



**TÉCNICO**  
LISBOA



**Modelling and Attitude Stabilization of a Tilt-Quadrotor:**  
Simulation and Experimental Validation

**Ricardo Jorge da Costa Andrade Marques**

Thesis to obtain the Master of Science Degree in

**Mechanical Engineering**

Supervisors: Prof. Alexandra Bento Moutinho  
Prof. Filipe Szolnoky Ramos Pinto Cunha

**Examination Committee**

Chairperson: Prof. Paulo Rui Alves Fernandes  
Supervisor: Prof. Alexandra Bento Moutinho  
Member of the Committee: Prof. Paulo Jorge Coelho Ramalho Oliveira

**June 2018**



## **Acknowledgments**

First I would like to thank Prof. Alexandra Moutinho and Prof. Filipe Cunha for their help and support given and especially for their patience. Second I would like to thank my friends for all the stupid moments which helped me push through this degree.

I would also like to thank my family, especially my mother who always gave me the support, help and quiet I so desperately needed and my brother who always believed in me. The biggest thank you of all goes to my girlfriend Carolina for all the help in the last five years and for all the crazy moments. We were never alone and I could not have kept my sanity without you.



## Resumo

Esta dissertação tem como objectivo a continuação do trabalho previamente desenvolvido no projeto ALIV, ao estudar e desenvolver o Tilt-Quadrotor, um veículo aéreo capaz de descolar e aterrar verticalmente, e também de vôo horizontal de grandes velocidades. Esta plataforma tem uma grande variedade de movimentos, possíveis devido à acção de quatro motores servo que permitem inclinar dois dos quatro rotores. Uma das principais vantagens desta plataforma é a possibilidade de movimento horizontal sem a necessidade de se inclinar.

Primeiro introduz-se os modelos dos actuadores do Tilt-Quadrotor, e os seus movimentos principais são explicados e provados matematicamente. De seguida as equações do movimento são definidas e são desenvolvidos os modelos dos sensores presentes no protótipo.

O modelo não linear do Tilt-Quadrotor é implementado em Simulink com um controlador LQR, obtido a partir de um modelo linearizado, para a atitude e posição do quadrirotor. São desenvolvidos estimadores, com base no Kalman Filter, para os três ângulos de atitude, e para a posição e velocidade vertical. O desempenho dos controladores, assim como dos estimadores são avaliados no simulador desenvolvido, onde se mostra que as técnicas implementadas podem estabilizar o quadrirotor e que este é capaz de efectuar todos os movimentos introduzidos.

O trabalho desenvolvido nesta dissertação culmina na validação experimental dos resultados obtidos no simulador. O protótipo ALIV3 foi testado com sucesso, tendo sido demonstrado que é capaz de vôo estável.

**Palavras-chave:** Quadrirotor-Tilt, Filtro de Kalman, Regulador Quadrático Linear, Controlo de Atitude, Controlo de Posição, Validação Experimental.



## Abstract

This dissertation aims to continue the work previously developed in the ALIV project, by studying and developing the Tilt-Quadrotor, a flying vehicle capable of both vertical take-off and landing and horizontal high speed flight with great efficiency. This platform has a great range of motions, made possible by the tilting of two of its four rotors by the action of servo motors, with the main advantage being the possibility of horizontal motion without the tilting of the platform central core.

First the Tilt-Quadrotor actuator models are introduced and then its principal motions are explained and proven possible mathematically. The equations of motion are defined and the models of the sensors present in the real life prototype are developed.

The Tilt-Quadrotor non-linear model is implemented in Simulink with an LQR controller for both position and attitude, obtained from a linearized model. Estimators for all three attitude angles, altitude and vertical velocity are developed based on the Kalman Filter. The performance of all controllers and estimators are assessed in the simulator, where it is shown that the controller and estimator pair can stabilise the platform which is capable of performing all motions.

The work presented in this dissertation culminates in the experimental validation of the results obtained in simulation. The ALIV3 prototype was tested and successfully achieved stable flight.

**Keywords:** Tilt-Quadrotor, Tilting rotors, Extended Kalman Filter, Linear-Quadratic-Regulator, Attitude Control, Experimental Validation.



# Contents

- Acknowledgments . . . . . iii
- Resumo . . . . . v
- Abstract . . . . . vii
- List of Tables . . . . . xiii
- List of Figures . . . . . xv
- Nomenclature . . . . . xvii
- Glossary . . . . . 1
  
- 1 Introduction . . . . . 1**
- 1.1 The ALIV concept . . . . . 2
  - 1.1.1 ALIV - IST . . . . . 2
- 1.2 Objectives and contributions . . . . . 4
- 1.3 Thesis structure . . . . . 5
  
- 2 Tilt-Quadrotor Modelling . . . . . 7**
- 2.1 General modelling assumptions . . . . . 7
- 2.2 Coordinate systems . . . . . 7
- 2.3 Actuation . . . . . 8
  - 2.3.1 DC motors . . . . . 8
  - 2.3.2 Servo motors . . . . . 10
  - 2.3.3 Propellers . . . . . 10
- 2.4 Tilt-Quadrotor movement analysis . . . . . 11
  - 2.4.1 Comparison between classical and Tilt-Quadrotor . . . . . 11
  - 2.4.2 Control allocation . . . . . 13
  - 2.4.3 Rebalancing equations . . . . . 13
- 2.5 Equations of motion . . . . . 22
  - 2.5.1 Kinematics . . . . . 22
  - 2.5.2 Dynamics . . . . . 24
- 2.6 Sensors . . . . . 25
  - 2.6.1 Accelerometer . . . . . 25
  - 2.6.2 Gyroscope . . . . . 26

|          |  |           |
|----------|--|-----------|
| 2.6.3    | Magnetometer . . . . .                                     | 26        |
| 2.6.4    | Barometer . . . . .  | 27        |
| 2.7      | Tilt-Quadcopter simulator . . . . .                        | 27        |
| 2.7.1    | Motors model block . . . . .                               | 28        |
| 2.7.2    | Servos model block . . . . .                               | 28        |
| 2.7.3    | Non-linear model block . . . . .                           | 28        |
| 2.7.4    | Sensors model block . . . . .                              | 29        |
| 2.8      | Linear model . . . . .                                     | 29        |
| 2.8.1    | Trim point - hovering . . . . .                            | 29        |
| 2.8.2    | Model linearization . . . . .                              | 30        |
| <b>3</b> | <b>ALIV3 Identification</b>                                | <b>33</b> |
| 3.1      | ALIV3 platform . . . . .                                   | 33        |
| 3.2      | Actuators . . . . .  | 33        |
| 3.2.1    | DC motor characteristics . . . . .                         | 33        |
| 3.2.2    | Propeller characteristics . . . . .                        | 35        |
| 3.2.3    | Servo motor characteristics . . . . .                      | 35        |
| 3.3      | Autopilot characteristics . . . . .                        | 36        |
| 3.4      | Sensors . . . . .  | 37        |
| <b>4</b> | <b>Control and estimation design</b>                       | <b>39</b> |
| 4.1      | LQR control . . . . .                                      | 39        |
| 4.2      | Observability and controlability . . . . .                 | 40        |
| 4.3      | Control design . . . . .                                   | 41        |
| 4.3.1    | Attitude Control . . . . .                                 | 41        |
| 4.3.2    | Altitude Control . . . . .                                 | 43        |
| 4.3.3    | Position Control . . . . .                                 | 44        |
| 4.4      | Estimation . . . . .                                       | 47        |
| 4.4.1    | Kalman Filter . . . . .                                    | 47        |
| 4.4.2    | Attitude estimation using extended Kalman filter . . . . . | 48        |
| 4.4.3    | Altitude estimation . . . . .                              | 50        |
| 4.5      | Simulation results . . . . .                               | 52        |
| <b>5</b> | <b>Experimental implementation</b>                         | <b>55</b> |
| 5.1      | Experimental procedure . . . . .                           | 55        |
| 5.2      | Autopilot implementation . . . . .                         | 56        |
| 5.3      | Experimental validation . . . . .                          | 58        |
| <b>6</b> | <b>Conclusions and future work</b>                         | <b>63</b> |
| 6.1      | Conclusions . . . . .                                      | 63        |
| 6.2      | Future work . . . . .                                      | 64        |

**Bibliography** 65

**A Autopilot Files** 67

A.1 Airframe file . . . . . 67

A.2 Main mixer . . . . . 68

A.3 Aux mixer . . . . . 69



# List of Tables

|     |  |    |
|-----|--|----|
| 3.1 | Platform moments of inertia . . . . .            | 33 |
| 3.2 | DC motor specifications . . . . .                | 34 |
| 3.3 | ESC specifications . . . . .                     | 34 |
| 3.4 | Servo motor specifications . . . . .             | 36 |
| 3.5 | Accelerometer specifications . . . . .           | 37 |
| 3.6 | Gyroscope specifications . . . . .               | 37 |
| 3.7 | Accelerometer specifications . . . . .           | 38 |
| 3.8 | Barometer specifications . . . . .               | 38 |
| 4.1 | Kalman filter mathematical formulation . . . . . | 48 |
| 4.2 | Attitude estimation performance . . . . .        | 50 |
| 4.3 | Altitude estimation performance . . . . .        | 52 |



# List of Figures

|      |  |    |
|------|--|----|
| 1.1  | The omnicopter, from [1]   | 1  |
| 1.2  | Different configurations of aircraft with tilting rotors           | 2  |
| 1.3  | ALIV platform and model, from [5]                                  | 3  |
| 1.4  | ALIV project in 2009, from [6]                                     | 3  |
| 1.5  | ALIV project in 2011, from [7]                                     | 4  |
|      |  |    |
| 2.1  | Inertial and body-Fixed frames, from [9]                           | 8  |
| 2.2  | Electric and mechanical schematic of the motor                     | 9  |
| 2.3  | Tilt-Quadrotor configurations                                      | 11 |
| 2.4  | Classical quadrotor operation principle, from [10]                 | 12 |
| 2.5  | Tilt-Quadrotor $x$ movement with classical quadrotor configuration | 12 |
| 2.6  | Hover  | 13 |
| 2.7  | Positive $x$ motion  | 14 |
| 2.8  | Positive $y$ motion  | 16 |
| 2.9  | Positive roll motion   | 17 |
| 2.10 | Positive pitch motion  | 18 |
| 2.11 | Positive yaw motion  | 20 |
| 2.12 | Pitch tilt angle position hold                                     | 21 |
| 2.13 | Tilt-Quadcopter Simulation model                                   | 27 |
| 2.14 | Motors Model block   | 28 |
| 2.15 | Servo motors Model block   | 28 |
|      |  |    |
| 3.1  | PROPDRIVE V2 2830 Brushless DC motor                               | 34 |
| 3.2  | Hobbyking™ 3-Blade Propeller 9x4.5 Orange                          | 35 |
| 3.3  | Futaba S3003 Standard  | 36 |
| 3.4  | Pixhawk 1 board  | 36 |
|      |  |    |
| 4.1  | Attitude control response to step references                       | 42 |
| 4.2  | Altitude control response to a step reference                      | 43 |
| 4.3  | Altitude control response to angular step references               | 44 |
| 4.4  | $x$ position control response to a step reference                  | 44 |
| 4.5  | $y$ position control response to a step reference                  | 45 |

|      |   |    |
|------|---|----|
| 4.6  | $x$ position hold response for a pitch manoeuvre . . . . .                | 46 |
| 4.7  | $y$ position hold response for a roll manoeuvre . . . . .                 | 46 |
| 4.8  | Attitude estimation . . . . .   | 50 |
| 4.9  | Altitude estimation . . . . .   | 51 |
| 4.10 | Tilt-Quadrotor performance to different angular step references . . . . . | 52 |
| 4.11 | Tilt-Quadrotor performance to an unfavourable initial position . . . . .  | 53 |
|      |   |    |
| 5.1  | Pixhawk power module . . . . .  | 55 |
| 5.2  | Simplified wiring schematic . . . . .                                     | 56 |
| 5.3  | Testing workbench . . . . .   | 56 |
| 5.4  | PX4 control diagram adapted from [23] . . . . .                           | 57 |
| 5.5  | Example of mixer block . . . . .  | 57 |
| 5.6  | ALIV3 in stable flight . . . . .  | 58 |
| 5.7  | Attitude response of the ALIV3 . . . . .                                  | 60 |
| 5.8  | PWM actuation signals . . . . .   | 61 |

# Nomenclature

## Acronyms

|      |                                  |
|------|----------------------------------|
| APM  | ArduPilot Mega                   |
| BEC  | Battery Eliminator Circuit       |
| BMT  | Blade Momentum Theory            |
| CA   | Control Allocation               |
| CCW  | Counter-ClockWise                |
| CG   | Center of Gravity                |
| CW   | ClockWise                        |
| DC   | Direct Current                   |
| EKF  | Extended Kalman Filter           |
| ESC  | Electronic Speed Controller      |
| GPS  | Global Positioning System        |
| IMU  | Inertial Measurement Unit        |
| LQR  | Linear Quadratic Regulator       |
| MEMS | Micro Electric Mechanical System |
| NED  | North East Down                  |
| PID  | Proportional-Integral-Derivative |
| PWM  | Pulse Width Modulation           |
| RMSE | Root Mean Squared Error          |
| UAV  | Unmanned Aerial Vehicle          |
| VTOL | Vertical Take-Off and Landing    |

## List of variables

|  |  |
|--|--|
| $\bar{\mathbf{a}}$                               | Accelerometer measurement $\bar{\mathbf{a}} = [\bar{a}_x, \bar{a}_y, \bar{a}_z]$ |
| $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ | Matrices of linear state space system  |
| $\phi, \theta, \psi$                             | Euler angles. Roll, Pitch, Yaw   |
| $a_g$  | Accelerometer bias   |
| $b_g$  | Gyroscope bias   |
| $C_Q$  | Dimensionless constant of power  |
| $C_T$  | Dimensionless constant of thrust   |
| $d$  | Distance of rotor center to quadrotor center of gravity                          |
| $\Phi$   | Representation of the Euler angles $\Phi = [\phi, \theta, \psi]$                 |
| $\mathbf{F}_c$                                   | Normalized command thrust  |
| $\bar{\mathbf{g}}$                               | Gyroscope measurement $\bar{\mathbf{g}} = [\bar{g}_x, \bar{g}_y, \bar{g}_z]$     |
| $\mathbf{g}^I$                                   | Gravity vector in earth inertial frame   |
| $g_o$  | Gravitational constant   |
| $h$  | Altitude of quadrotor  |
| $\mathbf{I}$                                     | Inertia matrix $\mathbf{I} = \text{diag}(I_{xx}, I_{yy}, I_{zz})$                |
| $\mathbf{J}$                                     | Cost function of LQR controller  |
| $K_d$  | Static gain of DC motor  |
| $K_E$  | Motor constant   |
| $\mathbf{K}_k$                                   | Kalman gain matrix for Kalman filter   |
| $K_Q$  | Coefficient of moment  |
| $K_s$  | Static gain of servo motor   |
| $K_T$  | Coefficient of thrust  |
| $L$  | Motor inductance   |
| $\bar{\mathbf{N}}$                               | Magnetometer measurement $\bar{\mathbf{a}} = [\bar{N}_x, \bar{N}_y, \bar{N}_z]$  |
| $\bar{\mathbf{N}}^I$                             | Earth magnetic field   |
| $m$  | Mass of the platform   |
| $\mu_a$  | RMS Noise of accelerometer   |
| $\mu_b$  | RMS Noise of barometer   |

|                                      |  |
|--------------------------------------|--|
| $\mu_g$                              | RMS Noise of gyroscope   |
| $\mu_m$                              | RMS Noise of magnetometer  |
| $O$                                  | Origin of earth inertial frame   |
| $\Omega^B$                           | Angular velocity of the platform, in body-fixed frame $\Omega = [p, q, r]$ |
| $\omega_0$                           | Trim angular velocity of propeller   |
| $\omega_i$                           | Angular velocity of propeller $i$  |
| $O_c$                                | Origin of body-fixed frame   |
| $\bar{p}$                            | barometer measurement  |
| $\mathbf{P}^I$                       | Quadrotor position in earth inertial frame $\mathbf{P}^I = [x, y, z]^T$    |
| $\rho$                               | Air density  |
| $p_0$                                | Athmospheric pressure at sea level   |
| $\mathbf{Q}$                         | Moment produced by propeller   |
| $\mathbf{Q}_{alt}, \mathbf{R}_{alt}$ | Weight matrices for altitude LQR controller                                |
| $\mathbf{Q}_{att}, \mathbf{R}_{att}$ | Weight matrices for attitude LQR controller                                |
| $\mathbf{Q}_{pos}, \mathbf{R}_{pos}$ | Weight matrices for position LQR controller                                |
| $R$                                  | Motor resistance   |
| $r$                                  | Propeller radius   |
| $\mathbf{S}$                         | Transformation matrix from the body-fixed frame to the inertial frame      |
| $\mathbf{S}_\phi$                    | Pitch rotation transformation matrix                                       |
| $\mathbf{S}_\psi$                    | Yaw rotation transformation matrix   |
| $\mathbf{S}_\theta$                  | Roll rotation transformation matrix  |
| $\tau_c$                             | Normalized command moment  |
| $\mathbf{T}$                         | Thrust produced by propeller   |
| $\tau$                               | Time constant  |
| $\phi_2, \phi_4, \theta_2, \theta_4$ | Servo motors tilting angles  |
| $\mathbf{x}(t)$                      | Input vector   |
| $U_N, U_E, U_D$                      | Axes of the earth inertial frame. North, East and Down                     |
| $u_x, u_y, u_z$                      | Axes of the body-fixed frame   |

|                 |  |
|-----------------|--|
| $\mathbf{v}$    | Measurement noise  |
| $\mathbf{V}^B$  | Quadrotor linear velocity in body-fixed frame $\mathbf{V}^B = [u, v, w]^T$ |
| $\mathbf{w}$    | Process noise  |
| $\mathbf{x}(t)$ | State vector   |
| $x_0$           | State vector   |
| $\mathbf{y}(t)$ | Output vector  |

### Subscripts

|              |                      |
|--------------|----------------------|
| $x, y, z$    | Cartesian components |
| $\mathbf{c}$ | Command signal       |
| $i$          | Index                |
| $k$          | Time step $k$        |
| ref          | Reference signal     |
| <i>trim</i>  | Trim vector          |

### Superscripts

|                     |                                 |
|---------------------|---------------------------------|
| $B$                 | Defined in Body-fixed frame     |
| $I$                 | Defined in earth Inertial frame |
| $T$                 | Transpose.                      |
| $\sim$              | Deviation                       |
| $\hat{\phantom{x}}$ | Estimated                       |
| $\bar{\phantom{x}}$ | Measured                        |

# Chapter 1

## Introduction

Unmanned aerial vehicle (UAV) adoption is on the rise. While in its inception UAVs were mostly created and used for military purposes, technological advances in both processing power, sensor development and most importantly battery technology means that many ideas that seemed infeasible or only achievable in large scale with enormous capital costs are nowadays a reality due to their possible implementation on a much smaller and in comparison inexpensive format.

Quadrotors in particular are widely popular in the academia as they provide a reliable and relatively cheap way to research and test new control strategies due to their unstable nature. However a new trend is rising, focusing more on the platform configuration and less on the control theory. Innovative configurations such as the omnicopter (figure 1.1) are developed and studied. The omnicopter [1] is a platform with 8 rotors pointing in many different directions, allowing it move in all 6 degrees of freedom, solving the inherent under actuation of classical multirotors.

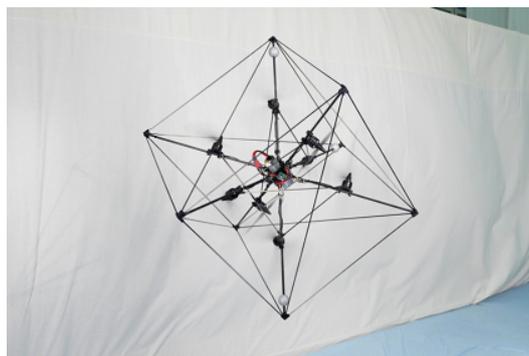
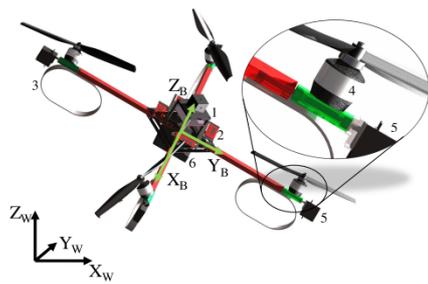
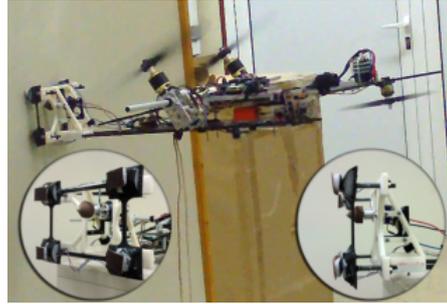


Figure 1.1: The omnicopter, from [1]

Studies have also been made with platforms more similar to standard multicopters. [2] models and controls a quadrotor where all 4 rotors are allowed to tilt about their axis with respect to the main quadrotor body (figure 1.2(a)), while [3] worked with a trirotor with two tilting rotors.



(a) Quadrotor UAV with tilting propellers, from [2]



(b) The Tri-TiltRotor, from [3]

Figure 1.2: Different configurations of aircraft with tilting rotors

This study follows the motivation but not the work of the previously mentioned studies, because as will be shown later, even small changes in the configuration geometry, means that all controlling motions have to be reformulated. It continues the work of the ALIV project on an fully actuated and innovative aircraft.

## 1.1 The ALIV concept

The ALIV (Autonomous Locomotive Individual Vehicle) project is based on the patent US8128033 - "System and Process of Vector Propulsion with Independent Control of three Translation and Three Rotation Axis" [4]. There are several different applications of this patent ranging for Vertical Take-off and Landing (VTOL) aircraft to multicopters.

The platform is very similar to a standard quadrotor with a cross-shaped structure, in fact it can operate exactly like one. However the two main arms are equipped with two servo motors each. These servo motors can tilt two of the four rotors in two different directions each. As such the platform can move in all six degrees of freedom, and unlike a standard multicopter it can move itself sideways without tilting its main core, which can be very useful for moving payloads. Horizontal motion is also more efficient especially if the coverage is optimized aerodynamically for travel in one or both directions. With the addition of airfoil in the main arms, the platform can become a copter-plane hybrid, with great efficiency for long distance travel. The Tilt-Quadrotor can also sustain a tilted orientation whilst staying in the same position.

### 1.1.1 ALIV - IST

The ALIV - IST is one of the projects based on the aforementioned patent and is being developed at Instituto Superior Técnico (IST) in a collaboration with the patent holder, Eng. Severino Raposo from the company ALIVAERO<sup>1</sup> and professors Alexandra Moutinho and Filipe Cunha from IST. Several students have contributed to this project.

