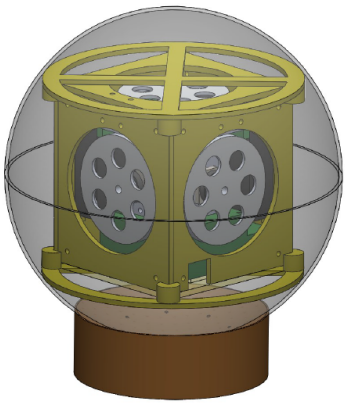
TÉCNICO LISBOA



Development of an Air-Bearing-Based Satellite Attitude Simulator and Testing of an ADCS Solution

Bruno Alexandre Morais Dias Ribeiro

Thesis to obtain the Master of Science Degree in

Aerospace Engineering

Supervisor: Prof. Paulo Jorge Coelho Ramalho Oliveira

Examination Committee

Chairperson: Prof. Fernando José Parracho Lau
Supervisor: Prof. Paulo Jorge Coelho Ramalho Oliveira
Member of the Committee: Prof. Rodrigo Martins de Matos Ventura

March 2021

Dedicated to the memory of my mother.

Acknowledgments

First of all, I would like to thank my supervisor, professor Paulo Oliveira, for the unwearying support and advice provided throughout the elaboration of this dissertation and for challenging me to pursue this endeavor and providing the equipment and facilities for the research performed during the past months.

I would like to thank professor Carlos Cardeira for the insightful suggestions and discussions throughout the past months.

I express my gratitude to the Calouste Gulbenkian Foundation for granting me a merit scholarship, which allowed me to focus on my studies.

I would also like to thank my colleagues, particularly Margarida Cosme and Francisco Carneiro, for all the companionship and for helping me keep my mental sanity during these 5 years. A special thanks to my friend Eng. Reydleon Paulo for always keeping me motivated.

Finally, I would like to thank my brothers, Catarina, Fábio and Fabiana, my father, José, and my girlfriend, Sofia Alves, for their unconditional love and for putting up with me during these challenging times.

Resumo

Nos últimos anos, o interesse por nano-satélites tem vindo a crescer. Entre a tecnologia envolvida nestes veículos, é de destacar o sistema de determinação e controlo de atitude (ADCS). Uma particularidade dos projetos espaciais prende-se com a dificuldade em testar as tecnologias num ambiente que simule as condições a que o veículo estará sujeito. O principal objetivo desta dissertação é desenvolver um protótipo que permita testar os sistemas de ADCS desenvolvidos no âmbito do projeto NANOSTAR. Para isso, um ADCS que utiliza rodas de momento e ”magnetorquers” é considerado. A determinação da atitude é baseada em medidas de um giroscópio, magnetómetro e acelerómetro. As rodas de momento são usadas para direcionar o satélite na direção Nadir. Dois controladores ótimos em regime estacionário são considerados, um controlador LQR e um controlador LQR com ação integral. Os ”magnetorquers” são usados para ”detumbling” e descarga de momento. Três algoritmos de estimação são considerados, dois dos quais baseados em filtros de Kalman, um deles formulado em ângulos de Euler e o outro em quaterniões. O terceiro filtro é baseado na técnica SLERP. Os algoritmos são testados num ambiente de software que simula de forma realística as condições a que o satélite estará sujeito no espaço. De seguida, o design de uma plataforma de testes e de um protótipo de um CubeSat são descritos. Resultados referentes ao desempenho do sistema ADCS são apresentados tanto no ambiente de simulação como no ambiente experimental.

Keywords: Nano-satélites, Controlo Ótimo, Filtros Complementares, Ensaios em Voo

Abstract

There is a recent growth on the interest for nanosatellites. A key particularity of space projects is the difficulty to test the technologies in a realistic environment before the mission is deployed. The Attitude Determination and Control System (ADCS) is a critical subsystem in a nanosatellite. The main goal of this thesis is to help the NANOSTAR project to overcome this difficulty by designing a ground prototype to test the ADCS system and provide grounded studies on the attitude control and determination algorithms that can be employed in the missions. To this end, a low-cost ADCS is projected, having momentum wheels and magnetorquers as actuators. Attitude estimation is based on vector measurements provided by a magnetometer, rate gyroscope and accelerometer. Two optimal control strategies are used for Nadir pointing using the momentum wheels, namely an LQR controller and an LQR controller with integral action. Detumbling and momentum dumping are accomplished through the use of magnetorquers. Three estimation filters are considered, two based on Kalman filtering, one of them formulated on Euler angles and the other on quaternion. The third filter is based on the SLERP technique. The control and estimation algorithms are tested in a software simulation platform that describes the space environment realistically, allowing orbit generation and propagation. Then, a test bench and a CubeSat prototype are desgined and built, allowing to test the algorithms on ground, thus reducing the risk of failure. Results for the ADCS system performance are presented both from the simulation and the experimental environments.

Keywords: Nanosatellites, Optimal Control, Complementary Filters, Ground-testing

Contents

List of Tables

List of Figures

Nomenclature

Greek symbols

α	Rotational angle.
$\boldsymbol{\lambda}$	Vector containing the Euler angles.
δ	Neighbourhood.
θ	Pitch Euler angle.
μ	Fluid viscosity.
$\mu_{\odot}$	Gravitational parameter.
ρ	Density.
ϕ	Roll Euler angle.
Ψ	Yaw Euler angle.
ω	Argument of the perigee.
Ω	Right ascension of the ascending node.

Operators

$\forall$	For all.

Roman symbols

a	Semi-major axis.
$\mathbf{a}_d$	Acceleration due to disturbances.
a_g	Non-spherical Earth disturbance acceleration.
A	Area.
A	Rotation matrix.
b	Bias.
$\mathbf{B}_m$	Geocentric magnetic flux density.

$\mathbf{B}_y$ Magnetometer measurement.

C_d Oriffice coefficient of discharge.

C_D Dimensionless drag coefficient.

$\mathbf{C}_{si}$ Soft iron distortion.

$\frac{d}{dt}$ Time derivative.

D Domain.

e Eccentricity.

e Rotational axis.

f Force.

g Gravity vector.

h Angular momentum.

h Angular momentum vector.

i_m Orbit inclination.

i_M Current at the motor armature.

i,j,k Basis for the $\mathbb{R}^3$.

I Identity matrix.

J Inertia matrix.

J2 Zonal coefficient associated with the second order harmonic of the model of the Earth's gravity.

$\mathbf{K}_L$ LQR gain matrix.

K_M Torque coefficient of the motor.

K_v Back EMF.

$\mathcal{L}$ Lyapunov function.

m Magnetic moment.

m Mass.

M Mean anomaly.

n Average rate of sweep.

$\mathbf{n}_h$ Hard iron distortion.

p Orbit period.

P Kalman Error Covariance Matrix.

P Pressure.

q Vector part of a quaternion.

$\bar{\mathbf{q}}$ Quaternion.

$\Delta\bar{\mathbf{q}}$ Orientation increment.

$\delta\bar{\mathbf{q}}$ Quaternion Error.

q_0 Scalar part of a quaternion.

Q Flow rate.

Q Kalman Process noise matrix.

$\mathbf{Q}(\boldsymbol{\lambda})$ Transformation from angular rate to Euler angle rate.

$\mathbf{Q}_L$ LQR state-weighting matrix.

r Euclidean Distance.

r Vector from the center of the Earth to the center of mass of the spacecraft.

R Radius of the Earth.

R Kalman Observation noise matrix.

$\mathbf{R}_L$ LQR control-weighting matrix.

R_M Electrical resistance at the motor armature.

Re Reynolds number.

$\mathbb{R}$ Set of real numbers.

S Spacecraft area projected along the direction of motion.

S Solution to an algebraic Ricatti equation.

t Time.

$\mathbf{t}_d$ Torque due to disturbances.

Δt Time interval.

T Sample time.

T External Torque.

$\mathbf{T}_{BI}$ Diagonal matrix with the correlation times of the BI.

u System input.

v	Electrical voltage.
V	Performance index.
w	Angular velocity vector.
w_0	Orbit rate.
$\hat{\mathbf{w}}_n^w$	Spin axis of the n^{th} wheel axis resolved in the Body frame.
$\mathbf{W}^w$	Distribution matrix.
x	System state.
y	System output.
y_k	Output of the Allan variance rate gyros noise model.
0	Zeros matrix.

Subscripts

a	Aerodynamic.
b	Body frame.
c	Corrected.
det	Detumbling.
Earth	Earth's.
gg	Gravity gradient.
I	ECI frame.
k	Value at the k^{th} sample.
lvlh	LVLH frame.
m	Magnetic.
md	Moment Dumping.
n	Normal.
ned	NED frame.
rel	Relative.
S	Satellite.
s	Solar.
t	Transverse.

w Momentum wheel.

wmd Moment Dumping - Wheel.

Superscripts

$*$ Complex conjugate.

-1 Inverse.

n Nominal Value.

T Matrix transpose.

Acronyms

ADCS Attitude Determination and Control System.

ARW Angle Random Walk.

BI Bias Instability.

COTS Commercial Off-The Shelf.

CSD The CubeSat Design Specification.

DC Direct-Current.

DOF Degrees Of Freedom.

ECI Earth Centered Inertial Frame.

EMF Electromotive Force.

ESA European Space Agency.

GPIO General-Purpose Input/Output pins.

I2C Inter-Integrated Circuit.

IMU Inertial Measurement Unit.

ISS International Space Station.

LEO Low Earth Orbit.

LQR Linear Quadratic Regulator.

LQRI Linear Quadratic Regulator with Integral Action.

LTI Linear-Time Invariant.

LVLH Local Vertical Local Horizontal Frame.

MWS Momentum Wheel System.

NED North-east Down Frame.

NSS NANOSTAR Software Suite.

PID Proportional-Integral-Derivative Control.

PSD Power Spectral Density.

PWM Pulse-Width Modulation.

RRW Rate Random Walk.

RWS Reaction Wheel System.

SLERP Spherical Linear Interpolation.

SPI Serial Peripheral Interface.

WMM World Magnetic Model.

Chapter 1

Introduction

1.1 Motivation

As of 2020 October 4, arround 1417 nanosatellites have been launched by 518 companies in 68 countries. In the next 6 years, 2500 more will be launched [1]. In here, the term nanosatellite also encompasses CubeSats, Picosatellites, and other small satellites. Strictly speaking, the term nanosatellite is used to describe small satellites between 1 and 10 Kg. The term small satellite is used to describe satellites under 500 Kg [2]. Three small satellites were the first to be launched between 1957 and 1958, beginning the space age: Sputnik, Explorer and Vanguard [3].

A particularly relevant type of small satellite is the CubeSat. In 1999, California Polytechnic State University and Stanford University developed this new class of satellite with educational purposes [4]. The motivation was to provide students with hands-on experience, allowing them to work on the entire life-cycle of a space project. This motivation required that the development, launching and operation costs were kept as low as possible, which led to project standardization, the use of Commercial Off-The Shelf (COTS) components and the reduction of the number of tests. Later, it became evident that the CubeSats could be used for other purposes rather than education. CubeSats are used for testing technologies and science missions such as those related to astronomy, space weather and Earth observation [5].

The Attitude Determination and Control System (ADCS) is a critical subsystem in a nanosatellite [6]. Only 20% of pico and nanosatellites are left tumbling free in space. Most of the times, the system is used simply for rotation damping. This is important for maintaining a reliable power generation and communications link. About 15% of pico and nanosatellites are equipped with control to point an instrument, generally a nadir pointing camera or a radiation detector pointing along the magnetic field lines. The involved control algorithms and technologies are rather simple when compared to larger satellites and one should not expect more than a rough pointing capability.

A key particularity of space projects is the difficulty to test the technologies in a realistic environment before the mission is deployed. This is due to the fact that the conditions in space are much different from those on Earth, namely in what concerns the atmosphere and gravity. To overcome this difficult,

the ADCS are tested in a computer simulation against realistic models of the conditions it will be subject to when in orbit.

Nevertheless, ADCS development, sensor integration, precision pointing algorithm validation, hardware in the loop verification and performance verification of momentum exchange devices can largely benefit from ground testing [7]. This reduces the risk of failure, by allowing the algorithms to be tested at an earlier stage. Since the beginning of the space race, air bearing based platforms have been used as testbeds for simulating the spacecraft torque-free conditions in space. An air bearing provides an environment as close as possible to that of space, and for this reason it is the preferred technology for ground-based research in spacecraft dynamics and control [8].

As far as the author is aware, there is no such an air-bearing as the one proposed in this work in Portugal yet. Furthermore, space education activities in Portugal lack a more hands-on experience such as the one that is provided in other countries. The main goal of this thesis is to build a low-cost air-bearing based testbed and use it to test the ADCS subsystem of a 1U CubeSat functional prototype, thus validating the technology and providing the know-how to continue with the development of a facility for ground testing of CubeSats.

Although primarily designed for educational purposes, this testbed application reaches beyond it to any payload onboard a nanosatellite requiring attitude control. The testbed allows any ADCS containable withing the 1U standard to go through testing prior to launch. This allows to assess the stability and pointing accuracy.

1.1.1 The NANOSTAR Project

The NANOSTAR Project [9], funded by the Interreg Sudoe Programme, aims at using the nanosatellite standard in Southern Europe universities and companies to attract the best students and engineers. This experience and the network that comes from the consortium of universities, aerospace clusters and ESA Business Incubation Centres provides the conditions for developing the nanosatellite sector in France, Spain and Portugal.

The training is to be provided through *student challenges*, which are specific problems related with the design and development of small satellite missions. There are two types of challenges: the pre-design challenges and the detailed design challenges [10].

Space mission predesign challenges are competitions in which teams of university students develop their design solutions to satisfy a set of mission requirements using the NANOSTAR Software Suite (NSS). Students are assisted by faculty members in the process of creating a satellite mission cooperatively in the framework of concurrent engineering.

This thesis is framed within the context of the *Detailed Design Challenges*. These are research based challenges in which an institution offers specific challenges on the detailed design, development and testing of nanosatellite components.

Up to now, two Space Mission Predesign Challenges were launched, one in February and the other in September 2019. In October 2019, each institution started offering specific *Detailed Design and Test*

Challenges. This work is one of the 38 active challenges as of November 30.

1.2 Topic Overview

Satellites are classified based on their applications, orbits, mass, etc. The mass-based classification is useful since in different classes different technologies are usually employed. The classification ranges of mass vary with organizations. The one in Tab. 1.1 is used by many of them [11].

Class	**Mass (Kg)**
Large Satellite	$>$1000
Medium Satellite	500 - 1000
Small Satellite	$<$500
Minisatellite	100 - 500
Microsatellite	10 - 100
Nanosatellite	1 - 10
Picosatellite	0.1 - 1
Femto-satellite	$<$0.1

Table 1.1: Classification of satellites.

The segment of nanosatellites has shown an incresing popularity and growth [1] due to the lower costs of development and the way they are launched into space. As opposite to traditional satellite launching logic, in which no more than a few satellites are launched by vehicle, there is a greater number of nanosatellites launched by a single vehicle. That, combined with the intense use of the International Space Station (ISS), significantly reduced the costs of deploying a satellite into orbit. These aspects granted easy access to space for a significant number of private companies [5].

CubeSats are a particular type of nanosatellite (or microsatellite in some cases), following The CubeSat Design Specification (CSD) [12], created in 1999. A CubeSat is made up of multiples of units, denoted as ”U”, each unit consisting of a 10 cm cube with a mass of up to 1.33 kg. The most used configuration is 3U [5], but there are several other configurations such as 1U, 2U, 6U and so forth, as depicted in Fig. 1.1.

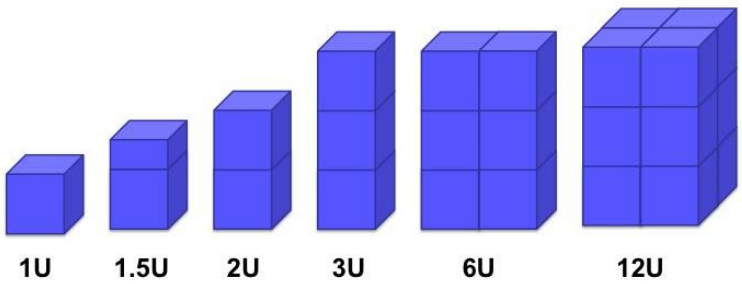


Figure 1.1: CubeSat configurations.From [13].

Attitude determination of a spacecraft means determining its orientation in space with respect to a given reference frame. There are static and recursive attitude determination methods. Static methods are memory-less, meaning that each time the attitude is determined is independent from the others,

ignoring statistical properties of the attitude measurements. The most used static attitude determination methods are, by chronological order: the TRIAD algorithm [14]; Davenport's solution to the Wahba's problem [15]; the QUEST algorithm [16]. Recursive attitude determination methods take advantage from past measurements to provide more accurate solutions and allow for estimates with less than two measurements. The most popular recursive algorithm is the Kalman Filter [17]. Other methods include the Extended Kalman Filter and the Recursive QUEST [17]. Cascade observers such as the one introduced by Batista, Silvestre and Oliveira in [18] can also be used for attitude determination.

Spacecraft attitude control is less well documented than attitude determination. However, a good review on the history of attitude control can be found in [19]. Early satellites in the 1950s-60s did not have any kind of pointing requirements. Spacecraft attitude stabilization methods can be divided into two types. Passive stabilization techniques such as Gravity Gradient stabilization, Passive Magnetic Stabilization, and Aerodynamic stabilization in Low Earth Orbit take advantage of the geometric and magnetic design of a satellite and the orbit properties to passively provide attitude stabilization and rough pointing. As on-board computers became more capable, more demanding pointing requirements became achievable, shifting the design of spacecraft control systems towards active control, namely three-axis stabilization. For CubeSats, magnetorquers are often the primary actuator for attitude control due to their simplicity and low cost [20]. Nowadays, designs adopting reaction / momentum wheels as primary actuators and magnetorquers for momentum dumping are being increasily adopted for better pointing accuracy.

Air bearing based platforms can be divided into 3 types.

Planar platforms provide two translational degrees of freedom and oftentimes also one rotational (vertical spin) [8]. These platforms usually carry their own compressed air tanks and create an air cushion, providing almost frictionless planar sliding motion on a flat surface. Vertical spin motion can be added using a reaction wheel or compressed air thrusters. Planar platforms are of great interest to test docking and rendezvous operations as in the case of this MIT platform [21] or the Naval Postgraduate School Space Robotics Laboratory [22].

Rotational systems aim at providing a frictionless rotational movement. Depending on the geometry of the platform, the pitch, yaw or roll rotations may be limited. *Tabletop* systems consist of a table attached to a spherical air bearing. *Umbrella* systems have a cylindrical bar connecting the table to the spherical air bearing. These two guarantee full freedom yaw rotation and limited pitch and roll rotations. Lastly, the *dumbbell* systems guarantee full freedom yaw and roll rotations and limited pitch rotation. Examples for the *tabletop*, *umbrella* and *dumbbell* systems can be found on [23], [24] and [25] respectively.

The last type are combinational systems that often provide 5 to 6 degrees of freedom. The Marshall Space Flight Center's Flight Robotics Laboratory provides 6 DOF [26]. The Advanced Autonomous Multiple Spacecraft (ADAMUS) laboratory's platform provides 6 degrees of freedom (DOF) [27] and also guarantees dynamical reproduction of motion about the 6 DOF. The attitude stage of this platform provides full 360 degrees of yaw freedom, but only ± 45 about the pitch and roll motions.

The air-bearing-based platform developed under the scope of this thesis is classified as a rotational

system. However, it does not fit into any of the three classical geometries previously described. It consists of an empty spherical air bearing. Inside it, one can fit any 1U CubeSat and test its ADCS.

1.3 Problem Statement

The efforts placed on this master thesis can be divided into four challenges. Firstly, attitude control and determination methods are developed for the selected set of actuators and sensors. Then, these algorithms are tested against a *MATLAB/Simulink* simulation environment that realistically describes the conditions the spacecraft will be subject to in space. A 1U CubeSat functional prototype is designed and built allowing for the test of the algorithms in the air-bearing-based testbed. Lastly, a testbed for ground-testing of 1U CubeSats is designed, built and tested. Experiments on the control of the yaw angle are conducted in order to validate the developed technology, setting the foundations for future work.

The testbed was developed following an Agile methodology (see appendix D). A CubeSat prototype is to be built, along with a testbed able to provide 3 Degrees Of Freedom (DOF) and full 360º rotation around the 3 axis. Previous work on the development of a functional CubeSat three-axis simulator with full 360º freedom of rotation around the 3 axis has been done by [28] and [7], but neither have fulfilled the task of having a functional platform. On 2014, EyasSAT demonstrated a three degreee of freedom CubeSat Air Bearing for classroom purposes, designed in cooperation with the United States Air Force [29]. Inspired on this work, the design and development of an unrestricted satellite motion simulator with a variable inertia model can be found in [30].

The testbed is required to have disturbance torques at most of the same order of magnitude of those those present in space. Requirements for the Attitude Determination and Control System (ADCS) include a pointing accuracy of 20º and a limit angular rate of 5º/s. Also, the satellite should be able to stop tumbling from at least 30º/s. These are the ADCS requirements established for the ISTsat-1 satellite [31] and are the baseline for the present work.

1.4 Thesis Outline

This thesis aims at providing the NANOSTAR project with grounded studies in ADCS algorithms that can be employed in the missions designed under the scope of the project. For that, a *MATLAB/Simulink* simulation environment is developed, allowing the algorithms to be tested against a simulation platform that realistically describes the nanosatellite environment, including orbit generation and propagation. Then, a test bench and a CubeSat prototype are built, allowing for the algorithms to be tested on the ground.

The work developed under the scope of this dissertation is documented in the following chapters:

- **Chapter 2** provides the technical background on attitude representations, the reference frames commonly used in space applications, the equations that describe orbital motion, the attitude dynamics and kinematics of a spacecraft and a model for the environmental forces and torques acting

upon a spacecraft in space;

- **Chapter 3** documents the theory behind the attitude determination and control methods used in this work;
- **Chapter 4** focuses on the design of the CubeSat functional prototype and the testbed, providing the rationale for the different design choices that led to the final configuration;
- **Chapter 5** addresses the implementation of the Attitude Determination and Control System (ADCS) presented in chapter 3, the simulation environment based on equations from chapter 2 and the code deployment and communication protocols involved in the project;
- **Chapter 6** describes and presents the results of the simulation case scenarios used for testing the ADCS algorithms;
- **Chapter 7** presents the experimental results of the testbed and ADCS algorithms validation;
- **Chapter 8** recaps the work performed with remarks and ideas for future work.

Chapter 2

Theoretical Background

2.1 Attitude Representations

To find a model that accurately represents a rigid body, namely, a spacecraft, one needs to find a way to represent the orientation of the body. One way to achieve this is to represent the orientation by the rotation that brings the spacecraft body frame to some reference frame. This means that we need to find a rotational axis and an appropriate rotational angle that rotates one given frame (**X**, **Y**, **Z**) to another given frame (**x**, **y**, **z**).

The attitude of a spacecraft is treated in this work as a mathematical concept. The attitude of a spacecraft belongs to the group of rotation matrices, the Special Orthogonal Group $SO(3) = \{\mathbf{A} \in O(3) : det(\mathbf{A}) = 1\}$.

2.1.1 Euler Angles

Euler angles represent a rotation by a series of rotations about the described coordinate, i.e., first the frame is rotated at an angle γ around the **Z** axis, then it is rotated at an angle β around the **Y** axis and finally it is rotated at an angle α around the **X** axis.

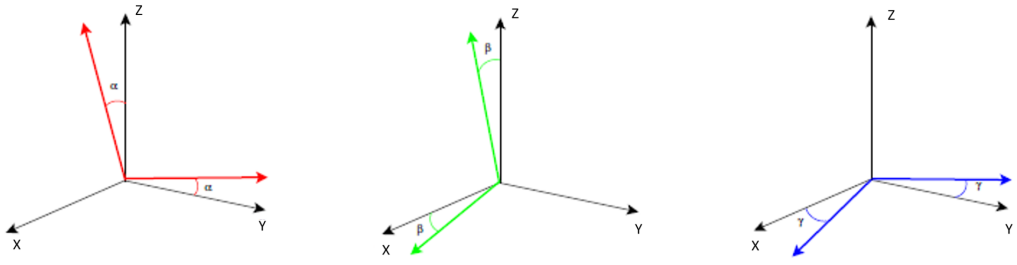


Figure 2.1: Rotations arround each axis. Adapted from [32].

The rotation matrix from the (**X**, **Y**, **Z**) frame to the (**x**, **y**, **z**) is obtained by multiplying the corresponding 3 rotation matrixes:

$$\mathbf{A}_{ZYX}(\gamma, \beta, \alpha) = \mathbf{A}_X(\alpha)\mathbf{A}_Y(\beta)\mathbf{A}_Z(\gamma). \tag{2.1}$$

There is a one-to-one correspondence between Euler angles and rotation matrices only if the Euler angle domains are restricted, e.g. to $0 \leq \alpha < 2\pi$, $0 \leq \beta \leq \pi$, $0 \leq \gamma < 2\pi$.

Euler angles have an intuitive interpretation, however they have the disadvantage of presenting a mathematical singularity. Quaternions are an alternative to overcome the singularity.

2.1.2 Axis-Angle representation

Any rotation represented by a rotation matrix **A** can be represented by a rotation about a fixed axis, **e**, as stated by Euler's theorem. The angle of rotation is denoted by α. Denoting the rotation matrix by

$$\mathbf{A} = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix}, \tag{2.2}$$

the rotation angle can be obtained from

$$cos(\alpha) = \frac{1}{2}(A_{11} + A_{22} + A_{33} - 1). \tag{2.3}$$

If $sin(\alpha) \neq 0$, the rotation axis **e**, with $|\mathbf{e}| = 1$ is defined by

$$\mathbf{e} = \frac{1}{2sin(\alpha)} \begin{bmatrix} A_{23} - A_{32} \\ A_{31} - A_{13} \\ A_{12} - A_{21} \end{bmatrix}. \tag{2.4}$$

The general rotational matrix can be expressed as

$$\mathbf{A} = cos(\alpha)\mathbf{I} + (1 - cos(\alpha))\mathbf{e}\mathbf{e}^T - sin(\alpha)\mathbf{S}(\mathbf{e}). \tag{2.5}$$

where $\mathbf{S}(.)$ is the skew-symmetric function that represents the cross product as defined in appendix A. **A** is also called *Direction Cossine Matrix*.

This equation shows that the rotation matrix is a periodic function (with period 2π) of the rotation angle. Also, note that a rotation by α around **e** is equivalent to a rotation of $-\alpha$ around -**e**.

2.1.3 Quaternion

Quaternion represent a rotation by a rotational angle around a rotational axis, which is not necessarily **X**, **Y** or **Z**. The concept was first introduced by the Irish mathematician William Rowan Hamilton in 1843. The mathematical development of the quaternion concept can be found on the appendix A.

In aerospace engineering one always use a special *normalized quaternion*:

$$\bar{\mathbf{q}} = q_0 + \mathbf{q} = cos(\frac{\alpha}{2}) + \mathbf{e}sin(\frac{\alpha}{2}). \tag{2.6}$$

where **e** is the rotational axis and α is the rotational angle.

The quaternion rotation operator is a linear operator that rotates a vector by an angle α around **e**:

$$\mathbf{w} = \bar{\mathbf{q}} \otimes \mathbf{v} \otimes \bar{\mathbf{q}}^* = (cos^2(\frac{\alpha}{2}) - sin^2(\frac{\alpha}{2}))\mathbf{v} + 2(\mathbf{q}.\mathbf{v})\mathbf{q} + 2q_0(\mathbf{q} \times \mathbf{v}). \tag{2.7}$$

Successive rotations can be easily computed by equation A.16.

The attitude matrix can be computed from the quaternion by equation A.14.

The derivative of a quaternion is given by

$$\frac{d\bar{\mathbf{q}}}{dt} = \bar{\mathbf{q}}(t) \otimes (0 + \frac{1}{2}\mathbf{w}(t)). \tag{2.8}$$

where $\Omega(t) = \frac{d\alpha}{dt}$ is a scalar, $\mathbf{w}(t) = \mathbf{e}(t)\Omega(t)$ is a vector and $(0+\frac{1}{2}\mathbf{w}(t)) = \frac{1}{2}(0, w_1, w_2, w_3)$ is a quaternion. This can also be represented as

$$\begin{bmatrix} \dot{q_0} \\ \dot{q_1} \\ \dot{q_2} \\ \dot{q_3} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & -q_3 & q_2 \\ q_2 & q_3 & q_0 & -q_1 \\ q_3 & -q_2 & q_1 & q_0 \end{bmatrix} \begin{bmatrix} 0 \\ w_1 \\ w_2 \\ w_3 \end{bmatrix}. \tag{2.9}$$

Quaternion can be obtained from the Euler angles, (ψ, θ, ϕ), for a 3-2-1 sequence (also known as Tait-Bryan angles) by the following expression

$$\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} cos(\phi/2)cos(\theta/2)cos(\psi/2) + sin(\phi/2)sin(\theta/2)sin(\psi/2) \\ sin(\phi/2)cos(\theta/2)cos(\psi/2) - cos(\phi/2)sin(\theta/2)sin(\psi/2) \\ cos(\phi/2)sin(\theta/2)cos(\psi/2) + sin(\phi/2)cos(\theta/2)sin(\psi/2) \\ cos(\phi/2)cos(\theta/2)sin(\psi/2) - sin(\phi/2)sin(\theta/2)cos(\psi/2) \end{bmatrix}. \tag{2.10}$$

The quaternion rotational error is defined as in [33]

$$\delta\bar{\mathbf{q}} = \bar{\mathbf{q}} \otimes \bar{\mathbf{q}}_r^{-1}, \tag{2.11}$$

where $\bar{\mathbf{q}}_r^{-1}$ is the quaternion inverse of the reference quaternion. $\delta\bar{\mathbf{q}}$ is a quaternion that defines the desired rotational motion from $\bar{\mathbf{q}}$ to $\bar{\mathbf{q}}_r$.

2.2 Reference Frames

A reference frame is defined by its origin and the orientation of its axis. The attitude representations defined in section 2.1 represent the components of a reference frame along the axis of some other reference frame.

Many reference frames are used in space. In the present section, only the reference frames used in this work will be discussed. For further reading see [34].

2.2.1 North-east Down Frame (NED)

The North-east Down Frame (NED) is a noninertial coordinate system with its origin fixed at the satellite's center of mass. Its $\mathbf{X}_{ned}$ and $\mathbf{Y}_{ned}$ axis are parallel to the geoid surface, meaning that they lie on the local horizontal plane. The $\mathbf{X}_{ned}$ axis points towards the north. The $\mathbf{Y}_{ned}$ axis points towards the east. The $\mathbf{Z}_{ned}$ axis points downwards, towards the Earth on Nadir direction.

2.2.2 Body-Fixed Frame

The origin of the Body-Fixed frame is located at the center of mass of the spacecraft. The axis definition will vary from mission to mission. In the present work, the $\mathbf{Z}_b$ axis is aligned with the main scientific payload. The $\mathbf{X}_b$ axis is aligned with the normal to the bottom plate of the spacecraft. The $\mathbf{Y}_b$ axis follows the right hand rule.

2.2.3 The Earth Centered Inertial (ECI) Frame

The ECI frame is important as the Newton's laws applied to the spacecraft are defined in the inertial frame. On the other hand, many satellites are inertial pointing and require an inertial frame to represent the pointing target. The origin of the ECI frame is the center of the Earth. The $\mathbf{X}_I$ is the direction from the Earth center to the vernal equinox. The $\mathbf{Z}_I$ axis is the Earth rotational axis. The $\mathbf{Y}_I$ follows the right-hand rule.

Any frame moving without rotation and at constant velocity with respect to an inertial frame is also an inertial frame. Since Earth is rotating arround the Sun, the ECI frame has a linear acceleration. However, this acceleration is negligible for attitude determination and control of a spacecraft orbiting the Earth [19].

