UNIVERSITÀ DEGLI STUDI DI PADOVA

Dipartimento di Ingegneria Industriale DII
Departamento de Engenharia Mecânica DEM (IST)

Corso di Laurea Magistrale in Ingegneria Aerospaziale

Simulation of a Commercial Off-The-Shelf ADCS with Design of a Pitch Sun Tracking Attitude Mode for the ORCASat

Relatori: Prof. Afzal Suleman
Ing. Francesco Branz

Zeno Pavanello
n° 1179767

Anno Accademico 2019/2020

A babbo

Acknowledgments

I give my most sincere thanks to the supervisor from Lisbon, Prof. Suleman for the great opportunity he gave me to work on an important project such as ORCASat and for his constant support during my time in Canada, and to the supervisor from Padova, Ing. Branz, who was always very helpful both with technical matters and, even more importantly, with bureaucratic ones. Also I could never have made it through the bureaucratic jungle without the help of Luisa, whose lighthouse of kindness and availability always shines for us T.I.M.E. students.

This thesis was developed in one of the strangest and more challenging periods of my life and I probably could not have gone through to see it finished if I had not had the support of the few people I consider important in my life.

So I want Lorenzo and Matteo to know that some of my days in Canada I would get up from bed just to hear their voices over my headphones while killing some aliens. I know you will always be there for me, not only to fight boredom.

I want to remind my friend Possana (or was it Possma? no, sorry *Possanza*) how great it was to roam with our minds the beloved north-American landscapes, even if I still have a question for him that will probably never be answered: *What's this about?*

A nice *obrigado* has to go to Inês for translating my abstract into a *resumo*.

Also I have to put Tasso e Test in here otherwise they might be offended. Sì... Giappone. And just not to exclude anyone I wanna give credits to everyone who has been close to me for an instant or for years (Mitja, Marco, Fede, Luca, Henrique, Adam, João Paulo, Francesca, Kai, Claudia, Belluz, Albi and many more).

Lastly I thank you mamma, without your presence I would probably still be wandering aimlessly. Grazie.

Resumo

Esta Tese de Mestrado tem o intuito de demonstrar que o CubeSpace Y-Momentum ADCS cumpre os requisitos da missão ORCASat: efetuar o detumbling do satélite, manter o erro de estimação de atitude inferior a 2° e o erro de atitude inferior a 10° durante o modo de Nadir Pointing (NP). Por forma a ter uma alternativa ao modo NP durante órbitas em que a potência fornecida pelos painéis solares é mínima, desenvolveu-se também o modo de atitude Pitch Sun Tracking (PST). Através de simulações detalhadas, demonstrou-se que, ao adotar o modo de atitude PST, os painéis solares são capazes de fornecer uma potência extra que é significativa sem que os atuadores ADCS exijam um aumento de potência considerável.

O conjunto de sensores Y-Momentum ADCS contém dez sensores solares simples, um sensor solar de elevada precisão, um magnetómetro e um giroscópio MEMS. As medições feitas com este conjunto de sensores são utilizadas por cinco modos de determinação distintos, implementados no computador do ADCS. As simulações permitem verificar que o Multiplicative Extended Kalman Filter (MEKF) e o Additive (AEKF) garantem uma estimação aceitável do estado do satélite no que diz respeito à atitude e velocidade angular.

O hardware de ADCS inclui três magnetómetros para controlo ativo da atitude e uma Momentum wheel para estabilização de rotação do eixo de inclinação. Quatro modos de detumbling foram simulados e comparados entre si em termos de performance e requisitos de potência. Estes resultam adequados ao detumbling do satélite, podendo diminuir a velocidade angular do satélite num período de tempo aceitável.

