TÉCNICO LISBOA

Linear and Nonlinear Control of UAVs: Design and Experimental Validation

Luís Carlos Cunha Martins

Thesis to obtain the Master of Science Degree in

Mechanical Engineering

Supervisors: Prof. Paulo Jorge Coelho Ramalho Oliveira
Prof. Carlos Baptista Cardeira

Examination Committee

Chairperson: Prof. Carlos Frederico Neves Bettencourt da Silva
Supervisor: Prof. Paulo Jorge Coelho Ramalho Oliveira
Members of the Committee: Prof. Miguel Afonso Dias de Ayala Botto
Prof. Pedro Daniel Graça Casau

June 2019

Acknowledgments

This dissertation is the culmination of five and a half years of intensive work and learning at IST. I want to start by thanking the professors that have made this time so fruitful in terms of knowledge.

I want to express my gratitude to my thesis supervisors, Professor Paulo Oliveira and Professor Carlos Cardeira, for their patient guidance, enthusiastic encouragement, useful critiques and constructive recommendations. Their confidence in my capabilities led me to write and submit a conference paper based on the work developed during this thesis.

I would also like to extend my thanks to ACCAII, for providing me the space and the resources required to conduct the experimental tests, and to FCT and IDMEC for the financial support.

To my colleagues, that have become great friends throughout this journey, I express my gratitude. To João Madeiras, Nuno Guerreiro and Tiago Bacelar, for their help during the projects, for inspiring me and for the companionship since the first year at IST, my thanks.

I sincerely thank my girlfriend, Cristina, for her crucial support and encouragement that kept me motivated during this final stretch.

Lastly, my deepest gratitude is expressed towards my mother, for being so supportive and believing in me throughout my life, and my father, that would undoubtedly be very proud of this accomplishment.

Resumo

A presente dissertação apresenta soluções de controlo posicional e angular para um *drone* recorrendo a métodos lineares e não lineares.

O modelo dinâmico do veículo aéreo é obtido. Os controladores não lineares são projetados considerando o modelo completo, ao passo que, para a síntese dos controladores lineares, uma linearização das dinâmicas do sistema é requerida. *Linear Quadratic Regulator* e *Feedback Linearization* foram as estratégias utilizadas para atacar o problema de controlo. Duas arquiteturas distintas tendo por base a abordagem linear foram concebidas e três diferentes estruturas de controlo, explorando os conceitos de realimentação estática e dinâmica, foram desenvolvidas com recurso à técnica não linear. A capacidade de seguimento de uma trajetória na presença de medições ruidosas de sensores e a robustez a variações significativas dos valores da massa e da inércia foram avaliadas em simulação, possibilitando, deste modo, a determinação das soluções lineares e não lineares mais promissoras com vista a serem posteriormente validadas experimentalmente.

A abordagem linear selecionada consiste numa estrutura de controlo em anel duplo na qual o anel interno é responsável por estabilizar as dinâmicas angulares e o anel externo serve o propósito de controlar o posicionamento. Em ambos os anéis, uma ação integrativa é incorporada. A segunda solução compreende um anel interno não linear que resulta da aplicação do método *Feedback Linearization* às dinâmicas angulares. Controladores quadráticos lineares com ação integrativa são implementados na cadeia de integradores resultante desta aplicação e também no anel externo de modo a controlar o movimento horizontal e estabilizar as *zero dynamics*. A realimentação completa dos estados é assegurada não só por medições provenientes dos sensores, mas também por estimativas efetuadas por filtros Kalman e filtros de orientação.

Os esquemas de controlo selecionados são implementados num quadricóptero disponível no mercado que possuí uma Unidade de Medida Inercial, um magnetómetro e um altímetro. A posição inercial do drone é obtida através de um sistema de captura de movimento. Os resultados obtidos validam a modelação e as arquitecturas dos sistemas de controlo desenvolvidos.

Palavras-chave: Controlo, *Linear Quadratic Regulator*, *Feedback Linearization*, *Ação Integrativa*, *Seguimento de Trajetória*

Abstract

This dissertation presents full control solutions for a quadrotor using linear and nonlinear methods.

The dynamical model of the quadrotor is derived. The nonlinear controllers are designed considering this model, a linearization of the dynamics is required to synthesize the linear controllers. Linear Quadratic Regulator and Feedback Linearization were the techniques applied to tackle the control problem. Two control structures were devised using the linear approach and three different architectures, exploring the concepts of static and dynamic feedback, were developed for the nonlinear method. The capacity of trajectory tracking, in the presence of measurements noise, and the robustness to significant deviations of the mass and inertia values were evaluated in simulation. Thereby, the most promising linear and nonlinear solutions were selected for implementation in the actual quadrotor.

The selected linear approach consists of an inner-outer loop control structure, where the innermost loop is responsible for the attitude control and the outermost solves the positioning control. In both loops, integrative action is incorporated. The second solution comprises a nonlinear inner loop that results from the application of static feedback linearization to the attitude and altitude dynamics. Linear quadratic controllers with integrative action are implemented not only to the resulting inner-loop chain of integrators, but also to the outer-loop, that controls the horizontal movement and, consequently, stabilizes the zero-dynamics. The required full state-feedback relies on measurements from motion sensors and on on-flight estimates provided by Kalman filters and an attitude filter.

The selected control systems are implemented in a commercially available drone, equipped with an Inertial Measurement Unit, a compass and an altimeter. A motion capture system gives the inertial position of the drone. The results obtained allow the validation of the modeling and control system architectures.

Keywords: Control, Linear Quadratic Regulator, Feedback Linearization, Integrative Action, Trajectory Tracking

Contents

List of Tables

List of Figures

.

Nomenclature

Greek symbols

β	Tuning parameter of the Madgwick attitude filter.
$\boldsymbol{\eta}$	Orientation vector, in terms of Euler angles, of the body-fixed frame $\{B\}$ with respect to the inertial frame $\{I\}$.
$\boldsymbol{\Lambda}$	Decoupling matrix
$\boldsymbol{\nu}$	Sensor noise.
$\boldsymbol{\omega}$	Angular velocity of frame $\{B\}$ relative to frame $\{I\}$ described in the reference frame $\{B\}$.
$\boldsymbol{\Phi}$	Diffeomorphism to obtain the transformed state-variables.
$\boldsymbol{\tau}$	Principal moments applied to the UAV airframe by the aerodynamics of the rotors.
$\boldsymbol{\vartheta}$	Process noise.
$\boldsymbol{\xi}$	Transformed state-variables vector.
$\boldsymbol{\zeta}$	Controller state-variables vector.
Δt	Sample time.
Ω_i	Rotation speed of the rotor i.
ψ	Yaw angle.
τ_ψ	Yaw Moment.
τ_{ψ_i}	Yaw Moment generated by rotor i.
τ_θ	Pitch Moment.
τ_φ	Roll Moment.
θ	Pitch angle.
φ	Roll angle.
ς	Integral of the tracking error over a given time interval.

Roman symbols

$\boldsymbol{b}_{\omega 1}$ Sensor bias of the bias instability noise.

$\boldsymbol{b}_{\omega 2}$ Sensor bias of the rate random walk noise.

$\boldsymbol{d}$ Normalized magnetometer vector after compensating for magnetic distortion.

$\boldsymbol{f}_{g,d}$ Objective fuction aimed to minimize in the Madgwick attitude filter.

$\boldsymbol{I}$ Identity matrix.

$\boldsymbol{J}$ Jacobian.

$\boldsymbol{q}$ Atitude vector expressed through quarternion representation.

$\boldsymbol{T_{BI}}$ Diagonal matrix whose entries correspond to the correlation time of each gyroscope axis bias instability noise.

$\bar{\mathbf{u}}$ Input vector of the extended system.

$\bar{\mathbf{x}}$ State-variables vector of the extended system.

$\mathbf{A}$ State-space matrix relating state derivatives to the states.

$\mathbf{a}$ Acceleration.

$\mathbf{B}$ State-space matrix relating state derivatives to the inputs.

$\mathbf{C}$ State-space matrix relating outputs to the states.

$\mathbf{D}$ State-space matrix relating outputs to the inputs.

$\mathbf{e^x}$ State vector reconstruction stochastic error.

$\mathbf{F_T}$ Principal nonconservative forces applied to the UAV airframe by the aerodynamics of the rotors.

$\mathbf{F}$ Discrete-time state-space matrix relating state derivatives to the states.

$\mathbf{f}$ Nonlinear control independent vector field.

$\mathbf{G}$ State-space matrix relating state derivatives to the inputs.

$\mathbf{g}$ Nonlinear control dependent vector field.

$\mathbf{H}$ State-space matrix relating outputs to the states.

$\mathbf{h}$ Output vector field.

$\mathbf{K^e}$ Steady-state optimal Kalman gain matrix.

$\mathbf{K}$ Steady-state optimal gain matrix.

$\mathbf{p}$ Position vector of the center of mass of the quadrotor.

$\mathbf{Q}$ State weighting matrix.

$\mathbf{ref}$ Reference.

$\mathbf{R}$ Control weighting matrix.

$\mathbf{T}(\boldsymbol{\eta})$ Matrix that transforms the angular velocities in angle rates.

$\mathbf{u}$ Input vector.

$\mathbf{v}$ Transformed input variables.

$\mathbf{x}$ Vector of state-variables.

$\mathbf{y}$ Output vector.

${}^{\mathbf{B}}\mathbf{v}$ Velocity expressed in frame $\{B\}$.

${}^{\mathbf{K}}\mathbf{Q}$ Process noise variance.

${}^{\mathbf{K}}\mathbf{R}$ Sensor noise variance.

$\mathcal{N}(\mu, \vartheta)$ Gaussian distribution with mean μ and covariance $\boldsymbol{\vartheta}$

$\{\vec{a}_1, \vec{a}_2, \vec{a}_3\}$ Unit vectors along the axis of the inertial frame $\{I\}$.

$\{\vec{b}_1, \vec{b}_2, \vec{b}_3\}$ Unit vectors along the axis of the body-fixed frame $\{B\}$.

$\{B\}$ Body-fixed frame.

$\{I\}$ Inertial frame.

e Absolute error.

g Gravity acceleration.

I_i Inertia for i component.

J Performance index.

k_1 Gain for the integrative action.

L Distance of the rotor to the x and y axis.

m Total mass of the quadcopter

M_p Overshoot.

p, q, r Angular velocities of frame $\{B\}$ relative to frame $\{I\}$ described in the reference frame $\{B\}$.

r_j Relative degree of the output $\mathbf{y}_j$.

r_T Sum of the relative degrees.

T Thrust.

T_i Thrust generated by rotor i.

t_s Settling time.

u, v, w Velocities in the body-fixed frame.

x, y, z Cartesian coordinates.

$\boldsymbol{K_{1\eta}}, \boldsymbol{K_{2\eta}}, \boldsymbol{K_{3\eta}}$ Diagonal matrices with the steady-state Kalman gains of the nonlinear attitude filter.

${}^{B}\mathbf{R}_{I}$ Rotation matrix from the inertial frame to the body-fixed frame.

${}^{I}\mathbf{R}_{B}$ Rotation matrix from the body-fixed frame to the inertial frame.

${}^{B}x_{I}$ Inertial coordinate x described with the body-fixed frame orientation.

${}^{B}y_{I}$ Inertial coordinate y described with the body-fixed frame orientation.

c_{τ_i} Coefficient that relates the rotation speed of a rotor and its generated Yaw Moment.

c_i Coefficient that relates the Thrust and Yaw Moment generated by a rotor.

c_{T_i} Coefficient that relates the rotation speed of a rotor and its generated Thrust.

$\mathbf{I}$ Constant inertia matrix.

Subscripts

$\boldsymbol{\psi}$ Yaw Subsystem.

$\boldsymbol{\theta}$ Pitch Subsystem.

$\boldsymbol{\varphi}$ Roll Subsystem.

$\boldsymbol{ref}$ Reference.

$\mathbf{ar}$ Angle rate loop dynamics.

$\mathbf{df}$ Dynamic Feedback Linearization appraoch.

$\mathbf{dl}$ Double-loop approach.

$\mathbf{in}$ Inner-loop dynamics.

$\mathbf{p}$ Position loop dynamics.

$\mathbf{s}$ Single-loop approach.

$\mathbf{x}$ X Inertial Subsystem.

$\mathbf{y}$ Y Inertial Subsystem.

$\mathbf{z}$ Height Subsystem.

i, j Computacional indexes.

k Discrete time instant k, $k \in \mathbb{Z}$.

Superscripts

$\mathbf{o}$ Outer-loop.

B Described in frame $\{B\}$.

e Error.

I Described in frame $\{I\}$.

S Obtained from sensor measumerents.

' Flutuation around a given equilibrium value.

Mathematical Operators

$\dot{s}$ First derivative of s.

$\ddot{s}$ Second derivative of s.

$s^{(3)}$ Third derivative of s.

$s^{(4)}$ Fourth derivative of s.

$\mathbf{S}^T$ Transpose of S.

$\mathbf{S}^{-1}$ Inverse of S.

$\mathcal{L}_{\mathbf{u}}\mathbf{s}$ Lie derivative of $\mathbf{s}$ in the direction of vector field $\mathbf{u}$.

$\nabla\mathbf{s}$ Gradient of $\mathbf{s}$.

$\otimes$ Quaternion product.

diag $(\mathbf{s})$ Diagonal matrix with ith diagonal element given by $\mathbf{s}_i$.

det$(\mathbf{S})$ Determinant of $\mathbf{S}$.

tr$(\mathbf{S})$ Trace of $\mathbf{S}$.

Acronyms

AC Adaptive Control. 11, 12

DFL Dynamic Feedback Linearization. 82

FL Feedback Linearization. 81, 82

FTP File Transfer Protocol. 84

GPIO General Purpose Input/Output. 85

IMU Inertial Measurement Unit. 13, 84

IP Internet Protocol. 84

LQR Linear Quadratic Regulator. 6, 9, 10, 23, 26, 33, 34, 39, 44, 83, 93

MPC Model Predictive Control. 11

PID Proportional Integrative Derivative. 9, 10

PWM Pulse Width Modulation. 4, 5, 59, 75, 84, 85, 87, 92, 103, 104

QTM Qualisys Track Manager. 83, 84

SFL Static Feedback Linearization. 82

SMC Sliding Mode Control. 10, 11

TCP Transmission Control Protocol. 84

UAV Unmanned Aerial Vehicle. 1–5, 10, 13, 14, 16, 20, 21, 33, 60, 83, 85, 86, 95

UDP User Datagram Protocol. 84

Chapter 1

Introduction

In recent years, the popularity of quadrotors has increased immensely as the embedded technology advanced. This growth translated into an intensive study of these Unmanned Aerial Vehicles. Its maneuverability, hovering capabilities, reduced price and small size not only enable the quadrotors to be equated in a wide range of applications, such as infrastructure inspection or area monitoring, but also constitute them as an excellent alternative for the experimentation of control and navigation techniques.

Similar to many other technologies that are currently used by the general population, the concept of the unmanned aerial vehicle started being explored in war context [1]. In World War I, the United States began the experiments but never deployed in combat. Years later, in World War II, Germany evidenced the numerous advantages of considering such vehicles on the battlefield. Military from various countries, like the United States, noticed the potential and started development programs that originated elaborated systems. Numerous ideas were studied and demonstrated in flight with varying degrees of success [2]. However, not only these prototypes presented poor performance [3] but also a skilful pilot was compulsory due to the poor stability and limited control authority [2]. The tremendous technological progress verified in the subsequent decades made possible to build and control unmanned aerial vehicles.

Most recently, due to advances in electronics and manufacturing processes, a miniaturization of the controllers, sensors and processors, without discarding the effectiveness of these components, became a reality. The evolution resulted in the emergence of small configurations of unmanned aerial vehicles. The potential inherent to this reduction in size is enormous and presents several advantages. In 2016, PwC released the report ”Clarity from Above” that stated that the addressable market value of drone-powered solutions is over $127 billion and that the drone revolution is disrupting a broad spectrum of industries [4]. In Fig. 1.1, the estimated value for some key industries is schematically presented.

A relevant part of the research conducted on these small-scale UAVs concerns the quadrotors. The quadrotor is a rotary-wing UAV that, as the name suggests, has four rotors whose arrangement generically resembles either an ”x” or a ”+” configuration. Also commonly referred to as ”drone”, this vehicle takes-off and lands vertically and has its hovering capacity and its high maneuverability as primary characteristics. The symmetry of the design leads to reduced gyroscopic effects [5] and simplicity makes the

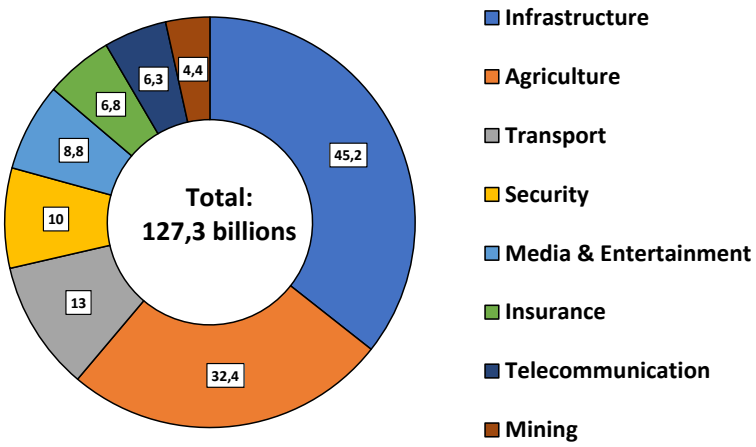

Figure 1.1: Potencial value of UAV-based solutions in key industries for global market (values in billions of dollars).

manufacturing less complicated. Moreover, the vertical take-off and landing capabilities present the opportunity of considering the quadcopter for applications with tighter spatial constraints when compared with fixed-wing UAVs.

The decreasing of the cost of quadrotors, that allows the spanning of their use beyond military applications and academic research, in conjunction with the characteristics formerly described results in equating them in a panoply of public and civil applications, for instance:

- Search and rescue: in situations of natural disasters, the quadcopters can provide crucial support in search and rescue operations and disaster management [6][7]. For instance, resorting to UAVs is possible to cover a large area by image, which is essential in finding missing persons. Moreover, these aerial vehicles can carry medical supplies to regions of difficult access and can provide communication coverage in cases of networks disrupt;

- Remote sensing: drones can be used to obtain data from hazardous or remote locations, perform geological and archaeological surveys [6] or to collect data from ground sensors and deliver to base stations [8];

- Infrastructure Inspection: lately, an interest in the usage of UAVs to monitor large construction projects has emerged [7]. Furthermore, the quadrotors are also often seen as an excellent tool to apply the strategy of condition-based maintenance. Therefore, for inspecting power lines, oil, water, and gas pipelines [9] or wind turbines [10], these vehicles are frequently employed.

- Agriculture: quadrotors can be used for crop management and monitoring [11], spraying pesticides [6] or for irrigation scheduling [7].

- Deliveries: one billion people around the world do not have access to all-season roads [12], which poses severe difficulties in access to medicine and critical supplies. Moreover, this connection problem constitutes a huge barrier to economic growth especially in developing countries since their inhabitants are unable to send the goods produced to markets [13]. To tackle this challenge, several projects using quadrotors have succeeded operationally in recent years. The utilization of

these vehicles ensured rapid and efficient transportation of food, medicine and goods, therefore cutting distances with hospitals, markets and other services. Furthermore, with the breakthrough of e-Commerce, postal and logistics companies have been avidly searching for new forms to distribute packages. Numerous experiments with UAV-based delivery systems have been conducted in countries like Australia, Singapore and Switzerland to assess its feasibility and profitability [14].

- Surveillance: in the scope of public safety, quadcopters with cameras can be used to recognize criminals and detect suspicious human activities, through crowd surveillance [15]. Real-time monitoring of road traffic is also a possibility with drones [7]. Compared with traditional monitoring devices, these aerial vehicles are more cost-effective and versatile, since it is possible to either monitor large continuous road segments or a specific road segment.

The benefits of considering quadrotors are indeed obvious and range from cost-effectiveness and improved speed to safety and efficiency. Furthermore, the quadcopters can substitute the man in hazardous or tedious and repetitive tasks. In the interest of extending the functionalities and abilities of these vehicles, thorough research from around the globe deepens the study of the topic.

1.1 Motivation

Invariably, at the core of the development of applications with quadrotors, a robust control structure is required. Nonetheless, devising controllers for a quadrotor is undoubtedly a challenging problem.

Throughout the years, an intensive study has been conducted by the research community in this topic once the vehicles are characterized for their mechanical simplicity and are fast-prototyping devices, which constitutes them an excellent tool for testing innovative control techniques and designs. In addition to that, the research community acknowledges the control problem inherent to UAVs as a stimulating challenge.

On the one hand, the reduced size of these vehicles, besides the clear advantages addressed formerly, poses serious difficulties. The small sensors used on these systems are much noisier and present more considerable biases when compared with navigation grades equivalents. Additionally, the compact structure results in a higher susceptibility to environmental effects and the reduced scale leads to the vibrations, which are frequent in these flight platforms, impacting the sensors. Therefore, not only the onboard estimation is more challenging, but also the control algorithm must not present significant performance deviations when measurements are noisy. On the other hand, the system is highly nonlinear, multivariable and has coupling between dynamics. Furthermore, its high maneuverability is a consequence of possessing fast dynamics, which increases the complexity of stabilizing it. Moreover, the fact of the drone having six degrees of freedom, position and orientation, and only four actuators, hence being an underactuated system, further intensifies the intricacy of controlling it.

In furtherance of these small aerial vehicles being autonomous, reliable onboard stabilization and trajectory tracking capabilities are imperative. The number and complexity of the applications of these systems are increasing at an impressive rate. Consequently, in order to keep track of this evolution,

the control methods applied must be enhanced aiming better performance and expanded versatility. The first control strategies considered, and still the most common nowadays, are linear and therefore present performance limitations. With the advances in microprocessors and modeling techniques, using real-time nonlinear control methods is an attainable objective and enables the desired improvements in performance and robustness.

In the interest of tackling the control problem, this work proposes and discusses different architectures based on linear and nonlinear control methods. The solutions are tested in simulation to draw comparisons and assess the most promising linear and nonlinear approaches. Subsequently, the selected control structures are experimentally validated.

1.2 Topic Overview

The quadcopter is a small size unmanned aerial vehicle that possesses four rotors disposed of at the end of a structure that resembles an "x". Each rotor is powered by a brushless motor that has its controller. Six variables define the pose of the quadrotor: the inertial coordinates (x, y, z), describing the position of the center of mass in the inertial frame, and the Euler angles (φ, θ, ψ), denoting the rotation of the body-fixed frame with respect to the inertial frame. The actuators receive Pulse Width Modulation (PWM) commands that regulate the rotation speed of the rotor. This rotation generates a thrust force, T_i, and a yaw moment τ_{ψ_i}. In Fig. 1.2, these forces and moments, as well as the Euler angles and the body-fixed frame, are represented. The propellers 1 and 3 rotate clockwise and the propellers 2 and 4 rotate counterclockwise.

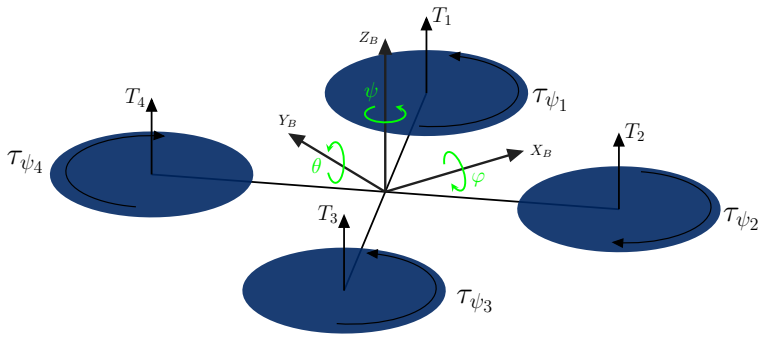

Figure 1.2: Scheme with the forces and moments generated by each rotor and the Euler angles represented.

The position and attitude of the quadcopter are controlled through variations of the rotational speed of the rotors. Increasing the PWM commands provided to the actuator leads to higher thrust and yaw moment values. Thereby, regarding vertical motion, to move upwards, the rotational speed of all the propellers must synchronously increase, and to move downwards, must synchronously decrease. Concerning the horizontal motion, the movement is performed by tilting or rolling the UAV. A positive variation of the roll angle is obtained with higher rotational speeds of the rotors 1 and 4. The motion along the positive direction of the Y_B axis is attained with negative roll values. If the pair formed by the rotors 4

and 3 rotates faster than the pair of rotors 1 and 2, the pitch angle becomes positive and the UAV moves forward. The total sum of the reactive yaw moments produced determines the yaw motion. Similarly to the individual thrust force, this reactive torque is higher with higher rotor speeds.

The control system computes the required total Thrust, T, roll moment, τ_φ, pitch moment, τ_θ, and yaw moment, τ_ψ to follow the references. Subsequently, the thrust forces that each rotor has to generate are determined by solving a system of four equations. This system of equations is often referred to as "motor mixing" and is a result of the movement dynamics being all interconnected. Posteriorly, the determined T_i are transformed in the equivalent PWM commands, through experimentally determined expressions, to send to the controllers of the motors.

Although the system is nonlinear, linear methods, such as the Linear Quadratic Regulator, can be applied to tackle the control problem. To this end, performing a linearization of the system dynamics is required. However, since the controllers are designed for the linearized system, the performance will present limitations. Alternatively, nonlinear control methods can be considered. These approaches and their implementation are indeed more complex than the linear ones. Nonetheless, the application of such methods allows achieving a higher level of control performance. Irrespective of the application of linear or nonlinear control methods, the state variables required for state-feedback that are not directly available through sensors have to be estimated. For this purpose, one can resort, for instance, to the Kalman filter theory.

1.3 Objectives

The ultimate objective of this work is to study and develop a full control solution for a quadrotor. To this end, linear and nonlinear methods are explored and applied.

In furtherance of designing and evaluating the controllers before implementing them in the actual aerial vehicle, the creation of a reliable dynamical model must be targeted. Furthermore, for the application of linear methods, a subsequent linearization of this model is also required. On the other hand, the nonlinear controllers are synthesized without relying on the linearized dynamics, but rather on the complete model. Consequently, a higher level of performance and a larger flight envelope can be achieved. Different control architectures will be explored both with the linear and nonlinear methods. It is aimed to devise control strategies capable of performing a predefined trajectory, presenting null steady-state position error and some level of robustness to model imprecision, uncertainty and disturbances.

The approaches that present the most promising results in the simulation will be implemented on a commercially available quadrotor. In order to estimate the state-variables that are not directly available from the sensory data, for instance, the Euler angles and the linear velocities, a Kalman filter and an attitude filter will be applied. Additionally, the Kalman filter will be used as well for filtering the readings provided by the motion sensors installed onboard. The estimates and the filtered measurements fulfill the requirement for full state-feedback.

1.4 Contributions

The main contributions documented in this thesis are the derivation of a simplified dynamical model for quadrotors with X configuration, the subsequent linearization for a trim position, the application of model-based LQR control with integrative action considering two different structures and the study of three distinct control architectures that rely on the nonlinear technique Feedback Linearization.

The work developed throughout this thesis resulted in the partipation in a robotics competition and in the submission and acceptance of a scientific paper, respectively:

- Madeiras, J., Martins, L., Cardeira, C. and Oliveira, P. (2019) Autonomous UAV Race: An onboard vision-based solution. 'FreeBots' Competion at "Festival Nacional de Robótica 2019" in Gondomar, Porto;
- Martins, L., Cardeira, C. and Oliveira, P. (2019). Linear Quadratic Regulator for Trajectory Tracking of a Quadrotor. Accepted on the 21st IFAC Symposium on Automatic Control in Aerospace.

1.5 Thesis Outline

This document is organized as follows:

Chapter 2: The literature related to the topic of modelling and controlling a quadrotor is reviewed. Linear and nonlinear control methods are explored;

Chapter 3: The aerial vehicle used in this project is presented by briefly describing its the main features and characteristics.. Additionally, the physical model in which the simulation relies on is detailed and the subsequent linearization is deduced;

Chapter 4: The theoretical formulation of the control methods chosen to tackle the control problem is addressed, starting with the Linear Quadratic Regulator with integrative action followed by the Feedback Linearization.

Chapter 5: With the description of the theoretical background of the control methods carried out, in this chapter, the referred techniques are particularized for the dynamics of the quadcopter. Concerning the linear control, two different architectures are developed, and, regarding the Feedback Linearization, three different control structures are designed relying on it.

Chapter 6: The state-estimation required for the full state-feedback is discussed. The fundamental theory of the Kalman filter is presented as well as the essential equations of the two attitude filters resorted. The structure of the filter used to estimate the first and third derivative of the position is also devised.

Chapter 7: Firstly, the gains of the controllers are obtained by analyzing the characteristics of the resulting step-responses. With these gains, the robustness of the proposed control schemes is assessed through the responses obtained in cases of significant deviations in the mass and inertia values. Next, the gains of the Kalman filters used to estimate the velocity and the third derivative of the position are determined. Lastly, with the controllers and estimators derived in this chapter, the trajectory tracking

capacity of the quadrotor is evaluated in simulation. In this test, measurement noise is modeled as Gaussian white noise and is inserted in the model.

Chapter 8: Aspects related to the implementation are addressed and the experimental results obtained with the most promising linear and nonlinear approaches are presented and analyzed. A comparison between the two control architectures is established.

Chapter 9: Lastly, some concluding remarks are drawn and possible future developments are envisioned.

Chapter 2

State of the Art

In the last twenty years, the topic of modeling and control of quadrotors has progressively become more popular. The potential regarding civil applications is unanimously recognized and transformed this area in one of the hottest in robotics. Thereby, the study of techniques to tackle the challenges posed by these aerial vehicles is vast and, consequently, the work produced is very diversified.

For the purpose of modeling the quadrotor dynamics, in the majority of the studies in the literature, the Newton-Euler formalism and the Lagrange equations are considered. Concerning the Newton-Euler approach, the work proposed in [16] derives the rigid-body dynamics of the quadcopter considering a simple model for the thrust force and moment generated by the rotor. Later works include a more accurate representation of the complex helicopter behavior exhibited by quadrotors. In particular, in [17] and [18] the phenomena of *blade flapping* and roll and pitch *rotor damping* are considered in the model of the quadcopter, thus enabling a better understanding of the oscillatory helicopter modes. In [19] a comprehensive study of quadrotor topic is presented, introducing not only the rigid-body dynamics of the airframe and the dominant aerodynamics, deepening the formulation of the induced drag, but also control and estimation methods.

The study of the control of quadrotors includes various linear and nonlinear techniques. Several previous works demonstrated the feasibility of controlling a quadcopter resorting to linear techniques. The bulk of the bibliography reports the application of classical Proportional Integrative Derivative (PID) controllers and modern Linear Quadratic Regulator (LQR) controllers. Bouabdallah et al. stabilized the attitude dynamics using PID control, achieving success in hovering with the presence of small disturbances [20]. In this work, the referred optimal control technique is also applied to the attitude stabilization, enabling establishing a comparison between methods. The latter presented average results, due to model imperfections, and steady-state error, as a consequence of not including an integrative action. On the other hand, the classic technique showed satisfactory behavior. Nonetheless, the authors state that the LQR should have attained better results. In [21], [22] and [23], the classical approach was successfully implemented to control the position and orientation in low speed indoor flights. A simple path following LQR controller was applied in simulation by Cowling et al. in [24]. The control structure leads to accuracy in path following even with the presence of modeled wind and other disturbances. Valenti

et al. proposed a LQR-based control for position and attitude in [25]. An integrative action was included in the position and heading subsystems and the control system is experimentally validated for hovering, obtaining maximum deviations of 10 cm, and waypoint tracking. Bauer et al. implemented a Linear Quadratic servo controller with a double integrator for trajectory tracking in simulation in order to null the steady-state error when the system is subjected to ramp inputs [26]. Lastly, an inner and outer control loop with LQR control with integration action in both loops is proposed by Raja in [27]. The control was designed for a linear model obtained through Jacobian linearization and was successfully implemented in a UAV, enabling an effective attitude response.

Notwithstanding that various works attest the success of tackling the control problem with linear techniques, applying nonlinear control methods that consider a more comprehensive model of vehicle dynamics can lead to better performance. Furthermore, a wider flight envelope is obtained and resorting to such approaches allows coping with challenges associated with parametric uncertainties, aerodynamic disturbances, actuators saturation or measurement errors. In the literature, it is possible to find a variety of nonlinear approaches applied to quadrotors.

The majority of these nonlinear strategies relies on Lyapunov stability criteria [28]. One of these control methods is the Sliding Mode Control. The SMC is widely used and is characterized by its robustness, accuracy and easiness in parameter tuning. The idea of the SMC consists in designing a switching hypersurface defined by the error and its derivatives of the state variables intended to control, such that the selected subspace is stable and its attractor is at the origin [29]. The control law is not continuous since it switches from one continuous control structure to another based on the current position in the state space. Bouabdallah and Siegwart applied this nonlinear approach to stabilize the quadrotor and maintain the roll, pitch and yaw angles to zero [30]. The proposed controller was tested in simulation and experimentally validated, evidencing the ability to stabilize the roll and pitch angles. However, disturbances originated by the chattering effect are present, being more noticeable in the yaw angle response. Xu and Ozguner developed sliding mode controllers with rate bounded PIDs capable of stabilizing and control the position and yaw angle in simulation [31]. A continuous approximation of the sign function is used to prevent chattering and good results are obtained in simulation with and without parametric uncertainties. Xia et al. presented an inner and outer loop control in which SMC is applied for both loops [32]. A comparison with classic derivative control is drawn through simulation and experimental results, with the nonlinear control attaining better results in either tests. Combining the SMC with other control strategies allows achieving better performance and provides additional robustness to unknown but bounded disturbances and modeling uncertainties [33][34].

Another nonlinear method frequently equated to tackle the control problem is Backstepping. This Lyapunov-based technique provides a powerful recursive approach for stabilizing systems that can be represented through nested loops. Thereby, this approach is well-suited for the cascaded structure of quadrotor dynamics. Furthermore, due to its fast convergence, leading to less computational resources required, and its capacity to handle disturbances well, this method is seen as an interesting solution. Nevertheless, some disadvantages are also associated, namely the sensitivity to parameter change and the necessity of full-state measurement. Madani and Benallegue successfully applied the Backstep-

ping approach in simulation to drive a quadrotor to track a desired trajectory and yaw angle [35]. The researchers divided the quadrotor model into three interconnected subsystems. Good performance of the control proposed is noticeable from the simulation results. In [30] the authors proposed and provided experimental results of a Backstepping controller for attitude stabilization. The results obtained demonstrate the aptitude to control the rotational dynamics even in the presence of relatively high perturbations. This approach, when compared with results obtained with SMC, performed favorably. In [1], an integral Backstepping controller is suggested. The developed technique proved to be capable of controlling attitude and altitude dynamics and the integrative action eliminated the steady-state errors. A Backstepping control algorithm for trajectory tracking for a quadrotor is presented in [36]. The proposed controller consists of position control, that tracks the desired trajectory and relies on position and velocity error, and attitude control, that is responsible for controlling not only the angles obtained from the outer loop but also the yaw angle, and that resorts to the rotation matrix and body angular velocity error. The experimental results show satisfactory tracking performance in following a small circular trajectory.

A different method commonly considered that has also attracted research interest throughout the years is Feedback Linearization. This control algorithm algebraically transforms, entirely or partly, a nonlinear system into a linear one resorting to change of coordinates and feedback, thus enabling the application of linear controllers. This approach differs entirely from conventional linearization since feedback linearization results from an exact state transformation and not from linear approximations of the dynamics. Hence, an accurate model is required for implementation. Das et al. in [37] discuss a two-loop approach where, in the inner loop, responsible for altitude and attitude control, feedback linearization was applied and, in the outermost, proportional derivative controllers were used to control the horizontal movement. Additionally, in both loops, Sliding Mode control is included in order to deal more effectively with disturbances. The simulation results validated the approach. In [38], Freddi et al. use feedback linearization to design a double loop control structure capable of performing not only trajectory tracking but also roll and pitch control in the event of a rotor failure. The simulation tests highlight this capacity of the fault tolerant controller proposed. A distinct two-loop architecture using feedback linearization is proposed in [39]. In this work, the attitude of the quadcopter only implicitly appears in the transformation matrix and is not a controlled state. The aerial vehicle proved to fly with good accuracy since the control errors obtained in hovering tests are within 3 cm for all Cartesian coordinates.

Finally, other approaches are often considered to address the challenge of quadrotor control, namely, Model Predictive Control (MPC), Adaptive Control (AC) or H-Infinity ($\mathcal{H}_\infty$). Within these methods, the MPC attempts to compute an admissible piecewise continuous control input, that ensures the system behaves like a reference model without violating the given state and input constraints [40], through optimization techniques that take into account the current and the future time-steps. Since MPC explicitly considers the operating constraints, it can operate closer to hard constraint boundaries than traditional control schemes. Furthermore, this control method can anticipate future responses of the plant and, therefore, can take control actions accordingly. The drawbacks of this approach are mainly associated with the requirement of an highly-accurate dynamic model and the possible calculation load of the optimization process. In [41], Alexis et al. suggested an algorithm using MPC to control the attitude of

a quadrotor and experimentally validated it in an environment with atmospheric disturbances [41]. The application of the AC algorithms aim the cases where the parameters of the system intended to control vary or are initially uncertain and consists in adapting the control law to this variation or incertitude. This approach, when applied to quadcopter control, allows attenuating the impact of unmodelled dynamics and unstructured disturbances. In [42], an adaptive position tracking controller achieving global asymptotic stability is proposed for a quadrotor in the presence of external disturbances. Pérez-Alcocer et al. introduced an adaptive controller for quadrotor position and orientation trajectory tracking in the presence of parametric inaccuracies, achieving better experimental results than the non-adaptive version of the referred controller [43]. $\mathcal{H}_\infty$ methods are applied to synthesize controllers through a mathematical min-max optimization problem. To that effect, a good model of the system to be controlled is required. Raffo et al. developed a hierarchical control scheme consisting of a model predictive controller to track a reference trajectory herewith a nonlinear $\mathcal{H}_\infty$ controller to stabilize the rotational dynamics of the quadcopter [44]. In both controllers, integrative action is included, allowing the achievement of a null steady-state error when sustained disturbances are acting on the system. The results obtained with the simulation model, which included the presence of aerodynamic disturbances and parametric and structural uncertainties, corroborate the effectiveness and robustness of the control structure.

Chapter 3

Aerial Vehicle

The Parrot AR. Drone 2.0 is the unmanned aerial vehicle used in this project. It is imperative to have knowledge about the drone and its dynamics prior to the beginning of the discussion about the control strategies. Consequently, in this section, the UAV and its actuators and sensors are briefly described, the dynamic model necessary to create the simulation is detailed and the linearized model will be deduced.

3.1 Overview

The Parrot AR. Drone 2.0 [45] is a quadrotor whose rotors arrangement resembles as "X". This drone was projected by the Parrot SA company aiming the mass market of videos games and home entertainment. The vehicle includes a built-in controller for its position and yaw angle, enabling an automatic static flight up to 50 meters high and an easiness in controlling by Wi-Fi, using a tablet or a smartphone, and ensuring stability.

3.2 Actuators and Sensors

The thrust generation is carried out by brushless motors that are individually controlled by its ATMEGA8L 8-bit controller and that have each one its exclusive cutout system to turn off the engine in case of blockage of the propellers. These actuators are controlled through Pulse Width Modulation commands that indicate the percentage of full speed of the rotor (0% being no speed and 100% being full speed) and that are converted to a 40-bit number. The arrangement of the rotors is depicted in Fig. 1.2. The rotors 1 and 3 rotate clockwise and the rotors 2 and 4 rotate counterclockwise.

The quadrotor possesses an inboard Inertial Measurement Unit, two cameras and a sonar board [46]. The IMU runs at a rate of 200 Hz and is constituted by a Bosch BMA150 3-axis accelerometer, an Invensense IMU-3000 3-axis gyroscope and a BMP180 barometric pressure sensor. The sonar board is equipped with two Kobitone ultrasonic transducers (one 400SR (receiver) and one 400ST (transmitter)) and an AKM Semiconductor 3-axis Compass. These sensors provide measurements of the acceleration minus the gravity, of the angular velocities, of the height of the UAV and of the magnetic field intensity.

The vertical QVGA (320x240) camera is used for ground speed measurement and video streams at 60 frames per second, while the frontal camera serves the purpose of capturing images and video with a resolution of 720p at 30 frames per second with a wide angle lens, enabling the implementation of tracking and identification algorithms.