A more pure inertial frame would be International Celestial Reference Frame (ICRF 3) [35].

2.2.4 Local Vertical Local Horizontal Frame

The Local Vertical Local Horizontal (LVLH) is broadly used in Earth-Observation satellites because its $\mathbf{Z}_{lvlh}$ direction is always pointing towards the center of the Earth (Nadir pointing). The origin of the LVLH frame is the center of mass of a given spacecraft. The $\mathbf{X}_{lvlh}$ direction is the same as the spacecraft velocity's and perpendicular to $\mathbf{Z}_{lvlh}$. Finally, the $\mathbf{Y}_{lvlh}$ direction is perpendicular to the orbital plane and follows the right-hand rule.

In Fig. 2.2, the LVLH frame is represented, along with the orbital path and the ECI frame.

2.3 Orbital Motion

In its simplest form, the orbit of a spacecraft can be modeled as if only two bodies are involved and both behave as spherical point-masses. For this case, where no other forces act on the body, the orbit model

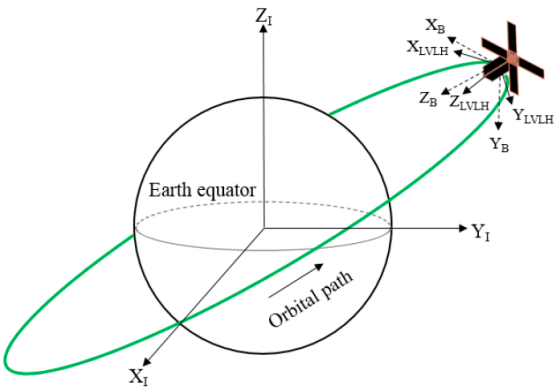


Figure 2.2: LVLH and ECI frames. From [36].

is simplified to a Keplerian orbit, than can be modeled as a conic section. According to this model, the distance, r, between a central body and an orbiting body is

$$r(\theta) = \frac{a(1-e^2)}{1+ecos(\theta)}, \tag{2.12}$$

where θ is the true anomaly, a is the semi-major axis and e is the eccentricity.

Given the ECI frame, the 6 classical keplerian orbital parameters describe a spacecraft on orbit:

- a - the semi-major axis defines the size of the orbit;
- e - the eccentricity defines the shape of the orbit;
- $M = n(t - t_0)$ - the mean anomaly defines the spacecraft position on the orbit (n is the rate of sweep, t is the current time and t_0 is the time the body passed the periapsis);
- i_m - the inclination is the angle between the orbit and the equator planes;
- Ω - the right ascension is the angle between the X_I axis (pointing towards the vernal equinox) and the node line pointing towards the ascending node;
- ω - the argument of the perigee is the angle between the node line pointing towards the ascending node and **P**, where **P** is the unit length vector from the primary focus (the center of the mass of the Earth) pointing to the perigee of the orbit.

The six orbital parameters, $[a, e, M, i_m, \Omega, \omega]$, are illustrated in Fig. 2.3. Clearly, there is another way to present the spacecraft moving around the orbit, given by the velocity, **v**, and position, **r**, at any time. For more details on Keplerian orbits and on the transformations between these representations, resort to [20].

Given the spacecraft position and Earth's gravitational parameter, the acceleration a spacecraft is subject to is obtained by combining Newton's second law with his law of gravitation to obtain the so called two-body equation of motion [37]:

$$\ddot{\mathbf{r}} = -\frac{\mu_\odot}{|\mathbf{r}|^3}\mathbf{r}. \tag{2.13}$$

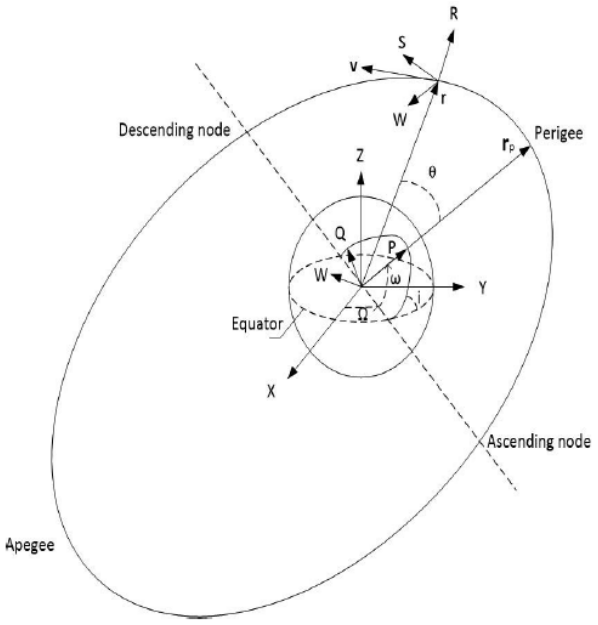


Figure 2.3: Classical Orbital Parameters. From [20].

During its orbit, a spacecraft is subject to forces that deviate it from a perfect Keplerian orbit. These disturbances must be taken into account in the equation of motion with an acceleration term, $\mathbf{a}_d$

$$\ddot{\mathbf{r}} = -\frac{\mu_{\odot}}{|\mathbf{r}|^3}\mathbf{r} + \mathbf{a}_d. \tag{2.14}$$

As a result, instead of the six classical orbital elements being constant, they will be time-varying.

The spacecraft velocity and position is obtained by consecutively integrating its acceleration, given by equation 2.14.

2.4 Attitude Kinematics and Dynamics

A full quaternion model does not have any singular point in any rotational sequence, being thereafter a suitable choice in control design methods.

Regarding a general spacecraft let its inertia matrix be denoted by

$$\mathbf{J} = \begin{bmatrix} J_{11} & J_{12} & J_{13} \\ J_{21} & J_{22} & J_{23} \\ J_{31} & J_{32} & J_{33} \end{bmatrix}. \tag{2.15}$$

Let $\mathbf{w}_I = [w_{I1}, w_{I2}, w_{I3}]^T$ denote the angular velocity of the spacecraft body with respect to the inertial frame and $\mathbf{h}_I$ the angular momentum vector of the spacecraft about its center of mass represented in the inertial frame and $\mathbf{h}$ the same vector as $\mathbf{h}_I$ but represented in the body frame.

The external torque, $\mathbf{T}$, acting on the body about its center of mass can be computed as follows

$$\mathbf{T} = \left(\frac{d\mathbf{h}_I}{dt}\right)_b = \left(\frac{d\mathbf{h}}{dt}\right) + \mathbf{w}_I \times \mathbf{h}, \tag{2.16}$$

which is equivalent to

$$\left(\frac{d\mathbf{h}}{dt}\right) = \mathbf{J}\dot{\mathbf{w}}_I = -\mathbf{w}_I \times \mathbf{J}w_I + \mathbf{T}. \tag{2.17}$$

Ignoring the disturbance torques, $\mathbf{t}_d$, **T** can be replaced by the control torque, **u** to obtain the Dynamics equation:

$$\mathbf{J}\dot{\mathbf{w}}_I = -\mathbf{w}_I \times (\mathbf{J}\mathbf{w}_I) + \mathbf{u}. \tag{2.18}$$

Denoting the quaternion that represents the rotation of the body frame relative to the reference frame by

$$\bar{\mathbf{q}} = [q_0, \mathbf{q}^T]^T = \left[cos(\frac{\alpha}{2}), \mathbf{e}^T sin(\frac{\alpha}{2})\right]^T, \tag{2.19}$$

the Kinematics equation of motion comes directly from 2.9:

$$\begin{cases} \dot{\mathbf{q}} = -\frac{1}{2}\mathbf{w} \times \mathbf{q} + \frac{1}{2}q_0\mathbf{w} \\ \dot{q}_0 = \quad -\frac{1}{2}\mathbf{w}^T\mathbf{q} \end{cases}. \tag{2.20}$$

Using the fact that $q_0 = \sqrt{1 - q_1^2 - q_2^2 - q_3^2}$, one can reduce the system dimension:

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} \sqrt{1 - q_1^2 - q_2^2 - q_3^2} & -q_3 & q_2 \\ q_3 & \sqrt{1 - q_1^2 - q_2^2 - q_3^2} & -q_1 \\ -q_2 & q_1 & \sqrt{1 - q_1^2 - q_2^2 - q_3^2} \end{bmatrix}\begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix}. \tag{2.21}$$

This can be denoted as $\dot{\mathbf{q}} = \frac{1}{2}\mathbf{Q}(\mathbf{q})\mathbf{w} = g(\mathbf{q}, \mathbf{w})$. Since det(**Q**(**q**))=$\frac{1}{\sqrt{1-q_1^2-q_2^2-q_3^2}}$, the matrix **Q** is a full rank matrix except for $\alpha = \pm\pi$, which makes this reduced quaternion representation valid, except for $\alpha = \pm\pi$.

The Inertial Pointing Spacecraft Model

Let $\mathbf{h}_w = [h_1, h_2, h_3]^T$ be the angular momentum of the momentum wheels' system in the body frame. Including $\mathbf{h}_w$, the dynamics equation 2.18 becomes

$$\mathbf{J}\dot{\mathbf{w}}_I = -\mathbf{w}_I \times (\mathbf{J}\mathbf{w}_I + \mathbf{h}_w) + \mathbf{u}, \tag{2.22}$$

and the reduced Kinematics equation is 2.21.

Applying a 1^{st} Order Taylor Expansion around the stationary point, $q_1 = q_2 = q_3 = 0, \mathbf{w}_l = [0, 0, 0]^T, \mathbf{h}_w = [h_1^n, h_2^n, h_3^n]^T$, the linearized model is obtained:

$$\begin{bmatrix} \dot{\mathbf{w}}_I \\ \dot{\mathbf{q}} \end{bmatrix} = \begin{bmatrix} \mathbf{J}^{-1}\mathbf{A}_w & \mathbf{0}_3 \\ \frac{1}{2}\mathbf{I}_3 & \mathbf{0}_3 \end{bmatrix}\begin{bmatrix} \mathbf{w}_I \\ \mathbf{q} \end{bmatrix} + \begin{bmatrix} \mathbf{J}^{-1} \\ \mathbf{0}_3 \end{bmatrix}\mathbf{u} \ = \mathbf{Ax + Bu}, \tag{2.23}$$

with $\mathbf{A}_w$ given by

$$\mathbf{A}_w = \begin{bmatrix} 0 & -h_3^n & h_2^n \\ h_3^n & 0 & -h_1^n \\ -h_2^n & h_1^n & 0 \end{bmatrix}. \tag{2.24}$$

The controllability matrix for an LTI system with dimension n is defined as

$$\mathscr{C} = [\mathbf{B} \quad \mathbf{AB} \quad \mathbf{A}^2\mathbf{B} \quad ... \quad \mathbf{A}^{n-1}\mathbf{B}]. \tag{2.25}$$

Replacing matrix **A** and **B** for the linearized inertial pointing spacecraft model

$$\mathscr{C} = \begin{bmatrix} \mathbf{J}^{-1} & \mathbf{J}^{-1}\mathbf{A}_w\mathbf{J}^{-1} & (\mathbf{J}^{-1}\mathbf{A}_w)^2\mathbf{J}^{-1} & (\mathbf{J}^{-1}\mathbf{A}_w)^3\mathbf{J}^{-1} & (\mathbf{J}^{-1}\mathbf{A}_w)^4\mathbf{J}^{-1} & (\mathbf{J}^{-1}\mathbf{A}_w)^5\mathbf{J}^{-1} \\ \mathbf{0}_3 & \frac{1}{2}\mathbf{J}^{-1} & \frac{1}{2}\mathbf{J}^{-1}\mathbf{A}_w\mathbf{J}^{-1} & \frac{1}{2}(\mathbf{J}^{-1}\mathbf{A}_w)^2\mathbf{J}^{-1} & \frac{1}{2}(\mathbf{J}^{-1}\mathbf{A}_w)^3\mathbf{J}^{-1} & \frac{1}{2}(\mathbf{J}^{-1}\mathbf{A}_w)^4\mathbf{J}^{-1} \end{bmatrix}, \tag{2.26}$$

which is full rank [20], meaning that the system is controllable in the vicinity of the linearization point.

Nadir Pointing Momentum Biased Spacecraft Model

Nadir describes the direction below a particular location, opposite to the Zenith. Nadir-pointing is the action of pointing directly below the satellite perpendicular to Earth. This operation can increase the performance of the attitude control system, allowing more efficient downlinks and solar charge for power.

Let $\mathbf{h}_w = [h_1, h_2, h_3]^T$ be the angular momentum of the momentum wheels' system in the body frame. The momentum of the entire system can be divided between the momentum of the momentum wheel and the momentum of the rigid body. The spacecraft model 2.16 becomes:

$$\mathbf{J}\dot{\mathbf{w}}_I = -\mathbf{w}_I \times (\mathbf{J}\mathbf{w}_I + \mathbf{h}_w) + \mathbf{t}_d + \mathbf{u}. \tag{2.27}$$

For a Nadir Pointing spacecraft, the attitude of the spacecraft is represented by the rotation of the spacecraft body frame relative to the local vertical and local horizontal (LVLH) frame. The quaternion and body rate are measured with respect to the LVLH frame.

In other words, $\mathbf{w} = [w_1, w_2, w_3]^T$ is the body rate with respect to the LVLH frame, represented in the body frame. Denote the LVLH frame rate with respect to the inertial frame represented in the LVLH frame by $\mathbf{w}_{lvlh} = [0, w_0, 0]^T$, where w_0 is the orbit rate. For orbits with low eccentricity, w_0 can be approximately computed assuming a circular orbit with period p at a distance r from the center of the Earth:

$$w_0 = \frac{2\pi}{p}. \tag{2.28}$$

The spacecraft body with respect to the inertial frame can be computed by

$$\mathbf{w}_I = \mathbf{w} + \mathbf{A}_l^b \mathbf{w}_{lvlh}, \tag{2.29}$$

where $\mathbf{A}_l^b$ is the rotation matrix from the LVLH frame to the body frame.

Assuming that $\dot{\mathbf{w}}_{lvlh}$ is small and can be neglected and denoting the rate of the LVLH frame with respect to the inertial frame, represented in the body frame by $\mathbf{w}_{lvlh}^b = \mathbf{A}_l^b \mathbf{w}_{lvlh}$ and noting that $\dot{\mathbf{A}}_l^b = -\mathbf{w} \times \mathbf{A}_l^b$, one can compute $\dot{\mathbf{w}}_I$:

$$\dot{\mathbf{w}}_I = \dot{\mathbf{w}} - \mathbf{w} \times \mathbf{A}_l^b \mathbf{w}_{lvlh} = \dot{\mathbf{w}} - \mathbf{w} \times \mathbf{w}_{lvlh}^b. \tag{2.30}$$

Rearranging and replacing equations 2.29 and 2.27 in the above equation:

$$\begin{aligned}
&\mathbf{J}\dot{\mathbf{w}} = \mathbf{J}\dot{\mathbf{w}}_I + \mathbf{J}(\mathbf{w} \times \mathbf{w}_{lvlh}^b) \\
&\Leftrightarrow \mathbf{J}\dot{\mathbf{w}} = -\mathbf{w}_I \times (\mathbf{J}\mathbf{w}_I + \mathbf{h}_w) + \mathbf{t}_d + \mathbf{u} + \mathbf{J}(\mathbf{w} \times \mathbf{w}_{lvlh}^b) \\
&\Leftrightarrow \mathbf{J}\dot{\mathbf{w}} = \mathbf{J}(\mathbf{w} \times \mathbf{w}_{lvlh}^b) - \mathbf{w} \times (\mathbf{J}\mathbf{w}) - \mathbf{w} \times (\mathbf{J}\mathbf{w}_{lvlh}^b) - \mathbf{w}_{lvlh}^b \times (\mathbf{J}\mathbf{w}) - \mathbf{w}_{lvlh}^b \times (\mathbf{J}\mathbf{w}_{lvlh}^b) \\
&\qquad\qquad - \mathbf{w} \times \mathbf{h}_w - \mathbf{w}_{lvlh}^b \times \mathbf{h}_w + \mathbf{t}_d + \mathbf{u}.
\end{aligned}$$

In short,

$$\mathbf{J}\dot{\mathbf{w}} = \mathbf{f}(\mathbf{w}, \mathbf{w}_{lvlh}^b, \mathbf{h}_w) + \mathbf{t}_d + \mathbf{u}. \tag{2.31}$$

Before writing the linearized attitude equations, one must analyze the main perturbations affecting the satellite. For low-earth orbit spacecraft, both the aerodynamic and the gravity-gradient torques are of great importance. However, the aerodynamic torque may be difficult to model due to its multi-factor provenience. According to [38] the gravity-gradient torque, on the other hand, can be modeled by

$$\mathbf{t}_{gg} = \begin{bmatrix} 3w_0^2(J_{33} - J_{22})\phi \\ 3w_0^2(J_{33} - J_{11})\theta \\ 0 \end{bmatrix}. \tag{2.32}$$

For small Euler angles, one can approximate the roll by $\phi = 2q_1$ and the pitch by $\theta = 2q_2$, leading to

$$\mathbf{t}_{gg} = \begin{bmatrix} 6w_0^2(J_{33} - J_{22})q_1 \\ 6w_0^2(J_{33} - J_{11})q_2 \\ 0 \end{bmatrix}. \tag{2.33}$$

Noticing that the spacecraft body rate with respect to the LVLH frame shall be as small as possible; and also that the spacecraft body frame shall be aligned with the LVLH frame, being the error as small as possible; it follows that the linearized model is the first order Taylor expansion of 2.21 and 2.31 about $\mathbf{h}_w = [h_1^n, h_2^n, h_3^n]^T$, $\mathbf{w} = [0, 0, 0]^T$ and q1 = q2 = q3 = 0. Following the steps from [20], but including 3 momentum wheels instead of 1, one can determine the linearized Nadir Pointing Spacecraft Model:

$$\begin{bmatrix} \dot{\mathbf{w}} \\ \dot{\mathbf{q}} \end{bmatrix} = \begin{bmatrix} \mathbf{J}^{-1}\mathbf{A}_1 & \mathbf{J}^{-1}\mathbf{A}_2 \\ \frac{1}{2}\mathbf{I}_3 & \mathbf{0}_3 \end{bmatrix} \begin{bmatrix} \mathbf{w} \\ \mathbf{q} \end{bmatrix} + \begin{bmatrix} \mathbf{J}^{-1} \\ \mathbf{0}_3 \end{bmatrix} \mathbf{u}, \tag{2.34}$$

where $\mathbf{A}_1$ and $\mathbf{A}_1$ are given by

$$\mathbf{A}_1 = \begin{bmatrix} 0 & -h_3^n & (J_{11} - J_{22} + J_{33})w_0 + h_2^n \\ h_3^n & 0 & -h_1^n \\ -(J_{11} - J_{22} + J_{33})w_0 - h_2^n & h_1^n & 0 \end{bmatrix}, \tag{2.35}$$

$$\mathbf{A}_2 = \begin{bmatrix} 8w_0^2(J_{33} - J_{22}) + 2h_2^n w_0 & 0 & 0 \\ -2h_1^n w_0 & 6w_0^2(J_{33} - J_{11}) & -2h_3^n w_0 \\ 0 & 0 & 2w_0^2(J_{11} - J_{22}) + 2h_2^n w_0 \end{bmatrix}. \tag{2.36}$$

Replacing matrix **A** and **B** in Eq. 2.25

$$\mathscr{C} = \begin{bmatrix} \mathbf{J}^{-1} & \mathbf{J}^{-1}\mathbf{A}_1\mathbf{J}^{-1} & ... \\ \mathbf{0}_3 & \frac{1}{2}\mathbf{J}^{-1} & ... \end{bmatrix}, \tag{2.37}$$

which is also full rank [20], meaning that this model is also controllable in the vicinity of the linearization point.

As a remark, note that equation 2.27 has a structure that is independent of the choice of coordinate system, as long as all quantities are referred to the same coordinate system. As a result, the inertia tensor does change. Rewriting the equation with $\mathbf{u} = -\dot{\mathbf{h}}_b$:

$$\mathbf{J}_b\dot{\mathbf{w}}_b = -\mathbf{w}_b \times (\mathbf{J}_b\mathbf{w}_b + \mathbf{h}_b) - \dot{\mathbf{h}}_b + \mathbf{t}_b, \tag{2.38}$$

where the subscript b is used to denote that the values are measured in the body frame. If the quantities were to be expressed in the principle axis, for example, the structure of the equation would not change. Denoting the rotation matrix from the body frame to the principal axis by $\mathbf{A}_p^b$, the quantities related to the body frame can be expressed by $\mathbf{w}_b = \mathbf{A}_p^b\mathbf{w}_p$ and $\mathbf{J}_b = \mathbf{A}_p^b\mathbf{J}_p\mathbf{A}_p^{bT}$, $\mathbf{h}_p = \mathbf{A}_p^b\mathbf{h}_b$, $\mathbf{t}_b = \mathbf{A}_p^b\mathbf{t}_p$. Replacing in 2.38 yields

$$\frac{d}{dt}(\mathbf{A}_p^b\mathbf{J}_p\mathbf{A}_p^{bT}\mathbf{A}_p^b\mathbf{w}_p) = -\mathbf{A}_p^b\mathbf{w}_p \times (\mathbf{A}_p^b\mathbf{J}_p\mathbf{A}_p^{bT}\mathbf{A}_p^b\mathbf{w}_p + \mathbf{A}_p^b\mathbf{h}_p) - \mathbf{A}_p^b\dot{\mathbf{h}}_p + \mathbf{A}_p^b\mathbf{t}_p. \tag{2.39}$$

Noting that $\mathbf{Aw} \times \mathbf{Ah} = \mathbf{A}(\mathbf{w} \times \mathbf{h})$:

$$\mathbf{J}_p\dot{\mathbf{w}}_p = \mathbf{w}_p \times (\mathbf{J}_p\mathbf{w}_p + \mathbf{h}_p) - \dot{\mathbf{h}}_p + \mathbf{t}_p, \tag{2.40}$$

which is the dynamics in its original structure but now expressed in the principal axis instead of the body axis. This is frequently useful since the inertia tensor in the principal axis is, by definition, diagonal.

2.5 Environment Model

In order to test the attitude control and determination algorithms in a realistic simulation scenario, an accurate model of the environment where the satellite operates is necessary. Disturbance forces affect the accelerations a spacecraft is subject to and, as result, its position. This affects the attitude desired for the spacecraft if one aims to point to a specific object, for example. Disturbance torques, on the other

hand, directly affect a spacecraft's attitude.

Even though the disturbance torques are normally not considered in the models used to design the controller, the designed controller should be able to compensate for these unmodelled disturbances. Given the spacecraft's geometry, position, velocity, the current time, etc, it is possible to simulate these disturbances and test the controller against the space model.

2.5.1 Disturbance Forces

Non-spherical Earth Perturbation

The most used approach to model the non-spherical gravity of the Earth uses a spherical harmonic expansion. In this work, only the second order zonal harmonic of the 6^{th} order geopotential model presented in [39] is considered, since this harmonic is much bigger than the higher order terms. The perturbing acceleration is given by

$$\mathbf{a}_g = -\frac{3}{2} J_2 \left(\frac{\mu_\odot}{|\mathbf{r}|^2}\right)\left(\frac{R}{|\mathbf{r}|}\right)^2 \begin{bmatrix} \left(1-5\left(\frac{r_z}{|\mathbf{r}|}\right)^2\right)\frac{r_x}{|\mathbf{r}|} \\ \left(1-5\left(\frac{r_z}{|\mathbf{r}|}\right)^2\right)\frac{r_y}{|\mathbf{r}|} \\ \left(3-5\left(\frac{r_z}{|\mathbf{r}|}\right)^2\right)\frac{r_z}{|\mathbf{r}|} \end{bmatrix}, \tag{2.41}$$

where J_2 is the zonal coefficient associated with the second order harmonic of the model, R is the radius of the Earth, $\mu_\odot$ is the geocentric gravitational constant, $|\mathbf{r}|$ is the distance of the spacecraft to the center of the earth, and $\mathbf{r} = (r_x, r_y, r_z)$ is the position of the spacecraft in the Earth Centered Earth Fixed frame.

Third Body Forces

The Keplerian motion only takes into consideration two bodies. Third bodies account for perturbations in the orbit of the spacecraft. The motion of a body with mass m_2 at position $\mathbf{r}_2$ about a body with mass m_1 at position $\mathbf{r}_1$, under the influence of N-2 other bodies with masses m_i at position $\mathbf{r}_i$ is, according to [19],

$$\ddot{\mathbf{r}} = -\frac{\mu_\odot}{|\mathbf{r}|^3}\mathbf{r} - \sum_{i=3}^{N} \mu_i \left(\frac{\mathbf{r}_1 - \mathbf{r}_i + \mathbf{r}}{|\mathbf{r}_1 - \mathbf{r}_i + \mathbf{r}|^3} - \frac{\mathbf{r}_1 - \mathbf{r}_i}{|\mathbf{r}_1 - \mathbf{r}_i|^3}\right), \tag{2.42}$$

where $\mathbf{r} = \mathbf{r}_2 - \mathbf{r}_1$, $\mu_\odot = G(m_1 + m_2)$ and $\mu_i = Gm_i$ for $i \geq 3$.

Aerodynamic Drag

An expression for aerodynamic drag follows from the analysis in [19]. The aerodynamic force can be approximated by the following equation

$$\mathbf{f} = -\frac{1}{2}\rho C_D S |\mathbf{v}_{rel}| \mathbf{v}_{rel}, \tag{2.43}$$

where ρ is the atmospheric density, S is the spacecraft area projected along the direction of motion, C_D is a dimensionless drag coefficient. The drag coefficient is determined empirically, and is usually in the

range between 1.5 and 2.5. The spacecraft's relative velocity to the atmosphere in the ECI frame, $\mathbf{v}_{rel}$, can be approximately computed, if one assumes that the atmosphere co-rotates with the Earth:

$$\mathbf{v}_{rel} = \mathbf{v}_I + \mathbf{w}_{Earth} \times \mathbf{r}_I, \tag{2.44}$$

where $\mathbf{v}_I$ and $\mathbf{r}_I$ are the spacecraft's velocity and position in the ECI frame and $\mathbf{w}_{Earth}$ is the Earth's angular velocity. In the ECI, $\mathbf{w}_{Earth} = [0, 0, 0.000072921158553]^T rad/s$

Solar Radiation Pressure

The mean momentum flux pressure acting on a surface normal to the Sun is, according to [20], $P = 4.563 \times 10^{-6} N/m^2$ at 1 astronomic unit from the Sun. For a flat plate, the solar radiation pressure induced force is, according to [40],

$$\mathbf{f} = F_n \mathbf{n} + F_t \mathbf{t}, \tag{2.45}$$

where **t** is the transverse unit vector, perpendicular to **n**, the unit evector normal to the surface and opposite to the vector of incoming photons.

The components of **f** are given by

$$F_n = PA\big[(1 + \rho_s)cos^2(\alpha) + \frac{2}{3}\rho_d cos(\alpha)\big], \tag{2.46}$$

$$F_t = PA(1 - \rho_s)cos(\alpha)sin(\alpha), \tag{2.47}$$

where A is the surface area, ρ_s the fraction of specularly reflected photons, ρ_d the fraction of diffusely reflected photons and ρ_a the fraction of absorved photons.

2.5.2 Disturbance Torques

The order of magnitude of the considered disturbance torques is summarized in table 2.1.

Torque	Magnitude [N.m]
Gravitational	4×10^{-8}
Atmosphere-induced	3×10^{-7}
Magnetic Field-induced	2×10^{-6}
Solar Radiation	8×10^{-9}

Table 2.1: Orders of magnitude of the main disturbance torques [41].

Gravitational Torques

Following the analysis from [20], the gravitational torque or gravity gradient torque, $\mathbf{t}_{gg}$ is given by

$$\mathbf{t}_{gg} = \frac{3\mu_\odot}{|\mathbf{r}|^5} \int (\mathbf{r}.\mathbf{p})(\mathbf{p} \times \mathbf{r}) dm = -\frac{3\mu_\odot}{|\mathbf{r}|^5} \mathbf{R} \times \int \mathbf{p}(\mathbf{p} dm.\mathbf{r}) = \frac{3\mu_\odot}{|\mathbf{r}|^5} \mathbf{r} \times \mathbf{J}\mathbf{r}, \tag{2.48}$$

where dm denotes a small element of the spacecraft, **p** the vector from the center of mass of the spacecraft to dm, **r** the vector from the center of Earth to the center of mass of the spacecraft and $\mu_{\odot} = Gm_1$ the geocentric gravitational constant of the Earth, where m_1 is the mass of the Earth and G is the universal constant of gravitation.

When the body frame is close to the LVLH frame and assuming that $\mathbf{J} = diag([J_{11}, J_{22}, J_{33}])$ and that the velocity of the spacecraft can be approximated by the velocity of a body in a circular orbit of radius $|\mathbf{r}|$, expression 2.32 is obtained.

Atmosphere-Induced Torques

Air density is the most significant factor that causes atmosphere-induced torques. Having the aerodynamic force from section 2.5.1, the aerodynamic torque, $\mathbf{t}_a$, can be evaluated by

$$\mathbf{t}_a = \mathbf{r}_a \times \mathbf{f}, \tag{2.49}$$

where $\mathbf{r}_a$ is the moment arm.

Magnetic Field-Induced Torques

Magnetic disturbance torques are a result of the interaction between the spacecraft's residual magnetic field and the geomagnetic field. The main source of the magnetic disturbance torque is the magnetic moment induced torque [20]. The magnetic moment induced torque is given by

$$\mathbf{t}_m = \mathbf{m} \times \mathbf{B}_m, \tag{2.50}$$

where **m** (in $A.m^2$) is the sum of the individual magnetic moments caused by permanent and induced magnetism and the spacecraft current loops, and $\mathbf{B}_m$ is the geocentric magnetic flux density (in Wb/m^2).

The Earth magnetic field at a specific location and time is obtained using the World Magnetic Model (WMM). The current model is WMM2020, which is valid for the year 2020 through the year 2025. The World Magnetic Models can be found at [42].

Solar Radiation Torques

Given the solar radiation pressure induced force, **f**, the solar pressure induced torque is given by

$$\mathbf{t}_s = \mathbf{r}_s \times \mathbf{f}, \tag{2.51}$$

where $\mathbf{r}_s$ is the vector from the body center of mass to the optical center of pressure.

Chapter 3

Attitude Determination and Control

3.1 Lyapunov Stability Theory

In this work, the concept of Lyapunov stability is used to study the behavior of the spacecraft. The definitions presented in this section follow from [43].

Consider the system

$$\dot{\mathbf{x}} = f(\mathbf{x}) \tag{3.1}$$

Without loss of generality, suppose that **x**=**0** is an equilibrium point of 3.1.

Definition 3.1.1 (Stability)**.** The equilibrium point **x**=**0** of 3.1 is

- *stable* if for each $\epsilon > 0$, there is a $\delta = \delta(\epsilon) > 0$ such that

$$|\mathbf{x}(0)| < \delta \Rightarrow |\mathbf{x}(t)| < \epsilon, \forall t \geq 0$$

- *unstable* if it is not stable;
- *asymptotically stable* if it is stable and δ can be chosen such that

$$|\mathbf{x}(0)| < \delta \Rightarrow \lim_{t \to 0} \mathbf{x}(t) = \mathbf{0}$$

Consider a scalar function $\mathcal{L}(\mathbf{x})$ of the system state. This function is called *positive definite* in a domain $D \subset \mathbb{R}^n$ if

$$\mathcal{L}(\mathbf{x}) > 0 \qquad \forall \mathbf{x} \in D - \{\mathbf{0}\}$$
$$\mathcal{L}(\mathbf{x}) = 0 \qquad if \qquad \mathbf{x} = \mathbf{0}$$

The function is called *positive semidefinite* in a domain $D \subset \mathbb{R}^n$ if

$$\mathcal{L}(\mathbf{x}) \geq 0 \qquad \forall \mathbf{x} \in D$$

A function $\mathcal{L}$ is called negative definite (negative semidefinite) if -$\mathcal{L}$ is poitive definite (positive semidefinite).