<sup>1</sup><http://www.alivaero.com/>

Costa [5] started the project in 2008 where he modelled, simulated and built the first iteration of the current platform (figure 1.3). However this platform was very limited and fragile, and as such Pedro [6] started the design of a second ALIV platform.

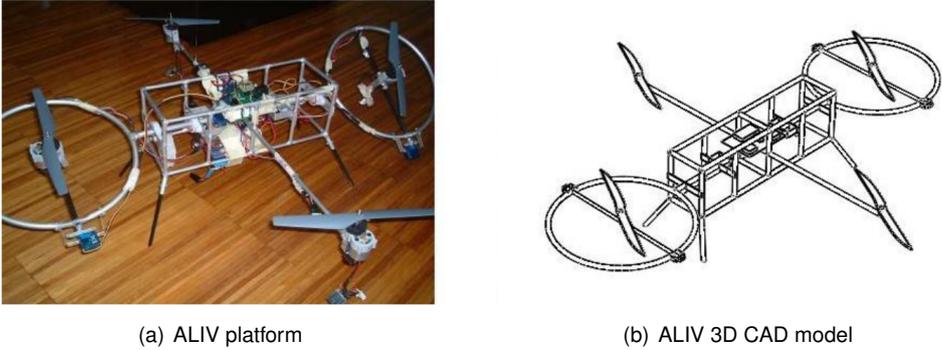


Figure 1.3: ALIV platform and model, from [5]

This time there was big focus on the theoretical design of the platform. A Genetic Algorithm was used to obtain the optimal rotor for the platform, and several Finite Element Method (figure 1.4(a)) studies were conducted on the platform structure elements. The possible application of propeller ducts to improve thrust efficiency was studied (figure 1.4(b)) but ultimately rejected because the added weight did not add enough efficiency to warrant its use.

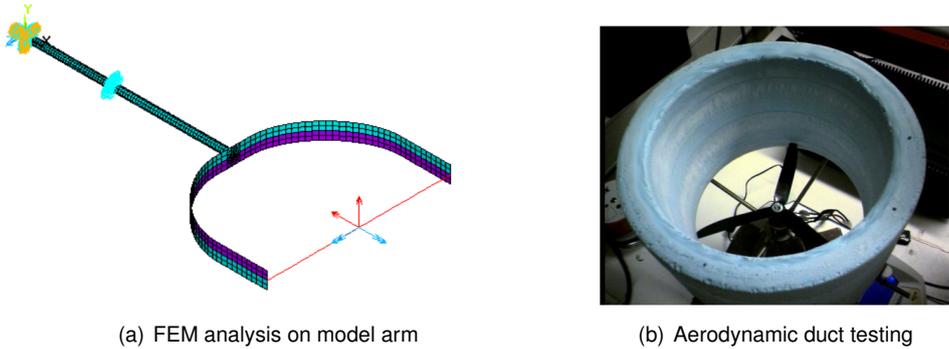
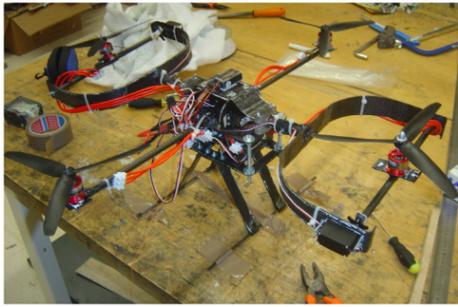
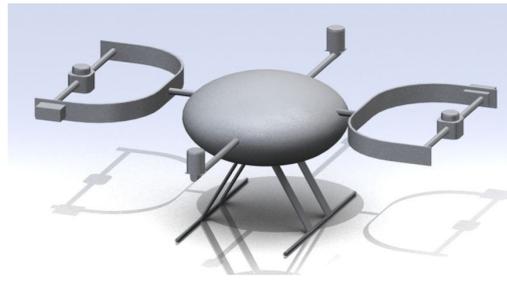


Figure 1.4: ALIV project in 2009, from [6]

Fernandes [7] continued this work. The platform was built from scratch using carbon fibre materials (figure 1.5(a)) and all the actuator components and avionics were chosen and purchased. Aerodynamic drag forces acting on the platform were also studied, and the possibility of an outer shell was considered (figure 1.5(b)).



(a) ALIV3 platform



(b) Aerodynamic outer shell study

Figure 1.5: ALIV project in 2011, from [7]

Mateos [8] was the last student to work on the ALIV3 platform. Some tweaking was made to the structure, assuring it was better balanced, stiffer and more robust. An effort was also made to reduce vibrations. As a proof of concept, Mateos then focused on the modelling, simulation and control of the platform in a classical quadrotor configuration, i.e. without using the servo motors. Several parameters of the ALIV3 platform were identified, including the motor curves and moments of inertia, accomplished by specially built structures.

Given that there was a somewhat large hiatus between this project and the last, the ageing effect had its consequences on several components and as such new motors had to be chosen and purchased, for the previous ones were no longer in working capacity. A new autopilot was also chosen taking advantage of increased processing power, more precise sensors and estimators, and last but not least an easier and more flexible architecture. New propellers were also purchased due to clearance issues with the main arm.

## 1.2 Objectives and contributions

The main purpose of this work is to achieve stabilisation of this new type of aircraft. It continues the work accomplished on the project thus far, and as such the prototype was ready, barring a few needed modifications and other improvements as already explained.

The defined steps to reach this goal are:

- Modelling of the Tilt-Quadrotor platform as well as its actuators and sensors;
- Development of control laws required to achieve stabilisation and horizontal motion;
- Development and tuning of a simulator model in MATLAB/Simulink;
- Adaptation of the developed controllers into an open source Autopilot;
- Implementation in a real life prototype with experimental validation of results;

Taking this into account the main contributions are:

- A detailed study on the principal motions of a new type of aircraft;
- A MATLAB/Simulink simulator including a non-linear model of the Tilt-Quadrotor and its peripherals;
- Experimental implementation and validation of the designed controllers;
- An attitude stabilized ALIV platform;
- Several commits to the PX4 open source flight stack and documentation on Github;

The work developed in the dissertation also resulted in an accepted paper with the title "The Tilt-Quadrotor: Modelling and Attitude Stabilization" to be presented in the 44th European Rotorcraft Forum, in Delft, The Netherlands<sup>2</sup>.

### 1.3 Thesis structure

This dissertation is mainly split in three different sections. In chapter 2 the Tilt-Quadrotor is introduced in more detail. Its operation principles are explained, and the controlling motions are proven possible mathematically. All the components of the Tilt-Quadrotor model are introduced and explained, from the actuator and sensor models, to the kinematic and dynamic equations. In chapter 3 the components, and their characteristics, present in the ALIV3 are introduced.

In chapter 4 the control of the platform in a simulation environment is developed and studied. Starting with a theoretical description of the LQR controller, its results are shown afterwards. The same approach is used for the estimation of the altitude and attitude of the Tilt-quadrotor with the Kalman Filter. In chapter 5 the experimental implementation is explained and the validation results are presented.

Lastly in chapter 6 the most relevant conclusions are presented and recommendations for future work are made.

---

<sup>2</sup><https://www.erf2018.org/>



## Chapter 2

# Tilt-Quadrotor Modelling

### 2.1 General modelling assumptions

Some assumptions were made regarding the quadrotor model in order to simplify it and, as such, to make this study easier. As such it is assumed that:

- The quadrotor is a rigid body;
- The quadrotor is symmetric about the  $x$  and  $y$  axis;
- All external aerodynamic forces acting on the quadrotor such as drag or wind are neglected;
- The propeller flapping effect is neglected;
- The ground effect is neglected;
- All motors have the same time constant;
- There is no slip between the rotor and the propeller;
- All propellers have the same coefficients of power and thrust;

### 2.2 Coordinate systems

In order to avoid confusion it is important to establish the coordinate system and its variables. Two reference frames will be adopted in this work according to which the forces, moments and movement of the quadrotor can be described. The inertial reference frame and the body-fixed frame, which are shown in Figure 2.1.

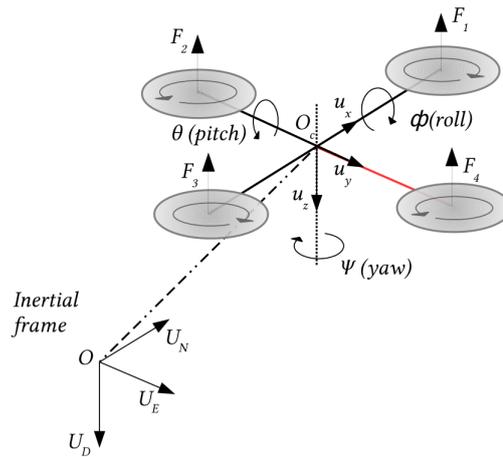


Figure 2.1: Inertial and body-Fixed frames, from [9]

The earth inertial frame adopted is the North-East-Down (NED) frame where the frame is located flat on the surface of the earth with the first axis, as the name describes, pointing North, the second axis pointing East and the last axis pointing downwards towards the central core of the earth. This frame is also centered on the quadrotor initial position  $O$ . The second frame is fixed on the quadrotor center of gravity  $O_c$ , with the  $u_x$  axis pointing forward, the  $u_y$  axis pointing to the right, and the  $u_z$  axis pointing downward. Vectors expressed in the earth inertial frame are marked with the superscript  $I$  and vectors expressed in the body-fixed frame are marked with the superscript  $B$ .

## 2.3 Actuation

The actuators consist of the four different motor-propeller sets and the four servo motors. They are the only way to influence the state of the system in order to stabilise and control it.

A more complete explanation will be given further ahead but it is important to note that all three different types of actuators are inherently connected to each other. The propeller blades are responsible for the thrust forces and momentums, but they are dependent on the action of the direct current (DC) motors which rotate these same propellers at very high speeds. Lastly the servo motors can tilt the motor-propeller sets, and with it, tilt the thrust forces and their respective moments.

### 2.3.1 DC motors

The motors used are brushless Direct Current (DC) electric motors. Compared to brushed DC motors with the same specifications, brushless motors are lighter, more reliable and with a longer life span especially considering high-RPM scenarios such as a quadrotor.

Due to its brushless nature the motor needs to continuously energize and de-energize the different coils in order to move its rotor and finally the propeller. As such these types of motors require an Electronic Speed Controller (ESC) that does this based on readings from a hall sensor.

The desired angular speed is sent to the ESCs using Pulse Width Modulation (PWM) technique. In a nutshell, the data is sent in a square digital signal, where the duration and frequency of the pulses has the encoded information. The ESCs can receive pulses ranging from  $1000\mu s$  to  $2000\mu s$ , and require calibration in order to obtain good results. After calibration a pulse lasting  $1000\mu s$  will make the rotor run at minimum velocity while a pulse lasting  $2000\mu s$  will make the rotor run at maximum velocity.

In figure 2.2 a representation of the electrical circuit of the motor is shown:

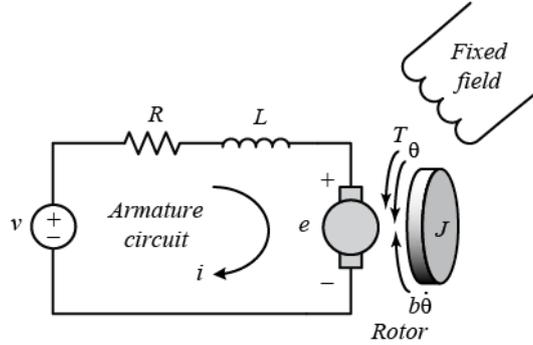


Figure 2.2: Electric and mechanical schematic of the motor

From Kirchhoff's Voltage Law:

$$v = K_E\omega + L\frac{di}{dt} + Ri \quad (2.3.1)$$

that can be used to model the electric circuit, where  $v$  is the voltage applied,  $i$  is the current drawn,  $L$  and  $R$  the motor's inductance and resistance,  $K_E$  the motor constant and  $\omega$  the rotor's speed relative to the stator.

Using Newton's Second Law of Motion the mechanical part of the motor can be modelled by:

$$J\dot{\omega} + b\omega = K_E i \quad (2.3.2)$$

where  $J$  is the rotor's moment of inertia and  $b$  the motor's viscous friction constant .

Since the input signal to the motors will be in PWM, as explained already, it is replaced over  $v$ . Applying the Laplace Transform to equations 2.3.1 and 2.3.2 produces:

$$\frac{\omega_i}{PWM_i} = \frac{K_E}{(Js + b)(Ls + R) + K_E^2} \quad (2.3.3)$$

However since the pole of the electrical part of the system is always significantly faster than its mechanical counterpart it can be neglected and as such the system can be simplified to the following first order system that models the dynamics of the motor:

$$\frac{\omega_i}{PWM_i} = \frac{K_i}{\tau s + 1} \quad (2.3.4)$$

where  $K_i$  is the dc gain of the motor and  $\tau$  its time constant.

### 2.3.2 Servo motors

Similarly to the DC motors, the servo motors also work with Pulse Width Modulation. Although the working range is the same,  $1000\mu s$  to  $2000\mu s$ , the servo motor differs from the DC motor because of its symmetry. As such an input signal of  $1000\mu s$  will rotate and hold the servo at the negative end of its spectrum. Conversely an input signal of  $2000\mu s$  will do the opposite. The trim point of most servo motors is  $0^\circ$  and it is obtained with a signal of  $1500\mu s$  while most servo motors have a range of  $[-0^\circ; 90^\circ]$ .

The servo motor model can be approximated to a first order system:

$$\frac{\theta_i}{PWM_i} = \frac{K_i}{\tau s + 1} \quad (2.3.5)$$

where once again  $K_i$  is the dc gain of the motor and  $\tau$  its time constant.

### 2.3.3 Propellers

Blade Momentum Theory states that the thrust force and moment produced by a propeller  $i$  is related to the radius  $r$  of the propeller, the air density  $\rho$ , the angular speed of the propeller  $\omega$  and dimensionless constants of thrust  $C_T$  and power  $C_P$  respectively.

$$T_i = \rho \pi r^4 C_T \omega_i^2 \quad (2.3.6)$$

$$Q_i = \rho \pi r^5 C_P \omega_i^2 \quad (2.3.7)$$

Since fixed propellers will be used, the only variable is the angular speed and defining:

$$K_T = \rho \pi r^4 C_T \quad (2.3.8)$$

$$K_Q = \rho \pi r^5 C_P \quad (2.3.9)$$

as the coefficients of thrust and moment respectively, (2.3.6) and (2.3.7) simplify to:

$$T_i = K_T \omega_i^2 \quad (2.3.10)$$

$$Q_i = K_Q \omega_i^2 \quad (2.3.11)$$

which is a simple relation between the propeller's angular speed and the force and moment that it produces. The propellers thrust and moment are then related through:

$$T_i = \frac{K_T}{K_Q} Q_i \quad (2.3.12)$$

## 2.4 Tilt-Quadrotor movement analysis

It is important to explain the platform in detail. Figure 2.3 shows the numbering of the rotors and the tilting actions which are performed by the servo motors. The rotors are numbered in a counter-clockwise order, meaning that rotors pairs 1-3 and 2-4 are opposite to each other. Rotors that are green rotate in a clockwise direction and thus produce an opposite torque as per Newton's third law of motion. On the other hand orange rotors rotate in a counter-clockwise direction. The forward direction, coinciding with  $u_x$ , is marked with the red arm.

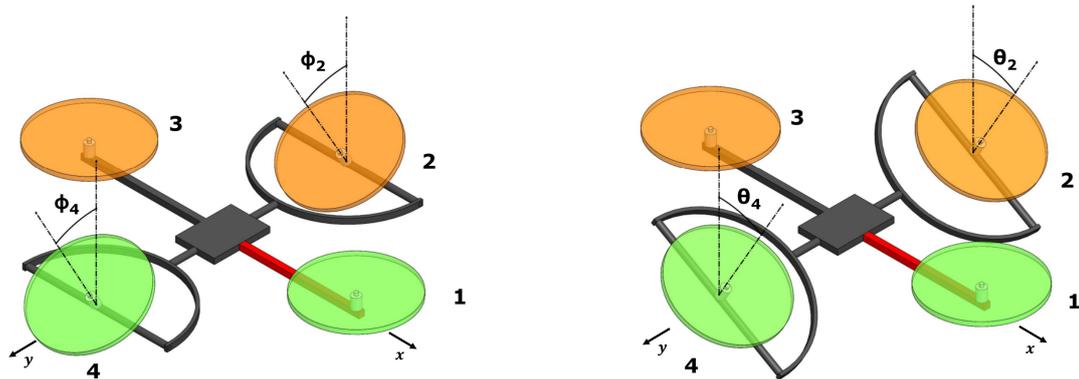


Figure 2.3: Tilt-Quadrotor configurations

The depicted configurations do not relate to any particular manoeuvre, but instead show the different controlling motions possible. The left side of the figure shows the tilting of rotors 2 and 4 around the  $x$  axis, of positive  $\phi_2$  and  $\phi_4$  angles respectively. Similarly the right side of the figure shows the tilting of the same rotors around the  $y$  axis, of a positive angle  $\theta_2$  and  $\theta_4$  respectively. It should be noted that the rotors are shown tilted in similar directions and angles for simplicity, and that they are allowed to move completely independent from each other.

### 2.4.1 Comparison between classical and Tilt-Quadrotor

Even though at first glance the Tilt-Quadrotor appears to be very similar to a classical quadrotor, which is not too far from the truth, there is a small yet crucial detail which drastically changes its behaviour and controlling motions. In a classical quadrotor the main principal motions, such as roll and pitch motions, are achieved by a difference in thrust of the motors on opposite sides of the axis of rotation. For example for a positive roll motion it is necessary to simultaneously increase and decrease the speed of the rotors 2 and 4 respectively, as shown in figure 2.4. Since each rotor produces a moment depending on its direction of rotation, it is required that rotors 2 and 4 rotate in the same direction so that these moments cancel each other out and as such maintain yaw stability during a roll motion. Similarly the same can be concluded for rotors 1 and 3 regarding a Pitch motion. As such in a classical quadrotor the rotors need to rotate in the same direction in opposite pairs, where usually rotors 1 and 3 rotate in a counter-clockwise (CCW) direction and rotors 2 and 4 in a clockwise (CW) direction.

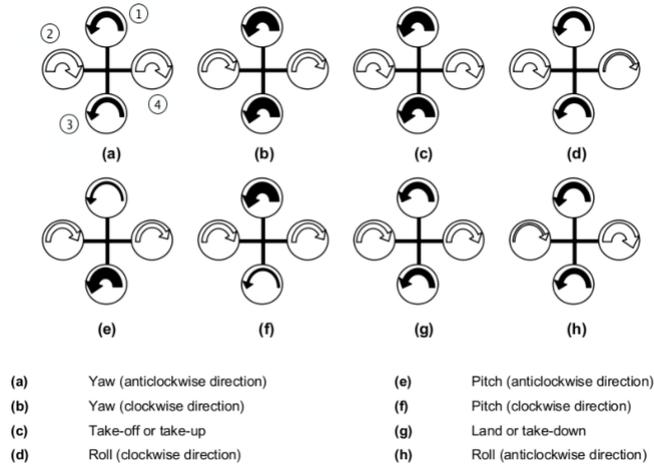


Figure 2.4: Classical quadrotor operation principle, from [10]

However one of the main objectives of the Tilt-Quadrotor configuration is the ability to move horizontally without the need to change its attitude i.e. achieve horizontal motion while still keeping its core center levelled. This is achieved, as will be shown in more detail later in this work, by tilting rotors 2 and 4 in the direction of travel desired. For standard forward motion, rotors 2 and 4 tilt towards the positive direction of the  $x$  axis as can be seen in figure 2.5. However this tilting means the rotor moment is partially induced on the  $x$  axis and if we were to adopt the configuration of a classical quadrotor these rotors would operate in the same direction of rotation and the resulting moments on the  $x$  axis would be added instead of cancelled. This would in turn make the platform perform an unwanted roll motion and possibly make it unstable. As such in order to achieve balance while performing a forward motion it is required that rotors 2 and 4 rotate in opposite directions.

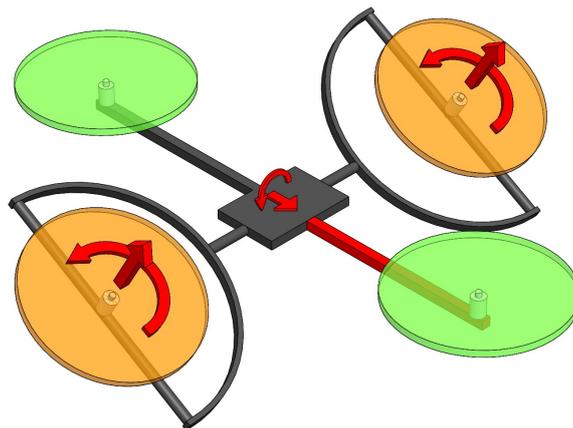


Figure 2.5: Tilt-Quadrotor  $x$  movement with classical quadrotor configuration

Similarly the same occurs for sideways horizontal motions where rotors 2 and 4 will tilt towards the same side of the  $y$  axis and will induce a pitching moment. As such it can be concluded that rotors 1 and 3 also need to rotate in opposite directions. This means that all previously known controlling motions for classical quadrotors cannot be used for this study as they would be unstable. Otherwise simple motions

such as roll and pitch motions now take on increased levels of complexity due to this new configuration, and will have to be reformulated.

## 2.4.2 Control allocation

In this section the different manoeuvres of the Tilt-Quadrotor will be explained in detail and proven possible analytically. First the equations of equilibrium of forces and moments have to be set:

$$\Sigma F_x = -T_2 \sin(\theta_2) \cos(\phi_2) - T_4 \sin(\theta_4) \cos(\phi_4) \quad (2.4.1a)$$

$$\Sigma F_y = T_2 \sin(\phi_2) \cos(\theta_2) + T_4 \sin(\phi_4) \cos(\theta_4) \quad (2.4.1b)$$

$$\Sigma F_z = -[T_1 + T_2 \cos(\theta_2) \cos(\phi_2) + T_3 + T_4 \cos(\theta_4) \cos(\phi_4)] + mg \quad (2.4.1c)$$

$$\Sigma M_x = T_2 \cos(\phi_2) \cos(\theta_2)d - T_4 \cos(\phi_4) \cos(\theta_4)d + Q_2 \sin(\theta_2) \cos(\phi_2) - Q_4 \sin(\theta_4) \cos(\phi_4) \quad (2.4.1d)$$

$$\Sigma M_y = (T_1 - T_3)d - Q_2 \sin(\phi_2) \cos(\theta_2) + Q_4 \sin(\phi_4) \cos(\theta_4) \quad (2.4.1e)$$

$$\Sigma M_z = -Q_1 + Q_2 \cos(\phi_2) \cos(\theta_2) + Q_3 - Q_4 \cos(\phi_4) \cos(\theta_4) - T_2 \cos(\phi_2) \sin(\theta_2)d + T_4 \cos(\phi_4) \sin(\theta_4)d \quad (2.4.1f)$$

where  $T_i$  and  $Q_i$  represent the thrust and moment respectively, generated by propeller  $i$ , as explained in section 2.3.3.

