POLITECNICO DI TORINO

Master's degree course in Aerospace Engineering
Academic year 2021/2022
Graduation Session October

Development of the
Active Attitude Determination and Control System
for a 3U Educational CubeSat

Supervisors:
Prof. Fabrizio Stesina
Prof. Sabrina Corpino

Defended By:
Manuel Pecorilla

"Time is relative; its only worth depends upon what we do as it is passing."

-Albert Einstein

Abstract

CubeSats are gaining a relevant position in the new space economy era. Thanks to the "low cost and fast delivery" paradigm and the miniaturisation of the electromechanical system, new missions are possible in support to larger spacecraft or as stand alone missions. CubeSats were born in the academia context and they maintain their importance as educational instruments and, also, as in orbit demonstration platforms for new systems and equipment. One of the most challenging subsystem for the CubeSats is the Attitude Determination and Control System because it involves hardware and software integration, advanced strategies and algorithms to determine and control the attitude. For this reason the ADCS is largely studied and still requires further investigations and analysis in order to achieve higher solutions with as reduced as possible efforts.

This thesis work fits into this context and aims at designing, developing, and verifying effective and reliable solutions for the Attitude Determination and Control System of the 3U CubeSat under development in the CubeSat Team for an Earth observation mission.

The proposed ADC system design involved all necessary aspects, from requirements identification to the selection and sizing of the Hardware suite. System performance and fulfilment of requirements are verified utilizing a complete simulator, developed using Matlab and Simulink, which allows both position and attitude dynamics reproduction to be realistically simulated. More specifically, an orbit propagator, designed with Cowell's method, using the fourth-order Runge-Kutta numerical integration technique, is employed to simulate position dynamics. While the attitude dynamic is modelled utilizing Euler's equation of motion, including all disturbance torques that characterise the space environment.

The defined control modes relate to the different mission phases in which the CubeSat will operate. These include the detumbling mode, the science mode, itself divided into nadir and target pointing, the desaturation mode, and other nominal and off-nominal modes.

Orbit-referenced angular rates during the detumbling phase are estimated with a robust Rate Kalman Filter using magnetometer measurements. Attitude determination, during fine pointing, involves a procedure that sequentially employs a smoothing filter, the Singular Value Decomposition method for deterministic determination, and an Extended Kalman Filter for recursive estimation.

The controllers required for detumbling and reaction wheels desaturation are implemented via the Cross Product Magnetic control law with magnetic torquers. Control during the science mode takes place using reaction wheels. A classical Quaternion feedback controller is used for the target pointing phase, while special attention focuses on advanced control algorithms implementation, including Model Predictive Control and H infinity control, for the Nadir Pointing mode. Different combinations between the optimal estimator and optimal controllers are studied to test their performance and eventually select the best solution that meets the performance requirements.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 The CubeSat standard

The CubeSat program began as a collaborative effort in 1999 between Jordi Puig-Suari, a professor at California Polytechnic State University (Cal Poly), and Bob Twiggs, a professor at Stanford University's Space Systems Development Laboratory (SSDL). One of the goals was to get universities closer to space activities through reasonable projects [10].

Another important goal was to involve students in the development of a spacecraft throughout its life cycle, therefore in the different phases [39]:

- Mission requirements development;

- Design, analysis, and testing;

- Fabrication, assembly, and quality control;

- Integration and launch;

- Satellite operations.

A first step in the development of the program was to define a standard that was not too binding but that left a certain degree of freedom to the so-called "CubeSat Developers". In addition, this has allowed many universities to experiment in the field and gain experience with space programs.

With the advancement of technological progress and the introduction of miniaturized technologies, CubeSats have attracted the attention not only of universities, but also of agencies and private commercial groups. The reason for this is due precisely to the possibility offered by the CubeSats to be able to carry out missions, whose costs of technical developments are reduced by standards, using state-of-the-art miniaturized payloads.
CubeSats come in several sizes, which are based on the standard CubeSat "unit" referred to as a 1U [9]. A 1U CubeSat is a 10 cm cube with a mass of approximately 1 to 1.33 kg. In the years larger sizes have become popular, such as the 1.5U, 2U, 3U, and 6U. Examples of a 1U and 3U are shown in Figure 1.1.

A key element of the CubeSats is that standardized dispensers have also been developed. The Poly Picosatellite Orbital Deployer (P-POD) is an aluminum, rectangular box with a spring mechanism. CubeSats slide along a series of rails during ejection into orbit. This allows launch providers to interface with different missions in the same way. Of course, the use of deployers requires the fulfillment of certain requirements by the CubeSats.

1U Standard Dimensions: 3U Standard Dimensions:
10 cm x 10 cm x 11 cm   10 cm x 10 cm x 34 cm
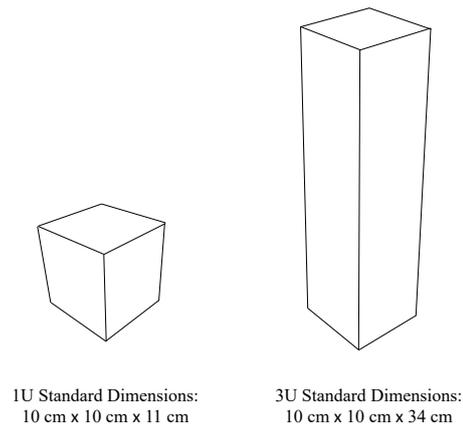
Figure 1.1: *1U CubeSat (left) 3U CubeSat (right)*

1.2   SILVA's Cubesat Project

The CubeSat PoliTO Team is a student team at the Politecnico di Torino engaged in the design and development of small space platforms for scientific missions and testing new technologies. The team was created in 2008 on the initiative of Aerospace Engineering students and professors. So far, two Cubesats have been developed: E-ST@R-I, launched into orbit in February 2012, and E-ST@R-II, launched in April 2016 thanks to the European Space Agency's *Fly your satellite!* programme.

Now the team is about to start a new mission focusing on one of the most relevant and challenging topics worldwide: climate change. In this context, the CubeSat team proposes a CubeSat Earth observation mission, SILVA (Satellite-based Innovative Land and Vegetation Analysis). The scientific purpose of Silva is to identify and analyse climate change effects on large green areas, collecting data that can be used to prevent desertification and to analyse the effectiveness of land restoration interventions or the consequences of deforestation. The aim is to accomplish this by developing a modular, cheap and efficient 3U CubeSat which will study the hydration status of vegetation by using an optical payload.

The mission is proposed and carried out by a team of students, and its educational nature heavily influenced the mission features. The necessity of keeping the complexity level and costs as low as possible made the team select a COTS optical RGB camera as the sensing instrument. A relatively low-budget instrument could easily degrade the quality of the collected data. To overcome this issue the images collected by the payload are processed and enhanced through data fusion techniques and reconstruction algorithms such as Super-Resolution Reconstruction Algorithm (SRR).

The peculiarities of this project are basically two: the first one is the on-board data processing, to increase the quality of available data, while the second is the dependable approach in spacecraft design. Indeed, Silva's CubeSat has been designed following the dependability techniques prescribed by the ECSS-Q-ST-30C.

1.3   Motivation and goals

This thesis work aims to design and develop the Attitude Determination and Control (ADCS) system for Silva's CubeSat realized by the CubeSat team of the Politecnico di Torino. As described in

Section 1.2, the quality of the images collected by the payload will be enhanced through on-board data processing, however this is not enough as a key point for obtaining good images is to have an ADCS that allows the satellite to be properly controlled, especially during the picture taking process.

The design of this sub-system is an iterative structured process that evolves in several steps where different essential inputs are considered to make proper design choices. Specifically, the CubeSat will be used in an Earth observation mission, so the ADCS shall be designed to guarantee adequate performances in terms of pointing accuracy and stability required by the observation payload.

This thesis work focuses in particular on the development and comparison of optimal control algorithms: Model Predictive Control (MPC) and H_∞ Control. In this regard, the performance, cost, robustness and stability of control are analysed and compared between the two algorithms. Determination algorithms are also implemented by considering the most common ones in the literature. Different combinations of estimation and control algorithms are considered evaluating their effectiveness and comparing their performances.

The software simulating the Attitude Determination and Control System is tested in the Matlab®-Simulink® environment as will be discussed in the following sections. The orbital simulator used for this purpose has been implemented in such a way that the simulations are reliable and allows testing of the algorithms that will later be equipped on board the CubeSat. The validation of these algorithms involves Monte Carlo simulations.

1.4 Attitude Determination and Control System

This section briefly describes the main characteristics of the Attitude Determination Control System. Given the purpose of this thesis work, attention is focused on the type of active attitude control which, as discussed in the analysis of the state of the art (in Section 1.5), currently represents the most widely used method among the CubeSats.

The ADCS represents one of the most important subsystems of the bus platform of a spacecraft, and is a critical system for the success of the mission. In particular, this system is designed to stabilise, detumble, and orient the spacecraft in a given direction. This system is necessary, for example, to avoid damage to sensitive components (for example an optical payload) due to pointing to the sun, or to control heat dissipation, or to allow communication with the ground station by pointing the antenna.

Figure 1.2 shows the feedback control and determination loop for the ADCS developed for the 3U CubeSat. The different blocks will be discussed in detail in the following sections, the diagram is presented here in order to highlight the main features of the ADCS.

The diagram schematically represents the operative logic behind spacecraft control and attitude determination by highlighting the flow of information.

The reference signal (desired attitude) is compared with the estimated attitude of the spacecraft and, from this comparison, an error is calculated which acts as an input for the controller. The controller has the task of calculating suitable command torques to be applied to the spacecraft in order to reduce the attitude error to zero. The application of these control torques takes place by activating suitable actuators which allow to change the attitude of the spacecraft. The dynamic response of the spacecraft is then evaluated using various sensors and starting from these measurements the real attitude of the satellite is estimated.
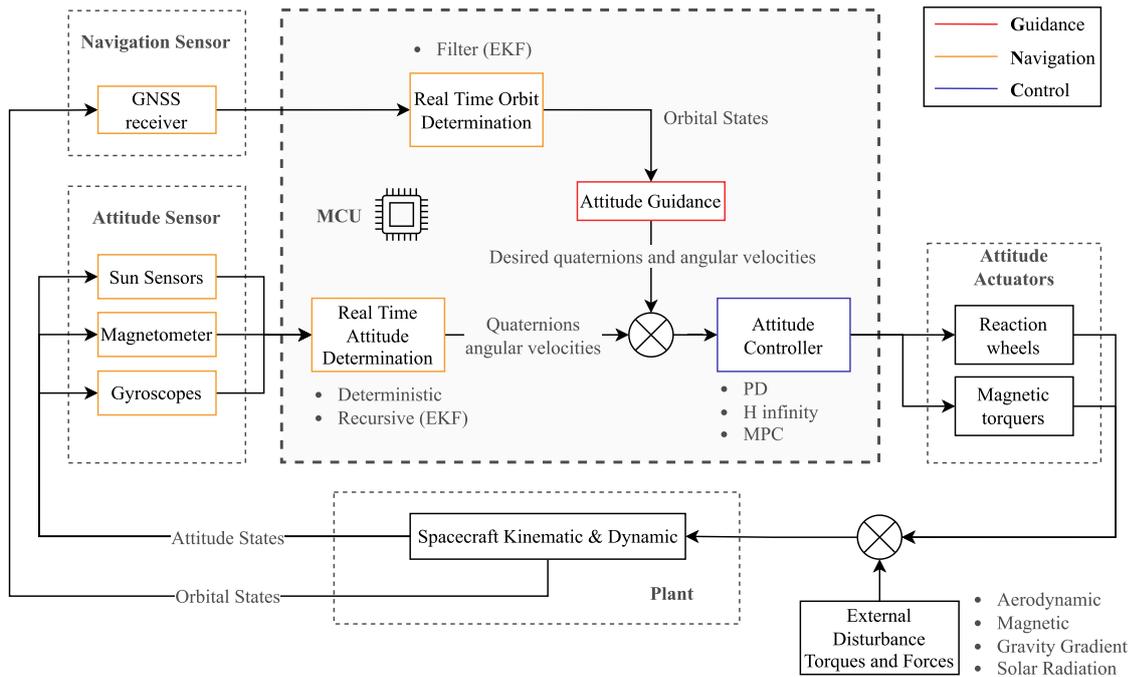
Figure 1.2: *Closed loop feedback control of a spacecraft*

From this diagram it is therefore possible to identify various elements both hardware and software characteristic of the system. In particular, as regards the hardware components it is possible to conclude that ADCS needs sensors and actuators respectively for the measurement of the attitude (and speed) and for the application of the torques. While, as far as the software is concerned, it is necessary that the system implements the estimation and control algorithms of the attitude. Figure 1.3 summarizes these elements of the system.
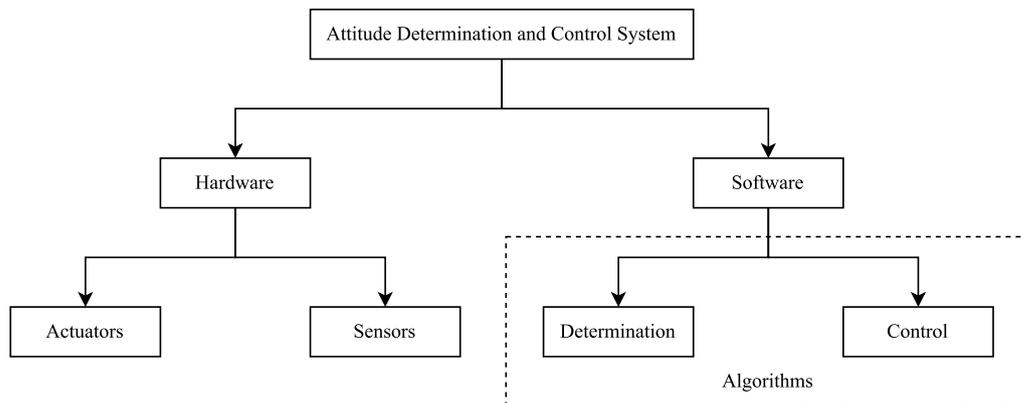
Figure 1.3: *ADCS Hardware and Software*

1.5 ADCS Current State of the Art

Figure 1.3 shows that ADCS involves hardware and software in order to perform its functions.

Nowadays, the development of the attitude determination and control system of CubeSats employs well-known software and miniaturised technology to achieve good performance with small dimensions. In this section, the state of the art of both the hardware and software of CubeSats is presented.

The following information presented is not intended to be exhaustive but provides an overview of the current state of the art technologies.

In [46] it is described the current state of the art in NanoSats/CubeSats technology for ADCS. In particular, the survey shows, NanoSat-compatible attitude actuators, miniaturized attitude sensors, GPS receivers and various attitude determination algorithms.

1.5.1 Hardware for CubeSats

In addition to the data presented in [46] further research was carried out by considering some of the recently developed CubeSats. Table 1.1 shows the list of CubeSats whose attitude determination and control system was analysed. The information is available on the *ESA Earth Observation Portal* [2].

GOMX-3	CuPID	CXBN-2
GOMX-4B	CXBN	CUAVA-1
SIMBA	UniCubeSat-GG	HORUS
PICASSO	Aeneas	Pathfinder
EDSN	PSAT2	SpooQySat
VELOX-II	VELOX-1	QBITO
e-st@r-II	TEMPEST-D	ITASat-1
CanX-7	MicroMAS-1	Delfi-C3
RAX	MeznSat—A	OUFTI-1
ARMADILLO	CUBESPEC	DustCube

Table 1.1: *ADCS CubeSats investigated*

Figure 1.4 shows an overview of the type of CubeSats considered, in terms of units, while Figure 1.5 shows percentages of mission types performed. As can be seen, the range considered goes from 1U to 6U. However, more 3U CubeSats were considered.

In Figure 1.5 it can be noticed as the highest percentage of missions is of the technology demonstration type. The reason can be attributed to the low cost of CubeSats, which thus enable the testing and validation of novel technology with inexpensive missions.

In the research carried out, all CubeSats provide some type of control, passive or active. This is highlighted in Figure 1.6 which also shows that most CubeSats use an active control. It should be emphasised, however, that it is possible to find CubeSats in the literature that do not provide a control of any kind.

Figures 1.7 and 1.8 show the percentages of use of the various types of sensors and actuators respectively. The characteristics, functions and typical performance of these components will be discussed in detail in Sections 3.2.5, 3.2.4.

The desired performance and mission requirements heavily affect sensor selection. For example, if tight requirements on pointing accuracy are imposed, then a star sensor would be an appropriate choice. However, almost every CubeSat among those considered equips a magnetometer or a sun sensor. As far as sun sensors are concerned, these are divided into Fine and Coarse, which in turn includes the use of solar panels for measuring the sun vector. Sun sensors and magnetometers are inexpensive, reliable and available in miniaturised technology, which is why they are the most commonly used sensors in CubeSats. Gyroscopes are also widely used as they allow a direct mea-

surement of the angular velocity of the CubeSat. The attitude determination algorithms combine measurements obtained from gyroscopes and other sensors, such as sun sensors and magnetometers, to determine the satellite's attitude under all conditions.

Some of the CubeSats considered are equipped with IR Earth Horizon Sensors, which are mainly used in Earth observation missions.
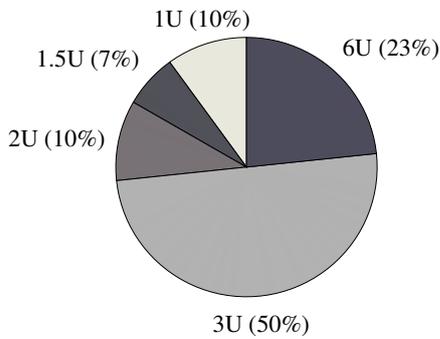
Figure 1.4: *CubeSats investigated*
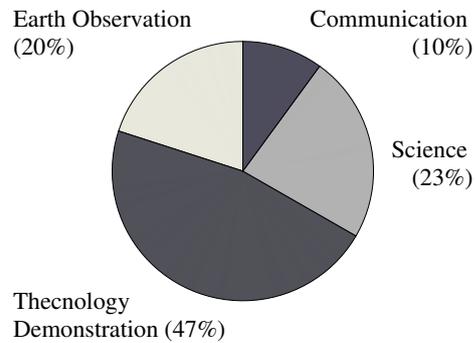
Figure 1.5: *Mission types of CubeSats*
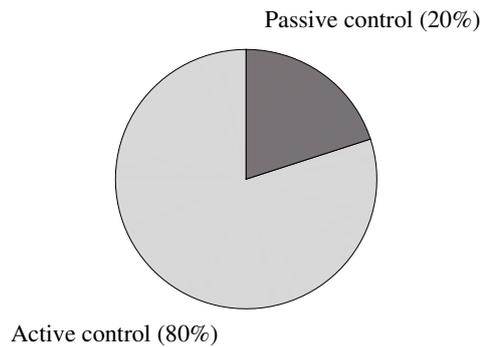
Figure 1.6: *Attitude control types*
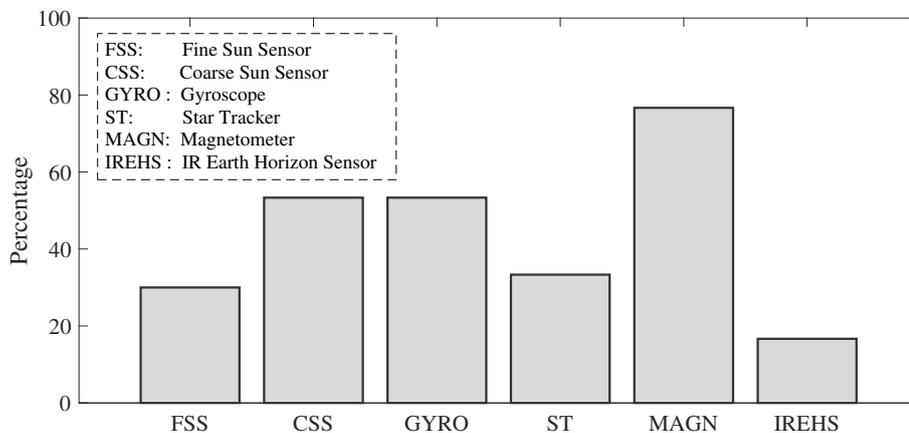
Figure 1.7: *Percentage distribution of the sensors equipped in the CubeSats*
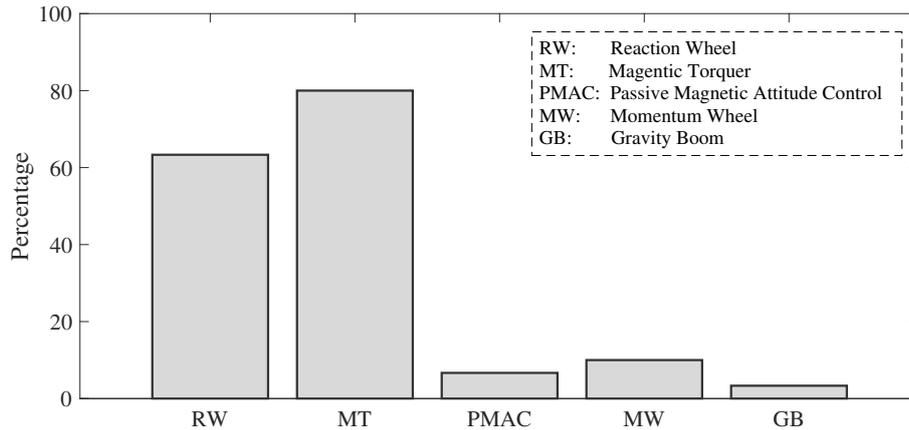
Figure 1.8: *Percentage distribution of the actuators equipped in the CubeSats*

Regarding actuators, since most CubeSats provide active attitude control, most of them are equipped with reaction wheels or magnetic torquers. The percentage of magnetic torquers used is higher than that of reaction wheels also because they are required for wheel desaturation and therefore are present both as main and secondary actuators. In the research conducted, the other types of actuators are less used, as they are related to passive control.

The results of this research, in agreement with [46], show that in general, CubeSats equip more sun sensors, magnetometers and gyroscopes as sensors, and reaction wheels and magnetic torquers as actuators. The reasons for this are to be found in the advantages offered by these suites of sensors and actuators, these will be discussed in the following sections.

1.5.2 Software for CubeSats

Once the actuators and sensors have been selected, it is necessary to put everything together through suitable attitude determination and control algorithms.

Attitude Control Algorithms

In the modern era, improvements in sensors, actuators and processors enable three-axis stabilisation of the spacecraft. Attitude control law theory has been widely studied and advanced. However, in aerospace engineering, reliability tends to be more important than other aspects. For this reason, when designing an ADCS, the engineer tries to use algorithms that have already been tested and verified in practice. The classic example of this is the PID (Proportional Integrative Derivative) controller, which still remains the simplest and most widely used controller in aerospace applications. This aspect is discussed by Xie et al. [47] who review the current status of advanced control methods in spacecraft from an engineer's perspective. On the other hand, it is also discussed how the traditional PID controller is gradually becoming unable to meet the increasingly stringent requirements. This is why various advanced control techniques are more frequently being considered in ADCS system design. Modern control methods differ in type but also in the system model to which they apply. The main and most common modern control techniques are:

- Optimal Control

- Robust Control

- Adaptive Control

- Neural Network Control

Optimal controllers include the Linear Quadratic Regulator (LQR) and the Model Predictive Control (MPC). A hybrid between the optimal and robust controllers is the H infinity controller (H_∞). Whereas, the Model Reference Adaptive Control (MRAC) is one of the best known adaptive control techniques.

In this thesis, the Model Predictive control and H infinity control algorithms are implemented and compared. The characteristics of these control methods will therefore be discussed in depth in Chapter 5.

Attitude Determination Algorithms

Attitude determination generally requires finding three independent quantities, such as a minimum parameterisation of the attitude matrix. The mathematics of attitude determination can be broadly characterised into approaches that use stochastic analysis and approaches that do not. In the literature the term 'estimation' is restricted to approaches that explicitly take stochastic variables into account in the mathematical formulation, such as the Kalman filter or the maximum likelihood approach [31].

It is useful to divide attitude determination techniques into two categories:

Deterministic solutions: These methods need at least two vector measurements obtained at a single time to determine a three-axis attitude. The TRIAD algorithm was the first method for determining the attitude using body reference observations. The limitation of this method is that it can only combine two measurements, which prompted Wahba to formulate a general criterion for attitude determination that would allow to combine two or more measurements. Some common solutions of Wahba's problem are: Davenport's q Method, QUEST, SVD Method, and ESOQ.

Recursive estimation algorithms: These methods use both present and past measurements for determining the attitude. This category includes Kalman filters that use the dynamic model of the system and sensor models to obtain an estimate of the system states that is the most accurate possible using a linear estimator based on present and past measurements. A survey of early attitude estimation methods can be found in [28]. Crassidis et al. in [22] discuss more modern attitude estimation approaches. In the survey it is stated that the Extended Kalman Filter (EKF) remains the most frequently used method among all for many applications. However, the choice of which method to use remains conditioned by the non-linearities of the dynamic model and the desired performances. Some common advanced methods of attitude estimation are: Extended Kalman Filter, Sigma-Point (Unscented) Kalman Filter, and Particle Filters.

In [32] a useful performance analysis of different attitude filtering algorithms (attitude determination algorithms, attitude estimation algorithms, and non linear observers) is discussed. The analysis is applied to a Low Earth Orbit Satellite in terms of accuracy, convergence time, amount of memory, and computation time. The results of this work indicate that deterministic methods are characterized by low computational efforts, less accuracy and amount of memory in comparison with attitude estimation algorithms.

1.6   Dissertation outline

The dissertation is developed according to the following chapters

Chapter 2 describes the necessary theoretical background for implementing the attitude determination and control algorithms. It thus provides a description of the reference systems

involved and the kinematics and dynamics of an Earth observation spacecraft. The theory behind the orbital propagator, developed to simulate position dynamics, is also presented;

Chapter 3 discusses the design of the Attitude Determination and Control System. The description begins with the functional analysis of the system, followed by the design process. After providing an overview of the hardware characteristics, the chapter presents the selected sensor and actuator suites and the chosen determination and control algorithms;

Chapter 4 outlines the algorithms sequence used for attitude determination and then discusses the selected algorithms in detail. The theory behind each algorithm is briefly described. Some preliminary simulation results are presented at the end of each section;

Chapter 5 deals with the implementation of the control algorithms. As in Chapter 4, the theoretical background for each controller is briefly discussed, with particular emphasis on aspects such as stability and other control properties. Next, ideal simulations with no measurement noise are reported to support the testing of the control algorithms. An exception concerns the $H_{3\infty}$ controller, where a non-ideal simulation is also presented to highlight the effectiveness of noise rejection.

Chapter 6 presents the final results. In particular, it first discusses the simulation setup and the sensor and actuator models employed. It then reports the Monte Carlo simulations carried out to validate the control algorithms. Finally, the final configuration of the Attitude Determination and Control System is discussed with appropriate engineering considerations.

Chapter 2

Spacecraft and Orbital Mechanics

To test and validate determination and control algorithms, it is necessary to use a complete orbital simulator that allows reliable and accurate simulations. The impementation of the orbital simulator therefore is part of the overall design of the CubeSat's Attitude Determination and Control System, as it allows the design process to be improved and the fulfilment of requirements to be verified. Since the reference mission is an Earth observation mission, it involves observing the Earth, therefore simulation environment is defined with the Earth itself as the main body.

Figure 2.1 illustrates a spacecraft orbiting in a generic orbit around the Earth. The figure highlights the general framework in which the dynamics of an Earth-Pointing spacecraft take place. Four different reference systems can be identified, which are:

- Earth-Centred Inertial Frame

- Earth-Centred Earth-Fixed Frame

- Local Orbital Frame

- Spacecraft Body Frame
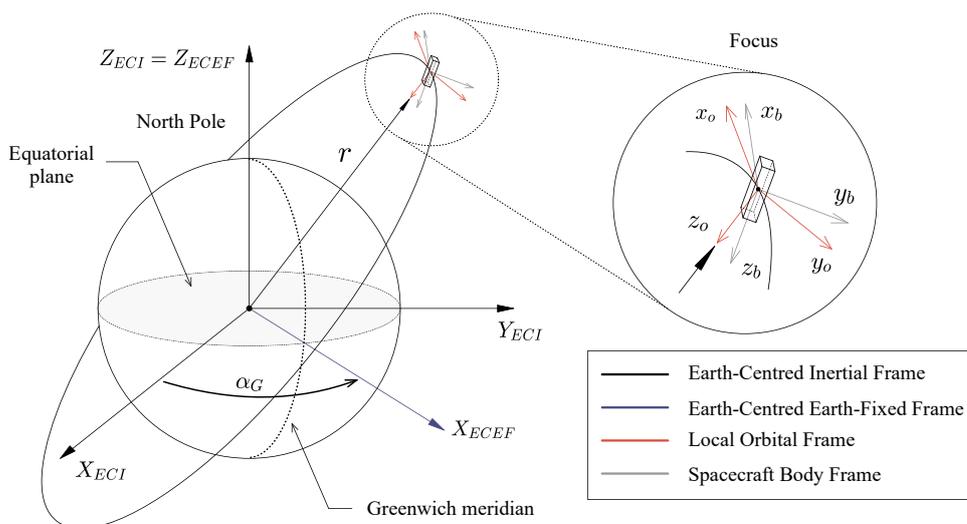
Figure 2.1: *Main reference frames for the description of the dynamics of Earth-Pointing Spacecraft*

The motion of a rigid spacecraft is defined by its position, velocity, attitude, and angular velocity. The first two quantities describe the translational motion of the center of mass of the spacecraft while the latter two quantities the rotational motion of the spacecraft about the center of mass. The definition of these reference frames allows the dynamics and kinematics of the spacecraft in space to be described by utilizing appropriate equations of motion for both position and attitude. This forms the basis of the orbital simulator whose basic functions are precisely the simulation of the orbit and dynamics of the spacecraft.

In addition to this, it is important to have models for the simulation of the Earth's magnetic field and gravitational field as they play an important role in the generation of disturbances acting on the spacecraft and attitude control and determination simulations. Similarly, modeling the position of the sun and moon also contributes to the accuracy of the simulator as it enables the introduction of disturbances and sensor models for attitude measurement.

In this chapter, the theoretical background necessary for the implementation of the orbital simulator, and algorithm development is presented.

2.1 Reference Frames

This section defines the coordinate frames that are used in this thesis work. Each reference Frame $\mathcal{F}_i$ is defined by three orthogonal vectors e_1, e_2, e_3, and the position of its origin O_i.

2.1.1 Earth-Centred Inertial Frame

The *Earth-Centred Inertial Frame* ($\mathcal{F}_{ECI}$) is used to describe the orbital motion of the spacecraft around the centre of the Earth. This coordinate frame is also known as *Geocentric Inertial Frame* (GCI). It is an inertial reference frame therefore Newton's law of motion is valid.

The reference frame is illustrated in Figure 2.2 and is defined as follows:

- The X_{ECI} axis, whose unit vector is $\hat{I}$, lies in the equatorial plane and points toward an inertial reference direction which is the *vernal equinox*;

- The Z_{ECI} axis, whose unit vector is $\hat{K}$, is normal to the equatorial plane and points toward the Earth's North pole;

- The Y_{ECI} axis, whose unit vector is $\hat{J}$, lies in the equatorial plane and it is defined such that $\hat{K} = \hat{I} \times \hat{J}$;

- The origin O_{ECI} is the Earth's centre.

2.1.2 Earth-Centred Earth-Fixed Frame

The Earth-Centred Earth-Fixed Frame ($\mathcal{F}_{ECEF}$) is similar to the ECI frame with $Z_{ECEF} \equiv Z_{ECI}$ and the origin is again the centre of the Earth $O_{ECEF} \equiv O_{ECI}$. The difference is that the ECEF frame rotates together with the Earth with the angular velocity

$$\omega_\oplus = \frac{1 + 365.25}{365.25 \cdot 24h} \frac{2\pi rad}{3600s/h} \simeq 7.29218 \cdot 10^{-5} \ rad/s \tag{2.1}$$

Figure 2.2 shows $\mathcal{F}_{ECI}$ and also $\mathcal{F}_{ECEF}$.
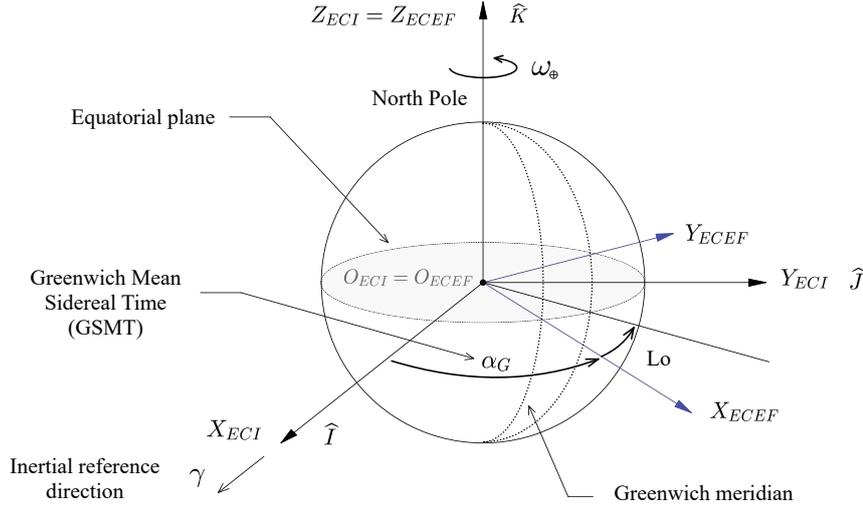
Figure 2.2: *ECI and ECEF reference frames*

The frame is defined by the following characteristics:

- The X_{ECEF} axis lies in the equatorial plane and points toward the Earth's prime meridian which is the *Greenwich meridian*;

- The Z_{ECEF} axis is normal to the equatorial plane and points toward the Earth's North pole;

- The Y_{ECEF} axis lies in the equatorial plane and it complete the right-handed system;

- The origin O_{ECEF} is the Earth's centre.

The α_G angle is known as the *Greenwich Mean Sidereal Time* (GMST) angle and represents the time elapsed since the Greenwich meridian passed through the vernal equinox [23]. The local sidereal time of a site is found by first determining the Greenwich sidereal time α_G and then adding the east longitude Lo of the site. Algorithms can be used to calculate α_G, which involve the use of the so-called Julian day J_D.

The Julian day is the number of days since noon UT on January 1, 4713 BCE. Denoting by J_0 the Julian day at 0 h UT, at any other UT the Julian day is given by

$$JD = J_0 + \frac{UT}{24} \tag{2.2}$$

where

$$UT = h + \frac{minutes}{60} + \frac{seconds}{3600} \tag{2.3}$$

For a given day (d), month (m) and year (y) it is possible to obtain J_0 by the formulas [23]

$$J_0 = 367y - INT\left\{\frac{7}{4}\left[y + INT\left(\frac{m+9}{12}\right)\right]\right\} + INT\left(\frac{275m}{9}\right) + d + 1,721,013.5 \tag{2.4}$$

where $INT(x)$ denotes retaining only the integer portion of x, without rounding.

To calculate α_G with existing algorithms it is necessary to compute T_0 which is the number of Julian days elapsed from the epoch J2000 that has the exact Julian day number $2,451,545.0$

$$T_0 = \frac{J_0 - 2,451,545}{36,525} \tag{2.5}$$

The Greenwich sidereal time α_{G_0} at 0 h UT can be found in terms of this dimensionless time [35]. The following formulas allows to compute α_{G_0} in degrees

$$\alpha_{G_0} = 100.4606184 + 36,000.77004T_0 + 0.000387933T_0^2 - 2.583(10^{-8})T_0^3 \qquad (2.6)$$

The Greenwich sidereal time α_G is found using the relation

$$\alpha_G = \alpha_{G_0} + 360.98564724\frac{UT}{24} \qquad (2.7)$$

where UT is given by Equation 2.3.

2.1.3 Local Orbital Frame

The *local Orbital Frame* ($\mathcal{F}_O$) is used to describe motions with respect to the moving position and direction towards the centre of the Earth of an orbiting body. This reference frame is convenient especially for Earth-pointing spacecraft.

Figure 2.3 shows $\mathcal{F}_O$, the frame is defined by the following characteristics:

- The y_o axis is normal to the orbital plane and points in the opposite direction of the angular momentum vector of the orbit;

- The z_o axis is radial and points from the spacecraft CoM to the Earth's centre;

- The x_o axis is defined such that complete the right-handed system;

- The origin O_O is the centre of mass of the spacecraft (CoM).

For a circular orbit the x_o axis is aligned with the orbital velocity vector. This frame is also known as the *Local-Vertical/Local-Horizontal Frame* (LVLH).
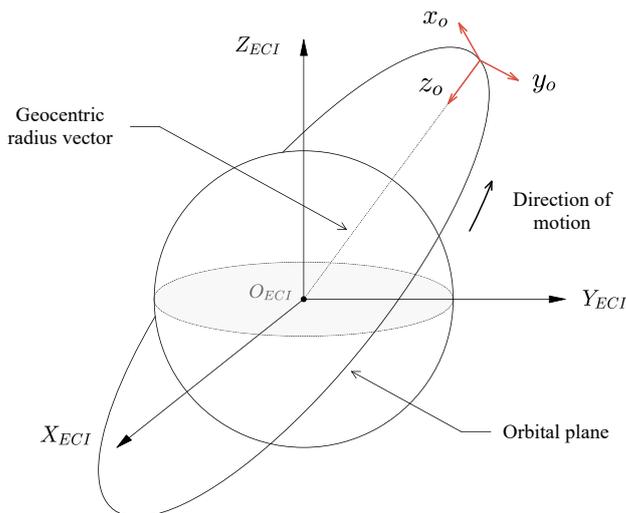
Figure 2.3: *Local Orbital Frame*
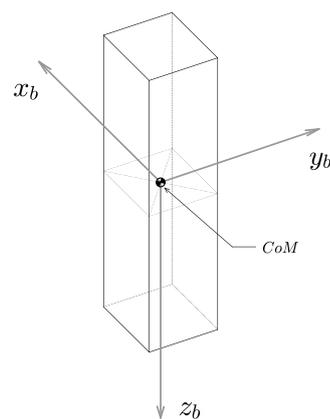
Figure 2.4: *Spacecraft Body Frame*

2.1.4 Spacecraft Body Frame

The *Spacecraft Body Frame* ($\mathcal{F}_B$) is used to describe all rotations of the spacecraft and the attitude with respect to $\mathcal{F}_O$. The frame is illustrated in Figure 2.4 and it is defined as follows

- The z_b axis points in the direction of the face where the observation payload is located;

- The x_b axis nominally points in the direction of the orbital velocity vector;

- The y_b axis is defined such that complete the right-handed system;

- The origin O_B is the centre of mass of the spacecraft (CoM).

In the case of CubeSats, the centre of mass rarely moves from the nominal point, so this frame can be assumed to be fixed.

2.2 Attitude Representations

To describe the attitude of the spacecraft and to determine its evolution it is necessary to introduce an attitude representation and Euler's equation of motion (presented in 2.5.1). The position of the spacecraft body frame $\mathcal{F}_B$ with respect to the local orbital frame $\mathcal{F}_O$, completely define the attitude of the spacecraft. Mathematically this latter can be described using a coordinate transformation matrix which is defined with 3 parameters.

2.2.1 Euler angles

One of the most widely used 3-parameter sets are the Euler angles, which define a sequence of rotations to make the starting reference frame coincide with the body reference frame.

The sequence of rotations is not unique and the rotations are not commutative. The original sequence of rotations proposed by Euler is the so-called 'Proper Euler' also known as 3-1-3. If $\hat{E}_1$, $\hat{E}_2$, $\hat{E}_3$ denote the unit vectors of the starting reference frame and $\hat{e}_1$, $\hat{e}_2$, $\hat{e}_3$ the unit vectors of the body frame, then the situation is as shown in Figure 2.5.

- The first rotation is about third axis of the initial frame $\hat{E}_3$; the rotation angle is called precession angle Ψ;

- The second rotation is about the first axis transformed after the first rotation $\hat{e}_1'$; the rotation angle is called nutation angle Θ;

- The final rotation is about $\hat{e}_3$; the rotation angle is called spin angle Φ.

The three angles represent the attitude of the frame $\mathcal{F}_B$ with respect to the initial frame. It is possible to obtain the transformation matrix, which describes the attitude of the spacecraft, by appropriately multiplying the rotation matrices for the sequence of 3 elementary Euler rotations. The three elementary rotation matrices of the Euler' sequence 3-1-3 are defined as

$$\boldsymbol{A}_3(\Psi) = \begin{bmatrix} \cos\Psi & \sin\Psi & 0 \\ -\sin\Psi & \cos\Psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.8}$$

$$\boldsymbol{A}_1(\Theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\Theta & \sin\Theta \\ 0 & -\sin\Theta & \cos\Theta \end{bmatrix} \tag{2.9}$$

$$\boldsymbol{A}_3(\Phi) = \begin{bmatrix} \cos\Phi & \sin\Phi & 0 \\ -\sin\Phi & \cos\Phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.10}$$
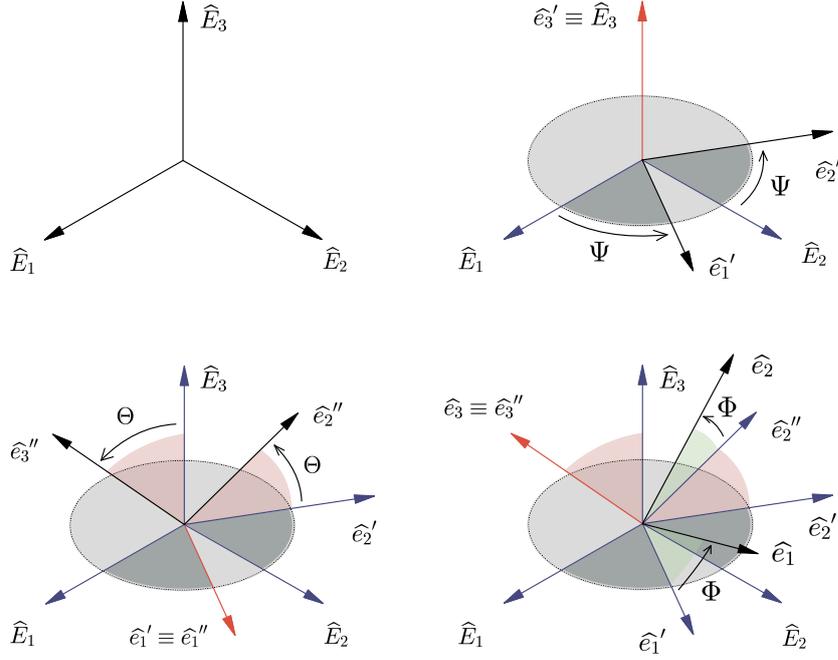
Figure 2.5: *Euler angles and rotations, adapted from [12]*

Therefore the coordinate transformation matrix which define the attitude of the spacecraft with respect to the initial reference frame is

$$A_{3-1-3}(\Phi, \Theta, \Psi) = A_3(\Phi)A_1(\Theta)A_3(\Psi) \tag{2.11}$$

In the case of interest where the attitude of the body frame $\mathcal{F}_B$ has to be described with respect to the local orbital frame $\mathcal{F}_O$ (starting reference frame), the coordinate transformation matrix is denoted by A_O^B.

The drawback of using Euler angles to represent spacecraft attitude lies in the fact that singularities can occur. This means that there may be situations in which the relative attitude between the two reference systems can be described in different ways and not uniquely. When a singularity occurs some angles go to infinity causing serious consequences in the attitude representation.

2.2.2 Quaternions

Using only three parameters, such as Euler angles, to describe the attitude of the spacecraft can lead to a singularity. To overcome this, the spacecraft attitude can be parameterised using 4 parameters known as quaternions.

A quaternion can be seen as a four-component vector

$$\bar{q} = \begin{bmatrix} q_0 \\ \vec{q} \end{bmatrix} \qquad\qquad \bar{q} = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \tag{2.12}$$

Many supplemental equations and explanations can be found for quaternions in [12, 31].

From the Euler eigenaxis rotation theorem, which states that is possible to rotate a fixed frame onto another frame with a simple rotation around an axis $\hat{a}$ fixed in both frames, it is possible to use quaternions to describe changes in attitude. The rotation from the inertial frame to the body

15

is then the rotation of the inertial frame about the unit vector $\hat{a}$ through angle α. The quaternion, whose components are also called Euler parameters, can then be written as

$$\bar{q} = \begin{bmatrix} \cos(\alpha/2) \\ a_1 \sin(\alpha/2) \\ a_2 \sin(\alpha/2) \\ a_3 \sin(\alpha/2) \end{bmatrix} \tag{2.13}$$

It is easy to demonstrate that the coordinate transformation matrix is given by

$$\boldsymbol{A}_O^B = (q_0^2 - \vec{q} \cdot \vec{q})\mathbf{1} + 2\,\vec{q}\,\vec{q}^T - 2q_0\mathbf{S}(\vec{q}) \tag{2.14}$$

where $\mathbf{S}(\vec{q})$ is the cross product matrix equivalent also known as skew matrix

$$\mathbf{S}(\vec{q}) = \begin{bmatrix} 0 & -q_3 & q_2 \\ q_3 & 0 & -q_1 \\ -q_2 & q_1 & 0 \end{bmatrix} \tag{2.15}$$

Equation 2.14 represents the transformation matrix which define the attitude of the spacecraft, it can be expanded as follows

$$\boldsymbol{A}_O^B = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 + q_0q_3) & 2(q_1q_3 - q_0q_2) \\ 2(q_1q_2 - q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 + q_0q_1) \\ 2(q_1q_3 + q_0q_2) & 2(q_2q_3 - q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \tag{2.16}$$

2.3 Coordinate Transformations

Once the reference frames have been defined, it is possible to consider the transformation matrices that allow one reference frame to be rotated on another, bringing them to coincide. Once the rotation matrix has been defined, it is sufficient to consider the inverse of the matrix to perform the opposite rotation.

Rotation from ECI to ECEF

To rotate the ECI reference frame on the ECEF frame, it is necessary to know the rotation angle, which in this case is given by the Greenwich Mean Sidereal Time (GMST) defined in Equation 2.7.

Therefore, the transformation of a position vector from its ECI representation r_I to its ECEF representation r_E follows

$$\bar{r}_E = \boldsymbol{A}_I^E \, \bar{r}_I = \begin{bmatrix} \cos\alpha_G & \sin\alpha_G & 0 \\ -\sin\alpha_G & \cos\alpha_G & 0 \\ 0 & 0 & 1 \end{bmatrix} \bar{r}_I \tag{2.17}$$

Rotation from LVLH to ECI

The rotation matrix from the $\mathcal{F}_O$ frame to the $\mathcal{F}_{ECI}$ frame can be expressed by

$$\boldsymbol{A}_O^I = \begin{bmatrix} \bar{o}_{1I} & \bar{o}_{2I} & \bar{o}_{3I} \end{bmatrix} \tag{2.18}$$

where $\bar{o}_{1I}$, $\bar{o}_{2I}$, and $\bar{o}_{3I}$ are the representations of the unit vectors which define the local orbital frame, as described in 2.1.3, in an inertial frame. These are given by

$$\bar{o}_{3I} = -\frac{\bar{r}_I}{\|\bar{r}_I\|}$$

$$\bar{o}_{2I} = -\frac{\bar{r}_I \times \bar{v}_I}{\|\bar{r}_I \times \bar{v}_I\|} \tag{2.19}$$

$$\bar{o}_{1I} = \bar{o}_{2I} \times \bar{o}_{3I}$$

Rotation from LVLH to Spacecraft Body Frame

To rotate the $\mathcal{F}_O$ frame to the $\mathcal{F}_B$ frame, it is necessary to know the attitude of the spacecraft with respect to the local orbital frame. As described in 2.2.2, the attitude can be represented by quaternions in order to avoid singularities. It follows that the transformation matrix is as shown in Equation 2.16.

2.4 Orbit and Reference Models

This section presents the characteristics of the orbital propagator and the reference models used.

2.4.1 Two-Body Problem

The classical problem of determining the motion of two bodies due solely to their own mutual gravitational attraction is resolved by the *Newton's Law of Universal Gravitation*. In [14, 23] it is shown that by considering an inertial frame, such as the ECI frame represented in Figure 2.2, making some simplifying assumptions, and by applying the Newton's second law, the path of one of the masses relative to the other is a conic section (circle, ellipse, parabola, or hyperbola) whose shape is determined by the eccentricity.

Through straightforward steps can be demonstrated that the vector differential equation of the relative motion for the two-body problem is given by

$$\ddot{\bar{r}} = -\frac{G(M+m)}{r^3}\bar{r} \tag{2.20}$$

where G is the universal gravitational constant, M is the mass of the main body, and m is the mass of the secondary body. In the case of interest, the main body is represented by the Earth and the secondary body by the spacecraft, therefore the mass m is much less than M. Hence it follows

$$G(M+m) \simeq GM \equiv \mu \tag{2.21}$$

where μ is called the gravitational parameter which for the Earth is

$$\mu = 3.986004418 \cdot 10^{14} \ m^3/s^2 \tag{2.22}$$

Then Equation 2.20 can be written again as

$$\ddot{\bar{r}} = -\frac{\mu}{r^3}\bar{r} \tag{2.23}$$

It is possible to define the so-called *state vector* of the spacecraft which comprises its position $\bar{r}$ and velocity $\bar{v}$. The time derivative of the state vector return the velocity and acceleration that is given by Equation 2.23. By integrating the six components of this vector, the instantaneous position and velocity of the spacecraft relative to the inertial frame $\mathcal{F}_{ECI}$ can be calculated to define the characteristics of the orbit. Hence, the six components of the state vector are sufficient to uniquely determine an orbit.
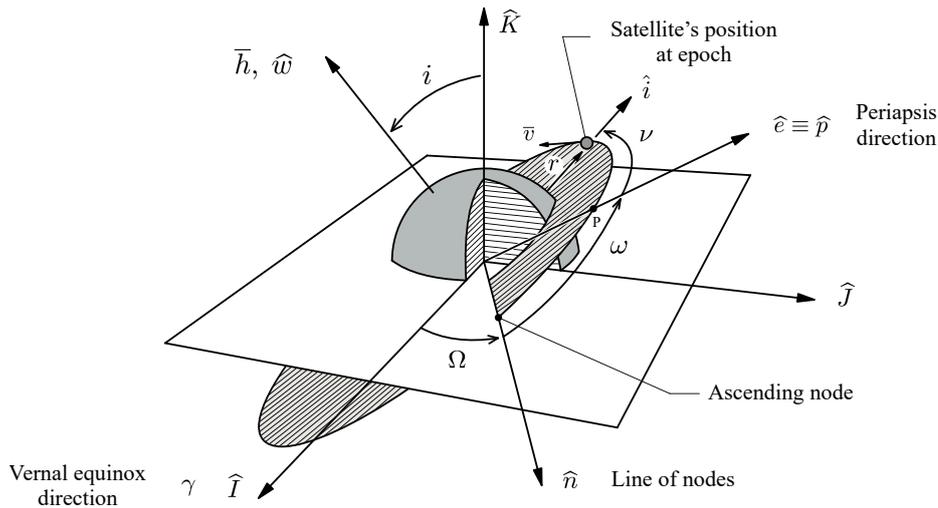
Figure 2.6: *Classical orbital elements, adapted from [14]*

As an alternative to the three components of the position vector $\bar{r}$ and the three components of the velocity vector $\bar{v}$, the six *classical orbital parameters* represented in Figure 2.6 can be used to describe the orbit and the position of the spacecraft. Among these six parameters, five are sufficient to define the shape, size and orientation of the orbit, while the sixth element is required to specify the position of the spacecraft along the orbit. In [14] the classical orbital elements are defined as

a, **semi-major axis**: a constant defining the size of the conic orbit;

e, **eccentricity**: a constant defining the shape of the conic orbit;

i, **inclination**: the angle between the $\hat{K}$ unit vector and the angular momentum vector $\bar{h}$;

Ω, **longitude of the ascending node**: also known as RAAN angle, in the fundamental plane, between the $\hat{I}$ unit vector and the point where the spacecraft crosses through the fundamental plane in a northerly direction measured counterclockwise when viewed from the north side of the fundamental plane;

ω, **argument of periapsis**: the angle, in the plan of the spacecraft's orbit, between the ascending node and the periapsis point, measured in the direction of the satellite's motion;

ν, **true anomaly at epoch**: the angle, in the plane of the spacecraft's orbit, between periapsis and the position of the spacecraft at a particular time, t, called the "epoch".

An orbit that can be described by the six orbital parameters, considered constant over time (except for the true anomaly), is called a Keplerian orbit.

The orbit period is a direct consequence of the Kepler's third law and can be calculated as

$$T = 2\pi\sqrt{\frac{a^3}{\mu}} \tag{2.24}$$

Further considerations and insights are made in [40, 14, 23], here it is only recalled that the conservation of energy and angular momentum applies to Keplerian orbits.

In general, an orbit is characterised by several aspects

- energy of the orbit, which in turn determines the type of conic;

- altitude;

- inclination;

- direction of motion.

In the case of interest, a circular low Earth orbit (LEO) is considered.

2.4.2   Perturbations

Keplerian orbits are the closed-form solutions of the two-body equation of relative motion 2.23. This equation is based on the folliwing simplifying assumptions

- there are only two bodies in space;

- their gravitational fields are spherically symmetric;

- the only source of interaction between the two bodies is the mutual gravitational attraction.

Any effect that causes the motion to deviate from a Keplerian trajectory is known as a perturbation. To account for the perturbations, contributions to the acceleration of the spacecraft are added. It follows that Equation 2.23 becomes

$$\ddot{\bar{r}} = -\frac{\mu}{r^3}\,\bar{r} + \bar{a}_p \tag{2.25}$$

where the vector $\bar{a}_p$ is the net perturbative acceleration from all sources other than the spherically symmetric gravitational attraction between the two bodies. Common perturbations of two-body motion include gravitational interactions with celestial objects like the moon and the sun, a non-spherical central body, solar radiation pressure, and atmospheric drag.

There are two main categories of perturbations techniques, these are described in Table 2.1. In this thesis work, attention was focused on the category of special perturbations, particularly in Cowell's method.

Cowell's method is the simplest and most straight forward of all perturbation methods. The application of the method consists on integrating the equations of motion, including all the perturbations, step-by-step numerically. Having the analytical formulation of the perturbation, the state vector can be integrated by applying a numerical integration scheme.

Type	Description	Examples
Special Perturbations	are techniques which deal with the direct numerical integration of the equations of motion including all necessary perturbing accelerations	Cowell, Encke, Variation elements techniques
General Perturbations	involve an analytic integration of series expansions of the perturbing accelerations	SGP, SGP4, BL

Table 2.1: *Categories of perturbation techniques, from [14]*

In [26] a comparative study is carried out where the performances of different orbital propagators are evaluated, computational time and root mean square errors are used as comparison metrics. The results of the study show that special perturbation techniques are more accurate than general perturbation techniques; however, this accuracy comes at the cost of computational efficiency.

On the other hand, analytic theories (general perturbation techniques) perform poorly but are efficient. As computational capacity becomes increasingly available even on board spacecraft, accuracy was prioritised over computational cost.

2.4.3 Orbit Propagator

The orbit propagator developed for the Attitude Determination and Control System implements a special perturbation solution of orbital motion using a fixed step size Runge-Kutta 4th integration method. The acceleration vector integrated at each step consists of the following contributions

$$\bar{a}_{tot} = \ddot{\bar{r}} = \bar{a}_g + \bar{a}_{sun} + \bar{a}_{srp} + \bar{a}_d + +\bar{a}_{moon} \tag{2.26}$$

where

$\bar{a}_g$ is the acceleration due to gravity

$\bar{a}_{sun}$ is the acceleration due to the Sun

$\bar{a}_{srp}$ is the acceleration due to solar radiation pressure

$\bar{a}_d$ is the acceleration due to atmospheric drag

$\bar{a}_{moon}$ is the acceleration due to the Moon

Figure 2.7 shows the schematic framework of the orbit propagator, implemented in this thesis work, which highlights the inputs required to calculate the different contributions.

Acceleration due to the Gravity Field of the Earth

The gradient of the potential of a central body yield the acceleration, if the body is spherically symmetric then the acceleration is that given in Equation 2.23. In this case, the non-ideal case is considered, therefore in order to derive the non-ideal acceleration, it is necessary to perform the gradient of a potential function that includes perturbations due to a nonspherical Earth. In [40] the procedure for deriving an *aspherical-potential function U* is shown in detail, here only the final results used in the Matlab code implementation are reported.

The *aspherical-potential function* is given by

$$
\begin{aligned}
U = \frac{\mu}{r} \Bigg\{ 1 + \sum_{l=2}^{\infty} C_{l,0} \left(\frac{R_\oplus}{r} \right)^l P_l(\sin\phi) + \\
+ \sum_{l=2}^{\infty} \sum_{m=1}^{l} \left(\frac{R_\oplus}{r} \right)^l P_{l,m}(\sin\phi) \Big[C_{l,m} \cos(m\lambda) + S_{l,m} \sin(m\lambda) \Big] \Bigg\}
\end{aligned}
\tag{2.27}
$$

where λ and ϕ are respectively the east longitude and the latitude of the spacecraft, μ is the gravitational parameter defined in 2.22, $r = \sqrt{x^2 + y^2 + z^2}$ is the geocentric radius of the spacecraft, S and C are unnormalized harmonic coefficients of the geopotential, and P_l are the Legendre polynomials of degree l while $P_{l,m}$ are the Legendre polynomials of degree l and order m.

The analysis of satellite motion allows to empirically determine the C and S coefficients from observations. These harmonic coefficients are available for example in Earth Geopotential Model 96 (EGM96) gravity model which is used in the orbit propagator.
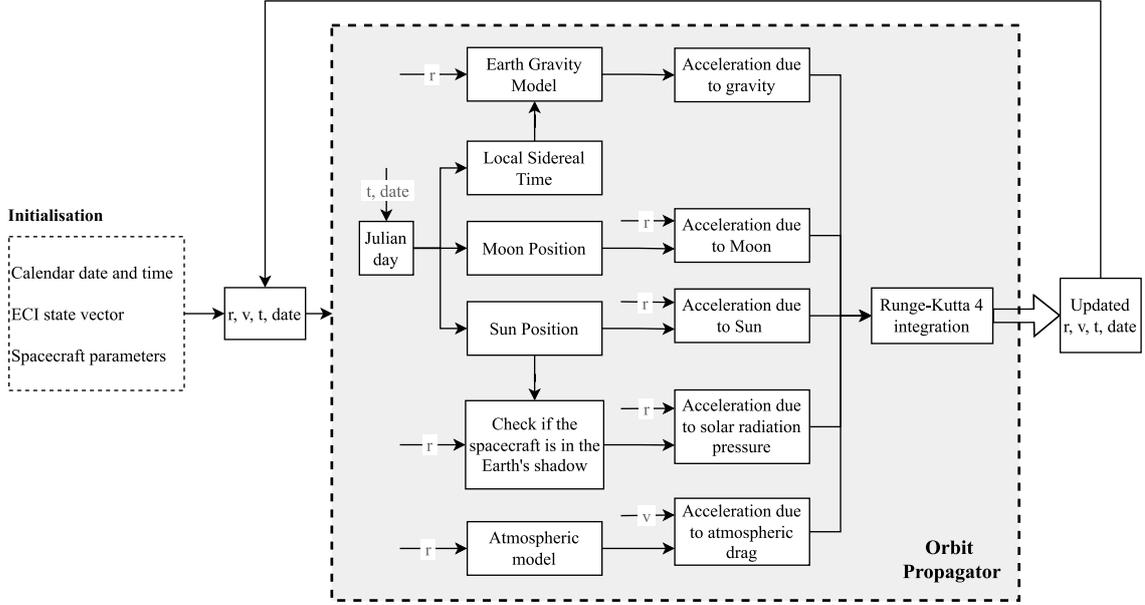
Figure 2.7: *Orbit Propagator Framework*

Coefficients $C_{l,0} = -J_l$ where J_l are the zonal harmonics of the planet. J_2 is the strongest perturbation due to the Earth's shape, in particular reflects the Earth's oblateness. There are other two type of spherical harmonics which are sectoral and tesseral harmonics. These are described in [40], however their effect is less important than that of the J_2 coefficient.

