



## **Estimation and Control of a Tilt-Quadrotor Attitude**

**Estanislao Cantos Mateos**

Thesis to obtain the Master of Science Degree in  
**Aerospace Engineering**

### **Examination Committee**

Chairperson: Prof. Fernando José Parracho Lau

Supervisors: Prof. Filipe Szolnoky Ramos Pinto Cunha  
Prof<sup>a</sup>. Alexandra Bento Moutinho

Members of the committee: Prof. Agostinho Rui Alves da Fonseca.  
Prof. José Raúl Carreira Azinheira.

**November 2013**



# Acknowledgment

I would like to thank Prof. Filipe Cunha and Prof. Alexandra Moutinho for the constant help provided, for all the counseling about following a systematic process, for let me to develop my creativity and teach me all the knowledge that was necessary.

I would also like to thank Alexandre Cruz and David Melo for all the help provided in the laboratory, showing me how work the different machines, the electronics of the Ardupilot and all the ideas that I could learn from their work.

I would also like to thank my mother, my father, my older brother for supporting me during all the process of my background, and specially I would thank to my Brother Juan Carlos for all the counseling that made me easier to overpass any obstacle during my degree.

I would also like to thank to my second family in Lisbon, Laura Cornejo, Esperanza Fernández, Carlos Garcia and Marta Castro for all the experiences lived in this wonderful city making feel my life there as I was at home.



# Abstract

The aim of the present work is to continue the development of a prototype of a tilt-quadrotor, focusing on the estimation and control of the attitude. A Tilt-quadrotor is a fusion of the quadrotor and tiltrotor concepts, enabling it to move in all six degrees of freedom with the advantage of maintaining its central core leveled. This possibility results from add a tilting movement in two opposed rotors while the other two rotors remain fixed.

First the differences in motion and configurations of a tilt-quadrotor in relation with a common quadrotor are explained. The mass properties and actuators of the prototype are measured through tests specially developed, like the identification of the motors and the Moments of Inertia. In the identification of the motors was considered the influence of the temperature and the battery discharge.

The Tilt-Quadrotor is equipped with 3-axes accelerometer, 3-axes gyroscope, 3-axes compass and a barometer, all included in the IMU shield of the Ardupilot Mega 1 (APM 1) .These sensors are modeled using real sensor measurements to a better approximation to the real case.

Once the entire model was identified, it was linearized around hovering, which is considered the linearization point. Then the model was implemented in *Simulink*. First a 12 states LQR controller was developed to achieve the stabilization of the ALIV3. Because the model was not either fully controllable or observable, it was developed a 6 states LQR controller.

The model was simulated using ideal sensor in the continuous and discrete case. Once the controller was tuned, it was simulated the model with estimation feedback introducing the Extended Kalman Filter (EKF). The final results suggest that the LQR and the EKF combined could stabilize the ALIV3.

**Keywords:** Quadrotor, Tilting rotors, Extended Kalman Filter, Linear-Quadratic-Regulator Identification



# Resumo

O objetivo do presente trabalho é continuar o desenvolvimento do protótipo do Tilt-Quadrotor, neste caso na estimativa e controlo da atitude. Um Tilt-Quadrotor é uma fusão dos conceitos Quadrotor e tiltrotor, permitindo a sua deslocação nos seis graus de liberdade, com a vantagem de manter o seu núcleo central nivelado. Esta possibilidade resulta de adicionar um movimento de inclinação em dois rotores opostos, enquanto os outros dois rotores permanecem fixos .

Primeiramente, são explicadas as diferenças de movimento e de configurações de um tilt- quadrotor em relação com um quadrotor comum. As propriedades de massa e atuadores do protótipo são medidos através de testes especialmente desenvolvidos, como a identificação dos motores e os momentos de inércia. Na identificação dos motores foi considerada a influência da temperatura e a descarga da bateria .

O Tilt-Quadrotor é equipado com um acelerómetro de 3 eixos , um giroscópio de 3 eixos, um magnetómetro de 3 eixos e um barómetro. Todos os sensores são incluídos na IMU shield do Ardupilot mega 1 (APM 1). Os sensores são modelados utilizando as medições reais do sensor para uma melhor aproximação para o caso real.

Uma vez que foi identificado o modelo inteiro, este foi linearizado em torno da situação a pairar. O modelo foi implementado em Simulink. Primeiro foi desenvolvido um controlador LQR de 12 estados para alcançar a estabilização do ALIV3. Como o modelo não era nem totalmente controlável nem observável, foi desenvolvido um controlador LQR de 6 estados.

O modelo foi simulado usando sensores ideais no caso contínuo e discreto. Uma vez que o controlador foi afinado, foi simulado o modelo com a introdução da estimativa da atitude mediante o Filtro de Kalman Estendido (EKF). Os resultados finais sugerem que o LQR e o EKF combinados poderiam estabilizar a ALIV3 .

**Palavras-chave:** Quadrirotor , Rotores de inclinação variável, Filtro de Kalman Estendido, Identificação, Regulador – Quadrático – Linear.

# Contents

- Contents ..... viii**
- List of tables ..... xii**
- List of figures ..... xiii**
- Abbreviations and Acronyms ..... xv**
- Nomenclature ..... xv**
- 1.Introduction..... 1**
  - 1.1 History of Quadrotors ..... 1
  - 1.2 Context and Motivation ..... 4
  - 1.3 Background..... 6
  - 1.4 Objectives and Contributions ..... 8
  - 1.5 Thesis Structure ..... 9
- 2.Tilt-Quadrotor Modeling..... 10**
  - 2.1 General Modeling Assumptions ..... 10
  - 2.2 ALIV3 Platform ..... 11
    - 2.2.1 Arms..... 11
    - 2.2.2 Central Core ..... 12
    - 2.2.3 Electrical and Electronic Components..... 13
  - 2.3 Tilt Quadrotor Model..... 13
    - 2.3.1 Coordinate System ..... 13
    - 2.3.2 Performance of a Tilt-Quadrotor..... 14
      - i) Leveled Motions..... 16
        - Hovering ..... 16
        - Climbing ..... 17
        - Forward Motion ..... 18
        - Lateral Motion..... 19
        - Yaw Motion ..... 20
      - ii) Rebalancing Motions ..... 22
        - Roll Rebalance ..... 22
        - Pitch Rebalance..... 25
        - Roll and Pitch overall Rebalance without tilting any angle..... 27

iii)	Conclusions .....	28
2.3.3	Kinematics Equations. ....	29
2.3.4	Dynamic Equations. ....	30
2.4	Model of the Sensors .....	31
2.4.1	Gyroscope.....	31
2.4.2	Accelerometer .....	32
2.4.3	Compass .....	33
2.4.4	Barometer .....	33
2.5	Model of the Actuators.....	34
2.5.1	Propellers .....	34
2.5.2	Model of the motors.....	35
2.6	Tilt-Quadrotor Simulator .....	37
2.6.1	Dynamic and Output Equations.....	37
2.6.2	Sensors .....	37
2.6.3	Motors.....	37
2.6.4	Simulation Model. Open Loop. ....	39
<b>3.</b>	<b>Tilt-Quadrotor Identification .....</b>	<b>40</b>
3.1	Prototype Identification .....	40
3.1.1	Center of Gravity.....	40
3.1.2	Moments of Inertia .....	42
3.2	Actuators Identification.....	44
3.2.1	Characteristics of the motors .....	45
3.2.2	Experimental Setup .....	46
3.2.3	Curve PWM-RPM .....	48
3.2.4	Curve Time-RPM .....	49
i)	Battery - Temperature analysis .....	49
ii)	RPM/V variable .....	50
3.2.5	Curve PWM-RPM/V .....	52
3.2.6	BEC influence.....	53
3.2.7	Flight Time .....	54
3.2.8	Propeller identification.....	55
3.3	Sensors Identification.....	58
3.3.1	Characteristics of the sensors .....	58
3.3.2	Experimental identification of the sensors .....	60

<b>4.Tilt-Quadrotor Flight Control .....</b>	<b>63</b>
4.1 Control Method.....	63
4.1.1 Open Loop Response .....	64
4.1.1 LQR Control Description.....	64
4.1.2 Observability and Controllability .....	66
4.1.3 Ideal LQR Full 12 States. ....	66
4.2 Estimation Method.....	69
4.2.1 Kalman Filter .....	69
4.2.2 Extended Kalman Filter .....	70
4.2.3 Height and Vertical Velocity Estimator .....	72
4.3 Simulation Results.....	72
4.3.1 Ideal Simulation.....	72
4.3.2 Real Simulation .....	76
<b>5.Conclusions and Future Work.....</b>	<b>79</b>
5.1 Conclusions .....	79
5.2 Future work.....	80
<b>Bibliography.....</b>	<b>81</b>
<b>Appendices .....</b>	<b>83</b>
Appendix A. Experimental Test Results.....	83
A.1 Moments of Inertia .....	83
A.2 Curve PWM-RPM: .....	83
A.3 Curve Time-RPM Motor 3: .....	84
A.4 Battery vs Power Supply: .....	84
A.5 Curve PWM-RPM/V: .....	85
A.6 Curve with BEC vs without BEC.....	86
A.7 Curve Endurance.....	87
Appendix B. Wiring.....	87
Appendix C. Linearization Matrices .....	90
Appendix D. Maintenance .....	92
Appendix E. Arduino Programs .....	92



# List of tables

Table 3.1: Inertia Measurements .....	44
Table 3.2: Motor Characteristics .....	45
Table 3.3: ESC Characteristics <sup>35</sup> .....	46
Table 3.4: Propeller Setup.....	56
Table 3.5: <i>CT</i> and <i>CP</i> All Motors .....	56
Table 3.6: Accelerometers Characteristics.....	58
Table 3.7: x,y Gyroscopes Characteristics .....	59
Table 3.8: z Gyroscope Characteristics .....	59
Table 3.9: Magnetometer Characteristics.....	60
Table 3.10: Barometer Characteristics.....	60
Table 3.11: Accelerometer Noise Measurements .....	61
Table 3.12: Gyroscopes Noise Measurements.....	61
Table 3.13: Magnetometer Noise Measurements .....	61
Table 3.14: Barometer Pressure Noise Measurement .....	62
Table 3.15: Barometer Altitude Noise Measurement .....	62
Table 4.1: Closed loop poles .....	68
Table 4.2: Rising time and Setting time of the continuous model.....	74
Table 4.3: Poles of the Discrete 6 States Model.....	74
Table 4.4: Attitude Error of the Ideal Model.....	75
Table 4.5: Attitude Error with estimation Feedback .....	77
Table 4.6: Rising Time and Setting time with Estimation Feedback.....	78
Table A.1: Moments of Inertia values .....	83
Table A.2: Curve PWM-RPM values .....	83
Table A.3: Curve Time-RPM Motor 3 values .....	84
Table A.4: Battery vs Power Supply values.....	85
Table A.5: Curve PWM- RPM/V values .....	86
Table A.6: Curve with BEC and without BEC values .....	86
Table A.7: Curve Endurance values .....	87
Table B.1: PDB Pin Description .....	88
Table B.2: ESC Pin Description .....	89

# List of figures

Figure 1.1: Breguet’s Gyroplane [20].....	2
Figure 1.2: Oehmichen 2 <sup>1</sup> .....	2
Figure 1.3: Bothezat Helicopter [20].....	2
Figure 1.4: D’Ascanio Helicopter <sup>3</sup> .....	2
Figure 1.5: Convertawings.....	2
Figure 1.6 Curtiss-Wright VZ <sup>6</sup> .....	3
Figure 1.7 Focke-Achgelis Fa.269 <sup>8</sup> .....	3
Figure 1.8 Bell XV-3 <sup>9</sup> .....	3
Figure 1.9: Curtiss-Wright X-19 <sup>10</sup> .....	4
Figure 1.10: Bell X-22 <sup>13</sup> .....	4
Figure 1.11: Bell Boeing Quad TiltRotor <sup>12</sup> .....	4
Figure 1.12: ArduCopter <sup>14</sup> .....	5
Figure 1.13: Parrot AR.Drone <sup>16</sup> .....	5
Figure 1.14: Swarm of nano Quadrotors <sup>17</sup> .....	5
Figure 1.15: ALIV [14].....	6
Figure 1.16: ALIV2 [14].....	7
Figure 1.17: ALIV3 U-arm [14].....	7
Figure 1.18: LIV3 Slim-Arm [14].....	7
Figure 1.19: ALIV3 [14].....	8
Figure 1.20: ALIV3 Covered [14].....	8
Figure 2.1: Final upgrade of the ALIV3.....	11
Figure 2.2: NED frame.....	14
Figure 2.3: ABC frame ALIV1 [14].....	14
Figure 2.4: Standard Forward Motion.....	15
Figure 2.5: forward motion ALIV platform.....	16
Figure 2.6: Hovering.....	17
Figure 2.7: Climbing.....	18
Figure 2.8: Forward Motion.....	19
Figure 2.9: Lateral Motion.....	20
Figure 2.10: Yaw Motion.....	21
Figure 2.11: Standard Yaw Motion.....	22
Figure 2.12: Roll Rebalance.....	23
Figure 2.13: Pitch Rebalance without tilting.....	25
Figure 2.14: Roll Rebalance tilting rotors.....	26
Figure 2.15: Roll and Pitch Rebalance.....	27
Figure 2.16: Electrical and Mechanical Schematic [13].....	36

Figure 2.17: Dynamic and Output Equations .....	37
Figure 2.18: Implementation of the Sensors .....	38
Figure 2.19: Implementation of the Motors.....	38
Figure 2.20: Implementation of the Open Loop .....	39
Figure 3.1: Determination of CG in the horizontal plane I .....	41
Figure 3.2: Determination of CG in the horizontal plane II .....	41
Figure 3.3: Determination of CG in the vertical plane .....	42
Figure 3.4: Moment of Inertia's Structure.....	43
Figure 3.5: Elastics of the Moment of inertia's Structure.....	43
Figure 3.6: Experimental Setup.....	47
Figure 3.7: Curve PWM-RPM .....	48
Figure 3.8: Time-RPM Motor 3.....	49
Figure 3.9: Battery vs Power Supply .....	50
Figure 3.10: Performance vs Temperature.....	51
Figure 3.11: <i>RPM/V</i> vs Time Motor 3, 1450 $\mu$ s.....	52
Figure 3.12: <i>RPM/V</i> vs PWM All Motors .....	53
Figure 3.13: Relative Error All Motors .....	54
Figure 3.14: BEC vs no BEC supply.....	54
Figure 3.15: Endurance .....	55
Figure 3.16: <i>CT</i> All Motors.....	57
Figure 3.17: <i>CP</i> All Motors .....	57
Figure 4.1: Open Loop response .....	64
Figure 4.2: Reference Set .....	67
Figure 4.3: Closed loop implantation .....	67
Figure 4.4: Response of the system to the reference set .....	68
Figure 4.5: Close Loop Poles .....	68
Figure 4.6: Extended Kalman Filter algorithm.....	70
Figure 4.7: Response to the first test.....	73
Figure 4.8: Response to the third test .....	73
Figure 4.9: Response to the second test .....	74
Figure 4.10: Response without the quantizer.....	74
Figure 4.11: Attitude Stabilization of the Discrete Model.....	75
Figure 4.12: Attitude Stabilization of the Discrete Model with Initial Conditions .....	75
Figure 4.13: Discrete Closed Loop Poles .....	76
Figure 4.14: Implantation of the Model with Estimation Feedback .....	77
Figure 4.15: Attitude Stabilization with Estimation Feedback.....	77
Figure 4.16: Attitude Stabilization with Estimation Feedback and with Initial Conditions .....	78

# Abbreviations and Acronyms

ABC – Aircraft-Body-Centered.

ALIV – Autonomous Locomotion Individual Vehicle.

APM – ArduPilot Mega.

CG – Center of Gravity.

DIY – Do It Yourself.

EKF – Extended Kalman Filter

ESC – Electronic Speed Control.

FAI – Fédération Aéronautique Internationale.

IMU – Inertial Measurement Unit.

LQR – Linear Quadratic Regulator.

NED – North East Down.

PWM – Pulse Width Modulation

UAV – Unmanned Aerial Vehicle.

VTOL – Vertical Take-Off and Landing.

# Nomenclature

## Greek Symbols

$\zeta$  – Damping ratio.

$\theta_i$  – Pitch angle of rotor  $i = 2,4$

$\mu_g$  – Gaussian measurement noise of the gyroscope.  $\mu_g = [\mu_{g-x}, \mu_{g-y}, \mu_{g-z}]$ .

$\mu_a$  – Gaussian measurement noise of the accelerometer.  $\mu_a = [\mu_{a-x}, \mu_{a-y}, \mu_{a-z}]$ .

$\mu_m$  – Gaussian measurement noise of the compass.  $\mu_m = [\mu_{m-x}, \mu_{m-y}, \mu_{m-z}]$ .

$\mu_p$  – Gaussian measurement noise of the barometer.

$\rho$  – Density of the air.

$\tau$  – Time constant of the of the motors.

$\Phi = [\phi, \theta, \psi]^T \in \mathbb{R}^3$  – Vector of attitude angles..

$\phi_i$  – Roll angle of rotor  $i = 2,4$ .

$\omega_i$  – Angular velocity of the rotor  $i$

$\Omega^B = [P, Q, R]^T \in \mathbb{R}^3$  – Angular velocity of the tiltquadrotor.

$\bar{\Omega}^B$  – Measured angular rates in the body-fixed frame.  $\bar{\Omega}^B = [\bar{g}_x, \bar{g}_y, \bar{g}_z] \in \mathbb{R}^3$

## Roman Symbols

$A$  – Amplitude of the free oscillation in the Inertia measurement.

$A_0$  – Initial amplitude of the free oscillation in the Inertia measurement.

$\mathbf{a}^B = [a_x, a_y, a_z] \in \mathbb{R}^3$  – Acceleration of the tilt-quadrotor in the body frame.

$\bar{\mathbf{a}}^B$  – Measured acceleration of the tilt-quadrotor in the body-fixed frame;  $\bar{\mathbf{a}}^B = [\bar{a}_x, \bar{a}_y, \bar{a}_z]$ .

$b_g$  – Constant bias of the gyroscope.

$b_a$  – Constant bias of the accelerometer.

$b_m$  – Constant bias of the compass.

$b$  – Motor viscous friction constant.

$c$  – Damping coefficient in the Inertia measurement.

$C_T$  – Thrust coefficient.

$C_p$  – Power coefficient.

$\mathbf{D}^B = [D_x, D_y, D_z]^T \in \mathbb{R}^3$  – Drag Force applied on the tilt-quadrotor.

$d$  – Distance from the rotors to the Center of Gravity.

$\mathbf{F}^B = [F_x, F_y, F_z]^T \in \mathbb{R}^3$  – Forces applied on the tilt-quadrotor in the body frame by the rotors.

$\mathbf{g}^I = [0, 0, g_0]^T \in \mathbb{R}^3$  – Gravity force in the inertial frame

$g_0$  – Constant of gravity.

$\mathbf{I} = \text{diag}([I_x, I_y, I_z])$  – Matrix of Inertia of the tilt-quadrotor.

$i$  – Intensity supplied to the motors.

$J$  – Moment of inertia of the motors.

$J_{lqr}$  – Cost function of the LQR controller.

$k$  – Stiffness in the Inertia measurement.

$K_T$  – Constant which relates thrust and angular velocity of a propeller.

$K_p$  – Constant which relates power and angular velocity of a propeller.

$\mathbf{K}_{lqr}$  – Optimal gain matrix of the LQR controller.

$k_e$  – Electromotive force constant.  
 $k_t$  – Motor torque constant.  
 $k_i$  – Static gain of the motor  $i$ .  
 $L$  – Motor electric inductance.  
 $\mathbf{M}^B = [M_x, M_y, M_z]^T \in \mathbb{R}^3$  – Moments applied on the tilt-quadrotor in the body frame by the rotors.  
 $m$  – Mass of the tilt-quadrotor.  
 $\bar{\mathbf{N}}$  – Measured magnetic vector, defined in the body-fixed frame;  $\bar{\mathbf{N}}^B = [\bar{N}_x, \bar{N}_y, \bar{N}_z]$ .  
 $O$  – Origin of the inertial frame.  
 $O_c$  – Origin of the body frame.  
 $\mathbf{P}^I = [X, Y, Z]^T \in \mathbb{R}^3$  – Cartesian components in the inertial frame.  
 $\mathbf{P}^B = [x, y, z]^T \in \mathbb{R}^3$  – Cartesian components in the body frame.  
 $\mathbf{PWM} = [PWM_1, PWM_2, PWM_3, PWM_4]^T \in \mathbb{R}^4$  – PWM signal sent to the motors.  
 $p$  – Atmospheric pressure.  
 $p_0$  – Atmospheric pressure at sea level  
 $\bar{p}$  – Measured pressure.  
 $Q_i$  – Moment of rotor  $i$ .  
 $\mathbf{Q}_{lqr}$  – Weighting matrix of the LQR controller.  
 $\mathbf{Q}_k$  – Weighting matrices of the Kalman Filter.  
 $\mathbf{R}_{lqr}$  – Weighting matrix of the LQR controller.  
 $\mathbf{R}_k$  – Weighting matrix of the Kalman Filter.  
 $R_e$  – Equivalent Radius of the rotors.  
 $r_r$  – Radius of the rotors.  
 $\mathbf{r}_s$  – Distance from the accelerometer to the CG.  $\mathbf{r}_s = [r_{s-x}, r_{s-y}, r_{s-z}]$ .  
 $R_m$  – Motor electric resistance.  
 $\mathbf{S}$  – Rotation matrix from the inertial to the body frame.  
 $T_i$  – Thrust of rotor  $i$ .  
 $\bar{T}$  – Medium thrust at hovering.  
 $T$  – Period of the free oscillation in the Inertia measurement.  
 $\mathbf{U}$  – Input vector of the tilt-quadrotor system.  
 $v$  – Voltage supplied to the motors  
 $\mathbf{V}^B = [U, V, W]^T \in \mathbb{R}^3$  – Vector velocity in the ABC frame.  
 $\mathbf{v}_k$  – Measurement noise.  
 $\mathbf{w}_k$  – Process noise of the Kalman filter.  
 $\mathbf{X}$  – State vector of the tilt-quadrotor system.  
 $\mathbf{Y}$  – Output vector of the tilt-quadrotor system.

# Chapter 1

## Introduction

The Master's thesis developed in next chapters is a continuation of Filipe Pedro's work [4] from 2009, in which the author made a preliminary design of a "nonconventional" quadrotor, which is called in the present work as Tilt-Quadrotor, based on the prototype of Eng. Severino Raposo's Autonomous Locomotion Individual Vehicle (ALIV) [17]. Filipe Pedro's work resulted in the ALIV2 prototype. In 2011, the research was continued by Nelson dos Santos Fernandes's work [14] resulting in an improved prototype, the ALIV3.

In this Chapter, a brief review of the history of quadrotors is presented. Then the state of the art of UAV (Unmanned Aerial Vehicle) quadrotors is mentioned. The work performed by the previous researchers is described. And finally, the objectives, contributions and thesis structure are talked about.

### 1.1 History of Quadrotors

A quadrotor is a multirotor that is lifted and propelled by four rotors. These aircrafts configuration was seen as possible solution in Vertical Take-Off and Landing (VTOL) and in torque-induced control problems in the early of 20<sup>th</sup> century.

Early in the history of the aviation, the design of the first quadrotors appears with the Gyroplane (Figure 1.1) of Breguet brothers and Prof. Richet in 1907 [21]. A few years later, in 1920 Etienne Oehmichen, created the first quadrotor able to perform a controlled and stable flight lifting a person, the Oehmichen 2. It had four rotors and eight propellers, all driven by a single engine as can be seen in Figure 1.2<sup>1</sup>. Five of the propellers, spinning in the horizontal plane, stabilized the machine laterally. Another propeller was mounted at the nose for steering. The remaining pairs of propellers were for forward propulsion. The aircraft made more than a thousand test flights during the middle 1920s,

---

<sup>1</sup> <http://ingaeronautica.wordpress.com/2010/11/18/oehmichen-2-el-primer-helicoptero-que-logro-elevar-un-hombre/>, August 2013.

establishing the first-ever FAI distance record for helicopters of 360 m (390 yd). It demonstrated the ability to complete a circular course and later, it completed the first 1 kilometer closed-circuit flight by a rotorcraft<sup>2</sup>.

About the same time, Dr. George de Bothezat and Ivan Jerome under a program of the United States Army Air Service, developed the “de Bothezat helicopter” or also called the “Flying Octopus” (Figure 1.3). It had four massive six-bladed rotors that allowed the craft to successfully fly, in fact, was said at the time the first successful helicopter. However, the control difficulties, the lack of power and the necessary favorable wind to achieve forwards flight lead to the cancellation of the project in 1924 [21].

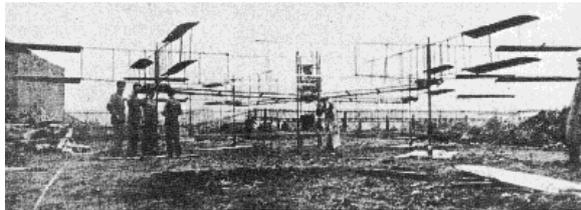


Figure 1.1: Breguet's Gyroplane [20]

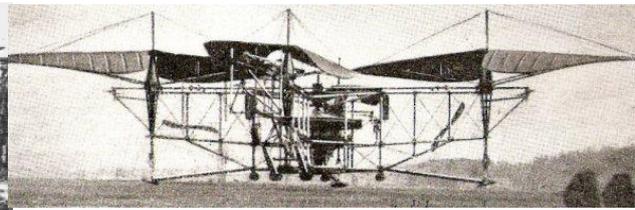


Figure 1.2: Oehmichen 2<sup>1</sup>

All quadrotors and helicopters developed in the 1920's were not capable of flying without the bonus of ground effect lifting [14]. It was not until 1930 that the possibility of flying without the bonus of ground was achieved by the coaxial helicopter of Corradino d'Ascanio (Figure 1.4)<sup>3</sup>. Then, the new models were based on helicopters and quadrotors were forgotten until the late 1950's

In 1956, D.H. Kaplan designed and piloted the “Convertawings Model A Quadrotor” (Figure 1.5)<sup>4</sup>, first capable of a truly controlled and without ground effect [3]. The design featured two engines driving four rotors through a system of “v” belts. No tail rotor was needed and control was obtained by varying the thrust between rotors, eliminated complex cyclic-pitch-control systems typical of standard helicopters<sup>5</sup>. It flew successfully many times in the mid-1950s, proving the quadrotor design and it was also the first four-rotor helicopter to demonstrate successful forward flight.

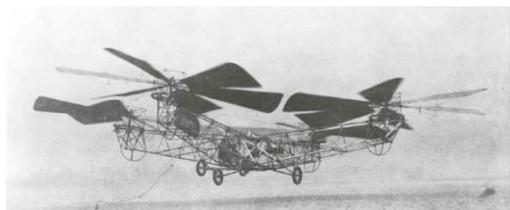


Figure 1.3: Bothezat Helicopter [20]

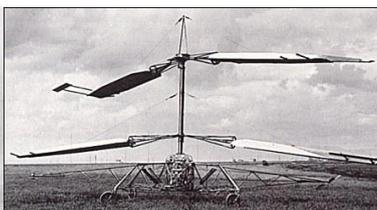


Figure 1.4: D'Ascanio Helicopter<sup>3</sup>

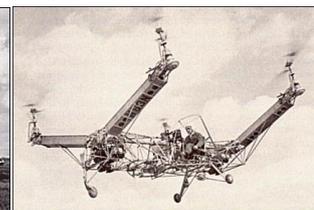


Figure 1.5: Convertawings Model A Quadrotor<sup>4</sup>

Since then more quadrotors were developed for heavy payloads. Other example is the “Curtiss-Wright VZ” (Figure 1.6)<sup>6</sup>, a VTOL designed by the Curtiss-Wright company for the US Army. It was also

<sup>2</sup> <http://www.flightglobal.com/pdfarchive/view/1924/1924%20-%200047.html>, August 2013

<sup>3</sup> [http://www.aviastar.org/helicopters\\_eng/askanio.php](http://www.aviastar.org/helicopters_eng/askanio.php), August 2013

<sup>4</sup> [http://www.aviastar.org/helicopters\\_eng/convertawings.php](http://www.aviastar.org/helicopters_eng/convertawings.php), August 2013

<sup>5</sup> <http://www.flightglobal.com/pdfarchive/view/1956/1956%20-%201564.html>, August 2013

<sup>6</sup> [http://www.diseno-art.com/news\\_content/2012/09/strange-vehicles-curtiss-wright-vz-7/](http://www.diseno-art.com/news_content/2012/09/strange-vehicles-curtiss-wright-vz-7/), August 2013

controlled by changing the thrust of each of the four propellers<sup>7</sup>. Like the Chrysler VZ-6 and the VZ-8 Airgeep, it was to be a "flying jeep".

In respect to tiltrotors, their history starts with George Lehberger. In May 1930, Lehberger registered the first patent of a tiltrotor, but the concept was only developed in 1942 by Focke-Achgelis in World War II, the "Focke-Achgelis Fa.269"(Figure 1.7)<sup>8</sup>, but it never flew [18]. In 1955 "Bell XV-3" (Figure 1.8)<sup>9</sup> became the first tiltrotor to fully accomplish the goal of be able of VTOL and also cruise flight. It was powered by a 450hp radial engine and had a maximum speed of 296Km/h, with 411km of autonomy and a maximum ceiling of 4600 meters. This aircraft was the proof of concept and many others followed it, including tilt-winged models.



Figure 1.6 Curtiss-Wright VZ<sup>6</sup>

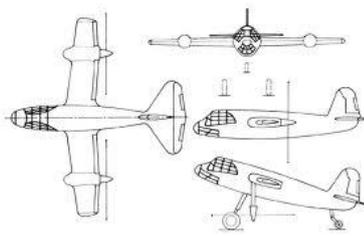


Figure 1.7 Focke-Achgelis  
Fa.269<sup>8</sup>



Figure 1.8 Bell XV-3<sup>9</sup>

An entirely new concept that combined a tiltrotor and a quadrotor was the "Curtiss-Wright X-19" (Figure 1.9)<sup>10</sup>, also idealized by the Curtiss-Wright Corporation. It was a transporter plane (capable to carry up to 500kg of cargo) with two sets of wings and four rotors that could rotate 90° mounted at the tips of both wings, creating the innovative idea of a tilting quadrotor<sup>11</sup>. This aircraft had its first flight in 1963 and had a range of 523Km and a maximum speed of 730Km/h. Two aircrafts were built but the project was cancelled after a crash in 1965.

More models of quadrotors and tiltrotors were developed. In fact, actually the "Bell Boeing Quad TiltRotor" (Figure 1.11)<sup>12</sup> is still under study. It is inspired by "Bell X-22" (Figure 1.10)<sup>13</sup> and "Curtiss-Wright X-19" and is sized to carry until 11,000kg of payload in hover. The vehicle presents the advantage of landing on unimproved sites thanks to its VTOL capabilities. However no new changes have been introduced since the "Curtiss-Wright X-19" until the last decades. This changed with the concept of Unmanned Aerial Vehicles (UAV).

<sup>7</sup> <http://www.flightglobal.com/pdfarchive/view/1960/1960%20-%201367.html>, August 2013

<sup>8</sup> [http://www.aviastar.org/helicopters\\_eng/focke\\_269.php](http://www.aviastar.org/helicopters_eng/focke_269.php), August 2013

<sup>9</sup> [http://www.anigrand.com/AA2013\\_XV-3.htm](http://www.anigrand.com/AA2013_XV-3.htm), August 2013

<sup>10</sup> [http://upload.wikimedia.org/wikipedia/commons/3/38/Curtiss-Wright\\_X-19\\_1963.jpg](http://upload.wikimedia.org/wikipedia/commons/3/38/Curtiss-Wright_X-19_1963.jpg), August 2013

<sup>11</sup> [http://www.unrealaircraft.com/gravity/cw\\_x19a.php](http://www.unrealaircraft.com/gravity/cw_x19a.php), August 2013

<sup>12</sup> [http://www.jetfly.hu/rovatok/jetfly/hirek/hatalmas\\_helikopterek\\_epulhetnek/](http://www.jetfly.hu/rovatok/jetfly/hirek/hatalmas_helikopterek_epulhetnek/), August 2013

<sup>13</sup> [http://www.disen-art.com/encyclopedia/strange\\_vehicles/bell\\_x-22.html](http://www.disen-art.com/encyclopedia/strange_vehicles/bell_x-22.html), August 2013



Figure 1.9: Curtiss-Wright X-19<sup>10</sup>



Figure 1.10: Bell X-22<sup>13</sup>



Figure 1.11: Bell Boeing Quad TiltRotor<sup>12</sup>

These vehicles use electronic control systems and electronic sensors to stabilize the aircraft, either remotely controlled or totally autonomous. Research and investment in these machines, whether in industry or academic sectors, was made possible by the advance of miniaturization, maturing of the technologies, more powerful processors and more reliable and cheaper sensors. The need for aircraft with greater maneuverability and hovering ability has led to the current rise in quadrotor investigation.

Quadrotor UAVs are nowadays under exhaustive investigation and a large number of quadrotors was introduced since 2004, both for military and civil use. Most projects come to life as a result of partnerships between universities and companies. Some significant current programs are the AeroQuad and ArduCopter (Figure 1.12)<sup>14</sup>, which are open-source hardware and software projects based on Arduino for the DIY (Do It Yourself)<sup>15</sup> construction of quadcopters. Another project is the Parrot AR.Drone (Figure 1.13)<sup>16</sup> which is a small radio controlled quadcopter with cameras attached to it built by Parrot SA, designed to be controllable by smartphones or tablet devices.

The present of quadrotors is more alive than never in their history. The progress in this sector is really astonishing, with a grade of potentiality very high. Examples of that are the investigation on swarm of nano quadrotors (Figure 1.14)<sup>17</sup> by the University of Pennsylvania with its formation flight<sup>18</sup> and even performing James Bond theme using musical instrument<sup>19</sup>. Other examples are the interaction between quadrotors and humans<sup>20</sup> via the Kinetic or quadrocopters capable of not only balancing an inverted pendulum, but also of launching it off the vehicle and catching it again<sup>21</sup>, both examples done by the ETH Zurich University.