## 2.4.3 Rebalancing equations

### Vertical motion

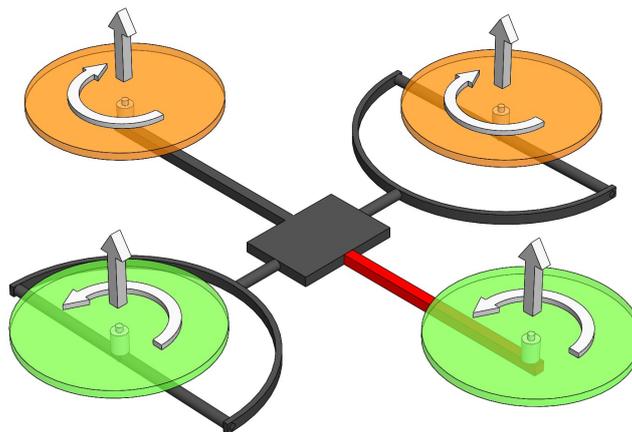


Figure 2.6: Hover

In order to achieve stable vertical motion it must satisfy that  $\Sigma F_z \neq 0$  while all other forces and moments are null. Figure 2.6 illustrates this configuration. Keeping all servos fully vertical ( $\theta_2 = \theta_4 = \phi_2 = \phi_4 = 0^\circ$ )

system (2.4.1) simplifies to:

$$\Sigma F_x = 0 \quad \Leftrightarrow 0 = 0 \quad (2.4.2a)$$

$$\Sigma F_y = 0 \quad \Leftrightarrow 0 = 0 \quad (2.4.2b)$$

$$\Sigma F_z \neq 0 \quad \Leftrightarrow -(T_1 + T_2 + T_3 + T_4) + mg = m\dot{w} \quad (2.4.2c)$$

$$\Sigma M_x = 0 \quad \Leftrightarrow (T_2 - T_4)d = 0 \quad \Rightarrow T_2 = T_4 \quad (2.4.2d)$$

$$\Sigma M_y = 0 \quad \Leftrightarrow (T_1 - T_3)d = 0 \quad \Rightarrow T_1 = T_3 \quad (2.4.2e)$$

$$\Sigma M_z = 0 \quad \Leftrightarrow -Q_1 + Q_2 + Q_3 - Q_4 = 0 \quad \Rightarrow 0 = 0 \quad (2.4.2f)$$

This system has infinite solutions. Assuming  $T_1 = T_2 = T_3 = T_4 = \bar{T}$  the solution for (2.4.2c) is:

$$\bar{T} = m \frac{\dot{w} + g}{4} \quad (2.4.3)$$

A special case is reached when  $\dot{w} = 0$ . This corresponds to a hovering position and it is obtained when:

$$\bar{T}_0 = \frac{mg}{4} \quad (2.4.4)$$

From a levelled position and assuming all rotors are providing the same thrust  $\bar{T}$ , if  $\bar{T} > \bar{T}_0$  the Tilt-Quadrotor will accelerate upwards and rise, if  $\bar{T} < \bar{T}_0$  the Tilt-Quadcopter will accelerate downwards and descend and lastly if  $\bar{T} = \bar{T}_0$  the Tilt-Quadrotor will stay hovering at the same altitude.

## Horizontal motion

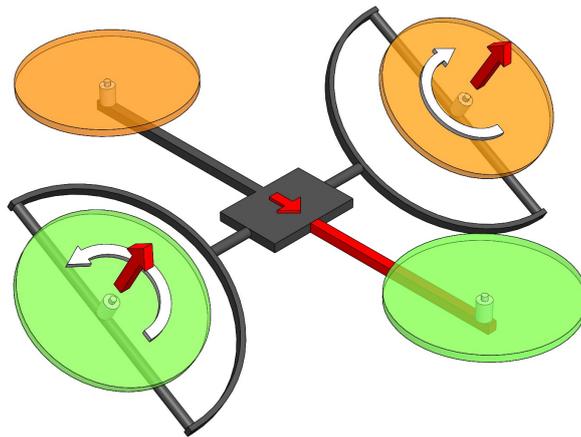


Figure 2.7: Positive  $x$  motion

For forward horizontal motion rotors 2 and 4 will tilt towards the positive  $x$  axis ( $\theta_2, \theta_4 < 0$ ) and no other tilting will be required. Figure 2.7 illustrates this configuration, where the smaller red arrow in the platform central core shows the desired direction of motion. The other two arrows are in red as they are directly responsible for the motion in question. The curve arrows represent the respective moment from each rotor and are displayed in white, because as they cancel each other out, they have no influence in the

motion. For simplicity reasons the arrows corresponding to the thrust and moment of rotors 1 and 3 are not shown. As such assuming  $\theta_2 = \theta_4 = \theta$  and  $\phi_2 = \phi_4 = 0^\circ$  system (2.4.1) simplifies to:

$$\Sigma F_x \neq 0 \quad \Leftrightarrow -(T_2 + T_4) \sin(\theta) = m\dot{u} \quad (2.4.5a)$$

$$\Sigma F_y = 0 \quad \Leftrightarrow 0 = 0 \quad (2.4.5b)$$

$$\Sigma F_z = 0 \quad \Leftrightarrow T_1 + T_2 \cos(\theta) + T_3 + T_4 \cos(\theta) = mg \quad (2.4.5c)$$

$$\Sigma M_x = 0 \quad \Leftrightarrow (T_2 - T_4) \cos(\theta)d + (Q_2 - Q_4) \sin(\theta) = 0 \quad \Rightarrow T_2 = T_4 \quad (2.4.5d)$$

$$\Sigma M_y = 0 \quad \Leftrightarrow (T_1 - T_3)d = 0 \quad \Rightarrow T_1 = T_3 \quad (2.4.5e)$$

$$\Sigma M_z = 0 \quad \Leftrightarrow -Q_1 + Q_3 + (Q_2 - Q_4) \cos(\theta) + (T_4 - T_2) \sin(\theta)d = 0 \quad (2.4.5f)$$

A possible solution to system (2.4.5) is:

$$T_4 = -\frac{m\dot{u}}{2 \sin(\theta)} \quad (2.4.6a)$$

$$T_2 = T_4 \quad (2.4.6b)$$

$$T_1 = T_3 \quad (2.4.6c)$$

$$T_1 = \frac{mg}{2} - T_4 \cos(\theta) \quad (2.4.6d)$$

obtained respectively from (2.4.5a), (2.4.5d), (2.4.5e) and (2.4.5c). From (2.4.5a) or (2.4.6a) it can be seen that increasing the speed of rotors 2 and 4, and in turn their respective thrust forces, or increasing their tilting angle  $\theta$  will further accelerate the Tilt-Quadrotor forward. Considering this a special situation arises. By tilting rotors 2 and 4 by  $-90^\circ$  such that  $\theta_2 = \theta_4 = -90^\circ$  these rotors will be fully horizontal and as such only propel the Tilt-Quadrotor forward (2.4.5a), meaning that altitude is only maintained by rotors 1 and 3, as seen on (2.4.5c). In this configuration the Tilt-Quadrotor turns into a plane/copter hybrid, able to travel at great horizontal speeds. It should be noted that the symmetrical scenario is also possible. By setting  $\theta_2 = \theta_4 = 90^\circ$  the Tilt-Quadrotor will in turn travel in the negative direction of the  $x$  axis.

However it is important to note that the Tilt-Quadrotor must maintain altitude when moving horizontally and even when tilted. Regarding the first scenario, as seen in (2.4.5a) tilting of rotors 2 and 4 will propel the Tilt-Quadrotor forward. However as this tilting angle increases the vertical component of each rotor decreases (2.4.5c). As such the rotor speeds need to increase to maintain altitude. In order to keep balance in other components, rotors 2 and 4 must have the same speed (2.4.5d) while the same applies to rotors 1 and 3((2.4.5e). Consequently there are several solutions. Increasing the speed of the rotors 1 and 3 is the most efficient way to maintain altitude, but increasing the speed of rotors 2 and 4 also further accelerates the quadrotor forward, which can be the desired solution in certain situations. It must be noted however, that the latter solution is not viable for large tilting angles as the vertical component decreases with an increase in the tilting angle. Consequently the most used solution will probably be a mix of the other two in order to both maintain altitude and to move forward at the desired speed.

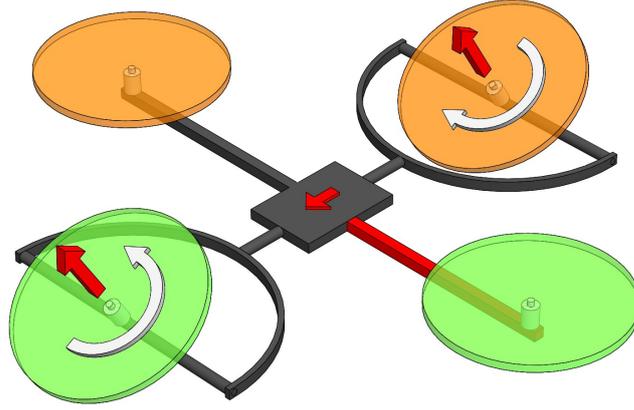


Figure 2.8: Positive  $y$  motion

For sideways horizontal motion the configuration is similar as shown in figure 2.8. Considering a sideways motion towards the positive  $y$  axis, rotors 2 and 4 will tilt in this direction such that  $\phi_2 = \phi_4 = \phi > 0$ , while  $\theta_2 = \theta_4 = 0$ . The system of equations (2.4.1) simplifies to:

$$\Sigma F_x = 0 \quad \Leftrightarrow \quad 0 = 0 \quad (2.4.7a)$$

$$\Sigma F_y \neq 0 \quad \Leftrightarrow \quad (T_2 + T_4) \sin(\phi) = m\dot{v} \quad (2.4.7b)$$

$$\Sigma F_z = 0 \quad \Leftrightarrow \quad T_1 + T_2 \cos(\phi) + T_3 + T_4 \cos(\phi) = mg \quad (2.4.7c)$$

$$\Sigma M_x = 0 \quad \Leftrightarrow \quad (T_2 - T_4) \cos(\phi)d = 0 \quad (2.4.7d)$$

$$\Sigma M_y = 0 \quad \Leftrightarrow \quad (T_1 - T_3)d + (Q_4 - Q_2) \sin(\phi) = 0 \quad (2.4.7e)$$

$$\Sigma M_z = 0 \quad \Leftrightarrow \quad -Q_1 + Q_3 + (Q_2 - Q_4) \cos(\phi) = 0 \quad (2.4.7f)$$

By solving system (2.4.7) it comes from (2.4.7b) to (2.4.7f) that:

$$T_4 = \frac{m\dot{v}}{2 \sin(\phi)} \quad (2.4.8a)$$

$$T_2 = T_4 \quad (2.4.8b)$$

$$T_1 = T_3 \quad (2.4.8c)$$

$$T_1 = \frac{mg}{2} - T_4 \cos(\phi) \quad (2.4.8d)$$

Similarly to the previous system the larger the angle  $\phi$  or the larger the speed of rotors 2 and 4, the faster will the Tilt-Quadrotor accelerate in the  $y$  axis direction. Once again it is possible to fully tilt rotors 2 and 4 so that they are fully horizontal  $\phi_2 = \phi_4 = 90^\circ$ . In this configuration the Tilt-Quadrotor altitude is only maintained by rotors 1 and 3, but it makes it possible to travel at great horizontal speeds. Once more the symmetrical scenario is also possible and the Tilt-Quadrotor can move in the same configuration but on the opposite direction by setting  $\phi_2 = \phi_4 = -90^\circ$ .

The solution for the sideways scenario is analogous where the only difference is that the tilting angle is now  $\phi$  instead of  $\theta$ . Similarly to the previous case, by tilting rotors 2 and 4, their vertical component of thrust will decrease (2.4.7c) and thus, the quadrotor will have to adapt in order to maintain altitude. This can be achieved by increasing rotor speeds but once again it must occur that rotor pairs 2-4 and 1-3 have the same speed, according to (2.4.7d) and (2.4.7e), to keep the quadcopter balanced. As such, the most efficient way to maintain altitude is by increasing the rotor pair 1-3. However increasing the speed of rotor pair 2-4 will also further accelerate the quadrotor sideways. It should be noted however, that, once again, the former solution is not viable for large tilting angles. The most probable solution is increasing the speed of both pairs, in order to maintain altitude and achieve the desired sideways acceleration  $\dot{v}$ .

## Roll motion

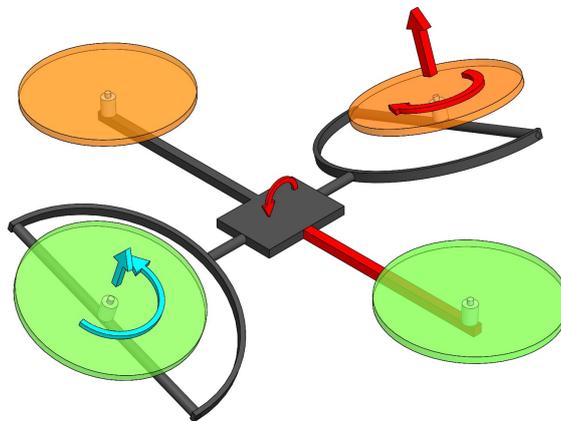


Figure 2.9: Positive roll motion

Very similarly to a classical quadrotor, roll motion is achieved by setting  $\phi_2 = \phi_4 = 0$ , and increasing the speed difference between rotors 2 and 4. Considering a positive rotation around the  $y$  axis it occurs that  $T_2 > T_4$ . However as mentioned in section 2.4.1, since rotors 2 and 4 do not rotate in the same direction this will induce a rotation around the  $z$  axis as can be seen in (2.4.9f). As such rotors 2 and 4 will tilt in different directions such that  $\theta_2 > 0$  and  $\theta_4 < 0$  in order to counteract this rotation. This configuration can be seen in figure 2.9, where the red and blue arrows mean larger and smaller speeds respectively, and consequently, larger and smaller thrusts and moments. Once again the red arrow in the central core

shows the desired motion. Applying these changes to system (2.4.1) and simplifying it results in:

$$\Sigma F_x = 0 \quad \Leftrightarrow -T_2 \sin(\theta_2) - T_4 \sin(\theta_4) = 0 \quad (2.4.9a)$$

$$\Sigma F_y = 0 \quad \Leftrightarrow 0 = 0 \quad (2.4.9b)$$

$$\Sigma F_z = 0 \quad \Leftrightarrow T_1 + T_2 \cos(\theta_2) + T_3 + T_4 \cos(\theta_4) = mg \quad (2.4.9c)$$

$$\Sigma M_x \neq 0 \quad \Leftrightarrow T_2 \cos(\theta_2)d - T_4 \cos(\theta_4)d + Q_2 \sin(\theta_2) - Q_4 \sin(\theta_4) = I_{xx}\dot{p} \quad (2.4.9d)$$

$$\Sigma M_y = 0 \quad \Leftrightarrow (T_1 - T_3)d = 0 \quad (2.4.9e)$$

$$\Sigma M_z = 0 \quad \Leftrightarrow -Q_1 + Q_3 + Q_2 \cos(\theta_2) - Q_4 \cos(\theta_4) - T_2 \sin(\theta_2)d + T_4 \sin(\theta_4)d = 0 \quad (2.4.9f)$$

This system has 7 unknown variables ( $T_1, T_2, T_3, T_4, \theta_2, \theta_4$ , and  $\dot{p}$ ) and 6 linearly independent equations and as such infinite solutions. One possible solution is:

$$T_4 = -\frac{I_{xx}\cdot\dot{p}}{2 \sin(\theta_4)(\frac{d^2}{K} + K)} \quad (2.4.10a)$$

$$\tan(\theta_2) = -\frac{K \sin(\theta_4)}{K \cos(\theta_4) - 2d \sin(\theta_4)} \quad (2.4.10b)$$

$$T_2 = -T_4 \frac{\sin(\theta_4)}{\sin(\theta_2)} \quad (2.4.10c)$$

$$T_1 = T_3 \quad (2.4.10d)$$

$$T_1 = \frac{mg}{2} - T_4[\cos(\theta_4) - \frac{d}{K} \sin(\theta_4)] \quad (2.4.10e)$$

In conclusion, the bigger the difference between the speed of rotors 2 and 4 the faster the Tilt-Quadrotor will roll and as such the larger the tilting angle will have to be in order to prevent rotation around the  $z$  axis. Once again the speed of rotors 1 and 3 will have to increase to maintain altitude, in order to compensate the decrease in the vertical thrust of rotors 2 and 4 that is a consequence of their tilt.

## Pitch motion

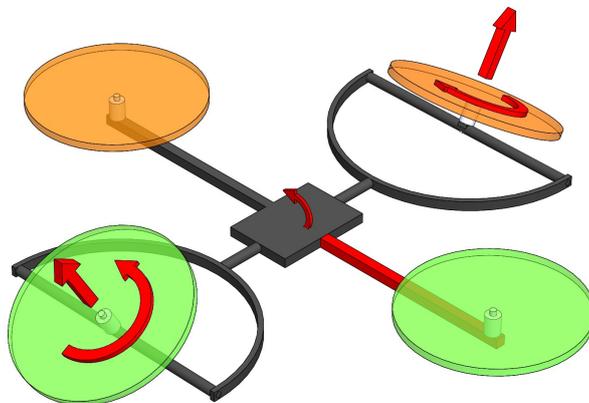


Figure 2.10: Positive pitch motion

For a positive rotation around the  $y$  axis the following configuration is adopted. Rotors 2 and 4 will tilt around the  $x$  axis, symmetrically, such that  $\phi_2 < 0$ ,  $\phi_4 > 0$  and  $\phi_4 = -\phi_2 = \phi^*$ . Lastly it is also set that  $\theta_2 = \theta_4 = 0$ . Implementing these changes the system of equations ((2.4.1)) turns into:

$$\Sigma F_x = 0 \quad \Leftrightarrow 0 = 0 \quad (2.4.11a)$$

$$\Sigma F_y = 0 \quad \Leftrightarrow -T_2 \sin(\phi^*) + T_4 \sin(\phi^*) = 0 \quad (2.4.11b)$$

$$\Sigma F_z = 0 \quad \Leftrightarrow T_1 + T_2 \cos(\phi^*) + T_3 + T_4 \cos(\phi^*) = mg \quad (2.4.11c)$$

$$\Sigma M_x = 0 \quad \Leftrightarrow T_2 \cos(\phi^*)d - T_4 \cos(\phi^*)d = 0 \quad (2.4.11d)$$

$$\Sigma M_y > 0 \quad \Leftrightarrow (T_1 - T_3)d + Q_2 \sin(\phi^*) + Q_4 \sin(\phi^*) = I_{yy}\dot{q} \quad (2.4.11e)$$

$$\Sigma M_z = 0 \quad \Leftrightarrow -Q_1 + Q_3 + Q_2 \cos(\phi^*) - Q_4 \cos(\phi^*) = 0 \quad (2.4.11f)$$

System (2.4.11) 6 unknown variables ( $T_1, T_2, T_3, T_4, \phi^*$  and  $\dot{q}$ ) and 5 linearly independent equations and as such it has infinite solutions. A possible solution is:

$$T_4 = \frac{I_{yy}\dot{q}}{2K \sin(\phi^*)} \quad (2.4.12a)$$

$$T_2 = T_4 \quad (2.4.12b)$$

$$T_1 = T_3 \quad (2.4.12c)$$

$$T_1 = \frac{mg}{2} - T_4 \cos(\phi^*) \quad (2.4.12d)$$

This motion is achieved by tilting rotors 2 and 4 towards opposite directions of the  $y$  axis. This induces two similar moments that add up to induce rotation about the  $y$  axis as can be seen in (2.4.11e). As such increasing the speed of rotors 2 and 4 and/or increasing the tilting angle  $\phi^*$  will cause the Tilt-Quadrotor to rotate faster about its  $y$  axis. For balancing reasons it is required for rotor pairs 2-4 and 1-3 to have the same speed as per (2.4.11f),(2.4.11d) and (2.4.11b).

Once again with the tilting of rotors 2 and 4, the vertical component of the thrust provided by them decreases (2.4.11c), and as such rotor speeds will have to increase in order to compensate this difference and maintain altitude. Therefore the quadrotor can maintain altitude by increasing the speed of the first pair of rotors, by increasing the speed of the second pair of rotors or by increasing the speed of both pairs, as long as each motor in the same pair has the same speed. The first solution, increasing the speed of rotors 1 and 3, is the most efficient way to maintain altitude, whilst the second solution has the added effect of increasing the speed of the pitch rotation which may also be desired.

One interesting note is the fact the the Tilt-Quadrotor can also perform a pitch motion with a configuration similar to a classical quadrotor. By increasing the difference of speeds between rotors 1 and 3 a pitching moment is produced. However since rotors 1 and 3 rotate in opposite directions their speed difference will create an unwanted rotation around the  $z$  axis. As such rotors 2 and 4 will need to tilt, and their velocities will need to be adjusted to prevent this yaw motion in a similar configuration to the one presented in the next section.

## Yaw motion

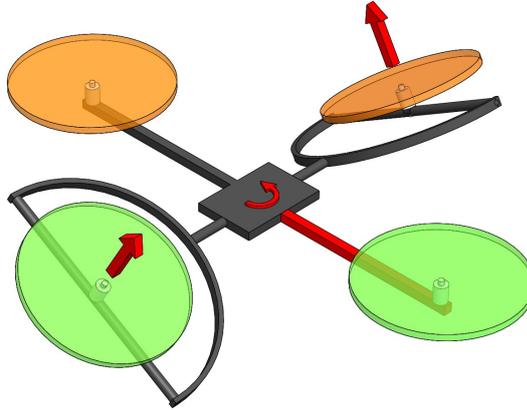


Figure 2.11: Positive yaw motion

For a positive rotation around the  $z$  axis the following configuration is adopted. Rotors 2 and 4 will rotate around the  $y$  axis, symmetrically such that  $\theta_2 < 0$  and  $\theta_4 > 0$ . This generates a rotating moment around the  $z$  axis. No other tilting movements are required ( $\phi_2 = \phi_4 = 0$ ). The system of equations (2.4.1) then turns into:

$$\Sigma F_x = 0 \quad \Leftrightarrow -T_2 \sin(\theta_2) - T_4 \sin(\theta_4) = 0 \quad (2.4.13a)$$

$$\Sigma F_y = 0 \quad \Leftrightarrow 0 = 0 \quad (2.4.13b)$$

$$\Sigma F_z = 0 \quad \Leftrightarrow T_1 + T_2 \cos(\theta_2) + T_3 + T_4 \cos(\theta_4) = mg \quad (2.4.13c)$$

$$\Sigma M_x = 0 \quad \Leftrightarrow T_2 \cos(\theta_2)d - T_4 \cos(\theta_4)d + Q_2 \sin(\theta_2) - Q_4 \sin(\theta_4) = 0 \quad (2.4.13d)$$

$$\Sigma M_y = 0 \quad \Leftrightarrow (T_1 - T_3)d = 0 \quad (2.4.13e)$$

$$\Sigma M_z \neq 0 \quad \Leftrightarrow -Q_1 + Q_2 \cos(\theta_2) + Q_3 - Q_4 \cos(\theta_4) - T_2 \sin(\theta_2)d + T_4 \sin(\theta_4)d = I_{zz}\dot{r} \quad (2.4.13f)$$

System (2.4.13) forms a system with 7 unknown variables ( $T_1, T_2, T_3, T_4, \theta_2, \theta_4$  and  $\dot{r}$ ), and 5 linearly independent equations so there is infinite solutions. From equation (2.4.13e) it can easily be concluded that  $T_1 = T_3$  and solving the system in function of  $\theta_4$  and  $\dot{r}$  the following solution is obtained:

$$T_4 = \frac{I_{zz}\dot{r}}{2 \sin(\theta_4)(\frac{K^2}{d} + d)} \quad (2.4.14a)$$

$$\tan(\theta_2) = -\frac{d \sin(\theta_4)}{d \cos(\theta_4) + 2K \sin(\theta_4)} \quad (2.4.14b)$$

$$T_2 = -T_4 \frac{\sin(\theta_4)}{\sin(\theta_2)} \quad (2.4.14c)$$

$$T_1 = \frac{mg}{2} - T_4 [\cos(\theta_4) + \frac{K}{d} \sin(\theta_4)] \quad (2.4.14d)$$

The solution for  $T_2, T_4, \theta_2$  and  $\theta_4$  is once again quite complex. The reason is since rotors 2 and 4 are

tilted in opposite directions, they will induce a moment around the  $x$  axis. This is clear upon inspection of equation (2.4.13d) where only the first term is positive because  $\theta_2 < 0$ . To maintain balance it must happen that  $T_2 > T_4$  and consequently  $\theta_4 > \theta_2$ . This configuration also ensures that the quadrotor does not move in the  $x$  direction whilst performing a yaw rotation as stated on equation (2.4.13a).