3.3 Nonlinear Model

In this section, the nonlinear model of the UAV based on the Newton-Euler formalism is presented. The nonlinear dynamics are described in the body-fixed $\{B\}$ and in the inertial $\{I\}$ frames, depicted in Fig. 1. The unit vectors along the axis of the body-fixed frame are denoted by $\{\vec{b}_1, \vec{b}_2, \vec{b}_3\}$ and the unit vector along the inertial frame $\{I\}$ axis are denoted by $\{\vec{a}_1, \vec{a}_2, \vec{a}_3\}$. It is assumed that the origin of the body-fixed frame $\{B\}$ is coincident with the center of mass of the quadrotor.

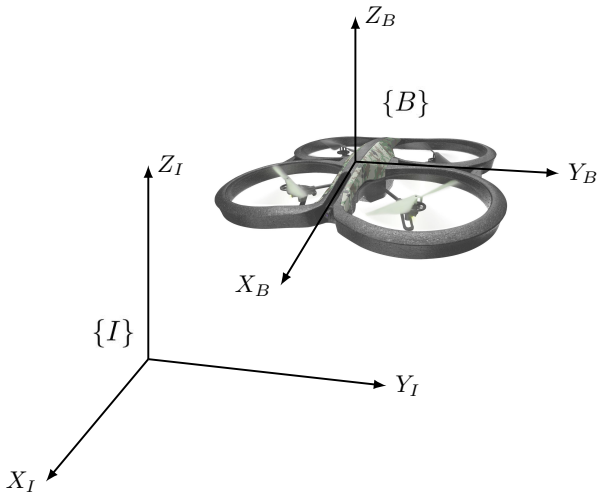

Figure 3.1: Reference frames in which the nonlinear dynamics are described.

Let $\mathbf{p} = [x \;\; y \;\; z]^T$ denote the position vector of the center of mass of the UAV in the inertial frame. Let $\boldsymbol{\eta} = [\varphi \;\; \theta \;\; \psi]^T$ describe the orientation vector, in terms of Euler angles, of the body-fixed frame with respect to the inertial frame, where φ, θ and ψ are the roll, pitch and yaw angles, respectively. Let $\boldsymbol{\omega} = [p \;\; q \;\; r]^T$ represent the angular velocity of frame $\{B\}$ relative to frame $\{I\}$ described in the reference frame $\{B\}$. The rigid body equations of motion of the quadcopter according to [19] are given by:

$$m\ddot{\mathbf{p}} = \; -mg\vec{a}_3 + {}^I\mathbf{R}_B\mathbf{F_T} \tag{3.1}$$

$$\mathbf{I}\dot{\boldsymbol{\omega}} = \; -\boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega} + \boldsymbol{\tau} \tag{3.2}$$

where $\mathbf{I}$ corresponds to the 3×3 constant inertia matrix described in the body fixed-frame, m is the total mass of the quadrotor, g denotes the gravity acceleration, $\mathbf{F_T}$ and $\boldsymbol{\tau}$ denote, respectively, the principal nonconservative forces and moments applied to the UAV airframe by the aerodynamics of the rotors, both described in the reference frame $\{B\}$, and ${}^I\mathbf{R}_B$ is the rotation matrix from the body-fixed to the

inertial frame. The Euler angles follow the sequence of rotation Z-Y-X that is described in [47]. The resultant rotation matrix is given by:

$$
{}^{I}\mathbf{R}_{B} = \begin{bmatrix} \cos(\theta)\cos(\psi) & \sin(\varphi)\sin(\theta)\cos(\psi) - \cos(\varphi)\sin(\psi) & \cos(\varphi)\sin(\theta)\cos(\psi) + \sin(\varphi)\sin(\psi) \\ \cos(\theta)\sin(\psi) & \sin(\varphi)\sin(\theta)\sin(\psi) - \cos(\varphi)\cos(\psi) & \cos(\varphi)\sin(\theta)\sin(\psi) - \sin(\varphi)\cos(\psi) \\ -\sin(\theta) & \sin(\varphi)\cos(\theta) & \cos(\varphi)\cos(\theta) \end{bmatrix} \tag{3.3}
$$

The angle rates $\dot{\boldsymbol{\eta}} = [\dot{\varphi} \;\; \dot{\theta} \;\; \dot{\psi}]^T$ are obtained from the body rotational rates using the following system of equations:

$$
\begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \mathbf{T}(\boldsymbol{\eta})\,\boldsymbol{\omega} = \begin{bmatrix} 1 & \sin(\varphi)\tan(\theta) & \cos(\varphi)\tan(\theta) \\ 0 & \cos(\varphi) & -\sin(\varphi) \\ 0 & \sin(\varphi)\sec(\theta) & \cos(\varphi)\sec(\theta) \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{3.4}
$$

According to [3], the steady-state thrust T_i and yaw moment τ_{ψ_i} generated by a rotor in free air can be modeled as follow:

$$
T_i = c_{T_i}\Omega_i^2 \tag{3.5}
$$

$$
\tau_{\psi_i} = c_{\tau_i}\Omega_i^2 \tag{3.6}
$$

where c_{T_i} and c_{τ_i} are coefficients possible to determine experimentally that are dependent on the area of the disk, the radius of the rotor, the density of air, the geometry and the profile of the rotor, and the effect of drag by the rotor flow, and Ω_i is the rotation speed of the rotor i. Therefore, the relation between the generated yaw moment τ_{ψ_i} by a rotor and its generated thrust T_i is described by the following expression:

$$
\tau_{\psi_i} = \frac{c_{\tau_i}}{c_{T_i}} T_i = c_i T_i \tag{3.7}
$$

The roll and pitch moments, τ_φ and τ_θ, result from the generated thrust of the rotor and its arrangement relative to the center of mass of the quadcopter. Hence, the resultant total thrust T and moments τ_φ, τ_θ and τ_ψ, for a quadrotor with a *X-configuration*, are computed through:

$$
\begin{bmatrix} T \\ \tau_\varphi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ L & -L & -L & L \\ -L & -L & L & L \\ c_1 & -c_2 & c_3 & -c_4 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \end{bmatrix} \tag{3.8}
$$

where L denotes the perpendicular distance of the rotor to the x or y axis of the body-fixed frame, as the case may be.

There are various aerodynamic and gyroscopic effects associated with the rotorcraft that increase the complexity of the model. However, a model with such level of precision is not required, not only because the control can overcome these secondary effects [19], but also since is widely shown in the literature that the control can achieve high performance with the simplified model of the rotor. Consequently, high

order effects such as *blade flapping*, *aerodynamic drag*, *translational lift* and *vortex states* caused by axial motion are neglected. Furthermore, it is assumed that the total thrust generated T is oriented along the $\vec{b}_3$ direction, i.e., parallel to the axis of the rotor, and the coefficients of the rotor are constant (static thrust and moments).

Moreover, the quadrotor is assumed to be exactly symmetrical, which implies that the inertia matrix is diagonal, and the rotor gyroscopic effects are neglected. Additionally, since the UAV flies at a height higher than 0.30 meters, except when the take-off or the landing occurs, the ground effect is ignored [48].

Considering the equations previously defined and the approximations and assumptions described, let

$$\mathbf{x} = [x \;\; y \;\; z \;\; \varphi \;\; \theta \;\; \psi \;\; \dot{x} \;\; \dot{y} \;\; \dot{z} \;\; p \;\; q \;\; r]^T \tag{3.9}$$

denote the vector of state-variables and let

$$\mathbf{u} = [T \;\; \tau_\varphi \;\; \tau_\theta \;\; \tau_\psi]^T \tag{3.10}$$

represent the input vector, the quadcopter dynamics can be written in the compact form

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\,\mathbf{u} \tag{3.11}$$

where

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ p + q\sin(\varphi)\,\tan(\theta) + r\cos(\varphi)\tan(\theta) \\ q\cos(\varphi) - r\sin(\varphi) \\ q\sin(\varphi)\sec(\theta) + r\cos(\varphi)\sec(\theta) \\ 0 \\ 0 \\ -g \\ \frac{I_y - I_z}{I_x} qr \\ \frac{I_z - I_x}{I_y} pr \\ \frac{I_x - I_y}{I_z} pq \end{bmatrix} \tag{3.12}$$

and

$$\mathbf{g}(\mathbf{x}) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{\cos(\varphi)\sin(\theta)\cos(\psi)+\sin(\varphi)\sin(\psi)}{m} & 0 & 0 & 0 \\ \frac{\cos(\varphi)\sin(\theta)\sin(\psi)-\sin(\varphi)\cos(\psi)}{m} & 0 & 0 & 0 \\ \frac{\cos(\varphi)\cos(\theta)}{m} & 0 & 0 & 0 \\ 0 & \frac{1}{I_x} & 0 & 0 \\ 0 & 0 & \frac{1}{I_y} & 0 \\ 0 & 0 & 0 & \frac{1}{I_z} \end{bmatrix} \tag{3.13}$$

3.4 Linear Model

With the nonlinear model detailed, the next step consists in linearize the plant in furtherance of enabling the implementation of linear control techniques.

The point of equilibrium for which the linearization will be deduced is the hover condition ($\mathbf{p} = [x \ y \ z \ ]^T$, $\boldsymbol{\eta} = [0 \ 0 \ 0]^T$), where the yaw angle is additionally considered zero. Note that the linearization could be performed for other conditions, however, this one was chosen given its simplicity. By resorting the Taylor series till the first order term, the following approximations are achieved:

$$\cos(\varphi) \simeq \cos(\theta) \simeq \cos(\psi) \simeq 1 \tag{3.14}$$

$$\sin(\varphi) \simeq \tan(\varphi) \simeq \varphi \tag{3.15}$$

$$\sin(\theta) \simeq \tan(\theta) \simeq \theta \tag{3.16}$$

$$\sin(\psi) \simeq \tan(\psi) \simeq \psi \tag{3.17}$$

Furthermore, for the equilibrium point referred, the Euler angles can be represented by

$$\varphi = \bar{\varphi} + \varphi' = \varphi' \tag{3.18}$$

$$\theta = \bar{\theta} + \theta' = \theta' \tag{3.19}$$

$$\psi = \bar{\psi} + \psi' = \psi' \tag{3.20}$$

where φ', θ' and ψ' correspond to fluctuations around the equilibrium values $\bar{\varphi}$, $\bar{\theta}$ and $\bar{\psi}$, respectively.

3.4.1 Height Subsystem

For the *height subsystem*, the application of the former equations to the differential equation yields:

$$\ddot{z} = \cos(\varphi)\cos(\theta)\frac{T}{m} - g \simeq \frac{1}{m}(T - mg) \tag{3.21}$$

Defining the following state variables and modified input of the subsystem,

$$\mathbf{x_z} = [z \;\; \dot{z}]^T, \;\; \mathbf{u_z} = T - mg \tag{3.22}$$

leads to the state-space representation described below:

$$\dot{\mathbf{x}}_\mathbf{z} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \mathbf{x_z} + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} \mathbf{u_z} \tag{3.23}$$

$$\mathbf{y_z} = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x_z} \tag{3.24}$$

3.4.2 X Inertial Subsystem

In fact, the variation of the quadcopter position along the direction of $\vec{b}_1$ is a direct result of the pitch angle variation. With this in mind, it becomes evident that the linearization of the equations that define the inertial coordinate x described with the body-fixed frame orientation, ${}^B x_I$, has the linearization carried out for the equations that relate the pitch moment, τ_θ, with the pitch angle, θ, as an integral part. Thus, firstly, the linear state-space representation for the pitch angle subsystem is obtained and, subsequently, the linear state-space representation for the position coordinate ${}^B x_I$ is attained.

Pitch Subsystem

The pitch rate is given by

$$\dot{\theta} = \cos(\varphi)\, q - \sin(\varphi)\, r \tag{3.25}$$

differentiating in respect to time originates

$$\ddot{\theta} = -\dot{\varphi}\sin(\varphi)\, q + \cos(\varphi)\, \dot{q} - \dot{\varphi}\cos(\varphi)\, r - \sin(\varphi)\, \dot{r} \tag{3.26}$$

By applying the approximations aforementioned, the former equation takes the form

$$\ddot{\theta} \simeq -\dot{\varphi}'\varphi' + \dot{q} - \dot{\varphi}' r - \varphi' \dot{r} \tag{3.27}$$

Considering small fluctuations, the product between two of them can be neglected, yielding

$$\ddot{\theta} \simeq \dot{q} - \dot{\varphi}' r - \varphi' \dot{r} \tag{3.28}$$

Inverting the relation expressed in (3.4) gives

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin(\theta) \\ 0 & \cos(\varphi) & \sin(\varphi)\cos(\theta) \\ 0 & -\sin(\varphi) & \cos(\varphi)\cos(\theta) \end{bmatrix} \begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{3.29}$$

By observation of the above equation, it is easy to conclude that, for the conditions established, any product between body angle rates and Euler angles or Euler rates will be zero since all the resulting terms will have a product of fluctuations. Replacing the term of (3.28) by the equation of interest present in (3.11) results

$$\ddot{\theta} \simeq \frac{I_z - I_x}{I_y} pr + \frac{\tau_\theta}{I_y} \simeq \frac{\tau_\theta}{I_y} \tag{3.30}$$

Defining the consequent state variables and modified input of the subsystem,

$$\mathbf{x_\theta} = [\theta \;\; \dot{\theta}]^T, \;\; \mathbf{u_\theta} = \tau_\theta \tag{3.31}$$

results in the state-space representation described below:

$$\dot{\mathbf{x}}_{\boldsymbol{\theta}} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \mathbf{x_\theta} + \begin{bmatrix} 0 \\ \frac{1}{I_y} \end{bmatrix} \mathbf{u_\theta} \tag{3.32}$$

$$\mathbf{y_\theta} = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x_\theta} \tag{3.33}$$

X Inertial State-Space Representation

The acceleration in the body-fixed frame vector, ${}^B\mathbf{a}$, is given by

$${}^B\mathbf{a} = \frac{\mathbf{F}}{m} - {}^B\mathbf{R}_I g \overrightarrow{a_3} - \boldsymbol{\omega} \times {}^B\mathbf{v} = \begin{bmatrix} \sin(\theta)\, g + rv - qw \\ -sin(\varphi)\, cos(\theta)\, g + pw - ru \\ -cos(\varphi)\cos(\theta)\, g + \frac{T}{m} + qu - pv \end{bmatrix} \tag{3.34}$$

where ${}^B\mathbf{v} = [u \;\; v \;\; w]^T$ denotes the velocities in the body-fixed frame and the term $\boldsymbol{\omega} \times {}^B\mathbf{v}$ corresponds to the centripetal acceleration. Considering the x component of this vector and applying the relations described in (3.29) and the approximations listed in the outset of this section, the acceleration along the direction of $\vec{b}_1$, ${}^B\mathbf{a}_x$, is equal to

$${}^B\mathbf{a}_x \simeq \theta g - \dot{\theta} w + \dot{\psi} v \tag{3.35}$$

Assuming that the movement along the direction of $\vec{b}_1$ occurs with negligible variations along the direction of $\vec{b}_3$ and with a constant yaw angle, the previous equations simplify into:

$$ {}^{B}\mathbf{a}_x \simeq \theta g \tag{3.36} $$

Noticing that the forward velocity of the UAV in the body-fixed frame, u, is the integral of ${}^{B}\mathbf{a}_x$, equation (3.36) can be rewritten as

$$ \dot{u} \simeq \theta g \tag{3.37} $$

The equation obtained relates the forward velocity, u, with the pitch angle, θ. Thereby, with the establishment of the next state-variables and modified input of the subsystem:

$$ \mathbf{x_x} = [{}^{B}x_I \;\; u \;\; \theta \;\; \dot{\theta}]^T, \;\; \mathbf{u_x} = \tau_\theta \tag{3.38} $$

the following state-space representation is attained

$$ \dot{\mathbf{x}}_\mathbf{x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & g & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{x_x} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{I_y} \end{bmatrix} \mathbf{u_x} \tag{3.39} $$

$$ \mathbf{y_x} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \mathbf{x_x} \tag{3.40} $$

3.4.3 Y Inertial Subsystem

Analogously to the subsection 3.4.2, the linearization that will result in the state-space representation for the y coordinate follows a similar logic by firstly performing the linearization of second derivative of the roll angle and, posteriorly, the linearization of the y component of the linear acceleration described in the body-fixed frame, ${}^{B}\mathbf{a}_y$.

Roll Subsystem

The roll rate is expressed by

$$ \dot{\varphi} = p + \sin(\varphi)\tan(\theta)\, q + \cos(\varphi)\tan(\theta)\, r \tag{3.41} $$

Differentiating in respect to time followed by the application of the same simplifications taking into account for the pitch subsystem yields

$$ \ddot{\varphi} \simeq \dot{p} + \dot{\theta} r + \theta \dot{r} \tag{3.42} $$

Once again, considering the assumption that the product of Euler rates and angular velocities origins products of fluctuations that can be neglected, equation (3.42) simplifies into

$$\ddot{\varphi} \simeq \frac{\tau_\varphi}{I_x} \tag{3.43}$$

Letting the following equations define the state variables and modified input of the subsystem

$$\mathbf{x_\varphi} = [\varphi \;\; \dot{\varphi}]^T, \;\; \mathbf{u_\varphi} = \tau_\varphi \tag{3.44}$$

the obtained state-space representation is described by

$$\dot{\mathbf{x}}_{\boldsymbol{\varphi}} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \mathbf{x}_{\boldsymbol{\varphi}} + \begin{bmatrix} 0 \\ \frac{1}{I_x} \end{bmatrix} \mathbf{u}_{\boldsymbol{\varphi}} \tag{3.45}$$

$$\mathbf{y}_{\boldsymbol{\varphi}} = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x}_{\boldsymbol{\varphi}} \tag{3.46}$$

Y Inertial State-Space Representation

Recalling (3.34), the y component of the acceleration in the body-fixed frame, after applying the same approximations considered in the case of ${}^B\mathbf{a}_x$, is given by

$${}^B\mathbf{a}_y \simeq -\varphi g - \dot{\varphi} w + \dot{\psi} u \tag{3.47}$$

Assuming that the movement along the direction of $\vec{b}_2$ occurs with negligible variations along the direction of $\vec{b}_3$ and with a constant yaw angle, the previous equations simplify into:

$${}^B\mathbf{a}_y \simeq -\varphi g \tag{3.48}$$

Noting that the sideway velocity of the UAV in the body-fixed frame, v, is the integral of ${}^B\mathbf{a}_y$, (3.48) can be rewritten as

$$\dot{v} \simeq -\varphi g \tag{3.49}$$

The equation obtained relates the sideway velocity, v, with the roll angle, φ. Thus, by defining the following state-variables and modified input of the subsystem:

$$\mathbf{x_y} = [{}^B y_I \;\; v \;\; \varphi \;\; \dot{\varphi}]^T, \;\; \mathbf{u_y} = \tau_\varphi \tag{3.50}$$

where ${}^B y_I$ denotes the y coordinate of the inertial position described according the orientation of the body-fixed frame, the following state-space representation is obtained

$$\dot{\mathbf{x}}_{\mathbf{y}} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -g & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{x}_{\mathbf{y}} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{I_x} \end{bmatrix} \mathbf{u}_{\mathbf{y}} \tag{3.51}$$

$$\mathbf{y_y} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \mathbf{x_y} \tag{3.52}$$

3.4.4 Yaw Subsystem

The yaw rate is described by

$$\dot{\psi} = \sin(\varphi)\sec(\theta)\, q + \cos(\varphi)\sec(\theta)\, r \tag{3.53}$$

Taking the time derivative and carrying out the simplifications considered in the previous angular subsystems yields

$$\ddot{\psi} \simeq \dot{r} + \dot{q}\varphi \tag{3.54}$$

Following an analogous logic to the presented in the sections 3.4.2 and 3.4.3 enables to linearize the second derivative of the yaw angle:

$$\ddot{\psi} \simeq \frac{\tau_\psi}{I_z} \tag{3.55}$$

Letting the state-variables and entry of the subsystem be described by

$$\mathbf{x_\psi} = [\psi \;\; \dot{\psi}]^T, \;\; \mathbf{u_\psi} = \tau_\psi \tag{3.56}$$

enables the following state-space representation

$$\dot{\mathbf{x}}_{\boldsymbol{\psi}} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \mathbf{x}_{\boldsymbol{\psi}} + \begin{bmatrix} 0 \\ \frac{1}{I_z} \end{bmatrix} \mathbf{u}_{\boldsymbol{\psi}} \tag{3.57}$$

$$\mathbf{y}_{\boldsymbol{\psi}} = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x}_{\boldsymbol{\psi}} \tag{3.58}$$

Chapter 4

Control Theory

In this chapter, the theoretical basis of the control strategies considered is addressed. The chapter starts with the presentation of a linear modern control technique, Linear Quadratic Regulator, with the inclusion of integrative action. Next, the concept of Feedback Linearization Control is enunciated, where both the static and dynamic feedback control laws are discussed in detail.

4.1 Linear Quadratic Regulator with Integrative Action

The first control technique chose to tackle the problem formulated is the Linear-Quadratic Regulator (LQR) with Integrative Action. The theory behind the LQR presented in this subsection is based on the formulation in [49]. The LQR is an optimal controller that uses full state-feedback and that is obtained as the solution of an optimization process where the system dynamics imposes the restrictions. The computation of the gains for the state-feedback consists in finding a gain matrix that minimizes a cost function, that relates the weighting of the control effort with the deviation of the states from the origin, without neglecting the stability of the closed-loop. Therefore, the resulting poles are a consequence of the optimization process.

This optimal control technique is a very appealing approach since it easily handles multiple actuators and complex system dynamics. Moreover, LQR offers very large stability margins to errors in the loop gain: gain margin of infinity, gain reduction margin of 1/2 and a minimum phase margin of 60° in each control input channel [50].

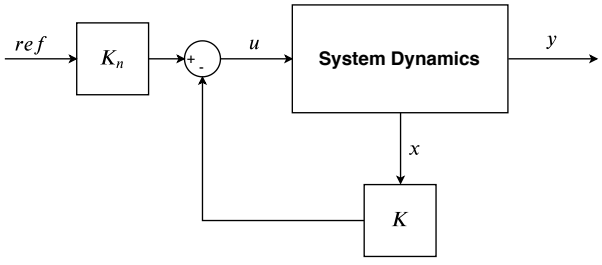

Figure 4.1: Structure of the Linear Quadratic Regulator.

For a system described by the state-space representation

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \tag{4.1}$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} \tag{4.2}$$

the optimal regulator problem determines the steady-state gain matrix $\mathbf{K}$ that is part of the optimal control vector

$$\mathbf{u} = -\mathbf{K}\mathbf{x} \tag{4.3}$$

and that ensures the minimization of the performance index

$$J = \int_0^\infty \left(\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{u}^T\mathbf{R}\mathbf{u}\right) dt \tag{4.4}$$

where the first quadratic form includes the real symmetric $(\mathrm{n} \times \mathrm{n})$ matrix $\mathbf{Q}$ (state weighting matrix), that enables the embodiment of a penalization on the deviation of the state $\mathbf{x}$ from the desired state, and the second quadratic term accounts for the energetic effort of the control signals by attempting to limit its magnitude resorting to the real symmetric $(\mathrm{m} \times \mathrm{m})$ matrix $\mathbf{R}$ (control weighting matrix). Since this is an infinite time control problem, the performance index J has no time limit and, additionally, as time tends to infinity the control solution turns into a steady-state solution and the gain matrix $\mathbf{K}$ becomes constant.

Substituting (4.3) into (4.1) and into (4.4), the system and the cost function can be described respectively, by:

$$\dot{\mathbf{x}} = \left(\mathbf{A} - \mathbf{B}\mathbf{K}\right)\mathbf{x} \tag{4.5}$$

$$J = \int_0^\infty \left(\mathbf{x}^T\left(\mathbf{Q} + \mathbf{K}^T\mathbf{R}\mathbf{K}\right)\mathbf{x}\right) dt \tag{4.6}$$

Setting

$$\mathbf{x}^T\left(\mathbf{Q} + \mathbf{K}^T\mathbf{R}\mathbf{K}\right)\mathbf{x} = -\frac{d}{dt}\left(\mathbf{x}^T\mathbf{P}\mathbf{x}\right) \tag{4.7}$$

with $\mathbf{P}$ being a real symmetric matrix, resolving the time derivative and considering (4.5) yields

$$\mathbf{x}^T\left(\mathbf{Q} + \mathbf{K}^T\mathbf{R}\mathbf{K}\right)\mathbf{x} = -\mathbf{x}^T\left(\left(\mathbf{A} - \mathbf{B}\mathbf{K}\right)^T\mathbf{P} + \mathbf{P}\left(\mathbf{A} - \mathbf{B}\mathbf{K}\right)\right)\mathbf{x} \tag{4.8}$$

The previous equation must hold true for any state $\mathbf{x}$, therefore, it is required that

$$-\left(\mathbf{Q} + \mathbf{K}^T\mathbf{R}\mathbf{K}\right) = \left(\mathbf{A} - \mathbf{B}\mathbf{K}\right)^T\mathbf{P} + \mathbf{P}\left(\mathbf{A} - \mathbf{B}\mathbf{K}\right) \tag{4.9}$$

It is provable that if $\mathbf{A} - \mathbf{B}\mathbf{K}$ is a stable matrix then a positive-definite matrix $\mathbf{P}$ that verifies the last

equation exists. It is important to stress that more than one matrix $\mathbf{P}$ constitute a possible solution, however, exists only one that is positive-definite if the system is stable.

On account of assuming $\mathbf{R}$ to be a real symmetric matrix, it is possible to write it as

$$\mathbf{R} = \mathbf{T}^T\mathbf{T} \tag{4.10}$$

in which $\mathbf{T}$ is a nonsingular matrix. As result, (4.9) gains the following form

$$\left(\mathbf{A}^T - \mathbf{K}^T\mathbf{B}^T\right)^T \mathbf{P} + \mathbf{P}\left(\mathbf{A} - \mathbf{B}\mathbf{K}\right) + \mathbf{Q} + \mathbf{K}^T\mathbf{T}^T\mathbf{T}\mathbf{K} = \mathbf{0} \tag{4.11}$$

By rearranging the terms, the above equation becomes

$$\mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} + \left(\mathbf{T}\mathbf{K} - \left(\mathbf{T}^T\right)^{-1}\mathbf{B}^T\mathbf{P}\right)^T \left(\mathbf{T}\mathbf{K} - \left(\mathbf{T}^T\right)^{-1}\mathbf{B}^T\mathbf{P}\right) - \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q} = \mathbf{0} \tag{4.12}$$

The minimization of the cost function $\mathbf{J}$ with respect to $\mathbf{K}$ occurs when

$$\mathbf{x}^T \left(\mathbf{T}\mathbf{K} - \left(\mathbf{T}^T\right)^{-1}\mathbf{B}^T\mathbf{P}\right)^T \left(\mathbf{T}\mathbf{K} - \left(\mathbf{T}^T\right)^{-1}\mathbf{B}^T\mathbf{P}\right)\mathbf{x} = \mathbf{0} \tag{4.13}$$

which originates

$$\mathbf{K} = \mathbf{T}^{-1}\left(\mathbf{T}^T\right)^{-1}\mathbf{B}^T\mathbf{P} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} \tag{4.14}$$

This equation gives the optimal gain matrix $\mathbf{K}$ that minimizes the cost function J. The matrix $\mathbf{P}$ present in (4.14) is given by the reduced-matrix *Riccati* equation:

$$\mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} - \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q} = \mathbf{0} \tag{4.15}$$

Due to the impracticability inherent in forecasting the effect of a given pair of weighting matrices $\mathbf{Q}$ and $\mathbf{R}$, the approach to follow, in order to find the gain matrix $\mathbf{K}$ that results in success regarding the control objectives predefined, consists in simulate the corresponding closed-loop response for a range of matrices $\mathbf{Q}$ and $\mathbf{R}$. Nonetheless, the $\mathbf{Q}$ and $\mathbf{R}$ matrices can be initially chosen based on the Bryson's rule [51]:

$$\mathbf{Q} = \mathrm{diag}\left(\frac{1}{x_{1_{\max}}^2}, ..., \frac{1}{x_{n_{\max}}^2}\right) \tag{4.16}$$

$$\mathbf{R} = \mathrm{diag}\left(\frac{1}{u_{1_{\max}}^2}, ..., \frac{1}{u_{m_{\max}}^2}\right) \tag{4.17}$$

where the $x_{i_{\max}}^2$ and $u_{i_{\max}}^2$ represent the largest desired response or control input for that component of the state/actuator signal.

The chosen gain matrix must be the one that origins a higher level of achievement of the closed-loop response requirements. It is important to stress that is absolutely indispensable the system being

controllable to apply the LQR control. Furthermore, the controllability is a sufficient condition for the closed-loop to be stabilizable.

As a way of dealing with the effect of perturbations and with the steady-state error, an integrator was embedded in the control structure. This inclusion translates into additional robustness of the control system, since it ensures a sufficiently high low-frequency gain in the loop transfer function [50], and eliminates the steady-state errors due to constant disturbances or reference input commands. In other words, the integral term can be perceived as constantly calculating the required value of the control at the trim point for the purpose of regulating the error [52]. The structure of the Linear-Quadratic Regulator with Integrative action is depicted in Fig. 4.2.

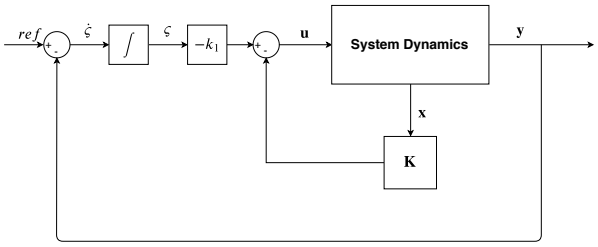

Figure 4.2: Structure of the Linear Quadratic Regulator with Integrative Action.

To project the LQR control for this system with the integrator included is necessary to modify the dynamical model considered in the computation of the optimal gains. Letting the reference signal be represented by $\mathbf{ref}$ and the difference between the output of the system, $\mathbf{y}$, and the reference, $\mathbf{ref}$, be

$$\dot{\boldsymbol{\varsigma}} = \mathbf{ref} - \mathbf{y} = \mathbf{ref} - \mathbf{C}\mathbf{x} \tag{4.18}$$

where $\boldsymbol{\varsigma}$ is the state-space variable that results from adding the referred integrator. The control vector $\mathbf{u}$ of the servo system is defined by:

$$\mathbf{u} = -\mathbf{K}\mathbf{x} + \mathbf{k}_1\boldsymbol{\varsigma} \tag{4.19}$$

Combining the last two equations with (4.1) and (4.2), the dynamics of the system can be described by

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\boldsymbol{\varsigma}} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ -\mathbf{C} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \boldsymbol{\varsigma} \end{bmatrix} + \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix} \mathbf{u} + \begin{bmatrix} \mathbf{0} \\ \mathbf{1} \end{bmatrix} \mathbf{ref} \tag{4.20}$$

The use of the matrices of the modified model

$$\bar{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ -\mathbf{C} & \mathbf{0} \end{bmatrix} \tag{4.21}$$

$$\bar{\mathbf{B}} = \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix} \tag{4.22}$$

enables the unequivocal application of the LQR gain computation presented formerly while considering the integrative action. The optimal gain matrix obtained

$$\bar{\mathbf{K}} = \begin{bmatrix} \mathbf{K} & -k_1 \end{bmatrix} \tag{4.23}$$

is constituted by the vector of gains for the state-variables, $\mathbf{K}$, and by the gain for the integrative action k_1. The gains are used in the control structure as detailed in Fig. 4.2. Note that is assumed not only that the plant is completely state controllable, but also that a zero at the origin in the plant, capable of canceling the integrative action, does not exist.

4.2 Feedback Linearization Control

Feedback Linearization consists in a nonlinear control approach that aims to algebraically transform nonlinear dynamics of systems, through a nonlinear change of coordinates and nonlinear state feedback, into a model that is linear in the new set of coordinates. In this way, the application of linear control techniques is possible. The linear model produced is an exact representation of the original nonlinear model over a large set of operating points [53].

In fact, most feedback linearization approaches are based either on input-output linearization or in state-space linearization. The two concepts are exposed in [53], [54], [55] and [56]. Since in a tracking control problem the output variables are of interest and since linearizing the state equations does not necessarily linearize the output equations [55], the first approach is chosen. For multi-input multi-output systems, the application of the input-output method is often referred as the feedback decoupling problem [56] or as noninteractive control [54], since it results in a reduction of the system, at least from an input-output point of view, to an aggregate of independent single-input single-output channels.

Given a nonlinear system of the form:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u} \tag{4.24}$$

$$\mathbf{y} = \mathbf{h}(\mathbf{x}) \tag{4.25}$$

where $\mathbf{f}(\mathbf{x})$ is an n-dimensional vector of sufficiently smooth nonlinear functions, $\mathbf{g}(\mathbf{x})$ is an $(\mathrm{n} \times \mathrm{m})$-dimensional matrix of sufficiently smooth nonlinear functions and $\mathbf{h}(\mathbf{x})$ is an m-dimensional vector of sufficiently smooth nonlinear functions. The derivative of the output $\mathbf{y}_j$, with $j = 1, ..., m$, is defined by

$$\dot{\mathbf{y}}_j = \nabla \mathbf{h}_j(\mathbf{x})\dot{\mathbf{x}} = \nabla \mathbf{h}_j(\mathbf{x})\mathbf{f}(\mathbf{x}) + \nabla \mathbf{h}_j(\mathbf{x})\mathbf{g}(\mathbf{x})\mathbf{u} \tag{4.26}$$

The directional derivative of $\mathbf{h}_j(\mathbf{x})$ in the direction of $\mathbf{f}(\mathbf{x})$ is denominated as *Lie derivative*, named in honour of the Norwegian mathematician Sophus Lie, and is represented by [56]:

$$\mathcal{L}_{\mathbf{f}}\mathbf{h}_j(\mathbf{x}) = \sum_{i=1}^{n} \frac{\partial \mathbf{h}_j}{\partial \mathbf{x}_i}\mathbf{f}_i(\mathbf{x}) = \nabla \mathbf{h}_j(\mathbf{x})\mathbf{f}(\mathbf{x}) \tag{4.27}$$

In other words, the *Lie derivative* of $\mathbf{h}_j(\mathbf{x})$ in the direction of a vector field $\mathbf{f}(\mathbf{x})$ is denoted as $\mathcal{L}_\mathbf{f}\mathbf{h}_j$. Thus, (4.26) can be rewritten as:

$$\dot{\mathbf{y}}_j = \mathcal{L}_\mathbf{f}\mathbf{h}_j(\mathbf{x}) + \mathcal{L}_\mathbf{g}\mathbf{h}_j(\mathbf{x})\mathbf{u} \tag{4.28}$$

If on the first derivative of the output $\mathbf{y}_j$ the equality $\mathcal{L}_\mathbf{g}\mathbf{h}_j(\mathbf{x})\mathbf{u} = \mathbf{0}$ is verified, the former equation simplifies into:

$$\dot{\mathbf{y}}_j = \mathbf{y}_j^{(1)} = \mathcal{L}_\mathbf{f}\mathbf{h}_j(\mathbf{x}) \tag{4.29}$$

Note that in this case the first derivative of $\mathbf{y}_j$ remains independent of the inputs of the system. Notwithstanding, higher order derivatives could present a dependence on the inputs of the system. By letting the derivatives of $\mathbf{y}_j$ be represented by:

$$\mathbf{y}_j^{(k)} = \mathcal{L}_\mathbf{f}^\mathbf{k}\mathbf{h}_j(\mathbf{x}) + \sum_{i=1}^{m} \mathcal{L}_{\mathbf{g}_i}\mathcal{L}_\mathbf{f}^{\mathbf{k-1}}\mathbf{h}_j(\mathbf{x})\mathbf{u}_i \tag{4.30}$$

it is possible to introduce the concept of relative degree, r_j, of the output $\mathbf{y}_j$. The nonlinear system is said to have *vector relative degree* $\begin{bmatrix} r_1 & r_2 & \cdots & r_m \end{bmatrix}^T$ at the point $\mathbf{x_0}$ if [53]:

- $\mathcal{L}_{\mathbf{g_i}}\mathcal{L}_\mathbf{f}^\mathbf{k}\mathbf{h}_j(\mathbf{x}) = \mathbf{0}$ for all $1 \leq i, j \leq m$, for all $k < r_j - 1$ and for all $\mathbf{x}$ in a neighborhood of $\mathbf{x_0}$.

- The m $\times$ m *decoupling matrix*

$$\mathbf{\Lambda}(\mathbf{x}) = \begin{bmatrix} \mathcal{L}_{\mathbf{g}_1}\mathcal{L}_\mathbf{f}^{\mathbf{r_1-1}}\mathbf{h}_1(\mathbf{x}) & \cdots & \mathcal{L}_{\mathbf{g}_m}\mathcal{L}_\mathbf{f}^{\mathbf{r_1-1}}\mathbf{h}_1(\mathbf{x}) \\ \vdots & \ddots & \vdots \\ \mathcal{L}_{\mathbf{g}_1}\mathcal{L}_\mathbf{f}^{\mathbf{r_m-1}}\mathbf{h}_m(\mathbf{x}) & \cdots & \mathcal{L}_{\mathbf{g}_m}\mathcal{L}_\mathbf{f}^{\mathbf{r_m-1}}\mathbf{h}_m(\mathbf{x}) \end{bmatrix} \tag{4.31}$$

  is non-singular at the point $\mathbf{x_0}$.

The integer r_j represents the smallest relative degree of the j-th output with respect to any of the m inputs. Additionally, if the equality $r_1 + ... + r_m = n$ is verified, this system can be modified into a fully linear and decoupled controllable system through state feedback and through the application of the following transformation for each output $\mathbf{y}_j$ [56]:

$$\boldsymbol{\xi}_{j,1} = \mathbf{y}_j \tag{4.32}$$

$$\boldsymbol{\xi}_{j,2} = \dot{\boldsymbol{\xi}}_{j,1} = \mathcal{L}_\mathbf{f}\mathbf{h}_j(\mathbf{x}) \tag{4.33}$$

$$\vdots$$

$$\boldsymbol{\xi}_{j,r_j} = \dot{\boldsymbol{\xi}}_{j,r_j-1} = \mathcal{L}_\mathbf{f}^{\mathbf{r_j-1}}\mathbf{h}_j(\mathbf{x}) \tag{4.34}$$

$$\dot{\boldsymbol{\xi}}_{j,r_j} = \mathcal{L}_\mathbf{f}^{\mathbf{r_j}}\mathbf{h}_j(\mathbf{x}) + \sum_{i=1}^{m} \mathcal{L}_{\mathbf{g}_i}\mathcal{L}_\mathbf{f}^{\mathbf{r_j-1}}\mathbf{h}_j(\mathbf{x})\mathbf{u}_i \tag{4.35}$$

From (4.35), for $j = 1, ..., m$, is possible to define the transformed input variables $\mathbf{v}_j$:

$$\mathbf{v}_1 = \dot{\boldsymbol{\xi}}_{1,r_1} = \mathcal{L}_{\mathbf{f}}^{\mathbf{r_1}} \mathbf{h}_1(\mathbf{x}) + \sum_{i=1}^{m} \mathcal{L}_{\mathbf{g}_i} \mathcal{L}_{\mathbf{f}}^{\mathbf{r_1}-\mathbf{1}} \mathbf{h}_1(\mathbf{x}) \mathbf{u}_i \tag{4.36}$$

$$\vdots$$

$$\mathbf{v}_m = \dot{\boldsymbol{\xi}}_{m,r_m} = \mathcal{L}_{\mathbf{f}}^{\mathbf{r_m}} \mathbf{h}_m(\mathbf{x}) + \sum_{i=1}^{m} \mathcal{L}_{\mathbf{g}_i} \mathcal{L}_{\mathbf{f}}^{\mathbf{r_m}-\mathbf{1}} \mathbf{h}_m(\mathbf{x}) \mathbf{u}_i \tag{4.37}$$

Given the equality $\sum_{j=1}^{m} r_j = n$, the transformation described formerly defines a local *diffeomorphism* [56]. The vector of transformed input variables, $\mathbf{v}$, is compactly expressed as:

$$\mathbf{v} = \mathbf{b}(\mathbf{x}) + \boldsymbol{\Lambda}(\mathbf{x})\mathbf{u} \tag{4.38}$$

where $\boldsymbol{\Lambda}(\mathbf{x})$ is defined by (4.31) and $\mathbf{b}(\mathbf{x})$ is given by

$$\mathbf{b}(\mathbf{x}) = \begin{bmatrix} \mathcal{L}_{\mathbf{f}}^{\mathbf{r_1}} \mathbf{h}_1(\mathbf{x}) \\ \vdots \\ \mathcal{L}_{\mathbf{f}}^{\mathbf{r_m}} \mathbf{h}_m(\mathbf{x}) \end{bmatrix} \tag{4.39}$$

Solving in order to the input vector of the system, $\mathbf{u}$, yields:

$$\mathbf{u} = -\boldsymbol{\Lambda}^{-1}(\mathbf{x}) \cdot \mathbf{b}(\mathbf{x}) + \boldsymbol{\Lambda}^{-1}(\mathbf{x}) \cdot \mathbf{v} = \boldsymbol{\alpha}(\mathbf{x}) + \boldsymbol{\beta}(\mathbf{x}) \cdot \mathbf{v} \tag{4.40}$$

where $\boldsymbol{\alpha}(\mathbf{x})$ is an m-dimensional vector of nonlinear functions and $\boldsymbol{\beta}(\mathbf{x})$ is an $(\mathrm{m} \times \mathrm{m})$-dimensional matrix of nonlinear functions. Equation (4.40) corresponds to the nonlinear static state feedback control law. It is overt that the input-output decoupling problem only has a solution if the *decoupling matrix* $\boldsymbol{\Lambda}(\mathbf{x})$ is nonsingular. Substituting the previous result into (4.30)

$$\begin{bmatrix} \mathbf{y}_1^{(r_1)} \\ \vdots \\ \mathbf{y}_m^{(r_m)} \end{bmatrix} = \begin{bmatrix} \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_m \end{bmatrix} \tag{4.41}$$

results in a set of m linear systems that are decoupled, i.e. the transformed input $\mathbf{v}_j$ only impacts the output $\mathbf{y}_j$, and formed by a chain of r_j integrators. Furthermore, once each system is in a linear and controllable form, it is possible to stabilize them by means of linear controllers. Consequently, the modern control approach addressed in the last subsection could be used to select an appropriate control law $\mathbf{v}_j$ in order to satisfy the desired response requirements.