## 1.2 Context and Motivation

Quadrotors and above all, small quadrotors, are more and more commons in our lives, either in military or civil applications. Their currently main functions are achieved working singly, but the work of

<sup>14</sup> <https://code.google.com/p/arducopter/>, August 2013

<sup>15</sup> <http://diydrones.com/>, August 2013

<sup>16</sup> [http://upload.wikimedia.org/wikipedia/commons/b/b8/Parrot\\_AR.Drone\\_2.0\\_-\\_indoor\\_hull.jpg](http://upload.wikimedia.org/wikipedia/commons/b/b8/Parrot_AR.Drone_2.0_-_indoor_hull.jpg), August 2013

<sup>17</sup> <http://www.gizmag.com/grasp-nano-quadrotor-robots-swarm/21302/pictures#3>, August 2013

<sup>18</sup> <https://www.youtube.com/watch?v=YQIMGV5vtd4>, August 2013

<sup>19</sup> [https://www.youtube.com/watch?feature=player\\_embedded&v=\\_sUeGC-8dyk](https://www.youtube.com/watch?feature=player_embedded&v=_sUeGC-8dyk), August 2013

<sup>20</sup> <https://www.youtube.com/watch?v=A52FqfOi0Ek>, August 2013

<sup>21</sup> <https://www.youtube.com/watch?v=pp89tTDxXuI>, August 2013

the University of Pennsylvania in swarm of nano-quadrotors already mentioned in the previous section, opens some new doors.

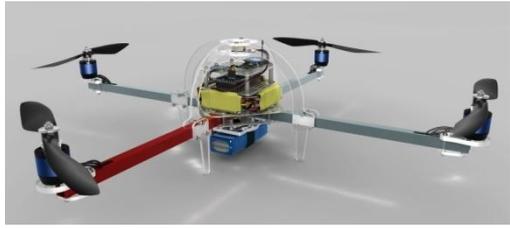


Figure 1.12: ArduCopter<sup>14</sup>



Figure 1.13: Parrot AR.Drone<sup>16</sup>

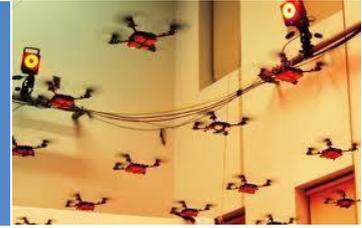


Figure 1.14: Swarm of nano Quadrotors<sup>17</sup>

The applications of quadrotors UAV until now in the military field are surveillance and reconnaissance by military and law enforcement agencies, as well as search and rescue missions in urban scenarios. In the field of civil applications<sup>22</sup>, the largest use of quadcopters has been in the field of aerial imagery. Using on-board cameras, users can stream from the sky to the ground what can be used in crowded scenarios as music festivals or sport events.

A quadcopter UAV solely has limited payload, but a group working together can increase this limit. For this reason the work of the University of Pennsylvania is important, because a swarm quadrotor working together would be used in more application than until now are being used, like is the field of construction.

The main advantages of these aerial vehicles over fixed wing aircrafts are the ability to hover and land in a wide variety of locations. Moreover a quadrotor in relation to a helicopter adds some major advantages considering the same scale<sup>23</sup>:

- Absence of tail rotor, hence making it more energy efficient.
- Simplifies the mechanical complexity, enabling the absence of gearing between motors and rotors.
- A payload increase can be achieved
- Easier to control.

The ALIV concept in comparison with a standard quadrotor, by the substitution of two normal rotors for two tilting ones, adds the advantage of maintaining a payload almost perpendicular to gravity, and independent of its motion or velocity. A priori also it also can contribute to a drag reduction, because in forward motion and lateral motion the surface facing the airflow does not change and it's possible to optimize the coverage.

---

<sup>22</sup> <http://www.popularmechanics.com/technology/military/planes-uavs/armed-quadrotors-are-coming-10720086>, August 2013

<sup>23</sup> <https://sites.google.com/site/quadrotordrone/home/history>, August 2013

### 1.3 Background

As is said previously, the creator of the concept of a small size multirotor with rotors of variable inclination was Eng. Severino Raposo [17], with his ALIV (Figure 1.15), a fully aluminum structure of 1811g. He designed the concept and built it, but did not develop the stabilization software, which was semi-attempted with several approximations by Sergio Costa in his 2008 master's thesis [16]. Costa developed a Linear Quadratic Regulator (LQR) for ALIV which was fully functional in flightgear's flight simulator; however the implementation in the real model was never attempted due to different fails that are going to be explained succinctly below.

The major factor that made Costas's control not able to function in the real ALIV, was that he tried to control the ALIV as a common quadrotor maintaining the direction of rotation of the rotors, and in future work enabling the tilting of the rotors. This issue makes the ALIV unstable in forward motion. Raposo placed correctly the direction of rotation of the rotors according to this new concept, but it had also problems in yaw motion, where it was also unstable. All these problems will be well detailed in the "Performance of a Tilt-Quadrotor" in section 2.3.2

Another important factor for the absence of a working control for the ALIV is due to its asymmetry (see Figure 1.15). Both servos are mounted on the same side of the edge of the swivel arms, moving the center of mass in the x-direction in respect to the place where it should be, in the intersection of the two virtual lines drawn by opposing rotors, like a common quadrotor. With a technique similar to the one employed by Henriques [1], the center of mass was found by Fernandes [14] in the horizontal plane (xy plane in the ABC frame, Figure 2.3) off-centered 7,06mm in the x coordinate and 9,88mm along y.



Figure 1.15: ALIV [14]

The total weight of the ALIV, without avionics, is 1811g and its dimensions are 563mm between motors in the fixed arm and 689mm between swivel arm's motors. The structure of the swivel arm is full on around the rotor, with 336mm in diameter allowing for an up to 330,2mm rotor diameter and leading to 1025mm of total span in the swivel arms axis (y).

Due to symmetry and weight problems of the original ALIV, an upgrade of the concept was projected by Filipe Pedro in his master's Thesis [4]. In his work he planned a new and lighter version of ALIV concept (Figure 1.16). He envisioned a structure built mainly in carbon fiber, also with a fixed arm and two swivel arms, embracing the structure only a semicircle of the rotor.

He study the aerodynamic of the rotor, with and without ducted rotor, concluding that introducing ducts around the rotors is not enough profitable related to the weight added. Pedro also made a Genetic Algorithm (GA) for the estimation of an optimum rotor to his ALIV2. He did a study on the electronic and mechanical components necessary to build the device and finally defined a structural project of the ALIV2. Nonetheless, he never approached the problem of rotor tilting and torque counteract.



Figure 1.16: ALIV2 [14]

Once the preliminary design of an upgrade of the original ALIV was made, Fernandes continued the project with a new version, the ALIV3, based basically on the ALIV2. He studied the concept of the swivel arm, comparing two main structures: the U-arm (Figure 1.17) and the Slim-arm (Figure 1.18). The U-arm is the symmetrical U-shaped and stronger swivel arm alternative, while the slim-arm is the lighter and more simplified version. However after construction and tests of both alternatives, Fernandes concluded that the U-arm was the best option for the ALIV3 [14].



Figure 1.17: ALIV3 U-arm [14]

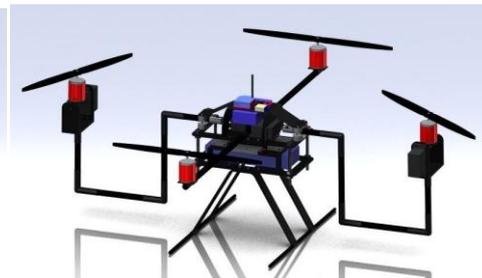


Figure 1.18: LIV3 Slim-Arm [14]

He also designed a new structure performing its corresponding analysis. This new ALIV3 has two main boards connected by screws. Over the first "Central board" rests the "Servo board", the bearings of the swivels arms and also the fixed arm. Over the second "Central board", connected to the landing gear, rests the electronic board and the battery.

Fernandes refined the selection of the components “off the shelf”: Motors, ESC’s, servos, avionics, battery and communication. He also developed an optimum rotor with the algorithm created by Filipe Pedro [4], which has not been built yet.

He also did a drag analysis focused in forward motion considering both covered (Figure 1.20) and uncovered (Figure 1.19) scenarios, performance that experimented a descend of 16,3% in drag with the structure covered. However, for lateral motion or for climbing the drag is heavily increased, 22% in lateral motion and 49% in climbing motion, adding an increase of weight, and therefore a decrease in payload. For that the final ALIV3 is uncovered.

Unlike Pedro’s work, Fernandes did a study of the performance of the device based on equations, noticing the differences with a common quadrotor. This study will be extended in the “Performance of a Tilt-Quadrotor” section of the present thesis.



Figure 1.19: ALIV3 [14]



Figure 1.20: ALIV3 Covered [14]

## 1.4 Objectives and Contributions

The present Master Thesis proposes to analyse the proof of concept of Eng. Severino Raposo tilt-quadrotor. In every quadrotor project there are two major aspects to be accounted for, the platform project and the aircraft’s control. Once the platform project was developed by Fernandes this thesis seek to make all components work, either mechanic or electronic, and progress in the quadrotor’s control.

Before it can fly and perform aggressive maneuvers like a common quadrotor, it is necessary to achieve the stabilization of the device, and this is the main objective of this work. In order to reach the objectives, different steps have to be reached:

- Guarantee a trustworthy ALIV3 platform;
- Study the behavior and obtain the motion model of the ALIV3;
- Identify platform parameters;
- Develop and implement a model of the prototype in Matlab/Simulink;
- Inclusion of data from the real sensors in the model;
- Inclusion of data from the real motors in the model;

The work developed resulted in the following contributions:

- A Matlab/Simulink model of the ALIV3 including identified data (Center of Gravity, mass, inertia, as well as sensors and actuators characteristics);
- A structure to measure the moments of inertia was developed and made available with proper how to use documentation;
- The motors were modeled from experimental data, considering also the propellers and the battery. For this purpose a motor test bed was created.
- A stabilization solution is presented and tested in simulation for the tilt-quadrotor model.

## 1.5 Thesis Structure

In this chapter the main aspects of this project are outlined and a historical overview is presented. The previous versions of the ALIV3's platform and past works have been described and lastly, it have been stated the objectives and contributions of this work.

There are two parts differentiated in this work. The first part are the Chapter 2 and Chapter 3, where the modeling and identification of the tilt-quadrotor is performed. In Chapter 2 the ALIV3 platform and the upgrade of the structure are described. The model of the ALIV3 platform is also completely described. Firstly a theoretical study of the elemental performance of the ALIV concept is presented. Then the kinematics and dynamics equations are presented. The models of the different sensors and actuators are also described and finally, the quadrotor simulator is introduced. In Chapter 3 the mass properties (total mass, Center of Mass, and inertia), actuators and sensors are identified.

The second part is related with the flight attitude control simulation. It is performed along the Chapter 4. Firstly a theoretical description of the LQR controller and the response to the ideal 12 States response is presented. Then, a theoretical description of Kalman Filter is also presented. Because the system is not fully controllable, the system is reduced to a system of 6 states. So, finally is presented the ideal response and the real response with estimation feedback.

Finally, the conclusions and some possible improvements for the structure and for the stabilization of the ALIV3 attitude are drawn and left for future work in Chapter 5.

## Chapter 2

# Tilt-Quadrotor Modeling

This Chapter describes the model of the ALIV3 platform. It has four main sections. The first one describes the ALIV3 platform. The second one presents the general model of the tilt-quadrotor. The system of coordinates is introduced, the performance of the platform is studied and the kinematics and dynamics equations are formulated. The performance is studied around two different motions: leveled motions and rebalancing motion. Then some conclusion about which type of motions and how it can perform these are added.

The third section of this Chapter describes the models of the sensors: Accelerometer, Gyroscopes and Magnetometer. All of them are part of the IMU (Inertial Measurement Unit) shield of the ArduPilot.

And finally, in the fourth section, the model of the actuators is presented: Including both propellers and rotors.

### 2.1 General Modeling Assumptions

This work is centered in the stabilization of the Tilt-Quadrotor. With this objective in mind, several general assumptions are introduced in order to simplify the model:

- In hover, the accelerations of the quadrotor can be neglected.
- The quadrotor is symmetric along the x-axis and y-axis.
- The quadrotor is a rigid body.
- All aerodynamic forces (drag forces) acting on the quadrotor are neglected.
- The magnetic inclination is neglected.
- The rotor flapping effect is neglected.
- There is no slip between the propeller and the rotor of the motor.
- All motors have the same time constant,  $\tau$ .
- All sensor are composing the IMU shield are located at its center

## 2.2 ALIV3 Platform

The ALIV3 built by Fernandes [14] has suffered some modifications and improvements. Some components have been changed and some structural modifications have been done to improve the quadrotor performance. This chapter presents the description of the ALIV3 platform and the upgrade of the structure. The ALIV3 platform consists in a structure with a central core and four arms. Figure 2.1 shows the ALIV3 after the upgrade. A relation of the final electric and electronic components is added.

### 2.2.1 Arms

The four motor and propeller set, are placed in the extreme of the four arms. The ALIV3 has two types of arms: two fixed arms and two swivel arms. The fixed arms accomplish the function of support the motor and propeller set and transmit the loads to the central core. The swivel arms accomplish the same function and in addition they allow the roll and pitch movement of the rotors 2 and 4. The pitch movement of the rotor is carried out by rotating the arm about its own axis, the y-axis. The roll movements of the rotor is carried out by rotating the tube that support the motor. The swivel arms have a U-shaped to harbor this tube.



Figure 2.1: Final upgrade of the ALIV3

The platform has 4 servos that ensure the tilting movement. Two servos are called in Fernandes's work [14] as first-servos. They are placed in the central core and are responsible of the pitch movement. They are directly connected to the tube that runs from the U-shape of the swivel arm towards the central core. The other two servos are called in Fernandes's work as second-servos. They are placed in the tip of the two U-shape arms and are responsible of the roll movement of the rotor.

The weight of the second-servos produces one asymmetry around the y-axis, making that the first-servo have to rebalance this asymmetry all the time. To balance the swivel arm, a counterweight in

lead has been placed on the other side of each U-arm. There are two bearing in the central core that support both swivel arms. They have also been reinforced.

Respect to the fixed arms, they were a tube made in carbon fibre. It had a bad performance to torsion and flexion efforts and it had also problems with the frequencies of resonance around the point of operation of the motors. So it was replaced by another one in aluminium with a square section. Thus, the stiffness has been increased the resonance problems solved. The weight increased by introducing the aluminium beam has been lower than 30g.

## **2.2.2 Central Core**

The central core supports all the efforts of the platform: the lifting forces and the gravity forces. It harbors the battery, the ArduPilot, the Power Distribution Board (PDB), the Electronic Speed Controllers (ESCs) and the landing gear is also attached.

Fernandes projected the APM to be placed into the servo board [14]. However, he did not project enough space to cover the APM with its case. One solution could be place the case (with the APM inside) above the electronic board. But this solution increase the distance between the two central boards, and then the weight of the heavy screws M6 increases, the electromagnetic interference with the battery also increases and it can involve some error in attitude estimation.

The solution taken were to modify the structure. The idea was to place the APM's case above the servos inside the servo board. For this purpose was necessary to retire the bearing board and cut the upper plate of the servo board. Then the servo board and the bearing supports were going to accuse a lack of stiffness. So, two carbon tubes inside the servo board and two plates at each side of the servo board, which link it to the bearing support, were employed to increase the stiffness.

The material for the four plates was used from the electronic board. Since the PDB was going to be attached from the aluminum beam and the battery was going to lie upon the down central board, the electronic board was not going to accomplish any function and so this material has been recycled. Then the APM lies upon the carbon tubes inside the servo board. The distance from the aluminum beam to the APM is leveled by two aluminum tubes.

To level the two central boards, four plastic tubes have been incorporated covering the screws M6. Also two rubber washers have been placed for each screw to damp the vibrations.

This final upgrade of the structure achieves a larger compactness, a higher stiffness and a weight reduction from the screws M6 and the plastic tubes. It also contributes to better sensor measurements decreasing the noise.

## 2.2.3 Electrical and Electronic Components

The model and operating principles some components as the motors and the speed controllers are going to be explained in following sections, but here a relation of the overall components is added. The connection between them is included in the Appendix B. The list of the different components of the final version of the ALIV3 is:

- 4 Motors *Turnigy 2217 16turn 1050kv 23A Outrunner*<sup>24</sup>
- 4 ESCs *RC-plus SKYSPORT 30 30A – 2-3S – BEC*<sup>25</sup>
- 1 Battery *ZIPPY Flightmax 5000mAh 3S1P 20C*<sup>26</sup>
- 1 PDB *Arducopter Power Distribution Board*<sup>27</sup>
- 1 APM1: *ArduPilot Atmega2560*<sup>28</sup> and *ArduPilot Mega IMU shield*<sup>29</sup>
- 4 Servos *Futaba S3003 Standard*<sup>30</sup>

## 2.3 Tilt Quadrotor Model

### 2.3.1 Coordinate System

For an accurate description of the ALIV's motion and its attitude, two coordinated frames must be defined: a reference inertial frame (Figure 2.2) and a body frame (Figure 2.3). The chosen reference frame is the North East Down (NED) frame, also known as local tangent plane (to Earth's surface) and is centered on O, here assumed as inertial.

The body frame is usually known in aeronautics as the Aircraft-Body-Centered (ABC) frame. The ABC frame (see Figure 2.3) is centered in the ALIV's center of mass ( $O_c$ ) and is identified by the (x; y; z) coordinated axis. The x-axis is associated to the fixed arm, where the rotor 1 is located in the positive direction of this axis; the y-axis is associated to the swivel arm, where the rotor 4 is located in the positive direction; and z axis is directed downwards.

---

<sup>24</sup> [http://www.hobbyking.com/hobbyking/store/\\_\\_5690\\_\\_Turnigy\\_2217\\_16turn\\_1050kv\\_23A\\_Outrunner.html](http://www.hobbyking.com/hobbyking/store/__5690__Turnigy_2217_16turn_1050kv_23A_Outrunner.html)

<sup>25</sup> [http://www.e-](http://www.e-rc.be/index.php?option=com_virtuemart&view=productdetails&virtuemart_product_id=4312&virtuemart_category_id=172)

[rc.be/index.php?option=com\\_virtuemart&view=productdetails&virtuemart\\_product\\_id=4312&virtuemart\\_category\\_id=172](http://www.e-rc.be/index.php?option=com_virtuemart&view=productdetails&virtuemart_product_id=4312&virtuemart_category_id=172)

<sup>26</sup> [http://www.hobbyking.com/hobbyking/store/\\_\\_8579\\_\\_ZIPPY\\_Flightmax\\_5000mAh\\_3S1P\\_20C.html](http://www.hobbyking.com/hobbyking/store/__8579__ZIPPY_Flightmax_5000mAh_3S1P_20C.html)

<sup>27</sup> [http://store.jdrones.com/Power\\_Distribution\\_Quad\\_RevD\\_p/quadpdpcb1.htm](http://store.jdrones.com/Power_Distribution_Quad_RevD_p/quadpdpcb1.htm)

<sup>28</sup> <http://store.3drobotics.com/products/apm-1-arduino-compatible-uav-controller-w-atmega2560-kit>

<sup>29</sup> <https://www.sparkfun.com/products/retired/10952>

<sup>30</sup> [http://www.servocity.com/html/s3003\\_servo\\_standard.html](http://www.servocity.com/html/s3003_servo_standard.html)

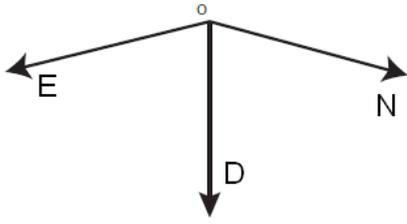


Figure 2.2: NED frame

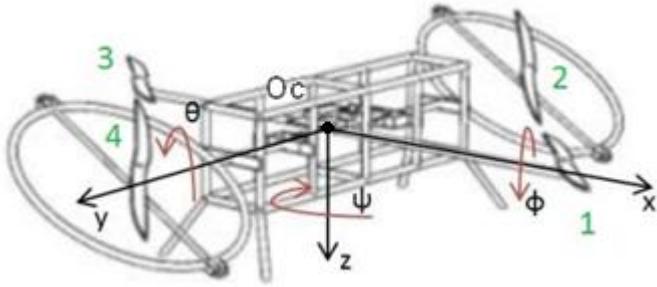


Figure 2.3: ABC frame ALIV1 [14]

Vectors expressed in the inertial frame are marked with the superscript  $I$  and vectors expressed in the body fixed frame have the superscript  $B$ .

The position  $P$  of the quadrotor is described in the NED frame and corresponds to the displacement of  $O_c$  relative to  $O$ :

$$P^I = [X; Y; Z]^T \quad (2.1)$$

The rotation of the ABC frame relatively to the NED frame defines the attitude of the aircraft. It can be described using the three Euler angles. In aeronautic literature the three angles correspond to the rolling motion ( $\phi$ ), which corresponds to a rotation about the x-axis; the pitching motion ( $\theta$ ), which corresponds to a rotation about the y-axis; and the yaw motion ( $\psi$ ), which corresponds to a rotation about the z-axis.:

$$\Phi = [\phi; \theta; \psi]^T \quad (2.2)$$

### 2.3.2 Performance of a Tilt-Quadrotor

In this section the principal movements of the ALIV3 are going to be studied, which are divided in levelled motions and in rebalanced motions. But firstly, the implications of the ALIV concept are going to be explained.

The introduction of two additional degrees of freedom in two of its rotors adds new possibilities for the ALIV's motion, theoretically making it faster and more stable than standard quadrotors, especially regarding the payload [17]. Unlike regular quadrotors, by tilting its rotors, the ALIV can maintain its center core levelled and at the same time perform translational motions. But the possibility of tilting two of its rotors also adds some problems in performance in respect to a common quadrotor.

The presence of the two tilting rotors, with 2 tilting angles each rotor, introduces a substantial change in respect to a common quadrotor. While in a common quadrotor the pair of rotors (1,3) rotates in a

clockwise direction and the pair (2,4) rotates in a counter-clockwise direction in order to annul the resulting moment, in the ALIV concept this balance is only possible in a hover situation.

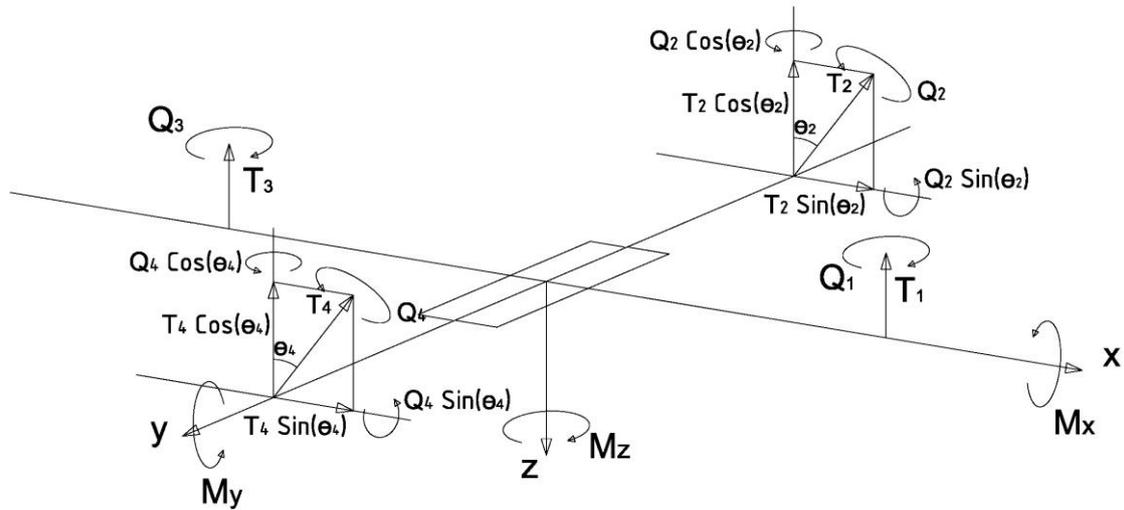


Figure 2.4: Standard Forward Motion

In forward motion, as it is shown in Figure 2.4, if rotors 2 and 4 rotate in the same direction, the moment  $Q_2 \sin(\theta_2)$  created in the x-axis by rotor 2 is not counteracted by the moment  $Q_4 \sin(\theta_4)$  created in the x-axis by rotor 4, in fact it is increased. So, it is compulsory that rotors 2 and 4 rotate in opposite directions in order to annul the resulting moment. To cancel the moment in the z-axis, rotors 1 and 3 also have to rotate in opposite directions.

For the lateral motion a similar question occurs. If rotors 2 and 4 rotate in the same direction, the moment  $Q_2 \sin(\phi_2)$  created in the y-axis by rotor 2 is not counteracted by the moment  $Q_4 \sin(\phi_4)$  created in the y-axis by rotor 4. So, it is also compulsory that rotors 2 and 4 rotate in opposite directions in order to cancel the resulting moment.

Resuming, the pair of rotors (2,3) must rotate in clockwise direction and the pair of rotors (1,4) must rotate in a counter-clockwise direction in order to cancel the resulting moments in the x-axis, y-axis, and z-axis in all possible movements of the ALIV concept, as is show in Figure 2.5 in a description of a forward motion.

As mentioned in section 1.4, the major problem in Costas's control was to control the ALIV as a common quadrotor maintaining the direction of rotation of the rotors. This issue makes the ALIV unstable in forward motion as is already explained. On the other hand, Raposo placed correctly the direction of rotation of the rotors according to this new concept, but it had also problems in yaw motion, where it was also unstable.

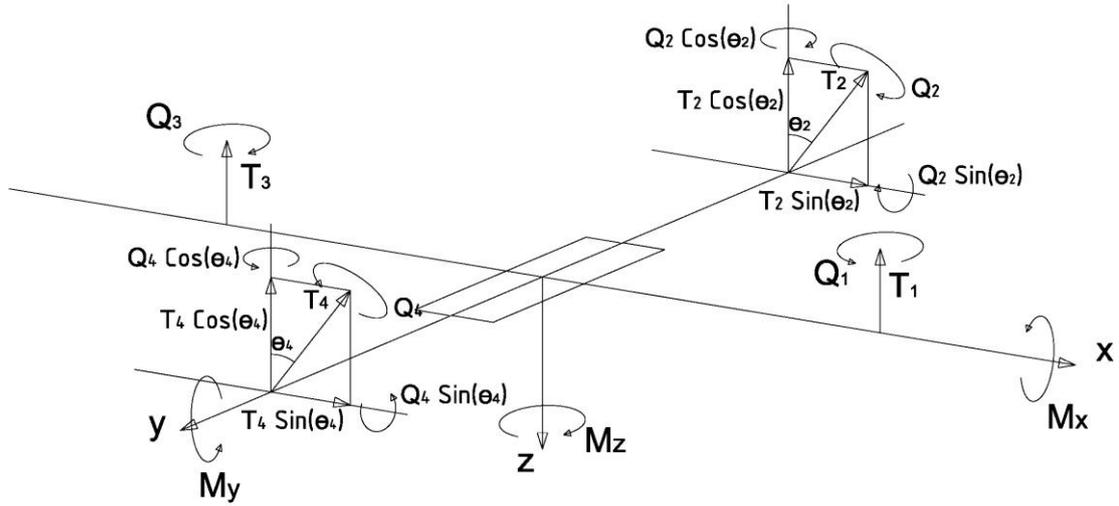


Figure 2.5: forward motion ALIV platform

Both Costa and Raposo had problems when the motors were tilted because none of them considered the torque from the motors in x-axis and y-axis when the motors are tilted.

A further analysis of the principal movements is going to be presented in the sequel.

### **2.3.2.1 Leveled Motions**

This section presents the study of the performance of the ALIV concept when it is levelled, i.e. when it is hovering, climbing, and in forward, lateral and yaw motions.

#### **Hovering**

The equations that govern the hovering motion are the equations of equilibrium plus the equations of power:

$$\sum F_x = 0 \Rightarrow 0 = 0 \quad (2.3)$$

$$\sum F_y = 0 \Rightarrow 0 = 0 \quad (2.4)$$

$$\sum F_z = 0 \Rightarrow T_1 + T_2 + T_3 + T_4 = mg_0 \quad (2.5)$$

$$\sum M_x = 0 \Rightarrow T_2 \cdot \frac{d}{2} = T_4 \cdot \frac{d}{2} \quad (2.6)$$

$$\sum M_y = 0 \Rightarrow T_1 \cdot \frac{d}{2} = T_3 \cdot \frac{d}{2} \quad (2.7)$$

$$\sum M_z = 0 \Rightarrow -Q_1 + Q_2 + Q_3 - Q_4 = 0 \quad (2.8)$$

$$Q_i = R_e T_i \quad (2.9)$$

Where  $T_i$  is the thrust force produced by the rotor  $i$ ,  $m$  is the mass of the Tilt-quadrotor,  $g_0$  is the gravity acceleration,  $d$  is the distance between rotors and the center of gravity (CG),  $Q_i$  is the moment created by the rotor  $i$ , and finally  $R_e = r_r \frac{C_P}{C_T}$  being  $r_r$  the radius of the rotor,  $C_P$  the coefficient of power and  $C_T$  the coefficient of thrust.

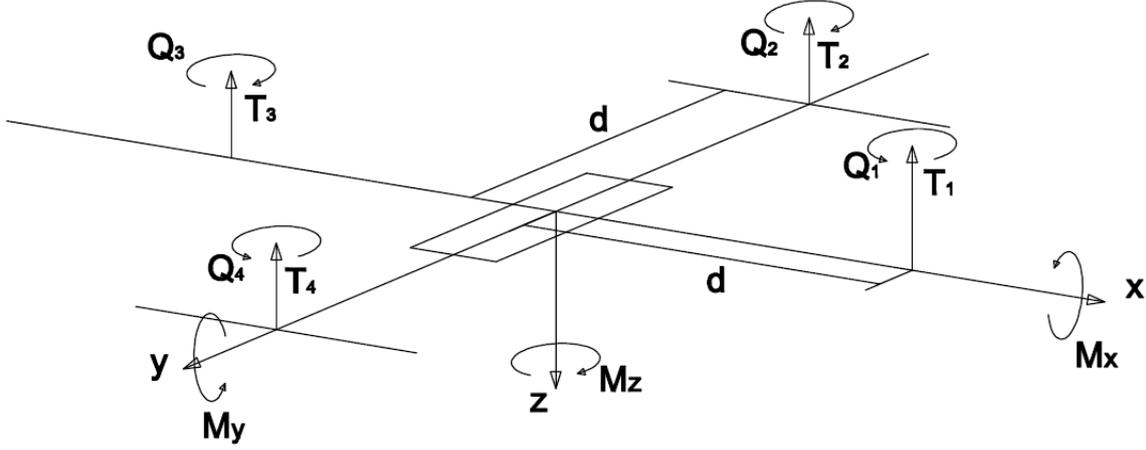


Figure 2.6: Hovering

This system has 4 unknown quantities and 3 equations linearly independent, so it has infinite solutions. One possible solution is:

$$T_1 = T_2 = T_3 = T_4 = \bar{T} \quad (2.10)$$

Where  $\bar{T} = \frac{mg_0}{4}$ .

### Climbing

As Figure 2.7 shows, to increase (or decrease) the altitude of the ALIV3 it is necessary to increase (or decrease) the power of the four rotors. The equations that govern the problem of Climb are the equations of equilibrium plus the equations of power:

$$\sum F_x = 0 \Rightarrow 0 = 0 \quad (2.11)$$

$$\sum F_y = 0 \Rightarrow 0 = 0 \quad (2.12)$$

$$\sum F_z = 0 \Rightarrow T_1 + T_2 + T_3 + T_4 = mg_0 - \dot{W} + D_z(W) \quad (2.13)$$

$$\sum M_x = 0 \Rightarrow T_2 \cdot \frac{d}{2} = T_4 \cdot \frac{d}{2} \quad (2.14)$$

$$\sum M_z = 0 \Rightarrow -Q_1 + Q_2 + Q_3 - Q_4 = 0 \quad (2.15)$$

$$Q_i = R_e T_i \quad (2.16)$$

Where  $\dot{W}$  is the acceleration in the z-axis,  $W$  is the velocity in the z-axis and  $D_z$  is the drag force in the z-axis. The sign minus before  $\dot{W}$  in equation 2.13 is due to considering  $z$  positive downwards.

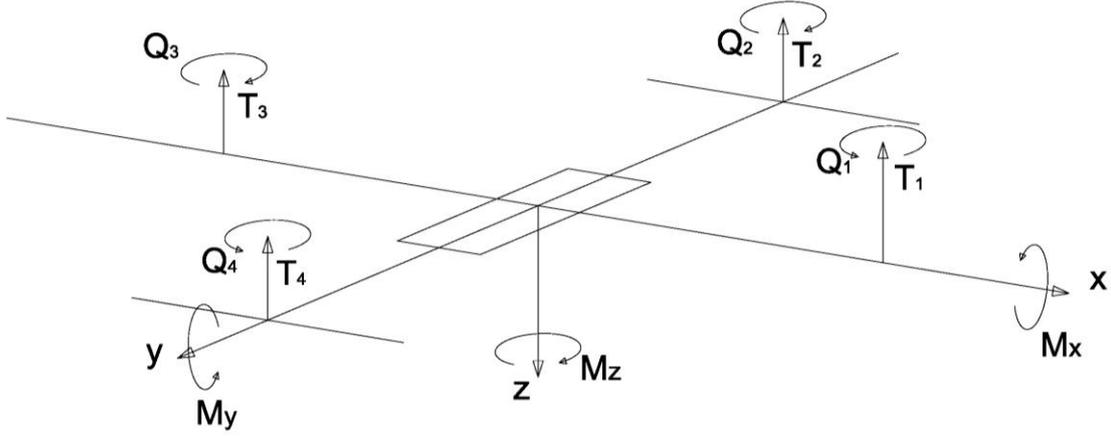


Figure 2.7: Climbing

This system has 5 unknown quantities:  $T_1, T_2, T_3, T_4, \ddot{z}$  and 3 equations linearly independent, so it has infinite solutions. One possible solution is:

$$T_1 = T_2 = T_3 = T_4 = \frac{m g_0 - \dot{W} + D_z(W)}{4} \quad (2.17)$$

### Forward Motion

As Figure 2.8 shows, for the ALIV to move forward (or backward) it is necessary to tilt rotors 2 and 4 with a pitching angle ( $\theta_i$ ) and increase the thrust of the rotors because when the thrust vector is tilted, the projection about the z axis is smaller. The respective quantities are related through the equations that govern the problem of forward motion, which are the equations of equilibrium and the equations of power:

$$\sum F_x = 0 \Rightarrow T_2 \sin \theta_2 + T_4 \sin(\theta_4) = m\dot{U} + D_x(U) \quad (2.18)$$

$$\sum F_y = 0 \Rightarrow 0 = 0 \quad (2.19)$$

$$\sum F_z = 0 \Rightarrow T_1 + T_2 \cos \theta_2 + T_3 + T_4 \cos \theta_4 = m g_0 \quad (2.20)$$

$$\sum M_x = 0 \Rightarrow (T_2 \cdot \cos \theta_2) \frac{d}{2} - Q_2 \sin \theta_2 - (T_4 \cos \theta_4) \frac{d}{2} + (Q_4 \sin \theta_4) = 0 \quad (2.21)$$

$$\sum M_y = 0 \Rightarrow T_1 \cdot \frac{d}{2} = T_3 \cdot \frac{d}{2} \quad (2.22)$$

$$\sum M_z = 0 \Rightarrow -Q_1 + Q_2 \cos \theta_2 + Q_3 - Q_4 \cos \theta_4 + (T_2 \sin \theta_2) \frac{d}{2} - (T_4 \sin \theta_4) \frac{d}{2} = 0 \quad (2.23)$$

$$Q_i = R_e T_i \quad (2.24)$$

Where  $\dot{U}$  is the acceleration in the x-axis,  $U$  is the velocity in the x-axis and  $D_x$  is the drag force in the x-axis.