The *conventional Legendre polynomials* are defined as

$$P_l(\sin\phi) = \frac{1}{2^l\, l!}\, \frac{d^l}{d(\sin\phi)^l}\big[\sin^2\phi - 1\big]^l \tag{2.28}$$

while the so-called *associated Legendre functions* are

$$P_{l,m}(\sin\phi) = (1 - \sin^2\phi)^{m/2}\, \frac{d^m}{d(\sin\phi)^m}\, P_l(\sin\phi) \tag{2.29}$$

Acceleration can be derived by taking the gradient of the aspherical-potential function in Equation 2.27.

$$\bar{a}_g = \nabla U \tag{2.30}$$

Breaking out individual components

$$\ddot{x} = \left\{\frac{1}{r}\frac{\partial U}{\partial r} - \frac{z}{r^2\sqrt{x^2 + y^2}}\frac{\partial U}{\partial\phi}\right\}x - \left\{\frac{1}{x^2 + y^2}\frac{\partial U}{\partial\lambda}\right\}y - \frac{\mu x}{r^3} \tag{2.31}$$

$$\ddot{y} = \left\{\frac{1}{r}\frac{\partial U}{\partial r} - \frac{z}{r^2\sqrt{x^2 + y^2}}\frac{\partial U}{\partial\phi}\right\}y - \left\{\frac{1}{x^2 + y^2}\frac{\partial U}{\partial\lambda}\right\}x - \frac{\mu y}{r^3} \tag{2.32}$$

$$\ddot{z} = \frac{1}{r}\frac{\partial U}{\partial r}z + \frac{\sqrt{x^2 + y^2}}{r^2}\frac{\partial U}{\partial\phi} - \frac{\mu z}{r^3} \tag{2.33}$$

where

$$\frac{\partial U}{\partial r} = -\frac{\mu}{r^2}\sum_{l=2}^{\infty}\sum_{m=0}^{l}\left(\frac{R_\oplus}{r}\right)^l(l+1)P_{l,m}(\sin\phi)\Big\{C_{l,m}\cos(m\lambda) + S_{l,m}\sin(m\lambda)\Big\} \tag{2.34}$$

$$\frac{\partial U}{\partial \phi} = \frac{\mu}{r} \sum_{l=2}^{\infty} \sum_{m=0}^{l} \left(\frac{R_\oplus}{r}\right)^l \left\{ P_{l,m+1}(\sin \phi) - m \tan(\phi) P_{l,m}(\sin \phi) \right\}$$
$$\times \left[C_{l,m} \cos(m\lambda) + S_{l,m} \sin(m\lambda) \right] \tag{2.35}$$

$$\frac{\partial U}{\partial \lambda} = \frac{\mu}{r} \sum_{l=2}^{\infty} \sum_{m=0}^{l} \left(\frac{R_\oplus}{r}\right)^l m P_{l,m}(\sin \phi) \left[S_{l,m} \cos(m\lambda) - C_{l,m} \sin(m\lambda) \right] \tag{2.36}$$

Acceleration due to Solar Gravity

The perturbing acceleration due to the presence of a third body is discussed in [23]. In particular, the acceleration due to solar gravity is given by

$$\bar{a}_{sun} = \mu_\odot \left(\frac{\bar{r}_{\odot/s}}{r_{\odot/s}^3} - \frac{\bar{r}_\odot}{r_\odot^3} \right) \tag{2.37}$$

where $\bar{r}_{\odot/s}$ is the vector from the spacecraft to the sun, $\bar{r}_\odot$ is the vector from the Earth to the Sun in $\mathcal{F}_{ECI}$. These vectors are represented in Figure 2.8.

The sun's geocentric position vector $\bar{r}_\odot$ in its apparent motion around the earth can be found following the indications given in [23, 33].

According to The Astronomical Almanac [33] the apparent solar ecliptic longitude is given by the formula

$$\lambda = L + 1.915° \sin M + 0.0200° \sin 2M \tag{2.38}$$

where L and M are expressed both in degree and are respectively the mean longitude and mean anomaly of the sun. These can be calculated by

$$L = 280.459° + 0.985\,647\,36° n \tag{2.39}$$

$$M = 357.529° + 0.985\,600\,23° n \tag{2.40}$$
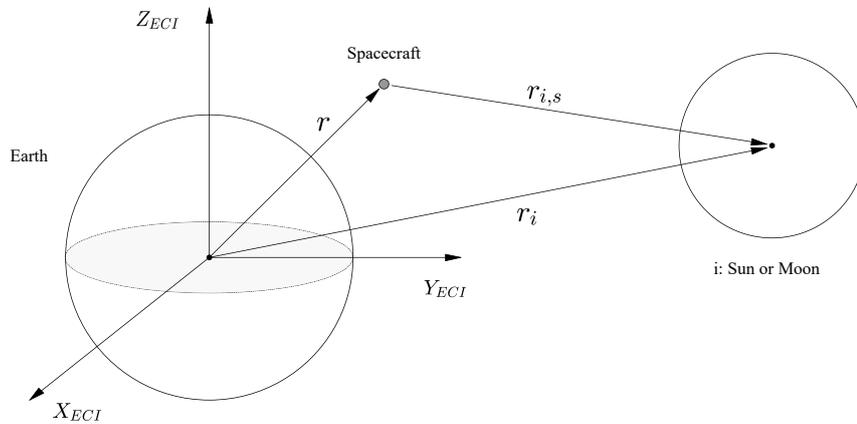
Figure 2.8: *Perturbation of a spacecraft's earth orbit by solar or moon gravity. Adapted from [23]. The scheme is not to scale.*

λ, L, and M are angles in the range $0°$ to $360°$. n is the number of day since $J200$

$$n = JD - 2,451,545.0 \tag{2.41}$$

The Julian day can be calculated as shown in Equations 2.2, 2.3, and 2.4.

The obliquity ε, which is the angle between earth's equatorial plane and the ecliptic plane, can be found in terms of n

$$\varepsilon = 23.439° - 3.56 \cdot 10^{-7} n \tag{2.42}$$

Then, the distance from the Earth to the sun is

$$r_{\odot} = (1.00014 - 0.01671 \cos M - 0.000140 \cos 2M) \; AU \tag{2.43}$$

This is only the norm of the vector, but it is necessary to know also the direction. To do so, the *geocentric ecliptic frame* is introduced. In this frame, the unit vector $\hat{u}$ along the earth-sunline is provided by the solar ecliptic longitude λ given in Equation 2.38

$$\hat{u}_{sun}^{ecliptic} = \cos\lambda \, \hat{I}' + \sin\lambda \, \hat{J}' \tag{2.44}$$

In Equation 2.37 the vector $\bar{r}_{\odot}$ is measured in the $\mathcal{F}_{ECI}$ frame, therefore it is necessary to evaluate $\hat{u}$ in $\mathcal{F}_{ECI}$. The transformation from the geocentric-ecliptic frame to the ECI frame is a clockwise rotation through the obliquity ε around the positive X axis

$$\hat{u}_{sun}^{ECI} = \boldsymbol{A}_1(\varepsilon)^T \hat{u}_{sun}^{ecliptic} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\varepsilon & -\sin\varepsilon \\ 0 & \sin\varepsilon & \cos\varepsilon \end{bmatrix} \begin{bmatrix} \cos\lambda \\ \sin\lambda \\ 0 \end{bmatrix} = \begin{bmatrix} \cos\lambda \\ \cos\varepsilon \, \sin\lambda \\ \sin\varepsilon \, \sin\lambda \end{bmatrix} \tag{2.45}$$

Therefore, the sun vector $\bar{r}_{\odot}$ in ECI frame can be calculated as

$$\bar{r}_{\odot} = r_{\odot} \, \hat{u}_{sun}^{ECI} \tag{2.46}$$

To avoid numerical problem, due to subtractions between two nearly equal numbers, Equation 2.37 is rewritten as suggested in [15]

$$\bar{a}_{sun} = \frac{\mu_{\odot}}{r_{\odot/s}^3} \left[F(q)\bar{r}_{\odot} - \bar{r} \right] \tag{2.47}$$

where $F(q)$ is given by

$$F(q) = \frac{q^2 - 3q + 3}{1 + (1-q)^{3/2}} \, q \tag{2.48}$$

with

$$q = \frac{\bar{r} \cdot (2\bar{r}_{\odot} - \bar{r})}{r_{\odot}^2} \tag{2.49}$$

Acceleration due to Solar Radiation Pressure

The output from the sun contain momentum, which produce an effective pressure on spacecraft surfaces. The perturbing force on the satellite due to the solar radiation pressure is given by

$$F = -\nu \frac{S}{c} \gamma A_{sc} \hat{u}_{\odot,s} \tag{2.50}$$

where

ν is the shadow function, which is equal to 0 when the spacecraft is in the Earth's shadow while is equal to 1 when the spacecraft is in sunlight

$\dfrac{S}{c}$ is the solar radiation pressure, where S is the solar constant and c the speed of light

$$\frac{S}{c} = \frac{1367 \, (N \cdot m/s)/m^2}{2.998 \cdot 10^8 \, m/s} = 4.56 \cdot 10^{-6} \, N/m^2 \tag{2.51}$$

γ is the radiation pressure coefficient, or reflectivity constant, which lies between 0 and 2

A_{sc} is the surface area of the spacecraft normal to the incident radiation

$\hat{u}_{\odot,s}$ is the unit vector pointing from the satellite toward the Sun

Denoting by m the mass of the spacecraft, the acceleration is given by

$$\bar{a}_{srp} = \frac{F}{m} \tag{2.52}$$

During the integration process, the software must determine if the satellite is in Earth shadow or sunlight. As indicated in [40], to determine when a satellite is in the earth's shadow it is possible to use the following procedure. The procedure is based on determining whether or not a line of sight exists between two given vectors. Considering Figure 2.10, assuming that body A is the spacecraft and body B is the sun then it is possible to check whether the spacecraft is in Earth shadow if the sum of θ_1 and θ_2 is greater than the angle θ.

The angle θ between the two position vectors may be found from the dot product operation

$$\theta = \cos^{-1}\left(\frac{\bar{r}_\odot \cdot \bar{r}}{r_\odot \, r}\right) \tag{2.53}$$

while it is easy to see from Figure 2.10 that

$$\theta_1 = \cos^{-1}\left(\frac{R_\oplus}{r}\right) \qquad \qquad \theta_2 = \cos^{-1}\left(\frac{R_\oplus}{r_\odot}\right) \tag{2.54}$$

Therefore, if $\theta_1 + \theta_2 < \theta$ then there is no line of sight which means that the spacecraft is in Earth shadow. In this case there is no perturbing force, hence $\nu = 0$. On the other hand if $\theta_1 + \theta_2 > \theta$ then there is line of sight and so the spacecraft is in sunlight $\nu = 1$.
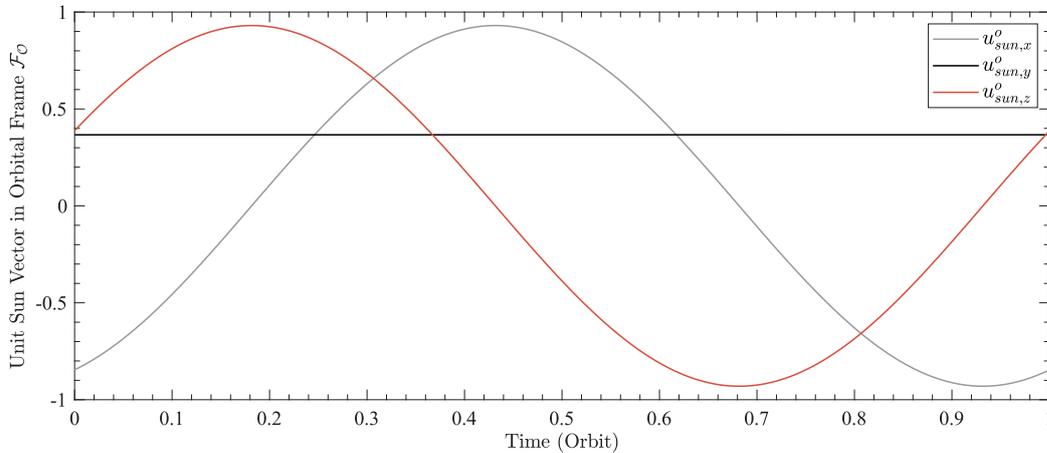
Figure 2.9: *Example of computed unit sun vector in orbital frame measured from an orbit with an altitude of 470 km*
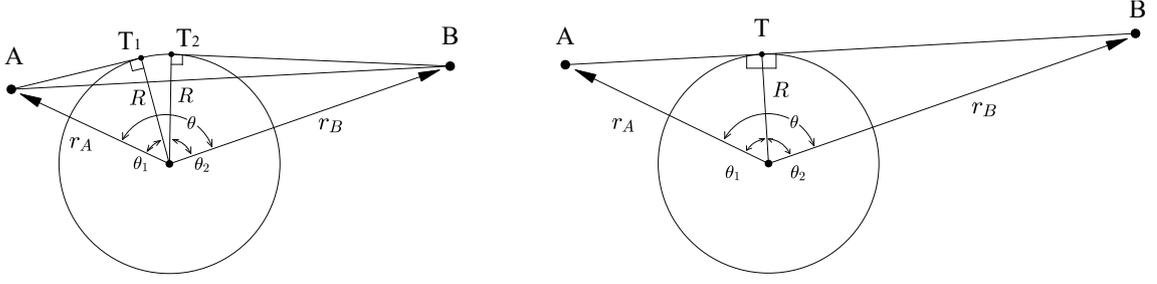
Figure 2.10: *Eclipse model, adapted from [23]*

Acceleration due to Atmospheric Drag

The acceleration of the spacecraft due to atmospheric drag is evaluated using the following expression

$$\bar{a}_d = -\frac{1}{2}\rho v_{rel}\left(\frac{C_D A}{m}\right)\bar{v}_{rel} \tag{2.55}$$

where ρ is the atmospheric density and it is calculated considering the *1976 U.S. Standard atmosphere* [36]. $\bar{v}_{rel}$ is the relative velocity of the spacecraft with respect to the atmosphere. In this implementation it is assumed that the atmosphere rotates with the earth, whose angular velocity is $\omega_\oplus$, it follows that $\bar{v}_{atm} = \bar{\omega}_\oplus \times \bar{r}$. Therefore, the relative velocity is given by

$$\bar{v}_{rel} = \bar{v} - \bar{v}_{atm} \tag{2.56}$$

The negative sign in Equation 2.55 is due to the fact that the drag force on an object acts in the direction opposite to the relative velocity vector.

C_D is the dimensionless drag coefficient, A is the spacecraft frontal area, and m is the spacecraft mass.

Acceleration due to the Moon

The acceleration experienced by the spacecraft due to the Moon follow the same scheme presented for the perturbation of the solar gravity. In this case, the Moon replaces the sun as the third body, the acceleration become

$$\bar{a}_{moon} = \mu_{\mathbb{C}}\left(\frac{\bar{r}_{\mathbb{C}/s}}{r_{\mathbb{C}/s}^3} - \frac{\bar{r}_{\mathbb{C}}}{r_{\mathbb{C}}^3}\right) \tag{2.57}$$

where $\bar{r}_{\mathbb{C}/s}$ is the vector from the spacecraft to the moon, $\bar{r}_{\mathbb{C}}$ is the vector from the Earth to the moon in F_{ECI}.

The unit vector $\hat{u}$ from the center of the Earth to that of the moon is given in the geocentric ecliptic frame by an expression similar to Equation 2.44

$$\hat{u}_{moon}^{ecliptic} = \cos\delta\,\cos\lambda\,\hat{I}' + \cos\delta\,\sin\lambda\,\hat{J}' + \sin\delta\,\hat{K}' \tag{2.58}$$

where λ is the lunar ecliptic longitude, and δ is the lunar ecliptic latitude. The components of $\hat{u}_{moon}^{ecliptic}$ in the geocentric equatorial frame are found as in Equation 2.45

$$\hat{u}_{moon}^{ECI} = \boldsymbol{A}_1(\varepsilon)^T \hat{u}_{moon}^{ecliptic} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\varepsilon & -\sin\varepsilon \\ 0 & \sin\varepsilon & \cos\varepsilon \end{bmatrix} \begin{bmatrix} \cos\delta\,\cos\lambda \\ \cos\delta\,\sin\lambda \\ \sin\delta \end{bmatrix} =$$

$$= \begin{bmatrix} \cos\delta\,\cos\lambda \\ \cos\varepsilon\,\cos\delta\,\sin\lambda - \sin\varepsilon\,\sin\delta \\ \sin\varepsilon\,\cos\delta\,\sin\lambda + \cos\varepsilon\,\sin\delta \end{bmatrix} \tag{2.59}$$

Then, the geocentric equatorial position of the moon is $\bar{r}_m = r_m\,\hat{u}_{moon}^{ECI}$. The distance to the moon r_m can be calculated as

$$r_m = \frac{R_\oplus}{\sin HP} \tag{2.60}$$

where HP is the horizontal parallax. The Astronomical Almanac [33] presents the following formulas for the time variation of lunar ecliptic longitude λ, lunar ecliptic latitude δ, and lunar horizontal parallax HP

$$\lambda = b_0 + c_0 T_0 + \sum_{i=1}^{6} a_i \sin(b_i + c_i T_0)$$

$$\delta = \sum_{i=1}^{4} d_i \sin(e_i + f_i T_0) \tag{2.61}$$

$$HP = g_0 + \sum_{i=1}^{4} \cos(h_i + k_i T_0)$$

where T_0 is the number of Julian day centuries since J2000 for the current Julian day JD given in Equation 2.5. The necessary coefficients are listed in Table 2.2.

	Longitude, λ			Latitude, δ			Horizontal Parallax, HP		
i	a_i	b_i	c_i	d_i	e_i	f_i	g_i	h_i	k_i
0	-	218.32	481267.881	-	-	-	0.9508	-	-
1	6.29	135.0	477198.87	5.13	93.3	483202.03	0.0518	135.0	477198.87
2	-1.27	259.3	-413335.36	0.28	220.2	960400.89	0.0095	259.3	-413335.38
3	0.66	235.7	890534.22	-0.28	318.3	6003.15	0.0078	253.7	890534.22
4	0.21	269.9	954397.74	-0.17	217.6	-407332.21	0.0028	269.9	954397.70
5	-0.19	357.5	35999.05	-	-	-	-	-	-
6	-0.11	106.5	966404.03	-	-	-	-	-	-

Table 2.2: *Coefficients for computing lunar position*

2.4.4 International Geomagnetic Reference Field

As mentioned earlier, in order to verify the performance of the determination algorithms, it is necessary to have a model of the magnetic field to simulate the operation of the magnetometer whose measurements are used to determine the attitude. The Earth's magnetic field model is also used to simulate the detumbling phase in which magnetic torquers are used.

In International System of Units the geomagnetic field is measured in Tesla, however given the very small values it is often used its submultiple nano-Tesla (nT). The Earth's magnetic field is produced by internal sources primarly inside Eart's core and is mainly that of a magnetic dipole.

It is not constant but subject to continuous variations and intensity due to external or local causes. Further insights can be found in [44], however it should be noted that the lack of surface electric currents implies that outside the Earth, the magnetic field B has zero curl

$$\nabla \times \bar{B} = 0 \tag{2.62}$$

therefore the field can be expressed as the gradient of a scalar potential, V

$$\bar{B} = -\nabla V \tag{2.63}$$

The absence of magnetic monopoles implies that the divergence of B is equal to zero which led to the Laplace's equation

$$\nabla^2 V = 0 \tag{2.64}$$

Given the spherical nature of the boundary at the Earth's surface, the Laplace's equation has a solution suitably expressed as a finite series expansion in terms of spherical harmonic coefficients

$$V(r, \theta, \phi, t) = a \sum_{n=1}^{N} \sum_{m=0}^{n} \left(\frac{a}{r}\right)^{n+1} \left[g_n^m(t) \cos m\phi + h_n^m(t) \sin m\phi\right] P_n^m(\cos \theta) \tag{2.65}$$

where a is a reference radius of the Earth; g_n^m and h_n^m are the spherical harmonic coefficients called *Gaussian coefficients*; r, θ, and ϕ are respectively the geocentric distance, east longitude from Greenwich, and coelevation. The $P_n^m(\cos \theta)$ are Schmidt semi-normalized associated Legendre functions of degree n and order m. The parameter N specify the maximum spherical harmonic degree.

To use Equation 2.65 in order to evaluate the magnetic field at any point, the Gaussian coefficients must be known. The model used in this work is the *International Geomagnetic Reference Field (IGRF): the thirteen generation*. As reported in [11], the IGRF is a set of spherical harmonic coefficients which can be input into the mathematical model to describe the large-scale, time-varying portion of Earth's internal magnetic field between epoch 1900 A.D. and the present. The coefficients of this thirteen generation has been obtained from ground observatories, from observations recorded by satellites and magnetic surveys (in [11] it is reported a list of World Data System data centers and services). The IGRF is produced and maintained by an international task force of scientists under the auspices of the International Association of Geomagnetism and Aeronomy (IAGA) Working group V-MOD.

Given the continuous unpredictably variations of the Earth's core field on timescales ranging from months to millions of years the IGRF must be regularly revised to account for temporal changes, typically every 5 years. It follows that Gauss coefficients $g_n^m(t)$ and $h_n^m(t)$ change in time at 5-years epoch intervals. Expressions for the time dependence of these parameters are given in [11].

The IGRF thirteen generation provides a Definitive Geomagnetic Reference Field (DGRF) model for epoch 2015, a Non-Definitive Geomagnetic Reference Field (basically labeled as IGRF) model for epoch 2020, and a predictive IGRF secular variation model for the 5-year time interval 2020 to 2025. To give an example, Figures 2.11a, and 2.11b shows respectively the global map of the IGRF-13 total field magnitude and its predicted secular variations. These Figures* can be found in [11]. While Figure 2.12 presents an example of magnetic field computed with the IGRF model.

---

*Included in the article's Creative Common license http://creativecommons.org/licenses/by/4.0/

(a) Total field

(b) Predicted secular variation in total field
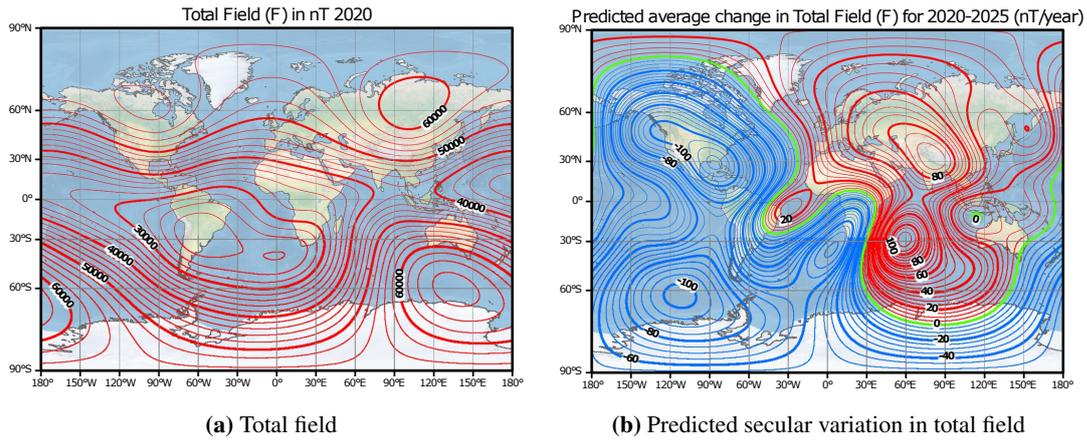
Figure 2.11: *Maps of total field for epoch 2020 (*a*), and maps of predicted annual secular variation in total field (*b*) over 2020 to 2025. Both at the WGS84 ellipsoid surface for epoch 2020. From [11]*
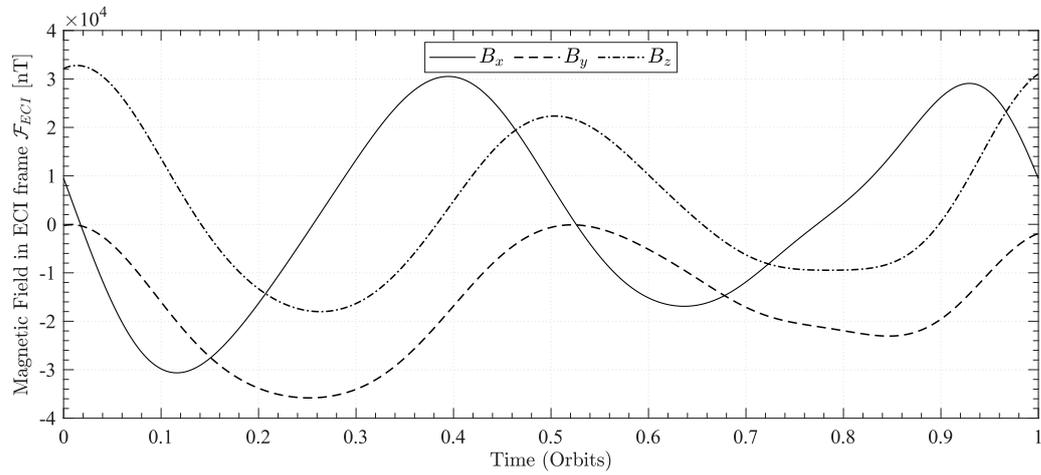
Figure 2.12: *Example of computed magnetic field vector in Earth-Centred Earth-Fixed frame measured from an orbit with an altitude of 470 km and $51.6°$ inclination*

2.5 Spacecraft Dynamics and Kinematics

In this section, the dynamics and kinematics of a spacecraft are presented. The difference is that dynamics involves the consideration of torques, whereas kinematics does not. Finally, the small angle approximation is then considered.

2.5.1 Attitude Dynamics

A detailed discussion in which the equations of dynamics are derived can be found in [31], here only the final result is presented.

Referring to the coordinate frames introduced in 2.1, it is possible to write the equation for the angular momentum of a rigid body with respect to the inertial reference frame F_{ECI}. The equation indicates how the angular momentum varies over time, also taking into account the variations due to the applied torques

$$\dot{\bar{h}}^B + \bar{\omega}_{IB}^B \times \bar{h}^B = \bar{T}^B \tag{2.66}$$

This is the well known *Euler's equation* which characterise the dynamic of a rigid body about its center of mass with respect to an inertial frame. In the equation

$\bar{\omega}_{IB}^B$ is the angular velocity of the spacecraft relative to $\mathcal{F}_{ECI}$ expressed in the body frame $\mathcal{F}_B$

$\bar{h}^B$ is the spacecraft's angular momentum vector measured in the body frame $\mathcal{F}_B$;

$\bar{T}^B$ includes all the external torques acting on the spacecraft with respect to its center of mass expressed in the body frame $\mathcal{F}_B$.

The angular momentum of the spacecraft expressed in $\mathcal{F}_B$ can be written as

$$\bar{h}^B = \boldsymbol{I}\bar{\omega}_{IB}^B \tag{2.67}$$

where $\boldsymbol{I}$ is the inertia matrix of the spacecraft. The assumption of rigid body implies that the time derivative of the inertia matrix is equals to zero, therefore the time derivative of the angular momentum is given by

$$\frac{d\boldsymbol{I}}{dt} = 0 \quad \longrightarrow \quad \dot{\bar{h}}^B = \boldsymbol{I}\dot{\bar{\omega}}_{IB}^B \tag{2.68}$$

By substituting in equation 2.66, angular acceleration can be calculated as

$$\dot{\bar{\omega}}_{IB}^B = \boldsymbol{I}^{-1}(\bar{T}^B - \bar{\omega}_{IB}^B \times \boldsymbol{I}\bar{\omega}_{IB}^B) \tag{2.69}$$

Integrating Equation 2.69 over time allows to obtain the angular velocity of the spacecraft. To evaluate the angular velocity relative to the local orbital frame expressed in $\mathcal{F}_B$ ($\bar{\omega}_{OB}^B$) it is necessary to consider that

$$\bar{\omega}_{IB}^B = \bar{\omega}_{OB}^B + \boldsymbol{A}_O^B \bar{\omega}_{IO}^O \tag{2.70}$$

where $\bar{\omega}_{IO}^O = \begin{bmatrix} 0 & -\omega_0 & 0 \end{bmatrix}$ is the angular velocity of the local orbital frame $\mathcal{F}_O$ relative to the inertial frame $\mathcal{F}_{ECI}$ expressed in $\mathcal{F}_O$. The scalar ω_0 is the angular velocity of the spacecraft about the Earth's center and can be calculated as

$$\omega_0 = \sqrt{\frac{\mu_\oplus}{r^3}} \tag{2.71}$$

2.5.2 Attitude Kinematics

Once the angular velocity $\bar{\omega}_{OB}^B$ is known from Equation 2.70, the attitude of the spacecraft can be calculated. Using quaternions as attitude representation the attitude is given by integrating the following formulas

$$\dot{q}_0 = -\frac{1}{2}\,\bar{\omega}_{OB}^B\,\vec{q} \tag{2.72}$$

$$\dot{\vec{q}} = \frac{1}{2}\,q_0\,\bar{\omega}_{OB}^B + \frac{1}{2}\,\vec{q}\times\bar{\omega}_{OB}^B \tag{2.73}$$

$\vec{q}$ and q_0 are respectively the vector and scalar part of the quaternion, as defined in 2.2.2.

2.5.3 Disturbance Torques

As can be seen in Equation 2.66, external torques acting on the spacecraft contribute to its dynamic. Therefore, it is important to take into account all the toques, due to the space environment, which perturb the spacecraft's dynamic. Some of the perturbations introduced in 2.4.3 generate forces which in turn yield non-negligible disturbance torques. Only four sources of torque matter for the typical Earth-orbiting spacecraft [45]

- gravity-gradient effects;

- magnetic field torques

- impingement by solar radiation

- aerodynamic torques

Gravity-Gradient Torque

The gravity-gradient torque is due to the fact that the spacecraft's center gravity is not aligned with its center of mass, with respect to the local vertical, and the Earth's gravitational force is not constant with distance from the Earth's center. The torque can be computed by summing the contributions of the gravitational forces on the point masses of the spacecraft.

Through straightforward steps, it can be demonstrated that the torque acting on the spacecraft is given by

$$\bar{T}_{gg}^B = 3\,\omega_0^2\,\hat{o}_3^B\times\boldsymbol{I}\hat{o}_3^B \tag{2.74}$$

The torque is expressed in the body frame $\mathcal{F}_B$; $\hat{o}_3^B$ is the unit vector of the z_o axis of the local orbital frame $\mathcal{F}_o$ expressed in the body frame; $\boldsymbol{I}$ is the inertia matrix.

The unit vector $\hat{o}_3^B$ can be easily computed by considering the transformation

$$\hat{o}_3^B = \boldsymbol{A}_O^B\begin{bmatrix}0 & 0 & 1\end{bmatrix} \tag{2.75}$$

Substituting in Equation 2.74 gives the torque as a function of spacecraft attitude

$$\bar{T}_{gg}^B = \begin{bmatrix}3\,\omega_0^2\,(I_z-I_y)A_{23}A_{33}\\3\,\omega_0^2\,(I_x-I_z)A_{13}A_{33}\\3\,\omega_0^2\,(I_y-I_x)A_{13}A_{23}\end{bmatrix} \tag{2.76}$$

where A_{13}, A_{23}, and A_{33} are the components of the attitude matrix defined in Equation 2.16.

Magnetic Field Torque

The magnetic torque is due to the interaction between the Earth's magnetic field and the residual magnetic moment of the spacecraft. These residual moments depends on the spacecraft's size and on the electronic components that can create an equivalent current loop which in turn result in a magnetic dipole. When a spacecraft's residual moment is not aligned with the local magnetic field then it is subjected to a magnetic torque which attempts to align the magnetic dipole with the local field.

The magnetic torque can be computed as

$$\bar{T}_m^B = \bar{m} \times \bar{B}^B \tag{2.77}$$

where $\bar{m}$ is the residual magnetic dipole moment, and $\bar{B}^B$ is the Earth's magnetic field expressed in the body frame.

Solar Radiation Pressure Torque

As mentioned earlier sunlight has momentum, and therefore it produce pressure on those objects it strikes. The magnitude of this disturbance torque depend on the spacecraft surface properties.

The solar radiation pressure torque acting on the spacecraft is given by

$$\bar{T}_{srp}^B = \bar{c}_{ps} \times \bar{F}_{srp}^B \tag{2.78}$$

where $\bar{c}_{ps}$ is the effective location of solar pressure application, also called center of solar pressure, while $\bar{F}_{srp}^B$ is the solar pressure force given in Equation 2.50 but expressed in the body frame.

Aerodynamic Torque

Similarly to the force, for spacecraft in low-Earth orbit the atmospheric drag is a significant source of perturbing torque.

It is easy to demonstrate that the aerodynamic torque is given by

$$\bar{T}_{aer}^B = \bar{c}_{pa} \times \bar{F}_{aer}^B \tag{2.79}$$

where $\bar{c}_{pa}$ is the center of aerodynamic pressure, and $\bar{F}_{aer}^B$ is the aerodynamic force given in Equation 2.55 expressed in the body frame.

Chapter 3

System Design

The Attitude Determination and Control System is the most important subsystem of the spacecraft and therefore requires special attention in its design. The procedure followed in this thesis work for the design is depicted in Figure 3.1. The design steps are derived from the European Cooperation for Space Standardisation (ECSS) documents, in particular ECSS-E-ST-60-30C [5], and from what is recommended in [45].
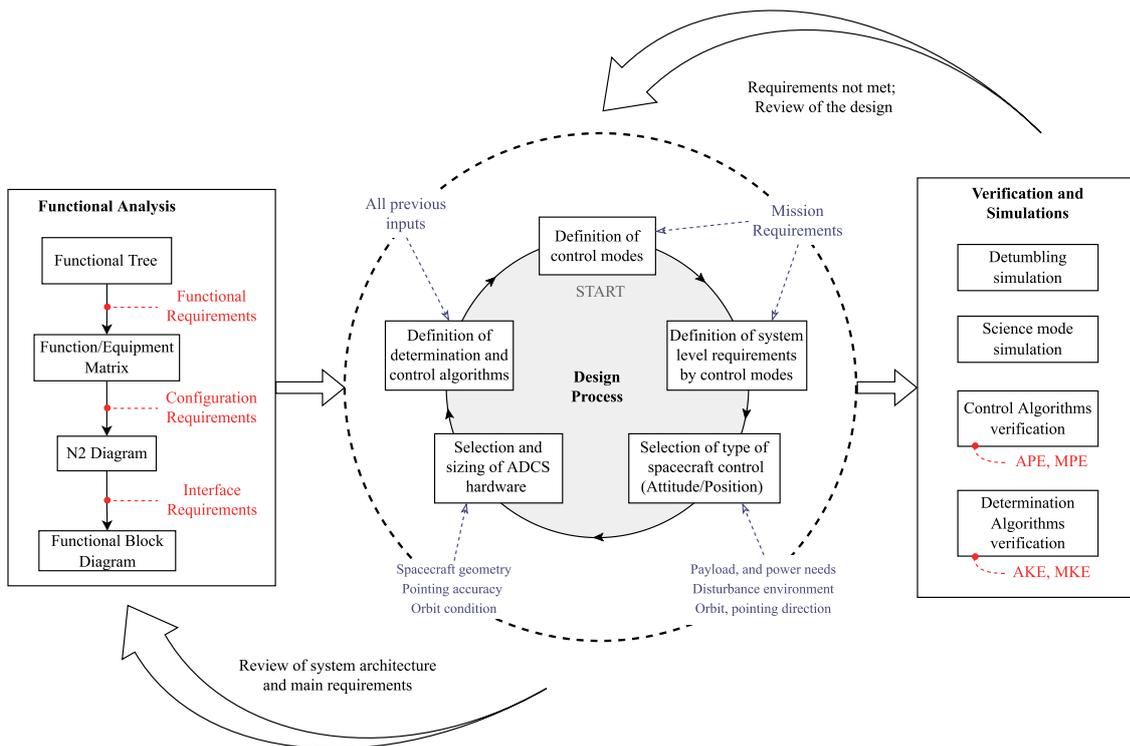
Figure 3.1: *Main design steps followed for the design of the Attitude Determination and Control System*

As can be seen from the figure, design starts with what is called *Functional Analysis*, which allows the identification of functions and the appropriate products to perform them. Once the functions and components of the system have been identified, the iterative design cycle follows in which the main characteristics of the system are defined. Finally, after carrying out an iteration, the verification of the fulfilment of the identified requirements and then the simulation of the implemented algorithms takes place. The result of the verification determines whether the design process is concluded or whether it is necessary to return to the previous steps. The verification may fail, for

example due to incorrect actuator sizing, so it may be necessary to return to the central design cycle. It may also be necessary to return to the functional analysis to review the system architecture or the definition of certain requirements. For this reason, the figure highlights the iterative aspect of the overall design process.

In the following sections, the discussed design steps applied to the 3U CubeSat used for a Earth observation mission are presented individually.

3.1 Functional Analysis

Functional analysis is part of system engineering process activities, and is used to define requirements and a set of functional architectures. Functional Analysis as a fundamental tool of the design process is discussed by a number of references, in this work the functional analysis is considered as presented in [41]. Table 3.1 summarises the steps followed in the functional analysis of the Attitude Determination and Control System (ADCS).

Step	Description	Tools
1. Identification/Decomposition	Definition of the top level function of the system. Breaking down of higher level functions into lower levels	Functional Tree
2. Choice	Determination of the components which can perform the identified functions. Conducting trade-off studies to confirm choices	Function/Equipment Matrix
3. Inner interfacing	Relating the chosen components to each other by highlighting the type of interface	N2 Diagram
4. Outer interfacing	Showing interfaces between the system components and between the system components and other subsystems or systems	Functional Block Diagram

Table 3.1: *Steps of the functional analysis followed for the design of the Attitude Determination and Control System*

3.1.1 Functional Tree

The functional tree allows breaking down the higher-level functions, which derive from the top-level system requirements, into lower-level functions and, finally, identifying the basic functions to be performed by the future product. The functional tree therefore starts with the identification of the primary function of the system. As seen in Section 1.4 ADCS is designed to stabilise, de-tumble, and orient the spacecraft in a given direction. Hence, the primary function is to guarantee the correct attitude of the satellite. In order to perform this function, it is necessary to carry out other, lower-level functions which are

- To determine the angular data

- To generate profile for attitude

- To calculate the attitude manoeuvers

- To generate torques

- To exchange data with other subsystems

- To manage the operative modes

These functions are reported in Figure 3.2 where the top-level of the functional tree is represented. Figures 3.3, 3.4 show the lower-level functions. As can be seen the functional tree generates various branches, starting from the most complex function to the basic functions (grey coloured), i.e. those functions at the bottom of the tree that cannot be split any further.
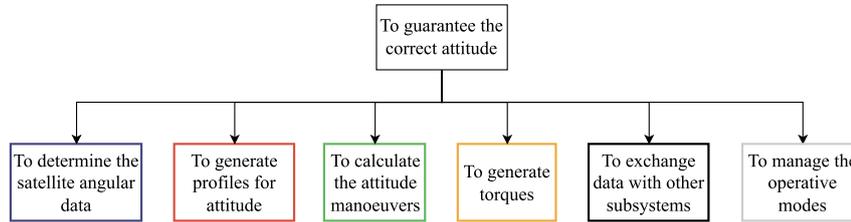
Figure 3.2: *Top level of the functional tree of the Attitude Determination and Control System*
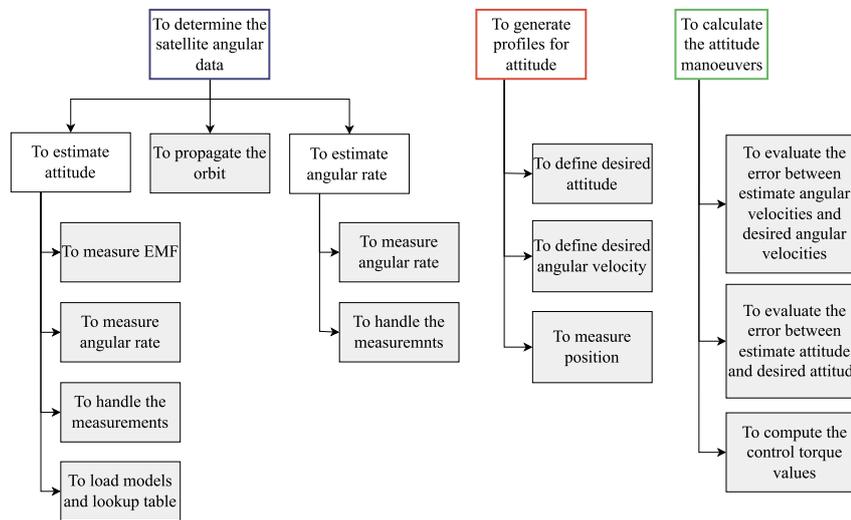
Figure 3.3: *Sub-levels of the functional tree of the Attitude Determination and Control System*
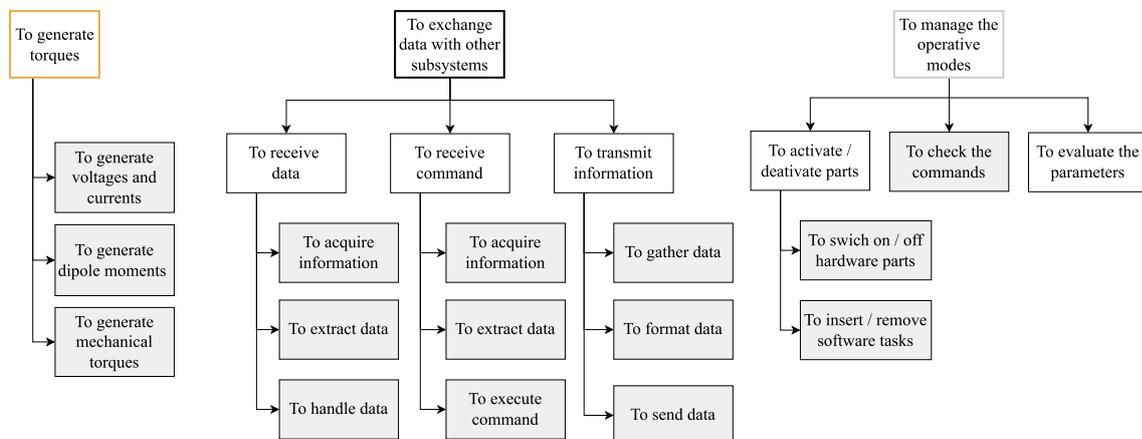
Figure 3.4: *Sub-levels of the functional tree of the Attitude Determination and Control System*

3.1.2 Function/Equipment Matrix

Once the basic functions have been identified, it is possible to choose the components that will perform those functions by means of the function/equipment matrix. The function equipment matrix, represented by Table 3.2, is therefore used to map functions to physical components.

As mentioned earlier, design, as well as functional analysis, are iterative in nature. Table 3.2 essentially shows the final result, i.e. the components identified after a series of iterations and trade-off studies, which will be presented in the following sections. It is noticeable how the chosen components are the most frequently used ones as seen from the results of the state-of-the-art survey in 1.5.1.

Function / Equipment	Magnetometers	Gyroscopes	Sun Sensors	Computer	GNSS Receiver	PWM Circuit	Magnetic Torquers	Reaction Wheels
To propagate the orbit				X				
To measure EMF	X							
To measure angular rate		X						
To handle the measurements data				X				
To measure sun incidence			X					
To load models and lookup table				X				
To define desired attitude				X				
To measure position					X			
To define desired angular velocity				X				
To evaluate the error between estimate and desired angular velocities				X				
To evaluate the error between estimate attitude and desired attitude				X				
To compute the control torque values				X				
To generate voltages and currents						X		
To generate dipole moments							X	
To generate mechanical torques								X
To acquire information				X				
To extract data				X				
To handle data				X				
To excute command				X				
To gather data				X				
To format data				X				
To send data				X				
To check the commands				X				
To swich on/off hardware parts				X				
To insert/remove software tasks				X				

Table 3.2: *Functions / Equipment matrix of the ADCS*

3.1.3 N2 Diagram

Having identified the basic components, the connections between the various system components can be determined. This is achieved by means of the N2 diagram, which highlights the interfaces between all components.
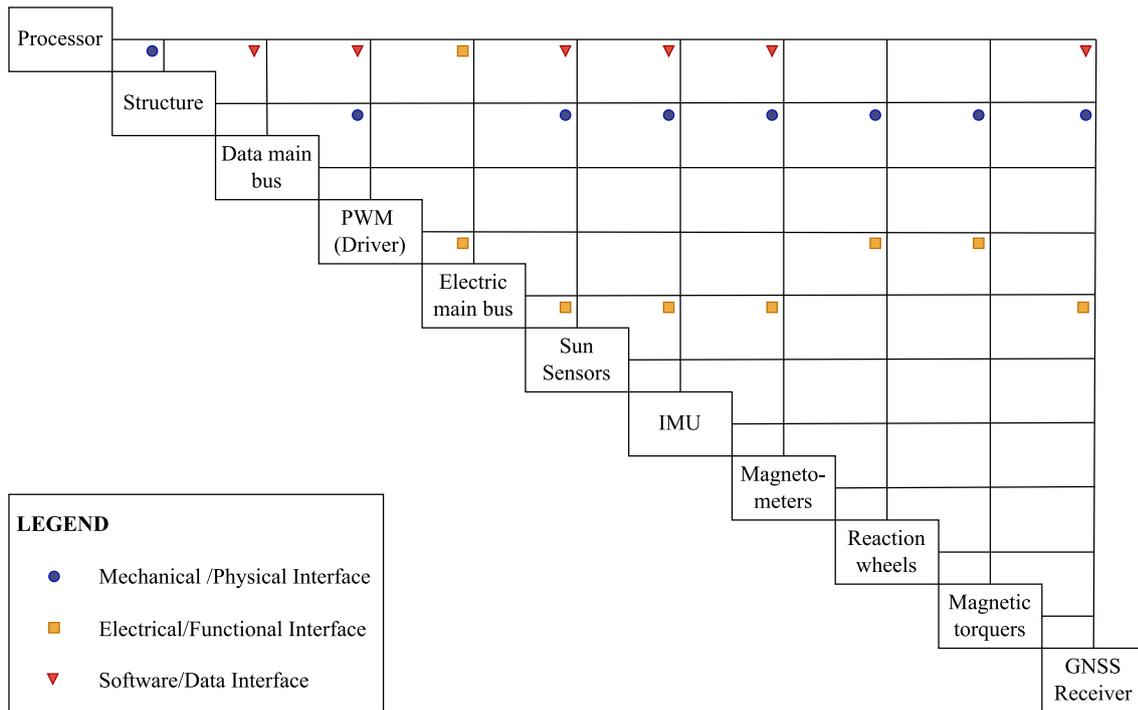
Figure 3.5: *N2 Diagram. Electric main bus represents the interface with the EPS. Data main bus represents the interface with the OBC. PWM (driver) represents the driver with a digital Pulse Width Modulation*

The diagram is shown in Figure 3.5 where there is also a key describing the type of interface between the components.

3.1.4 Functional Block Diagram

A different representation of the same concept, expressed by the N2 diagram, is obtained by means of the functional block diagram represented in Figure 3.6, where blocks are interconnected through point-to-point connections. By indicating the direction of the connection, more information about the data exchange can be obtained. In this case, the type of interconnection that exists between two elements is also indicated in order to provide more details.

In addition to showing the connections between the components of the system itself, connections to other subsystems are also shown. Indeed, the figure shows the connection with the Electrical Power System (EPS) and the On Board Computer (OBC).

3.2 Design Process

Once the functional analysis is complete, the system is defined from a functional point of view and the components that can perform the basic functions have been identified. Furthermore, thanks to the analysis, a first set of requirements, shown in Table C.1, was defined, which the system will have to fulfil.

The next step is the design process, which refers to what is suggested in [45]. Table 3.3 summarises the steps that should be followed, which match what is shown in the central part of the diagram in Figure 3.1.
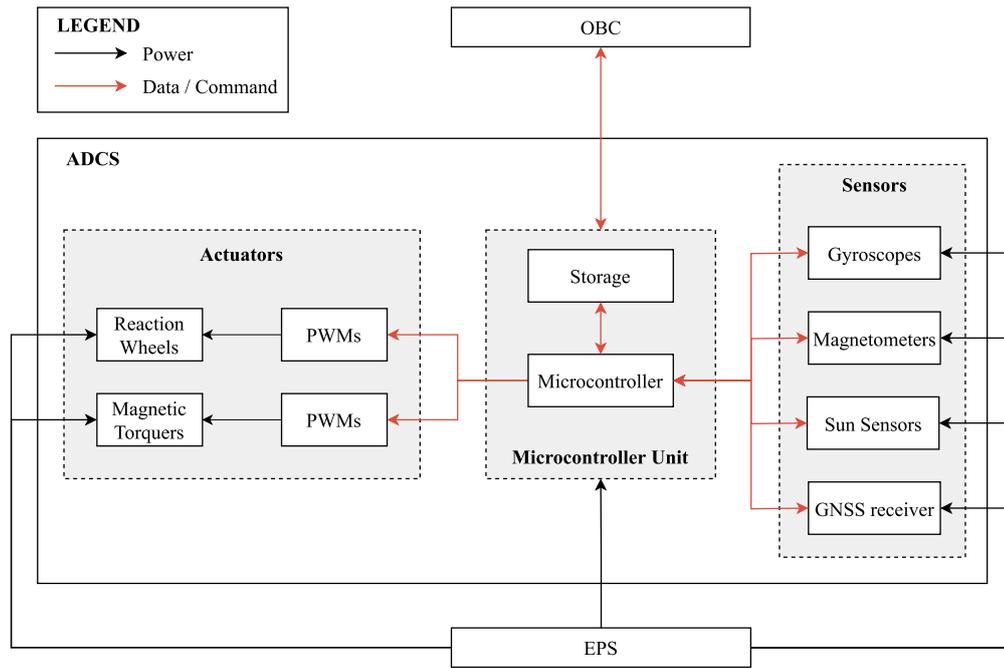
Figure 3.6: *Functional Block Diagram*

	Step	Inputs	Outputs
1.	Define Control Modes and Define or Derive System-Level Requirements by Control Mode	Mission requirements, mission profile, type of insertion for launch vehicle	List of different control modes during mission, requirements and constraints
2.	Quantify Disturbance Environment	Spacecraft geometry, orbit, solar/magnetic models, mission profile	Values for torques from external and internal sources
3.	Select Type of Spacecraft Control by Attitude Control Mode	Payload, thermal & power needs, orbit, pointing and direction, disturbance environment, accuracy requirements	Method for stabilization & control
4.	Select and Size ADCS Hardware	Spacecraft geometry and mass properties, required Accuracy, orbit geometry, mission lifetime, space environment, pointing direction, slew rates. Failure detection and redundancy	Sensor suite, control actuators, data processing avionics
5.	Define Determination and Control Algorithms	Performance considerations, stabilization method, attitude knowledge & control accuracy, slew rates, balanced against system-Level limitations, power & thermal needs, lifetime, jitter sensitivity, spacecraft processor capability	Algorithms and parameters for each determination and control mode, and logic for changing from one mode to another
6.	Iterate and Document	All of above	Refined mission and subsystem requirements. More detailed ADCS design. Subsystem and component specifications

Table 3.3: *Steps in Attitude Determination and Control System design. Adapted from [45]*

As can be seen, the design of the Attitude Determination and Control System starts by defining the control modes of the system to which requirements are subsequently associated. This is followed by the analysis of the disturbance environment, whose purpose is to quantify the disturbance torques acting on the spacecraft. Next, based on the accuracy requirements, objectives and mission needs, the selection of the control type is made. This step is not discussed in the following subsections because, as mentioned earlier, this thesis aims to develop an Active-Attitude Determination and Control System (A-ADCS), where Active indicates that the spacecraft provides active attitude control on the three axes. Next, hardware selection and sizing is carried out. This step is important because if done incorrectly, it can lead to a system that does not meet the requirements. Then follows the step that substantially defines the performance of the system. The definition of the determination and control algorithms takes into consideration various aspects such as robustness, cost, simplicity, and so on. Based on the type of algorithms, the characteristics of the spacecraft from a performance point of view are thus partly defined. The final step concerns the documentation and review of the sub-system and mission requirements.

3.2.1 Mission and System Level Requirements

Before the first design step can be carried out, the requirements and mission profile needs to be analysed.

The 3U CubeSat will be used in an Earth observation mission whose purpose is to take pictures of certain areas of the Earth. The mission involves the use of an optical observation payload for the visible range, which means that images can only be taken when the spacecraft is flying over an area exposed to sunlight. Such payloads introduce requirements for pointing accuracy and stability defined by the desired output quality. In the context of this mission, the requirements are those shown in Table 3.4. These requirements result from the study and design of the payload conducted by the *Payload Research and Development group* of the CubeSat Team Polito.

ID Requirement	Description	
PLD-SYS-ADCS-001	Pointing accuracy	The payload shall be designed to operate with a platform pointing accuracy of **49.11 arcmin** (0.81 deg)
PLD-SYS-ADCS-002	Pointing stability	The payload shall be designed to operate with a payload pointing accuracy stability of **7.41 arcmin/s** (0.12 deg/s)
PLD-SYS-ADCS-003	Ground track velocity	The payload shall be designed to operate with a platform that provides ground track velocity pointing compensation.

Table 3.4: *Technical requirements specification related with ADCS, provided by Payload Research and Development group of the CubeSat team Polito*

Table 3.4 defines requirements concerning the performance of the spacecraft in terms of stability and pointing accuracy. To show how the requirements for ADCS were derived from these, it is necessary to introduce some concepts and tools.

Figure 3.7 represents a schematic framework which allows to define some terminology indicated in the standard [4]. Pointing accuracy is defined by the Absolute Performance Error (APE), which indicates the difference between the target (desired) output of the system and the actual achieved output.

Target	Desired pointing direction
True	Actual pointing direction (mean)
Estimate	Estimate of true (instantaneous)
APE	Absolute Performance Error, pointing accuracy (long-term). It is the attitude error
s	Stability (peak-peak motion). It is the attitude jitter
AKE	Absolute Knowledge error
c	Control error
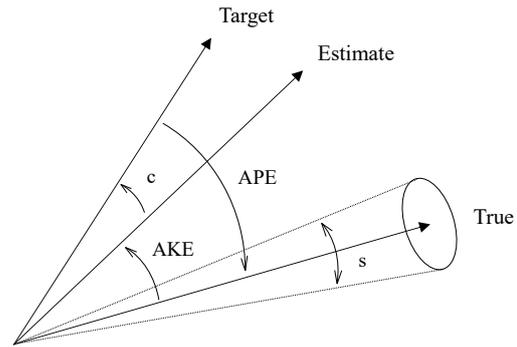
Figure 3.7: *Terms, and definitions according to [4]*

As can be seen from the figure, it can be assumed as a first approximation that the APE is the sum of the control error (c) and the Absolute Knowledge Error (AKE). Indeed, it is easy to verify that if the control error were ideally null then the APE would coincide with the AKE, vice versa if the AKE were null. However, this is an approximation as other sources of error can also contribute to the pointing error such as position estimation error and uncertainties in observation time. The reason for this is that errors on position estimation and timing increase both the AKE error but also errors on guidance, which then identify an incorrect target position.

To derive requirements on APE, control error, AKE, position estimation error, and timing error, the procedure discussed in [45] for the realisation of Mapping and Pointing Budgets is applied. In this context, *pointing* means orienting the spacecraft (camera) to a target having specific geographic coordinates, while, *mapping* is determining the coordinates of the look-point of a camera.

First of all, it is necessary to define which error sources contribute to the mapping and pointing error. Figure 3.8, and Table 3.5 are from [45] and defines the components of pointing and mapping budgets for an Earth-oriented observation mission.

ε	Elevation angle
lat	Latitude of the target
ϕ	Target azimuth relative to the ground track
λ	Earth central angle from the target to the S/C
D	Distance from S/C to the target
R_T	Distance from the Earth's center to the target
R_S	Distance from the Earth's center to the satellite
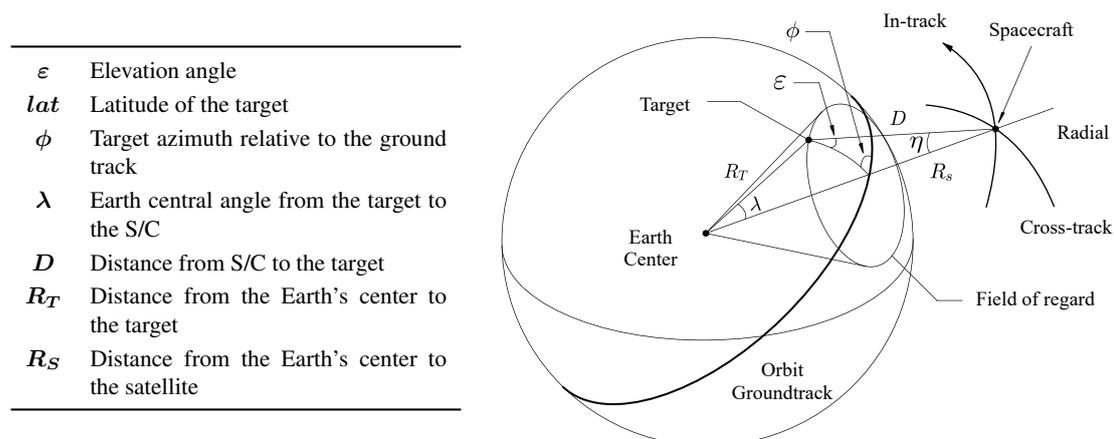
Figure 3.8: *Definition of pointing and mapping error components for an Earth-oriented observation mission. Adapted from [45]*

Spacecraft Position Errors

ΔI	In- or along-track	Displacement along the spacecraft's velocity vector
ΔC	Cross-track	Displacement normal to the spacecraft's orbit plane
ΔR_S	Radial	Displacement toward the center of the Earth (nadir)

Sensing Axis Orientation Errors

$\Delta \eta$	Elevation	Error in angle from nadir to sensing axis
$\Delta \phi$	Azimuth	Error in rotation of the sensing axis about nadir

Other Errors

ΔR_T	Target altitude	Uncertainty in the attitude of the observed object
ΔT	Clock error	Uncertainty in the real observation time (results in) uncertainty in the rotational position of the Earth

Table 3.5: *Sources of pointing and mapping errors. Adapted from [45]*

As mentioned earlier errors on position, attitude, and other types contribute to the pointing error. Attitude errors, called *Sensing Axis Orientation Errors* in Table 3.5, include errors in

- Attitude determination

- Instrument mounting

- Control

This is in agreement with what is defined in Figure 3.7.

The first step is to allocate the budget among the various components. As represented in Figure 3.9 an accuracy goal can be achieved in many ways.
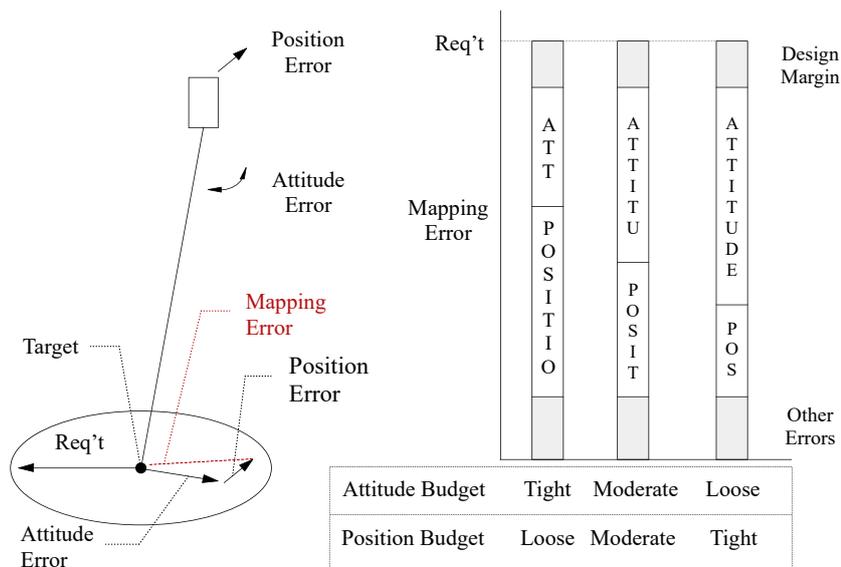
Figure 3.9: *Typical options in error budgets for attitude and position. Adapted from [45]*

For example, it is possible to minimise attitude errors and know the position of the spacecraft inaccurately, alternatively, to know the position precisely and allow more attitude errors. As suggested

in [45] the practical solution is that of allocate the budget equally among all the components.

Once the errors have been allocated, it is necessary to relate them to pointing and mapping errors. The formulas presented in [45] are shown in Table B.1, these provides the basic algebraic information which can be used to transform attitude, position, and other error sources into specific mapping and pointing requirements.

At this point, the procedure followed to realise the mapping and pointing error budget differs from that presented in [45] on a few points. The changes made concern the calculation of the total mapping and pointing error. Whereas in [45] the calculation of the total error is carried out utilizing a root sum square of all partial errors, in this case a root sum square of the partial errors calculated for each category is first carried out, after which a 10% margin is added and finally, the partial RSSs are added linearly. In addition, the value of the total pointing error relative to the allocated attitude errors is doubled. To justify this last operation, it is necessary to consider what has been seen in Figure 3.7. Previously it was mentioned that as a first approximation, it can be assumed that the pointing error is the sum of the control error and the AKE. Therefore, neglecting in a first approximation the influence of position error on attitude determination, and neglecting any errors due to instrument mounting, the total pointing error (related to attitude errors) would only be due to c and AKE. In this case, a possible hypothesis would be to divide the error equally between the two sources. However, to perform a conservative calculation, rather than dividing the error between the two sources, the error is distributed without making any division, i.e. the total error is doubled as if the entire pointing error initially calculated was totally due to a control error or a determination error.