Therefore, after applying the input-output feedback linearization, which consists in the nonlinear change of coordinates, described from (4.32) to (4.37), and in the nonlinear static-state feedback presented in (4.40), the input-output model is linear in the new set of coordinates and the system has the

following representation:

$$\dot{\boldsymbol{\xi}} = \mathbf{A}\boldsymbol{\xi} + \mathbf{B}\mathbf{v} \tag{4.42}$$

$$\mathbf{y} = \mathbf{C}\boldsymbol{\xi} \tag{4.43}$$

where $\boldsymbol{\xi}$ denotes the n-dimensional vector of the transformed state variables and is given by

$$\boldsymbol{\xi} = \begin{bmatrix} \boldsymbol{\xi}_{1,1} & \cdots & \boldsymbol{\xi}_{1,r_1} & \boldsymbol{\xi}_{2,1} & \cdots & \boldsymbol{\xi}_{2,r_2} & \cdots & \boldsymbol{\xi}_{m,1} & \cdots & \boldsymbol{\xi}_{m,r_m} \end{bmatrix}^T \tag{4.44}$$

The matrices $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ of the system defined by (4.42) and (4.43) have a very simple canonical structure.

Considering now the case in which $\sum_{j=1}^{m} r_j = r_T < n$. Assuming that the *decoupling matrix* $\boldsymbol{\Lambda}(\mathbf{x})$ is nonsingular, the diffeomorphism $\boldsymbol{\Phi}(\mathbf{x}) = \begin{bmatrix} \boldsymbol{\xi}^T & \boldsymbol{\Phi}_{r_T+1} & \cdots & \boldsymbol{\Phi}_n \end{bmatrix}^T$ can be defined by setting the first r_T coordinates as [53]

$$\boldsymbol{\xi}_{j,k} = \boldsymbol{\Phi}_k^j(\mathbf{x}) = \mathcal{L}_{\mathbf{f}}^{\mathbf{k-1}} \mathbf{h}_j(\mathbf{x}) \tag{4.45}$$

for $1 \leq k \leq r_j$ and $1 \leq j \leq m$. In fact, $n - r_T$ additional coordinates $\boldsymbol{\kappa}^T = \begin{bmatrix} \boldsymbol{\Phi}_{r_T+1}, \cdots, \boldsymbol{\Phi}_n \end{bmatrix}$ are always possible to be found such that $\boldsymbol{\Phi}(\mathbf{x})$ has a Jacobian matrix which is nonsingular at the point $\mathbf{x}_0$ [54]. Furthermore, if the functions present in $\mathbf{g}(\mathbf{x})$ are involutive near $\mathbf{x}_0$, choosing the additional coordinates such that $\mathcal{L}_{\mathbf{g}_j} \boldsymbol{\Phi}_{r_T+i}(\mathbf{x}) = 0$ for all $1 \leq i \leq n - r_T$ and for all $1 \leq j \leq m$ is feasible. However, generally, the involutivity condition is not fulfilled and the $n - r_T$ functions $\boldsymbol{\kappa}$ can only be written generically with the following vector notation:

$$\dot{\boldsymbol{\kappa}} = \mathbf{q}(\xi, \kappa) + \mathbf{p}(\xi, \kappa)\mathbf{u} \tag{4.46}$$

The *normal form* of the equations that describe (locally around a point $\mathbf{x_0}$) a nonlinear system, with m inputs and m outputs, with relative degree $r_T = \sum_{j=1}^{m} r_j$ at $\mathbf{x_0}$ is characterized by (4.46) and by the following equations [54]:

$$\dot{\boldsymbol{\xi}}_{i,1} = \boldsymbol{\xi}_{i,2} \tag{4.47}$$

$$\vdots$$

$$\dot{\boldsymbol{\xi}}_{i,r_i-1} = \boldsymbol{\xi}_{i,r_i} \tag{4.48}$$

$$\dot{\boldsymbol{\xi}}_{i,r_i} = \mathbf{b}_i(\boldsymbol{\xi}, \boldsymbol{\kappa}) + \sum_{j=1}^{m} \boldsymbol{\Lambda}_{i,j}(\boldsymbol{\xi}, \boldsymbol{\kappa}) \mathbf{u}_j \tag{4.49}$$

for $1 \leq i \leq m$ and where

$$\boldsymbol{\Lambda}_{i,j}(\boldsymbol{\xi}, \boldsymbol{\kappa}) = \mathcal{L}_{\mathbf{g}_j} \mathcal{L}_{\mathbf{f}}^{\mathbf{r_i}-\mathbf{1}} \mathbf{h}_i(\boldsymbol{\Phi}^{-1}(\boldsymbol{\xi}, \boldsymbol{\kappa})) \tag{4.50}$$

$$\mathbf{b}_i(\boldsymbol{\xi}, \boldsymbol{\kappa}) = \mathcal{L}_{\mathbf{f}}^{\mathbf{r_i}} \mathbf{h}_i(\boldsymbol{\Phi}^{-1}(\boldsymbol{\xi}, \boldsymbol{\kappa})) \tag{4.51}$$

for $1 \leq i, j \leq m$. From (4.46), the concept of *zero dynamics* of a system with *relative degree* r_T can be explained. The refered dynamics are derived by solving the *Problem of Zeroing the Output* [54], i.e, to find initial conditions and inputs with the constraint that the output function $\mathbf{y}(t)$ is identically zero for all

times in a neighborhood of $t = 0$, and, subsequently, to analyze the corresponding internal dynamics of the system. If $\mathbf{y}(t) = \mathbf{0}$, then $\boldsymbol{\xi}(t) = \mathbf{0}$ for all t near 0. Imposing the derivative of order r_i of $\mathbf{y}_i(t)$ to be zero, for all $1 \leq i \leq m$, results in the inputs $\mathbf{u}_i(t)$ to be solutions of the system of equations:

$$\mathbf{0} = \dot{\boldsymbol{\xi}}_{i,r_i} = \mathbf{b}_i(\boldsymbol{\xi}, \boldsymbol{\kappa}) + \sum_{j=1}^{m} \boldsymbol{\Lambda}_{i,j}(\boldsymbol{\xi}, \boldsymbol{\kappa})\mathbf{u}_j \tag{4.52}$$

which can be rewritten with the following vector notation:

$$\mathbf{b}(\mathbf{0}, \boldsymbol{\kappa}(t)) + \boldsymbol{\Lambda}(\mathbf{0}, \boldsymbol{\kappa}(t))\mathbf{u}(t) = \mathbf{0} \tag{4.53}$$

Bearing in mind the nonsingularity of the *decoupling matrix* at $x = x_0$ by definition, is possible to describe the input vector $\mathbf{u}(t)$ by

$$\mathbf{u}(t) = -\boldsymbol{\Lambda}^{-1}(\mathbf{0}, \boldsymbol{\kappa}(t))\mathbf{b}(\mathbf{0}, \boldsymbol{\kappa}(t)) \tag{4.54}$$

Replacing the former result in (4.46) yields:

$$\dot{\boldsymbol{\kappa}}(t) = \mathbf{q}(\mathbf{0}, \kappa(t)) - \mathbf{p}(\mathbf{0}, \kappa)\boldsymbol{\Lambda}^{-1}(\mathbf{0}, \boldsymbol{\kappa}(t))\mathbf{b}(\mathbf{0}, \boldsymbol{\kappa}(t)) \tag{4.55}$$

The dynamics characterized by (4.55) describe the internal behaviour of the system when it is forced to track the output $\mathbf{y}(t) = 0$ correspond to the *zero dynamics* and are rather important. In order to the system to be locally asymptotically stable after applying feedback linearization and the further linear controllers, these *zero dynamics* are required to be asymptotically stable [54].

In fact, so far, the existence of the inverse of the *decoupling matrix* $\boldsymbol{\Lambda}(\mathbf{x})$ was assumed. However, in some cases, when $\sum_{j=1}^{m} r_j < n$, the matrix is singular and, consequently, is not possible to apply the static state feedback control law since the relative degree is invariant under this type of feedback. In order to achieve the equality $\sum_{j=1}^{m} r_j = n$ and the consequent nonsingularity of the decoupling matrix $\boldsymbol{\Lambda}(\mathbf{x})$, is required to resort to a more general feedback law capable of incorporating an additional set of state variables. This is attained by adding a dynamic compensator that results in a dynamic control transformation. The dynamic state feedback control law is modelled by:

$$\mathbf{u} = \boldsymbol{\alpha}(\mathbf{x}, \boldsymbol{\zeta}) + \boldsymbol{\beta}(\mathbf{x}, \boldsymbol{\zeta})\mathbf{v} \tag{4.56}$$

$$\dot{\boldsymbol{\zeta}} = \boldsymbol{\gamma}(\mathbf{x}, \boldsymbol{\zeta}) + \boldsymbol{\delta}(\mathbf{x}, \boldsymbol{\zeta})\mathbf{v} \tag{4.57}$$

where $\boldsymbol{\zeta}$ is a q-dimensional vector of controller state variables, $\boldsymbol{\gamma}$ is a q-dimensional vector of nonlinear functions and $\boldsymbol{\delta}$ is a $(q \times m)$-dimensional matrix of nonlinear functions. Note that the number of inputs and outputs of the system remains the same. The most common approach is to perform the compensation through the addition of integrators, which translates into delaying the appearance of the input dependence to higher-order derivatives of the output. Therefore, an increase in the relative degree of some of the output occurs which increases the possibility of the decoupling matrix being nonsingular.

Chapter 5

Control Design

With the description of the theoretical basis effected in the preceding chapter, the control strategies will now be discussed in detail for the dynamics of a quadcopter. Regarding the linear control method, two different approaches will be presented. As for the nonlinear control technique, three different ways of tackling the control problem are proposed.

5.1 Linear Quadratic Regulator with Integrative Action

The linearization performed in section 3.4 resulted in four linear decoupled subsystems, with the x and y subsystems encompassing the pitch and roll subsystems, respectively. Thus, the implementation of linear controllers to the dynamics of the UAV is possible.

The control approaches relying only on the LQR theory presented in section 4.1 are now addressed. To this end, firstly, an architecture based on the direct application of the LQR with integrative action depicted in Fig. 4.2 on each subsystem is derived. Subsequently, with the view of study an inner-outer loop control structure, where the innermost loop is responsible for the attitude control and the outermost solves the positioning control, a double-loop architecture is developed to tackle the control problem. In order to implement these structures, the position and the time derivative of the error in the xy inertial plan must be described with the body-fixed frame orientation.

5.1.1 Single Loop Structure

The first architecture using only linear controllers is illustrated in Fig. 5.1.

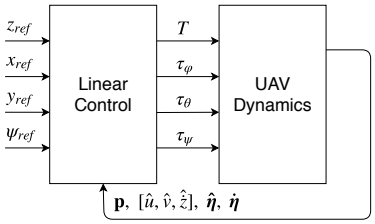

Figure 5.1: Structure of the single loop control structure.

The input vector, $\mathbf{u_s} = [T \;\; \tau_\varphi \;\; \tau_\theta \;\; \tau_\psi]$ is computed through:

$$T = \mathbf{u_z} + mg = -\mathbf{K_z}\mathbf{x_z} - k_{1_z}\varsigma_z + mg \tag{5.1}$$

$$\begin{bmatrix} \tau_\theta \\ \tau_\varphi \end{bmatrix} = \begin{bmatrix} \mathbf{u_\theta} \\ \mathbf{u_\varphi} \end{bmatrix} = {}^B\mathbf{R}_{I_{ij}} \begin{bmatrix} -\mathbf{K_{x_1}}x - k_{1_x}\varsigma_x \\ -\mathbf{K_{y_1}}y - k_{1_y}\varsigma_y \\ -\mathbf{K_{z_1}}z - k_{1_z}\varsigma_z \end{bmatrix} - \begin{bmatrix} \mathbf{K_{x_2}}u + \mathbf{K_{x_3}}\theta + \mathbf{K_{x_4}}\dot{\theta} \\ \mathbf{K_{y_2}}v + \mathbf{K_{y_3}}\varphi + \mathbf{K_{y_4}}\dot{\varphi} \end{bmatrix} \tag{5.2}$$

$$\tau_\psi = \mathbf{u_\psi} = -\mathbf{K_\psi}\mathbf{x_\psi} - k_{1_\psi}\varsigma_\psi \tag{5.3}$$

with $i = 1, 2$ and $j = 1, 2, 3$.

5.1.2 Inner-Outer Loop Structure

The LQR with integrative action is applied for each subsystem present in the outer (position control) and in the inner (attitude control) loops. The control structure is schematized in Fig. 5.2, from which it is noted that the references for the pitch θ and roll φ angles result from the position control.

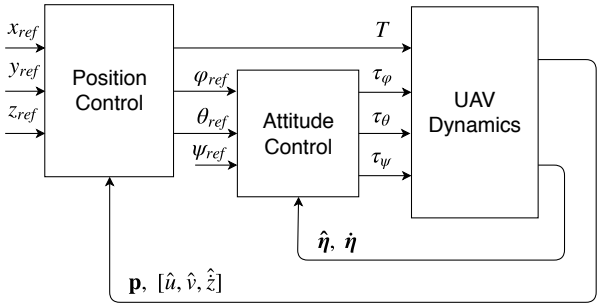

Figure 5.2: Structure of the inner-outer loop control structure.

In order to develop this architecture, the translational and the rotational dynamics present in the x and y subsystems have to be considered separately. Therefore, the inner loop of the control structure will include the pitch θ and roll φ subsystems, defined in subsections 3.4.2 and 3.4.3, and the outermost loop will comprise the state-spaces obtained from (3.37) and (3.49). With the establishment of the next state-variables and modified input of the subsystem:

$$\mathbf{x_x^o} = [{}^Bx_I \;\; u]^T, \;\; \mathbf{u_x^o} = \theta \tag{5.4}$$

the following state-space representation, that relates the ${}^Bx_I$ coordinate of the position with the pitch angle, θ, is obtained

$$\dot{\mathbf{x}}_\mathbf{x}^\mathbf{o} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \mathbf{x_x^o} + \begin{bmatrix} 0 \\ g \end{bmatrix} \mathbf{u_x^o} \tag{5.5}$$

$$\mathbf{y_x^o} = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x_x^o} \tag{5.6}$$

By defining the following state-variables and modified input of the subsystem:

$$\mathbf{x_y^o} = [{}^B y_I \;\; v]^T, \;\; \mathbf{u_y^o} = \varphi \tag{5.7}$$

the following state-space representation, that relates the ${}^B y_I$ coordinate of the position with the roll angle, φ, is obtained

$$\dot{\mathbf{x}}_\mathbf{y}^\mathbf{o} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \mathbf{x_y^o} + \begin{bmatrix} 0 \\ -g \end{bmatrix} \mathbf{u_y^o} \tag{5.8}$$

$$\mathbf{y_y^o} = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x_y^o} \tag{5.9}$$

The input vector is computed through:

$$\mathbf{u_{dl}} = \begin{bmatrix} T \\ \tau_\varphi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} \mathbf{u_z} + mg \\ \mathbf{u_\varphi} \\ \mathbf{u_\theta} \\ \mathbf{u_\psi} \end{bmatrix} = \begin{bmatrix} -\mathbf{K_z x_z} - k_{1_z}\varsigma_z + mg \\ -\mathbf{K_\varphi x_\varphi} - k_{1_\varphi}\varsigma_\varphi \\ -\mathbf{K_\theta x_\theta} - k_{1_\theta}\varsigma_\theta \\ -\mathbf{K_\psi x_\psi} - k_{1_\psi}\varsigma_\psi \end{bmatrix} \tag{5.10}$$

where the time derivative of ς_φ and ς_θ result from the outer loop:

$$\begin{bmatrix} \dot{\varsigma}_\theta \\ \dot{\varsigma}_\varphi \end{bmatrix} = {}^B\mathbf{R}_{I_{ij}} \begin{bmatrix} -\mathbf{K_{x_1}^o} x - k_{1_x}^o \varsigma_x \\ -\mathbf{K_{y_1}^o} y - k_{1_y}^o \varsigma_y \\ -\mathbf{K_{z_1}} z - k_{1_z}\varsigma_z \end{bmatrix} - \begin{bmatrix} \mathbf{K_{x_2}^o} u \\ \mathbf{K_{y_2}^o} v \end{bmatrix} - \begin{bmatrix} \theta \\ \varphi \end{bmatrix} \tag{5.11}$$

with $i = 1, 2$ and $j = 1, 2, 3$.

5.2 Feedback Linearization Control

The control approaches based on Feedback Linearization are derived while considering the nonlinear model presented in section 3.3.

5.2.1 Static Feedback Linearization with zero-dynamics stabilization

The first approach using feedback linearization control consists in applying it to the inner loop dynamics, formed by the attitude and altitude equations, and in stabilizing the outer loop with linear controllers. The zero-dynamics in this case will correspond to the translational dynamics in the xy plan. This solution was first proposed in [37], where the simplification $\begin{bmatrix} \dot{\varphi} & \dot{\theta} & \dot{\psi} \end{bmatrix} = \begin{bmatrix} p & q & r \end{bmatrix}$ is considered. However, this equality will not be established here due to the fact that this assumption only holds for smaller angles of movement.

Let $\mathbf{y_{in}} = \begin{bmatrix} z & \varphi & \theta & \psi \end{bmatrix}^T$ define the output vector of the inner loop. The first derivative of the output

is independent of the input. The second derivative of the output vector is given by:

$$\ddot{\mathbf{y}}_{\mathbf{in}} = \begin{bmatrix} \ddot{z} & \ddot{\varphi} & \ddot{\theta} & \ddot{\psi} \end{bmatrix}^T \tag{5.12}$$

where

$$\ddot{z} = \frac{T\,\cos(\theta)\,\cos(\varphi)}{m} - g \tag{5.13}$$

$$\begin{aligned}
\ddot{\varphi} \;=\; & \frac{\tau_\varphi}{I_x} + \cos(\varphi)\,\tan(\theta)\,\left(\frac{\tau_\psi}{I_z} + \frac{p\,q\,(I_x - I_y)}{I_z}\right) \\
& + \tan(\theta)\,\sin(\varphi)\,\left(\frac{\tau_\theta}{I_y} - \frac{p\,r\,(I_x - I_z)}{I_y}\right) \\
& + \cos(\varphi)\,\tan(\theta)\,q\,(p + \cos(\varphi)\,\tan(\theta)\,r + \tan(\theta)\,\sin(\varphi)\,q) \\
& - \tan(\theta)\,\sin(\varphi)\,r\,(p + \cos(\varphi)\,\tan(\theta)\,r + \tan(\theta)\,\sin(\varphi)\,q) \\
& + \frac{q\,r\,(I_y - I_z)}{I_x} + \cos(\varphi)\,r\,\left(\tan^2(\theta) + 1\right)\,(\cos(\varphi)\,q - \sin(\varphi)\,r) \\
& + \sin(\varphi)\,q\,\left(\tan^2(\theta) + 1\right)\,(\cos(\varphi)\,q - \sin(\varphi)\,r)
\end{aligned} \tag{5.14}$$

$$\begin{aligned}
\ddot{\theta} \;=\; & \cos(\varphi)\,\left(\frac{\tau_\theta}{I_y} - \frac{p\,r\,(I_x - I_z)}{I_y}\right) - \sin(\varphi)\,\left(\frac{\tau_\psi}{I_z} + \frac{p\,q\,(I_x - I_y)}{I_z}\right) \\
& - \cos(\varphi)\,r\,(p + \cos(\varphi)\,\tan(\theta)\,r + \tan(\theta)\,\sin(\varphi)\,q) \\
& - \sin(\varphi)\,q\,(p + \cos(\varphi)\,\tan(\theta)\,r\, + \tan(\theta)\,\sin(\varphi)\,q)
\end{aligned} \tag{5.15}$$

$$\begin{aligned}
\ddot{\psi} \;=\; & \frac{-\cos(\theta)\,p\,{I_y}^2\,\cos(\varphi)\,q + 4\,\sin(\theta)\,I_y\,I_z\,\cos^2(\varphi)\,q\,r}{I_y\,I_z\,\cos^2(\theta)} \\
& + \frac{2\,\sin(\theta)\,\sin(\varphi)\,I_y\,I_z\,\cos(\varphi)\,q^2 + \cos(\theta)\,p\,I_y\,I_z\,\cos(\varphi)\,q}{I_y\,I_z\,\cos^2(\theta)} \\
& + \frac{-2\,\sin(\theta)\,\sin(\varphi)\,I_y\,I_z\,\cos(\varphi)\,r^2 - 2\,\sin(\theta)\,I_y\,I_z\,q\,r}{I_y\,I_z\,\cos^2(\theta)} \\
& + \frac{-\cos(\theta)\,\sin(\varphi)\,p\,I_y\,I_z\,r + I_x\,\cos(\theta)\,p\,I_y\,\cos(\varphi)\,q}{I_y\,I_z\,\cos^2(\theta)} \\
& + \frac{\tau_\psi\,\cos(\theta)\,I_y\,\cos(\varphi) + \cos(\theta)\,\sin(\varphi)\,p\,{I_z}^2\,r}{I_y\,I_z\,\cos^2(\theta)} \\
& + \frac{-I_x\,\cos(\theta)\,\sin(\varphi)\,p\,I_z\,r + \tau_\theta\,\cos(\theta)\,\sin(\varphi)\,I_z}{I_y\,I_z\,\cos^2(\theta)}
\end{aligned} \tag{5.16}$$

From the previous equations the vector relative degree is determined:

$$\begin{bmatrix} r_1 & r_2 & r_3 & r_4 \end{bmatrix}^T = \begin{bmatrix} 2 & 2 & 2 & 2 \end{bmatrix}^T \tag{5.17}$$

Let $\mathbf{x_{in}}$ denote the vector of state-variables of the inner dynamics

$$\mathbf{x_{in}} = \begin{bmatrix} z & \dot{z} & \varphi & \theta & \psi & p & q & r \end{bmatrix}^T \tag{5.18}$$

and $\mathbf{u_{in}}$ describe the vector of inputs of the system

$$\mathbf{u_{in}} = \begin{bmatrix} T & \tau_\varphi & \tau_\theta & \tau_\psi \end{bmatrix}^T \tag{5.19}$$

Since the altitude and attitude dynamics combined have a total of 8 state-variables and $\sum_{j=1}^{4} r_j = 8$, the feedback linearization can be performed through a static state feedback law. Recalling (4.38) and (4.41), the vector $\ddot{\mathbf{y}}_{\mathbf{in}}$ transforms into:

$$\begin{bmatrix} \ddot{z} \\ \ddot{\varphi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \mathbf{b}(\mathbf{x_{in}}) + \mathbf{\Lambda}(\mathbf{x_{in}})\mathbf{u_{in}} \tag{5.20}$$

yielding the following *decoupling matrix*

$$\mathbf{\Lambda}(\mathbf{x_{in}}) = \begin{bmatrix} \frac{\cos(\theta)\,\cos(\varphi)}{m} & 0 & 0 & 0 \\ 0 & \frac{1}{I_x} & \frac{\tan(\theta)\,\sin(\varphi)}{I_y} & \frac{\cos(\varphi)\,\tan(\theta)}{I_z} \\ 0 & 0 & \frac{\cos(\varphi)}{I_y} & -\frac{\sin(\varphi)}{I_z} \\ 0 & 0 & \frac{\sin(\varphi)}{I_y\,\cos(\theta)} & \frac{\cos(\varphi)}{I_z\,\cos(\theta)} \end{bmatrix} \tag{5.21}$$

whose determinant is

$$\det\left(\mathbf{\Lambda}(\mathbf{x_{in}})\right) = \frac{\cos\left(\varphi\right)}{I_x\,I_y\,I_z\,m} \tag{5.22}$$

Thus, the *decoupling matrix* $\mathbf{\Lambda}(\mathbf{x_{in}})$ is invertible at any point defined by $\frac{-\pi}{2} < \varphi < \frac{\pi}{2}$. Therefore, by resorting to the static state feedback control law, the inner dynamics are possible to be input-output linearized. The vector $\mathbf{b}(\mathbf{x_{in}})$ is directly obtained by extracting the terms that are independent of the input of each of the second derivatives. Consequently, the static state feedback law denoted by

$$\mathbf{u_{in}} = \boldsymbol{\alpha}(\mathbf{x_{in}}) + \boldsymbol{\beta}(\mathbf{x_{in}})\mathbf{v_{in}} \tag{5.23}$$

has the matrix $\boldsymbol{\beta}(\mathbf{x_{in}})$ with the form

$$\boldsymbol{\beta}(\mathbf{x_{in}}) = \begin{bmatrix} \frac{m}{\cos(\theta)\,\cos(\varphi)} & 0 & 0 & 0 \\ 0 & I_x & 0 & -I_x\,\sin\left(\theta\right) \\ 0 & 0 & I_y\,\cos\left(\varphi\right) & I_y\,\cos\left(\theta\right)\,\sin\left(\varphi\right) \\ 0 & 0 & -I_z\,\sin\left(\varphi\right) & I_z\,\cos\left(\theta\right)\,\cos\left(\varphi\right) \end{bmatrix} \tag{5.24}$$

and the entries of the vector $\boldsymbol{\alpha}(\mathbf{x_{in}}) = \left[\boldsymbol{\alpha}_1\ \boldsymbol{\alpha}_2\ \boldsymbol{\alpha}_3\ \boldsymbol{\alpha}_4\right]^T$ expressed by

$$\boldsymbol{\alpha}_1(\mathbf{x_{in}}) = \frac{m\,g}{\cos\left(\theta\right)\,\cos\left(\varphi\right)} \tag{5.25}$$

$$\boldsymbol{\alpha}_2(\mathbf{x_{in}}) = I_z\,q\,r - I_y\,q\,r - \frac{I_x\,\sin\left(2\varphi\right)\,q^2}{2} + \frac{I_x\,\sin\left(2\varphi\right)\,r^2}{2} - I_x\,\cos\left(2\varphi\right)\,q\,r \tag{5.26}$$

$$
\begin{aligned}
\boldsymbol{\alpha}_3(\mathbf{x_{in}}) \;=\; & \frac{2\,I_y\,\cos(\varphi)\,\sin(\theta)\,r^2 - I_y\,\cos(\varphi)\,\sin(\theta)\,q^2 + I_x\,\cos(\theta)\,p\,r}{\cos(\theta)} \\
& + \frac{I_y\,\cos(\theta)\,p\,r - I_z\,\cos(\theta)\,p\,r + I_y\,\cos^3(\varphi)\,\sin(\theta)\,q^2}{\cos(\theta)} \\
& + \frac{-I_y\,\cos^3(\varphi)\,\sin(\theta)\,r^2 + 2\,I_y\,\sin(\theta)\,\sin(\varphi)\,q\,r}{\cos(\theta)} \\
& + \frac{-2\,I_y\,\cos^2(\varphi)\,\sin(\theta)\,\sin(\varphi)\,q\,r}{\cos(\theta)}
\end{aligned}
\tag{5.27}
$$

$$
\begin{aligned}
\boldsymbol{\alpha}_4(\mathbf{x_{in}}) \;=\; & -\frac{I_z\,\sin(\theta)\,\sin(\varphi)\,q^2 + I_x\,\cos(\theta)\,p\,q - I_y\,\cos(\theta)\,p\,q}{\cos(\theta)} \\
& - \frac{I_z\,\cos(\theta)\,p\,q + I_z\,\cos^2(\varphi)\,\sin(\theta)\,\sin(\varphi)\,q^2}{\cos(\theta)} \\
& - \frac{-I_z\,\cos^2(\varphi)\,\sin(\theta)\,\sin(\varphi)\,r^2 + 2\,I_z\,\cos^3(\varphi)\,\sin(\theta)\,q\,r}{\cos(\theta)}
\end{aligned}
\tag{5.28}
$$

From the previous equations, one states that the pitch angle must respect the condition $\frac{-\pi}{2} < \theta < \frac{\pi}{2}$. Applying the diffeomorphism described in (4.45) yields the vector of the transformed state-variables:

$$
\boldsymbol{\xi_{in}} = \boldsymbol{\Phi}(\mathbf{x_{in}}) = \begin{bmatrix} z & \dot{z} & \varphi & \dot{\varphi} & \theta & \dot{\theta} & \psi & \dot{\psi} \end{bmatrix}^T \tag{5.29}
$$

As a result of the static state feedback control law, illustrated in Fig. 5.3, and of the change of coordinates, the inner dynamics are now linearized and decoupled. These dynamics are translated into four single-input single-output chains of two integrators and are depicted in Fig. 5.4.

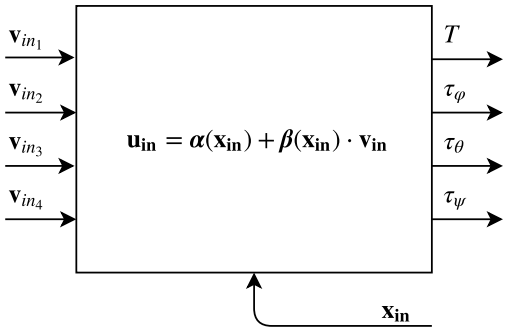

Figure 5.3: Block diagram representation of the static state feedback law derived for the inner dynamics.

Given the linear, decoupled and controllable form of the inner-loop dynamics, the Linear Quadratic Regulator control technique can be employed to each chain.

Concerning the remaining dynamics of the quadcopter, the *zero dynamics* can be specified through [57]:

$$
\begin{aligned}
\ddot{x} \;&=\; g\left(\frac{\cos(\varphi)\,\sin(\theta)\,\cos(\psi) + \sin(\varphi)\,\sin(\psi)}{\cos(\varphi)\,\cos(\theta)}\right) \\
&= g\left(\tan(\theta)\,\cos(\psi) + \frac{\sin(\psi)}{\cos(\theta)}\tan(\varphi)\right)
\end{aligned}
\tag{5.30}
$$

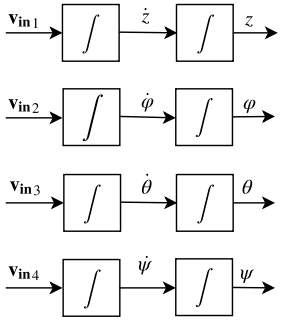

Figure 5.4: Block diagram representation of the resulting dynamics after applying feedback linearization to the altitude and attitude dynamics.

$$\begin{aligned} \ddot{y} &= g\left(\frac{\cos(\varphi)\,\sin(\theta)\,\sin(\psi) - \sin(\varphi)\,\cos(\psi)}{\cos(\varphi)\,\cos(\theta)}\right) \\ &= g\left(\tan(\theta)\,\sin(\psi) - \frac{\cos(\psi)}{\cos(\theta)}\tan(\varphi)\right) \end{aligned} \tag{5.31}$$

where the thrust $T = \frac{mg}{\cos(\theta)\cos(\varphi)}$ derives from zeroing the linear acceleration $\ddot{z}$. In fact, as expected, the *zero dynamics* are unstable. Therefore, an outer position control loop is necessary. The control strategy for this loop relies on the linearization performed in section 3.4 and on the LQR based outer-loop with integrative action presented in subsection 5.1.2.

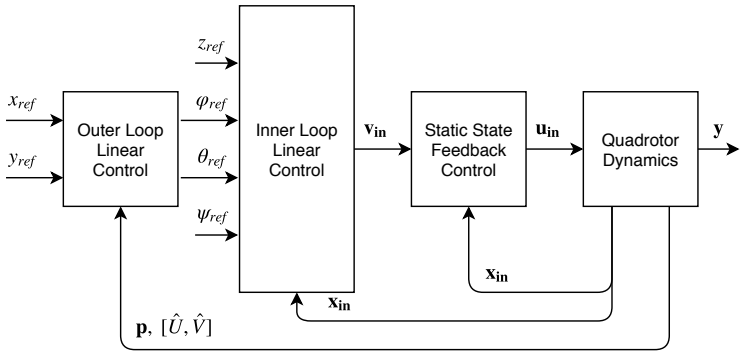

Figure 5.5: Block diagram representation of the control structure derived for the first approach using Feedback Linearization.

This first approach using Feedback Linearization is schematized in Fig. 5.5. The " Inner Loop Linear Control" block is formed by the LQR controllers with integrative action designed for the set of integrators represented in Fig. 5.4. In each chain, an additional integrator was embedded into the feedback control to obtain additional robustness. Thus, each subsystem present in the inner loop is now represented by the following completely controllable state-space representation:

$$\begin{bmatrix} \dot{\boldsymbol{\xi}}_{\mathbf{in}_i} \\ \dot{\varsigma}_{\mathbf{in}_i} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{(r_i-1)\times 1} & \boldsymbol{I}_{r_i-1} & \mathbf{0}_{(r_i-1)\times 1} \\ \mathbf{0} & \mathbf{0}_{1\times(r_i-1)} & 0 \\ -1 & \mathbf{0}_{1\times(r_i-1)} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi}_{\mathbf{in}_i} \\ \varsigma_{\mathbf{in}_i} \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{(r_i-1)\times 1} \\ 1 \\ 0 \end{bmatrix} \mathbf{v}_{\mathbf{in}i} \tag{5.32}$$

$$\mathbf{y}_{\mathbf{in}_i} = \begin{bmatrix} 1 & \mathbf{0}_{1\times r_i} \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi}_{\mathbf{in}_i} \\ \varsigma_{\mathbf{in}_i} \end{bmatrix} \tag{5.33}$$

5.2.2 Non-Interacting Control via Dynamic Feedback Linearization

In this subsection, a different application of the Feedback Linearization is considered. Contrary to the previous approach, based on a double-loop control structure in which the internal dynamics were linearized through static feedback linearization, the approach presently described proceeds to linearize the entire dynamics of the quadcopter through a dynamic nonlinear control law and originates a control structure with a single loop. The idea behind the exact linearization problem applied to the dynamics of an aerial vehicle is present in [54]. This concept is extended to quadrotors in [58], [59] and [60].

In fact, the drone is considered an underactuated system since it has six outputs and only four inputs. In furtherance of avoiding a non-square *decoupling matrix* $\mathbf{\Lambda}(\mathbf{x})$ and further difficulties in obtaining a feasible linearizing input, the output is defined as:

$$\mathbf{y_{df}} = \begin{bmatrix} x & y & z & \psi \end{bmatrix}^T \tag{5.34}$$

Recalling the equations that define the translational dynamics and (5.16), it is noted that the system has relative degree:

$$\begin{bmatrix} r_1 & r_2 & r_3 & r_4 \end{bmatrix}^T = \begin{bmatrix} 2 & 2 & 2 & 2 \end{bmatrix}^T \tag{5.35}$$

which yields

$$\sum_{j=1}^{4} r_j = 8 < n = 12 \tag{5.36}$$

Hence, the *decoupling matrix* $\mathbf{\Lambda}(\mathbf{x_{df}})$ is singular, which is confirmed by (5.37), and the feedback linearization can not be performed through a static state feedback law.

$$\mathbf{\Lambda}(\mathbf{x_{df}}) = \begin{bmatrix} \frac{\sin(\varphi)\,\sin(\psi)+\cos(\varphi)\,\cos(\psi)\,\sin(\theta)}{m} & 0 & 0 & 0 \\ -\frac{\cos(\psi)\,\sin(\varphi)-\cos(\varphi)\,\sin(\theta)\,\sin(\psi)}{m} & 0 & 0 & 0 \\ \frac{\cos(\theta)\,\cos(\varphi)}{m} & 0 & 0 & 0 \\ 0 & 0 & \frac{\sin(\varphi)}{I_y\,\cos(\theta)} & \frac{\cos(\varphi)}{I_z\,\cos(\theta)} \end{bmatrix} \tag{5.37}$$

Let $\mathbf{u_{df}}$ describe the vector of inputs of the system

$$\mathbf{u_{df}} = \begin{bmatrix} T & \tau_\varphi & \tau_\theta & \tau_\psi \end{bmatrix}^T \tag{5.38}$$

By analyzing (5.37), one notes that the second derivatives $\ddot{x}$, $\ddot{y}$ and $\ddot{z}$ are only dependent of the input T. In order to achieve a nonsingular *decoupling matrix*, the appearance of this input could be postponed to higher order derivatives that are influenced by the other inputs. As a result, the number of null entries of the matrix $\mathbf{\Lambda}(\mathbf{x})$ is reduced and the possibility of being invertible increases.