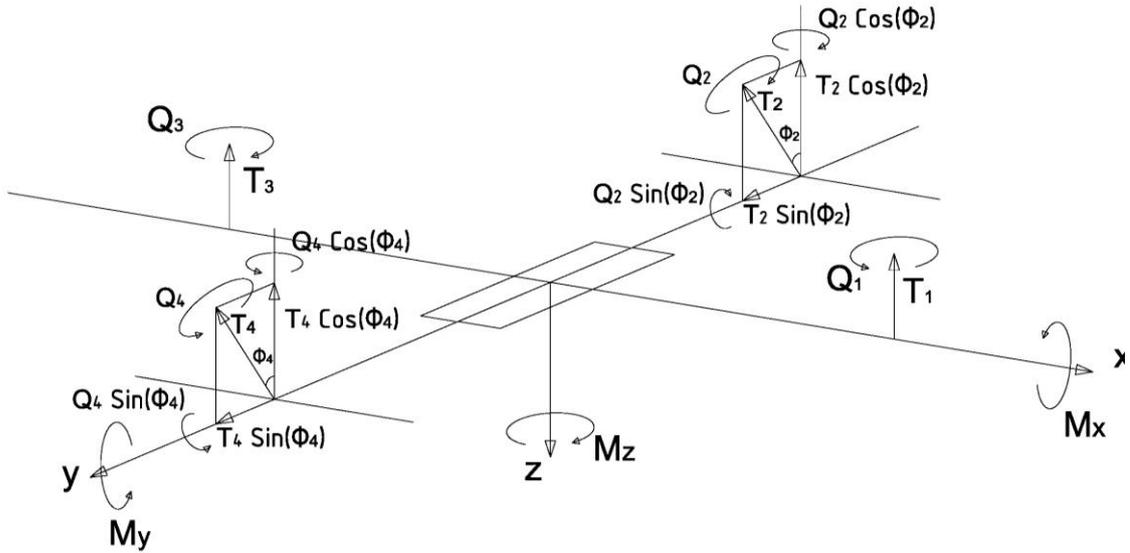


Figure 2.9: Lateral Motion

This system has 7 unknown quantities:  $T_1, T_2, T_3, T_4, \dot{y}, \phi_2, \phi_4$  and 5 equations linearly independent, so it has infinite solutions. Assuming the symmetric problem, i.e.  $\phi_2 = \phi_4$ , it is obtained that  $T_1 = T_3$   $T_2 = T_4$ . So, one possible solution is:

$$T_2 = \frac{m\dot{V} + D_y(V)}{2 \sin \phi_2} \quad (2.34)$$

$$T_1 = \frac{mg_0}{2} - \frac{m\dot{V} + D_y(V)}{2 \tan \phi_2} \quad (2.35)$$

## Yaw Motion

As Figure 2.10 shows, for the ALIV3 to perform a yaw movement it is necessary to tilt the rotors 2 and 4 by a pitching angle ( $\theta_i$ ) and increase the thrust of the rotors. In this action the rotor 4 tilts in the opposite direction of the rotor 2.  $\theta_2$  and  $\theta_4$  are considered positive as is represented in the figure to perform a positive Yaw according to the direction of z-axis. The respective quantities are related through the equations that govern the problem of yaw motion, which are the equations of equilibrium and the equations of power:

$$\sum F_x = 0 \Rightarrow T_2 \sin \theta_2 - T_4 \sin(\theta_4) = 0 \quad (2.36)$$

$$\sum F_y = 0 \Rightarrow 0 = 0 \quad (2.37)$$

$$\sum F_z = 0 \Rightarrow T_1 + T_2 \cos \theta_2 + T_3 + T_4 \cos \theta_4 = mg_0 \quad (2.38)$$

$$\sum M_x = 0 \Rightarrow -Q_2 \sin \theta_2 - Q_4 \sin \theta_4 + (T_2 \cos \theta_2) \frac{d}{2} - (T_4 \cos \theta_4) \frac{d}{2} = 0 \quad (2.39)$$

$$\sum M_y = 0 \Rightarrow T_1 \frac{d}{2} = T_3 \frac{d}{2} \quad (2.40)$$

$$\sum M_z = 0 \Rightarrow -Q_1 + Q_2 \cos \theta_2 + Q_3 - Q_4 \cos \theta_4 + (T_2 \sin \theta_2) \frac{d}{2} + (T_4 \sin \theta_4) \frac{d}{2} = I_z \ddot{\psi} \quad (2.41)$$

$$Q_i = R_e T_i \quad (2.42)$$

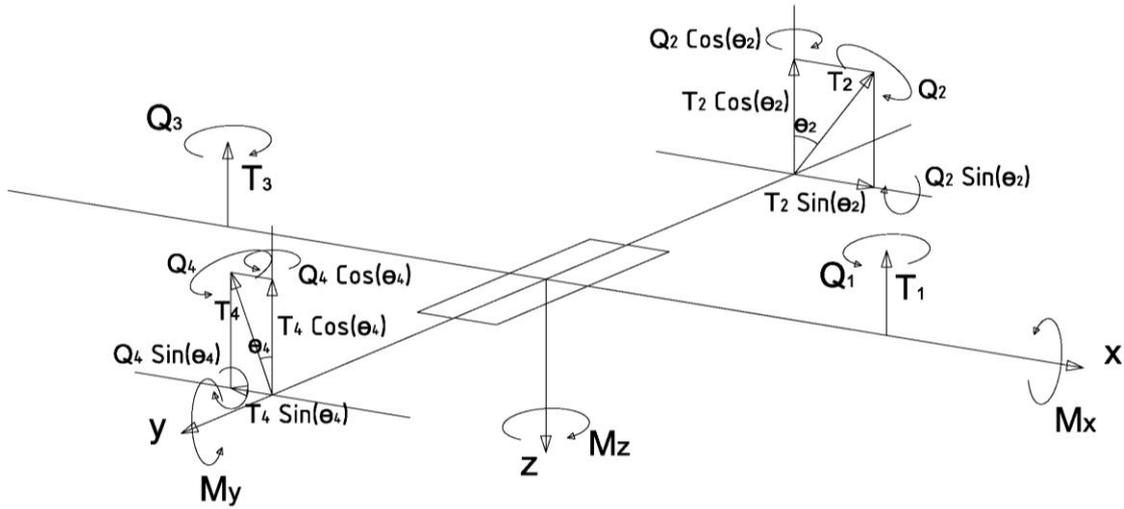


Figure 2.10: Yaw Motion

This system has 7 unknown quantities:  $T_1, T_2, T_3, T_4, \ddot{\psi}, \theta_2, \theta_4$  and 5 equations linearly independent, so it has infinite solutions. From (2.40) it is known that:

$$T_1 = T_3 \quad (2.43)$$

Then, one possible solution is solving the system in function of  $\ddot{\psi}$  and  $\theta_4$ . Then, the result is:

$$\tan \theta_2 = \frac{d \tan \theta_4}{4R_e \tan \theta_4 + d} \quad (2.44)$$

$$T_2 = \frac{I_z \ddot{\psi}}{(R_e - R_e \frac{d}{4R_e \tan \theta_4 + d} + \frac{d \tan \theta_4}{4R_e \tan \theta_4 + d} d)} \quad (2.45)$$

$$T_4 = T_2 \frac{\sin \theta_2}{\sin \theta_4} \quad (2.46)$$

$$T_1 = \frac{1}{2} (mg - T_2 \cos \theta_2 - T_4 \cos \theta_4) \quad (2.47)$$

Another idea to perform the yaw motion (Figure 2.11) could be to vary the moments of the different rotors as a common quadrotor does. However, here, due to the change on the rotors rotation, this way of performing is impossible because the  $\ddot{\psi}$  obtained is null:

$$\sum F_x = 0 \Rightarrow 0 = 0 \quad (2.48)$$

$$\sum F_y = 0 \Rightarrow 0 = 0 \quad (2.49)$$

$$\sum F_z = 0 \Rightarrow T_1 + T_2 + T_3 + T_4 = mg_0 \quad (2.50)$$

$$\sum M_x = 0 \Rightarrow T_2 \cdot \frac{d}{2} = T_4 \cdot \frac{d}{2} \quad (2.51)$$

$$\sum M_y = 0 \Rightarrow T_1 \cdot \frac{d}{2} = T_3 \cdot \frac{d}{2} \quad (2.52)$$

$$\sum M_z = 0 \Rightarrow -Q_1 + Q_2 + Q_3 - Q_4 = I_z \ddot{\psi} \quad (2.53)$$

$$Q_i = R_e T_i \quad (2.54)$$

In fact, if we introduce equations (2.51), (2.52) and (2.54) into equation (2.53), the  $\ddot{\psi}$  obtained would be null and no yaw motion would take place.

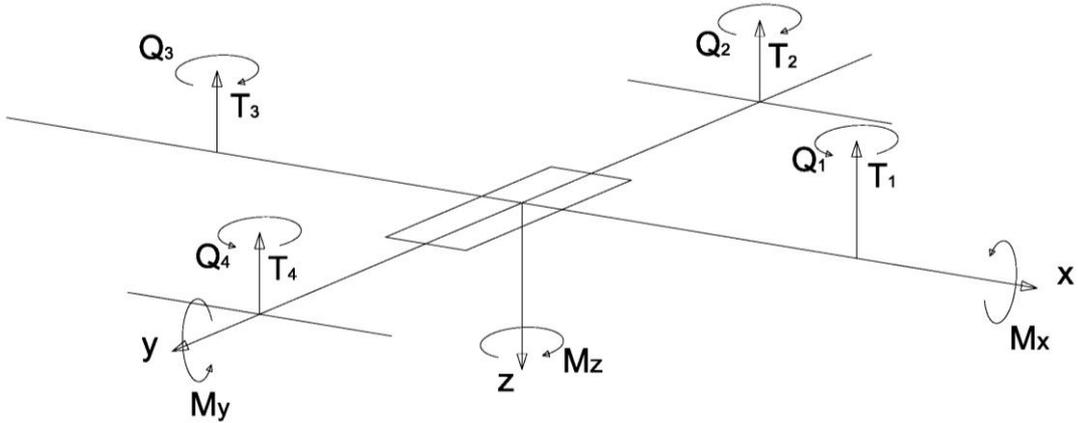


Figure 2.11: Standard Yaw Motion

### **2.3.2.2 Rebalancing Motions**

In this section the performance of the ALIV is going to be studied when it is not levelled. When it suffers a perturbation, it changes its attitude in relation to the equilibrium position with a  $\phi \neq 0$  or/ and  $\theta \neq 0$ . So the goal is to level the quadrotor, taking it to the original position.

Different cases are going to be studied in order to analyse what happens when some rotor is tilted and the ALIV is not levelled. First the roll rebalance is going to be analysed, then the pitch rebalance and finally the overall rebalance.

### **Roll Rebalance**

In this section the roll rebalance is going to be studied for different tilted rotor angles. Figure 2.12 and the followings equations are refer to the case in which rotors 2 and 4 are tilted in their own degrees of freedom. From this picture the other cases which are going to be studied can be perceived.

$$\sum F_x = 0 \Rightarrow T_2 \cos(\phi_2 - \phi) \sin \theta_2 - T_4 \cos(\phi_4 - \phi) \sin \theta_4 = 0 \quad (2.55)$$

$$\sum F_y = 0 \Rightarrow T_1 \sin \phi - T_2 \sin(\phi_2 - \phi) \cos \theta_2 + T_3 \sin \phi - T_4 \sin(\phi_4 - \phi) \cos \theta_4 = 0 \quad (2.56)$$

$$\sum F_z = 0 \Rightarrow T_1 \cos \phi + T_2 \cos(\phi_2 - \phi) \cos \theta_2 + T_3 \cos \phi + T_4 \cos(\phi_4 - \phi) \cos \theta_4 = mg_0 \quad (2.57)$$

$$\begin{aligned} \sum M_x = 0 \Rightarrow & T_2 \cos(\phi_2 - \phi) \cos \theta_2 \frac{d}{2} \cos \phi - T_4 \cos(\phi_4 - \phi) \cos \theta_4 \frac{d}{2} \cos \phi - \\ & - Q_2 \cos(\phi_2 - \phi) \sin \theta_2 - Q_4 \cos(\phi_4 - \phi) \sin \theta_4 = -I_x \ddot{\phi} \end{aligned} \quad (2.58)$$

$$\begin{aligned} \sum M_y = 0 \Rightarrow T_1 \cos \phi \frac{b}{2} - T_3 \cos \phi \frac{d}{2} + Q_1 \sin \phi + Q_2 \sin(\phi_2 - \phi) \cos \theta_2 - \\ - Q_3 \sin \phi - Q_4 \sin(\phi_4 - \phi) \cos \theta_4 = 0 \end{aligned} \quad (2.59)$$

$$\begin{aligned} \sum M_z = 0 \Rightarrow -Q_1 \cos \phi + Q_2 \cos(\phi_2 - \phi) \cos \theta_2 + Q_3 \cos \phi - Q_4 \cos(\phi_4 - \phi) \cos \theta_4 + \\ + T_1 \sin \phi \frac{d}{2} - T_3 \sin \phi \frac{d}{2} + T_2 \cos(\phi_2 - \phi) \sin \theta_2 \frac{d}{2} \cos \phi \\ + T_4 \cos(\phi_4 - \phi) \sin \theta_4 \frac{d}{2} \cos \phi = 0 \end{aligned} \quad (2.60)$$

$$Q_i = R_e T_i \quad (2.61)$$

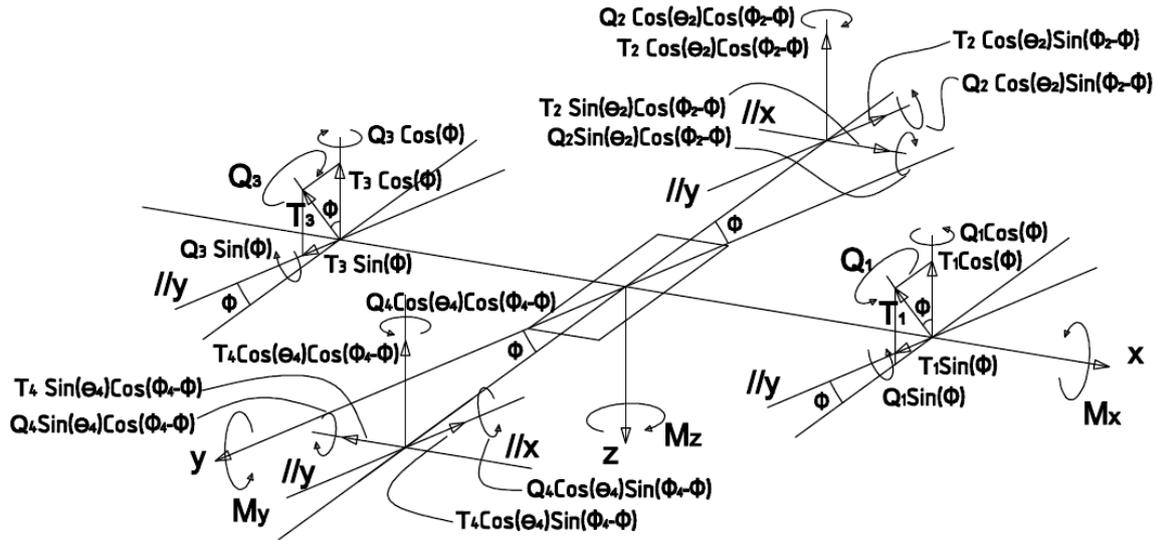


Figure 2.12: Roll Rebalance

### A) Without tilting any angle

Solving the system of equations (2.55)-(2.61) for the nontilting case ( $\phi_2 = 0$ ,  $\phi_4 = 0$ ,  $\theta_2 = 0$ ,  $\theta_4 = 0$ ), it is obtained that the roll acceleration  $\ddot{\phi}$  is null. But it is also obtained that  $\sin^2 \theta + \cos^2 \theta = 0$ , which leads to an incompatible system. If a moment in the y-axis or in the z-axis is introduced, i.e. introduce a  $I_y \ddot{\theta}$  in equation 2.59 or a  $I_z \ddot{\psi}$  in equation 2.60, the system becomes compatible. This means that this performance is impossible unless a moment appears in the y-axis or z-axis.

### B) Tilting $\phi_2, \phi_4 > \phi$

Considering the system of equations (2.55)-(2.61), the implication of tilting  $\phi_2$  and  $\phi_4$  leads to a system with 7 unknown quantities ( $T_1, T_2, T_3, T_4, \phi_2, \phi_4, \ddot{\phi}$ ) and 5 equations linearly independent, so it is possible to solve it making two assumptions. Considering two of the following assumptions:  $T_1 = T_3$ ,  $\phi_2 = \phi_4$  and  $T_2 = T_4$ , the final result is the same, a null roll acceleration,  $\ddot{\phi} = 0$ .

This means that the ALIV platform is completely stabilized, i.e. is in equilibrium, but it does not return to the leveled position.

### **C) Tilting $\phi_2, \phi_4 > \phi$ and $\theta_4$**

Considering the system of equations (2.55)-(2.61) tilting  $\phi_2, \phi_4$  and  $\theta_4$  leads to a system with 9 unknown quantities,  $T_1, T_2, T_3, T_4, \phi_2, \phi_4, \theta_4, \ddot{x}, \ddot{\phi}$ , and 6 equations linearly independent, so it is possible to solve it making three assumptions. There are three cases that are going to be studied in function of the different assumptions.

- a) If it is considered that  $\phi_2 = \phi_4$  and  $T_1 = T_3$ . This case becomes the case presented before in section B. The roll acceleration is null,  $\ddot{\phi} = 0$ , and the ALIV platform is completely stabilized, but it does not return to the leveled position.
- b) If it is considered that  $T_1 = T_3$  and  $T_2 = \bar{T}$ , the system has a solution with a nonnull roll acceleration,  $\ddot{\phi} \neq 0$  and then the rebalance motion occurs, together with a displacement on the xy-plane.
- c) If it is considered that  $\phi_2 = \phi_4$  and  $T_2 = \bar{T}$  the system also has a solution with a nonnull roll acceleration,  $\ddot{\phi} \neq 0$ . The rebalance motion occurs, together with a displacement on the xy-plane.

### **D) Tilting $\phi_2, \phi_4 > \phi$ and $\theta_2, \theta_4$**

Considering the system of equations (2.55)-(2.61), tilting  $\phi_2, \phi_4, \theta_2$  and  $\theta_4$  leads to a system with 9 unknown quantities,  $T_1, T_2, T_3, T_4, \phi_2, \phi_4, \theta_2, \theta_4, \ddot{\phi}$ , and 6 equations linearly independent, so it is possible to solve it making three assumptions. There are four interesting cases that are going to be studied in function of the different assumptions.

- i) If it is considered that  $\phi_2 = \phi_4$  and  $T_1 = T_3$ . This case becomes the section B, where  $\theta_2 = \theta_4 = 0$  and so the roll acceleration is null,  $\ddot{\phi} = 0$ . Then the ALIV platform is completely stabilized in a certain  $\phi_2 = \phi_4$  but it does not return to the original position.
- ii) If it is considered that  $T_1 = T_3$  and  $\theta_2 = \theta_4$ , this case becomes the section B where  $\theta_2 = \theta_4 = 0$ , and so the roll acceleration is null,  $\ddot{\phi} = 0$ . The ALIV platform is completely stabilized but it does not return to the original position.
- iii) If it is considered that  $T_1 = T_3$  and  $T_2 = \bar{T}$ , the system has a solution where the roll acceleration is nonnull,  $\ddot{\phi} \neq 0$ , and then the rebalance motion takes place. Also it has not a displacement on the x,y-plane.
- iv) If it is considered that  $\phi_2 = \phi_4$  and  $T_2 = \bar{T}$ , the system has a solution where the roll acceleration is nonnull,  $\ddot{\phi} \neq 0$ , and then the rebalance motion take place. However it suffers a displacement on the x,y-plane.

## Pitch Rebalance

This section analyses the pitch rebalance when the rotors are not tilted and when the two tilting rotors are tilted in the two possible angles each one.

### A) Without tilting any angle

Figure 2.13 shows the case of pitch rebalance without tilting any rotors. Again, the equations that describe the problem are the equations of equilibrium and power:

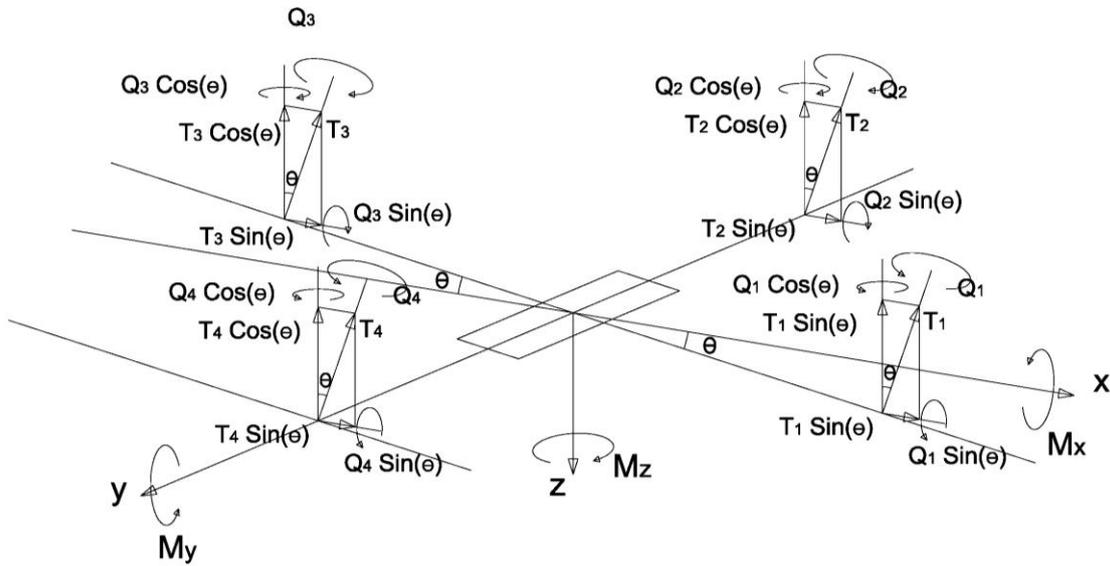


Figure 2.13: Pitch Rebalance without tilting

$$\sum F_x = 0 \Rightarrow (T_1 + T_2 + T_3 + T_4) \sin \theta = m\dot{U} + D_x(U) \quad (2.62)$$

$$\sum F_y = 0 \Rightarrow 0 = 0 \quad (2.63)$$

$$\sum F_z = 0 \Rightarrow (T_1 + T_2 + T_3 + T_4) \cos \theta = mg_0 \quad (2.64)$$

$$\sum M_x = 0 \Rightarrow (T_2 - T_4) \cos \theta \frac{d}{2} + (Q_1 - Q_2 - Q_3 + Q_4) \sin \theta = 0 \quad (2.65)$$

$$\sum M_y = 0 \Rightarrow T_1 \frac{d}{2} \cos^2 \theta \frac{d}{2} - T_3 \cos^2 \theta \frac{d}{2} = I_y \ddot{\theta} \quad (2.66)$$

$$\sum M_z = 0 \Rightarrow (-Q_1 + Q_2 + Q_3 - Q_4) \cos \theta + (T_2 - T_4) \sin \theta \frac{d}{2} = 0 \quad (2.67)$$

$$Q_i = R_\theta T_i \quad (2.68)$$

Like in section A of the Roll Rebalance, dividing (2.65) by (2.67) we obtain:

$$\frac{(Q_1 - Q_2 - Q_3 + Q_4) \sin \theta}{(-Q_1 + Q_2 + Q_3 - Q_4) \cos \theta} = \frac{(T_4 - T_2) \cos \theta \frac{d}{2}}{(T_2 - T_4) \sin \theta \frac{d}{2}} \Rightarrow \sin^2 \theta + \cos^2 \theta = 0 \quad (2.69)$$

This means this motion is not possible for the ALIV platform without introducing a moment in the x or z-axes. It has also a displacement in the xy- plane.

### **B) Tilting $\theta_2, \theta_4 > \theta$ and $\phi_2, \phi_4$**

Regarding Figure 2.14,  $\theta_2, \theta_4$  are considered positive tilting to counteract a negative pitch,  $\phi_4$  is positive in the same way as  $\phi$  is and  $\phi_2$  is positive in the opposite way as  $\phi$  is. The equations that govern the problem of pitch rebalancing tilting  $\theta_2, \theta_4 > \theta$  and  $\phi_2, \phi_4$  are:

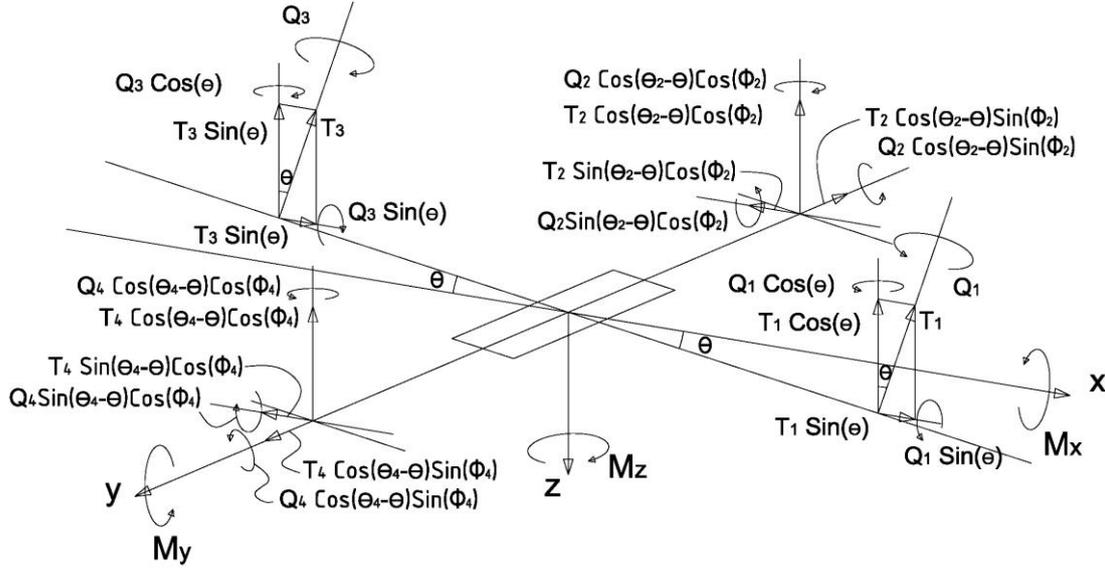


Figure 2.14: Roll Rebalance tilting rotors

$$\sum F_x = 0 \Rightarrow T_1 \sin \theta + T_3 \sin \theta - T_2 \sin(\theta_2 - \theta) \cos \phi_2 - T_4 \cos(\theta_4 - \theta) \cos \phi_4 = 0 \quad (2.70)$$

$$\sum F_y = 0 \Rightarrow T_2 \cos(\theta_2 - \theta) \sin \phi_2 - T_4 \cos(\theta_4 - \theta) \sin \phi_4 = 0 \quad (2.71)$$

$$\sum F_z = 0 \Rightarrow T_1 \cos \theta + T_2 \cos(\theta_2 - \theta) \cos \phi_2 + T_3 \cos \theta + T_4 \cos(\theta_4 - \theta) \cos \phi_4 = m g_0 \quad (2.72)$$

$$\sum M_x = 0 \Rightarrow T_2 \cos(\theta_2 - \theta) \cos \phi_2 \frac{d}{2} - T_4 \cos(\theta_4 - \theta) \cos \phi_4 \frac{d}{2} + Q_1 \sin \theta - Q_3 \sin \theta + Q_2 \sin(\theta_2 - \theta) \cos \phi_2 - Q_4 \sin(\theta_4 - \theta) \cos \phi_4 = 0 \quad (2.73)$$

$$\sum M_y = 0 \Rightarrow T_1 \frac{d}{2} \cos^2 \theta - T_3 \frac{d}{2} \cos^2 \theta + Q_2 \cos(\theta_2 - \theta) \sin \phi_2 + Q_4 \cos(\theta_4 - \theta) \sin \phi_4 = I_y \ddot{\theta} \quad (2.74)$$

$$\sum M_z = 0 \Rightarrow -Q_1 \cos \theta + Q_2 \cos(\theta_2 - \theta) \cos \phi_2 + Q_3 \cos \theta - Q_4 \cos(\theta_4 - \theta) \cos \phi_4 - T_2 \sin(\theta_2 - \theta) \cos \phi_2 \frac{d}{2} + T_4 \sin(\theta_4 - \theta) \cos \phi_4 \frac{d}{2} = 0 \quad (2.75)$$

$$Q_i = R_e T_i \quad (2.76)$$

This system has 9 unknown quantities,  $T_1, T_2, T_3, T_4, \phi_2, \phi_4, \theta_2, \theta_4, \ddot{\theta}$ , and 6 equations linearly independent, so it has infinite solutions. One possible solution is assuming that  $\theta_2 = \theta_4, T_1 = T_3$  and  $\phi_2 = \phi_4$  and consider  $\ddot{\theta}$  and  $\phi_2$  as parameters. Then the system has solution, but it must be found

through an iterative process. This solution corroborate that this performance applied to the ALIV platform can to rebalance the machine.

### Roll and Pitch overall Rebalance without tilting any angle

In section A of the Roll Rebalance section it has been noted that it is not possible to rebalance a roll angle without introducing a moment in the x or z-axes. Also in section A of the Pitch Rebalance section, it has been noted that it is not possible to rebalance a roll angle without introducing a moment in the y or z-axes. But if in the case of the section A of the Roll Rebalance section a moment in the x-axis is introduced and in the case of the section A of the Pitch Rebalance section a moment in the y-axis is introduced, both cases will be complementary.

In a normal situation, we have to rebalance both angles, pitch and roll (Figure 2.15). That problem is formulated in the following system of compatible equations and so it can be solved:

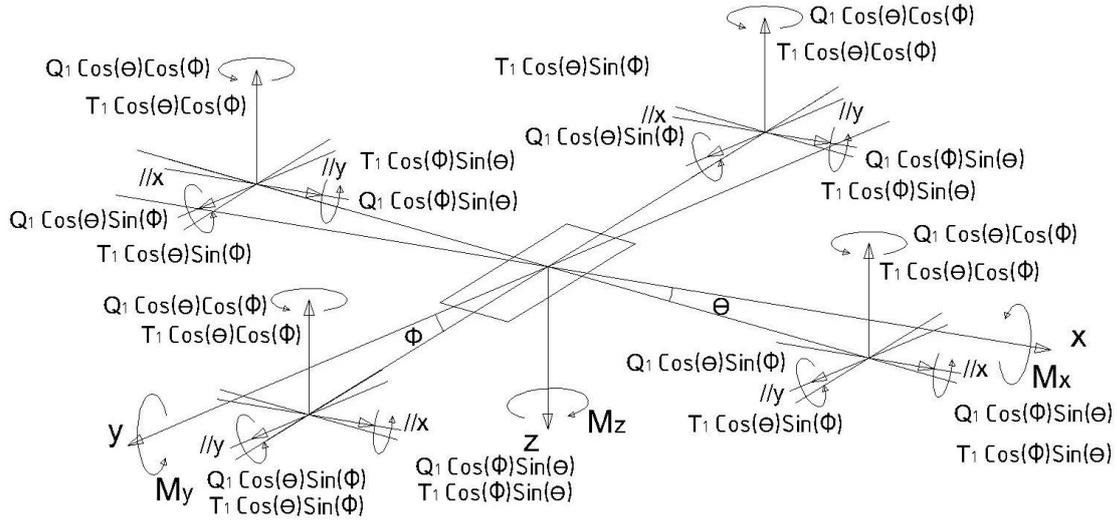


Figure 2.15: Roll and Pitch Rebalance

$$\sum F_x = 0 \Rightarrow (T_1 + T_2 + T_3 + T_4) \cos \phi \sin \theta = m\dot{U} + D_x(U) \quad (2.77)$$

$$\sum F_y = 0 \Rightarrow (T_1 + T_2 + T_3 + T_4) \sin \phi \cos \theta = m\dot{V} + D_y(V) \quad (2.78)$$

$$\sum F_z = 0 \Rightarrow (T_1 + T_2 + T_3 + T_4) \cos \phi \cos \theta = mg_0 \quad (2.79)$$

$$\sum M_x = 0 \Rightarrow (Q_1 - Q_2 - Q_3 + Q_4) \cos \phi \sin \theta + (T_2 - T_4) \cos^2 \phi \cos \theta \frac{d}{2} = -I_x \ddot{\phi} \quad (2.80)$$

$$\sum M_y = 0 \Rightarrow (Q_1 - Q_2 - Q_3 + Q_4) \sin \phi \cos \theta + (T_1 - T_3) \cos \phi \cos^2 \theta \frac{d}{2} = I_y \ddot{\theta} \quad (2.81)$$

$$\sum M_z = 0 \Rightarrow (-Q_1 + Q_2 + Q_3 - Q_4) \cos \phi \cos \theta + (T_1 - T_3) \cos^2 \theta \sin \phi \frac{d}{2} + (T_2 - T_4) \cos^2 \phi \sin \theta \frac{d}{2} = 0 \quad (2.82)$$

$$Q_i = R_e T_i \quad (2.83)$$

This system has 8 unknown quantities,  $T_1, T_2, T_3, T_4, \ddot{\theta}, \ddot{\phi}, \ddot{x}, \ddot{y}$ , and 6 equations linearly independent, so it is possible to solve it making two assumptions. Since  $\ddot{\phi}$  is a function of  $\ddot{\theta}$ , it is not possible to choose them as parameters. For example, if  $\ddot{\theta}$  and  $T_3$  are chosen as parameters, the system has a solution but it is not trivial.

It is also important to notice that if the hypotheses  $T_2 = T_3$  and  $T_1 = T_4$  are made, then the system becomes:

$$2(T_1 + T_2) \cos \phi \cos \theta = mg_0 \quad (2.84)$$

$$(T_1 - T_2) \left( 2R \cos \phi \sin \theta - \cos^2 \phi \cos \theta \frac{d}{dt} \right) = -I_x \ddot{\phi} \quad (2.85)$$

$$(T_1 - T_2) \left( 2R \sin \phi \cos \theta + \cos \phi \cos^2 \theta \frac{d}{dt} \right) = I_y \ddot{\theta} \quad (2.86)$$

$$(T_1 - T_2) \left( \cos^2 \theta \sin \phi \frac{d}{dt} - \cos^2 \phi \sin \theta \frac{d}{dt} - 2R \cos \phi \cos \theta \right) = 0 \quad (2.87)$$

From equation (2.8) it is obtained that  $T_1 = T_2$ , and then  $\ddot{\phi} = \ddot{\theta} = 0$ . So, it can be concluded that the ALIV platform remains stabilized if  $T_1 = T_2 = T_3 = T_4$  although the ALIV is not leveled. In this situation it cannot return to the leveled position.