## Pitch position hold

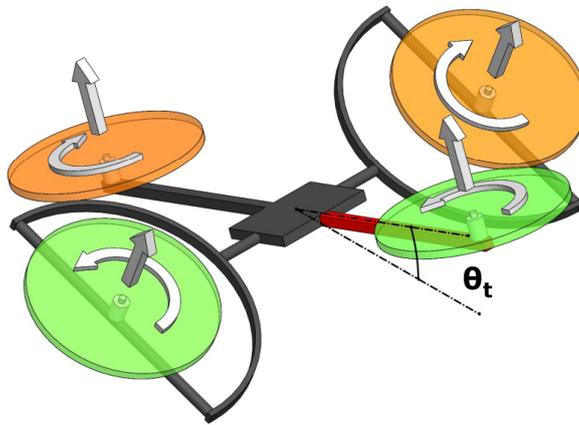


Figure 2.12: Pitch tilt angle position hold

Unlike a classical quadrotor that will necessarily drift from its position if its central core is tilted, the Tilt-Quadrotor has the ability to stay still horizontally even if its tilted in a roll and/or pitch angle. Considering a pitch tilt angle  $\theta$ , and assuming that  $\phi_2 = \phi_4 = 0$  and  $\theta_2 = \theta_4 = \theta_c$  system of equations (2.4.1) simplifies to:

$$\Sigma F_x^I = 0 \quad \Leftrightarrow -(T_1 + T_3) \sin(\theta) - (T_2 + T_4) \sin(\theta + \theta_c) = 0 \quad (2.4.15a)$$

$$\Sigma F_y^I = 0 \quad \Leftrightarrow 0 = 0 \quad (2.4.15b)$$

$$\Sigma F_z^I = 0 \quad \Leftrightarrow -[(T_1 + T_3) \cos(\theta) + (T_2 + T_4) \cos(\theta + \theta_c)] + mg = 0 \quad (2.4.15c)$$

$$\Sigma M_x^B = 0 \quad \Leftrightarrow T_2 \cos(\theta_c)d - T_4 \cos(\theta_c)d + Q_2 \sin(\theta_c) - Q_4 \sin(\theta_c) = 0 \quad (2.4.15d)$$

$$\Sigma M_y^B = 0 \quad \Leftrightarrow (T_1 - T_3)d = 0 \quad (2.4.15e)$$

$$\Sigma M_z^B = 0 \quad \Leftrightarrow -Q_1 + Q_2 \cos(\theta_c) + Q_3 - Q_4 \cos(\theta_c) - T_2 \sin(\theta_c)d + T_4 \sin(\theta_c)d = 0 \quad (2.4.15f)$$

where the first three equations are evaluated with respect to the earth inertial frame, and the last three with respect to the body frame for simplicity. System 2.4.15 has 6 unknown variables ( $T_1, T_2, T_3, T_4, \theta$  and  $\theta_c$ ) and 5 linearly independent equations, and as such infinite solutions. From (2.4.15d) to (2.4.15f)

a simple solution is achieved if  $T_1 = T_2 = T_3 = T_4 = \bar{T}$ . Solving the system in function of  $\theta_t$ :

$$\theta_c = -2\theta \quad (2.4.16a)$$

$$\bar{T} = \frac{mg}{4 \cos(\theta)} \quad (2.4.16b)$$

So in order to maintain position while tilted the quadrotor will tilt rotors 2 and 4 on the opposite direction of the tilting angle, and twice that amount, so that the thrust of rotors 2 and 4 nulify the thrust of rotors 1 and 3 in the  $x$  direction as seen in (2.4.15a). In turn all 4 rotors increase in speed to compensate the decrease in the vertical component of thrust, where this increase in speed is proportional to the tilting angle.

The procedure and solution is analogous for a roll tilting angle.

## 2.5 Equations of motion

In previous section the different actuators were defined and their relations explained. However in order to model the Tilt-Quadrotor behaviour one must first define how it moves and how its inputs can influence its movement.

### 2.5.1 Kinematics

Kinematics describes the motions of bodies without taking into account their mass or the influence of forces acting on them. According to [11] the object of study of kinematics is the geometry of motion. It aims to describe and define the position and attitude of the quadrotor.

The quadrotor position is described by:

$$\mathbf{P}^I = [x, y, z]^T \quad (2.5.1)$$

while its velocity is given by:

$$\mathbf{V}^B = [u, v, w]^T \quad (2.5.2)$$

The attitude of the quadrotor is given by:

$$\Phi^B = [\phi, \theta, \psi]^T \quad (2.5.3)$$

These are know as the Euler angles, roll ( $\phi$ ), pitch ( $\theta$ ) and yaw( $\psi$ ). The angular velocity is defined by:

$$\Omega^B = [p, q, r]^T \quad (2.5.4)$$

where  $p$ ,  $q$  and  $r$  represent the derivative of the roll, pitch and yaw angles.

Also important is defining the transformation matrix that allow the transformation of a vector from the quadrotor body fixed frame to the earth inertial frame:

$$\mathbf{S} = \mathbf{S}_z(\psi) * \mathbf{S}_y(\theta) * \mathbf{S}_x(\phi) \quad (2.5.5)$$

where,

$$\mathbf{S}_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi) & \cos(\phi) \end{bmatrix} \quad (2.5.6)$$

$$\mathbf{S}_y(\theta) = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \quad (2.5.7)$$

$$\mathbf{S}_z(\psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.5.8)$$

Which yields:

$$\mathbf{S}(\phi, \theta, \psi) = \begin{bmatrix} c(\theta)c(\psi) & c(\psi)s(\theta)s(\psi) - c(\phi)s(\psi) & s(\phi)s(\psi) + c(\phi)c(\psi)s(\theta) \\ c(\theta)s(\psi) & c(\phi)c(\psi) + s(\phi)s(\theta)s(\psi) & c(\phi)s(\theta)s(\psi) - c(\psi)s(\phi) \\ -s(\theta) & c(\theta)s(\phi) & c(\phi)c(\theta) \end{bmatrix} \quad (2.5.9)$$

where  $s(\cdot)$  and  $c(\cdot)$  are respectively the  $\sin(\cdot)$  and  $\cos(\cdot)$  of a given angle.

It is worth mentioning that the Euler angles representation is not unique in the entirety of its domain. In fact when either  $\theta = \pm\frac{\pi}{2}rad$  or  $\phi = \pm\frac{\pi}{2}rad$  a singularity occurs, which means that there are 2 possible orientations for the same notation. The quaternion representation solves this problem at the expense of added complexity, and could be adopted, however since the aim of this works is the stabilization of the quadrotor,  $\theta$  and  $\phi$  have low values and thus, this model is far from the known singularities. For these reasons the Euler angle representation is sufficient with the added bonus that is both simpler and more intuitive.

By using  $\mathbf{S}$  it is possible to describe the quadrotor velocity in the inertial frame from the velocity measurements made in the body fixed frame:

$$\dot{\mathbf{P}}^I = \mathbf{S}\mathbf{V}^B \quad (2.5.10)$$

Integrating  $\dot{\mathbf{P}}^I$  the quadrotor position in the inertial frame is obtained. Similarly the Euler rates can be obtained from the angular velocities according to the following relation:

$$\dot{\Phi}^B = \mathbf{T}\Omega^B \quad (2.5.11)$$

where:

$$\mathbf{T}(\phi, \theta, \psi) = \begin{bmatrix} 1 & \sin(\phi) \tan(\theta) & \cos(\phi) \tan(\theta) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi) \sec(\theta) & \cos(\phi) \sec(\theta) \end{bmatrix} \quad (2.5.12)$$

## 2.5.2 Dynamics

Dynamics makes the connection between the quadrotor forces and its movement. According to Newton's second law of motion:

$$\mathbf{F} = m\mathbf{a} \quad (2.5.13)$$

There are two types of forces acting on the quadrotor, external and internal. Regarding external forces, only the force of gravity will be considered as aerodynamic forces such as drag and wind are considered small and thus neglected. The internal forces are then the thrust forces produced by the four motor-propeller sets. It should be noted that since the servos can tilt the motors, this thrust force is not exclusively vertical as in classical quadrotors.

$T_i$  and  $Q_i$  stand for the thrust force and moment respectively, produced by the motor-propeller set number  $i$ , as defined in section 2.3.3. The forces produced by the quadrotor are:

$$\mathbf{F}^B = \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = \begin{bmatrix} -T_2 \sin(\theta_2) \cos(\phi_2) - T_4 \sin(\theta_4) \cos(\phi_4) \\ T_2 \sin(\phi_2) \cos(\theta_2) + T_4 \sin(\phi_4) \cos(\theta_4) \\ -[T_1 + T_2 \cos(\theta_2) \cos(\phi_2) + T_3 + T_4 \cos(\theta_4) \cos(\phi_4)] \end{bmatrix} \quad (2.5.14)$$

Considering the earth inertial frame the external forces are added through:

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \frac{1}{m} \mathbf{S} \mathbf{F} + \begin{bmatrix} 0 \\ 0 \\ g_0 \end{bmatrix} \quad (2.5.15)$$

where  $g_0 = 9.81m/s^2$  is the Earth's gravitational acceleration and, once again,  $\mathbf{S}$  is the transformation matrix from the body-fixed frame to the inertial frame.

The quadrotor is also subject to rotations in all three axis since the motor-propeller sets are placed at a certain distance from the quadrotor center of gravity (CG). The servo motors also affect these moments greatly by tilting the rotors as already explained. As such the moments produced by the quadrotor actuators are defined by:

$$\mathbf{M}^B = \begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} = \begin{bmatrix} T_2 c(\phi_2) c(\theta_2) d - T_4 c(\phi_4) c(\theta_4) d + Q_2 s(\theta_2) c(\phi_2) - Q_4 s(\theta_4) c(\phi_4) \\ (T_1 - T_3) d - Q_2 s(\phi_2) c(\theta_2) + Q_4 s(\phi_4) c(\theta_4) \\ -Q_1 + Q_2 c(\phi_2) c(\theta_2) + Q_3 - Q_4 c(\phi_4) c(\theta_4) - T_2 c(\phi_2) s(\theta_2) d + T_4 c(\phi_4) s(\theta_4) d \end{bmatrix} \quad (2.5.16)$$

where  $s(\cdot)$  and  $c(\cdot)$  are respectively the  $\sin(\cdot)$  and  $\cos(\cdot)$  of a given angle and thus according to [12] and

Euler's second law of motion:

$$\mathbf{I} \dot{\boldsymbol{\Omega}}^B = \mathbf{M}^B - \boldsymbol{\Omega}^B \times \mathbf{I} \boldsymbol{\Omega}^B \quad (2.5.17)$$

where  $\mathbf{I}$  is the inertia matrix defined by:

$$\mathbf{I} = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \quad (2.5.18)$$

Rearranging 2.5.17 :

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \frac{M_x}{I_x} \\ \frac{M_y}{I_y} \\ \frac{M_z}{I_z} \end{bmatrix} - \begin{bmatrix} \frac{(I_z - I_y)qr}{I_x} \\ \frac{(I_x - I_z)pr}{I_y} \\ \frac{(I_y - I_x)pq}{I_z} \end{bmatrix} \quad (2.5.19)$$

With this set of Kinematic and Dynamic equations it is now possible to model the Quadrotor movement in order to study its behaviour.

## 2.6 Sensors

In order to control the quadrotor it is required to have information about its behaviour and state. Some sensors like the Gyroscope allow the direct measurement of important variables, in this case, the angular velocities. However, other sensors, for example the accelerometer, measure other quantities like proper acceleration which can then be used to estimate the variables of interest such as the quadrotor attitude. As such in the next sections the models of the different sensors present in the quadrotor will be defined, which in this case, are the ones included in the inertial measurement unit (IMU) embedded in the Pixhawk autopilot board. These are, a 3-axis accelerometer, a 3-axis gyroscope, a 3-axis magnetometer and a barometer.

### 2.6.1 Accelerometer

The accelerometer is a Microelectromechanical System (MEMS) that measures proper acceleration [13] which is "the acceleration of a particle relative to its instantaneous rest frame" [14]. This means that if the accelerometer is at rest flat in a non-moving surface it will read the acceleration of gravity which is approximately  $g \approx 9.81m/s^2$ . On the other hand if the accelerometer is free falling it will read a value of approximately zero. As such with a tilting of the quadrotor central core and consequently of the IMU in which the accelerometer is present, the acceleration of gravity will be measured in the other axis of the accelerometer. By analysing the direction of the gravity vector  $\mathbf{g}^I$  it is possible to estimate the roll and pitch angles.

However the accelerometer is also sensitive to the accelerations caused by the movement of the quadrotor  $\mathbf{a}^B$ , as well as measurement bias  $\mathbf{b}_a$  and Gaussian noise  $\boldsymbol{\mu}_a$ . As such the readings on the

accelerometer can be modelled by:

$$\bar{\mathbf{a}}^B = \mathbf{S}^T \mathbf{g}^I + \mathbf{a}^B + \mathbf{b}_a + \boldsymbol{\mu}_a \quad (2.6.1)$$

where  $\bar{\mathbf{a}}^B$  is the measurement output in  $m/s^2$  and  $\mathbf{S}^T$  is the transformation matrix from the earth's inertial frame to the quadrotor body-fixed frame. Furthermore since the accelerometer is not located in the center of gravity of the quadrotor, it will also be affected by accelerations, which arise from the angular velocity of the quadrotor, with tangential and centripetal terms. Lastly according to [15] the bias term can be neglected through calibration of the accelerometer. The model can be simplified for a near-hovering situation to be independent of the platform as proposed in [16] and as such (2.6.1) turns into:

$$\bar{\mathbf{a}}^B = \mathbf{S}^T \mathbf{g}^I + \dot{\boldsymbol{\Omega}} \times \mathbf{r} + \boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \mathbf{r}) + \boldsymbol{\mu}_a \quad (2.6.2)$$

where  $\mathbf{r}$  is the distance between the accelerometer and the center of gravity of the quadrotor and  $\boldsymbol{\Omega}$  the quadrotor angular velocity.

## 2.6.2 Gyroscope

The gyroscope is a MEMS that measures angular velocities. The sensor measurement vector is written as:

$$\bar{\boldsymbol{\Omega}}^B = [\bar{g}_x, \bar{g}_y, \bar{g}_z] \quad (2.6.3)$$

According to [17] the gyroscope readings are mostly affected by a stochastic Gaussian noise  $\boldsymbol{\mu}_g$  and a linear slow time varying non-stochastic bias  $\mathbf{b}_g$ . As such the gyroscope readings can be modelled by:

$$\bar{\boldsymbol{\Omega}}^B = \boldsymbol{\Omega}^B + \mathbf{b}_g + \boldsymbol{\mu}_g \quad (2.6.4)$$

where  $\bar{\boldsymbol{\Omega}}^B$  is the measured angular rate and  $\boldsymbol{\Omega}^B$  the real angular rate in body-fixed frame.

## 2.6.3 Magnetometer

The magnetometer is a MEMS that measures the variation of the earth magnetic field. This change of the magnetic reading can be related to the earth inertial frame allowing the estimation of the quadrotor attitude. According to [9] with proper calibration the bias can be neglected and as such the magnetometer measurement is given by:

$$\bar{\mathbf{N}}^B = \mathbf{S}^T \mathbf{N}^I + \boldsymbol{\mu}_m \quad (2.6.5)$$

where  $\mathbf{N}^I$  is the earth magnetic field, and  $\boldsymbol{\mu}_m$  the Gaussian measurement noise.

## 2.6.4 Barometer

The barometer is a MEMS that measures the atmospheric pressure. Since the atmospheric pressure is dependent on the altitude, this simple sensor is very useful to measure the altitude of the quadrotor. One particular advantage of this sensor is that it is independent of the attitude of quadrotor, meaning it will always measure the altitude in earth inertial frame. However the pressure is also very dependant on the ambient temperature and as such it is required to calibrate this unit frequently. This calibration means that the bias term can be neglected and as such the measurement can be modeled by:

$$\bar{p} = p_0 * (1 - \frac{h}{44330})^{5.255} + \mu_b \quad (2.6.6)$$

where  $\bar{p}$  is the measured pressure in  $Pa$ ,  $p_0$  the atmospheric pressure at sea level in  $Pa$ ,  $h$  the altitude of the quadrotor in  $m$  and  $\mu_b$  the Gaussian noise.

## 2.7 Tilt-Quadcopter simulator

Having defined all laws of motion it is now possible to simulate the behaviour of the Tilt-Quadrotor. The simulator takes the configuration shown in figure 2.13, where *Motors Model* receives a PWM array from the *LQR Controller* block, for each of the four motors and turns it into an angular speed array taking into account the motor dynamic response and limitations. Similarly the same is made in the *Servos Model* block for each of the four servos, where in the case the output is an array of tilting angles. All eight input PWM values then enter the *TiltQuad\_model* block which simulates the Quadrotors behaviour taking into account its current state which is fed to the system via a feedback loop. The output vector is then connected to the *Sensors and Estimation* block which adds the non-linearities such as noise to the sensor readings. It also performs the state estimation that will be discussed in section 4.4. Lastly the estimates are subtracted to the reference signals which are then used in the *LQR Controller* block to compute the controlling actions, which will be explained in chapter 4.

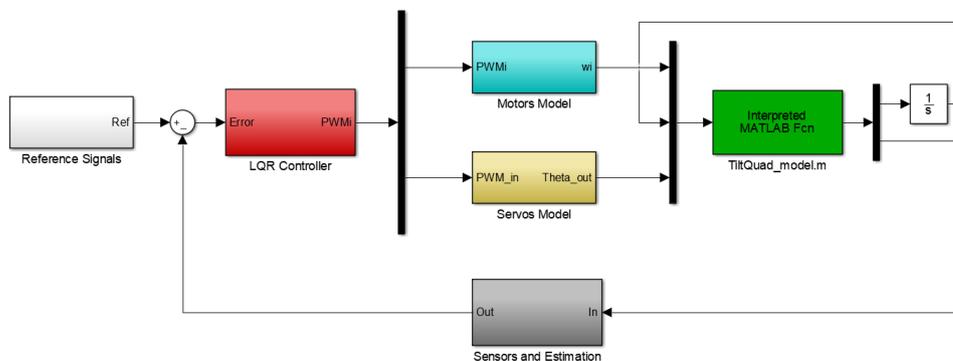


Figure 2.13: Tilt-Quadcopter Simulation model

### 2.7.1 Motors model block

As seen in figure 2.14 each of the four input PWM signals goes through a saturation block, following by a dead zone block. The signal is then discretized before entering the State-Space block with the first order approximation of the motion stated by equation (2.3.4). Afterwards all four signals, consisting of each motor speed in radians are concatenated into an array.

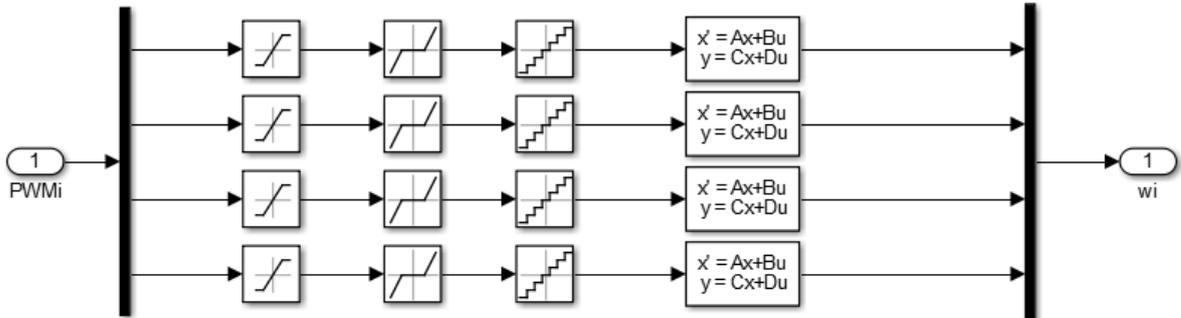


Figure 2.14: Motors Model block

### 2.7.2 Servos model block

Similarly to the DC motors block the input signal is in the form of an array containing four different PWM values. All four values are then saturated to ensure they are between the lower and upper limits of the Servo motor. Once again a dead zone block is present, followed by a discretization block to ensure all values are integer. Afterwards is the first order approximation of the servo motor. Lastly the trim point is subtracted from the signal.

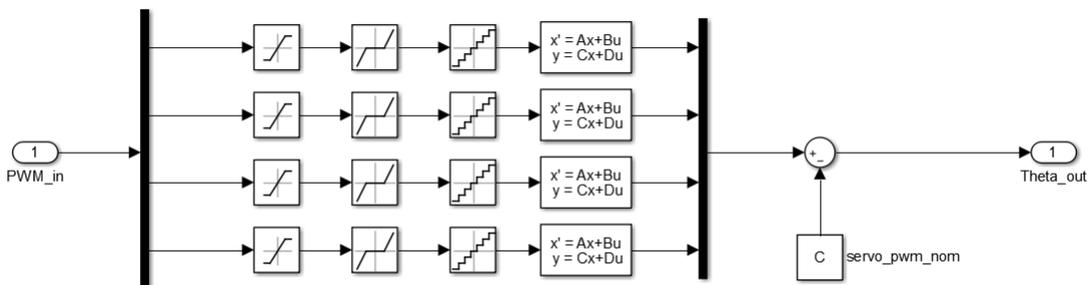


Figure 2.15: Servo motors Model block

### 2.7.3 Non-linear model block

The non-linear model block takes as inputs the rotor speeds and tilting angles from the Motors Model block and Servos Model block respectively. First it computes the individual forces and moments from

the rotor speeds using the equations derived in section 2.3.3. It then computes the total forces and moments from 2.4.1 and using the kinematics and dynamic equations set in section 2.5 it calculates the state outputs. It also calculates the sensor readings according to the models described in section 2.6.