In subsection 5.2.1 it was shown that the entries of the system τ_φ, τ_θ and τ_ψ begin to appear in the second derivative of the Euler angles. Thus, since the linear accelerations depend on these angles, differentiating the position outputs x, y and z two additional times results in obtaining terms that are dependent on other entries besides the thrust T. In an effort to increase the relative degree of each

position output by two units, compensation through the addition of two integrators on the thrust input channel is performed [54]. This incorporation origins two new state-variables that are characterized by:

$$\begin{cases} \varepsilon &= T \\ \chi &= \dot{\varepsilon} \\ \dot{\chi} &= \bar{\mathbf{u}}_{\mathbf{df_1}} \end{cases} \tag{5.39}$$

where $\bar{\mathbf{u}}_{\mathbf{df_1}}$ denotes the first entry of the input vector of the extended system, $\bar{\mathbf{u}}_{\mathbf{df}}$. The other inputs remain unchanged, yielding:

$$\begin{bmatrix} \bar{\mathbf{u}}_{\mathbf{df_1}} & \bar{\mathbf{u}}_{\mathbf{df_2}} & \bar{\mathbf{u}}_{\mathbf{df_3}} & \bar{\mathbf{u}}_{\mathbf{df_4}} \end{bmatrix}^T = \begin{bmatrix} \dot{\chi} & \tau_\varphi & \tau_\theta & \tau_\psi \end{bmatrix}^T \tag{5.40}$$

The vector of state-variables of the extended system is represented by:

$$\bar{\mathbf{x}}_{\mathbf{df}} = [x \;\; y \;\; z \;\; \varphi \;\; \theta \;\; \psi \;\; \dot{x} \;\; \dot{y} \;\; \dot{z} \;\; \varepsilon \;\; \chi \;\; p \;\; q \;\; r]^T \tag{5.41}$$

After the inclusion of the dynamic compensator, the linear accelerations are expressed through:

$$\ddot{x} = (\cos(\varphi)\sin(\theta)\cos(\psi) + \sin(\varphi)\sin(\psi))\,\frac{\varepsilon}{m} \tag{5.42}$$

$$\ddot{y} = (\cos(\varphi)\sin(\theta)\sin(\psi) - \sin(\varphi)\cos(\psi))\,\frac{\varepsilon}{m} \tag{5.43}$$

$$\ddot{z} = \cos(\varphi)\cos(\theta)\,\frac{\varepsilon}{m} - g \tag{5.44}$$

Differentiating twice each one of the previous equations yields:

$$\begin{aligned} x^{(4)} = {} & \frac{I_x\, I_y\, T\, \sin(\varphi)\, \sin(\psi) + I_x\, \tau_\theta\, \cos(\theta)\, \cos(\psi)\, \varepsilon + I_y\, \tau_\varphi\, \cos(\varphi)\, \sin(\psi)\, \varepsilon}{I_x\, I_y\, m} \\ & + \frac{-I_y\, \tau_\varphi\, \cos(\psi)\, \sin(\theta)\, \sin(\varphi)\, \varepsilon + 2\, I_x\, I_y\, \cos(\theta)\, \cos(\psi)\, q\, \chi}{I_x\, I_y\, m} \\ & + \frac{2\, I_x\, I_y\, \cos(\varphi)\, \sin(\psi)\, p\, \chi - {I_x}^2\, \cos(\theta)\, \cos(\psi)\, p\, \varepsilon\, r}{I_x\, I_y\, m} \\ & + \frac{I_x\, I_y\, T\, \cos(\varphi)\, \cos(\psi)\, \sin(\theta) + {I_y}^2\, \cos(\varphi)\, \sin(\psi)\, q\, \varepsilon\, r}{I_x\, I_y\, m} \\ & + \frac{-I_x\, I_y\, \sin(\varphi)\, \sin(\psi)\, p^2\, \varepsilon - I_x\, I_y\, \sin(\varphi)\, \sin(\psi)\, q^2\, \varepsilon}{I_x\, I_y\, m} \\ & + \frac{-I_x\, I_y\, \cos(\varphi)\, \cos(\psi)\, \sin(\theta)\, p^2\, \varepsilon - I_x\, I_y\, \cos(\varphi)\, \cos(\psi)\, \sin(\theta)\, q^2\, \varepsilon}{I_x\, I_y\, m} \\ & + \frac{I_x\, I_y\, \cos(\theta)\, \cos(\psi)\, p\, \varepsilon\, r + I_x\, I_z\, \cos(\theta)\, \cos(\psi)\, p\, \varepsilon\, r}{I_x\, I_y\, m} \\ & + \frac{-I_x\, I_y\, \cos(\varphi)\, \sin(\psi)\, q\, \varepsilon\, r - I_y\, I_z\, \cos(\varphi)\, \sin(\psi)\, q\, \varepsilon\, r}{I_x\, I_y\, m} \\ & + \frac{-2\, I_x\, I_y\, \cos(\psi)\, \sin(\theta)\, \sin(\varphi)\, p\, \chi - {I_y}^2\, \cos(\psi)\, \sin(\theta)\, \sin(\varphi)\, q\, \varepsilon\, r}{I_x\, I_y\, m} \\ & + \frac{I_x\, I_y\, \cos(\psi)\, \sin(\theta)\, \sin(\varphi)\, q\, \varepsilon\, r + I_y\, I_z\, \cos(\psi)\, \sin(\theta)\, \sin(\varphi)\, q\, \varepsilon\, r}{I_x\, I_y\, m} \end{aligned} \tag{5.45}$$

$$
\begin{aligned}
y^{(4)} \;=\; & \frac{I_x\,\tau_\theta\,\cos(\theta)\,\sin(\psi)\,\varepsilon - I_y\,\tau_\varphi\,\cos(\varphi)\,\cos(\psi)\,\varepsilon}{I_x\,I_y\,m} \\
& + \frac{-I_x\,I_y\,T\,\cos(\psi)\,\sin(\varphi) - I_y\,\tau_\varphi\,\sin(\theta)\,\sin(\varphi)\,\sin(\psi)\,\varepsilon}{I_x\,I_y\,m} \\
& + \frac{-2\,I_x\,I_y\,\cos(\varphi)\,\cos(\psi)\,p\,\chi + 2\,I_x\,I_y\,\cos(\theta)\,\sin(\psi)\,q\,\chi}{I_x\,I_y\,m} \\
& + \frac{-{I_y}^2\,\cos(\varphi)\,\cos(\psi)\,q\,\varepsilon\,r - {I_x}^2\,\cos(\theta)\,\sin(\psi)\,p\,\varepsilon\,r}{I_x\,I_y\,m} \\
& + \frac{I_x\,I_y\,T\,\cos(\varphi)\,\sin(\theta)\,\sin(\psi) + I_x\,I_y\,\cos(\psi)\,\sin(\varphi)\,p^2\,\varepsilon}{I_x\,I_y\,m} \\
& + \frac{I_x\,I_y\,\cos(\psi)\,\sin(\varphi)\,q^2\,\varepsilon - I_x\,I_y\,\cos(\varphi)\,\sin(\theta)\,\sin(\psi)\,p^2\,\varepsilon}{I_x\,I_y\,m} \\
& + \frac{-I_x\,I_y\,\cos(\varphi)\,\sin(\theta)\,\sin(\psi)\,q^2\,\varepsilon + I_x\,I_y\,\cos(\varphi)\,\cos(\psi)\,q\,\varepsilon\,r}{I_x\,I_y\,m} \\
& + \frac{I_y\,I_z\,\cos(\varphi)\,\cos(\psi)\,q\,\varepsilon\,r + I_x\,I_y\,\cos(\theta)\,\sin(\psi)\,p\,\varepsilon\,r}{I_x\,I_y\,m} \\
& + \frac{I_x\,I_z\,\cos(\theta)\,\sin(\psi)\,p\,\varepsilon\,r - 2\,I_x\,I_y\,\sin(\theta)\,\sin(\varphi)\,\sin(\psi)\,p\,\chi}{I_x\,I_y\,m} \\
& + \frac{-{I_y}^2\,\sin(\theta)\,\sin(\varphi)\,\sin(\psi)\,q\,\varepsilon\,r + I_x\,I_y\,\sin(\theta)\,\sin(\varphi)\,\sin(\psi)\,q\,\varepsilon\,r}{I_x\,I_y\,m} \\
& + \frac{I_y\,I_z\,\sin(\theta)\,\sin(\varphi)\,\sin(\psi)\,q\,\varepsilon\,r}{I_x\,I_y\,m}
\end{aligned}
\tag{5.46}
$$

$$
\begin{aligned}
z^{(4)} \;=\; & -\frac{2\,\cos(\theta)\,\sin(\varphi)\,I_x\,I_y\,p\,\chi + 2\,\sin(\theta)\,I_x\,I_y\,q\,\chi\,2}{I_x\,I_y\,m} \\
& - \frac{-\sin(\theta)\,\varepsilon\,r\,{I_x}^2\,p + \cos(\theta)\,\cos(\varphi)\,\varepsilon\,I_x\,I_y\,p^2}{I_x\,I_y\,m} \\
& - \frac{\sin(\theta)\,\varepsilon\,r\,I_x\,I_y\,p + \cos(\theta)\,\cos(\varphi)\,\varepsilon\,I_x\,I_y\,q^2}{I_x\,I_y\,m} \\
& - \frac{-\cos(\theta)\,\sin(\varphi)\,\varepsilon\,r\,I_x\,I_y\,q - T\,\cos(\theta)\,\cos(\varphi)\,I_x\,I_y}{I_x\,I_y\,m} \\
& - \frac{I_z\,\sin(\theta)\,\varepsilon\,r\,I_x\,p + \tau_\theta\,\sin(\theta)\,\varepsilon\,I_x + \cos(\theta)\,\sin(\varphi)\,\varepsilon\,r\,{I_y}^2\,q}{I_x\,I_y\,m} \\
& - \frac{-I_z\,\cos(\theta)\,\sin(\varphi)\,\varepsilon\,r\,I_y\,q + \tau_\varphi\,\cos(\theta)\,\sin(\varphi)\,\varepsilon\,I_y}{I_x\,I_y\,m}
\end{aligned}
\tag{5.47}
$$

From the previous equations and (5.16) is easily concluded that the condition $\sum_{j=1}^{4} r_j = n = 14$ was achieved. Hence, the corresponding *decoupling matrix* is nonsingular and is expressed by:

$$
\mathbf{\Lambda}(\bar{\mathbf{x}}_{\mathbf{df}}) = \begin{bmatrix}
\mathbf{\Lambda}(\bar{\mathbf{x}}_{\mathbf{df}})_{11} & \mathbf{\Lambda}(\bar{\mathbf{x}}_{\mathbf{df}})_{12} & \frac{\cos(\theta)\,\cos(\psi)\,\varepsilon}{I_y\,m} & 0 \\
\mathbf{\Lambda}(\bar{\mathbf{x}}_{\mathbf{df}})_{21} & \mathbf{\Lambda}(\bar{\mathbf{x}}_{\mathbf{df}})_{22} & \frac{\cos(\theta)\,\sin(\psi)\,\varepsilon}{I_y\,m} & 0 \\
\frac{\cos(\theta)\,\cos(\varphi)}{m} & -\frac{\cos(\theta)\,\sin(\varphi)\,\varepsilon}{I_x\,m} & -\frac{\sin(\theta)\,\varepsilon}{I_y\,m} & 0 \\
0 & 0 & \frac{\sin(\varphi)}{I_y\,\cos(\theta)} & \frac{\cos(\varphi)}{I_z\,\cos(\theta)}
\end{bmatrix}
\tag{5.48}
$$

with

$$
\mathbf{\Lambda}(\bar{\mathbf{x}}_{\mathbf{df}})_{11} = \frac{\sin(\varphi)\,\sin(\psi) + \cos(\varphi)\,\cos(\psi)\,\sin(\theta)}{m} \tag{5.49}
$$

$$
\mathbf{\Lambda}(\bar{\mathbf{x}}_{\mathbf{df}})_{12} = \frac{\varepsilon\,(\cos(\varphi)\,\sin(\psi) - \cos(\psi)\,\sin(\theta)\,\sin(\varphi))}{I_x\,m} \tag{5.50}
$$

$$\mathbf{\Lambda}(\bar{\mathbf{x}}_{\mathbf{df}})_{21} = -\frac{\cos(\psi)\,\sin(\varphi) - \cos(\varphi)\,\sin(\theta)\,\sin(\psi)}{m} \tag{5.51}$$

$$\mathbf{\Lambda}(\bar{\mathbf{x}}_{\mathbf{df}})_{22} = -\frac{\varepsilon\,(\cos(\varphi)\,\cos(\psi) + \sin(\theta)\,\sin(\varphi)\,\sin(\psi))}{I_x\,m} \tag{5.52}$$

The respective determinant is

$$\det(\mathbf{\Lambda}(\bar{\mathbf{x}}_{\mathbf{df}})) = \frac{\cos(\varphi)\,\varepsilon^2}{I_x\,I_y\,I_z\,m^3\,\cos(\theta)} \tag{5.53}$$

Thus, the *decoupling matrix* $\mathbf{\Lambda}(\bar{\mathbf{x}}_{\mathbf{df}})$ is invertible at any point defined by $\frac{-\pi}{2} < \varphi < \frac{\pi}{2}$, $\frac{-\pi}{2} < \theta < \frac{\pi}{2}$ and $\varepsilon \neq 0$. In fact, the last condition is not verified when the quadrotor is on the ground waiting to take-off, however, this singularity is easily overcome in the implementation by setting a different initial value.

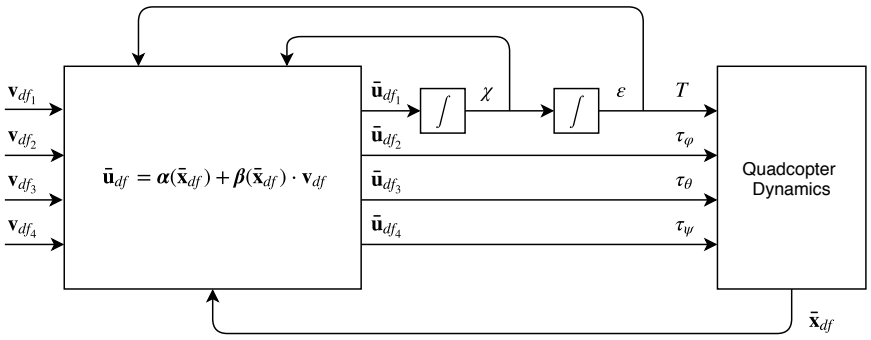

Figure 5.6: Block diagram representation of the resulting dynamic state feedback control law.

The consequent dynamic state feedback law, illustrated in Fig. 5.6, is defined by

$$\bar{\mathbf{u}}_{\mathbf{df}} = \boldsymbol{\alpha}(\bar{\mathbf{x}}_{\mathbf{df}}) + \boldsymbol{\beta}(\bar{\mathbf{x}}_{\mathbf{df}})\mathbf{v}_{\mathbf{df}} \tag{5.54}$$

where the matrix $\boldsymbol{\beta}(\bar{\mathbf{x}}_{\mathbf{df}})$ has the form

$$\boldsymbol{\beta}(\bar{\mathbf{x}}_{\mathbf{df}}) = \begin{bmatrix} \boldsymbol{\beta}(\bar{\mathbf{x}}_{\mathbf{df}})_{11} & \boldsymbol{\beta}(\bar{\mathbf{x}}_{\mathbf{df}})_{12} & m\,\cos(\theta)\,\cos(\varphi) & 0 \\ \boldsymbol{\beta}(\bar{\mathbf{x}}_{\mathbf{df}})_{21} & \boldsymbol{\beta}(\bar{\mathbf{x}}_{\mathbf{df}})_{22} & -\frac{I_x\,m\,\cos(\theta)\,\sin(\varphi)}{\varepsilon} & 0 \\ \frac{I_y\,m\,\cos(\theta)\,\cos(\psi)}{\varepsilon} & \frac{I_y\,m\,\cos(\theta)\,\sin(\psi)}{\varepsilon} & -\frac{I_y\,m\,\sin(\theta)}{\varepsilon} & 0 \\ -\frac{I_z\,m\,\cos(\theta)\,\cos(\psi)\,\sin(\varphi)}{\cos(\varphi)\,\varepsilon} & -\frac{I_z\,m\,\cos(\theta)\,\sin(\varphi)\,\sin(\psi)}{\cos(\varphi)\,\varepsilon} & \frac{I_z\,m\,\sin(\theta)\,\sin(\varphi)}{\cos(\varphi)\,\varepsilon} & \frac{I_z\,\cos(\theta)}{\cos(\varphi)} \end{bmatrix} \tag{5.55}$$

with

$$\boldsymbol{\beta}(\bar{\mathbf{x}}_{\mathbf{df}})_{11} = m\,(\sin(\varphi)\,\sin(\psi) + \cos(\varphi)\,\cos(\psi)\,\sin(\theta)) \tag{5.56}$$

$$\boldsymbol{\beta}(\bar{\mathbf{x}}_{\mathbf{df}})_{12} = m\,\cos(\varphi)\,\sin(\theta)\,\sin(\psi) - m\,\cos(\psi)\,\sin(\varphi) \tag{5.57}$$

$$\boldsymbol{\beta}(\bar{\mathbf{x}}_{\mathbf{df}})_{21} = \frac{m\,(I_x\,\cos(\varphi)\,\sin(\psi) - I_x\,\cos(\psi)\,\sin(\theta)\,\sin(\varphi))}{\varepsilon} \tag{5.58}$$

$$\boldsymbol{\beta}(\bar{\mathbf{x}}_{\mathbf{df}})_{22} = -\frac{m\,(I_x\,\cos(\varphi)\,\cos(\psi) + I_x\,\sin(\theta)\,\sin(\varphi)\,\sin(\psi))}{\varepsilon} \tag{5.59}$$

and the entries of the vector $\boldsymbol{\alpha}(\bar{\mathbf{x}}_{\mathbf{df}}) = [\boldsymbol{\alpha}_1\ \boldsymbol{\alpha}_2\ \boldsymbol{\alpha}_3\ \boldsymbol{\alpha}_4]^T$ are expressed by

$$\boldsymbol{\alpha}_1(\bar{\mathbf{x}}_{\mathbf{df}}) = \varepsilon\,\left(p^2 + q^2\right) \tag{5.60}$$

$$\boldsymbol{\alpha}_2(\bar{\mathbf{x}}_{\mathbf{df}}) = -\frac{2\,I_x\,p\,\chi - I_x\,q\,\varepsilon\,r + I_y\,q\,\varepsilon\,r - I_z\,q\,\varepsilon\,r}{\varepsilon} \tag{5.61}$$

$$\boldsymbol{\alpha}_3(\bar{\mathbf{x}}_{\mathbf{df}}) = -\frac{2\,I_y\,q\,\chi - I_x\,p\,\varepsilon\,r + I_y\,p\,\varepsilon\,r + I_z\,p\,\varepsilon\,r}{\varepsilon} \tag{5.62}$$

$$\begin{aligned}\boldsymbol{\alpha}_4(\bar{\mathbf{x}}_{\mathbf{df}}) \;=\; & -\frac{I_z\,\sin(\theta)\,\sin(2\,\varphi)\,q^2\,\varepsilon - I_z\,\sin(\theta)\,\sin(2\,\varphi)\,\varepsilon\,r^2}{\cos(\theta)\,\cos(\varphi)\,\varepsilon} \\ & - \frac{-2\,I_z\,\cos(\theta)\,\sin(\varphi)\,q\,\chi + 2\,I_z\,\sin(\theta)\,\cos(2\,\varphi)\,q\,\varepsilon\,r}{\cos(\theta)\,\cos(\varphi)\,\varepsilon} \\ & - \frac{I_x\,\cos(\theta)\,\cos(\varphi)\,p\,q\,\varepsilon - I_y\,\cos(\theta)\,\cos(\varphi)\,p\,q\,\varepsilon}{\cos(\theta)\,\cos(\varphi)\,\varepsilon} \\ & - \frac{I_z\,\cos(\theta)\,\cos(\varphi)\,p\,q\,\varepsilon - 2\,I_z\,\cos(\theta)\,\sin(\varphi)\,p\,\varepsilon\,r}{\cos(\theta)\,\cos(\varphi)\,\varepsilon}\end{aligned} \tag{5.63}$$

Using the transformation of coordinates expressed by the diffeomorphism described in (4.45) yields the vector of the transformed state-variables:

$$\boldsymbol{\xi}_{\mathbf{df}} = \boldsymbol{\Phi}(\mathbf{x}_{\mathbf{df}}) = \begin{bmatrix} x & \dot{x} & \ddot{x} & x^{(3)} & y & \dot{y} & \ddot{y} & y^{(3)} & z & \dot{z} & \ddot{z} & z^{(3)} & \psi & \dot{\psi} \end{bmatrix}^T \tag{5.64}$$

As a result of the dynamic state feedback control law detailed and of the change of coordinates, the quadcopter dynamics are now linearized and decoupled. Moreover, the position dynamics corresponds now to three chains of four integrators and the yaw angle dynamics transformed into a chain of two integrators. The linear and decoupled dynamics of the aerial vehicle are depicted in Fig. 5.7.

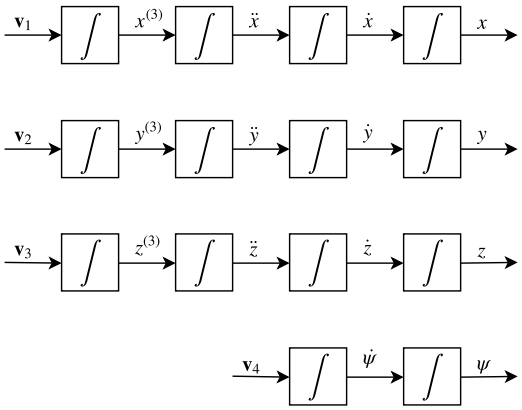

Figure 5.7: Block diagram representation of the resulting dynamics after applying dynamic feedback.

This second approach using Feedback Linearization is schematized in Fig. 5.8. The " Linear Control" block is formed by the LQR controllers with integrative action designed for the set of integrators represented in Fig. 5.7. In each chain, an additional integrator was included into the feedback control to obtain additional robustness. Hence, each subsystem is now represented by the following completely controllable state-space representation:

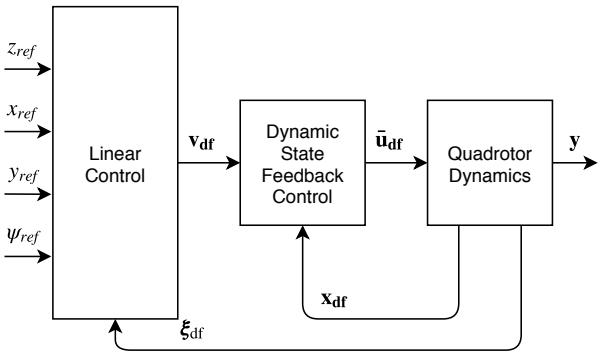

Figure 5.8: Block diagram representation of the control structure derived for the second approach using Feedback Linearization.

$$\begin{bmatrix} \dot{\boldsymbol{\xi}}_{\mathbf{df}_i} \\ \dot{\varsigma}_{\mathbf{df}_i} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{(r_i-1)\times 1} & \boldsymbol{I}_{r_i-1} & \mathbf{0}_{(r_i-1)\times 1} \\ 0 & \mathbf{0}_{1\times(r_i-1)} & 0 \\ -1 & \mathbf{0}_{1\times(r_i-1)} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi}_{\mathbf{df}_i} \\ \varsigma_{\mathbf{df}_i} \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{(r_i-1)\times 1} \\ 1 \\ 0 \end{bmatrix} \mathbf{v}_{\mathbf{df}_i} \tag{5.65}$$

$$\mathbf{y}_{\mathbf{df}_i} = \begin{bmatrix} 1 & \mathbf{0}_{1\times r_i} \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi}_{\mathbf{df}_i} \\ \varsigma_{\mathbf{df}_i} \end{bmatrix} \tag{5.66}$$

5.2.3 Inner-Outer Control Loop using Feedback Linearization

The two solutions previously presented either applied Dynamic Feedback Linearization considering the position and yaw angle as outputs of the system, resulting in a single loop control structure, or applied Static Feedback Linearization to the altitude and attitude dynamics, originating a double loop control structure where only the innermost loop was designed based on this nonlinear control technique. As a way to deepen the study, an inner-outer loop control structure using Dynamic Feedback in the outermost loop and Static Feedback in the innermost is now derived. This approach is inspired by the work present in [39], where a control structure with an inner loop, responsible for the angle rates, and an outer loop, accountable for the position, velocity and acceleration, is developed. However, the variant of the feedback linearization resorted, dynamic inversion, is different from the differential geometry method that is addressed in this section. The dynamic inversion method constructs the integral inverse system for the controlled plant to obtain the decoupled linear system, whereas the input-output feedback linearization aims the attainment of an exact linearization of the system while resorting to tools such as the Lie algebra or differential manifold theory [61]. In fact, is possible to apply the Nonlinear Dynamic Inversion technique to an outer loop of relative degree 2 [33]. Nonetheless, since the second derivatives of the inertial position are not affine functions, is necessary to resort to simplifications, such as the Taylor series, to overcome this particularity. Alternatively, the third derivative of the position is an affine function and the outer-loop can be defined in order to provide references for the angle rates. This idea is explored in this third approach.

Let the vector of state-variables implied in the inner dynamics be denoted by

$$\mathbf{x_{ar}} = \begin{bmatrix} \dot{\varphi} & \dot{\theta} & \psi & \dot{\psi} \end{bmatrix}^T \tag{5.67}$$

the $\mathbf{u_{ar}}$ describe the vector of inputs

$$\mathbf{u_{ar}} = \begin{bmatrix} \tau_\varphi & \tau_\theta & \tau_\psi \end{bmatrix}^T \tag{5.68}$$

and $\mathbf{y_{ar}}$ represent the output vector of the angle rate dynamics

$$\mathbf{y_{ar}} = \begin{bmatrix} \dot{\varphi} & \dot{\theta} & \psi \end{bmatrix}^T \tag{5.69}$$

From (5.14), (5.15) and (5.16) is noted that the relative degree of the innermost dynamics is

$$\begin{bmatrix} r_1 & r_2 & r_3 \end{bmatrix}^T = \begin{bmatrix} 1 & 1 & 2 \end{bmatrix}^T \tag{5.70}$$

yielding

$$\sum_{j=1}^{3} r_j = 4 = n \tag{5.71}$$

Therefore, the inner dynamics are possible to be input-output feedback linearized by means of a static state feedback control law. The *decoupling matrix* $\mathbf{\Lambda(x_{ar})}$ is expressed by

$$\mathbf{\Lambda(x)} = \begin{bmatrix} \frac{1}{I_x} & \frac{\tan(\theta)\ \sin(\varphi)}{I_y} & \frac{\cos(\varphi)\ \tan(\theta)}{I_z} \\ 0 & \frac{\cos(\varphi)}{I_y} & -\frac{\sin(\varphi)}{I_z} \\ 0 & \frac{\sin(\varphi)}{I_y\ \cos(\theta)} & \frac{\cos(\varphi)}{I_z\ \cos(\theta)} \end{bmatrix} \tag{5.72}$$

Note that, as expected, the matrix $\mathbf{\Lambda(x)}$ is equal to the submatrix of $\mathbf{\Lambda(x_{in})}$ that refers to the attitude dynamics, i.e:

$$\mathbf{\Lambda(x)} = \mathbf{\Lambda}_{ij}\mathbf{(x_{in})} \text{ with } i,j = 2,3,4 \tag{5.73}$$

The determinant of $\mathbf{\Lambda(x)}$ is expressed by

$$\det\left(\mathbf{\Lambda(x)}\right) = \frac{1}{I_x\, I_y\, I_z\, \cos\left(\theta\right)} \tag{5.74}$$

Hence, the *decoupling matrix* $\mathbf{\Lambda(x)}$ is invertible at any point defined by $\frac{-\pi}{2} < \theta < \frac{\pi}{2}$. The static state feedback law

$$\mathbf{u_{ar}} = \boldsymbol{\alpha}(\mathbf{x}) + \boldsymbol{\beta}(\mathbf{x})\mathbf{v_{ar}} \tag{5.75}$$

has the matrix $\boldsymbol{\beta}(\mathbf{x})$ given by

$$\boldsymbol{\beta}(\mathbf{x}) = \boldsymbol{\beta}_{ij}\mathbf{(x_{in})} \text{ with } i,j = 2,3,4 \tag{5.76}$$

and the vector $\boldsymbol{\alpha}(\mathbf{x})$ defined by

$$\boldsymbol{\alpha}(\mathbf{x}) = \boldsymbol{\alpha}_i\mathbf{(x_{in})} \text{ with } i = 2,3,4 \tag{5.77}$$

The static state feedback law combined with the diffeomorphism defined by (4.45) leads to the trans-

formed state-variables:

$$\boldsymbol{\xi}_{\mathbf{ar}} = \mathbf{x}_{\mathbf{ar}} \tag{5.78}$$

Having the angle rate dynamics linearized and decoupled, the focus is now on the outer-loop dynamics. Let

$$\mathbf{x_p} = [x \;\; y \;\; z \;\; \dot{x} \;\; \dot{y} \;\; \dot{z} \;\; \ddot{x} \;\; \ddot{y}]^T \tag{5.79}$$

denote of the vector of state-variables present in the outer-loop dynamics,

$$\mathbf{u_p} = \begin{bmatrix} \dot{\varphi} & \dot{\theta} & T \end{bmatrix}^T \tag{5.80}$$

describe the vector of inputs and

$$\mathbf{y_p} = \begin{bmatrix} x & y & z \end{bmatrix}^T \tag{5.81}$$

represent the output vector of the translational dynamics. From the equations linear accelerations. Considering the inputs, the linear accelerations are only dependent on the thrust and the relative degree is $[r_1 \;\; r_2 \;\; r_3]^T = [2 \;\; 2 \;\; 2]^T$, which results in the sum of the relative degrees being inferior to the number of state-variables. Thus, a dynamic compensation must be included in order to perform the input-output linearization. Following a similar logic applied in subsection 5.2.2, an integrator is included on the thrust input channel. This incorporation origins a new state-variable, ε characterized by:

$$\begin{cases} \varepsilon = T \\ \dot{\varepsilon} = \bar{\mathbf{u}}_{\mathbf{p_1}} \end{cases} \tag{5.82}$$

where $\bar{\mathbf{u}}_{\mathbf{p_1}}$ denotes the first entry of the input vector of the extended system, $\bar{\mathbf{u}}_p$. The other inputs remain unchanged, yielding:

$$\begin{bmatrix} \bar{\mathbf{u}}_{\mathbf{p_1}} & \bar{\mathbf{u}}_{\mathbf{p_2}} & \bar{\mathbf{u}}_{\mathbf{p_3}} \end{bmatrix}^T = \begin{bmatrix} \dot{\varepsilon} & \dot{\varphi} & \dot{\theta} \end{bmatrix}^T \tag{5.83}$$

The vector of state-variables of the extended system is represented by:

$$\bar{\mathbf{x}}_{\mathbf{p}} = [x \;\; y \;\; z \;\; \dot{x} \;\; \dot{y} \;\; \dot{z} \;\; \ddot{x} \;\; \ddot{y} \;\; \varepsilon]^T \tag{5.84}$$

The linear accelerations are again expressed by (5.45), (5.46) and (5.47). Differentiating each one of the referred equations results in

$$\begin{aligned} x^{(3)} = & \frac{\varepsilon \, (\cos(\varphi) \, \sin(\psi) \, \dot{\varphi} - \cos(\psi) \, \sin(\theta) \, \sin(\varphi) \, \dot{\varphi})}{m} \\ & + \frac{\varepsilon \left(\cos(\psi) \, \sin(\varphi) \, \dot{\psi} + \cos(\theta) \, \cos(\varphi) \, \cos(\psi) \, \dot{\theta} \right)}{m} \\ & + \frac{\varepsilon \left(- \cos(\varphi) \, \sin(\theta) \, \sin(\psi) \, \dot{\psi} \right)}{m} \\ & + \frac{\dot{\varepsilon} \, (\sin(\varphi) \, \sin(\psi) + \cos(\varphi) \, \cos(\psi) \, \sin(\theta))}{m} \end{aligned} \tag{5.85}$$

$$
\begin{aligned}
y^{(3)} \;=\; & \frac{\varepsilon\,\left(\cos\left(\theta\right)\,\cos\left(\varphi\right)\,\sin\left(\psi\right)\,\dot{\theta} - \cos\left(\varphi\right)\,\cos\left(\psi\right)\,\dot{\varphi}\right)}{m} \\
& + \frac{\varepsilon\,\left(-\sin\left(\theta\right)\,\sin\left(\varphi\right)\,\sin\left(\psi\right)\,\dot{\varphi} + \sin\left(\varphi\right)\,\sin\left(\psi\right)\,\dot{\psi}\right)}{m} \\
& + \frac{\varepsilon\,\left(\cos\left(\varphi\right)\,\cos\left(\psi\right)\,\sin\left(\theta\right)\,\dot{\psi}\right)}{m} \\
& - \frac{\dot{\varepsilon}\,\left(\cos\left(\psi\right)\,\sin\left(\varphi\right) - \cos\left(\varphi\right)\,\sin\left(\theta\right)\,\sin\left(\psi\right)\right)}{m}
\end{aligned}
\tag{5.86}
$$

$$
z^{(3)} \;=\; \frac{\dot{\varepsilon}\,\cos\left(\theta\right)\,\cos\left(\varphi\right)}{m} - \frac{\cos\left(\varphi\right)\,\sin\left(\theta\right)\,\varepsilon\,\dot{\theta}}{m} - \frac{\cos\left(\theta\right)\,\sin\left(\varphi\right)\,\varepsilon\,\dot{\varphi}}{m} \tag{5.87}
$$

From the preceding expressions the *decoupling matrix* for the outer dynamics is derived

$$
\mathbf{\Lambda}(\mathbf{x},\varepsilon) = \begin{bmatrix}
\frac{\sin(\varphi)\sin(\psi)+\cos(\varphi)\cos(\psi)\sin(\theta)}{m} & \frac{\varepsilon\,(\cos(\varphi)\sin(\psi)-\cos(\psi)\sin(\theta)\sin(\varphi))}{m} & \frac{\cos(\theta)\cos(\varphi)\cos(\psi)\varepsilon}{m} \\
-\frac{\cos(\psi)\sin(\varphi)-\cos(\varphi)\sin(\theta)\sin(\psi)}{m} & -\frac{\varepsilon\,(\cos(\varphi)\cos(\psi)+\sin(\theta)\sin(\varphi)\sin(\psi))}{m} & \frac{\cos(\theta)\cos(\varphi)\sin(\psi)\,\varepsilon}{m} \\
\frac{\cos(\theta)\cos(\varphi)}{m} & -\frac{\cos(\theta)\sin(\varphi)\varepsilon}{m} & -\frac{\cos(\varphi)\sin(\theta)\varepsilon}{m}
\end{bmatrix}
\tag{5.88}
$$

The respective determinant is

$$
\det\left(\mathbf{\Lambda}(\mathbf{x},\varepsilon)\right) = \frac{\cos\left(\varphi\right)\,\varepsilon^2}{m^3} \tag{5.89}
$$

which translates to the matrix $\mathbf{\Lambda}(\mathbf{x},\varepsilon)$ being nonsingular for all the points characterized by $-\frac{\pi}{2} \leq \varphi \leq \frac{\pi}{2}$ and $\varepsilon \neq 0$. Once again, its stressed that the last condition can be guaranteed in the implementation by setting a positive non-zero initial value. The resulting dynamic state feedback law has the following form

$$
\bar{\mathbf{u}}_{\mathbf{p}} = \boldsymbol{\alpha}(\mathbf{x},\varepsilon) + \boldsymbol{\beta}(\mathbf{x},\varepsilon)\mathbf{v}_{\mathbf{p}} \tag{5.90}
$$

where the matrix $\boldsymbol{\beta}(\mathbf{x},\varepsilon)$ is represented as follow

$$
\boldsymbol{\beta}(\mathbf{x},\varepsilon) = \begin{bmatrix}
\boldsymbol{\beta}(\mathbf{x},\varepsilon)_{11} & \boldsymbol{\beta}(\mathbf{x},\varepsilon)_{12} & m\,\cos\left(\theta\right)\,\cos\left(\varphi\right) \\
\boldsymbol{\beta}(\mathbf{x},\varepsilon)_{21} & \boldsymbol{\beta}(\mathbf{x},\varepsilon)_{22} & -\frac{m\,\cos(\theta)\,\sin(\varphi)}{\varepsilon} \\
\frac{m\,\cos(\theta)\,\cos(\psi)}{\cos(\varphi)\,\varepsilon} & \frac{m\,\cos(\theta)\,\sin(\psi)}{\cos(\varphi)\,\varepsilon} & -\frac{m\,\sin(\theta)}{\cos(\varphi)\,\varepsilon}
\end{bmatrix}
\tag{5.91}
$$

with

$$
\boldsymbol{\beta}(\mathbf{x},\varepsilon)_{11} = m\,\left(\sin\left(\varphi\right)\,\sin\left(\psi\right) + \cos\left(\varphi\right)\,\cos\left(\psi\right)\,\sin\left(\theta\right)\right) \tag{5.92}
$$

$$
\boldsymbol{\beta}(\mathbf{x},\varepsilon)_{12} = m\,\cos\left(\varphi\right)\,\sin\left(\theta\right)\,\sin\left(\psi\right) - m\,\cos\left(\psi\right)\,\sin\left(\varphi\right) \tag{5.93}
$$

$$
\boldsymbol{\beta}(\mathbf{x},\varepsilon)_{21} = \frac{m\,\left(\cos\left(\varphi\right)\,\sin\left(\psi\right) - \cos\left(\psi\right)\,\sin\left(\theta\right)\,\sin\left(\varphi\right)\right)}{\varepsilon} \tag{5.94}
$$

$$
\boldsymbol{\beta}(\mathbf{x},\varepsilon)_{22} = -\frac{m\,\left(\cos\left(\varphi\right)\,\cos\left(\psi\right) + \sin\left(\theta\right)\,\sin\left(\varphi\right)\,\sin\left(\psi\right)\right)}{\varepsilon} \tag{5.95}
$$

where the vector $\boldsymbol{\alpha}(\mathbf{x}, \varepsilon)$ is represented by

$$\boldsymbol{\alpha}(\mathbf{x}, \varepsilon) = \begin{bmatrix} 0 \\ \sin(\theta)\ \dot{\psi} \\ -\cos(\theta)\ \tan(\varphi)\ \dot{\psi} \end{bmatrix} \tag{5.96}$$

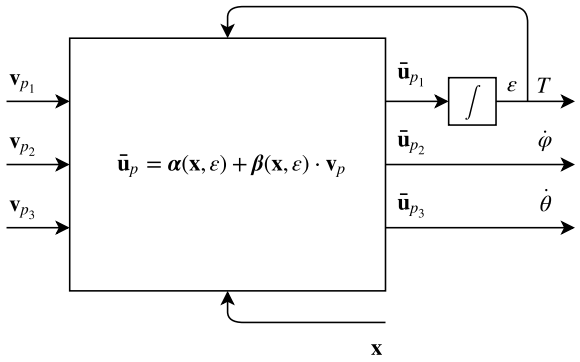

Figure 5.9: Block diagram representation of the resulting dynamics after applying dynamic feedback linearization to the translational dynamics.

The dynamic state feedback law, depicted in Fig. 5.9, combined with the diffeomorphism defined by (4.45) leads to the transformed state-variables:

$$\boldsymbol{\xi}_{\mathbf{p}} = \boldsymbol{\Phi}(\mathbf{x}_{\mathbf{p}}) = \begin{bmatrix} x & \dot{x} & \ddot{x} & y & \dot{y} & \ddot{y} & z & \dot{z} & \ddot{z} \end{bmatrix}^T \tag{5.97}$$

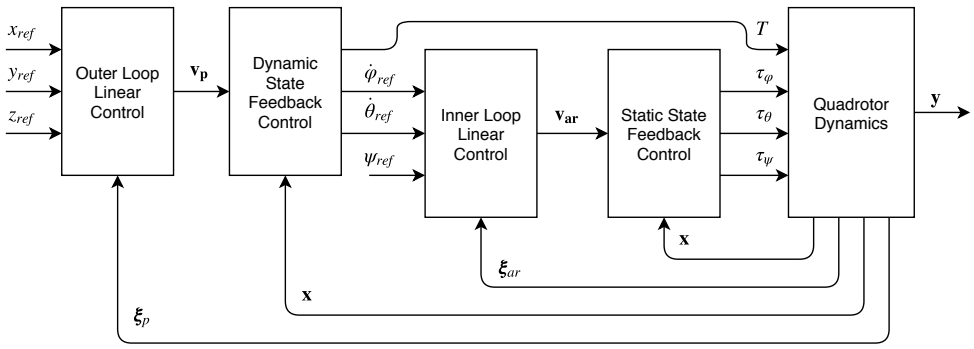

Figure 5.10: Block diagram representation of the control structure derived for the third approach using Feedback Linearization.

The third approach using Feedback Linearization is schematized in Fig. 5.10. Once again, an integrator is incorporated into the feedback control of the angular subsystems, obtaining the following state-space representation for the inner subsystems:

$$\begin{bmatrix} \dot{\boldsymbol{\xi}}_{\mathbf{ar}_i} \\ \dot{\varsigma}_{\mathbf{ar}_i} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{(r_i-1)\times 1} & \boldsymbol{I}_{r_i-1} & \mathbf{0}_{(r_i-1)\times 1} \\ 0 & \mathbf{0}_{1\times(r_i-1)} & 0 \\ -1 & \mathbf{0}_{1\times(r_i-1)} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi}_{\mathbf{ar}_i} \\ \varsigma_{\mathbf{ar}_i} \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{(r_i-1)\times 1} \\ 1 \\ 0 \end{bmatrix} \mathbf{v}_{\mathbf{ar}i} \tag{5.98}$$

$$\mathbf{y}_{\mathbf{ar}i} = \begin{bmatrix} 1 & \mathbf{0}_{1\times r_i} \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi}_{\mathbf{ar}_i} \\ \varsigma_{\mathbf{ar}_i} \end{bmatrix} \tag{5.99}$$

The integrative action is also included in the outer subsystems, yielding the following state-space representation for each position subsystem:

$$\begin{bmatrix} \dot{\boldsymbol{\xi}}_{\mathbf{p}_i} \\ \dot{\varsigma}_{\mathbf{p}_i} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{(r_i-1)\times 1} & \boldsymbol{I}_{r_i-1} & \mathbf{0}_{(r_i-1)\times 1} \\ 0 & \mathbf{0}_{1\times(r_i-1)} & 0 \\ -1 & \mathbf{0}_{1\times(r_i-1)} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi}_{\mathbf{p}_i} \\ \varsigma_{\mathbf{p}_i} \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{(r_i-1)\times 1} \\ 1 \\ 0 \end{bmatrix} \mathbf{v}_{\mathbf{p}i} \tag{5.100}$$

$$\mathbf{y}_{\mathbf{p}_i} = \begin{bmatrix} 1 & \mathbf{0}_{1\times r_i} \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi}_{\mathbf{p}_i} \\ \varsigma_{\mathbf{p}_i} \end{bmatrix} \tag{5.101}$$

Chapter 6

Estimation

In fact, the optimal control method considered relies on full state-feedback. The reconstruction of the state variables that are not directly available through sensors is performed resorting to estimators. Therefore, in this chapter, the theoretical basis of the Kalman filter and the formulation of the Madgwick attitude filter [65] and of the nonlinear attitude filter proposed by João Madeiras [66] are addressed.