Lyapunov's Direct Method makes use of a Lyapunov function, a continuously differentiable function $\mathcal{L} : D \to \mathbb{R}$, for demonstrating stability.

Definition 3.1.2 (Lyapunov's Stability Theorem)**.** Let **x**=**0** be an equilibrium point for the system 3.1 and suppose there is a continuously differentiable function $\mathcal{L}(\mathbf{x})$, defined on a domain $D \subset \mathbb{R}^n$ which contains the equilibrium. If

$$\mathcal{L}(\mathbf{x}) > 0, \quad and$$
$$\dot{\mathcal{L}}(\mathbf{x}) \leq 0$$

in D, then **x**=**0** is stable. If

$$\mathcal{L}(\mathbf{x}) > 0, \quad and$$
$$\dot{\mathcal{L}}(\mathbf{x}) < 0$$

in D, then **x**=**0** is assymptotically stable.

3.1.1 Unwinding

No continuous vector field on SO(3) has a globally asymptotically stable equilibrium, due to the topology of SO(3). This means that the attitude of a spacecraft cannot be globally stabilized by means of a continuous feedback [44]. Consider the unit quaternion parameterization of SO(3) in which $\bar{\mathbf{q}} = -\bar{\mathbf{q}}$. This means that there are two quaternion representing each possible attitude which may lead to large distances traveled to bring the state to a given value even if the initial and desired states are close to each other. This is a well-known phenomenon called unwinding. A continuous feedback based on the unit quaternion representation has two closed-loop equilibrium points. Therefore, global asymptotic stability is not achievable. In the literature, the term almost global stability was coined to express this situation.

3.2 Attitude Control

The attitude control of the spacecraft can be divided into three main algorithms. After its launch, the detumbling algorithm is applied aiming at reducing the angular rate of the spacecraft by means of magnetic torques. This is the so called detumbling mode. The nominal mode corresponds to pointing the spacecraft to the Nadir direction. Finally, the momentum dumping algorithm is applied simultaneously with the pointing algorithm, allowing for continuous dumping of the accumulated angular moment by the momentum wheels. The speed control of the wheels is achieved through a proportional-integral control law and considered as an inner control loop.

3.2.1 Proportional-Integral-Derivative Control

A Proportional-Integral-Derivative (PID) controller is applied for the speed control of the momentum wheels. A PID controller takes the error $e(t) = r(t) - y(t)$, where r(t) is the reference for the output y(t) of the system, and computes the control input for the system:

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau)d\tau + K_d \frac{de(t)}{dt}. \tag{3.2}$$

The proportional term, $K_p e(t)$ compensates for the current error, while the integral term, $\int_0^t e(\tau)d\tau$, takes into account the past error and the differential term, $K_d \frac{de(t)}{dt}$, antecipates the future error, based on its current rate of change. The integral term is capable of compensating for steady-state errors [45].

Note that one may not need to include the 3 terms. Oftentimes, a P, PI or PD controller is enough to attain the requirements of a controller. The design of the weights can be made manually or with more systematic techniques, such as Ziegler–Nichols's method [46].

3.2.2 Nominal Mode

The nominal model corresponds to Nadir Pointing, i.e. aligning the body and LVLH frames. For the Nadir pointing momentum biased spacecraft model from section 2.4, this corresponds to regulating the state $[\mathbf{w}, \mathbf{q}]$. A Linear Quadratic Regulator with Integral Action solution will be studied and compared against the baseline LQR without Integral Action.

The Discrete-Time Linear Quadratic Regulator

Optimal control is one particular branch of modern control. The system that results of an optimal design is not supposed merely to be stable, but is supposed to be the best possible system of a particular type. In linear optimal control the plant that is controlled is assumed to be linear, and the controller is constrained to be linear.

The Linear Quadratic Regulator (LQR) is an optimal controller that minimizes a quadratic performance index.

Consider the discrete-time LTI system:

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k \\ \mathbf{y}_k &= \mathbf{C}x_k, \end{aligned} \tag{3.3}$$

where the subscript k identifies the sample at time kT, where T is the sampling period and k is an integer number. The state is $\mathbf{x}_k = [\mathbf{w}_k, \mathbf{q}_k]^T$ and the matrices **A** and **B** are obtained applying the forward Euler method to equation 2.34

$$\begin{bmatrix} \mathbf{w}_{k+1} \\ \mathbf{q}_{k+1} \end{bmatrix} = \begin{bmatrix} \mathbf{I}_3 + T\mathbf{J}^{-1}\mathbf{A}_1 & T\mathbf{J}^{-1}\mathbf{A}_2 \\ \frac{T}{2}\mathbf{I}_3 & \mathbf{I}_3 \end{bmatrix} \begin{bmatrix} \mathbf{w}_k \\ \mathbf{q}_k \end{bmatrix} + \begin{bmatrix} T\mathbf{J}^{-1} \\ \mathbf{0}_3 \end{bmatrix} \mathbf{u}_k. \tag{3.4}$$

The steady state LQR performance index can be formulated as follows

$$V = \frac{1}{2}\sum_{k=0}^{\infty}(\mathbf{x}_k^T\mathbf{Q}_L\mathbf{x}_k + \mathbf{u}_k^T\mathbf{R}_L\mathbf{u}_k). \tag{3.5}$$

The minimization leads to the linear state feedback [47]

$$\mathbf{u}_k = -\mathbf{K}_L\mathbf{x}_k, \tag{3.6}$$

with $\mathbf{K}_L$ given by

$$\mathbf{K}_L = (\mathbf{R}_L + \mathbf{B}^T\mathbf{S}\,\mathbf{B})^{-1}\mathbf{B}^T\mathbf{S}\,\mathbf{A}, \tag{3.7}$$

where S is the steady-state solution to the algebraic Ricatti equation:

$$\mathbf{S} = \mathbf{A}^T(\mathbf{S} - \mathbf{S}\mathbf{B}\mathbf{R}_L^{-1}\mathbf{B}^T\mathbf{S})\mathbf{A} + \mathbf{Q}_L. \tag{3.8}$$

LQR Extension - Integral Action

When there are disturbances or the model to be controlled is different from the dynamical model considered when computing the LQR gains, there can be a steady-state error.This drawback can be overcome by augmenting the state space with an extra state that corresponds to the integral of the attitude **q**. The concept of integral feedback is described in detail in [48]. The regulation error can be defined as $\mathbf{z}_k = \mathbf{r} - \mathbf{C}\mathbf{x}_k$, where **r** is a constant reference. The reformulated dynamic model is given by

$$\begin{bmatrix}\mathbf{z}_{k+1}\\ \Delta\mathbf{x}_{k+1}\end{bmatrix} = \begin{bmatrix}\mathbf{I} & -\mathbf{C}\mathbf{A}\\ \mathbf{0} & \mathbf{A}\end{bmatrix}\begin{bmatrix}\mathbf{z}_k\\ \Delta\mathbf{x}_k\end{bmatrix} + \begin{bmatrix}-\mathbf{C}\mathbf{B}\\ \mathbf{B}\end{bmatrix}\Delta\mathbf{u}_k, \tag{3.9}$$

where $\Delta\mathbf{u}_k = \mathbf{u}_k - \mathbf{u}_{k-1}$ and $\Delta\mathbf{x}_k = \mathbf{x}_k - \mathbf{x}_{k-1}$. Applying the regulator problem, the constant gain $\mathbf{K} = [\mathbf{K}_z, \mathbf{K}_x]$ is obtained and the control input is given by

$$\mathbf{u}_k = \sum \Delta\mathbf{u}_k = \mathbf{K}_z\sum \mathbf{z}_k + \mathbf{K}_x\mathbf{x}_k. \tag{3.10}$$

This way, an LQR with integral action is obtained and, if a compensator that stabilizes the system is found, a steady state null error for a step is attained.

3.2.3 Momentum Dumping

The main actuator of the proposed ADCS system is the momentum wheels' system. When using a Reaction/ Momentum Wheels' system to counteract perturbations, a periodic torque typically results in a cyclic variation of the wheel's angular rate, whilst a constant torque results in a linear increase in angular rate, so as to compensate for the excess angular momentum due to the external disturbance sources. This is only possible up to the limits for the spin of rate of the wheel, resulting in a saturation of the actuator. A desaturation process of the wheels must be carried out, generating external torques resorting to an additional suite of actuators.

Magnetic torquers are commonly used in low-Earth orbiting spacecraft for momentum dumping. The torque generated by magnetic torquers is given by

$$\mathbf{T} = \mathbf{m} \times \mathbf{B}_m, \tag{3.11}$$

where $\mathbf{m}$ [Am^2] is the commanded magnetic dipole moment generated by the torquers and $\mathbf{B}_m$ [T] is the local geomagnetic field expressed in body-frame coordinates.

An issue that affects the magnetic torquers is that their actuation is constrained to lie in a two-dimensional plane orthogonal to the magnetic field. Nevertheless, full three-axis control is available provided that the spacecraft's orbital plane does not coincide with the geomagnetic equatorial plane and does not contain the magnetic poles [19].

A common approach to design a magnetic torquer control law for momentum dumping is to command a magnetic dipole moment [19]:

$$\mathbf{m} = \frac{k_{md}}{|\mathbf{B}_m|} \Delta\mathbf{h}_w \times \mathbf{b}, \tag{3.12}$$

where $\Delta\mathbf{h}_w = \mathbf{h}_w^n - \mathbf{h}_w$ is the momentum wheel system angular momentum tracking error, $\mathbf{b} = \mathbf{B}_m/|\mathbf{B}_m|$ and k_{md} is a positive scalar gain.

The resulting torque is given by

$$\mathbf{T}_{md} = \frac{k_{md}}{|\mathbf{B}_m|} \Delta\mathbf{h}_w \times \mathbf{b} \times \mathbf{B}_m = k_{md}(\Delta\mathbf{h}_w \times \mathbf{b}) \times \mathbf{b} = -k_{md}(\mathbf{I}_3 - \mathbf{b}\mathbf{b}^T)\Delta\mathbf{h}_w. \tag{3.13}$$

As referred before, the torque cannot be exerted when $\Delta\mathbf{h}_w$ is parallel to $\mathbf{b}$, but, according to [19], this is not a concern for practical reasons.

The use of this torque in addition to the momentum wheel's torque that comes from the linear quadratic regulator feedback, $\mathbf{T}_c$, results in an excess of torque in the spacecraft. To counteract this, a de-spin torque is imposed on each wheel:

$$\mathbf{T}_{wmd} = \mathbf{T}_{md}. \tag{3.14}$$

The total control torque applied to the spacecraft is $\mathbf{T}_c - \mathbf{T}_{wmd} + \mathbf{T}_{md} = \mathbf{T}_c$. This way, the resulting torque acting on the spacecraft due to the desaturation process is null and the momentum dumping controller may be applied continuously without influencing the nominal control solution (LQR).

3.2.4 Detumbling Mode

After its launch, a spacecraft is characterized by large angular velocities. Thus, it is said to be tumbling. Detumbling is hard to be accomplished with the momentum wheels alone, since the saturation limits would be rapidly achieved.

The proposed controller for detumbling consists on commanding the following magnetic dipole moment:

$$\mathbf{m} = \frac{k_{det}}{|\mathbf{B}_m|}\mathbf{w}_I \times \mathbf{b}, \tag{3.15}$$

where $\mathbf{w}_I$ is the spacecraft's angular rate, $\mathbf{b} = \mathbf{B}_m/|\mathbf{B}_m|$ and k_{det} is a positive scalar gain. The theory behind this control law can be found on [49].

By analogy to 3.13, the control torque is given by

$$\mathbf{T}_{det} = -k_{det}(\mathbf{I}_3 - \mathbf{b}\mathbf{b}^T)\mathbf{w}_I. \tag{3.16}$$

Again, the control torque is perpendicular to **b**. Reference [49] provides an expression for k_{det} based on analyzing the closed-loop dynamics of the component of $\mathbf{w}_I$ perpendicular to the $\mathbf{B}_m$:

$$k_{det} = \frac{4\pi}{p}(1 + sin(i_m))J_{min}, \tag{3.17}$$

where p is the orbit period, i_m is its inclination and J_{min} is the minimum principal moment of inertia.

Stability of the Detumbling Control Law

Stability of the Detumbling Control Law can be proven when the body rate is relative to the inertial frame. Considering the following Lyapunov function:

$$\mathcal{L} = \frac{1}{2}\mathbf{w}_I^T\mathbf{J}\mathbf{w}_I. \tag{3.18}$$

If **J** is diagonal and recalling expression 2.22, the derivative of the Lyapunov function is

$$\dot{\mathcal{L}} = \frac{d\mathcal{L}}{d\mathbf{w}_I}\frac{d\mathbf{w}_I}{dt} \tag{3.19}$$

$$= \frac{1}{2}\mathbf{w}_I^T(\mathbf{J} + \mathbf{J}^T)(-\mathbf{J}^{-1}\mathbf{w}_I \times (\mathbf{J}\mathbf{w}_I) - \mathbf{J}^{-1}\mathbf{w}_I \times \mathbf{h}_w - \mathbf{J}^{-1}\mathbf{T}_{det}) \tag{3.20}$$

$$= -\mathbf{w}_I^T\mathbf{J}\mathbf{J}^{-1}\mathbf{T}_{det} \tag{3.21}$$

$$= -k_{det}\mathbf{w}_I^T(\mathbf{I}_3 - \mathbf{b}\mathbf{b}^T)\mathbf{w}_I, \tag{3.22}$$

which is negative semi-definite, since when **w** is parallel to **b**, then $\dot{\mathcal{L}} = 0$, meaning that the control law stabilizes the system, but asymptotic stability is not guaranteed. According to [49], this is not a concern in practice, though.

Note that if no angular velocity information is available, one can find an approximation based on the derivative of the magnetic field vector:

$$\dot{\mathbf{B}}_m = \mathbf{A}_I^b\dot{\mathbf{B}}_{m_I} - \mathbf{w}_I \times \mathbf{B}_m. \tag{3.23}$$

Assuming that $\dot{\mathbf{B}}_{m_I} << \dot{\mathbf{B}}_m$, then $\dot{\mathbf{B}}_m \approx \mathbf{w} \times \mathbf{B}_m$ and expression 3.15 can be approximated by

$$\mathbf{m} = -\frac{k_{det}}{|\mathbf{B}_m|}\dot{\mathbf{B}}_m. \tag{3.24}$$

This is an alternative version of the well-known B-dot control [19].

3.3 Attitude Determination

Regarding the rate gyroscope, the integration of its data makes the estimated angles drift from the real value with time, highlighting the need of a high-pass filter. With a 3-axis accelerometer, an inclinometer can be obtained, in which long time measurements are reliable. However, disturbances affect the measurement and, furthermore, do not respond as fast as the gyroscope for fast maneuvers. As a result, the use of a low-pass filter is common. The estimation of a so called complementary filter is the combination of the low frequency component of the inclinometer and high frequency component of the rate gyroscope integral.

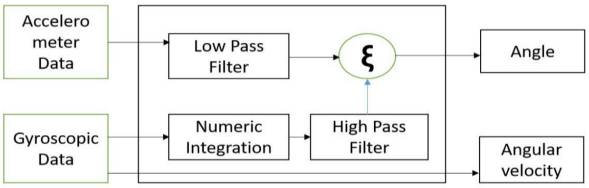


Figure 3.1: Schematic of the structure of a complementary filter based on gyroscope and accelerometer data. From [50].

3.3.1 Euler-Based Complementary Filter

The description of the filter presented in this section is based on the work of [51].

Let $\boldsymbol{\lambda} = [\phi \quad \theta \quad \psi]^T$ be the vector containing the Euler angles roll, pitch and yaw respectively. The kinematics is given by

$$\dot{\boldsymbol{\lambda}} = \mathbf{Q}(\boldsymbol{\lambda})\boldsymbol{w}, \tag{3.25}$$

where $\boldsymbol{w} = [w_x w_y w_z]^T$ and

$$\mathbf{Q}(\boldsymbol{\lambda}) = \begin{bmatrix} 1 & sin(\phi)tan(\theta) & cos(\phi)tan(\theta) \\ 0 & cos(\phi) & -sin(\phi) \\ 0 & sin(\phi)sec(\theta) & cos(\phi)sec(\theta) \end{bmatrix}. \tag{3.26}$$

The discrete-time equivalent is obtained applying the Euler method (also called forward Euler method):

$$\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + \Delta t \mathbf{Q}(\boldsymbol{\lambda}_k)\boldsymbol{w}_k. \tag{3.27}$$

The gyroscopes provide the angular velocity measurement subject to noise:

$$\boldsymbol{w}_{sk} = \boldsymbol{w}_k + \boldsymbol{y}_k, \tag{3.28}$$

where $\boldsymbol{\omega}_k$ represents the nominal angular velocity and $\boldsymbol{y}_k$ is the output of the Allan variance rate gyros noise model. For one axis, the noise model is given by

$$\begin{aligned} \mathbf{b}_{wk+1} = \begin{bmatrix} b_{w1} \\ b_{w2} \end{bmatrix}_{k+1} &= \begin{bmatrix} \left(1 - \frac{1}{T_{BI}}\Delta t\right) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} b_{w1} \\ b_{w2} \end{bmatrix}_k + \begin{bmatrix} \sqrt{\sigma_b^2(1 - e^{-2\Delta t/T_{BI}})} \\ \sqrt{K^2 \Delta t} \end{bmatrix} \upsilon_k, \\ y_k &= \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} b_{w1} \\ b_{w2} \end{bmatrix}_k + \left[\sqrt{\frac{N^2}{\Delta t}} \zeta_k\right] \end{aligned}, \tag{3.29}$$

where υ_k and ζ_k are uncorrelated Gaussian zero-mean white noises; T_{BI} is the correlation time of the bias instability (BI) noise; σ_b^2 is the variance of the BI; K is the Allan variance rate random walk (RRW); N is the angle random walk (ARW).

Incorporating the model in equation 3.29 into 3.27 yields

$$\begin{bmatrix} \boldsymbol{\lambda} \\ \boldsymbol{b_{w1}} \\ \boldsymbol{b_{w2}} \end{bmatrix}_{k+1} = \begin{bmatrix} \mathbf{I} & -\Delta t \mathbf{Q}(\boldsymbol{\lambda}_k) & -\Delta t \mathbf{Q}(\boldsymbol{\lambda}_k) \\ \mathbf{0} & \mathbf{I} - \Delta t \mathbf{T}_{BI}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{\lambda} \\ \boldsymbol{b_{w1}} \\ \boldsymbol{b_{w2}} \end{bmatrix}_k + \begin{bmatrix} \Delta t \mathbf{Q}(\boldsymbol{\lambda}_k) \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \boldsymbol{w}_{sk} + \begin{bmatrix} -\Delta t \mathbf{Q}(\boldsymbol{\lambda}_k) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{n}_{arw} \\ \mathbf{n}_{bi} \\ \mathbf{n}_{rrw} \end{bmatrix}, \tag{3.30}$$

where $\boldsymbol{b_{\omega 1}}$ and $\boldsymbol{b_{\omega 2}}$ are the sensor bias vectors related to the angular velocity $\boldsymbol{\omega}$; $\mathbf{n}_{arw}$, $\mathbf{n}_{bi}$ and $\mathbf{n}_{rrw}$ are the white-noises from the gyro noise model; $\mathbf{T}_{BI}$ is a diagonal matrix with the correlation times of the BI noise.

The observer proposed in [51] is

$$\begin{aligned} \begin{bmatrix} \hat{\boldsymbol{\lambda}} \\ \hat{\boldsymbol{b}}_{\boldsymbol{w1}} \\ \hat{\boldsymbol{b}}_{\boldsymbol{w2}} \end{bmatrix}_{k+1} &= \begin{bmatrix} \mathbf{I} & -\Delta t \mathbf{Q}(\boldsymbol{\lambda}_k) & -\Delta t \mathbf{Q}(\boldsymbol{\lambda}_k) \\ \mathbf{0} & \mathbf{I} - \Delta t \mathbf{T}_{BI}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\lambda}} \\ \hat{\boldsymbol{b}}_{\boldsymbol{w1}} \\ \hat{\boldsymbol{b}}_{\boldsymbol{w2}} \end{bmatrix}_k + \begin{bmatrix} \Delta t \mathbf{Q}(\boldsymbol{\lambda}_k) \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \boldsymbol{w}_{sk} \\ &+ \begin{bmatrix} \mathbf{Q}(\boldsymbol{\lambda}_k)(\boldsymbol{K_{1\lambda}} - \mathbf{I}) + \mathbf{Q}(\boldsymbol{\lambda}_{k-1}) \\ \boldsymbol{K_{2\lambda}} \\ \boldsymbol{K_{3\lambda}} \end{bmatrix} (\boldsymbol{y_{\lambda k}} - \hat{\boldsymbol{y}}_{\boldsymbol{\lambda k}}) \end{aligned}, \tag{3.31}$$

where

$$\begin{aligned} \hat{\mathbf{y}}_{\lambda_k} &= \mathbf{Q}^{-1}(\lambda_{k-1}) \hat{\boldsymbol{\lambda}}_k \\ \boldsymbol{y}_{\lambda k} &= \mathbf{Q}^{-1}(\lambda_{k-1}) \boldsymbol{\lambda}_k + \boldsymbol{\Theta}_k \end{aligned}, \tag{3.32}$$

where $\boldsymbol{y}_{\lambda k}$ is the measured Euler angles transformedd to the space of the angular rate and corrupted by zero-mean white Gaussian-noise $\boldsymbol{\Theta}$. The attitude observation $\hat{\boldsymbol{\lambda}}$ can be obtained by measuring the

Earth's gravitational and magnetic fields by an IMU. $\boldsymbol{K_{1\lambda}}$, $\boldsymbol{K_{2\lambda}}$, $\boldsymbol{K_{3\lambda}} \in M(3,3)$ are observer gains equal to the Kalman gain, $\mathbf{K} = [\mathbf{K}_1, \mathbf{K}_2, \mathbf{K}_3]^T$, from the auxiliary LTI system obtained replacing $\mathbf{Q}(\boldsymbol{\lambda})$ with $\mathbf{I}$ and $\omega_k = \mathbf{0}$:

$$\begin{aligned}
\begin{bmatrix} \boldsymbol{x_\lambda} \\ \boldsymbol{x_{b_{w1}}} \\ \boldsymbol{x_{b_{w2}}} \end{bmatrix}_{k+1} &= \begin{bmatrix} \mathbf{I} & -\Delta t\mathbf{I} & -\Delta t\mathbf{I} \\ \mathbf{0} & \mathbf{I} - \Delta t\mathbf{T}_{BI}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{x_\lambda} \\ \boldsymbol{x_{b_{w1}}} \\ \boldsymbol{x_{b_{w2}}} \end{bmatrix}_k + \begin{bmatrix} -\Delta t\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{n}_{arw} \\ \mathbf{n}_{bi} \\ \mathbf{n}_{rrw} \end{bmatrix}. \\
\mathbf{y}_{xk} &= \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{x_\lambda} \\ \boldsymbol{b_{w1}} \\ \boldsymbol{b_{w2}} \end{bmatrix} + \boldsymbol{\Theta}_k
\end{aligned} \tag{3.33}$$

In [51], the asymptotic stability of the observer 3.31 is proved if the pitch is bounded $|\theta| < \theta_{max} < \frac{\pi}{2}$.

3.3.2 Quaternion-Based Complementary Filter

This filter was inspired by the work of [52]. Firstly, the quaternion from Earth's gravity and magnetic fields observations is computed.

Prediction

In the prediction step, the angular velocity vector from the rate gyros is used to compute an estimation of the orientation in quaternion form. Denoting the gyro measurements by $\boldsymbol{w} = [w_x, w_y, w_z]^T$ and the quaternion with the filter estimation by $\hat{\mathbf{q}} = [\hat{q}_0, \hat{q}_1, \hat{q}_2, \hat{q}_3]^T$, the quaternion resulting from the prediction step comes directly from applying the Euler method to equation 2.20, yielding

$$\begin{aligned}
q_{0k+1} &= q_{0k} + \frac{1}{2}(-w_x q_{1k} - w_y q_{2k} - w_z q_{3k})\Delta t \\
q_{1k+1} &= q_{1k} + \frac{1}{2}(w_x q_{0k} + w_z q_{2k} - w_y q_{3k})\Delta t \\
q_{2k+1} &= q_{2k} + \frac{1}{2}(w_y q_{0k} - w_z q_{1k} + w_x q_{3k})\Delta t \\
q_{3k+1} &= q_{3k} + \frac{1}{2}(w_z q_{0k} + w_y q_{1k} - w_x q_{2k})\Delta t.
\end{aligned} \tag{3.34}$$

At the end of the prediction step, the quaternion is normalized. If the gyros' output was the exact angular velocity, no update step would be required. However, as depicted in section 3.3.1, the gyro is corrupted by noise, including a bias component which would make this estimate drift with time preventing it from being an adequate estimate.

Update

In this step, the Spherical Linear Interpolation (SLERP) technique (see [53]) is used to find an intermediate quaternion between the one from the prediction step and the one that comes from the Earth's gravity and magnetic fields observations.

Spherical Linear Interpolation (SLERP)

Given two quaternions $\bar{\mathbf{q}}_0$ and $\bar{\mathbf{q}}_1$, SLERP can be represented by a function $\bar{\mathbf{q}}(t), t \in [0,1]$ that linearly interpolates from $\bar{\mathbf{q}}(0) = \bar{\mathbf{q}}_0$ and $\bar{\mathbf{q}}(1) = \bar{\mathbf{q}}_1$. As t evolves from 0 to 1, a body will continuously rotate from $\bar{\mathbf{q}}_0$ to $\bar{\mathbf{q}}_1$ at a constant rate along a fixed axis. This function is given by

$$\bar{\mathbf{q}}(t) = \bar{\mathbf{q}}_0 \otimes (\bar{\mathbf{q}}_0^* \otimes \bar{\mathbf{q}}_1)^t. \tag{3.35}$$

An approach to compute $\bar{\mathbf{q}}(t)$ comes from treating $\bar{\mathbf{q}}_0$ and $\bar{\mathbf{q}}_1$ as unit vectors in the unit sphere as depicted in Fig. 3.2. $\bar{\mathbf{q}}(t)$ is the unit vector that follows at a constant speed the shortest spherical path joining $\bar{\mathbf{q}}_0$ and $\bar{\mathbf{q}}_1$.

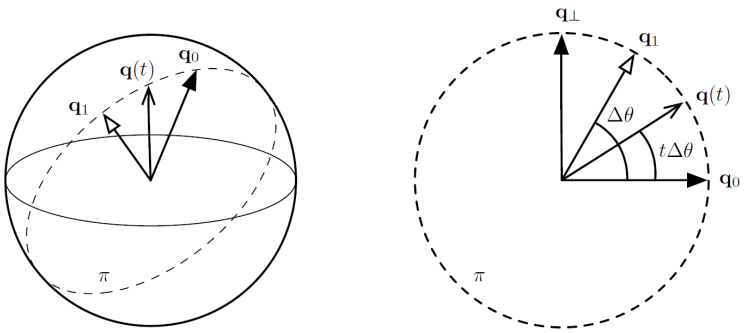


Figure 3.2: Quaternion interpolation in the unit sphere of $\mathbb{R}^4$ and a frontal view of the situation on the rotation plane π of $\mathbb{R}^4$. From [53].

The angle between $\bar{\mathbf{q}}_0$ and $\bar{\mathbf{q}}_1$ is derived from the scalar product.

Remembering that according to 2.6 an unit quaternion is defined by its rotational axis and rotational angle, the following methodology can be used to compute $(\bar{\mathbf{q}}_0^* \otimes \bar{\mathbf{q}}_1)^t = [cos(t\Delta\theta), \mathbf{u}sin(t\Delta\theta)]$. The rotational axis, **u**, is the same for $\Delta\bar{\mathbf{q}}$ and $(\bar{\mathbf{q}}_0^* \otimes \bar{\mathbf{q}}_1)^t$. Only the rotational angle changes proportionally to t.

1. Compute the orientation increment: $\Delta\bar{\mathbf{q}} = \bar{\mathbf{q}}_0^* \otimes \bar{\mathbf{q}}_1$;
2. Normalize it $\Delta\bar{\mathbf{q}} = \frac{\Delta\bar{\mathbf{q}}}{||\Delta\bar{\mathbf{q}}||}$;
3. Denote the orientation increment by $\Delta\bar{\mathbf{q}} = [cos(\Delta\theta), \mathbf{u}sin(\Delta\theta)]$;
4. Compute $\Delta\theta = cos^{-1}(\Delta\bar{\mathbf{q}}_0)$;
5. Compute $sin(t\Delta\theta)$ and $cos(t\Delta\theta)$;
6. Take only the vector part of $\Delta\bar{\mathbf{q}}$, $\mathbf{v} = [0, \mathbf{u}sin(\Delta\theta)]$;
7. Normalize it, yielding $\mathbf{u}' = \frac{\vec{\mathbf{v}}}{||\mathbf{v}||} = [0, \mathbf{u}]$;
8. Compute $(\bar{\mathbf{q}}_0^* \otimes \bar{\mathbf{q}}_1)^t = [cos(t\Delta\theta), \mathbf{u}sin(t\Delta\theta)]$.

This method can be used to interpolate between the predicted quaternion and the one computed from the gravity and magnetic fields. Using SLERP has the advantage of reducing computational complexity by avoiding matrix operations.

3.3.3 Discrete Kalman Filter

The kalman filter is a current observer, meaning that for a given system

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k$$
$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}_k,$$

the estimation is based on the following equation

$$\hat{\mathbf{x}}_{k+1} = \mathbf{A}\hat{\mathbf{x}}_k + \mathbf{B}\mathbf{u}_{k-1} + \mathbf{K}(\mathbf{y}_k - \mathbf{C}(\mathbf{A}\hat{\mathbf{x}}_{k-1} + \mathbf{B}\mathbf{u}_{k-1})). \tag{3.36}$$

In the Kalman filter, the gain **K** is determined such that it optimizes the signal to noise ratio. For that, two terms are added to the state-space equations: the process noise, **w**; the sensor noise, **v**. Both noises are modeled as incorrelated additive white-gaussian noises, with variances $E[\mathbf{w}_k^T\mathbf{w}_k] = \mathbf{Q}$ and $E[\mathbf{v}_k^T\mathbf{v}_k] = \mathbf{R}$ respectively (E denotes the expected value).

The Kalman filter estimator equation is often written as

$$\hat{\mathbf{x}}_{k|k} = \mathbf{A}\hat{\mathbf{x}}_{k-1|k-1} + \mathbf{B}\mathbf{u}_{k-1} + \mathbf{K}[\mathbf{y}_k - \mathbf{C}(\mathbf{A}\hat{\mathbf{x}}_{k-1|k-1} + \mathbf{B}u_{k-1}], \tag{3.37}$$

which is a recursive equation and can be rewritten as

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{A}\hat{\mathbf{x}}_{k-1|k-1} + \mathbf{B}\mathbf{u}_{k-1}, \tag{3.38}$$

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}[\mathbf{y}_k - \hat{\mathbf{x}}_{k|k-1}]. \tag{3.39}$$

Equation 3.38 is the prediction and equation 3.39 is the update, correcting the prediction with the Kalman gain multiplied by the difference between the observation and the predicted observation. The Kalman filter gain is obtained minimizing the expected sum of the squares of the successive estimation errors:

$$\mathbf{V}_k = E\Big[\frac{1}{2}\sum_{k=0}^{\infty}||\mathbf{x}_k - \hat{\mathbf{x}}_k||^2\Big]. \tag{3.40}$$

Quaternion-Based Kalman Filter

The kinematics of the quaternion is given by equation 2.21 that can be rewritten, for simplification, as

$$\dot{\boldsymbol{q}} = \mathbf{F}(\mathbf{q})\mathbf{w}. \tag{3.41}$$

The discrete-time equivalent is once again obtained applying the Euler method:

$$\mathbf{q}_{k+1} = \mathbf{q}_k + \Delta t\mathbf{F}(\mathbf{q})\mathbf{w}. \tag{3.42}$$

The angular velocity, measured by a rate gyro, is modeled as being affected by noise and rate random-walk (RRW):

$$\begin{aligned} \boldsymbol{w}_{rk} &= \boldsymbol{w}_k + \mathbf{b}_{wk} + \mathbf{w}_{w_r k} \\ \mathbf{b}_{wk+1} &= \mathbf{b}_{wk} + \mathbf{w}_{bk} \end{aligned}, \tag{3.43}$$

where $\mathbf{w}_{w_r} \sim \mathscr{N}(\mathbf{0}, \boldsymbol{\Theta}_w)$ is zero-mean white Gaussian-noise; $\mathbf{b}_w$ is the rate gyro bias; $\mathbf{w}_b \sim \mathscr{N}(\mathbf{0}, \boldsymbol{\Theta}_b)$ is the white Gaussian noise that affects the sensor bias.