### **2.3.2.3 Conclusions**

In this Section, it has been analytically proved that when rotors (2,3) turn clockwise and rotors (1,4) turn counter-clockwise, the ALIV platform is capable of performing the principals movements: hover, climb, forward motion, lateral motion, yaw motion and rebalancing motion. Moreover, it is important to say:

- a) Yaw motion can only be performed by tilting rotors 2 and 4. In a standard configuration, changing the direction of rotation of the rotors without tilting the rotors, this manoeuvre cannot be achieved because the yaw acceleration obtained is null.
- b) A common rebalance, without tilting rotors, has been proved to be possible.
- c) To rebalance by tilting rotors is also possible, but it is a manoeuvre very complicated, so a priori it seems better to rebalance the ALIV platform by a standard way, i.e. without tilting the rotors.
- d) When the ALIV platform is not levelled, but  $T_1 = T_3, T_2 = T_4, \phi_2 = \phi_4, \theta_2 = \theta_4$ , its attitude does not change.
- e) When the ALIV concept has no rotors tilted,  $T_1 = T_2 = T_3 = T_4$  and it supports its own weight, its attitude also does not change.

### 2.3.3 Kinematics Equations.

The position  $\mathbf{P}^I = [X; Y; Z]^T$  of the vehicle is expressed in the inertial frame. However its velocity is expressed in the body-fixed frame, so it is necessary to be able to pass from one frame to the other.

The sequence of rotations expressed by Euler angles conventionally used in the aircraft literature to change from the NED frame to the ABC frame is [19]:

1. Rotation about the z-axis, (positive “yaw”  $\psi$ ),  $\mathbf{B}(\psi)$
2. Rotation about the new y-axis, (positive “pitch”  $\theta$ ),  $\mathbf{B}(\theta)$
3. Rotation about the new x-axis, (positive “roll”  $\phi$ ),  $\mathbf{B}(\phi)$

with:

$$\mathbf{B}(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix}, \mathbf{B}(\theta) = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix}, \mathbf{B}(\psi) = \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.88)$$

The vector  $\mathbf{P}^I = [X; Y; Z]^T$  defines the position of the quadrotor in the inertial frame and  $\mathbf{V}^B = [U; V; W]^T$  defines the linear velocity of the quadrotor in the body-fixed frame. These the two vectors are then related by:

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = \mathbf{S}^T \begin{bmatrix} U \\ V \\ W \end{bmatrix} \quad (2.89)$$

With:

$$\mathbf{S} = \begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ -\cos\phi\sin\psi + \sin\phi\sin\theta\cos\psi & \cos\phi\cos\psi + \sin\phi\sin\theta\sin\psi & \sin\phi\cos\theta \\ \sin\phi\sin\psi + \cos\phi\sin\theta\cos\psi & -\sin\phi\cos\psi + \cos\phi\sin\theta\sin\psi & \cos\phi\cos\theta \end{bmatrix} \quad (2.90)$$

Where  $\mathbf{S} = \mathbf{B}(\phi)\mathbf{B}(\theta)\mathbf{B}(\psi)$  is the rotation matrix which transforms a vector from the inertial frame (NED) to the body-fixed frame (ABC).

When  $\theta = \frac{\pi}{2}$ , the rotation matrix reveals a singularity:

$$\mathbf{S} = \begin{bmatrix} 0 & 0 & -1 \\ -\cos\phi\sin\psi + \sin\phi\cos\psi & \cos\phi\cos\psi + \sin\phi\sin\psi & 0 \\ \sin\phi\sin\psi + \cos\phi\cos\psi & -\sin\phi\cos\psi + \cos\phi\sin\psi & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & -1 \\ \sin(\phi - \psi) & \cos(\phi - \psi) & 0 \\ \cos(\phi - \psi) & -\sin(\phi - \psi) & 0 \end{bmatrix} \quad (2.91)$$

This phenomenon is called the gimbal lock and corresponds to the loss of a degree of freedom in a three-dimensional space when two gimbals start spinning in the same plane. As seen in equation (2.902.9), a change in  $\phi$  or in  $\psi$  has the same effect, thus a situation of ambiguity is created: one notation may represent two orientations. To avoid this ambiguity quaternion formulation can be used. However, since these attitudes are not considered for the ALIV3, no other formulation is going to be introduced.

The angular velocities vector  $\Omega^B = [P; Q; R]$  is related with the Euler angles rates,  $\dot{\Phi} = [\dot{\phi}; \dot{\theta}; \dot{\psi}]$  by:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \tan\theta\sin\phi & \tan\theta\cos\phi \\ 0 & \cos\phi & -\sin\phi \\ 0 & \frac{\sin\phi}{\cos\theta} & \frac{\cos\phi}{\cos\theta} \end{bmatrix} \begin{bmatrix} P \\ Q \\ R \end{bmatrix} \quad (2.92)$$

### 2.3.4 Dynamic Equations.

As referred in section 2.1, the aerodynamic forces applied to the quadrotor are neglected. The only forces considered are the thrust and gravity.

Each of the 4 motor-propeller system produces a force  $F_i$ . This work focuses on the stabilization of the Tilt-Quadrotor without tilting any rotor ( $\theta_i = \phi_i = 0$ ), so it is assumed all 4 forces have the same direction, all along de z-axis. So it is possible write the net force applied to the quadrotor (excluding gravity) as  $\mathbf{F}^B = [F_x; F_y; F_z]^T = [0; 0; \sum_{i=1}^4 F_i]^T$ . In following works, when the stabilization would be achieved, this point would have be reconsidered because when the ALIV platform tilts a rotor, the force is also created in the x and y axes.

The rotation of the rotors also produces moments around the axes of the quadrotor. The net moments are written  $\mathbf{M}^B = [M_x; M_y; M_z]^T$ . In section 2.5.1, the model of the propellers will be introduced and these forces and moments will be properly explained.

According to [19], the dynamics of the quadrotor concerning the rotations are given by:

$$\mathbf{I}\dot{\Omega}^B = -\Omega^B \times (\mathbf{I}\Omega^B) + \mathbf{M}^B \quad (2.93)$$

where  $\mathbf{I} = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix}$  is the inertia matrix and  $\mathbf{M}^B = [M_x; M_y; M_z]^T$  are moments produced by the

rotors on the quadrotor. The last equation leads to:

$$\begin{bmatrix} \dot{P} \\ \dot{Q} \\ \dot{R} \end{bmatrix} = \begin{bmatrix} \frac{M_x}{I_x} \\ \frac{M_y}{I_y} \\ \frac{M_z}{I_z} \end{bmatrix} - \begin{bmatrix} \frac{(I_z - I_y)QR}{I_x} \\ \frac{(I_x - I_z)PR}{I_y} \\ \frac{(I_y - I_x)PQ}{I_z} \end{bmatrix} \quad (2.94)$$

Applying the second law of Newton, the acceleration of the quadrotor in the ABC frame results in:

$$m\dot{\mathbf{V}}^B = \mathbf{F}^B + m\mathbf{S}\mathbf{g}^I - \boldsymbol{\Omega}^B \times m\mathbf{V}^B \quad (2.95)$$

where  $\dot{\mathbf{V}}^B = [\dot{U}; \dot{V}; \dot{W}]^T$ ,  $\mathbf{g}^I = [0; 0; g_0]^T$  is the gravity vector and  $g_0 = 9,81 \text{ m/s}^2$  is the gravity constant and  $\mathbf{F}^B = [F_x; F_y; F_z]^T$  is the force produced by the rotors. Thus:

$$\begin{bmatrix} \dot{U} \\ \dot{V} \\ \dot{W} \end{bmatrix} = \frac{1}{m} \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} + \mathbf{S} \begin{bmatrix} 0 \\ 0 \\ g_0 \end{bmatrix} + \begin{bmatrix} RV - QW \\ PW - RU \\ QU - PV \end{bmatrix} \quad (2.96)$$

## 2.4 Model of the Sensors

Sensors allow us to know the attitude and the other variables like acceleration, orientation and position. So, through sensors state variables can be estimated. The sensors noise and bias are the principal sources of measurement imprecision. But once we know the functioning principles and the experimental noise data, it becomes possible to reduce the noise affecting the measurements and guarantee the correct behavior of the quadrotor. Experimental data will be presented in Chapter 3.

In this section, a model of the existing sensors is presented. 3-axis accelerometer, gyroscope and magnetometer, as well as a barometer are part of the IMU (Inertial Measurement Unit) shield of the APM1 connected with the Arduino board, providing information each 0.02s.

### 2.4.1 Gyroscope

Gyroscopes are sensors designed to measure angular velocities  $\boldsymbol{\Omega}^B$ . So, the sensor measurement vector is written as:

$$\bar{\boldsymbol{\Omega}}^B = [\bar{g}_x, \bar{g}_y, \bar{g}_z] \quad (2.97)$$

According to [11] and [15], Gyroscopes are mostly affected in two ways: a stochastic Gaussian noise component  $\boldsymbol{\mu}_g$  and a slowly time varying non-stochastic bias  $\mathbf{b}_g$  corrupting the readings:

$$\bar{\boldsymbol{\Omega}}^B = \boldsymbol{\Omega}^B + \boldsymbol{\mu}_g + \mathbf{b}_g \quad (2.98)$$

The gyroscopes are not located exactly on the center of mass of the quadrotor. According to Henriques [1], if the sensor is located near to the center of mass, the effect can be neglected.

## 2.4.2 Accelerometer

Accelerometers are sensors which measure accelerations, but they are designed to measure the direction of the gravity vector,  $\mathbf{g}^I$  in order to know the pitch and roll angles. Since  $\mathbf{g}^I$  is always pointing down in the NED frame with an intensity of  $g_0 = 9,81m/s^2$ , through the accelerometer measurement vector  $\bar{\mathbf{a}}^B$  the pitch and roll angles can be obtained.

However, the accelerometer is not only sensitive to the gravity, it is also sensitive to accelerations due to the movement of the quadrotor, written  $\mathbf{a}^B$ . According to [1], for the ABC frame, one can write  $\bar{\mathbf{a}}^B = \mathbf{g}^B + \mathbf{a}^B$ . Like in the gyroscopes, accelerometers are also affected by a Gaussian noise and a bias, introducing them in the expression before, the expression of the accelerometer measurement can be written as:

$$\bar{\mathbf{a}}^B = \mathbf{S}\mathbf{g}^I + \mathbf{a}^B + \boldsymbol{\mu}_a + \mathbf{b}_a \quad (2.99)$$

where  $\bar{\mathbf{a}}^B$  is the sensor output in  $m/s^2$ ;  $\mathbf{S}$  is the rotation matrix;  $\mathbf{g}^I$  is the gravity vector defined in the inertial frame;  $\mathbf{a}^B$  is the acceleration of the tilt-quadrotor due to its movement defined in the body fixed frame;  $\boldsymbol{\mu}_a$  is the Gaussian noise component and finally  $\mathbf{b}_a$  is the bias of the accelerometer.

In this case, the accelerometer readings are also affected by the position of the accelerometer in respect to the center of gravity. So, since in the ALIV3 the accelerometer is not positioned at the center of mass, centripetal and tangential terms appear. Developing equation (2.99), we obtain:

$$\bar{\mathbf{a}}^B = \mathbf{S}\mathbf{g}^I + \dot{\boldsymbol{\Omega}} \times \mathbf{r}_s + \boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \mathbf{r}_s) + \boldsymbol{\mu}_a \quad (2.100)$$

where  $\mathbf{r}$  is the distance of the accelerometer from the CG (Center of Gravity). The acceleration due to the bias has been neglected because it is a function of the temperature, then with a calibration routine the problem is eliminated. Equation (2.100) can also be written as:

$$\begin{bmatrix} \bar{a}_x \\ \bar{a}_y \\ \bar{a}_z \end{bmatrix} = g_0 \begin{bmatrix} -\sin\theta \\ \cos\theta\sin\phi \\ \cos\theta\cos\phi \end{bmatrix} + \begin{bmatrix} \dot{Q}r_{s-z} - \dot{R}r_{s-y} \\ \dot{R}r_{s-x} - \dot{P}r_{s-z} \\ \dot{P}r_{s-y} - \dot{Q}r_{s-x} \end{bmatrix} + \begin{bmatrix} Q(Pr_{s-y} - Qr_{s-x}) - R(Rr_{s-x} - Pr_{s-z}) \\ R(Qr_{s-z} - Rr_{s-y}) - P(Pr_{s-y} - Qr_{s-x}) \\ P(Rr_{s-x} - Pr_{s-z}) - Q(Qr_{s-z} - Rr_{s-y}) \end{bmatrix} + \begin{bmatrix} \mu_{a-x} \\ \mu_{a-y} \\ \mu_{a-z} \end{bmatrix} \quad (2.101)$$

In equations (2.100) and (2.101) the acceleration of the quadrotor has been neglected. This simplification is only valid for a near-hovering situation. For example, if the tilt-quadrotor is in free fall in a horizontal position, the output of the accelerometer should be 0, but with the simplification, the measurement would remain  $\mathbf{S}\mathbf{g}^I$ .

### 2.4.3 Compass

The compass is a sensor designed to detect the magnetic North direction, written  $N^I$ . By the definition of the reference frame and neglecting the magnetic inclination or magnetic dip,  $N^I = [1; 0; 0]^T$ . The sensor measurement vector is:

$$\bar{N}^B = \mathcal{S}N^I + \boldsymbol{\mu}_m + \mathbf{b}_m \quad (2.102)$$

where  $\bar{N}^B$  is the sensor measurement in the ABC frame,  $\boldsymbol{\mu}_m$  is the Gaussian measurement noise, and  $\mathbf{b}_m$  is the bias term. Equation (2.1022.10) also can be written as:

$$\begin{bmatrix} \bar{N}_x \\ \bar{N}_y \\ \bar{N}_z \end{bmatrix} = \begin{bmatrix} \cos\theta\cos\psi \\ \cos\psi\sin\phi\sin\theta - \cos\phi\sin\psi \\ \cos\psi\cos\phi\sin\theta + \sin\phi\sin\psi \end{bmatrix} + \begin{bmatrix} \mu_{m-x} \\ \mu_{m-y} \\ \mu_{m-z} \end{bmatrix} + \begin{bmatrix} b_{m-x} \\ b_{m-y} \\ b_{m-z} \end{bmatrix} \quad (2.103)$$

For a near hover situation, it can be assumed that the compass is held horizontally ( $\theta = \phi = 0$ ) with neglected tilt affecting the projection on the x-axis and y-axis. The model can be further simplified to:

$$\begin{bmatrix} \bar{N}_x \\ \bar{N}_y \\ \bar{N}_z \end{bmatrix} = \begin{bmatrix} \cos\psi \\ -\sin\psi \\ 0 \end{bmatrix} + \begin{bmatrix} \mu_{m-x} \\ \mu_{m-y} \\ \mu_{m-z} \end{bmatrix} + \begin{bmatrix} b_{m-x} \\ b_{m-y} \\ b_{m-z} \end{bmatrix} \quad (2.104)$$

The measurements of the compass are affected by soft iron and hard iron distortions. According to [9], hard iron distortions are caused by the presence of magnet fields; they produce a constant additive error regardless of the orientation. Soft iron distortions are similar to hard iron distortions but the error varies with the orientation. As a consequence, the hard iron distortions can be included in a constant bias term of equation (2.1032.10), but the soft iron effect cannot be easily accounted for and will be neglected.

### 2.4.4 Barometer

The barometer is a sensor used to obtain the altitude by means of pressure. The sensor used is also capable of make temperature measurements. Referring to the model of the standard atmosphere it is possible to compute the data to obtain the altitude.

With the measured pressure  $p$  and the pressure at sea level  $p_0 = 101325Pa$ , the altitude in meters can be calculated from the international barometric formula [2]:

$$Z = -44330 \cdot \left( 1 - \left( \frac{p}{p_0} \right)^{\frac{1}{5.255}} \right) \quad (2.105)$$

Thus, a pressure change of  $\Delta p = 1hPa$  corresponds to  $8.43m$  above sea level.

In this case the sensor is also affected the noise, so the sensor measurement is:

$$\bar{p} = p + \mu_p \quad (2.106)$$

where  $\bar{p}$  is the sensor measurement and  $\mu_p$  is the Gaussian measurement noise. The readings are not affected by the sensor position in the frame.

## 2.5 Model of the Actuators

Actuators are employed to produce forces on the system. Through these forces we bring the tilt-quadrotor to a desired state. In this case the actuators are the motor and propeller sets.

The Motors used are brushless, and controlled by a speed controller which receives the PWM signal from the Arduino board and sends the respective voltage to the motor. A better explication of the brushless motor and how it is controlled is carried out in section 3.2.1. Propellers are connected to the motors, thus, when motors start to turn, they make the propellers spin at the same angular velocity and produce a force. This force depends on the angular velocity.

### 2.5.1 Propellers

The friction forces of the air in contact with the propellers, the flapping of the blades and the ground effect will be neglected.

According to the Blade Momentum Theory [6], the thrust,  $T_i$  and the moment,  $Q_i$  created by each propeller are proportional to the squared angular velocity of the blade, as shown in equation (2.1072.10) and (2.1082.10):

$$T_i = \rho \pi r_r^4 C_T \omega_i^2 \quad (2.107)$$

$$Q_i = \rho \pi r_r^5 C_p \omega_i^2 \quad (2.108)$$

where  $C_T$  is the dimensionless constant of thrust and  $C_p$  is the dimensionless constant of power.

In order to simplify these equations we can also write:

$$T_i = K_T \omega_i^2 \quad (2.109)$$

$$Q_i = K_Q \omega_i^2 \quad (2.110)$$

where  $K_T$  is a constant that relates the thrust to the square of the angular velocity of the propellers and  $K_Q$  is a constant that relates the moment to the square of the angular velocity of the propellers. So, the two constants are defined by:

$$K_T = \rho \pi r_r^4 C_T \quad (2.111)$$

$$K_Q = \rho \pi r_r^5 C_p \quad (2.112)$$

All these constant will be identified in Chapter 3.

## 2.5.2 Model of the motors

As it is presented in the introduction of the section 2.5, the motors employed are brushless which need a speed controller that receives a PWM signal from the Arduino board associated to an angular velocity. The PWM signal is a square digital signal, a common method to provide an analog signal with digital means<sup>31</sup>. So, Arduino board works with a normalized 50Hz square wave to code the signal as a pulse width varying from 1 *ms* to 2 *ms*. Once the speed controller receives the PWM signal, it turn the PWM signal into a triphasic signal to feed the brushless motor.

Each motor receives the triphasic signal corresponding to an order of angular velocity  $\omega_i$ . The behavior of the motor is a consequence of an interaction between an electric and mechanical dynamics. The electrical and mechanical systems can be represented as Figure 2.16, where  $e$  is the counter electromotive force,  $v$  is the voltage,  $i$  is the intensity,  $\theta$  is the angular rotation of the rotor relative to the stator,  $R$  is the electric resistance,  $L$  is the electric inductance  $b$  motor viscous friction constant.

From the figure above, it is derived the following governing equations based on Newton's 2<sup>nd</sup> law and Kirchhoff's voltage law [13].

$$v = k_e \dot{\theta} + L \frac{di}{dt} + Ri \quad (2.113)$$

$$J\ddot{\theta} + b\dot{\theta} = k_t i \quad (2.114)$$

where  $k_e$  is the electromotive force constant and  $k_t$  is the motor torque constant.

---

<sup>31</sup> <http://www.arduino.cc/en/Tutorial/PWM>

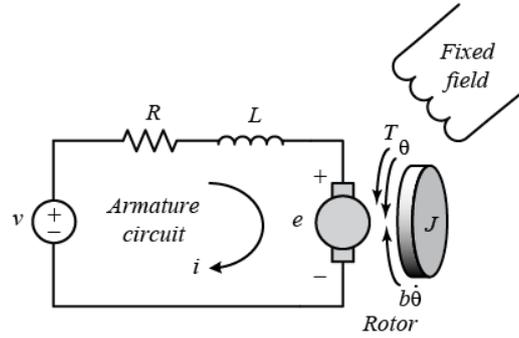


Figure 2.16: Electrical and Mechanical Schematic [13]

Using the input  $PWM_i$  instead of voltage, and  $\omega_i$  instead of  $\dot{\theta}$ :

$$\frac{\omega_i}{PWM_i} = \frac{K_e}{(Js+b)(Ls+R)+K_e^2} \quad (2.115)$$

where  $K_e$  represent both the motor torque constant and the electromotive force constant because in SI units, the motor torque and the electromotive force constants are equal [13].

Equation (2.115) presents an accurate model of the motor; however it is difficult to measure all the parameters of the motor. The model can be simplified to a first-order system corresponding to the mechanical dynamics [1]:

$$\frac{\omega_i}{PWM_i} = \frac{k_i}{\tau s + 1} \quad (2.116)$$

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

In this approach it is considered that the influence of the electrical dynamics of the motor is considerably faster than the mechanical dynamics of the motor with the propellers. Through the equation (2.116) it is possible to say that in the permanent regime the angular speed of rotation is proportionally to the PWM signal. In the following Chapter it is going to be presented the identification of the actuators system, seek for a linear relation between  $\omega_i$  and  $PWM_i$ .

In addition to the presented model, the actuator set, contains some nonlinearities since the PWM signal take discrete values. They are also going to be explained better in following chapters.

## 2.6 Tilt-Quadrotor Simulator

In this section, the implementation of the described model in *Simulink* is explained.

### 2.6.1 Dynamic and Output Equations

The dynamic and the output equations are implemented in the same block in *Simulink* since they are related directly. The *Matlab*'s function is called "nonlinear\_system.m" and it contains the kinematics and dynamics equations that relate the inputs of the model and the outputs. Figure 2.17 shows this implementation in *Simulink*.

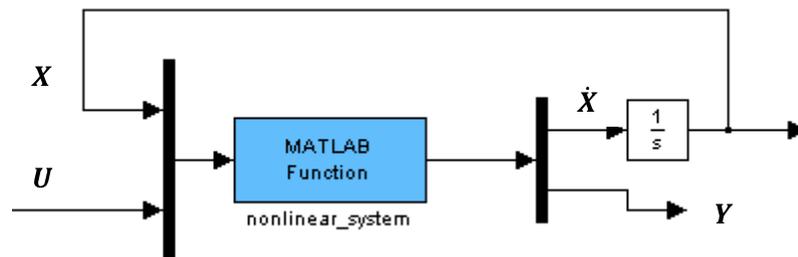


Figure 2.17: Dynamic and Output Equations

The input vector corresponds to the frequency of rotation of the motors  $U = [\omega_1, \omega_2, \omega_3, \omega_4]^T$  which has already suffered the implementation of the motors. The state vector is  $X = [V, \Omega, P, \Phi]^T$  and the output vector is  $Y = [\bar{a}, \bar{\Omega}, \bar{N}, \bar{Z}, \bar{W}]^T$ , where  $\bar{Z}$  and  $\bar{W}$  are the altitude estimated and the vertical velocity estimated respectively.

### 2.6.2 Sensors

The output equation corresponding to the sensor measurements obtained from "nonlinear\_system.m" file does not include the bias and noise terms. So, to introduce those terms the model presented in Figure 2.18 is implemented:

The user can choose to perform simulations using either theoretical values for the noise affecting the sensors or use the experimentally defined values by setting a flag during the simulation. Note also that different seeds were used to generate the noise, avoiding a correlation between the different sensors.

### 2.6.3 Motors

The motor sets (Motors and ESCs) receive a PWM signal and produce a corresponding rotational speed for the propellers. The motor input has several nonlinearities; first the saturation block ensures

that the signal is within the defined boundaries, then the dead zone of the motors is simulated and finally the quantizer ensures that the signal is an integer number. Then, the signal is converted to obtain the angular velocities through a *Matlab* function where each curve PWM-RPM/V is implemented. The battery discharge time is too large compared with the simulation times, so a constant battery voltage can be considered. A nominal battery voltage of 11.1V is used. Figure 2.19 shows the implementation of the motors model.

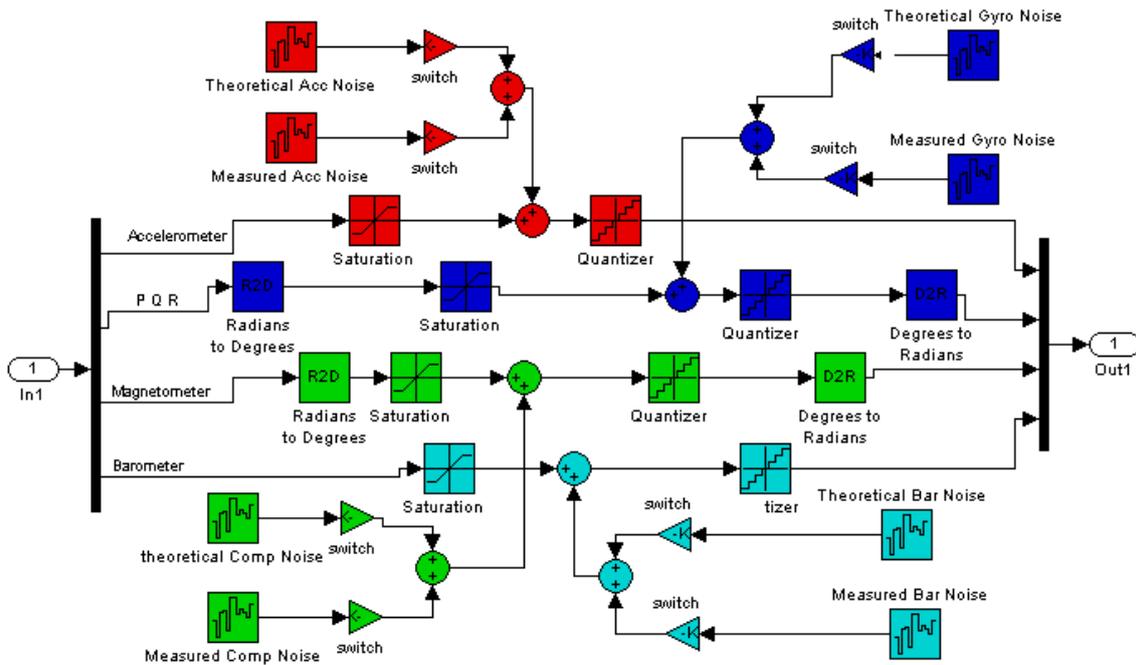


Figure 2.18: Implementation of the Sensors

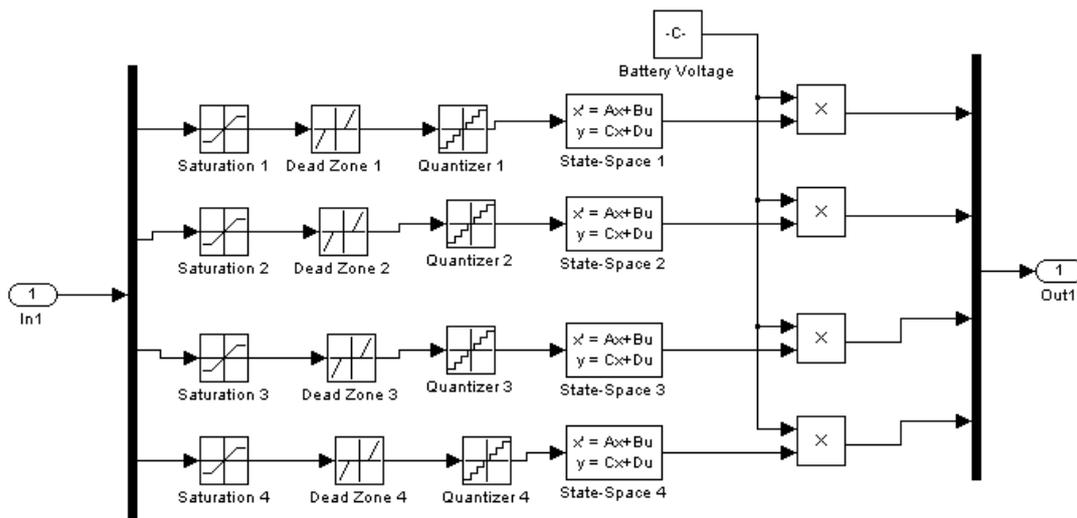


Figure 2.19: Implementation of the Motors

### 2.6.4 Simulation Model. Open Loop.

The overall implementation of the model in open loop is shown in Figure 2.20, with the details of the model of the motors and the model of the sensors described in figures. It also contains the controller LQR and the estimator EKF (Extended Kalman Filter) which will be presented in Chapter 4. They are disconnected when the open loop is simulated. The results of the open-loop will be also presented in Chapter 4.

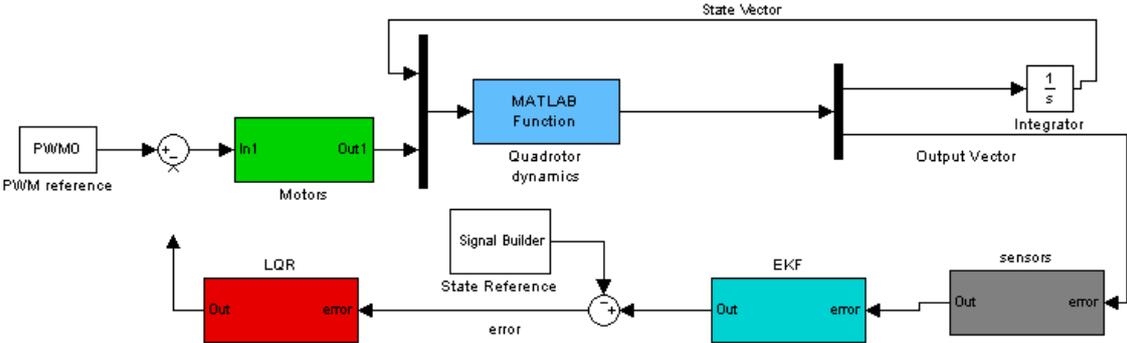


Figure 2.20: Implementation of the Open Loop

## Chapter 3

# Tilt-Quadrotor Identification

In this Chapter the identification of the ALIV3 platform is carried out. It consists in three parts. The first part presents the identification of the mass properties of the device: the Center of Gravity (CG), the total mass and the moments of Inertia.

In the second part the actuators are identified. Firstly the characteristics of the implemented actuators are presented, followed by a description of the experimental setup. Then a discussion of which curve represents the motor's performance is carried out. The influence of the battery discharge, the temperature and the BEC of the speed controllers is studied. A first measurement of the endurance is realized and the identification of the propellers is also accomplished.

Finally, in the third part the sensors are identified. A description of the characteristics of the different sensors is followed by the noise measurements.

### 3.1 Prototype Identification

In this Chapter mass and dimension properties of the ALIV3 are going to be measured. In respect to mass properties, the magnitudes measured are the Center of Gravity, Total Mass and the Moments of Inertia. Finally, the main geometrical dimensions are going to be presented.

#### 3.1.1 Center of Gravity

The Center of mass, which is also the Center of Gravity (CG) in this case, is measured by a graphic experimental procedure. It consists in hanging the ALIV3 by a string from different points. First the CG in the horizontal plane is measured and later in the vertical plane. This string draws a line upon the plane we are measuring. So, the CG is placed where these lines intersect.

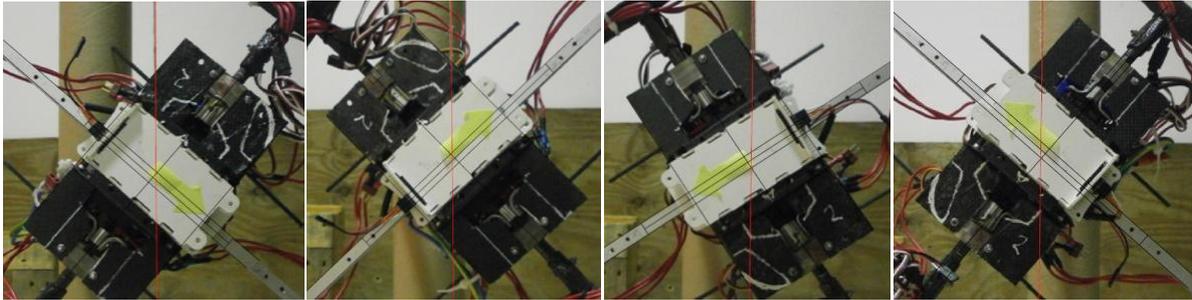


Figure 3.1 a)

Figure 3.1 b)

Figure 3.1 c)

Figure 3.1 d)

Figure 3.1: Determination of CG in the horizontal plane I

Figure 3.1 shows the identification of the CG in the horizontal plane. All these figures are shown overall in Figure 3.2. It can be appreciated that this method is not completely accurate but the error can be considered acceptable. The lines intersect in 5 points. However, one of them is too far (14mm), so it is not considered. The remaining points are confined in a circle with a radius of 4mm and centered where x-axis and y-axis intersect, so it can be assumed that the CG is placed in this point.

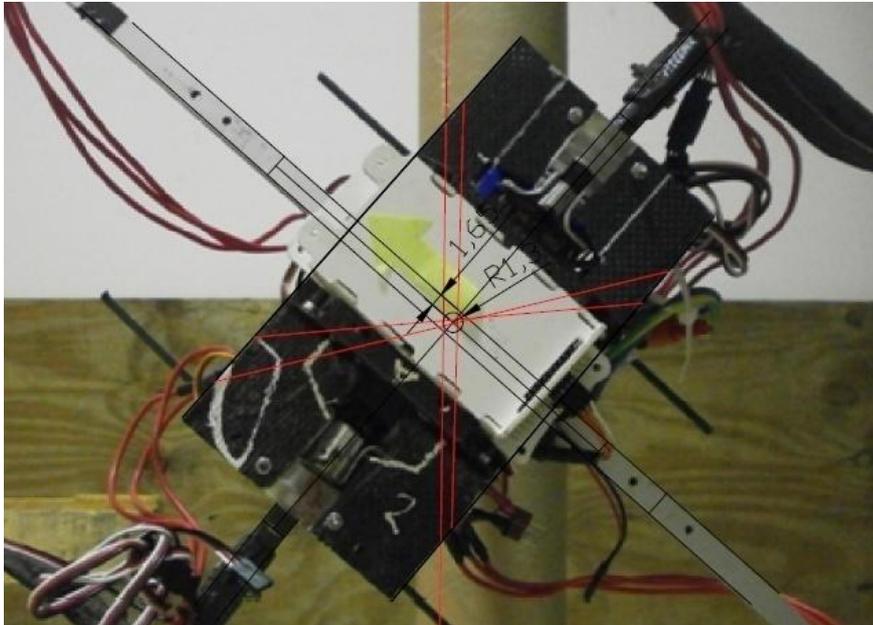


Figure 3.2: Determination of CG in the horizontal plane II

Figure 3.3 shows the identification of the CG in the vertical plane. In this case, the intersections of the red lines with the center of central board are not exactly coincident. However, these points distance less than 3,1mm from the central board. So, it can be assumed that the CG is placed in the upper Central board. Thus, the CG is considered to be placed in the geometric center and at height of 161mm from the base.