6.1 Discrete Kalman Filter

The Kalman filter constitutes a solution to optimal stochastic estimation problem for linear systems and aims the minimization of the estimation mean-squared-error under the assumptions of zero-mean Gaussian process and sensor noises. It was first proposed by Rudolf Kalman [62] in 1960 and is a recursive estimator since only the estimated state from the previous time step and the current measurement are needed to compute the estimate for the current state.

Considering the discrete-time state-space representation for a linear time-invariant system with noise

$$\mathbf{x_{k+1}} = \mathbf{F}\mathbf{x_k} + \mathbf{G}\mathbf{u_k} + \boldsymbol{\vartheta} \tag{6.1}$$

$$\mathbf{y_k} = \mathbf{H}\mathbf{x_k} + \boldsymbol{\nu} \tag{6.2}$$

where $\boldsymbol{\vartheta}$ is the process noise that models the uncertainty in the system and that affects the evolution of the states, and $\boldsymbol{\nu}$ is the sensor noise that models uncertainty in the measurements and that impacts the states measurements [40]. It is typically assumed that $\boldsymbol{\vartheta}$ and $\boldsymbol{\nu}$ are zero mean ($\mathsf{E}[\boldsymbol{\vartheta}] = 0$, $\mathsf{E}[\boldsymbol{\nu}] = 0$) and uncorrelated Gaussian white random noises so that:

$$\boldsymbol{\vartheta} \sim \mathcal{N}\left(0, {}^{\mathbf{K}}\mathbf{Q}\right) \tag{6.3}$$

$$\boldsymbol{\nu} \sim \mathcal{N}\left(0, {}^{\mathbf{K}}\mathbf{R}\right) \tag{6.4}$$

$$\mathsf{E}\left[\boldsymbol{\vartheta\nu}\right] = 0 \tag{6.5}$$

with the matrices ${}^{\mathbf{K}}\mathbf{Q}$ and ${}^{\mathbf{K}}\mathbf{R}$ being, respectively, positive semi-definite and positive definite. With this assumption, an optimal estimator can be developed while aiming the balance of the effect of the various types of random noise in the system on the estimation error. The Kalman filter gains result from an optimization process that targets the reduction of this error. The algorithm to compute the steady-state optimal Kalman gain $\mathbf{K}_{\mathbf{k}}^{\mathbf{e}}$ is initialized by defining the initial values of the state vector $\mathbf{x}_{\mathbf{0|0}}$ and the covariance matrix $\mathbf{P}_{\mathbf{0|0}}$ and can be divided in two phases [63]. In the first phase, the predicted state-estimate $\hat{\mathbf{x}}_{\mathbf{k|k-1}}$ and the predicted error covariance matrix $\mathbf{P}_{\mathbf{k|k-1}}$ are determined. The values are computed *a priori*, i.e., using the data available at the instant $k-1$. The prediction phase has the following equations:

$$\hat{\mathbf{x}}_{\mathbf{k|k-1}} = \mathbf{F}\hat{\mathbf{x}}_{\mathbf{k-1|k-1}} + \mathbf{G}\mathbf{u}_{\mathbf{k}} \tag{6.6}$$

$$\mathbf{P}_{\mathbf{k|k-1}} = \mathbf{F}\mathbf{P}_{\mathbf{k-1|k-1}}\mathbf{F}^{T} + {}^{\mathbf{K}}\mathbf{Q} \tag{6.7}$$

The second part of this recursive estimator is the update phase where the residue of the previous predictions, $\tilde{\mathbf{y}}_{\mathbf{k|k-1}}$ and $\mathbf{S}_{\mathbf{k}}$, respectively, the Kalman gain vector $\mathbf{K}_{\mathbf{k}}^{\mathbf{e}}$, and the state vector and covariance matrix estimations *a posteriori*, $\hat{\mathbf{x}}_{\mathbf{k|k}}$ and $\mathbf{P}_{\mathbf{k|k}}$ (i.e. using the data available at the present instant k) are calculated. The update phase is formed by the ensuing equations:

$$\tilde{\mathbf{y}}_{\mathbf{k}} = \mathbf{y}_{\mathbf{k}} - \mathbf{H}\hat{\mathbf{x}}_{\mathbf{k|k-1}} \tag{6.8}$$

$$\mathbf{S}_{\mathbf{k}} = \mathbf{H}\mathbf{P}_{\mathbf{k|k-1}}\mathbf{H}^{T} + {}^{\mathbf{K}}\mathbf{R} \tag{6.9}$$

$$\mathbf{K}_{\mathbf{k}}^{\mathbf{e}} = \mathbf{P}_{\mathbf{k|k-1}}{\mathbf{H}_{\mathbf{k}}}^{T}{\mathbf{S}_{\mathbf{k}}}^{-1} \tag{6.10}$$

$$\hat{\mathbf{x}}_{\mathbf{k|k}} = \hat{\mathbf{x}}_{\mathbf{k|k-1}} + \mathbf{K}_{\mathbf{k}}^{\mathbf{e}}\tilde{\mathbf{y}}_{\mathbf{k}} \tag{6.11}$$

$$\mathbf{P}_{\mathbf{k|k}} = \left(I - \mathbf{K}_{\mathbf{k}}^{\mathbf{e}}\mathbf{H}_{\mathbf{k}}\right)\mathbf{P}_{\mathbf{k|k-1}}\left(I - \mathbf{K}_{\mathbf{k}}^{\mathbf{e}}\mathbf{H}_{\mathbf{k}}\right)^{T} + \mathbf{K}_{\mathbf{k}}^{\mathbf{e}}\mathbf{R}{\mathbf{K}_{\mathbf{k}}^{\mathbf{e}}}^{T} \tag{6.12}$$

The performance index is given by the trace of the estimation error covariance matrix $\mathbf{P}_{\mathbf{k|k}}$:

$$J = \mathsf{tr}\left(\mathbf{P}_{\mathbf{k|k}}\right) = \mathsf{tr}\left(E\left[\mathbf{e}_{\mathbf{k}}^{\mathbf{x}}{\mathbf{e}_{\mathbf{k}}^{\mathbf{x}}}^{T}\right]\right) \tag{6.13}$$

where $\mathbf{e}_{\mathbf{k}}^{\mathbf{x}}$ denotes the state vector reconstruction stochastic error at instant k

$$\mathbf{e}^{\mathbf{x}}_{\mathbf{k+1}} = \mathbf{x_k} - \hat{\mathbf{x}}_{\mathbf{k|k}} \tag{6.14}$$

The equations (6.6) to (6.12) describe the discrete Kalman Filter recursive algorithm for a system represented by the state-space equations (6.1) and (6.2). Since the system considered is time invariant, the Kalman gain vector $\mathbf{K^e}$ will converge to a given value.

In order to use Kalman filters, the system for which they are being designed is required to be observable since the sensory data must contain information regarding all states. The structure of the filter is depicted in Fig. 6.1.

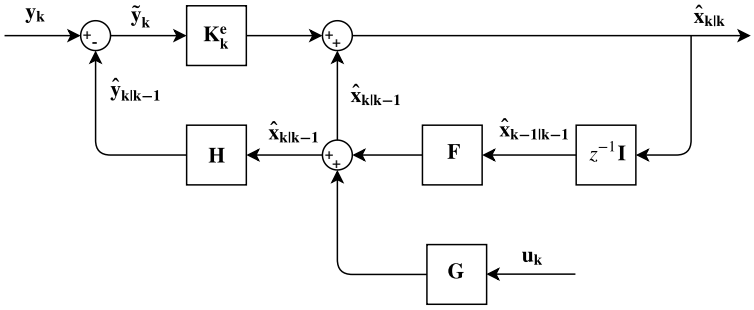

Figure 6.1: Structure of the Discrete Kalman Filter.

6.1.1 Complementary Filters Design

The estimation theory formerly addressed will be presently considered in the design of the velocity and jerk estimators. To this end, complementary filters are used. These filters are independent of the dynamics of the system since they rely on the kinematic model to establish a relation between the sensory data and the state-variables that are intended to estimate. Therefore, these filters are a particularly useful tool that has the advantage of not being affected by unmodelled dynamics or unknown parameters.

Linear Velocity Estimation

The following discrete-time kinematic equations describe the position and the velocity of the vehicle along the direction of $\vec{a}_1$:

$$x_{k+1} = x_k + \dot{x}_k \Delta t + \frac{1}{2} a_{x_k} (\Delta t)^2 \tag{6.15}$$

$$\dot{x}_{k+1} = \dot{x}_k + a_{x_k} \Delta t \tag{6.16}$$

Let $\mathbf{x}_k = [x_k \;\; \dot{x}_k]$ define the state variables vector. The former equations lead to the ensuing discrete-time state-space representation:

$$\mathbf{x}_{k+1} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \mathbf{x}_k + \begin{bmatrix} \frac{(\Delta t)^2}{2} \\ \Delta t \end{bmatrix} {}^{I}a_{x_k} \tag{6.17}$$

Assuming that, between the instant $k-1$ and the instant k, uncontrolled forces cause a constant acceleration ${}^{I}a_{x_k}$ that has a normal distribution, characterized by zero mean and variance ϑ, in order to model the acceleration effect as white random process noise, leads to the following representation

$$\mathbf{x}_{k+1} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \mathbf{x}_k + \vartheta \tag{6.18}$$

The previous assumption is referred in the theory as the white-noise acceleration model [64]. Since only the position is available through sensory data, the output vector is defined as

$$\mathbf{y}_k = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x}_k + \nu \tag{6.19}$$

Note that the process noise ϑ and the sensor noise ν considered have the normal distribution described in (6.3) and (6.4), respectively. The kinematic model presented was derived for the velocity along the direction described by $\vec{a}_1$. Nonetheless, for the remaining inertial velocities, an equivalent model can be used. Furthermore, by applying the required rotation to the inertial velocities, the body velocities ${}^{B}\widehat{\mathbf{v}}$ are computed:

$${}^{B}\widehat{\mathbf{v}} = \begin{bmatrix} \widehat{u} \\ \widehat{v} \\ \widehat{w} \end{bmatrix} = {}^{B}\mathbf{R}_I \begin{bmatrix} \widehat{\dot{x}} \\ \widehat{\dot{y}} \\ \widehat{\dot{z}} \end{bmatrix} \tag{6.20}$$

Jerk Estimation

The estimate of the jerk follows a similar logic to the one applied in the estimation of the linear velocities. The acceleration in the inertial frame is computed by removing the effect of the centripetal acceleration from the measurements provided by the accelerometer and by, subsequently, applying the rotation to the inertial frame:

$$\widehat{\ddot{\mathbf{x}}} = {}^{I}\widehat{\mathbf{a}} = {}^{I}\mathbf{R}_B \left({}^{B}\mathbf{a} + \boldsymbol{\omega} \times {}^{B}\widehat{\mathbf{v}} \right) \tag{6.21}$$

Thus, from sensors, one has access to the integral of the state-variable that intends to estimate. The effect of the fourth derivative of the position can be modeled as random white process noise. Hence, letting $\mathbf{x}_k = \begin{bmatrix} \ddot{x}_k & x_k^{(3)} \end{bmatrix}$ define the state variables vector, considering the following discrete model can be used to estimate the third derivative of the x coordinate:

$$\mathbf{x}_{k+1} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \mathbf{x}_k + \vartheta \tag{6.22}$$

$$\mathbf{y}_k = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x}_k + \nu \tag{6.23}$$

A similar model is used for the estimation of $y^{(3)}$ and $z^{(3)}$.

6.2 Attitude Filters

6.2.1 Madgwick Attitude Filter

In fact, the Kalman filter is widely used for sensor fusion to estimate the attitude, however, for practical reasons, the orientation filter developed by [65] was opted.

The computation is performed using the quaternion representation and fuses the measurements of the gyroscope, accelerometer and magnetometer through an optimized gradient-descent algorithm.

The attitude filter can be summarized through the following expressions:

$$\widehat{\boldsymbol{q}}_k = \widehat{\boldsymbol{q}}_{k-1} + \dot{\widehat{\boldsymbol{q}}}_k \Delta t \tag{6.24}$$

$$\dot{\widehat{\boldsymbol{q}}}_k = {}^{\omega}\dot{\widehat{\boldsymbol{q}}}_k - \beta {}^{e}\dot{\widehat{\boldsymbol{q}}}_k \tag{6.25}$$

$${}^{e}\dot{\widehat{\boldsymbol{q}}}_k = \frac{\nabla \boldsymbol{f}}{\|\nabla \boldsymbol{f}\|} \tag{6.26}$$

where $\widehat{\boldsymbol{q}}_k$ represents the estimated orientation at instant k, β is a tuning parameter, that can be interpreted as the magnitude of the gyroscope measurement error, and ${}^{e}\dot{\widehat{\boldsymbol{q}}}_k$ corresponds to the direction of the error of $\dot{\widehat{\boldsymbol{q}}}_k$, determined from the accelerometer and magnetometer measurements at instant k, and, together with the constant β, define a corrective step. The filter computations start with $\widehat{\boldsymbol{q}}_0 = [1 \ 0 \ 0 \ 0]^T$. The rate of change of orientation estimated from the gyroscope measurement at instant k, ${}^{\omega}\dot{\widehat{\boldsymbol{q}}}_k$, is obtained with the following formula

$${}^{\omega}\dot{\widehat{\boldsymbol{q}}}_k = \frac{1}{2}\widehat{\boldsymbol{q}}_{k-1} \otimes {}^{S}\boldsymbol{\omega}_k \tag{6.27}$$

in which ${}^{S}\boldsymbol{\omega}_k = [0 \ p \ q \ r]^T$ considers the angular velocities at instant k and the operator $\otimes$ denotes a product of quaternions. The direction of the error of $\dot{\widehat{\boldsymbol{q}}}_k$ is computed through a gradient descent algorithm. Thus, results from the normalization of the gradient of a solution surface that is defined by the objective function and its Jacobian. The optimization problem aims the minimization of this objective function that defines the difference between a predefined reference direction of the field in the inertial frame oriented according $\widehat{\boldsymbol{q}}_{k-1}$ and the measured direction of the field in the body-fixed frame. The gradient of the objective function is computed through:

$$\nabla \boldsymbol{f} = \boldsymbol{J}_{g,d}^{T}\left(\widehat{\boldsymbol{q}}_{k-1}, \widehat{\boldsymbol{d}}_k\right) \boldsymbol{f}_{g,d}\left(\widehat{\boldsymbol{q}}_{k-1}, {}^{S}\mathbf{a}_k, \widehat{\boldsymbol{d}}_k, {}^{S}\mathbf{m}_k\right) \tag{6.28}$$

where

$$\boldsymbol{J}_{g,d}^{T}\left(\widehat{\boldsymbol{q}}_{k-1},\widehat{\boldsymbol{d}}_{k}\right)=\begin{bmatrix}\boldsymbol{J}_{g}^{T}\left(\widehat{\boldsymbol{q}}_{k-1}\right)\\ \boldsymbol{J}_{d}^{T}\left(\widehat{\boldsymbol{q}}_{k-1},\widehat{\boldsymbol{d}}_{k}\right)\end{bmatrix} \tag{6.29}$$

$$\boldsymbol{f}_{g,d}\left(\widehat{\boldsymbol{q}}_{k-1},{}^{S}\mathbf{a}_{k},\widehat{\boldsymbol{d}}_{k},{}^{S}\mathbf{m}_{k}\right)=\begin{bmatrix}\boldsymbol{f}_{g}\left(\widehat{\boldsymbol{q}}_{k-1},{}^{S}\mathbf{a}_{k}\right)\\ \boldsymbol{f}_{d}\left(\widehat{\boldsymbol{q}}_{k-1},\widehat{\boldsymbol{d}}_{k},{}^{S}\mathbf{m}_{k}\right)\end{bmatrix}=\begin{bmatrix}\widehat{\boldsymbol{q}}_{k-1}^{*}\otimes{}^{E}\boldsymbol{g}\otimes\widehat{\boldsymbol{q}}_{k-1}-{}^{S}\mathbf{a}_{k}\\ \widehat{\boldsymbol{q}}_{k-1}^{*}\otimes\widehat{\boldsymbol{d}}_{k}\otimes\widehat{\boldsymbol{q}}_{k-1}-{}^{S}\mathbf{m}_{k}\end{bmatrix} \tag{6.30}$$

with ${}^{E}\boldsymbol{g}=[0\ 0\ 0\ 1]$ denoting the direction of gravity, ${}^{S}\mathbf{a}_{k}=\begin{bmatrix}0 & {}^{B}\bar{a}_{x} & {}^{B}\bar{a}_{y} & {}^{B}\bar{a}_{z}\end{bmatrix}$ being formed by the normalized accelerometer measurements at instant k, ${}^{S}\mathbf{m}_{k}=[0\ \bar{m}_{x}\ \bar{m}_{y}\ \bar{m}_{z}]$ being constituted by the normalized magnetometer measurements at instant k and $\hat{\boldsymbol{d}}_{k}=[0\ d_{x}\ 0\ d_{z}]$ representing the resulting normalized magnetometer measurements vector after compensating for magnetic distortion. This compensation resorts to the following expressions:

$$\hat{\boldsymbol{h}}_{k}=[0\ h_{x}\ h_{y}\ h_{z}]=\widehat{\boldsymbol{q}}_{k-1}\otimes{}^{S}\mathbf{m}_{k}\otimes\widehat{\boldsymbol{q}}_{k-1}^{*} \tag{6.31}$$

$$\hat{\boldsymbol{d}}_{k}=\begin{bmatrix}0 & \sqrt{h_{x}^{2}+h_{y}^{2}} & 0 & h_{z}\end{bmatrix} \tag{6.32}$$

Note that, provided that $d_{x}\neq 0$, the solution surface created by the objective function expressed in (6.30) has a minimum defined by a single point. This objective function and its Jacobian are computed through the next equations:

$$\boldsymbol{f}_{g,d}\left(\widehat{\boldsymbol{q}}_{k-1},{}^{S}\mathbf{a}_{k},\widehat{\boldsymbol{d}}_{k},{}^{S}\mathbf{m}_{k}\right)=\begin{bmatrix}2\left(q_{2}q_{4}-q_{1}q_{3}\right)-{}^{B}\bar{a}_{x}\\ 2\left(q_{1}q_{2}+q_{3}q_{4}\right)-{}^{B}\bar{a}_{y}\\ 2\left(\frac{1}{2}-q_{2}^{2}-q_{3}^{2}\right)-\\ {}^{B}\bar{a}_{z}2d_{x}\left(0.5-q_{3}^{2}-q_{4}^{2}\right)+2d_{z}\left(q_{2}q_{4}-q_{1}q_{3}\right)-\bar{m}_{x}\\ 2d_{x}\left(q_{2}q_{3}-q_{1}q_{4}\right)+2d_{z}\left(q_{1}q_{2}+q_{3}q_{4}\right)-\bar{m}_{y}\\ 2d_{x}\left(q_{1}q_{3}+q_{2}q_{4}\right)+2d_{z}\left(0.5-q_{2}^{2}-q_{3}^{2}\right)-\bar{m}_{z}\end{bmatrix} \tag{6.33}$$

$$\boldsymbol{J}_{g,d}^{T}\left(\widehat{\boldsymbol{q}}_{k-1},\widehat{\boldsymbol{d}}_{k}\right)=\begin{bmatrix}-2q_{3} & 2q_{2} & 0\\ 2q_{4} & 2q_{1} & -4q_{2}\\ -2q_{1} & 2q_{4} & -4q_{3}\\ 2q_{2} & 2q_{3} & 0\\ -2d_{z}q_{3} & -2d_{x}q_{4}+2d_{z}q_{2} & 2d_{x}q_{3}\\ 2d_{z}q_{4} & 2d_{x}q_{3}+2d_{z}q_{1} & 2d_{x}q_{4}-4d_{z}q_{2}\\ -4d_{x}q_{3}-2d_{z}q_{1} & 2d_{x}q_{2}+2d_{z}q_{4} & 2d_{x}q_{1}-4d_{z}q_{3}\\ -4d_{x}q_{4}+2d_{z}q_{2} & -2d_{x}q_{1}+2d_{z}q_{3} & 2d_{x}q_{2}\end{bmatrix}^{T} \tag{6.34}$$

where q_{1}, q_{2}, q_{3} and q_{4} are the entries of the vector $\widehat{\boldsymbol{q}}_{k-1}$. The structure of the Madgwick Attitude Filter is squematized in Fig. 6.2.

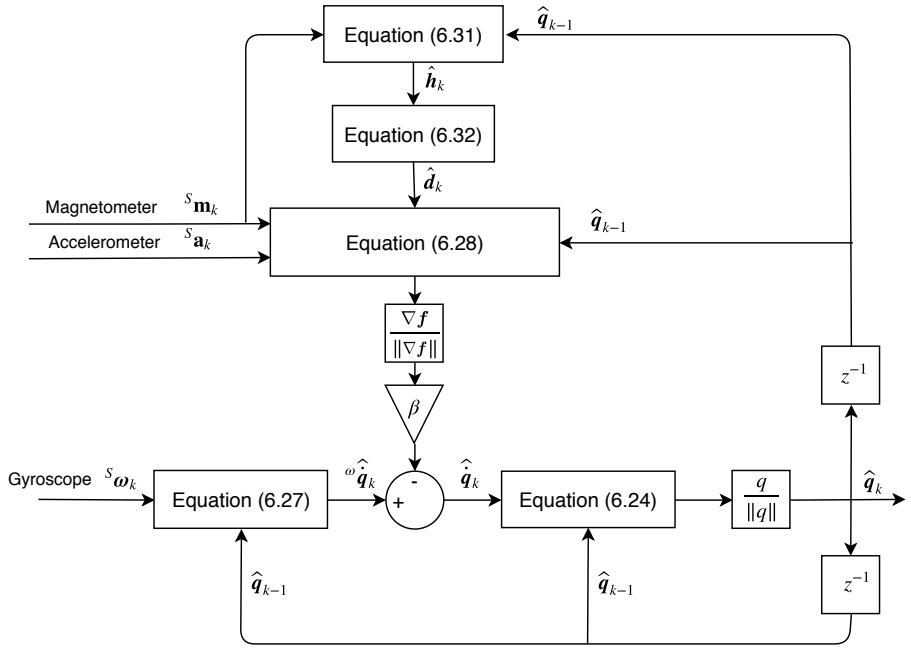

Figure 6.2: Structure of the Madgwick Attitude Filter.

6.2.2 Nonlinear Attitude Filter

The second filter considered, developed by João Madeiras [66], is a nonlinear filter that fuses the accelerometer and gyroscope measurements. The formulation is presented in the following equation:

$$\begin{aligned}\begin{bmatrix}\hat{\boldsymbol{\eta}}\\ \hat{\boldsymbol{b}}_{\boldsymbol{\omega}1}\\ \hat{\boldsymbol{b}}_{\boldsymbol{\omega}2}\end{bmatrix}_{k+1} &= \begin{bmatrix}\boldsymbol{I} & -\Delta t\mathbf{T}\left(\boldsymbol{\eta}_k\right) & -\Delta t\mathbf{T}\left(\boldsymbol{\eta}_k\right)\\ \mathbf{0} & \boldsymbol{I}-\boldsymbol{T}_{\boldsymbol{BI}}^{-\mathbf{1}}\Delta t & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \boldsymbol{I}\end{bmatrix}\begin{bmatrix}\hat{\boldsymbol{\eta}}\\ \hat{\boldsymbol{b}}_{\boldsymbol{\omega}1}\\ \hat{\boldsymbol{b}}_{\boldsymbol{\omega}2}\end{bmatrix}_k + \begin{bmatrix}\Delta t\mathbf{T}\left(\boldsymbol{\eta}_k\right)\\ \mathbf{0}\\ \mathbf{0}\end{bmatrix}\boldsymbol{\omega}_{\mathbf{k}}\\ &+ \begin{bmatrix}\mathbf{T}\left(\boldsymbol{\eta}_k\right)\left(\boldsymbol{K}_{\mathbf{1}\boldsymbol{\eta}}-\boldsymbol{I}\right)+\mathbf{T}\left(\boldsymbol{\eta}_{k-1}\right)\\ \boldsymbol{K}_{\mathbf{2}\boldsymbol{\eta}}\\ \boldsymbol{K}_{\mathbf{3}\boldsymbol{\eta}}\end{bmatrix}\mathbf{T}^{-\mathbf{1}}\left(\boldsymbol{\eta}_{k-1}\right)\left(\boldsymbol{\eta}_k-\hat{\boldsymbol{\eta}}_k\right)\end{aligned} \tag{6.35}$$

where $\hat{\boldsymbol{b}}_{\boldsymbol{\omega}1}$ and $\hat{\boldsymbol{b}}_{\boldsymbol{\omega}2}$ denote sensor bias vectors correspondent to the angular velocities $\boldsymbol{\omega}$, $\boldsymbol{K}_{\mathbf{1}\boldsymbol{\eta}}$, $\boldsymbol{K}_{\mathbf{2}\boldsymbol{\eta}}$ and $\boldsymbol{K}_{\mathbf{3}\boldsymbol{\eta}}$ are diagonal matrices whose entries are the steady-state Kalman gains and $\boldsymbol{T}_{\boldsymbol{BI}}$ is a diagonal matrix whose entries correspond to the correlation time of each gyroscope axis bias instability noise. The gains of this attitude complementary filter can be computed considering a linear system that is obtained resorting to a Lyapunov transformation. These gains are optimal for small angular variations. Nonetheless, for more aggressive maneuvers, the trace of the error covariance matrix does not deviate much from the obtained when computing the optimal gains for each instant. Furthermore, this filter is proved to be uniformly asymptotically stable assuming a bounded pitch angle $\left(|\theta| < \frac{\pi}{2}\right)$.

Chapter 7

Simulation Results

In the last chapters, the control and estimation strategies were mainly addressed from a theoretical point of view. In order to forecast the performance of the approaches presented in the actual quadcopter, simulation tests should be performed.

The prior validation of the referred strategies in the simulation is crucial since it allows to study the impact of uncertainties and perturbations, such as external disturbances, saturation, measurement noise or inaccuracy of the model parameters, that might significantly affect the closed-loop behavior. Furthermore, possible adjustments or corrections are more easily identified in the simulation than during the testing in the actual quadcopter. Moreover, a comparison between the performance of the different methods can be made in simulation, which allows the selection of the best methods and the avoidance of the time-consuming task of implement all in the real system. Finally, implementation errors not related to the mathematical basis of the estimation and control solutions must be debugged before advancing to the experimental tests.

For these reasons, a simulation model of the quadcopter was developed in furtherance of studying the approaches and determining the most promising solutions beforehand in order to select them to be tested experimentally. The computation of the PWM commands (Appendix A) is also included as a way to consider the limitations imposed by saturation.

This chapter is organized in the following manner: firstly the control gains are obtained by analyzing the characteristics of the resulting step-responses, namely the settling time, the overshoot and the steady-state position error; secondly, to ascertain the robustness of the proposed controllers and its model dependency, the impact of considering inaccurate values of the mass and the inertia of the quadcopter is studied; subsequently, the gains of the complementary filters are computed while taking into account the estimation error; lastly, noise is inserted in the sensory data and the capacity of tracking a predefined trajectory is evaluated. The attitude filters are evaluated by computing the estimation error obtained during the trajectory tracking.

The simulation model with both the estimation and control solutions follow the structure depicted in Fig. 7.1. The selected sampling time for the simulation is 0.01 seconds. The relevant physical quantities of the quadcopter consider in the simulation are detailed in Table. 7.1.

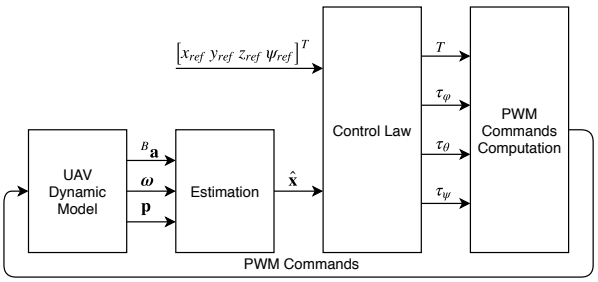

Figure 7.1: Block diagram representation of the simulation model implemented.

Concerning the design parameters, it is intended to control the position of the UAV while focusing in obtain responses with a settling time inferior to 6 seconds, with a maximum overshoot and oscillation inferior to 2 % and without steady-state position error. Regarding the rotational responses, since they are responsible for the stabilization of the quadcopter, a faster response, in furtherance of coping with the references that result from the translational control in the x and y directions of the inertial frame, is required. Therefore, the pitch/pitch rate and roll/roll rate subsystems must necessarily present step responses with a settling time inferior to 1 second and fulfill the formerly defined overshoot, oscillation and error requirements. Considering the yaw subsystem, once it does not present such an important role in the stabilization of the drone as the other Euler angles, its response is not required to have this settling time. Therefore, for this Euler angle, the aim is a settling time inferior to 3 seconds and the already defined null static-error and overshoot and oscillation inferior to 2%. For the estimation approaches, the objective is to minimize the absolute error of the estimation.

L (m)	m (kg)	I_x (kg $\cdot$ m^2)	I_y (kg $\cdot$ m^2)	I_z (kg $\cdot$ m^2)
0.127	0.460	2.24×10^{-3}	2.90×10^{-3}	5.30×10^{-3}

Table 7.1: Important physical quantities of the quadcopter considered in the simulation model

7.1 Step Responses

7.1.1 Linear Quadratic Regulator with Integrative Action

Single Loop Structure

Concerning the single-loop architecture that relies solely on linear controllers, the step-responses obtained with the $\mathbf{Q}$ and $\mathbf{R}$ matrices detailed in Table 7.2 are displayed in Fig. 7.2.

Subsystem	$\mathbf{Q}$	$\mathbf{R}$
Height	diag$(5, 1, 3)$	3
X Inertial	diag$(3, 1, 1, 10, 2)$	25
Y Inertial	diag$(3, 1, 1, 10, 2)$	25
Yaw	diag$(7, 1, 15)$	1

Table 7.2: $\mathbf{Q}$ and $\mathbf{R}$ matrices used in the optimal gains computation for each subsystem.

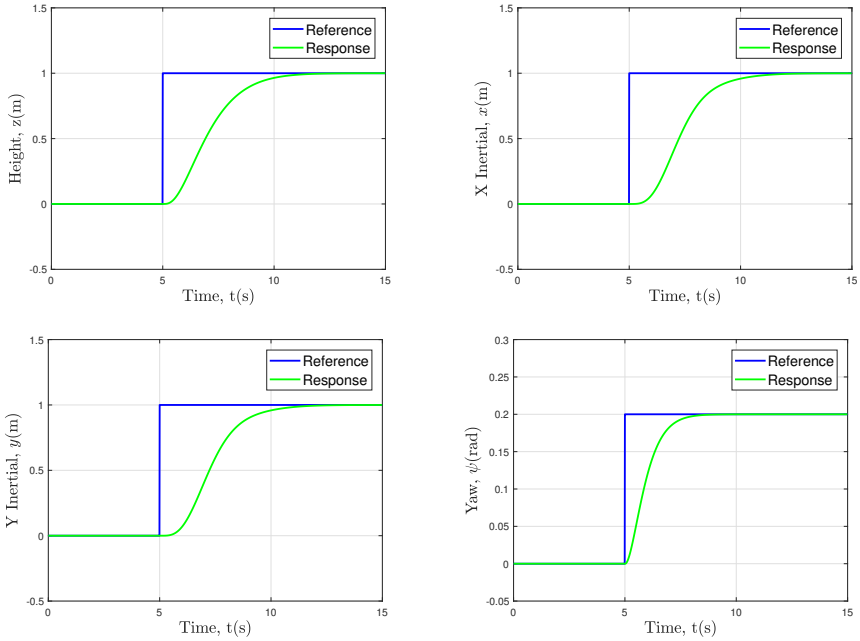

Figure 7.2: Step-responses obtained in simulation with the controllers designed for the single loop approach using LQR. From left to right, top to bottom: (a) Height; (b) X Inertial; (c) Y Inertial; (d) Yaw.

These responses and the Table 7.3 allow to state that the control objectives outlined were achieved.

Subsystem	Overshoot, M_p (%)	Settling time, t_s (s)
Height	0.00	5.62
X Inertial	0.00	5.76
Y Inertial	0.00	5.78
Yaw	0.00	2.81

Table 7.3: Overshoot, M_p, and settling time, t_s, of the step responses of each subsystem considered in the first linear approach.

Concerning the stability, from the Table 7.4 one notes that all eigenvalues are inside the unit circle and therefore the closed loop systems are stable. The actuation required during the step responses is presented in Fig. 7.3.

Subsystem	Closed-Loop Eigenvalues
Height	$\{0.9875 \pm 0.0104i,\ 0.9920\}$
X Inertial	$\{0.0402,\ 0.9887 \pm 0.0153i,\ 0.9892 \pm 0.0006i\}$
Y Inertial	$\{0.0151,\ 0.9887 \pm 0.0153i,\ 0.9892 \pm 0.0006i\}$
Yaw	$\{0.0938,\ 0.9810 \pm 0.0042i\}$

Table 7.4: Discrete time closed-loop eigenvalues of each subsystem considered in the first linear approach.

Inner-Outer Loop Structure

Comparatively to the first architecture derived, the control design is equal for the height and the yaw subsystems and, therefore, will not be addressed again. The obtainment of the gains for this architecture

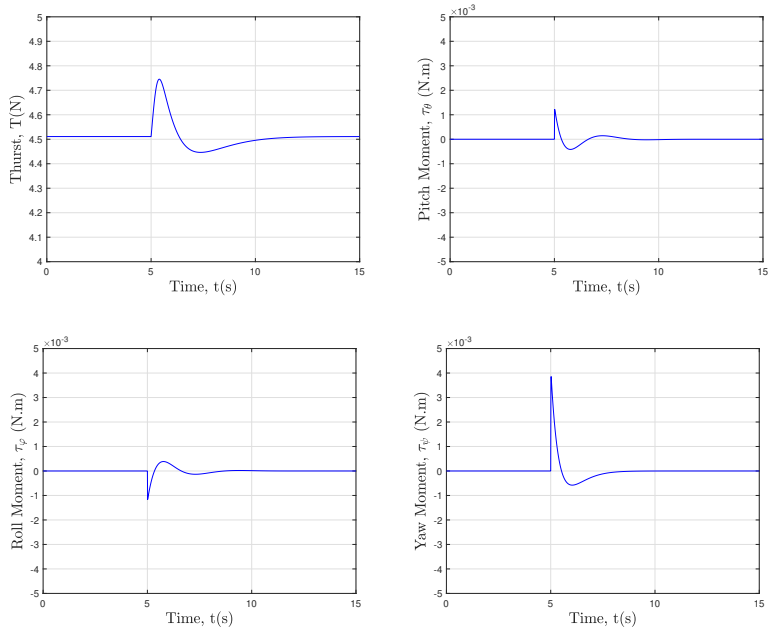

Figure 7.3: Actuation obtained during the step responses in simulation with the controllers designed for the single loop approach using LQR. From left to right, top to bottom: (a) Thrust; (b) Pitch Moment; (c) Roll Moment; (d) Yaw Moment.

initiates with the design of the inner controllers and is followed by the outermost control.

The step-responses for the inner loop subsystems obtained with the $\mathbf{Q}$ and $\mathbf{R}$ matrices specified in Table 7.5 are depicted in Fig. 7.4.

Subsystem	$\mathbf{Q}$	$\mathbf{R}$
Roll	$\text{diag}(25, 1, 1500)$	200
Pitch	$\text{diag}(25, 1, 1500)$	200

Table 7.5: $\mathbf{Q}$ and $\mathbf{R}$ matrices used in the optimal gains computation for the inner loop subsystems.

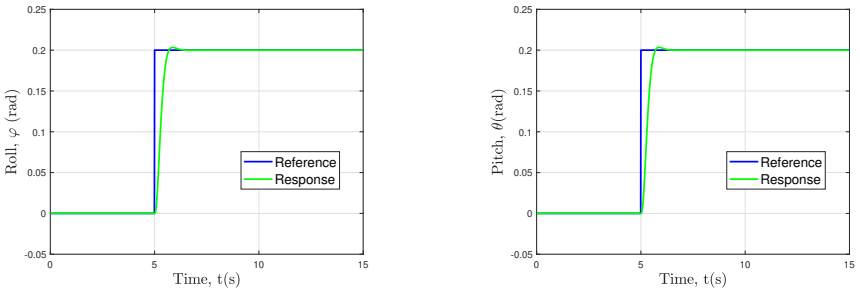

Figure 7.4: Step-responses obtained in simulation with the controllers designed for the inner loop dynamics of the second linear approach. From left to right: (a) Roll; (b) Pitch.

It is possible to verify, from these responses and from the Table 7.6, the attainment of the control requirements.

Concerning the stability, from the Table 7.7 one notes that all eigenvalues are within the unit circle. Thus, the closed loop systems are stable.

Subsystem	Overshoot, M_p (%)	Settling time, t_s (s)
Roll	1.79	0.63
Pitch	1.79	0.63

Table 7.6: Overshoot, M_p, and settling time, t_s, of the step responses of each inner subsystem considered in the second linear approach.

Subsystem	Closed-Loop Eigenvalues
Roll	$\{0.7309.\ 0.9500 \pm 0.0350i\}$
Pitch	$\{0.7869,\ 0.9500 \pm 0.0355i\}$

Table 7.7: Discrete time closed-loop eigenvalues of each inner subsystem considered in the second linear approach.

With the controllers for the inner loop defined, the focus is now on the outer loop linear controllers. The $\mathbf{Q}$ and $\mathbf{R}$ matrices specified in Table 7.8 origin the step responses illustrated in Fig. 7.5.

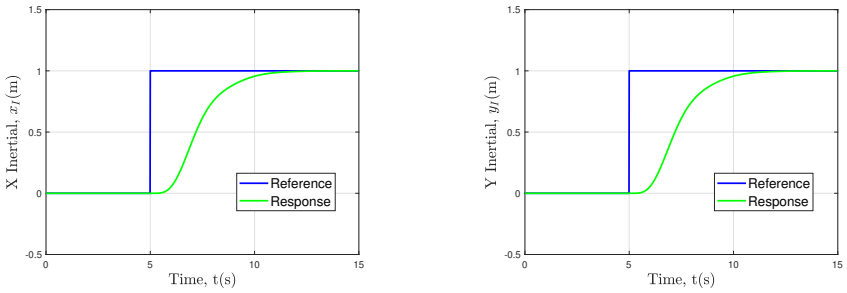

Figure 7.5: Step-responses obtained in simulation with the controllers designed for the outer loop subsystems of the second approach using linear control. From left to right: (a) X Inertial; (b) Y Inertial

Through the step responses illustrated in Fig. 7.5, the Tab. 7.8 is constructed. It is noticeable from these results that the control design parameters aforementioned were achieved.

With regard to the eigenvalues of the consequent closed-loop systems, the Table 7.10 highlights that all eigenvalues are within the unit circle. Hence, the closed-loop systems are stable.

The actuation required during the x and y step responses is displayed in Fig. 7.6. It is noticeable, when comparing with the roll and pitch moments required with the single-loop approach, that the inner-

Subsystem	$\mathbf{Q}$	$\mathbf{R}$
X Inertial	diag$(15, 1, 10)$	1×10^3
Y Inertial	diag$(15, 1, 10)$	1×10^3

Table 7.8: $\mathbf{Q}$ and $\mathbf{R}$ matrices used in the optimal gains computation for each outer subsystem of the second linear control approach.

Subsystem	Overshoot, M_p (%)	Settling time, t_s (s)
X Inertial	0.00	5.72
Y Inertial	0.00	5.73

Table 7.9: Overshoot, M_p, and settling time, t_s, of the step responses of each outer subsystem considered in the second linear approach.

Subsystem	Closed-Loop Eigenvalues
X Inertial	$\{0.9925,\ 0.9927 \pm 0.0087i\}$
Y Inertial	$\{0.9925,\ 0.9927 \pm 0.0087i\}$

Table 7.10: Discrete time closed-loop eigenvalues of each outer subsystem considered in the second linear approach.

loop structure led to an actuation less abrupt.