## 2.7.4 Sensors model block

The signals are received from the non-linear model block which contain the measurement readings. However these don't account for noise and other limitations. As such for each sensor there is a saturation block where the upper and lower limits of each sensor are defined. Secondly the noise is added to the signal which is then processed by a quantizer block with the respective sensor resolution. A noise seed is chosen randomly to provide more realistic results.

## 2.8 Linear model

Up until this point our system has only been described by a set of nonlinear equations, however this poses some difficulties as these equations are quite complex and not at all easy to work with. As such it is convenient to model our system using a different format, one that allows and facilitates the use of modern control theory. One of the most commonly used models is the state-space representation:

At its most general state-space representation we have:

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}, t) \quad (2.8.1)$$

$$\dot{\mathbf{y}} = g(\mathbf{x}, \mathbf{u}, t) \quad (2.8.2)$$

where (2.8.1) is called the Dynamic Equation and (2.8.2) the Output Equation,  $x$  the state vector,  $u$  the input vector and  $y$  the output vector of the system. Since this system is time invariant it is possible to drop the  $t$  term and obtain:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \quad (2.8.3)$$

$$\dot{\mathbf{y}} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} \quad (2.8.4)$$

The main goal of this work is to control the Tilt-Quadrotor attitude and position and as such it is defined  $\mathbf{x} = [\Omega, \dot{\mathbf{P}}, \Phi, \mathbf{P}]^T$  and  $\mathbf{u} = [PWM_1, PWM_2, PWM_3, PWM_4, PWM_5, PWM_6, PWM_7, PWM_8]^T$  where the first four input signals correspond to the first four DC motors in their respective order and the last four signals correspond to the servo motors responsible for the tilting angles  $\phi_2, \phi_4, \theta_2$  and  $\theta_4$  respectively.

### 2.8.1 Trim point - hovering

The system can also be linearised about an operating point, the trim point. Any state can be related to this trim point by:

$$\mathbf{x} = \mathbf{x}_0 + \tilde{\mathbf{x}} \quad \text{and} \quad \mathbf{u} = \mathbf{u}_0 + \tilde{\mathbf{u}} \quad (2.8.5)$$

where  $(x_0, u_0)$  is the trim point and  $\tilde{x}$  and  $\tilde{u}$  any state or input deviation from that point.

Since the aim is to stabilize the platform, the trim point is the hovering point where the quadrotor is fully levelled and at an arbitrary height  $z$  and so  $x_0 = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, z]^T$ . In this situations all servos are at their starting position, i.e. all motors are fully vertical, the quadrotor is levelled in all 3 axis, and the sum of all forces and moments acting on the quadrotor amount to zero, meaning:

$$\Sigma F_x = 0 \quad (2.8.6a)$$

$$\Sigma F_y = 0 \quad (2.8.6b)$$

$$\Sigma F_z = \sum_{i=1}^4 T_i = mg_o \quad (2.8.6c)$$

$$\Sigma M_x = (T_2 - T_4)d = 0 \quad (2.8.6d)$$

$$\Sigma M_y = (T_1 - T_3)d = 0 \quad (2.8.6e)$$

$$\Sigma M_z = -Q_1 + Q_2 + Q_3 - Q_4 = 0 \quad (2.8.6f)$$

From the equations above and from (2.3.6) and (2.3.7) it is then trivial to compute the trim speeds for each motor.

## 2.8.2 Model linearization

Considering (2.8.8) and performing a Taylor series it turns into:

$$\dot{x} \approx \dot{x}_o + \left. \frac{\partial f}{\partial x} \right|_{x=x_0, u=u_0} (x - x_0) + \left. \frac{\partial f}{\partial u} \right|_{x=x_0, u=u_0} (u - u_0) \quad (2.8.7)$$

and assuming that the deviations from the trim point are small the higher order terms can be neglected. As such the linear time invariant state space representation of the Tilt-Quadcopter is then given by:

$$\dot{\tilde{x}} = \mathbf{A}\tilde{x} + \mathbf{B}\tilde{u} \quad (2.8.8)$$

$$\dot{\tilde{y}} = \mathbf{C}\tilde{x} + \mathbf{D}\tilde{u} \quad (2.8.9)$$

where

$$\mathbf{A} = \left. \frac{\partial f}{\partial x} \right|_{x=x_0, u=u_0} \quad \text{and} \quad \mathbf{B} = \left. \frac{\partial f}{\partial u} \right|_{x=x_0, u=u_0}$$

Performing this for all system equations the following state space system is obtained:

$$\mathbf{A} = \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ I_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & I_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \end{bmatrix} \quad (2.8.10)$$

$$\mathbf{B} = \begin{bmatrix} \mathbf{B}_v \\ 0_{6 \times 8} \end{bmatrix} \quad (2.8.11)$$

where:

$$\mathbf{B}_v = \begin{bmatrix} 0 & M_{x,2}/I_{xx} & 0 & M_{x,4}/I_{xx} & 0 & 0 & Q_3/I_{xx} & Q_4/I_{xx} \\ M_{y,1}/I_{yy} & 0 & M_{y,3}/I_{yy} & 0 & Q_1/I_{yy} & Q_2/I_{yy} & 0 & 0 \\ M_{z,1}/I_{zz} & M_{z,2}/I_{zz} & M_{z,3}/I_{zz} & M_{z,4}/I_{zz} & 0 & 0 & Q_5/I_{zz} & Q_6/I_{zz} \\ 0 & 0 & 0 & 0 & 0 & 0 & F_{x,2}/m & F_{x,4}/m \\ 0 & 0 & 0 & 0 & F_{y,2}/m & F_{y,4}/m & 0 & 0 \\ F_{z,1}/m & F_{z,2}/m & F_{z,3}/m & F_{z,4}/m & 0 & 0 & 0 & 0 \end{bmatrix} \quad (2.8.12)$$

and:

$$\mathbf{F}_{x,i} = \begin{bmatrix} 0 \\ -2K_t^2 K_s \omega_0^2 \\ 0 \\ -2K_t^2 K_s \omega_0^2 \end{bmatrix} \quad \mathbf{F}_{y,i} = \begin{bmatrix} 0 \\ 2K_t^2 K_s \omega_0^2 \\ 0 \\ 2K_t^2 K_s \omega_0^2 \end{bmatrix} \quad \mathbf{F}_{z,i} = \begin{bmatrix} -2K_t K_d \omega_0 \\ -2K_t K_d \omega_0 \\ -2K_t K_d \omega_0 \\ -2K_t K_d \omega_0 \end{bmatrix} \quad (2.8.13)$$

$$\mathbf{M}_{x,i} = \begin{bmatrix} 0 \\ 2K_t K_d \omega_0 d \\ 0 \\ -2K_t K_d \omega_0 d \end{bmatrix} \quad \mathbf{M}_{y,i} = \begin{bmatrix} 2K_t K_d \omega_0 d \\ 0 \\ -2K_t K_d \omega_0 d \\ 0 \end{bmatrix} \quad \mathbf{M}_{z,i} = \begin{bmatrix} -2K_q K_d \omega_0 \\ 2K_q K_d \omega_0 \\ 2K_q K_d \omega_0 \\ -2K_q K_d \omega_0 \end{bmatrix} \quad (2.8.14)$$

$$\mathbf{Q}_i = \begin{bmatrix} -2K_q^2 K_s \omega_0^2 \\ 2K_q^2 K_s \omega_0^2 \\ 2K_q^2 K_s \omega_0^2 \\ -2K_q^2 K_s \omega_0^2 \\ -2K_t^2 K_s \omega_0^2 d \\ 2K_t^2 K_s \omega_0^2 d \end{bmatrix} \quad (2.8.15)$$

In the equations above  $\omega_0$  is the trim speed of the DC motors and  $K_d$  and  $K_s$  are the static gains for the DC and servo motors respectively.

Finally the trim point vectors are:

$$\mathbf{X}_{\text{trim}} = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, z] \quad (2.8.16)$$

$$\mathbf{U}_{\text{trim}} = [PWM_1^0, PWM_2^0, PWM_3^0, PWM_4^0, PWM_5^0, PWM_6^0, PWM_7^0, PWM_8^0]$$



# Chapter 3

## ALIV3 Identification

Before advancing to the control part of this work there are parameters that need to be identified, regarding the Tilt-Quadrotor geometry as well as its actuators and sensors.

### 3.1 ALIV3 platform

Since the changes to the platform structure were minimal, its geometry remains roughly the same and it is take advantage of the work done by [8] were several tests were carried out to determine these parameters.

It was concluded that the quadrotor CG matches the geometric center in the horizontal plane and at a height of 161mm from the base. The total mass of the Tilt-Quadrotor is 1903g.

The moments of inertia are presented in the table below:

Table 3.1: Platform moments of inertia

| Inertia  | $[kg \cdot m^2]$ |
|----------|------------------|
| $I_{xx}$ | 0.0367           |
| $I_{yy}$ | 0.0262           |
| $I_{zz}$ | 0.0504           |

### 3.2 Actuators

In the following sections the characteristics of all actuator components are shown.

#### 3.2.1 DC motor characteristics

The motors chosen for the platform were ordered from *HobbyKing's* website. As already explained the previous motors were not in good condition due to bad use and ageing, and since they are fulcral components, replacement units were purchased.

The previous motors had also the limitation of operating at speeds very close to the maximum even while at hovering and as such the new motors are more powerful to provide more leeway for manoeuvres.

However one must also take into consideration that the chosen motors are compatible with the ESCs and battery already in the prototype which are in good condition as as such don't warrant the need to be replaced. Furthermore the motors chosen are also compatible with the 3-cell batteries, and don't exceed the maximum current of 30A of the ESCs. In fact as a rule of thumb the current should not be above 75% of the ESCs maximum current as they can reach high temperatures in such situations and possibly ignite. It should be noted however, that this value can be surpassed for short periods of time.

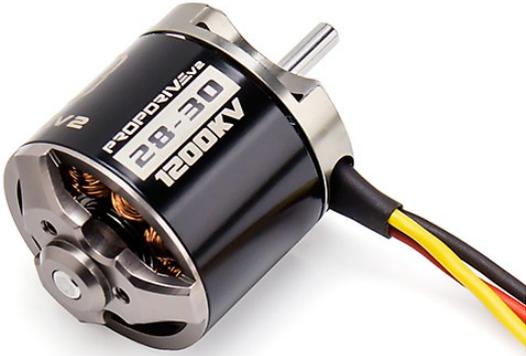


Figure 3.1: PROPDRIVE V2 2830 Brushless DC motor

The motors used are PROPDRIVE v2 2830 1200KV Brushless Outrunner Motor <sup>1</sup> whose characteristics are shown in the table below:

Table 3.2: DC motor specifications

|                 |            |
|-----------------|------------|
| Kv [rpm/V]      | 1200       |
| Power [W]       | 400        |
| Weight [g]      | 69         |
| Max current [A] | 28         |
| Batt. type      | LiPo 3S-4S |

In turn the ESC specifications are shown in the table below:

Table 3.3: ESC specifications

|                  |            |
|------------------|------------|
| BEC              | 5V/2A      |
| Load current [A] | 30         |
| Weight [g]       | 35         |
| Peak current [A] | 45         |
| Batt. type       | LiPo 2S-3S |

<sup>1</sup> [https://hobbyking.com/en\\_us/propdrive-v2-2830-1200kv-brushless-outrunner-motor.html](https://hobbyking.com/en_us/propdrive-v2-2830-1200kv-brushless-outrunner-motor.html)

### 3.2.2 Propeller characteristics

The previous propellers were  $10 \times 4.5$ , 2 blade propellers. The first number is the size of the propeller, in this case the propeller is 10 inches wide in diameter whilst the second number indicates the pitch of the blade. A higher pitched blade will meet airflow at a higher angle and in turn provide more thrust at a cost of higher power usage. The number indicates that the blade would travel 6 inches upwards in one revolution if it were in a fluid with no slippage. However these blades did not provide the required clearance against the frame arms when the motors tilted towards the center, which would cause the blades to collide with the structure. As such a smaller blades were required. Decreasing the blade size has a significant influence in the produced thrust and to mitigate this difference a propeller with 3 blades was chosen.



Figure 3.2: Hobbyking™ 3-Blade Propeller 9x4.5 Orange

As such the propellers on the platform are Hobbyking™ 3-Blade Propeller 9x4.5, as shown in figure 3.2 and were ordered in two colors, green<sup>2</sup> and orange<sup>3</sup> in order to facilitate the visual identification of the orientation of the platform.

### 3.2.3 Servo motor characteristics

The servo motors used are the Futaba S3003 Standard<sup>4</sup>, shown in figure 3.3, since they are the ones already present in the platform. It should be noted that their condition is not ideal, and this is especially noticeable in two of them, since they show signs of wear and ageing. As such it is important to monitor their performance and it is not unlikely that they will need to be replaced in the future in order to improve the platform performance. If this happens it is important to consider the replacement dimensions so as to modify the platform as little as possible.

The servo motors characteristics are shown in table 3.4.

<sup>2</sup>[https://hobbyking.com/en\\_us/hobbykingtm-3-blade-propeller-9x4-5-green-cw-ccw-2pcs.html](https://hobbyking.com/en_us/hobbykingtm-3-blade-propeller-9x4-5-green-cw-ccw-2pcs.html)

<sup>3</sup>[https://hobbyking.com/en\\_us/hobbykingtm-3-blade-propeller-9x4-5-orange-cw-ccw-2pcs.html](https://hobbyking.com/en_us/hobbykingtm-3-blade-propeller-9x4-5-orange-cw-ccw-2pcs.html)

<sup>4</sup><https://www.gpdealera.com/cgi-bin/wgainf100p.pgm?l=FUTM0031>



Figure 3.3: Futaba S3003 Standard

Table 3.4: Servo motor specifications

|                        |            |
|------------------------|------------|
| Torque [Kg-cm]         | 3.17 @4.8V |
| Speed [sec/60°]        | 0.23       |
| Weight [g]             | 37         |
| Range [°]              | +60        |
| Pulse Width [ $\mu$ s] | 1000-2000  |

### 3.3 Autopilot characteristics

As explained before the autopilot hardware was changed from an Ardupilot Mega (APM) 1.4 board to a more recent and capable Pixhawk 1 board, seen on figure 3.4, as the previous hardware was much outdated and already discontinued by the open-source project. Similar to the APM the Pixhawk board is equipped with a 3-axis accelerometer, 3-axis gyroscope, 3-axis magnetometer and a barometer. It also allows the inclusion of an optical flow sensor board. Compared to its predecessor it also has a much higher processing power and memory capacity making it a great solution for this project taking into account both present and future needs.



Figure 3.4: Pixhawk 1 board

Having chosen the board a new decision had to be made, now regarding the flight stack to be used PX4 vs ArduPilot . While both are open-source and fully compatible with the chosen board the decision

fell with using PX4 for the new flight stack as opposed to using the ArduPilot flight stack which had been used on the ALIV3 project thus far. There are several reasons for this being the main reason the fact that PX4 is aimed at developers, meaning there is much more detailed documentation when it comes to creating custom applications, new types of airframes and the architecture itself is purposely designed to allow and facilitate such changes. It also supports a wider range of peripherals and sensors as well as tools either for simulation or for other applications.

### 3.4 Sensors

The pixhawk autopilot board has an embedded Inertial Measurement Unit (IMU) which is equipped with a 3-axis accelerometer, a 3-axis gyroscope, a 3-axis magnetometer and a barometer. Also present in the Tilt-Quadrotor is an external GPS and an external compass. In the next sections the characteristics of these sensors are presented:

#### Accelerometer

The accelerometer used is the one present in the InvenSense MPU-600<sup>5</sup> IMU. It is a 16-bit accelerometer which can be set to a variety of different measurement ranges. Since it will be expected for the quadrotor to have slow movements it has been set for the lowest, and more precise setting.

Table 3.5: MPU-600 16-bit Accelerometer specifications

| MPU-600 16-bit Accelerometer                 | typ     |
|--|---------|
| Measurement Range [ $g$ ]                    | $\pm 2$ |
| Sensitivity [LSB/ $g$ ]                      | 16 384  |
| Noise Spectral Density [ $\mu g/\sqrt{Hz}$ ] | 400     |
| Resolution [ $mg$ ]                          | 0.061   |

#### Gyroscope

The gyroscope is also present in the same IMU as the accelerometer and it is also equipped with a 16-bit Analog-to-Digital Converter (ADC). The characteristics of this sensor are presented in table

Table 3.6: MPU-600 16-bit Gyroscope specifications

| MPU-600 16-bit Gyroscope                            | typ       |
|---|-----------|
| Measurement Range [ $^{\circ}/s$ ]                  | $\pm 250$ |
| Sensitivity [LSB/ $(^{\circ}/s)$ ]                  | 131       |
| Noise Spectral Density [ $(^{\circ}/s)/\sqrt{Hz}$ ] | 0.005     |
| Resolution [ $(^{\circ}/s)$ ]                       | 0.008     |

<sup>5</sup>[https://store.invensense.com/datasheets/invensense/MPU-6050\\_DataSheet\\_V3%204.pdf](https://store.invensense.com/datasheets/invensense/MPU-6050_DataSheet_V3%204.pdf)

## Magnetometer

Table 3.7 presents the characteristics of the ST Micro LSM303M 14-bit Magnetometer<sup>6</sup> included in the IMU of the autopilot board.

Table 3.7: MPU-600 16-bit Accelerometer specifications

| MPU-600 16-bit Accelerometer | typ     |
|------------------------------|---------|
| Measurement Range [gauss]    | $\pm 2$ |
| Sensitivity [LSB/mgauss]     | 13      |
| Noise [mgauss/rms ]          | 5       |
| Resolution [mgauss]          | 0.244   |

## Barometer

The IMU is also equipped with a MEAS MS5611 24-bit Barometer<sup>7</sup>. Table 3.8 presents the characteristics of this sensor.

Table 3.8: MEAS MS5611 24-bit Barometer specifications

| MEAS MS5611 24-bit Barometer | typ       |
|------------------------------|-----------|
| Measurement Range [hPa]      | 10 - 1200 |
| Noise [hPa]                  | 0.05      |
| Resolution [hPa]             | 0.065     |

<sup>6</sup><http://www.st.com/resource/en/datasheet/lsm303d.pdf>

<sup>7</sup>[http://www.amsys.info/sheets/amsys.en.ms5611\\_01ba03.pdf](http://www.amsys.info/sheets/amsys.en.ms5611_01ba03.pdf)

# Chapter 4

## Control and estimation design

In this chapter the Tilt-Quadcopter control will be presented. [8] studied the Tilt-Quadrotor control without using any servo actuation, i.e. just like a standard quadrotor. He showed that the open loop response of the system is disastrous because, as expected, the quadrotor quickly unstabilizes without any form of feedback, proving once again its very unstable nature.

As such a control method is required and based on the conclusions obtained in [18] a Linear-Quadratic Regulator (LQR) controller will be used over the simpler PID controller. The decision is quite straightforward because not only does it mean an easier and less computationally demanding implementation, it also provides a faster and accurate response.

The control is split into three parts, Attitude Control, Altitude Control and Position Control.

### 4.1 LQR control

First a brief explanation of the LQR controller. From optimal control theory the LQR controller aims to control a dynamic system at minimum cost. As such an optimal control action must be obtained:

$$\mathbf{u}(t) = -\mathbf{K}\mathbf{x}(t) \quad (4.1.1)$$

which results in the dynamic closed loop:

$$\dot{\mathbf{x}} = (\mathbf{A} - \mathbf{BK})\mathbf{x} \quad (4.1.2)$$

where the matrix  $\mathbf{K}$  minimizes the cost function:

$$J = \int_0^{\infty} (\mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{u}^T \mathbf{R} \mathbf{u}) dt \quad (4.1.3)$$

and is given by:

$$\mathbf{K} = \mathbf{R}^{-1} \mathbf{B}^T \mathbf{P} \quad (4.1.4)$$

where  $\mathbf{P}$  is obtained by solving the Riccati equation:

$$\mathbf{A}^T \mathbf{P} + \mathbf{P} \mathbf{A} - \mathbf{P} \mathbf{B} \mathbf{R}^{-1} \mathbf{B}^T \mathbf{P} + \mathbf{Q} = 0 \quad (4.1.5)$$

with both  $\mathbf{Q}$  and  $\mathbf{R}$  being the weighing matrices for the state and control errors. By changing matrices  $\mathbf{Q}$  and  $\mathbf{R}$  it is possible to adapt the controller to follow the desired state more rigorously, or on the other hand, to have a more fluid, albeit slower, response but with less actuation.

Using the Bryson's rule, one can define matrices  $\mathbf{Q}$  and  $\mathbf{R}$  as diagonal matrices by setting:

$$Q_{ii} = \frac{1}{\text{minimum acceptable value of } x_i^2} \quad (4.1.6)$$

$$R_{ii} = \frac{1}{\text{minimum acceptable value of } u_i^2} \quad (4.1.7)$$

Of course these values are not full proof and will have to be fine tuned both for simulation and afterwards for the actual implementation but this means that it is possible to set our goals for the maximum errors and using Bryson's Rule have a decent first iteration for the control matrix.

## 4.2 Observability and controllability

A system is said to be fully controllable if it is possible to transform the transfer the system from any initial state  $x_{t_0}$  to any other desired state in a finite interval of time. It can be accessed by analysing the rank of the controllability matrix which is given by:

$$\left[ \mathbf{B} \quad \mathbf{A}\mathbf{B} \quad \dots \quad \mathbf{A}^{n-1}\mathbf{B} \right] \quad (4.2.1)$$

and the observability matrix is given by:

$$\begin{bmatrix} \mathbf{C} \\ \dots \\ \mathbf{C}\mathbf{A} \\ \dots \\ \vdots \\ \dots \\ \mathbf{C}\mathbf{A}^{n-1} \end{bmatrix} \quad (4.2.2)$$

Using MATLAB commands  $Co = ctrb(A, B)$  and  $rank(Co)$  it is concluded that this matrix has rank 12 and since the system also has 12 states it can be concluded that the system is fully controllable. However the observability matrix has rank 8 and as such the system is not fully observable. As mentioned before the quadrotor position in the  $x$  and  $y$  coordinates, as well as their respective velocities  $u$  and  $v$  can not be measured, and will be assumed as known.