Finalmente, é mostrado através de simulações detalhadas que durante o modo NP o ADCS, utlizando um Regulador Quadrático Linear de Ganho Constante (CGLQR), é capaz de manter o erro de atitude dentro do requisito, enquanto que durante o modo PST pode manter a potência disponível nos painéis solares quase constantemente por volta do valor máximo obtido no modo NP. Quantitativamente, isso significa que a disponibilidade de energia ao longo uma órbita aumenta em média em mais de 1 W (17%), sem que seja necessário um aumento significativo de potência pedida aos atuadores ($\simeq 60$ mW). Além disso, o controlador CGLQR prova ser capaz de alternar facilmente entre os dois modos de atitude, com um tempo de transição inferior a cinco minutos e sem exigir uma mudança na matriz de ganho.

Palavras-chave: ADCS, Pitch Sun Tracking, CubeSat, Extended Kalman Filter, Disponibilidade de Poténcia

Sommario

Questa tesi magistrale nasce per dimostrare la capacità dell'ADCS Y-Momentum di Cubespace di soddisfare i requisiti di missione di ORCASat: detombolare il satellite, mantenere l'errore di stima dell'assetto entro 2° e l'errore d'assetto sotto 10° durante il modo d'assetto Puntamento a Nadir (NP). Inoltre, per avere un'alternativa al modo NP durante le orbite in cui i pannelli solari forniscono il minimo di energia, è sviluppato e testato con simulazioni numeriche il modo d'assetto di Puntamento al Sole sul piano di Beccheggio (PST). É dimostrato che adottando questo nuovo modo d'assetto i pannelli solari sono in grado idealmente di fornire un surplus di potenza di 0.7 W (24%) nelle orbite in cui la disponibilità di potenza è minima.

Il set di sensori del Y-Momentum ADCS presenta dieci sensori di Sole grossolani, un sensore di Sole preciso, un magnetometro e tre giroscopi MEMS. Le misure di questi sensori vengono utilizzate da cinque diversi modi di determinazione implementati nel computer dell'ADCS. Le simulazioni mostrano che l'Extended Kalman Filter Moltiplicativo (MEKF) e l'Additivo (AEKF) sono in grado di garantire una stima accettabile dello stato del satellite in termini di puntamento e velocità angolare.

L'hardware dell'ADCS comprende tre magnetorquers per il controllo attivo dell'assetto e una ruota di momento per la stabilizzazione del moto di pitch. Quattro modi di detombolamento sono stati simulati e comparati in termini di performance e requisiti di potenza. Risultano essere adatti a detombolare il satellite, dato che sono in grado di ridurre la velocità angolare in un lasso di tempo accettabile.

Infine è dimostrato che durante il modo NP, usando un Linear Quadratic Regulator a Guadagni Costanti (CGLQR), l'ADCS è in grado di mantenere l'errore d'assetto entro il requisito, mentre durante il modo PST è capace di mantenere la potenza disponibile ai pannelli solari attorno al valore massimo raggiunto dal modo NP. Dal punto di vista quantitativo, questo significa che durante un'orbita la potenza disponibile aumenta in media di più di 1 W (17%), senza comportare un aumento significativo di potenza richiesta dagli attuatori ($\simeq 60$ mW). Inoltre, il controllore CGLQR si dimostra capace di passare facilmente da un modo d'assetto all'altro in un tempo inferiore a cinque minuti e senza richiedere modifiche alla matrice dei guadagni.

Parole chiave: ADCS, Pitch Sun Tracking, CubeSat, Extended Kalman Filter, Disponibilità di Potenza

Abstract

This Master's thesis was created to demonstrate the ability of CubeSpace Y-Momentum ADCS to meet the mission requirements of ORCASat: detumbling the satellite, keeping the attitude estimation error beneath 2° and the attitude error within 10° during the Nadir Pointing (NP) attitude mode. Furthermore, in order to have an alternative to the NP mode during the orbits in which the power provided by the solar panels is minimum, the Pitch Sun Tracking (PST) attitude mode is developed and numerically tested. It is demonstrated that, by adopting this new attitude mode, the solar panels are able to ideally provide a mean surplus of power of 0.7 W (24%) in those orbits where the power availability is minimum.