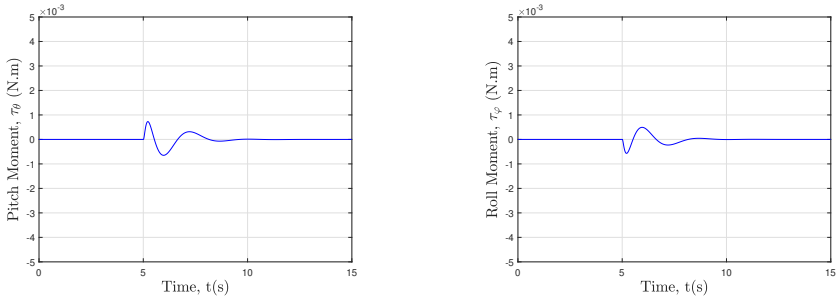

Figure 7.6: Actuation obtained during the step responses in simulation with the controllers designed for the double loop approach using LQR. From left to right, top to bottom. From left to right: (a) Pitch Moment; (b) Roll Moment.

The closed-loop frequency responses of the three subsystems present in the outer loop, with the gains obtained, are depicted in Fig. 7.7. For each position subsystem, the Bode diagram of the transfer function that relates the reference with the output, the reference with the actuation and the position measurement noise with the output are displayed. This figure allows concluding not only that the high-frequency noise is highly attenuated, but also that the reference is always attenuated in the transfer function that outputs the actuation. Moreover, this transfer function has a frequency response similar to the one verified for a band-pass filter. Given the band of frequencies that are less attenuated, one states that the actuation can cope effectively with the references. Furthermore, the static gain of the reference to the output transfer function is 0 dB for all position subsystems, which were forecastable given the null steady-state error. Considering the bandwidth of the referred plots, the position subsystems have the capacity to track frequencies up to roughly 1 rad $\cdot$ s^{-1}.

7.1.2 Feedback Linearization Control

Static Feedback Linearization with zero-dynamics stabilization

Starting with the set of chains of integrators that resulted from applying static feedback linearization to the attitude and altitude dynamics, the step-responses obtained with the $\mathbf{Q}$ and $\mathbf{R}$ matrices detailed in Table 7.11 are displayed in Fig. 7.8.

It is possible to verify, from these responses and from the Table 7.12, the attainment of the predefined control objectives.

Concerning the stability, from the Table 7.13 one notes that all eigenvalues are within the unit circle and, therefore, the closed loop systems are stable.

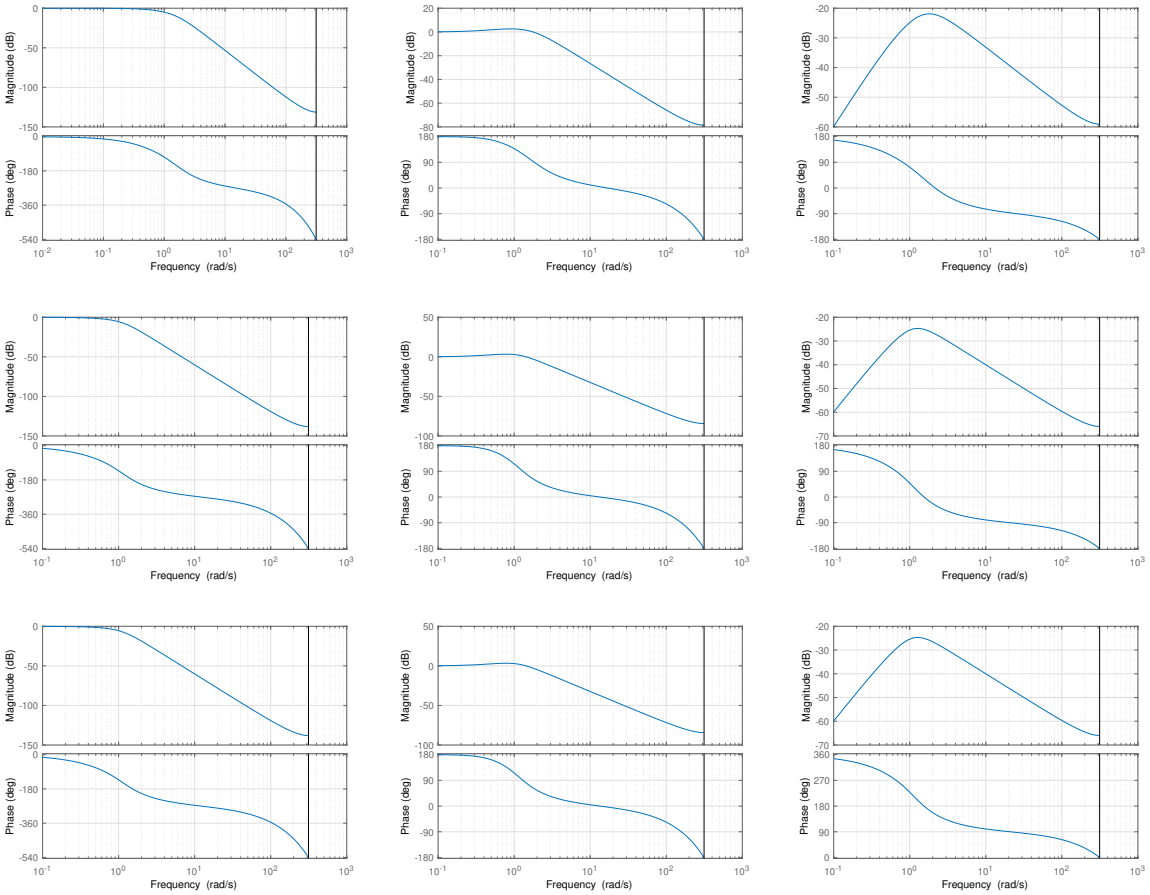

Figure 7.7: Bode plots obtained in simulation with the controllers designed for the outer loop of the double loop approach using linear control. From left to right, top to bottom: (a) Height Subsystem (Reference to Output); (b) Height Subsystem (Noise to Output); (c) Height Subsystem (Reference to Actuation); (d) X Inertial Subsystem (Reference to Output); (e) X Inertial Subsystem (Noise to Output); (f) X Inertial Subsystem (Reference to Actuation); (g) Y Inertial Subsystem (Reference to Output); (h) Y Inertial Subsystem (Noise to Output); (i) Y Inertial Subsystem (Reference to Actuation).

Subsystem	$\mathbf{Q}$	$\mathbf{R}$
Height	$\text{diag}(15, 1, 15)$	1
Roll	$\text{diag}(6 \times 10^4, 7.5 \times 10^2, 2 \times 10^6)$	1
Pitch	$\text{diag}(6 \times 10^4, 7.5 \times 10^2, 2 \times 10^6)$	1
Yaw	$\text{diag}(7.5 \times 10^3, 20, 2 \times 10^5)$	1

Table 7.11: $\mathbf{Q}$ and $\mathbf{R}$ matrices used in the optimal gains computation for each subsystem.

Subsystem	Overshoot, M_p (%)	Settling time, t_s (s)
Height	0.00	4.32
Roll	0.00	0.73
Pitch	0.00	0.73
Yaw	0.00	0.81

Table 7.12: Overshoot, M_p, and settling time, t_s, of the step responses of each inner subsystem considered in the first approach.

With the controllers for the inner loop outlined, the focus is now on the outer loop linear controllers. Since the inner loop has its dynamics linearized for all points defined by $|\varphi|, |\theta| < \frac{\pi}{2}$, its capacity to cope

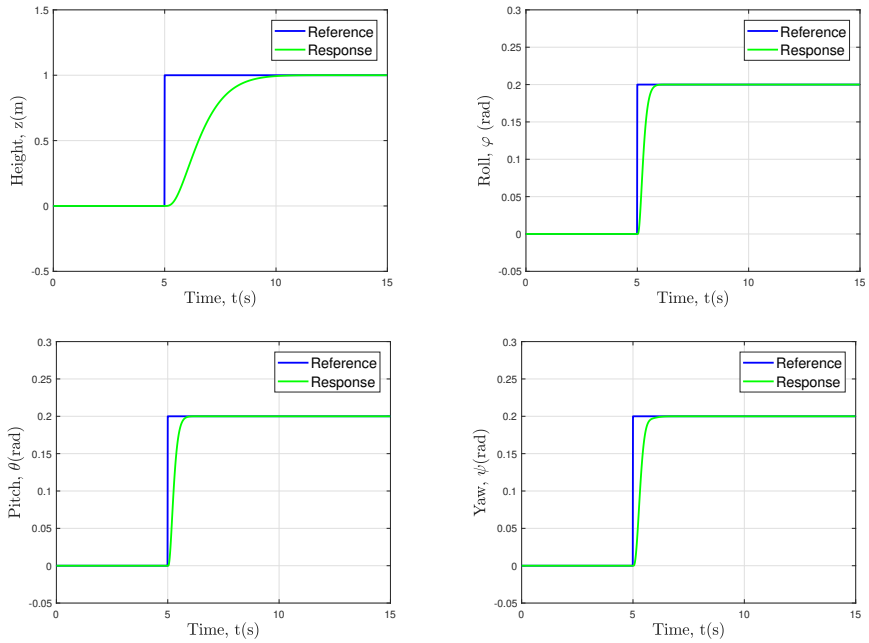

Figure 7.8: Step-responses obtained in simulation with the controllers designed for the inner-loop dynamics of the first approach using nonlinear control. From left to right, top to bottom: (a) Height; (b) Roll; (c) Pitch; (d) Yaw.

Subsystem	Closed-Loop Eigenvalues
Height	$\{0.9861 \pm 0.0137i,\ 0.990\}$
Roll	$\{0.7722,\ 0.9322 \pm 0.0228i\}$
Pitch	$\{0.7722,\ 0.9322 \pm 0.0228i\}$
Yaw	$\{0.9350 \pm 0.0630i,\ 0.9502\}$

Table 7.13: Discrete-time closed-loop eigenvalues of each inner subsystem considered in the first nonlinear approach.

effectively with the angular references provided by the outer loop increased. Therefore, the position control in the XY inertial plan can be faster compared to the designed in the subsection 7.1.1 . The $\mathbf{Q}$ and $\mathbf{R}$ matrices specified in Table 7.14 origin the step responses illustrated in Fig. 7.9.

Through the step responses illustrated in Fig. 7.9, the Tab. 7.14 is constructed. It is noticeable from these results that the control design parameters outlined were achieved.

Subsystem	$\mathbf{Q}$	$\mathbf{R}$
X Inertial	$\text{diag}(5, 1, 30)$	2×10^3
Y Inertial	$\text{diag}(5, 1, 30)$	2×10^3

Table 7.14: $\mathbf{Q}$ and $\mathbf{R}$ matrices used in the optimal gains computation for each outer subsystem of the first nonlinear control approach.

Subsystem	Overshoot, M_p (%)	Settling time, t_s (s)
X Inertial	1.80	3.36
Y Inertial	1.80	3.36

Table 7.15: Overshoot, M_p, and settling time, t_s, of the step responses of each outer subsystem considered in the first nonlinear approach.

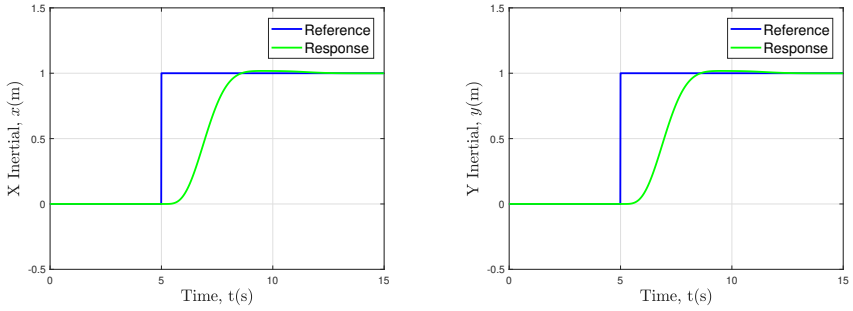

Figure 7.9: Step-responses obtained in simulation with the controllers designed for the *zero dynamics* of the first approach using nonlinear control. From left to right: (a) X Inertial; (b) Y Inertial

With regard to the eigenvalues of the consequent closed-loop systems, the Table 7.16 highlights that all eigenvalues are within the unit circle and thus the closed-loop systems are stable.

Subsystem	Closed-Loop Eigenvalues
X Inertial	$\{0.9897,\ 0.9943 \pm 0.0091i\}$
Y Inertial	$\{0.9897,\ 0.9943 \pm 0.0091i\}$

Table 7.16: Discrete-time closed-loop eigenvalues of each outer subsystem considered in the first nonlinear approach.

Non-Interacting Control via Dynamic Feedback Linearization

The controller for the yaw angle designed for the first nonlinear control architecture is also used in this approach. In respect to the position control, since the respective subsystems obtained with the Feedback Linearization are equal and the design parameters are alike, the same linear controller is considered for these inertial coordinates. Hence, the design analysis is only performed for the height subsystem.

Subsystem	$\mathbf{Q}$	$\mathbf{R}$
Height	diag$(1200, 300, 100, 1, 2000)$	1

Table 7.17: $\mathbf{Q}$ and $\mathbf{R}$ matrices used in the optimal gains computation for the height subsystem.

The step-response obtained with the $\mathbf{Q}$ and $\mathbf{R}$ matrices detailed in Table 7.17 is presented in Fig. 7.10.

Taking into account the null steady-state error, the values detailed in Table 7.18 and the eigenvalues of the closed-loop system, Table 7.19, the desired characteristics were achieved and the closed-loop system is proved to be stable.

Subsystem	Overshoot, M_p (%)	Settling time, t_s (s)
Height	0.10	3.74

Table 7.18: Overshoot, M_p, and settling time, t_s, of the height subsystem step response of the second approach using nonlinear control.

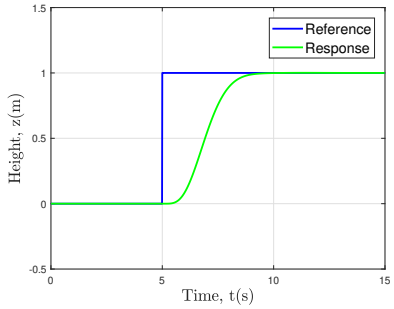

Figure 7.10: Step-response obtained in simulation with the controller designed for the height subsystem of the second approach using nonlinear control.

Subsystem	Closed-Loop Eigenvalues
Height	$\{0.9072,\ 0.9771,\ 0.9870,\ 0.9930 \pm 0.0117i\}$

Table 7.19: Discrete-time closed-loop eigenvalues of the height subsystem of the second approach using nonlinear control.

Inner-Outer Control Loop using Feedback Linearization

Regarding the inner loop linear control, the yaw angle controller is equal to the one devised for the first nonlinear control architecture and the control of the roll and pitch rates is designed while aiming the same target idealized for the previous roll and pitch subsystems. Therefore, it is intended to obtain fast angular responses in order to cope with the references generated by the outer loop.

Subsystem	$\mathbf{Q}$	$\mathbf{R}$
Roll Rate	$\text{diag}(6 \times 10^2, 5 \times 10^4)$	1

Table 7.20: $\mathbf{Q}$ and $\mathbf{R}$ matrices used in the optimal gains computation for each angle rate subsystem.

For the $\mathbf{Q}$ and $\mathbf{R}$ matrices presented in Table 7.20, the resulting step response is depicted in Fig. 7.11.

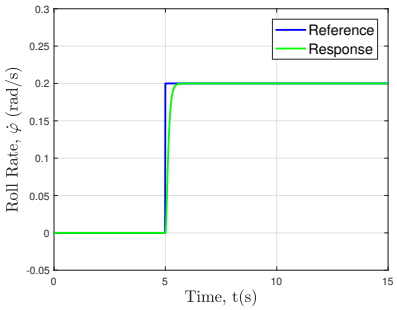

Figure 7.11: Step-response obtained in simulation with the controller designed for the roll rate subsystem of the third approach using nonlinear control.

It is overt when observing the former step response and the Table 7.21 that the predefined control objectives were achieved.

Concerning the stability, from the Table 7.22 one notes that all eigenvalues are inside the unit circle. Hence, the closed loop systems are stable.

Subsystem	Overshoot, M_p (%)	Settling time, t_s (s)
Roll Rate	0.00	0.43

Table 7.21: Overshoot, M_p, and settling time, t_s, of the step response of roll rate subsystem of the third approach using nonlinear control.

Subsystem	Closed-Loop Eigenvalues
Roll Rate	$\{0.7992,\ 0.9048\}$

Table 7.22: Discrete-time closed-loop eigenvalues of the roll rate subsystem of the third nonlinear approach.

With the controllers for the inner loop defined, the outer loop linear controllers are now obtained. Once more, the inertial coordinates subsystems are all equal after the transformation and nonlinear feedback and, consequently, the same controller is used for all.

The $\mathbf{Q}$ and $\mathbf{R}$ matrices specified in Table 7.23 origin the step response illustrated in Fig. 7.12.

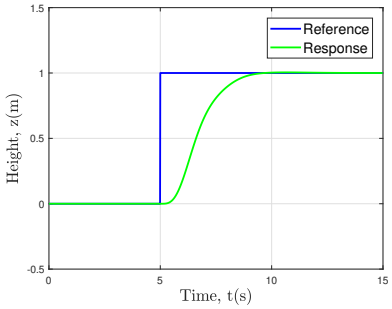

Figure 7.12: Step-response obtained in simulation with the controller designed for the height subsystem of the third approach using nonlinear control.

In the Table 7.24 the characteristics of the previous step response are detailed. It is noticeable from these results that the control design parameters outlined were achieved. Regarding the stability, the Table 7.25 evidences that all eigenvalues are within the unit circle and thus the closed-loop systems are stable.

Subsystem	$\mathbf{Q}$	$\mathbf{R}$
Height	diag$(200, 1, 20, 300)$	1

Table 7.23: $\mathbf{Q}$ and $\mathbf{R}$ matrices used in the optimal gains computation for the height subsystem of the third nonlinear control approach.

Subsystem	Overshoot, M_p (%)	Settling time, t_s (s)
Height	0.47	3.94

Table 7.24: Overshoot, M_p, and settling time, t_s, of the step response of the height subsystem considered in the third nonlinear approach.

Subsystem	Closed-Loop Eigenvalues
Height	$\{0.9558,\ 0.9887,\ 0.9887 \pm 0.0143i\}$

Table 7.25: Discrete-time closed-loop eigenvalues of the height subsystem considered in the third nonlinear approach.

7.2 Robustness to model parameters variation

In fact, since the Feedback Linearization method resorts to the system dynamics to define the control input in furtherance of canceling nonlinear terms, this nonlinear control technique can present high sensitivity to modeling errors. Similarly, this model dependency is also verified in the linear control resorted since the gains of the Linear Quadratic Regulator controllers are obtain as solution of an optimization process where the system dynamics impose the restrictions. Consequently, as a way to assess the robustness of the proposed architectures, several tests were performed with variations of the mass and inertia. The tests are conducted with significant variations of the mass and of the inertia. The references for the position are steps with an amplitude of 3 m and the reference for the yaw angle is a ramp with a slope of 0.1 rad $\cdot$ s^{-1}.

7.2.1 Linear Quadratic Regulator with Integrative Action

Single Loop Structure

The responses obtained with the model parameters variation are depicted in Fig. 7.13. It is visible that the responses altered significantly when only a reduction in the inertia was considered. It is important to stress that, in this case, the height response does not fulfill the control objectives. However, in general, the system demonstrated some level of robustness, since the majority of the responses obtained complied with the requirements established.

Inner-Outer Loop Structure

The responses obtained with the model parameters variation are displayed in Fig. 7.14. It is visible that the step responses of the inertial coordinates x and y presented negligible alterations. In fact, the settling time varied less than 0.02 seconds and the overshoot and the static error are null in all responses. Concerning the height response, the performance altered very slightly, with the settling time varying less than 0.3 seconds and the overshoot less than 0.2% and the static error was kept null. Regarding the yaw angle, the fact that the responses are indistinguishable evidences that the control performance did not changed in the various tests conducted.

In order to deepen the analysis, larger variations of the inertia matrix were considered. The results are presented in Fig. 7.15. This further testing allows verifying that the height and yaw responses present an oscillatory behavior, characteristic of a marginally stable system, when a decrease of 75% is considered for the inertia matrix values.

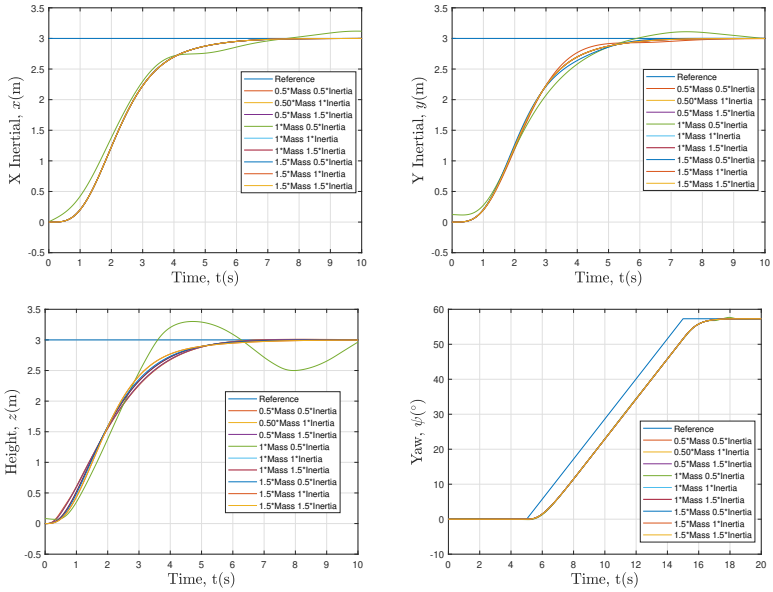

Figure 7.13: Responses obtained with the first approach using only LQR in the inertia and mass variation robustness test. From left to right, top to bottom: (a) X Inertial; (b) Y Inertial; (c) Height; (d) Yaw.

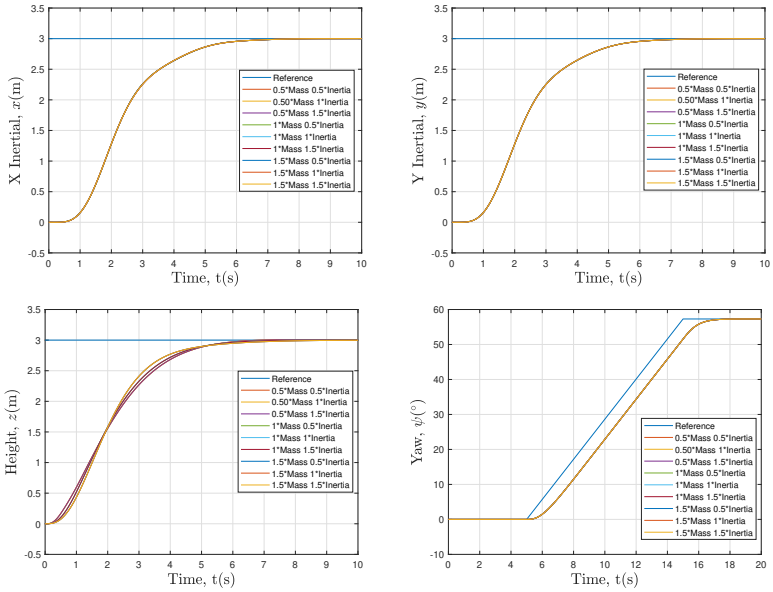

Figure 7.14: Responses obtained with the second approach using only LQR in the inertia and mass variation robustness test. From left to right, top to bottom: (a) X Inertial; (b) Y Inertial; (c) Height; (d) Yaw.

Therefore, given the results obtained, the inner-outer loop structure presented a higher level of robustness when compared with the single-loop control architecture. Furthermore, although significant changes in the model parameters were consider and the amplitude of the steps required very aggressive maneuvers, the responses met the performance criteria when combined variations of 50% in the mass and inertia were studied, which demonstrates an interesting level of robustness for a linear approach.

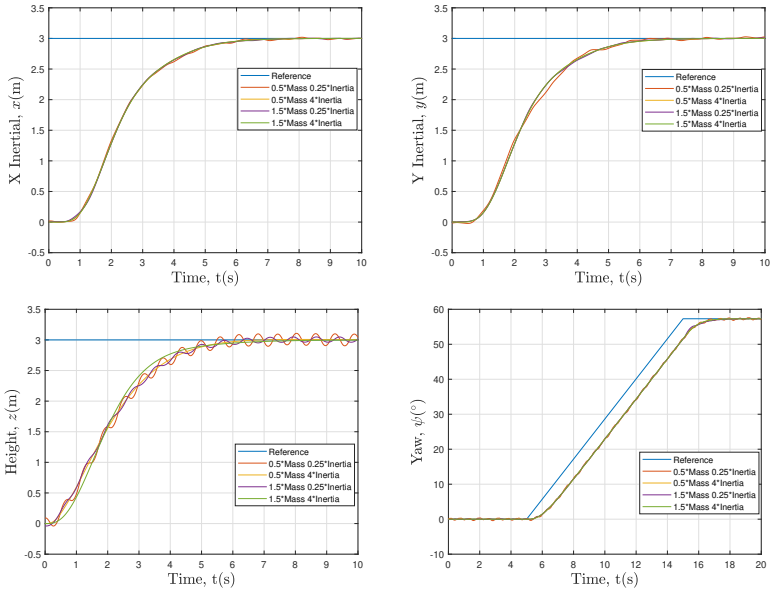

Figure 7.15: Responses obtained with the second approach using only LQR in the second robustness test considering larger variations. From left to right, top to bottom: (a) X Inertial; (b) Y Inertial; (c) Height; (d) Yaw.

7.2.2 Feedback Linearization Control

Static Feedback Linearization with zero-dynamics stabilization

The position responses obtained with the model parameters variation are displayed in Fig. 7.16. It is overt that the step responses of the inertial coordinates x and y presented negligible alterations. In fact, the settling time varied less than 0.01 seconds, the overshoot did not alter more than 0.1% and the static error is null in all responses. Concerning the height response, the performance altered very slightly, with the settling time varying less than 0.2 seconds, the overshoot less than 0.2% and the static error was kept null. These are very promising and important results since the performance did not deteriorate even tough significant changes in the model parameters were consider and the amplitude of the steps required very aggressive responses. Due to the fact that the yaw responses did not change with the variation of the mass, the graphs displayed are for tests conducted only varying the inertia of the quadcopter.

In order to deepen the analysis and to compare to the second linear approach, larger variations were considered. The results are depicted in Fig. 7.17. Compared to the results obtained for the same variations with the inner-outer loop architecture using linear control, Fig. 7.15, the static feedback linearization applied to the altitude and attitude dynamics achieved better results. Not only the responses remained stable, but also the control performance presented a higher consistency throughout the tests, which is reflected in the fact that all responses respected the design criteria.

As a way to study the impact of the integrative action included, new gains for the subsystems without this inclusion were computed. It was aimed to attain very similar responses in each subsystem in order to establish a better comparison between the results of the robustness test. The details of this computation

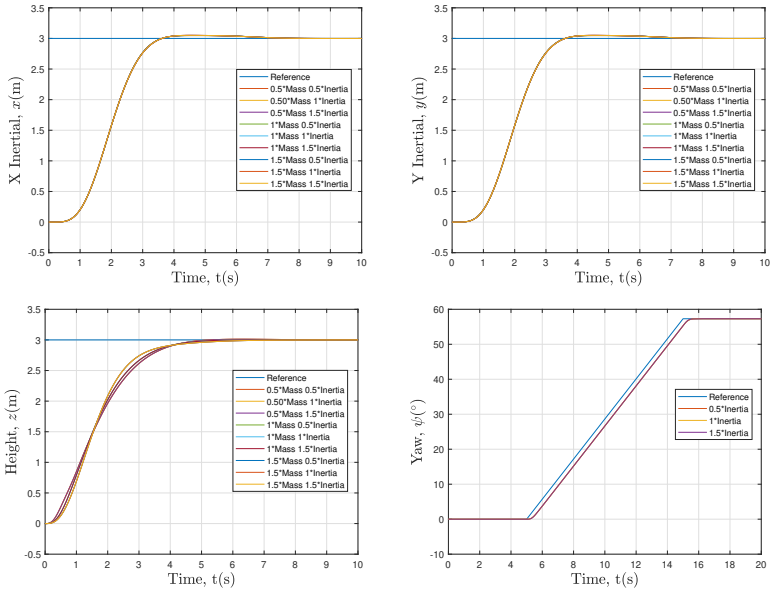

Figure 7.16: Responses obtained with the first approach using Feedback Linearization in the inertia and mass variation robustness test. From left to right, top to bottom: (a) X Inertial; (b) Y Inertial; (c) Height; (d) Yaw.

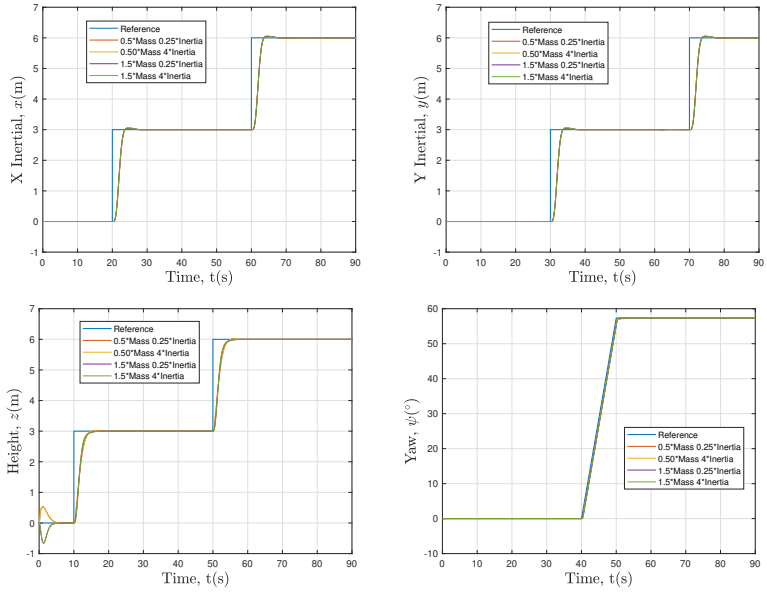

Figure 7.17: Responses obtained with the first approach using Feedback Linearization in the robustness test considering larger variations. From left to right, top to bottom: (a) X Inertial; (b) Y Inertial; (c) Height; (d) Yaw.

are presented in Table 7.26, from which is noted that the characteristics of the step responses are almost identical to the obtained when the integrator was embedded in each subsystem.

The responses obtained are depicted in Fig. 7.18 The absence of integrative action originated the appearance of undesired static error, especially in the height response. Furthermore, a coupling effect between this response variation and the other subsystems is detected and the deviations from the responses obtained without model parameters variation are noticeable. Consequently, the outcome

Subsystem	$\mathbf{Q}$	$\mathbf{R}$	Overshoot, M_p (%)	Settling time, t_s (s)
Height	$\text{diag}(8, 7)$	1	0.00	4.36
Roll	$\text{diag}(3 \times 10^5, 9 \times 10^3)$	1	0.00	0.68
Pitch	$\text{diag}(3 \times 10^5, 9 \times 10^3)$	1	0.00	0.68
Yaw	$\text{diag}(2.9 \times 10^5, 1.3 \times 10^4)$	1	0.00	0.83
X Inertial	$\text{diag}(12, 4)$	1200	1.82	3.23
Y Inertial	$\text{diag}(12, 4)$	1200	1.82	3.23

Table 7.26: $\mathbf{Q}$ and $\mathbf{R}$ matrices used in the test without integrative action and the overshoot and settling time of the resulting step responses.

of this robustness test evidences the important role of the integrative action in diminishing the model dependence and in increasing the robustness of the control architecture.

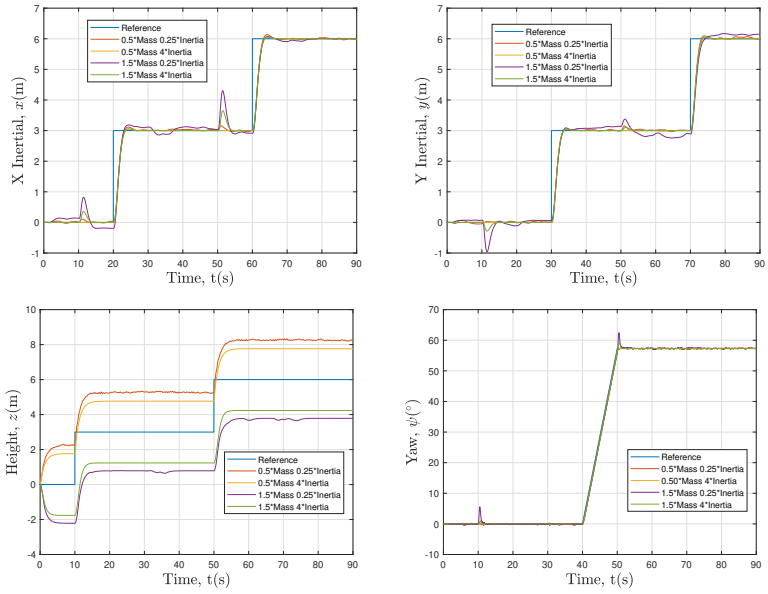

Figure 7.18: Responses obtained with the first approach using Feedback Linearization without integrative action in the robustness test considering large mass and inertia variations. From left to right, top to bottom: (a) X Inertial; (b) Y Inertial; (c) Height; (d) Yaw.

In summary, the results of the model parameters variation test for the first architecture using nonlinear control demonstrate the robustness of this approach and the importance of the integrative component. The control performance was roughly equal in every test, which translated in the fulfillment of the design parameters even in cases of large variations simultaneously in the values of the mass and inertia.

Non-Interacting Control via Dynamic Feedback Linearization

Carrying out the robustness tests with the second approach using Feedback Linearization led to the responses depicted in Fig. 7.19 and in Fig. 7.20.

Contrary to the previous architecture, with the Non-Interacting Control via Dynamic Feedback Linearization a fraction of the tests conducted caused an unstable behavior. This behavior is presented in Fig. 7.20 through the height response. By analyzing the tests displayed in Fig. 7.19, it is noted that the performance did not deteriorate significantly when one of the model parameters in the study increased.

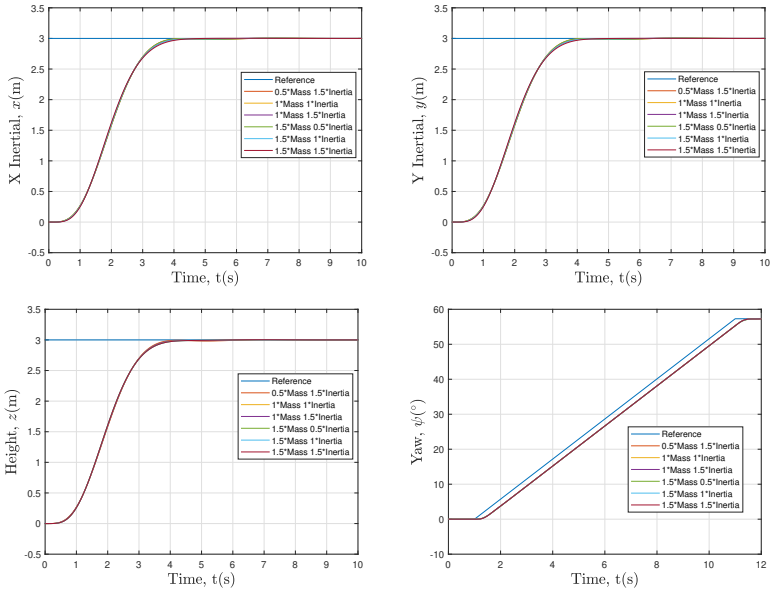

Figure 7.19: Responses obtained with the second approach using Feedback Linearization in the inertia and mass variation robustness test. From left to right, top to bottom: (a) X Inertial; (b) Y Inertial; (c) Height; (d) Yaw.

In fact, considering only this tests, the height response varied less than 0.1% and the settling time had a variation inferior to 0.1 seconds, which corresponds to a higher robustness to the increase of mass than the verified in the first architecture using nonlinear control. When the system presented a stable behaviour, the control objectives were achieved. However, in general, the non-interacting control via dynamic feedback demonstrated a higher level of control performance worsening than the verified for the architecture with the static feedback inner loop.

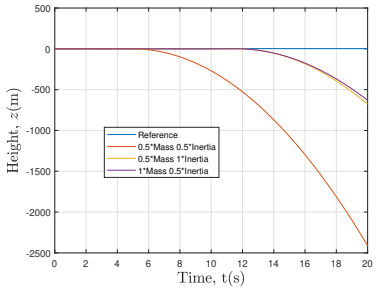

Figure 7.20: Unstable height responses obtained with the second approach using FL in the inertia and mass variation robustness test.

Inner-Outer Control Loop using Feedback Linearization

Finally, by extending the robustness tests to the last architecture derived, the responses depicted in Fig. 7.21 and in Fig. 7.22 are obtained.

This architecture exhibited robustness to increases of the mass and inertia since the design parameters were fulfilled even in situations of simultaneous increases in these two model parameters. Similar to the former architecture, the control structure did not present robustness to all the variations under study.

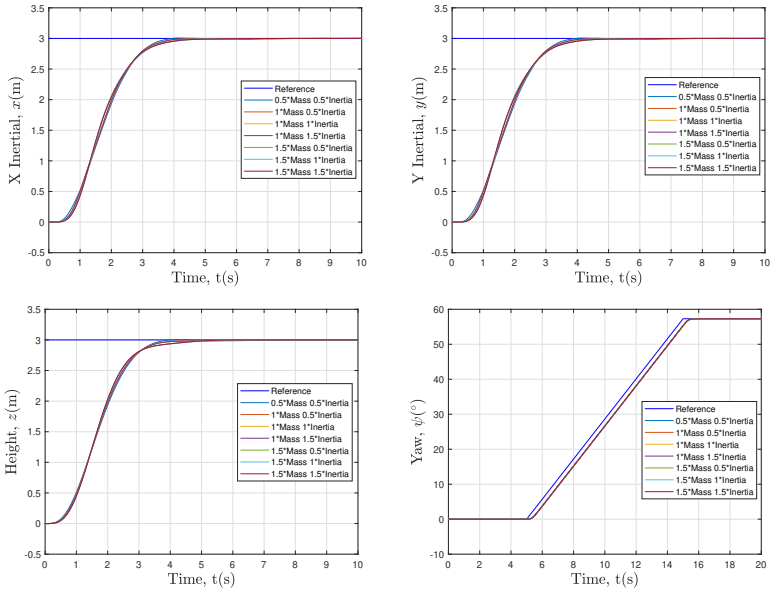

Figure 7.21: Responses obtained with the third approach using Feedback Linearization in the inertia and mass variation robustness test. From left to right, top to bottom: (a) X Inertial; (b) Y Inertial; (c) Height; (d) Yaw.

Nonetheless, the height response did not vary its overshoot more than 0.2% and its settling time more than 0.5 seconds in all of this tests, even though the other position responses demonstrated an unstable behavior under certain variations. Overall, the results indicate a level of robustness inferior to the one displayed by the first nonlinear control approach.

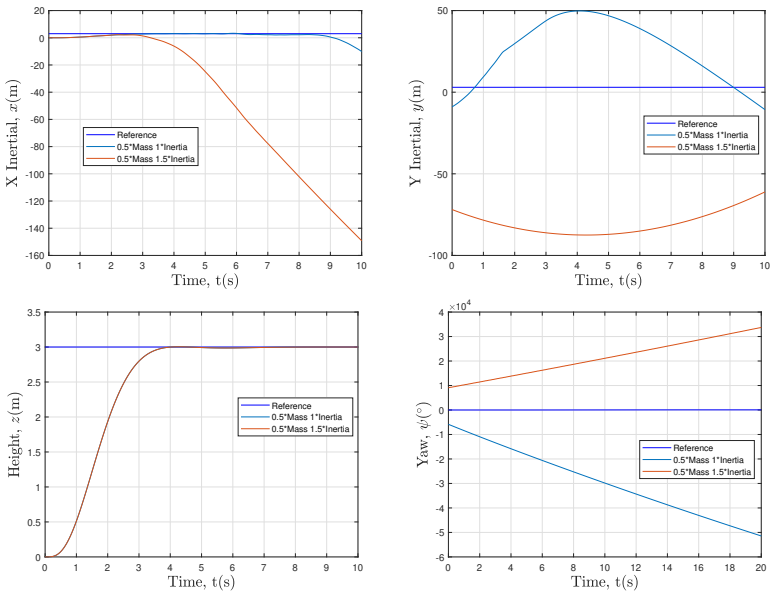

Figure 7.22: Responses obtained with the third approach using Feedback Linearization in the robustness tests that resulted in unstable behaviour. From left to right, top to bottom: (a) X Inertial; (b) Y Inertial; (c) Height; (d) Yaw.

7.3 Estimation

The complementary filters, designed in subsection 6.1.1, are applied in this section. The resulting estimates are evaluated considering the maximum error. The attitude filters resorted are not evaluated since they were developed and experimentally validated in other works.