Figure 3.10 summarises the procedure followed in this thesis work to define the requirements of the ADCS from the requirements imposed by the payload.
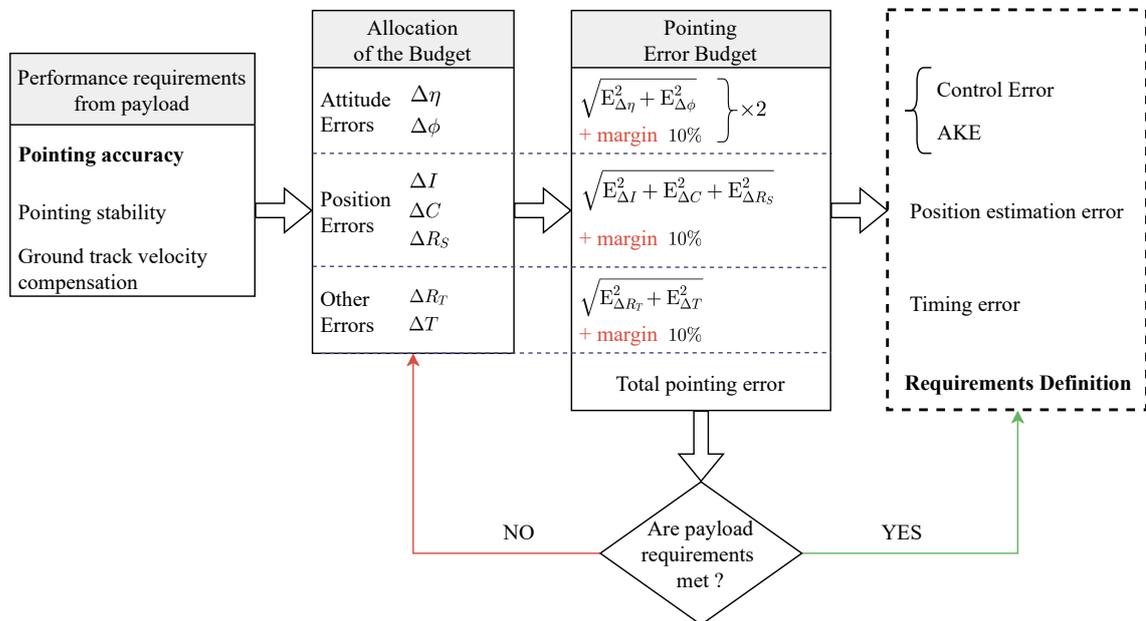
Figure 3.10: *Procedure to define requirements. E_i indicate the error of i, which can represent $\Delta\eta$, $\Delta\phi$, ΔI, ΔC, ΔR_S, ΔR_T, ΔT, calculated with formulas in Table B.1*

The procedure begins by analysing the requirements imposed by the payload, then from these, the errors defined in Table 3.5 are allocated based on analysis, simulations, and experience. Next, the pointing and mapping errors are calculated for each of the categories and then summed up

according to the changes just discussed. At the end of the process, it is checked whether the total pointing error is greater than that imposed by the payload, if so then the allocation must be re-defined, otherwise, the requirements for ADCS can be defined according to the obtained results. Thus, the error allocation for the various categories must be such that the total pointing error does not exceed the requirement imposed by the payload. In this way, if the total pointing error is less than that imposed by the payload, then the allocations and results obtained can be seen as maximum error values that must not be exceeded, i.e. requirements are defined.

The results obtained are shown in Table 3.7, while the data used in the computation are shown in Table 3.6. Two cases are considered, the first one is related to a nadir pointing, while, the second one is related to a pointing where the spacecraft forms an angle of $56°$ with respect to nadir. This angle was considered because it represents a medium-high angle, close to the limiting angle discussed in Section 3.2.5, which may be required during target pointing.

	Elevation Angle. ε (deg)	Spacecraft Altitude. H (km)	Target Latitude. lat (deg)*	Target Azimuth. Φ (deg)*	Azimuth Relative to East. Φ_E (deg)*
Case 1	89.99	470	0	0	90
Case 2	27.00	470	0	0	90

	R_S (km)	D (km)	Nadir Angle. η (deg)	Earth Central Angle. λ (deg)
Case 1	6.848	470	0.0	0.0
Case 2	6.848	926	56.1	6.9

Table 3.6: *Data used in the computation of the mapping and pointing error budget.*
*Values are used for maximum errors

Source	Error In Source	Error Budgets			
		Mapping Error (km)		Pointing Error (deg)	
		Case 1	Case 2	Case 1	Case 2
Attitude Errors:					
Azimuth	0.25 *deg*	0.000	3.351	0.0000	0.2075
Nadir Angle	0.2 *deg*	1.641	7.116	0.2000	0.2000
$RSS \cdot 1.1 \cdot 2 =$				0.4400	0.6340
Position Errors:					
In-Track	0.1 *km*	0.093	0.093	0.0122	0.0035
Cross-Track	0.1 *km*	0.093	0.092	0.0122	0.0062
Radial	0.1 *km*	0.000	0.183	0.0000	0.0051
$RSS \cdot 1.1 =$				0.0190	0.0096
Other Errors:					
Target Altitude	0.01 *km*	0.000	0.020	-	-
S/C Clock	0.5 *sec*	0.233	0.233	0.0283	0.0144
$RSS \cdot 1.1 =$				0.0312	0.0158
Total Sum:		**1.662**	**7.872**	**0.4901**	**0.6594**

Table 3.7: *Mapping and pointing error budget for the 3U CubeSat. Grey cells represent the input for the calculation*

To consider a worst-case scenario a low altitude was used for pointing error. However, Table B.3 shows the results using a higher altitude, it is easy to see that the pointing error decreases, the opposite is true for the mapping error. As can be seen in Table 3.7, the pointing error is higher

with non-nadir pointing. However, the total error still remains below the payload requirement by some margin. Therefore, the requirements shown in Table 3.8 can be defined.

ID Requirement	Description	
ADCS-FUN-415-a	Absolute Performance Error (APE)	The ADCS shall ensure during the operational mission phase an absolute pointing performance of **0.65 *deg***, at **90%** confidence level
ADCS-FUN-415-b	Control Error (c)	The ADCS shall ensure during the operational mission phase a control error less than **0.31 *deg***, at **90%** confidence level
ADCS-FUN-420-c	Absolute Knowledge Error (AKE)	The ADCS shall ensure during the operational mission phase an on-board absolute knowledge performance of **0.31 *deg***, at **90%** confidence level
ADCS-FUN-415-d	Performance Stability Error (PSE)	The ADCS shall ensure during the operational mission phase a pointing accuracy stability of **7.41 arcmin/s** (0.12 deg/s)
ADCS-FUN-416	Orbit Knowledge	The navigation function shall provide the on-board orbit estimation with an accuracy of **100 *m*** for In-Track, Cross-Track, and Radial position
ADCS-FUN-417	Clock error	Uncertainty in the real observation time shall be less than **0.5 *sec***
ADCS-FUN-409	APE on angular velocity	The ADCS shall ensure at the end of the detumbling phase an angular velocity less than **0.2 *deg/sec***

Table 3.8: *Technical requirements specification for the ADCS*

3.2.2 Control Modes

Once the guiding requirements and mission objectives have been defined, the design process can begin by defining the control modes. Table 3.9 shows the control modes defined for this mission. As stated in [5] the ADCS shall define a strategy for the implementation of the mode transitions, for this reason a state diagram represented in Figure 3.11 has been produced in which the mode transition conditions are shown.

Given the nature of the mission (Earth observation), it is necessary to have control modes that allow the spacecraft's observation payload to be oriented towards the Earth so that pictures can be taken. The required pointing can be of different types: nadir pointing or target pointing. The difference lies in the execution of the pointing, the complexity of the control, and the quality of the final observation result. While in the nadir pointing case the spacecraft is stabilised to constantly point at the nadir, in the target pointing case the spacecraft continuously points at the target thus requiring continuous control of both attitude and speed. As shown in Table 3.4 a requirement imposed by the payload is to provide *ground track velocity pointing compensation* i.e. to try to minimise as much as possible the look-point velocity of the spacecraft camera on the ground to obtain high-quality non-blurred images. This requirement can be fulfilled by adopting a target pointing that ideally cancels out the camera's ground look-point speed. In this thesis work, both options have been considered, so functionality is included in the spacecraft design to perform both nadir pointing and target pointing. For this reason, the spacecraft has two control modes relating to the operational mission phase, which are *nadir pointing* and *target pointing*.

To ensure that the mission runs smoothly, it is also necessary to have control modes for other mission phases in addition to the operational one. Indeed, it is necessary to define control modes for the dormant, and idle phases, while a safe mode is needed for the off-nominal phases.

Mode	Description
Dormant	ADCS components are powered off
Sensor acquisition	A IMU and magnetometer evaluate angular velocities from the telemetries B Attitude determination is performed
Detumbling	Controller aims at reducing spacecraft angular velocity under a pre-defined threshold
Target Pointing	The spacecraft is Target pointing
Nadir Pointing	The spacecraft is Nadir pointing
Desaturation	The reaction wheels are dumped by the magnetic torquers
Slew	A desired attitude is required and the ADCS reorients the spacecraft to satisfy the new pointing
Safe	Used in emergencies if regular mode fails. This mode uses less power or fewer components to meet minimal power and thermal needs
Idle	No control is required, system idle power consumption

Table 3.9: *Description of the mission control modes*
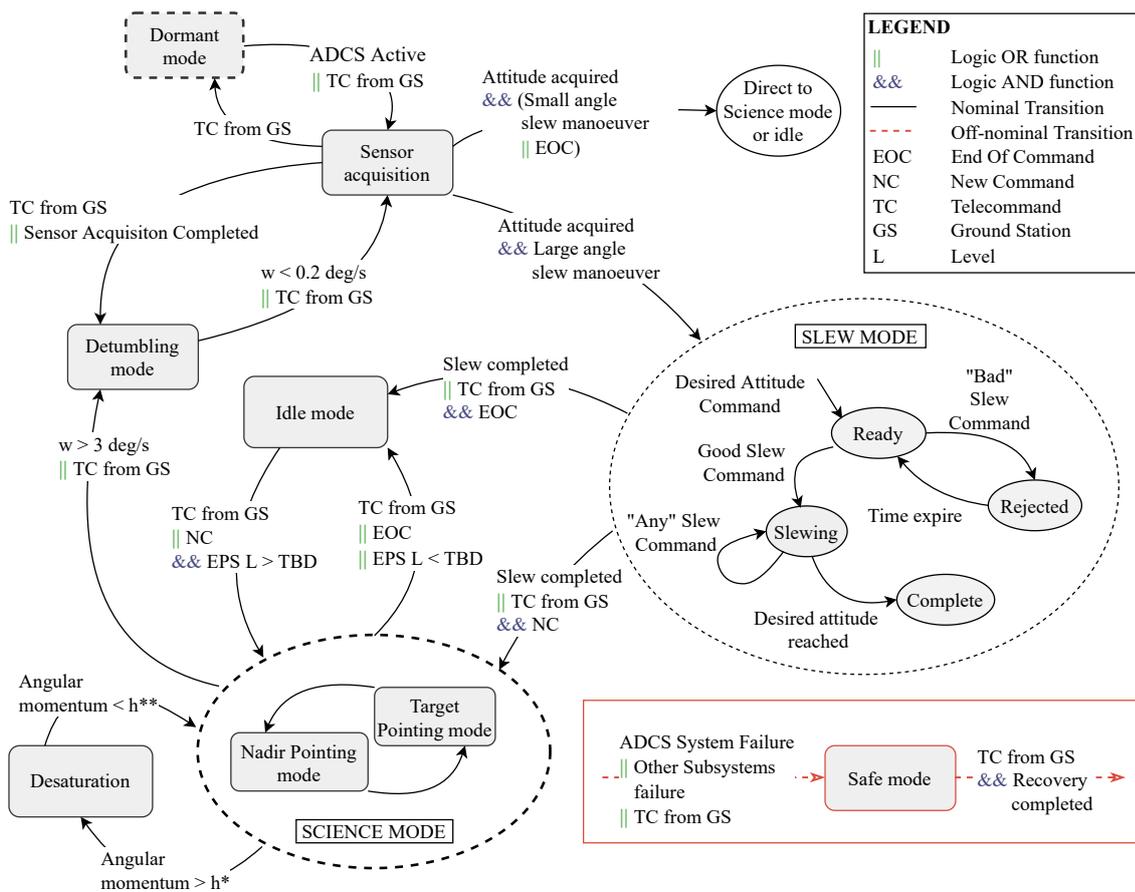
Figure 3.11: *State diagram for the description of the mode transitions strategy*

Another control mode is the one used mainly in the early stages of the mission, namely the *detumbling mode*, used to dampen the angular velocity that the spacecraft has acquired due to the release from the orbital deployer. To detumble the spacecraft angular velocities and magnetic field measurements are required. Therefore, the *Sensor Acquisition mode A* is defined and used before the detumbling phase, where angular velocity and magnetometer data are acquired.

Once the spacecraft has an angular velocity below a certain threshold, attitude acquisition using the estimation algorithms can be performed. Hence, the *Sensor Acquisition mode B* responsible for determining the spacecraft attitude is defined.

At the end of the detumbling phase, the satellite can have an attitude that deviates significantly from that of interest (e.g. nadir pointing), therefore a corrective attitude manoeuvre must be performed. Such a manoeuvre is problematic for the control algorithms used in the nadir pointing mode since, as discussed in the following sections, these algorithms work well around the operating point, i.e. for small angles relative to the orbital system. To overcome this, the slew manoeuver mode is defined, which allows to execute large angle slew manoeuvers.

A further mode is the *Idle mode* activated when the spacecraft is in a phase where it is not required to perform any activity or is simply in a power-saving mode.

Finally, as explained in detail in the following sections, the spacecraft is equipped with reaction wheels. This type of actuator can experience the saturation of the angular momentum of the wheels, therefore a mode is defined in which a control is performed whose objective is to desaturate the wheels using the secondary actuators which are the magnetic torquers.

Figure 3.11 shows the transitions among the various modes of the ADCS. In some cases, transitions can take place by telecommand from the ground station or automatically commanded by the OBC when the conditions specified for the transition are met. At this point in the project, some values defining conditions for transitions have not yet been defined, an example is the power consumption level. The safe mode is accessible from all other modes if anomalies or failures occur in either the ADCS or other subsystems. Return to nominal control modes is possible when system recovery is completed or by telecommand from the ground station.

3.2.3 Satellite Geometry and Disturbance Torques Evaluation

To select and size the right actuators, it is necessary to evaluate all the disturbances which will act on the spacecraft. As mentioned earlier, only four sources of toque matter for the typical Earth-orbiting spacecraft: gravity gradient effects, magnetic field torques, impingement by solar radiation, and aerodynamic torques. Section 2.5.3 presented formulas that allow the evaluation of disturbance torques acting on the spacecraft. In this section, these formulas are used for a preliminary assessment of the disturbances with some simplifications considering the worst-case scenario regarding orbit and spacecraft characteristics for a conservative calculation. Figure 3.12 and Table 3.10 present the spacecraft and orbit characteristics used to perform the calculation. As far as the orbit is concerned, the worst case is at low altitudes where there is a greater intensity of disturbance torques. Particularly the torque due to aerodynamic drag is considerably greater than in a high altitude case. Concerning the spacecraft characteristics, conservative quantities for the various parameters were assumed. It can be seen by comparing the expected values for the geometric and inertial characteristics of the spacecraft shown in Figure 3.12 and the parameter values used for the calculation shown in Table 3.10. The disturbances considered, the type of disturbance (cyclic or secular), the formulas used, and the final result are shown in Table 3.11. For the sake of completeness, the calculation was also repeated for an orbit at an altitude of 700 km. Table B.5 confirms that at a higher altitude, the intensity of the disturbance torques is lower.

Expected linear dimension	0.34	m
Expected volume	0.0034	m^3
Expected loaded mass	4.5	kg
Expected projected area	0.034	m^2
Expected moment of inertia X, Y	0.041	$kg\,m^2$
Expected moment of inertia Z	0.0067	$kg\,m^2$
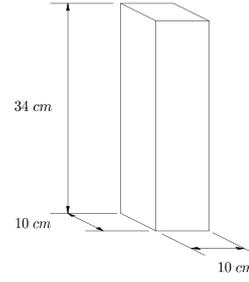
Figure 3.12: *Spacecraft characteristics*

	Value	Unit	Name
Φ	$= 1.366$	W/m^2	Solar constant
c	$= 3 \cdot 10^8$	m/s	Speed of light
A_s	$= 0.034$	m^2	Sunlit surface area
q	$= 0.263$	-	Reflectance factor
φ	$= 45°$	deg	Angle of incidence of the Sun
b_{srp}	$= 0.2$	m	Solar radiation pressure arm $(cp_s - cm)$
ρ	$= 1.13 \cdot 10^{-12}$	kg/m^3	Atmospheric density at 470 km of altitude
C_d	$= 2.2$	-	Drag coefficient
A_r	$= 0.034$	m^2	Ram area
V	$= 7.62 \cdot 10^3$	m/s	Spacecraft's orbital velocity
b_a	$= 0.2$	m	Center of aerodynamic pressure arm $(cp_a - cm)$
D	$= 0.08$	Am^2	Spacecraft's residual dipole moment
B	$= 5.11 \cdot 10^{-5}$	T	Magnetic field strength
M	$= 7.8 \cdot 10^{15}$	Tm^3	Magnetic moment of the Earth
R	$= 6848$	km	Distance between the spacecraft and the Earth's center
μ	$= 398600$	km^3/s^2	Earth's gravitational constant
θ	$= 10$	deg	Angle between the local vertical and the Z principal axis
I_z	$= 0.0095$	kgm^2	Moments of inertia about Z
I_y	$= 0.062$	kgm^2	Moments of inertia about Y
m	$= 6$	kg	Spacecraft's mass

Table 3.10: *Data used in the calculation of the disturbance environment*

Disturbance	Type	Formula	Result
Solar Radiation	Cyclic for Earth-oriented	$T_s = \dfrac{\Phi}{c} A_s (1 + q)(cp_s - cm) \cos \varphi$	$2.76 \cdot 10^{-8}\ Nm$
Atmospheric Drag	Constant for Earth-oriented	$T_a = \dfrac{1}{2} \rho C_d A_r V^2 (cp_a - cm)$	$4.91 \cdot 10^{-7}\ Nm$
Magnetic Field	Cyclic	$T_m = DB = D\left(\dfrac{M}{R^3}\lambda\right)$	$3.96 \cdot 10^{-6}\ Nm$
Gravity Gradient	Constant for Earth-oriented	$T_g = \dfrac{3\mu}{2R^3}\left|I_z - I_y\right| \sin 2\theta$	$3.37 \cdot 10^{-8}\ Nm$

Table 3.11: *Disturbance torques results, data involves in the calculation are shown in Table 3.10. Formulas used for the calculation are described in [45].*

3.2.4 Sensor Selection

The next design step is the selection of ADCS hardware. As seen in Section 1.5, most CubeSats equip similar sensors and actuators. The results obtained from the state-of-the-art study show that the most commonly used sensors are sun sensors, magnetometers, and gyroscopes while the most widely used actuators are reaction wheels and magnetic torquers.

At this point, it is necessary to present the characteristics, advantages and disadvantages of the sensors and actuators used in ADCS in more detail.

To perform attitude determination is required to have suitable sensors that provide a direct or indirect measurement of the attitude. These measurements are then processed using specific algorithms to estimate the satellite's attitude. The results of the state-of-the-art study showed that some types of sensors are more chosen than others. The reason for this lies in the accuracy required to perform a given mission and the monetary budget available to purchase the sensor suite.

Possible sensors that can be used on board the satellite for attitude determination are sun sensors, magnetometers, gyroscopes, star sensors, and horizon sensors. Table 3.12 summarises advantages and disadvantages of these sensors.

Sun sensors are the most widely used sensor type [44]; are visible-light or infrared detectors that measure one or two angles between their mounting base and incident sunlight. Because the Sun is bright and easy identifiable, sun sensors are very reliable and therefore often used for fault detection and recovery. However, they require clear fields of view. There are two macrocategories of sun sensors: Fine Sun Sensors (FSS) and Coarse Sun Sensors (CSS). Coarse sun sensors are simply small solar cells that issue a current roughly proportional to the cosine of the sun angle. Such sensors are so small and inexpensive that it is often possible to use several of them on all faces of the spacecraft and then obtain the sun vector by solving the resulting linear system of equations;

Magnetometers are widely used as spacecraft attitude sensors for a variety of reasons: they are simple, lightweight sensors that measure both the direction and size of the Earth's magnetic field. The magnetometer output, when compared to the Earth's magnetic field known thanks to onboard models (e.g. IGRF), gives information about the spacecraft's attitude; the Earth's magnetic field varies along the orbit, therefore, in the case of spacecraft equipped with magnetic torquers, magnetometers are used to control the intensity and direction of the output torque from the torquers;

Horizon sensors are infrared devices that work by detecting the contrast between the heat of the Earth's atmosphere and the cold of deep space. The main goal is to detect the spacecraft orientation relative to the Earth. They can have high performances but can be expensive, and complex due to rotating parts which reduce their reliability;

Star sensors are the most accurate of attitude sensors, achieving accuracies to the arc-second range. They are sensors with very sensitive light detectors, typically charge-coupled devices (CCD); a star sensor can provide an accurate, standalone estimate of three-axis attitude by comparing an image captured with a detector to an onboard star catalog. They can be heavy, expensive and require more power and memory than other attitude sensors;

Gyroscopes are inertial sensors that measure the speed or angle of rotation from an inertial reference without any knowledge of an external, absolute reference; they are used in combination with other sensors for precision attitude determination. When used with an accurate external reference, gyros can provide smoothing and higher frequency information.

Sensor	Reference	Advantages	Disadvantages
Sun sensor	Sun	Small and inexpensive	Cannot measure during eclipse
		Low power and weight	Disturbed by Earth Albedo
		Almost never fail	Limited accuracy if coarse
		Used in initial acquisition, low power acquisition, fault recovery modes	More sensors are needed to determine the sun vector
Magnetometer	Magnetic field	Low power requirements	Sensitive to biases
		Magnetic field is always available for low altitude spacecraft	Spacecraft must be magnetically clean
		Simple, lightweight, and reliable	Measurements good only in LEO orbit
Horizon Sensor	Earth	High performance	High cost and complexity
		Earth always available for nearby spacecraft	Typically require scan motion; rotating parts reduce reliability
			Sensors must be protected from sun
Star sensor	Stars, including distant planets	High accuracy	Expensive and complex
		Stars available anywhere in sky; lost in space function	Sensors can be heavy and have high volume
		Directly provide 3-axis attitude information	Identification of stars is complex and time consuming
			Sensors must be protected from sun
			Need a lot of memory
Gyroscope	Inertial space	High accuracy	High accuracy for limited time
		Orbit independent	Sensor have rapidly moving parts
		Requires no external reference	Changes in attitude can be measured from some point; Subject to drift

Table 3.12: *Advantages and disadvantages discussed in [45, 44] of sensors used for spacecraft attitude determination*

Sensors selected

By examining the requirements defined in Table 3.8 and the characteristics of the sensors shown in Table 3.12, it is possible to define a candidate sensor suite for the mission of interest.

The identification of candidate sensors was carried out through straightforward trade-offs, without involving any particular tools, to find the best cost-effective solution that meets the mission requirements.

Full 3-axis knowledge requires at least two external non-parallel vector measurements. Magnetometers and coarse sun sensors are the most cost-effective choice in agreement with the above. In addition, the performance requirements for pointing accuracy and attitude determination knowledge are not too tight, so there is no need to consider star sensors. Horizon sensors might be a reasonable alternative, but they are more expensive and space-consuming than sun sensors and magnetometers.

Table 3.13 shows the selected sensor suite and the rationale for the 3U CubeSat. The characteristics and performance of the selected sensors are shown in Appendix D.

	Components	Rationale
6	Sun Sensors	Initially acquire satellite attitude from unknown orientation
	(Osram SFH 2430)	Coarse attitude data
		Provide an external measurement to be used in attitude determination algorithms
1	3-Axis Magnetometer	Needed for more precise magnetic torquing
	(Honeywell HMC2003)	Provide an external measurement to be used in attitude determination algorithms
1	IMU	Determining rotational rates
	(ADIS16446)	Needed to propagate attitude between external measurements

Table 3.13: *Baseline ADCS sensors selected for the 3U CubeSat*

With regard to the sun sensors, photodiodes were chosen, which are basically coarse sun sensors with very small size and low cost. The photodiode is a light sensor consisting of a photocell that, when exposed to light, produces a current that is roughly proportional to the intensity of the light falling on the cell through the cosine of the inclination angle of the light beam with respect to the normal of the photocell.

$$I_i = \begin{cases} I_{max}(\bar{n}_i \cdot \bar{s}) & \bar{n}_i \cdot \bar{s} > 0 \\ 0, & \bar{n}_i \cdot \bar{s} \leqslant 0 \end{cases} \tag{3.1}$$

where $\bar{s}$ is the unit sun vector in the direction from the spacecraft to the Sun, and $\bar{n}_i$ is the normal unit vector to the photocell.

The low cost of these sensors makes it feasible to put several of them on each face of the CubeSat. If a second photodiode is put on the opposite side of the spacecraft his normal would be $\bar{n}_{-i} = -\bar{n}_i$, and the output current

$$I_{-i} = \begin{cases} I_{max}(\bar{n}_{-i} \cdot \bar{s}) = -I_{max}(\bar{n}_i \cdot \bar{s}) & \bar{n}_{-i} \cdot \bar{s} = -\bar{n}_i \cdot \bar{s} > 0 \\ 0, & \bar{n}_{-i} \cdot \bar{s} = -\bar{n}_i \cdot \bar{s} \leqslant 0 \end{cases} \tag{3.2}$$

Therefore, differencing the two outputs gives

$$I_i - I_{-i} = I_{max}(\bar{n}_i \cdot \bar{s}) \ \text{ for all } \ \bar{n}_i \cdot \bar{s} \tag{3.3}$$

By equipping the satellite with six sun sensors, one for each face with normals $\pm n_i \pm n_j \pm n_k$, the direction of the sun vector can be estimated by solving the following system of linear equations as suggested in [31]

$$\begin{bmatrix} I_i - I_{-i} \\ I_j - I_{-j} \\ I_k - I_{-k} \end{bmatrix} = I_{max} \begin{bmatrix} \bar{n}_i \cdot \bar{s} \\ \bar{n}_j \cdot \bar{s} \\ \bar{n}_k \cdot \bar{s} \end{bmatrix} = I_{max} \begin{bmatrix} \bar{n}_i^T \\ \bar{n}_j^T \\ \bar{n}_k^T \end{bmatrix} \bar{s} \tag{3.4}$$

Therefore, the sun unit vector can be computed as

$$\bar{s} = \frac{1}{I_{max}} \begin{bmatrix} \bar{n}_i^T \\ \bar{n}_j^T \\ \bar{n}_k^T \end{bmatrix}^{-1} \begin{bmatrix} I_i - I_{-i} \\ I_j - I_{-j} \\ I_k - I_{-k} \end{bmatrix} \tag{3.5}$$

More photodiodes can be placed on each face of the satellite for redundancy reasons and to obtain a more robust result.

3.2.5 Actuator Selection and Sizing

Actuator selection depends heavily on the pointing accuracy requirements imposed by the mission. Other factors which affect the choice are the geometry of the spacecraft, the orbit which in turn defines the disturbance environment, and any need for slew manoeuvres with specific slew rates.

Several types of actuators can be considered for controlling a CubeSat. One widely used class of actuators is momentum exchange devices that conserve angular momentum in the spacecraft. These devices are wheels which can be used to add stability against disturbance torques, to absorb cyclic torques, to provide a variable momentum, and to transfer momentum to the spacecraft body for the execution of slewing manoeuvres [44]. Wheel control provide smooth changes in torque, allowing very accurate pointing of spacecraft. Within the context of CubeSats, two types of momentum exchange devices can be used

Reaction wheel is a flywheel with the axis fixed to the spacecraft designed to operate at zero bias. It is basically a torque motor with high-inertia rotor. Reaction wheels can spin in any direction and provide one axis of control for each wheel. In practice, a reaction wheel typically is driven by a brushless DC motor which can drive the wheel to spin at a speed up to a certain limit; For this reason, the wheels can experience the so-called angular momentum saturation, which causes the wheel to be unable to store additional angular momentum. Suitable secondary actuators must be provided on board the spacecraft to carry out the desaturation of the wheels.

Momentum wheel is a flywheel designed to operate at a nonzero, or biased, momentum. Therefore, are wheels with a nominal spin rate above zero to provide a nearly constant angular momentum which in turn provide gyroscopic stiffness to two axes.

Another class is that of external torque actuators, which change the angular momentum of the spacecraft when they are activated.

Magnetic torquers create a magnetic dipole moment which in turn creates a torque. With three orthogonal torques it is possible to generate a torque in any direction. A common use of the magnetic torquers is to desaturate momentum-exchange devices.

Thrusters produce a force by expelling propellant in the opposite direction. For this reason, they can have a dual use: attitude control and orbit control. They provide torques proportional to their moment arm therefore the magnitude of the generated torque depends on how thrusters are mounted on the spacecraft. When mounted to maximize torque authority, thrusters have the advantage of being able to provide lagre control torques at any time in the orbit [45]. They can also be used to desaturate reaction wheels.

Actuators Selected

Given the pointing accuracy requirement to be met, reaction wheels were chosen as primary actuators for the 3U CubeSat. Three orthogonal magnetic torquers will be used for momentum dumping, and to perform the detumbling of the spacecraft.

Table 3.14 shows the selected actuator suite and the rationale for the 3U CubeSat.

The choice to equip the CubeSat with four reaction wheels results from a trade-off study between different possible configurations and number of reaction wheels. Figure 3.13 represents the reaction wheel configurations compared.

	Components	Rationale
4	Reaction Wheels	3-axis stabilization
		Storing momentum
		Slew manoeuvre
		Smooth changes in torque
3	Magnetic Torquers	Wheels desaturation
		Satellite detumbling

Table 3.14: *Baseline ADCS actuators selected for the 3U CubeSat*

(a) Pyramidal

(b) Tetrahedral

(c) NASA 3+1

(d) Three orthogonal wheels
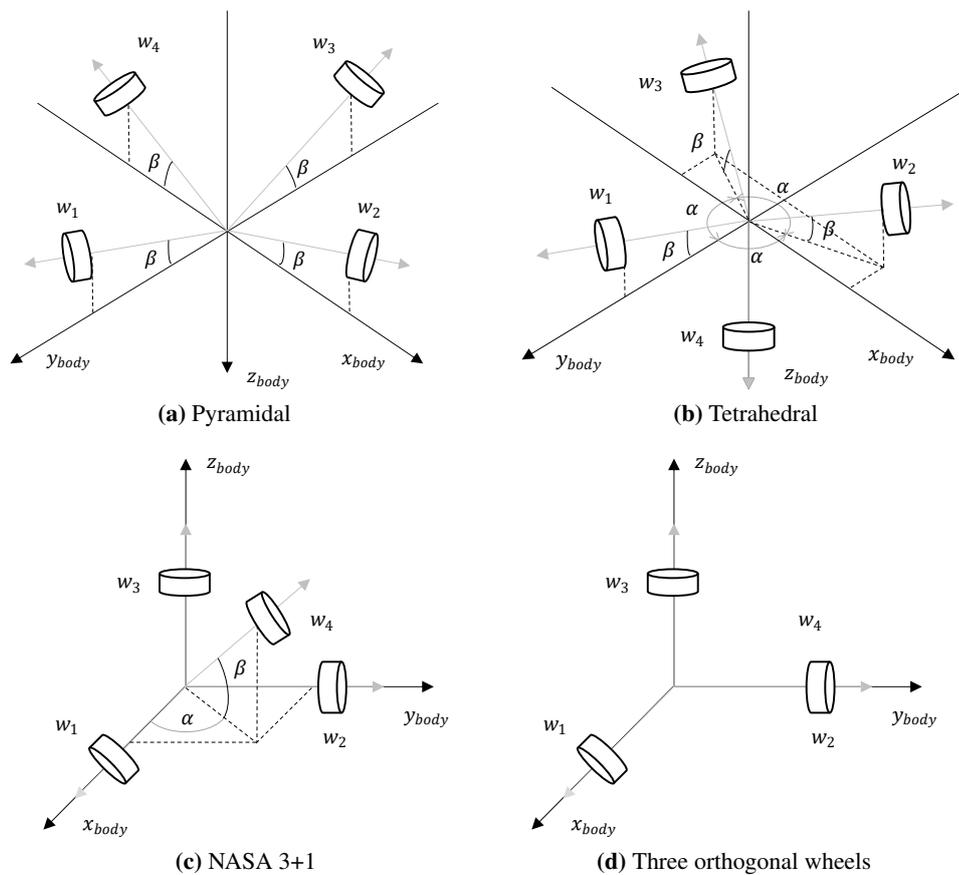
Figure 3.13: *Types of reaction wheel configurations for a spacecraft. The reference frame represented is the spacecraft body frame $\mathcal{F}_B$*

As reported in [31] the most common layout of a pyramid configuration of reaction wheels for Earth-pointing spacecraft is that in which the wheels assume a preferential direction along the y_b body axis because a constant rate along the pitch axis is required to point the instruments toward the Earth's surface. However, placing a cluster of reaction wheels in a pyramid configuration with this layout within a CubeSat results in inefficient utilisation of the limited available space. For this reason, the pyramid configuration considered is that shown in Figure 3.13a. The same argument justifies the tetrahedral configuration in Figure 3.13b.

The trade-off study was carried out according to the Analytic Hierarchy Process (AHP) scheme recommended in the Nasa System Engineering Handbook [27].

The attributes used to compare the configurations are

- *Power* required by the entire configuration to perform a given manoeuvre;

- *Cost* of the configuration;

- The *dimensions* occupied by the set of wheels;

- The *weight* of the configuration;

- The *probability* that the reaction wheels will *not saturate*;

- The *reliability* of the configuration, also intended as the ability to operate after a failure.

The results are shown in Table 3.15 where are also summarised the scores for the various attributes obtained by each configuration and the weight associated with each attribute. Figure 3.14 shows the result graphically. As can be seen, the winning configuration is the pyramidal one, although without a clear difference in the final score. By analysing the partial scores, it can be seen that the tetrahedral and pyramidal configurations often obtain the same scores except in certain attributes.

Report	Power		Cost		Dimension		Weight		Non-saturation probability of wheels		Reliability		Total Score	Rank
	Scores	Weight	Scores	Weight	Scores	Weight	Scores	Weight	Scores	Weight	Scores	Weight		
Pyramidal	0,17		0,17		0,15		0,16		0,56		0,38		0,2827	1
Tetrahedral	0,17		0,17		0,18		0,16		0,28		0,38		0,2556	3
3 ORWs	0,33	0,17	0,50	0,11	0,45	0,14	0,50	0,10	0,08	0,11	0,05	0,35	0,2570	2
NASA 3+1	0,33		0,17		0,22		0,16		0,08		0,20		0,2045	4

Table 3.15: *Results of the trade-off study using the AHP tool. The partial scores for the different attributes for each configuration are shown. The rank coloumn represents the final scores. ORW stands for Orthogonal Reaction Wheels*
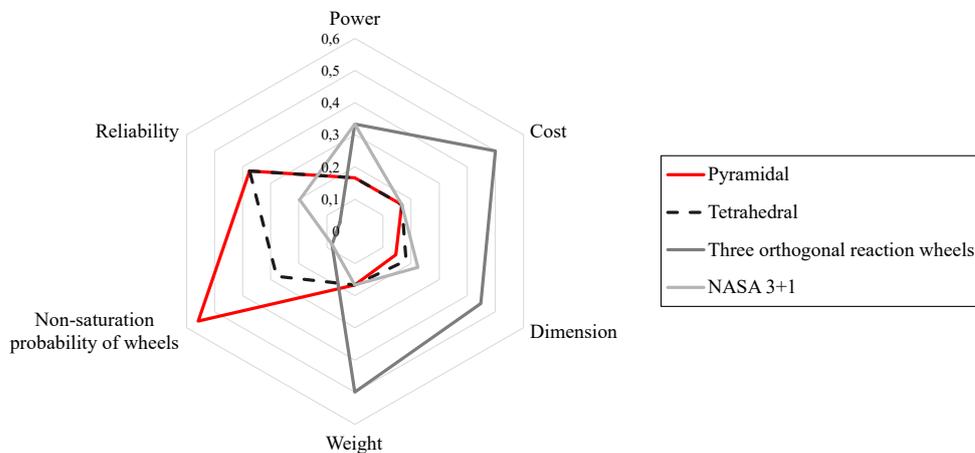
Figure 3.14: *Radar plot of the final scores obtained from the trade-off study*

Differences occur in the attribute dimension and non-saturation probability of the wheels. Regarding the dimension given that in the tetrahedral configuration one wheel is positioned orthogonally to the z-axis of the spacecraft the top three wheels can be compacted more occupying less space. A similar argument applies to the non-saturation probability of the wheels. Since in the tetrahedral configuration a reaction wheel is arranged along the z-axis, the angular momentum components along x and y can only be stored by three wheels. In contrast, in the pyramidal configuration, the angular momentum components can be stored in all four wheels, thus ensuring a lower saturation probability.

The second-ranked configuration is the three orthogonal wheels (indicated by 3 ORWs in the table). This is justified by the fact that this configuration scored higher than the other configurations in the attributes of power, cost, size, and weight. However, the configuration with only three wheels has a very low score on the reliability attribute, which has the highest weight, because it has no redundancy. The best configurations in terms of reliability are the pyramidal and tetrahedral as they have a redundant fourth wheel that acts as a hot backup. The NASA 3+1 configuration scored lower because it has three working wheels plus a redundant wheel that acts as a cold backup.

Actuator Sizing

Once the actuators have been selected, the sizing can be performed. Figure 3.15 shows the sizing process followed for both reaction wheels and magnetic torquers.
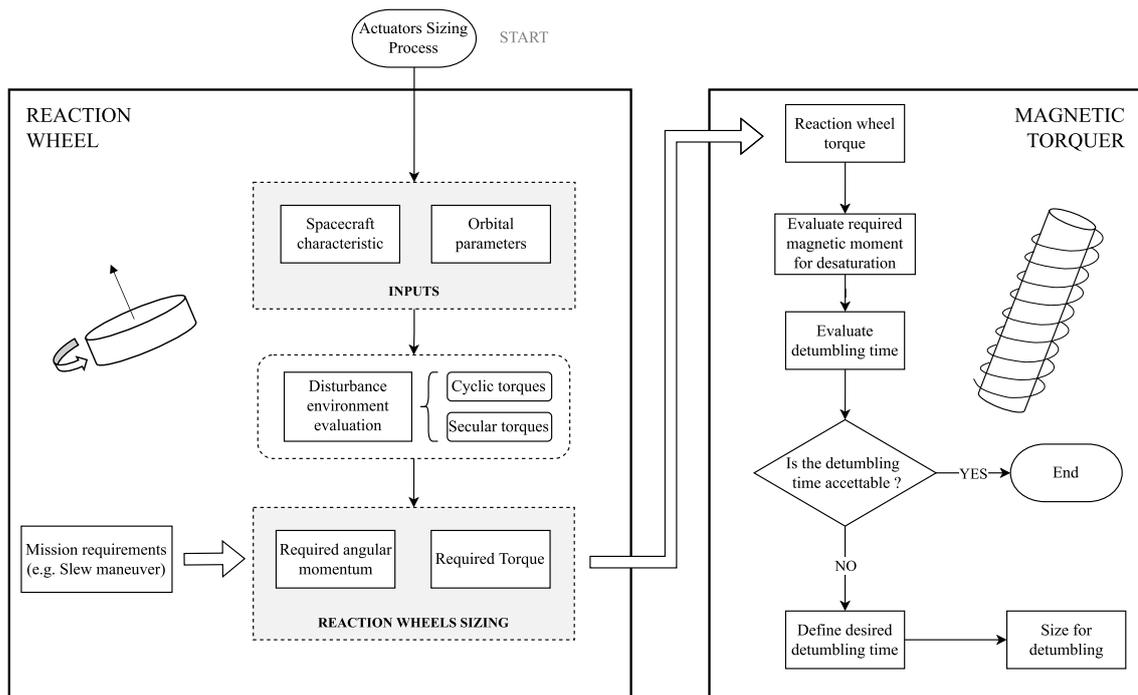
Figure 3.15: *Actuators sizing flow*

The process begins by evaluating the characteristics of the spacecraft and orbit before moving to the calculation of the external disturbance environment. To determine the necessary momentum capacity, it is important to distinguish between cyclic and secular disturbances in the spacecraft's environment. The distinction is shown in Table 3.11. Reaction wheels are typically sized to be able to store the full cyclic component of momentum without the need for frequent momentum dumping [45]. Therefore, the average disturbance torque for $1/4$ an orbit must be considered. The secular component of momentum will also need to be stored for the amount of time the spacecraft must be operational without a momentum dump being performed.

For conservative sizing purposes, it is assumed that the spacecraft points at nadir for the entire orbit. In practice, this may not be true since the observation payload equipped on board the Cube-Sat cannot take pictures while in eclipse, so continuous pointing at nadir would be unnecessary. Furthermore, it is also assumed that the spacecraft is equipped with a single reaction wheel, which therefore handles all angular momentum components.

As shown in Figure 3.15, before proceeding with the estimation of the required angular momentum and torque of the reaction wheel, it is necessary to assess whether the mission includes any slew manoeuvres. The observation payload requires to provide ground velocity compensation to obtain non-blurred images. This can be achieved by performing target pointing where the spacecraft continuously points to the target.

Figure 3.16 shows a situation where the spacecraft sees the target at the tangent to the earth, i.e. at the furthest point. Assuming this is the situation where the spacecraft starts target pointing, then before the spacecraft is commanded to track the ground target it is in a nadir pointing orientation. In this case, a large-angle slew manoeuvre will initially be required to point the camera towards the target. Therefore, if the spacecraft is commanded immediately to point to the target from an initial nadir pointing attitude the torque demanded for the reaction wheels can be excessive. For this reason, a pre-manoeuvre should be carried out to achieve the necessary attitude but within a longer time. As shown in Figure 3.16, it has been chosen to begin the pre-manoeuvre ahead of the point considered so far by an angle ϕ.
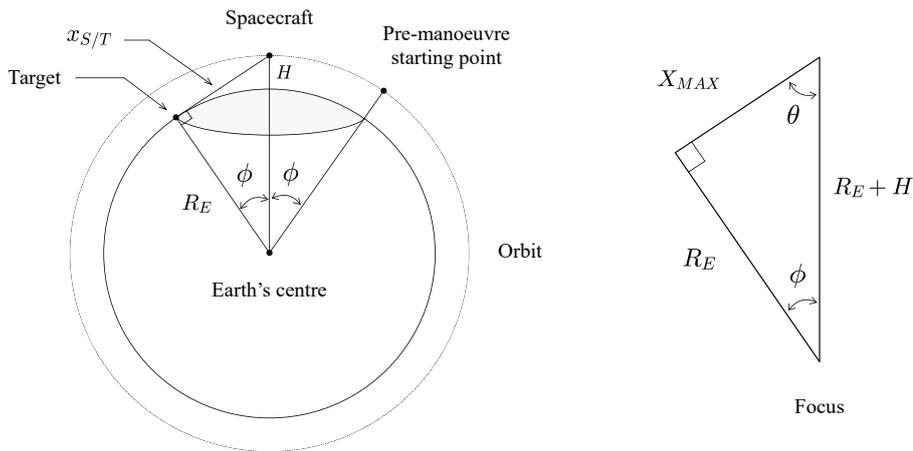
Figure 3.16: *Schematic representation for target pointing considerations. On the right is a focus of the right triangle formed by the spacecraft, the target, and the Earth's centre.*

Given a certain altitude of the orbit, it is possible to calculate the quantities of interest for the sizing of the reaction wheels. It is easy to demonstrate that higher altitudes result in less demanding slew manoeuvres as the available time increases. Therefore, a low-altitude orbit of 470 km is considered for the calculation.

The maximum distance, called X_{MAX} in the figure, can be calculated as

$$X_{MAX} = \sqrt{R_E^2 + (R_E + H)^2} \tag{3.6}$$

while the angle θ required for the satellite to point to the target can be calculated as

$$\theta = \arccos\left(\frac{X_{MAX}}{R_E + H}\right) \tag{3.7}$$

Assuming a circular orbit, then the time available to perform the pre-manoeuvre is given by the following equation

$$t_{pre} = t_{orb} \frac{\phi}{360°} \tag{3.8}$$

where t_{orb} is the orbital period, and ϕ is given by

$$\phi = \arccos\left(\frac{R_E}{R_E + H}\right) \tag{3.9}$$

Carrying out the calculations for an orbit with an altitude of $470 \ km$, yields a slew pre-manoeuvre of $68.65°$ in $334.5 \ sec$. If a bang-bang slew maneuver is performed, then the reaction wheel shall be able to store the peak momentum due to the slew in addition to all the other sources of momentum. The formulas required to calculate the peak angular momentum and torque of the slew manoeuvre are given in [45]. The slew torque for reaction wheels is given by

$$T_{slew} = 4\frac{\psi I}{t_{slew}^2} \tag{3.10}$$

where ψ is the slew angle, and I is the moment of inertia. While the peak angular momentum is given by

$$h_{peak} = \frac{1}{2} t_{slew} T_{slew} \tag{3.11}$$

Another component that adds up to the angular momentum to be stored is that due to the torque required to maintain nadir pointing when target pointing is not being performed. This component can be calculated using Equations 3.10 and 3.11 by substituting $t_{slew} = (T_{orb} - t_{pre})$ and $\psi = 360°$.

It follows that the required angular momentum that the reaction wheel must be able to store in an orbit is given by

$$h_{RW,req} = \frac{1}{2} t_{pre} T_{pre} + \frac{1}{2} t_{nadir} T_{nadir} + \frac{1}{4} t_{orb} T_{cyclic} + t_{orb} T_{secol} \tag{3.12}$$

While the torque for sizing is

$$T_{RW,req} = T_{pre} + T_{nadir} + T_{cyclic} + T_{secol} \tag{3.13}$$

It is possible to estimate the number of revolutions after which the reaction wheel saturates. It has been seen that the angular momentum stored in one revolution considering the peak due to the slew manoeuvre is that given by Equation 3.12. However, after one complete revolution, the angular momentum related to the cyclic torques and the slew manoeuvre is recovered except for small amounts. Therefore, the second revolution starts with a stored angular momentum equals to the residual angular momentum of the previous revolution, which essentially corresponds to the angular momentum accumulated due to the secular torques and the torque required for continuous nadir pointing.

By indicating with ε and η respectively the percentage recovered of angular momentum related to the slew manoeuvre and cyclic torques, the residual angular momentum after one revolution can be calculated as

$$h_1^{ar} = h_1^{peak} - \frac{1}{4} t_{orb} T_{cyclic} \cdot \eta - \frac{1}{2} t_{pre} T_{pre} \cdot \varepsilon \tag{3.14}$$

where η and ε are real numbers between 0 and 1, in the ideal case both would be equal to 1. h_1^{peak} is calculated with Equation 3.12 which include the peak due to the slew manoeuvre.

Therefore, the angular momentum that needs to be stored at the second revolution is given by

$$h_2^{peak} = h_1^{ar} + \frac{1}{2}\,t_{pre}\,T_{pre} + \frac{1}{2}\,t_{nadir}\,T_{nadir} + \frac{1}{4}t_{orb}\,T_{cyclic} + t_{orb}\,T_{secol} \tag{3.15}$$

This can be extended to the n-th revolution as shown in Table 3.16.

First revolution

$h_1^{peak} = \frac{1}{2}\,t_{pre}\,T_{pre} + \frac{1}{2}\,t_{nadir}\,T_{nadir} + \frac{1}{4}t_{orb}\,T_{cyclic} + t_{orb}\,T_{secol}$

$h_1^{ar} = h_1^{peak} - \frac{1}{4}t_{orb}\,T_{cyclic}\cdot\eta - \frac{1}{2}\,t_{pre}\,T_{pre}\cdot\varepsilon$

Second revolution

$h_2^{peak} = h_1^{ar} + \frac{1}{2}\,t_{pre}\,T_{pre} + \frac{1}{2}\,t_{nadir}\,T_{nadir} + \frac{1}{4}t_{orb}\,T_{cyclic} + t_{orb}\,T_{secol}$

$h_2^{ar} = h_2^{peak} - \frac{1}{4}t_{orb}\,T_{cyclic}\cdot\eta - \frac{1}{2}\,t_{pre}\,T_{pre}\cdot\varepsilon$

$\vdots$

$\vdots$

n-th revolution

$h_n^{ar} = n\cdot A - (n-1)\cdot B - (n-1)\cdot C$

$A = \frac{1}{2}\,t_{pre}\,T_{pre} + \frac{1}{2}\,t_{nadir}\,T_{nadir} + \frac{1}{4}t_{orb}\,T_{cyclic} + t_{orb}\,T_{secol}$

$B = \frac{1}{4}t_{orb}\,T_{cyclic}\cdot\eta$

$C = \frac{1}{2}\,t_{pre}\,T_{pre}\cdot\varepsilon$

Table 3.16: h_i^{peak} *angular momentum required in a orbit,* h_i^{ar} *angular momentum after a revolution*

Assuming $h_n^{ar} = h_{RW_{max}}$, the number of orbits n_{max} required to saturate the reaction wheel, which has an angular momentum storage capacity of $h_{RW_{max}}$, can be calculated as

$$n_{max} = \frac{h_{RW_{max}} - B - D}{A - B - D} \tag{3.16}$$

where

$$A - B - D = t_{orb}\,T_{secol} + \frac{1}{2}\,t_{nadir}\,T_{nadir}$$
$$+ \frac{1}{4}t_{orb}\,T_{cyclic}\cdot(1-\eta) + \frac{1}{2}\,t_{pre}\,T_{pre}\cdot(1-\varepsilon) \tag{3.17}$$

Therefore, Equation 3.16 highlights that the dominant contribution defining the number of revolutions required for wheel saturation is that given by the secular torques. If the secular torques have a lower intensity then the wheel will saturate after a higher number of revolutions.

Once the reaction wheel torque is known, the magnetic torquers can be sized. Indeed, the sizing process shown in Figure 3.15 involves sizing the magnetic torquers so that the wheel desaturation can be correctly performed. Once the value of the dipole moment of the magnetic torquers has been obtained, a detumbling simulation is carried out to check whether this allows the detumbling of the spacecraft in a reasonable time. If the time required for detumbling is acceptable, then the sizing is considered completed otherwise the magnetic torquers are sized to guarantee an appropriate detumbling time. The desaturation of the reaction wheels may require different amounts of

time depending on the torque used for desaturation. For example, if desaturation of the wheels needs to be performed quickly, then the magnetic torquers must be able to deliver higher torque than if desaturation is performed slower. The maximum desaturation speed is achieved by using the maximum torque of the reaction wheels to desaturate. If this is the case then the torque used to size the dipole moment of the magnetic torquers is equal to the maximum torque of the reaction wheels. However, in the mission of interest, the maximum torque of the reaction wheels is not high as the only slew manoeuvre that is performed is not demanding. Therefore, sizing the magnetic torquers with the maximum torque of the reaction wheels results in a dipole moment that implies excessive detumbling times. Consequently, it was decided to size the magnetic torquers so that the detumbling phase ends in a maximum of 3 orbits for a certain range of initial angular velocities.

The results obtained for the momentum budget for an altitude of 470 km are shown in Table 3.17, while the data used are shown in Table 3.10. Table B.6 shows the results obtained by repeating the calculation for an altitude orbit of 700 km.

Reaction wheels			
Parameters	**Values**	**Units**	**Notes**
Torque for sizing	$6.67 \cdot 10^{-6}$	Nm	-
Required angular momentum	0.0092	Nms	-
Required slew rate	0.2052	deg/s	-
Number of orbits required for saturation	6.59	rev	Assuming: $h_{RW_{max}} = 0.03\ Nms$, $\eta = \varepsilon = 0.9$
Time required for saturation	10.32	h	Same as above

Magnetic torquers			
Parameters	**Values**	**Units**	**Notes**
Required torquer dipole*	0.5	$A\,m^2$	Guarantees detumbling in less than 3 orbits for initial angular velocities less than $10\,deg/s$ in all three axes
Time required for desaturation	370	s	Calculated considering the required angular momentum and the torque for sizing of the reaction wheels

* Defined through detumbling simulations, see Section 5.1

Table 3.17: *Momentum budget results for an altitude orbit of 470 km, the data used are shown in Table 3.10*

Considering these results, the actuators required for the mission can be selected. The characteristics of the reaction wheels and magnetic torquers chosen for the 3U CubeSat are shown in Appendix D.

3.2.6 Determination and Control Algorithms

Suitable algorithms must be implemented to put together the operation of actuators and sensors. The necessary algorithms concern both attitude determination and attitude control. Nowadays, as seen in Section 1.5.2, advanced algorithms exist to accomplish both tasks, although there is a tendency to use long-established algorithms for reliability reasons. In this thesis work, some of these advanced algorithms are implemented to ensure that the CubeSat can correctly perform the different phases of the mission while meeting all requirements.

Figure 3.17 depicts the framework representing how the information from the sensors and the actuator commands are looped together through the algorithms executed in the micro-controller unit.
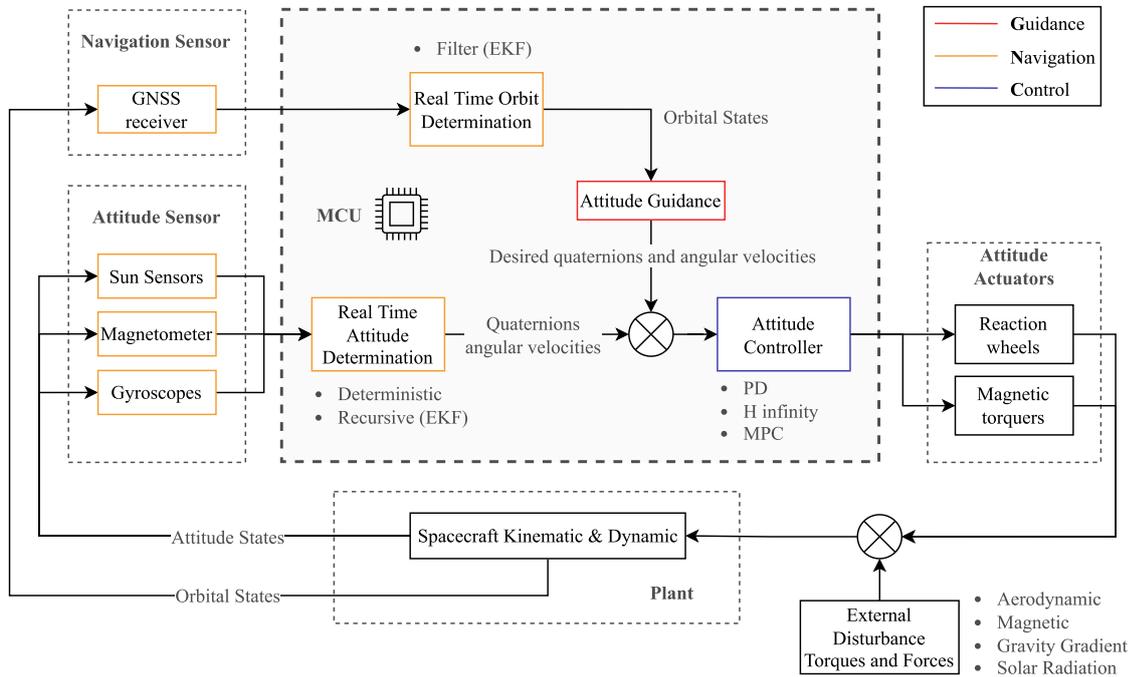
Figure 3.17: *Closed loop feedback control of a spacecraft*

Mode	Guidance		Attitude Determination	Attitude Control
Detumbling	ω_{OB}^{B}	$= \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}$	KRE	B dot
Nadir Pointing MPC	$\bar{q}$	$= \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}$	EKF + SVD	MPC
Nadir Pointing H_∞	$\bar{q}$	$= \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}$	EKF + SVD	H_∞
Target Pointing	$q_c(t)$	$= \begin{bmatrix} \cos\delta/2 & \bar{u}_c\sin\delta/2 \end{bmatrix}$	EKF + SVD	QFC
Desaturation	$\bar{q}$	$= \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}$	EKF + SVD	MPC or H_∞
	h_{RW}	$= \begin{bmatrix} h^{**} & h^{**} & h^{**} \end{bmatrix}$		CPMC

KRE: Kalman Rate Estimator
EKF: Extended Kalman Filter
SVD: Singular Value Decomposition Method
MPC: Model Predictive Controller
QFC: Quaternion Feedback Controller
CMPC: Cross Product Magnetic Controller
H_∞: H infinity controller

Table 3.18: *Attitude determination and control algorithms related to the control modes of the ADCS. Two different types of controllers have been developed for the nadir pointing mode*

Attitude determination consists of determining the attitude of the spacecraft using the measurements of sensors available on board. The sensors used are not perfect and suffer from measurement noise, so data must be filtered through efficient algorithms to obtain good measurements even with low-accuracy and inexpensive sensors. Attitude control is responsible for ensuring the correct orientation of the spacecraft, both for functional reasons and to perform mission-related tasks. Control algorithms apply to the actuators, which are the components that, when properly

controlled, allow governing the spacecraft.

Table 3.18 shows the attitude determination and control algorithms used in the operative control modes along with the guidance function that represents the desired to be achieved. Their characteristics will be discussed in detail in the following sections.

The performance requirements defined in Table 3.8 for fine pointing apply only to these control modes in which pointing is required, i.e. nadir and target pointing. Requirements can be relaxed to *coarse pointing* when the spacecraft is not performing the scientific measurement, for example when it is in eclipse. For other control modes there are no pointing requirements. The technical specifications of the requirements for fine and coarse pointing can be found in Table C.1.

Chapter 4

Attitude Determination

Before attitude control can be discussed, the current attitude of the spacecraft must be estimated. As mentioned in Section 1.5.2, two techniques can be used to do this, which differ in the nature of the mathematics involved: *deterministic* attitude methods and state *estimation* methods. The fist category comprises static determination approaches that depend on measurements taken at the same time, while the second category comprises filtering approaches that make explicit use of knowledge of the motion of the spacecraft to accumulate a "memory" of past measurements. Both state estimators (also called recursive methods) and deterministic processors have advantages and disadvantages.

The **deterministic methods** nearly always provide a solution and require a rough a priori estimate of the attitude. The methods and results are easy to interpret physically and geometrically. However, it is algebraically difficult to model biases, a time-varying attitude, or other attitude-related parameters with deterministic processors [44].

The **recursive methods*** can provide statistically optimal solutions. However, state estimators may diverge and provide no solution at all. Furthermore, they require a dynamic model or a more accurate estimate of the a priori attitude than do deterministic methods, and the geometrical and physical meaning of the results can be very difficult to interpret.

Before going into detail about the theory behind these techniques, the logical framework in which they are used in this thesis work for attitude and angular velocity estimation is presented. Figure 4.1 depicts a detailed description of the workflow used for fine attitude determination. The determination process is carried out by combining deterministic and recursive methods. The workflow begins with acquiring measurements, referred to the body frame $\mathcal{F}_B$, provided by the sun sensors and magnetometer at a frequency of 10Hz. These measurements are noisy, so they are appropriately filtered using a complementary filter before being used for attitude determination. At the same frequency, the processing unit calculates the corresponding vectors related to the local orbital frame $\mathcal{F}_O$ using the models discussed in the previous sections. For example Figure 2.9 shows the unit sun vector expressed in orbital frame. Thus, two pairs of measurements are available for deterministic attitude estimation using the Singular Value Decomposition Method. This deterministic solution is then applied to update the estimate made by the extended kalman filter, which also estimates the satellite's angular velocity using measurements from the gyroscopes.

Since the deterministic attitude calculation uses the measurement from the sun sensors, it cannot be performed during the eclipse phase in orbit.