Integrating the quaternion kinematics 3.42 and the gyro noise model in equation 3.43 into a state-space model:

$$\begin{bmatrix} \mathbf{q}_{k+1} \\ \mathbf{b}_{k+1} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & -\Delta t \mathbf{F}(\mathbf{q}) \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{q}_k \\ \mathbf{b}_k \end{bmatrix} + \begin{bmatrix} \Delta t \mathbf{F}(\mathbf{q}) \\ \mathbf{0} \end{bmatrix} \mathbf{w}_{rk} + \begin{bmatrix} -\Delta t \mathbf{F}(\mathbf{q}) & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{w}_{w_r k} \\ \mathbf{w}_{bk} \end{bmatrix}. \tag{3.44}$$

If the noise model from 3.29 was to be considered, the state space model would be similar to the one in equation 3.30, replacing $\boldsymbol{\lambda}$ by $\mathbf{q}$ and $\mathbf{Q}(\boldsymbol{\lambda}_k)$ by $\mathbf{F}(\mathbf{q})$.

Linearizing the model 3.44 around $\bar{\mathbf{q}} = [1, 0, 0, 0]^T$ and $\omega = [0, 0, 0]^T$ yields

$$\begin{bmatrix} \mathbf{q}_{k+1} \\ \mathbf{b}_{k+1} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & -\frac{\Delta t}{2}\mathbf{I} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{q}_k \\ \mathbf{b}_k \end{bmatrix} + \begin{bmatrix} \frac{\Delta t}{2}\mathbf{I} \\ \mathbf{0} \end{bmatrix} \mathbf{w}_{rk} + \begin{bmatrix} -\frac{\Delta t}{2}\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{w}_{w_r k} \\ \mathbf{w}_{bk} \end{bmatrix}. \tag{3.45}$$

The Kalman gains $\mathbf{K}_1$ and $\mathbf{K}_2$ for this LTI system can be obtained solving the *Algebraic Ricatti Equation*, resulting in the observer

$$\begin{bmatrix} \hat{\boldsymbol{q}}_{k+1} \\ \hat{\boldsymbol{b}}_{k+1} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & -\frac{\Delta t}{2}\mathbf{I} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{q}}_k \\ \hat{\boldsymbol{b}}_k \end{bmatrix} + \begin{bmatrix} \frac{\Delta t}{2}\mathbf{I} \\ \mathbf{0} \end{bmatrix} \mathbf{w}_{rk} + \begin{bmatrix} \mathbf{K}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_2 \end{bmatrix} (\mathbf{q}_k - \hat{\mathbf{q}}_k). \tag{3.46}$$

Chapter 4

Prototype

The hardware involved in this thesis can be divided into two main groups. On the one side, the functional satellite prototype. On the other side, the testbed and support components to attach the prototype to the testbed. The satellite prototype is a 1U CubeSat, meaning that its dimensions are $100 \times 100 \times 113.5mm$ and its maximum weight is 1.33Kg [9].

The testbed was designed so as to allow the testing of the Attitude Determination and Control System (ADCS) of any 1U CubeSat. In this chapter the physics of the testbed are presented.

The satellite prototype is not intended to go to space. It is a 1U satellite functional prototype, with actuators, sensors and a microcontroller with an educational purpose. Since it aims to be a low-cost solution, all of its components are Commercial Off-The Shelf (COTS) components, except for the outer structure and other components designed to fit the COTS components to the outer structure, which were 3D printed. These parts were designed using *SolidWorks* and printed in PLA using the *Ultimaker 2* 3D printer. *Ultimaker* has its own *slicing* software: *Ultimaker Cura*. *Cura* is used to setup the part and printing properties. 3D printing keeps production costs low and allows for rapid construction of different designs, allowing for an iterative design process. The main disadvantage of 3D printing is that it is hard to have exceptionally well polished surfaces.

Figure 4.1: Picture of the Ultimaker 2 3D printer. From [54].

4.1 Satellite Prototype

The *Raspberry Pi 4 Model B* [55] is the microcontroller selected to be responsible for running the control and attitude determination algorithms, communicating with the sensors and driving the actuators. The *Raspberry Pi* is connected to a *PiJuice Hat* [56], a fully uninterruptible power supply, an LSM9DS1 IMU [57], 3 hall-effect sensors [58] and 2 DRV8835 dual motor drivers [59]. Each motor driver is powered by a 6V power source (set of 4 1.5V AA cells) and drives the momentum wheel according to the PWM value and signal indicated by the Raspberry Pi.

The Raspberry Pi has 29 GPIO (general-purpose input/output) pins. GPIO pins can be used with a variety of alternative functions, some are available on all pins, others on specific pins. These functions are Pulse-Width Modulation (PWM), Serial Peripheral Interface (SPI), Inter-Integrated Circuit (I2C) and Serial. Table 4.1 has a list of the electronic components with its mass, quantity and GPIO's used.

Component	Unit Mass (g)	Quantity	Total Mass (g)	GPIO and Voltage Pins used
Raspberry Pi 4 Model B	62	1	62	-
PiJuice	60	1	60	-
Motor Driver	22	2	44	4, 5, 6, 12, 19, 25, 3V3, GND
Battery	90	2	180	-
Motor	17	3	51	-
IMU	26	1	26	0, 1, 2, 3, 3V3, 5V, GND
Wheel encoder	4	3	12	10, 14, 15, 3V3

Table 4.1: List of electronic components.

The non-electronic components - beside nuts and screws - include the wheels, the supports from the wheels to the CubeSat, the supports from the motors to the wheels, the CubeSat faces, and the corners that connect the faces. The CAD's of the designed parts can be found in [60]. A list of the non-electronic components can be found on Tab. 4.5.

Component	Unit Mass (g)	Quantity	Total Mass (g)
Motor Face 1	29.04	1	29.04
Motor Face 2	24.01	1	24.01
Battery Face 1	34.74	1	34.74
Battery Face 2	30.19	1	30.19
Bottom Face	35.9	1	35.9
Top Face	23.93	1	23.93
Wheel Suport 1	9.19	2	18.38
Wheel Suport 2	10.19	1	10.19
Motor to Wheel Connector	0.23	3	0.69
Corner	1.54	8	12.32
Wheel	25.21	3	75.63

Table 4.2: List of non electronic components.

An exploded view of the designed prototype is found on Fig. 4.2

The total mass of the cube, including wires, nuts, screws and other added weights is 797 grams. The moments of inertia of the cube in the body frame was obtained from *SolidWorks* as

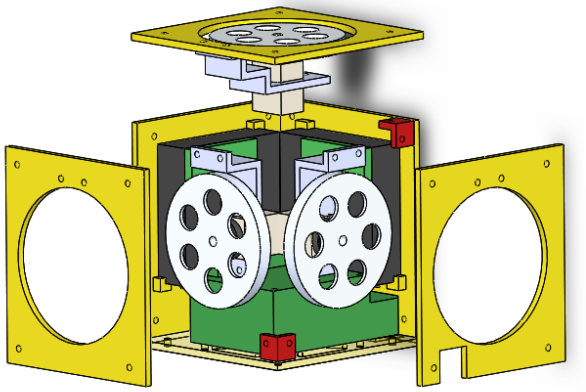


Figure 4.2: Exploded view of the CubeSat prototype and its components.

$$\mathbf{J} = \begin{bmatrix} 1.12 & 0.0298 & 0.0907 \\ 0.0298 & 1.16 & -0.0359 \\ 0.0907 & -0.0359 & 1.03 \end{bmatrix} \times 10^{-3} Kg.m^2. \tag{4.1}$$

4.1.1 Actuators

As actuators, the satellite will be provided with 3 momentum wheels aligned with the 3 body axis, the minimum amount to control the attitude in space [38]. Reaction wheels and momentum wheels are very similar. They both have flywheels and are driven by electric motors. However, unlike a reaction wheel, a momentum wheel is always spinning at a very high speed, which creates a momentum bias, making it resistant to changing its attitude. The momentum wheel can be accelerated and decelerated near the momentum biased high speed (instead of the zero speed), so as to create a torque and force the spacecraft to rotate, allowing for attitude control.

If there were N wheels not aligned with the body axis, then, for each wheel, its angular momentum expressed in the spacecraft body axes can be expressed as $h_n = \hat{\mathbf{w}}_n^w h_n = \hat{\mathbf{w}}_n^w I_n w_n$, where I_n is the inertia of the n-th wheel along its rotation axis, $\hat{\mathbf{w}}_n^w$ is the spin axis of the n-th wheel resolved in the body frame and w_n is its spin rate. The distribution matrix is defined as

$$\mathbf{W}^w = \begin{bmatrix} \hat{\mathbf{w}}_1^w & \hat{\mathbf{w}}_2^w & ... & \hat{\mathbf{w}}_N^w \end{bmatrix}. \tag{4.2}$$

The angular momentum generated by the momentum wheel's system with respect to the center of mass of the system and expressed in the body frame is given by

$$\mathbf{h}_w = \mathbf{W}^w [h_1 \quad h_2 \quad ... \quad h_N]^T, \tag{4.3}$$

$$h_n = I_n(w_n + (\hat{\mathbf{w}}_n^w)^T \mathbf{w}_s) \quad , \quad n = 1, 2, ..., N. \tag{4.4}$$

The control torque of the momentum wheels' system follows the relation

$$\mathbf{u} = -\dot{\mathbf{h}}_w = -\mathbf{W}^w[\dot{h}_1 \quad \dot{h}_2 \quad ... \quad \dot{h}_N]^T, \tag{4.5}$$

$$\dot{h}_n = I_n(\dot{w}_n + (\hat{\mathbf{w}}_n^w)^T\dot{\mathbf{w}}_s) \quad , \quad n = 1, 2, ..., N. \tag{4.6}$$

The term $\dot{\mathbf{w}}_s$ is generally small and does not significantly affect the evolution of the wheels' speeds. As a result, it is frequently omitted [61]:

$$\mathbf{u} = -\dot{\mathbf{h}}_w \approx -\mathbf{W}^w[I_1 w_1 \quad I_2 w_2 \quad ... \quad I_N \dot{w}_N]^T. \tag{4.7}$$

In the present case, the three wheels are aligned with the body axis, meaning that $\mathbf{W}^w = \mathbf{I}_3$.

Momentum Wheel Model

In this model, a flywheel attached to a DC brushed motor in a given axis is considered. When modeling a DC motor, there are some constants that define the behaviour of the model, namely the torque coefficient of the motor, the electrical resistance of the motor armature and the back EMF (electromotive force) constant denoted by K_M, R_M and K_v, respectively.

The current at the motor is given by

$$i_M = \frac{1}{R_M}(v - K_v.w_{REL}), \tag{4.8}$$

where v is the input voltage to the electrical motor and w_{REL} is the angular velocity of the rotating part of the motor relative to the stator. Denoting the angular velocity of the stator (or satellite) by w_S:

$$w_{REL} = w_w - w_S. \tag{4.9}$$

The torque produced by the electrical motor is

$$\dot{h}_w = K_M.i_M - B.w_{REL} - f(w_{REL}), \tag{4.10}$$

where B is the viscosity damping coefficient sensed by the rotor. As denoted by equation 4.10, the damping torque is proportional to the angular velocity of the rotor relative to the satellite body frame. The function *f(.)* represents the coulomb and dry-friction which will be ignored in the subsequent analysis so as to allow for a linear transfer function.

The wheel angular velocity is related to the wheel torque by

$$\dot{w}_w = \frac{\dot{h}_w}{I_w}, \tag{4.11}$$

where I_w denotes the wheel moment of inertia.

According to the Euler's equation of motion, in order to apply a torque on the body, a torque in the opposite direction must be produced by the rotor of the DC motor:

$$\dot{h}_S + \dot{h}_w = 0, \tag{4.12}$$

where $\dot{h}_S$ denotes the satellite's torque, which is related to the satellite angular velocity by

$$w_S = \frac{\dot{h}_S}{I_S}, \tag{4.13}$$

where I_S denotes the satellite's moment of inertia along the axis where the momentum exchange device is mounted.

The momentum wheel model can be represented by the scheme in Fig. 4.3.

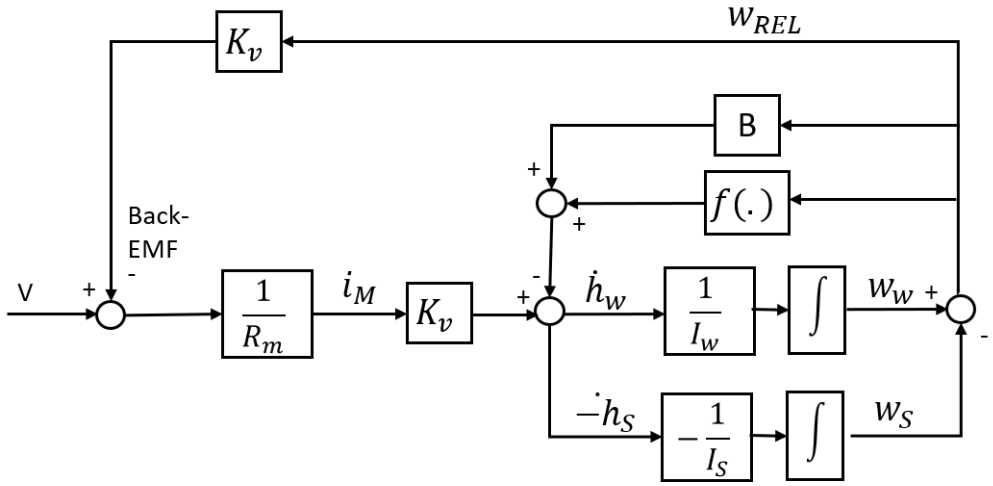


Figure 4.3: Momentum Wheel Model. Adapted from [38].

Once established the differential equations that rule the momentum wheel functioning, the goal is to find a transfer function from the motor input voltage, v, to the satellite torque, $\dot{h}_S$.

By neglecting the Coulomb and dry friction, equation 4.10 simplifies to

$$\dot{h}_w = K_M.i_M - B.w_{REL}.$$

Replacing equation 4.8 in the above expression and rearranging the terms:

$$\begin{aligned}
\dot{h}_w &= \frac{k_M}{R_M}(v - K_v.w_{REL}) - Bw_{REL} \\
&= \frac{K_M}{R_M}.v - (\frac{K_M K_v}{R_M} + B)w_{REL} \\
&= \frac{K_M}{R_M}.v - (\frac{K_M K_v}{R_M} + B)(w_w - w_S).
\end{aligned}$$

Applying the Laplace transform to equations 4.11 and 4.13 and replacing in the above expression one obtains:

$$
\begin{aligned}
\dot{h}_w &= \frac{K_M}{R_M}.v - (\frac{K_M K_v}{R_M} + B)(\frac{\dot{h}_w}{I_w s} - \frac{\dot{h}_S}{I_S s}) \\
&= \frac{K_M}{R_M}.v - (\frac{K_M K_v}{R_M} + B)(\frac{\dot{h}_w}{I_w s} + \frac{\dot{h}_w}{I_S s}) \\
&= \frac{K_M}{R_M}.v - (\frac{K_M K_v}{R_M} + B)(\frac{1}{I_w} + \frac{1}{I_S})\frac{\dot{h}_w}{s}.
\end{aligned}
$$

At last, the transfer function from the motor input voltage to the wheel torque is

$$
\frac{\dot{h}_w}{v} = \frac{I_w w_w s}{v} = \frac{\frac{K_M}{R_M} s}{s + (\frac{K_M k_v}{R_M} + B)(\frac{1}{I_w} + \frac{1}{I_S})}. \tag{4.14}
$$

By analyzing equation 4.15, it is obvious that a step in the motor voltage does not produce a pure angular torque, because the system will have a time constant and has a derivative relating the input to the output. The transfer function can be further simplified, if one assumes that the viscosity damping coefficient is close to zero and the wheel moment of inertia is much smaller than the satellite's:

$$
\frac{\dot{h_w}}{v} = \frac{\frac{I_w}{K_v} s}{s\frac{R_M I_w}{K_M K_v} + 1}. \tag{4.15}
$$

Momentum Wheel Design

To achieve a given desired performance, a momentum wheel system must have certain technical features, such as the maximum achievable torque and momentum storage capacity, low torque noise and Coulomb friction torques [38].

The sizing of the momentum wheels was based on a simplified approach that considers time-optimal control. Time-optimal control is achieved by delivering maximum angular accelerations and decelerations to the flywheel.

As discussed in the previous subsection and depicted by equation 4.12 since there are no external moments, the spacecraft's variation in angular momentum is symmetric to the momentum wheels'.

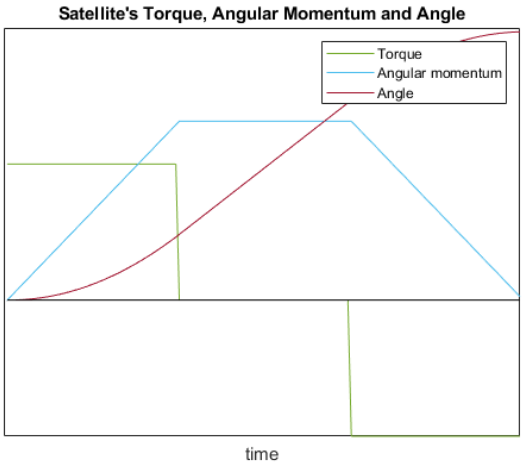


Figure 4.4: Time-optimal Control of a reaction wheel.

To obtain a positive rotation of the satellite, a negative momentum must be delivered ($\dot{h}_w = -\dot{h}_S$) between a time frame $[t_1, t_2]$. The angular momentum of the satellite will increase by

$$h_S(t_2) = -\dot{h}_w(t_2 - t_1). \tag{4.16}$$

The satellite will rotate to

$$\theta(t_2) = -\frac{\dot{h}_w}{I_S}\frac{(t_2 - t_1)^2}{2}. \tag{4.17}$$

In a time-optimal strategy, there is no interval between the acceleration and deceleration of the momentum wheel and the final angular rotation is simply

$$\theta_f = 2\theta(t_2) = -\frac{\dot{h}_w(t_2 - t_1)^2}{I_S}. \tag{4.18}$$

By denoting the time in which one wants to achieve the final rotation by $t_f = 2(t_2 - t_1)$, the above formula can be used to compute the torque one needs to achieve in this maneuver:

$$\dot{h}_w = \frac{4I_S\theta_f}{t_f^2}. \tag{4.19}$$

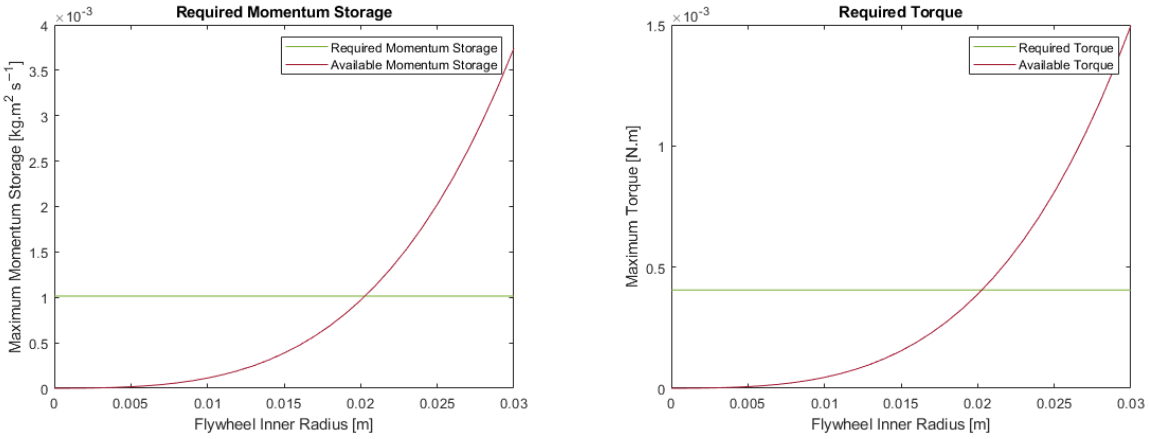
During acceleration, the wheel angular momentum increases to

$$h_w = \frac{t_f}{2}\dot{h}_w = \frac{2I_S\theta_f}{t_f}, \tag{4.20}$$

which is the amount of angular momentum the inertia wheel must be able to store.

The above formulas can be used to size the momentum wheel. As a design requirement, it is assumed that one wants to be able to rotate the satellite by 1 rad in 5 seconds. Assuming that a motor can put the flywheel into rotation up to 500 rad/s in a linear manner as depicted in Fig. 4.4. Assuming also the inertia for a 1U satellite with mass 1.33Kg and designing the wheel for the axis with the biggest inertia. At last, assume that the inertia wheel is an aluminum wheel with 1mm thickness except for an outer rim that extends the wheel by 2mm and has 4mm of thickness.

The torque and momentum available can be compared to the torque and momentum required so as to size the inner radius of the momentum wheel. The design point is depicted in Fig. 4.5. This analysis yields a minimum inner radius of 2.1 cm.



(a) Input (b) Angular Rate

Figure 4.5: Design Point for the Momentum Wheel.

The flywheel was not manufactured in-house due to time and resource constraints. As a result, its inner radius is not 2.1cm but 3cm. The real flywheel was modeled in detail in SolidWorks (see Fig. 4.6), so as to find its moment of inertia. The real momentum wheel moment of inertia is $1.658552 \times 10^{-5} Kg.m^2$, which complies with the referred maneuver requirement.

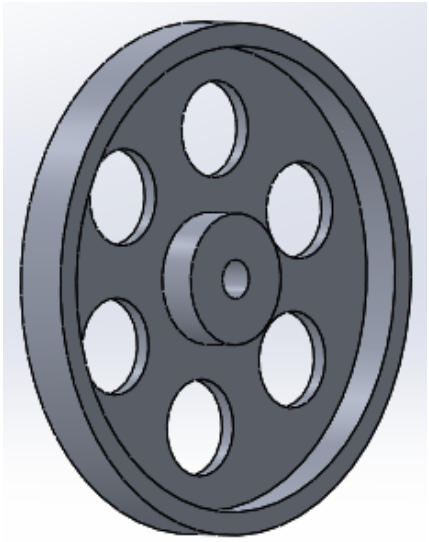


Figure 4.6: SolidWorks capture of the 3D model of the flywheel.

From [41], a typical value for the maximum torque to be provided by the Reaction Wheel System is $4.24 \times 10^{-6} N.m$ and typical values for the external torques acting on a 1U CubeSat are: $4.3 \times 10^{-8} N.m$ for the gravitational torques; $2.9 \times 10^{-7} N.m$ for the atmosphere-induced torques; $8.2 \times 10^{-9} N.m$ for the solar radiation torques; $2.0 \times 10^{-6} N.m$ for the magnetic field-induced torques. Assuming a worst case scenario, in which all the perturbations act on the same direction in which one wants to provide the maximum torque, the maximum required torque is $\dot{h}_{w_{max}} = 6.64 \times 10^{-6} N.m$. In the same work, the maximum angular momentum storage capacity comes from integrating the maximum required torque over a maneuver of 400s, yielding $h_{w_{max}} = 1.09 \times 10^{-3} N.m.s$. Following a similar analysis as before, but

using these new constraints, yields an inner radius of at least 2.7cm. Therefore, the previously chosen momentum wheel is still a good fit. The designed momentum wheel is to be used in the ground testing facility's functional satellite prototype. However, in a real space application, the flywheel could be put into rotation up to 2000 rad/s [41], which would allow for a much smaller momentum wheel system.

Pulse-Width Modulation (PWM)

The DC motors of the inertia wheel will be controlled resorting to a PWM signal (instead of a continuous signal). A PWM signal is characterized by having only two possible values: on and off. This type of modulation uses a rectangular pulse wave whose pulse width is modulated resulting in a variation of the average value of the waveform [62].

Denoting the waveform by x(t), the period by T, the on value by x_{max}, the off value by x_{min} and the duty cycle by D, the average value of the waveform is given by

$$\bar{x} = \frac{1}{T}\int_0^T x(t)dt = D.x_{max} + (1-D).x_{min}. \tag{4.21}$$

In the present case, $x_{min} = 0$, meaning that $\bar{x} = D.x_{max}$.

The proportion of on-time to the period is the duty cycle. A duty cycle of 0% corresponds to a 0 constant signal and a duty cycle of 100% corresponds to a constant signal equal to maximum value.

The use of pulse width modulation to control a small motor results in lower power loss in the switching transistor because it is either fully on or fully off. Class D power amplifiers are used to implement PWM signals and power efficiencies well over 90% are common. PWM is particularly suited for running inertial loads such as motors, as they are not as easily affected by the discrete switching, since their inertia causes them to react slowly. The PWM switching frequency has to be high enough not to affect the load.

4.1.2 Sensors

Inertial Measurement Unit

An Inertial Measurement Unit (IMU) is an electrical unit generally consisting of a 3-axis accelerometer, a 3-axis gyroscope and, in some cases, a 3-axis magnetometer. The selected IMU was the 9-axis LSM9DS1 inertial module by STMicroelectronics which communicates with the Raspberry Pi via *I2C* which is a synchronous, multi-master, multi-slave, packet switched, single-ended, serial communication bus invented in 1982 by Philips Semiconductor.

The basic performance characteristics of the LSM9DS1 are presented in Tab. 4.3. The measurement range of the accelerometer is expressed in units of g, the gravitational acceleration on Earth. The gyroscope measurement range is expressed in degrees per second, dps. The sensitivity of the accelerometer is 0.732 mg/LSB, meaning that the output is expressed in units of 0.000732 g. A similar interpretation can be done for the sensitivity of the gyroscope and magnetometer.

The 3-axis prefix indicates that the acceleration, angular velocity and magnetic field are measured in the 3-dimensional space.

To characterize the noise affecting the IMU, sample data corresponding to 1.5 hours was collected. Since the platform was at rest, the measurements from the gyroscope correspond to noise. Regarding

	Accelerometer	Gyroscope	Magnetometer
Measurement Range	$\pm 2g - \pm 16g$	$\pm 245dps - \pm 2000dps$	$\pm 4Gauss - \pm 16Gauss$
Output Resolution	16 bits	16 bits	16 bits
Sensitivity	0.732 mg/LSB($\pm 16g$)	70 mDPS/LSB($\pm 2000dps$)	0.58 mGauss/LSB($\pm 16Gauss$)
Output Data Rate (max)	952 Hz	952 Hz	80 Hz

Table 4.3: Basic performance characteristics of the LSM9DS1 IMU.

the accelerometer, only the gravity acceleration is being read, and it is aligned with the z-axis. Subtracting the gravity component, the measurement from the accelerometer also corresponds to noise. This way, the average of the measurements from the acelerometer and gyroscope can be used to retrieve the average bias. The noise covariance and bias are resumed in Tab. 4.4.

	Accelerometer	Gyroscope
Noise Bias	$[-0.11475, 0.10982, -0.044702]m/s^2$	$[0.1096, -0.11303, -0.20123]o/s$
Noise Variance	$[0.0029346; 0.0052722; 0.0013492](m/s^2)^2$	$[7.1815e-05, 0.0001305, 0.0001102](o/s)^2$

Table 4.4: Noise bias an covariance identification for the LSM9DS1 IMU.

When it comes to the magnetometer, identifying its measurement model is not trivial and a specific calibration procedure is described in subsection 5.3.1. The values to be used in the simulation environment regarding the noise variance were obtained similarly as for the gyroscope and accelerometer: $[0.15055, 0.15971, 0.37153](\mu T)^2$.

Hall-Effect Sensor

A neodymium 8-pole magnets with rubber hubs is connected to each momentum wheel axis, thus rotating at the same angular speed as the momentum wheels themselves. An hall-effect sensor is put close to the magnets, so that it can sense the change in magnetic polarization as the magnet rotates.

4.2 Test Bench for Nanosatellite Ground Testing

The test bench provides an inexpensive platform for ground-based testing while imitating the torque-free conditions of a satellite in space. This simulator is composed of 2 main parts: a rotor and a stator. The stator is composed of a small chamber and a plate with holes. The rotor is a hollow sphere made of plastic that sits on top of the rotor.

A 3D computer model of the satellite prototype sitting on the rotor is depicted in Fig. 4.7. The rotor was bought off-the-shelf. The 2 parts of the stator were manufactured by a CNC machining company with a precision of 10 nm. The supports for the 1U CubeSat were 3D printed. The drawings can be found on [60]. A list of the items can be found on Tab. 4.5.

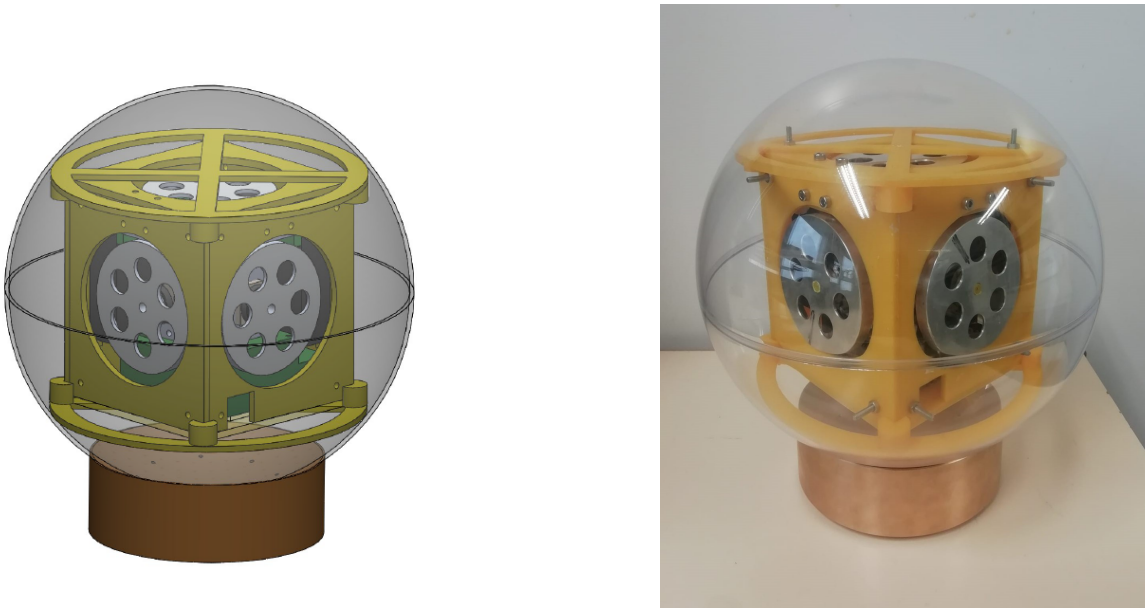
An air compressor is connected to the small chamber, forcing air through the holes of the plate. This keeps the rotor suspended on a cushion of air and allows the simulator to rotate 360º in roll, pitch and yaw. The main advantage of this setup is the low friction between the plate and the sphere.

This setup reduces the friction between moving parts, allowing relative motion along the desired directions. The load is supported by a thin layer of rapidly moving pressurized air between the surfaces.

