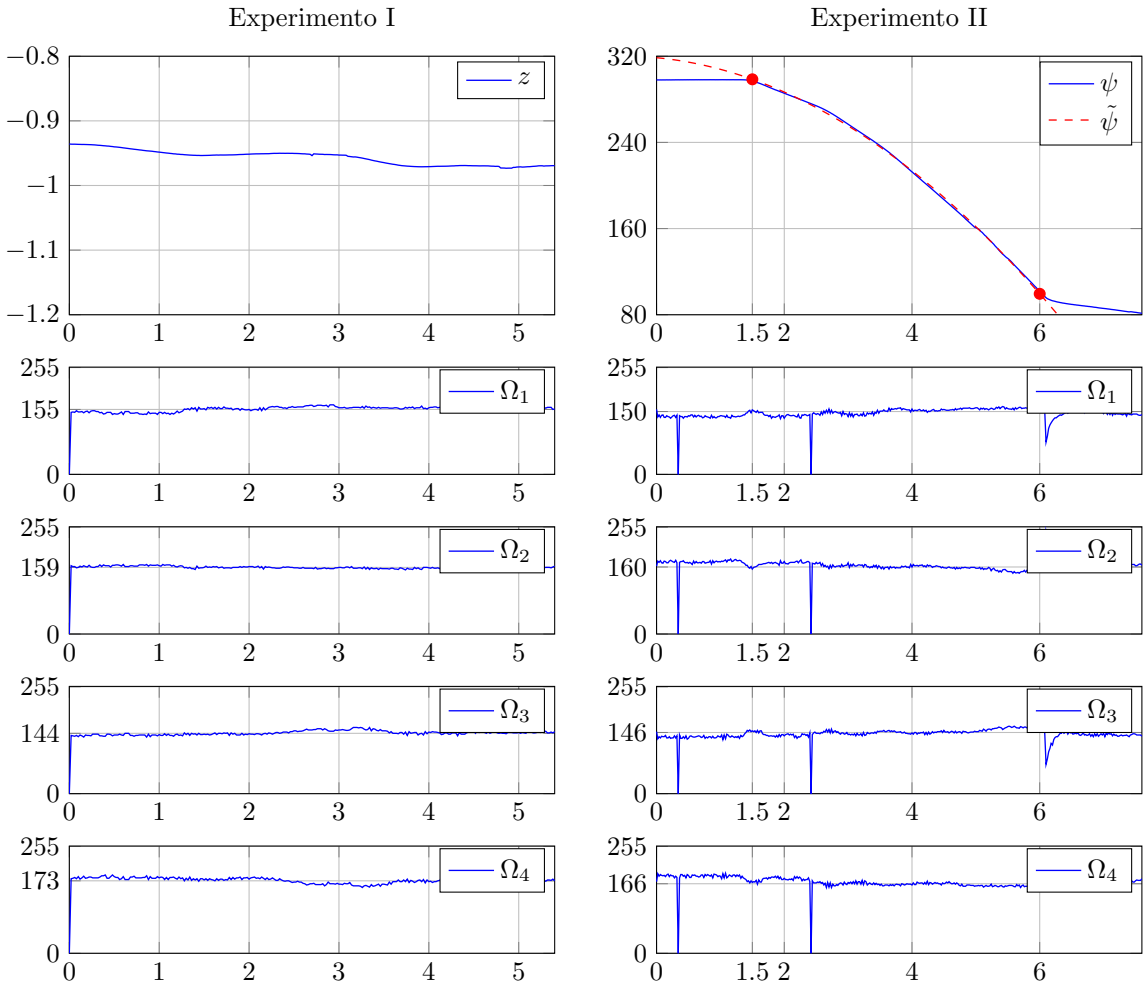


Figura B.1: Pose capture data of the vehicle resting on the ground.

no altitude variation, and the mean motor speeds $\bar{\Omega}_i$, one can write the following equality:

$$4\kappa_T \sum_{i=1}^{4} \bar{\Omega}_i^2 = mg. \tag{B.1}$$

Substituting the mean motor speeds from the first experiment, the vehicle mass, and the gravitational acceleration, one obtains:

$$\kappa_T = 8.7 \times 10^{-5}\ [\text{N.s}^2]. \tag{B.2}$$

B.2 Estimation of the Drag Constant

Since the vehicle exhibits low angular velocities in the second experiment, as well as small roll and pitch angles, the yaw dynamics can be written as:

$$I_{zz}\ddot{\psi} = \kappa_Q \sum_{i=1}^{4} \Omega_i^2. \tag{B.3}$$

Note that when a constant torque is applied to the system, the equation predicts that the yaw angle exhibits a parabolic response.