## 4.3 Control design

The Tilt-Quadrotor control will be divided into two parts, attitude control and position control. An obvious first step is to define a set of goals for the platform response in both cases which will act as guidelines when tuning the model. A greater focus is given to the roll and pitch angles in contrast to the yaw angle, as these are crucial to achieve stabilisation. Similarly a greater importance is given to altitude in contrast to  $x$  and  $y$  position. As such the following performance metrics are set:

### Attitude Control:

- $\phi$  and  $\theta$  should have rising times smaller than 1 second and settling times smaller than 3 seconds for a step response.

- $\psi$  should have a rising time smaller than 3 seconds and a settling time smaller than 10 seconds for a step response.

- If required a small overshoot is allowed to provide a faster response.

### Position Control:

- Altitude control should have a rising time inferior to 3 seconds and a settling time inferior to 10 seconds for a step response.

- $x$  and  $y$  position control should have a rising time inferior to 5 seconds and rising time inferior to 10 seconds for a step response.

- Small drifts are allowed only in  $x$  and  $y$  position when performing tilting manoeuvres.

- No overshoot is allowed.

### 4.3.1 Attitude Control

As said before the use of Bryson's rule is a good option for a first iteration. However it is required to choose the maximum allowed errors for each of the state parameters. As such the chosen maximum errors are the following:

- $1^\circ$  for both  $\phi$  and  $\theta$  and  $1^\circ/s$  for their respective angular rates  $p$  and  $q$ ;

- $5^\circ$  for  $\psi$  and  $1^\circ/s$  for the respective angular rate  $r$ ;

- $1 \mu s$  for all actuators;

Converting these angular values to radians and using (4.1.6 - 4.1.7) the following matrices are obtained:

$$\mathbf{Q}_{\text{att}} = \text{diag}([3265, 3265, 131, 3265, 3265, 131]); \quad (4.3.1a)$$

$$\mathbf{R}_{\text{att}} = \text{diag}([1, 1, 1, 1, 1, 1, 1, 1]); \quad (4.3.1b)$$

where the first three elements of  $\mathbf{Q}_{\text{att}}$  are in  $[(s/rad)^2]$  and the last three in  $[rad^{-2}]$ , while the elements of  $\mathbf{R}_{\text{att}}$  are all in  $[\mu s^{-2}]$ .

Even though the controller manages to stabilise the quadrotor it does not have the desired performance. As such after some tuning the attitude response is much improved as can be seen on figure 4.1.

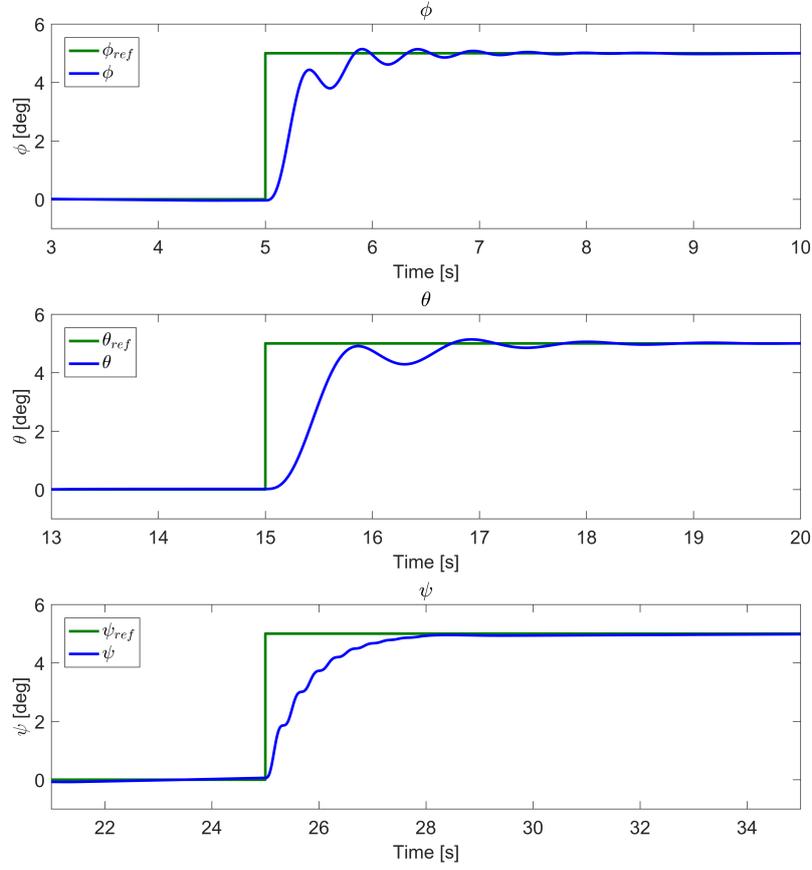


Figure 4.1: Attitude control response to step references

which was obtained with the following weight matrices:

$$\mathbf{Q}_{att} = \text{diag}([4000, 10000, 3500, 12000, 15000, 7000]); \quad (4.3.2a)$$

$$\mathbf{R}_{att} = \text{diag}([0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1]); \quad (4.3.2b)$$

Although there is some oscillations the angular response is fast and presents no static errors and is therefore acceptable. It should be noted that compared to a classical quadrotor the Tilt-Quadrotor has a much faster yaw response, due to the action of the servo motors. Another interesting observation is that the pitch motion is performed not by the action of the servo motors but by differences of thrust between rotors 1 and 3. The reason for this is fairly obvious, and it has to do with the fact that for the same rotation speed the moment produced by the thrust of the propellers with the moment arm is about one order of magnitude larger than the induced moment produced by the rotation of the blades. This means a slight difference in the speed of rotors 1 and 3 produces a large moment around the  $y$  axis, while a large tilting angle would be required to produce an equally strong moment. As such although this configuration mathematically induces a slight rotation around the  $z$  axis, it is very small and can easily

be corrected after the pitch rotation is completed, and in turn the pitch motion is more efficient and faster.

As a proof of concept the LQR controller was changed to perform the pitch motion with the tilting of the arms by the action of the servo motors and the results obtained were still satisfactory.

### 4.3.2 Altitude Control

For altitude control the same procedure was followed. Start the first iteration with the values obtained using Bryson's Rule and tune the parameters to improve the response as desired. The system response shown in figure 4.2 is obtained using the following weight matrices:

$$\mathbf{Q}_{\text{alt}} = \text{diag}([2500, 10000]); \quad (4.3.3a)$$

$$\mathbf{R}_{\text{alt}} = \text{diag}([0.01, 0.01, 0.01, 0.01, 0.01, 0.01, 0.01, 0.01]); \quad (4.3.3b)$$

where the elements of  $\mathbf{Q}_{\text{alt}}$  are in  $[(s/m)^2]$  and  $[m^{-2}]$  respectively, and the elements of  $\mathbf{R}_{\text{alt}}$  are in  $[\mu s^{-2}]$ .

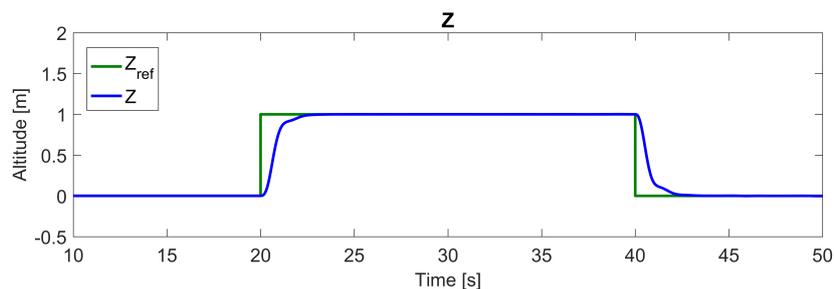


Figure 4.2: Altitude control response to a step reference

It is also important to show that the quadrotor can maintain altitude even when performing other manoeuvres. Figure 4.3 shows that the quadrotor altitude is barely affected by the tilting of its central core.

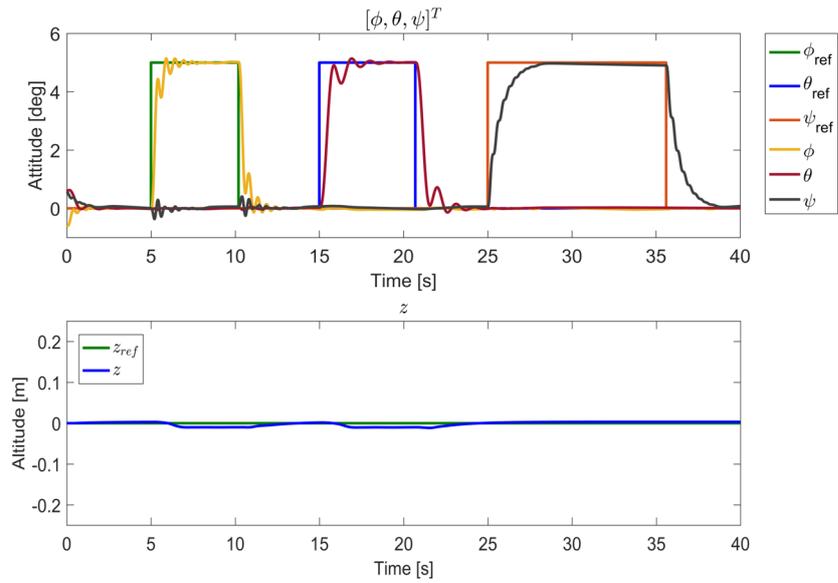


Figure 4.3: Altitude control response to angular step references

### 4.3.3 Position Control

Having shown that the quadrotor can perform all angular motions, stabilize itself and maintain altitude while tilted it is now time to show how the quadrotor behaves when moving horizontally. As mentioned in previous chapters one of the main advantages and particularities of this platform is the ability to move horizontally without the need to tilt its central core.

This can be seen in figures 4.4 and 4.5 which show a response to a unit step reference in the  $x$  and  $y$  position respectively. It can be seen in the bottom part of each figure that the attitude of the quadrotor is not affected by the control actions required for translational motion.

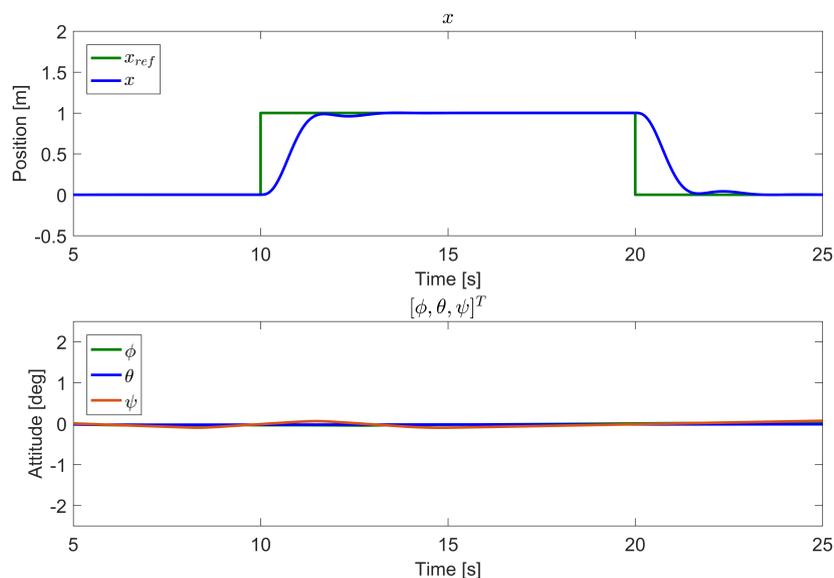


Figure 4.4:  $x$  position control response to a step reference

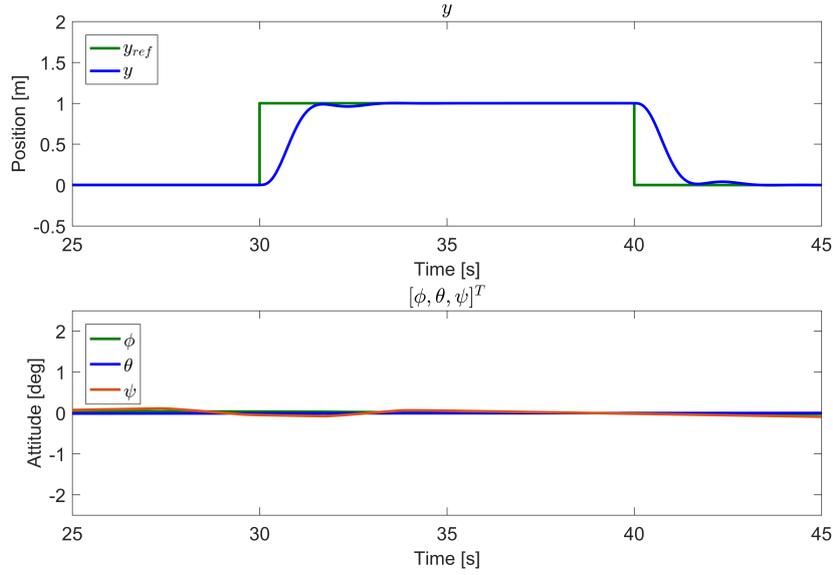


Figure 4.5:  $y$  position control response to a step reference

The tuned values for the weight matrices  $\mathbf{Q}_{\text{pos}}$  and  $\mathbf{r}_{\text{pos}}$  are:

$$\mathbf{Q}_{\text{pos}} = \text{diag}([2500, 2500, 3500, 3500]); \quad (4.3.4a)$$

$$\mathbf{R}_{\text{pos}} = \text{diag}([0.01, 0.01, 0.01, 0.01, 0.01, 0.01, 0.01, 0.01]); \quad (4.3.4b)$$

where the first two elements of  $\mathbf{Q}_{\text{pos}}$  are in  $[(s/m)^2]$  and the last two in  $[m^{-2}]$ , and the elements of  $\mathbf{R}_{\text{pos}}$  are in  $[\mu s^{-2}]$ .

Lastly another possible manoeuvre of the Tilt-Quadrotor is shown. As explained in section 2.4.1 a classical quadrotor, due to having its 4 motors in the same direction will move horizontally if its central core is not levelled. This phenomenon is often referred to as drift. The classical quadrotor makes use of this to perform its lateral motions. However due to the increased number of actuators the Tilt-Quadrotor has the ability to maintain its initial position even while tilted.

When the quadrotor performs a roll motion it will influence its position in the  $y$  axis, while a pitch motion will influence the position in the  $x$  axis. Figures 4.4 and 4.5 show the position of the quadrotor when it is performing the respective angular motion. Since this is the same position controller all matrices used are the ones showed before. The only difference is the reference position given to the controller which in this case is always zero.

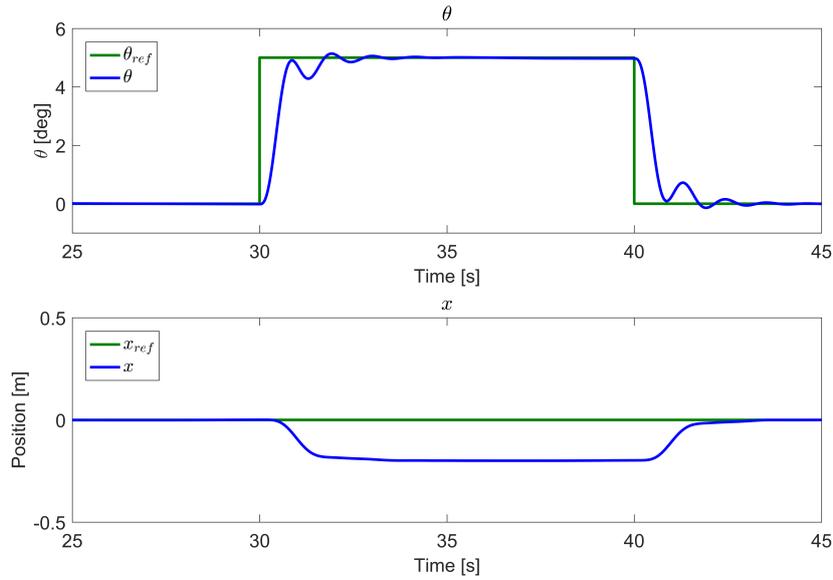


Figure 4.6:  $x$  position hold response for a pitch manoeuvre

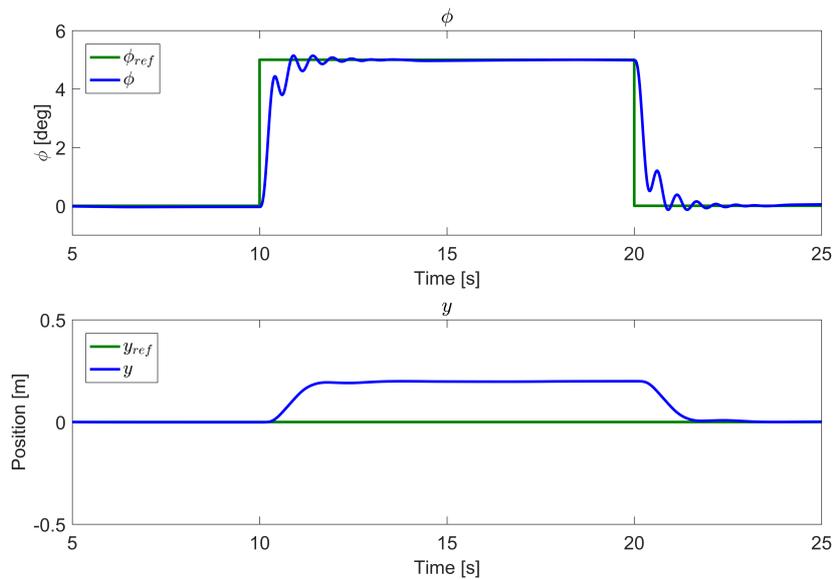


Figure 4.7:  $y$  position hold response for a roll manoeuvre

It can be seen that in both cases there is a slight drift, albeit small ( $\approx 20cm$ ). However once the quadrotor reaches the new reference attitude it very quickly stops the drift, and in fact the quadrotor manages to hold its position while tilted even for long periods of time. This is easily explained by the fact that the attitude controller has much higher gains as it requires a more rigorous control, and once the errors in the attitude return to smaller values the position controller can have a greater influence and completely stop the drift.

It should also be noted that without the position controller this drift would grow much faster even in the first seconds, meaning that while it can't completely stop the drift in a first instance this controller still plays an important role. Another important observation is the fact that with different gains it was possible to obtain much smaller drifts but this meant that the attitude responses would be affected negatively. In

any quadrotor project the attitude response should be the primary focus of the controller design, since it crucial to a stable system, and as such a small drift was accepted to allow better attitude responses.

## 4.4 Estimation

An effective control loop is only possible with good and reliable information. All sensor signals are corrupted with noise which depending on the sensor can be quite significant. Furthermore some states such as the attitude angles cannot be directly measured and as such state estimation is a vital part of this study.

In the next sections the different approaches to each state estimation will be explained. First the well known Kalman Filter is introduced, as well as a common adaptation, the Extended Kalman Filter (EKF), which will be used for the attitude estimation. The angular velocities will be determined directly from a low-pass filtered signal of the gyroscope measurements. A simple Kalman Filter is developed for altitude estimation with the barometer. The X and Y position estimation is not the focus of this study and as such will be assumed as known and no estimation for these states will be discussed.

### 4.4.1 Kalman Filter

The Kalman Filter is a recursive algorithm that makes use of all the sensor measurements over a period of time, which contain noise, to estimate the desired state  $\mathbf{x} \in \mathbb{R}^N$  of the system through the minimization of the error covariance [19]. According to [20] it "supports estimations of past, present, and even future states, and it can do so even when the precise nature of the modeled system is unknown". It is widely used for guidance, navigation, and control of vehicles, especially for aircraft and spacecraft, and as such it is an obvious choice for this work.

Considering the discrete-time linear system:

$$\mathbf{x}_k = \mathbf{A}\mathbf{x}_{k-1} + \mathbf{B}\mathbf{u}_{k-1} + \mathbf{w}_{k-1} \quad (4.4.1)$$

with a measurement:

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{v}_k \quad (4.4.2)$$

where  $\mathbf{w}_{k-1}$  and  $\mathbf{v}_k$  are stochastic variables that represent the process noise and measurement noise respectively. These are assumed to be independent, white, and with normal probability distributions given by:

$$p(\mathbf{w}) \sim N(0, \mathbf{Q}) \quad (4.4.3)$$

$$p(\mathbf{v}) \sim N(0, \mathbf{R}) \quad (4.4.4)$$

where  $\mathbf{Q}$  and  $\mathbf{R}$  are the process noise covariance and measurement noise covariance matrices respectively, and are assumed to be constant.

The Kalman filter is divided in two steps. It begins by making a prediction of the current state given

past state estimations and control actions. These *a priori* estimates are then updated using the new measurements from the sensors using a weighted sum through the Kalman gain matrix  $K_k$ .

Table 4.1: Kalman filter mathematical formulation

Time update:

$$\hat{\mathbf{x}}_k^- = \mathbf{A}\hat{\mathbf{x}}_{k-1} + \mathbf{B}\mathbf{u}_{k-1} \quad (4.4.5)$$

$$\mathbf{P}_k^- = \mathbf{A}\mathbf{P}_{k-1}\mathbf{A}^T + \mathbf{Q} \quad (4.4.6)$$

Measurement update:

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{C}^T (\mathbf{C}\mathbf{P}_k^- \mathbf{C}^T + \mathbf{R})^{-1} \quad (4.4.7)$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}_k^-) \quad (4.4.8)$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{C}) \mathbf{P}_k^- \quad (4.4.9)$$

#### 4.4.2 Attitude estimation using extended Kalman filter

A common improvement for the estimation of non-linear systems is the EKF where the model is linearized about the current time step. According to [21] a good approach is using the gyroscope measurements as input signals to the EKF, which will function as a predictor of the attitude angles, as opposed to using PWM signals. The information from the accelerometer and compass is then corrected from the gyroscope measurements.