Component	Unit Mass (g)	Quantity	Total Mass (g)
Rotor Half Sphere	152	2	304
Stator Bottom Part	369	1	369
Stator Upper Part	1394	1	1394
CubeSat Support	33	2	66

Table 4.5: List of components that compose the Test Bench.

As a result, this can also be called an air bearing.



(a) 3D Model (b) Picture

Figure 4.7: 3D model (Left) and Picture (Right) of the Satellite Prototype Sitting on the Developed Test Bench.

As it is evident in Fig. 4.7, the CubeSat Prototypes does not rotate alone. The rotating parts also include the rotor and the supports. From *SolidWorks*, the moment of inertia of the Cubesat with the supports and the rotor in the body frame is

$$\mathbf{J}_{Total} = \mathbf{J}_S + \mathbf{J}_{rotor} + \mathbf{J}_{supports} = \begin{bmatrix} 3.58 & -0.0297 & 0.0298 \\ -0.0297 & 3.54 & 0.0122 \\ 0.0298 & 0.0122 & 3.35 \end{bmatrix} \times 10^{-3} Kg.m^2. \tag{4.22}$$

The added moments of inertia act as parasitic moments of inertia. The moment exchange of the satellite under tests is less effective. The factor that describes this loss of effectiveness can de computed by $k = J_S/J_{Total}$.

Furthermore, a slight miscenter between the center of rotation and center of mass of the prototype induces a pendulum-like dynamics to the system. Replacing $\mathbf{h}_w = \mathbf{u} = \mathbf{0}$ and adding a term to describe this dynamics, equation 2.22 becomes

$$\mathbf{J}\dot{\mathbf{w}}_I = -\mathbf{J}\mathbf{w}_I \times \mathbf{w}_I + mg\mathbf{r}_{cm} \times \mathbf{A}_I^b \mathbf{e}_z \tag{4.23}$$

where $\mathbf{r}_{cm}$ is the vector from the origin of the body frame to the center of mass, m is the mass of the rotating parts, g is the gravity acceleration, $\mathbf{A}_I^b$ is the rotation matrix from the inertial to the body frame and $\mathbf{e}_z = [0, 0, 1]^T$ is a vector representing the gravity acceleration direction in the inertial frame. This miscenter shall be compensated by means of a manual or automated balancing procedure. In this work, the miscenter was partially compensated adding lead weights until $\mathbf{r}_{cm} \approx \mathbf{0}$.

4.2.1 Physics of the Test Bench

The physics of the Test Bench are rather complex even for the case of a single hole in the center of the stator's plate. Even though the present chapter focuses on the case of a single hole in the center of the stator, experience showed that adding holes around the center hole in a circular pattern results in a better performance.

The levitation force produced by the air flow, F_{up}, must be enough to compensate the gravity force, F_g, due to the rotor's weight, m_r. Denoting the gravitational field strength by $g = 9.81m/s^2$, the gravity force is given by:

$$F_g = m_r.g. \tag{4.24}$$

The levitation force is provided by the gauge pressure $P(\theta) - P_{Atm}$ inside the fuid layer. The associated force follows from

$$F_{up} = \int\int [P(\theta) - P_{Atm}] \cos(\theta) dA, \tag{4.25}$$

where dA represents the submerged surface.

In order to proceed with the characterization of the test bench, one needs to know the gauge pressure distribution inside the fluid layer.

Along with the pressure, the shear stress and the buoyant force contribute to the force on the cylinder. However, according to [63], their contribution is much smaller than the pressure.

The pressure field follows from the mass and momentum balance within the fluid.

The mass balance is represented by the continuity equation:

$$\frac{\partial \rho}{\partial t} + \nabla.(\rho \mathbf{u}) = 0, \tag{4.26}$$

where ρ is the density of the air and **u** is the fluid velocity distribution.

The Mach number is defined as $Ma = u/c$, where u is the velocity of the flow and c the speed of sound in air. Since c = 330 m/s, the Ma of the fluid film flow is much smaller than 1 and, as a result, the compressibility of the air is negligible. For an incompressible fluid ($\rho = constant$), the mass balance simplifies to $\nabla \mathbf{u} = 0$.

In spherical coordinates, the mass balance yelds

$$\frac{1}{r^2}\frac{\partial(\rho r^2 u_r)}{\partial r} + \frac{1}{rsin(\theta)}\frac{\partial(\rho u_\theta sin(\theta))}{\partial \theta} + \frac{1}{rsin(\theta)}\frac{1\partial(\rho u_\phi)}{\partial \phi} = 0. \tag{4.27}$$

Consider the gap coordinates (x,y, ϕ) parallel and perpendicular to the curved surface and related to the spherical coordinates (r, θ, ϕ) as $x = R\theta$, $y = R + h - r$, where h is the width of the gap between the rotor and the stator, as depicted in Fig. 4.8.

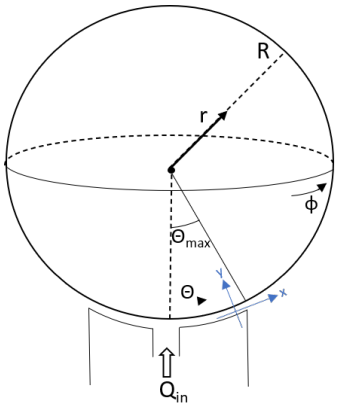


Figure 4.8: Sphere coordinates (r, θ, ϕ) and auxiliary gap coordinates (x, y, ϕ).

Assuming that there are no imperfections in the surfaces, the velocity in the ϕ direction is zero. Also, after leaving the nozzle, the flow rapidly orients itself in the θ-direction, so the velocity in the r-direction will also be zero. Thus in steady-state $\mathbf{u} = u_\theta(r, \theta)\mathbf{e}_\theta$ or $\mathbf{u} = u(y, \theta)\mathbf{e}_\theta$ The continuity equation then reduces to

$$\frac{\partial u_\theta}{\partial \theta} sin(\theta) + u_\theta cos(\theta) = 0. \tag{4.28}$$

Creeping Flow Approximation

The momentum balance is expressed by the Navier-Stokes equation, which for fluids with constant density ρ and viscosity μ is given by

$$\rho \left[\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}.\nabla)\mathbf{u} \right] = \rho \mathbf{g} - \nabla P + \mu \nabla^2 \mathbf{u}. \tag{4.29}$$

The flow is radially outward from the inlet nozzle so that $\mathbf{u} = (u_r, u_\theta, u_\phi) = u_\theta(r, \theta)\mathbf{e}_\theta$. As the fluid is being spread over a region of increasing area its speed must decrease with θ to ensure conservation of mass.

This means that the term $\rho(\mathbf{u}.\nabla\mathbf{u})$ is not identically zero. However, for simplicity, one can neglect it in the Navier-Stokes equation, since it is small in comparison with the terms ∇P and $\mu\nabla^2\mathbf{u}$ as long as the inflow rate is sufficiently small [63].

Considering steady flow, the first term on the left side vanishes. The gravitational effect is also negligible, considering that its only adds a hydrostatic component to the pressure. With $\rho = 1.225 Kg/m^3$ and with a film thickness of $h = 0.15mm$, the gravitational contribution to the fluid pressure is at most $\rho g h \approx 0.002 Pa = 1 \times 10^{-8} atm$, much smaller than the required gauge pressure at the inlet nozzle, which is in the order of a tenth of an atmosphere.

The Navier-Stokes equation reduces to

$$0 = -\nabla P + \mu \nabla^2 \mathbf{u}, \tag{4.30}$$

which is known as "Stokes flow" or "creeping flow".

As the y and ϕ components of the velocity profile are zero, equation 4.30 implies that the pressure is a function of x only. Since the fluid speed does not depend on ϕ, from equation 4.30 follows

$$\frac{dP}{dx} = \mu \frac{d^2 u_x}{dx^2} + \mu \frac{d^2 u_x}{dy^2}. \tag{4.31}$$

Or, in spherical coordinates

$$\frac{1}{r}\frac{\partial P}{\partial \theta} = \mu \left[\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2 \frac{\partial u_\theta}{\partial r} \right) + \frac{1}{r^2}\frac{\partial}{\partial \theta}\left(\frac{1}{sin(\theta)}\frac{\partial}{\partial \theta}(u_\theta sin(\theta)) \right) \right]. \tag{4.32}$$

This equation can be solved inserting an auxiliary function $K(\theta)$:

$$\frac{dP}{dx} = \frac{1}{R}\frac{dP}{d\theta} = -\mu K(\theta). \tag{4.33}$$

Replacing equation 4.33 in 4.32, a parabolic velocity profile across the fluid layer is obtained:

$$u_\theta(y, \theta) = \frac{1}{2} K(\theta) y (h - y). \tag{4.34}$$

At an angle θ, the total flux per unit of time through the cross-section with circumference $2\pi R sin(\theta)$ is

$$Q = 2\pi R sin(\theta) \int_0^h u_\theta(y, \theta) dy. \tag{4.35}$$

Replacing equation 4.34 in 4.35 yields

$$Q = \frac{1}{6} \pi R sin(\theta) K(\theta) h^3. \tag{4.36}$$

Thus, $K(\theta)$ decreases with $1/sin(\theta)$:

$$K(\theta) = \frac{6Q}{\pi R h^3 sin(\theta)}. \tag{4.37}$$

Replacing the value of $K(\theta)$ from 4.37 in the equation for the pressure gradient 4.33 and noting that $P(\theta_{max}) = P_{atm}$, the gauge pressure can be computed

$$P(\theta) - P_{atm} = \frac{6\mu Q}{\pi h^3} ln \left[\frac{(1 - cos(\theta_{max}))sin(\theta)}{(1 - cos(\theta))sin(\theta_{max})} \right]. \tag{4.38}$$

By replacing equation 4.38 in 4.25, the levitation force is obtained:

$$F_{up} = \frac{6\mu Q R^2}{h^3}(1 - cos(\theta_{max})). \tag{4.39}$$

Now, by equating the levitation force to the weight of the rotor 4.24, the thickness of the fluid layer is obtained:

$$h = \left(\frac{6\mu Q R^2}{m_r g}\right)^{1/3}\left(1 - cos(\theta_{max})\right)^{1/3}. \tag{4.40}$$

The values for the developed test bench are: $\mu = 1.81 \times 10^{-5} Kg/(m.s)$, $R = 0.1m$, $g = 9.81 m/s^2$, $m_r = 1.33Kg$, $Q = 0.00023 m^3/s$, $\theta_{max} = 0.6rad$, $\rho = 1.225 Kg/m^3$, orifice radius of 1mm. These values can be used to compute the thickness of the fluid layer, obtaining $h = 0.15mm$.

Inertial Effects: Bernoulli Suction

The inertia of the fluid is accounted in the Navier-Stokes equation 4.29 by the advection term $\rho(\mathbf{u}.\nabla)\mathbf{u}$, which was neglected in the preceding analysis. However, the fact that the velocity decreases as $\frac{1}{sin(\theta)}$ means that the inertia of the fluid can become important if the contribution of the fluid deceleration to the pressure field is not negligible.

The dimensional analysis method will be employed to verify if $\rho u \nabla u << \mu \nabla^2 u$, i.e. if the viscous effects dominate the inertial. The order of magnitude estimates are $\frac{\partial u}{\partial x} \sim U/R$, $\frac{\partial u}{\partial y} \sim U/h$, $\frac{\partial^2 u}{\partial x^2} \sim U/R^2$, $\frac{\partial^2 u}{\partial y^2} \sim U/h^2$, where U is a typical value for the fluid velocity. Then, the condition $\rho u \nabla u << \mu \nabla^2 u$ can be written as

$$\rho u_x \frac{\partial u_x}{\partial x} << \mu \left(\frac{\partial^2 u_x}{\partial x^2} + \frac{\partial^2 u_x}{\partial y^2}\right) \tag{4.41}$$

$$\rho U \left(\frac{U}{R}\right) << \mu \left(\frac{U}{R^2} + \frac{U}{h^2}\right), \tag{4.42}$$

or, equivalently,

$$Re << 1 + \frac{R^2}{h^2}, \tag{4.43}$$

where $Re = \rho U R/\mu$ is the Reynolds number based on the sphere radius R.

Replacig the values for the present case (consider $Q = 0.00023 m^3/s$ and an orifice with 1mm radius), one obtains $Re = 5 \times 10^5$, and $1 + \frac{R^2}{h^2} = 4.44 \times 10^5$, meaning that inertial effects cannot be neglected.

The idealized case of inviscid flow ilustrates the effect of inertia in its purest form. This is the opposite case of creeping Stokes flow. One now assumes that the inertia (or kinetic energy) dominates viscous friction. If in addition the flow is steady and irrotational, the Navier-Stokes equation can be integrated to obtain the Bernoulli's law:

$$\frac{1}{2}\rho|\mathbf{u}|^2 + P = constant, \tag{4.44}$$

where the gravity was neglected. This equation expresses the conservation of energy for an inviscid flow in which regions of high kinetic energy correspond to low pressure. For the present case, the velocity decreases from the nozzle to the outlet region, meaning that if it were for the inertial effects alone, the pressure would be below atmospheric in the submerged area, creating a downward force.

A description that includes both viscous and inertial effects is addressed in [64], where the following approximate expression is derived for the pressure field in the fluid layer:

$$P(r) - P_{atm} = \frac{6\mu Q}{\pi h^3} ln\left(\frac{r_{max}}{r}\right) - \frac{27\rho Q^2}{140\pi^2 h^2}\left(\frac{1}{r^2} - \frac{1}{r_{max}^2}\right), \tag{4.45}$$

where r denotes the distance from the inlet nozzle.

The first term is very similar to the previously derived exact expression for the limiting case of Stokes flow 4.38. So, it is recognizable that the second is the pressure contribution from inertia.

Note that the viscous pressure grows with Q but the inertial contribution grows with Q^2. The dominance shifts from the viscous to the inertial regime as Q increases. At large flow rates, the generated lifting force will no longer be able to levitate the rotor.

In Fig. 4.9 the pressure distribution is plotted for the extreme case of creeping flow (solid curve) given by equation 4.38. The dashed curve includes both inertial and vicous effects (equation 4.45). The pressure is not plotted in the nozzle region (from $\theta = 0rad$ to $\theta = 0.05rad$), where the actual pressure will deviate from the previous analysis.

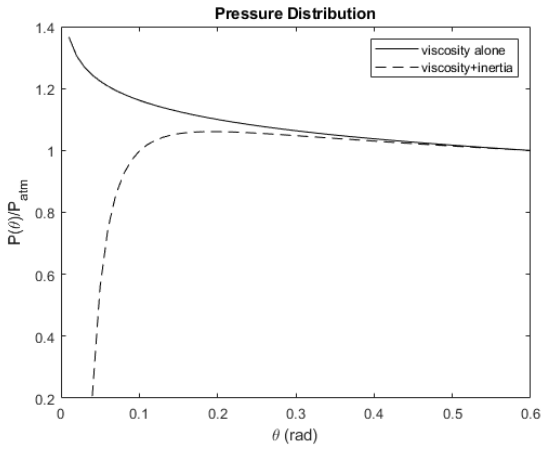


Figure 4.9: Pressure distribution $P(\theta)/P_{atm}$.

Damping of Rotations

Since there is no contact between the rotor and the stator, the friction is very low. In fact, the only source of drag is the viscous drag that the fluid exerts on the sphere and this is the exact same principle on which roller bearings work. This viscous drag causes the angular speed to slow down exponentially as

$$w(t) = w(0)e^{-t/t_{rel}}, \tag{4.46}$$

where t_{rel} is the relaxation time.

The computation of the relation time will be based on a scaling argument. First, it will be computed for a cylinder with width L and the same radius R as the sphere.

When the cylinder is set into rotation with angular frequency w, the velocity at the cylinder surface

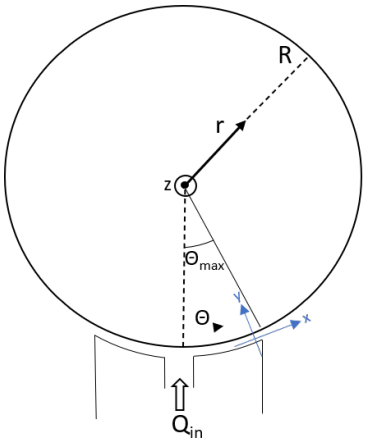


Figure 4.10: Auxiliary geometry: cylinder.

becomes $u = wR$ instead of u=0. This affects the velocity profile, that can now be seen as the superposition of the Poiseuille parabolic profile, $u_\theta(y)$ and a simple linear Couette profile, $u_w(y)$:

$$u(y) = u_\theta(y) + u_w(y), \tag{4.47}$$

where the Couette profile is given by

$$u_w(y) = wR\frac{y}{h}. \tag{4.48}$$

Note that $u_w(y)$ has zero second derivative and, as a result, does not contribute to the momentum balance in equation 4.31. Therefore, the pressure distribution is unaffected by the rotation.

The main contribution of the rotation is to break the left-right symmetry and hence it produces a nonzero friction torque. The force that the fluid applies in the cylinder (as a result of the force the cylinder applies in the fluid) is accomplished via the shear stress, τ:

$$\tau = \tau_\theta + \tau_w = \mu\left(\frac{du_\theta}{dy} + \frac{du_w}{dy}\right). \tag{4.49}$$

By symmetry, the Poiseuille τ_θ does not contribute to the net torque, which comes entirely from the Couette's contribution $\tau_w = \mu Rw/h$. The torque on the cylinder is computed by integrating the force moment $dT = -R\tau_w dA$ over the submerged area, yielding

$$T = -\frac{2\theta_{max}\mu LR^3}{h}w. \tag{4.50}$$

The torque will slow down the rotation according to the equation

$$T = I_{cyl}\frac{dw}{dt}, \tag{4.51}$$

where the moment of inertia of the cylinder, I_{cyl}, is given by

$$I_{cyl} = \frac{1}{2}MR^2 = \frac{1}{2}\rho_{cyl}\pi LR^4, \tag{4.52}$$

where ρ_{cyl} is the mean density of the cylinder.

Replacing 4.52 in 4.51 and the resulting expression in 4.50, yelds

$$\frac{dw}{dt} = -\frac{4\theta_{max}\mu}{\pi\rho_{cyl}Rh}w = -\frac{1}{t_{rel}}w. \tag{4.53}$$

The relaxation time for the auxiliary cylinder body is

$$t_{rel} = \frac{\pi}{4\theta_{max}}\frac{\rho_{cyl}Rh}{\mu}, \tag{4.54}$$

which can be viewed as a characteristic time scale for the damping $\frac{\rho_{cyl}Rh}{\mu}$ multiplied by a dimensionless geometric prefactor $\frac{\pi}{4\theta_{max}}$.

The sphere is much more complex than the cylinder since the axis of rotation is no longer fixed. The effectiveness of the viscous drag will be different if the sphere rotates arround the horizontal or vertical axis.

However, the essential physics are the same. In analogy to the torque on the cylinder, given by 4.50, the torque for the sphere must be, aside from a scaling prefactor that will depend on the mode of rotation:

$$T \sim \frac{\mu R^4 w}{h}, \tag{4.55}$$

where the width of the cylinder was replaced by R.

The relaxation time can now be approximately found:

$$T = I_{rotor}\frac{dw}{dt} \Rightarrow \frac{\mu R^4 w}{h} \sim MR^2\frac{dw}{dt} \Rightarrow \frac{\mu R^4 w}{h} \sim \rho_{rotor}R^5\frac{dw}{dt} \Rightarrow t_{rel} \sim \frac{\rho_{rotor}Rh}{\mu}. \tag{4.56}$$

Replacing the values for the developed test bench, one obtains $t_{rel} = 4.4$ minutes.

Orifice Sizing

The orifice sizing is done based on the analogy between pneumatic and electrical systems presented in [65]. In this text, the generalized system variables effort(e) and flow(f) are introduced. The energy coupling of many systems can be represented by this variables, whose product is the instantaneous power being transmitted through an energy port. The act of delivering energy is associated with one intensive variable (e.g current, flow rate) giving the flux of energy flow and an extensive variable (e.g voltage, pressure) giving the pitch of energy flow.

The general constitutive relation of an energy dissipator takes the form

$$e = \Psi(f). \tag{4.57}$$

In the linear form this becomes

$$e = R.f, \tag{4.58}$$

where R represents a resistance.

In an electrical system, the effort variable is the voltage, v, and the flow variable is the current, i. Electrical resistance is associated with electrical dissipation:

$$v = R.i, \tag{4.59}$$

The total power dissipated is given by

$$P = R.i^2 = \frac{v^2}{R}. \tag{4.60}$$

Consider the simple electric circuit composed by a voltage source, V, and a resistor, R. The voltage source has itself an inner resistance, r_i. Kirchoff's voltage law implies

$$V = (r_i + R)i \Leftrightarrow i = \frac{V}{r_i + R}. \tag{4.61}$$

The total power dissipated by the resistor, R is then

$$P = R.i^2 = \frac{R}{(r_i + R)^2}V^2. \tag{4.62}$$

If one wants to choose R so as to maximize the power dissipation, one must force $dP/dR = 0$:

$$\frac{dP}{dR} = 0 \Leftrightarrow r_i = R. \tag{4.63}$$

The value of R should be equal to the inner resistance of the voltage source. This is called impedance matching. The orifice of the stator shall be sized so as to maximize the energy transfer to the rotor.

In a pneumatic system, the effort variable is the pressure, P, and the flow variable is the flow rate, Q. Fluid dissipation may be the result of several effects, since there are many ways in which viscous effects can cause the conversion of fluid kinetic energy to thermal energy and dissipate power.

Regarding the viscous forces between the fluid and the retaining surfaces, the fluid resistance will be computed in a different manner whether in the presence of a porous medium or a capillary pipe where the flow will be laminar. In the case of a laminar flow (small numbers of Reynolds), the fluid resistance comes from the Haigen-Poiseuille's law:

$$R_f = \frac{P_{21}}{Q} = \frac{128\mu l}{\pi d^4}, \tag{4.64}$$

where P_{21} represents the pressure difference measured upstream and downstream of the power dissipator, μ the fluid viscosity, l the capillary pipe length and d the capillary pipe diameter.

Regarding the viscous forces between fluid particles, note that in the case of the incompressible flow through an orifice an approximate expression for the constitutive relation of an orifice dissipator is

$$P_{21} = \frac{\rho}{2C_d^2 A_0^2}Q|Q|, \tag{4.65}$$

where ρ is the fluid density, C_d is the orifice coefficient of discharge and A_0 is the orifice cross-sectional

area.

Orifice means a sudden restriction or change in cross-sectional area over a short length. Fluid is rapidly accelerated through the orifice with resultant turbulent flow downstream of the restriction.

The fluid resistance resulting from an orifice is then

$$R_f = \frac{P_{21}}{Q} = \frac{\rho}{2C_d^2 A_0^2}|Q|. \tag{4.66}$$

Assuming that a mechanical fluid pump is a pure pressure source, P_{21}, and that one can measure the flow rate at its outlet, Q_i when the pump pipe is in contact with the atmospheric air, the internal fluid resistance, r_i is

$$r_i = \frac{P_{21}}{Q_i}. \tag{4.67}$$

By analogy to the simple electric circuit previously described, matching the impedances ($R_f = r_i$) yelds the orifice area that maximizes the power transfer:

$$A_0^2 = \frac{\rho|Q|Q_i}{2C_d^2 P_{21}}. \tag{4.68}$$

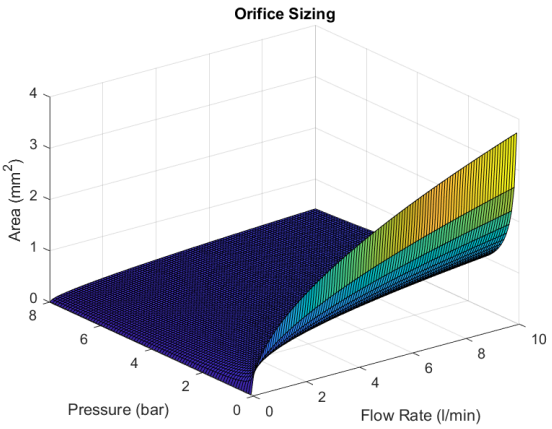


Figure 4.11: Orifice area as a function of the flow rate and pressure.

An orifice plate is considered, $C_d = 0.61$ [66]. For a given Q_i the orifice area is directly proportional to the flow rate, Q, and decreases with the pressure, P_{21}, as depicted in Fig. 4.11. For the current test bench, $Q_i = 0.0015m^3/s$. Experiments show that the pressure does not need to be too high in order to generate the necessary lifting force. So, a pressure of $P = 1.1bar$ is used. This means that the area is ultimately chosen depending on the desired Q. The thickness of the fluid layer should not be too small. Otherwise, small imperfections on the surfaces of the testbed could disturb the air flow. Too high a flow rate and the generated lift force may not be able to levitate the rotor. Another reason so as to not to choose the biggest possible value for Q is to augment the amount of time the air compressor can be used without needing to refill its tank. A value of $Q = 0.00023m^3/s$ was chosen empirically, resulting in an optimal orifice radius of approximately 1mm.

Chapter 5

Implementation

In section 2.4, two different Spacecraft models were presented. The model to be used will depend on the satellite's mission objectives. For example, in the case of an Earth observation mission, the Nadir Pointing Spacecraft Model is the most adequate. However, if the satellite's mission is to study more distant space bodies, like the Sun or other stars, then the Inertial Pointing Spacecraft Model shall be used.

In the present work, the satellite's attitude control with momentum wheels is considered. The system to be controlled is depicted in Fig. 5.1 with a high level of abstraction.

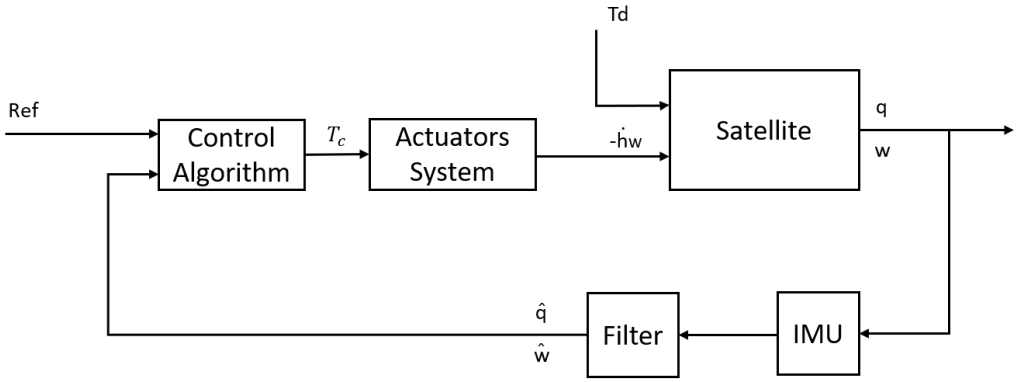


Figure 5.1: Scheme of the system to be controlled.

The satellite is subject to control and disturbance torques, which will affect its attitude (denoted by **q** and **w**). The spacecraft angular velocity, linear acceleration and the magnetic field around it are sensed by an IMU. The data from IMU is filtered and the state estimation ($\hat{\mathbf{q}}, \hat{\mathbf{w}}$) is used in the control algorithm which provides the reference for the control torque, $\mathbf{T}_c$. The reference control torque is used in the actuators' system to control the momentum wheels, so as to provide a torque as close as possible to the reference. Ideally, the actuators' system is able to provide instantaneously the reference torque: $\mathbf{T}_c = -\dot{\mathbf{h}}_w$.

5.1 Simulation Environment

The simulation environment aims at realistically describing the conditions the spacecraft will be subject to in order to validate the developed attitude determination and control algorithms. The environment was implemented in *MATLAB* and *Simulink* and is mainly composed of 8 parts.

The first is the ”orbit propagator”, which takes the spacecraft force perturbations and computes the spacecraft's position and velocity in the ECI frame according to equation 2.14.

The ”environment” block takes the spacecraft position and computes the Earth to Sun vector, Earth to Moon vector, Solar Radiation Perturbation Acceleration, air density and magnetic field. This block was adapted from the work of [67].

The ”orbit perturbations” block computes the following perturbation forces and torques:

- taking the spacecraft position, computes the Non-spherical Earth perturbation from equation 2.41;
- taking the spacecraft position and also the Earth to Sun and Earth to Moon vector, computes the Third-body forces from equation 2.42;
- taking the spacecraft velocity, position and air density, computes the aerodynamic drag force from equation 2.43;
- the Solar Radiation acceleration comes directly from the ”environment” block.

This block also outputs the torque perturbations:

- taking the Earth's magnetic field and the spacecraft attitude relative to the ECI frame, the Magnetic Field-Induced Torques are computed from equation 2.50;
- taking the spacecraft position and attitude relative to the ECI frame, the gravitational torque is computed according to 2.32;
- given the Solar Radiation acceleration from the ”environment” block, the Solar Radiation torque can be computed from equation 2.51;
- given the aerodynamic drag force, the Atmosphere-Induced Torque can be computed from equation 2.49.

The ”spacecraft model” block takes the momentum wheel angular momentum, disturbance and control torques, spacecraft position and velocity to implement equation 2.20 and 2.22 or 2.31 in continuous-time.

The ”sensors model” block takes the spacecraft acceleration from the ”orbit propagator” block, the magnetic field from the ”environment” block and the spacecraft angular velocity from the ”spacecraft model” block, adds white Gaussian Noise and a bias according to the values that come from the identification process. For the gyroscope, a slow time-varying bias was considered. This phenomenon was taken into account when designing the filters from sections 3.3.1 and 3.3.3.

The ”actuators model” block takes a torque command and outputs the wheels angular momentum and the actual torque provided by the momentum wheel system. A model for the DC motors that includes

the deadzone, saturation and discretization (finite number of possible PWM values) is included, along with the control algorithms for the momentum wheels. More on this on section 5.2.1.

The ”attitude determination” block takes the data from the ”sensors model” block and filters it according to the attitude determination algorithms from chapter 3.

Lastly, the ”control algorithms” block takes the estimated attitude from the ”attitude determination” block and wheels angular momentum from the ”actuators model” block and applies the control algorithms from chapter 3 to output the torque command that is the input to the ”actuators model” block.

5.2 Control Methods

5.2.1 Momentum Wheel Control

The Actuators' System block holds the inner control logic that determines the momentum wheel input voltage from the Torque Command. Two control strategies are considered.

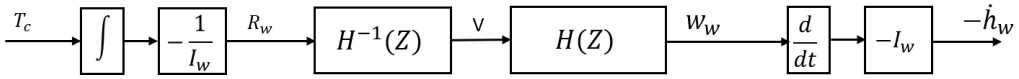


Figure 5.2: First momentum wheel control strategy.

The first control strategy (fig 5.2) performs a discrete-time integration of the torque command to compute a reference flywheel angular velocity, R_w. This value is used to invert the momentum wheel model, $H(Z)$, and find the input voltage, v, that produces R_w. The applied control torque, $-\dot{h_w}$, is hard to compute in real-time, because it would require measuring and computing the derivative of the flywheel angular velocity, w_w. The main drawback of this strategy is that it relies on the accuracy of the momentum wheel model.

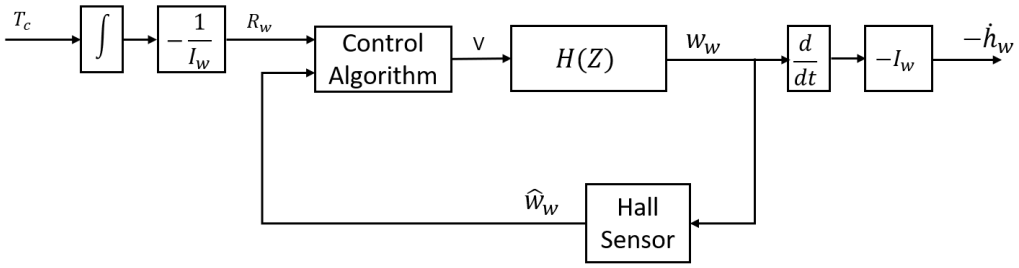


Figure 5.3: Second momentum wheel control strategy.