From real sensor data that was acquired while fixing the drone position and while subjecting the propellers to the PWM commands required for hovering (see Appendix A), the variances of the measurement noises were computed. The values obtained are detailed from Table 7.27 to Table 7.29. Moreover, for the position and acceleration measurements, the variances presented correspond to the ${}^{K}\mathbf{R}$ considered in the Kalman filters. The matrix ${}^{K}\mathbf{Q}$ was used as a tuning parameter.

x	y	z
1.25×10^{-9}	1.51×10^{-9}	1.18×10^{-6}

Table 7.27: Variances (m) considered in the position measurement noise included in the nonlinear model.

${}^{B}a_x$	${}^{B}a_y$	${}^{B}a_z$
1.87×10^{-5}	3.62×10^{-5}	1.08×10^{-5}

Table 7.28: Variances ($\text{m} \cdot \text{s}^{-2}$) considered in the accelerometer measurement noise included in the nonlinear model.

p	q	r
6.54×10^{-4}	2.82×10^{-4}	2.04×10^{-4}

Table 7.29: Variances ($\text{rad} \cdot \text{s}^{-1}$) considered in the gyroscope measurement noise included in the nonlinear model.

7.3.1 Velocity Estimation

For the estimation of the inertial velocities, the matrices ${}^{\mathbf{K}}\mathbf{Q}$ detailed in Table 7.30 were considered.

Estimate	$\dot{\widehat{x}}$	$\dot{\widehat{y}}$	$\dot{\widehat{z}}$
${}^{\mathbf{K}}\mathbf{Q}$	$\text{diag}(1 \times 10^{-8}, 25 \times 10^{-4}$	$\text{diag}(1 \times 10^{-8}, 25 \times 10^{-4})$	$\text{diag}(25 \times 10^{-8}, 4 \times 10^{-4})$

Table 7.30: ${}^{\mathbf{K}}\mathbf{Q}$ considered in the inertial velocities estimation.

The estimates can be evaluated through Fig. 7.23 and Fig. 7.24. From the low maximum error obtained is possible to conclude that the filters designed demonstrate a good level of precision. Moreover, the performance of the estimator when the velocity rapidly varied is highly satisfactory, which is a characteristic of paramount importance.

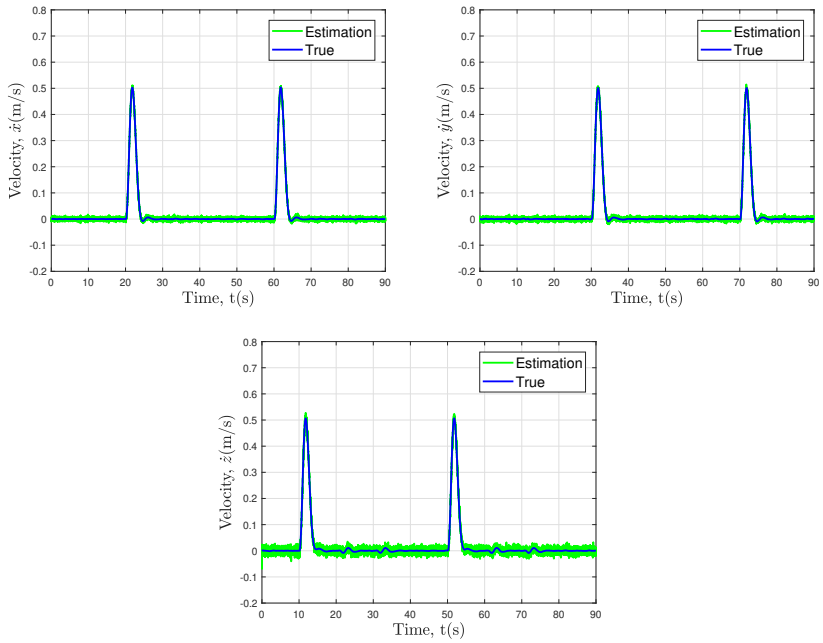

Figure 7.23: Comparison between the velocities obtained in simulation and the estimated. From left to right, top to bottom: (a) $\dot{x}$; (b) $\dot{y}$; (c) $\dot{z}$.

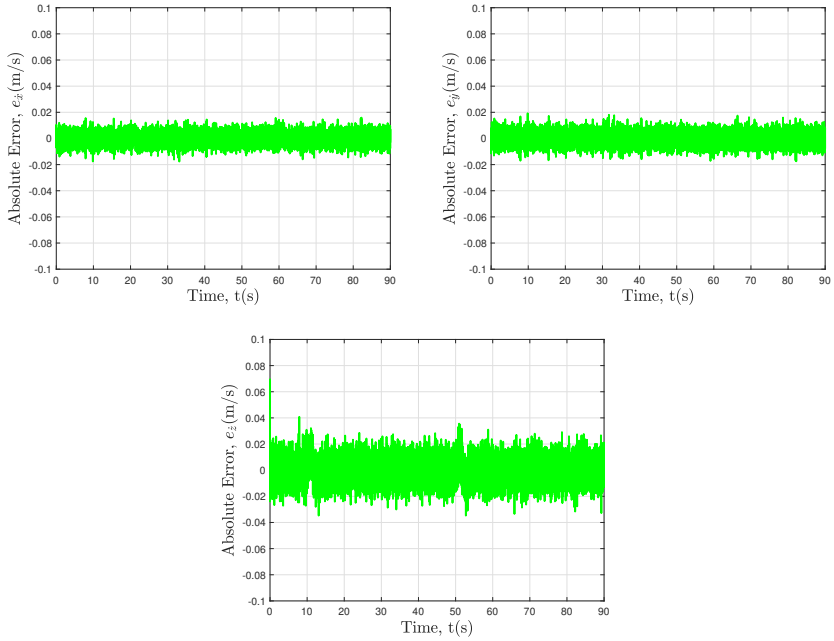

Figure 7.24: Absolute error obtained in the estimation of the inertial velocities. From left to right, top to bottom: (a) $e_{\dot{x}}$; (b) $e_{\dot{y}}$; (c) $e_{\dot{z}}$.

7.3.2 Jerk Estimation

For the estimation of the third derivatives of the position, the matrices ${}^{\mathbf{K}}\mathbf{Q}$ detailed in Table 7.31 were used.

The accelerometer measurements are more corrupted by noise than the position observations. Consequently, jerk estimation is more difficult than reconstructing the inertial velocities.

Estimate	$\widehat{x}^{(3)}$	$\widehat{y}^{(3)}$	$\widehat{z}^{(3)}$
${}^{\mathbf{K}}\mathbf{Q}$	diag$(1 \times 10^{-6}, 9 \times 10^{-4})$	diag$(4 \times 10^{-6}, 16 \times 10^{-4})$	diag$(1 \times 10^{-6}, 9 \times 10^{-4})$

Table 7.31: ${}^{\mathbf{K}}\mathbf{Q}$ considered in the jerk estimation.

To establish a comparison, in Fig. 7.25 the jerk obtained in simulation is plotted with the reconstructed with the filter. It is visible that the third derivative of the position is far more corrupted by noise, this is due to the amplification of the acceleration noise originated by being differentiated. Nonetheless, the filters demonstrate the ability to correctly estimate the third derivative, which is sustained by the absolute error displayed in Fig. 7.26.

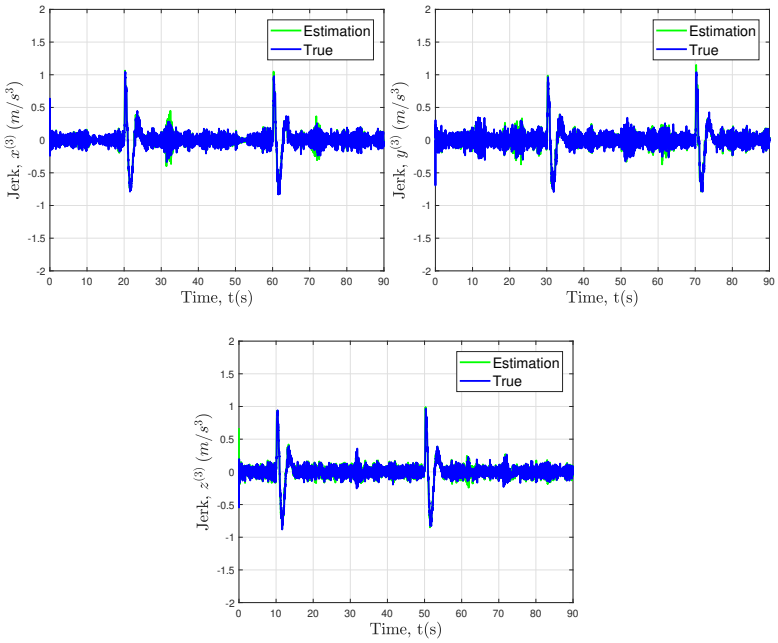

Figure 7.25: Comparison between the jerk obtained in simulation and the estimated. From left to right, top to bottom: (a) $x^{(3)}$; (b) $y^{(3)}$; (c) $z^{(3)}$.

7.4 Trajectory Tracking

To evaluate the performance of the proposed control architectures, a trajectory was created and given as a reference for the subsystems. This trajectory is defined with a constant yaw angle equal to zero and, excluding the take-off, is formed by rectilinear sections, with a constant velocity of $0.05\,\mathrm{m}\cdot\mathrm{s}^{-1}$, and semicircular sections, with a constant angular velocity of 0.05π rad $\cdot\,\mathrm{s}^{-1}$.

In an effort to approximate even more the simulation to the actual quadcopter and to account and study the impact of the noise of the sensors, noise disturbances were included in the nonlinear model. These disturbances were modeled as zero-mean Gaussian white noise with the variances detailed in Table 7.27 to Table 7.29.

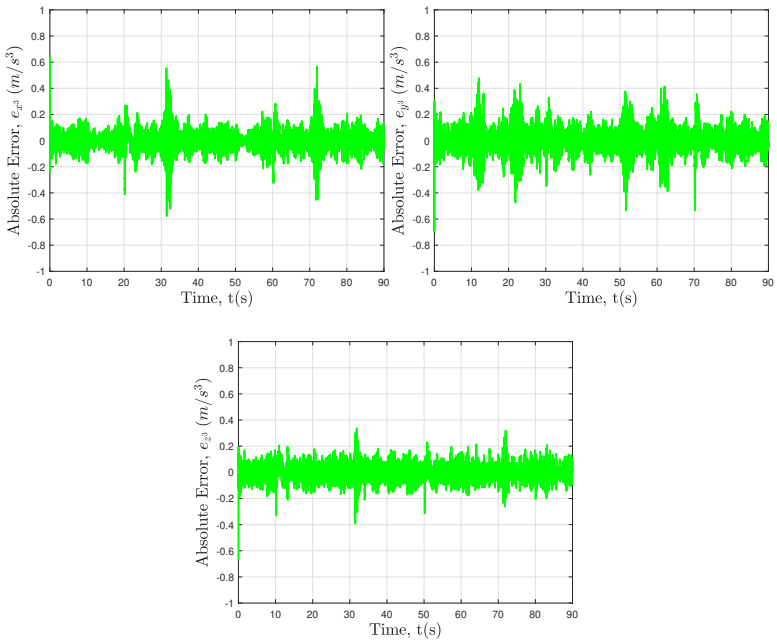

Figure 7.26: Absolute error obtained in the estimation of jerk. From left to right, top to bottom: (a) $e_{x^{(3)}}$; (b) $e_{y^{(3)}}$; (c) $e_{z^{(3)}}$.

7.4.1 Linear Quadratic Regulator with Integrative Action

Starting with the linear approaches, from the observation of the Fig. 7.27, it can be concluded that the control structures implemented allow a good following of the trajectory. The time responses have a steady-state error for references with constant velocity, which was predictable since the integrator included in the control structure only has the capacity to eliminate the steady-state error for constant references. Notwithstanding, once the ramp references are followed up by constant inputs, the responses converge without error to the desired coordinate. It is important to emphasize that the noise insertion did not cause relevant perturbations in the position responses.

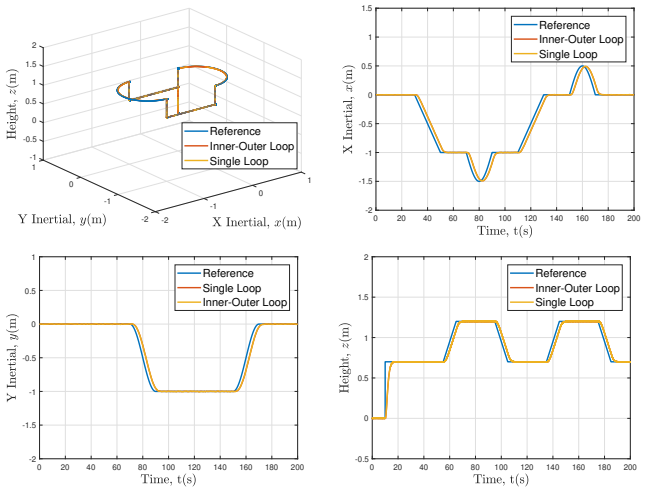

Figure 7.27: Responses obtained during trajectory tracking in simulation with the linear control approaches. From left to right, top to bottom: (a) Trajectory Tracking in 3D space; (b) X Inertial; (c) Y Inertial; (d) Height.

In fact, the two control approaches did not present relevant differences so far. However, by analyzing the angular responses depicted in Fig. 7.28, one notes that the single loop architecture led to a slightly more oscillatory roll and pitch responses. Concerning the yaw angle, due the fact that the same controller was used in both structures, the response is equal.

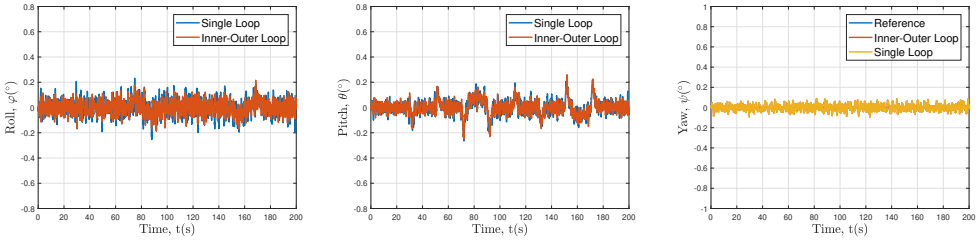

Figure 7.28: Euler angles responses obtained during trajectory tracking in simulation with the linear control approaches. From left to right: (a) Roll; (b) Pitch; (c) Yaw.

The forces and moments computed during the trajectory are depicted in Fig. 7.29. The smaller roll and pitch moments required along the trajectory, when comparing the double-loop approach with the single-loop control strategy, corroborate not only the idea of the former architecture to have a more efficient angular response that the latter, but also the idea of the inner-outer loop being less impacted by measurement noise. In the real system, once the perturbations can possibly be larger, the importance of this inner loop in obtaining a smoother flight is even more evident.

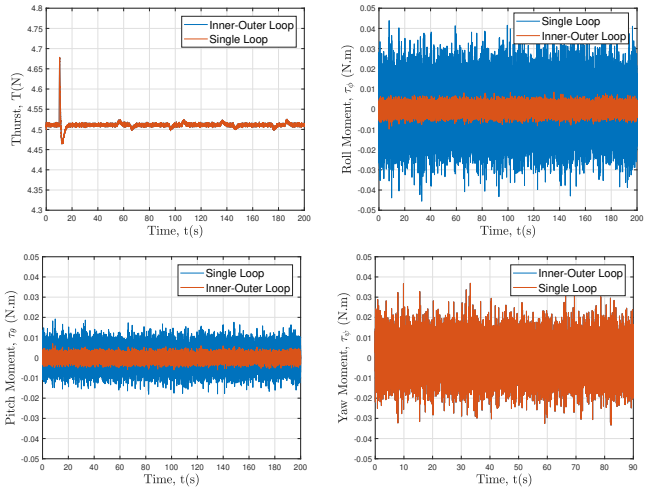

Figure 7.29: Forces and moments computed during trajectory tracking in simulation with the linear control approaches. From left to right, top to bottom: (a) Thrust; (b) Roll Moment; (c) Pitch Moment; (d) Yaw Moment.

7.4.2 Feedback Linearization Control

The trajectory tracking capacity of the Feedback Linearization based control structures is now evaluated. From the position responses obtained, displayed in Fig. 7.30, one can state that the control architectures implemented enabled a good following of the trajectory. Once again, the static error for references with constant velocity is present. However, due to the fact that these solutions are faster than

the linear approaches developed, the error is inferior to the previously verified. Furthermore, is equally relevant to stress that the noise insertion did not cause relevant perturbations in these responses.

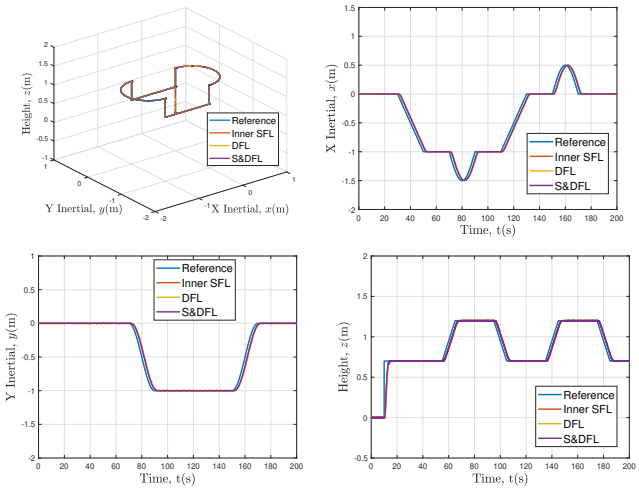

Figure 7.30: Responses obtained during trajectory tracking in simulation with the nonlinear control approaches. From left to right, top to bottom: (a) Trajectory Tracking in 3D space; (b) X Inertial; (c) Y Inertial; (d) Height.

Concerning the attitude, Fig. 7.31, the roll and pitch angles responses are less impacted by corrupted measurements with the Non-Interacting Control solution. These results are explained by the fact that the transformed inputs of this solution are computed without considering either the angular velocity nor the Euler rates, which are the variables corrupted by the noise with higher variance.

Since the nonlinear control has faster responses, obtaining higher maximum angle values, comparatively with the linear control solutions, was expected. Note that this is easily verified in the pitch response once it is the most solicited angle during the trajectory. The control structure with the Static FL applied to the inner loop presents an angular response less affected by the noise inserted than the linear approaches presented in the former subsection. Thus, the application of nonlinear control to the attitude dynamics seems to present better results. On the other hand, the inner-outer control loop using FL is undoubtedly the most affected by the noise, regarding the angular responses, between the five architectures tested.

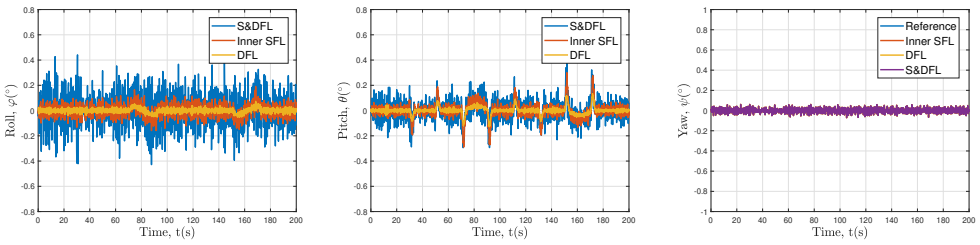

Figure 7.31: Euler angles responses obtained during trajectory tracking in simulation with the nonlinear control approaches. From left to right: (a) Roll; (b) Pitch; (c) Yaw.

The forces and moments computed during the trajectory are displayed in Fig. 7.32. These graphs corroborate the idea of the Dynamic FL solution being the control structure that originate the responses

less impacted by the noise. Furthermore, establishing a comparison between the nonlinear and the liner architectures, from the point of view of the actuation, the single loop linear control structure is the most influenced by the noise considered.

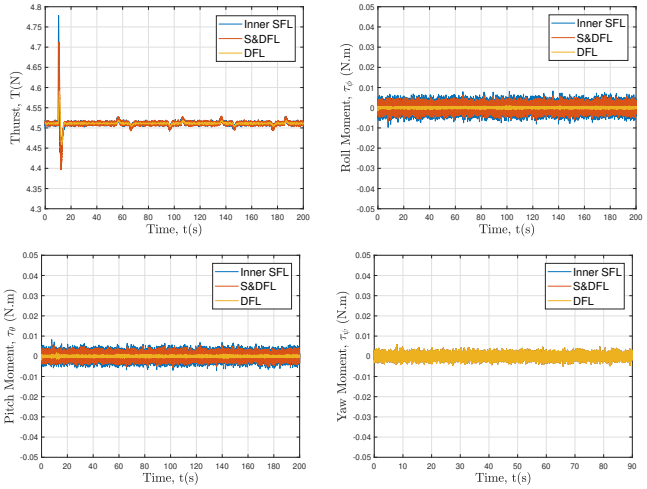

Figure 7.32: Forces and moments computed during trajectory tracking in simulation with the nonlinear control approaches. From left to right, top to bottom: (a) Thrust; (b) Roll Moment; (c) Pitch Moment; (d) Yaw Moment.

Finally, the root-mean-square error obtained with the control architectures tested is presented in Table. 7.32. It is noticeable that the nonlinear control approaches achieved smaller root-mean-squared values.

	x (m)	y (m)	z (m)	ψ (°)
Single Loop	0.0812	0.0631	0.0480	0.0302
Inner-Outer Loop	0.0814	0.0633	0.0479	0.0302
Inner SFL Loop	0.0666	0.0508	0.0386	0.0216
DFL	0.0676	0.0519	0.0449	0.0216
Inner-Outer FL	0.0599	0.0459	0.0398	0.0216

Table 7.32: Root-mean-square error obtained in simulation.

Chapter 8

Experimental Results

In this chapter are discussed aspects regarding the implementation and presented and analyzed experimental results.

Since the simulation allowed to assess the most promising approaches and once testing all strategies developed would be very time-consuming, one linear and one nonlinear control strategies were chosen to test in the actual UAV. Thereby, a comparison between the linear and nonlinear control approaches is possible. This selection is based not only on the results presented in the trajectory following but also on the robustness to model parameters variation. Consequently, for the linear approach, the inner-outer loop control structure using LQR with integrative action was opted and, for the nonlinear control, the strategy selected was the static feedback linearization with zero-dynamics stabilization. The ”Inner-Outer Control Loop using Feedback Linearization” approach presented slightly lower root-mean-squared error in the trajectory tracking when compared with the static feedback linearization approach. However, the later presents a higher level of robustness and is computationally less demanding, given its reduced number of calculations. For this reason, the later was opted instead.

The experimental tests consist of evaluating the capacity of the quadcopter, with the control developed, to perform the trajectory considered in the simulation. The control and estimation solutions were implemented with a sampling time of 0.01 seconds. Posteriorly, comparisons between the experimental and the simulation results are drawn. Histograms of the position and yaw angle tracking error are presented to deepen the analysis.

8.1 Implementation

In order to implement the control structure in the quadrotor, the ”AR Drone 2.0 Quadcopter Embedded Coder” developed by [67] was used. This *Simulink* project enables direct access to the sensors and the actuators of the quadcopter. The inertial position of the vehicle is provided by a motion capture system [68].

In Fig. 8.1, the connections established between the computer running the Qualisys Track Manager (QTM), the host computer, where the *Simulink* model is compiled, and the hardware board of the UAV

are summarized. The computer where the QTM is running transmits the data, at a rate of 100 Hz, through an ethernet cable to the host computer using the User Datagram Protocol (UDP) and has the port 9091 as the endpoint of communication. The data received is sent through a Wi-Fi connection to the port 6000 of the quadrotor also using the UDP as the transport layer. The *Simulink* model is compiled to C-Code before being executed and a binary file, possible to run on the operating system of the quadrotor, is generated and deployed to the hardware board. A wireless File Transfer Protocol (FTP) connection to the port 5551 assures the deployment. An additional Transmission Control Protocol (TCP) connection to the port 23 of the vehicle is created to send string commands. The indications are given to clear old executable files and to manage permissions. The Simulink model on the host computer is executed in normal mode; therefore, to the purpose of sending the take-off/land and emergency commands, the UDP connection on port 6000 is used. The quadcopter sends the flight data in real-time through a UDP connection that uses the port 6001 of the host computer as the endpoint of communication. The UDP connections enable real-time communication with the generated code deployed to the hardware board. The IP addresses and the ports used are schematically represented in Fig. 8.1.

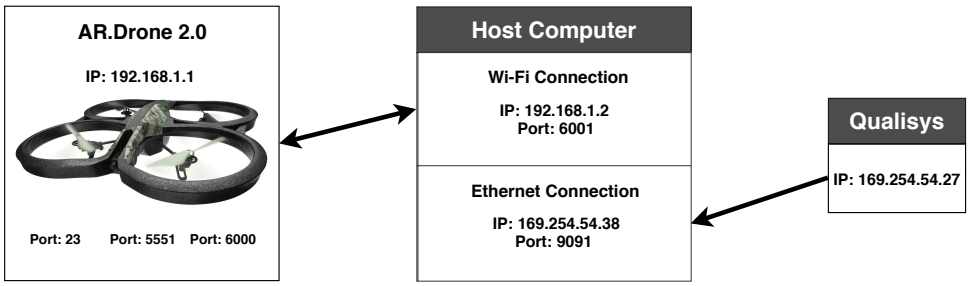

Figure 8.1: Network Diagram describing the connections between the computer running the Qualisys Track Manager, the host computer and the UAV.

The control techniques and the estimation approaches considered were implemented in the Embedded Coder *Simulink* environment. The structure of the resulting control software designed is displayed in Fig. 8.2.

As was previously stated, communication in real-time with the quadrotor relies on UDP connections. Thereby, the blocks ”UDP Receiver” and ”UDP Sender” are accountable for these connections. The first block receives the x and y inertial coordinates, flight instructions and position and yaw references. The flight instructions correspond to indications to take-off or land and to an emergency flag. This block operates at 200Hz to minimize data loss. To send the state-variables and PWM commands to the host computer the second block is used. These indications affect after the first 10 seconds, in which not only the correction of the sensor measurements takes place but also the convergence of the estimators.

The ”IMU Sensors” block outputs a bus signal directly from the IMU board containing multiple measurements. Since the IMU board operates at 200 Hz, a higher sampling rate, 400 Hz, was defined in this block in furtherance of avoiding the fragmentation of data due to the loss of packets. This block outputs, additionally, a checksum flag that indicates an error in the transmission of the data when it is 1 and indicates success when is 0. Posteriorly, the signals mentioned above are processed, filtered and used in the estimations. The pre-processing, in short, corresponds to the calibration and the removal of

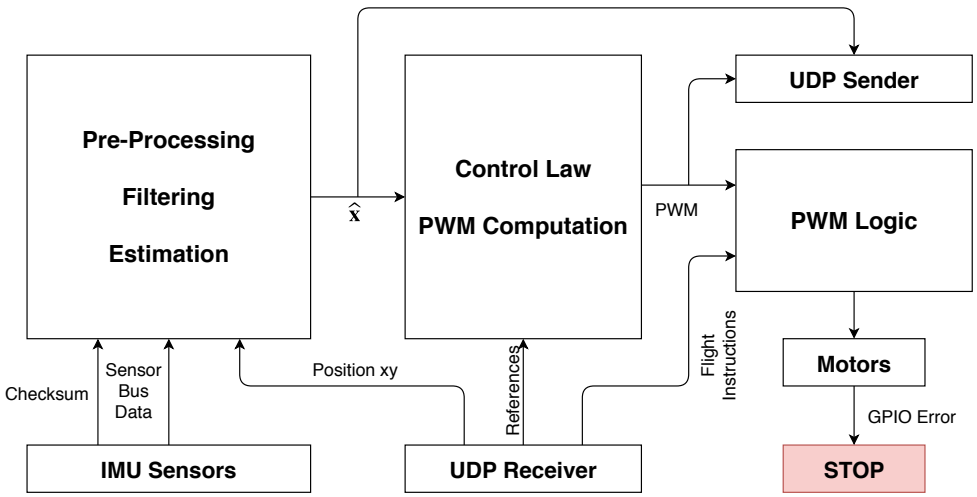

Figure 8.2: Structure of the control software implement in Simulink that originates the binary file deployed to the UAV operating system.

the offsets of the sensor measurements. Afterwards, the gyroscope, accelerometer and magnetometer measurements are filtered using Kalman filters. After the filtering, these vectors are used in the estimation of the Euler angles, resorting to one of the attitude filters described in section 6.2. The velocities are also estimated in this block, using for this end the Kalman filters designed in 6.1.1.

The control law is applied in the "Control Law / PWM Computation" block, whose inputs are the state-variables, obtained from sensory data and estimation, and the references, yielding the thrust and moments required. The resultant control vector is subsequently transformed into PWM Commands using the formulas detailed in Appendix A.

The "PWM Logic" block manages the after-effect of the application of the control law by considering the flight instructions received from the host computer. When the emergency command is 0 and the take-off signal is 1, the PWM commands sent to the motors are equal to the computed in the previous block. If the emergency command is equal to 1, a vector of zeros is sent to the motors to stop the flight immediately. Finally, if the take-off signal is equal to 0, the PWM commands are reduced progressively to land the UAV smoothly.

The "Motors" block converts the four PWM values arising from the "PWM Logic" block into a 40-bit number that is fed to the controllers of the rotors. Additionally, it outputs a GPIO ERROR signal that is 1 when the propellers are blocked, causing the "Stop" block to terminate the control software, and 0 otherwise.

8.2 Linear Approach

The experimental results obtained with the selected linear control approach are now presented and discussed. The trajectory tested is equal to the simulated one and, by observing Fig. 8.3, it is possible to verify a good tracking by the quadrotor. It is important to stress that the take-off is present in the results depicted and that the altimeter does not measure heights inferior to 30 cm.

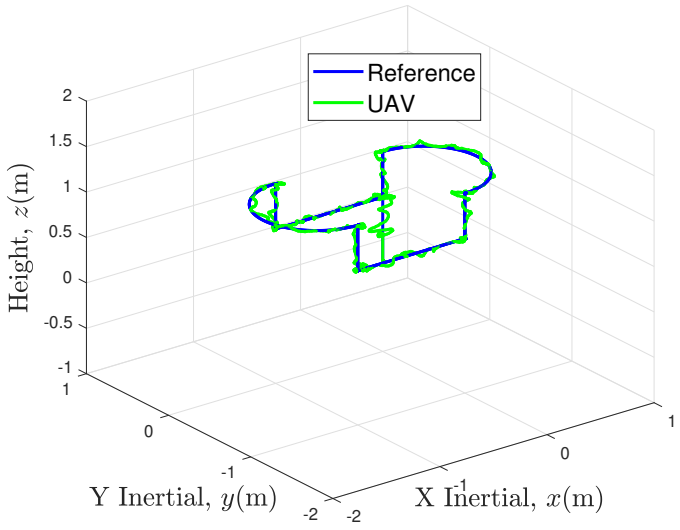

Figure 8.3: Trajectory tracking in 3D space obtained with the selected linear control approach.

The controllers had to be adjusted when implemented in the UAV. The matrices $\mathbf{Q}$ and $\mathbf{R}$ used in this adjustment are detailed in Table. 8.1.

Subsystem	$\mathbf{Q}$	$\mathbf{R}$
Yaw	diag$(18, 2, 2)$	30
X Position	diag$(3, 1, 2)$	300
Y Position	diag$(3, 1, 2)$	500
Height	diag$(3, 1, 1)$	1

Table 8.1: $\mathbf{Q}$ and $\mathbf{R}$ matrices used in the adjustment of the linear controllers implemented in the UAV.

This new set of controllers was also used for generating new simulation responses to compare with the experimental results. By comparing the UAV results and the simulation responses displayed in Fig. 8.4, the similarities between them are manifest, which indicates that the nonlinear model considered possesses a satisfactory degree of proximity. Note that, for the yaw angle case, only the reference and the quadrotor response are depicted since the simulation presented residual error and, consequently, considering the scale used, would be indistinguishable from the reference.

For attitude estimation, the Madgwick attitude filter was considered in this approach. The tuning parameter β used was 0.05. The roll and pitch angles estimated during the take-off and the following of the trajectory are depicted in Fig. 8.5. It is noticeable that these angles presented reduced values during the totality of the experimental test. Hence, the small angle approximation considered in the linearization of the dynamics was verified, which contributed to attaining a good performance overall with the linear control implemented.

The thrust force and moments computed by the control structure are shown in Fig. 8.6. A heavier battery was used in the tests, leading to higher values of thrust. Nonetheless, despite this change in the total mass of the UAV, since the integral action was included, the results did not deteriorate.

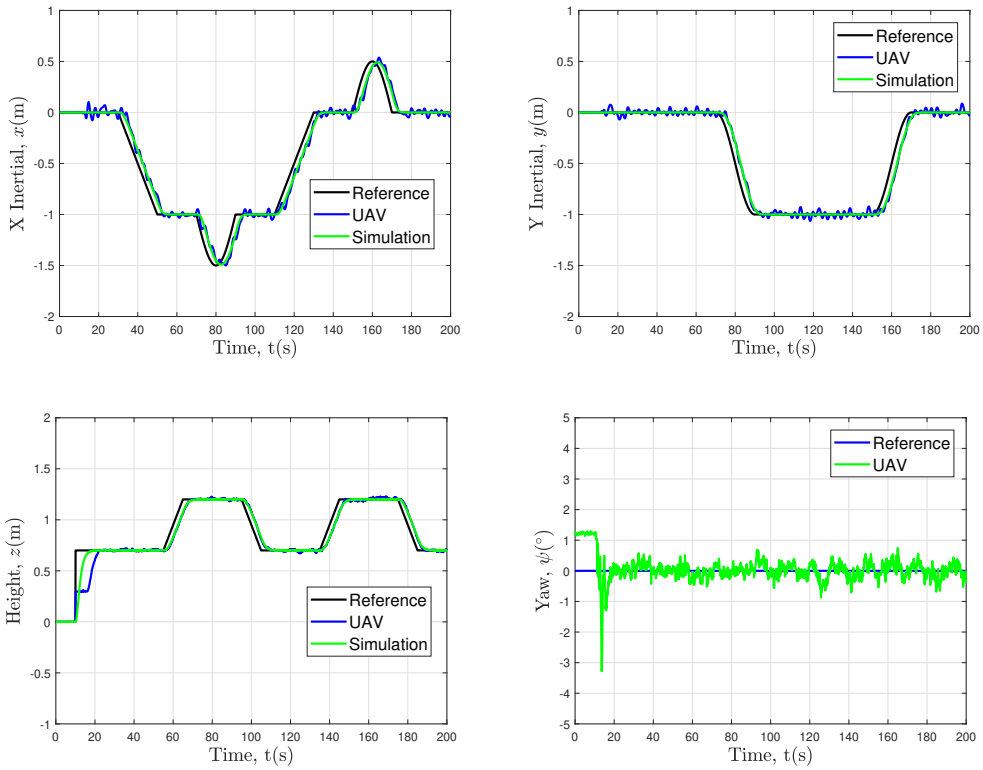

Figure 8.4: Responses obtained during trajectory tracking in simulation and in the experimental tests with the selected linear control approach. From left to right, top to bottom: (a) X Inertial; (b) Y Inertial; (c) Height; (d) Yaw.

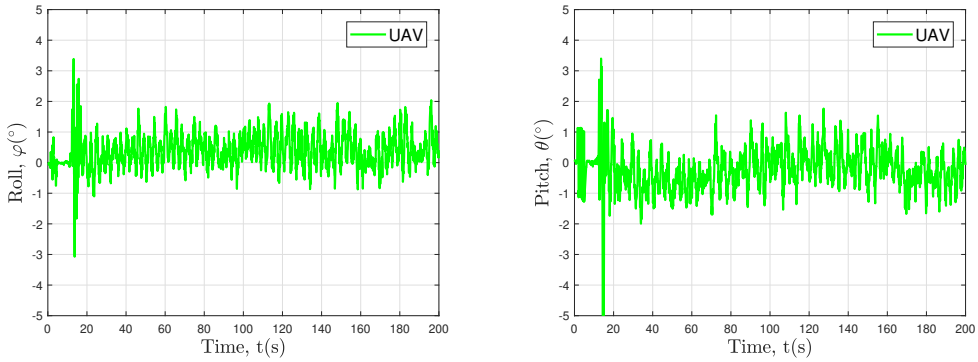

Figure 8.5: Roll and pitch angles responses obtained during trajectory tracking in experimental test using the selected linear control approach. From left to right: (a) Roll; (b) Pitch.

Regarding the moments, the higher values were required when the position control was activated, since the quadcopter had to compensate for the positional drift that occurred during the take-off. Throughout the tracking, due to the trajectory not being aggressive, the actuation did not present abrupt variations.

From Fig. 8.7, where the PWM commands are displayed, one notes that the actuation did not saturate during the take-off and the trajectory tracking.

The root-mean-square error for the trajectory tracking in simulation an in the real system is detailed in Table. 8.2. The height response obtained with the quadrotor has almost the same error that was obtained in simulation, whereas the other subsystems showed a forecastable increase in the real system due to

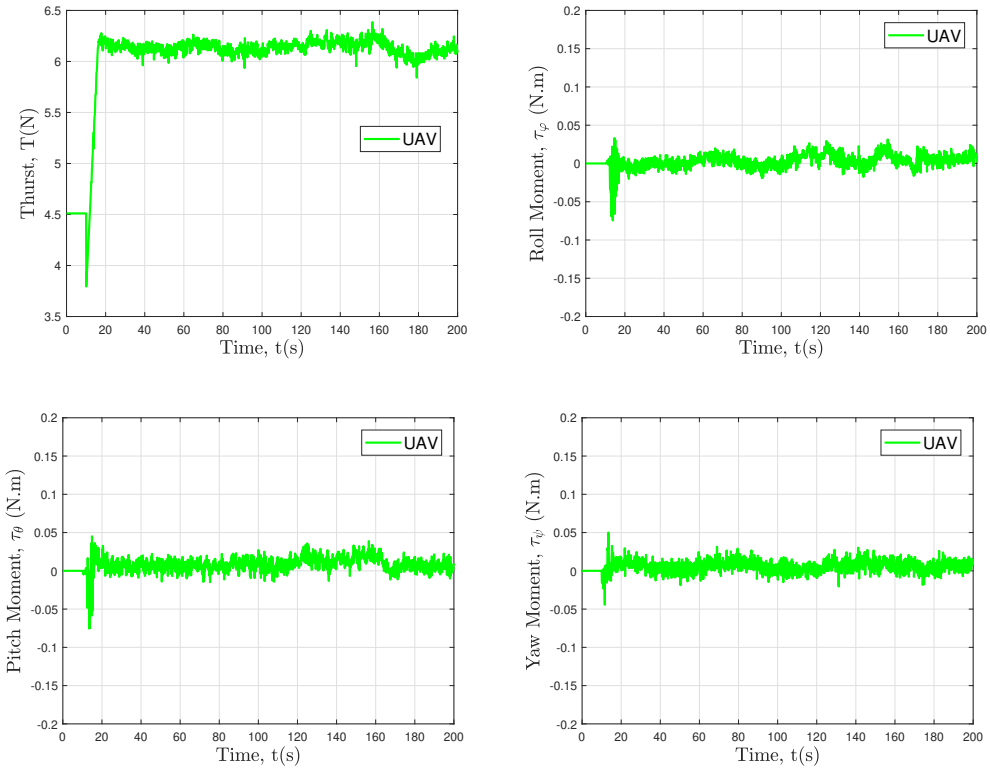

Figure 8.6: Actuation during experimental test with the selected linear control approach. From left to right, top to bottom: (a) Thrust; (b) Roll Moment; (c) Pitch Moment; (d) Yaw Moment.

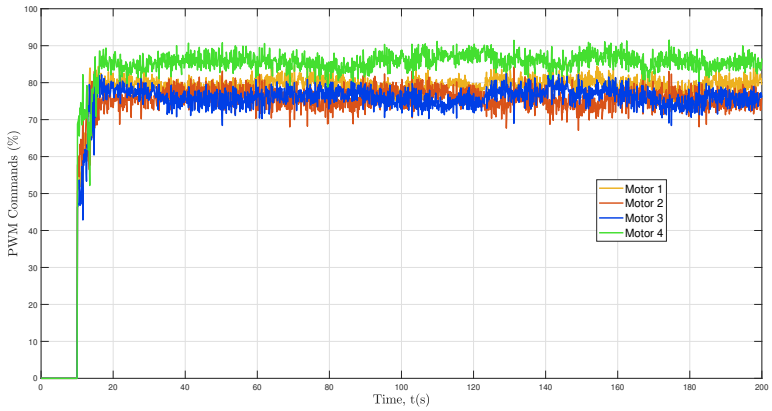

Figure 8.7: PWM Commands computed during trajectory tracking with the selected linear control approach.

the discarding of the higher order dynamics effects. In general, the errors were kept under reasonable values with the transition to the drone.

	x (m)	y (m)	z (m)	ψ (°)
Simulation	0.0865	0.0714	0.0556	0.0295
Experimental	0.1010	0.0781	0.0570	0.2244

Table 8.2: Root-mean-square error obtained in simulation and in the experimental test with the selected linear control approach.