This formulation relies on readings from the accelerometer which can be corrupted by horizontal linear accelerations from the quadrotor. In a classical quadrotor this is an acceptable approximation, as considering the quadrotor body frame all thrust forces act on the  $u_z$  axis whilst the only force acting on the quadrotor axis  $u_x$  and  $u_y$  is the gravitational force. As such for a classical quadrotor the developed estimator would be satisfactory, however, the Tilt-Quadrotor is capable of producing forces on the horizontal axis of its body frame, which will be included in the accelerometer readings and thus in the attitude estimation. As such another improvement is made, the magnetometer readings are used to determine the attitude. According to [22] it is possible to determine the roll and pitch angles from the readings of the magnetometer, which are independent of the quadrotor velocity and accelerations. The angles can be computed by:

$$\bar{\phi} = \arctan \left[ \frac{\bar{N}_y}{\bar{N}_z} \right] \quad (4.4.10)$$

$$\bar{\theta} = \arcsin \left[ -\bar{N}_x \right] \quad (4.4.11)$$

For the yaw angle estimation the external compass readings are used. The computed angles are then used in the EKF, together with the accelerometer and gyroscope measurements. Therefore defining  $\mathbf{y} = [\bar{a}_x, \bar{a}_y, \bar{g}_x, \bar{g}_y, \bar{g}_z, \bar{\phi}, \bar{\theta}, \bar{N}_z]^T$ ,  $\mathbf{x} = [\phi, \theta, \psi]^T$  and  $\mathbf{u} = [\bar{g}_x, \bar{g}_y, \bar{g}_z]^T$  the EKF model formulation is:

$$\begin{bmatrix} \phi_k \\ \theta_k \\ \psi_k \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \phi_{k-1} \\ \theta_{k-1} \\ \psi_{k-1} \end{bmatrix} + \begin{bmatrix} \tau & 0 & 0 \\ 0 & \tau & 0 \\ 0 & 0 & \tau \end{bmatrix} \begin{bmatrix} \bar{g}_{x,k-1} \\ \bar{g}_{y,k-1} \\ \bar{g}_{z,k-1} \end{bmatrix} + w_{k-1} \quad (4.4.12)$$

$$\begin{bmatrix} \bar{a}_{x,k} \\ \bar{a}_{y,k} \\ \bar{g}_{x,k} \\ \bar{g}_{y,k} \\ \bar{g}_{z,k} \\ \bar{\phi}_k \\ \bar{\theta}_k \\ \bar{N}_{z,k} \end{bmatrix} = \mathbf{C}_k \begin{bmatrix} \phi_k \\ \theta_k \\ \psi_k \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \bar{g}_{x,k} \\ \bar{g}_{y,k} \\ \bar{g}_{z,k} \end{bmatrix} + v_k \quad (4.4.13)$$

where  $\mathbf{C}_k$  will have to be defined at each iteration. Algorithm 1 was the one used:

---

#### Algorithm 1 EKF for Attitude Estimation

---

$$\mathbf{y}_k = [\bar{a}_x, \bar{a}_y, \bar{g}_x, \bar{g}_y, \bar{g}_z, \bar{N}_z, \bar{N}_y, \bar{N}_z]^T$$

$$\mathbf{u}_k = \bar{\mathbf{g}}^B$$

$$\text{define the state trim condition for iteration } k: \mathbf{x}_k^0 = [0, 0, \hat{\psi}_{k-1}]^T$$

$$\text{define the output trim condition for iteration } k: \mathbf{y}_k^0 = [0, 0, 0, 0, 0, -\sin(\psi_{k-1})]^T$$

**Time Update:**

$$\mathbf{x}_k^- = \mathbf{x}_k^0 + \mathbf{A}_d(\hat{\mathbf{x}}_{k-1} - \mathbf{x}_k^0) + \mathbf{B}_d \mathbf{u}_k$$

$$\mathbf{P}_k^- = \mathbf{A}_d \mathbf{P}_{k-1} \mathbf{A}_d^T + \mathbf{Q}_k$$

Linearization:

$$\mathbf{C}_k = \begin{bmatrix} 0 & -g & 0 \\ g & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -\cos(\mathbf{x}_k^-(3)) \end{bmatrix}$$

**Measurement Update:**

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{C}_k^T (\mathbf{C}_k \mathbf{P}_k^- \mathbf{C}_k^T + \mathbf{R}_k)^{-1}$$

$$\mathbf{x}_k = \mathbf{x}_k^- + \mathbf{K}_k (\mathbf{y}_k - (\mathbf{C}_k (\mathbf{x}_k^- - \mathbf{x}_k^0) + \mathbf{D}_d \mathbf{u}_k) - \mathbf{y}_k^0)$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{C}_k) \mathbf{P}_k^-$$


---

The EKF manages to estimate all three attitude angles in its entire range with great accuracy as shown in figure 4.8.

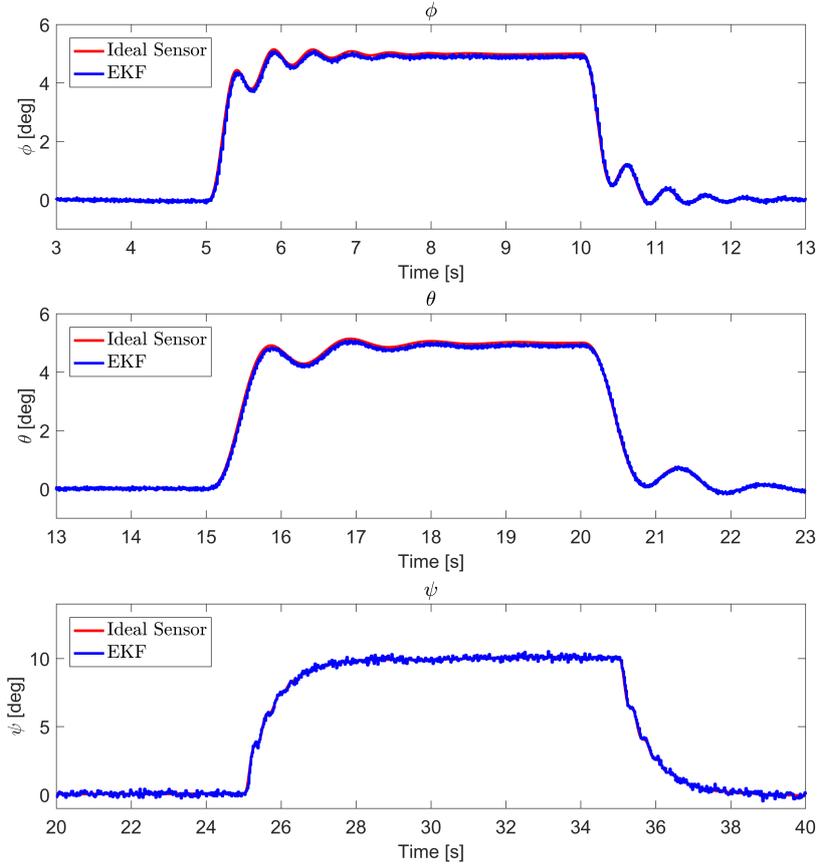


Figure 4.8: Attitude estimation

in which the following weight matrices were used:

$$\mathbf{Q}_{EKF} = \text{diag}([0.02, 0.02, 5]); \quad (4.4.14a)$$

$$\mathbf{R}_{EKF} = \text{diag}([0.093, 0.088, 0.245, 0.301, 0.196, 0.054, 0.054, 0.029]); \quad (4.4.14b)$$

where the elements in  $\mathbf{Q}_{EKF}$  are in  $[rad^2]$  and the elements of  $\mathbf{R}_{EKF}$  are in  $[(rad/s)^2]$ .

Table 4.2 shows the performance of the estimator.

| Error |           | $\phi$ | $\theta$ | $\psi$ |
|-------|-----------|--------|----------|--------|
| RMSE  | $[deg^2]$ | 0.114  | 0.161    | 0.245  |
| Mean  | $[deg]$   | 0.242  | 0.233    | 0.416  |
| Max   | $[deg]$   | 0.551  | 0.306    | 0.612  |

### 4.4.3 Altitude estimation

A reliable and precise value of both the quadrotor vertical position ( $z$ ) and vertical velocity ( $\dot{v}$ ) is required to enable a good altitude control. However in the platform only the barometer is capable of providing such information, and even then it only provides a value for the vertical position ( $z$ ). As such it is required to implement an estimator for the vertical position and velocity. It should also be noted that the barometer,

and thus the altitude control loop in the Pixhawk autopilot runs at a frequency of 20 Hz, whilst the attitude control loop runs at double the frequency (40 Hz).

[9] tested different estimators, and concluded that using a Simple Kalman Filter with no input variable yielded the best results, while still being independent of the platform which is always desired. The estimator model is presented in equation (4.4.15):

$$\begin{bmatrix} \dot{z}_k \\ z_k \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \tau & 1 \end{bmatrix} \begin{bmatrix} \dot{z}_{k-1} \\ z_{k-1} \end{bmatrix} + w_{k-1} \quad (4.4.15)$$

$$\mathbf{y}_k = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{z}_k \\ z_k \end{bmatrix} + v_k \quad (4.4.16)$$

where the following matrices were used:

$$\mathbf{Q}_{KFalt} = \text{diag}([45, 0.5]); \quad (4.4.17a)$$

$$R_{KFalt} = 400; \quad (4.4.17b)$$

$$\mathbf{P}_{KFalt} = \mathbf{I}_{2 \times 2} \quad (4.4.17c)$$

The first and second elements of  $\mathbf{Q}_{KFalt}$  and  $\mathbf{P}_{KFalt}$  are in  $[(m/s)^2]$  and  $[m^2]$  respectively while  $R_{KFalt}$  is in  $[Pa^2]$ . Lastly the initial state vector is  $\mathbf{x}_o = [0, z_0]^T$  where  $z_0$  is the quadrotor initial altitude.

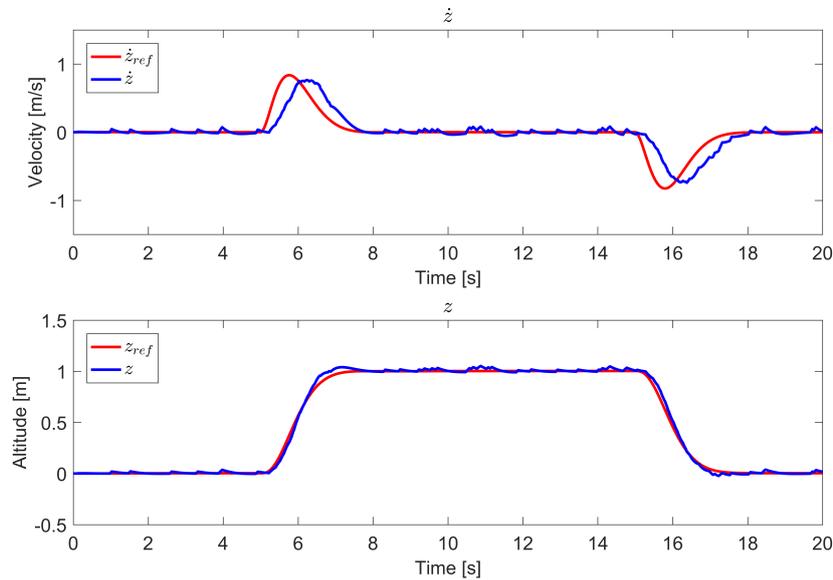


Figure 4.9: Altitude estimation

Even though the only used sensor is the barometer the altitude estimation is quite satisfactory. The barometer has the added bonus of being independent of the quadrotor attitude and velocity. However since no input signal was used the velocity estimation has a delay which is expected as the velocity is obtained by the derivative of the barometer signal and especially because this signal not updated very frequently. Table 4.3 presents the performance of this estimator.

Table 4.3: Altitude estimation performance

| Vertical Velocity |                  |                  | Altitude |                |                |
|-------------------|------------------|------------------|----------|----------------|----------------|
| RMSE [m/s]        | Mean error [m/s] | Max. error [m/s] | RMSE [m] | Mean error [m] | Max. error [m] |
| 0.141             | 0.019            | 0.509            | 0.027    | 0.003          | 0.075          |

## 4.5 Simulation results

In order to simplify our study up until this point the estimation and control parts of the simulator were separated from each other, however it is important to access the simulator performance with all working parts. As such two tests were performed with the following control weight matrices:

$$\mathbf{Q}_{\text{att}} = \text{diag}([750, 3000, 450, 4000, 10500, 800]); \quad (4.5.1a)$$

$$\mathbf{Q}_{\text{alt}} = \text{diag}([50, 100]); \quad (4.5.1b)$$

$$\mathbf{Q}_{\text{pos}} = \text{diag}([2500, 2500, 3500, 3500]); \quad (4.5.1c)$$

$$\mathbf{R} = \text{diag}([0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1]); \quad (4.5.1d)$$

where the same matrix  $R$  was used for all 3 controllers. Matrices  $\mathbf{Q}_{\text{att}}$  and  $\mathbf{Q}_{\text{alt}}$  had to be adjusted to account for the less precise signals, and as such have lower gains. Because ideal sensors are still assumed for  $x$  and  $y$  position, as well as the respective velocities, matrix  $\mathbf{Q}_{\text{pos}}$  did not warrant any changes. Figure 4.10 shows the performance to three distinct angular stimuli and their effect on the quadrotor position, similarly to what was done in section 4.3.1.

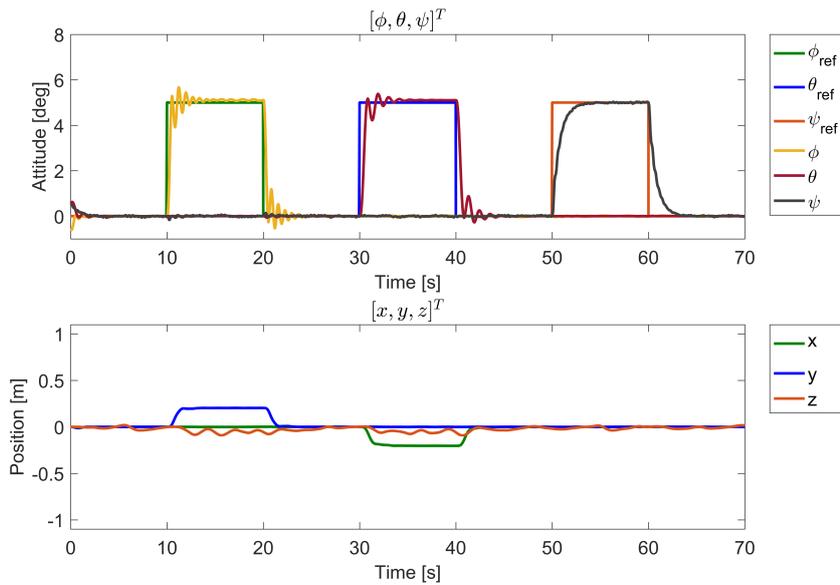


Figure 4.10: Tilt-Quadrotor performance to different angular step references

Figure 4.11 shows the quadrotor response to an unfavourable initial position with a roll angle of

$-15^\circ$  and a pitch angle of  $15^\circ$ . The quadrotor manages to reach a stable position in a few seconds. Position wise the response is good with altitude falling  $0.5m$  as expected because the altitude estimator has a delay. Horizontal position is maintained between  $0.3m$  of the initial and reference position, for both  $x$  and  $y$  position, meaning that the quadrotor is capable of performing several controlling motions simultaneously.

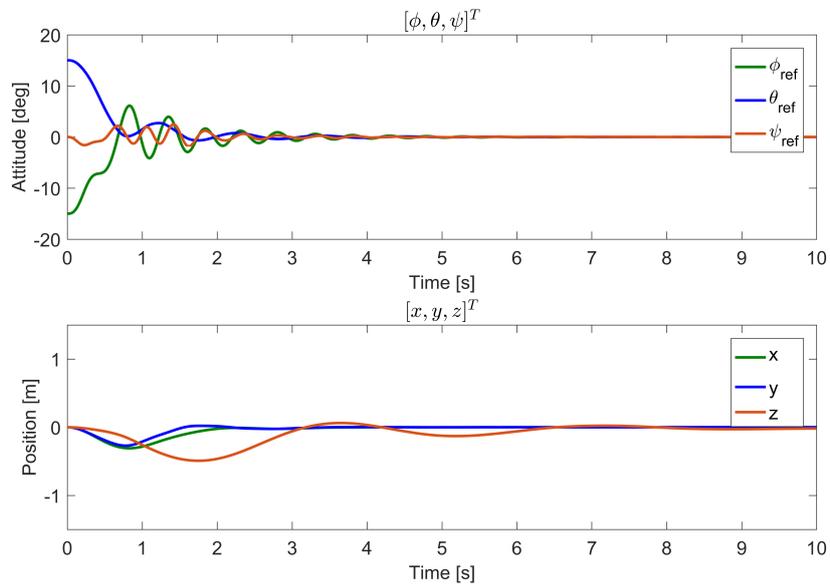


Figure 4.11: Tilt-Quadrotor performance to an unfavourable initial position



## Chapter 5

# Experimental implementation

In this chapter the experimental procedure will be explained.

### 5.1 Experimental procedure

A rough explanation of the platform wiring and functioning should be included here. The battery is first connected to the Pixhawk Power module<sup>1</sup>, as can be seen in figure 5.1, that will provide the necessary power to the Pixhawk autopilot. It also has voltage and current sensors that allows the autopilot to estimate the battery charge with great precision, providing this information to the pilot, preventing damages to the battery or unexpected failures that would most likely end in crashes.



Figure 5.1: Pixhawk power module

The other output connects to the Power Distribution Board which was designed by [8]. This board will distribute power to all 4 ESCs. As stated in section 3.2.1 each ESC has a battery eliminator circuit (BEC) which will provide the 5V power to one servo motor. This is also accomplished with the use of the PDB as well as the signal cable routing from the Pixhawk. The ESCs are then connected to their respective motor and their signal cable is routed from the Pixhawk board through the PDB. A simplified schematic of this wiring is presented in figure 5.2 with only one servo and one dc motor.

In a first approach the platform was tested attached to a pole present in the laboratory which was later improved with the addition of a kneecap like workbench developed by [9] shown in figure 5.3. This workbench allowed a safe environment to test the autopilot implementation, so that all outputs behaved as expected. This workbench allows the ALIV3 to move freely in the yaw angle, and to move in  $\pm 15$

---

<sup>1</sup>[https://pixhawk.org/peripherals/power\\_module](https://pixhawk.org/peripherals/power_module)

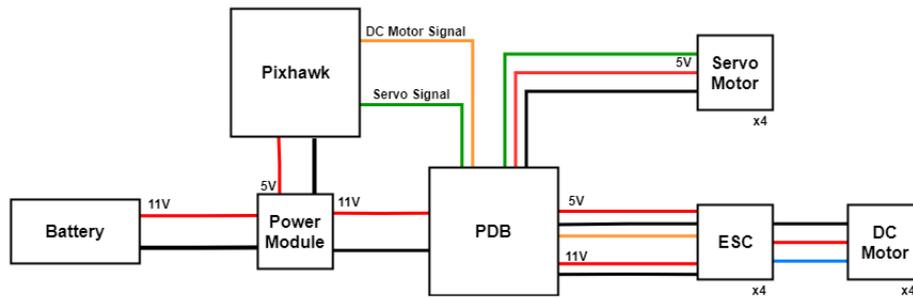


Figure 5.2: Simplified wiring schematic

degrees in both roll and pitch angles. Although a bit far from the real scenario, it is very useful to test the ALIV3 performance in a safe and controlled environment.



Figure 5.3: Testing workbench

## 5.2 Autopilot implementation

The final goal of this project is to implement these findings in a real life prototype, the ALIV3. However the Pixhawk autopilot does not have the same control structure as the developed simulator. As such some adaptation is required to ensure compatibility between the autopilot and the custom code from the simulator.

In figure 5.4 the PX4 flight stack multicopter controller is depicted. Before arming the quadrotor a signal is sent to the autopilot identifying the desired control group, for example Altitude Hold, Position Hold, Manual control etc. The selected control group will determine the response to the signals sent using the remote joysticks were in most cases its either a reference velocity or reference attitude. In this case the control group used will be Stabilise mode, meaning that the left joystick will command a yaw rate (horizontal joystick movement) and thrust signal (vertical joystick movement), whilst the right joystick will command roll (horizontal joystick movement) and pitch angles (vertical joystick movement).

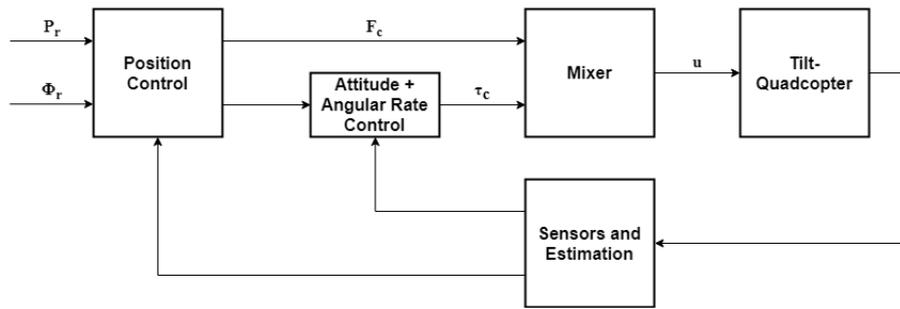


Figure 5.4: PX4 control diagram adapted from [23]

The Position Control function receives these signals and outputs a command attitude  $\Phi_c = [\phi_c, \theta_c, \psi_c]^T$  in Euler angles to the Attitude + Angular Rate control function which in turn outputs a normalized torque command  $\tau_c = [\tau_x, \tau_y, \tau_z]^T \in [-1, 1]$  to the Mixer. Additionally the Position Control function also outputs a normalized thrust command  $F_c \in [0, 1]$  directly to the Mixer. It should be noted that up to this point no function takes into account the precise geometry of platform it controls, other than the fact that it is a multicopter. The unique characteristics of the platform are only present in the Mixer which receives as inputs the normalized forces and moments and in turn outputs the PWM signals for each individual motor and is usually known as control allocation (CA).

Since building custom code for the PX4 flight stack is very time consuming it was decided to use the original Position and Attitude + Angular Rate Controllers, and implement the new CA in the mixer which is derived from the linearized model developed in section 2.8. Since the mixer defines the linear relation between the required movements and the PWM outputs, matrix  $B_v$  defined in (2.8.12) was used.

Since the Mixer is normalized for each actuator, each column was divided by the respective absolute maximum, so that all values in the matrix are in the interval  $[-1, 1]$ . The Mixer is encoded in normalized units in the range  $[-10000, 10000]$  and as such all elements in the matrix are multiplied by 10 000.

It should be noted that in practice the Mixer is divided in two files, one for the main actuators (DC motors) and one for the auxiliary actuators (servo motors). However their structure is the same. The Mixer is written in C++ and at its core is comprised of 8 blocks of code, one for each actuator. An example of one of these blocks is shown below for Servo Motor number 3 in figure 5.5.

|        | Scale factor  | Offset | Output range |
|--------|---------------|--------|--------------|
| M: 2   |               | ↓      |              |
| O:     | 10000 10000   | 0      | -10000 10000 |
| S: 0 0 | 1000 1000     | 0      | -10000 10000 |
| S: 0 2 | -10000 -10000 | 0      | -10000 10000 |
|        | Input ID      |        |              |

Figure 5.5: Example of mixer block

where first line states the number of inputs to this mixer in this case 2, as this servo is only requested in roll and yaw motions. The second line defines the output scaling. In order of the columns the numbers

mean that the output will be scaled by 1 (10 000) in the negative and in the positive regions of the interval. It will apply no offset (0) and it will output to the full range  $[-1, 1]$  (-10 000, 10 000). This output is then computed to the respective PWM range defined in the ground station according to the actuator in question.

The next and last two lines represent the output scalers. The first number (0 in both lines) defines the control group which is always the same in this implementation. The second number defines the input command to be used in that specific scaler (0 - Roll, 1 - Pitch, 2 - Yaw, 3 - Thrust). The following numbers are just like in the output scaling line. Regarding the roll command (third line) the input is scaled by a factor of 0.1 (1 000) in both the negative and positive ranges. Then it applies no offset and outputs to the full range. On the other hand a yaw input is scaled by a factor of 1.0 (10 000).