---

*The term *recursive* is used to indicate that a new estimate state vector is obtained after each observation, as stated in [44]
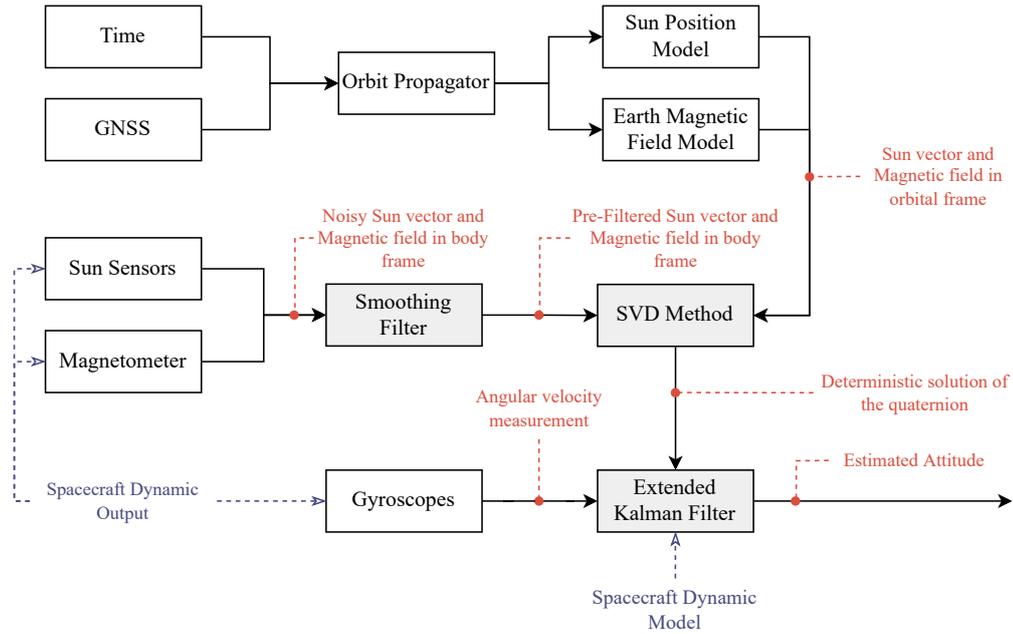
Figure 4.1: *Schematic representation of the attitude determination algorithms operating logic*

As seen in the previous sections, the models used to calculate the sun vector and the Earth's magnetic field in the orbital reference frame require the satellite's position in the ECI reference frame. For this reason, it is essential to have an orbital propagator on board the satellite to provide the satellite's position. Similarly to the attitude case, it is also possible to implement estimation algorithms for the position using measurements from sensors. In this case, the satellite's position can be estimated by an Extended Kalman Filter that uses the dynamic model of the orbital propagator and the measurements obtained from the GNSS receiver for updating the estimate. However, this is beyond the scope of this thesis work and so will not be discussed.

4.1 Deterministic Methods

Deterministic methods use sensor outputs at one given point in time and does not include dynamics of the system. These methods require at least two vector measurements to determine the attitude. Unit vectors that can be used are typically the unit vector to the Sun and the Earth's magnetic field vector for coarse Sun-magnetic attitude determination. However, this can be a problem when the satellite is in eclipse because the Sun vector measurement cannot be used.

Black's 1964 TRIAD algorithm was the first published method for determining the attitude of a spacecraft using body and reference observations, but his method could only combine the information from two measurements [31, 17]. TRIAD method was improved one year later by Grace Wahba by introducing arbitrary weighting of the measurements and by allowing the use of more than two measurements [42]. There are several solutions to the so-called Wahba's problem. The SVD method was chosen among them, which has good performance and accuracy.

4.1.1 Wahba's Problem and SVD Method

In 1965, Grace Wahba framed the problem of finding the orthogonal matrix A with determinant 1 that minimises the loss function

$$L(\boldsymbol{A}) = \frac{1}{2} \sum_{i=1}^{N} a_i \|\bar{b}_i - \boldsymbol{A}\bar{r}_i\|^2 \tag{4.1}$$

where $\bar{b}_i$ is a set of N unit vectors measured in spacecraft's body frame, $\bar{r}_i$ are the corresponding unit vectors in the orbital reference frame, and a_i are non-negative weights.

Using the orthogonality of $\boldsymbol{A}$, the unit norm of the unit vectors, and the cyclic invariance of the trace, Equation 4.1 can be written as

$$L(\boldsymbol{A}) = \lambda_0 - \text{tr}(\boldsymbol{A}\boldsymbol{B}^T) \tag{4.2}$$

with

$$\lambda_0 = \sum_{i=1}^{N} a_i \tag{4.3}$$

and the *attitude profile matrix* $\boldsymbol{B}$ defined by

$$\boldsymbol{B} = \sum_{i=1}^{N} a_i \, \bar{b}_i \, \bar{r}_i^T \tag{4.4}$$

From Equation 4.1 it is clear that the loss function is minimised when $\text{tr}(\boldsymbol{A}\boldsymbol{B}^T)$ is maximised.

Algorithms for solving Wahba's problem fall into two classes [31]. The first solves for the attitude matrix directly, and the second for the quaternion representation of the attitude matrix. However, in the first case, it is still possible to derive quaternions from the A-matrix. The SVD method, used in this work, belongs to the first class.

Singular Value Decomposition (SVD) Method

One of the first solutions of Wahba's problem was presented by J.L. Farrell and J.C. Stuelpnagel [31, 25]. They performed a two-step process which is equivalent to the single-step procedure known as the singular value decomposition, for which very robust algorithms have been developed.

The singular value decomposition of the attitude profile matrix is given by

$$\boldsymbol{B} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T = \boldsymbol{U} \, \text{diag}(\begin{bmatrix} s_1 & s_2 & s_3 \end{bmatrix}) \, \boldsymbol{V}^T \tag{4.5}$$

where U and V are orthogonal, and $s_1 \geqslant s_2 \geqslant s_3 \geqslant 0$. By following the steps shown in [31, 25], it is possible to prove that the trace $\text{tr}(\boldsymbol{A}\boldsymbol{B}^T)$ is maximised with the following optimal attitude matrix

$$\boldsymbol{A}_{opt} = \boldsymbol{U} \, \text{diag}(\begin{bmatrix} 1 & 1 & \det\boldsymbol{U} & \det\boldsymbol{V} \end{bmatrix}) \, \boldsymbol{V}^T \tag{4.6}$$

Once the attitude matrix is known, the quaternion elements can be extracted by using the following equations

$$\begin{aligned}
q_0 &= \pm\sqrt{1 + A_{11} - A_{22} - A_{33}}/2 \\
q_1 &= (A_{12} + A_{21})/(4\,q_0) \\
q_2 &= (A_{13} + A_{31})/(4\,q_0) \\
q_3 &= (A_{23} - A_{32})/(4\,q_0)
\end{aligned} \tag{4.7}$$

Other formulations are also available to arrive at the same result. Figure 4.2 shows an example of the application of SVD method for attitude determination. Noise was removed from the sensor models in the simulation in order to verify the error-free operation of the method. Naturally, in the simulations that will be discussed in the following sections, the method will receive noisy measurements as input.
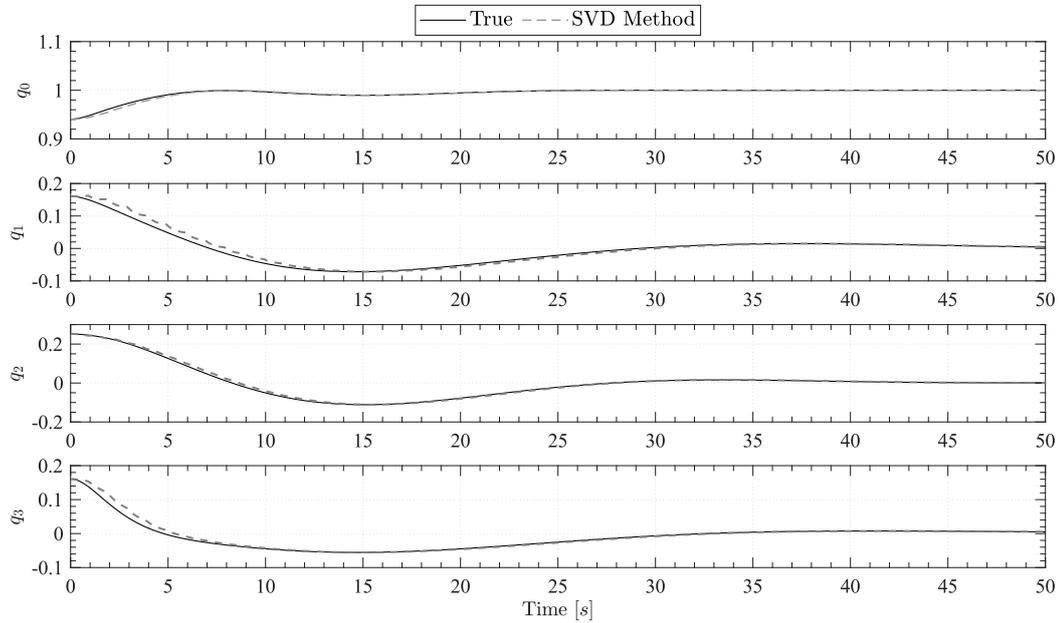
Figure 4.2: *Example of application of the SVD method on a nadir pointing simulation starting with a certain misalignment from nadir. Sensor noise was removed to verify the correct functioning of the method.*

4.2 Recursive Methods

4.2.1 Smoothing Filter

Before using the measurements obtained from the magnetometer and sun sensors in the SVD method, they are pre-filtered using a smoothing filter. The term *Smoothing Filter* is used in this work to denote a variant of the well-known Exponential smoothing method. Exponential smoothing is a rule-of-thumb technique for smoothing time series data using the exponential window function and acting as low-pass filters to remove high-frequency noise. In this work, smoothing is performed according to the following equation

$$\bar{x} = (1 - \alpha)\bar{x}^- + \alpha y \tag{4.8}$$

where x is the output of the smoothing, α is a weight factor between 0 and 1, $\bar{x}^-$ is the past measurement, and y is the present measurement. In the case of interest, this formula is applied to the measurements obtained from the sun sensors and the magnetometer. Figure 4.3 shows an example in which the smoothing filter is applied to the noisy magnetic field measurement, while Figure 4.4 shows the difference in the output of the SVD method with and without the smoothing filter.
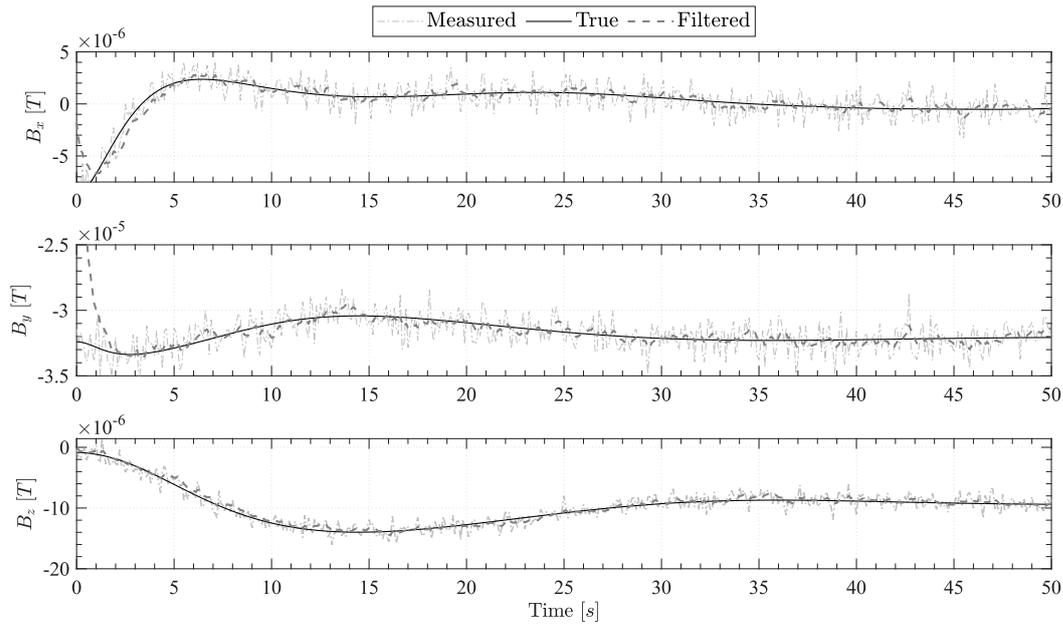
Figure 4.3: *Example of application of the smoothing filter on the noisy magnetic field measurement.*
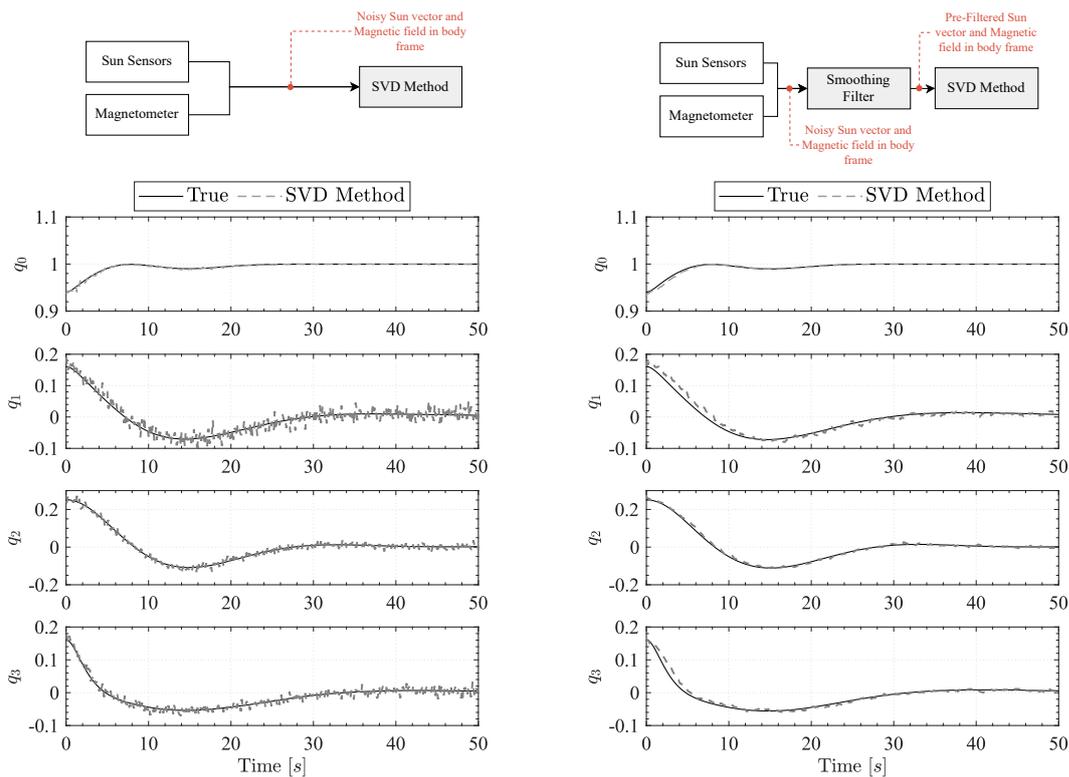
Figure 4.4: *Differences on the SVD output with and without smoothing filter. In the block diagram above, only the input, of the SVD method, of the measurements in the body frame is shown for the purpose of exposition.*

4.2.2 Extended Kalman Filter

The Kalman Filter is an algorithm used to solve the state estimation problem, also known as *sequential probabilistic inference*, which involves finding the best estimate of the true system using a dynamic model and measurements that are both corrupted with random noise of known statistics [31]. Thanks to its properties, the Kalman Filter is an optimal filter for Gaussian errors with zero mean acting on the system, indeed the main assumption of the Kalman Filter is that all error terms and measurements have a Gaussian distribution.

The variables to be estimated are collected into a *state vector*, which typically includes other variables in addition to the attitude. For example, a common choice is to compose the state vector with angular velocities and quaternions.

The simplest version of the Kalman filter is the so-called Linear Kalman Filter (LKF) in which the dynamic model of the system is linear. There are two basic nonlinear generalizations to Kalman Filter

- Extended Kalman Filter (EKF): Analytic linearization of the model at each point in time;

- Sigma-Point (Unscented) Kalman filter (SPKF/UKF): Statistical/empirical linearization of the model at each point in time.

The solution process followed by any version of the Kalman filter involves two macrosteps:

1. **Prediction**: The present state value is predicted based on all past available data;

2. **Estimation/Updating**: The present state value is estimated by updating the prediction based on all presently available data.

In this work the EKF has been chosen as optimal filter for SILVA's CubeSat. Table 4.1 shows a summary of the process steps. The output of the process has two part: the state estimate and the covariance estimate. In fact, at the end of every iteration the best guess of the present value $\hat{x}_k^+$ is computed, however, the estimation is not perfect and the covariance matrix gives the uncertainty of $\hat{x}_k^+$.

A common choice to represent the spacecraft attitude is to use quaternions but they must respect the unit length constraint. This leads to problems when using the EKF especially in the update step. In this case, the term Additive Quaternion Representation is used, and the EKF is called AEKF. For this reason, several methods and solutions have been proposed to solve this problem. As discussed in [31], there are methods applied to additive quaternion representation which only partially solve the problem. Other representations as the Multiplicative Quaternion Representation (MQP), solve the problem by writing the true quaternion as the product of the estimated quaternion and an error quaternion rather than as the sum.

In this work, the approach followed in [49, 50] is adopted in which a reduced representation is used that incorporates the quaternions unit length requirement.

System Model

The nonlinear model of the spacecraft's dynamic is given by 2.69. If the spacecraft is equipped with reaction wheels Equation 2.69 become

$$\dot{\bar{\omega}}_{IB}^B = \boldsymbol{I}^{-1}(\bar{T}^B - \bar{\omega}_{IB}^B \times (\boldsymbol{I}\bar{\omega}_{IB}^B + \boldsymbol{I}_{RW}\bar{\omega}_{RW})) + \bar{w}_2 \tag{4.9}$$

Nonlinear state-space model

$$x_k = f(x_{k-1}, u_{k-1}, w_{k-1}) \qquad y_k = h(x_k, u_k, v_k)$$

Definitions

$$\hat{A}_k = \left.\frac{df(x_k, u_k, w_k)}{dx_k}\right|_{x_k=\hat{x}_k^+} \qquad \hat{B}_k = \left.\frac{df(x_k, u_k, w_k)}{dw_k}\right|_{w_k=\bar{w}_k}$$

$$\hat{C}_k = \left.\frac{dh(x_k, u_k, v_k)}{dx_k}\right|_{x_k=\hat{x}_k^-} \qquad \hat{D}_k = \left.\frac{dh(x_k, u_k, v_k)}{dv_k}\right|_{v_k=\bar{v}_k}$$

Initialization

$$\hat{x}_0^+ = \mathbb{E}[x_0] \qquad\qquad \boldsymbol{\Sigma}_{\tilde{x},0}^+ = \mathbb{E}\left[\left(x_0 - \hat{x}_0^+\right)\left(x_0 - \hat{x}_0^+\right)^T\right]$$

Computation

1.a	State estimate:	$\hat{x}_k^- = f(x_{k-1}^+, u_{k-1}, w_{k-1})$
1.b	Error covariance:	$\boldsymbol{\Sigma}_{\tilde{x},k}^- = \hat{A}_{k-1}\boldsymbol{\Sigma}_{\tilde{x},k-1}^+\hat{A}_{k-1}^T + \hat{B}_{k-1}\boldsymbol{\Sigma}_{\tilde{w}}\hat{B}_{k-1}^T$
1.c	Output prediction:	$\hat{y}_k = h(\hat{x}_k^-, u_k, \bar{v}_k)$
2.a	Estimation gain matrix:	$\boldsymbol{L}_k = \boldsymbol{\Sigma}_{\tilde{x},k}^-\hat{C}_k^T\left[\hat{C}_k\boldsymbol{\Sigma}_{\tilde{x},k}^-\hat{C}_k^T + \hat{D}_k\boldsymbol{\Sigma}_{\tilde{v}}\hat{D}_k^T\right]$
2.b	State estimate update:	$\hat{x}_k^+ = \hat{x}_k^- + L_k(y_k - \hat{y}_k)$
2.c	Error covariance update:	$\boldsymbol{\Sigma}_{\tilde{x},k}^+ = (\boldsymbol{I} - \boldsymbol{L}_k\hat{C}_k)\boldsymbol{\Sigma}_{\tilde{x},k}^-$

Table 4.1: *Summary of the steps of the Extended Kalman Filter (EKF)*

where $\boldsymbol{I}_{RW}$ is the inertia matrix of the wheels with respect to the body frame, and $\bar{\omega}_{RW}$ is the angular velocity of the wheels.

As shown in [48], a reduced quaternion model can be considered in which the unit norm constraint is directly incorporated into the model. In this case, the time derivative of the quaternion vector part is given by

$$\dot{\vec{q}} = \frac{1}{2}\boldsymbol{\Omega}\left(\bar{\omega}_{OB}^B + \bar{w}_1\right) \tag{4.10}$$

where the angular velocity in body frame is derived from 2.70,

$$\bar{\omega}_{OB}^B = \bar{\omega}_{IB}^B - \boldsymbol{A}_O^B\bar{\omega}_{IO}^O \tag{4.11}$$

where $\bar{\omega}_{IO}^O = \begin{bmatrix} 0 & -\omega_0 & 0 \end{bmatrix}$, while $\boldsymbol{\Omega}$ is given by

$$\boldsymbol{\Omega} = \begin{bmatrix} q_0(\vec{q}) & -q_3 & q_2 \\ q_3 & q_0(\vec{q}) & -q_1 \\ -q_2 & q_1 & q_0(\vec{q}) \end{bmatrix} \tag{4.12}$$

The scalar part of the quaternion is a function of the vector part

$$q_0(\vec{q}) = \sqrt{1 - q_1^2 - q_2^2 - q_3^3} \tag{4.13}$$

In [48] it is discussed how this representation is always a full rank matrix except for the case where α, in Equation 2.13, equals $\pm\pi$.

$\bar{w} = \begin{bmatrix} \bar{w}_1, & \bar{w}_2 \end{bmatrix}$ is the process Gaussian noise.

Measurement Model

As presented in Section 3.2.4, the CubeSat is equipped with gyroscopes, sun sensors and the triaxial magnetometer, so the gyroscopes provide a measure of angular velocity while sun sensors and magnetometer provide a quaternion measurement through the SVD method. The simple gyroscope model used is given by [44]

$$\bar{\omega}_y = \bar{\omega} + \bar{\beta} + \bar{v}_2$$
$$\dot{\bar{\beta}} = \bar{\eta}_u$$
$$\bar{q}_y = \vec{q} + \bar{v}_1$$

(4.14)

where $\bar{v} = \begin{bmatrix} \bar{v}_1, & \bar{v}_2 \end{bmatrix}$ is the measurement noise vector and $\bar{\eta}_u$ is the rate random walk. The quaternion perturbation model is that of [50] which is more general than the multiplicative perturbation as shown in the paper.

EKF Implementation

The overall system equations are given as follows:

$$\dot{\vec{q}} = \frac{1}{2}\boldsymbol{\Omega} \left(\bar{\omega}_{OB}^B + \bar{w}_1\right)$$
$$\dot{\bar{\omega}}_{IB}^B = \boldsymbol{I}^{-1}(\bar{T}^B - \bar{\omega}_{IB}^B \times (\boldsymbol{I}\bar{\omega}_{IB}^B + \boldsymbol{I}_{RW}\bar{\omega}_{RW})) + \bar{w}_2$$
$$\bar{q}_y = \vec{q} + \bar{v}_1$$
$$\bar{\omega}_y = \bar{\omega}_{IB}^B + \bar{\beta} + \bar{v}_2$$
$$\dot{\bar{\beta}} = \bar{\eta}_u$$

(4.15)

The discrete version, with sampling period dt, of 4.15 is given by

$$\vec{q}_{k+1} = \vec{q}_k + \frac{1}{2}\boldsymbol{\Omega}_k \left(\bar{\omega}_{OB_k}^B + \bar{w}_{1_k}\right)dt$$
$$\bar{\omega}_{IB_{k+1}}^B = \bar{\omega}_{IB_k}^B + \boldsymbol{I}^{-1}(\bar{T}_k^B - \bar{\omega}_{IB_k}^B \times (\boldsymbol{I}\bar{\omega}_{IB_k}^B + \boldsymbol{I}_{RW_k}\bar{\omega}_{RW_k}))dt + \bar{w}_{2_k}$$
$$\bar{\beta}_{k+1} = \bar{\beta}_k + \bar{\eta}_{u_k}dt$$
$$\bar{q}_{y_k} = \vec{q}_k + \bar{v}_{1_k}$$
$$\bar{\omega}_{y_k} = \bar{\omega}_{IB_k}^B + \bar{\beta}_k + \bar{v}_{2_k}$$

(4.16)

which can be written as a standard state space model as follows

$$\begin{bmatrix} \vec{q}_{k+1} \\ \bar{\omega}_{IB_{k+1}}^B \\ \bar{\beta}_{k+1} \end{bmatrix} = \left(\begin{bmatrix} \vec{q}_k \\ \bar{\omega}_{IB_k}^B \\ \bar{\beta}_k \end{bmatrix} + \begin{bmatrix} \frac{1}{2}\boldsymbol{\Omega}_k \, \bar{\omega}_{OB_k}^B \\ \bar{\omega}_{IB_k}^B + \boldsymbol{I}^{-1}(\bar{T}_k^B - \bar{\omega}_{IB_k}^B \times (\boldsymbol{I}\bar{\omega}_{IB_k}^B + \boldsymbol{I}_{RW_k}\bar{\omega}_{RW_k})) \\ \mathbf{0} \end{bmatrix} dt \right)$$
$$+ \begin{bmatrix} \frac{1}{2}\boldsymbol{\Omega}_k \, \bar{w}_{1_k} \\ \bar{w}_{2_k} \\ \bar{\eta}_{u_k} \end{bmatrix} dt = f(\bar{x}_k, \bar{u}_k, \bar{w}_k) = \bar{F}(\bar{x}_k, \bar{u}_k) + \bar{G}(\bar{x}_k, \bar{u}_k)\bar{w}_k$$

(4.17)

$$\begin{bmatrix} \bar{q}_{y_k} \\ \bar{\omega}_{y_k} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\mathcal{I}}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \boldsymbol{\mathcal{I}}_3 & \boldsymbol{\mathcal{I}}_3 \end{bmatrix} \begin{bmatrix} \vec{q}_k \\ \bar{\omega}_{IB_k}^B \\ \bar{\beta}_k \end{bmatrix} + \begin{bmatrix} \bar{v}_{1_k} \\ \bar{v}_{2_k} \end{bmatrix} = h(\bar{x}_k, \bar{u}_k, \bar{v}_k) = \bar{H}\bar{x}_k + \bar{v}_k$$

(4.18)

It is assumed that process and measurement noise signals are Gaussian white noise satisfying the following equations

Zero mean	$\mathbb{E}[\bar{w}_k] = 0$	$\mathbb{E}[\bar{v}_k] = 0$	$\forall k$
Uncorrelated	$\mathbb{E}[\bar{w}_i \bar{v}_j^T] = 0$		$\forall i,\, j$
White noise	$\mathbb{E}[\bar{w}_i \bar{w}_j^T] = 0$	$\mathbb{E}[\bar{v}_i \bar{v}_j^T] = 0$	$\forall i, \neq j$
Notation	$\mathbb{E}[\bar{w}_k \bar{w}_k^T] = \mathbf{\Sigma}_w$	$\mathbb{E}[\bar{v}_k \bar{v}_k^T] = \mathbf{\Sigma}_v$ $\mathbb{E}[\bar{\eta}_{u_k} \bar{\eta}_{u_k}^T] = \mathbf{\Sigma}_u$	$\forall k$

In order to implement the process shown in Table 4.1, the matrices $\hat{A}$, $\hat{B}$, $\hat{C}$, and $\hat{D}$ must be derived. It is easy to see that

$$\hat{C}_k = \frac{dh(x_k,\, u_k,\, v_k)}{dx_k} = \begin{bmatrix} \mathcal{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathcal{I}_3 & \mathcal{I}_3 \end{bmatrix} \tag{4.19}$$

$$\hat{D}_k = \frac{dh(x_k,\, u_k,\, v_k)}{dv_k} = \begin{bmatrix} \mathcal{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathcal{I}_3 \end{bmatrix} \tag{4.20}$$

$$\hat{B}_k = \frac{df(x_k,\, u_k,\, w_k)}{dw_k} = \begin{bmatrix} \frac{1}{2}\mathbf{\Omega}_k & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathcal{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathcal{I}_3 \end{bmatrix} dt \tag{4.21}$$

To calculate the matrix $\hat{A}$ it is convenient to consider the individual components of the vectors F and G, which in turn are vectors. It is known from calculus that the derivative of a vector with respect to another vector returns a matrix. Given that

$$\bar{F}_1 = \vec{q}_{k+1} = \vec{q}_k + \frac{1}{2}\mathbf{\Omega}_k \bar{\omega}_{OB_k}^B dt \tag{4.22}$$

and

$$\bar{F}_2 = \bar{\omega}_{IB_k}^B + \mathbf{I}^{-1}(\bar{T}_k^B - \bar{\omega}_{IB_k}^B \times (\mathbf{I}\bar{\omega}_{IB_k}^B + \mathbf{I}_{RW_k}\bar{\omega}_{RW_k})) dt \tag{4.23}$$

it follows that

$$\hat{A}_k = \frac{df(x_k,\, u_k,\, w_k)}{dx_k} = \begin{bmatrix} \dfrac{\partial \bar{F}_1}{\partial \vec{q}_k} & \dfrac{\partial \bar{F}_1}{\partial \bar{\omega}_{IB_k}^B} & \dfrac{\partial \bar{F}_1}{\partial \bar{\beta}_k} \\[2ex] \dfrac{\partial \bar{F}_2}{\partial \vec{q}_k} & \dfrac{\partial \bar{F}_2}{\partial \bar{\omega}_{IB_k}^B} & \dfrac{\partial \bar{F}_2}{\partial \bar{\beta}_k} \end{bmatrix} \tag{4.24}$$

where $\frac{\partial \bar{F}_1}{\partial \vec{q}_k}$, $\frac{\partial \bar{F}_1}{\partial \bar{\omega}_{IB_k}^B}$, $\frac{\partial \bar{F}_2}{\partial \vec{q}_k}$, $\frac{\partial \bar{F}_2}{\partial \bar{\omega}_{IB_k}^B}$, $\frac{\partial \bar{F}_1}{\partial \bar{\beta}_k}$, and $\frac{\partial \bar{F}_2}{\partial \bar{\beta}_k}$ are three-by-three matrices.

It is important to note that in Equation 4.22, the angular velocity of the spacecraft with respect to the orbital frame is a function of the angular velocity with respect to the inertial frame and the rotation matrix as a result of Equation 4.11. This relationship can be shown by substituting equation 4.11 into expression 4.22

$$\begin{aligned} \bar{F}_1 &= \vec{q}_k + \frac{1}{2}\mathbf{\Omega}_k\,(\bar{\omega}_{IB}^B - \mathbf{A}_O^B\bar{\omega}_{IO}^O) dt \\ &= \vec{q}_k + \frac{1}{2}\mathbf{\Omega}_k\,\bar{\omega}_{IB}^B\,dt - \frac{1}{2}\mathbf{\Omega}_k\,\mathbf{A}_O^B\bar{\omega}_{IO}^O\,dt \end{aligned} \tag{4.25}$$

where $\mathbf{A}_O^B$ is a function of the attitude expressed by quaternions. It follows that

$$\frac{\partial \bar{F}_1}{\partial \vec{q}_k} = \mathcal{I} + \frac{1}{2}\frac{\partial}{\partial \vec{q}_k}\left(\mathbf{\Omega}_k\,\bar{\omega}_{IB}^B\right) dt - \frac{1}{2}\frac{\partial}{\partial \vec{q}_k}\left(\mathbf{\Omega}_k \mathbf{A}_O^B \bar{\omega}_{IO}^O\right) dt \tag{4.26}$$

$$\frac{\partial \bar{F}_1}{\partial \bar{\omega}_{IB_k}^B} = \frac{1}{2}\,\Omega_k\,dt \tag{4.27}$$

Another important observation concerns $\bar{F}_2$ in which the term representing the torques applied to the spacecraft is present. As seen in Section 2.5.3 the torque due to the gravity gradient can be written as a function of the satellite attitude as expressed in Equation 2.76. Therefore, this contribution must be taken into account in the derivation of $\bar{F}_2$ with respect to quaternions

$$\frac{\partial \bar{F}_2}{\partial \vec{q}_k} = \boldsymbol{I}^{-1}\,\frac{\partial \bar{T}_k^B}{\partial \vec{q}_k}\,dt \tag{4.28}$$

While there are no particular considerations in the derivation of $\bar{F}_2$ with respect to angular velocity

$$\frac{\partial \bar{F}_2}{\partial \bar{\omega}_{IB_k}^B} = \mathcal{I} - \frac{\partial}{\partial \bar{\omega}_{IB_k}^B}\left(\boldsymbol{I}^{-1}\bar{\omega}_{IB_k}^B \times \boldsymbol{I}\bar{\omega}_{IB_k}^B\right) \tag{4.29}$$

Appendix E gives the analytical expressions of all derivatives required to calculate the matrix $\hat{\boldsymbol{A}}$.

EKF Simulation

Figures 4.5, 4.6 and 4.7 show the simulation results where the extended Kalman Filter is used to estimate the state vector used by the controller to stabilise the spacecraft from an initial attitude condition to a nadir pointing attitude. Data utilised in the simulation are shown in Table 4.2.

Initial attitude	$[\Phi, \quad \Theta, \quad \Psi] = [15, \quad 25, \quad 15]\,[deg]$
Initial angular rates	$[\omega_x, \quad \omega_y, \quad \omega_z] = [0.01, \quad 0.0011, \quad 0.01]\,[deg/s]$
Rates	Control rate = 0.5 s, Sampling rate = 0.1 s
Kalman Filter gains	$\Sigma_w = \text{diag}[5, \quad 5, \quad 5 \;\; 5, \quad 5, \quad 5, \quad 100, \quad 100, \quad 100]10^{-5}$
	$\Sigma_{\bar{x},0} = \text{diag}[1, \quad 1, \quad 1 \;\; 1, \quad 1, \quad 1, \quad 10, \quad 10, \quad 10]10^{-5}$
	$\Sigma_v = \text{diag}[0.1, \quad 0.1, \quad 0.1 \;\; 0.005, \quad 0.005, \quad 0.005]$

Table 4.2: *Extended Kalman filter simulation setup*
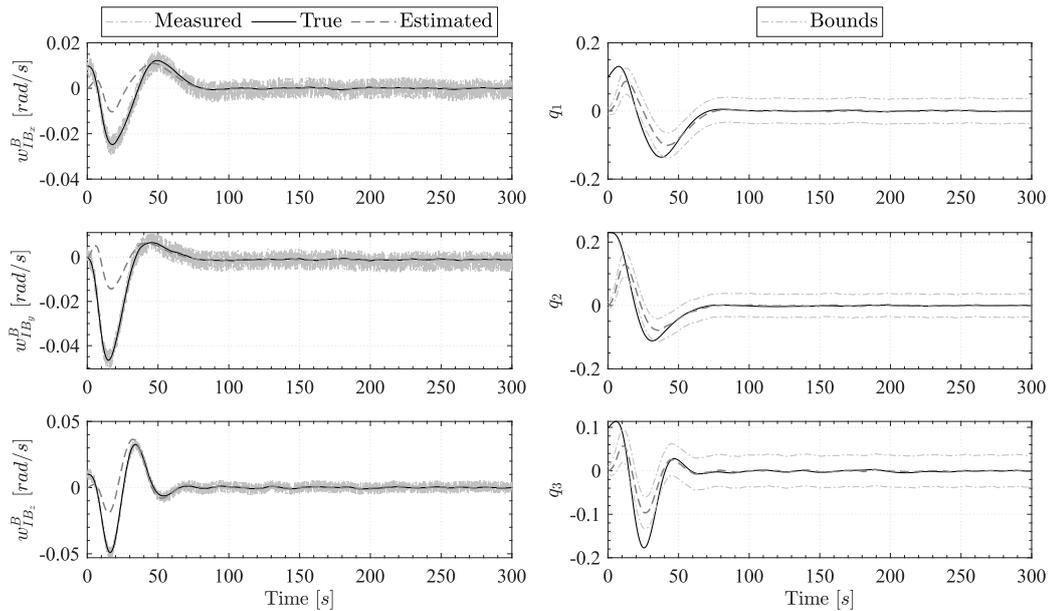
Figure 4.5: *Quaternions and spacecraft angular velocities resulting from the EKF simulation. The bounds on the right represent the 3σ bounds*

Figures 4.5 and 4.6 show that after a short initial transient of about 50 seconds, the filter converges to true values. The gyroscope bias is also estimated correctly, resulting in a better angular velocity measurement. From Figure 4.5, particularly when considering the second angular velocity component, it can be seen that the measurements are affected by bias, but this does not affect the estimation.
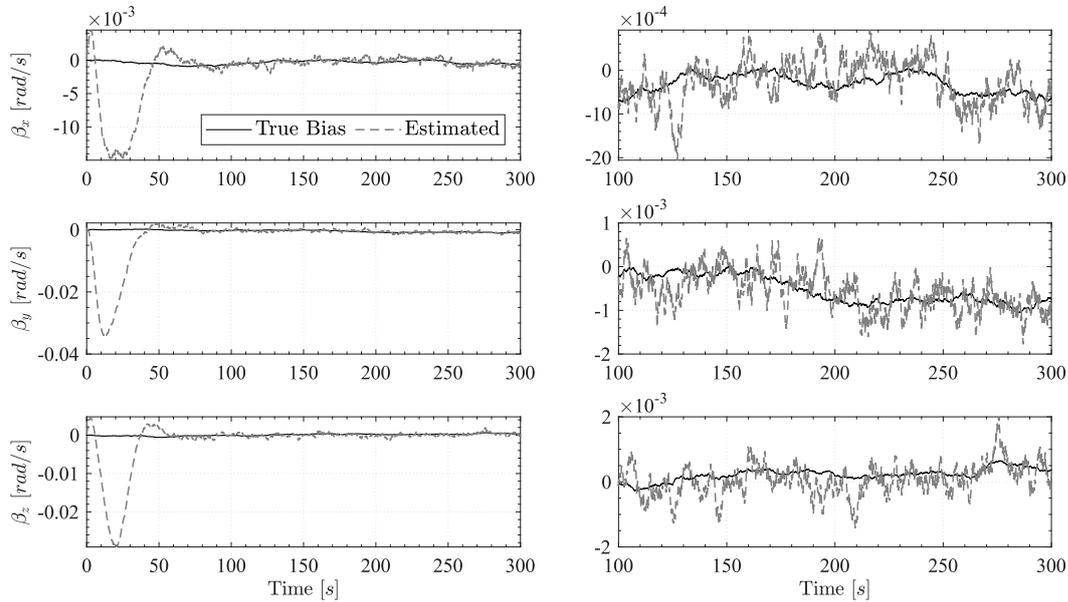
Figure 4.6: *True and estimated gyroscopes bias resulting from the EKF simulation. The diagram on the right shows a focus*

Diagram 4.7 highlights the errors between the EKF estimation and the true values in terms of Euler angles. Also shown are the boundaries indicating the values that should not be exceeded to meet the Absolute Knowledge Error requirement discussed in Section 3.2.1. It is easy to see that the requirement is successfully met.
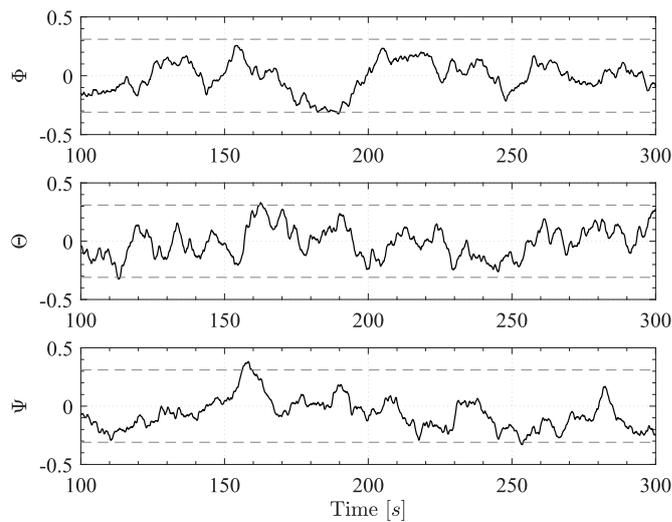
Figure 4.7: *Euler errors resulting from the EKF simulation. Dashed lines represent the AKE requirement shown in Table 3.8*

4.2.3 Kalman Rate Estimator

After being released from the P-POD, the satellite holds angular rates that need to be damped before it can perform mission-related operations. In fact, during the detumbling phase, a controller will be active to drive the satellite's angular rates below a certain threshold. This control, however, can only be performed if a measurement of the satellite's angular rates is available.

The Kalman Rate Estimator (KRE) is a robust method that can be used to derive the satellite's angular rates directly without employing gyroscopes but only by using attitude information. The filter is discussed in [38] where it is implemented for the satellite Sunsat.

The body measured rate of change of the geomagnetic field vector direction can be used as a measurement input for this rate estimator. However, is not inertially fixed as it rotates twice per polar orbit so an expected average error in the measured, orbit referenced body angular rate components will be ω_0 in the orbit normal direction. Another option is to use sun vector measurements, obtained using sun sensors, to estimate the rate vector. The sun vector only rotates inertially once per year and so represent an inertial reference. In this work, magnetometer measurements are supplied to the rate estimator.

System Model

The dynamic equation of motion can be written as

$$\dot{\bar{\omega}}_{IB}^{B} = \boldsymbol{I}^{-1}(\bar{T}_{MT}^{B} + \bar{T}_{gg}^{B} - \bar{\omega}_{IB}^{B} \times \boldsymbol{I}\bar{\omega}_{IB}^{B}) \tag{4.30}$$

Using equation 4.11 and assuming a near circular orbit, Equation 4.30 can be written in terms of the orbit referenced angular rates

$$\dot{\bar{\omega}}_{OB}^{B} = \boldsymbol{I}^{-1}(\bar{T}_{MT}^{B} + \bar{T}_{gg}^{B} - \bar{\omega}_{IB}^{B} \times \boldsymbol{I}\bar{\omega}_{IB}^{B}) - \dot{\boldsymbol{A}}_{O}^{B}\bar{\omega}_{IO}^{O} \tag{4.31}$$

The last term in the equation is typically in the same order of magnitude as the disturbance torque term. It can therefore be modeled with the disturbance torque as system noise [38]. The continuous time model is given by

$$\dot{\bar{x}}(t) = \boldsymbol{F} + \boldsymbol{G}\bar{u}(t) + \bar{s}(t) \tag{4.32}$$

with

$$\begin{aligned}
\boldsymbol{F} &= [\boldsymbol{0}], \quad \boldsymbol{G} = \boldsymbol{I}^{-1}, \quad \bar{u}(t) = \bar{T}_{MT}^{B}, \\
\bar{s}(t) &= (\bar{T}_{gg}^{B} - \bar{\omega}_{IB}^{B} \times \boldsymbol{I}\bar{\omega}_{IB}^{B}) - \dot{\boldsymbol{A}}_{O}^{B}\bar{\omega}_{IO}^{O}
\end{aligned} \tag{4.33}$$

The discrete system model, with sampling period T_s, will then be

$$\bar{x}(k+1) = \boldsymbol{\Phi}\bar{x}(k) + \boldsymbol{\Gamma}\bar{u}(k) + \bar{s}(k) \tag{4.34}$$

with

$$\begin{aligned}
\boldsymbol{\Phi} &= [\boldsymbol{\mathcal{I}}_3], \quad \boldsymbol{\Gamma} = \boldsymbol{I}^{-1}T_s, \\
\bar{s}(k) &= N\{0, Q(k)\}
\end{aligned} \tag{4.35}$$

where $\bar{s}(k)$ is a zero mean system noise vector with covariance $\boldsymbol{Q}$.

Measurement Model

The measurements for the Kalman Filter are obtained by using the small-angle approximation of the attitude matrix between two successive sampling instances. If almost constant angular rates during the sampling period are assumed then the rotation matrix can be approximated as

$$\Delta\boldsymbol{A} \approx \begin{bmatrix} 1 & \omega_{Oz}(k)T_s & -\omega_{Oy}(k)T_s \\ -\omega_{Oz}(k)T_s & 0 & \omega_{Ox}(k)T_s \\ \omega_{Oy}(k)T_s & -\omega_{Ox}(k)T_s & 0 \end{bmatrix} \approx \boldsymbol{\mathcal{I}_3} + \boldsymbol{\Lambda}\{\bar{\omega}_{OB}^B\} \tag{4.36}$$

An orbit referenced vector in body coordinates can then be updated between sampling periods as

$$\bar{v}(k) = \Delta\boldsymbol{A}\bar{v}(k-1) \tag{4.37}$$

The Kalman Filter measurement model then becomes

$$\Delta\bar{v}(k) = \bar{v}(k) - \bar{v}(k-1) = \boldsymbol{\Lambda}\{\bar{\omega}_{OB}^B\}\bar{v}(k-1) \tag{4.38}$$

Therefore

$$\bar{y}(k) = \Delta\bar{v}(k) = \boldsymbol{H}(k)\bar{x}(k) + \bar{m}(k) \tag{4.39}$$

where

$$\boldsymbol{H}(k) = \begin{bmatrix} 0 & -v_z(k-1)T_s & v_y(k-1)T_s \\ v_z(k-1)T_s & 0 & -v_x(k-1)T_s \\ -v_y(k-1)T_s & v_x(k-1)T_s & 0 \end{bmatrix} \tag{4.40}$$

and $\bar{m}(k) = N\{0, \boldsymbol{R}(k)\}$ as zero measurement noise, with covariance $\boldsymbol{R}$.

KRE implementation

In this case, a different notation is used from the one used previously in the EKF, in particular, $\boldsymbol{Q}$ denotes the covariance matrix of the process noise, $\boldsymbol{R}$ the covariance matrix of the measurement noise, and $\boldsymbol{P}$ the state covariance matrix. Table 4.3 shows the steps to be executed every sampling period T_s.

---

1. **State prediction**

   $$\hat{x}_{k+1}^- = \hat{x}_k^+ + 0.5T_s(3\Delta\bar{x}_k - \Delta\bar{x}_{k-1})$$

   $$\Delta\bar{x}_k = \boldsymbol{I}^{-1}(\bar{T}_{MT}^B(k) - \hat{\omega}_{OB}^B(k) \times \boldsymbol{I}\hat{\omega}_{OB}^B(k)))$$

2. **State covariance matrix propagation**

   $$\boldsymbol{P}_{k+1}^- = \boldsymbol{P}_k^+ + \boldsymbol{Q}$$

3. **Gain update**

   $$\boldsymbol{K}_{k+1} = \boldsymbol{P}_{k+1}^-\boldsymbol{H}_{k+1}^T\big[\boldsymbol{H}_{k+1}\boldsymbol{P}_{k+1}^-\boldsymbol{H}_{k+1}^T + \boldsymbol{R}\big]^{-1}$$

4. **System state update**

   $$\hat{x}_{k+1}^+ = \hat{x}_{k+1}^- + \boldsymbol{K}_{k+1}(\bar{y}_{k+1} - \boldsymbol{H}_{k+1}\hat{x}_{k+1}^-)$$

   $$\bar{y}_{k+1} = \bar{v}(k+1) - \bar{v}(k)$$

5. **State covariance matrix update**

   $$\boldsymbol{P}_{k+1}^+ = \big[\boldsymbol{\mathcal{I}_3} + \boldsymbol{K_{k+1}}\boldsymbol{H_{k+1}}\big]\boldsymbol{P}_{k+1}^-$$

Table 4.3: *Summary of the steps of the Kalman Rate Estimator (KRE)*

KRE Simulation

The performance of the Kalman Rate estimator was analysed through a simulation, whose results are shown in Figure 4.8. In the simulation, the spacecraft is in an uncontrolled tumbling state resulting from the initial conditions shown in Table 4.4.

Initial attitude	$[\Phi, \quad \Theta, \quad \Psi] = [25, \quad 35, \quad 15] \, [deg]$
Initial angular rates	$[\omega_x, \quad \omega_y, \quad \omega_z] = [4, \quad -2, \quad 3] \, [deg/s]$
Rates	Control rate = -, Sampling rate = 0.1 s
Kalman Rate gains	$\Sigma_w = \text{diag}[1, \quad 1, \quad 1]10^{-4}$
	$\Sigma_v = \text{diag}[1, \quad 1, \quad 1]10^{-3}$
	$\Sigma_{\bar{x},0} = \text{diag}[0.1, \quad 0.1, \quad 0.1]$

Table 4.4: *Kalman Rate Estimator simulation setup*

The sensor noise data used for the simulation are those discussed in Section 6.1.1 and shown in Table 6.1.

From Figure 4.8, it can be seen that the filter converges after approximately 200 seconds from the start of the simulation. The steady-state performance is satisfactory as the estimation error remains below 0.005 rad/s.
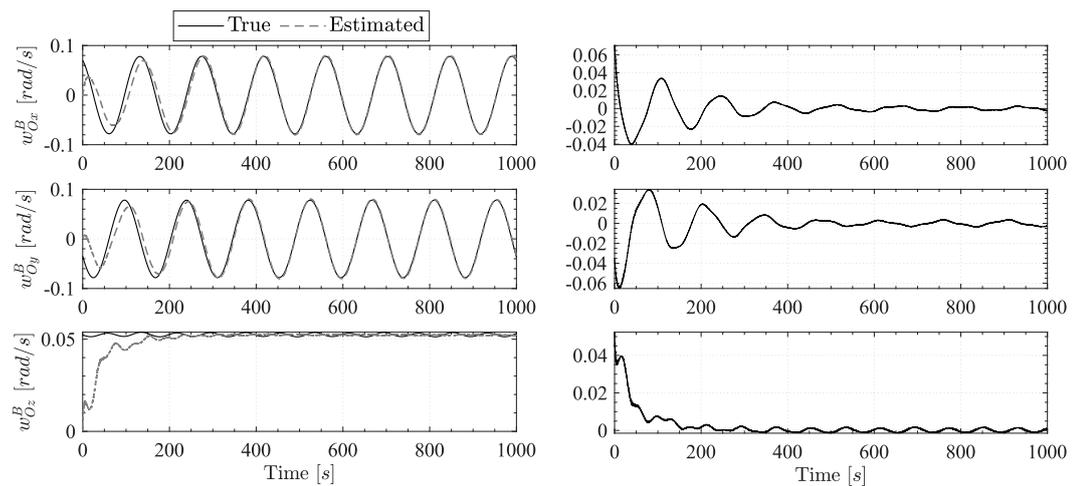
Figure 4.8: *Spacecraft angular velocities resulting from the KRE simulation. The diagrams on the right show the error defined by the difference between the true and estimated value*

Chapter 5

Attitude Control

This chapter presents the theoretical background of the controllers implemented for the different control modes. Only some points will be discussed in detail, while for the detailed discussion of others, references will be made to the appropriate sources.

When it comes to attitude control, the goal is the synthesis of a feedback command torque, which is capable of forcing the kinematic variables to track their respective target/reference trajectories. In general the problem of controlling a system has been studied for years. The research involving the application of control theories to the control of satellites has yielded many good results, leading to various control techniques that differ in aspects as the variables used to perform the control or the performances they offer. As mentioned in Section 1.5.2, some types of controllers have been consolidated over the years, earning the podium of the most commonly used controllers. However, although the latter is less complex and simple to implement, they sometimes fail to guarantee the performance needed to meet requirements. For this reason, research focuses on advanced control methods that manage to bridge this gap. Some of these modern control techniques are the following.

Optimal Control it is a particular control technique in which the control signal optimizes a certain cost index. Two optimal control design methods have been widely used in industrial applications. These are Linear-Quadratic-Gaussian control (LQG) and Model Predictive Control (MPC);

Robust Control is used when uncertainties on the spacecraft parameters (mass, inertia, etc...), disturbances, and noise are unknown. A modern example of a robust control technique is H-infinity (H_∞);

Adaptive Control is the control method used by a controller which must adapt to a controlled system with parameters which vary, or are initially uncertain [19];

Intelligent Control uses various AI computing approaches like neural networks, fuzzy logic, machine learning, evolutionary computation and genetic algorithms to control the spacecraft dynamics.

Within this thesis work, Model Predictive Control and H_∞ Control are implemented as will be discussed below.

To verify the performances of a controller the guidelines given in [4] and discussed in [3, 37] are followed. In particular, the performance elements of a control system are quantified through suitable mathematical indicators and can be divided into intrinsic and extrinsic.

74

Some of extrinsic performance indicators have already been introduced in Section 3.2.1. The definition of other extrinsic performance indicators, that will be calculated for the controllers, is given in Table 5.1.

Absolute Performance Error	**APE**	is the instantaneous values of the performance error and indicates the difference between the target (desired) output of the system and the actual achieved output
Mean Performance Error	**MPE**	is the mean value of the performance error over a specified time interval
Relative Performance Error	**RPE**	is the difference between the instantaneous performance error at a given time, and its mean value over a time interval containing that time
Performance Stability Error	**PSE**	is defined as the difference between the instantaneous performance error at a given time t and the error value at an earlier time $t - \delta t$
Absolute knowledge Error	**AKE**	is defined as the instantaneous value of the knowledge error and indicates the difference between the known (estimated) output of the system and the actual achieved output
Mean Knowledge Error	**MKE**	is defined as the mean value of the knowledge error over a specified time interval
Relative Knowledge Error	**RKE**	is defined as the difference between the instantaneous knowledge error at a given time, and its mean value over a time interval containing that time

Table 5.1: *Extrinsic performance indicators, from [3, 4, 37]*

Simulations will be carried out to verify whether the extrinsic performance indicators meet the requirements and to validate the algorithms.

Regarding intrinsic indicators, it is important to consider

Stability is an intrinsic property of the system and refers to the effects on state motion caused by perturbations on the initial state. The system is said to be stable if its evolution is insensitive to perturbations on the initial state and remains indefinitely around the equilibrium point or trajectory;

Robustness is the property of a controlled system to achieve the control objectives against the disturbances and uncertainties;

Controllability describes the possibilities of transferring the system's state to a particular predefined final state by appropriately acting on the control torque;

Observability describes the possibility of estimating the initial state of the system by measuring the output and control torque over a given time interval.

As with the extrinsic indicators, these properties will be evaluated and used as an index for comparison.

5.1 Magnetic Control Law

Together with active attitude stabilization and disturbance rejection, the attitude control system uses magnetic torquers to dump excess momentum of the reaction wheels and detumbles the spacecraft immediately after launch.

Magnetic torquers use the Earth's magnetic field to produce a torque, which is given by

$$\bar{T}_{MT} = \bar{m} \times \bar{B} \tag{5.1}$$

where $\bar{m}$ is the commanded dipole moment generated by the torquers and $\bar{B}$ is the local geomagnetic field expressed in body-frame coordinates. As can be seen from Equation 5.1 the torques are constrained to lie in a two-dimensional plane orthogonal to the magnetic field. However, 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 [31, 16].

The control laws used for detumbling and wheels momentum dumping differ in the definition of the dipole moment $\bar{m}$.

5.1.1 Detumbling

The magnetic control law that will be used to detumble the spacecraft is an alternative version of the well-known B-dot controller. Detailed theory behind this control law can be found in [13]. The controller is characterised by the following dipole moment law

$$\bar{m} = \frac{k}{\|\bar{B}\|} \bar{\omega} \times \hat{b} \tag{5.2}$$

where $\bar{B}$ is the magnetic field, $\hat{b} = \bar{B}/\|\bar{B}\|$, $\bar{\omega}$ is the spacecraft angular velocity, and k is a positive scalar gain.

This gives a control torque

$$\bar{T}_{MT} = -k(\boldsymbol{\mathcal{I}}_3 - \hat{b}\,\hat{b}^T)\,\bar{\omega} \tag{5.3}$$

As shown in [13], the stability of this control law can be proven using the Lyapunov stability theorems. The rotational Kinetic energy is chosen as the candidate Lyapunov function

$$V = \frac{1}{2}\bar{\omega}^T \boldsymbol{I} \bar{\omega} \tag{5.4}$$

It can be shown that $\dot{V}$ is only negative semi-definite because when $\bar{\omega}$ is parallel to $\hat{b}$ then $\dot{V} = 0$. However, this is not a concern in practice as discussed in [13].

Always in [13] a gain expression it is provided by analysing the closed-loop dynamics of the component $\bar{\omega}$ perpendicular to the Earth's magnetic field

$$k = \frac{4\pi}{T_{orb}}(1 + \sin \xi_m)I_{min} \tag{5.5}$$

where T_{orb} is the orbital period, ξ_m is the inclination of the spacecraft orbit relative to the geomagnetic equatorial plane and I_{min} is the minimum principal moment of inertia.

Detumbling Simulation

Figures 5.2, 5.3 and 5.4 show the results of a detumbling simulation. The simulation has been performed to verify the working of the controller. For this reason, neither measurement noise nor estimators were simulated. The initial conditions and orbit characteristics in the simulation are shown in Table 5.2. Figure 5.1 shows the orbit used in the simulation.
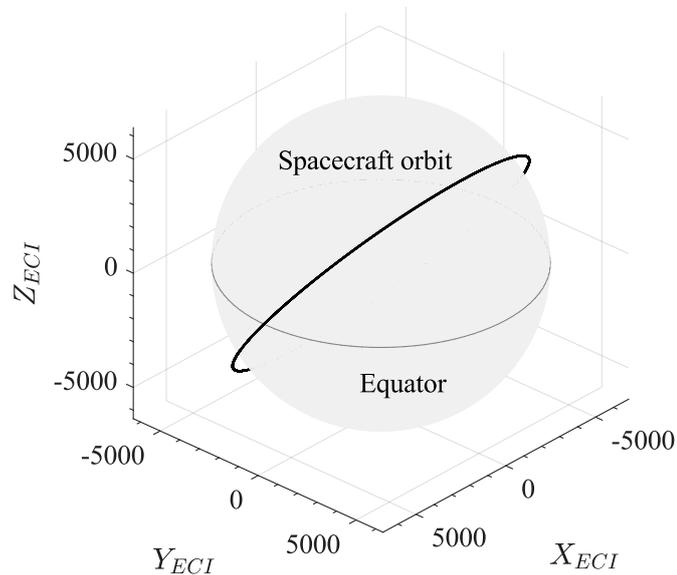
Figure 5.1: *Spacecraft orbit used in the detumbling simulation*

Orbit characteristics	Circular orbit, Altitude = 470 km, Inclination = 51.6°
Initial attitude	$[\Phi,\quad \Theta,\quad \Psi] = [5,\quad 5,\quad 5]\ [deg]$
Initial angular rates	$[\omega_x,\quad \omega_y,\quad \omega_z] = [10,\quad 10,\quad 10]\ [deg/s]$
Rates	Control rate = 5 s, Sampling rate = 0.5 s
Constraints	Maximum dipole moment = 0.5 Am^2

Table 5.2: *Detumbling simulation setup*
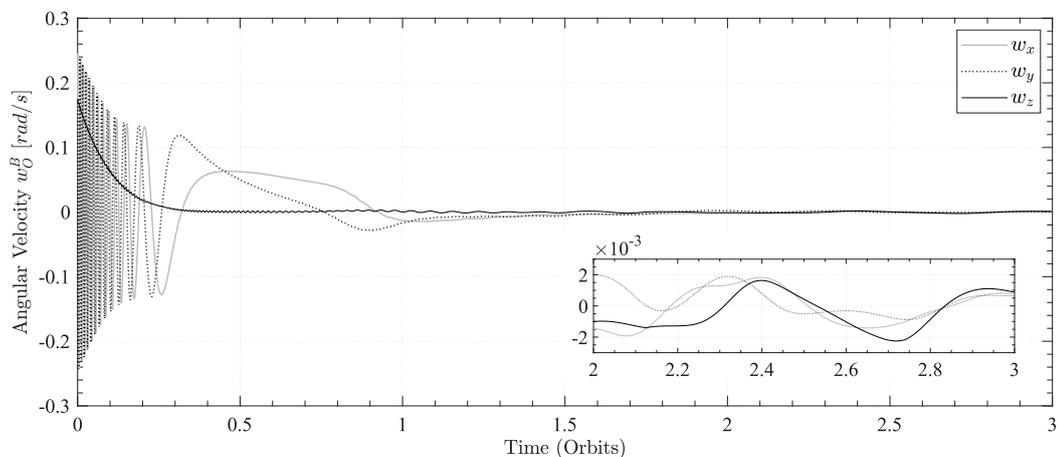
Figure 5.2: *Orbit referenced angular velocities of the spacecraft during the detumbling simulation. A focus of the third orbit is shown in the figure to show the detail of the angular velocity value*
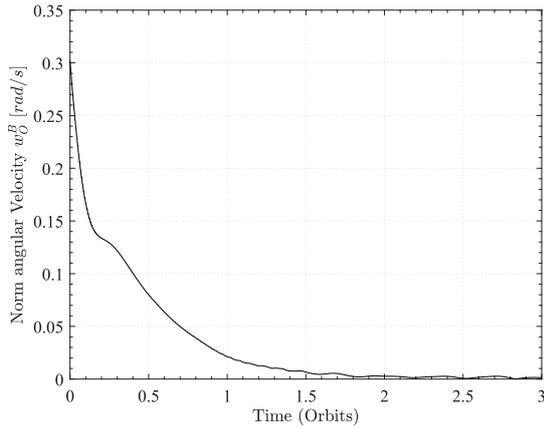
Figure 5.3: *Norm of angular velocities*
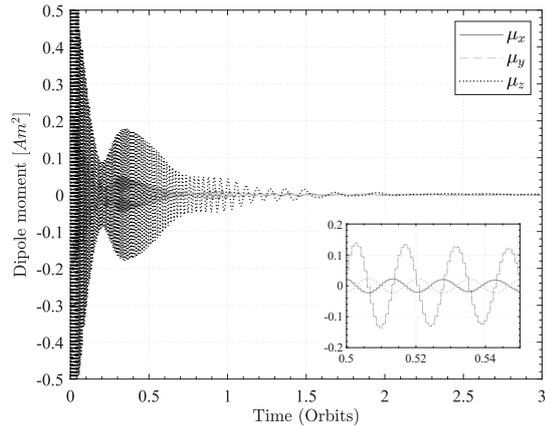
Figure 5.4: *Dipole moment*

From Figure 5.2, it can be seen that in the ideal case the spacecraft is detumbled in 2 orbits since the angular velocities are already below the imposed requirement given in Table 3.8. However, this is the ideal case, in the non-ideal case with disturbances three orbits will be needed to fulfil the requirement. A focus of the control signal was included in Figure 5.4 to highlight the stepwise nature due to ZOH.