To complete this section, the total mass measured with all components mounted is  $m = 1903g$ .

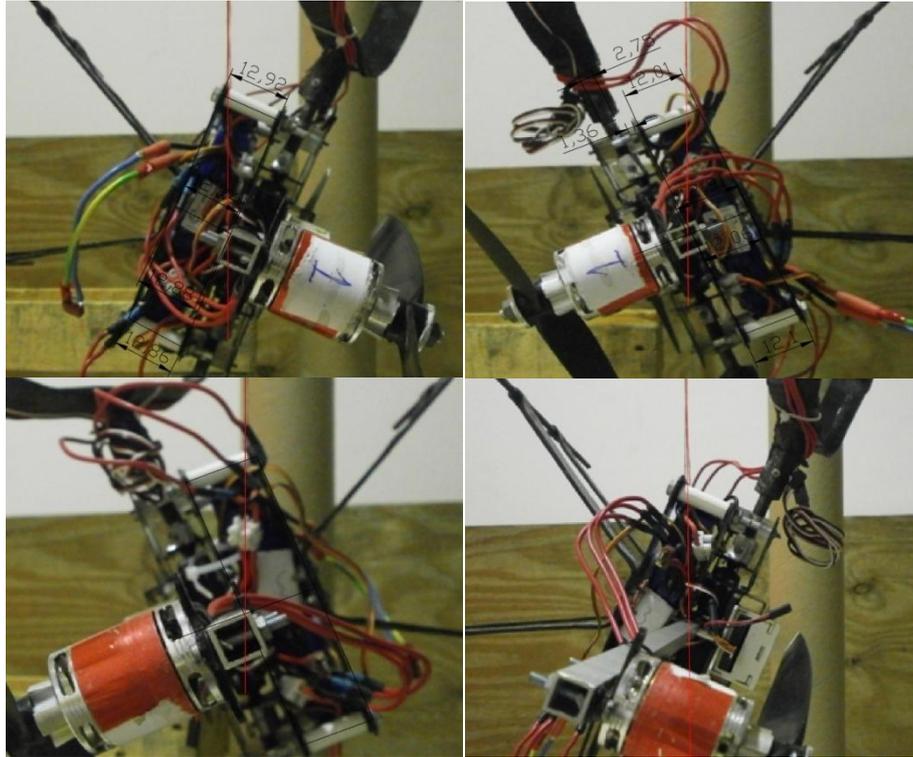


Figure 3.3: Determination of CG in the vertical plane

### 3.1.2 Moments of Inertia

The ALIV3 is assumed symmetric about the x-axis and also about the y-axis. So, the products of Inertia in x and y are null and it is possible to assume that the Matrix of Inertia is diagonal [10]:

$$I = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \quad (3.1)$$

Fernandes use in his work a simplification of the mass distribution of his quadrotor to calculate the moments of inertia. However, since the structure of the ALIV3 is much more complex, it was decided to design a procedure to measure them.

According to [5] there are mainly two experimental methods to measure the moments of inertia: oscillation methods and acceleration methods; being the firsts more accurate. Oscillation methods consider a linear system made by a rigid body constrained in such a way that the only possible motion is a rotation about an axis fixed in space. The body is subject to an elastic restoring torque with stiffness  $k$  and a viscous drag torque with damping coefficient  $c$ . Introducing the damping ratio  $\zeta = c/2\sqrt{kI}$ , the period  $T$  of the free oscillations is:

$$T = 2\pi \sqrt{\frac{I}{k} \frac{1}{\sqrt{1-\zeta^2}}} \quad (3.2)$$

The test is performed by making the ALIV3 oscillate about the axis under study, measuring its period and by computing the respective moment of inertia  $I$ ; once the stiffness and damping are known.

So, it was developed and built a structure specially designed to measure the moments of inertia of a quadrotor. This structure shown in Figure 3.4 works by constraining the motion to a rotation about an axis only. It has elastics to restore the position (see Figure 3.5).



Figure 3.4: Moment of Inertia's Structure



Figure 3.5: Elastics of the Moment of Inertia's Structure

The idea is to force a rotation with a certain amplitude, and leave it oscillate freely. Since this system has a damping, the amplitude suffers a reduction in each period. So, according to the equations of an oscillation movement<sup>32</sup>, this reduction in amplitude is related to the damping:

$$A(t) = A_0 e^{(-\gamma t)} \quad (3.3)$$

where  $\gamma$  is defined by  $\gamma = \frac{c}{2I}$ . So, the damping and the period are:

$$c = 2I \frac{1}{T} \ln \left( \frac{A_0}{A_1} \right) \quad (3.4)$$

$$T = 2\pi \sqrt{\frac{I}{k} \frac{1}{\sqrt{1 - \frac{c^2}{4kI}}}} \quad (3.5)$$

In order to know the stiffness of the system, a prism whose moment of inertia about the rotation allowed is known was employed:

$$I_z = m \cdot \frac{l^2 + a^2}{12} \quad (3.6)$$

where  $l$  is the length and  $a$  is the width of the prism.

Thus, the test is performed fixing the quadrotor about the axis whose moment of inertia will be identified. Then, leaving the quadrotor to oscillate freely, the period and the amplitude reduction are measured. The period is measured with the help of a tachometer and the amplitude reduction was

<sup>32</sup> <http://www.sc.ehu.es/sbweb/fisica/oscilaciones/amortiguadas/amortiguadas.htm>

measured visually. The test data is presented in the Appendix A.1. Table 3.1 resumes the quadrotor identified moments of inertia

Inertia	$[kg \cdot m^2]$
$I_x$	0,0367
$I_y$	0,0262
$I_z$	0,0504

Table 3.1: Inertia Measurements

## 3.2 Actuators Identification

Following equation (2.11) it was expected to find a linear relationship between the PWM signal and the RPMs. So, the first idea was to take 12 values of RPMs corresponding to 12 PWM signal values: [950,1050,1150, 1250, 1350, 1450, 1550, 1650, 1750,1850,1950, 2050, 2150]  $\mu s$ ; and take some more points near the point of operation, which is the point at the ALIV3 is expected to hover, i.e. when the thrust is equal to the weight of the quadrotor. All this points were thought to be taken without stopping each motor. However, at the end of the first test some observations were made :

- a) The relationship between RPM and PWM was not linear.
- b) For the same PWM, different RPMs were measured.
- c) The temperature and the battery influence the test

The second idea was to stop the motor after each measurement in order to minimize the temperature effect and take three values for each PWM in order to minimize the error. The results of these tests are presented and commented in section 3.2.3.

In section 3.2.4 the curve Time-RPM is represented. With this curve was pretended to study what happens with the RPM along the time for each PWM. Then it was studied the effect of the temperature and the battery discharge effect. In this test, the values were taken in constant intervals of time maintaining the motors working. In this section, another variable RPM/V is considered in order to eliminate the effect of the battery discharge.

Finally, in section 3.2.5 the curve PWM-RPM/V is presented, which is an acceptable way to estimate the behavior of the motors removing the effect of the temperature and the battery discharge.

In section 3.2.6 the effect of the BEC is studied, concluding that it has not an important effect over the performance. In section 3.2.7 the ALIV3 flight time is tested, which is the maximum time that the ALV3 can fly. In section 3.2.8 the propeller identification is made.

### 3.2.1 Characteristics of the motors

The motors used in the ALIV3 project are Brushless DC Motors. The main characteristic of a Brushless motor is that it works without brushes. Unlike DC brushed motor, they have the permanent magnets placed on the rotor. No connection needs to be done with the rotor, thus, no brush-commutator pair needs to be made. These motors are three-phase, they are fed with AC current, and so that they need an Electronic Speed Controller (ESC) which convert DC into AC.

The main advantages of these motors over DC brushed motors are better performance (higher speeds and torque), smaller size and higher efficiency which means less heat to get out of the machine, lower power consumption and less noise.

The motor that has been finally mounted in the ALIV3 is the one chosen by Fernandes in his Master Thesis [14]. It is the Turnigy 2217 16turn 1050kv 23A Outrunner<sup>33</sup> whose characteristics are shown above:

$K_v$ [rpm/V]	1050
Operating Current [A]	6~20
Weight [g]	73
Dimensions [mm]	27.6 × 36
Shaft Size [mm]	3.175

Table 3.2: Motor Characteristics

As it is said, an ESC is needed to control the Brushless DC Motor. It is an electronic circuit with the purpose of varying the speed and the rotor direction of rotation. The ESC receives a nominal 50HZ PWM signal from the APM1 board whose pulse width varies from 1 ms to 2 ms, and provides to the motor a three-phase electric power low voltage source of energy. A 1 ms width pulse at 50Hz corresponds to a null speed. A 1.5 ms pulse-width input signal results in a 50% duty cycle output signal that drives the motor at approximately half-speed. And finally a 2.0 ms width pulse corresponds to full speed.

The ESCs incorporate also a battery eliminator circuit (BEC) to regulate voltage for the receiver, or as in this case, to feed the APM1 board through the Power Distribution Board (PDB). Only one of the four ESCs is used to feed the APM1 board. Later it will be studied that this fact does not imply a reduction in power for that motor.

The ESC chosen by Fernandes [14] for the ALIV3 was the Hobbyking SS Series 25-30A<sup>34</sup>. However, due to some malfunctioning of these ESCs, it was decided to replace them by the RC-plus

---

<sup>33</sup> [http://www.hobbyking.com/hobbyking/store/\\_5690\\_\\_Turnigy\\_2217\\_16turn\\_1050kv\\_23A\\_Outrunner.html](http://www.hobbyking.com/hobbyking/store/_5690__Turnigy_2217_16turn_1050kv_23A_Outrunner.html)

SKYSPORT 30 30A - 2-3S - BEC<sup>35</sup> which is programmable and it has a higher quality. The characteristics of the new ESC are shown in Table 3.3:

Cont Current [A]	30
Burst [A]	40
BEC	5V/2A
Weight [g]	26
Size [mm]	52 × 24 × 8

Table 3.3: ESC Characteristics<sup>35</sup>

### 3.2.2 Experimental Setup

The goals of the test are the identification of the curve PWM-RPM and the coefficients of thrust and power of the propellers. The experimental setup of the Figure 3.6 allows to measure both things at the same time. This setup consist of a motor support, the Motor/ESC set, a balance, a tachometer, a tachometer support, a multimeter, two computers, a power supply, the APM1 and the Power Distribution Board (PDB).

One computer is connected to the Arduino board in order to send at each moment the desired PWM signal. The power supply, which in most of tests is the battery employed in the ALIV3 (ZIPPY Flightmax 5000mAh 3S1P 20C<sup>36</sup>), is connected to the PDB through the multimeter in order to measure at each moment the voltage and the intensity supplied by the battery. The APM1 board is also connected to the PDB.

The PDB is a board which distributes the power and PWM signals. It is needed because while motors work at 11.1V, the ArduPilot and Servos work at 5V. So, the PDB receives as inputs the power from the battery, the BEC from the ESCs and the control signals from the ArduPilot. It has also as outputs the power to ESCs, the power to the ArduPilot and the control signals to the ESCs. It is important to emphasize that this board does not modify any tension or signal. It only distributes the tension of 11.1V from the battery to the four ESCs. Then the board receives the tension of 5V from the BEC of and connects only one of them to the Ardupilot in order to supply the power needed for this device. It also receives each control signal from the Ardupilot and sends them separately to each ESC.

---

<sup>34</sup> [http://www.hobbyking.com/hobbyking/store/\\_\\_26032\\_\\_Hobbyking\\_SS\\_Series\\_25\\_30A\\_ESC\\_UK\\_Warehouse\\_.html](http://www.hobbyking.com/hobbyking/store/__26032__Hobbyking_SS_Series_25_30A_ESC_UK_Warehouse_.html)

<sup>35</sup> <http://www.e->

[rc.be/index.php?option=com\\_virtuemart&view=productdetails&virtuemart\\_product\\_id=4312&virtuemart\\_category\\_id=172](http://www.e-rc.be/index.php?option=com_virtuemart&view=productdetails&virtuemart_product_id=4312&virtuemart_category_id=172)

<sup>36</sup> [http://www.hobbyking.com/hobbyking/store/\\_\\_8579\\_\\_ZIPPY\\_Flightmax\\_5000mAh\\_3S1P\\_20C.html](http://www.hobbyking.com/hobbyking/store/__8579__ZIPPY_Flightmax_5000mAh_3S1P_20C.html)

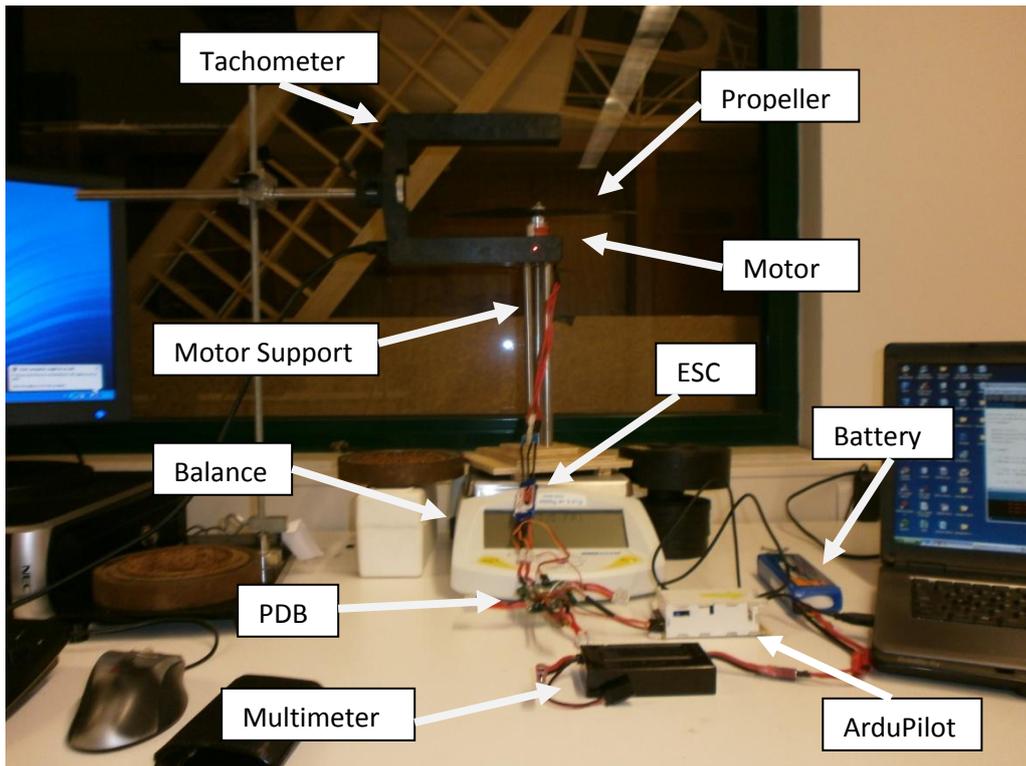


Figure 3.6: Experimental Setup

The tachometer consists of two arms and a light beam that is passed from one arm to the other. When the light beam is interrupted it makes a measurement. RPMs are measured dividing the number of interruptions by the time. In fact, if we consider a time interval of 30 seconds, for a propeller of two blades, the RMPs are the number of interruptions:

$$RPMs = \frac{\# \text{ of interruptions}}{\text{time}} = \frac{\# \text{ of interruptions}}{30s} \frac{1}{2 \text{ blades } 1min} \frac{60s}{1min} = \# \text{ of interruptions}$$

The tachometer is set up in such a way that the propeller crosses the two arms and so interrupts the laser beam. It is connected to the other computer, where the information of the number of interruption is received.

The motor/propeller set is attached to the motor support, which has been specially designed and built for this test. The propeller is inverted to impulse the wake upwards making the force downwards in order for the force produced to be measured (weighted) by the balance. The wake is considered to be free and the force not affected by the ground effect.

Thus, the test is performed by sending a PWM signal from the computer to the ArduPilot. Then the motors start to turn, the RPMs are presented in the computer, the voltage and the intensity in the multimeter and the force in the balance.

All tests were done in the Mechanical laboratory, where the atmospheric conditions were assumed to remain constant:

$$T = 26^{\circ}\text{C}$$

$$p = 102 \text{ kPa}$$

### 3.2.3 Curve PWM-RPM

After the first observations described at the introduction of this Chapter, the average of three measurements for each PWM signal was considered, in order to minimize the error. The motor was stopped after each measurement in order to minimize the effect of the temperature.

The curve obtained for the four motors is shown in Figure 3.7, where the points represented by a circle are presented in the Appendix A.2 and the curves correspond to an interpolation with the function *spline* in *MATLAB*:

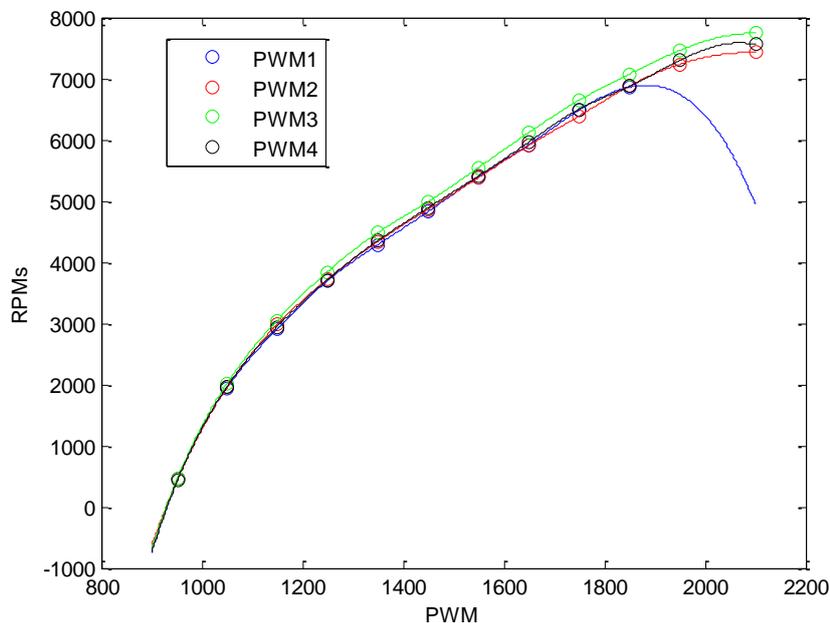


Figure 3.7: Curve PWM-RPM

From this curve obtained, it could be said that the relationship between the RPM is not linear for all the range of PWM. However, around the point of operation which is supposed between  $1400 \mu\text{s}$  and  $1600 \mu\text{s}$ , the relationship can be considered as linear.

From Figure 3.7 can be also observed that motors 2 and 4 have a very similar behavior versus the motor 1 and 3. Maybe the similitude between motors 2 and 4 lies in the fact of having the same longitude connection cables, so it would be expected that motor 1 and 3 had also a similar behavior. However motor 1 has a lower performance versus motor 3. At full load, motor 1 begins to vibrate a lot and so, the curve after  $1850 \mu\text{s}$  cannot be taken into account. Nevertheless, the differences between the motors are in a range of 2,7% around the point of operation.

### 3.2.4 Curve Time-RPM

In order to analyze the effect of the discharge of the battery and the effect of the temperature over the performance of the motors, the curve Time-RPM was obtained. With this purpose, RPM measurements were taken without stopping the engine in intervals of 35s approximately, for the same PWM signal. It is assumed that the behavior of all motors is the same, so only one motor (motor 3) was tested.

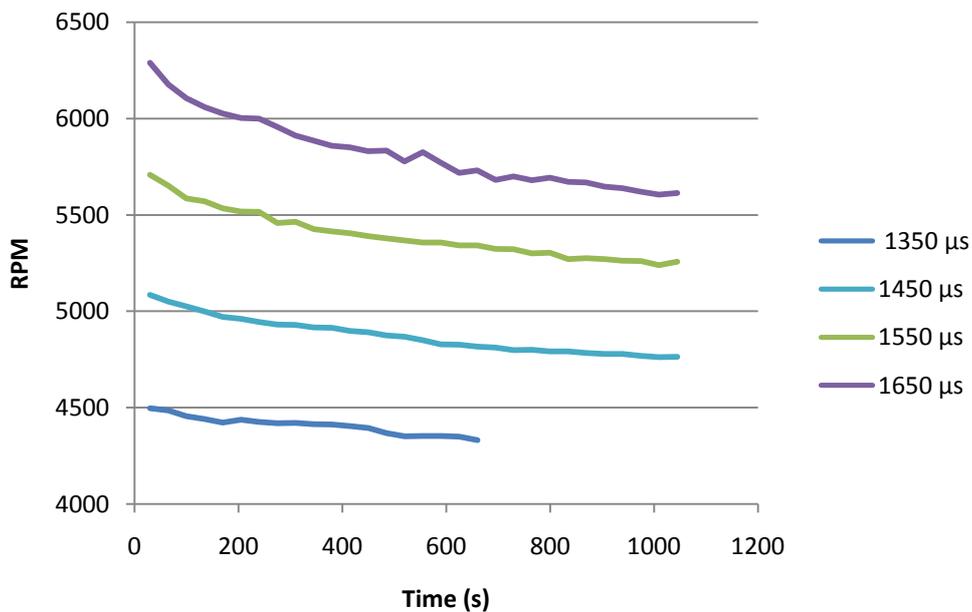


Figure 3.8: Time-RPM Motor 3

#### i) Battery - Temperature analysis

In order to study the influence of the battery discharge two types of power supplies were considered. The first values were obtained with the battery connected to the PDB and the second values obtained with the battery replaced by a power supply where the voltage could be maintained constant at 11.1V.

Figure 3.8 represents the response of motor 3 during 1050s approximately for different PWM signals. For the four PWM signals, two tendencies can be observed: a high negative slope during the first 200 seconds, and a smoother slope during the rest of the test.

From Figure 3.8 it can be already drawn that the first slope is the sum of the effect of the temperature and the battery discharge, and the second negative slope is due to the effect of the battery discharge solely since the temperature is supposed to remain constant after a certain time.

To prove this hypothesis two more testes were done. The first test consists in replacing the battery by the power supply and comparing both behaviors. The second test consists in measuring the

temperature of the base of the motor with a sensor in order to verify if it remains constant after the first 200 seconds approximately.

Figure 3.9 shows the results of the first test, where the performance of the motor 3 connected to the battery is compared with its performance connected to the power supply at 11,1V. From this figure it is proved that the second slope is due to the effect of the battery discharge, because when the power supply is connected, the RPMs vary in a constrained range.

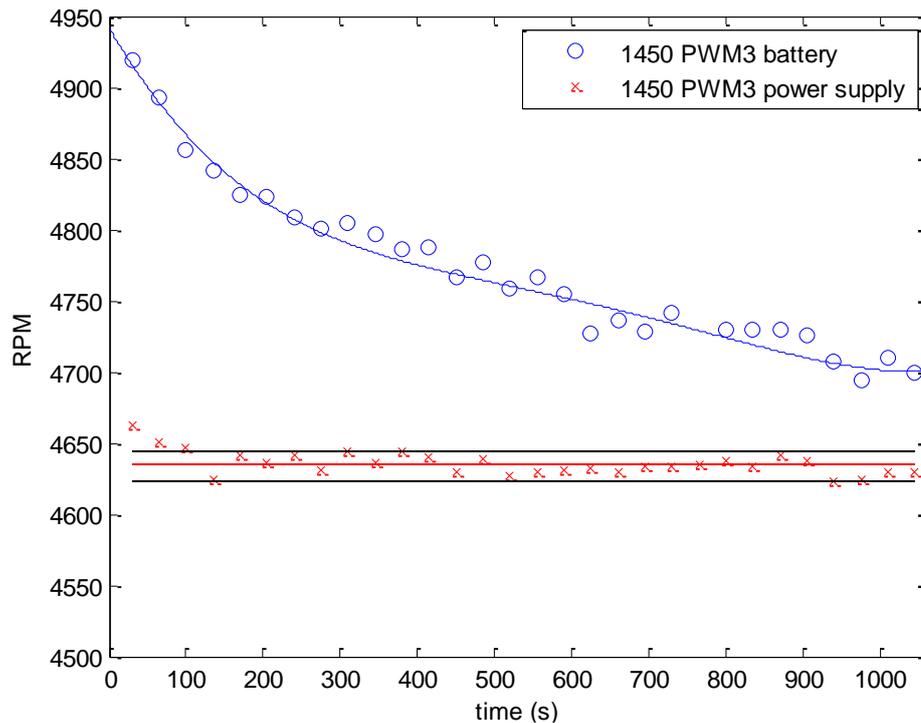


Figure 3.9: Battery vs Power Supply

Finally, Figure 3.10 shows the results of the second test, where a LM35 temperature sensor was used. This figure represents the response of the temperature of motor 3 with time for different PWM signals. From this test it can be concluded that the temperature has a first order response. Increasing the duty cycle of the PWM signal, the temperature also increases, rising values very high for 1850  $\mu$ s. As it was predicted, after the first 200s the effect of the temperature in the performance of the motors can be neglected for the PWM signal used in Figure 3.9.

## **ii) RPM/V variable**

Once the effects of the temperature and the battery discharge have been studied, we can conclude that the curve PWM-RPM is not a good approximation of the performance of the motors. That curve does not take into account the influence of discharge of the battery. The effect of the temperature is not so important because after some seconds, their effect can be neglected. However, in the case of the battery, different states of the battery are going to be found during the flight. Therefore another variable must to be considered.

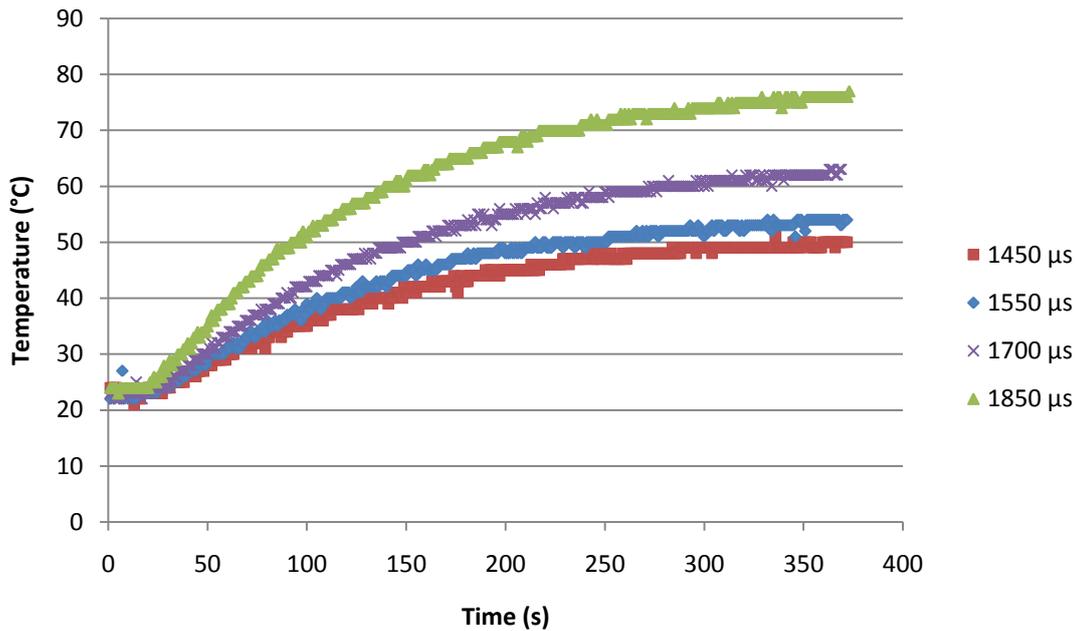


Figure 3.10: Performance vs Temperature

The idea is to introduce another variable to eliminate the effect of the battery discharge. Due to the variable which controls the battery discharge is the Voltage, it was considered the  $RPM/V$  as the variable chosen, where  $V$  is the voltage of the battery.

If we represent the  $RPM/V$  over the time from the data of the former test ( $RPM$ -Time of the motor 3 for a signal of  $1450 \mu s$ ), we obtain Figure 3.11. It can also be observed the effect of the temperature in the first 200s. However, the most important of these values is not the shape of the graph, it is that the maximum relative difference between values of  $RPM/V$  achieved are lower than 1,74% for all the time.

Although the relationship seems to be linear, the slope is very small and therefore one can consider that this magnitude remains constant along the time, i.e. one can take its average. Then, if we do an estimation of the  $RPMs$  for  $1450 \mu s$  with the voltage of the battery measured in each measurement and the average of  $RPM/V$  for  $1450 \mu s$ , the maximum relative error between experimental values and estimated values is 1,0%; and the average of the relative errors is 0,47%.

Relative errors of 1% can be considered acceptable and the magnitude  $RPM/V$  can become a good way to model the behavior of the motors when the power is supplied from the battery.

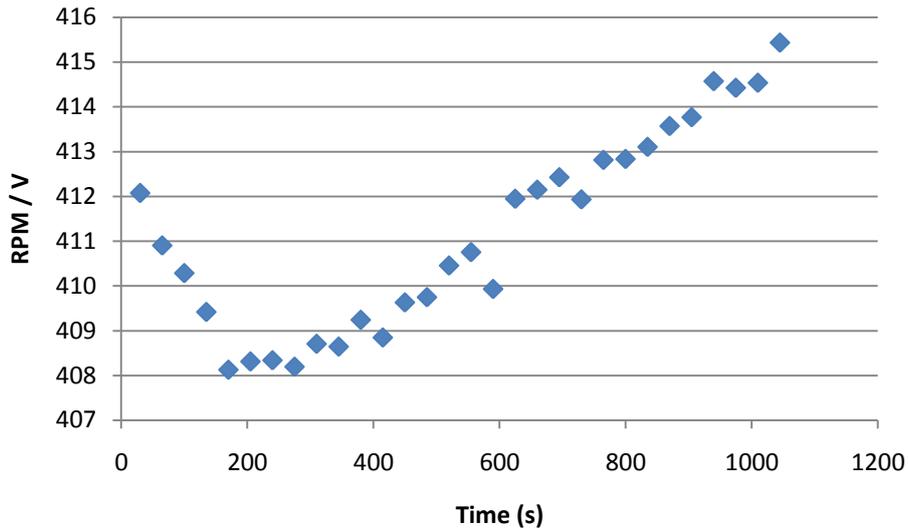


Figure 3.11: *RPM/V* vs Time Motor 3, 1450  $\mu$ s

### 3.2.5 Curve PWM-RPM/V

In the previous section the curve *RPM/V* was considered to model the performance of each motor, and it was assumed that the value *RPM/V* was constant for each PWM signal. To obtain the curve *PWM – RPM/V* for each motor, the *RPM* and the Voltage were measured during 1000s for each PWM:

$$PWM=[952,1150,1350,1450,1550,1650,1750,1900] \mu s.$$

Figure 3.12 shows the average of *RPM/V* for each PWM acquired during the test (see Appendix A.5). In this figure are also shown the regression line of each motor, where the first point has not been considered. These linear regressions are written a continuation, where it is important notice that the coefficient of determination is very high, so we can confirm that the relationship between PWM and *RPM/V* is linear:

$$Motor\ 1 \equiv (RPM/V)_1 = 0,5272 \cdot PWM - 364,17; R^2 = 0,9987$$

$$Motor\ 2 \equiv (RPM/V)_2 = 0,5003 \cdot PWM - 331,63; R^2 = 0,9976$$

$$Motor\ 3 \equiv (RPM/V)_3 = 0,5180 \cdot PWM - 341,19; R^2 = 0,9932$$

$$Motor\ 4 \equiv (RPM/V)_4 = 0,5250 \cdot PWM - 360,84; R^2 = 0,9986$$

To validate this model, a comparison has been done between experimental *RPM* data and *RPM* estimated from the former regression lines and the experimental voltages measured in each measurement of the *RPMs*. The comparison of all motors has been done through the average of the relative error for each PWM. The results are shown in Figure 3.13.

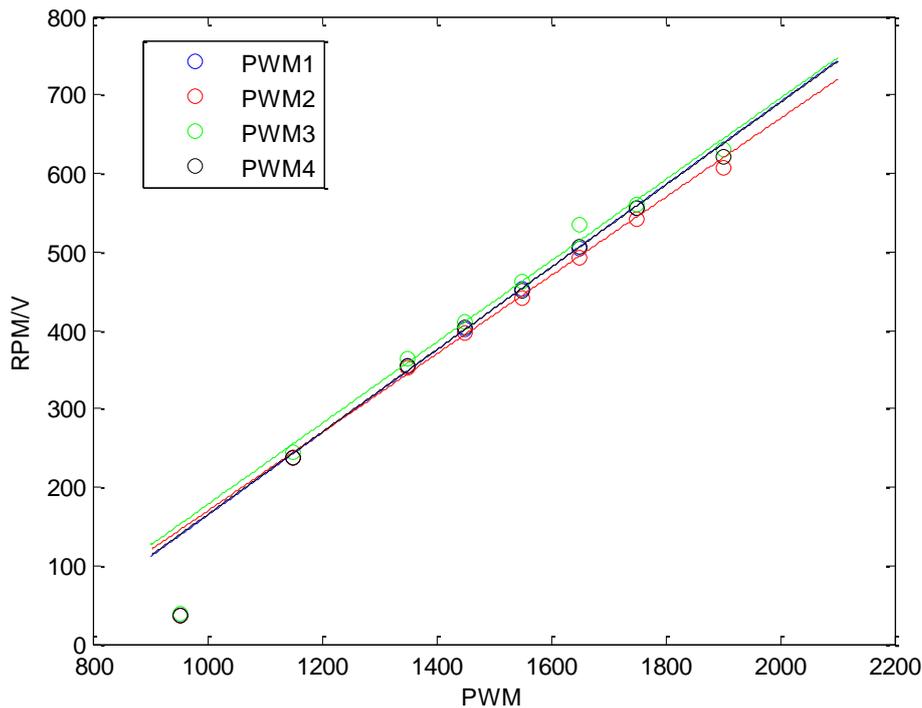


Figure 3.12: RPM/V vs PWM All Motors

For all motors, the average of the relative error of the estimation around the point of operation is under 1%. In fact, it is not never up to 4,5% in all PWM range. Thus, it is possible to conclude that the curve  $PWM - RPM/V$  is a good approximation for the performance of the motors.