The Y-Momentum ADCS sensor set features ten coarse Sun sensors, an accurate Sun sensor, a magnetometer and three MEMS gyroscopes. Measurements from these sensors are used by five different determination modes implemented in the ADCS computer. The simulations show that the Multiplicative Extended Kalman Filter (MEKF) and the Additive (AEKF) are able to guarantee an acceptable estimate of the state of the satellite in terms of attitude and angular velocity.

The ADCS hardware includes three magnetorquers for active attitude control and a moment wheel for pitch rotation stabilization. Four detumbling modes were simulated and compared in terms of performance and power requirements. They result suitable to detumble the spacecraft, being able to decrease the angular rate of the satellite in an acceptable time frame.

Finally it is shown via detailed simulations that during the NP mode the ADCS, employing a Constant Gain Linear Quadratic Regulator (CGLQR), is able to keep the attitude error within the requirement, while during the PST mode it can maintain the power available to the solar panels almost constantly on the highest value enabled by the NP mode. Quantitatively, this means that the power availability over one orbit increases on average by more than 1 W (17%), implying no significantly increased demand in the power required by the actuators ($\simeq 60$ mW). Moreover, the CGLQR controller proves to be capable of easily switching between the two attitude modes, with a transition time of less than five minutes, even without requiring a change in the gain matrix.

Keywords: ADCS, Pitch Sun Tracking, CubeSat, Extended Kalman Filter, Power Availability

Contents

List of Tables

List of Figures

Nomenclature

Greek symbols

β	β angle.
μ	Sun gravitational constant.
Ω	RAAN.
ω	Argument of perigee.
ρ	Density.
$\boldsymbol{\eta}$	Gyro Bias White Gaussian Noise vector.
$\boldsymbol{\nu}$	White Gaussian Noise vector.
$\boldsymbol{\omega}$	Angular rate.
$\boldsymbol{\tau}$	Torque vector.

Roman symbols

A	Area.
B	Ballistic coefficient.
i	Orbit inclination.
k	Gain.
m	Mass.
r	Distance.
T	Period.
t	Time.
v	Velocity.
$\mathbf{A}$	Attitude Matrix.
$\mathbf{I}$	Identity matrix.
$\mathbf{J}$	Tensor of the moment of inertia.
$\mathbf{a}$	Acceleration.
$\mathbf{b}$	Magnetic field vector.
$\mathbf{f}$	Force.
$\mathbf{h}$	Angular momentum.
$\mathbf{m}$	Magnetic dipole.
$\mathbf{q}$	Quaternion
$\mathbf{s}$	Poynting vector.

Subscripts

0	Initial state.
$1, 2, 3$	Cartesian components.
3	3×1 column vector.
3×3	3×3 matrix.
B	Body frame.
c	Coriolis.
d	Drag.
e	Eccentricity/counter-electromotive.
est	Estimated.
ext	External.
I	ECI frame.
m	Magnetic.
max	Maximum.
min	Minimum.
O	Orbital frame.
orb	Orbtial.
p	Pressure.
rel	Relative.
sp	Solar pressure.
$tumb$	Tumbling.
w	Momentum wheel.

Superscripts

$-$	Mean value
true	True state.
$\sim$	Measured state.
$\wedge$	Estimation/Vector of unit norm.
*	Conjugate.
$+$	After measurement update.
-	Before measurement update.
T	Transpose.

Acronyms

ACP Attitude Control Program.

ADCS Attitude Determination and Control Subsystem.

AEKF Additive Extended Kalman Filter.

ALTAIR Airborne Laser for Telescopic Atmospheric Interference Reduction.

APEX Asteroid Prospection EXplorer.

ARW Angular Random Walk.

BOL Beginning Of Life.

CAN Controller Area Network.

CfAR Center for Aerospace Research.