In the experiment conducted, a constant signal was maintained on the motors during the period from 1.5 to 6 seconds and it was possible to observe this predicted response. Figure B.1 shows these experimental results, as well as the mean motor speeds within the aforementioned time interval and the best-fit parabola approximating the yaw data,

$$\tilde{\psi}(t) = -5.1467t^2 - 5.6547t + 318.6370 \text{ [deg]} \quad \therefore \quad \ddot{\tilde{\psi}} = -5.1467 \text{ [deg/s}^2\text{]}. \tag{B.4}$$

Using this angular acceleration value, the mean speeds within the interval of interest, and the known inertia value, the drag constant is:

$$\kappa_Q = 1.4 \times 10^{-5} \text{ [N.m.s}^2\text{]}. \tag{B.5}$$

Apêndice C

Simulation Platform

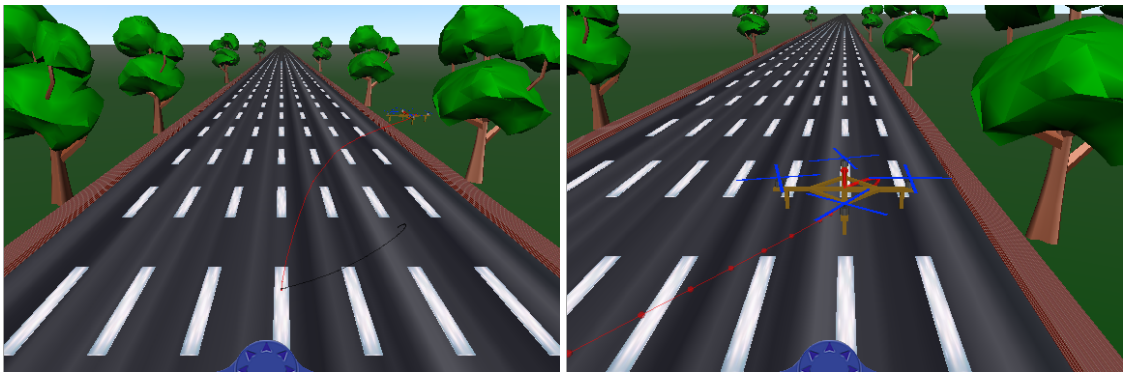
The platform for simulating the vehicle dynamics, as well as the control strategies, was developed in Matlab® and Simulink®. Through this simulation environment it was possible to easily test the algorithms proposed in this work, as well as visualize the system response. This visualization can be done both quantitatively, through the analysis of the generated plots, and qualitatively, by observing the animation that shows the vehicle's flight.

C.1 Virtual World

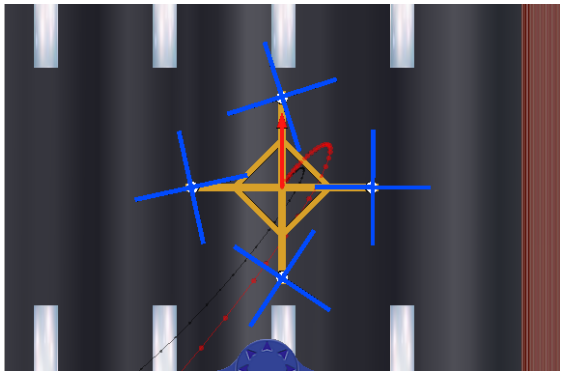
For the animation a virtual world was created, following the VRML (Virtual Reality Modeling Language) standard. This standard allows the creation of an interactive three-dimensional vector drawing environment. Despite the existence of the more modern X3D standard, which succeeds VRML, the latter was chosen because up to the current Matlab version, R2015a, there is no support for the X3D standard. Figure C.1 shows three predefined views in the VRML file created.