### 3.2.6 BEC influence

All test mentioned until now have been carried being the BEC of the motor under study the one which powered the ArduPilot. In a real situation only one motor is powering the Ardupilot. So it is interesting to know if powering the ArduPilot has some negative effect over the performance of the motor.

In order to resolve this question, another test was done where two motors were connected to the PDB because one motor have to be always powering the ArduPilot. So, another motor was powering the ArduPilot while the motor 3 was working with the BEC disconnected. The Figure 3.14 shows the curve  $RPM/V$  versus Time for the motor 3 with BEC connected to the PDB and disconnected for the PWM signal of  $1450 \mu s$ . The values with BEC connected are the same as the ones obtained in section 3.2.4.

The curve with the BEC disconnected has a higher value of  $RPM/V$ . However, the maximum relative difference between both situations is 1,19% and the average of the relative differences is 0,82%. Since the differences are not very high, it is possible to neglect the fact of being or not being the motor powering the Ardupilot.

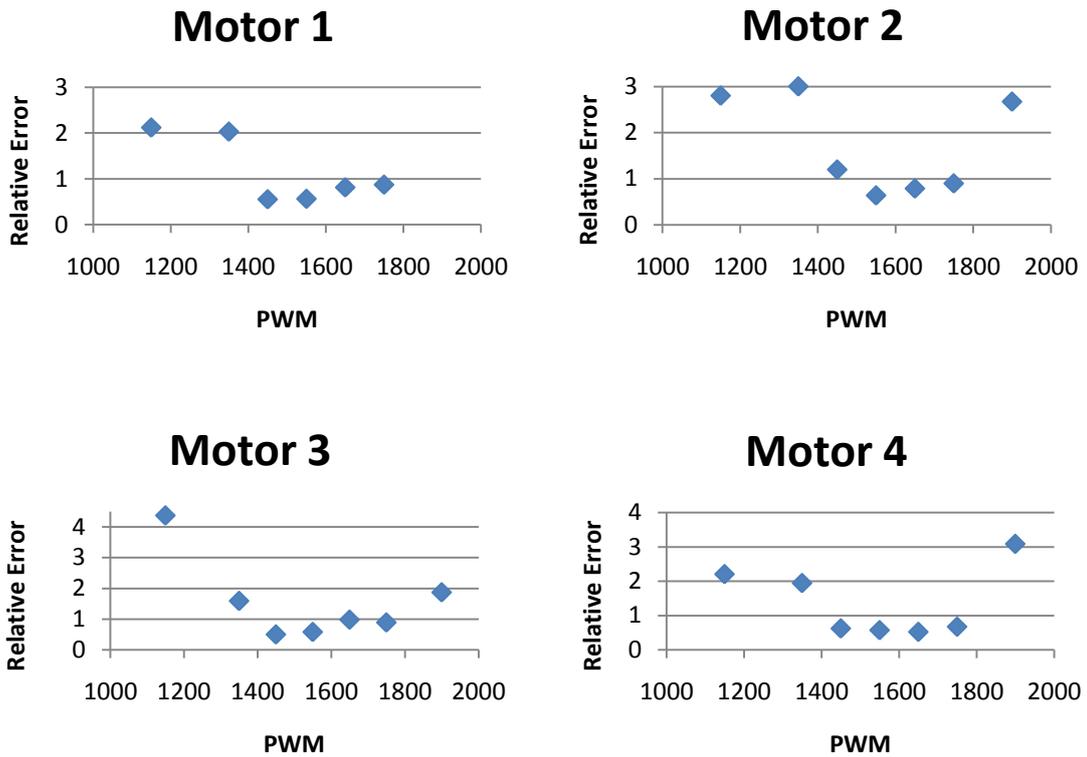


Figure 3.13: Relative Error All Motors

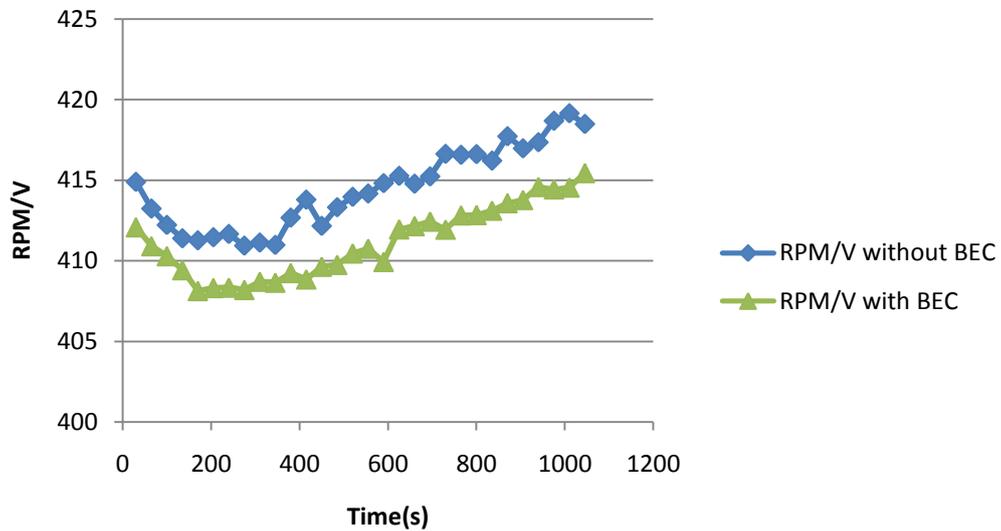


Figure 3.14: BEC vs no BEC supply

### 3.2.7 Flight Time

When Fernandes sized the battery, he considered an endurance of 10 min [14]. In fact he estimated the intensity that it was going to be required in hover and then the minimum capacity required for the battery choice. He considered an intensity of 18,805A and an endurance of 10 min:

$$Cap_{min} = I \times Endurance = 18805mA \times \frac{10min}{60} = 3134mAh \quad (3.7)$$

However, the Battery chosen has a  $Cap = 5000mAh$ , so with the intensity estimated by Fernandes, the final Endurance in hover would be:

$$Endurance = \frac{Cap}{I} = \frac{5000mAh}{18805} \cdot 60min = 15,95 min \quad (3.8)$$

To corroborate this result, a test of endurance has been realized. In this test a PWM signal of  $1500 \mu s$  has been considered as representative of the signal that is going to be sent to all motors in a hover situation during all the time. Figure 3.15 shows the evolution of the battery voltage during the test:

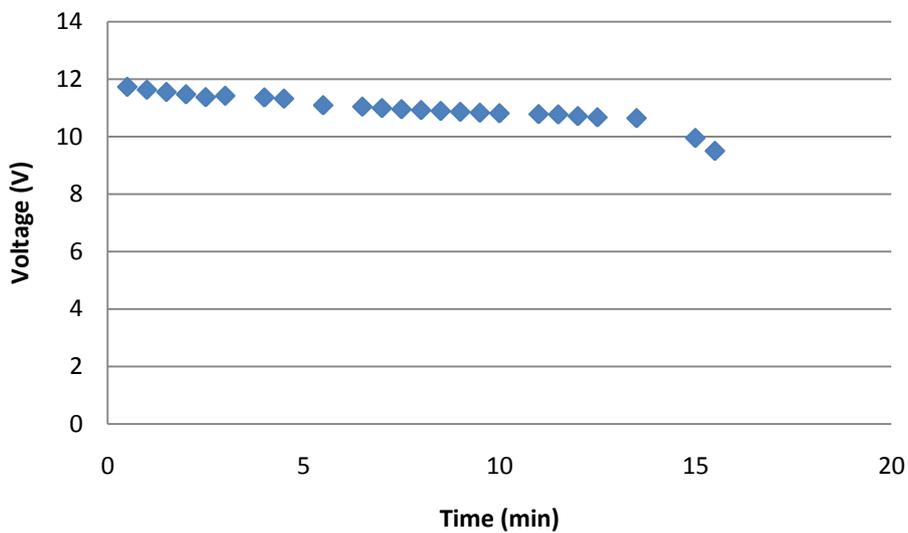


Figure 3.15: Endurance

During the test some problems took place. So this result is not really accurate, but gives as an idea of the performance of the battery in hover and the endurance. In this figure we can observe that from minute 14.5 the voltage of the battery suffers a great descend. In this test the servos were not connected and so the intensity required is going to be higher. Although this test give as an approximation of the real endurance, which is close to the expected, more tests to characterize the battery and have a better approximation of the flight time should be done in the future.

### 3.2.8 Propeller identification

Filipe Pedro did a study of the aerodynamics of different rotors and developed an algorithm to design the optimum rotor [17]. Fernandes employed this algorithm to design the optimum rotor [14]. However it has not built yet, and there were not stock of the rotors studied by Filipe Pedro. The available propellers were a two blade propeller not studied by Filipe Pedro, and so an identification of the characteristics of this propeller has been done.

Normally, in propellers used in aeromodelism the extrados is marked, the intrados is completely smooth and the leading edge is flat while the trailing edge is curve. However, in this propeller it is different. If we place the propeller on the motor with the surface marked upwards and the direction of rotation to generate lift, the leading edge is the curve one. So a huge confusion was generated and a test has been done to confirm which one is the leading edge and the extrados.

In tests the logic is the inverse to the flight situation because we must to direct the wake upwards in order to take accurate force measurements. So, it was tested what happens when the extrados is considered the smooth surface and then the leading edge is the curve one; and what happens when the extrados is considered the marked surface and then the leading edge is the flat one. For the PWM signals of  $1450 \mu s$  and  $1550 \mu s$  the results are:

		smooth surface	marked surface
$1450 \mu s$	<i>RPM</i>	4725	4545
	<i>T(gr)</i>	357	155
$1550 \mu s$	<i>RPM</i>	5241	5042
	<i>T(gr)</i>	431	199

Table 3.4: Propeller Setup

It is clear that the propeller has a better performance when the extrados is the smooth surface directing the wake upwards. Thus, in a real situation the situation is the inverse. The extrados must be the marked surface and then the leading edge is the curve one.

But the important work in this section is to identify the thrust coefficient ( $C_T$ ) and the power coefficient ( $C_P$ ). These coefficients have been measured during the test to obtain the curve  $PWM - RPM/V$  (section 3.2.5). In Figure 3.17 and Figure 3.17 show the average of the coefficients of thrust and power respectively obtained for each pair motor-propeller, for each PWM signal.

In order to obtain the coefficients of each pair motor-propeller it has been taken the average of the results obtained near to the point of operation, i.e. the PWM signals  $1350 \mu s$ ,  $1450 \mu s$ ,  $1550 \mu s$ , and  $1650 \mu s$ . In the case of  $C_T$  of the motor 2, since the point corresponding to  $1550 \mu s$ , and  $1650 \mu s$  are away from they should be, it has been considered to the average of the points  $1350 \mu s$ ,  $1450 \mu s$  and  $1750 \mu s$ . Also in the case of  $C_T$  of the motor 3, since the point corresponding to  $1650 \mu s$  is away from where it should be, it has been considered the average of the points  $1350 \mu s$ ,  $1450 \mu s$ ,  $1550 \mu s$  and  $1750 \mu s$ . Theses averages are shown in the Table 3.5.

motor	$C_T$	$C_P$
1	0,0143	0,0033
2	0,0141	0,0035
3	0,0135	0,0032
4	0,0140	0,0033

Table 3.5:  $C_T$  and  $C_P$  All Motors

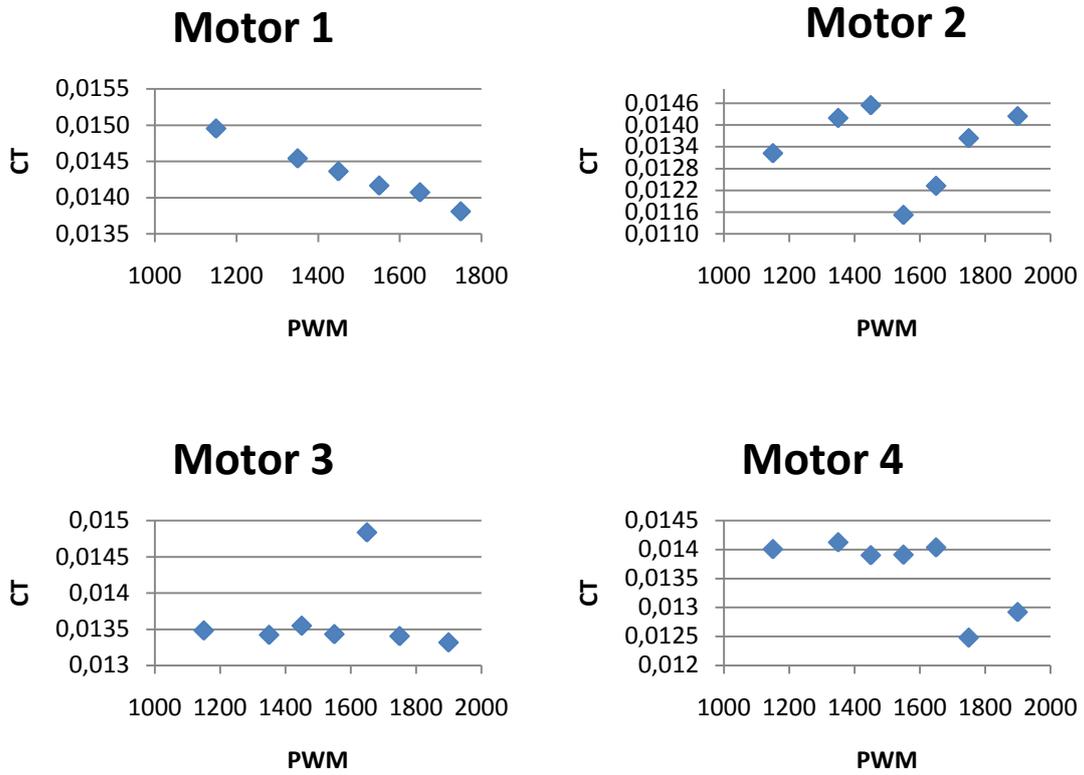


Figure 3.16:  $C_T$  All Motors

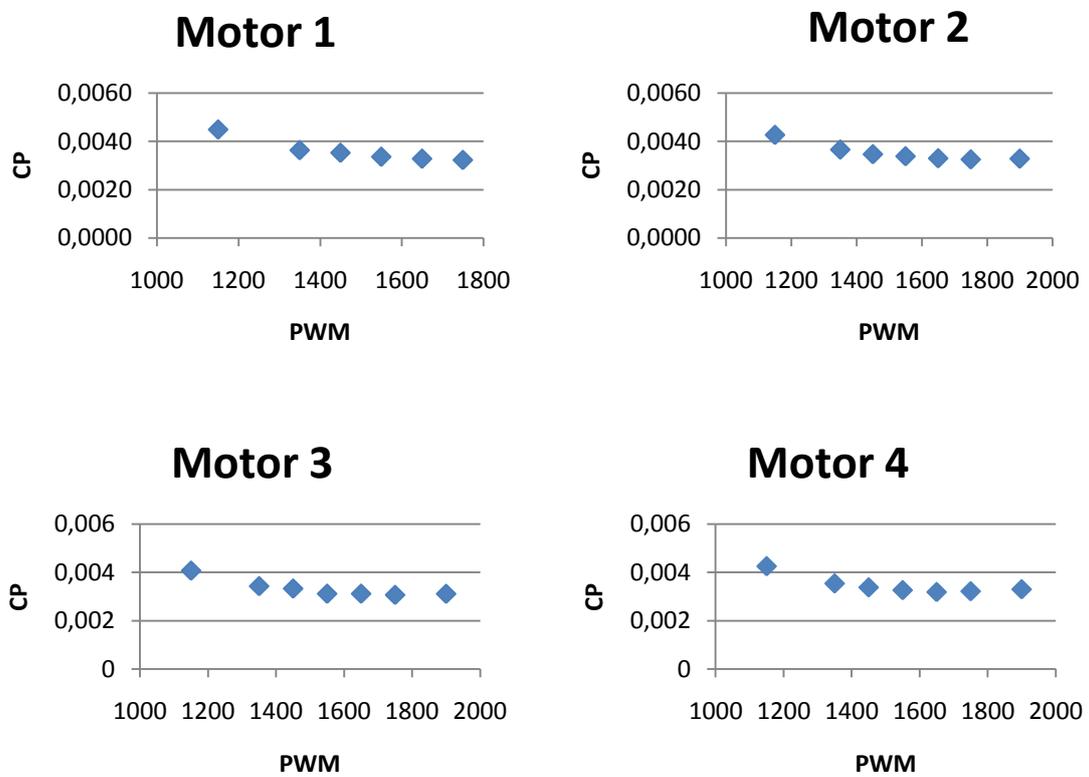


Figure 3.17:  $C_P$  All Motors

Knowing the coefficients of each propeller mounted to each motor, they can be included in the controller. However, if some propeller broke during testes or in flight, it should be changed and it should be necessary to obtain the coefficients of the propeller and include in the controller. Since we want the controller to be robust, the coefficients used correspond to the average of the four propellers.

The final averages of the coefficients are:

$$C_T = 0,0140$$

$$C_P = 0,0033$$

Since the radius of the propeller is  $r_r = 0,127$  m and the density of the air considered is  $\rho = 1,225$  kg/m<sup>3</sup>, we obtain:

$$K_T = 1,40 * 10^{-5} gr \cdot s^2$$

$$K_Q = 4,2 * 10^{-7} gr \cdot m \cdot s^2$$

### 3.3 Sensors Identification

In this chapter, a short description of the different sensors chosen by Fernandes is given. Then the noise associated to the sensors is measured on the prototype for greater accuracy of the simulations.

#### 3.3.1 Characteristics of the sensors

All sensor used to control the attitude of the ALIV3 are integrated in the IMU shield of the APM1 board. This board contains gyroscopes and accelerometers in the 3-axis, a magnetometer also in the 3-axis and a barometer. The characteristics of all these sensors are presented:

##### A) Accelerometers

<b>XL 335B</b>	<u>Min</u>	<u>typ</u>	<u>max</u>
Measured range	$\pm 3 g$	$\pm 3.6 g$	
Sensitivity	$270 mV/g$	$300 mV/g$	$330 mV/g$
Noise Xout, Yout		$150 \mu g/\sqrt{Hz} rms$	
Noise Zout		$300 \mu g/\sqrt{Hz} rms$	
Resolution		$0.02395 m/s^2$	

Table 3.6: Accelerometers Characteristics

Table 3.7 shows the characteristics of the accelerometers. Because the analog signal from the sensor is converted to a digital signal, there is an imposed resolution associated to the number of bits of the processor. In this case, the analog signal is converted to a digital signal with  $2^{12} = 4096$  intervals. As a consequence, to calculate the resolution, the reference voltage of 3,3V must be divided by the sensitivity and the number of intervals. The resolution of the accelerometers is:

$$Accres = \frac{3,3V \cdot 9,81 \text{ m/s}^2}{4096 \cdot 0,3 \text{ V}} = 0,02395 \text{ m/s}^2 \tag{3.9}$$

**B) Gyroscopes**

The IMU shield employs two types of gyroscopes, one for the x and y axis, the IDG 500 and another one for the z axis, the ISZ 500. Table 3.8 shows the characteristic of the IDG 500 gyroscope and Table 3.9 shows the characteristics of the ISZ 500.

<b>IDG 500</b>	
Measured range	$\pm 500 \text{ }^\circ/s$
Sensitivity	$2,0 \text{ mV}/(^\circ/s)$
Noise	$0,8 \text{ mV rms}$
Bias	$2,8 \text{ }(^\circ/s)/s$
Bias varying rate	$0,1 \text{ Hz}$
Resolution	$0,4028 \text{ }(^\circ/s)$

Table 3.7: x,y Gyroscopes Characteristics

$$Gyrores_{x,y} = \frac{3,3V}{4096 \cdot 2,0 \text{ mV}/(^\circ/s)} = 0,4028 \text{ }(^\circ/s) \tag{3.10}$$

The resolution is calculated following the same idea as in the accelerometer, and the Bias is obtained according to [8].

<b>ISZ- 500</b>	
Measured range	$\pm 500 \text{ }^\circ/s$
Sensitivity	$2,0 \text{ mV}/(^\circ/s)$
Noise	$0,5 \text{ mV rms}$
Bias	$2,8 \text{ }(^\circ/s)/s$
Bias varying rate	$0,1 \text{ Hz}$
Resolution	$0,4028 \text{ }(^\circ/s)$

Table 3.8: z Gyroscope Characteristics

$$Gyrores_z = \frac{3,3V}{4096 \cdot 2,0 \text{ mV}/(^\circ/s)} = 0,4028 \text{ }(^\circ/s) \tag{3.11}$$

### C) Magnetometer

Table 3.10 shows the characteristic of the Magnetometer included in the IMU shield, the Honeywell HMC5843.

<b>HMC5843</b>	
Measured range	360°
Resolution	7 milli – gauss
Signal to Noise Ratio	70 dB

Table 3.9: Magnetometer Characteristics

### D) Barometer

Table 3.11 shows the characteristic of the Magnetometer included in the IMU shield, the Bosch BMP085.

<b>BMP085</b>	
Measured range	300...1100 hPa (+9000...-500 m above sea level)
Noise	0.05 hPa/0.4m
Resolution	0.01 hPa/0.08m

Table 3.10: Barometer Characteristics

## 3.3.2 Experimental identification of the sensors

The measurement of the sensors noise is performed in this section in order to be included in the simulation for a better approximation of the platform. Because the measurements of the sensors are also affected by the vibrations of the motors, the noise of the gyroscopes, accelerometer and magnetometer were obtained both when motors were running and when motors were stopped. With this test the influence of the vibrations of the motors are estimated. The noise of the sensors is also compared to the theoretical noise specified in the respective datasheet.

The spectral power of the sensors measurements is given by:

$$Pow = \sigma^2(1/freq) \quad (3.12)$$

where  $Pow$  is the spectral power,  $\sigma$  the standar deviation and  $freq$  is the sampling frequency.

The reading of the accelerometer is affected by its location relative to the center of mass as it has been explained in section 2.4.2. It is considered that all sensors gathered in the IMU shield are located at a height of  $r_z = -3.2cm$ . In the case of the accelerometer, its location is:  $r_{acc} = [0,0,-3.2]cm$ .

### A) Accelerometers

Components	Spectral Power $(m/s^2)^2/Hz$		
	Theoretical	Motors off	Motors on
x	$2.1653 \times 10^{-6}$	$3.3103 \times 10^{-6}$	0.0212
y	$2.1653 \times 10^{-6}$	$2.7038 \times 10^{-6}$	0.013
z	$8.6612 \times 10^{-6}$	$1.0384 \times 10^{-5}$	0.0186

Table 3.11: Accelerometer Noise Measurements

Table 3.12 shows the spectral power of accelerometer noise measurements in  $(m/s^2)^2/Hz$ . Note that units have been converted to  $m/s^2$  instead of  $g$ . It can be observed that theoretical values and the values obtained with the motors turned off are similar. However, when the motors are turned on, the spectral power increases a lot.

### B) Gyroscopes

Components	Spectral Power $(^\circ/s)^2/Hz$		
	Theoretical	Motors off	Motors on
x	$3.2 \times 10^{-3}$	$3.1921 \times 10^{-4}$	0.2024
y	$3.2 \times 10^{-3}$	$6.6720 \times 10^{-4}$	0.0799
z	$1.25 \times 10^{-3}$	$5.5948 \times 10^{-4}$	0.1009

Table 3.12: Gyroscopes Noise Measurements

Table 3.13 shows the spectral power of gyroscopes noise measurements in  $(^\circ/s)^2/Hz$ . Note that in gyroscope's datasheet the units of the noise are given in *rms*. So they must be translated to  $^\circ/s/\sqrt{Hz}$  with the help of the sensitive and the sampling frequency. It can be observed that theoretical values are smaller than values obtained with the motors turned off. However, when the motors are turned on, the spectral power increases even more.

### C) Magnetometer

Spectral Power $(deg)^2/Hz$		
Theoretical	Motors off	Motors on
—	0.0023	0.0500

Table 3.13: Magnetometer Noise Measurements

Table 3.14 shows the spectral power of the magnetometer noise measurements in  $(deg)^2/Hz$ . The theoretical value is given in the datasheet by Signal to Noise Ratio in dB, so the Spectral Power in  $(deg)^2/Hz$  cannot be obtained. It can be observed that the value obtained with the motors turned off is slightly smaller than the value obtained when the motors are turned on.

### D) Barometer

Spectral Power $(Pa)^2/Hz$		
Theoretical	Motors off	Motors on
0.5	0.1663	0.4269

Table 3.14: Barometer Pressure Noise Measurement

Spectral Power $(m)^2/Hz$		
Theoretical	Motors off	Motors on
0.0032	0.0012	0.0030

Table 3.15: Barometer Altitude Noise Measurement

Table 3.15 shows the spectral power of the barometer noise measurements in  $(Pa)^2/Hz$  and Table 3.16 shows the spectral power of the barometer noise measurements in  $(m)^2/Hz$  (once the altitude is accessed). Note that theoretical values and the values obtained with the motors turned off and turned on do not differ much.

## Chapter 4

# Tilt-Quadrotor Flight Control

This Chapter presents the simulation of the tilt-quadrotor flight attitude control considering the ALIV3 without tilting any rotor. The approachment followed is to achieve the stabilization of the Tilt-Quadrotor in a first step like a standard quadrotor. Once this goal will be accomplished, it would be considered the Flight Control including the servos as actuators. That is why the servo's identification and modelling has not been done. Then, the stabilization is achieved with the 4 PWM signals of the motors as input of the system.

In Section 4.1 the control method is presented and the simulation of the full system is performed. In Section 4.2 the estimation method is described. Finally, in Section 4.3 the simulations results are presented for two different cases. First, the ideal response of the system reduced to 6 states is performed, and finally the real response of the discrete 6 states with feedback estimation is accessed.

### 4.1 Control Method

In section 4.1.1 is presented the open loop response of the system. There, is evidenced the needed of a control method. The necessary control system for the ALIV3 works with a state vector of 12 components  $\mathbf{X} = [\mathbf{V}^T, \mathbf{\Omega}^T, \mathbf{P}^T, \mathbf{\Phi}^T]^T$  and an input vector of 4 components  $\mathbf{U} = [PWM_1, PWM_2, PWM_3, PWM_4]^T$ . As input components only the motors are going to be considered to stabilize the ALIV3. The control including the servos would be addressed in following works.

There are two main methods of control implemented in quadrotors: one based on a PID controller, and another one based on LQR controller. In [1], Henriques did the study of which one was the best. He defended that the LQR is faster than the PID because the LQR is an optimal controller presented in the form of a matrix and its implementation is easier and less computationally demanding than the PID method. Therefore, the LQR is the chosen control method.

### 4.1.1 Open Loop Response

Before describe the control method, it is shown the response of the system to the open loop in Figure 4.1. It can be noted that when the system is left alone from the initial conditions set corresponding to the hovering situation, the quadrotor diverges quickly from the hovering state. That reflects the necessity of a control method to stabilize the attitude of the ALIV3.

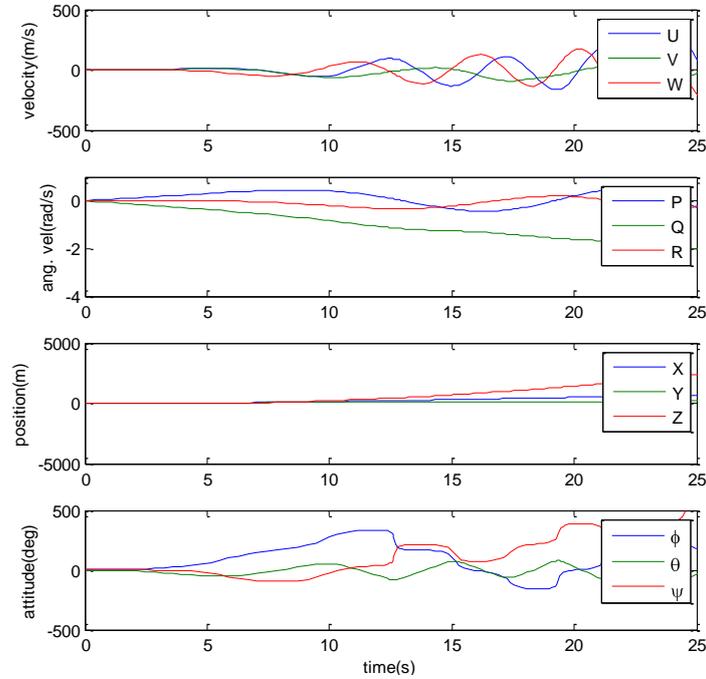


Figure 4.1: Open Loop response

### 4.1.1 LQR Control Description

The model of the system can be reduced to two functions which satisfy the general nonlinear state space formulation:

$$\dot{\mathbf{X}} = g(\mathbf{X}, \mathbf{U}) \quad \text{Dynamic Equation} \quad (4.1a)$$

$$\mathbf{Y} = h(\mathbf{X}, \mathbf{U}) \quad \text{Output Equation} \quad (4.1b)$$

where  $\mathbf{X} = [\mathbf{V}^T, \mathbf{\Omega}^T, \mathbf{P}^T, \mathbf{\Phi}^T]^T$  is the state vector of the system;  $\mathbf{U} = [PWM_1, PWM_2, PWM_3, PWM_4]^T$  is the input vector of the system; and  $\mathbf{Y} = [\bar{\mathbf{a}}^T, \bar{\mathbf{\Omega}}^T, \bar{\mathbf{N}}^T, \bar{\mathbf{Z}}, \bar{\mathbf{W}}]^T$  is the output vector of the system and obtained from the sensors.

Considering the generic linear state space model:

$$\dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{Bu} \quad (4.2a)$$

$$\mathbf{y} = \mathbf{Cx} + \mathbf{Du} \quad (4.2b)$$

The control problem consists in finding the optimal control action  $\mathbf{u} = -\mathbf{K}_{lqr}\mathbf{x}$  where  $\mathbf{K}_{lqr}$  is a simple matrix gain given by:

$$\mathbf{K}_{lqr} = \mathbf{R}_{lqr}^{-1}\mathbf{B}^T\mathbf{P} \quad (4.3)$$

and  $\mathbf{P}$  is found by solving the algebraic Riccati equation:

$$\mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} - \mathbf{P}\mathbf{B}\mathbf{R}_{lqr}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q}_{lqr} = 0 \quad (4.4)$$

The matrix gain is chosen by the criteria of minimize the following cost function:

$$J_{LQR} = \int_0^{\infty} (\mathbf{x}^T\mathbf{Q}_{lqr}\mathbf{x} + \mathbf{u}^T\mathbf{R}_{lqr}\mathbf{u})dt \quad (4.5)$$

where  $\mathbf{Q}_{lqr}$  and  $\mathbf{R}_{lqr}$  are the weighting matrices. They are initially determined by the Bryson's rule which say that  $\mathbf{Q}_{lqr}$  and  $\mathbf{R}_{lqr}$  can be written as diagonal matrices as:

$$Q_{ii} = \frac{1}{\text{maximum acceptable value of } x_i^2} \quad (4.6)$$

$$R_{ii} = \frac{1}{\text{maximum acceptable value of } u_i^2} \quad (4.7)$$

The dynamic closed loop is given by:

$$\dot{\mathbf{x}} = (\mathbf{A} - \mathbf{B}\mathbf{K}_{lqr})\mathbf{x} \quad (4.8)$$

Thus, it is necessary to carry out a linearization of the system around the operation point, which in this case is when the ALIV3 is hovering. A linearization consists in approximating the nonlinear state space by a linear formulation:

$$\dot{\hat{\mathbf{X}}} \approx \mathbf{A}\hat{\mathbf{X}} + \mathbf{B}\mathbf{U} \quad (4.9a)$$

$$\mathbf{Y} \approx \mathbf{C}\hat{\mathbf{X}} + \mathbf{D}\mathbf{U} \quad (4.9b)$$

where  $\hat{\mathbf{X}} = \mathbf{X} - \mathbf{X}_0$ . This approximation is only valid in the vicinity of the operating point  $\mathbf{X}_0$ . This linearization has been achieved by a computationally method and tested analytically. The linearization matrices are shown in the Appendix C. The operation point considered is:

$$\mathbf{X}_0 = [0,0,0,0,0,0,0,0,-1,0,0,0]$$

corresponding to when the quadrotor is hovering at a height of 1 meter.

### 4.1.2 Observability and Controllability

A twelve-state LQR is used in this section to close the loop and ensure that the quadrotor follows the reference. In this first step of control development, ideal sensors are used as output vector.

Firstly, it is necessary to study the system controllability and observability. This study is achieved by checking the controllability and observability matrices. According to [7], one system is fully controllable if all states are controllable. The Matrix of controllability::

$$[B \ AB \ \dots \ A^{n-1}B] \quad (4.10)$$

where  $n$  is the number of states. So, the system is fully controllable if the rank of the controllability matrix is equal to  $n$ . In respect to the observability, one system is fully observable if all elements of the output vector are affected eventually by any transition of the state vector. One system is fully observable if the rank of the observability matrix:

$$\begin{bmatrix} C \\ \dots \\ CA \\ \dots \\ \vdots \\ \dots \\ CA^{n-1} \end{bmatrix} \quad (4.11)$$

is equal to  $n$ , where  $n$  is the number of states.

Regarding the ALIV3 model, the rank of the controllability matrix is 10 and the rank of the observability matrix is 8, so there are 2 magnitudes that are not controllable and 4 that are not observable.

Through the Kalman decomposition [12] it is possible to separate the reachable subspace and its complement combined with the unobservable subspace and its complement. Doing it, it is possible to know that the subspace which is fully controllable and observable has a dimension of 6. The magnitudes that are not controllable are  $\psi$  and  $R$ , which is coherent with the analytic study performed in Chapter 2. There, it was observed that the yaw motion cannot be achieved through the standard form, it is necessary to tilt the rotors. The magnitudes unobservable are  $X, Y, U$  and  $V$ . In following works it must be introduced the GPS to get the system fully observable. In section 4.3 the system will be reduced to a 6 states system.

### 4.1.3 Ideal LQR Full 12 States.

In the simulation of the ideal LQR full 12 States a reference set, which is shown in Figure 4.2, is used to test if the system follows it. Linear velocities are expressed in  $m/s$ , angular rates in  $rad/s$ , positions in  $m$  and angles in  $deg$ . Figure 4.3 shows the closed loop implementation.