CGLQR Constant Gain Linear Quadratic Regulator.

CHIME Canadian Hydrogen Intensity Mapping Experiment.

COTS Commercial Off-the-Shelf.

CPU Central Processing Unit.

DCM Direction Cosine Matrix.

ECI Earth Centered Inertial.

EKF Extended Kalman Filter.

ESA European Space Agency.

FOV Field Of View.

HIL Hardware-In-the-Loop.

I2C Inter-Integrated Circuit.

IAE Integral Absolute Error.

LEO Low Earth Orbit.

LQR Linear Quadratic Regulator.

LTI Linear and Time Invariant.

LVLH Local Vertical Local Horizon.

MEKF Multiplicative Extended Kalman Filter.

MEMS Micro-Electro-Mechanical System.

MISO Multiple Input Single Output.

MREKF Magnetic Rate Extended Kalman Filter.

NP Nadir Pointing.

NRLMSISE-00 US Naval Research Laboratory Mass Spectrometer & Incoherent Scatter Radar-2000.

OBC On Board Computer.

ORCASat Optical and Radio CAlibration Satellite.

PCB Printed Circuit Board.

PST Pitch Sun Tracking.

RAAN Right Ascension of the Ascending Node.

RRW Rate Random Walk.

TRIAD TRIaxial Attitude Determination.

UART Universal Asynchronous Receiver-Transmitter.

UHF Ultra High Frequency.

UVic University of Victoria.

WMM World Magnetic Model.

Chapter 1

Introduction

In this chapter a brief introduction is given on the context in which the research work for the Master's thesis took place: an overview is given on the general concept of CubeSat and on the particular project of ORCASat, which is being developed at the Center for Aerospace Research (CfAR) of the University of Victoria (UVic). Secondly, the motivation and the objectives of the thesis are reported and a preliminary summary of the whole document is given.

1.1 Background

1.1.1 An overview on CubeSats

In the end of the year 1999, a team from the University of Stanford, lead by Prof. Robert Twiggs and Prof. Jordi Puig-Suari, conceived the idea of a new class of small satellites: CubeSat [1]. CubeSat was originally defined as a cube of $10 \times 10 \times 10$ cm^3 with each face covered by solar cells, and containing two rechargeable batteries that split the functions of the spacecraft in two: one half of the satellite contained the On Board Computer (OBC) and all the subsystems necessary to its functioning, leaving the other half available for the payload. The CubeSat standard, as intended by most of the scientific literature, though, is not the spacecraft, but the container [2]. Single units (1U) can be assembled into CubeSats of different sizes, namely 1.5U, 2U, 3U, 6U, 8U, 12U, 16U and 27U, as it is shown in Fig. 1.1 [1].

Originally CubeSats were perceived merely as educational toys for students to get hands-on experience in working on a space project, but thanks to their rapid increase in popularity, which started in 2013, as shown in Fig. 1.2, and the birth of many companies developing standardized Commercial Off-the-Shelf (COTS) components, which allowed to reduce the size of the engineering teams and the costs and times of development, they have progressively started to be used also for scientific [4], remote sensing [5, 6] and technology development [7] projects. The trend reported in Fig. 1.3 suggests that the purely educational missions are progressively disappearing, also because most of the times the educational purpose of the university projects is coupled with a relevant scientific or technological objective. To support this it is sufficient to remember that the total number of scientific contributions related to CubeSats between 2005 and 2017 amounts to a total of 2306 documents [4] and that large CubeSats, like Juventas and