5.1.2 Momentum Dumping

As mentioned earlier, another important use of magnetic torquers for low-Earth orbiting spacecraft is momentum dumping. In Section 3.2.5, the difference between cyclic and secular torques was discussed. While cyclic torques result in a cyclic variation of reaction wheels speed, secular torques result in a linear increase because the wheel constantly absorbs angular momentum. Eventually saturation of the wheels will occur due to the excess momentum, which can only be dumped through external torques.

A common approach to design a magnetic torquer control law for momentum dumping is to command a magnetic dipole moment [18, 31]

$$\bar{m} = \frac{k}{\|\bar{B}\|}\bar{h} \times \hat{b} \tag{5.6}$$

which is similar to Equation 5.2 but with the angular velocity vector replaced by the wheel angular momentum.

Momentum Dumping Simulation

Here again, an ideal simulation was carried out to verify the correct desaturation of the reaction wheels. The simulation starts from a nadir pointing condition with unit quaternion and zeros angular velocities, with a certain angular momentum stored in the reaction wheels defined by the random choice of the initial reaction wheel velocities. The momentum dumping magnetic controller is implemented simultaneously with the nadir pointing controller to desaturate the wheels almost entirely in one orbit. The orbit characteristics are the same as those shown in Table 5.2. The simulation setup is summarised in Table 5.3.

The controller gain indicated by k in Equation 5.6 determines how fast the wheels will be desaturated. However, as discussed in Section 3.2.5, this desaturation speed is limited by the maximum dipole moment deliverable by the magnetic torquers. The gain chosen is such that the momentum dumping is spread within an orbit.

Initial attitude	$[\Phi, \quad \Theta, \quad \Psi] = [0, \quad 0, \quad 0] \, [deg]$
Initial angular rates	$[\omega_x, \quad \omega_y, \quad \omega_z] = [0, \quad 0, \quad 0] \, [deg/s]$
Initial RWs* angular rates	$[\omega_1, \quad \omega_2, \quad \omega_3, \quad \omega_4] = [1000, \quad 1500, \quad -500 \quad -500] \, [rpm]$
Rates	Control rate = 1 s, Sampling rate = 1 s
Constraints	Maximum dipole moment = 0.5 Am^2
Controller gain	$k = \text{diag}[50, \quad 50, \quad 50]$

* Reaction wheels

Table 5.3: *Momentum dumping simulation setup*

Results are shown in diagrams 5.5, 5.6, 5.7, 5.8, 5.9 and 5.10. As represented in Figures 5.5 and 5.6, during momentum dumping the spacecraft maintains nadir pointing, which means that the torques applied by the magnetic torquers are opposed by the reaction wheels and are such that the angular momentum of the wheels decreases over time. This can be seen in Figure 5.9 which shows how the wheel momenta approach zero thanks to the execution of the momentum dumping control law.
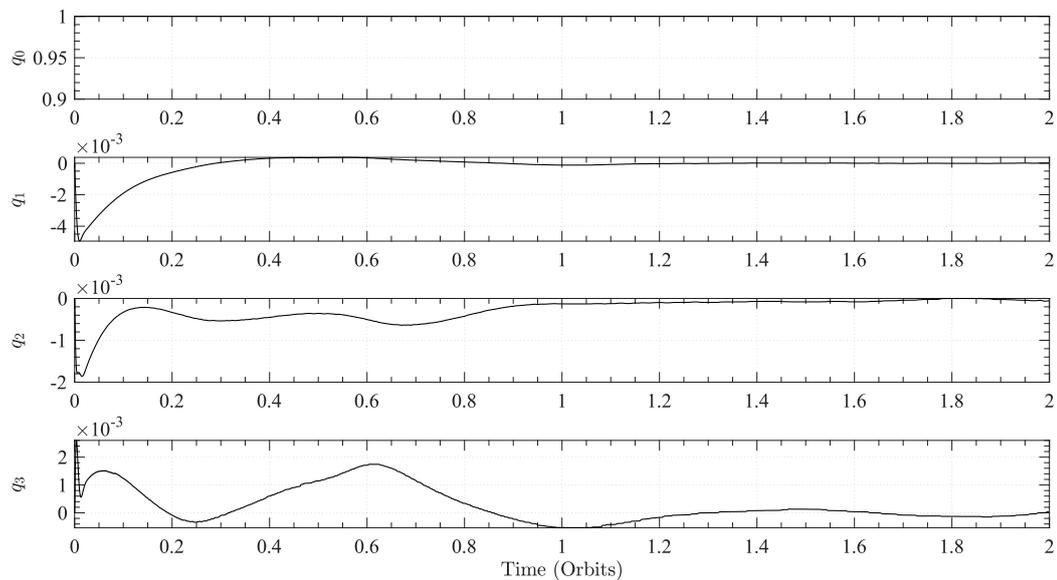
Figure 5.5: *Quaternions resulting from the reaction wheels momentum dumping simulation*
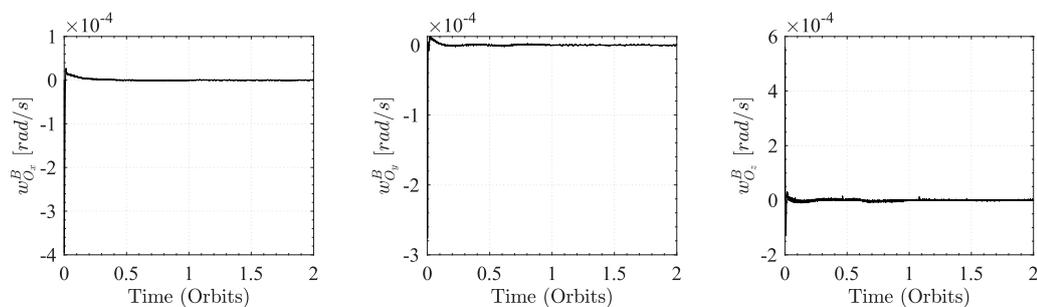
Figure 5.6: *Spacecraft angular velocities resulting from the reaction wheels momentum dumping simulation*

From Figure 5.7, it can be seen that the angular momentum is almost totally damped in one orbit as intended. The chosen gain results in no costly desaturation manoeuvres in terms of applied control torques. Indeed, Figure 5.8 shows that the dipole moment assumes small values during the simulation without ever reaching the maximum value indicated in Table 5.3.
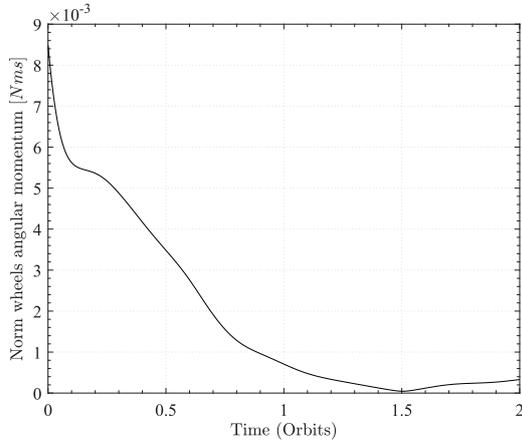
Figure 5.7: *Angular momentum magnitude*
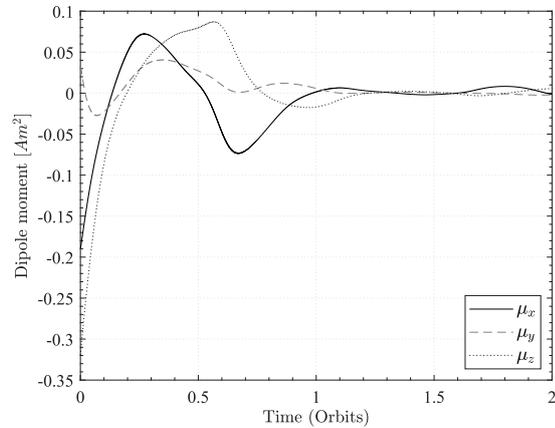
Figure 5.8: *Dipole moment*

Although the reaction wheels model is discussed in Section 6.1.2, diagrams showing their angular velocities and accelerations are also shown for completeness. There are four reaction wheels and they are arranged in the pyramidal configuration previously discussed. The angular velocity and inertia of the wheels determine the angular momentum they store. The technological limit of maximum permissible wheel speed defines a maximal storable angular momentum. Once the maximum speed limit is reached, desaturation is necessary to continue operating the wheel. In practice, efforts are made to avoid approaching this limit and not to reduce the angular momentum to zero to prevent problems with the motors that drive the wheels.
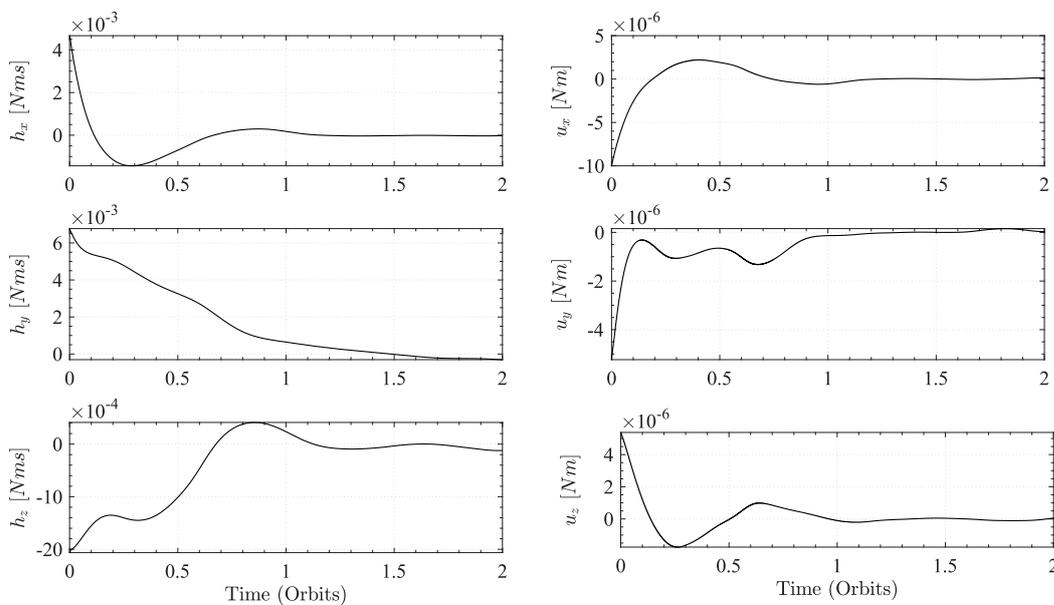
Figure 5.9: *Angular momentum and magnetic torques resulting from the reaction wheels momentum dumping simulation. On the left angular momentum, on the right magnetic torques*

From Figure 5.10, it can be seen that during the simulation, desaturation occurred correctly as the angular velocity of the wheels, and therefore the angular momentum decreases over time.

The wheels acceleration shown in Figure 5.10 is closely related to the control torques applied by the nadir pointing controller. This torque is not shown here, but it is equal to and opposite to that shown in Figure 5.9.
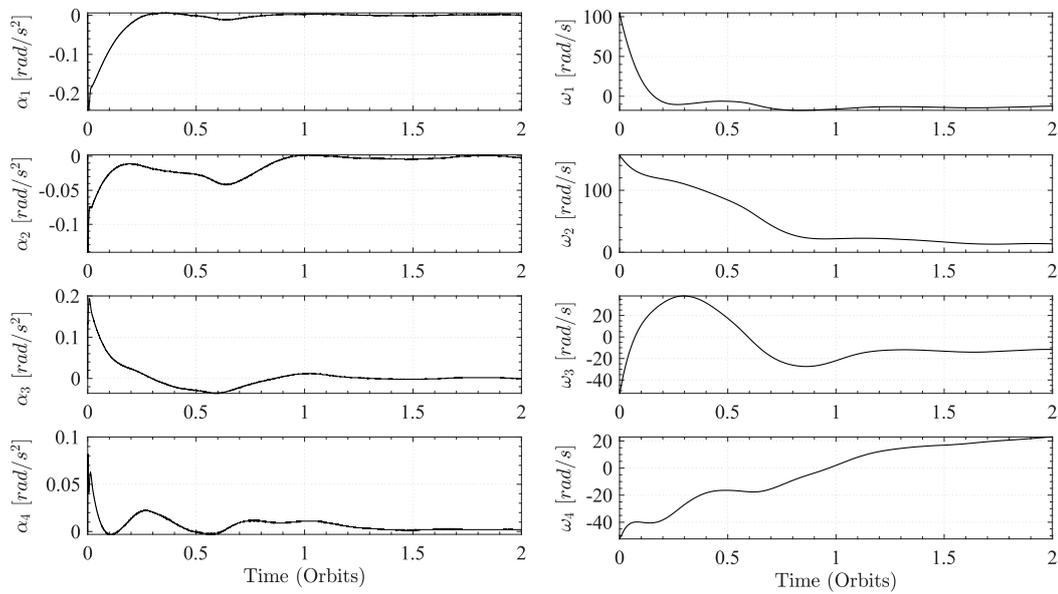
Figure 5.10: *Reaction wheels angular rates and acceleration resulting from the reaction wheels momentum dumping simulation*

5.2   Model Predictive Control

The Model Predictive Control (MPC) is a method of designing feedback control systems. By definition MPC uses an explicit dynamic plant model to predict the effect of future reactions of the manipulated variables on the output and the control signal obtained by minimising a pre-defined cost function.

There are key principles required

- An internal dynamic model used to predict what might happen over a future *prediction horizon*

- Calculation of a control sequence minimising a certain objective function

- Moving prediction horizon window

- Receding horizon control that only executes the next step

These principles and the methodology of a model predictive controller are characterised by the strategy represented in Figure 5.11.
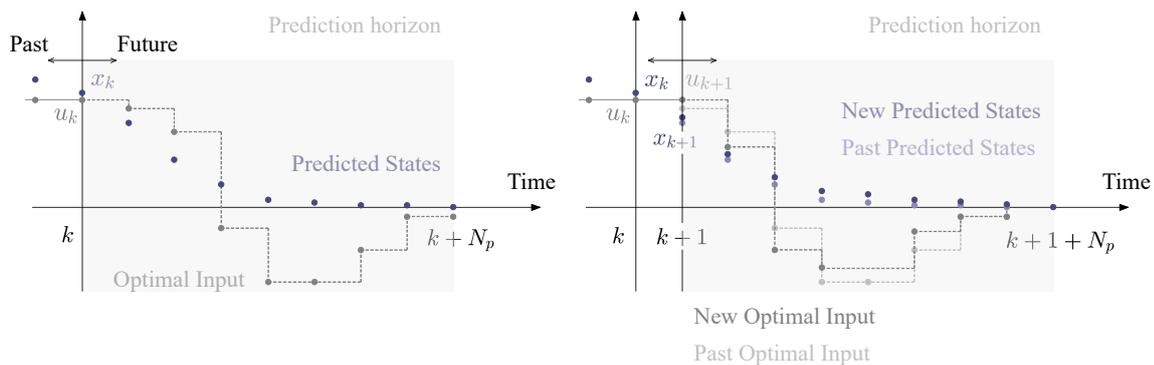
Figure 5.11: *Example of Model Predictive Control strategy. On the left, the current timestep is k, while on the right it is $k + 1$.*

The future outputs, or states, for a determined horizon window N_p, called the prediction horizon, are predicted using the dynamic model of the plant. These predicted states depend on the known values up to the time instant k and on the future control signals. The latter is calculated by optimising a determined objective function, over a prediction horizon N_p, to force the states to track the reference trajectory with a control horizon N_c. Only the first control signal is applied to the plant according to the logic of receding horizon control. The other inputs are rejected as in the next sampling instant the optimum sequence is recalculated incorporating the new measurements available up to time-step $k + 1$.

Three popular approaches to system modeling for predictive control design are

- Finite Impulse Response (FIR)

- Transfer Function (TF)

- State-Space (SS)

Each approach exhibits certain advantages and disadvantages, the development in this work focuses on state space models.

5.2.1 State-Space Model

Model predictive control systems are designed based on a mathematical model of the plant. The mathematical model chosen is the state space model in which the variables representing the system's information are contained in the so-called state vector. The selected variables in this work are quaternions for attitude and angular velocities. The relationships describing the kinematics and attitude dynamics are non-linear, so they cannot be written directly in the state space form. Linearisation of the mathematical model is, therefore, necessary before it can be used in the model predictive control strategy.

The linearisation of the system mathematical model is performed around the equilibrium point, which in the considered case will be the nadir pointing. In other words, unit quaternion approximation can be assumed. In Appendix A, the linearised system is derived from these assumptions. The state space model is given by

$$\dot{\bar{x}} = A\bar{x} + B\bar{u} \tag{5.7}$$

where

$$A = \begin{bmatrix} 0 & 0 & 0 & 1/2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/2 \\ -8\omega_0^2\frac{I_y-I_z}{I_x} & 0 & 0 & 0 & 0 & \omega_0\frac{I_z+I_x-I_y}{I_x} \\ 0 & -6\omega_0^2\frac{I_x-I_z}{I_y} & 0 & 0 & 0 & 0 \\ 0 & 0 & -2\omega_0^2\frac{I_y-I_x}{I_z} & -\omega_0\frac{I_z+I_x-I_y}{I_z} & 0 & 0 \end{bmatrix} \tag{5.8}$$

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1/I_x & 0 & 0 \\ 0 & 1/I_y & 0 \\ 0 & 0 & 1/I_z \end{bmatrix} \tag{5.9}$$

Modern control system are typically implemented on a digital processor. The processor samples the sensors at discrete sampling instants and computes the corresponding control torques. The control torque is held constant according to the control frequency, thus following the logic of the so-called zero-order hold (ZOH) control implementation. For this reason, the controllers developed in this thesis work are implemented in discrete time.

The discrete version of the state space model is given by

$$\bar{x}(k+1) = A_d\bar{x}(k) + B_d\bar{u}(k) \tag{5.10}$$

where A_d, and B_d can be calculated as

$$A_d = e^{AT_s} \tag{5.11}$$

$$B_d = \int_0^{T_s} e^{A\sigma} B d\sigma \tag{5.12}$$

in the equations T_s is the control sampling time which is the inverse of the control frequency.

Assuming that a measure of all state variables is available, the output equation can be written as

$$\bar{y}(k) = \boldsymbol{C}_d\,\bar{x}(k) \tag{5.13}$$

where

$$\boldsymbol{C}_d = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.14}$$

With the matrices $\boldsymbol{A}_d$, $\boldsymbol{B}_d$, and $\boldsymbol{C}_d$ representing the mathematical model of the system, it is possible to evaluate the previously defined intrinsic indicators. To determine if the system is stable, it is necessary to check whether the eigenvalues of matrix $\boldsymbol{A}_d$ lie within the unit disc in the complex plane. Instead, verification of controllability and observability involves calculating two matrices and checking their rank. If the latter is maximum, then the properties are satisfied in both cases, otherwise the system has non-controllable/observable modes.

Studying these properties reveals that the system is completely controllable and observable. On the other hand, the system's internal stability property is lacking since some eigenvalues lie outside the unit disc. However, since the modes are all controllable, the plant is stabilisable, i.e. it is possible to realise a controller that stabilises the closed-loop system.

Embedded Integrator

To eliminate steady-state error from a control system it is necessary to include integral action. In [30, 43] the state space model with embedded integrator is derived.

For model predictive controllers, a convenient way to introduce integral action is by modifying the state-space model. By denoting the model with the subscript m, the discrete version of Equation 5.7 can be written as follows

$$\bar{x}_m(k + 1) = \boldsymbol{A}_m\bar{x}_m(k) + \boldsymbol{B}_m\bar{u}(k) \tag{5.15}$$

where $\boldsymbol{A}_m$ and $\boldsymbol{B}_m$ are $\boldsymbol{A}_d$ and $\boldsymbol{B}_d$ and it is assumed that the matrices are time invariant. The system considered is MIMO* so the output equation is a vector equation which is given by

$$\bar{y}_m(k) = \boldsymbol{C}_m\bar{x}_m(k) \tag{5.16}$$

Denoting by $n = 6$ the number of states, by $m = 3$ the number of inputs, and by $q = 6$ the number of outputs, it follows that

$$\begin{aligned} \boldsymbol{A}_m \text{ is a } n \times n \text{ matrix} \\ \boldsymbol{B}_m \text{ is a } n \times m \text{ matrix} \\ \boldsymbol{C}_m \text{ is a } q \times n \text{ matrix} \end{aligned} \tag{5.17}$$

Taking a difference operation on both sides of the state equation follows

$$\bar{x}_m(k + 1) - \bar{x}_m(k) = \boldsymbol{A}_m(\bar{x}_m(k) - \bar{x}_m(k - 1)) + \boldsymbol{B}_m(\bar{u}(k) - \bar{u}(k - 1)) \tag{5.18}$$

Denoting the differences as

---

*Multiple Input Multiple Output (MIMO)

$$\Delta \bar{x}_m(k+1) = \bar{x}_m(k+1) - \bar{x}_m(k)$$
$$\Delta \bar{x}_m(k) = \bar{x}_m(k) - \bar{x}_m(k-1) \tag{5.19}$$
$$\Delta \bar{u}(k) = \bar{u}(k) - \bar{u}(k-1)$$

Therefore, the difference of the state equation is

$$\Delta \bar{x}_m(k+1) = \boldsymbol{A}_m \Delta \bar{x}_m(k) + \boldsymbol{B}_m \Delta \bar{u}(k) \tag{5.20}$$

which is a standard discrete time evolution of a steady equation in terms of finite differences instead of state vectors.

A new augmented state vector is defined as

$$\bar{x}(k) = \begin{bmatrix} \Delta \bar{x}_m(k) \\ \bar{y}(k) \end{bmatrix} \tag{5.21}$$

Considering the difference equation for $\bar{y}$

$$\bar{y}(k+1) - \bar{y}(k) = \boldsymbol{C}_m(\bar{x}_m(k+1) - \bar{x}_m(k)) = \boldsymbol{C}_m \Delta \bar{x}_m(k+1) =$$
$$= \boldsymbol{C}_m \boldsymbol{A}_m \Delta \bar{x}_m(k) + \boldsymbol{C}_m \boldsymbol{B}_m \Delta \bar{u}(k) \tag{5.22}$$

Putting expression 5.22 together with 5.20 it is possible to define the augmented model

$$\begin{bmatrix} \Delta \bar{x}_m(k+1) \\ \bar{y}(k+1) \end{bmatrix} = \begin{bmatrix} \boldsymbol{A}_m & \boldsymbol{0}_m^T \\ \boldsymbol{C}_m \boldsymbol{A}_m & \boldsymbol{\mathcal{I}}_{q \times q} \end{bmatrix} \begin{bmatrix} \Delta \bar{x}_m(k) \\ \bar{y}(k) \end{bmatrix} + \begin{bmatrix} \boldsymbol{B}_m \\ \boldsymbol{C}_m \boldsymbol{B}_m \end{bmatrix} \Delta \bar{u}(k) \tag{5.23}$$

$$\bar{y}(k) = \begin{bmatrix} \boldsymbol{0}_m & \boldsymbol{\mathcal{I}}_{q \times q} \end{bmatrix} \begin{bmatrix} \Delta \bar{x}_m(k) \\ \bar{y}(k) \end{bmatrix} \tag{5.24}$$

where $\boldsymbol{0}_m$ is a $q \times n$ zero matrix. The augmented state space equation is then

$$\bar{x}(k+1) = \boldsymbol{A}\bar{x}(k) + \boldsymbol{B}\Delta \bar{u}(k) \tag{5.25}$$

$$\bar{y}(k) = \boldsymbol{C} \begin{bmatrix} \Delta \bar{x}_m(k) \\ \bar{y}(k) \end{bmatrix} \tag{5.26}$$

This new system is characterised by matrices $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{C}$, with which the MPC controller will work. Consequently, it is important to check whether the augmented model respects the properties of controllability, observability and stability.

Looking at these properties it turns out that the observability is satisfied while controllability is not. The system also remains unstable. That is problematic because not only is the system no longer fully controllable, but through the so-called PBH test, it can be seen that the uncontrollable modes are also unstable, which means that the system is not stabilisable by the controller.

The stabilisability of the system can be ensured by introducing the use of Exponential Data Weighting as will be discussed below.

5.2.2   Formulation of MPC

Once the form[†] of the state-space model has been chosen, with an embedded integrator or not, the model predictive control can be formulated.

As mentioned earlier, the MPC aims to find the optimal control sequence that minimises a given objective function considering the constraints. The chosen objective function is based on the minimization of the error between the set-point signal and the output signal. If state-space model without embedded integrator is chosen, the objective function is given by the following equation

$$J = \sum_{i=1}^{N_p} \left[(\bar{r}_s(k) - \bar{y}(k+i|k))^T \, \boldsymbol{Q} \, (\bar{r}_s(k) - \bar{y}(k+i|k)) \right] + \sum_{i=0}^{N_c} \left[\bar{u}(k+i)^T \, \boldsymbol{R}_w \, \bar{u}(k+i) \right] \quad (5.27)$$

where

$\bar{r}_s(k)$ is the set-point signal for the output $\bar{y}$ at time k

$\bar{y}(k+i|k)$ is the output predicted at timestep $k+i$ given all the information at time k

$\bar{u}(k+i) = \begin{bmatrix} u_x & u_y & u_z \end{bmatrix}^T$ is the control command at timestep $k+i$

$\boldsymbol{Q} \geqslant 0$ is the output cost matrix

$\boldsymbol{R}_w > 0$ is the input cost matrix

N_p is the length of the prediction horizon

N_c is the length of the control horizon

This objective function can be written equivalently in vector form as

$$J = (\bar{R}_s - \bar{Y})^T \boldsymbol{\mathcal{Q}} (\bar{R}_s - \bar{Y}) + \bar{U}^T \boldsymbol{\mathcal{R}}_w \bar{U} \quad (5.28)$$

where

$$\bar{Y} = \begin{bmatrix} \bar{y}(k+1|k) & \bar{y}(k+2|k) & \bar{y}(k+3|k) & ... & \bar{y}(k+N_p|k) \end{bmatrix}^T$$
$$\bar{y}(k|k) = \boldsymbol{C}_d \, \bar{x}(k|k) \quad (5.29)$$

$$\bar{U} = \begin{bmatrix} \bar{u}(k+1|k) & \bar{u}(k+2|k) & \bar{u}(k+3|k) & ... & \bar{u}(k+N_p|k) \end{bmatrix}^T$$
$$\bar{u}(k|k) = \begin{bmatrix} u_x & u_y & u_z \end{bmatrix}^T \quad (5.30)$$

$$\boldsymbol{\mathcal{Q}} = \text{diag} \begin{bmatrix} \boldsymbol{Q} & \boldsymbol{Q} & ... & \boldsymbol{Q} \end{bmatrix}$$
$$\boldsymbol{\mathcal{R}}_w = \text{diag} \begin{bmatrix} \boldsymbol{R}_w & \boldsymbol{R}_w & ... & \boldsymbol{R}_w \end{bmatrix} \quad (5.31)$$

As shown in [43], considering equation 5.10 and 5.13, it is possible to obtain the following vector relationship for the predicted outputs

$$\bar{Y} = \boldsymbol{F}\bar{x}(k) + \boldsymbol{\Phi}\bar{U} \quad (5.32)$$

where

[†]The choice of which form to use influences the definition of the objective function involved in the calculation

$$\boldsymbol{F} = \begin{bmatrix} \boldsymbol{C}_d \boldsymbol{A}_d \\ \boldsymbol{C}_d \boldsymbol{A}_d^2 \\ \boldsymbol{C}_d \boldsymbol{A}_d^3 \\ \vdots \\ \boldsymbol{C}_d \boldsymbol{A}_d^{N_p} \end{bmatrix} \tag{5.33}$$

$$\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{C}_d \boldsymbol{B}_d & 0 & 0 & \dots & 0 \\ \boldsymbol{C}_d \boldsymbol{A}_d \boldsymbol{B}_d & \boldsymbol{C}_d \boldsymbol{B}_d & 0 & \dots & 0 \\ \boldsymbol{C}_d \boldsymbol{A}_d^2 \boldsymbol{B}_d & \boldsymbol{C}_d \boldsymbol{A}_d \boldsymbol{B}_d & \boldsymbol{C}_d \boldsymbol{B}_d & \dots & 0 \\ \vdots & & & & \\ \boldsymbol{C}_d \boldsymbol{A}_d^{N_p-1} \boldsymbol{B}_d & \boldsymbol{C}_d \boldsymbol{A}_d^{N_p-2} \boldsymbol{B}_d & \boldsymbol{C}_d \boldsymbol{A}_d^{N_p-3} \boldsymbol{B}_d & \dots & \boldsymbol{C}_d \boldsymbol{A}_d^{N_p-N_c} \boldsymbol{B}_d \end{bmatrix} \tag{5.34}$$

Substituting Equation 5.32 into 5.28 gives

$$J = (\bar{R}_s - \boldsymbol{F}\bar{x})^T \boldsymbol{\mathcal{Q}} (\bar{R}_s - \boldsymbol{F}\bar{x}) + \bar{U}^T (\boldsymbol{\Phi}^T \boldsymbol{\mathcal{Q}} \boldsymbol{\Phi} + \boldsymbol{\mathcal{R}}_w) \bar{U} - 2\bar{U}^T \boldsymbol{\Phi}^T \boldsymbol{\mathcal{Q}} (\bar{R}_s - \boldsymbol{F}\bar{x}) \tag{5.35}$$

This objective function is in the form of the so-called standard quadratic programming problem, which has been extensively studied in the literature. The required numerical solution is often viewed as an obstacle in the application of MPC, especially in real-time applications where minimum computational time is needed. In this thesis work the essential computational programs required to solve the quadratic problem have been produced following *Hildreth's quadratic programming procedure*. The procedure involved is not presented here, but a detailed description can be found in [43].

The optimization problem is

$$\begin{aligned} \min_{\bar{U}} \quad & J = \frac{1}{2}\bar{U}^T \boldsymbol{H} \bar{U} + \bar{U}^T \bar{f} \\ \text{s.t.} \quad & \bar{x}(k+1+i|k) = \boldsymbol{A}_d \bar{x}(k+i) + \boldsymbol{B}_d \bar{u}(k+i) \qquad i = 0 \dots N_p \\ & \boldsymbol{A}_{cons} \bar{U} \leqslant \bar{b} \end{aligned} \tag{5.36}$$

The matrix $\boldsymbol{H}$ and the vector $\bar{f}$ can be derived by comparison with Equation 5.35

$$\begin{aligned} \boldsymbol{H} &= 2(\boldsymbol{\Phi}^T \boldsymbol{\mathcal{Q}} \boldsymbol{\Phi} + \boldsymbol{\mathcal{R}}_w) \\ \bar{f} &= -2\boldsymbol{\Phi}^T \boldsymbol{\mathcal{Q}} (\bar{R}_s - \boldsymbol{F}\bar{x}) \end{aligned} \tag{5.37}$$

One of the main advantages of model predictive control is the possibility of including constraints within the optimisation problem. The classic example which shows its significance is when control input cannot exceed a certain level. It can be seen how including constraints on the control input in the optimisation problem gives better performance than the solution using a simple saturator. In 5.36, the term $\boldsymbol{A}_{cons} \bar{U} \leqslant \bar{b}$ represents inequality constraints included in the optimisation problem and handled by Hildreth's quadratic programming procedure. There are three major types of constraints, the first two deal with contraints imposed on the control variables $\bar{u}$, and the third type of constraint deals with output $\bar{y}$ or state variable $\bar{x}$ constraints. Here, only the constraints imposed on the control variables are considered. In particular, in the context of CubeSat control, it is relevant to be aware of the maximum torque that can be delivered by the reaction wheels and

the variation capability of the torque. It follows that constraints are imposed on the amplitude and incremental variation of the control variable. By indicating with $\bar{u}_{max}$ and $\bar{u}_{min}$ the maximum and minimum amplitude of torque allowed and with $\Delta\bar{u}_{max}$ and $\Delta\bar{u}_{min}$ the maximum and minimum torque variation allowed, the constraints can be written as

$$
\begin{aligned}
\bar{u}_{min} \leqslant \bar{u}(k+i) \leqslant \bar{u}_{max} && i = 0 \ldots N_c \\
\Delta\bar{u}_{min} \leqslant \Delta\bar{u}(k+i) \leqslant \Delta\bar{u}_{max} && i = 0 \ldots N_c
\end{aligned}
\tag{5.38}
$$

Both these must be written involving the optimisation variable, i.e. $\bar{u}$, so the constraints on incremental variation must be rewritten considering that

$$
\Delta\bar{u}(k+i) = \bar{u}(k+i) - \bar{u}(k+i-1)
\tag{5.39}
$$

Writing the constraints in the form indicated in 5.36 gives the following matrix and vector

$$
\boldsymbol{A}_{cons} =
\begin{bmatrix}
\boldsymbol{\mathcal{I}}_m & \boldsymbol{0} & \boldsymbol{0} & \ldots & \boldsymbol{0} & \boldsymbol{0} \\
\boldsymbol{0} & \boldsymbol{\mathcal{I}}_m & \boldsymbol{0} & \ldots & \boldsymbol{0} & \boldsymbol{0} \\
\vdots & & \ddots & & & \\
\boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \ldots & \boldsymbol{\mathcal{I}}_m & \boldsymbol{0} \\
\boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \ldots & \boldsymbol{0} & \boldsymbol{\mathcal{I}}_m \\
& & -(*) & & & \\
\boldsymbol{\mathcal{I}}_m & \boldsymbol{0} & \boldsymbol{0} & \ldots & \boldsymbol{0} & \boldsymbol{0} \\
-\boldsymbol{\mathcal{I}}_m & \boldsymbol{\mathcal{I}}_m & \boldsymbol{0} & \ldots & \boldsymbol{0} & \boldsymbol{0} \\
\boldsymbol{0} & -\boldsymbol{\mathcal{I}}_m & \boldsymbol{\mathcal{I}}_m & \ldots & \boldsymbol{0} & \boldsymbol{0} \\
\vdots & & \ddots & & & \\
\boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \ldots & -\boldsymbol{\mathcal{I}}_m & \boldsymbol{\mathcal{I}}_m \\
& & -(*) & & &
\end{bmatrix}
\qquad
\bar{b} =
\begin{bmatrix}
\bar{u}_{max} \\
\bar{u}_{max} \\
\vdots \\
\bar{u}_{max} \\
\bar{u}_{min} \\
\bar{u}_{min} \\
\vdots \\
\bar{u}_{min} \\
\Delta\bar{u}_{max} + \bar{u}(k-1) \\
\Delta\bar{u}_{max} \\
\vdots \\
\Delta\bar{u}_{max} \\
\Delta\bar{u}_{min} - \bar{u}(k-1) \\
\Delta\bar{u}_{min} \\
\vdots \\
\Delta\bar{u}_{min}
\end{bmatrix}
\tag{5.40}
$$

where $\boldsymbol{\mathcal{I}}_m$ is a $m \times m$ identity matrix, and m is the number of input which in this case is equals to three. The symbol $-(*)$ indicates that there is a block matrix equal to the one above but with the opposite sign.

5.2.3 MPC Stability

The stability is the first issue that must be faced in the design problem. Without stability any other performance measure or qualification are meaningless. In practice, a reasonable selection of tuning parameters (N_p, N_c, $\boldsymbol{\mathcal{R}}_w$, $\boldsymbol{\mathcal{Q}}$) and a well-posed optimization problem usually deliver stable performance for model predictive control algorithms. However, there is still a risk that the resulting closed loop might be unstable because the optimised control inputs could put the plant in a state that may be impossible to stabilise.

There are several ways of guaranteeing nominal stability of the MPC controller

- Terminal Constraints

- Infinite horizon, terminal cost penalty

- Exponential data weighting

Exponential Data Weighting

The idea of exponential data weighting in the design of model predictive control system is introduced in [43], here the same idea is applied but adapted to the considered case.

Previously, it was seen that the augmented model leads to an unstable system that is not fully controllable. Furthermore, the uncontrollable modes are unstable, and thus the system is also non-stabilisable. However, as mentioned earlier, exponential data weighting makes it possible to obtain a stabilisable closed-loop system, even using the augmented model.

Considering the augmented model, the exponential data weighting method involves minimising the following modified objective function

$$J = \sum_{i=1}^{N_p} \alpha^{-2i} \left[(\bar{x}(k+i|k))^T \, \boldsymbol{Q} \, (\bar{x}(k+i|k)) \right] + \sum_{i=0}^{N_c} \alpha^{-2i} \left[\Delta\bar{u}(k+i)^T \, \boldsymbol{R}_w \, \Delta\bar{u}(k+i) \right] \qquad (5.41)$$

Note that the objective function differs from that defined by Equation 5.27 for both the state variables and the incremental variation of the control variable. To obtain a stable closed-loop system, the weights must be exponentially decreasing. In this case $\alpha > 1$, and the exponential weights place more emphasis on the state $\bar{x}(k+i|k)$ at the current time and less emphasis on those at future times.

As shown in [43], this objective function is equivalent to the following

$$J = \sum_{i=1}^{N_p} \left[(\hat{x}(k+i|k))^T \, \boldsymbol{Q} \, (\hat{x}(k+i|k)) \right] + \sum_{i=0}^{N_c} \left[\Delta\hat{u}(k+i)^T \, \boldsymbol{R}_w \, \Delta\hat{u}(k+i) \right] \qquad (5.42)$$

where $\hat{x}(k+i|k)$ and $\Delta\hat{u}(k+i)$ are governed by the following difference equation

$$\hat{x}(k+i+1|k) = \frac{\boldsymbol{A}}{\alpha}\hat{x}(k+i|k) + \frac{\boldsymbol{B}}{\alpha}\Delta\hat{u}(k+i) \qquad (5.43)$$

with

$$\begin{aligned} \hat{x}(k+i|k) &= \alpha^{-i}\bar{x}(k+i|k) \\ \Delta\hat{u}(k+i) &= \alpha^{-i}\Delta\bar{u}(k+i) \end{aligned} \qquad (5.44)$$

The key point is that with this modified objective function the augmented state-space model $(\boldsymbol{A}, \boldsymbol{B})$ is changed into $(\alpha^{-1}\boldsymbol{A}, \alpha^{-1}\boldsymbol{B})$. Therefore, by choosing $\alpha > |\lambda_{max}(\boldsymbol{A})|$, the system matrix $\alpha^{-1}\boldsymbol{A}$ is transformed to a matrix with all eigenvalues inside the unit disc.

To study the controllability of the system, the rank of the controllability matrix obtained by considering $(\alpha^{-1}\boldsymbol{A}, \alpha^{-1}\boldsymbol{B})$ is analysed. The result is that the system remains not fully controllable, however, now the uncontrollable modes are stable and thus the system is stabilisable.

To guarantee an asymptotic stability with exponential data weighting, the matrices $\boldsymbol{Q}$ and $\boldsymbol{R}_w$ need to be appropriately selected. The full discussion can be found in [43], here only the final result is presented. The new matrices $\boldsymbol{Q}_\alpha$ and $\boldsymbol{R}_{w,\alpha}$ to be considered in the objective function are obtained by means of the following relations

$$\begin{aligned} \gamma &= \frac{1}{\alpha} \\ \boldsymbol{Q}_\alpha &= \gamma^2\boldsymbol{Q} + (1-\gamma^2)\boldsymbol{P}_\infty \\ \boldsymbol{R}_{w,\alpha} &= \gamma^2\boldsymbol{R}_w \end{aligned} \qquad (5.45)$$

where $\boldsymbol{P}_\infty$ is the solution of the algebraic Riccati equation

$$\boldsymbol{A}^T\left[\boldsymbol{P}_\infty - \boldsymbol{P}_\infty \boldsymbol{B}\left(\boldsymbol{R}_w + \boldsymbol{B}^T \boldsymbol{P}_\infty \boldsymbol{B}\right)^{-1} \boldsymbol{B}^T \boldsymbol{P}_\infty\right]\boldsymbol{A} + \boldsymbol{Q} - \boldsymbol{P}_\infty = \boldsymbol{0} \tag{5.46}$$

Using the exponentially weighted cost function and these new matrices the prediction horizon can be selected to be sufficiently large without numerical problems and asymptotic closed-loop stability is guaranteed.

Infinite Horizon, Terminal Cost Penalty

A further way of ensuring the asymptotic stability of the closed-loop system is to include a terminal cost penalty in the objective function. The basic idea is presented in [30], only the final results are summarised here.

The procedure consists in adding a terminal cost penalty in the objective function which guarantees asymptotic stability. This can be done by simply modifying the matrix $\boldsymbol{\mathcal{Q}}$ in Equation 5.28 as follows

$$\boldsymbol{\mathcal{Q}} = \begin{bmatrix} \boldsymbol{Q} & \boldsymbol{0} & \dots & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{Q} & \dots & \boldsymbol{0} & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \boldsymbol{0} & \boldsymbol{0} & \dots & \boldsymbol{Q} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \dots & \boldsymbol{0} & \bar{\boldsymbol{Q}} \end{bmatrix} \tag{5.47}$$

The procedure to derive the $\bar{\boldsymbol{Q}}$ matrix depends on the stability of the system, in the case of interest only the main steps related to the unstable plant case are recalled. As shown in [30] the first step is decompose the plant into stable and unstable parts, by means of an eigenvalue-eigenvector decomposition

$$\boldsymbol{A}_d = \boldsymbol{V}\boldsymbol{D}\boldsymbol{V}^{-1} = \begin{bmatrix} \bar{V}_u & \bar{V}_s \end{bmatrix} \begin{bmatrix} \boldsymbol{D}_u & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{D}_s \end{bmatrix} \begin{bmatrix} \tilde{V}_u \\ \tilde{V}_s \end{bmatrix} \tag{5.48}$$

The $\bar{\boldsymbol{Q}}$ matrix can be obtained from

$$\bar{\boldsymbol{Q}} = \tilde{V}_s^T \boldsymbol{\Pi} \tilde{V}_s \tag{5.49}$$

where $\boldsymbol{\Pi}$ is the solution of the matrix Lyapunov equation

$$\boldsymbol{\Pi} = \bar{V}_s^T \boldsymbol{C}_d^T \boldsymbol{Q} \boldsymbol{C}_d \bar{V}_s + \boldsymbol{D}_s^T \boldsymbol{\Pi} \boldsymbol{D}_s \tag{5.50}$$

To ensure asymptotic stability, in addition to the change in the $\boldsymbol{\mathcal{Q}}$ matrix it is necessary to add a terminal equality constraint to impose that the unstable modes are null at time instant $k + N_c$.

$$\bar{\xi}_u(k + N_c) = \tilde{V}_u \bar{x}(k + N_c) = 0 \tag{5.51}$$

Comparison Simulation

A nadir pointing simulation with unitary quaternion and null angular velocities guidance (desired) is carried out to analyse and compare the performance of the two methodologies: terminal cost penalty and exponential data weighting. In particular, only the goodness of control is investigated, so ideal sensors providing noise-free measurements are considered. Data used in the simulation are shown in Table 5.4. In the simulation, the torque calculated by solving the constrained optimisation problem is applied to the non-linear model of the system through the 4 reaction wheels model that will be discussed in Section 6.1.2. The results shown in Figures 5.12 and 5.13 are similar, however the method with exponential data weighting seems to converge faster with less oscillation.

Control rate	$T_s = 0.5\,s$		
Initial Euler Angles	$\Phi = 15°$	$\Theta = 15°$	$\Psi = 15°$
Amplitude Torque	$\bar{u}_{max} = 0.007\,Nm$	$\bar{u}_{min} = 0.007\,Nm$	
Incremental Variation Torque	$\Delta\bar{u}_{max} = 0.001\,Nm$	$\Delta\bar{u}_{min} = 0.001\,Nm$	
Cost Matrices	$\boldsymbol{Q} = 1000 \cdot \boldsymbol{\mathcal{I}}_n$	$\boldsymbol{R}_w = \boldsymbol{\mathcal{I}}_m$	
Horizons length	$N_p = 10$	$N_c = 7$	
Exponential Data Weights	$\alpha = e^{\lambda T_s}$	$\lambda = 0.1$	

Table 5.4: *Data used in the comparison simulation*
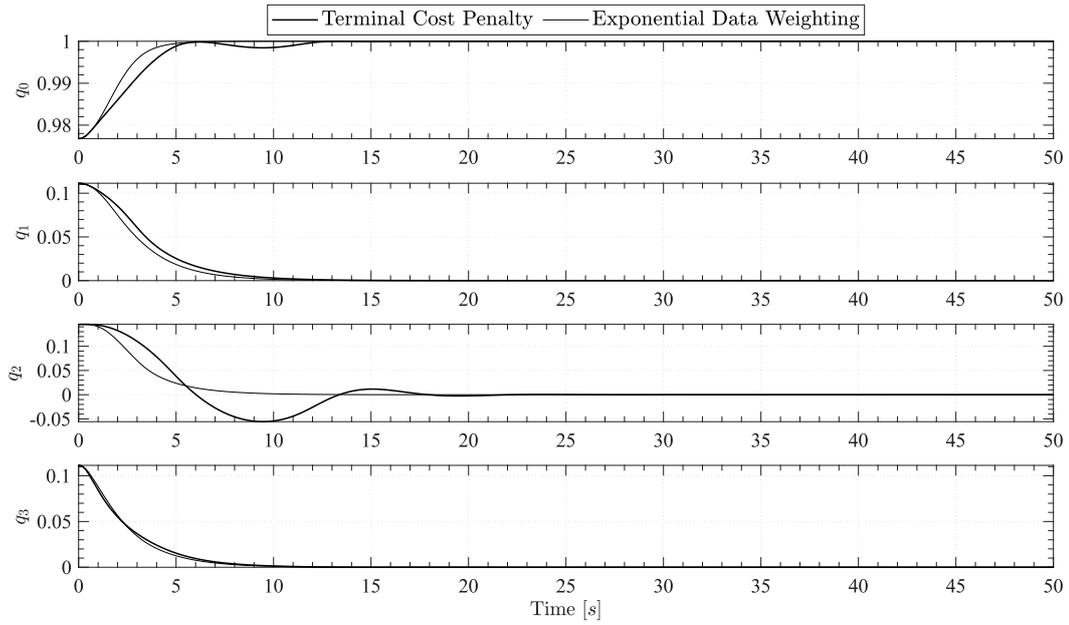
Figure 5.12: *Quaternions resulting from the two stabilisation methods of the MPC*
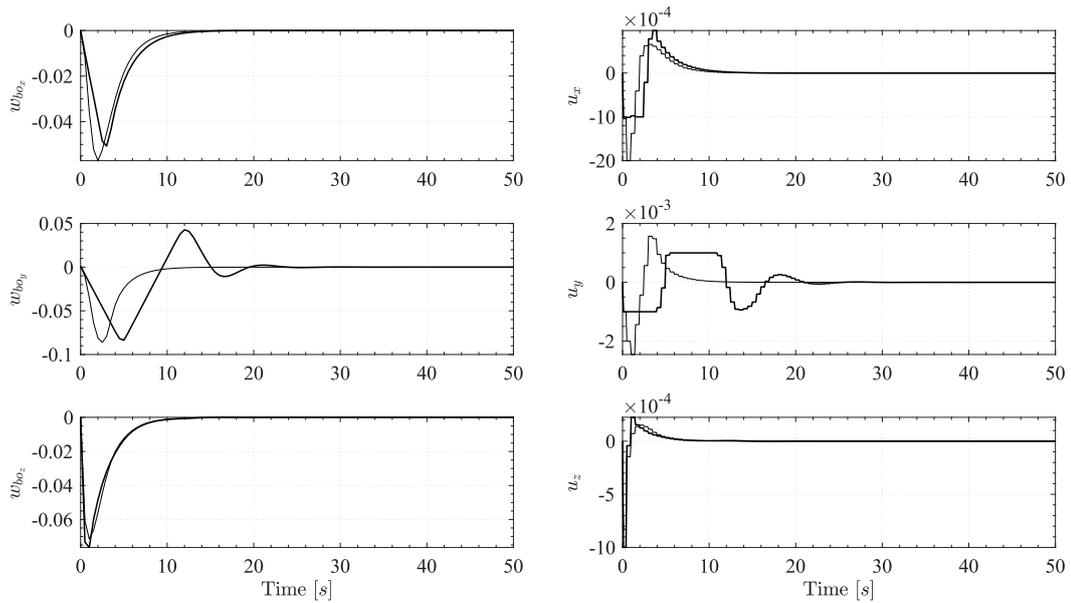
Figure 5.13: *Comparison of angular rates and optimal torques resulting from the two stabilisation methods of the MPC. Angular velocities are measured in rad/s, while torques are measured in Nm*

91

5.3   H Infinity Optimal Output Feedback Control

In this section, the implementation of the H_∞ controller is discussed, recalling only the relevant aspects of the mathematical machinery required for controller synthesis.

Systems for which controllers are designed are inevitably subject to uncertainties and disturbances related to imperfect modelling, errors in measurements, imperfections in system components, and so on. For this reason, it is essential to design controllers that can provide good performances while minimising the effect of these sources of disturbances. The purpose of the H_∞ controller is precisely the latter and can be effectively stated as follows.

Given a linear system, it is possible to introduce the control framework by separating input signals from output signals

- **Output signals**

  - Regulated outputs z, outputs to be controlled
  - Sensed outputs y, outputs used by the controller

- **Input signals**

  - Exogenous inputs w, disturbances, tracking signals, etc.
  - Actuator inputs u, output from the controller

The controller closes the loop from y to u as shown in Figure 5.14
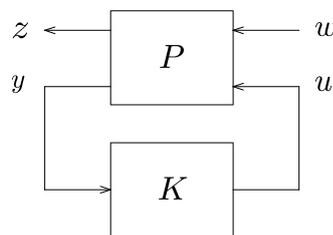
Figure 5.14: *The optimal control framework*

This sets up the framework for H_∞ optimal control whose goal is to choose the controller K such that the map from exogenous inputs (disturbances) to regulated outputs is minimised. As will be mentioned below, this is equivalent to minimising the closed-loop system L_2 induced norm which is the H_∞ norm of the transfer function realisation.

5.3.1   Background Theory

This subsection merely sets out to retrace the logical procedure leading to the synthesis of the controller, highlighting the main results utilised. A complete discussion of the concepts can be found in [34, 24] where references are also provided for detailed proofs of the most technical results.

State space or the transfer function is a representation of a system in which matrices or complex-valued are utilised to parameterise the representation. However, the system is not a transfer function, nor a set of matrices, but a set of behaviours. At the same time, system representations are important to infer the properties of the system through appropriate norms. For this reason, the theory involved in the design of this controller starts with the definition of the important mathematical concept of normed space. The only norm available for systems is the induced norm, for which systems form an algebra.

Denoting by G the system and by $\hat{G}$ its transfer function representation, as shown in [24] it turns out that if G is a *casual linear time invariant system* then the H_∞ norm of the transfer function is equal to the induced norm of the system

$$\|G\|_{\mathcal{L}(L_2)} = \|\hat{G}\|_{H_\infty} \tag{5.52}$$

where

$$\|\hat{G}\|_{H_\infty} = \operatorname*{ess\,sup}_{\omega \in \mathbb{R}} \bar{\sigma}(\hat{G}(j\omega)) \tag{5.53}$$

Therefore, the gain of the system G can be calculated as the H_∞ norm of the transfer function. This is an important result, in fact it is possible to minimise the norm of the closed loop system with respect to the L_2 induced norm from exogenous inputs to regulated outputs by choosing the optimal controller which minimises the H_∞ norm of the transfer function representation.

However, computers work well with optimisation algorithms using matrices within the so-called Linear Matrix Inequalities (LMI). Therefore, the idea is to relate the transfer function to the state space representation characterised by the matrices $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, and $\boldsymbol{D}$.

This can be done via the Kalman-Popov-Yakubovich (KYP) lemma [20, 24], also known as the Bounded Real Lemma. The KYP lemma can be defined for both continuous-time and discrete-time systems. As previously noted, modern control systems are typically implemented on digital processors, so it has been chosen to implement the H_∞ controller in discrete time. Therefore, the following is the discrete-time KYP lemma.

Assuming that $G : l_{2e} \to l_{2e}$ is a discrete-time LTI system with transfer function representation $\hat{G}$ and state-space realization $(\boldsymbol{A}_d, \boldsymbol{B}_d, \boldsymbol{C}_d, \boldsymbol{D}_d)$, where $\boldsymbol{A}_d \in \mathbb{R}^{n\times n}$ $\boldsymbol{B}_d \in \mathbb{R}^{n\times m}$, $\boldsymbol{C}_d \in \mathbb{R}^{q\times n}$, $\boldsymbol{D}_d \in \mathbb{R}^{q\times m}$, then the following are equivalent

- $\|\hat{G}\|_{H_\infty} \leqslant \gamma$

- There exist $\boldsymbol{P} > 0$ such that

$$\begin{bmatrix} \boldsymbol{A}_d^T \boldsymbol{P} \boldsymbol{A}_d - \boldsymbol{P} & \boldsymbol{A}_d^T \boldsymbol{P} \boldsymbol{B}_d & \boldsymbol{C}_d^T \\ * & \boldsymbol{A}_d^T \boldsymbol{P} \boldsymbol{B}_d - \gamma \boldsymbol{\mathcal{I}} & \boldsymbol{D}_d^T \\ * & * & -\gamma \boldsymbol{\mathcal{I}} \end{bmatrix} < 0 \tag{5.54}$$

In this case the lemma has been applied to the state space realisation $(\boldsymbol{A}_d, \boldsymbol{B}_d, \boldsymbol{C}_d, \boldsymbol{D}_d)$, which represents the system. However, the goal is not to minimise the H_∞ norm of the system but of the closed-loop system which maps exogenous inputs to outputs. Therefore, the next step is to isolate the regulated outputs as a function of the exogenous inputs and separate them from the other signals internal to the closed loop system.

With the signals previously identified it is possible to formulate the optimal control framework shown in Figure 5.14 in which appears the so-called aggregate plant P. The latter can be decomposed into four subsystems which maps the inputs to the outputs

$$\begin{bmatrix} z \\ y \end{bmatrix} = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix} \tag{5.55}$$

Different representations can be chosen for these subsystems: transfer functions or the so-called 9 matrix representation. Thinking of these subsystems as transfer functions it is easy to derive the relation between w and z. Two equations can be written from 5.55 together with the interconnection feedback equation

$$z = P_{11}w + P_{12}u$$
$$y = P_{21}w + P_{22}u \qquad (5.56)$$
$$u = Ky$$

Thus, it follows that

$$z = \big[P_{11} + P_{12}(I - KP_{22})^{-1}KP_{21}\big]w \qquad (5.57)$$

The latter is an algebraic expression for the regulated output as a function of the exogenous inputs in which the control variable now appears internally. This expression is called Linear Fractional Transformation (LTF) of (P, K) and denoted by

$$\underline{S}(P, K) := P_{11} + P_{12}(I - KP_{22})^{-1}KP_{21} \qquad (5.58)$$

Once this map from inputs to outputs has been defined it is possible to formulate the optimal control problem. However, as mentioned earlier there are certain disadvantages in implement optimization algorithms with the transfer function representation. While, results easier to work with the 9 matrix state space representation of P. Furthermore, a basic assumption that is made when formulating these optimal control problems is the *well-posedness*, i.e. there exist solutions to these systems, which can be easily verified with the state space representation.

Therefore, assuming a state space representation for P and K

$$P = \left[\begin{array}{c|cc} \boldsymbol{A}_d & \boldsymbol{B}_1 & \boldsymbol{B}_{d2} \\ \hline \boldsymbol{C}_{d1} & \boldsymbol{D}_{d11} & \boldsymbol{D}_{d12} \\ \boldsymbol{C}_{d2} & \boldsymbol{D}_{d21} & \boldsymbol{D}_{d22} \end{array}\right] \qquad K = \left[\begin{array}{c|c} \boldsymbol{A}_K & \boldsymbol{B}_K \\ \hline \boldsymbol{C}_K & \boldsymbol{D}_K \end{array}\right] \qquad (5.59)$$

Through multiple substitutions it is possible to obtain the following state-space representation of the interconnection $\underline{S}(P, K)$

$$\begin{aligned} \begin{bmatrix} \bar{x}(k+1) \\ \bar{x}_K(k+1) \end{bmatrix} &= \left(\begin{bmatrix} \boldsymbol{A}_d & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{A}_K \end{bmatrix} + \begin{bmatrix} \boldsymbol{B}_{d2} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{B}_K \end{bmatrix} \begin{bmatrix} \boldsymbol{\mathcal{I}} & -\boldsymbol{D}_K \\ -\boldsymbol{D}_{d22} & \boldsymbol{\mathcal{I}} \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{0} & \boldsymbol{C}_K \\ \boldsymbol{C}_{d2} & \boldsymbol{0} \end{bmatrix} \right) \begin{bmatrix} \bar{x}(k) \\ \bar{x}_K(k) \end{bmatrix} \\ &\quad + \begin{bmatrix} \boldsymbol{B}_{d1} + \boldsymbol{B}_{d2}\boldsymbol{D}_K\boldsymbol{Q}\boldsymbol{D}_{d21} \\ \boldsymbol{B}_K\boldsymbol{Q}\boldsymbol{D}_{d21} \end{bmatrix} \bar{w}(k) \end{aligned}$$
$$\qquad (5.60)$$
$$\begin{aligned} \bar{z}(k) &= \left(\begin{bmatrix} \boldsymbol{C}_{d1} & \boldsymbol{0} \end{bmatrix} + \begin{bmatrix} \boldsymbol{D}_{d12} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\mathcal{I}} & -\boldsymbol{D}_K \\ -\boldsymbol{D}_{d22} & \boldsymbol{\mathcal{I}} \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{0} & \boldsymbol{C}_K \\ \boldsymbol{C}_{d2} & \boldsymbol{0} \end{bmatrix} \right) \begin{bmatrix} \bar{x}(k) \\ \bar{x}_K(k) \end{bmatrix} \\ &\quad + \begin{bmatrix} \boldsymbol{D}_{d11} + \boldsymbol{D}_{d12}\boldsymbol{D}_K\boldsymbol{Q}\boldsymbol{D}_{d21} \end{bmatrix} w(k) \end{aligned}$$

where the iterconnection is well-posed if and only if $\boldsymbol{Q} = (\boldsymbol{\mathcal{I}} - \boldsymbol{D}_{d22}\boldsymbol{D}_K)^{-1}$ exists, i.e. $(\boldsymbol{\mathcal{I}} - \boldsymbol{D}_{d22}\boldsymbol{D}_K)$ is invertible. These forms the closed loop map from w to z

$$\left[\begin{array}{c|c} \boldsymbol{A}_{CL} & \boldsymbol{B}_{CL} \\ \hline \boldsymbol{C}_{CL} & \boldsymbol{D}_{CL} \end{array}\right] = \left[\begin{array}{c|c} \begin{bmatrix} \boldsymbol{A}_d & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{A}_K \end{bmatrix} + \begin{bmatrix} \boldsymbol{B}_{d2} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{B}_K \end{bmatrix} \begin{bmatrix} \boldsymbol{\mathcal{I}} & -\boldsymbol{D}_K \\ -\boldsymbol{D}_{d22} & \boldsymbol{\mathcal{I}} \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{0} & \boldsymbol{C}_K \\ \boldsymbol{C}_{d2} & \boldsymbol{0} \end{bmatrix} & \begin{bmatrix} \boldsymbol{B}_{d1} + \boldsymbol{B}_{d2}\boldsymbol{D}_K\boldsymbol{Q}\boldsymbol{D}_{d21} \\ \boldsymbol{B}_K\boldsymbol{Q}\boldsymbol{D}_{d21} \end{bmatrix} \\ \hline \begin{bmatrix} \boldsymbol{C}_{d1} & \boldsymbol{0} \end{bmatrix} + \begin{bmatrix} \boldsymbol{D}_{d12} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\mathcal{I}} & -\boldsymbol{D}_K \\ -\boldsymbol{D}_{d22} & \boldsymbol{\mathcal{I}} \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{0} & \boldsymbol{C}_K \\ \boldsymbol{C}_{d2} & \boldsymbol{0} \end{bmatrix} & \begin{bmatrix} \boldsymbol{D}_{d11} + \boldsymbol{D}_{d12}\boldsymbol{D}_K\boldsymbol{Q}\boldsymbol{D}_{d21} \end{bmatrix} \end{array}\right] \qquad (5.61)$$

Therefore, the optimal H_∞ control problem is

$$\min_{K \in H_\infty} \left\| \underline{S}\left(\left[\begin{array}{c|c} \boldsymbol{A}_{CL} & \boldsymbol{B}_{CL} \\ \hline \boldsymbol{C}_{CL} & \boldsymbol{D}_{CL} \end{array}\right], \left[\begin{array}{c|c} \boldsymbol{A}_K & \boldsymbol{B}_K \\ \hline \boldsymbol{C}_K & \boldsymbol{D}_K \end{array}\right] \right) \right\|_{H_\infty} = \|\underline{S}(P, K)\|_{\mathcal{L}(L_2)} \qquad (5.62)$$

5.3.2 Controller Synthesis

Once the procedure leading to the definition of the optimisation problem has been briefly described, it is possible to proceed with the controller synthesis. The goal is to design a controller for the nadir pointing phase capable of stabilising the spacecraft by nullifying its angular velocities and the vector part of the quaternion.