The histograms of the absolute error for the inertial coordinates and the yaw angle during the tracking of the trajectory (initiates at 20 seconds) are presented in Fig. 8.8. Even though the deviations are larger in the real system, they are still considered reduced. The local maxima visible in the position histograms are a consequence of the static error in the following of ramp inputs. Nevertheless, the maximum instances of error for all of these coordinates correspond approximately to zero, which is symptomatic of the capacity of the control system to maintain these coordinates at a constant value. The inertial coordinate x presented the higher deviations, which was expected since that was the subsystem more solicited. The values of the yaw angle obtained demonstrate that the goal defined for this Euler angle was achieved.

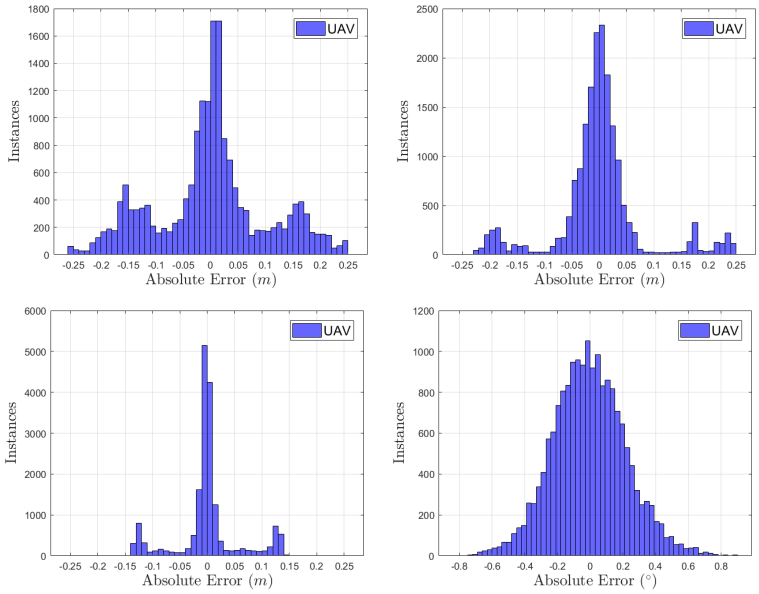

Figure 8.8: Absolute error histograms obtained with the selected linear control approach. Left to right, top to bottom: (a) X Inertial; (b) Y Inertial; (c) Height and (d) Yaw.

8.3 Nonlinear Approach

Implementing the static feedback linearization with zero-dynamics stabilization led to the experimental results that are exhibited and discussed in this subsection. Through observation of Fig. 8.9, one notices that this approach achieved a successful tracking of the predefined trajectory. Similar to the linear control results, the take-off is displayed as well and the previously referred saturation of the altimeter measurements is also evident.

The first attitude controllers designed in simulation led to an oscillatory behavior in the actual aerial vehicle. In furtherance of correcting this issue, the third diagonal entry of the matrix $\mathbf{Q}$ was reduced so the integral of the error would not be so penalized in the cost function and could achieve slightly bigger values. The remaining diagonal entries were adjusted while targeting the fulfillment of the simulation design criteria. Concerning the outer loop control, tests before the trajectory tracking experiment demonstrated the need to decrease the penalization considered in the quadratic cost function associated

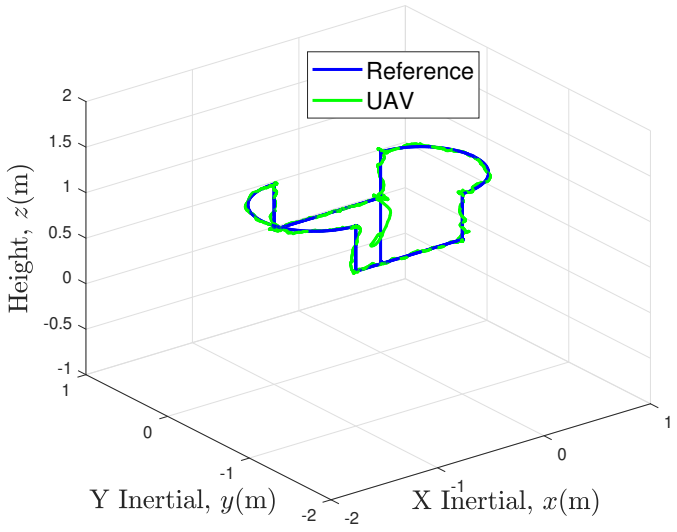

Figure 8.9: Trajectory tracking in 3D space obtained with the selected nonlinear control approach.

Subsystem	$\mathbf{Q}$	$\mathbf{R}$
Roll	diag$(7.5 \times 10^3, 20, 2 \times 10^5)$	1
Pitch	diag$(7.5 \times 10^3, 20, 2 \times 10^5)$	1
Yaw	diag$(7.5 \times 10^3, 50, 1 \times 10^5)$	1
X Position	diag$(5, 1 \times 10^{-3}, 40)$	2×10^3
Y Position	diag$(5, 1 \times 10^{-3}, 40)$	2×10^3

Table 8.3: $\mathbf{Q}$ and $\mathbf{R}$ matrices used in the adjustment of the controllers of the selected nonlinear approach implemented in the UAV.

with the velocity. New simulations responses were obtained with these new controllers in furtherance of comparing with the experimental results. In Fig. 8.10, the simulation and experimental responses are depicted. Identically to the linear control approach, similarities between the two results are evident, demonstrating once more the proximity between the nonlinear model developed and the actual system.

For attitude estimation, the nonlinear attitude filter proposed by João Madeiras [66] was implemented in this approach. The filters parameters used are equal to the reported in his work [66]. The roll and pitch angles estimated during the take-off and the following of the trajectory are depicted in Fig. 8.11.

The thurst force and moments computed by the control structure are shown in Fig. 8.12. Once again, a heavier battery was used, leading to higher values of thrust, and the integral action was able to overcome this change.

In order to compare the actuation computed by each control scheme implemented, the variance of the resulting force and moments is computed. To this end, the control actions are smoothed resorting to a moving-average with a sliding window of 400 points. Then, these smoothed signals are subtracted from the forces and moments to extract the variations. Finally, the variance of the computed difference for each actuation is obtained. From Table 8.4, one verifies that the nonlinear control origins thrust and moments with lower variance than the linear control considered.

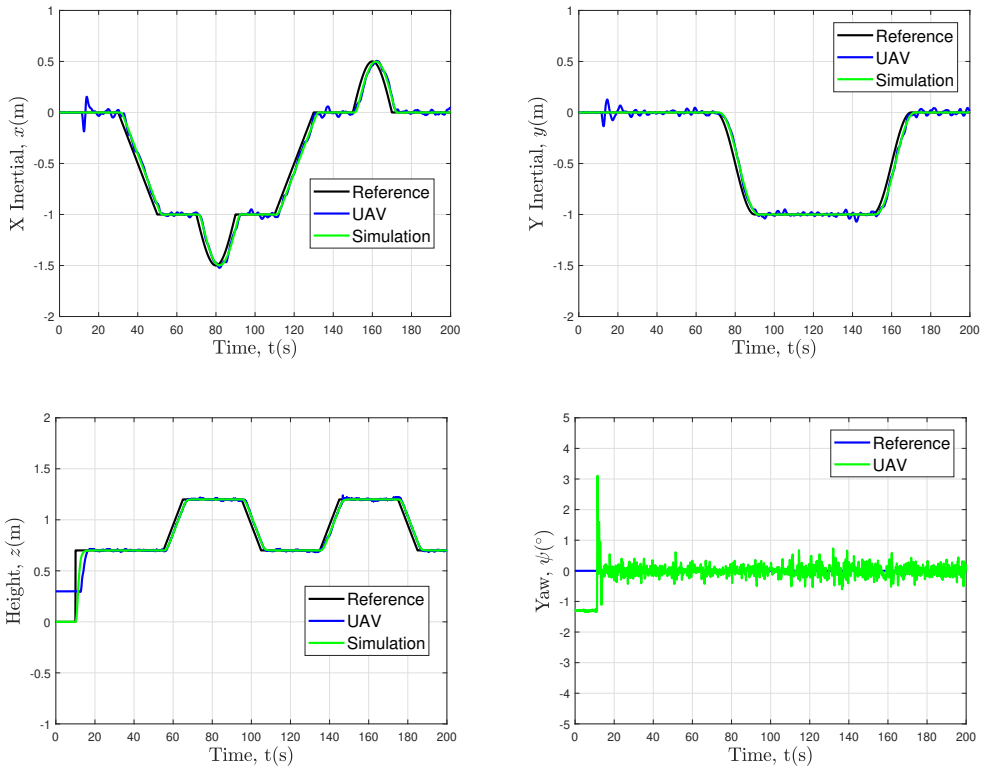

Figure 8.10: Responses obtained during trajectory tracking in simulation and in the experimental tests with the selected nonlinear control approach. From left to right, top to bottom: (a) X Inertial; (b) Y Inertial; (c) Height; (d) Yaw.

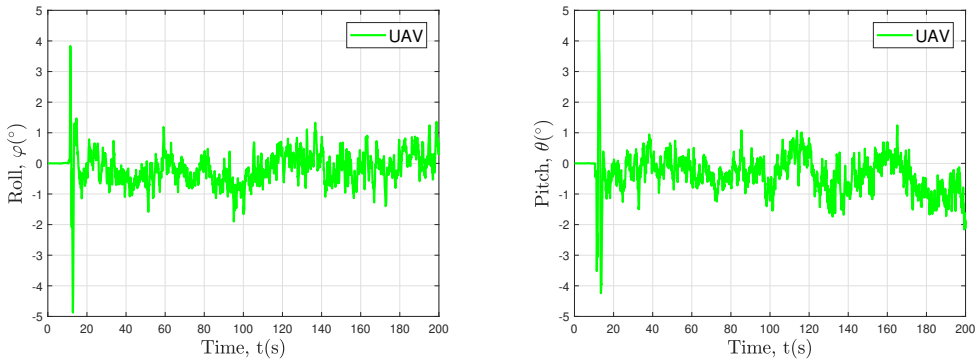

Figure 8.11: Roll and pitch angles responses obtained during trajectory tracking in experimental test using the selected nonlinear control approach. From left to right: (a) Roll; (b) Pitch.

Control Approach	T (N)	τ_φ (N · m)	τ_θ (N · m)	τ_ψ (N · m)
Linear	1.50×10^{-3}	4.47×10^{-5}	3.20×10^{-5}	3.81×10^{-5}
Nonlinear	1.30×10^{-3}	5.92×10^{-6}	9.99×10^{-6}	2.62×10^{-5}

Table 8.4: Variance of the Thrust and Moments computed by each control approach implemented in the quadcopter.

From Fig. 8.13, is visible that the PWM commands did not saturate during the experimental test.

The root-mean-square error for the trajectory tracking in simulation an in the real system obtained

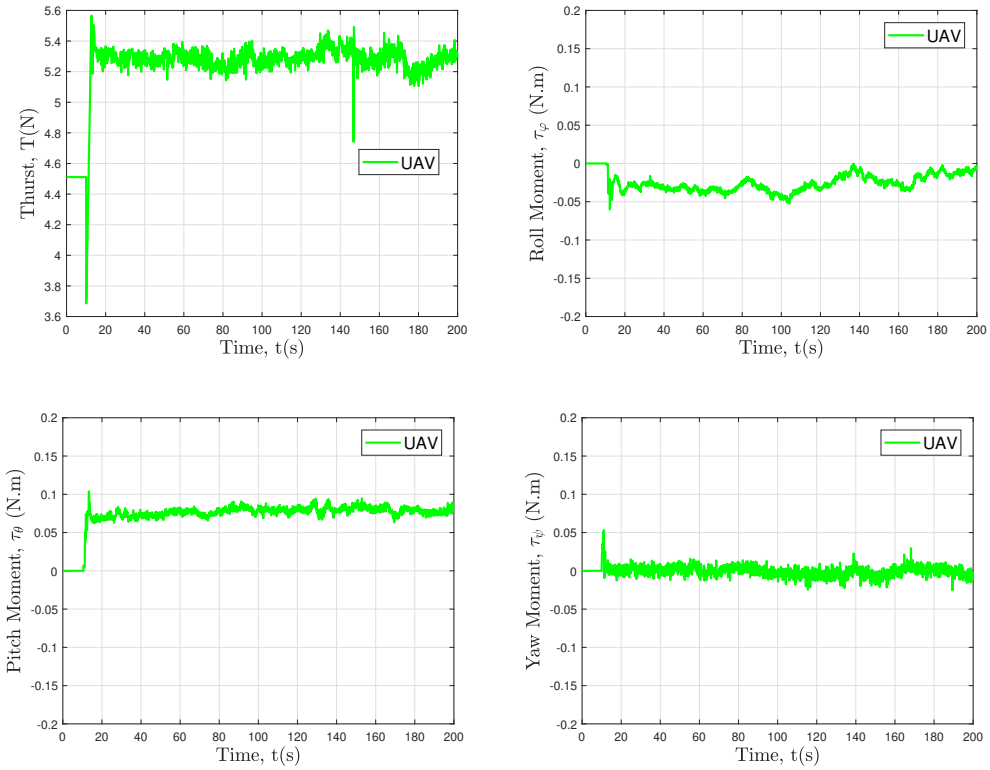

Figure 8.12: Actuation during the experimental test with the selected nonlinear control approach. From left to right, top to bottom: (a) Thrust; (b) Roll Moment; (c) Pitch Moment; (d) Yaw Moment.

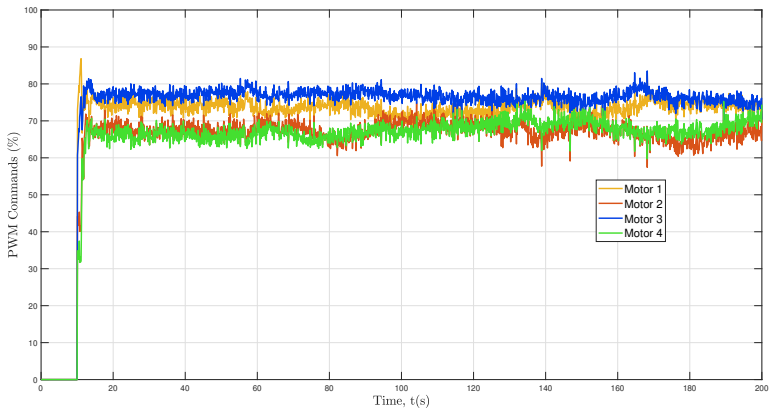

Figure 8.13: PWM Commands computed during trajectory tracking with the selected nonlinear control approach.

with the nonlinear control approach is detailed in Table. 8.5. Compared to the linear control implemented, it is noticeable that the error decreased with the application of the static feedback linearization technique to the altitude and attitude dynamics. Furthermore, this nonlinear method enabled achieving experimental results more similar to the simulation. The improvement achieved with the nonlinear control is partly due to the faster position responses, which has as a consequence lower steady-state error in the following of ramp inputs.

By analyzing the histograms of the absolute error for the inertial coordinates and the yaw angle

	x (m)	y (m)	z (m)	ψ (°)
Simulation	0.0632	0.0482	0.0385	0.0223
Experimental	0.0666	0.0518	0.0394	0.1715

Table 8.5: Root-mean-square error obtained in simulation and in the experimental test with the selected nonlinear control approach.

during the tracking trajectory (initiates at 20 seconds) displayed in Fig. 8.14, it is noted that a higher number of approximately zero absolute error instances was obtained with the second control approach implemented. Moreover, as a result of presenting faster responses, this approach has the local maxima of the histograms closer to zero than the previous control structure considered. Analogously to the inner-outer control using LQR with integrative action, the response with more deviations is the inertial coordinate x and the one with less is the height, which is the affected at the smallest scale by unmodelled high-order effects. Overall, the implementation of the static feedback linearization inner-loop improved the tracking performance of the control structure, not only attaining the objectives but also leading to better results.

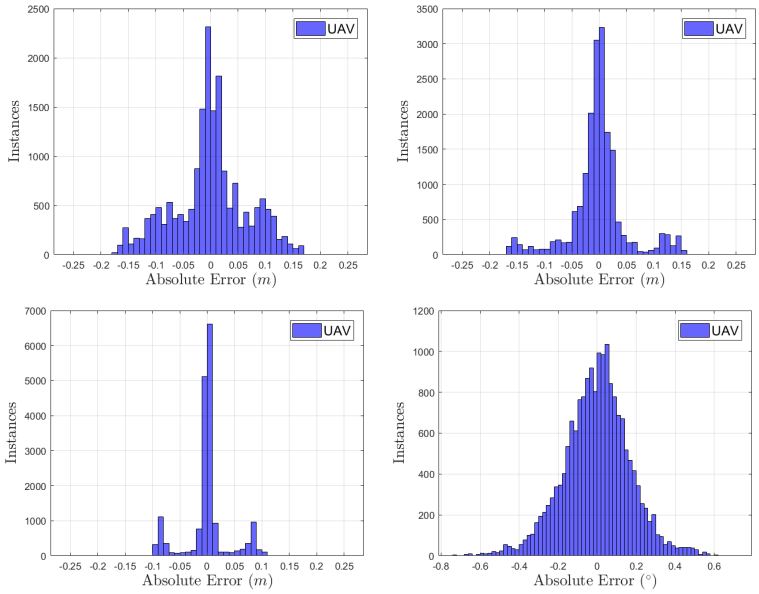

Figure 8.14: Absolute error histograms obtained with the selected nonlinear control approach. Left to right, top to bottom: (a) X Inertial; (b) Y Inertial; (c) Height and (d) Yaw.

Chapter 9

Conclusions

The work developed throughout this thesis aimed to provide solutions to full control a quadcopter. Linear and nonlinear techniques were applied to tackle this problem, with different architectures being studied and devised for each technique. The developed strategies were all tested in simulation, which allowed to determine the most auspicious. The selected approaches were successfully validated in a commercially available quadcopter.

The dynamical model of the UAV considered, obtained resorting to the Newton-Euler formalism, proved to be accurate and sufficient, even though higher order effects were neglected. The first solution presented consists of an inner-outer loop control structure, constituted by linear quadratic controllers with integrative action whose design relied on the linearization of the nonlinear dynamics, an attitude filter, based on an optimized gradient-descent algorithm, and Kalman filters to estimate the velocities. The second solution conceived, in terms of control, results from the application of Feedback Linearization to the attitude and altitude dynamics and has its horizontal movement controlled resorting to an LQR-based outer-loop with integrative action. Concerning the estimation, this solution relies on Kalman filters to estimate the Euler angles and the velocities. These two strategies correspond, respectively, to the linear and nonlinear control structures that presented better results in the simulation tests conducted. Both schemes demonstrated the capacity to seamlessly follow a predefined trajectory in the presence of measurement noises in simulation. Furthermore, in the robustness test to model parameters variation, the first architecture was able to meet the performance criteria when simultaneous variations of 50% in the mass and inertia were considered, which constitutes an interesting level of robustness for a linear approach. The nonlinear solution was capable of handling variations of 50% in the mass together with inertia values four times lower and four times higher than the value considered in the model without considerably affecting its responses. Further tests, in which the integrative action was removed from the architecture of the nonlinear strategy, evidenced the critical role of this action in diminishing the model dependence and in increasing the robustness of the control architecture.

Both control solutions were successfully implemented on an off-the-shelf quadrotor. The results obtained in trajectory tracking validated the proposed strategies and, given the manifest similarities between the attained responses with the actual aerial vehicle and in simulation, evidenced the satisfactory

degree of accuracy of the nonlinear model considered. The integrative action present in both architectures led to null steady-state position error, even when a heavier battery was used. The small angle approximation considered in the linearization of the dynamics was verified, which contributed to achieving good overall performance with the linear controller. Notwithstanding, the application of the static feedback linearization to the attitude and altitude dynamics improved the performance since the obtained position and yaw angle error was inferior compared with the obtained with the first solution. The highest root-mean-square error obtained with the linear approach was $0.1010\ m$, for the x inertial coordinate and, with the nonlinear control scheme, was $0.0666\ m$, verified for the same coordinate. Moreover, the second control scheme originated an actuation with less variation than the first solution.

In summary, the primary objective of this work was achieved. Two different control solutions, using linear and nonlinear methods, capable of following a predefined trajectory and presenting null steady-state position error and a reasonable level of robustness to model imprecision, uncertainty and disturbances, were successfully devised, tested in simulation and validated in experimental tests resorting to a commercially available quadcopter.

9.1 Future Work

In the interest of proceeding with the work developed during this thesis, the future work should not only focus on extending the capacities of the proposed solutions but also on improving the results. Eliminating the steady-state velocity error and applying sliding mode control simultaneously with Feedback Linearization, thus increasing the robustness and dealing more effectively with disturbances, are some possibilities that can enhance the solutions proposed.

Disturbances and actuators saturation, for instance, can be equated in the formulation of the problem and defining a Lyapunov function in order to prove stability can be explored.

Ultimately, the future work must aim outdoor flight through a versatile nonlinear control solution that only relies on onboard sensors. In furtherance of achieving this goal, a reliable optical flow has to be implemented and the disturbance rejection capacity of the control structure should be improved.

Bibliography

[1] S. Bouabdallah. *Design and control of quadrotors with application to autonomous flying*. PhD thesis, 2007.

[2] S. B. Anderson and V. Lebacqz. Historical overview of v/stol aircraft technology. February 1997.

[3] J. G. Leishman. *Principles of helicopter aerodynamics*. Cambridge University Press, 2000.

[4] Clarity from above. pwc global report on the commercial applications of drone technology, May 2016. URL `https://www.pwc.pl/clarityfromabove`.

[5] H. t. M. N. ElKhody. Dynamic modeling and control of a quadrotor using linear and nonlinear approaches. Master's thesis, School of Sciences and Engineering, Cairo, Egypt, 2014.

[6] L. R. G. Carrillo, A. E. López, R. Lozano, and C. Pégard. *Quad rotorcraft control: Vision-Based Hovering and Navigation*. Springer, 2013.

[7] H. Shakhatreh, A. Sawalmeh, A. Al-Fuqaha, Z. Dou, E. Almaita, I. Khalil, N. Othman, A. Khreishah, and M. Guizani. Unmanned aerial vehicles: A survey on civil applications and key research challenges, April 2018.

[8] E. Tuyishimire, A. Bagula, S. Rekhis, and N. Boudriga. Cooperative data muling from ground sensors to base stations using uavs. In *2017 IEEE Symposium on Computers and Communications (ISCC)*, pages 35–41, July 2017.

[9] F. Mohamadi. Vertical takeoff and landing (vtol) small unmanned aerial system for monitoring oil and gas pipelines, November 2014. URL `http://www.google.it/patents/US4741207`. US Patent 8,880,241.

[10] B. E. Schäfer, D. Picchi, T. Engelhardt, and D. Abel. Multicopter unmanned aerial vehicle for automated inspection of wind turbines. In *2016 24th Mediterranean Conference on Control and Automation (MED)*, pages 244–249, June 2016.

[11] J. P. Sinha, H. Kushwaha, D. Kushwaha, N. Singh, and M. Purushottam. Prospect of unmanned aerial vehicle (uav) technology for agricultural production management. In *International Conference on Emerging Technologies in Agricultural and Food Engineering*, December 2016.

[12] A. Raptopoulos. No roads? there's a drone for that, November 2013. URL https://www.youtube.com/watch?v=9yEl0-bCA9M.

[13] E. McCollom. The future of distribution - part ii: Product distribution in emerging markets, June 2018. URL https://redstagfulfillment.com/future-distribution-part-ii-product-distribution-emerging-markets/.

[14] Drones going postal - a summary of postal service delivery drones, October 2016. URL http://unmannedcargo.org/drones-going-postal-summary-postal-service-delivery-drone-trials/.

[15] S. Qazi, A. S. Siddiqui, and A. I. Wagan. Uav based real time video surveillance over 4g lte. *2015 International Conference on Open Source Systems & Technologies (ICOSST)*, December 2015.

[16] T. Hamel, R. Mahony, R. Lozano, and J. Ostrowski. Dynamic modelling and configuration stabilization for an x4-flyer. *IFAC Proceedings Volumes*, 35(1):217–222, 2002.

[17] P. Pounds, R. Mahony, and P. Corke. Modelling and control of a quad-rotor robot. In *Australasian Conference on Robotics and Automation 2006*, Auckland, New Zealand, 2006. Australian Robotics and Automation Association Inc.

[18] P. Pounds, R. Mahony, and P. Corke. Modelling and control of a large quadrotor robot. *Control Engineering Practice*, 18(7):691–699, 2010.

[19] R. Mahony, V. Kumar, and P. Corke. Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor. *IEEE Robotics & Automation Magazine*, 19(3):20–32, 2012.

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

[21] G. Hoffmann, H. Huang, S. Waslander, and C. Tomlin. Quadrotor helicopter flight dynamics and control: Theory and experiment. *AIAA Guidance, Navigation and Control Conference and Exhibit*, 2007.

[22] J. H. Gillula, H. Huang, M. P. Vitus, and C. J. Tomlin. Design of guaranteed safe maneuvers using reachable sets: Autonomous quadrotor aerobatics in theory and practice. *2010 IEEE International Conference on Robotics and Automation*, page 1649–1654, 2010.

[23] J. Li and Y. Li. Dynamic analysis and pid control for a quadrotor. *2011 IEEE International Conference on Mechatronics and Automation*, page 573–578, 2011. doi: 10.1109/icma.2011.5985724.

[24] I. D. Cowling, O. A. Yakimenko, J. F. Whidborne, and A. K. Cooke. A prototype of an autonomous controller for a quadrotor uav. In *European Control Conference (ECC)*, pages 4001–4008, July 2007.

[25] M. Valenti, B. Bethke, G. Fiore, J. How, and E. Feron. Indoor multi-vehicle flight testbed for fault detection, isolation, and recovery. *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2006.

[26] P. Bauer and J. Bokor. Lq servo control design with kalman filter for a quadrotor uav. *Periodica Polytechnica Transportation Engineering*, 36(1-2):9–14, 2008.

[27] M. Raja. Extended kalman filter and lqr controller design for quadrotor uavs. Master's thesis, Wright state University, Dayton,Ohio, 2017.

[28] J.-J. E. Slotine and W. Li. *Applied Nonlinear Control: An Introduction*. Prentice-Hall, 1991.

[29] N. S. Özbek, M. Önkol, and M. n. Efe. Feedback control strategies for quadrotor-type aerial robots: a survey. *Transactions of the Institute of Measurement and Control*, 38(5):529–554, 2015.

[30] S. Bouabdallah and R. Siegwart. Backstepping and sliding-mode techniques applied to an indoor micro quadrotor. *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, page 2247–2252, 2005.

[31] R. Xu and U. Ozguner. Sliding mode control of a quadrotor helicopter. In *Proceedings of the 45th IEEE Conference on Decision and Control*, pages 4957–4962. IEEE, 2006.

[32] D. Xia, L. Cheng, and Y. Yao. A robust inner and outer loop control method for trajectory tracking of a quadrotor. *Sensors*, 17(9):2147, 2017.

[33] U. Ansari and A. H. Bajodah. Robust generalized dynamic inversion quadrotor control. *IFAC-PapersOnLine*, 50(1):8181–8188, 2017.

[34] İlyas Eker. Second-order sliding mode control with experimental application. *ISA Transactions*, 49 (3):394–405, 2010.

[35] T. Madani and A. Benallegue. Backstepping control for a quadrotor helicopter. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3255–3260. IEEE, 2006.

[36] H. Lee, S. Kim, T. Ryan, and H. J. Kim. Backstepping control on se (3) of a micro quadrotor for stable trajectory tracking. In *2013 IEEE International Conference on Systems, Man, and Cybernetics*, pages 4522–4527. IEEE, 2013.

[37] A. Das, K. Subbarao, and F. Lewis. Dynamic inversion of quadrotor with zero-dynamics stabilization. *2008 IEEE International Conference on Control Applications*, 2008.

[38] A. Freddi, A. Lanzon, and S. Longhi. A feedback linearization approach to fault tolerance in quadrotor vehicles. *IFAC Proceedings Volumes*, 44(1):5413–5418, 2011.

[39] J. Wang, T. Bierling, M. Achtelik, L. Hocht, F. Holzapfel, W. Zhao, and T. H. Go. Attitude free position control of a quadcopter using dynamic inversion. *Infotech@Aerospace 2011*, Mar 2011.

[40] K. P. Valavanis and G. J. Vachtsevanos. *Handbook of Unmanned Aerial Vehicles*. Springer, 2015.

[41] K. Alexis, G. Nikolakopoulos, and A. Tzes. Switching model predictive attitude control for a quadrotor helicopter subject to atmospheric disturbances. *Control Engineering Practice*, 19(10):1195–1207, 2011.

[42] A. Roberts and A. Tayebi. Adaptive position tracking of vtol uavs. *IEEE Transactions on Robotics*, 27(1):129–142, 2010.

[43] R. Pérez-alcocer and J. Moreno-valenzuela. Adaptive control for quadrotor trajectory tracking with accurate parametrization. *IEEE Access*, 2019.

[44] G. V. Raffo, M. G. Ortega, and F. R. Rubio. An integral predictive/nonlinear $\mathcal{H}_\infty$ control structure for a quadrotor helicopter. *Automatica*, 46(1):29–39, 2010.

[45] Parrot ar.drone 2.0 elite edition, Oct 2018. URL `https://www.parrot.com/global/drones/parrot-ardrone-20-elite-edition#parrot-ardrone-20-elite-edition`.

[46] J. Martin, Aug 2014. URL `https://archive.eetasia.com/www.eetasia.com/ART_8800702856_1034362_NT_0c5640bf.HTM`.

[47] G. Oriolo, L. Sciavicco, B. Siciliano, and L. Villani. *Robotics: Modelling, Planning and Control*. Springer, 2010.

[48] L. Qianying. Grey-box system identification of a quadrotor unmanned aerial vehicle. Master's thesis, Delft University of Technology, Delft, Netherlands, 2014.

[49] K. Ogata. *Modern Control Engineering*. Prentice Hall, 2002.

[50] R. Stengel. *Optimal Control and Estimation*. Dover Publications, 1994.

[51] A. E. Bryson and Y.-C. Ho. *Applied optimal control optimization, estimation, and control*. Hemisphere Publishing Corporation, 1986.

[52] G. F. Franklin and J. D. Powell. *Digital control of dynamic systems*. Addison-Wesley, 1980.

[53] M. Henson and D. E. Seborg. *Nonlinear process control*. Prentice Hall PTR, 1997.

[54] A. Isidori. *Nonlinear Control Systems*. Springer-Verlag, 1995.

[55] H. K. Khalil. *Nonlinear systems*. Prentice Hall, 2002.

[56] R. Vepa. *Nonlinear Control of Robots and Unmanned Aerial Vehicles: an integrated approach*. CRC Press Taylor & Francis Group, 2017.

[57] D. Lee, H. J. Kim, and S. Sastry. Feedback linearization vs. adaptive sliding mode control for a quadrotor helicopter. *International Journal of Control, Automation and Systems*, 7(3):419–428, 2009.

[58] F. Sabatino. Quadrotor control: modeling, nonlinear control design, and simulation. Master's thesis, KTH Electrical Engineering, Stockholm, 2015.

[59] R. Bonna and J. Camino. Trajectory tracking control of a quadrotor using feedback linearization. 02 2015.

[60] A. Roza and M. Maggiore. Path following controller for a quadrotor helicopter. *2012 American Control Conference (ACC)*, page 4655–4660, 2012.

[61] S. Zhang, F. Liao, Y. Chen, and K.-F. He. Aircraft trajectory control with feedback linearization for general nonlinear system. *CoRR*, abs/1712.10077, 12 2017.

[62] R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45, 1960.

[63] A. Kelly. A 3d state space formulation of a navigation kalman filter for autonomous vehicles. 1994.

[64] Y. Bar-Shalom, T. Kirubarajan, and X.-R. Li. *Estimation with Applications to Tracking and Navigation*. Wiley, 2007.

[65] S. O. H. Madgwick, A. J. L. Harrison, and R. Vaidyanathan. Estimation of imu and marg orientation using a gradient descent algorithm. In *2011 IEEE International Conference on Rehabilitation Robotics*, pages 1–7, June 2011.

[66] J. Madeiras, C. Cardeira, and P. Oliveira. Complementary filter vision-aided for attitude and position estimation: Design, analysis and experimental validation. 2019.

[67] D. Lee. Ar.drone 2.0 support from embedded coder, 2016. URL `https://www.mathworks.com/hardware-support/ar-drone.html`.

[68] 5 , 6 and 7 series. URL `https://www.qualisys.com/hardware/5-6-7/`.

[69] N. Jeurgens. Identification and control implementation of an ar.drone 2.0. Master's thesis, Eindhoven University of Technology, Eindhoven, Netherlands, 2017.

Appendix A

PWM Commands Computation

The Parrot AR.Drone 2.0 has four rotors that are respectively driven by four motors, generating forces and moments that are applied to the body of the quadrotor. The angular velocity of these motors are controlled by an electronic speed controller that receives Pulse Width Modulation (PWM) signal [69].The PWM Commands are computed from the Thrust and Moments that result from the control law and range from 0 to 100%.

In order to determine the referred relations, an experiment, similar to the one conducted in [48], was performed. As a result, for each rotor, the thrust generated with PWM commands ranging from 0 to 100% with increments of 5% was computed. From the previous works followed, one notes that the relation between the PWM command and the Thrust generated by a rotor is quadratic. Therefore, a second-order polynomial was used to approximate the trends of each data set. The fitting is displayed in Fig. A.1, from which is noted that slightly different results were achieved for each rotor.

$$\begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \end{bmatrix} = \begin{bmatrix} 1.1400 \times 10^{-4} \cdot PWM_1^2 + 1.0180 \times 10^{-2} \cdot PWM_1 + 6.5833 \times 10^{-2} \\ 1.1544 \times 10^{-4} \cdot PWM_2^2 + 7.7509 \times 10^{-3} \cdot PWM_2 + 9.5961 \times 10^{-2} \\ 1.2605 \times 10^{-4} \cdot PWM_3^2 + 9.2952 \times 10^{-3} \cdot PWM_3 + 7.2140 \times 10^{-2} \\ 1.3045 \times 10^{-4} \cdot PWM_4^2 + 7.5567 \times 10^{-3} \cdot PWM_4 + 8.2877 \times 10^{-2} \end{bmatrix} \tag{A.1}$$

where PWM_i denotes the PWM commands sent to the rotor i. Concerning the yaw moments τ_{ψ_i}, since the experiment is more complicated to be performed and requires extra tools that were unavailable, the equations presented in [69] were considered. Note that the quadrotor used in this work is also the Parrot AR. Drone 2.0. Hence, these relations will not differ significantly. The formulas for each rotor that relate the generated yaw moment with the PWM commands received are the following:

$$\begin{bmatrix} \tau_{\psi_1} \\ \tau_{\psi_2} \\ \tau_{\psi_3} \\ \tau_{\psi_4} \end{bmatrix} = \begin{bmatrix} 3.4103 \times 10^{-6} \cdot PWM_1^2 + 1.9283 \times 10^{-4} \cdot PWM_1 + 1.3043 \times 10^{-3} \\ 2.9756 \times 10^{-6} \cdot PWM_2^2 + 3.8093 \times 10^{-4} \cdot PWM_2 + 5.1923 \times 10^{-3} \\ 4.5795 \times 10^{-6} \cdot PWM_3^2 + 3.1301 \times 10^{-4} \cdot PWM_3 + 2.3996 \times 10^{-3} \\ 7.2886 \times 10^{-6} \cdot PWM_4^2 + 1.2779 \times 10^{-4} \cdot PWM_4 + 4.6184 \times 10^{-3} \end{bmatrix} \tag{A.2}$$

From equation 3.7, one notes the linearity between the yaw moment and the thrust generated by

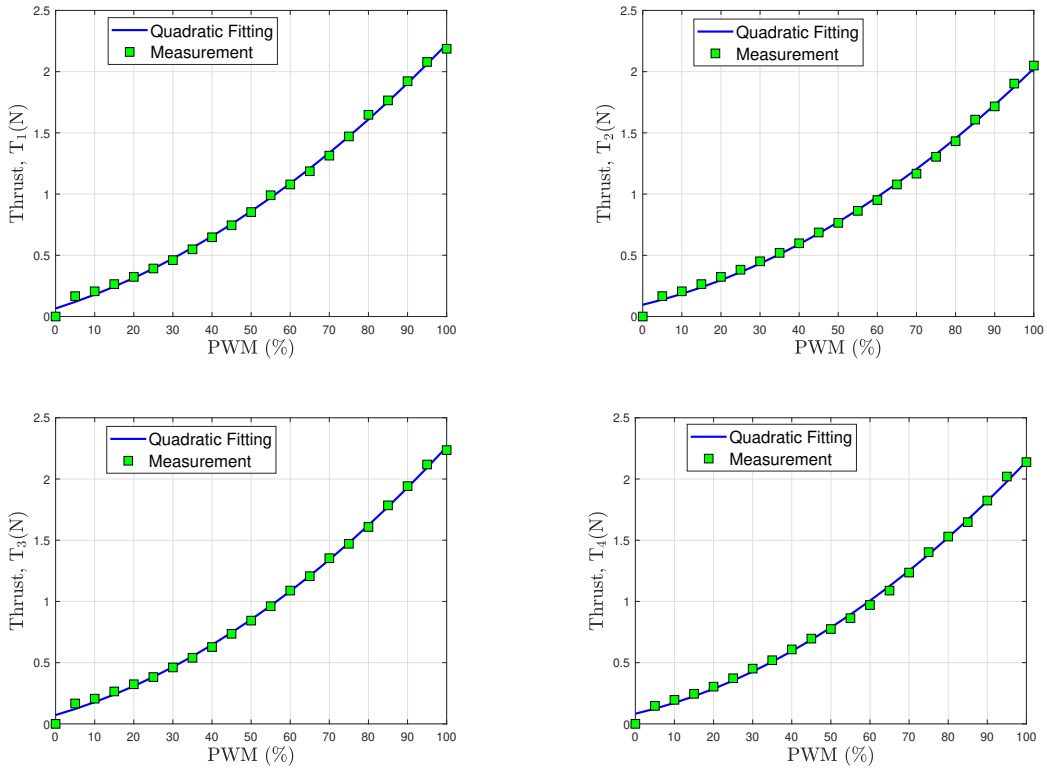

Figure A.1: Thrust computed for each PWM command that each rotor was subjected along with the quadratic fitting. From left to right, top to bottom: (a) Rotor 1; (b) Rotor 2; (c) Rotor 3; (d) Rotor 4.

a rotor. Thus, in order to determine the constants c_i, for an interval of PWM commands ranging from 60% to 90%, the referred force and moment were computed using the previous equations. This interval was considered since during a stable flight the PWM commands are within it. By approximating the set of points obtained by a linear function that crosses the origin, the constants detailed in Table A.1 were attained.

c_1	c_2	c_3	c_4
0.02915	0.03594	0.04062	0.03892

Table A.1: c_i obtained with the linear fitting

Since the control system computes the moments and thrust required to cope with the references given, the necessity to obtain the PWM commands from these quantities arises. Therefore, by inverting the matrix present in (3.8), the thrust that each rotor has to provide is calculated. Subsequently, using the set of experimentally determined equations $T_i = \bar{a}_i PWM_i^2 + \bar{b}_i PWM_i + \bar{c}_i$, is possible to extract the PWM values through the expression:

$$PWM_i = \frac{-\bar{b}_i + \sqrt{\bar{b}_i^2 - 4\bar{a}_i\left(\bar{c}_i - T_i\right)}}{2\bar{a}_i} \tag{A.3}$$

With the equations derived is possible to determine the PWM commands necessary for hovering:

$$\begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ L & -L & -L & L \\ -L & -L & L & L \\ c_1 & -c_2 & c_3 & -c_4 \end{bmatrix}^{-1} \begin{bmatrix} mg \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1.282 \\ 0.974 \\ 1.282 \\ 0.974 \end{bmatrix} (N) \tag{A.4}$$

$$\begin{bmatrix} PWM_1 \\ PWM_2 \\ PWM_3 \\ PWM_4 \end{bmatrix} = \begin{bmatrix} 63.3175 \\ 65.0246 \\ 63.3731 \\ 63.4722 \end{bmatrix} (\%) \tag{A.5}$$