The second control strategy (fig 5.2) is similar to the previous one in that the Torque Command is used to compute a reference for the flywheel's angular velocity, R_w. However, the wheel angular velocity measured by the Hall Sensor is used in a feedback control loop to compute the input voltage of the momentum wheel. The main disadvantage of this strategy is that it requires extra hardware and polling it at a high rate.

PID control was described in section 3.2.1. The derivative term will not be used when controlling the momentum wheel angular velocity. A feedback of the torque provided by the inertia wheel is difficult in

practice as it would imply computing the derivative of the flywheel angular velocity computed from the Hall sensor data. As an alternative, the control of the momentum wheel will be based on its angular velocity:

$$\frac{w_w}{v} = \frac{K_v}{s\frac{R_M I_w}{K_M K_v} + 1}. \tag{5.1}$$

The proportional and integral terms of the control law are found by converting the transfer function 5.1 to a state-space model with state w_w and input v and augmenting the state with the integral of the error velocity $e = w_w - R_w$, where R_w is the reference for the angular velocity, discretizing it for a sampling rate of 100Hz, and computing the LQ control law for this state-space model. The gains used to compute the control law were 100 for the state cost corresponding to w_w, 1 for the integral of w_w and 1 for the input cost. The sampling time of this loop is 10 times smaller, since this is an inner loop and the pointing control law assumes $\mathbf{T}_c = -\dot{\mathbf{h}}_w$.

Some phenomena was neglected when computing the LQ control law so as to allow for a linear relation between the input voltage and the wheel angular speed. In fact, there is a deadzone near the zero angular velocity. In fact, the wheel rotates only down to an input voltage of 0.9 Volts. Also, the input voltage is subject to saturation, with a maximum of 6 Volts.

The hall-effect sensor is sampled each $t_{hall} = 0.0001s$ and a counter is incremented each time. When there is a change in polarization, it means that the wheel rotated $2\pi/8rad = \pi/4rad$, since each rotation corresponds to 8 polarization changes. The estimated angular velocity is then given by

$$\hat{w}_w = \frac{\pi}{4 \times t_{hall} \times counter}. \tag{5.2}$$

And the maximum angular velocity that can be estimated is $\hat{w}_{w_{max}} = pi/(4 \times 0.0001 \times 1) = 7854rad/s$, which is high enough. Note however, that the accuracy of the estimates is lower for high angular velocities. For example, the second highest measurable angular velocity is $pi/(4 \times 0.0001 \times 2) = 3927rad/s$. The third one is $pi/(4 \times 0.0001 \times 3) = 2618rad/s$ and so on. So, one wants to choose t_{hall} so that the accuracy in the range of angular speeds of the momentum wheel is acceptable. In the present case, $w_w \in [100, 770]rad/s$, the precision is on average 9 rad/s. There is a trade-off between the precision of this measurement and the computational burden of polling the sensor. In fact, a precision of 9 rad/s is not enough to use feedback control in the present context. As a result, the first control strategy is used on the experimental setup, even though the second control strategy was used in simulation (with $t_{hall} = 1 \times 10^{-6}s$), as it is a better solution. As an alternative, it is advisable the use of a brushless DC motor with integrated speed controller. Another issue to be noticed is that the saturation limits of the momentum wheel system are tighter on the experimental setup than on simulation. This is not a concern in practice, since the experimental setup is used to test short time maneuvers.

Pulse-Width Modulation (PWM)

The input V in equation 5.1 is implemented in hardware by a PWM signal. The Raspberry Pi base clock is 19,2 MHz. The PWM duty cycle should be high enough so that the wheel's angular velocity is not affected by applying a PWM signal instead of the actual desired voltage. So, one should use the

minimum clock divisor in the Raspberry Pi: 2. Then, the PWM counter is incremented at a frequency of $19.2/2 = 9.6MHz$. The last parameter to choose is the range. A range of 4 means that the PWM frequency is $f_{PWM} = 19.2/2/4 = 2.3MHz$. However, there are only 5 possible duty cycle values : $0/4, 1/4, 2/4, 3/4$ and $4/4$. In this work, a range of 480 is used, meaning that the PWM frequency is $f_{PWM} = 480KHz$.

Anti-windup

If the control signal saturates, the integral part of the PI controlller keeps integrating. This effect is known as integrator-windup or reset-windup. This effect is visible, for example, when the input to a momentum wheel motor builds up to the maximum duty cycle value, 480. From then on, if the desired control torque would require the wheel angular velocity to continue increasing, the saturation of the actuator prevents this from happening, but the integrator keeps on integrating the error. If a desaturation procedure is carried out, it would take too much time, because the integrator would require some time to discharge.

To solve this issue, the integrator is only allowed to charge up to an upper and discharged down to a lower values. The upper and lower saturation limits are $u_{i_{max}} = u_{max} - u_p$ and $u_{i_{min}} = u_{min} - u_p$, where $u_{max} = 480$, $u_{min} = -480$ and u_p is the proportional control term.

5.2.2 Pointing Algorithm - LQR

The linear state feedback that minimizes the performance index from equation 3.5 is obtained using the *MATLAB* command *dlqr*. For the Nadir Pointing Spacecraft model, matrices A and B are given by equation 2.34. The C and D matrices are $\mathbf{I}_6$ and $\mathbf{0}_3$ respectively. The matrices $\mathbf{Q}_L$ and $\mathbf{R}_L$ were again obtained by trial and error and fixed on $\mathbf{Q}_{L_w} = 50\mathbf{I}_3$, $\mathbf{Q}_{L_q} = 0.5\mathbf{I}_3$ and $\mathbf{R}_L = 1 \times 10^6.\mathbf{I}_3$. Given the matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}, \mathbf{Q}_L, \mathbf{R}_L$ and the sampling time, tc, the matlab code that allows to obtain the gain for the linear state feedback in 3.6 is below.

```
sys = ss(A, B, C, D);
sysD = c2d(sys, tc);
K = dlqr(sysD.A, sysD.B, Q, R);
```

LQR Extension: Integral Action

Applying a simple LQR shows that in the presence of disturbance torques the attitude cannot be regulated and the spacecraft is kept pointing to a point different from $\mathbf{q} = [1, 0, 0, 0]^T$. To overcome this difficulty, the state space model is augmented with an extra state that corresponds to the integral of the attitude **q**. Using the same **A**, **B**, **C** and **D** matrices described in the previous section, the linear state feedback can be obtained in Matlab using the following set of commands.

```
A_aug = [A, zeros(6,3); zeros(3,3), eye(3), zeros(3,3)];
B_aug = [B;zeros(3,3)];
C_aug = [C,zeros(6,3)];
D_aug = D;
```

```
sys = ss(A_aug,B_aug,C_aug,D_aug);
sysD = c2d(sys, tc);

K = dlqr(sysD.A,sysD.B,[Q,zeros(6,3);zeros(3,6),QI],R);
```

Matrices $\mathbf{Q}_L$ and $\mathbf{R}_L$ were maintained as in the previous section and the $\mathbf{Q}_I$ was obtained by trial and error and fixed on $\mathbf{Q}_I = 1 \times 10^{-2}.\mathbf{I}_3$.

5.2.3 Detumbling Algorithm

After its launch, the spacecraft is said to be tumbling, since its angular rates are usually very large, which cannot be effectively controlled by the Momentum Wheel System (MWS). The detumbling mode employs either thrusters or magnetorquers. As referred in section 3.2.4, the commanded magnetic dipole moment is given by 3.15. Resorting to equation 3.17, the value $k_{det} = 1.35 \times 10^{-5}$ is obtained. The detumbling algorithm is applied at 10 Hz.

5.2.4 Momentum Dumping Algorithm

As referred in section 3.2.3, the commanded magnetic dipole for momentum dumping purposes is given by equation 3.12. In order to avoid commanding any extra torque besides the one from the main controller, an equal term is added to the commanded momentum wheels' torque. This way, the total torque affecting the spacecraft due to the momentum dumping algorithm is null and it can be used continuously. By trial and error, a value of $k_{md} = -1$ was obtained. The momentum dumping algorithm is applied at 10 Hz.

5.3 Attitude Determination Methods

5.3.1 Magnetometer Calibration

Denoting the magnetometer measurement by $\mathbf{B}_y$:

$$\mathbf{B}_y = \mathbf{C}_{si}\mathbf{B}_{m_B} + \mathbf{n}_h + \mathbf{n}_m, \tag{5.3}$$

where $\mathbf{n}_m \in \mathbb{R}^3$ is the measurement noise; $\mathbf{A}_I^B \in M(3,3)$ is the rotation matrix from the inertial to the body frame; $\mathbf{B}_{m_B} = \mathbf{A}_I^B\mathbf{B}_{m_I} \in \mathbb{R}^3$ is the Earth's magnetic field expressed in the body frame; $\mathbf{n}_h \in \mathbb{R}^3$ is the hard iron distortion; $\mathbf{C}_{si} \in M(3,3)$ is the soft iron distortion.

To retrieve an ideal measurement, absent any magnetic interference, $\mathbf{B}_c = \mathbf{A}_I^B\mathbf{B}_{m_I}$, one needs to correct for the hard and soft iron effects:

$$\mathbf{B}_c = \mathbf{C}_{si}^{-1}(\mathbf{B}_y - \mathbf{n}_h). \tag{5.4}$$

The measurements of an ideal magnetometer would lie on an origin-centered sphere with radius equal to the magnetic field strength.

Hard iron effects are stationary interfering magnetic noise sources, such as metallic objects on the cirtuit board with the magnetometer, motors and currents, as well as other external magnetic fields. These, deviate the origin of the sphere of measurements.

Soft iron effects arise from objects near the sensor which distort the surrounding magnetic field, stretching and tilting the sphere of ideal measurements. The resulting measurements lie on an ellipsoid.

$\mathbf{C}_{si}$ and $\mathbf{n}_h$ can be retrieved by the best-fitted ellipsoid, minimizing the following expression by value decomposition such as in [68]:

$$min((\mathbf{B}_y - \mathbf{n}_h)^T(\mathbf{C}_{si}^{-T}\mathbf{C}_{si}^{-1})(\mathbf{B}_y - \mathbf{n}_h) - 1). \tag{5.5}$$

The results of a sample magnetometer calibration procedure are depicted in Fig. 5.4.

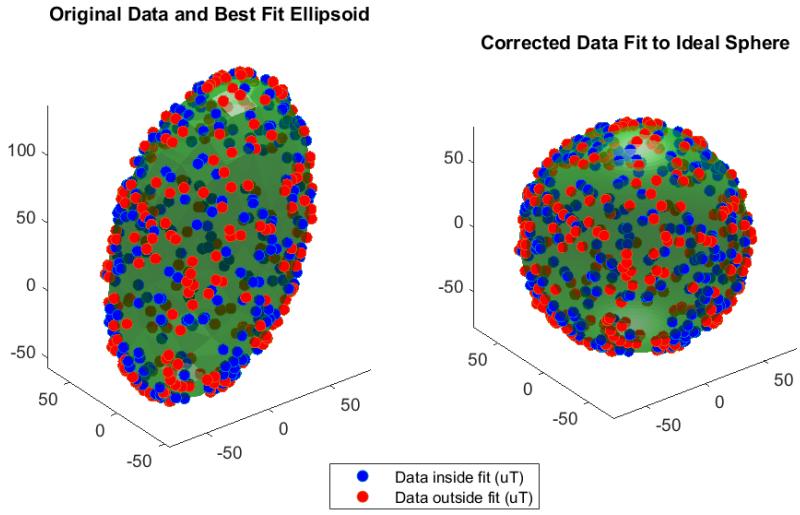


Figure 5.4: Sample magnetometer data, best-ftted ellipsoid and corrected data to fit the ideal sphere.

5.3.2 Quaternion from Earth's Gravity and Magnetic Field Observations

In the experimental setup, the accelerometer can be used as an inclinometer to retrieve the spacecraft pitch, θ, and roll, ϕ, angles. Even though this would not be possible in orbit, this is not a concern, since this setup has an educational purpose. Firstly, bias and other accelerations should be subtracted from the accelerometer readings to retrieve the gravity field estimation, $\hat{\mathbf{g}}$, as in [51]. For $\phi = \theta = 0$, estimated gravity field would be $\hat{\mathbf{g}} = [\hat{g}_x, \hat{g}_y, \hat{g}_z]^T = [0, 0, g]^T$, with $g = 9.81ms^{-2}$. It is well known that the pitch and roll angles can be retrieved as follows

$$\phi = atan2(\hat{g}_y, -\hat{g}_z), \tag{5.6}$$

$$\theta = -atan2(-\hat{g}_x, \sqrt{\hat{g}_y^2 + \hat{g}_z^2}). \tag{5.7}$$

The Euler representation $(\phi, \theta, 0)$ can then be used in formula 2.10 to obtain the equivalent quaternion.

The magnetometer reading is first corrected for hard and soft iron distortions as in eq. 5.4. Then, it is rotated by $-\theta$ and $-\phi$, resulting in an intermediate vector $\mathbf{B} = [B_x, B_y, B_z]^T$. The yaw angle related to the NED frame is given by

$$\psi_{NED} = -atan2(B_y, B_x) - \delta, \tag{5.8}$$

where δ is the magnetic declination at the spacecraft's position. Having the yaw related to the NED frame, orbit data is needed to obtain the yaw related to the LVLH frame.

5.3.3 Euler-Based Complementary Filter

The Euler-based complementary filter runs at 10Hz, the same frequency at which the satellite is actuated. The values for matrices **Q** and **R** were obtained by trial and error and set on

$$\mathbf{Q} = diag(8, 8, 8, 0.005, 0.0016, 0.0016, 0.0005, 0.025, 0.025),$$
$$\mathbf{R} = diag(10, 25, 5).$$

The filter gains were obtained resorting to the function *kalman* from *MATLAB* applied to the linearized system 3.33 with the previous values for the gain matrices **Q** and **R**.

The parameters of the noise model were obtained by the Allan deviation plot. Data corresponding to 1.5 hours was collected and the plot was obtained resorting to the *allanvar* function from *MATLAB*. From Fig. 5.5,

$$\mathbf{T}_{BI} = [153.6 \quad 256 \quad 153.6]^T s.$$

5.3.4 Quaternion-Based Complementary Filter

The quaternion-based complementary filter was implemented as described in section 3.3.2. The accelerometer and magnetometer readings are used to compute an observation quaternion. The gyroscope readings are used to find a predicted quaternion. In the update step, the SLERP technique is used to find an intermediate quaternion between these two, that is the output of the filter.

The filter runs at 10Hz with a parameter $t = 0.2$, meaning that for the most part the output quaternion is similar to the one that comes from the acccelerometer and magnetometer readings, but it is corrected by a fraction of 1/5 by the output of the prediction step. This is important because the integration of the gyroscope readings are crucial to have better results in rapid maneuvers, but need to be corrected to prevent it from drifting due to the integration of sensor errors. Most of the maneuvers in the satellite are

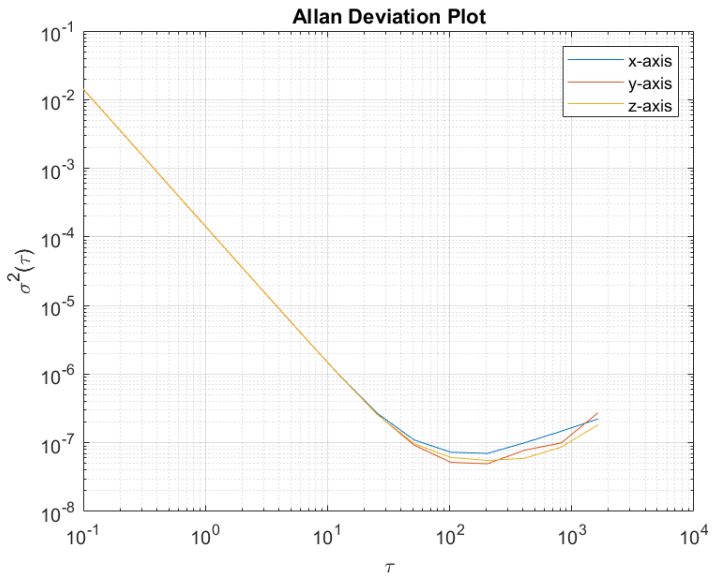


Figure 5.5: Allan Deviation Plot.

slow maneuvers, meaning that the relative importance of the gyroscope must be smaller.

5.3.5 Quaternion-Based Kalman Filter

The quaternion-based kalman filter runs at 10Hz, the same frequency at which the satellite is actuated. The values for the matrices **Q** and **R** were obtained by trial and error and set on

$$\mathbf{Q} = diag(10, 10, 10, 0.01, 0.01, 0.01),$$
$$\mathbf{R} = diag(1, 1, 1).$$

The filter gains were obtained resorting to the function *kalman* from *MATLAB* with the previous values for the gain matrices **Q** and **R**.

5.4 Code Deployment and Communication Protocols

MATLAB/Simulink has support for Target Hardware, namely, the Raspberry Pi. *Simulink Coder*^{TM} (formerly known as Real-Time Workshop) generates and executes C and C++ code from Simulink models. The generated code can be used for real-time applications, rapid prototyping and hardware-in-the-loop testing.

Monitor and Tune (External Mode) can be used to tune parameters and monitor a Simulink model running on the target hardware. The effects of different parameter values can be viewed in *Simulation Data Inspector*. The Raspberry Pi 4 Model B comes with an on-board 802.11n Wireless LAN adapter, meaning that a router can serve as the interface between the computer where the MATLAB/Simulink is running and the Raspberry Pi, as long as they are both connected to the same router and the Raspberry Pi IP address is known. The scheme on Fig. 5.6 illustrates the structure of this communication service.

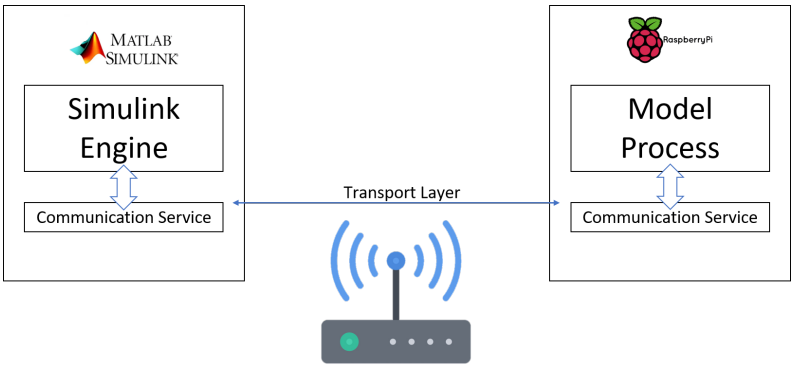


Figure 5.6: Signal Monitoring and Parameter Tuning.

The Simulink Support Package for Raspberry Pi Hardware includes a library of Simulink blocks for configuring and accessing I/O peripherals and communication interfaces. This library is used to retrieve data from the IMU, reading and writing on digital pins. As for the other blocks, the PWM block implementation is not available. This does not allow to configure the PWM frequency and number of possible duty cycles. So, in a quest to have control of these parameters, custom Device Driver Blocks [69] were created, making use of the WiringPi library [70].

Chapter 6

Simulation Results

6.1 Problem Description

The considered spacecraft inertia matrix is given by equation 4.22.

In section 6.2, the results for the proposed estimation algorithms are presented. For that, the simulation is initialized with the spacecraft body frame aligned with the LVLH frame and no angular velocity between these two frames. No control is applied to regulate the spacecraft's attitude. Throughout the simulation, the spacecraft's attitude evolves due to the movement of the satellite in orbit and modeled perturbations. The simulation is left running for the time corresponding to one orbit period. The considered orbit is the same as the ISS. This is a quasi-circular LEO orbit. The considered orbit characteristics are summarized in Tab. 6.1.

	ISS orbit
Epoch	01 Jan 2019 00h:00min:00s
Semi-major axis	6781.16 Km
Eccentricity	0.000845
Initial Mean Anomaly	2.8298 rad
Inclination	0.90278 rad
Right ascension of ascending node:	1.966 rad
Argument of perigee:	1.396 rad

Table 6.1: Orbit parameters.

In section 6.3, to compare the several solutions proposed for the attitude control of a satellite, two simulation cases are considered. To test the pointing capabilities, the Euler angle errors of the spacecraft are set to 10 degrees in pitch, roll and yaw and there's no initial body frame angular velocity relative to the LVLH frame. To test the detumbling mode, the spacecraft angular attitude is set to $\bar{\mathbf{q}}_i = [1, 0, 0, 0]^T$ and the body frame rate to 30º/s in pitch, roll and yaw.

In section 6.4, the same simulation scenario from section 6.3 is considered, but the controllers are fed with the output of each estimator, instead of the real state. This results in 6 simulation cases resumed in Tab. 6.2.

The accelerometers were modeled as being affected by a constant bias. The rate gyros were mod-

	LQR	LQRI
Euler-based complementary filter	Case 1	Case 2
Quaternion-based complementary filter	Case 3	Case 4
Quaternion-based kalman filter	Case 5	Case 6

Table 6.2: Simulation cases in section 6.4.

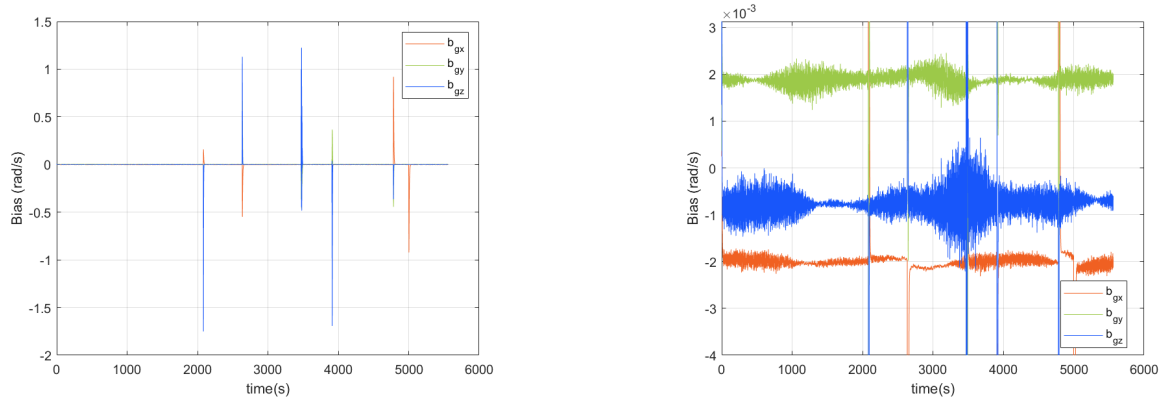
eled as being affected by a slow time-varying bias. The magnetometer was modeled by equation 5.3. The three sensors were modelled as being affected by additive white gaussian noise. The values used in simulation correspond to the ones from the real IMU, LSM9DS1, obtained through recorded data, as described in section 4.1.2. The covariance of the data is obtained. Then, the Power Spectral Density (PSD) can be estimated by $PSD = covariance \times sampling frequency$.

6.2 Estimation

This section presents the results of the estimation algorithms for an entire orbit around the Earth. The satellite is not actuated during this period, so that the satellite achieves unrestricted attitude values. The momentum wheels rotate at their nominal rate. The filters parameters are resumed in the following table.

	Parameters
Euler-Based Complementary Filter	t_s=0.1s; $\mathbf{T}_{BI}$ = diag([153.6, 256, 153.6]); **Q** = diag([8, 8, 8, 0.005, 0.0016, 0.0016, 0.0005, 0.025, 0.025]); **R** = diag([10, 25, 5]);
Quaternion-Based Complementary Filter	t_s=0.1s; t=0.2;
Quaternion-Based Kalman Filter	t_s=0.1s; **Q** = diag([8,8,8,2e-04,2e-04,2e-04]); **R** = diag([10,25,5]);

Table 6.3: Filters parameters.



(a) Zoom on the vector part of the quaternion error.

(b) Zoom on the bias estimation.

Figure 6.1: Bias estimation for the Euler-based complementary filter(Left) and Zoom (Right).

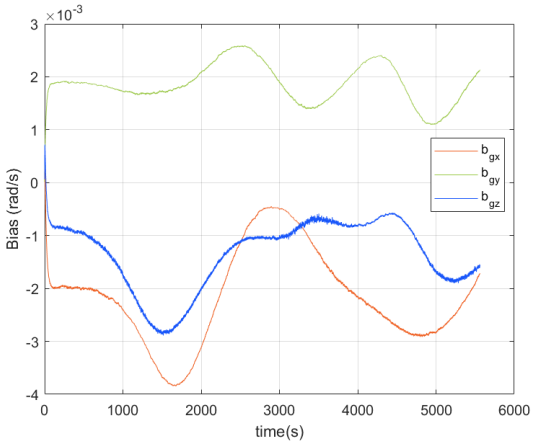


Figure 6.2: Bias estimation for the quaternion-based kalman filter.

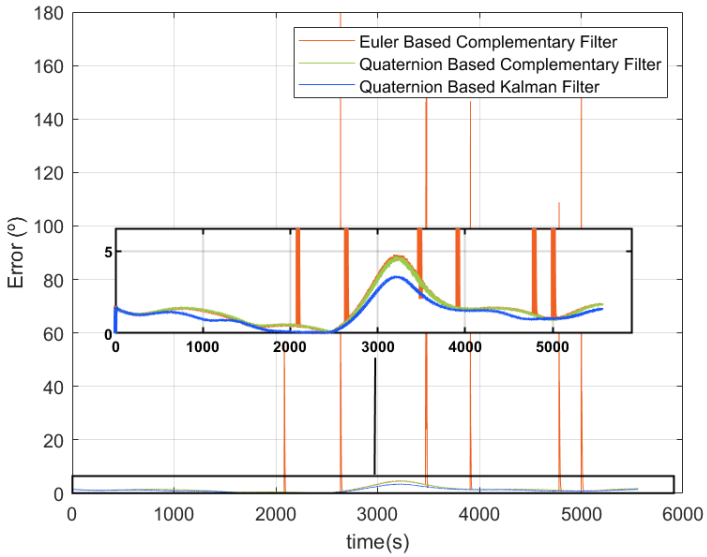


Figure 6.3: Estimation error.

6.2.1 Discussion

Regarding the Euler-based complementary filter (see Fig. 6.3), it is evident that the estimation is disturbed by peaks in the estimation error (computed from the error quaternion). Naturally, these are accompanied by disturbances in the rate gyros' bias estimation (see Fig. 6.1(a)). In spite of being converted to quaternion, the estimation is originally done based on the Euler angles formulation as described in section 3.3.1. Being the Euler angles domain restricted, each time one of the limits is crossed and there is a discontinuity on the estimation, the filter will deviate from the nominal estimation for some time. Another drawback is that in [51], asymptotic stability of the filter is only proved for pitch angles no bigger in absolute value than $\pi/2$. For a satellite this is not sufficient. However, on the zoom of Fig. 6.3, it is evident that the normal performance of this filter is comparable to the quaternion-based complementary filter, regarding the estimation error. Furthermore, the Euler-based complementary filter is

able to estimate the gyroscope bias (see Fig. 6.1(b)). The quaternion-based kalman filter performance is slightly better than the one from the other filters when the estimation error is taken into account. The bias estimation, however, is more oscillating than the one from the Euler-based complementary filter. This is explainable by the fact that the kalman filter was based on a linearization of the kinematics equation arround a zero pointing error and body frame rate. When no control is applied, the system deviates significantly from the linearization point. When compared to the other two filters, the SLERP-based filter has the disadvantage of not being able to estimate the gyroscope biases. The estimation error remains below 5 degrees for the three filters. For the Quaternion-based Kalman filter, the estimation error remains below 3.5 degrees.

6.3 Control

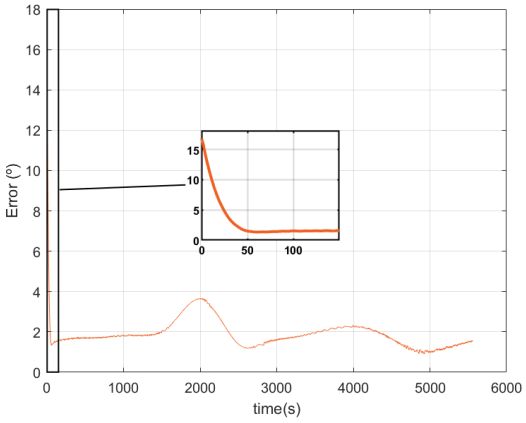
6.3.1 Nominal mode

Figures 6.4, 6.5 and 6.6 correspond to the recovery from a 10º error in pitch, roll and yaw. A simulation time corresponding to one orbit is considered for demonstrating the pointing capability for a considerable amount of time. The momentum dumping algorithm is continuously active. The control feedbacks are fed with the real attitude and body rate, so that the results are free from errors induced by the estimation algorithms. The parameters used to obtain the control linear feedback laws are resumed in Tab. 6.4.

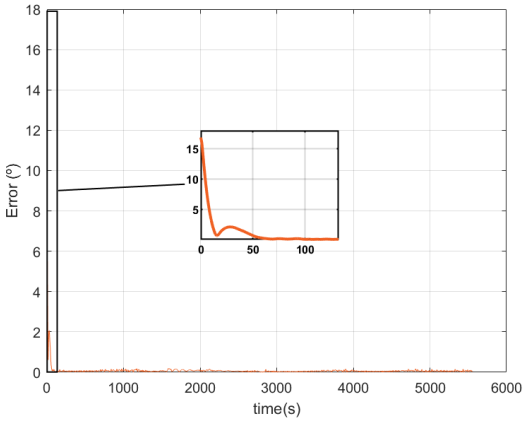
To test the robustness of the designed controller, the LQR controller with integral action was applied to the nonlinear spacecraft system described with a Monte Carlo perturbation model: in the inertia matrix, the off diagonal elements are randomly selected between 0 and $1 \times 10^{-4} Kg.m^2$; the initial Euler angle errors are randomly selected between 0 and 90 degrees; the initial angular rates are randomly selected between 0 and 1 deg/sec. 300 Monte Carlo Simulation runs are conducted and the results shown in Fig. 6.7. A first order model of the Earth's magnetic field was considered along with the ideal case of $\mathbf{T}_c = -\dot{\mathbf{h}}_w$ and only the magnetic and gravity gradient disturbance torques were taken into account.

	Parameters
Linear Quadratic Regulator	tc = 0.1s; $\mathbf{Q}_{L_w} = 50\mathbf{I}_3$; $\mathbf{Q}_{L_q} = 0.5\mathbf{I}_3$; $\mathbf{R}_L$=1e06.$\mathbf{I}_3$
LQR with Integral Action	tc = 0.1s; $\mathbf{Q}_{L_w} = 50\mathbf{I}_3$; $\mathbf{Q}_{L_q} = 0.5\mathbf{I}_3$; $\mathbf{R}_L$=1e06.$\mathbf{I}_3$; $\mathbf{Q}_I$=0.01$\mathbf{I}_3$
Detumbling	$k_{det} = 1.35 \times 10^{-5}$
Momentum Dumping	$k_{md} = -1$

Table 6.4: Control parameters.

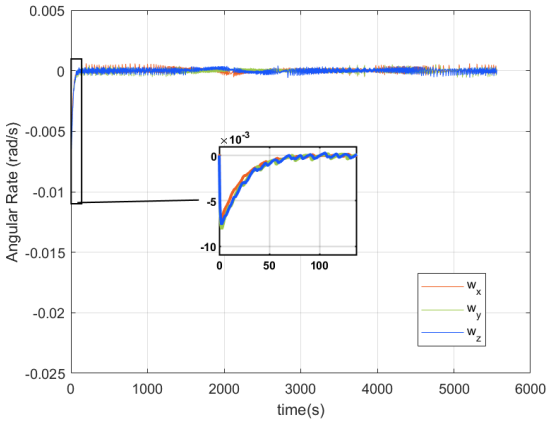


(a) Pointing error for the LQR.