The weighting matrices are defined in equation 4.6 and 4.7. Instead of using the Bryson's rule to start the iteration process, the weighting matrices of Henriques's [1] work have been employed since the system is similar. The angles are in *rad* and the angular velocities in *rad/s*:

$$Q_{lqr} = \text{diag}([5,5,10,100,100,50,4,4,30,150,150,150]) \quad (4.12)$$

$$R_{lqr} = \text{diag}([0.01,0.01,0.01,0.01]) \quad (4.13)$$

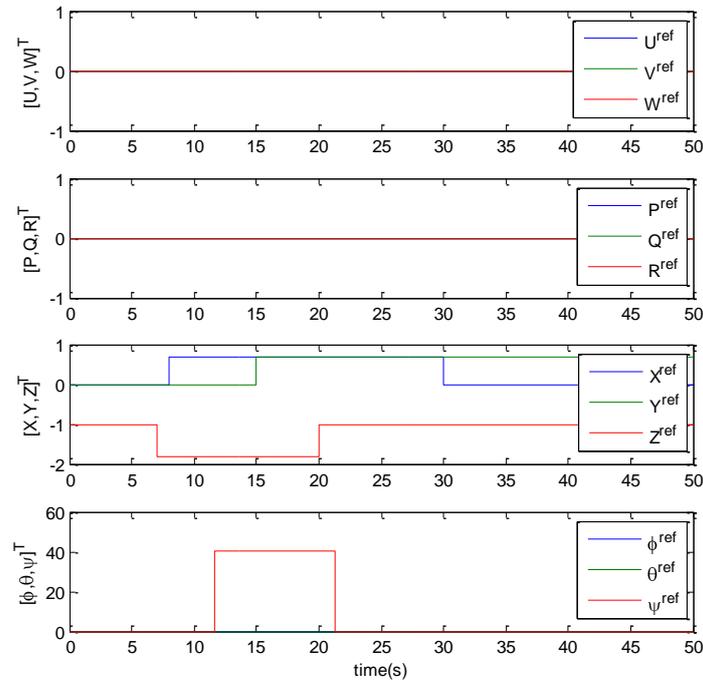


Figure 4.2: Reference Set

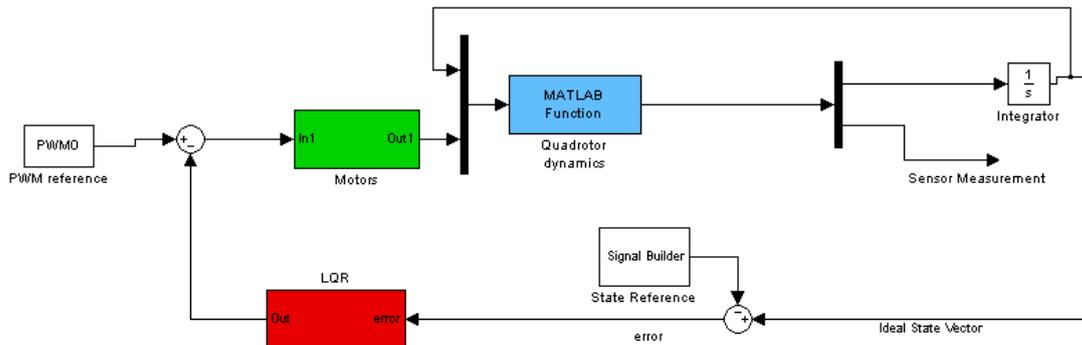


Figure 4.3: Closed loop implantation

The system is not fully controllable, but the controller gets to stabilize it. To analyze the closed loop dynamic, the poles of the system expressed in equation (4.8) are obtained. Table 4.1 presents the location of the closed loop poles and Figure 4.5 presents the poles in the Argand plane.

Real	-15.9	-11.1	-1.09	-1.09	-1.09	-1.09
Imaginary	0	0	9.42	-9.42	0.945	-0.945
Real	-0.95	-0.95	-0.55	-0.55	2.19e-17	2.19e-17
Imaginary	0	0	0.50	-0.50	1.97e-10	1.97e-10

Table 4.1: Closed loop poles [rad/s]

As can be observed, all poles have a negative real part (on the left half of the Argand plane) but two poles are placed very near to zero with a positive real part. This means that the system is marginally stable. Figure 4.4 shows the response of the system to the reference set. It can be seen that  $X, Y$  and  $Z$  follow the reference, however  $\psi$  does not because as it is already said, this state cannot be controlled.

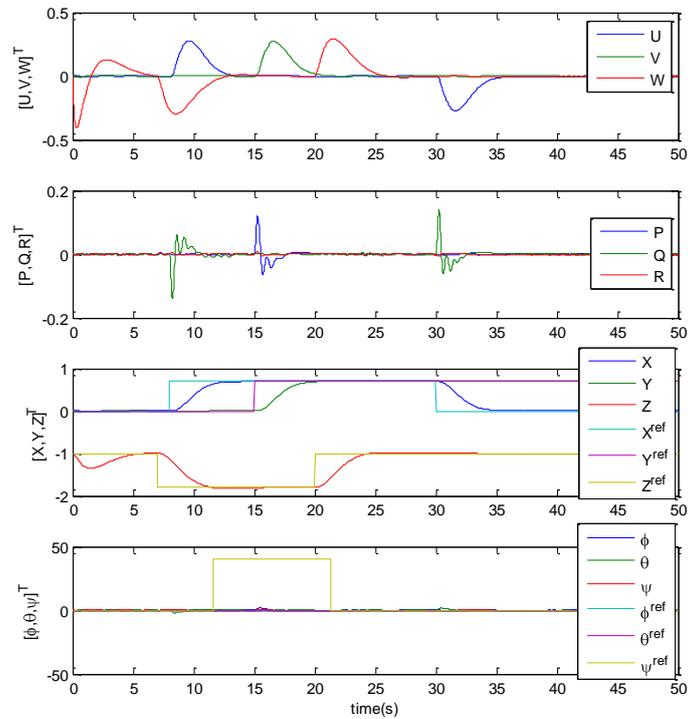


Figure 4.4: Response of the system to the reference set

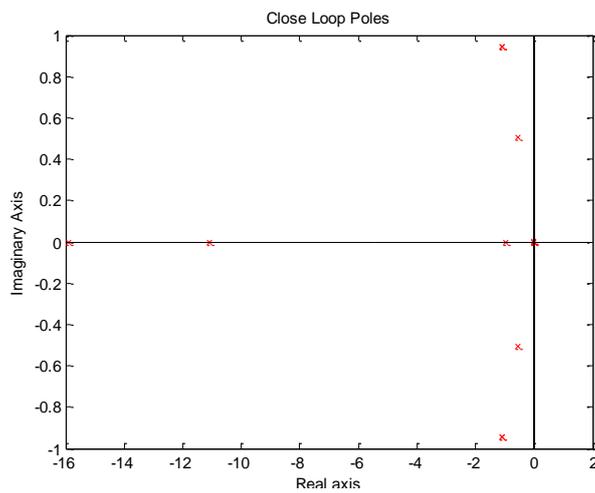


Figure 4.5: Close Loop Poles

## 4.2 Estimation Method

The information provided by the sensors is not completely useful. It has inaccuracies as noise and bias. So, it is necessary to include an estimator in order to extract the information from the sensors, filtering these perturbations.

There are many types of estimators, but in [1] Henriques did a discussion of which estimator achieved the best results. He studied different Complementary Filters and Kalman filters, resulting that the EKF (Extended Kalman Filter) was the best option. He said that EKF filters out slightly less noise than the other Kalman filters, but it reaches a better stationary estimation. It is also computationally more efficient than the complementary filter with bias estimation and it is able to overcome the linearization limits. Therefore, the filter that is going to be implemented in this work is the EKF.

### 4.2.1 Kalman Filter

The Kalman Filter is an algorithm that works in two steps. First it acts as a predictor: the Kalman filter predicts the future state considering the covariance error with the help of the model of the system, the current state and the input vector. Then in the second step it corrects the predicted state and the estimated covariance error according to the measurements and the noise covariance. The Kalman filter is optimal in the sense that it minimizes the estimated covariance error.

The main assumption of the Kalman filter is that the system is a linear dynamical system. Thus, it is necessary to carry out a linearization of the system around the operation point. However, the APM board runs at a frequency  $freq = 50Hz$ , what means that the observable states will only be reconstructed at intervals of  $T = 0.02s$ , and so the system must be also discretized:

$$\mathbf{X}_{k+1} \approx \mathbf{A}_d \mathbf{X}_k + \mathbf{B}_d \mathbf{U}_k \quad (4.14a)$$

$$\mathbf{Y}_k \approx \mathbf{C}_d \mathbf{X}_k + \mathbf{D}_d \mathbf{U}_k \quad (4.14b)$$

where  $\mathbf{A}_d$ ,  $\mathbf{B}_d$ ,  $\mathbf{C}_d$ ,  $\mathbf{D}_d$  are the discrete state space matrices linearized;  $\mathbf{X}_k$ ,  $\mathbf{U}_k$ ,  $\mathbf{Y}_k$  are respectively the state vector, the input vector and the output vector at time  $k$ ; and  $\mathbf{X}_{k+1}$  is the state vector at time  $k + 1$ . The discretization process of the matrices followed is:

$$\mathbf{A}_d = e^{\mathbf{A}T} \quad (4.15a)$$

$$\mathbf{B}_d = \mathbf{A}^{-1}(\mathbf{A}_d - \mathbf{I})\mathbf{B} \quad (4.15b)$$

$$\mathbf{C}_d = \mathbf{C} \quad (4.15c)$$

$$\mathbf{D}_d = \mathbf{D} \quad (4.15d)$$

Returning to Kalman's algorithm, consider the following discrete linear system:

$$\mathbf{X}_{k+1} = \mathbf{A}_d \mathbf{X}_k + \mathbf{B}_d \mathbf{U}_k + \mathbf{w}_k \quad (4.16a)$$

$$\mathbf{Y}_k = \mathbf{C}_d \mathbf{X}_k + \mathbf{D}_d \mathbf{U}_k + \mathbf{v}_k \quad (4.16b)$$

where  $\mathbf{w}_k$  is the process noise which is assumed to be drawn from a zero mean multivariate normal distribution with covariance  $\mathbf{Q}_k$  and  $\mathbf{v}_k$  is the measurement noise which is assumed to be zero mean Gaussian white noise with covariance  $\mathbf{R}_k$ :

$$\mathbf{w}_k \sim N(0, \mathbf{Q}_k) \quad (4.17a)$$

$$\mathbf{v}_k \sim N(0, \mathbf{R}_k) \quad (4.17b)$$

The matrices  $\mathbf{R}_k$  and  $\mathbf{Q}_k$  are diagonal matrices with values corresponding to the noise characteristics of the sensors. Because the Kalman filter is going to be implemented in discrete mode, the discrete matrices are:

$$\mathbf{R}_d = \mathbf{R}_k / T \quad (4.18a)$$

$$\mathbf{Q}_k = \int_{\tau=0}^T e^{\mathbf{A}\tau} \mathbf{Q}_k e^{\mathbf{A}^T \tau} d\tau \quad (4.18b)$$

## 4.2.2 Extended Kalman Filter

The modification of the basic Kalman Filter that is going to be implemented is the Extended Kalman Filter (EKF). In this version, the non-linearities of the system are approximated by a linearized version of the non-linear system about the current estimation. Figure 4.6 presents the algorithm:

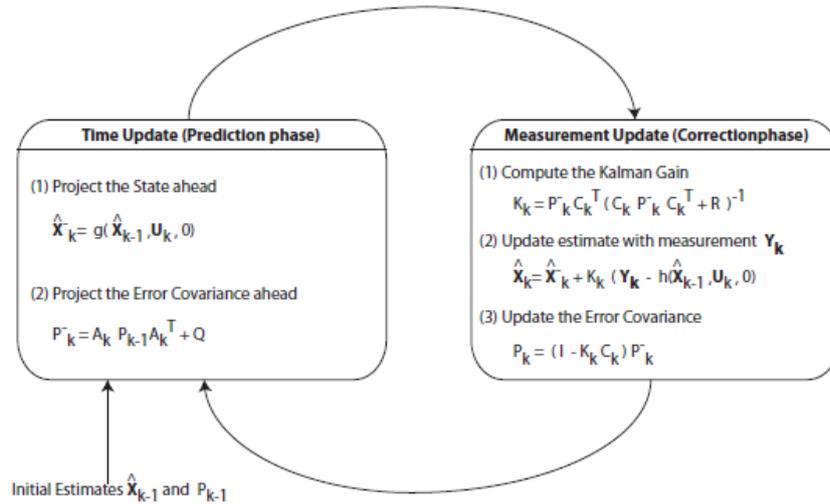


Figure 4.6: Extended Kalman Filter algorithm [1]

Instead of using the matrices  $\mathbf{A}_d$ ,  $\mathbf{B}_d$ ,  $\mathbf{C}_d$ ,  $\mathbf{D}_d$ , the nonlinear output and dynamic functions are used to linearize around the current state, resulting in the use of  $\mathbf{A}_k$  and  $\mathbf{C}_k$ , obtained at each iteration.

The improvement of the EKF in respect to the basic Kalman filter comes from integrate the gyroscopes to “predict” the evolution of the attitude and correct that prediction with the information

from the accelerometer and the compass. Then, instead of using the PWM signal as the input vector of the Kalman filter, one can write the system taking the gyroscopes as input vector. As the system is going to be reduced to a 6 state space system because only 6 states are fully controllable and observable, the yaw angle  $\psi$  and the angular velocity about the z-axis  $R$  are also removed in the Kalman filter:

$$\dot{\mathbf{X}} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \mathbf{X} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \mathbf{U} \quad (4.19)$$

where  $\mathbf{X} = [\phi, \theta]^T$  and  $\mathbf{U} = [P, Q]^T$ .

Then, matrices  $\mathbf{C}$  and  $\mathbf{D}$  have also to be adapted to the new system:

$$\dot{\mathbf{Y}} = \begin{bmatrix} 0 & -9.81 \\ 9.81 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \mathbf{X} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \mathbf{U} \quad (4.20)$$

where  $\mathbf{Y} = [\bar{a}_x, \bar{a}_y, \bar{g}_x, \bar{g}_y]^T$ .

This approach leads to a benefit in the robustness of the estimation, but only the estimation of the attitude is obtained. The output of the gyroscopes is used as direct measurement of the angular velocities.

Because the dynamic equation is already linear by nature (see equation 4.19), the computations are greatly simplified and the time update can be performed with  $\mathbf{A}_k = \mathbf{A}_d$  and  $\mathbf{B}_k = \mathbf{B}_d$  and the same to  $\mathbf{C}_k = \mathbf{C}_d$  and  $\mathbf{D}_k = \mathbf{D}_d$ . The following algorithm is the one used:

---

**Algorithm 4.1** Estimation using the EKF

---

$$\mathbf{Y} = [\bar{a}_x, \bar{a}_y, \bar{g}_x, \bar{g}_y]$$

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

defining the state trim condition for iteration k  $\mathbf{X}_k^0 = [\mathbf{0}, \mathbf{0}]^T$

defining the output trim condition for iteration k  $\mathbf{Y}_k^0 = [\mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}]^T$

Time update:

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

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

Measurement Update:

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

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

$$\mathbf{P}_k = (\mathbf{I}(3) - \mathbf{K}_k \mathbf{C}_k) \mathbf{P}_k^-$$


---

### 4.2.3 Height and Vertical Velocity Estimator

Because the height estimation is not the main concern of this work, it has not been achieved, and ideal values of the height have been used instead. In next works, both  $\psi$  and  $Z$  have to be estimated taking into account the noise of the Compass and the Barometer.

However, the vertical velocity has been estimated. There are two main forms. One is to derive the altitude barometer measurement, and the other is to integrate the accelerometer. The implemented estimator is based on the first approach. A high pass filter is used since it does not affect the high frequency variation like the vertical velocity and so those variations can be approximated to the differentiation [1]. For the high pass filter, a time constant of 0.005s has been used.

## 4.3 Simulation Results

In section 4.1 a controller based on the 12 states model considering ideal sensors has been developed. However, the model is not fully controllable and observable. In fact, to get a fully controllable and observable system it is necessary to remove 6 states. So, in this section it is performed a controller based on 6 states, where the state vector is  $X = [W, P, Q, Z, \phi, \theta]^T$  and the output vector is  $Y = [\bar{a}_x, \bar{a}_y, \bar{g}_x, \bar{g}_y, \bar{W}, \bar{Z}]^T$ . Firstly, the controller is designed in the ideal case, using the output vector without noise, and then the real case, using the data from the estimated values.

In following sections linear velocities are expressed in  $m/s$ , angular rates in  $rad/s$ , positions in  $m$  and angles in  $deg$ .

### 4.3.1 Ideal Simulation

#### A) Continuous Model

Considering the ideal case, the stabilization problem without tilting any rotor consist in keeping the tilt-quadrotor hovering at a certain altitude. So, the 6 states to be controlled are:  $X = [W, P, Q, Z, \phi, \theta]^T$ . The closed loop is written in equation (4.21). The space state matrices of the system are obtained eliminating from the 12 states model the columns and rows corresponding to the removed states. They are written in Apendix C:

$$\dot{X} = (A - B \cdot K_{lqr})X \quad (4.21)$$

The weighting matrices have been slightly modified in order to improve the response. The ones used are:

$$\mathbf{Q}_{lqr} = \text{diag}([50,100,100,100,250,250]) \quad (4.22)$$

$$\mathbf{R}_{lqr} = \text{diag}([0.01,0.01,0.01,0.01]) \quad (4.23)$$

In order to test the controller, three cases have been considered. First it has been considered a reference of  $\mathbf{X}_{ref} = [0,0,0,-1,0,0]^T$  and the initial conditions as the reference:  $\mathbf{X}_0 = [0,0,0,-1,0,0]^T$ . The second test is performed with the same reference and initial conditions but now, perturbations in the attitude have also been introduced. The third test is performed with the initial condition  $\mathbf{X}_0 = [0,0,0,-1,0,2,0.2]^T$ .

Figure 4.7 show the response of the quadrotor model to the first test and Figure 4.8 represents the third test, the response of the quadrotor to a perturbation. Both cases give a first idea about the stability and the performance of the controller. On a large scale, the controller is able to stabilize the ALIV3, even under perturbations, keeping it at the attitude and height ordered.

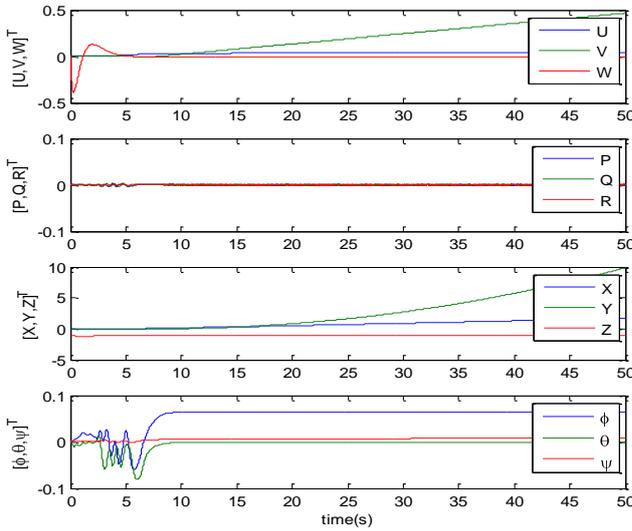


Figure 4.7: Response to the first test

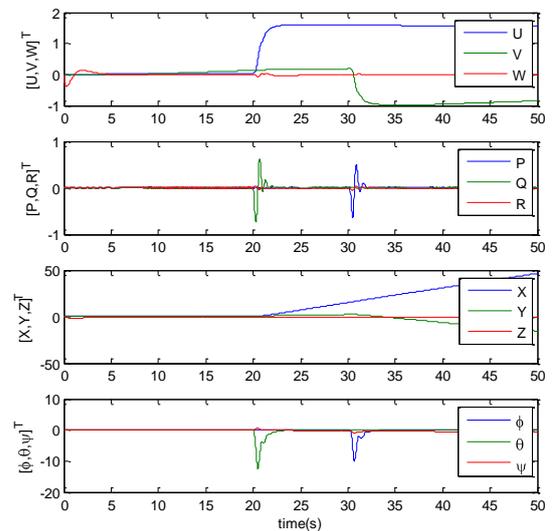


Figure 4.8: Response to the third test

However, in Figure 4.7 it can be seen that a small static error appears in the attitude. This small error is due to the nonlinearities of the model, like the quantizers of the motors. It can be also appreciated the uncontrolled position noticed by a random walk. The error in the attitude implies that the thrust force have components in the x and y axes, then the quadrotor is accelerated and it moves. Figure 4.10 shows the simulation without the quantizer in the block of the motors. The error in the attitude almost disappears and the random walk is reduced over ten times.

From Figure 4.9 it is possible to measure the rising time and the setting time. The rising time is the time required for the response to reach the 90% of its final value. The setting time is the time required for the response to reach stationary state, within 2% of the final value. Analyzing the curves, the rising time and the setting time of  $Z$ ,  $\phi$ , and  $\theta$  are measured and presented in Table 4.2:

	$Z$	$\phi$	$\theta$
--	-----	--------	----------

Rising Time (s)	2.60	1.52	1.54
Setting Time(s)	3.04	2.60	2.52

Table 4.2: Rising time and Setting time of the continuous model

Finally, analyzing the closed loop dynamic, all the poles are shown in Table 4.3 and they all have a negative real part. This proves the stability of the system:

Real	-15.80	-11.00	-0.80	-0.80	-1.60	-1.59
------	--------	--------	-------	-------	-------	-------

Table 4.3: Poles of the Discrete 6 States Model [rad/s]

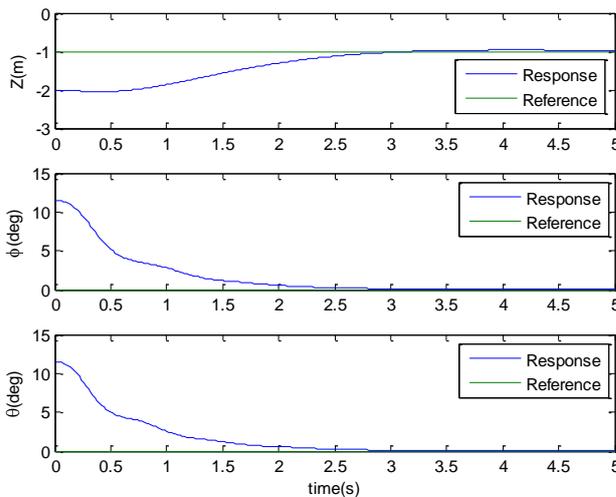


Figure 4.9: Response to the second test

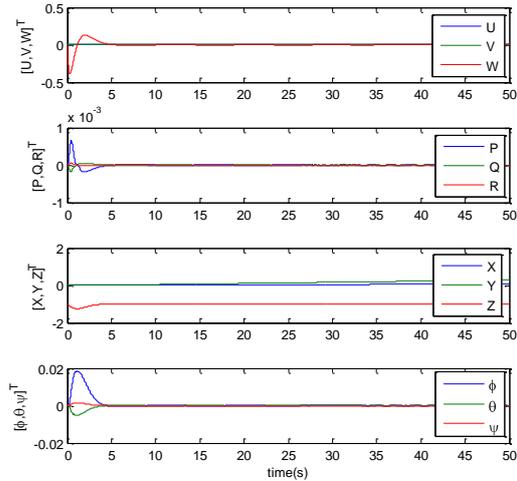


Figure 4.10: Response without the quantizer

## B) Discrete Model

Because the controller will be implemented in the APM, and the processor Atmel Mega 2560<sup>37</sup> operates at 50Hz, it is necessary to develop a discrete LQR controller. This is achieved by the function *lqrd()* in *Matlab*. It receives the weighting matrices, the state space matrices and the sampling time. Then, it performs the discretization of the state space matrices and the weighting matrices. The weighting matrices used in the following simulations are the same as in the continuous model.

In Figure 4.11 is shown the response of the discrete model to the set reference  $X_{ref} = [0,0,0 - 1,0,0]^T$  and the initial conditions equal to the set reference,  $X_0 = [0,0,0 - 1,0,0]^T$ . It can be seen that the means of the attitude angles are very similar to the continuous model but with a higher standard deviation. The random walk is also similar, because the random walk is caused by the error in the attitude angles. Table 4.4 presents the comparison of  $\phi$  and  $\theta$ , between the continuous model and the discrete model. There, it is proved that  $\phi$  is slightly smaller in the discrete model but  $\theta$  is ten times higher. However,  $\theta$  is not big enough to create large differences in the random walk.

<sup>37</sup> <http://www.atmel.com/devices/atmega2560.aspx>

	$\phi$			$\theta$		
	mse (deg) <sup>2</sup>	std (deg)	mean (deg)	mse (deg) <sup>2</sup>	std (deg)	mean (deg)
Continuous	0.0042	0.0002	0.0647	0.0000	0.0003	(-)0.0014
Discrete	0.0029	0.0041	0.0536	0.0002	0.0043	0.0122

Table 4.4: Attitude Error of the Ideal Model

In Figure 4.12 is shown the response of the discrete model to the same set reference, but now the initial conditions are:  $X_0 = [0,0,0,-2,0.2,0.2]^T$ . It can be observed that the discrete LQR controller achieved to stabilize properly the ALIV3, taking it from the initial conditions to the set reference.

Finally in Figure 4.13, all the poles of the discrete close loop are within the unit circle, proving that the controller make the system stable.

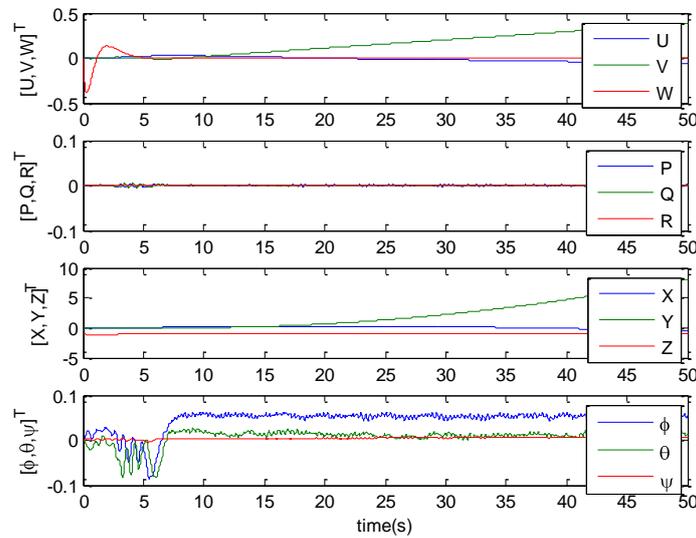


Figure 4.11: Attitude Stabilization of the Discrete Model

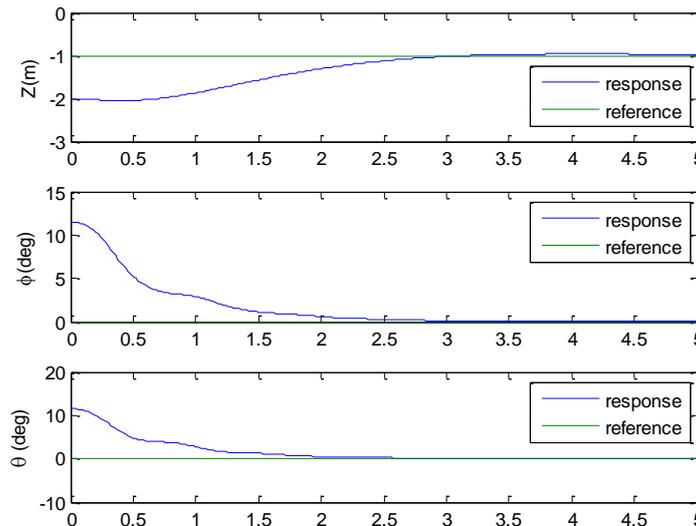


Figure 4.12: Attitude Stabilization of the Discrete Model with Initial Conditions

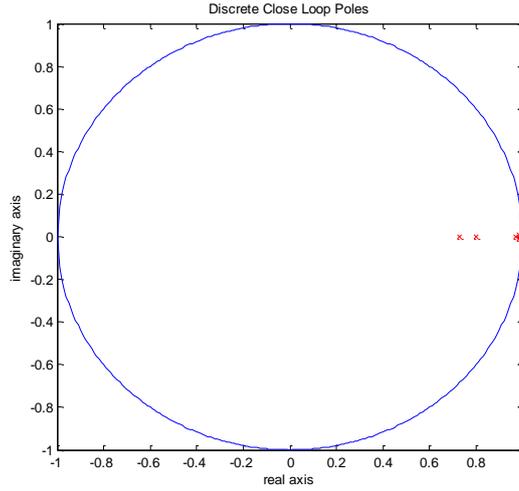


Figure 4.13: Discrete Closed Loop Poles

### 4.3.2 Realistic Simulation

After test the LQR controller in close loop with the system reduced to 6 states, the continuous and the discrete models, in this section the simulation of the model with estimation feedback is performed. The values of  $\phi$  and  $\theta$  are estimated by an EKF;  $P$  and  $Q$  are used directly from the gyroscope;  $W$  is estimated by a high pass filter and finally, because the concern of this work is the attitude stabilization,  $Z$  is used as ideal data. All sensors are considerate to work at  $freq = 50Hz$  and the high pass filter used to obtain the vertical velocity uses a cut-off frequency of  $\frac{1}{0.005} rad/s$ . The implantation of the model in *Simulink* is shown in Figure 4.14.

Like in the previous sections, two results are presented. First using the set reference  $\mathbf{X}_{ref} = [0,0,0,-1,0,0]^T$  and the initial conditions  $\mathbf{X}_0 = [0,0,0,-1,0,0]^T$ . The second test is performed with the same reference set but now the initial conditions are  $\mathbf{X}_0 = [0,0,0,-2,0.2,0.2]^T$ . In both cases, the weighting matrices of the controller and the estimator are:

$$\mathbf{Q}_{lqr} = diag([1, 100, 100, 400, 250, 250]) \quad (4.24)$$

$$\mathbf{R}_{lqr} = diag([0.01, 0.01, 0.01, 0.01]) \quad (4.25)$$

$$\mathbf{Q}_k = diag([0.00005, 0.00005]) \quad (4.26)$$

$$\mathbf{R}_k = diag([0.0212, 0.013, 0.2024, 0.0799]) \quad (4.27)$$

The weightings of  $\mathbf{Q}_{lqr}$  that are related with the height and the vertical velocity have been changed respect to the ideal case. With the vertical velocity estimated by the high pass filter, appears an error of  $0.4 m$  in the heigh because the weight related to the vertical velocity was too big compared with the weight of the height. So, the controller stabilized the ALIV3 around a null vertical velocity but with an

error in  $Z$ . Therefore, the weight related to  $Z$  has been increased and the one related to  $W$  has been decreased.

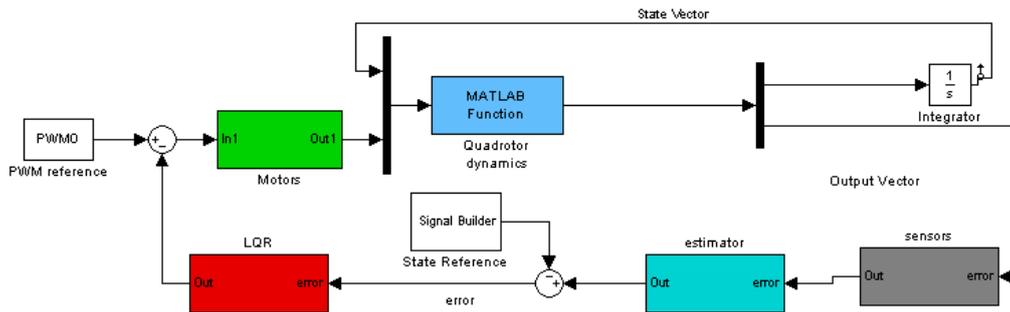


Figure 4.14: Implantation of the Model with Estimation Feedback

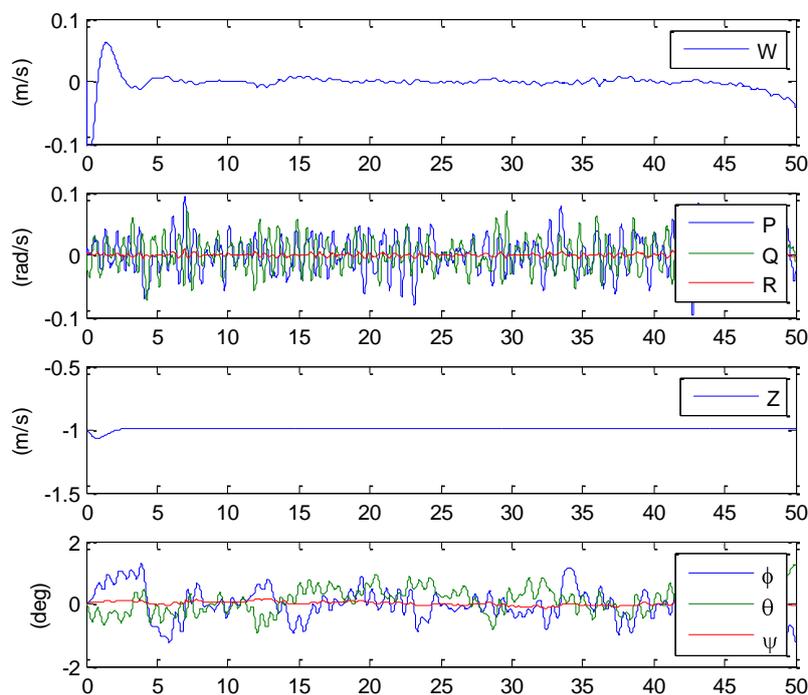


Figure 4.15: Attitude Stabilization with Estimation Feedback

Figure 4.15 presents the response of the model to the first test. It can be observed that the controller achieve stabilize the tilt-quadrotor around the set reference with a maximum error in  $\phi$  of  $1.43^\circ$  and in  $\theta$  of  $1.25^\circ$ . Table 4.5 presents the mean square error (mse), the standard deviation (std) and mean of the attitude angles  $\phi$  and  $\theta$ :

	$\phi$	$\theta$
mse (deg) <sup>2</sup>	0.2983	0.1781
std (deg)	0.5386	0.4006
mean (deg)	-0.0917	0.1330

Table 4.5: Attitude Error with estimation Feedback

Figure 4.16 presents the response of the model to the second test. Table 4.6 presents the setting time and the rising time of  $Z$ ,  $\phi$  and  $\theta$ . In the case of the angles, due to the error, it is not possible to measure the setting time.