The discrete-time generalised LTI plant P shown in Figure 5.14, can be described by the following state-space realisation

$$
\begin{aligned}
\bar{x}(k+1) &= \boldsymbol{A}_d\,\bar{x}(k) + \boldsymbol{B}_{d1}\,\bar{w}(k) + \boldsymbol{B}_{d2}\,\bar{u}(k) \\
\bar{z}(k) &= \boldsymbol{C}_{d1}\,\bar{x}(k) + \boldsymbol{D}_{d11}\,\bar{w}(k) + \boldsymbol{D}_{d12}\,\bar{u}(k) \\
\bar{y}(k) &= \boldsymbol{C}_{d2}\,\bar{x}(k) + \boldsymbol{D}_{d21}\,\bar{w}(k) + \boldsymbol{D}_{d22}\,\bar{u}(k)
\end{aligned}
\tag{5.63}
$$

where

$\boldsymbol{A}_d, \boldsymbol{B}_{d1}, \boldsymbol{B}_{d2}, \boldsymbol{C}_{d1}, \boldsymbol{D}_{d11}, \boldsymbol{D}_{d12}, \boldsymbol{C}_{d2}, \boldsymbol{D}_{d21}, \boldsymbol{D}_{d22}$ represent the 9 matrix representation;

$\bar{x}(k) = \begin{bmatrix} q_1 & q_2 & q_3 & \omega_x & \omega_y & \omega_z \end{bmatrix} \in \mathbb{R}^n$ is the system state at time step k;

$\bar{z}(k) = \begin{bmatrix} \bar{z}_1 & \bar{z}_2 \end{bmatrix} \in \mathbb{R}^{q+m}$ is the performance signal at time step k;

$\bar{y}(k) \in \mathbb{R}^q$ is the sensed output at time step k;

$\bar{w}(k) \in \mathbb{R}^q$ is the exogenous input at time step k;

$\bar{u}(k) \in \mathbb{R}^q$ is the control input at time step k;

The first step is to derive an expression for the 9 matrices as a function of the matrices ($\boldsymbol{A}_d$, $\boldsymbol{B}_d$, $\boldsymbol{C}_d$, $\boldsymbol{D}_d$) constituting the state space representation 5.10 of the nominal plant P_0. This can be done by formulating the control problem in the 2-input/2-output framework choosing between the regulator or tracking controller configuration. In order to have a controller that can track a desired, other than nadir pointing, the tracking control framework,shown in Figure 5.15, was selected.
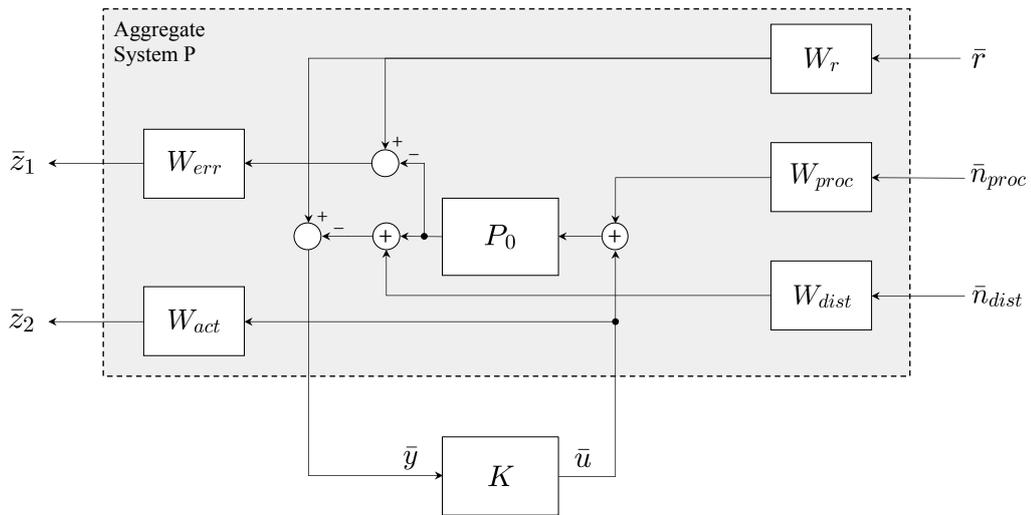

Figure 5.15: *Tracking control framework*

Figure 5.15 shows the aggregate system P which can be seen in Figure 5.14. The exogenous inputs includes the reference signal $\bar{r}$, process noise $\bar{n}_{proc}$, and sensor noise indicated as $\bar{n}_{dist}$.

To allow adjust performances it is possible to introduce weights within the framework. Although these could be tuned as a function of frequency, it was decided to use constant weights and thus diagonal matrices. The weights considered are the following

$\boldsymbol{W}_{proc}$ is a $m \times m$ diagonal matrix for process noises;

$\boldsymbol{W}_{dist}$ is a $q \times q$ diagonal matrix for sensor noises;

$\boldsymbol{W}_{err}$ is a $q \times q$ diagonal matrix for errors between the desired and the actual signal;

$\boldsymbol{W}_{act}$ is a $m \times m$ diagonal matrix for control authority;

$\boldsymbol{W}_r$ is a $q \times q$ diagonal matrix for the reference signal;

For clarity, from now on the quantities relating to the plant P_0 are indicated by the subscript p, while only the initial of the weights' subscript is kept. By referring to Figure 5.15, the following equations can be written

$$
\begin{aligned}
\bar{z}_1 &= \boldsymbol{W}_e \left[\boldsymbol{W}_r \bar{r} - P_0 (\boldsymbol{W}_p \bar{n}_{proc} + \bar{u}) \right] \\
\bar{z}_2 &= \boldsymbol{W}_a \bar{u} \\
\bar{y} &= \boldsymbol{W}_r \bar{r} - \left[P_0 (\boldsymbol{W}_p \bar{n}_{proc} + \bar{u}) + \boldsymbol{W}_d \bar{n}_{dist} \right]
\end{aligned}
\tag{5.64}
$$

The input and the output of the system are

$$
\begin{aligned}
\bar{u}_p &= \boldsymbol{W}_p \bar{n}_{proc} + \bar{u} \\
\bar{y}_p &= P_0 (\boldsymbol{W}_p \bar{n}_{proc} + \bar{u})
\end{aligned}
\tag{5.65}
$$

While the state-space equations for the states and the output are

$$
\begin{aligned}
\bar{x}_p(k+1) &= \boldsymbol{A}_d \bar{x}_p(k) + \boldsymbol{B}_d \bar{u}_p(k) \\
\bar{y}_p(k) &= \boldsymbol{C}_d \bar{x}_p(k) + \boldsymbol{D}_d \bar{u}_p(k)
\end{aligned}
\tag{5.66}
$$

Substituting $\bar{y}_p$ defined in Equation 5.66 into relation 5.64 gives

$$
\begin{aligned}
\bar{z}_1 &= -\boldsymbol{W}_e \boldsymbol{C}_d \bar{x}_p + \boldsymbol{W}_e \boldsymbol{W}_r \bar{r} - \boldsymbol{W}_e \boldsymbol{D}_d \boldsymbol{W}_p \bar{n}_{proc} + \boldsymbol{W}_e \boldsymbol{D}_d \bar{u} \\
\bar{z}_2 &= \boldsymbol{W}_a \bar{u} \\
\bar{y} &= -\boldsymbol{C}_d \bar{x}_p + \boldsymbol{W}_r \bar{r} - \boldsymbol{D}_d \boldsymbol{W}_p \bar{n}_{proc} - \boldsymbol{W}_d \bar{n}_{dist} - \boldsymbol{D}_d \bar{u}
\end{aligned}
\tag{5.67}
$$

Therefore, the state-space realisation defined in 5.63 can be written as

$$
\begin{aligned}
\bar{x}(k+1) &= \boldsymbol{A}_d \bar{x}(k) + \begin{bmatrix} \boldsymbol{0} & \boldsymbol{B}_d \boldsymbol{W}_p & \boldsymbol{0} \end{bmatrix} \bar{w}(k) + \boldsymbol{B}_d \bar{u}(k) \\
\bar{z}(k) &= \begin{bmatrix} -\boldsymbol{W}_e \boldsymbol{C}_d \\ \boldsymbol{0} \end{bmatrix} \bar{x}(k) + \begin{bmatrix} \boldsymbol{W}_e \boldsymbol{W}_r \boldsymbol{\mathcal{I}} & -\boldsymbol{W}_e \boldsymbol{D}_d \boldsymbol{W}_p & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \bar{w}(k) + \begin{bmatrix} -\boldsymbol{W}_e \boldsymbol{D}_d \\ \boldsymbol{W}_a \boldsymbol{\mathcal{I}} \end{bmatrix} \bar{u}(k) \\
\bar{y}(k) &= -\boldsymbol{C}_d \bar{x}(k) + \begin{bmatrix} \boldsymbol{W}_r \boldsymbol{\mathcal{I}} & -\boldsymbol{D}_d \boldsymbol{W}_p & -\boldsymbol{W}_d \boldsymbol{\mathcal{I}} \end{bmatrix} \bar{w}(k) - \boldsymbol{D}_d \bar{u}(k)
\end{aligned}
\tag{5.68}
$$

where

$\bar{x} = \bar{x}_p$

$\bar{w}^T(k) = \begin{bmatrix} \bar{r}^T(k) & \bar{n}_{proc}^T(k) & \bar{n}_{dist}^T(k) \end{bmatrix}$

$\bar{u}(k) = \bar{u}_c(k)$

$\bar{y}(k) = \bar{r}(k) - \bar{y}_p(k) - \bar{n}_{dist}(k)$

Equations 5.68 give expressions for the 9 matrix representation to be used in the optimisation problem. Thus, all the matrices necessary to completely define the state space representation 5.61 of the closed loop map from w to z are now available.

The next step is to use the matrices $(\boldsymbol{A}_{CL}, \boldsymbol{B}_{CL}, \boldsymbol{C}_{CL}, \boldsymbol{D}_{CL})$ within the KYP lemma to minimise the H_∞ norm of the interconnection $\underline{\mathsf{S}}(P, K)$. However, the step is not that straightforward, as proceeding straightforwardly would result in a non-linear and non-convex expression. Multiple substitutions and the introduction of new variables are required to make the problem convex and linear. Once the problem is solved, the variables of interest can be derived by going backward. Here, the final result used for controller synthesis is presented directly, omitting the intermediate steps. Nevertheless, the detailed procedure can be found in [34].

As reported in [20] the synthesis method for the H_∞ optimal output feedback controller is the following.

$$\min_{\gamma, \boldsymbol{X}_1, \boldsymbol{Y}_1, \boldsymbol{A}_{dn}, \boldsymbol{B}_{dn}, \boldsymbol{C}_{dn}, \boldsymbol{D}_{dn}} \gamma$$

subject to

$$\begin{bmatrix} \boldsymbol{X}_1 & \mathcal{I} & \boldsymbol{X}_1\boldsymbol{A}_d + \boldsymbol{B}_{dn}\boldsymbol{C}_{d2} & \boldsymbol{A}_{dn} & \boldsymbol{X}_1\boldsymbol{B}_{d1} + \boldsymbol{B}_{dn}\boldsymbol{D}_{d21} & \boldsymbol{0} \\ * & \boldsymbol{Y}_1 & \boldsymbol{A}_d + \boldsymbol{B}_{d2}\boldsymbol{D}_{dn}\boldsymbol{C}_{d2} & \boldsymbol{A}_d\boldsymbol{Y}_1 + \boldsymbol{B}_{d2}\boldsymbol{C}_{dn} & \boldsymbol{B}_{d1} + \boldsymbol{B}_{d2}\boldsymbol{D}_{dn}\boldsymbol{D}_{d21} & \boldsymbol{0} \\ * & * & \boldsymbol{X}_1 & \mathcal{I} & \boldsymbol{0} & \boldsymbol{C}_{d1}^T + \boldsymbol{C}_{d2}^T\boldsymbol{D}_{dn}^T\boldsymbol{D}_{d12}^T \\ * & * & * & \boldsymbol{Y}_1 & \boldsymbol{0} & \boldsymbol{Y}_1\boldsymbol{C}_{d1}^T + \boldsymbol{C}_{dn}^T\boldsymbol{D}_{d12}^T \\ * & * & * & * & \gamma\mathcal{I} & \boldsymbol{D}_{d11}^T + \boldsymbol{D}_{d21}^T\boldsymbol{D}_{dn}^T\boldsymbol{D}_{d12}^T \\ * & * & * & * & * & \gamma\mathcal{I} \end{bmatrix} > 0 \tag{5.69}$$

$$\begin{bmatrix} \boldsymbol{X}_1 & \mathcal{I} \\ \mathcal{I} & \boldsymbol{Y}_1 \end{bmatrix} > 0$$

The controller is recovered by

$$\begin{aligned} \boldsymbol{D}_K &= (\mathcal{I} + \boldsymbol{D}_{dK}\boldsymbol{D}_{d22})^{-1}\boldsymbol{D}_{dK} \\ \boldsymbol{C}_K &= (\mathcal{I} - \boldsymbol{D}_K\boldsymbol{D}_{d22})\boldsymbol{C}_{dK} \\ \boldsymbol{B}_K &= \boldsymbol{B}_{dK}(\mathcal{I} - \boldsymbol{D}_{d22}\boldsymbol{D}_K) \\ \boldsymbol{A}_K &= \boldsymbol{A}_{dK} - \boldsymbol{B}_K(\mathcal{I} - \boldsymbol{D}_{d22}\boldsymbol{D}_K)^{-1}\boldsymbol{D}_{d22}\boldsymbol{C}_K \end{aligned} \tag{5.70}$$

where

$$\left[\begin{array}{c|c} \boldsymbol{A}_{dK} & \boldsymbol{B}_{dK} \\ \hline \boldsymbol{C}_{dK} & \boldsymbol{D}_{dK} \end{array}\right] = \begin{bmatrix} \boldsymbol{X}_2 & \boldsymbol{X}_1\boldsymbol{B}_{d2} \\ \boldsymbol{0} & \mathcal{I} \end{bmatrix}^{-1} \left(\begin{bmatrix} \boldsymbol{A}_{dn} & \boldsymbol{B}_{dn} \\ \boldsymbol{C}_{dn} & \boldsymbol{D}_{dn} \end{bmatrix} - \begin{bmatrix} \boldsymbol{X}_1\boldsymbol{A}_d\boldsymbol{Y}_1 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \right) \begin{bmatrix} \boldsymbol{Y}_2^T & \boldsymbol{0} \\ \boldsymbol{C}_{d2}\boldsymbol{Y}_1 & \mathcal{I} \end{bmatrix}^{-1} \tag{5.71}$$

and where $\boldsymbol{X}_2$ and $\boldsymbol{Y}_2$ are any matrices which satisfy $\boldsymbol{X}_2\boldsymbol{Y}_2^T = \mathcal{I} - \boldsymbol{X}_1\boldsymbol{Y}_1$.

The optimisation problem has been solved in Matlab using the YALMIP‡ toolbox [29]. The controller resulting from the optimisation is a full order controller with internal dynamics which ensures the asymptotic stability of the closed-loop system while minimising the gain between $\bar{w}$ and $\bar{z}$.

H_∞ Optimal Output Simulation Results

The controller's performance can be tested through simulations. The same initial conditions used in the simulation with the MPC controller, shown in Table 5.4, were used for comparison. In particular, two simulations were performed: one without disturbances, namely the ideal case, and another with disturbances to test the controller's capacity to minimise exogenous inputs.

---

‡YALMIP is a Toolbox for Modeling and Optimization in MATLAB

Using the weights previously defined with unit values results in an H_∞ gain with the controller of approximately 1200. Therefore, it is necessary to set the weights appropriately to achieve the desired performance. The controller tuning seems acceptable with the weights shown in Table 5.5, which gives an H_∞ gain with the controller of $3.5 \cdot 10^{-6}$.

$$
\begin{aligned}
\boldsymbol{W}_{act} &= 1 \cdot 10^{-2} \ \text{diag} \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \\
\boldsymbol{W}_{proc} &= 2.5 \cdot 10^{-4} \ \text{diag} \begin{bmatrix} 1.2 & 1.2 & 0.25 \end{bmatrix} \\
\boldsymbol{W}_{dist} &= 5 \cdot 10^{-2} \ \text{diag} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \\
\boldsymbol{W}_{err} &= 1 \cdot 10^{-5} \ \text{diag} \begin{bmatrix} 1 & 1 & 1 & 0.1 & 0.1 & 0.1 \end{bmatrix} \\
\boldsymbol{W}_{r} &= 2 \cdot 10^{-2} \ \text{diag} \begin{bmatrix} 1 & 1 & 1 & 0.1 & 0.1 & 0.1 \end{bmatrix}
\end{aligned}
$$

Table 5.5: *Weights used in the H_∞ controller simulations*

The disturbances used in the simulation are those defined in Section 6.1.1. In particular, the controller inputs are the noisy measurement of quaternions obtained with the SVD method, which uses sun sensors and magnetometer measurements, and the noisy measurement of angular velocities obtained with the gyroscopes. Therefore, these simulations, like those of the MPC controller, do not involve the Extended Kalman Filter. Instead, the latter will be used in the simulations discussed in Chapter 6. As in the MPC simulation, the control torques are applied to the non-linear model of the system via reaction wheels.

The results of the simulations are shown in Figures 5.16, 5.17, and 5.18. By comparing Figures 5.16 and 5.12, it can be seen that the system converges to the desired state in a longer time than in the simulation with the MPC controller. One reason for this is the tuning process in which it was necessary to decrease the control authority to obtain a more robust controller. That highlights the differences between the two controllers; while Model Predictive Control prioritises performance over robustness, the H_∞ controller is much more robust but offers poorer performance.
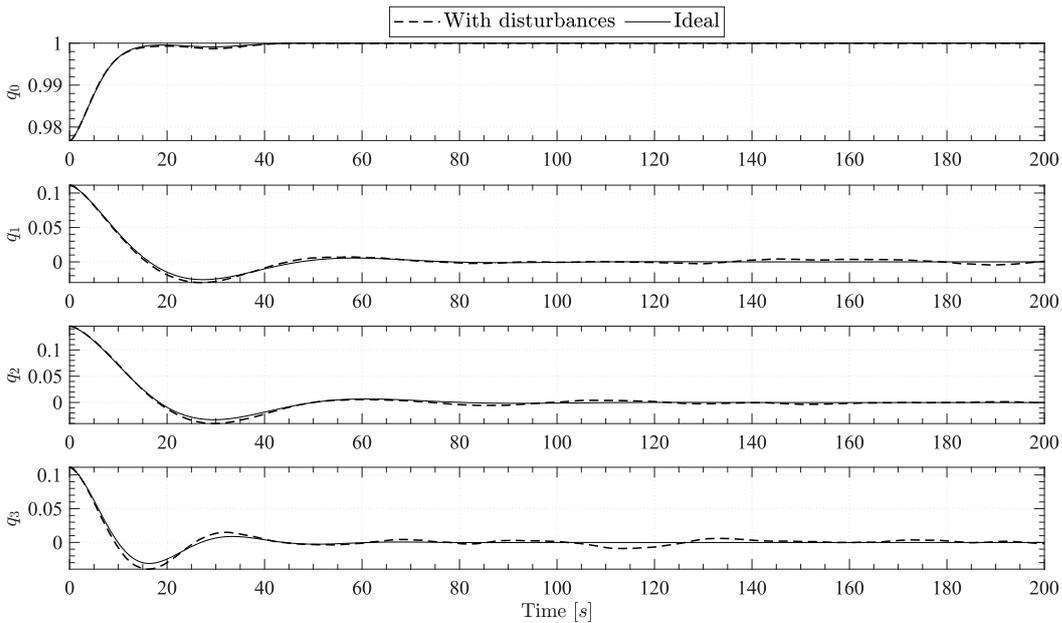

Figure 5.16: *Quaternions from the H_∞ controller simulation. The diagram shows the simulation results with and without disturbances*

However, simulation results with disturbances show how the controller rejects exogenous inputs while still providing good performance. This can be seen especially in Figure 5.17 where the noisy angular velocity measurement is also shown.

The diagrams are plotted over a longer simulation time to show steady-state performance, but this makes it difficult to see the stepwise nature of the control torque in Figure 5.18.
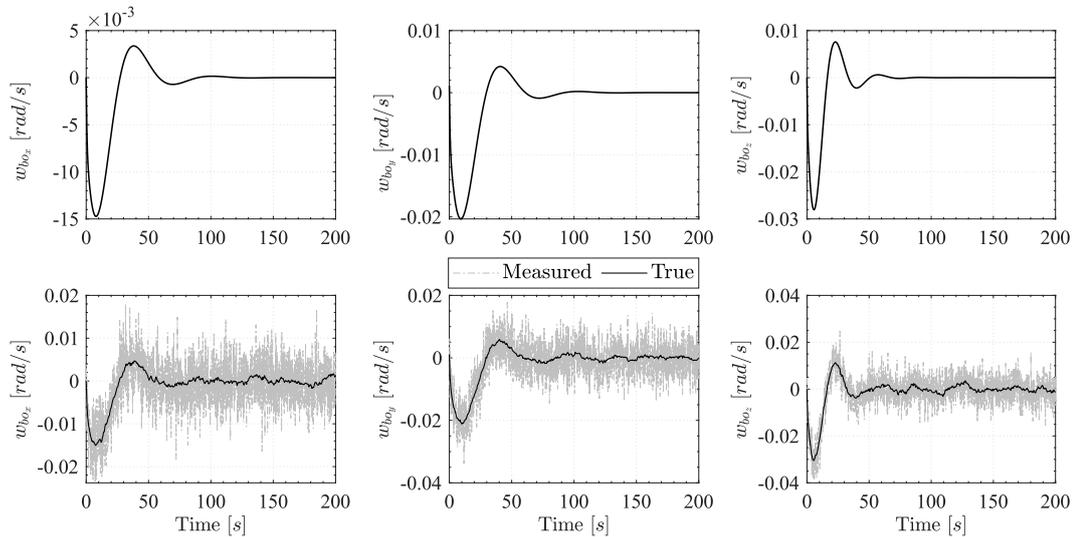

Figure 5.17: *Angular velocities from the H_∞ controller simulation. Above diagrams show the results of the simulation without disturbances, while below diagrams show the results with disturbances*
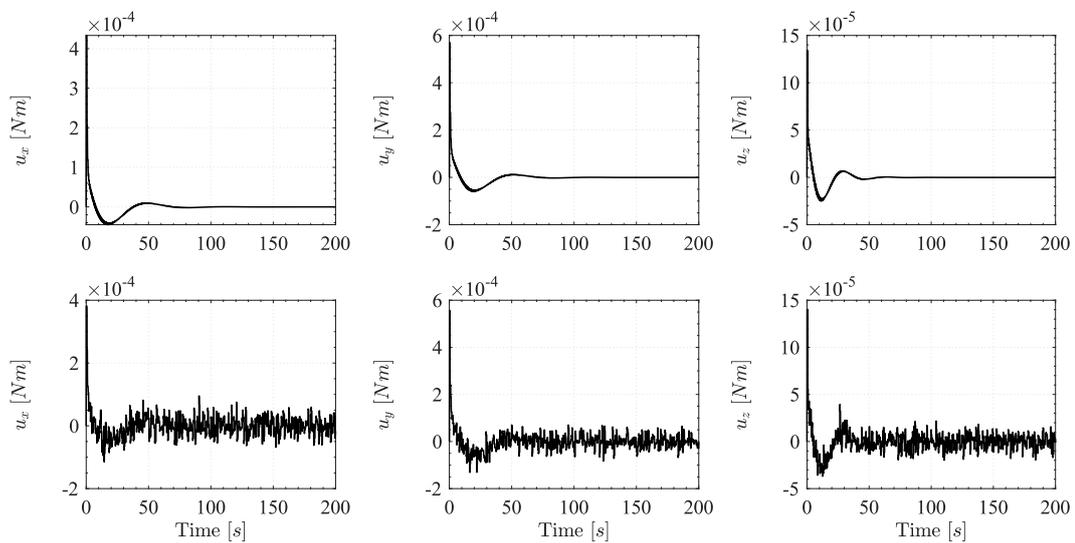

Figure 5.18: *Torques from the H_∞ controller simulation. Above diagrams show the results of the simulation without disturbances, while below diagrams show the results with disturbances*

5.4 Quaternion Feedback Control

As seen in the previous sections, the synthesis of the controllers used in the nadir pointing phase (MPC, H_∞) requires the linearised model of the spacecraft dynamics. That causes the controllers to perform best around the equilibrium point. For this reason, even though the controllers could work outside the equilibrium point, the preference is to develop a controller that can perform slew manoeuvres with large angular excursions without experiencing performance degradation.

A suitable controller for this purpose is one that implements the following Quaternion Feedback Control law defined as

$$\bar{T}^B = -\boldsymbol{K}\bar{q}_e - \boldsymbol{D}\bar{\omega}_e^B \tag{5.72}$$

where $\vec{q}_e$ is the quaternion error, $\bar{\omega}_e^B$ is the angular velocity error, and $\boldsymbol{K}$ and $\boldsymbol{D}$ are gain matrices.

As stated in [12], it is possible to demonstrate that the control law is globally asymptotically stabilising onto any arbitrary desired attitude, $\vec{q}_d$, for a wide choice of the gain matrices $\boldsymbol{K}$ and $\boldsymbol{D}$.

The quaternion error can be calculated as

$$\vec{q}_e = \bar{q}_{des}^{-1} \otimes \vec{q}_{true} \tag{5.73}$$

where

$$\bar{q}_d = \bar{q}_{des}^{-1} = \frac{\bar{q}_{des}^*}{\|\bar{q}_{des}\|_2} = \frac{\begin{bmatrix} q_0 & -q_1 & -q_2 & -q_3 \end{bmatrix}^T}{(q_0^2 + q_1^2 + q_2^2 + q_3^2)} \tag{5.74}$$

the product is given by

$$\vec{q}_e = \bar{q}_d \otimes \vec{q}_{true} = \begin{bmatrix} q_{d_0} & -q_{d_1} & -q_{d_2} & -q_{d_3} \\ q_{d_1} & q_{d_0} & -q_{d_3} & q_{d_2} \\ q_{d_2} & q_{d_3} & q_{d_0} & -q_{d_1} \\ q_{d_3} & -q_{d_2} & q_{d_1} & q_{d_0} \end{bmatrix} \begin{bmatrix} q_{true_0} \\ q_{true_1} \\ q_{true_2} \\ q_{true_3} \end{bmatrix} \tag{5.75}$$

5.4.1 Target Tracking Control

As discussed in Section 3.2.2 the observation payload requires target tracking control to obtain high-quality non-blurred images. The procedure shown in [21] is taken as a reference in this regard.

Figure 5.19 shows an overview of the quantities involved in tracking control. The goal is to control the spacecraft to point the axis of the observation payload at the ground-target of interest. That requires time-varying guidance of the desired quaternion and angular velocity to be used in the control law 5.72. First, the vector pointing from the Earth's centre to the target must be defined. Given the coordinates of the target in terms of latitude (La) and longitude (Lo), this vector can be calculated in $\mathcal{F}_{ECEF}$ as follows

$$\bar{x}_T^E = R_\oplus \begin{bmatrix} \cos Lo \cos La \\ \cos Lo \sin La \\ \sin La \end{bmatrix} \tag{5.76}$$

Defining the angle α_G as

$$\alpha_G = \alpha_{G_0} + \omega_\oplus t \tag{5.77}$$
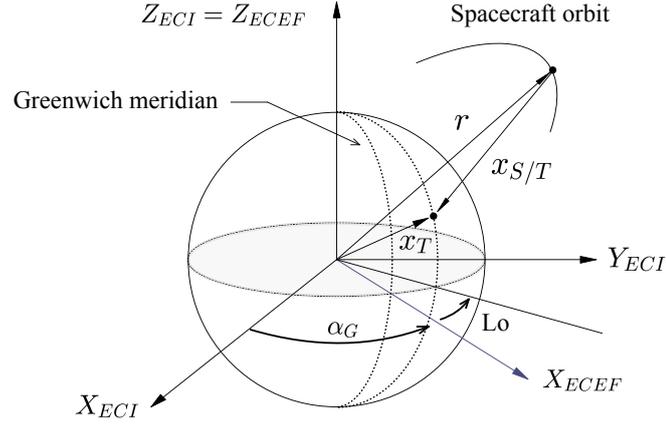
Figure 5.19: *Overview of vectors involved in tracking control*

where α_{G_0} is given in Equation 2.6, the transpose of the transformation matrix in Equation 2.17 can be used to compute the vector in $\mathcal{F}_{ECI}$

$$\bar{x}_T^I = \boldsymbol{A}_E^I \bar{x}_T^E \tag{5.78}$$

The spacecraft position vector expressed in $\mathcal{F}_{ECI}$ is denoted by $\bar{r}$ as shown in Figure 5.19. It follows that the vector pointing from the satellite to the target in $\mathcal{F}_{ECI}$ can be calculated as

$$\bar{x}_{S/T}^I = \bar{x}_T^I - \bar{r} \tag{5.79}$$

The same vector can be transformed in the local orbital frame $\mathcal{F}_O$ through the transpose of the matrix defined in Equation 2.18

$$\bar{x}_{S/T}^O = \boldsymbol{A}_I^O \bar{x}_{S/T}^I \tag{5.80}$$

The unit direction vector of $\bar{x}_{S/T}^O$ in $\mathcal{F}_O$ is given by

$$\bar{u}_{S/T}^O = \frac{\bar{x}_{S/T}^O}{\|\bar{x}_{S/T}^O\|} \tag{5.81}$$

The spacecraft axis representing the observation payload is denoted by the constant unit vector $\bar{u}_p^B = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$. Therefore, the vector $\bar{u}_{S/T}^O$ can be seen as the desired orientation for $\bar{u}_p^B$ during tracking. The desired quaternion can be defined using the following unit vector

$$\bar{u}_d = \frac{\bar{u}_p^B \times \bar{u}_{S/T}^O}{\|\bar{u}_p^B \times \bar{u}_{S/T}^O\|} \tag{5.82}$$

and the angle between the time-dependent vector $\bar{u}_{S/T}^O$ and the constant vector $\bar{u}_p^B$ given by

$$\delta = \cos^{-1}\left(\bar{u}_p^B \cdot \bar{u}_{S/T}^O\right) \tag{5.83}$$

With these two elements, the desired quaternion to be used in Equation 5.74 can be calculated as shown in Equation 2.13

$$\bar{q}_{des} = \begin{bmatrix} \cos \delta/2 \\ \bar{u}_d \, \sin \delta/2 \end{bmatrix} \tag{5.84}$$

To investigate the tracking accuracy it is necessary to compute $\bar{u}_{S/T}^B$ as

$$\bar{u}_{S/T}^B = \boldsymbol{A}_O^B \bar{u}_{S/T}^O \tag{5.85}$$

where A_O^B is the attitude matrix defined in 2.16. Then, the tracking error, i.e. the angle between the vector $\bar{u}_{S/T}^B$ and $\bar{u}_p^B$, is

$$\theta_{err} = \cos^{-1}\left(\bar{u}_p^B \cdot \bar{u}_{S/T}^B\right) \tag{5.86}$$

The procedure for calculating the desired angular velocity is more elaborate and is not presented here. The final result is given below, but the procedure is discussed in detail in [21]

$$\omega_{OB}^{des} = \frac{\bar{x}_{S/T}^O \times \dot{\bar{x}}_{S/T}^O}{\|\bar{x}_{S/T}^O\|^2} \tag{5.87}$$

where

$$
\begin{aligned}
\dot{\bar{x}}_{S/T}^O &= \dot{A}_I^O\left(\bar{x}_T^I - \bar{r}\right) + A_I^O\left(\dot{A}_E^I \bar{x}_T^E - \bar{v}\right) \\
\dot{A}_E^I &= \omega_\oplus \begin{bmatrix} -\sin(\alpha_G) & -\cos(\alpha_G) & 0 \\ \cos(\alpha_G) & -\sin(\alpha_G) & 0 \\ 0 & 0 & 0 \end{bmatrix} \\
\dot{A}_I^O &= \begin{bmatrix} \dot{\bar{o}}_{1I} & \dot{\bar{o}}_{2I} & \dot{\bar{o}}_{3I} \end{bmatrix}^T \\
\dot{\bar{o}}_{1I} &= \dot{\bar{o}}_{2I} \times \bar{o}_{3I} + \bar{o}_{2I} \times \dot{\bar{o}}_{3I} \\
\dot{\bar{o}}_{2I} &= -\frac{\left(\mathcal{I} - \bar{o}_{2I}\bar{o}_{2I}^T\right)\left(\bar{r} \times \dot{\bar{v}}\right)}{\|\bar{r} \times \bar{v}\|} \\
\dot{\bar{o}}_{3I} &= -\frac{\left(\mathcal{I} - \bar{o}_{3I}\bar{o}_{3I}^T\right)\bar{v}}{\|\bar{r}\|}
\end{aligned}
\tag{5.88}
$$

The desired time-varying quaternion and angular velocity can be used in control law 5.72 to perform target tracking control. The tracking mode is only activated when the target is visible to the spacecraft, the boundary situation is the one discussed in Section 3.2.5 and shown in Figure 3.16. The problem of large angle slew manoeuvering, which may be encountered if an instantaneous manoeuvre is performed from an attitude very different from that required for target tracking, was previously mentioned. It was anticipated that to account for the limited actuation capacity of reaction wheels it is necessary to perform a pre-manoeuvre, extended in time, before target tracking. This is also discussed in [21] where it is suggested to define the following desired pre-manoeuvre quaternion as

$$\bar{q}_{des}(t_0) = \begin{bmatrix} \cos\delta(t_0)/2 \\ \bar{u}_d(t_0)\sin\delta(t_0)/2 \end{bmatrix} \tag{5.89}$$

where t_0 is the time when starting tracking control, and $\bar{u}_d(t_0)$ and $\delta(t_0)$ can be calculated from Equations 5.82 and 5.83 using the unit vector $\bar{u}_{S/T}^O(t_0)$ calculated at t_0.

As shown in Figure 3.16, it has been chosen to begin the pre-manoeuvre ahead of the starting tracking point by an angle ϕ. This makes it possible to begin the slew manoeuvre before arriving at the tracking start point and thus avoid large, sudden angular excursions.

Target Tracking Control Simulation

A simulation was carried out to validate the correct implementation of the algorithm, in which the target to be tracked lies in correspondence with the ground track of the satellite's orbit as shown in Figure 5.20. In this way, it can be verified whether when the target is precisely below the spacecraft during tracking, the guidance provides the unit quaternion as desired. An ideal simulation is considered in which the sensors are noise-free. The spacecraft starts from an initial condition of nadir pointing.
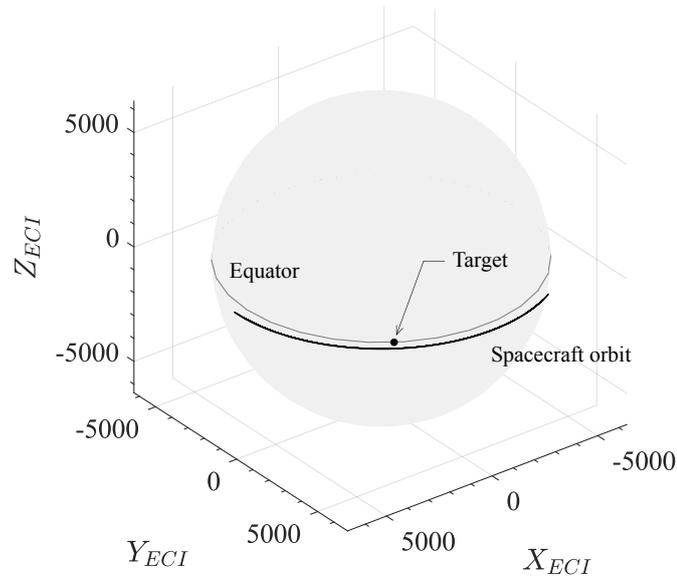
Figure 5.20: *Spacecraft orbit and target position in the target tracking control simulation. Target lies on the spacecraft ground track*

Once it reaches the pre-manoeuvre start position in orbit, it begins to change its attitude to achieve the required attitude for tracking control. The simulation results are shown in Figures 5.21, 5.22, and 5.23. The diagrams show the guidance, hence the desired quaternion or angular velocity, and the actual output of the spacecraft, and also highlight the pre-manoeuvre and tracking manoeuvre regions. The results show that the spacecraft correctly tracks the desired output provided by the guidance function. In Figure 5.22, it can be seen that when the spacecraft is in correspondence with the target, the desired quaternion coincides with the unit quaternion, thus indicating the correct implementation of the algorithm. Figure 5.21 shows that the spacecraft body angular rate does not exactly track the desired angular velocity.
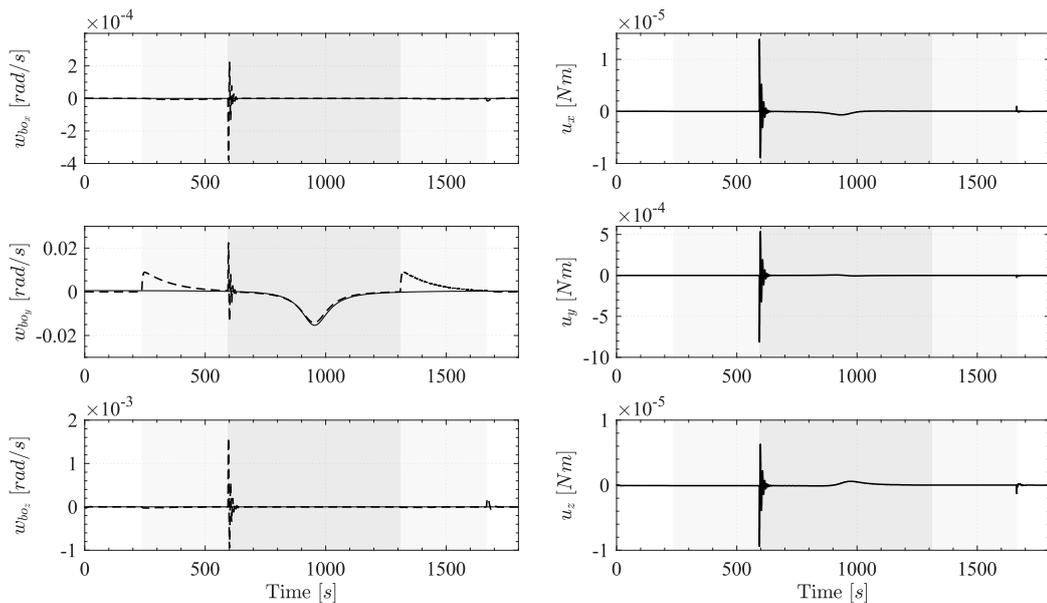

Figure 5.21: *Angular velocities and torques output in the target tracking control simulation. The legend is the same as in Figure 5.22*

However, from Figure 5.23 it can be seen that the tracking error is very small. It reaches a maximum where the spacecraft is manoeuvring closest to the target. Due to the pre-and post-manoeuvre, the torques delivered during the simulation are very small, as represented in Figure 5.21. In this way, the technological actuation limit of the reaction wheels is not exceeded, and the risk of wheel saturation is minimised.
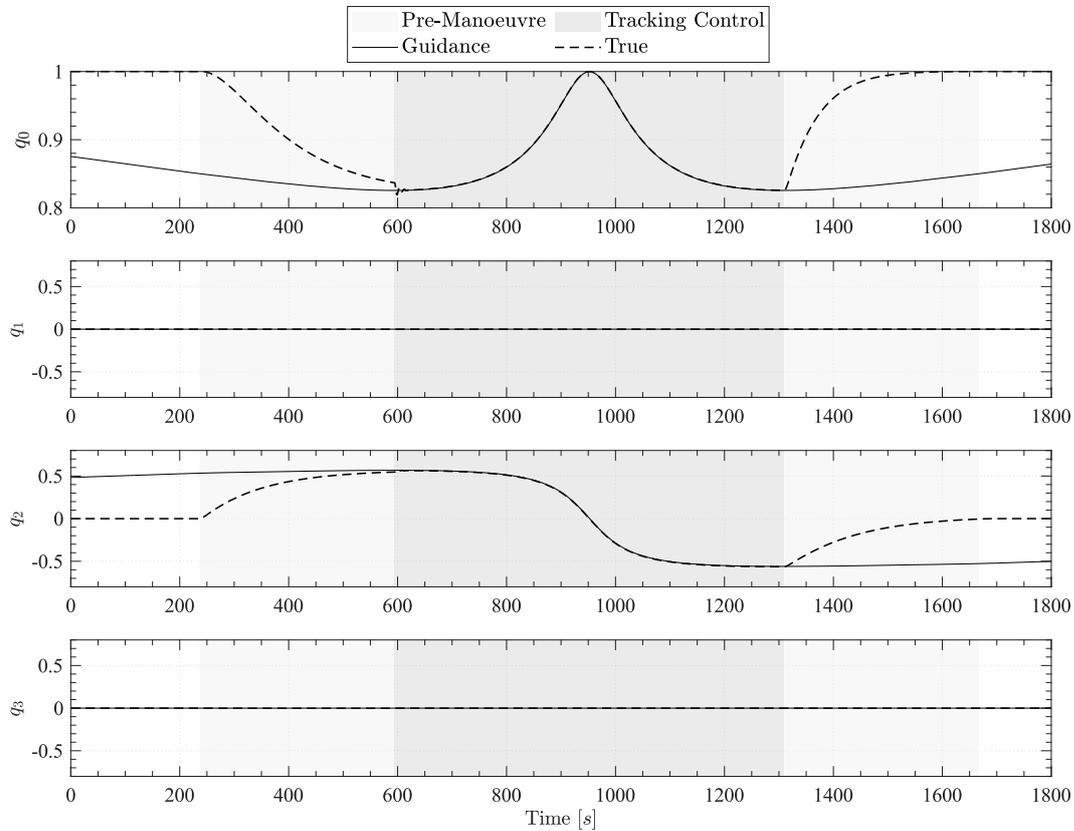

Figure 5.22: *Quaternions output in the target tracking control simulation. Light gray indicates the pre-manoeuvre, while dark gray the tracking control*
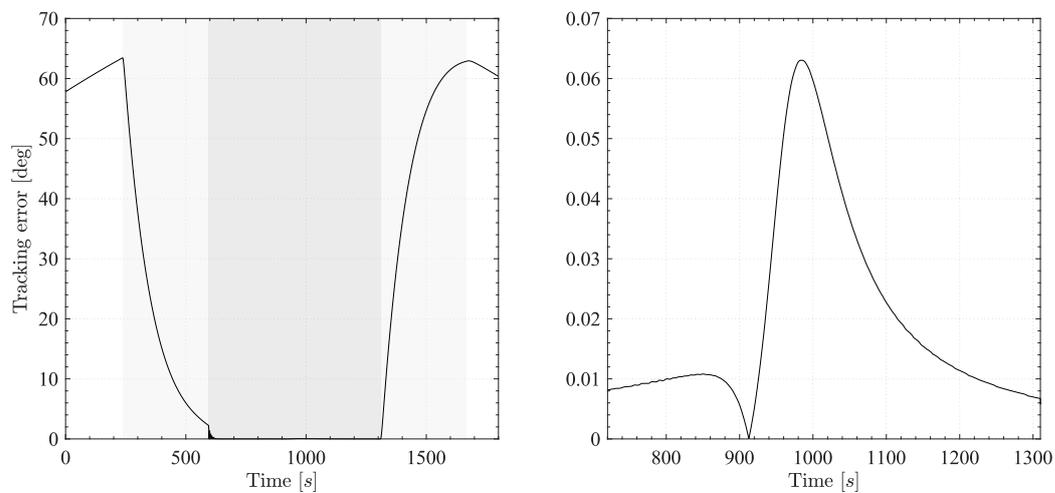

Figure 5.23: *Tracking error output in the target tracking control simulation. The error angle is defined in Equation 5.86. Left diagram has the same legend as in Figure 5.22. Right diagram is a focus of the tracking control region*

Chapter 6

Simulations and Results

This chapter presents the results from complete non-ideal simulations. In this regard, the simulation setup is defined by introducing the models of the actuators and sensors and the possible combinations of the determination and control algorithms whose performance is studied.

6.1 Simulations Setup

The simulations are carried out using a six-degree-of-freedom orbital simulator. A representative diagram of the simulator is shown in Figure 6.1. The diagram begins and ends with thicker boxes marking the beginning and end of the iterative loop.
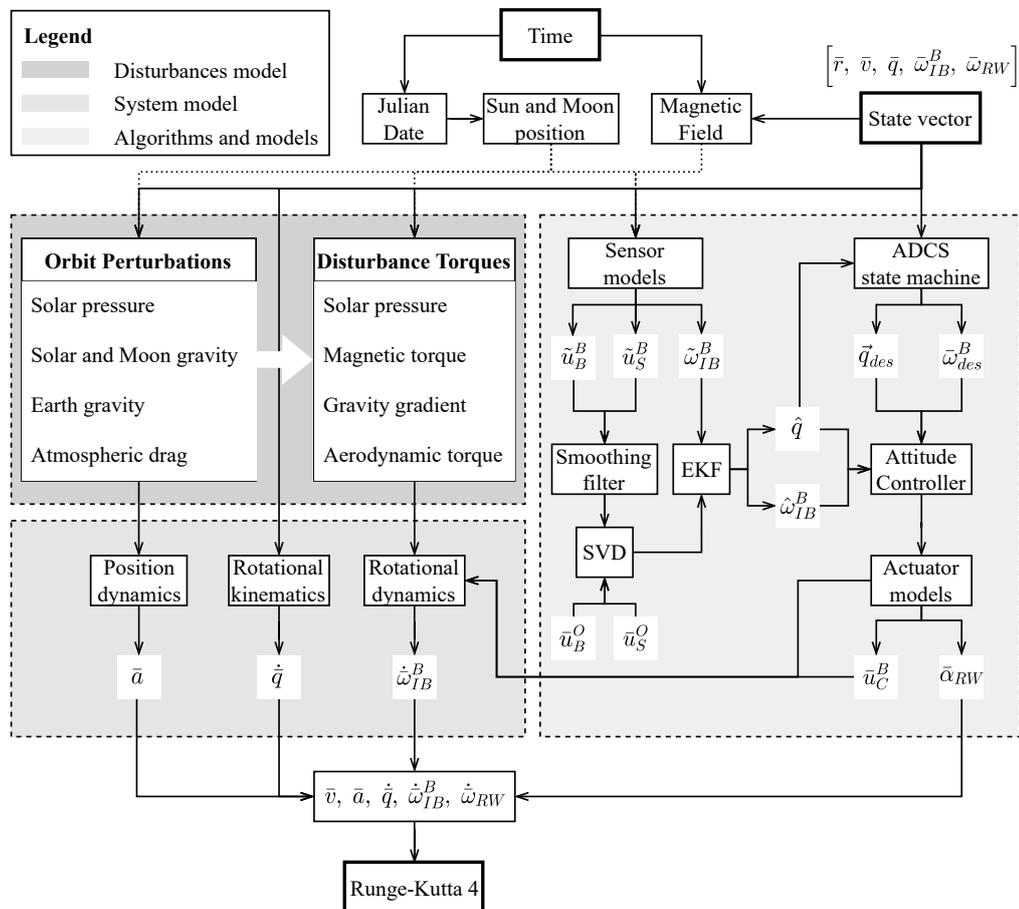

Figure 6.1: *Representative diagram of the orbital simulator*

The scheme in Figure 6.1 is an extension of the diagram previously shown in Figure 3.17. The simulator consists of the disturbances models, the dynamics and kinematics model of the plant, the determination and control algorithms, and the actuator and sensor models. The new state vector and time are available at the beginning of each timestep. Time is required for the Julian date calculation, which is necessary to estimate the Moon and Sun's position and serves as input for computing the magnetic field. The state vector is defined as follows

$$\bar{x}_{sim} = \begin{bmatrix} \bar{r} & \bar{v} & \bar{q} & \bar{\omega}_{IB}^{B} & \bar{\omega}_{RW} \end{bmatrix}^{T} \tag{6.1}$$

where

$\bar{r}$ is the spacecraft position

$\bar{v}$ is the spacecraft velocity

$\bar{q}$ is the quaternion

$\bar{\omega}_{IB}^{B}$ is the inertial referenced spacecraft angular velocity measured in the body frame

$\bar{\omega}_{RW}$ is the angular velocity of the reaction wheels

The spacecraft position and velocity, with the Sun and Moon positions, are used to calculate the orbital perturbations discussed in Section 2.4.3. Some of the perturbative forces are used then for the calculation of disturbance torques acting on the spacecraft.

The sensor model receives as input the angular velocity of the spacecraft, the measurement of the sun vector, and the Earth's magnetic field. These measurements are corrupted with noise resulting from the characteristics of the selected hardware. The noisy measurements of the sun vector and magnetic field are filtered by the smoothing filter and given as input, together with the respective models measured in the orbital frame, to the SVD method presented in Section 4.1.1. The attitude obtained by the deterministic algorithm, represented by the quaternion, serves as an innovation for the Extended Kalman Filter with the angular velocity measurement acquired by gyroscopes. The output of the EKF is the recursive estimate of the attitude and angular velocity. Given these two quantities, it is possible to calculate the orbit-referenced angular rate of the spacecraft.

According to the current control mode, the ADCS state machine provides guidance for the controller. For example, in target pointing mode, the guidance function requires as input the estimated quaternion, the position of the spacecraft in orbit, and other data pre-loaded in memory to calculate the desired attitude and angular velocity needed to track the ground target.

The estimated and desired quaternion and angular velocity are the input for the controller, which is responsible for reducing the error to zero. The controller produces a control torque applied by driving the reaction wheels. The latter and disturbance torques contribute to the dynamics of the spacecraft. The angular momentum of the reaction wheels must also be considered in the dynamics, as shown in Equation 4.9.

The kinematic and dynamic models of the spacecraft allow computing the variances of the quantities constituting the state vector. The angular acceleration of the reaction wheels is also part of the state vector to account for the dynamics of the wheels. The derivative of the state vector is integrated using the fourth-order Runge-Kutta numerical integration method. It marks the end of the current time step simulation loop and the beginning of the new one. Minor variations of the depicted scheme in Figure 6.1 may occur for specific control modes. For example, in the case of detumbling mode the estimator used is the Rate Kalman Estimator, which therefore replaces the EKF, and the control law requires the magnetic field as input rather than the attitude.

6.1.1 Sensor Models

The sensors were modelled for the simulations by considering an appropriate noise in the measurement resulting from considerations regarding the selected hardware and based on previous experience. An exception is made for the gyroscope whose model used is that shown in [45]. The equation describing the model is 4.14, which has already been presented previously in the context of the EKF implementation. The noise characteristics used for the sensors is shown in Table 6.1.

Sensor	Noise	Value	σ	Units	Notes
Sun Sensors	Bias	$[-0.001, \ 0.001]$	-	-	Uniformly distributed
	Output noise	$[-0.05, \ 0.05]$	-	-	Uniformly distributed
Gyroscopes	Bias†	$[-0.045, \ 0.045]$	0.0135	$deg/s^{3/2}$	Normal distributed, zero mean
	Output noise	$[-0.81 \ 0.81]$	0.27	deg/s	Normal distributed, zero mean
Magnetometers	Output noise	$[-5, \ 5]$	-	μT	Uniformly distributed

†RRW: Rate random walk utilised in the gyroscope model and which causes drift rate

Table 6.1: *Characteristic noises utilised to model the non-perfect working of the sensors in the simulations. All noises listed are white noises*

6.1.2 Actuator Models

Detailed modelling of actuator models is beyond the scope of this thesis work. For this reason, the essential models describing actuator dynamics are used.

Reaction Wheels Model

As seen in Section 3.2.5, reaction wheels are used as the primary attitude control actuators on the spacecraft. Each reaction wheel is characterised by inertia ($I_{r_i}^R$), an angular velocity (ω_{RW_i}), and acceleration (α_{RW_i}) that must be modelled appropriately.

The inertia of the reaction wheels contributes to that of the spacecraft and can be calculated by assuming reaction wheels as discs of radius r_{RW}, height h_{RW}, and mass m_{RW}. The inertia matrix about the centre of mass of the reaction wheel is given by

$$\boldsymbol{I}_{r_i}^R = m_{RW} \begin{bmatrix} r_{RW}^2/2 & 0 & 0 \\ 0 & r_{RW}^2/4 + h_{RW}^2/12 & 0 \\ 0 & 0 & r_{RW}^2/4 + h_{RW}^2/12 \end{bmatrix} \tag{6.2}$$

The transformation from the wheel frame to the body frame requires the definition of the unit vectors in the body frame $\hat{n}_i$ along the spin axes of the wheels. As previously discussed, the reaction wheels are arranged in the pyramidal configuration shown in Figure 6.2. The unit vectors for this configuration are

$$\boldsymbol{\mathcal{N}} = \begin{bmatrix} \hat{n}_1 & \hat{n}_2 & \hat{n}_3 & \hat{n}_4 \end{bmatrix} = \begin{bmatrix} \cos\beta & 0 & -\cos\beta & 0 \\ 0 & \cos\beta & 0 & -\cos\beta \\ -\sin\beta & -\sin\beta & -\sin\beta & -\sin\beta \end{bmatrix} \tag{6.3}$$

where angle β has been chosen as $30°$. Given a single unit vector, it is possible to extract the Euler angles, shown in Figure 6.3, as follows

$$\Psi = \tan^{-1}\left(\frac{\hat{n}_i(2)}{\hat{n}_i(1)}\right) \qquad\qquad \Theta = \tan^{-1}\left(\frac{\hat{n}_i(3)}{\sqrt{\hat{n}_i(1)^2 + \hat{n}_i(2)^2}}\right) \tag{6.4}$$
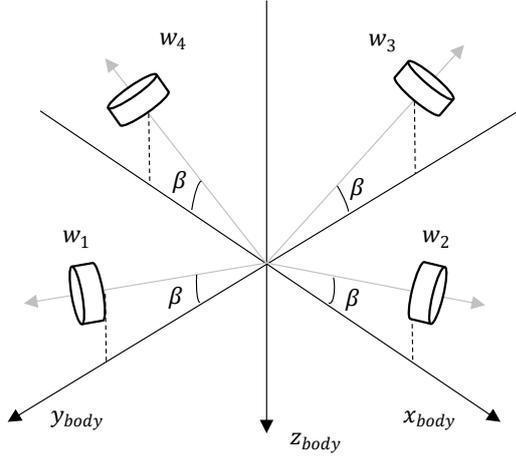
Figure 6.2: *Reaction wheels pyramidal configuration*
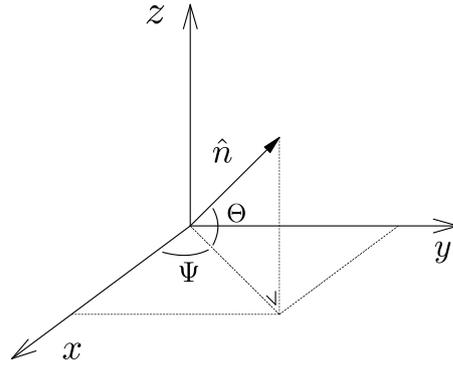
Figure 6.3: *Unit vector Euler angles*

Using these two angles in the rotation matrix defined in Equation 2.11, the inertia matrix in the body reference system can be calculated as

$$\boldsymbol{I}^B_{r_i} = \boldsymbol{A}_{3-1-3}(\Phi, \Theta, \Psi)^T \, \boldsymbol{I}^R_{r_i} \, \boldsymbol{A}_{3-1-3}(\Phi, \Theta, \Psi) \tag{6.5}$$

For reasons related to the position of the payload and positioning of other equipment within the CubeSat, it was decided to position the reaction wheels at the top of the CubeSat as shown in Figure 6.4.
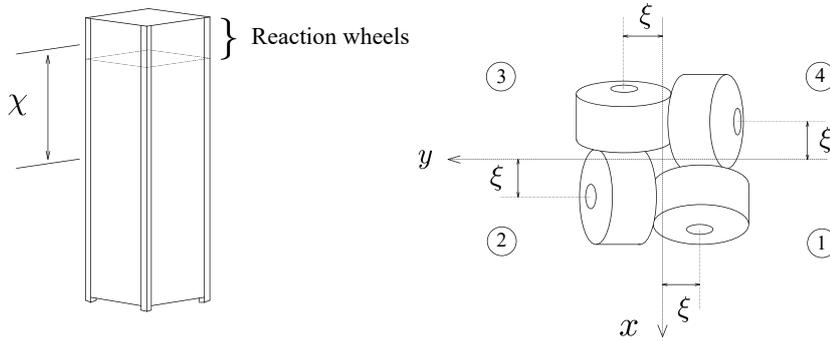

Figure 6.4: *Position of the reaction wheel assembly in the CubeSat. The figure on the right shows a simplified top view of the arrangement of the chosen reaction wheels. The view is adapted from the original configuration shown in [7]*

The parallel axis theorem can be then used to shift the inertias to the center of mass of the vehicle

$$\boldsymbol{I}^B_{rB_i} = \boldsymbol{I}^B_{r_i} + m_{RW_i} \boldsymbol{S}(\bar{d}_{cm_i}) \boldsymbol{S}(\bar{d}_{cm_i})^T \tag{6.6}$$

where $\bar{d}_{cm_i}$ is the distance of the reaction wheel's centre of mass from the spacecraft's centre of mass. Assuming that the centre of mass of the CubeSat is approximately at the geometric centre of gravity of the parallelepiped, the $\bar{d}_{cm_i}$ vectors are defined as

$$
\begin{aligned}
\bar{d}_{cm_1} &= \begin{bmatrix} h_{RW} & -\xi & -\chi \end{bmatrix} \\
\bar{d}_{cm_2} &= \begin{bmatrix} \xi & h_{RW} & -\chi \end{bmatrix} \\
\bar{d}_{cm_3} &= \begin{bmatrix} -h_{RW} & \xi & -\chi \end{bmatrix} \\
\bar{d}_{cm_4} &= \begin{bmatrix} -\xi & -h_{RW} & -\chi \end{bmatrix}
\end{aligned}
\tag{6.7}
$$

where h_{RW} is the height of the wheel, ξ is approximately equal to the radius of the wheel, and χ is approximately equal to half the height of the CubeSat minus half the height of the wheel. The wheel data are shown in Table D.4.