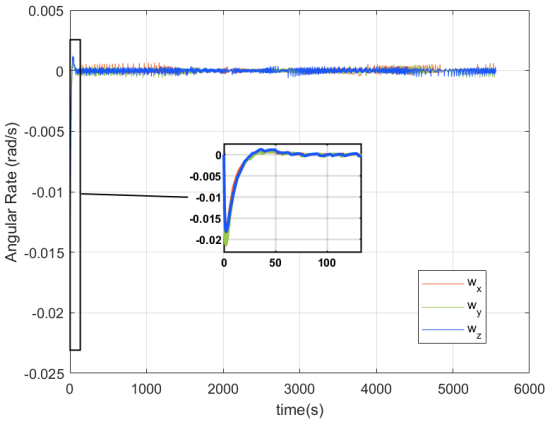


(b) Pointing error for the LQR with Integral Action.

Figure 6.4: Pointing error for the LQR (Left) and the LQR with Integral Action (Right).

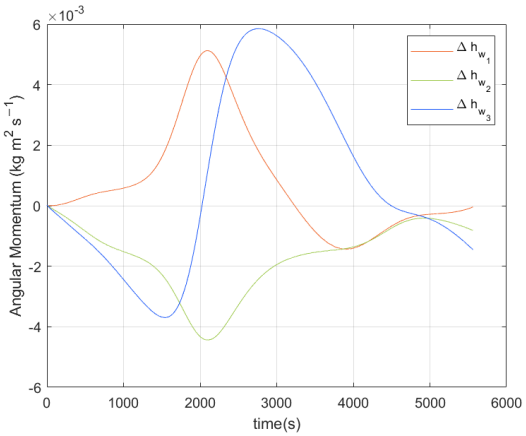


(a) Angular rate for the LQR.

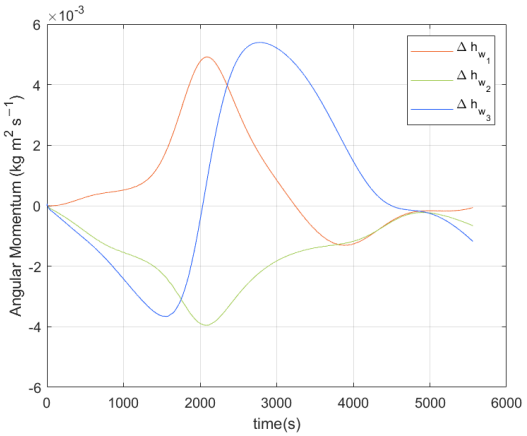


(b) Angular rate for the LQR with Integral Action.

Figure 6.5: Angular rate for the LQR (Left) and the LQR with Integral Action (Right).



(a) Tracking error for the LQR.



(b) Tracking error for the LQR with Integral Action.

Figure 6.6: Wheels' angular momentum tracking error for the LQR (Left) and the LQR with Integral Action (Right).

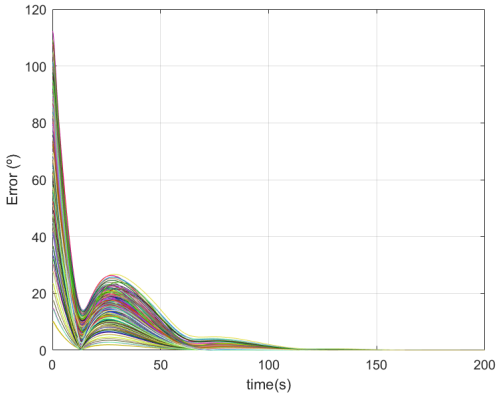
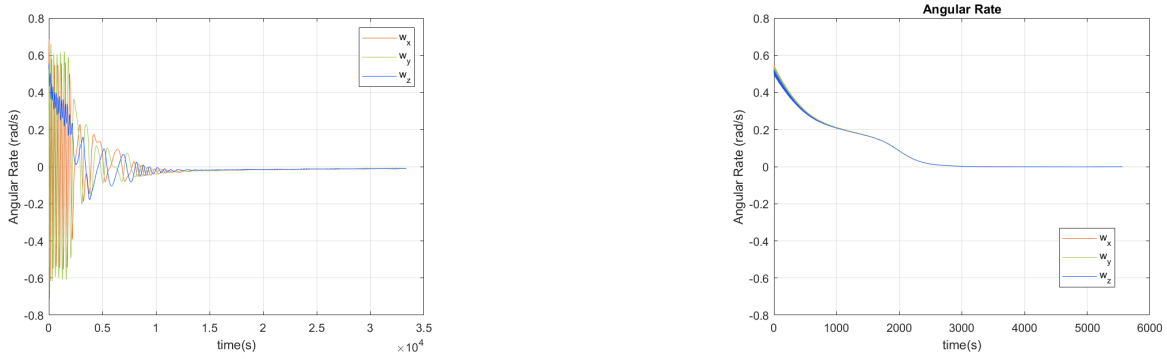


Figure 6.7: Monte Carlo Simulation for the nominal mode of the spacecraft (Nadir Pointing).

6.3.2 Detumbling

Figure 6.8 corresponds to the spacecraft angular rate during the detumbling procedure from an initial angular rate of [30, 30, 30] º/s. The detumbling law is fed with the raw gyroscope data. Two situations are considered. In the first (Fig. 6.8(a)), the momentum wheels were not spinning, while in the second (Fig. 6.8(b)), they were spinning at their nominal rate.

As referred in section 3.2.4, the selected detumbling controller is stable. To test this and also the controller robustness to parameter uncertainty, it was applied to the nonlinear spacecraft system described with a Monte Carlo perturbation model: in the inertia matrix, the off diagonal elements are randomly selected between 0 and $1 \times 10^{-4} Kg.m^2$; the initial Euler angle errors are randomly selected between 0 and 360 degrees; the initial angular rates are randomly selected between 0 and 30 deg/sec. The wheels are not spinning. A first order model of the Earth's magnetic field was considered and only the magnetic and gravity gradient disturbance torques were taken into account. 300 Monte Carlo Simulation runs are conducted and the results shown in Fig. 6.9.



(a) Wheels not spinning.

(b) Wheels spinning at the nominal rate.

Figure 6.8: Angular rate of the satellite during the detumbling procedure when the wheels are not spinning (Left) and spinning at the nominal rate (Right).

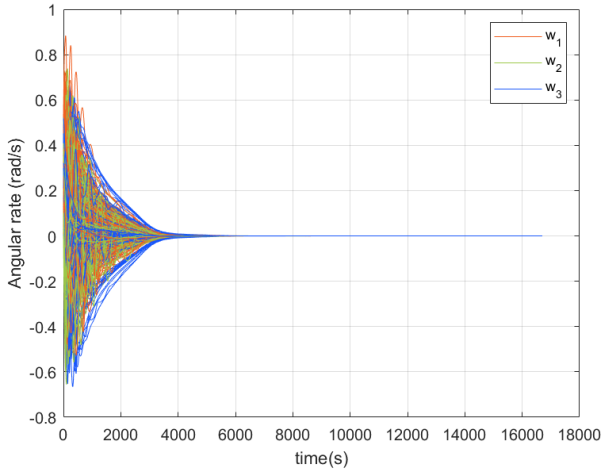


Figure 6.9: Monte Carlo Simulation for the Detumbling Mode.

6.3.3 Discussion

Regarding the LQR (see Fig. 6.4(a), 6.5(a) and 6.6(a)), it is evident that the LQR controller is sufficient to stabilize the system. The pointing error, however, is not regulated to zero, even though it remains below 4 degrees. Regarding the adding of the integral action (see Fig. 6.4(b), 6.5(b) and 6.6(b)), the pointing error is now regulated to zero. This controller uses a slightly smaller portion of the range of actuation of the momentum wheels. Also to be noted is the undershoot in the attitude.

The momentum dumping algorithm successfully prevents the momentum wheels angular momentum from drifting indefinitely from its nominal value. In Fig. 6.6(a) and 6.6(b) it is noticeable that the tracking error is close to zero around t=5000s. Note that the dumping of the angular momentum is not uniform around the orbit. This is due to the fact that magnetic torque is not available at all times in all directions. However, it is available in all directions at different times, meaning that momentum dumping using magnetorquers is possible.

The Monte Carlo simulation for the nominal mode of the spacecraft (Fig. 6.7) showed that, even though the controller was designed based on a linearization around the reference attitude and zero angular velocity, it is sufficient to bring the pointing error to zero even when the spacecraft is far from its desired attitude.

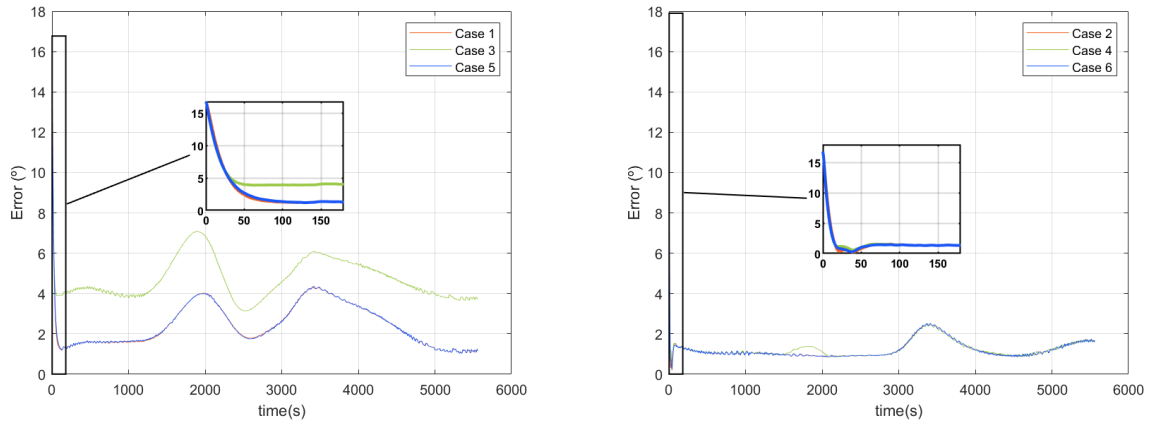
Magnetic detumbling with the magnetic torques takes a long time to detumble the spacecraft. This is due to the fact that magnetic momentum is not available in all directions all the time. The results depend whether the momentum wheels are spinning or not, since they provide a resistance to the spacecraft rotation. If the wheels are not rotating, the spacecraft is considered detumbled after three orbits. If the wheels are rotating, it takes only one orbit to achieve that state. It is preferable to detumble the spacecraft with the momentum wheels spinning at their nominal rate. Otherwise, the action of putting them to rotation afterwards may induct a tumbling behavior to the spacecraft.

In Fig. 6.9, it is evident that the detumbling is effective for various initial angular rate and orientation. Even though global stability could only be proven for a diagonal moment of inertia, the results from the Monte Carlo simulation show that the controller is robust to parameter uncertainty in the spacecraft

moment of inertia. Detumbling is possible even if the off-diagonal entries of the inertia matrix are not zero. Also, stability was proven for a continuous control signal. However, the control law was applied at 10Hz, meaning that stability is not guaranteed. Nevertheless, the results from Fig. 6.9 show that this is not a concern in practice.

6.4 Coupled Estimation and Control

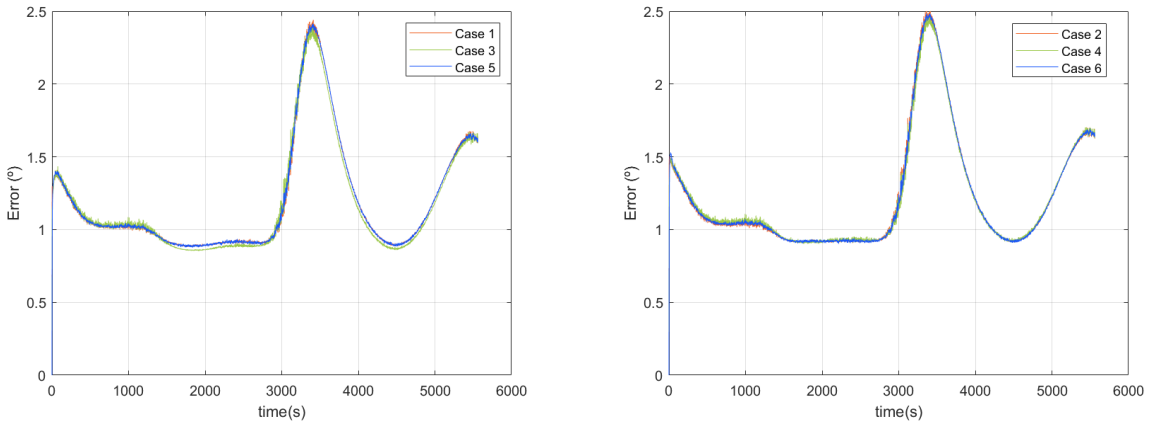
In this section each coupled pointing and estimation algorithms are tested as depicted in Tab. 6.2. The control and estimation parameters are the ones from tables 6.3 and 6.4 respectively. The simulation scenario is the same described in section 6.3.



(a) Pointing error for the LQR.

(b) Pointing error for the LQR with Integral Action.

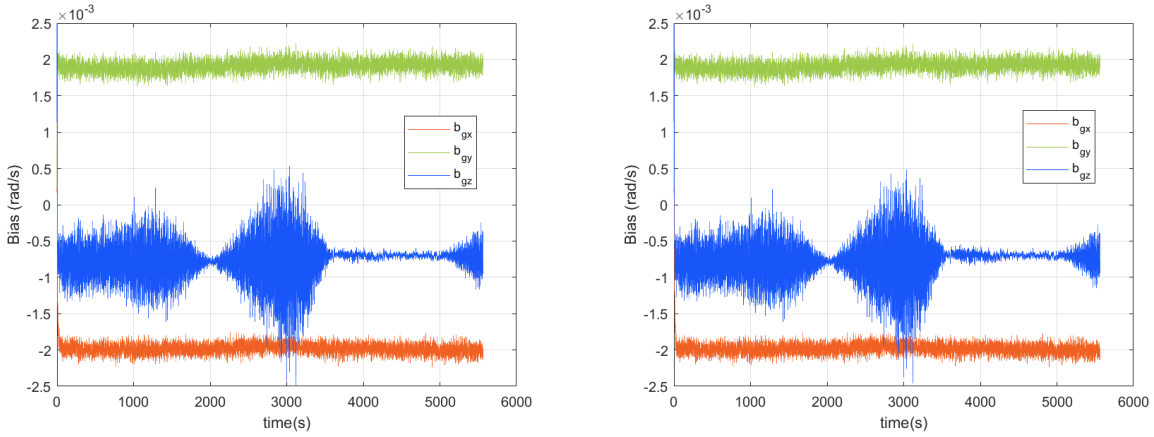
Figure 6.10: Pointing error for the LQR (Left) and LQR with Integral Action (Right).



(a) Estimation error for the LQR.

(b) Estimation error for the LQR with Integral Action.

Figure 6.11: Estimation error for the LQR (Left) and LQR with Integral Action (Left).



(a) Case 1.

(b) Case 2.

(c) Case 5.

(d) Case 6.

Figure 6.12: Bias estimation for cases 1, 2, 5 and 6.

6.4.1 Discussion

Simulation cases 1, 3 and 5 regard the LQR controller. The pointing accuracy stays bellow 4.5 degrees for cases 1 and 5. For case 3, a pointing accuracy of only roughly 7 degrees was attained (see Fig. 6.10(a)). Even though the estimation error is similar to the three estimators (see Fig. 6.11(a)), the Quaternion-based complementary filter is not able to compensate for the gyroscope bias, resulting in a poorer pointing accuracy.

Simulation cases 2, 4 and 6 regard the LQR with integrative action controller (Fig. 6.10(b)). In all 3 cases, the controller is able to align the body frame with the LVLH frame with a pointing accuracy of less than 2.5 degrees. With this controller, the unavailability of bias estimation in case 4 is compensated, resulting in a similar pointing accuracy.

Regarding the bias estimation (Fig. 6.12) both the Euler-based complementary filter and the Quaternion-based Kalman filter are capable of providing an adequate estimation, as opposite to the results from section 6.2. This is due to the fact that with the control action the system does not deviate significantly from the linearization point of the Kalman filter. Regarding the estimation error (Fig. 6.11), it was below 2.5 degrees for all cases. The Euler-based complementary filter has the smallest convergence time.

The bias estimation of the Quaternion-based Kalman filter is less noisy. The filters performance in the nominal mode are, for the most part, comparable.

Chapter 7

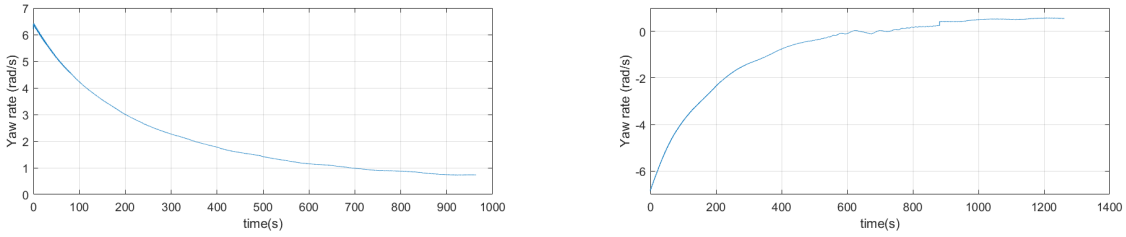
Experimental Results

The results from this section regard the control of the developed prototype in 1 direction. With that in mind, the upper part of the rotor sphere was removed and a weight was added at the bottom of the CubeSat, forming a pendulum, so that the pitch and roll errors were compensated by gravity. The inertia on the z axis remains approximately the same and the satellite is not controlled on the x and y body axis. This procedure aims at verifying the integration of the several subsystems, i.e., the momentum wheels system, the Raspberry Pi, IMU and batteries.

The controller and estimator parameters are the same from chapter 6. The spacecraft model is different because only the yaw momentum wheel is rotating. Several experiments are performed. In section 7.1, the prototype is set to rotation and the angular velocity decay is measured so as to study the damping of rotations. In section 7.2, the 6 simulation cases from chapter 6 are applied to the CubeSat functional prototype.

Videos demonstrating the use of the platform can be found on [71].

7.1 Perturbation Torque Characterization



(a) Clockwise rotation decay.

(b) Counter-clockwise rotation decay.

Figure 7.1: Angular velocity decay without actuation when the prototype is set to rotate clockwise (Left) and counter-clockwise (Right).

7.1.1 Discussion

As depicted in equation 4.46, the viscous drag causes the angular speed to slow down exponentially. This is an important aspect to characterize, since it will cause the experiment results to deviate from what would happen in space. This means that the external torque is proportional to the angular velocity $T = -k\omega$. Denoting the differential equation of the angular velocity as $\dot{\omega} = -\lambda\omega$, the decay constant, λ, is related to the external torque by the following relation $\lambda = k/I$, where I is the rotating part moment of inertia.

Using the function *fit* from the *Curve Fitting Toolbox* of *MATLAB*, the angular velocity decay for the clockwise rotation was best identified by the expression $5.506exp(-0.004196.t) + 0.687$. The counter-clockwise rotation was best identified by $-6.891exp(-0.004301.t) + 0.4926$. This means that the rotations decay more rapidly for counter-clockwise rotations. This goes along with observation, since it was evident that the rotor had a deterministic tendency to rotate clockwise, settling at a given positive yaw rate. So, it is not only the viscous drag that affects the testbed. This other external torque should be eliminated so as to augment the reliability of the obtained test results.

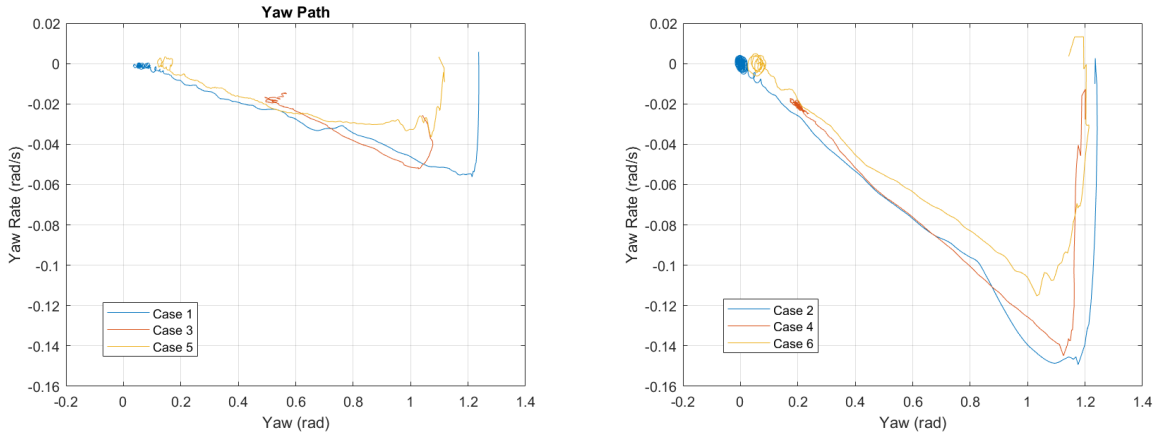
The testbed was required to have disturbance torques at most of the same order of magnitude of those present in space. Approximating the decay constant for the viscous drag by the average of the two previous decay constants: $0.004285s^{-1}$. Note that $\lambda \sim 4 \times 10^{-3}s^{-1}$, $I \sim 3 \times 10^{-3}Kg.m^2$, meaning that $k \sim 1.2 \times 10^{-5}Kg.m^2s^{-1}$. Taking into account that $w \sim 0.1rad/s$, the external torque due to the viscous drag $T \sim 10^{-6}N.m$. The disturbance torques present in space are of the order of $10^{-6}N.m$, meaning that this requirement is attained. A lower disturbance torque is theoretically possible. Imperfections on the surfaces of the testbed may be the source of these other friction torques. Also, a slight miscenter or tilt angle of the orifice may induce a spontaneous rotation to the sphere, affecting the damping of rotations.

7.2 Pointing

In this section, the 6 cases previously studied in chapter 6 are tested in the ground-testing facility, for the yaw axis only. The results are plotted in a phase plane in Fig. 7.2. The original plots, as a function of time, can be found on appendix C. The table with the definition of the 6 cases is reproduced for convenience.

	LQR	**LQRI**
Euler-based complementary filter	Case 1	Case 2
Quaternion-based complementary filter	Case 3	Case 4
Quaternion-based kalman filter	Case 5	Case 6

Table 7.1: Simulation cases in section 7.2.



(a) LQR controller.

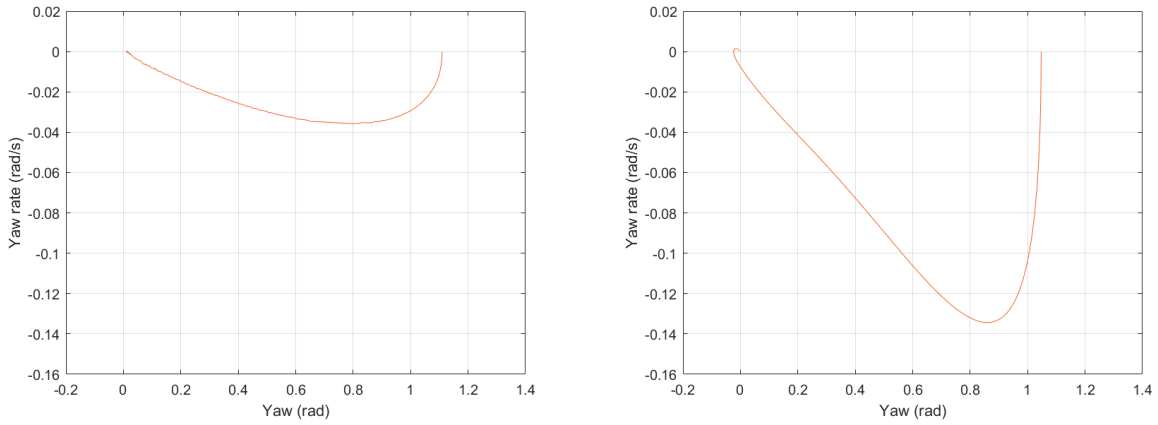
(b) LQR with Integral Action controller.

Figure 7.2: Experimental results plotted in the Phase Plane.

7.2.1 Discussion

For the LQR controller (cases 1, 3 and 5), it is evident that the yaw angle is not regulated to zero (see also Fig. C.1, C.3 and C.5). The use of the Euler-based complementary filter produces better pointing accuracy, less than 6 degrees. The pointing accuracy of the Quaternion-based Kalman filter is twice as large. The use of the Quaternion-based complementary filter results in a pointing accuracy of about 40 degrees. This was not predicted by the simulations results. In fact, the gyroscope biases used in simulation were identified with the motors turned off. These values are more in hand with those that would be present on a satellite in orbit. When the motors are turned on, the biases increase significantly. Since this filter is not able to estimate the biases, its use produces the worst results. For the LQR with integrative action, the steady state error is smaller (see also Fig. C.2, C.4 and C.6). The pointing accuracies for cases 2, 4 and 6 are 5 degrees, 15 degrees and 6 degrees respectively. The pointing accuracy required for the ISTsat-1 is 20 degrees, meaning that all estimators are able to attain this requirement, when coupled with an LQR controller with integrative action.

In Fig. 7.3, the paths from simulation, with a constant disturbance of $1 \times 10^{-6} N.m$, are shown in the Phase Plane. The results from the testbed are similar to the ones from these figures. This validates the testbed ability to be used to test the ADCS system. Also, it validates the software simulation environment previously described.



(a) LQR controller.

(b) LQR with Integral Action controller.

Figure 7.3: Ideal (Yaw, Yaw Rate) paths.

Note that the testbed was not equipped with a measurement system that could provide true attitude measurements. So, as opposite to the simulation plots, the attitude and angular rate from the experimental plots are not the true values, but the ones that come from the estimation algorithms.

As a final remark on the validity of the obtained results, note that the testbed is affected by an external torque different from the viscous drag predicted on chapter 4. This is evident by the results from section 7.1 and the visual evidence that if not actuated, the testbed's rotor has a tendency to start rotating. The gravity torque due to the displacement between the center of mass and the center of rotation could not be completely compensated and contributes to some wobbling of the platform. Other than that, it is inevitable that de viscous drag would deviate the results from the ones on a perfectly drag-free environment. The gravity gradient torque disturbance, introduced in section 2.5.2, is similarly felt on the surface of the Earth but with a higher magnitude, since the gravity force is bigger on the surface of the Earth than on orbit. The other disturbance torques felt by the spacecraft on orbit are not naturally present on the testbed and additional hardware should be added if those conditions were to be simulated.

Chapter 8

Conclusions and Future Work

The main aim of this thesis was to perform an exploratory study on the development of a low-cost prototype for testing of the ADCS of nanosatellites. This prototype can be divided into 2 parts: a 1U CubeSat functional prototype and an air-bearing based platform for simulation of the spacecraft motion. Algorithms for the attitude control and determination with the simple suite of actuators and sensors were to be selected and tested against a simulation environment in *MATLAB/Simulink* and later in the testbed itself.

8.1 Conclusions

In this first iteration of the prototype, the only sensor to be used, besides the Hall sensors, is the LSM9DS1 IMU, providing acceleration, angular rate and magnetic field measurements. The momentum-wheel system was designed based on sample requirements for a typical 1U Cubesat. The air-bearing testbed physics was studied so as to provide a good foundation on the sizing and construction of the air-bearing based platform. Then, the satellite prototype and testbed were drawn in *SolidWorks* and COTS components were incorporated. The satellite prototype structure was 3D-printed in PLA and the testbed's was built by a CNC machining company. The choice of the 3D-printing manufacture method was due to the low-cost of this process and also to the possibility of rapid prototyping and testing of different configurations.

Regarding the testbed, theory shows that it is possible to create a full 360 degrees pitch, roll and yaw simulator, even though many other attempts to create such a simulator were unsuccessful. The contribution of this work for the effort of creating such a platform is two-folded. Firstly, a review of previous efforts and existent platforms is undertaken. The physics of air-bearing platforms is detailed, after which a platforms was designed aiming at full 3 DOF testing. It was able to provide an environment for 360 degrees yaw testing. A functional full 3 DOF testing platform requires further work. One needs to overcome the aerodynamic torque that puts the platform into rotation. Other areas of improvement include the gravity torque due to the displacement between the center of rotation and center of mass, and the disturbance caused by the rib in the junction of the 2 rotor half-spheres.

Regarding the Attitude Determination and Control System (ADCS) system, one momentum dumping two detumbling, two pointing and three estimation algorithms were studied. Detumbling with magnetic torques is achieved in less than one orbit period if the wheels are spinning at their nominal rate. Regarding the pointing capability, simulation shows that in nominal conditions a pointing accuracy of 2.5 degrees is achievable. Coupling of the LQR controller with integral action with the Quaternion-based Kalman filter showed to be the best choice.

Taking the established requirements into consideration, regarding the ADCS, a pointing accuracy of (much) less than 20º was achieved without reaching the limit of 5º/s. The satellite can be detumbled from 30º/s (and even higher angular velocities). Regarding the air-bearing based spacecraft simulator, a full 3 DOF concept was studied and built. Theoretical analysis of the physics behind it and the practical challenges to its construction were explored. The platform was tested for full 1 DOF on the yaw axis. Further work was identified to achieve 3 DOF.

8.2 Future Work

Due to the multidisciplinary nature of the present work, the suggestions for future work will be divided in topics.

8.2.1 Displacement between the center of mass and the center of rotation

In order to provide a torque-free simulator, a testbed must be designed and constructed in such a way that the center of gravity and the center of rotation are co-located.

A manual approach for this challenge, would be the adding of a simple system with counterweights. A set of four rods can be used to manually select the position of counterweights and fix their position with nuts.

Fine resolution, however, is only attained through the real time implementation of adaptive mass-balancing onboard the simulator. An automatic mass balancing system consisting of three sliding masses independently actuated by stepper motors along three orthogonal directions is proposed. In the literature, the Least Squares Method (LSM) is the most common algorithm used to provide an estimation to the location of the center of gravity and the values of the inertia tensor. The spherical design may make it hard to accommodate the mass-balancing hardware. A *tabletop*, *umbrella* or *dumbbell* system may be preferred.

8.2.2 Other disturbances

Acording to [72], anisoelastic deformation of the platform may lead to a non-negligible source of disturbance, in the same order of magnitude of the disturbance torques expected on orbit. To cancel out this and other disturbance torques, the platform may be equipped with directional air nozzles that enable external torque generation. In addition, these nozzles can be used to simulate disturbances from the space environment.

Finally, the mass distribution of the inertia wheels and its position should be precisely calibrated, so as to avoid disturbance torques generated by the momentum wheel system.

When these disturbances are eliminated, only the viscous drag affects the platform. According to the results from chapter 4, the relaxation time regarding the damping of rotations is very large, meaning that the testbed can be used to conduct insightful testing of ADCS systems. More complexity can be added to the platform to broaden the amount of algorithms it can be used to test.

8.2.3 ADCS

Only momentum wheels were integrated into the experimental setup, meaning that the momentum wheels saturation problem could not be addressed. If an Helmholtz cage was built around the platform and a magnetorquer included in the CubeSat prototype, magnetic torque based control laws could also be tested.

Nonlinear control laws should be studied, so as to allow the system to recover from large errors in attitude, that deviate the state from the linearization point. A *sliding mode* controller could be used to circumvent the unwinding phenomenon. Also, a complementary filter similar to the Euler-based complementary filter, but formulated in quaternion (see equation 3.44) could provide the same asymptotic stability properties, without the disadvantage of the mathematic singularities of the Euler angles attitude representation. Finally, when several testbeds are available, the testing of formation control techniques is possible.

At last, it was not possible to apply the PID control in the physical system. A brushless DC motor with integrated speed controller is preferable, since it avoids the overhead of polling the Hall sensor at high rates. Also, an high quality DC motor should have a more linear behavior around the zero speed. A different approach to the speed control of the momentum wheel would be the use of a tachometer generator (tachogenerator). When attached to the momentum wheel's motor, this device outputs a voltage proportional to its rotation speed.