As seen in section 2.4.3, when a roll motion is performed, rotors 2 and 4 need to tilt slightly in the  $\theta_2$  and  $\theta_4$  angles to prevent a yaw motion, hence the low scaling factor in the third line of the Mixer. However a yaw motion is mostly achieved by actions of servos 3 and 4 and as such the scaling factor is maximum.

Appendix A shows the two Mixer files required to operate the ALIV3 and the corresponding Airframe file. It should also be noted that these files have also been added to the PX4 open source flight stack.

### 5.3 Experimental validation

In a first approach the ALIV3 was tested in the workbench. The controller parameters were tuned and once the ALIV3 showed to be stable in the workbench the platform was tested in the arena of the Mechatronics Laboratory. Even though this arena is equipped with a Qualisys motion capture system, because of the tilting of quadrotor arms the signal was incomplete and as such the presented results are obtained from the autopilot log. Figure 5.6 shows the ALIV3 in stable flight in the arena.



Figure 5.6: ALIV3 in stable flight

Figure 5.7 shows the results of one of the ALIV3 test flights. Starting with the roll angle analysis, figures 5.7(a) and 5.7(b) show the roll angle  $\phi$  and speed  $p$ . The ALIV3 has a tendency to drift to the left and as such a small positive tilt is required to maintain position. This is mostly likely due to vibrations in the main arms, or bad trim point in one of the servos. Nonetheless the response is fast and accurate,

although with overshoots and oscillations which can be removed with fine tuning. The angular velocity measurements are filled with unavoidable noise from the motors and vibrations.

The platform does not drift forward nor backwards and as such the pitch angle has a more stable reference and thus a more stable response. The forward and backward motion of the platform was tested, albeit with the classical configuration achieved by the tilting of the central core, as can be seen in figure 5.7(c) around the 35sec to the 40sec mark. The platform travelled several metres forward, stopped, and then travelled backwards, once again reaching a stable position.

Lastly the yaw angle response is discussed. The platform responds well, keeping its yaw angle close to the reference and is only slightly affected from the controlling motions of the other angles, meaning that the linearized model and corresponding mixer are a good approximation. Special attention should be given to the spikes around the 25sec and 42sec mark in figures 5.7(e) and 5.7(f). The first spike corresponds to a yaw rotation command of about  $90^\circ$ , which is then quickly reversed. The platform performs this motion at speeds reaching  $200^\circ/s$ , which is a great result. The second spike corresponds to a full rotation about the  $z$  axis in one quick and successive motion. Once again the platform performs this motion with great speeds.

Figure 5.8 shows the PWM signals sent to each of the eight actuators. These results confirm the previously made predictions. The pitching motion is achieved by a difference in torque between motors 1 and 3 and not by the action of the servo motors (figures 5.8(b) and 5.8(b)). However as said before, simulation results show that this manoeuvre is possible with both configurations, with the classical configuration being more efficient at the cost of a small disturbance in the equilibrium of moments and thus the one being implemented in the ALIV3. It should be noted that servo motors 1 and 2 are still used, albeit in very slight responses, whenever there are pitching commands, as a way to counteract the expected yaw motion from the classical configuration.

Another interesting result is the action of servo motors number 3 and 4 in the yaw motion. There is a clear correlation between the yaw angle and speed shown in figures 5.7(e) and 5.7(f), and the servo signals in figures 5.8(f) and 5.8(h). As expected the actuation of servos 3 and 4 is symmetrical for a yaw motion, as shown. It is also clear that servo motors 3 and 4 are much more active, even outside of yaw commands, as they are required to maintain a yaw moment balance, whenever the platform performed a roll motion, which as discussed previously, are more frequent and intense than the pitch motions.

The next step is to perform fine tuning in both the autopilot controllers as well as in the platform actuators. The propellers can and should be more finely balanced so as to reduce motor vibrations. The servo arms should be tightened or attached in a different way to reduce platform vibrations. And lastly the servo motors should be calibrated in the workbench during testing to make sure the rotors are fully vertical when the servo is in the trim position.

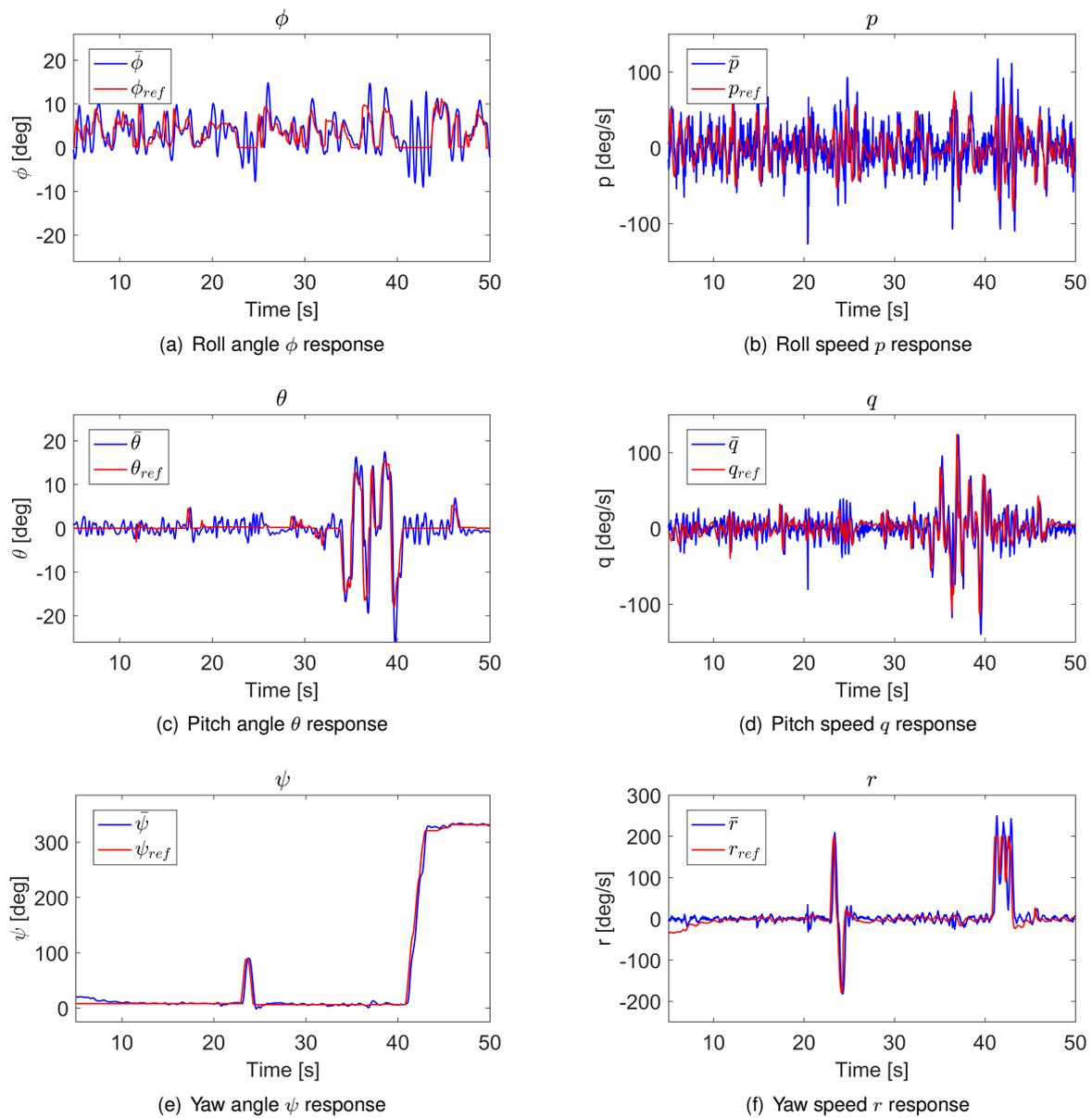
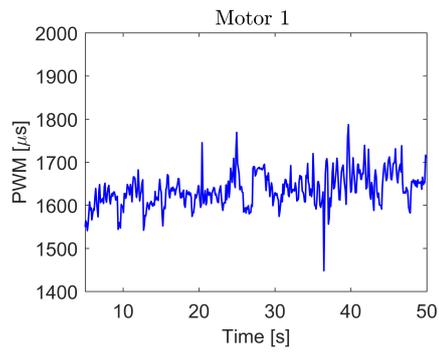
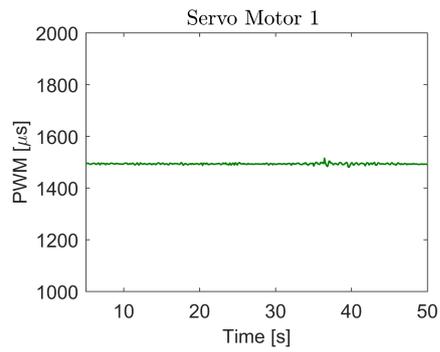


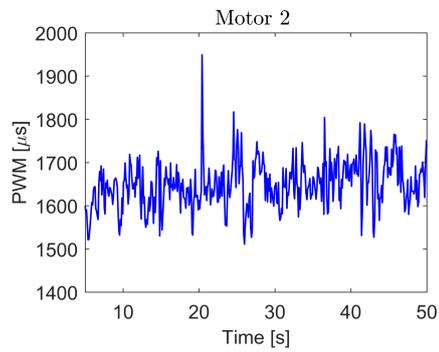
Figure 5.7: Attitude response of the ALIV3



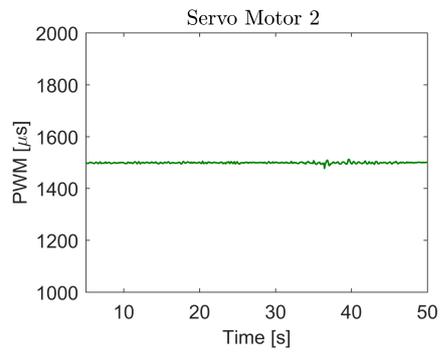
(a) DC motor 1 PWM signal



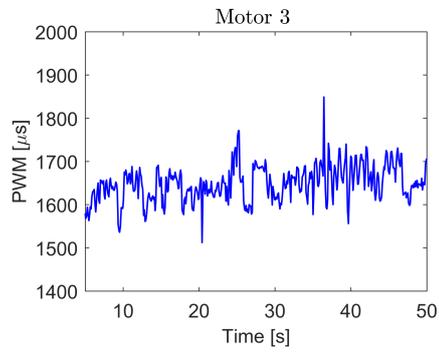
(b) Servo Motor 1 PWM signal



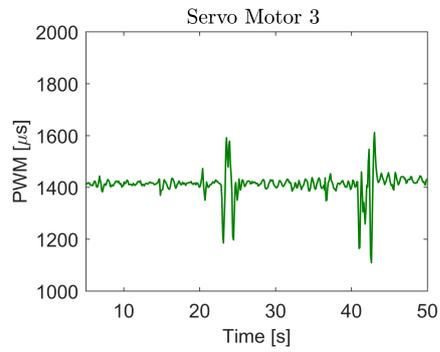
(c) DC motor 2 PWM signal



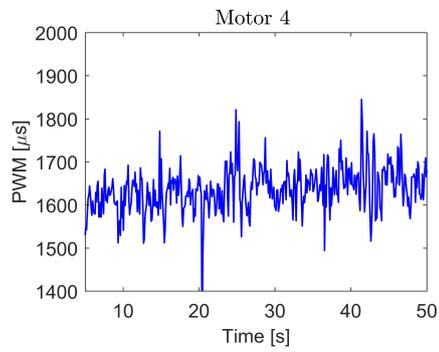
(d) Servo Motor 2 PWM signal



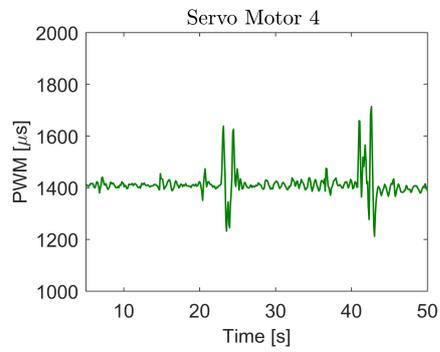
(e) DC motor 3 PWM signal



(f) Servo Motor 3 PWM signal



(g) DC motor 4 PWM signal



(h) Servo Motor 4 PWM signal

Figure 5.8: PWM actuation signals



## Chapter 6

# Conclusions and future work

This chapter presents the main conclusions drawn from the study developed and presented in the previous chapters. It also presents the main suggestions for future work to be done regarding the ALIV project.

### 6.1 Conclusions

After a brief introduction to the ALIV project, this work began by presenting the models for the different actuators present in the ALIV3, i.e. the DC motors, the servo motors and the propellers. Afterwards the principal motions of the Tilt-Quadrotor were presented, explained in detail, and proven possible mathematically. It was shown that the Tilt-Quadrotor can perform all three angular motions required for stabilisation, and in the case of a Pitch motion, it can achieve it by two totally different configurations. Next the Tilt-Quadrotor non-linear model was presented as well as the sensors model. The model was linearized which was later used in the development of the LQR controller. The developed controller proved capable of controlling both the Tilt-Quadrotor attitude and position.

A simple Kalman Filter was developed for altitude and velocity estimation using the barometer, which proved to be good for the altitude estimation but could do with some improvements to the velocity estimation. As discussed there were some expected delays due to the lack of predictive inputs. For the attitude estimation an Extended Kalman Filter was developed using the accelerometer and magnetometer readings, with the gyroscope readings as inputs. Contrary to most works the magnetometer was used for the estimation of all 3 attitude angles, as a way to compensate the accelerometer readings which could be corrupted from the horizontal forces produced by the tilted rotors. The estimator proved to be robust and provided accurate results. The  $x$  and  $y$  position, as well as their respective velocities, were not estimated and were assumed as known. The controllers, the estimators and the platform were implemented in a simulator in MATLAB Simulink, which allowed the tuning of all parts and analysis of the performance of the Tilt-Quadrotor.

The final goal of this work was to achieve experimental validation. First the ALIV3 platform required a much needed restoration. Many parts were replaced and the wiring redone to account for the servo

motors. The linearized model was implemented in the autopilot, tuned and tested first in a workbench and afterwards in the arena where it achieved stable flight.

As such his work culminates in mainly two fulfilled objectives. The development of a working simulator where different control or estimation techniques can be tested for the Tilt-Quadrotor, and a working prototype, the ALIV3, to experimentally validate these results.

## 6.2 Future work

Even though the main goals of the this work were achieved there are several improvements which, due to time constraints, were not implemented. As such regarding the ALIV3 in no particular order:

- The motors, propeller and sensor characteristics were not identified for the ones present in the ALIV3. It would be useful to introduce them in the simulator in order to better model the ALIV3 platform;
- Continue the ardupilot implementation to achieve the final goal of horizontal motion with a levelled central core.
- The GPS model was not developed, which would allow a more realistic simulation of the  $x$  and  $y$  position control. Regarding the ALIV3 it would be very interesting to implement position control with the Qualisys motion tracking system already present in the laboratory.
- The platform could be improved. The inner servo motors could be replaced by higher torque alternatives or even completely different alternatives like an ODrive<sup>1</sup>. The goal should be fast and precise actuation with minimal vibrations.

As more of long term goals the following ideas are suggested:

- The Tilt-Quadrotor is a great platform for the testing of non-linear control techniques, due to its very non-linear nature. For example the LQR controller does not account for the loss of thrust when the rotors are tilted, even though it manages to correct it.
- If the platform is not required to do high speed, sideways motion, then the rotors will have no need to tilt in large angles towards the central core. This means that the large and complex, "U" shaped main arms are not required and can thus be replaced by much simpler straight arms. If the motors were to be raised at the end of these arms than it would provide enough clearance for all motions and it would greatly simplify the structure.
- A new platform where the main arms have airfoils would allow the Tilt-Quadrotor to maintain altitude solely by aerodynamic forces once the horizontal speed would be large enough. This would mean a VTOL aircraft with great flexibility of motions and with great long range efficiency, which could be useful for a variety of applications.

---

<sup>1</sup><https://odriverobotics.com/>

# Bibliography

- [1] D. Brescianini and R. D'Andrea. Design, modeling and control of an omni-directional aerial vehicle. *Proc. IEEE Int'l Conference on Robotics and Automation*, 2016.
- [2] M. Ryll, H. H. Bulthoff, and P. R. Giordano. Modeling and control of a quadrotor UAV with tilting propellers. *2012 IEEE International Conference on Robotics and Automation*, 2012.
- [3] C. Papachristos, K. Alexis, and A. Tzes. Technical activities execution with a TiltRotor UAS employing explicit model predictive control. *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 2014.
- [4] S. Raposo. System And Process Of Vector Propulsion With Independent Control Of Three Translation And Three Rotation Axis, 2012.
- [5] S. Costa. Controlo e Simulação de um Quadrirotor convencional. Master's thesis, Instituto Superior Técnico, Universidade Técnica de Lisboa, Lisbon, Portugal.
- [6] F. Pedro. Projecto Preliminar de um Quadrirotor. Master's thesis, Instituto Superior Técnico, Universidade Técnica de Lisboa, Lisbon, Portugal.
- [7] N. D. S. Fernandes. Design and construction of a multi-rotor with various degrees of freedom . Master's thesis, Instituto Superior Técnico, Universidade Técnica de Lisboa, Lisbon, Portugal.
- [8] E. Mateos. Estimation and Control of a Tilt-Quadrotor Attitude Estanislao Cantos Mateos Thesis to obtain the Master of Science Degree in Aerospace Engineering Examination Committee. Master's thesis, Instituto Superior Técnico, Universidade Técnica de Lisboa, Lisbon, Portugal.
- [9] D. Esteves. Development and Experimental Validation of an Indoor Low Cost Quadrotor: Hover Stabilization with Altitude Control. Master's thesis, Instituto Superior Técnico, Universidade Técnica de Lisboa, Lisbon, Portugal.
- [10] S. Bouadallah, M. Becker, et al. Toward obstacle avoidance on quadrotors. In *Proceedings of the XII International Symposium on Dynamic Problems of Mechanics*, 2007.
- [11] P. P. Teodorescu. *Mechanical Systems, Classical Models*. Springer, 2007.
- [12] B. L. Stevens, F. L. Lewis, and E. N. Johnson. *Aircraft control and simulation : dynamics, controls design, and autonomous systems*. Wiley, 2016.
- [13] J. A. Wheeler and E. F. Taylor. *Spacetime physics*. WH Freeman San Francisco, CA, USA:, 1966.

- [14] W. Rindler. *Essential relativity: special, general, and cosmological*. Springer Science & Business Media, 2012.
- [15] W. T. Ang, S. Y. Khoo, P. K. Khosla, and C. N. Riviere. Physical model of a mems accelerometer for low-g motion tracking applications. volume 2, pages 1345–1351 Vol.2, 2004.
- [16] T. H. Robert Mahony and S.-H. Cha. A coupled estimation and control analysis for attitude stabilization of mini aerial vehicles. *Proceedings of the Australasian Conference on Robotics and Automation*, 2006.
- [17] R. M. J. K. Mark Euston, Paul Coote and T. Hamel. A complementary filter for attitude estimation of a fixed-wing uav. *International Conference on Intelligent Robots and Systems*, pages 340–345, 2008.
- [18] B. S. M. Henriques. Estimation and Control of a Quadrotor Attitude. Master’s thesis, Instituto Superior Técnico, Universidade Técnica de Lisboa, Lisbon, Portugal.
- [19] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 1960.
- [20] G. Welch and G. Bishop. An Introduction to the Kalman Filter. *Proc. Siggraph Course*, 2006.
- [21] R. Yadin. Attitude determination and bias estimation using kalman filtering. *US Air Force Academy Report*, 2009.
- [22] M. J. Wilson. Attitude Determination With Magnetometers for Gun-Launched Munitions. Technical report, 2004.
- [23] R. Coelho. Quadrotor attitude control using incremental nonlinear dynamics inversion. Master’s thesis, Instituto Superior Técnico, Universidade Técnica de Lisboa, Lisbon, Portugal.

# Appendix A

## Autopilot Files

### A.1 Airframe file

```
#!/nsh
#
# @name Tilt-Quadrotor
#
# @url http://www.alivaero.com/the-project.html
#
# @type Tilt-Quad
# @class Copter
#
# @output MAIN1 motor 1
# @output MAIN2 motor 2
# @output MAIN3 motor 3
# @output MAIN4 motor 4
#
# @output AUX1 servo motor 1
# @output AUX2 servo motor 2
# @output AUX3 servo motor 3
# @output AUX4 servo motor 4
#
# @maintainer Ricardo Marques <ricardojmarques@tecnico.ulisboa.pt>
#

sh /etc/init.d/rc.mc_defaults

#Parameters here:
```

```

param set LED_RGB_MAXBRT 8

# Set mixer
set MIXER tilt_quad
set MIXER_AUX tilt_quad
set PWM_OUT 1234

```

## A.2 Main mixer

Tilt-Quadrotor mixer for PX4FMU (1/2) V2.0

=====

This file defines the main outputs mixer for a Tilt-Quadrotor in the + configuration.

```

# @output MAIN1 motor 1
# @output MAIN2 motor 2
# @output MAIN3 motor 3
# @output MAIN4 motor 4

```

Motor 1

M: 2

```

O:      10000  10000      0 -10000  10000
S: 0 1  10000  10000      0 -10000  10000
S: 0 3  10000  10000      0 -10000  10000

```

Motor 2

M: 2

```

O:      10000  10000      0 -10000  10000
S: 0 0  10000  10000      0 -10000  10000
S: 0 3  10000  10000      0 -10000  10000

```

Motor 3

M: 2

```

O:      10000  10000      0 -10000  10000
S: 0 1 -10000 -10000      0 -10000  10000

```

```
S: 0 3 10000 10000 0 -10000 10000
```

Motor 4

```
M: 2
```

```
O: 10000 10000 0 -10000 10000
```

```
S: 0 0 -10000 -10000 0 -10000 10000
```

```
S: 0 3 10000 10000 0 -10000 10000
```

### A.3 Aux mixer

Tilt-Quadrotor mixer for PX4FMU (2/2) V2.0

```
=====
```

This file defines the aux outputs mixer for a Tilt-Quadrotor in the + configuration.

```
# @output AUX1 servo motor 1
```

```
# @output AUX2 servo motor 2
```

```
# @output AUX3 servo motor 3
```

```
# @output AUX4 servo motor 4
```

Servo 1

```
M: 1
```

```
O: 10000 10000 0 -10000 10000
```

```
S: 0 1 -1000 -1000 0 -10000 10000
```

Servo 2

```
M: 1
```

```
O: 10000 10000 0 -10000 10000
```

```
S: 0 1 1000 1000 0 -10000 10000
```

Servo 3

```
M: 2
```

```
O: 10000 10000 0 -10000 10000
```

S: 0 0 1000 1000 0 -10000 10000  
S: 0 2 -10000 -10000 0 -10000 10000

Servo 4

M: 2

O: 10000 10000 0 -10000 10000  
S: 0 0 -1000 -1000 0 -10000 10000  
S: 0 2 10000 10000 0 -10000 10000