The total inertia of the spacecraft-reaction wheels assembly is given by

$$\boldsymbol{I} = \boldsymbol{I}_S + \sum_{i=1}^{4} \boldsymbol{I}_{rB_i}^B \tag{6.8}$$

where $\boldsymbol{I}_S$ is the inertia of the spacecraft alone. The total angular momentum of the spacecraft is then

$$\bar{h}^B = \boldsymbol{I}_S \, \bar{\omega}_{IB}^B + \sum_{i=1}^{4} \boldsymbol{I}_{rB_i}^B \, \omega_{RW_i} \, \hat{n}_i \tag{6.9}$$

Similarly, the total torque of the spacecraft is given by

$$\bar{u}_C^B = \sum_{i=1}^{4} \boldsymbol{I}_{rB_i}^B \, \alpha_{RW_i} \, \hat{n}_i \tag{6.10}$$

where $\bar{\alpha}_{RW}$ is the reaction wheels angular acceleration. It is assumed that the angular acceleration of each reaction wheel can be directly controlled. Therefore, knowing the control torque, it is necessary to derive the accelerations of the reaction wheels. As stated in [31], there is no unique way to distribute the torque for a number of reaction wheels greater than three. There are different approaches to solve this problem, in this work the pseudoinverse distribution law is considered.

Equation 6.10 can be rewritten equivalently as

$$\bar{u}_C^B = \begin{bmatrix} \boldsymbol{I}_{rB_1}^B \, \hat{n}_1 & \boldsymbol{I}_{rB_2}^B \, \hat{n}_2 & \boldsymbol{I}_{rB_3}^B \, \hat{n}_3 & \boldsymbol{I}_{rB_4}^B \, \hat{n}_4 \end{bmatrix} \begin{bmatrix} \alpha_{RW_1} \\ \alpha_{RW_2} \\ \alpha_{RW_3} \end{bmatrix} = \boldsymbol{J} \bar{\alpha}_{RW} \tag{6.11}$$

Therefore, as shown in [31] the angular accelerations can be calculated as

$$\bar{\alpha}_{RW} = \boldsymbol{J}^\dagger \bar{u}_C^B \tag{6.12}$$

where the pseudoinverse[*] is defined as

$$\boldsymbol{J}^\dagger = \boldsymbol{J}^T \left(\boldsymbol{J} \boldsymbol{J}^T \right)^{-1} \tag{6.13}$$

Magnetic Torquers Model

On board the CubeSat, three magnetic torquers are used mainly for detumbling and momentum dumping of the reaction wheels. Each magnetic torquer is independently controlled by a current i whose intensity defines the desired dipole moment according to the following relationship

$$\bar{m} = n \, A \, \bar{i} \tag{6.14}$$

where A is the area of the magnetic torquers, n is the number of turns in the coil of each magnetic torquer, and $\bar{i}$ is the vector composed by the three current components. It is assumed that all magnetic torquers have the same area and same number of turns. Once the dipole moment is produced, the magnetic torque is generated according to the law shown in Equation 5.1.

---

[*]also known as Moore-Penrose generalized inverse

6.1.3 Attitude Determination and Control Configuration

Chapters 4 and 5 present the determination and control algorithms taken as a reference and developed within this thesis. These algorithms can be combined in different ways, and one of the aims of this thesis is to study such combinations to find the one that provides the best performance.

The control algorithms considered utilise full state knowledge, hence attitude and angular velocities feedback. As stated in [31], the problem of controlling a spacecraft without full sate feedback is more complex. The approaches used to solve this problem can be divided into methods which estimate the unmeasured states using a filter algorithm, such as the Kalman Filter, and methods which develop control laws directly from output feedback [31].

The H_∞ optimal output feedback controller implemented in this thesis falls into the last category, i.e. it is a controller designed to receive measured output as input to control the plant. However, this thesis also investigates its performance when it receives, as input, an estimated vector from an Extended Kalman Filter. Therefore, there are three combinations to be analysed as shown in figure 6.5.
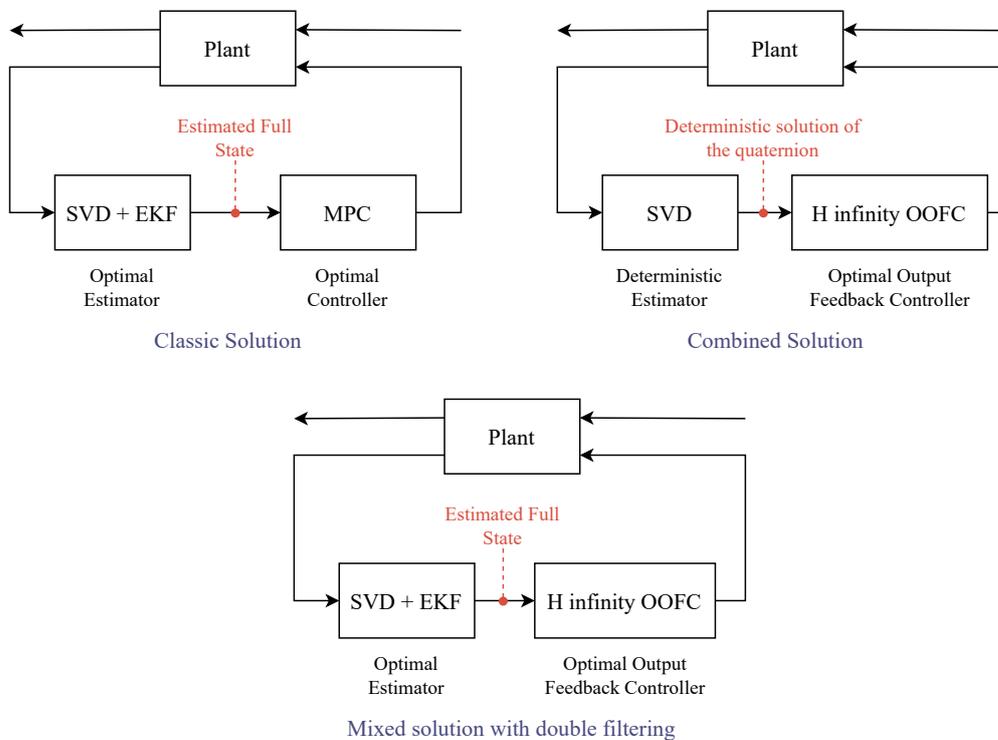
Figure 6.5: *Attitude control loop alternatives configurations*

1. **Optimal estimator and optimal controller**: This configuration involves the classic combination of an Extended Kalman Filter and a Model Predictive controller;

2. **Deterministic method and optimal output feedback controller**: In this case, only the H_∞ controller is used together with the SVD method required for the quaternion calculation;

3. **Optimal estimator and optimal output feedback controller**: In this case, the H_∞ controller is used together with the Extended Kalman Filter.

The performances of these combinations, in terms of extrinsic indicators, are compared and analysed in Section 6.3.

6.2 Detumbling Simulation

The detumbling simulation carried out in Section 5.1.1 did not include noise in the sensors. Naturally, this leads to better results since, as seen from Figure 5.2, the spacecraft can be detumbled in only two orbits. This section presents the results of a simulation inclusive of sensor noises but with the same initial conditions to compare the results.

The initial conditions of the simulation are shown in Table 6.2, while the sensor noise data are those shown in Table 6.1. For completeness, Appendix B reports the results of a detumbling simulation with the same initial conditions but at an altitude of 700 km.

Orbit characteristics	Circular orbit, Altitude = $470\ km$, Inclination = $51.6°$
Initial attitude	$[\Phi,\quad \Theta,\quad \Psi] = [5,\quad 5,\quad 5]\ [deg]$
Initial angular rates	$[\omega_x,\quad \omega_y,\quad \omega_z] = [10,\quad 10,\quad 10]\ [deg/s]$
Rates	Control rate = $5\ s$, Sampling rate = $0.5\ s$
Constraints	Maximum dipole moment = $0.5\ Am^2$

Table 6.2: *Detumbling simulation setup*

As mentioned earlier, the estimation of orbit-referenced angular velocities in the detumbling phase is performed with the robust Kalman Rate Estimator discussed in Section 4.2.3. A good estimation of angular velocities is essential for the correct operation of the control law and detumbling of the spacecraft. Although the KRE performs well, as shown in Figure 6.6, the non-perfect knowledge of the spacecraft's angular velocity results in a longer detumbling time than in the ideal case. Figure 6.7 confirms the last statement as it reports a focus of the satellite's third orbit during detumbling, where the angular velocity requirement is only met towards the end and not already at the second orbit as in the ideal case.
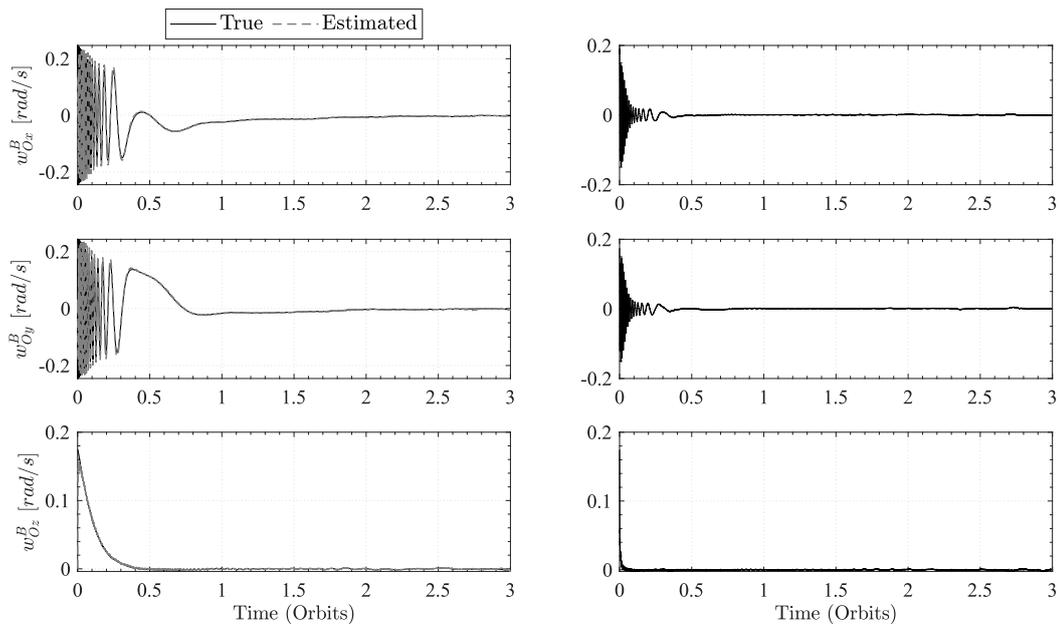

Figure 6.6: *Results of the Kalman Rate Estimator in the detumbling simulation. The diagrams on the right show the error, i.e. the difference between the true and estimated value*

111
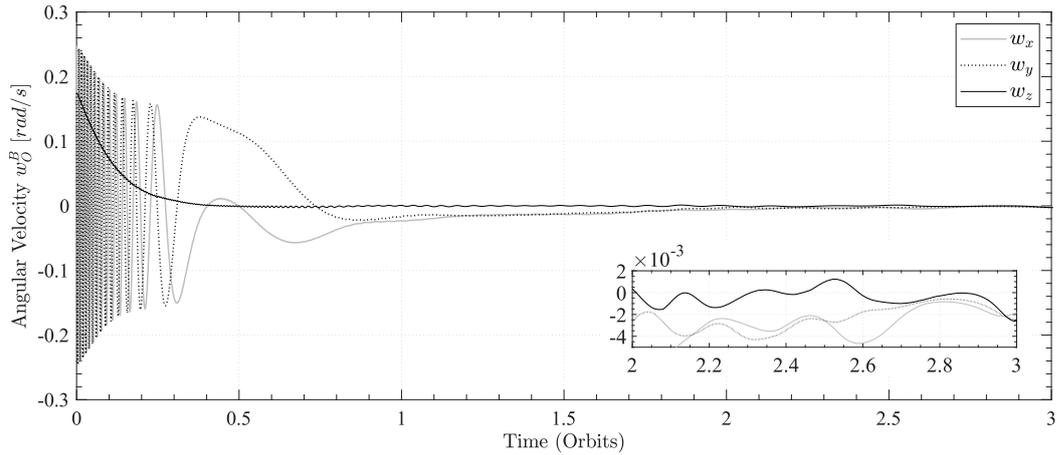
Figure 6.7: *Orbit referenced angular velocities of the spacecraft during the detumbling simulation. A focus of the third orbit is shown in the figure to show the detail of the angular velocity value*

However, the requirement is considered fulfilled as once the angular velocity falls below the threshold specified in Table 3.8 the ADCS state machine commands the transition to another control mode.

Regarding the control cost during detumbling, Figure 6.9 shows that only the magnetic torquer along z saturates and only in the initial phase. The obtained dipole moment profile differs from that of the ideal case due to sensor noise.
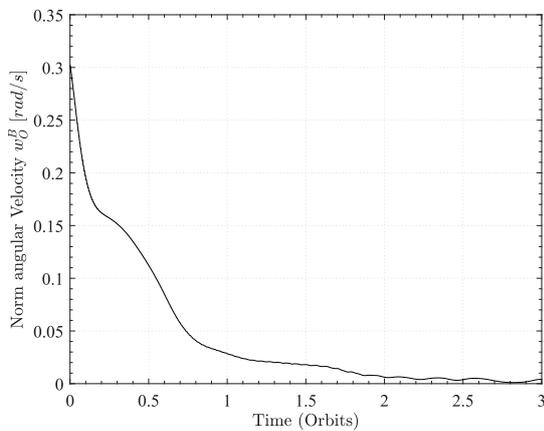

**Figure 6.8:** *Norm of angular velocities*
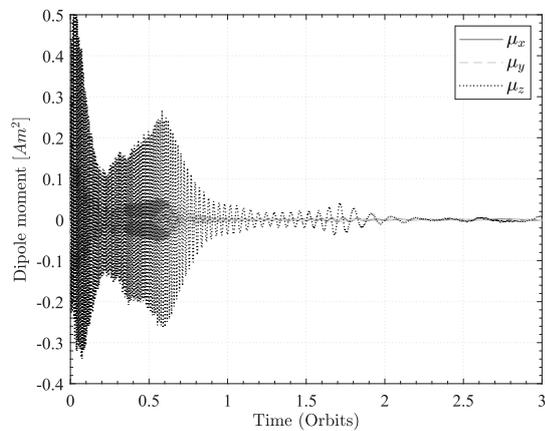
Figure 6.9: *Dipole moment*

Monte Carlo Simulation

A Monte Carlo simulation was carried out to test the working of the detumbling phase controller for different initial conditions. Figure 6.10 shows the final summary result of 30 detumbling simulations. The angular velocities magnitude gives a general indication of the value of angular velocities during detumbling, but its value does not ensure that each angular velocity component meets the requirement. For this reason, the fulfilment of the requirement was verified at the end of the simulation.
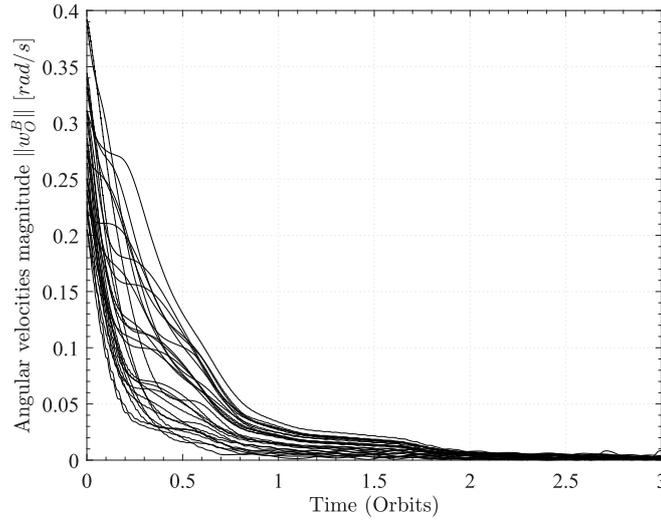
Figure 6.10: *Angular velocities magnitude resulting from the Monte Carlo simulation*

6.3   Nadir Pointing Simulation

This section presents the results from the performance comparison of the configurations presented in Figure 6.5. The comparison was performed based on the extrinsic indicators defined in Chapter 5 and considering other aspects characterising the control efficiency as, for example, the cost of the attitude manoeuvre. Three simulations were carried out with the same initial conditions, shown in Table 6.3, one for each configuration considered.

Orbit characteristics	Circular orbit, Altitude = 470 km, Inclination = $51.6°$
Initial attitude	$[\Phi, \quad \Theta, \quad \Psi] = [-15, \quad 25, \quad 10]\ [deg]$
Initial angular rates	$[\omega_x, \quad \omega_y, \quad \omega_z] = [0, \quad 0.0011, \quad 0]\ [rad/s]$
Rates	Control rate = 0.5 s, Sampling rate = 0.1 s
Controller gains	
H_∞	Same weights as in Table 5.5
MPC	$Q = 1000 \cdot \mathcal{I}_n$, $R_w = 1000 \cdot \mathcal{I}_m$, $N_p = 10$, $N_c = 7$
Filter matrices	Same matrices as in Table 4.5
Filter initial condition	Two cases considered: *a*) $\hat{x}_0 = \bar{0}$ and *b*) $\hat{x}_0 = \bar{x}_0$

Table 6.3: *Nadir pointing simulation setup*

Two cases were considered for the initial state vector of the filter: one case (*a*) where the initial state vector is zero, and another (*b*) where the vector coincides with the true quantities. These two cases allow highlighting the behaviour of the controllers when the state estimate is incorrect.
Each simulation consists of using the previously discussed algorithms to control the spacecraft to reach the nadir pointing attitude from the specified initial conditions. These simulations were carried out over a time span of approximately 1000 seconds, but the plots are presented over shorter time intervals to highlight the trend of some variables. Since in the SVD + H_∞ configuration there is no estimation of the bias in the angular velocity measurement, an additional filter is used solely for this purpose so that the results are comparable and do not diverge.

The first results presented are those for quaternions and angular velocities shown in Figures 6.11 and 6.12. One difference that is immediately noticeable concerns how the quantities approach steady-state values.
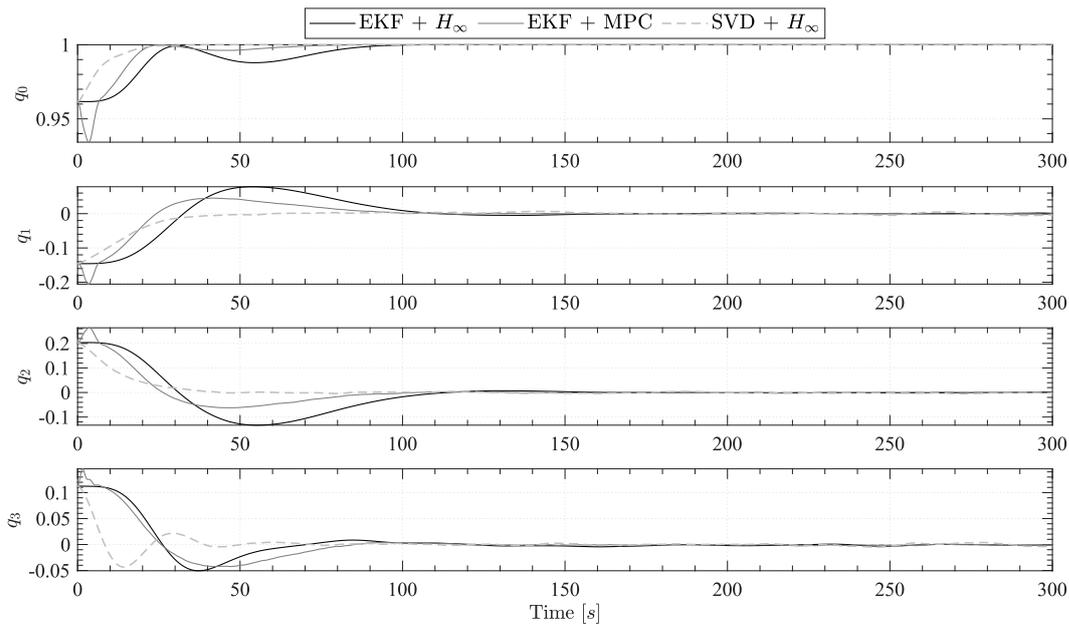
Figure 6.11: *Comparison of the quaternions resulting from the nadir pointing simulation. The diagram refers to the case named a*
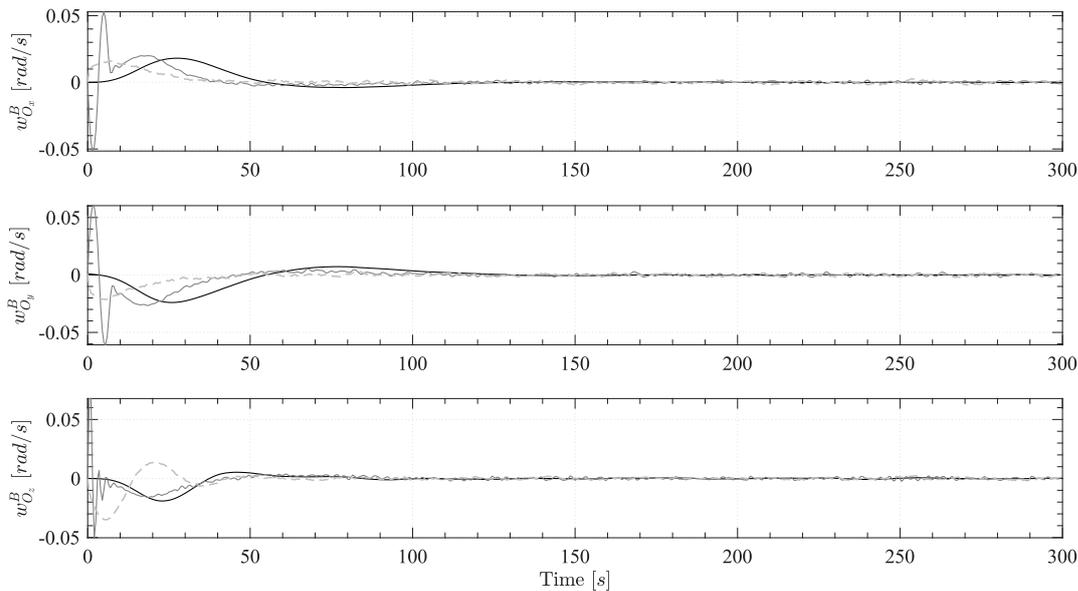

Figure 6.12: *Comparison of the spacecraft angular velocities resulting from the nadir pointing simulation. The diagram refers to the case named a*

The latter aspect depends on many factors including the control authority of the controller and the estimated spacecraft state. For example, although the H_∞ controller has the same tuning, it performs differently in the two configurations with EKF and SVD. This is easily explained by the fact that while with SVD the controller directly receives the noisy measurement of the quaternion and angular velocity (from the gyroscopes), which minus the noise follows the true value, with EKF it receives the estimate which may not coincide with the true value. This difference is highlighted by diagrams B.5 and B.6 in the Appendix which show the same simulation but with the filter's initial state vector coinciding with the true values. From these diagrams, it can be seen that the

114

quaternion trend for the SVD+H_∞ and EKF+H_∞ configurations is almost the same. Furthermore, the trend for the EKF+MPC configuration is much more direct in driving the state to the desired value. Further confirmation of this is provided by diagrams 6.13 and 6.14 for case *a* and B.7 and B.8 for case *b* that show the performance of the Extended Kalman Filter, for quaternions and angular velocities, in the two cases differentiated by the initial state vector. The results show that in the case of the EKF+MPC configuration, the filter struggles to estimate the state as the controller attempts to send it quickly to the desired based on the input it receives. It also happens to a lesser extent in case *b*, where the filter starts with perfect knowledge of the state at that instant.
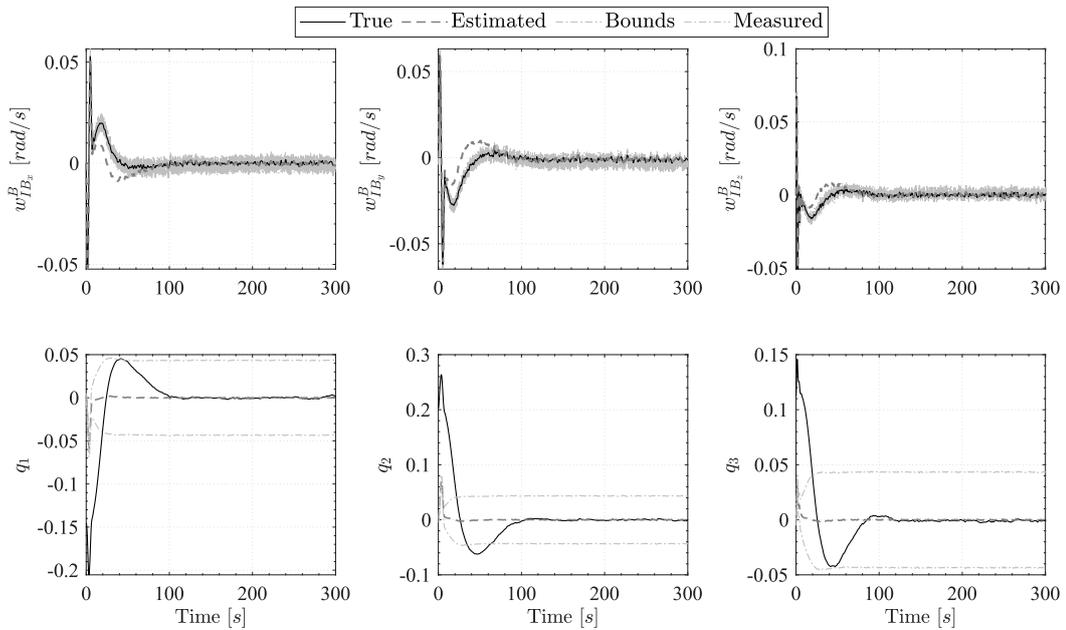
**Figure 6.13:** *EKF+MPC filter results from the nadir pointing simulation. The diagram refers to the case named a*
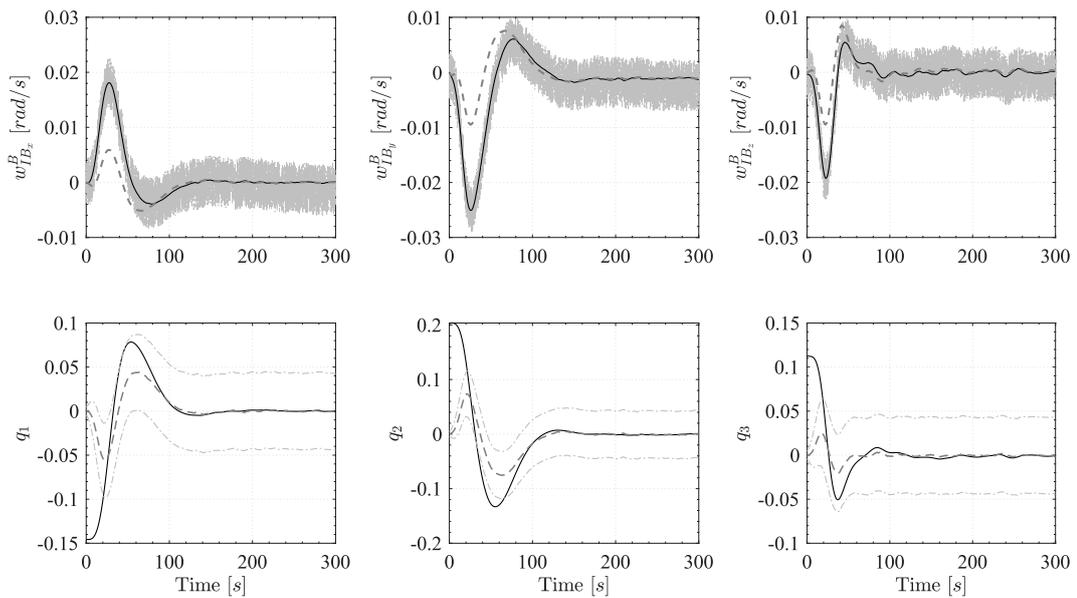

Figure 6.14: *EKF+H_∞ filter results from the nadir pointing simulation. The diagram refers to the case named a*

115

The fact that the state is driven faster to the desired and that the filter fails to converge quickly indicates that the MPC controller exerts more costly control. Although the MPC controller has been tuned to reduce the control authority, it still delivers high torques as shown in Figure 6.15. However, due to the inclusion of constraints in the optimisation, characteristic of the MPC controller, the reaction wheels do not reach the saturation value. In contrast, in configurations that include the H_∞ controller, the control torque is lower and less demanding for the reaction wheels.
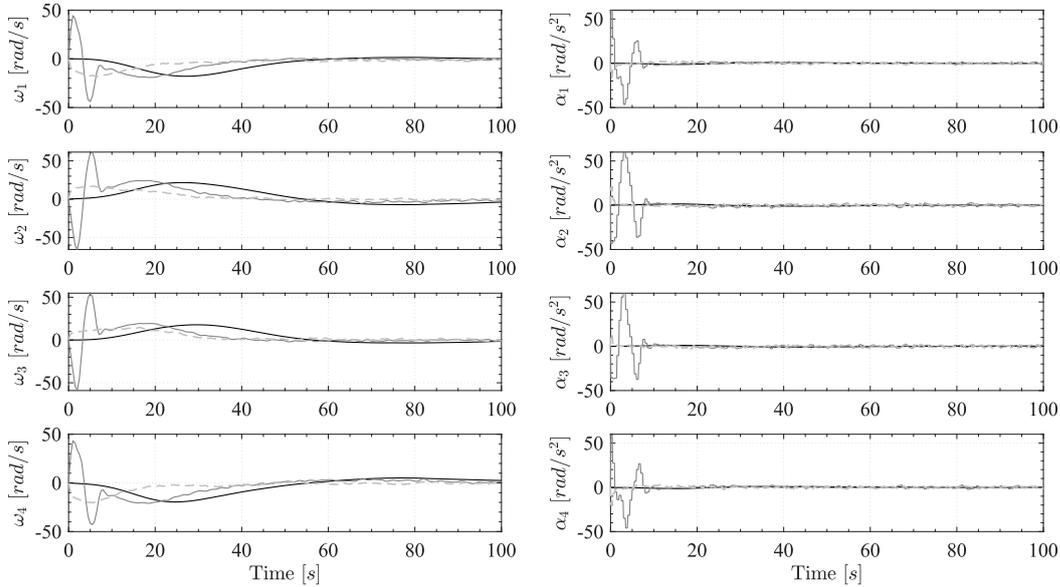

Figure 6.15: *Comparison of the reaction wheels' accelerations and angular velocities resulting from the nadir pointing simulation. The diagram refers to the case named a. The legend is the same as in Diagram 6.11*

From these initial comparisons, it can be concluded that the EKF+MPC configuration is characterised by a more costly control when compared to the configurations with the H_∞ controller, but that allows better performance in terms of, for example, rise time[†] (see Figure B.5). Extrinsic indicators APE, AKE, PSE, and the control error are considered to more closely investigate and compare the performance of these configurations. The definition of these indicators is given in Table 3.2.1, while how they can be calculated mathematically is discussed in [4].

ADCS requirements imposed by the payload were discussed in Section 5.1 and reported, in terms of angles, in Table 3.8. For this reason, it is convenient to shift the focus to Euler angles rather than quaternions as they have an immediate interpretation and allow easy verification of whether the requirement is met.

Figures 6.16, 6.17, 6.18, and 6.19 show these indicators with a focus on the steady-state region. The diagrams also show the boundaries within which the requirement is fulfilled. To calculate the indicators in the case of the SVD+H_∞ configuration, the output of the SVD method is assumed to be the same as the estimate in the case of the EKF.

From Figure 6.16, it can be seen that approximately the EKF+MPC and EKF+MPC configurations perform similarly in terms of Absolute Knowledge Error. In contrast, the SVD+ H_∞ configuration does not meet the requirement stably. Consequently, it indicates that the application of the Extended Kalman Filter results in a better estimation than the application of the SVD method alone.

[†]Time required for the state to reach 90% of the desired steady state value

116
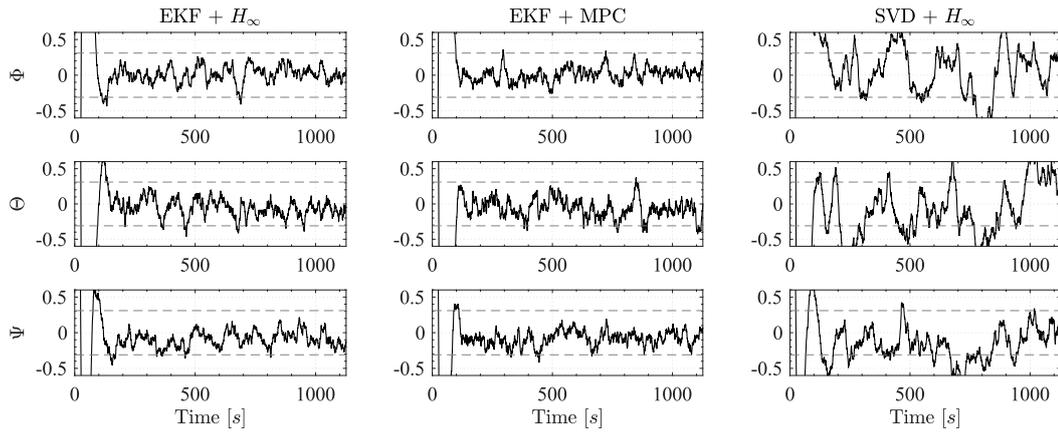
Figure 6.16: *Absolute Knowledge Error (AKE) on Euler angles for the three config-
uration, resulting from the nadir pointing simulation. The boundaries
indicate the region in which the requirement defined in Table 3.8 is met*
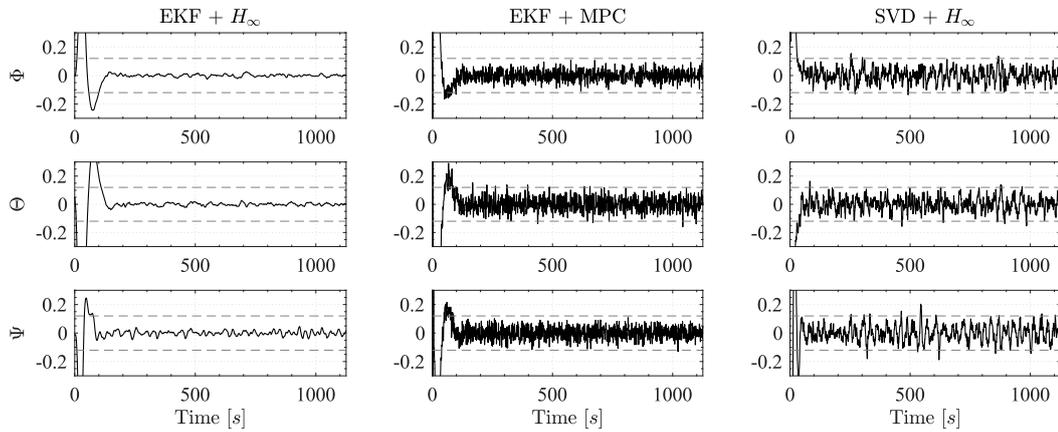
Figure 6.17: *Performance Stability Error (PSE) on Euler angles for the three config-
uration, resulting from the nadir pointing simulation. The boundaries
indicate the region in which the requirement defined in Table 3.8 is met*
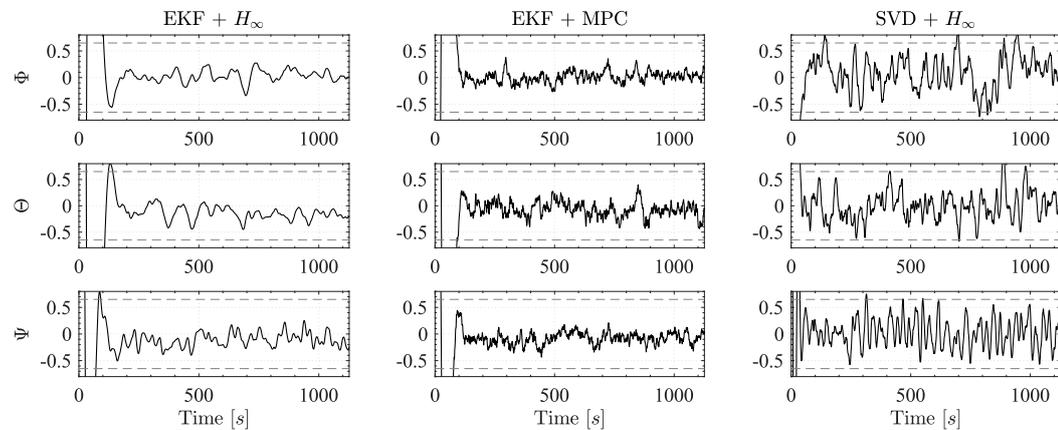
Figure 6.18: *Absolute Performance Error (APE) on Euler angles for the three con-
figuration, resulting from the nadir pointing simulation. The boundaries
indicate the region in which the requirement defined in Table 3.8 is met*

117
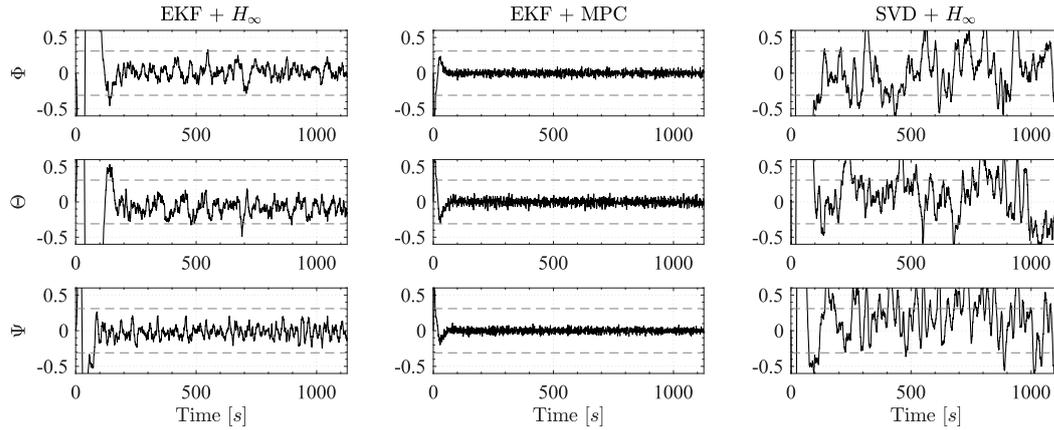
Figure 6.19: *Control Error (c) on Euler angles for the three configuration, resulting from the nadir pointing simulation. The boundaries indicate the region in which the requirement defined in Table 3.8 is met*

However, it should be emphasised that the latter comparison is only possible due to the noise-attenuation properties of the H_∞ controller. The SVD method, in contrast to the EKF, cannot handle measurement noise, so a noisy input produces a noisy output. The same configuration but with the MPC controller instead of the H_∞ controller would not even be able to stabilise the spacecraft.

A point in favour of the EKF+MPC configuration can be seen in Figure 6.19, which shows the control error. It is easy to see that the MPC controller allows for a lower control error than the other two configurations. As a matter of fact, this is one of the advantages of adopting an optimal control law. When looking at the remaining configurations, it is clear that in the EKF+H_∞ case, the requirement is fulfilled to a large margin, whereas the SVD+H_∞ configuration fails to fit within the boundaries.

The real advantage of combining the Extended Kalman Filter and H_∞ optimal output feedback controller can be seen in Figure 6.17. The figure shows the Performance Stability Error (PSE), thus the difference between the instantaneous performance error at a given time t and the error value at an earlier time $t - \delta t$ [3]. It is immediately noticeable that the EKF+H_∞ configuration has a much smaller PSE than the other two and exhibits a trend with less noise, indicating that the properties of the H_∞ controller combined with those of the EKF result in good stabilisation. In contrast, the EKF+MPC combination is the worst but at least meets the PSE requirement.

From Figures 6.16 and 6.19, it can be seen that both the EKF+MPC and EKF+H_∞ configurations meet the requirements on AKE and control error, while the SVD+H_∞ combination does neither. From what was said earlier about APE in Section 3.2.1, it would seem that the EKF+MPC and EKF+H_∞ combinations undoubtedly fulfil the requirement on APE while the SVD+H_∞ configuration does not. However, seeing APE as the sum of AKE and control error is a worst-case view and can therefore be seen as a sufficient condition for compliance with the requirement. It means that if both AKE and control error meets the requirement then certainly APE also fulfils it. The reverse is not valid in the sense that if both AKE and control error does not meet the requirement this does not mean that APE does not fulfil the requirement as well. The diagram in Figure 6.18 confirms the above by showing that all three configurations meet the Absolute Performance Error requirement. Also in this case, it can be seen that the EKF+H_∞ combination results in an APE with less oscillation and noise.

A final comparison can be made by considering the mean indicators calculated over the entire

118

period in which the state is stabilised. The value of these indicators calculated for the three configurations is shown in Table 6.4.

	Mean Performance Error			Mean Knowledge Error		
	Φ	Θ	Ψ	Φ	Θ	Ψ
EKF + H_∞	0.03564	-0.04505	-0.07655	0.01224	-0.06311	-0.07898
EKF + MPC	0.01493	-0.07093	-0.08264	0.01475	-0.07020	-0.08178
SVD + H_∞	0.07059	-0.00227	0.02172	0.03814	-0.10021	-0.16425

Table 6.4: *Mean extrinsic indicators calculated for the three configurations*

The results show that the EKF+H_∞ configuration has the lowest MKE and MPE errors. Taking into account all the EKF+H_∞-related aspects discussed so far: the best performance in terms of stability, the lowest cost of control while still obtaining good performance, and having the smallest MKE and MPE errors, it can be concluded that the EKF+H_∞ configuration is the best of the three. Other factors leading to the selection of this configuration are the greater control robustness due to the H_∞ controller and the simplicity of practical implementation. As seen in Chapter 5, the Model Predictive Control requires solving an optimisation problem at each sampling time; in contrast, the H_∞ controller requires offline synthesis and can then be implemented discretely without requiring any special numerical arrangements. Naturally, this results in a lower computational cost, which is very important in practice.

Monte Carlo Simulation

Once the combination of algorithms providing the best performance was chosen, it was validated with a Monte Carlo simulation. The results of the 30 simulations are shown in Figures 6.20, 6.21 and 6.22. The simulations were carried out by setting the initial condition of the Extended Kalman Filter state vector equal to zero. This was also done to test the ability to converge to the desired state from an incorrect state estimate. From Figure 6.22, it can be seen that all requirements are met with a few exceptions.
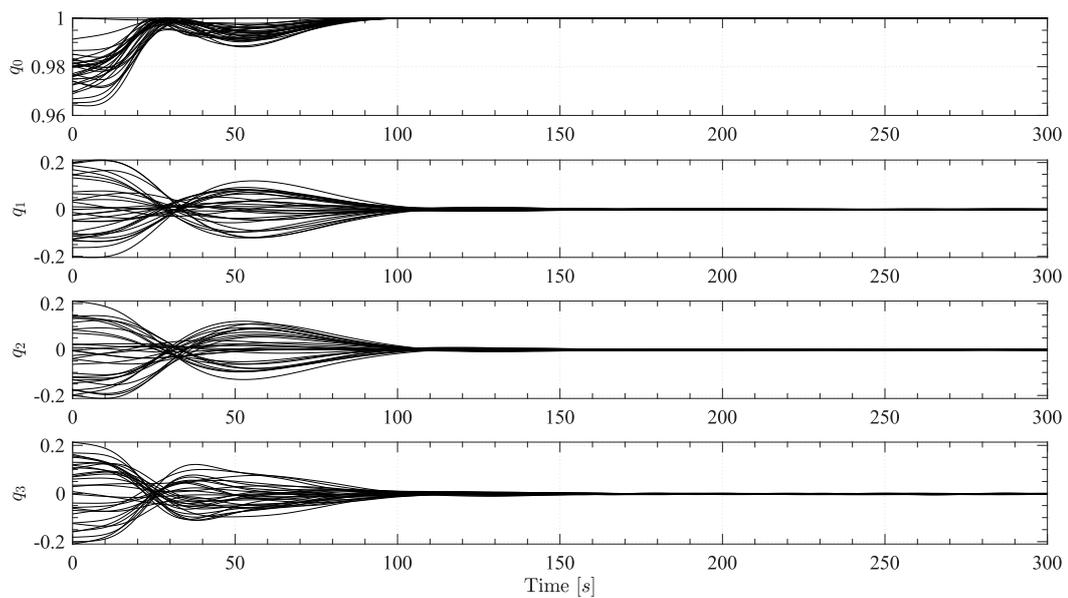
Figure 6.20: *Quaternions resulting from the nadir pointing Monte Carlo simulation*
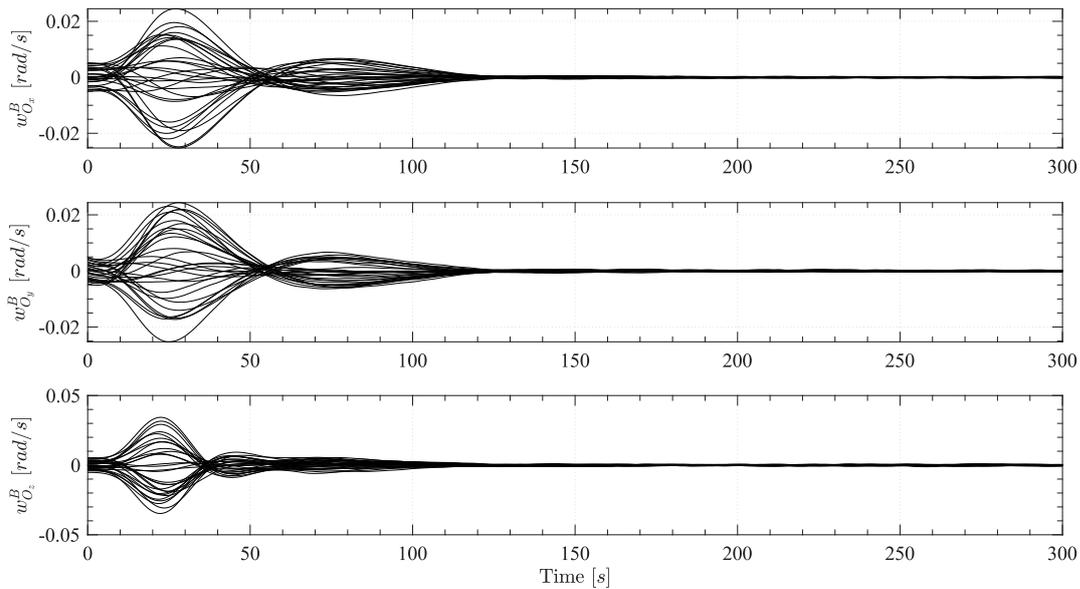
Figure 6.21: *Angular velocities resulting from the nadir pointing Monte Carlo simulation*
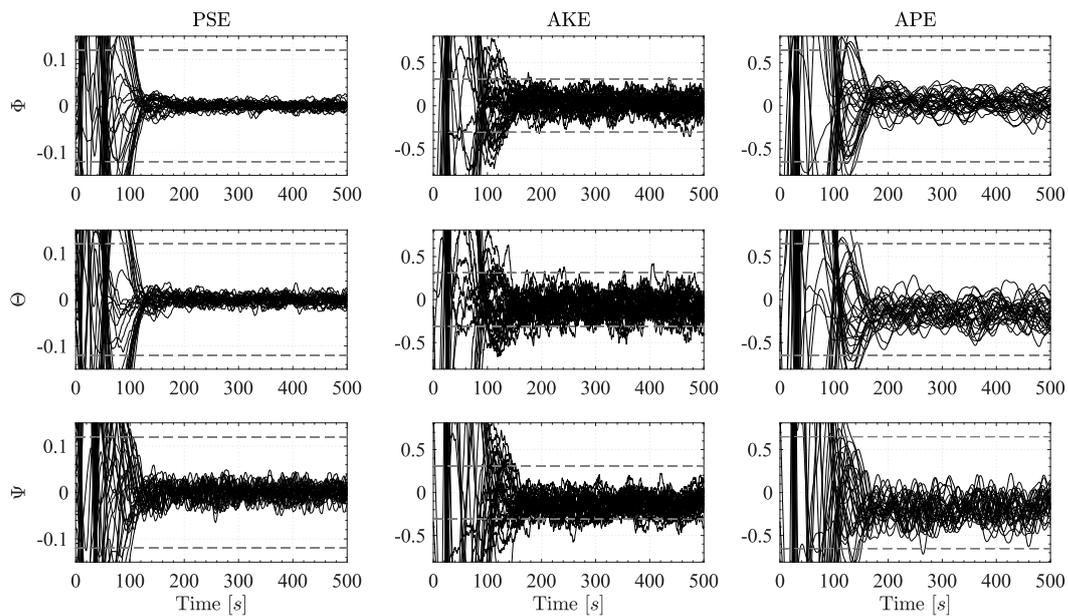
Figure 6.22: *Extrinsic indicators resulting form the nadir Monte Carlo simulation*

6.4   Target Pointing Simulation

This section presents the results of a non-ideal target pointing simulation with noise in sensor measurements. The initial conditions used in the simulation are the same as those used in the ideal simulation to obtain comparable results. Therefore, a target located at the satellite's ground track is used in the simulation. If the target is so placed, then the inclination of the orbit does not affect the target pointing mode, and this result can be extended to all orbit inclinations. The situation changes when the target is not at the ground track but offset. In the latter case, the spacecraft must
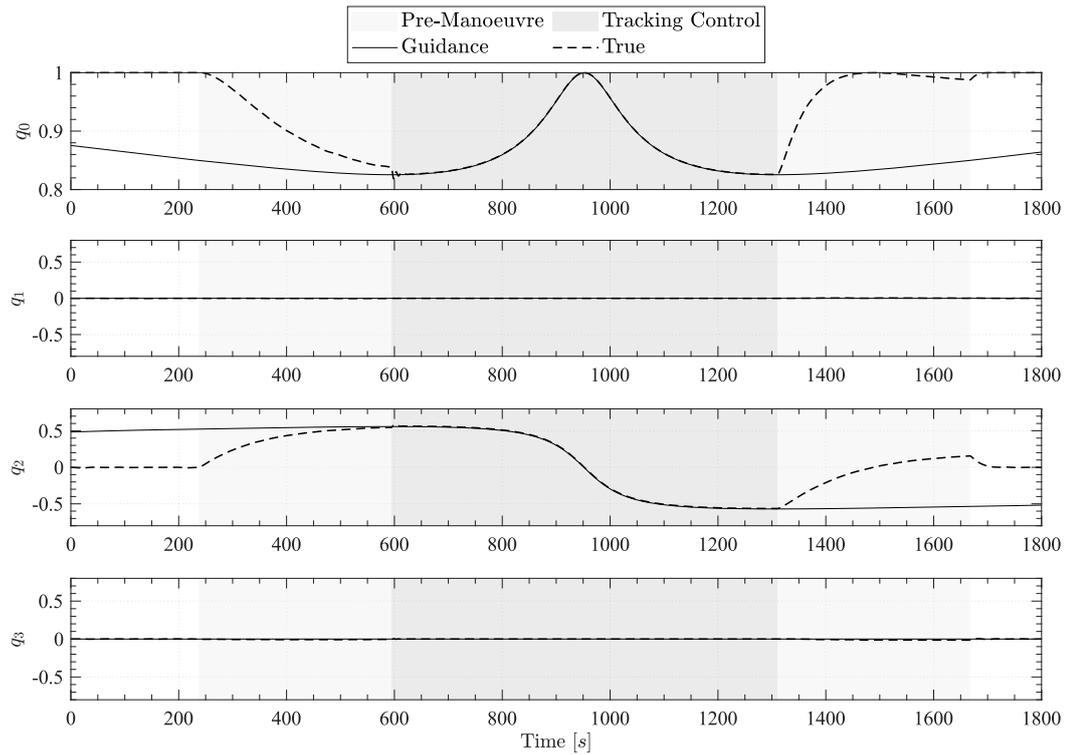
Figure 6.23: *Quaternions resulting from the target pointing simulation*

perform a slew manoeuvre implying angular velocity along both the y and x-axis.

As can be seen from Figure 6.23, the simulation results are very similar to those presented in Section 5.4.1. In the non-ideal case, the quaternion feedback controller receives as input the estimated state vector output of the EKF; thus, good results depend on valid estimates. Figure 6.24 shows that the filter correctly estimates quaternions, angular velocities, and bias on the measurement.
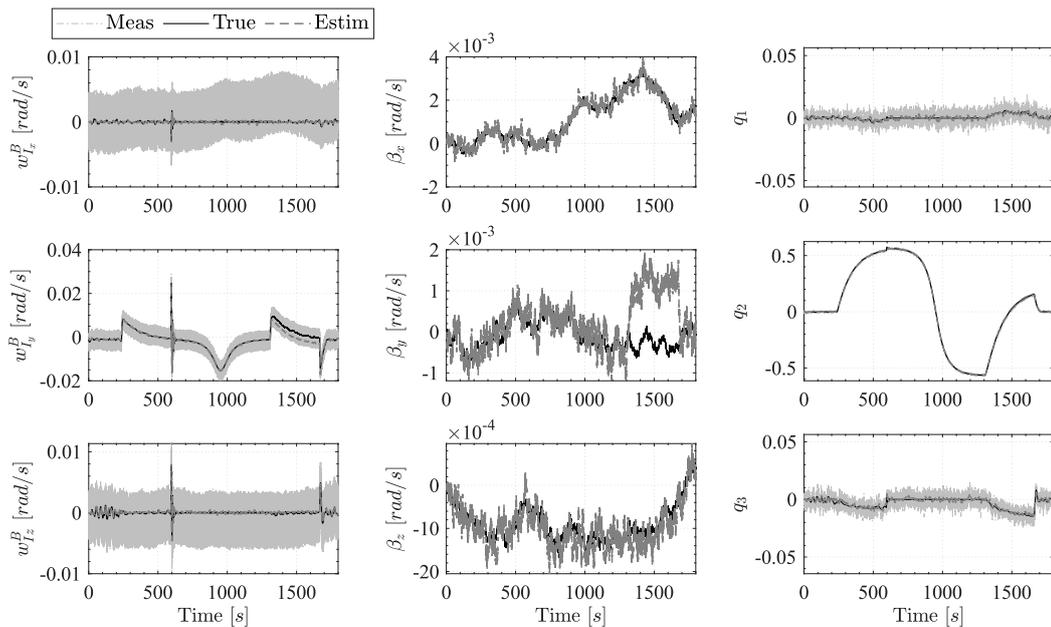
Figure 6.24: *Extended Kalman Filter results for the target pointing simulation*

From Figure 6.24 it can be seen that the angular velocity measurement is very biased, but the filter still manages to provide a reasonable estimate. At the most sudden manoeuvres, the filter struggles to work correctly but soon afterward converges.
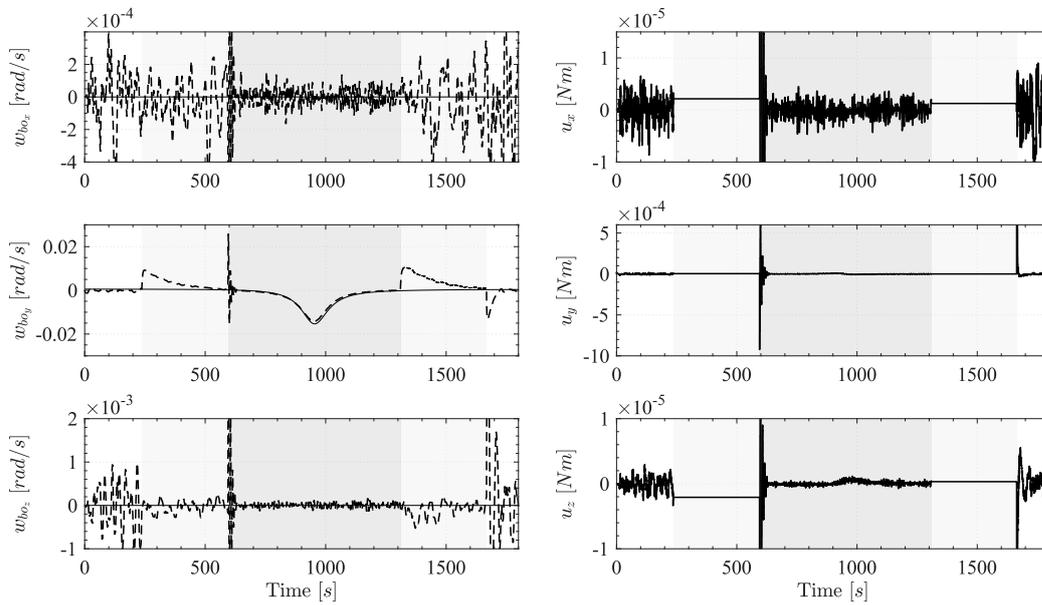
Figure 6.25: *Angular velocities and torques resulting from the target pointing simulation*
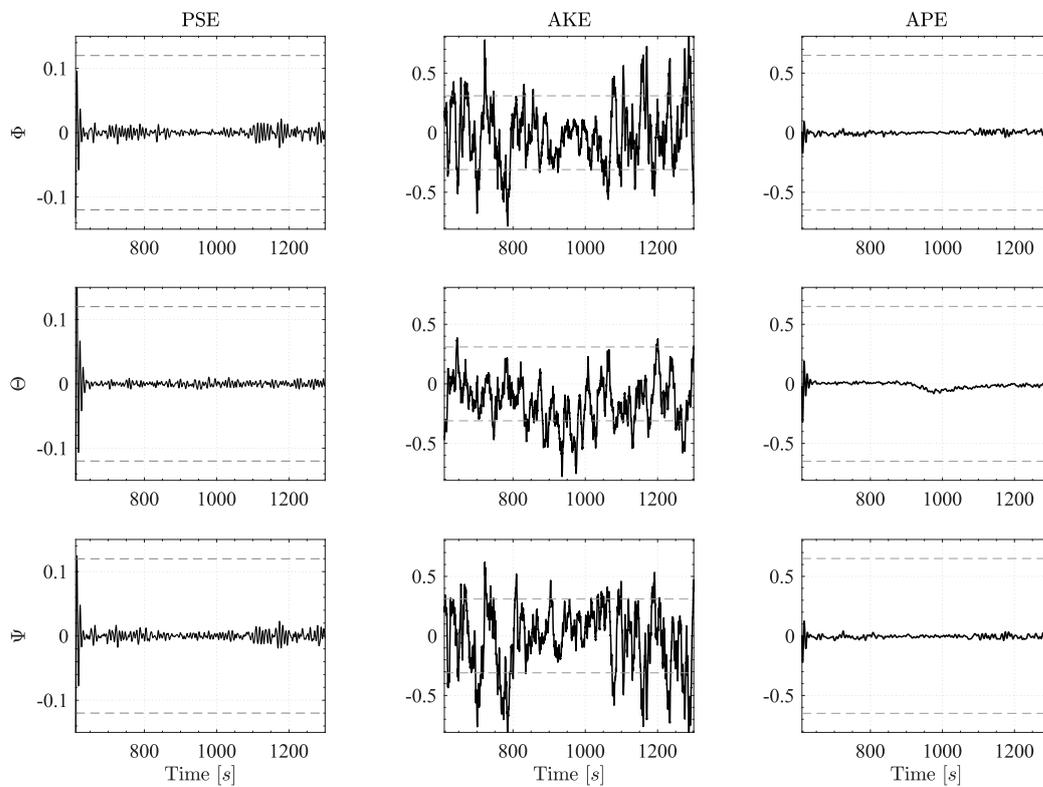
Figure 6.26: *Extrinsic indicators AKE, APE, PSE resulting from the target pointing simulation*

The extrinsic indicators shown in Figure 6.26 indicate that the APE and PSE requirements are met while the AKE goes out of bounds at some points during the tracking manoeuvre.

Finally, the diagram in Figure 6.27 highlights that due to the pre- and post-target tracking manoeuvre, the control torque required by the reaction wheels remains very low.
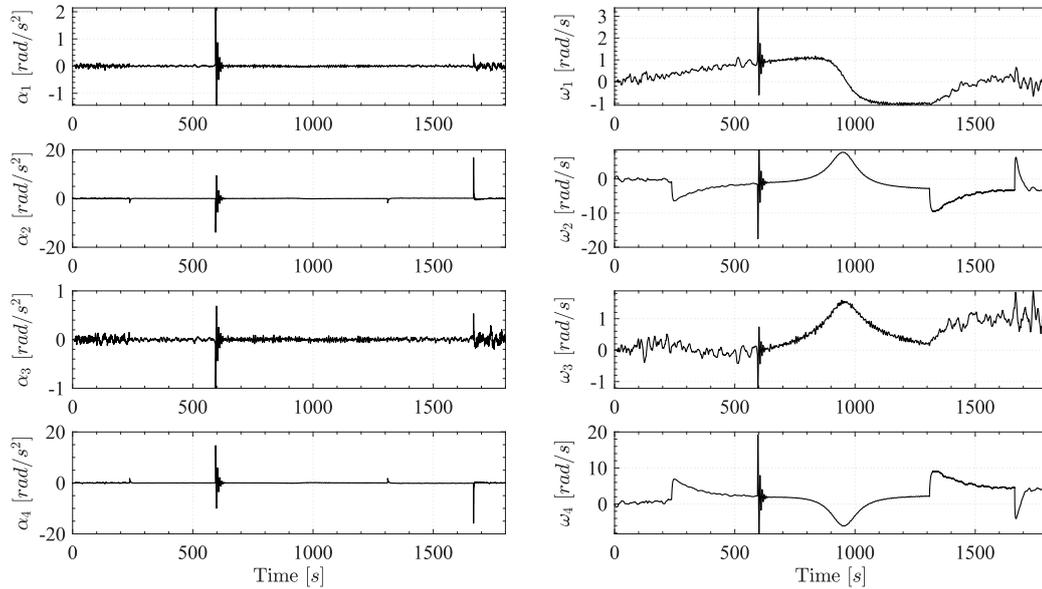
Figure 6.27: *Accelerations and angular velocities of reaction wheels resulting from the target pointing simulation*

Chapter 7

Closure

The work discussed in this thesis concerns the design of the Attitude Determination and Control system for a 3U CubeSat used in an Earth observation mission. All aspects were considered for the successful design, from developing an orbital simulator for preliminary simulations to selecting and sizing the hardware and software.