8.2.4 Ground Truth Vision System

Adding a vision system to track the attitude of the spacecraft and provide ground-truth values is a major improvement to consider. This way, the estimation algorithms accuracy may also be tested experimentally.

8.2.5 Attitude Sensors

Nanosatellites are often equipped with Sun sensors and Earth sensors. Testing the use of this sensors can be included if Sun and Earth simulators are added to the platform. A Sun simulator aims at delivering a collimated light-beam resembling the Sun light as in [72]. An hot aluminum plate can be used as an Earth simulator as in [73].

Bibliography

[1] E. Kulu. Nanosats database. URL: https://www.nanosats.eu/, 2020.

[2] E. Buchen and D. Depasquale. *2014 Nano / Microsatellite Market Assessment.* SpaceWorks Enterprises, Inc (SEI), 1 edition.

[3] H. Helvajian and S. W. Janson. *Small Satellites: Past Present and Future.* The Aerospace Press, 1 edition.

[4] J. Puig-Suari, C. Turner, and W. Ahlgren. Development of the standard cubesat deployer and a cubesat class picosatellite. *IEEE Aerospace Conference Proceedings*, page 347–353, 2001, Big Sky, MT, USA,.

[5] T. Villela, C. Costa, and A. Brandão. Towards the thousandth cubesat: A statistical overview. *International Journal of Aerospace Engineering*, 2019(1):1687–5966, 2018.

[6] J. Bouwmeester and J. Guo. Survey of worldwide pico- and nanosatellite missions, distributions and subsystem technology. *Acta Astronautica*, 67(7-8):854–862, 2010.

[7] D. M. Meissner. A three degrees of freedom test-bed for nanosatellite and cubesat attitude dynamics, determination, and control. Master's thesis, Naval Postgraduate School, Monterey, CA, 2009.

[8] J. Schwartz, M. Peck, and C. Hall. Historical review of air-bearing spacecraft simulators. *Journal of Guidance, Control, and Dynamics*, 26, 05 2003. doi: 10.2514/1.1035.

[9] I. S. Programme. Nanostar: Hands-on higher aerospace education through nanosatellite student challenfes. URL https://nanostarproject.eu/, 2020.

[10] J. B. Monteiro and A. Guerman. Nanostar, a collaborative approach to nanosatellite education. *Advances in the Astronautical Sciences*, 173:289–296, 11 2020.

[11] P. Murugan and Y. Agrawal. Small satellites applications, classification and technologies. *International Journal of Science and Research (IJSR)·*, 9(7):1682–1687, 7 2020. DOI: 10.21275/SR20723213825.

[12] CDS. *CubeSat Design Specification, Rev. 13.* California Polytechnic State University, 2015.

[13] NASA. What are smallsats and cubesats? URL: https://www.nasa.gov/content/what-are-smallsats-and-cubesats, 2015.

[14] H. D. Black. A passive system for determining the attitude of a satellite. *AIAA journal*, 2(7): 1350–1351, 1964.

[15] J. Keat. Analysis of least-squares attitude determination routine doaop. *Technical Report*, 1977.

[16] M. D. Shuster and S. Oh. Three-axis attitude determination from vector observations. *Journal of Guidance, Control, and Dynamics*, 4(1):70–77, 1981.

[17] F. L. Markley. Spacecraft attitude determination methods. *40th Israel Annual Conference on Aerospace Sciences, Tel Aviv and Haifa, Israel*, 2000.

[18] P. Batista, C. Silvestre, and P. Oliveira. Globally exponentially stable cascade observers for attitude estimation. *Control Engineering Practice*, 20(2):148–155, 2 2012.

[19] F. L. Markley and J. L. Crassidis. *Fundamentals of Spacecraft Attitude Determination and Control.* Springer New York Heidelberg Dordrecht London, 2014. ISBN:9781493908011.

[20] Y. Yang. *Spacecraft modeling, attitude determination, and control : quaternion-based approach*. CRC Press, 2019. ISBN:978-1138331501.

[21] P. Boning and A. Dubowsky. Proceedings of 9th international symposium on artificial intelligence and robotics in space (isairs 2008. *8th International ESA Conference on Guidance, Navigation 461 and Control Systems*, pages 26–29, 2 2008.

[22] M. Romano, D. A. Friedman, and T. J. Shay. Laboratory experimentation of autonomous spacecraft approach and docking to a collaborative target. *Journal of Spacecraft and Rockets*, 44(1), 2007. doi 10.2514/1.22092.

[23] B. Agrawal and R. E. Rasmussen. Air bearing based satellite attitude dynamics simulator for control software research and development. *Proceedings of the SPIE Conference on Technologies for Synthetic Environments: Hardware-in-the-Loop Testing VI, (Orlando, Florida)*, pages 204–214, 4 2001. doi 10.2514/1.22092.

[24] B. Kim, E. Velenis, P. Kriengsiri, and P. Tsiotras. A spacecraft simulator for research and education. *Proceedings of the AIAA-AAS Guidance, Dynamics and Control Conference*, 1(367):897–914, 2001.

[25] S. Cho, J. Shen, N. McClamroch, and D. S. Bernstein. Mathematical models for the triaxial attitude control testbed. *Mathematical and Computer Modelling of Dynamical Systems*, 9(2):165 – 192, 6 2003. doi 10.1076/mcmd.9.2.165.16520.

[26] F. Roe, D. W. Mitchell, B. M. Linner, and D. Kelley. Simulation techniques for avionics systems-an introduction to a world class facility. *Proceedings of the AIAA Flight Simulation Technologies Conference*, 96-3535:9535–543, 1996.

[27] D. Gallardo and R. Bevilacqua. Six degrees of freedom experimental platform for testing 460 autonomous satellites operations. *8th International ESA Conference on Guidance, Navigation 461 and Control Systems,*, pages 1–11, 2011.

[28] N. Jovanovic, J. M. Pierce, and J. Praks. Design and testing of a low-cost, open-source, 3-d printed air-bearing-based attitude simulator for cubesat satellites. *Journal of Small Satellites*, 9(2):859–880, 2019.

[29] EyasSat. The 3dof air bearing test chamber for eyassat3u is here! URL: http://eyassat.com/the-3dof-air-bearing-test-chamber-for-eyassat3u-is-here/, 2014.

[30] E. Cukton, J. King, and P. Ward. Design and development of an unrestricted satellite motion simulator. 31st Annual AIAA/USU Conference on Small Satellites.

[31] D. P. Neves. Control algorithm for the istsat-1. Master's thesis, Instituto Superior Técnico, 2019.

[32] A. Janota, V. Šimák, D. Nemec, and J. Hrbček. Improving the precision and speed of euler angles computation from low-cost rotation sensor data. *Sensors*, 2015.

[33] A. B. Younes, J. Turner, M. Majji, and J. Junkins. Nonlinear tracking control of maneuvering rigid spacecraft. *presented at 21st AAS/AAS Space Flight Mechanics Meeting*, (Number AAS 11-168), 02 2011.

[34] D. Vallado. *Fundamentals of astrodynamics and applications.* Microcosm Press, El Segundo, CA., 2004. ISBN:0-521-55072-6.

[35] F. A. for Cartography and Geodesy. The 3rd realization of the international celestial reference frame (icrf3). URL https://www.iers.org/IERS/EN/DataProducts/ICRF/ICRF3/ icrf3.html., 2013.

[36] A. A. Ibrahim and E. Yamaguchi. Comparison of solar radiation torque and power generation of deployable solar panel configurations on nanosatellites. *Aerospace by MDPI*, 6(5):50–72, 2019.

[37] J. R. Wertz and W. J. Larson. *Space Mission Analysis and Design.* Microcosm Press, 3 edition.

[38] M. Sidi. *Spacecraft Dynamics and Control: A Practical Engineering Approach.* Cambridge University Press, Cambridge, UK, 1997. ISBN:0-521-55072-6.

[39] M. R. Haneveer. *Orbital Lifetime Predictions - An assessment of model-based ballistic coefficient estimations and adjustment for temporal drag coefficient variations.* PhD thesis, TU Delft, 2017.

[40] B. Wie. Solar sail attitude control and dynamics, part i. *S Journal of Guidance, Control, and Dynamics.*, 27:526–535, 2004.

[41] R. Gomes. Development of a reliable and low-cost reaction wheel system for cubesat applications. Master's thesis, Instituto Superior Técnico, 2016.

[42] N. G.-I. A. (NGA). The world magnetic model. URL: http://www.ngdc.noaa.gov/geomag/WMM/DoDWMM.shtml, 2019.

[43] H. K. Khalil and J. Grizzle. *Nonlinear systems*. 3 edition.

[44] S. P. Bhat and D. S. Bernstein. A topological obstruction to continuous global stabilization of rotational motion and the unwinding phenomenon. *Systems and Control Letters.*, 30(10):63–70, 2000.

[45] G. F. Franklin, J. D. Powell, A. Emami-Naeini, and J. D. Powell. *Feedback control of dynamic systems.* Addison-Wesley Reading, 2 edition, .

[46] N. Nichols and J. Ziegler. Optimum settings for automatic controllers. *Trans. ASME*, 64:759–768, 1942.

[47] G. F. Franklin, J. D. Powell, and M. Workman. *Digital control of dynamic systems.* Ellis Kagle Press, 3 edition, .

[48] K. J. Âstom and R. M. Murray. *Analysis and Design of Feedback Systems.* Princeton University Press, preprint available at http://www.cds.caltech.edu/ murray/am05. edition.

[49] G. Avanzini and F. Giulietti. Magnetic detumbling of a rigid spacecraft. *J. Guid. Contr. Dynam.*, 4 (35):1326–1334, 2012.

[50] T. Islam, S. Islam, and S. Mahmud. Comparison of complementary and kalman filter based data fusion for attitude heading reference system. *AIP Conference Proceedings 1919.*, 2017. doi:10.1063/1.5018520.

[51] J. Madeiras, C. Cardeira, and P. Oliveira. Vision-aided complementary filter for attitude and position estimation: Design, analysis and experimental validation. *IFAC-PapersOnLine*, 52(12):388–393, 2019. doi:10.1016/j.ifacol.2019.11.274.

[52] R. Valenti, I. Dryanovski, and J. Xiao. Keeping a good attitude: A quaternion-based orientation filter for imus and margs. *Sensors*, 8 2015.

[53] J. Solà. Quaternion kinematics for the error-state kalman filter. 2015. hal-01122406v5.

[54] N. Atlas. Ultimaker 2 3d printer pushes the limits of speed and accuracy. URL: https://newatlas.com/ultimaker-2-3d-printer-speed-accuracy/29268/, 2013.

[55] R. P. Foundation. Raspberry pi 4. URL: https://www.raspberrypi.org/products/raspberry-pi-4-model-b/?resellerType=home, 2020.

[56] P. Supply. Pijuice hat - a portable power platform for every raspberry pi. URL: https://uk.pi-supply.com/products/pijuice-standard, 2020.

[57] STMicroelectronics. 9-axis inemo inertial module (imu): 3d magnetometer, 3d accelerometer, 3d gyroscope with i2c and spi. URL: https://www.st.com/en/mems-and-sensors/lsm9ds1.html, 2020.

[58] Honeywell. Ss340rt, ss440r series. URL: https://sensing.honeywell.com/sensors/unipolar-position-sensor-ics/SS340RT-SS440R-series, 2020.

[59] Pololu. Pololu drv8835 dual motor driver kit for raspberry pi. URL: https://www.pololu.com/product/2753, 2020.

[60] B. Ribeiro. Technical drawings of the testbed and satellite prototype. URL: https://cutt.ly/ikLXj8F, 2020.

[61] E. A. Hogan and H. Schaub. Three-axis attitude control using redundant reaction wheels with continuous momentum dumping. *Journal of Guidance, Control, and Dynamics.*, 38(10):1865–1871, 2015.

[62] S. K. Peddapelli. *Pulse width modulation.* Peddapelli, S. K. (2017). Pulse width modulation. Berlin: De Gruyter., 2017.

[63] J. Snoeijer and K. der Weele. Physics of the granite sphere fountain. *American Journal of Physics*, 82(11):1029–1039, 2014. doi:10.1119/1.4886365.

[64] J. Armengol, J. Calbó, T. Pujol, and P. Roura. Bernoulli correction to viscouslosses: Radial flow between two parallel discs. *American Journal of Physics*, 76:730–737, 2009.

[65] P. E. Wellstead. *Introduction to Physical System Modelling*. Academic Press Ltd, 1979. ISBN:0-12-744380-0.

[66] Neutrium. Discharge coefficient for nozzles and orifices. URL: www.neutrium.net/fluid-flow/discharge-coefficient-for-nozzles-and-orifices/, February 2015.

[67] H. Marques. Attitude determination for the nanostar project. Master's thesis, Instituto Superior Técnico, 2019.

[68] J. F. Vasconcelos, G. Elkaim, C. Silvestre, P. Oliveira, and B. Cardeira. Geometric approach to strapdown magnetometer calibration in sensor frame. *IEEE Transactions on Aerospace and Electronic Systems*, 4 2011.

[69] T. M. Inc. Device driver blocks. URL: https://www.mathworks.com/help/supportpkg/raspberrypi/device-driver-blocks.html, 2020.

[70] G. Henderson. Wiring pi - gpio interface library for the raspberry pi. URL: https://www.http://wiringpi.com/, 2020.

[71] B. Ribeiro. Videos demonstrating the used of the air-bearing-based sattelite simulator. URL: https://cutt.ly/0kLXd2f, 2020.

[72] D. Modenini, A. Bahu, G. Curzi, and A. Togni. A dynamic testbed for nanosatellites attitude verification. *Aerospace.*, 7(31), 2020. doi:10.3390/aerospace7030031.

[73] J. Prado, G. Bisiacchi, L. Reyes, E. Vicente, F. Contreras, M. Mesinas, and A. Jurez. Three-axis air-bearing based platform for small satellite attitude determination and control simulation. *J. Appl. Res. Technol.*, 3:222–237, 2005.

Appendix A

Quaternion and Its Properties

There are different conventions to the quaternion. The present analysis is based on and follows the convention from [20].

Let the standard basis **i**, **j** and **k** for the $\mathbb{R}^3$ satisfy

$$\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1 \tag{A.1}$$

A quaternion is defined as the sum of a scalar and a vector

$$\bar{\mathbf{q}} = q_0 + \mathbf{i}q_1 + \mathbf{j}q_2 + \mathbf{k}q_3 = q_0 + \mathbf{q} \tag{A.2}$$

where q_0 is called the scalar part of the quaternion and **q** is the vector part.

From A.1 one derives that

$$\mathbf{ij} = \mathbf{k} = -\mathbf{ji} \tag{A.3}$$

$$\mathbf{jk} = \mathbf{i} = -\mathbf{kj} \tag{A.4}$$

$$\mathbf{ki} = \mathbf{j} = -\mathbf{ik} \tag{A.5}$$

which means that the product of two quaternions in non-commutative.

The sum of two quaternion, $\bar{\mathbf{q}} = q_0 + \mathbf{i}q_1 + \mathbf{j}q_2 + \mathbf{k}q_3$ and $\bar{\mathbf{p}} = p_0 + \mathbf{i}p_1 + \mathbf{j}p_2 + \mathbf{k}p_3$ is given by the sum of the scalar and vector parts

$$\bar{\mathbf{q}} + \bar{\mathbf{p}} = (q_0 + p_0) + (q_1 + p_1)\mathbf{i} + (q_2 + p_2)\mathbf{j} + (q_3 + p_3)\mathbf{k} \tag{A.6}$$

The multiplication of two quaternion is defined by

$$\bar{\mathbf{p}} \otimes \bar{\mathbf{q}} = p_0 q_0 - \mathbf{p}.\mathbf{q} + p_0\mathbf{q} + q_0\mathbf{p} + \mathbf{p} \times \mathbf{q} \tag{A.7}$$

This operation is distributive and associative, even though it is not commutative. The matrix form of

the quaternion product is

$$\bar{\mathbf{r}} = \bar{\mathbf{p}} \otimes \bar{\mathbf{q}} = \begin{bmatrix} p_0 & -p_1 & -p_2 & -p_3 \\ p_1 & p_0 & -p_3 & p_2 \\ p_2 & p_3 & p_0 & -p_1 \\ p_3 & -p_2 & p_1 & p_0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & q_3 & -q_2 \\ q_2 & -q_3 & q_0 & q_1 \\ q_3 & q_2 & -q_1 & q_0 \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{bmatrix} \tag{A.8}$$

The complex conjugate of a quaternion $\bar{\mathbf{q}}$ is denoted by $\bar{\mathbf{q}}^*$ and is given by

$$\bar{\mathbf{q}}^* = q_0 - \bar{\mathbf{q}} \tag{A.9}$$

Given two quaternions $\bar{\mathbf{p}}$ and $\bar{\mathbf{q}}$, the conjugate of the quaternion product follows the following relation

$$(\bar{\mathbf{p}} \otimes \bar{\mathbf{q}})^* = \bar{\mathbf{q}}^* \otimes \bar{\mathbf{p}}^* \tag{A.10}$$

The norm of a quaternion is defined by $|\bar{\mathbf{q}}| = \sqrt{\bar{\mathbf{q}}^* \otimes \bar{\mathbf{q}}}$ and verifies

$$|\bar{\mathbf{q}}| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2} \tag{A.11}$$

The inverse of a quaternion is defined by $\bar{\mathbf{q}}^{-1} \otimes \bar{\mathbf{q}} = \bar{\mathbf{q}} \otimes \bar{\mathbf{q}}^{-1} = 1$ and verifies

$$\bar{\mathbf{q}}^{-1} = \frac{\bar{\mathbf{q}}^*}{|\bar{\mathbf{q}}|^2} \tag{A.12}$$

A vector is considered as a pure quaternion, which means that its scalar part is zero. The quaternion operator rotates a vector, **v**, into another vector, or pure quaternion. The quaternion operator is defined by

$$\mathbf{w} = \bar{\mathbf{q}} \otimes \mathbf{v} \otimes \bar{\mathbf{q}}^* = (2q_0^2 - 1)\mathbf{v} + 2(\mathbf{q}.\mathbf{v})\mathbf{q} + 2q_0(\mathbf{q} \times \mathbf{v}) \tag{A.13}$$

The quaternion operator can be expressed by a direction cosine matrix:

$$\begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} = \begin{bmatrix} 2q_0^2 - 1 + 2q_1^2 & 2q_1q_2 - 2q_0q_3 & 2q_1q_3 + 2q_0q_2 \\ 2q_1q_2 + 2q_0q_3 & 2q_2^2 + 2q_0^2 - 1 & 2q_2q_3 - 2q_0q_1 \\ 2q_1q_3 - 2q_01_2 & 2q_2q_3 + 2q_0q_1 & 2q_3^2 + 2q_0^2 - 1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$$

The above equation defines a general rotational matrix as

$$\mathbf{A}(\bar{\mathbf{q}}) = (q_0^2 - \mathbf{q}^T\mathbf{q})\mathbf{I} + 2\mathbf{q}\mathbf{q}^T - 2q_0\mathbf{S}(q) \tag{A.14}$$

where **S**(**x**) is a skew-symmetric matrix function of $\mathbf{x} = [x_1, x_2, x_3]^T$ defined by

$$\mathbf{S}(\mathbf{x}) = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix} \tag{A.15}$$

The cross product of $\mathbf{x} \times \mathbf{y}$ can then be represented by a matrix multiplication S(x)y.

Successive rotations by $\bar{\mathbf{p}}$ followed by $\bar{\mathbf{q}}$ of a vector $\mathbf{v}$ can be expressed as

$$(\bar{\mathbf{q}} \otimes \bar{\mathbf{p}}) \otimes \mathbf{v} \otimes (\bar{\mathbf{q}} \otimes \bar{\mathbf{p}})^* = \begin{bmatrix} 0 \\ \mathbf{A}(\bar{\mathbf{q}} \otimes \bar{\mathbf{p}})\mathbf{v} \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{A}(\bar{\mathbf{q}})\mathbf{A}(\bar{\mathbf{p}})\mathbf{v} \end{bmatrix} \tag{A.16}$$

Appendix B

Electrical Connections

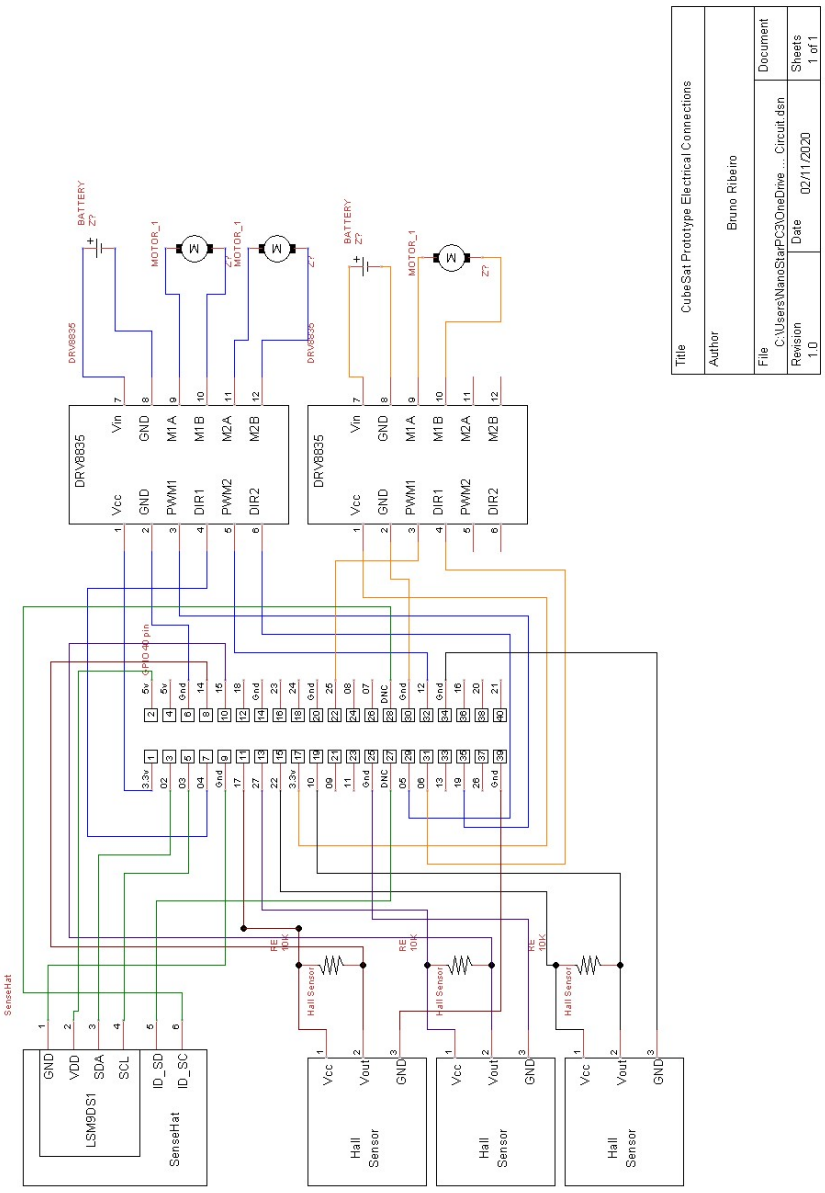
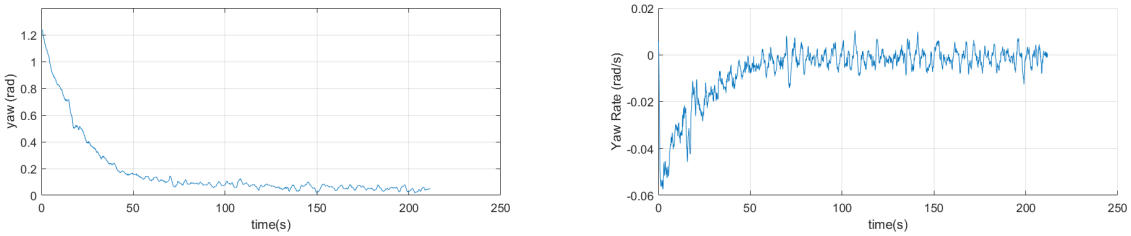


Figure B.1: CubeSat Prototype Electrical Connections.

Appendix C

Plots from the Experimental Results

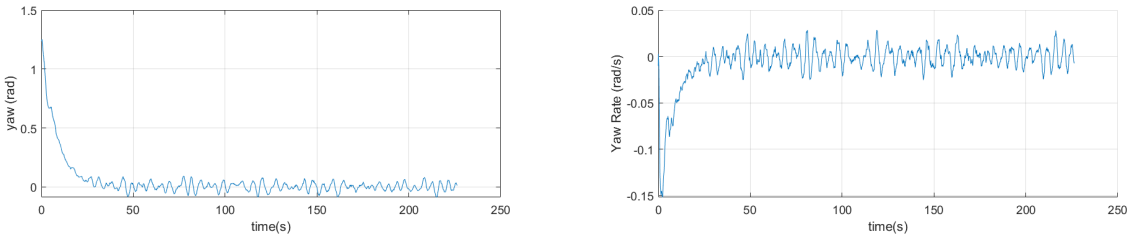
Case 1



(a) Yaw angle. (b) Yaw rate.

Figure C.1: Experimental results for case 1.

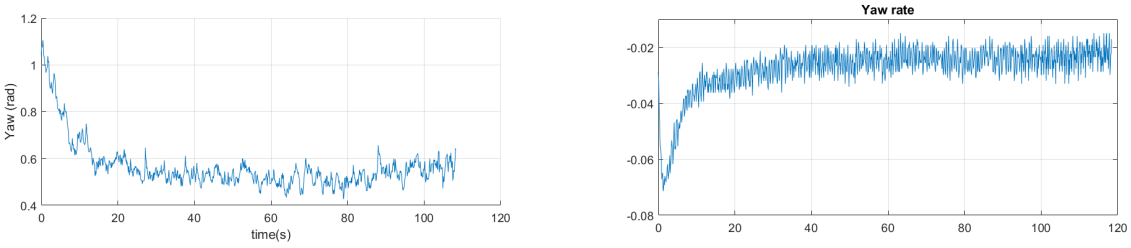
Case 2



(a) Yaw angle. (b) Yaw rate.

Figure C.2: Experimental results for case 2.

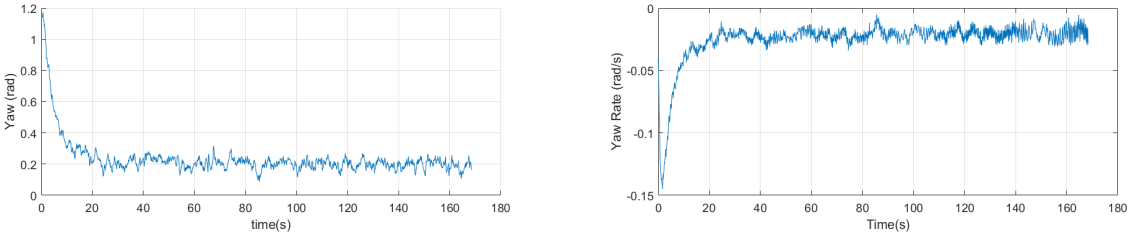
Case 2



(a) Yaw angle. (b) Yaw rate.

Figure C.3: Experimental results for case 3.

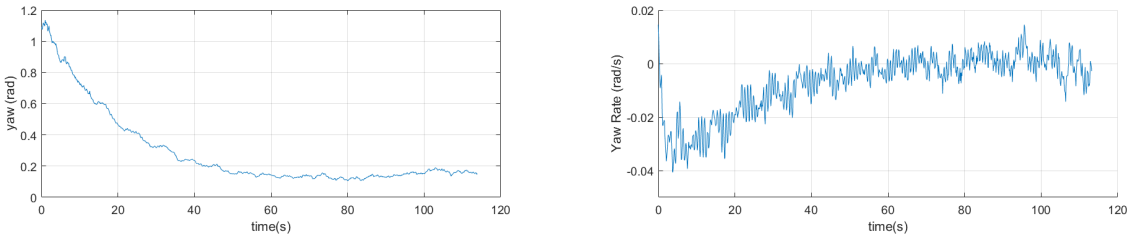
Case 4



(a) Yaw angle. (b) Yaw rate.

Figure C.4: Experimental results for case 4.

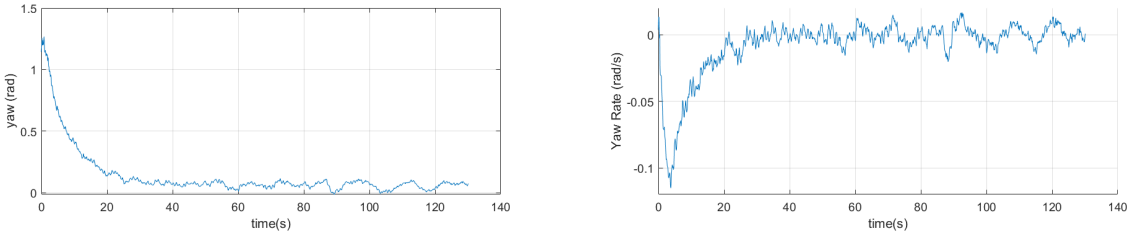
Case 5



(a) Yaw angle. (b) Yaw rate.

Figure C.5: Experimental results for case 5.

Case 6



(a) Yaw angle. (b) Yaw rate.

Figure C.6: Experimental results for case 6.

Appendix D

Prototype Iterations

The presented prototype on the main body of this dissertation is the culmination of an iterative design process. There were 3 previous main designs.

D.1 Prototype 1

In this design, the components are mounted on a flat plate and suspended by ropes. Control is performed on the yaw axis only.

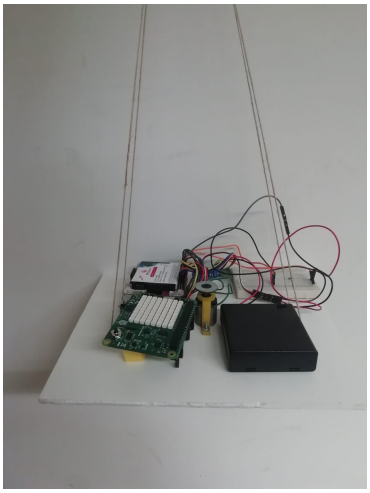


Figure D.1: Picture of prototype 1.

D.2 Prototype 2

In this design the components are mounted on a flat plate which is attached to a semi-sphere that levitated on the air cushion. The semi-sphere has a weight so as to provide a pendulum-like behavior for the pitch and roll axis. Control is performed on the yaw axis only, even though this is the typical example of a *tabletop* system, meaning that if the center of rotation and center of mass were co-located,

restricted motion on the pitch and roll axis would also be possible.

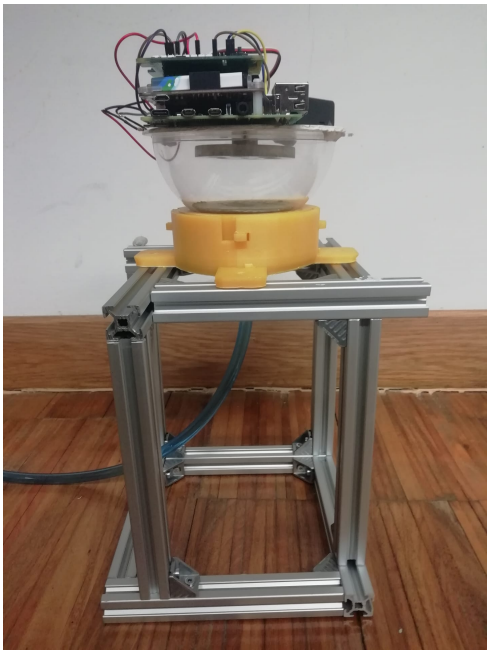


Figure D.2: Picture of prototype 2.

D.3 Prototype 3

This is similar to the final iteration, previously described in chapter 4, but the testbed was also 3D-printed in PLA.

Figure D.3: Picture of prototype 3.