	$Z$	$\phi$	$\theta$
Rising time (s)	3.5	4.96	6.88
Setting time (s)	5.00	-	-

Table 4.6: Rising Time and Setting time with Estimation Feedback

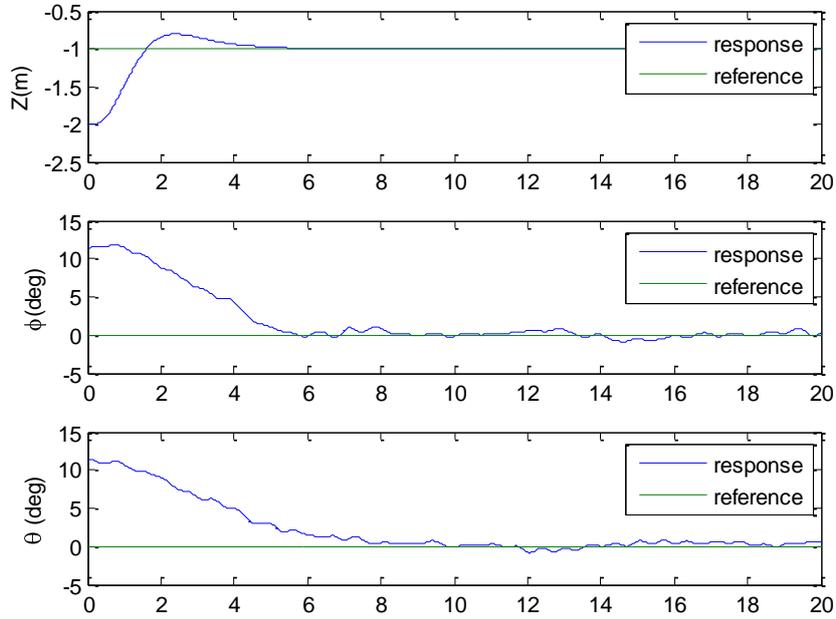


Figure 4.16: Attitude Stabilization with Estimation Feedback and with Initial Conditions

The simulation suggests that the tilt-quadrotor attitude could be stabilized for a hover flight using the implemented control and estimation methods. The standard variation and the mean error of the roll angle and pitch angle are greater using the estimation feedback than the ideal case, as could be predicted. However, these errors are small and so it can be considered that the LQR and EKF solution for attitude is an efficient method. It also have been checked that the yaw angle is not able to be controlled without tilting any rotor, but it remains stabilized.

## Chapter 5

# Conclusions and Future Work

### 5.1 Conclusions

The goal of this work was to continue the ALIV project with the developing of the control system. That enables stabilize the attitude when it is hovering. Different steps have been reached:

- The structure has been improved and the electronics components checked. The platform had resonance problems that were solved, the swivel arms were counterbalanced, the motors were changed by the ones chosen by Fernandes [14], and the ESCs were also replaced by better ones.
- A study of the general performs of a tilt-quadrotor have been achieved, analyzing what happen when the rotors are tilted and when a tilt-quadrotors fly like a standard quadrotor. It has been observed that the yaw motion is only possible by tilting the rotors.
- The model of the platform has been obtained experimentally. The Center of mass has been determinate via pictures of the platform when it is dropped by a string. The inertia Matrix has also been determined experimentally by the special design and construction of a structure for its determination in any quadrotor.
- The models of the actuators have been obtained considering the electronic speed controllers, the motors, the propellers and the battery. Because main goal was stabilize the tilt-quadrotor without tilt any rotor, the model of the servos was not considered. The behavior of the motors with the temperature and the battery discharge has been studying, concluding that the motors can be approximated to a first order system were the PWM signal is proportional to the rpm divided by the battery voltage. All the motors have been considered to have the same time constant, and this value was considered to be the same as in Bernardo's work [1]. The identification of the propellers and the motors has been achieved with the help of a laser based tachometer to measure the angular speed and a balance to measure the weight.

- The model of the sensors has been obtained, measuring the experimental noise when the motors are running. For greater realism in simulations, the sensors were modeled using the noise when the motors are turned on. This implied the development of Arduino based code to read the sensors measurements and interpret them.
- The models of the motors, sensors and quadrotors dynamics have been linearized and implemented in *Simulink* considering that any rotors can tilt. The linearization has been done around the hover situation by computationally method and checked by analytically method.
- The controllability and observability of the 12 states model have been studied, resulting that the model is neither fully controllable nor observable. As the analysis of the performance indicated, the yaw motion is not controllable but it is stabilized. The fully controllable and observable model without considering the GPS is a model of 6 States.
- Finally, a LQR controller has been developed based on the 6 observable and controllable states. It has been tuned and tested using ideal data from the sensors in the case of the continuous model and in the case of the discrete model. Then, the EKF has been implemented on the *Simulink* model. The LQR has been retuned and it has been tested that the LQR and the EKF combined could stabilize the ALIV3 in hover without tilting any rotor.

## 5.2 Future work

- The estimation of the height should be completely implemented, using a filter to the barometer measurements.
- Implement the control and estimation methods developed in this work in the real platform and check them.
- Develop the model and the control method when the rotors can tilt and the GPS is also connected. Note that when rotors are tilted, this fact affect to the dynamic of the model. The definition of the vector  $\mathbf{F}^B = [F_x; F_y; F_z]^T$  changes and the components  $F_x$  and  $F_y$  are not null, being dependent of the angle that the rotors are tilted.
- Develop the communication method via the Xbee Wire Antenna.
- The structure can be also improved. The down central board could be removed. The landing gear could be attached to the upper central board and the battery could also be attached to the landing gear rotating 90° its actual position and building a piece in composites for the attachment.
- A better cover study can be realized, being it optimized to increase the performance in all directions, not just forward flight as Fernandes did.

# Bibliography

- [1] Bernardo Sousa Machado Henriques. Estimation and control of a quadrotor attitude. Master's thesis, Instituto Superior Técnico, 2011.
- [2] BMP085, Digital pressure sensor Data sheet. Bosch Sensortec. 15 Oct 2009.
- [3] C.Gablehouse. Helicopters and autogiros. J. B. Lippincott Company, Philadelphia, PA, 1969
- [4] Filipe M. S. Pedro. Projecto preliminar de um quadrirotor. Master's thesis, Instituto Superior Técnico, Universidade Técnica de Lisboa, 2009.
- [5] G.Genta and C. Delpretê. Some Consideration on the Experimental Determination of Moments of Inertia. Politecnico di Torino, 1994.
- [6] J. Seddon. Basic Helicopter Aerodynamics. BSP Professional Books, 1990
- [7] Katsuhiko Ogata. Ingeniería de Control Moderna. Pearson Educación, 2010.
- [8] Mahony, R., T. Hamel, 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.
- [9] Mahony, R., T. Hamel, and J.-M. Pflimlin. Nonlinear complementary filters on the special orthogonal group. Volume 53 n-5, pp. 1203–1218. IEEE Transactions on Automatic Control, 2008
- [10] Marcelo Rodríguez Danta. Mecánica. Universidad de Sevilla, Secretariado de Publicaciones, 2010.
- [11] Mark Euston, Paul Coote, Robert Mahony, Jonghyuk Kim, and Tarek Hamel. A complementary filter for attitude estimation of a fixed-wing uav. pages 340–345. International Conference on Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ, 2008.

- [12] Mohammed Dahleh, Munther A. Dahleh, George Verghese. Lectures on Dynamic Systems and Control. Department of Electrical Engineering and Computer Science. Massachusetts Institute of Technology, 2013.
- [13] Motor model  
<http://ctms.engin.umich.edu/CTMS/index.php?example=MotorSpeed&section=SystemModeling>,  
August 2013
- [14] Nelson dos Santos Fernandes. Design and construction of a multi-rotor with various degrees of freedom. Master's thesis, Instituto Superior Técnico, Universidade Técnica de Lisboa, 2011.
- [15] Robert Mahony, Tarek Hamel, and Sung-Han Cha. A coupled estimation and control analysis for attitude stabilization of mini aerial vehicles. Proceedings of the Australasian Conference on Robotics and Automation, 2006.
- [16] Sérgio Eduardo Aurélio Pereira da Costa. Controlo e simulação de um quadricóptero convencional. Master's thesis, Instituto Superior Técnico, 2008.
- [17] Severino M. O. Raposo. System and process of vector propulsion with independent control of three translation and three rotation axis. Intellectual Property, 2010
- [18] Springmann, Enno; Gottfried Hilscher. *Focke: Flugzeuge und Hubschrauber von Heinrich Focke 1912-1961*. Aviatic-Verlag GmbH, 1997.
- [19] Stevens and Lewis. Aircraft Control and Simulation, 1992
- [20] Syed Ali Raza and Wail Gueaieb. Intelligent Flight Control of an Autonomous Quadrotor. University of Ottawa, Canada, 2010.
- [21] Young, Warren R. *The Helicopters*. "The Epic of Flight". Chicago: Time-Life Books, 1982.

# Appendices

## Appendix A. Experimental Test Results

Experimental values obtained during the tests described in Chapter 3 are added in this appendix.

### A.1 Moments of Inertia

	Probeta	Structure	x	y	z
$A_0$ [°]	33,75	40	33,75	40	40
$A_1$ [°]	20	25	18,25	24,25	26
$\Delta T_1$ [s]	0,17	0,73	0,739	0,7396	0,7817
$\Delta T_2$ [s]	0,78	3,86	3,89	3,6842	4,0993
$T$ [s]	0,95	3,86	4,63	4,4237	4,8811
$\gamma$ [s <sup>-1</sup> ]	0,5502	0,1217	0,1328	0,1131	0,0883
$I$ [kg × m <sup>2</sup> ]	0,0051	0,0841	0,1208	0,1103	0,1345

Table A.1: Moments of Inertia values

### A.2 Curve PWM-RPM:

PWM	RPM Motor 1	RPM Motor 2	RPM Motor 3	RPM Motor 4
952	449	449	470	435
1050	1945	1957	2019	1965
1150	2903	2974	3043	2941
1250	3686	3714	3838	3708
1350	4279	4331	4478	4347
1450	4821	4859	4983	4887
1550	5392	5376	5542	5405
1650	5917	5903	6113	5957
1750	6480	6386	6642	6496
1850	6861	6874	7073	6871
1950	-	7234	7458	7295
2100	-	7425	7741	7556

Table A.2: Curve PWM-RPM values

### A.3 Curve Time-RPM Motor 3:

Time(s)	1350 $\mu s$	1450 $\mu s$	1450 bis	1550 $\mu s$	1650 $\mu s$
30	4498	4919	5085	5708	6290
65	4486	4893	5050	5652	6177
100	4456	4856	5026	5586	6106
135	4441	4841	4999	5571	6059
170	4423	4824	4971	5534	6027
205	4438	4823	4961	5517	6003
240	4427	4808	4945	5515	5999
275	4419	4801	4931	5458	5957
310	4421	4805	4929	5464	5912
345	4415	4797	4916	5426	5885
380	4413	4786	4915	5415	5858
415	4404	4787	4898	5405	5850
450	4394	4766	4891	5389	5830
485	4367	4777	4876	5378	5834
520	4351	4758	4868	5565	5778
555	4352	4766	4851	5356	5826
590	4353	4754	4829	5356	5771
625	4350	4727	4828	5341	5718
Time(s)	1350 $\mu s$	1450 $\mu s$	1450 $\mu s$	1550 $\mu s$	1650 $\mu s$
660	4331	4736	4818	5342	5731
695	-	4728	4813	5324	5682
730	-	4742	4799	5322	5699
765	-	4729	4801	5300	5680
800	-	4729	4793	5303	5693
835	-	4729	4792	5271	5672
870	-	4726	4785	5276	5668
905	-	4707	4779	5270	5647
940	-	4694	4780	5263	5638
975	-	4710	4770	5260	5620
1010	-	4699	4763	5239	5605
1045	-	-	4765	5257	5613

Table A.3: Curve Time-RPM Motor 3 values

### A.4 Battery vs Power Supply:

Time (s)	Power Supply	Battery	Time (s)	Power Supply	Battery
30	4662	5085	555	4630	4851
65	4651	5050	590	4631	4829
100	4647	5026	625	4632	4828
135	4625	4999	660	4630	4818

170	4641	4971	695	4633	4813
205	4636	4961	730	4633	4799
240	4642	4945	765	4635	4801
275	4631	4931	800	4637	4793
310	4644	4929	835	4633	4792
345	4636	4916	870	4642	4785
380	4644	4915	905	4638	4779
415	4640	4898	940	4623	4780
450	4629	4891	975	4625	4770
485	4639	4876	1010	4629	4763
520	4627	4868	1045	4630	4765

Table A.4: Battery vs Power Supply values

## A.5 Curve PWM-RPM/V:

PWM MOTOR 1	RPM/V	Relative Error	T	I	V	CT	CP
1150	237,1	2,12	142	1,31	12,29	0,0149	0,0045
1350	354,8	2,03	297	3,43	12,04	0,0145	0,0036
1450	401,3	0,55	365	4,69	11,88	0,0144	0,0035
1550	452	0,56	444	6,2	11,71	0,0142	0,0034
1650	504,8	0,81	534	8,2	11,54	0,0141	0,0033
1750	556,7	0,87	620	10,51	11,38	0,0138	0,0032
1900	-	-	677	13,74	11,01	-	-
PWM MOTOR 2	RPM/V	Relative Error	T	I	V	CT	CP
1150	237,1	2,8	126	1,26	12,34	0,0132	0,0043
1350	352,2	3	285	3,38	12,03	0,0142	0,0037
1450	397,6	1,2	366	4,53	11,93	0,0145	0,0035
1550	441,9	0,64	354	5,9	11,76	0,0115	0,0034
1650	491,4	0,79	447	7,68	11,59	0,0123	0,0033
1750	542,3	0,9	578	9,75	11,35	0,0136	0,0033
1900	606	2,67	722	13,15	11,11	0,0142	0,0033
PWM MOTOR 3	RPM/V	Relative Error	T	I	V	CT	CP
1150	243,9	4,37	136	1,3	12,3	0,0135	0,0041
1350	363,9	1,59	289	3,5	12,05	0,0134	0,0034
1450	411,2	0,5	360	4,74	11,86	0,0135	0,0033
1550	461,4	0,58	439	6,12	11,7	0,0134	0,0031
1650	533,6	0,98	564	7,95	11,3	0,0148	0,0031
1750	561	0,89	609	10,17	11,36	0,0134	0,0031
1900	631,3	1,87	719	13,8	11	0,0133	0,0031
PWM MOTOR 4	RPM/V	Relative Error	T	I	V	CT	CP
1150	237,7	2,2	135	1,27	12,33	0,014	0,0043
1350	354,8	1,94	290	3,36	12,07	0,0141	0,0035
1450	402,6	0,62	358	4,56	11,91	0,0139	0,0034
1550	450,9	0,57	435	6	11,73	0,0139	0,0033

1650	505,7	0,52	519	7,75	11,34	0,014	0,0032
1750	555,4	0,67	540	10,1	11,22	0,0125	0,0032
1900	620,4	3,08	677	13,74	11,01	0,0129	0,0033

Table A.5: Curve PWM- RPM/V values

## A.6 Curve with BEC vs without BEC

Time (s)	RPM without BEC	RPM with BEC	RPM/V without BEC	RPM/V with BEC
30	5128	5085	415	412
65	5091	5050	413	411
100	5062	5026	412	410
135	5023	4999	411	409
170	5005	4971	411	408
205	4995	4961	411	408
240	4981	4945	412	408
275	4964	4931	411	408
310	4954	4929	411	409
345	4944	4916	411	409
380	4952	4915	413	409
415	4953	4898	414	409
450	4917	4891	412	410
485	4906	4876	413	410
520	4893	4868	414	410
555	4883	4851	414	411
590	4874	4829	415	410
625	4871	4828	415	412
660	4857	4818	415	412
695	4854	4813	415	412
730	4862	4799	417	412
765	4853	4801	417	413
800	4841	4793	417	413
835	4828	4792	416	413
870	4833	4785	418	414
905	4816	4779	417	414
940	4812	4780	417	415
975	4819	4770	419	414
1010	4816	4763	419	415
1045	4800	4765	418	415

Table A.6: Curve with BEC and without BEC values

## A.7 Curve Endurance

Time (min)	Voltage (V)	Time (min)	Voltage (V)
0,5	11,73	8,5	10,89
1	11,63	9	10,86
1,5	11,55	9,5	10,83
2	11,47	10	10,81
2,5	11,37	10,5	-
3	11,42	11	10,78
3,5	-	11,5	10,77
4	11,36	12	10,71
4,5	11,32	12,5	10,67
5	-	13	-
5,5	11,09	13,5	10,64
6	-	14	-
6,5	11,04	14,5	-
7	10,99	15	9,95
7,5	10,95	15,5	9,5
8	10,92	-	-

Table A.7: Curve Endurance values

## Appendix B. Wiring

The scheme of the electrical connections is shown in the Figure B.1: Electric scheme. The pin description of the PDB scheme is presented in Table B.1: **PDB Pin Description** and the pin description of the ESC scheme is presented in Table B.2: ESC Pin Description. In Figure B.2: APM Wiring is also shown the wiring of the APM. On the right are connected the outputs, which in this case are the wires that come from the PDB, and on the left are connected the inputs, which in this figure are the wires that come from the RC receiver.

Because our system has elements that work at different values of tension, the PDB board to distribute the power: motors work at 11.1V while the servos work in the range of 4.8V to 6V and the Ardupilot at 5V. Then, the battery is connected to the PDB, which send the power to the motors. Then the motors return the BEC to the PDB to feed the Ardupilot and the RC receiver. The PDB also receives the PWM signals from the Ardupilot and sends them to the ESCs.

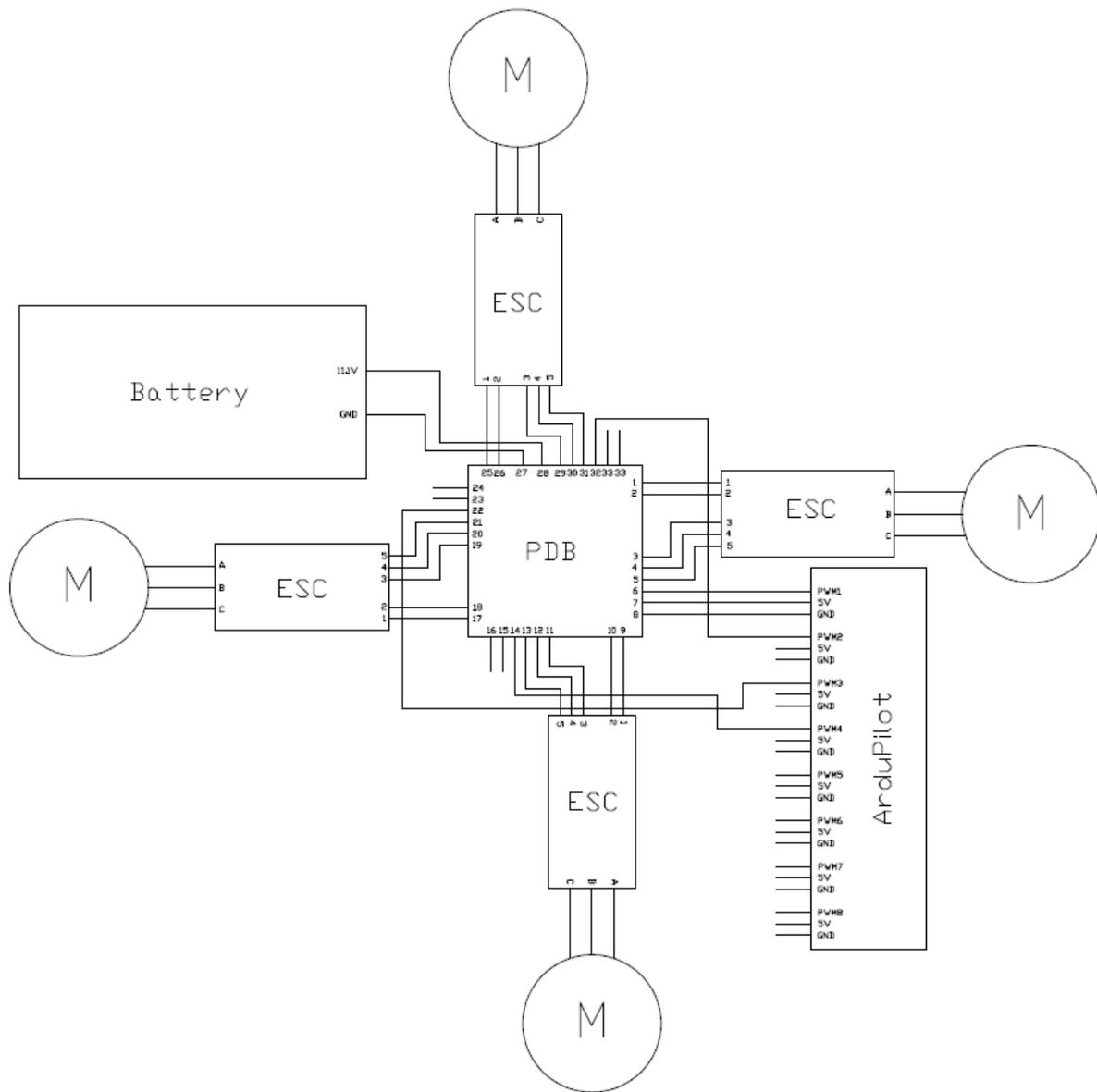


Figure B.1: Electric scheme

PIN	Reference	PIN	Reference	PIN	Reference	PIN	Reference
1	GND	10	GND	19	PWM3 (OUT)	28	11.1V
2	11.1V	11	PWM4 (OUT)	20	5V	29	PWM2 (OUT)
3	PWM1 (OUT)	12	5V	21	GND	30	5V
4	5V	13	GND	22	PWM3 (INP)	31	GND
5	GND	14	PWM4 (INP)	23	5V	32	PWM2 (INP)
6	PWM1 (INP)	15	5V	24	GND	33	5V
7	5V	16	GND	25	11.1V	34	GND
8	GND	17	11.1V	26	GND		
9	11.1V	18	GND	27	GND		

Table B.1: PDB Pin Description

PIN	Reference
1	11.1V
2	GND
3	PWM
4	5V
5	GND
A	Phase 1
B	Phase 2
C	Phase 3

Table B.2: ESC Pin Description

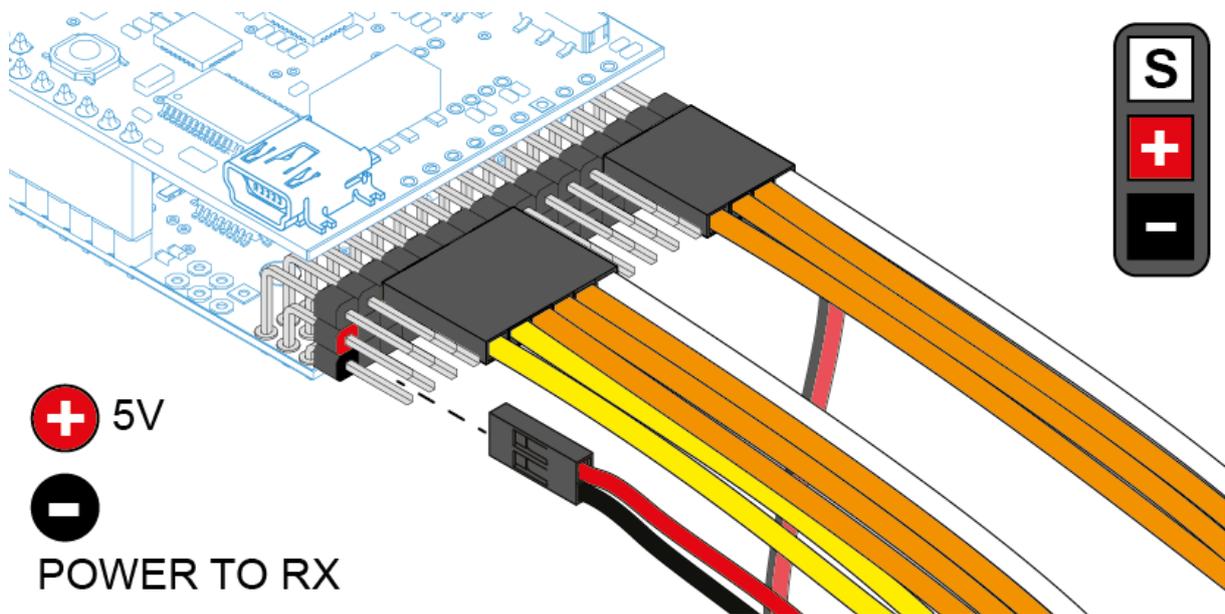


Figure B.2: APM Wiring

## Appendix C. Linearization Matrices

In this appendix are presented the linearization matrices of the 12 state-space model and of the 6 states-space model.

### 12 State-Space Model

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -9.81 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 9.81 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -0.0052 & -0.00490 & -0.0051 & -0.0052 \\ 0 & 0.0769 & 0 & -0.0806 \\ 0.1134 & 0 & -0.1115 & 0 \\ 0.0059 & 0.0056 & 0.0058 & -0.0059 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -9.81 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$D = \begin{bmatrix} 0.0036 & 0 & -0.0036 & 0 \\ 0 & -0.0025 & 0 & 0.0026 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

## 6 State-Space Model

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} -0.00521 & -0.00494 & -0.00512 & -0.00518 \\ 0 & 0.07685 & 0 & -0.08065 \\ 0.11344 & 0 & -0.11146 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & -9.81 \\ 0 & 0 & 0 & 0 & 9.81 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$D = \begin{bmatrix} 0.00363 & 0 & -0.00357 & 0 \\ 0 & -0.00246 & 0 & 0.00258 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

## Appendix D. Maintenance

The battery charge must be always between 9.5V-12.5V. The speed controllers have the ability of cut off the battery supply if its charge drops bellow a defined limit. This protects the battery. To charge the battery a "Lithium-Polymer Battery Charger" must be used. These chargers allow parallel and independent charging of the different cells, as well as a balancing and a cut-off action. This means that if one cell is more charged than another, they will first be balanced and then charged at the same pace; once the charging is completed, the charger cuts the supply off, protecting the cells.

The charger can works in a range of 1.5A to 3A and at 10V. Special care must be given to the connections since a short-circuit could mean the death of the battery's cells.

The prototype must also be protected from humidity and dust which could affect the bearings in the outrunner motor, the bearing of the swivel arms and the electronics.

It is also convenient to make sure the screws are not missing before each flight test. This can prevent severe damage on the prototype. The most affected by the vibrations seems to be the ones that fix the motors. A check should be done before each flight because a fail in the sky is critical.

## Appendix E. Arduino Programs

ArduPilot is an open source platform created by Chris Anderson and Jordi Muñoz of DIY Drones. The hardware consists in the core autopilot board, sold by Sparkfun, and various sensors and accessories to add to its functionality. The software is based on Arduino, but with the particularity that all input and output ports are already defined. So, to send and receive information it is necessary work with the open source libraries specially developed for the ArduPilot.

The libraries used in this work are: APM\_RC, APM\_ADC, APM\_compass and APM\_BMP085. The library APM\_RC is used to receive information from the inputs and to send information through the outputs. The library APM\_ADC is used to read information from the accelerometers and form the gyroscopes. The library APM\_compass is used to read information from the compass. Finally the library APM\_BMP085 is used to read information from the barometer.

The different codes used for the Motor tests and for take data from the sensors are:

### Motor test

```
#include <APM_RC.h> // ArduPilot Mega RC Library
```

```

String incomingString;
void setup()
{
  APM_RC.Init(); // APM Radio initialization
  Serial.begin(57600);
  Serial.println("initializing test_motors");
  delay(1000);
}
void loop()
{
  if(Serial.available() > 0) // If there is incoming value
  {
    char ch = Serial.read();// read the value
    if (ch != 97){
      Serial.print("I have received: ");
      Serial.print(ch, DEC);
      Serial.print('\n');
      incomingString += ch; //Add the character to the incomingString
    }
    // received a newline (linefeed) character
    // this means we are done making a string
    Else // print the incoming string
    {
      Serial.println("I am printing the entire string");
      Serial.println(incomingString);
      int val = incomingString.toInt();// Convert the string to an integer
      Serial.println("Printing the value:");// print the integer
      Serial.println(val);
      if (val > 899 && val < 2101) // We only want to write an integer between 900 and 2100
      {
        // Print confirmation that the value is between 900 and 2100
        Serial.println("Value is between 900 and 2100");
        // Write to Servo
        APM_RC.OutputCh(4,val);
        APM_RC.OutputCh(5,val);
        APM_RC.OutputCh(6,val);
        APM_RC.OutputCh(7,val);
      }
      // If the value is not between 900 and 2100 We do not want write this value to the motor.
      else
      {
        Serial.println("Value is NOT between 900 and 2100");
        Serial.println("Error with the input");
      }
      incomingString = ""; // Reset the value of the incomingString
    }
  }
}

```

```
}  
}
```

## Accelerometer and Gyroscope test

```
#include <APM_RC.h> // ArduPilot Mega RC Library  
#include <APM_ADC.h>  
String incomingString;  
unsigned long timer;  
void setup()  
{  
  APM_RC.Init(); // APM Radio initialization  
  APM_ADC.Init(); // APM ADC initialization  
  Serial.begin(57600);  
  Serial.println("initializing test_motors");  
  delay(1000);  
  timer = millis();  
  Serial.print("gyro_z[] gyro_x[] gyro_y[] temp[°] ax[] ay[] az[] pitot\n");  
}  
void loop()  
{  
  int chs;  
  
  if((millis()- timer) > 20)  
  {  
    timer = millis();  
    for (chs=0;chs<8;chs++)  
    {  
      Serial.print(APM_ADC.Ch(chs),DEC);  
      Serial.print(" ");  
    }  
    Serial.println();  
  }  
  if(Serial.available() > 0) // If there is incoming value  
  {  
    char ch = Serial.read();// read the value  
    if (ch != 97){  
      Serial.print("I have received: ");  
      Serial.print(ch, DEC);  
      Serial.print("\n");  
      incomingString += ch; //Add the character to the incomingString  
    }  
    // received a newline (linefeed) character  
    // this means we are done making a string  
    Else // print the incoming string  
    {  
      Serial.println("I am printing the entire string");  
      Serial.println(incomingString);  
      int val = incomingString.toInt();// Convert the string to an integer
```

```

Serial.println("Printing the value: "); // print the integer
Serial.println(val);
if (val > 899 && val < 2101) // We only want to write an integer between 900 and 2100
{
  // Print confirmation that the value is between 900 and 2100
  Serial.println("Value is between 900 and 2100");
  // Write to Servo
  APM_RC.OutputCh(4,val);
  APM_RC.OutputCh(5,val);
  APM_RC.OutputCh(6,val);
  APM_RC.OutputCh(7,val);
}
// If the value is not between 900 and 2100 We do not want write this value to the motor.
else
{
  Serial.println("Value is NOT between 900 and 2100");
  Serial.println("Error with the input");
}
  incomingString = ""; // Reset the value of the incomingString
}

delay (500);
}

```

## Compass test

```

#include <APM_RC.h> // ArduPilot Mega RC Library
#include <Wire.h>
#include <APM_Compass.h> // Compass Library

#define ToDeg(x) (x*57.2957795131) // *180/pi
String incomingString;
unsigned long timer;
void setup()
{
  APM_RC.Init(); // APM Radio initialization
  APM_Compass.Init(); // Initialization
  Serial.begin(57600);
  Serial.println("initializing test_motors_compas");
  delay(1000);
  timer = millis();
  Serial.print("RumoMagnetico[0] Raw_X Raw_Y Raw_Z");
}
void loop()
{
  float tmp;

  if((millis()- timer) > 100)
  {

```

```

timer = millis();
APM_Compass.Read();
APM_Compass.Calculate(0,0); // roll = 0, pitch = 0 for this example

Serial.print(ToDeg(APM_Compass.Heading));
Serial.print(" ");
Serial.print(APM_Compass.Mag_Y);
Serial.print(" ");
Serial.print(APM_Compass.Mag_X);
Serial.print(" ");
Serial.print(APM_Compass.Mag_Z);

Serial.println();
}
if(Serial.available() > 0) // If there is incoming value
{
    char ch = Serial.read();// read the value
    if (ch != 97){
        Serial.print("I have received: ");
        Serial.print(ch, DEC);
        Serial.print("\n");
        incomingString += ch; //Add the character to the incomingString
    }
    // received a newline (linefeed) character
    // this means we are done making a string
    Else // print the incoming string
    {
        Serial.println("I am printing the entire string");
        Serial.println(incomingString);
        int val = incomingString.toInt();// Convert the string to an integer
        Serial.println("Printing the value:");// print the integer
        Serial.println(val);
        if (val > 899 && val < 2101) // We only want to write an integer between 900 and 2100
        {
            // Print confirmation that the value is between 900 and 2100
            Serial.println("Value is between 900 and 2100");
            // Write to Servo
            APM_RC.OutputCh(4,val);
            APM_RC.OutputCh(5,val);
            APM_RC.OutputCh(6,val);
            APM_RC.OutputCh(7,val);
        }
        // If the value is not between 900 and 2100 We do not want write this value to the motor.
        else
        {
            Serial.println("Value is NOT between 900 and 2100");
            Serial.println("Error with the input");
        }
    }
}

```

```

    }
    incomingString = ""; // Reset the value of the incomingString
  }
  delay (500);
}

```

## Barometer Test

```

#include <APM_RC.h> // ArduPilot Mega RC Library
#include <Wire.h>
#include <APM_BMP085.h>
String incomingString;
unsigned long timer;
void setup()
{
  APM_RC.Init(); // APM Radio initialization
  APM_BMP085.Init(); // APM ADC initialization
  Serial.begin(57600);
  Serial.println("initializing test_motors_baro");
  delay(1000);
  timer = millis();
  Serial.print("Pressure[Pa] Temperature[°C] Altitude[m]\n");
}
void loop()
{
  float tmp_float;
  float Altitude;

  if((millis()- timer) > 50)
  {
    timer=millis();
    APM_BMP085.Read();

    Serial.print(APM_BMP085.Press);
    Serial.print(" ");
    Serial.print(APM_BMP085.Temp/10.0);
    Serial.print(" ");
    tmp_float = (APM_BMP085.Press/101325.0); // calculos auxiliares
    tmp_float = pow(tmp_float,0.190295); // para calculo da Altitude
    Altitude = 44330*(1.0-tmp_float);
    Serial.print(Altitude);

    Serial.println();
  }
  if(Serial.available() > 0) // If there is incoming value
  {
    char ch = Serial.read();// read the value
    if (ch != 97){
      Serial.print("I have received: ");
      Serial.print(ch, DEC);
      Serial.print('\n');
    }
  }
}

```

```

    incomingString += ch; //Add the character to the incomingString
}
// received a newline (linefeed) character
// this means we are done making a string
Else // print the incoming string
{
    Serial.println("I am printing the entire string");
    Serial.println(incomingString);
    int val = incomingString.toInt();// Convert the string to an integer
    Serial.println("Printing the value: ");// print the integer
    Serial.println(val);
    if (val > 899 && val < 2101) // We only want to write an integer between 900 and 2100
    {
        // Print confirmation that the value is between 900 and 2100
        Serial.println("Value is between 900 and 2100");
        // Write to Servo
        APM_RC.OutputCh(4,val);
        APM_RC.OutputCh(5,val);
        APM_RC.OutputCh(6,val);
        APM_RC.OutputCh(7,val);
    }
    // If the value is not between 900 and 2100 We do not want write this value to the motor.
    else
    {
        Serial.println("Value is NOT between 900 and 2100");
        Serial.println("Error with the input");
    }
    incomingString = ""; // Reset the value of the incomingString
}
delay (500);
}

```