7.1 Conclusions

Simulations play an essential role in understanding the spacecraft's performance in orbit. The developed orbital simulator on MATLAB®-Simulink® allows the position and attitude dynamics of the satellite to be simulated utilizing fairly well-established mathematical models. Cowell's method has been used to develop the orbital propagator for position dynamics simulation. The perturbations included in the propagator are aerodynamic drag due to the Earth's residual atmosphere, solar radiation pressure, acceleration due to the Sun and Moon gravity, and the gravitational field generated by the non-perfectly spherical Earth.
A simple eclipse model was implemented to simulate the correct operation of the sun sensors, while the IGRF 13 model was used for the Earth's magnetic field.
The attitude dynamic is simulated according to the well-known Euler Equation. Also, in this case, the various disturbance torques acting on the satellite are considered, including gravity-gradient, magnetic field, solar radiation, and aerodynamic torque.

The design process starts with a functional analysis of the system, followed by design steps focusing more on the hardware and software aspects of the system, and eventually verification of the final result through simulations. The functional analysis made it possible to highlight the system's principal functions and to identify an initial set of functional and interface requirements with other subsystems and between the system's components.
The main design steps involved defining the control modes for the different mission phases and associating some of them with pointing requirements. In these phases, therefore, it was essential to study the performance of the payload and the mission to derive the pointing requirements that the ADCS must satisfy to obtain satisfactory results. After knowing the mission characteristics, it was possible to set up an initial estimate of the magnitude of the disturbances acting on the spacecraft due to the space environment and thus be able to select a compatible hardware suite. Hardware selection, indeed, was based on the requirements imposed on the system and on trade-off processes carried out to choose the best alternatives.
A suite consisting of 6 coarse sun sensors (represented by photodiodes), a 3-axis magnetometer, and an IMU was chosen for the sensors. Coarse sun sensors are arranged in the faces of the satellite. A 3-axis magnetometer allows to measure the local vector of the Earth's magnetic field, its operation is alternated with the operation of the magnetic torquers in order to avoid interference.

The MEMS IMU consists of 3 gyroscopes and 3 accelerometers. The gyroscopes allow to measure speed of rotation from an initial reference, but without any knowledge of an external, absolute reference.

Reaction wheels and magnetic torquers were chosen as actuators. Reaction wheels control provide smooth changes in torque, allowing very accurate pointing of spacecraft. In case of the primary wheels fails a fourth reaction wheel is carried for redundancy as hot backup. The wheel configuration chosen is the pyramidal layout. The magnetic torquers are used during the detumbling phase to dampen the angular velocity of the satellite below a certain threshold imposed by the requirement. They are also used to desaturate the reaction wheels after some operating cycles.

The study of the determination algorithms available today has led to the selection of the Extended Kalman Filter for the recursive estimation of the attitude and bias in angular velocity measurement and the Singular Value Decomposition method for deterministic attitude determination. In particular, the SVD method output serves as attitude innovation in the EKF. Furthermore, the attitude determination process also includes a smoothing filter for pre-filtering noise from the sun sensors and magnetometer measurements. Orbit-referenced angular rates during the detumbling phase are estimated with a robust Rate Kalman Estimator using magnetometer measurements.

The results of the simulations shows that with the selected algorithms, the Absolute Knowledge error requirement of 0.31 deg is met in most cases.

The controllers required for detumbling and reaction wheels desaturation are implemented via the Cross Product Magnetic control law with magnetic torquers. The combination of the detumbling magnetic control law with the Rate Kalman Estimator allows the satellite to be detumbled in approximately 3 orbits while meeting the Absolute Performance Error requirement on angular velocities of 0.2 deg/s.

Control during the science mode takes place using reaction wheels. A classical Quaternion feedback controller is used for the target pointing phase, while special attention focuses on advanced control algorithms implementation, including Model Predictive Control and H_∞ control, for the Nadir Pointing mode. Different combinations between the optimal estimator and optimal controllers were studied to test their performance. The obtained results show that the EKF+H_∞ combination meets to a large margin the requirements and has the best performance in terms of stability and mean extrinsic indicators; Therefore has been employed and implemented in the system.

7.2 Recommendations

Further work for the development of the SILVA ADCS involves hardware testing. The algorithms used in the simulations were written to facilitate the conversion into the C programming language to be implemented later in the hardware. Therefore, the next step is to test the individual hardware components and then realise the so-called Hardware In The Loop simulation to properly observe system behaviour.

To further improve the reliability of the simulations, it is possible to implement more accurate sensor and actuator models that also take into account more advanced aspects that may deteriorate system performance.

A further point that needs revision is the knowledge of the spacecraft position for the determination algorithms. In the simulations performed, it has been assumed that the spacecraft position is known without errors. However, in practice, even this measurement will be affected by errors and noise and therefore needs a dedicated filter. In the design process, the GNSS receiver was selected as a hardware part of the system for satellite position measurement. Therefore, a suitable Kalman Filter applied to the GNSS receiver measurements is a possible practical solution.

Appendix A

Linearisation of the Mathematical Model

The linearisation point is selected as given in the following equations

$$\bar{q} = \begin{bmatrix} q_0 \\ \vec{q} \end{bmatrix} = \bar{q} = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \approx \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{A.1}$$

Inertial-referenced angular velocity is given by

$$\bar{\omega}_{IB}^B = \bar{\omega}_{OB}^B + \boldsymbol{A}_O^B \bar{\omega}_{IO}^O \tag{A.2}$$

where

$$A_O^B \bar{\omega}_{IO}^O = -\omega_0 \begin{bmatrix} 2(q_1 q_2 + q_0 q_3) \\ q_0^2 - q_1^2 + q_2^2 - q_3^2 \\ 2(q_2 q_3 - q_0 q_1) \end{bmatrix} \approx -\omega_0 \begin{bmatrix} 2q_3 \\ 1 \\ -2q_1 \end{bmatrix} \tag{A.3}$$

It follows that

$$\bar{\omega}_{IB}^B = \begin{bmatrix} \omega_x - 2\omega_0 q_3 \\ \omega_y - \omega_0 \\ \omega_z + 2\omega_0 q_1 \end{bmatrix} \tag{A.4}$$

The kinematic equation which relates quaternions to angular rates is given by

$$\dot{q} = \frac{1}{2} \begin{bmatrix} -q_1 & -q_2 & -q_3 \\ q_0 & -q_3 & q_2 \\ q_3 & q_0 & -q_1 \\ -q_2 & q_1 & q_0 \end{bmatrix} \bar{\omega}_{OB}^B \approx \frac{1}{2} \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \bar{\omega}_{OB}^B \approx \begin{bmatrix} 0 \\ \frac{1}{2} \bar{\omega}_{OB}^B \end{bmatrix} \tag{A.5}$$

Only the disturbance torque due to the gravity gradient is considered in the Euler Equation 2.69, which is given by Equation 2.76. The torque can be linearised as

$$\bar{T}_{gg}^B = 3\,\omega_0^2 \begin{bmatrix} (I_z - I_y)2\,q_1 \\ (I_x - I_z)2\,q_2 \\ 0 \end{bmatrix} \tag{A.6}$$

By substituting in the Euler equation gives

$$\begin{aligned} I_x \dot{\omega}_x - \omega_0(I_z + I_x - I_y)\omega_z + 8\omega_0^2(I_y - I_z)q_1 &= u_x \\ I_y \dot{\omega}_y + 6\omega_0^2(I_x - I_z)q_2 &= u_y \\ I_z \dot{\omega}_z + \omega_0(I_z + I_x - I_y)\omega_x + 8\omega_0^2(I_y - I_x)q_3 &= u_z \end{aligned} \tag{A.7}$$

where $\bar{u} = \begin{bmatrix} u_x & u_y & u_z \end{bmatrix}$ are the components of the command torque.

Considering the state vector $\bar{x} = \begin{bmatrix} q_1 & q_2 & q_3 & \omega_x & \omega_y & \omega_z \end{bmatrix}^T$ the linearized state space model can be written as

$$\dot{\bar{x}} = \boldsymbol{A}\bar{x} + \boldsymbol{B}\bar{u} \tag{A.8}$$

where

$$\boldsymbol{A} = \begin{bmatrix} 0 & 0 & 0 & 1/2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/2 \\ -8\omega_0^2 \frac{I_y - I_z}{I_x} & 0 & 0 & 0 & 0 & \omega_0 \frac{I_z + I_x - I_y}{I_x} \\ 0 & -6\omega_0^2 \frac{I_x - I_z}{I_y} & 0 & 0 & 0 & 0 \\ 0 & 0 & -2\omega_0^2 \frac{I_y - I_x}{I_z} & -\omega_0 \frac{I_z + I_x - I_y}{I_z} & 0 & 0 \end{bmatrix} \tag{A.9}$$

$$\boldsymbol{B} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1/I_x & 0 & 0 \\ 0 & 1/I_y & 0 \\ 0 & 0 & 1/I_z \end{bmatrix} \tag{A.10}$$

Appendix B

Other Results

Error Sources	Error Magnitude (units)	Magnitude of Mapping Error (km)	Magnitude of Pointing Error (rad)	Direction of Error
Attitude Errors:[1]				
Azimuth	$\Delta\phi$ (rad)	$\Delta\phi\, D \sin\eta$	$\Delta\phi \sin\eta$	Azimuthal
Nadir Angle	$\Delta\eta$ (rad)	$\Delta\eta\, D / \sin\varepsilon$	$\Delta\eta$	Toward nadir
Position Errors:				
In-Track	ΔI (km)	$\Delta I\ (R_T/R_S) \cos H$ [2]	$(\Delta I/D) \sin Y_I$ [5]	Parallel to ground track
Cross-Track	ΔC (km)	$\Delta C\ (R_T/R_S) \cos G$ [3]	$(\Delta C/D) \sin Y_C$ [6]	Perpendicular to ground track
Radial	ΔR_S (km)	$\Delta R_S \sin\eta / \sin\varepsilon$	$(\Delta R_S/D) \sin\eta$	Toward nadir
Other Errors:				
Target altitude	ΔR_T (km)	$\Delta R_T / \tan\varepsilon$	-	Toward nadir
S/C Clock	Δt (s)	$\Delta T\, V_e \cos(lat)$ [4]	$\Delta T\, (V_e/D) \cos(lat) \cdot \sin J$ [7]	Parallel to Earth's equator

Notes:
(1)   Includes attitude determination error, instrument mounting error, stability over exposure time (mapping only), and control error (pointing only). The formulas given assume that the attitude is measured with respect to the earth

(2) $\sin H = \sin\lambda \sin\phi$ \qquad\qquad (5) $\cos Y_I = \cos\phi \sin\eta$

(3) $\sin G = \sin\lambda \cos\phi$ \qquad\qquad (6) $\cos Y_C = \sin\phi \sin\eta$

(4) $V_e = 464\ m/s$ $(Earth\ rotation\ velocity\ at\ equator)$ \qquad (7) $\cos J = \cos\phi_E \cos\varepsilon$ $where$ $\phi_E = azimuth\ relative\ to\ East$

Table B.1: *Mapping and pointing error formulas, the nomenclature refers to Figure 3.8. Adapted from [45]*

	Elevation Angle. ε (deg)	Spacecraft Altitude. H (km)	Target Latitude. lat (deg)*	Target Azimuth. Φ (deg)*	Azimuth Relative to East. Φ_E (deg)*
Case 1	89.99	700	0	0	90
Case 2	27.00	700	0	0	90

	R_S (km)	D (km)	Nadir Angle. η (deg)	Earth Central Angle. λ (deg)
Case 1	7.078	700	0,0	0,0
Case 2	7.078	1.324	53,4	9,6

Table B.2: *Data used in the computation of the mapping and pointing error budget. *Values are used for maximum errors*

Source	Error In Source		Error Budgets			
			Mapping Error (km)		Pointing Error (deg)	
			Case 1	Case 2	Case 1	Case 2
Attitude Errors:						
Azimuth	0.25	*deg*	0,000	4,638	0,0000	0,2007
Nadir Angle	0.2	*deg*	2,443	10,179	0,2000	0,2000
$RSS \cdot 1.1 \cdot 2 =$					0,4400	0,6234
Position Errors:						
In-Track	0.1	*km*	0,090	0,090	0,0082	0,0026
Cross-Track	0.1	*km*	0,090	0,089	0,0082	0,0043
Radial	0.1	*km*	0,000	0,177	0,0000	0,0035
$RSS \cdot 1.1 =$					0,0127	0,0067
Other Errors:						
Target Altitude	0.01	*km*	0,000	0,020	-	-
S/C Clock	0.5	*sec*	0,233	0,233	0,0190	0,0101
$RSS \cdot 1.1 =$					0,0209	0,0111
Total Sum:			**2,458**	**11,190**	**0,4737**	**0,6412**

Table B.3: *Mapping and pointing error budget for the 3U CubeSat. Grey cells represent the input for the calculation*

	Value	Unit	Name
Φ	$= 1.366$	W/m^2	Solar constant
c	$= 3 \cdot 10^8$	m/s	Speed of light
A_s	$= 0.034$	m^2	Sunlit surface area
q	$= 0.263$	-	Reflectance factor
φ	$= 45°$	deg	Angle of incidence of the Sun
b_{srp}	$= 0.2$	m	Solar radiation pressure arm $(cp_s - cm)$
ρ	$= 2.72 \cdot 10^{-14}$	kg/m^3	Atmospheric density at 470 km of altitude
C_d	$= 2.2$	-	Drag coefficient
A_r	$= 0.034$	m^2	Ram area
V	$= 7.50 \cdot 10^3$	m/s	Spacecraft's orbital velocity
b_a	$= 0.2$	m	Center of aerodynamic pressure arm $(cp_a - cm)$
D	$= 0.08$	Am^2	Spacecraft's residual dipole moment
B	$= 5.11 \cdot 10^{-5}$	T	Magnetic field strength
M	$= 7.8 \cdot 10^{15}$	Tm^3	Magnetic moment of the Earth
R	$= 7078$	km	Distance between the spacecraft and the Earth's center
μ	$= 398600$	km^3/s^2	Earth's gravitational constant
θ	$= 10$	deg	Angle between the local vertical and the Z principal axis
I_z	$= 0.0095$	kgm^2	Moments of inertia about Z
I_y	$= 0.062$	kgm^2	Moments of inertia about Y
m	$= 6$	kg	Spacecraft's mass

Table B.4: *Data used in the computation of the disturbance environment*

Disturbance	Type	Formula	Result
Solar Radiation	Cyclic for Earth-oriented	$T_s = \dfrac{\Phi}{c} A_s (1 + q)(cp_s - cm) \cos \varphi$	$\mathbf{2.76 \cdot 10^{-8}\ Nm}$
Atmospheric Drag	Constant for Earth-oriented	$T_a = \dfrac{1}{2} \rho C_d A_r V^2 (cp_a - cm)$	$\mathbf{1.14 \cdot 10^{-8}\ Nm}$
Magnetic Field	Cyclic	$T_m = DB = D \left(\dfrac{M}{R^3} \lambda \right)$	$\mathbf{3.59 \cdot 10^{-6}\ Nm}$
Gravity Gradient	Constant for Earth-oriented	$T_g = \dfrac{3\mu}{2R^3} \left| I_z - I_y \right| \sin 2\theta$	$\mathbf{3.05 \cdot 10^{-8}\ Nm}$

Table B.5: *Disturbance torques, data involves in the calculation are shown in Table B.4*

Reaction wheels			
Parameters	**Values**	**Units**	**Notes**
Torque for sizing	$5.16 \cdot 10^{-6}$	Nm	-
Required angular momentum	0.0061	Nms	-
Required slew rate	0.1520	deg/s	-
Number of orbits required for saturation	25.87	rev	Assuming: $h_{RW_{max}} = 0.03\ Nms$, $\eta = \varepsilon = 0.9$
Time required for saturation	42.59	h	Same as above

Magnetic torquers			
Parameters	**Values**	**Units**	**Notes**
Required torquer dipole*	0.5	$A\,m^2$	Guarantees detumbling in less than 3 orbits for initial angular velocities less than $10\ deg/s$ in all three axes
Time required for desaturation	271	s	Calculated considering the required angular momentum and the torque for sizing of the reaction wheels

* Defined through detumbling simulations, see Section 5.1

Table B.6: *Momentum budget results for an altitude orbit of 700 km, the data used are shown in Table B.4*
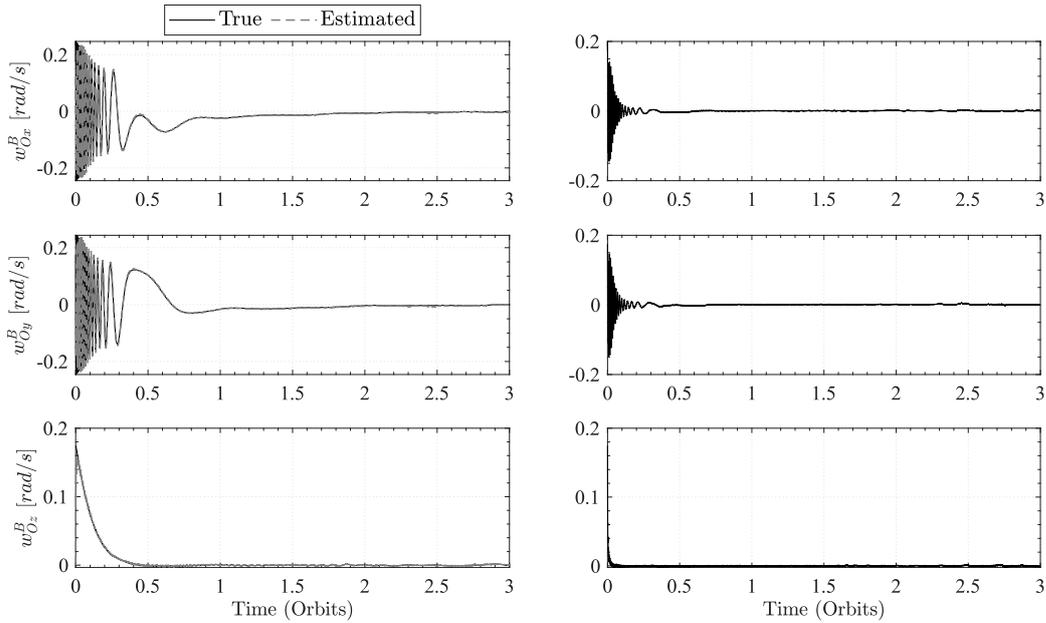

Figure B.1: *Results of the Kalman Rate Estimator in the detumbling simulation with an altitude of 700 km. The diagrams on the right show the error, i.e. the difference between the true and estimated value*
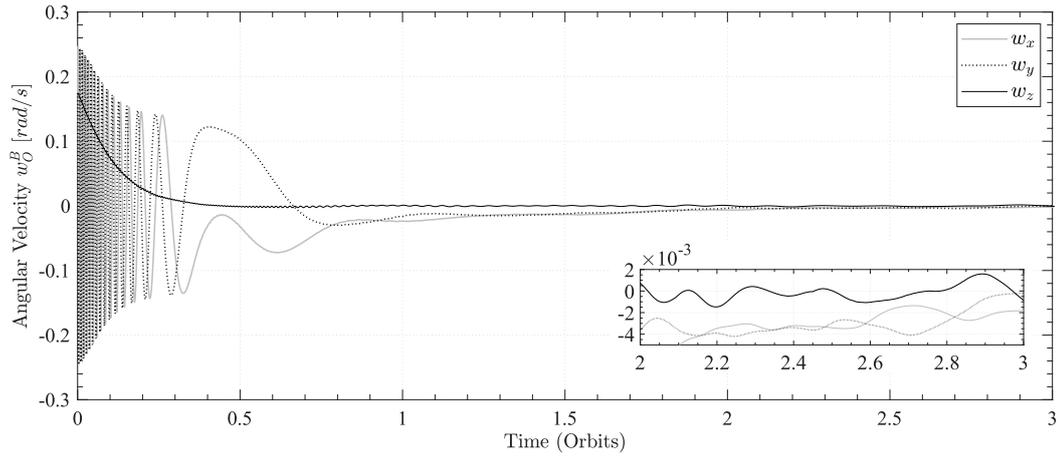
Figure B.2: *Orbit referenced angular velocities of the spacecraft during the detumbling simulation with an altitude of 700 km. A focus of the third orbit is shown in the figure to show the detail of the angular velocity value*
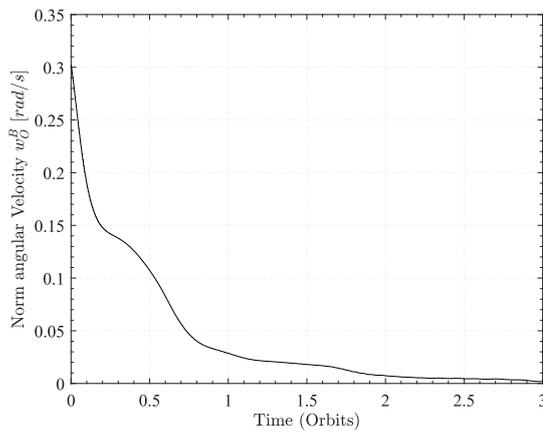
**Figure B.3:** *Norm of angular velocities*
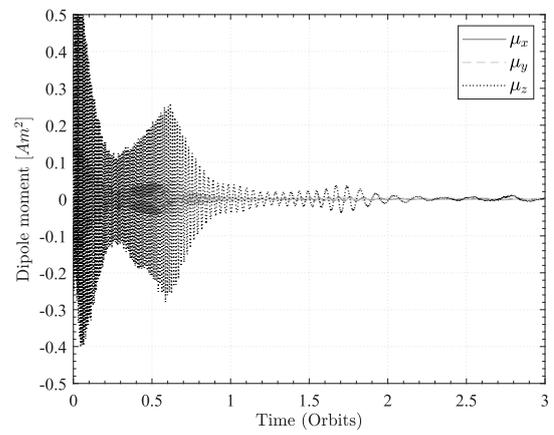
Figure B.4: *Dipole moment*

133
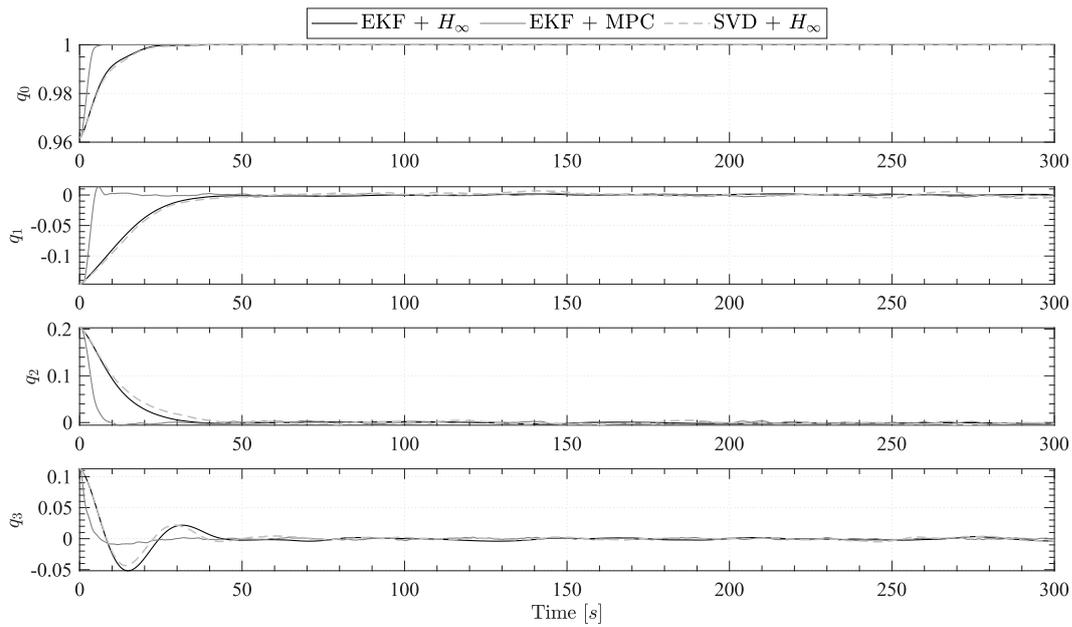
Figure B.5: *Comparison of the quaternions resulting from the nadir pointing simulation. The diagram refers to the case named 'b' in Section 6.3*
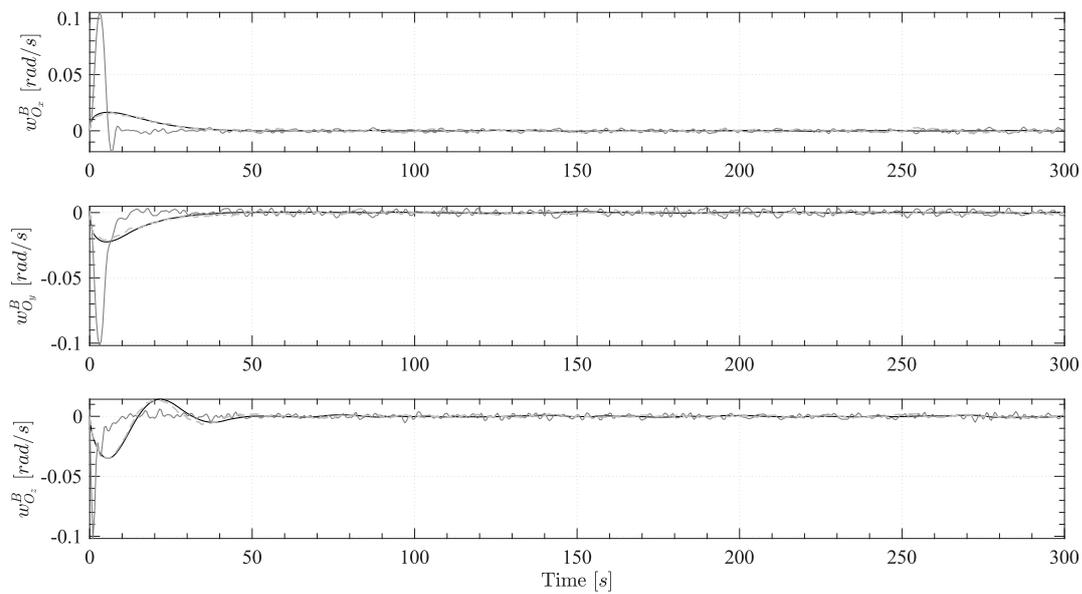
Figure B.6: *Comparison of the spacecraft angular velocities resulting from the nadir pointing simulation. The diagram refers to the case named 'b' in Section 6.3*
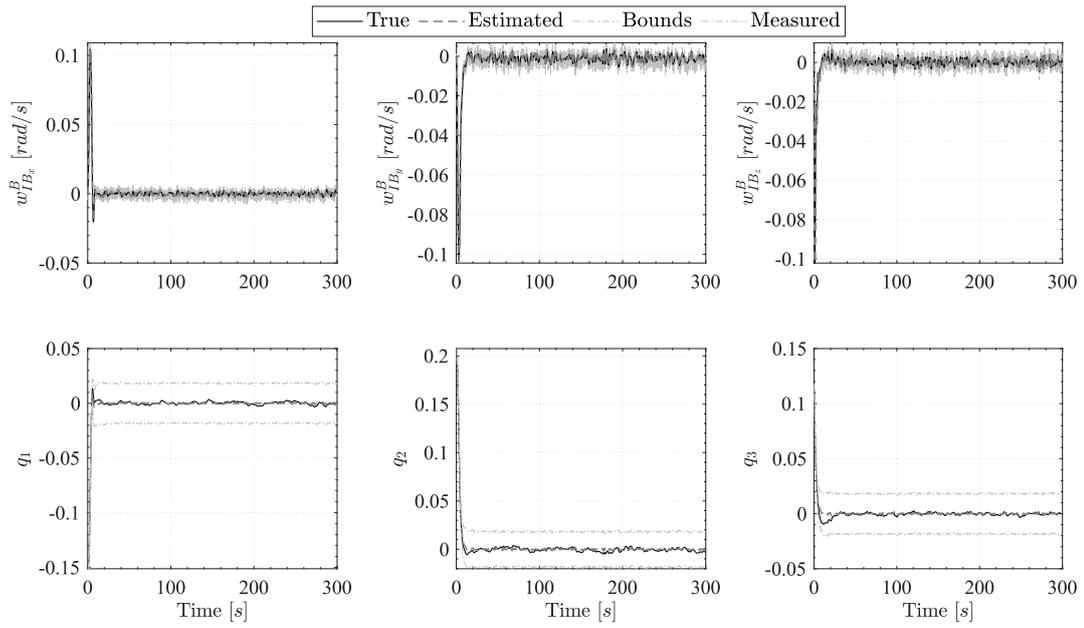
Figure B.7: *EKF+MPC filter results from the nadir pointing simulation. The diagram refers to the case named 'b'*
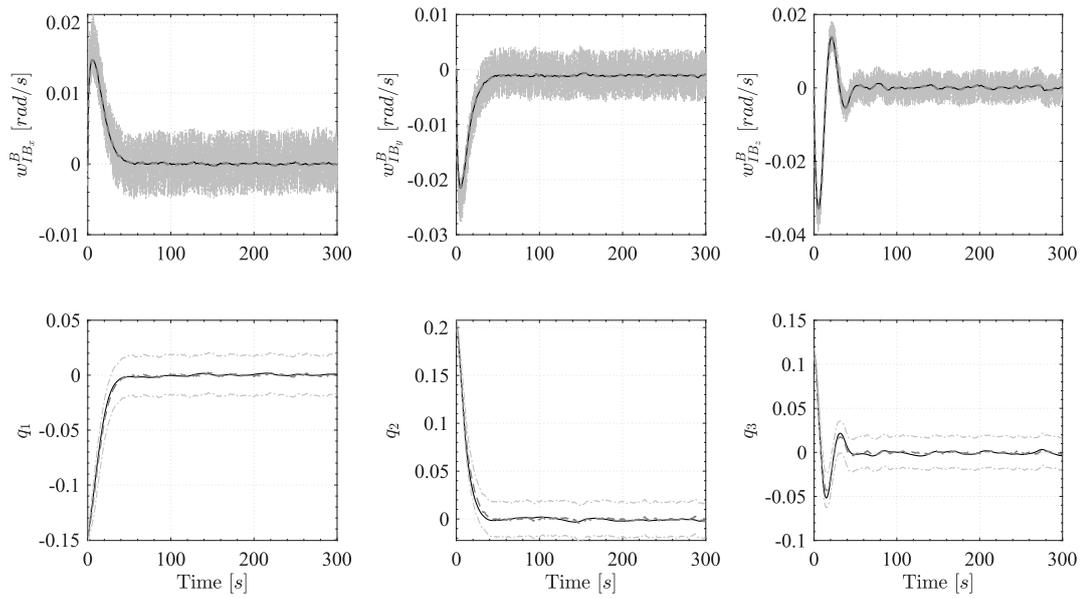
**Figure B.8:** *EKF+H_∞ filter results from the nadir pointing simulation. The diagram refers to the case named 'b'*

Appendix C

Requirements

Functional Requirements			
ID	Requirement	ID	Requirement
FUN-400	The attitude determination and control system (ADCS) shall guarantee the desired attitude	FUN-465	The ADCS PU shall compute control torque values
FUN-401	The ADCS shall determine spacecraft attitude data	FUN-470	The ADCS PU shall update control torques every 0.5 s
FUN-402	The ADCS shall generate profile for attitude	FUN-475	The ADCS PU shall switch on/off ADCS hardware parts
FUN-403	The ADCS shall calculate the control torques	FUN-485	The ADCS PU shall insert/remove ADCS software task
FUN-404	The ADCS shall generate the required torques	FUN-495	The ADCS PU shall acquire health status data every TBD s
FUN-405	The ADCS shall manage its operations	FUN-500	The ADCS PU shall acquire power consumption
FUN-406	The ADCS shall exchange data with other subsystems	FUN-505	The ADCS PU shall compare health status with thresholds
FUN-409	The ADCS shall guarantee an Absolute Performance Error (APE) on angular velocity lower than 0.2 deg/s when in detumbling mode	FUN-510	The ADCS PU shall activate recovery mode when required
FUN-410	The ADCS, when in coarse pointing mode, shall guarantee: An APE lower than 10 deg; An AKE lower than 5 deg	FUN-525	The ADCS PU shall receive data
FUN-415	The ADCS, when in fine pointing mode, shall guarantee: An APE lower than 0.65 deg; An AKE lower than 0.65 deg; A PSE lower than 0.12 deg/s	FUN-530	The ADCS PU shall receive commands
FUN-424	The magnetometer shall measure the electromagnetic field with an accuracy of TBD	FUN-535	The ADCS PU shall transmit information
FUN-427	The gyroscope shall measure the angular rates with an accuracy of TBD rad/s	FUN-540	The ADCS PU shall acquire information
FUN-431	The sun sensor shall measure the inclination of the sun vector with respect to the normal of the photocell	FUN-545	The ADCS PU shall extract data
FUN-435	The ADCS PU shall process the measurements data	FUN-550	The ADCS PU shall process data
FUN-437	The ADCS PU shall gather measurement data	FUN-552	The ADCS PU shall check the commands
FUN-438	An A/D converter shall convert the analog signal coming from the sensors into a digital one directed to the ADCS board MCU	FUN-555	The ADCS PU shall gather data
FUN-439	The ADCS PU shall load models and lookup table	FUN-560	The ADCS PU shall format data
FUN-441	The ADCS shall update navigation parameters every 0.1 s	FUN-565	The ADCS PU shall send data
FUN-445	The ADCS PU shall define the desired attitude according to the operative mode	FUN-570	The ADCS power consumption shall be less than 6.635 W
FUN-450	The ADCS PU shall define the desired angular rate according to the operative mode	FUN-575	The PWM circuit shall regulate currents
FUN-451	The ADCS shall update guidance parameters every 0.1 s during target pointing	FUN-580	The PWM circuit shall regulate the voltage
FUN-455	The ADCS PU shall calculate the error between estimated angular rate and desired angular rate	FUN-585	Every magnetorquer shall generate a minimum dipole moment of 0.5 Am^2
FUN-460	The ADCS PU shall calculate the error between estimated attitude and desired attitude	FUN-589	Every reaction wheel shall have a minimum angular momentum of 0.01 Nms
		FUN-590	Every reaction wheel shall generate a minimum mechanical torque of $6.67 \cdot 10^{-6}$ (TBC) Nm

Table C.1: *ADCS Functional requirements*

Hardware Characteristics

Parameter		Values
Spectral sensitivity	typ.	6.3 nA/lx
Wavelength of max sensitivity	typ.	570 nm
Spectral range of sensitivity	typ.	400...900 nm
Radiant sensitive area	typ.	7.02 mm^2
Short-circuit current	typ.	6.1 μA
Photocurrent	typ.	6.3 μA
Dark current	typ.	0.1 nA
Operating temperature	min	$-40°C$
	max	$100°C$
Half angle	typ.	$60°$
Weight		44.0 mg
Analog interface		

Table D.1: *SFH 2430 silicon photodiode characteristics. Information can be found in [8]*

Characteristics	Min	Typ	Max	Units
Dynamic Range		± 1200		deg/sec
Initial Sensitivity		0.04		$deg/sec/LSB$
Misalignment		±0.05		deg
In-Run Bias Stability		14.5		deg/hr
Angular Random Walk		0.66		$deg/\sqrt{hr}$
Output Noise		0.27		deg/sec rms
Temperature Range	-45		85	$°C$

Features

Triaxial digital gyroscope with digital range scaling

SPI-compatible

Power supply voltage range: 3.15 V to 3.45 V

Table D.2: *Compact, precision six degrees of freedom inertial sensor ADIS16446. Informations can be found in [1]*

Characteristics	Min	Typ	Max	Units
Magnetic Field				
Sensitivity	0.98	1	1.02	$V/gauss$
Resolution		40		$\mu gauss$
Field Range	-2		2	$gauss$
Errors				
Linearity Error		0.5	2	$\%FS$
Hysteresis Error		0.05	0.1	$\%FS$
Environments				
Temperature	-40		+85	$^\circ C$
Electrical				
Supply Voltage	6		15	VDC
Supply Current			20	mA

Features

Analog Output at 1 Volt/gauss (2.5V @ 0 gauss)

Precision 3-axis Capability

20-pin Wide DIP Footprint (1" by 0.75")

Table D.3: *3-Axis Magnetic Sensor Hybrid HMC2003 characteristics. Information can be found in [6]*

Parameter	Values	Units	Notes
Maximum Torque Around X axis	5.9	mNm	-
Maximum Torque Around Y axis	5.9	mNm	-
Maximum Torque Around Z axis	5	mNm	when all 4 RWs are used
Maximum Momentum Storage Around X axis	37	$mNms$	-
Maximum Momentum Storage Around y axis	37	$mNms$	-
Maximum Momentum Storage Around z axis	31.3	$mNms$	when all 4 RWs are used
DC Voltage	5.0	V	-
Power Consumption (Idle)	180	mW	-
Power Consumption (Steady state)	600	mW	1000 RPM each
Mass (excluding harness)	665	g	typ

Features

Operational Temperature: -40 °C to +85 °C

Interfaces: SPI / UART

Dimensions: $92.5 \times 92.5 \times 51.20 \ mm$

Table D.4: *Integral four-reaction wheels redundant 3-axis control system (4RW0). Information can be found in [7]*
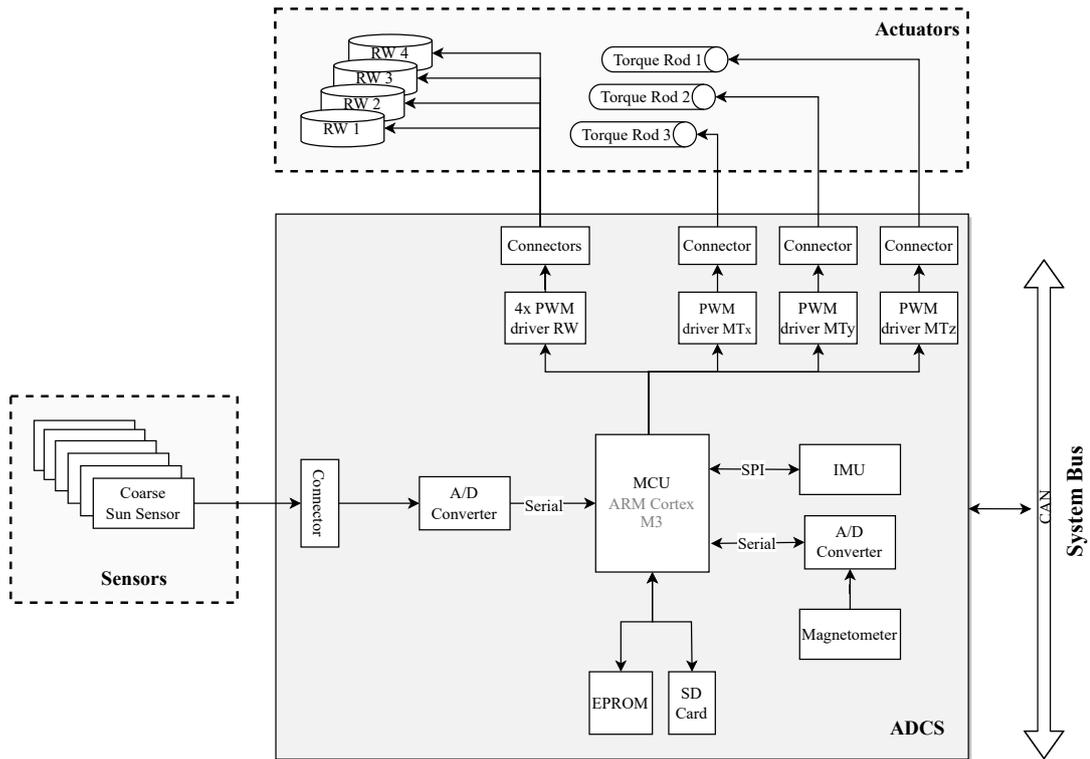
Figure D.1: *PCB schematic for ADCS. Any redundancy components are not shown in the diagram. For information on sensors and actuators, please refer to the datasheets mentioned in the appropriate tables*

Component	Number	Mass	Volume	Power consumption [W]		
	[-]	[g]	[mm x mm x mm]	Max	Nominal	Idle
Reaction wheel	4	665	92.5 x 92.5 x 51.20	13	0.6	0.18
Sun Sensors	4	4	4.5 x 3.7 x 1.2	0	0	0
IMU	1	10	24.53 x 38.08 x 11.10	0.36	0.25	0.24
Magnetometer	1		7.35 x 35 x 3.6	0.3	0.24	0.12
Magnetic Torquers	3	TBD	TBD	TBD	TBD	TBD

Table D.5: *Summary of hardware components characteristics*

Mode	Dormant	Detumbling	Commissioning	Basic	Mission	Transmission
Reaction wheel	Off	Off	Nominal	Idle	Nominal*	Nominal
Sun Sensors	Off	Nominal	Nominal	Idle	Nominal	Nominal
IMU	Off	Off	Nominal	Idle	Max	Nominal
Magnetometer	Off	Max	Nominal	Idle	Max	Nominal
Magnetic Torquers	Off	Max	Off	Off	Off	Off

* Assuming Nadir pointing, if target pointing is active then consider Max

Table D.6: *ADCS components mapped with operative modes*

Kalman Filter Derivations

With reference to the group of equations 4.17, it is possible to conclude immediately that

$$\frac{\partial \bar{F}_1}{\partial \bar{\beta}_k} = 0 \qquad\qquad \frac{\partial \bar{F}_2}{\partial \bar{\beta}_k} = 0 \tag{E.1}$$

$$\frac{\partial \bar{F}_1}{\partial \vec{q}_k} = \mathcal{I} + \frac{1}{2}\frac{\partial}{\partial \vec{q}_k}\left(\mathbf{\Omega}_k\,\bar{\omega}_{IB}^B\right)dt - \frac{1}{2}\frac{\partial}{\partial \vec{q}_k}\left(\mathbf{\Omega}_k \mathbf{A}_O^B \bar{\omega}_{IO}^O\right)dt \tag{E.2}$$

$$\frac{\partial \bar{F}_1}{\partial \bar{\omega}_{IB_k}^B} = \frac{1}{2}\,\mathbf{\Omega}_k\,dt \tag{E.3}$$

where

$$\frac{1}{2}\frac{\partial}{\partial \vec{q}_k}\left(\mathbf{\Omega}_k\,\bar{\omega}_{IB}^B\right)dt = \begin{bmatrix} -\dfrac{q_1\omega_{IB_1}^B}{2q_0(\vec{q})} & \dfrac{\omega_{IB_3}^B}{2} - \dfrac{q_2\omega_{IB_1}^B}{2q_0(\vec{q})} & -\dfrac{\omega_{IB_2}^B}{2} - \dfrac{q_3\omega_{IB_1}^B}{2q_0(\vec{q})} \\[3mm] -\dfrac{\omega_{IB_3}^B}{2} - \dfrac{q_1\omega_{IB_2}^B}{2q_0(\vec{q})} & -\dfrac{q_2\omega_{IB_2}^B}{2q_0(\vec{q})} & \dfrac{\omega_{IB_1}^B}{2} - \dfrac{q_3\omega_{IB_2}^B}{2q_0(\vec{q})} \\[3mm] \dfrac{\omega_{IB_2}^B}{2} - \dfrac{q_1\omega_{IB_3}^B}{2q_0(\vec{q})} & -\dfrac{\omega_{IB_1}^B}{2} - \dfrac{q_2\omega_{IB_3}^B}{2q_0(\vec{q})} & -\dfrac{q_3\omega_{IB_3}^B}{2q_0(\vec{q})} \end{bmatrix}_k dt \tag{E.4}$$

$$\begin{aligned} \mathbf{\Omega}_k \mathbf{A}_O^B \bar{\omega}_{IO}^O &= -\omega_0 \begin{bmatrix} q_0(\vec{q}) & -q_3 & q_2 \\ q_3 & q_0(\vec{q}) & -q_1 \\ -q_2 & q_1 & q_0(\vec{q}) \end{bmatrix}_k \begin{bmatrix} 2(q_1 q_2 + q_0(\vec{q})q_3) \\ q_0(\vec{q})^2 - q_1^2 + q_2^2 - q_3^2 \\ 2(q_2 q_3 - q_0(\vec{q})q_1) \end{bmatrix}_k = \\[2mm] &= -\omega_0 \begin{bmatrix} q_0(\vec{q})^2 + q_1^2 q_3 + q_3^3 + q_2^2 q_3 \\ q_0(\vec{q})q_3^2 + q_0(\vec{q})^3 + q_0(\vec{q})q_2^2 \\ -q_1 q_2^2 - q_1^3 - q_1 q_3^2 \end{bmatrix}_k = \\[2mm] &= -\omega_0 \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \end{bmatrix}_k \end{aligned} \tag{E.5}$$

$$\frac{\partial}{\partial \vec{q}_k}\left(\mathbf{\Omega}_k \mathbf{A}_O^B \bar{\omega}_{IO}^O\right) = \begin{bmatrix} \dfrac{d\varepsilon_1}{dq_1} & \dfrac{d\varepsilon_1}{dq_2} & \dfrac{d\varepsilon_1}{dq_3} \\[3mm] \dfrac{d\varepsilon_2}{dq_1} & \dfrac{d\varepsilon_2}{dq_2} & \dfrac{d\varepsilon_2}{dq_3} \\[3mm] \dfrac{d\varepsilon_3}{dq_1} & \dfrac{d\varepsilon_3}{dq_2} & \dfrac{d\varepsilon_3}{dq_3} \end{bmatrix}_k \tag{E.6}$$

The individual components are given by the following equations obtained by assuming that the scalar component of the quaternion is a function of the three components of the vector part of the quaternion

$$\frac{d\varepsilon_1}{dq_1} = -\omega_0 \left(2q_0(\vec{q})q_3 \frac{dq_0(\vec{q})}{q_1} + 2q_1q_3 \right)_k$$

$$\frac{d\varepsilon_1}{dq_2} = -\omega_0 \left(2q_0(\vec{q})q_3 \frac{dq_0(\vec{q})}{q_2} + 2q_2q_3 \right)_k$$

$$\frac{d\varepsilon_1}{dq_3} = -\omega_0 \left(2q_0(\vec{q})q_3 \frac{dq_0(\vec{q})}{q_3} + q_0(\vec{q})^2 + q_1^2 + 3q_3^2 + q_2^2 \right)_k$$

$$\frac{d\varepsilon_2}{dq_1} = -\omega_0 \left(q_3^2 \frac{dq_0(\vec{q})}{q_1} + 3q_0(\vec{q})^2 \frac{dq_0(\vec{q})}{q_1} + q_2^2 \frac{dq_0(\vec{q})}{q_1} \right)_k$$

$$\frac{d\varepsilon_2}{dq_2} = -\omega_0 \left(q_3^2 \frac{dq_0(\vec{q})}{q_2} + 3q_0(\vec{q})^2 \frac{dq_0(\vec{q})}{q_2} + 2q_0(\vec{q})q_2 + q_2^2 \frac{dq_0(\vec{q})}{q_2} \right)_k \qquad \text{(E.7)}$$

$$\frac{d\varepsilon_2}{dq_3} = -\omega_0 \left(2q_0(\vec{q})q_3 + q_3^2 \frac{dq_0(\vec{q})}{q_3} + 3q_0(\vec{q})^2 \frac{dq_0(\vec{q})}{q_3} + q_2^2 \frac{dq_0(\vec{q})}{q_3} \right)_k$$

$$\frac{d\varepsilon_3}{dq_1} = -\omega_0 \left(-q_2^2 - 3q_1^2 - q_3^2 \right)_k$$

$$\frac{d\varepsilon_3}{dq_2} = -\omega_0 \left(-2q_1q_2 \right)_k$$

$$\frac{d\varepsilon_3}{dq_1} = -\omega_0 \left(-2q_1q_3 \right)_k$$

where

$$\frac{dq_0(\vec{q})}{q_1} = -\frac{q_1}{q_0(\vec{q})}$$

$$\frac{dq_0(\vec{q})}{q_2} = -\frac{q_2}{q_0(\vec{q})} \qquad \text{(E.8)}$$

$$\frac{dq_0(\vec{q})}{q_3} = -\frac{q_3}{q_0(\vec{q})}$$

The derivative of the second component of $\boldsymbol{F}$ with respect to the quaternion results in the derivative of the gravity-gradient torque which is a function of the spacecraft attitude.

$$\frac{\partial \bar{\boldsymbol{F_2}}}{\partial \vec{\boldsymbol{q_k}}} = \boldsymbol{I}^{-1} \frac{\partial \bar{T}_k^B}{\partial \vec{q}_k} \, dt = \boldsymbol{I}^{-1} \frac{\partial \bar{T}_{gg_k}^B}{\partial \vec{q}_k} \qquad \text{(E.9)}$$

The torque is modelled as follows

$$\bar{T}_{gg}^B = \begin{bmatrix} 3\,\omega_0^2\,(I_z - I_y)A_{23}A_{33} \\ 3\,\omega_0^2\,(I_x - I_z)A_{13}A_{33} \\ 3\,\omega_0^2\,(I_y - I_x)A_{13}A_{23} \end{bmatrix} = \begin{bmatrix} k_1\,A_{23}A_{33} \\ k_2\,A_{13}A_{33} \\ k_3\,A_{13}A_{23} \end{bmatrix} \qquad \text{(E.10)}$$

Therefore, the derivatives are as follows

142

$$\frac{\partial \bar{T}_{gg_1}^B}{\partial q_1} = k_1 \left(A_{23} \frac{dA_{33}}{q_1} + A_{33} \frac{dA_{23}}{q_1} \right)$$

$$\frac{\partial \bar{T}_{gg_1}^B}{\partial q_2} = k_1 \left(A_{23} \frac{dA_{33}}{q_2} + A_{33} \frac{dA_{23}}{q_2} \right)$$

$$\frac{\partial \bar{T}_{gg_1}^B}{\partial q_3} = k_1 \left(A_{23} \frac{dA_{33}}{q_3} + A_{33} \frac{dA_{23}}{q_3} \right)$$

$$\frac{\partial \bar{T}_{gg_2}^B}{\partial q_1} = k_2 \left(A_{13} \frac{dA_{33}}{q_1} + A_{33} \frac{dA_{13}}{q_1} \right)$$

$$\frac{\partial \bar{T}_{gg_2}^B}{\partial q_2} = k_2 \left(A_{13} \frac{dA_{33}}{q_2} + A_{33} \frac{dA_{13}}{q_2} \right) \tag{E.11}$$

$$\frac{\partial \bar{T}_{gg_2}^B}{\partial q_3} = k_2 \left(A_{13} \frac{dA_{33}}{q_3} + A_{33} \frac{dA_{13}}{q_3} \right)$$

$$\frac{\partial \bar{T}_{gg_1}^B}{\partial q_1} = k_3 \left(A_{13} \frac{dA_{23}}{q_1} + A_{23} \frac{dA_{13}}{q_1} \right)$$

$$\frac{\partial \bar{T}_{gg_1}^B}{\partial q_2} = k_3 \left(A_{13} \frac{dA_{23}}{q_2} + A_{23} \frac{dA_{13}}{q_2} \right)$$

$$\frac{\partial \bar{T}_{gg_1}^B}{\partial q_3} = k_3 \left(A_{13} \frac{dA_{23}}{q_3} + A_{23} \frac{dA_{13}}{q_3} \right)$$

The derivative of the second component of $\boldsymbol{F}$ with respect to angular velocity is simply given by

$$\frac{\partial \bar{\boldsymbol{F}}_2}{\partial \bar{\boldsymbol{\omega}}_{IB_k}^B} = \boldsymbol{\mathcal{I}} - \frac{\partial}{\partial \bar{\boldsymbol{\omega}}_{IB_k}^B} \left(\boldsymbol{I}^{-1} \bar{\omega}_{IB_k}^B \times \boldsymbol{I} \bar{\omega}_{IB_k}^B \right) \tag{E.12}$$

Bibliography

[1] Analog devices, inc. - adis16446 isensor device. "https://www.analog.com/media/en/technical-documentation/data-sheets/adis16446.pdf".

[2] Earth observation directory amp; news. "https://directory.eoportal.org/web/eoportal/home".

[3] Ecss-e-hb-60-10a – control performance guidelines (14 december 2010).

[4] Ecss-e-st-60-10c – control performance (15 november 2008).

[5] Ecss-e-st-60-30c: Satellite attitude and orbit control system (aocs) requirements (30 august 2013).

[6] Honeywell - 3-axis magnetic sensor hybrid hmc2003. "https://neurophysics.ucsd.edu/Manuals/Honeywell/HMC_2003.pdf".

[7] Nanoavionics, corp. 2022, technical overview na-4rw0-to-r000. "https://nanoavionics.com/cubesat-components/".

[8] Osram opto semiconductors - sfh 2430. "https://www.osram.com/ecat/com/en/class_pim_web_catalog_103489/prd_pim_device_2219613/".

[9] Cubesat design specification rev. 13. Technical report, California Polytechnic State University, San Luis Obispo, 2014.

[10] *CubeSat 101: Basic Concepts and Processes for First-Time CubeSat Developers*. 2017.

[11] Patrick Alken, Erwan Thébault, Ciarán D Beggan, Hagay Amit, J Aubert, J Baerenzung, TN Bondar, WJ Brown, S Califf, A Chambodut, et al. International geomagnetic reference field: the thirteenth generation. *Earth, Planets and Space*, 73(1):1–25, 2021.

[12] Giulio Avanzini. Spacecraft attitude dynamics and control. *Politecnico di Torino*, 2009, 2008.

[13] Giulio Avanzini and Fabrizio Giulietti. Magnetic detumbling of a rigid spacecraft. *Journal of guidance, control, and dynamics*, 35(4):1326–1334, 2012.

[14] Roger R Bate, Donald D Mueller, Jerry E White, and William W Saylor. *Fundamentals of astrodynamics*. Courier Dover Publications, 2020.

[15] Richard H Battin. *An introduction to the mathematics and methods of astrodynamics*. Aiaa, 1999.

[16] Sanjit P Bhat and Ajit S Dham. Controllability of spacecraft attitude under magnetic actuation. In *42nd IEEE International Conference on Decision and Control (IEEE Cat. No. 03CH37475)*, volume 3, pages 2383–2388. IEEE, 2003.

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

[18] Peter J Camillo and FL Markley. Orbit-averaged behavior of magnetic control laws for momentum unloading. *Journal of Guidance and Control*, 3(6):563–568, 1980.

[19] Chengyu Cao, Lili Ma, and Yunjun Xu. Adaptive control theory and applications, 2012.

[20] Ryan James Caverly and James Richard Forbes. Lmi properties and applications in systems, stability, and control theory. *arXiv preprint arXiv:1903.08599*, 2019.

[21] Xiaojiang Chen, Willem Steyn, and Yoshi Hashida. Ground-target tracking control of earth-pointing satellites. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, page 4547, 2000.

[22] John L Crassidis, F Landis Markley, and Yang Cheng. Survey of nonlinear attitude estimation methods. *Journal of guidance, control, and dynamics*, 30(1):12–28, 2007.

[23] Howard Curtis. *Orbital mechanics for engineering students*. Butterworth-Heinemann, 2013.

[24] Geir E Dullerud and Fernando Paganini. *A course in robust control theory: a convex approach*, volume 36. Springer Science & Business Media, 2013.

[25] JL Farrell, JC Stuelpnagel, RH Wessner, JR Velman, and JE Brook. A least squares estimate of satellite attitude (grace wahba). *Siam Review*, 8(3):384–386, 1966.

[26] Daniel J Fonte Jr, Beny Neta, Chris Sabol, Donald A Danielson, and Walter R Dyar. Comparison of orbit propagators in the research and development goddard trajectory determination system (r & d gtds). part 1. simulated data. Technical report, NAVAL POSTGRADUATE SCHOOL MONTEREY CA DEPT OF MATHEMATICS, 1995.

[27] Stephen J Kapurch. *NASA systems engineering handbook*. Diane Publishing, 2010.

[28] Ern J Lefferts, F Landis Markley, and Malcolm D Shuster. Kalman filtering for spacecraft attitude estimation. *Journal of Guidance, Control, and Dynamics*, 5(5):417–429, 1982.

[29] J. Löfberg. Yalmip : A toolbox for modeling and optimization in matlab. In *In Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.

[30] Jan Marian Maciejowski. *Predictive control: with constraints*. Pearson education, 2002.

[31] F Landis Markley and John L Crassidis. *Fundamentals of spacecraft attitude determination and control*, volume 1286. Springer, 2014.

[32] MA Si Mohammed, H Boussadia, A Bellar, and A Adnane. Performance comparison of attitude determination, attitude estimation, and nonlinear observers algorithms. In *Journal of Physics: Conference Series*, volume 783, page 012017. IOP Publishing, 2017.

[33] United States Naval Observatory. Nautical Almanac Office, Great Britain. Nautical Almanac Office, Engineering Research Council (Great Britain), Rutherford Appleton Laboratory, and Council for the Central Laboratory of the Research Councils (Great Britain). *The Astronomical Almanac*. US Government Printing Office, 2008.

[34] Matthew M Peet. Introduction to optimal control via lmis, 2016.

[35] P Kenneth Seidelmann. *Explanatory supplement to the astronomical almanac*. University Science Books, 2006.

[36] Norman Sissenwine, Maurice Dubin, and Harry Wexler. The us standard atmosphere, 1962. *Journal of Geophysical Research*, 67(9):3627–3630, 1962.

[37] Fabrizio Stesina. *Design and verification of Guidance, Navigation and Control systems for space applications*. PhD thesis, Politecnico di Torino, 2014.

[38] Willem Hermanus Steyn. *A multi-mode attitude determination and control system for small satellites*. PhD thesis, Stellenbosch: Stellenbosch University, 1995.

[39] Armen Toorian, Ken Diaz, and Simon Lee. The cubesat approach to space access. In *2008 IEEE Aerospace Conference*, pages 1–14, 2008.

[40] David A Vallado. *Fundamentals of astrodynamics and applications*, volume 12. Springer Science & Business Media, 2001.

[41] Nicole Viola, Sabrina Corpino, Marco Fioriti, and Fabrizio Stesina. Functional analysis in systems engineering: Methodology and applications. In Boris Cogan, editor, *Systems Engineering*, chapter 3. IntechOpen, Rijeka, 2012.

[42] Grace Wahba. A least squares estimate of satellite attitude. *SIAM review*, 7(3):409–409, 1965.

[43] Liuping Wang. *Model predictive control system design and implementation using MAT-LAB®*. Springer Science & Business Media, 2009.

[44] James R Wertz. *Spacecraft attitude determination and control*, volume 73. Springer Science & Business Media, 2012.

[45] James R. Wertz, David F. Everett, and Jeffery J. Puschell. Space mission engineering : the new smad. 2011.

[46] Xiwang Xia, Guowen Sun, Keke Zhang, Shufan Wu, Tian Wang, Lei Xia, and Shanwu Liu. Nanosats/cubesats adcs survey. In *2017 29th Chinese Control And Decision Conference (CCDC)*, pages 5151–5158. IEEE, 2017.

[47] Yong-chun Xie, Huang Huang, Yong Hu, and Guo-qi Zhang. Applications of advanced control methods in spacecrafts: progress, challenges, and future prospects. *Frontiers of Information Technology & Electronic Engineering*, 17(9):841–861, 2016.

[48] Yaguang Yang. Quaternion based model for momentum biased nadir pointing spacecraft. *Aerospace Science and Technology*, 14(3):199–202, 2010.

[49] Yaguang Yang. *Spacecraft Modeling, Attitude Determination, and Control Quaternion-based Approach: Quaternion-Based Approach*. CRC Press, 2019.

[50] Yaguang Yang and Zhiqiang Zhou. Spacecraft dynamics should be considered in kalman filter attitude estimation. In *AAS/AIAA Space Flight Mechanics Meeting*, number NF1676L-22885, 2016.