C.2 Vehicle Dynamics Implementation

Obviously, the core of the vehicle simulation is the implementation of its dynamics. This implementation is done using the Matlab API (Application Programming Interface) called Level-2 Matlab® S-function. Through this API it is possible to easily program the dynamics of nonlinear MIMO systems and integrate the code directly with Simulink. Together



(a) Fixed front view (b) Front view



(c) Top view

Figura C.1: Predefined views of the quadrotor in the VRML file.

with this file that implements the dynamics, an initialization file is used where the quadrotor parameters are defined. When calling this initialization file it is possible to freely change the default values defined therein. The code used in these two files is shown below.

Referências Bibliográficas

[1] ÅSTRÖM, K. J., WITTENMARK, B., 2013, *Computer-controlled systems: theory and design*. Courier Corporation.

[2] BOUABDALLAH, S., 2007, *Design and Control of Quadrotors with Application to Autonomous Flying*. Tese de Doutorado, École Polytechnique Fédérale de Lausanne.

[3] BOUABDALLAH, S., SIEGWART, R., 2005, "Backstepping and sliding-mode techniques applied to an indoor micro quadrotor". In: *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pp. 2247–2252. IEEE.

[4] BOUABDALLAH, S., NOTH, A., SIEGWART, R., 2004, "PID vs LQ control techniques applied to an indoor micro quadrotor". In: *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, v. 3, pp. 2451–2456. IEEE.

[5] GOMES, L. L., LEAL, L., 2014. "Controle de um Quadricóptero por Servovisão". .

[6] GOMES, L. L., LEAL, L., OLIVEIRA, T. R., et al., "Controle de um Veículo quadrirrotor usando um sistema de captura de movimentos", .

[7] GORDON, L. J., 2006, "Principles of helicopter aerodynamics", *Second editions*, , n. 12.

[8] HARKEGARD, O., 2002, "Efficient active set algorithms for solving constrained least squares problems in aircraft control allocation". In: *Decision and Control, 2002, Proceedings of the 41st IEEE Conference on*, v. 2, pp. 1295–1300. IEEE.

[9] HEHN, M., D'ANDREA, R., 2011, "A flying inverted pendulum". In: *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pp. 763–770. IEEE.

[10] HOFFMANN, G. M., HUANG, H., WASLANDER, S. L., et al., 2007, "Quadrotor helicopter flight dynamics and control: Theory and experiment". In: *Proc. of the AIAA Guidance, Navigation, and Control Conference*, v. 2.

[11] HUANG, H., HOFFMANN, G. M., WASLANDER, S. L., et al., 2009, "Aerodynamics and control of autonomous quadrotor helicopters in aggressive maneuvering". In: *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pp. 3277–3282. IEEE.

[12] JOHANSEN, T. A., FOSSEN, T. I., 2013, "Control allocation – A survey", *Automatica*, v. 49, n. 5, pp. 1087–1103.

[13] KHALIL, H. K., GRIZZLE, J., 1996, *Nonlinear systems*, v. 3. Prentice hall New Jersey.

[14] LUPASHIN, S., SCHOLLIG, A., SHERBACK, M., et al., 2010, "A simple learning strategy for high-speed quadrocopter multi-flips". In: *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pp. 1642–1648. IEEE.

[15] MELLINGER, D., KUMAR, V., 2011, "Minimum snap trajectory generation and control for quadrotors". In: *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pp. 2520–2525. IEEE.

[16] MELLINGER, D., MICHAEL, N., KUMAR, V., 2012, "Trajectory generation and control for precise aggressive maneuvers with quadrotors", *The International Journal of Robotics Research*, p. 0278364911434236.

[17] MICHAEL, N., FINK, J., KUMAR, V., 2011, "Cooperative manipulation and transportation with aerial robots", *Autonomous Robots*, v. 30, n. 1, pp. 73–86.

[18] MULLER, M., LUPASHIN, S., D'ANDREA, R., 2011, "Quadrocopter ball juggling". In: *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pp. 5113–5120. IEEE.

[19] POUNDS, P., MAHONY, R., CORKE, P., 2010, "Modelling and control of a large quadrotor robot", *Control Engineering Practice*, v. 18, n. 7, pp. 691–699.

[20] SEDDON, J. M., NEWMAN, S., 2011, *Basic helicopter aerodynamics*, v. 40. John Wiley & Sons.