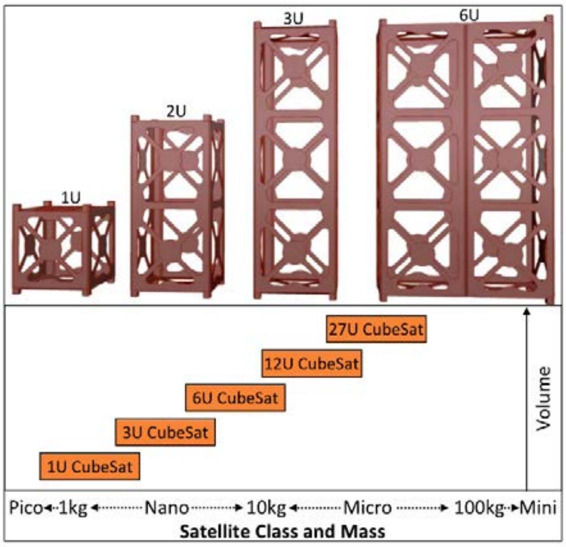


Figure 1.1: Cubesat sizes [3].

APEX on ESA’s Hera mission, are starting to be utilized also for Solar System and deep space exploration [8, 9].

Since like any industry the space sector is mainly driven by economic reasons, to understand the growing popularity of CubeSats it is relevant to notice that, as of May 2018, 57% of the total missions had a commercial end user [4].

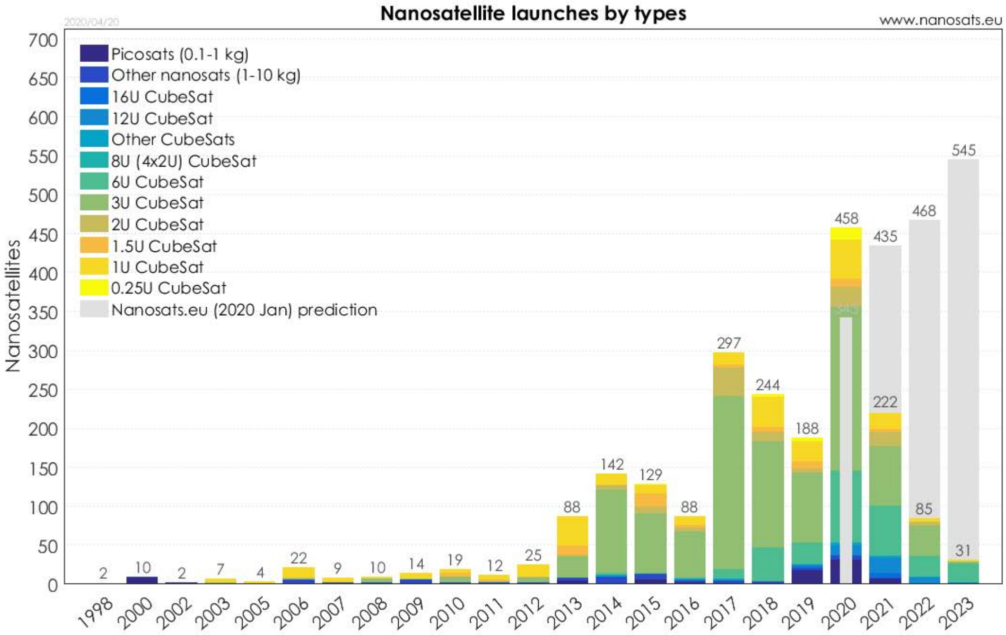


Figure 1.2: Number of CubeSats and picosatellites launched since 1998 and prediction for the future years[1].

This is most significant because, as Twiggs and Puig-Suari effectively wrote in the original paper, this new concept can fulfill the constant request from the space industry for “Smaller, Cheaper, Faster, Better”

[1] `https://alen.space/basic-guide-nanosatellites`

missions [1]. In fact, CubeSats can grant frequent launches and low development and production costs [10], even if this comes at the expense of a short life expectancy. CubeSats typically experience a life-cycle that does not exceed the two years because, since most of them operate in Low Earth Orbit (LEO), they undergo a fast orbital decay. Even when they are provided with thrusters for orbital maintenance, they cannot perform operations of station keeping for long, because of the inevitable low fuel capacity that comes with their small size [11].

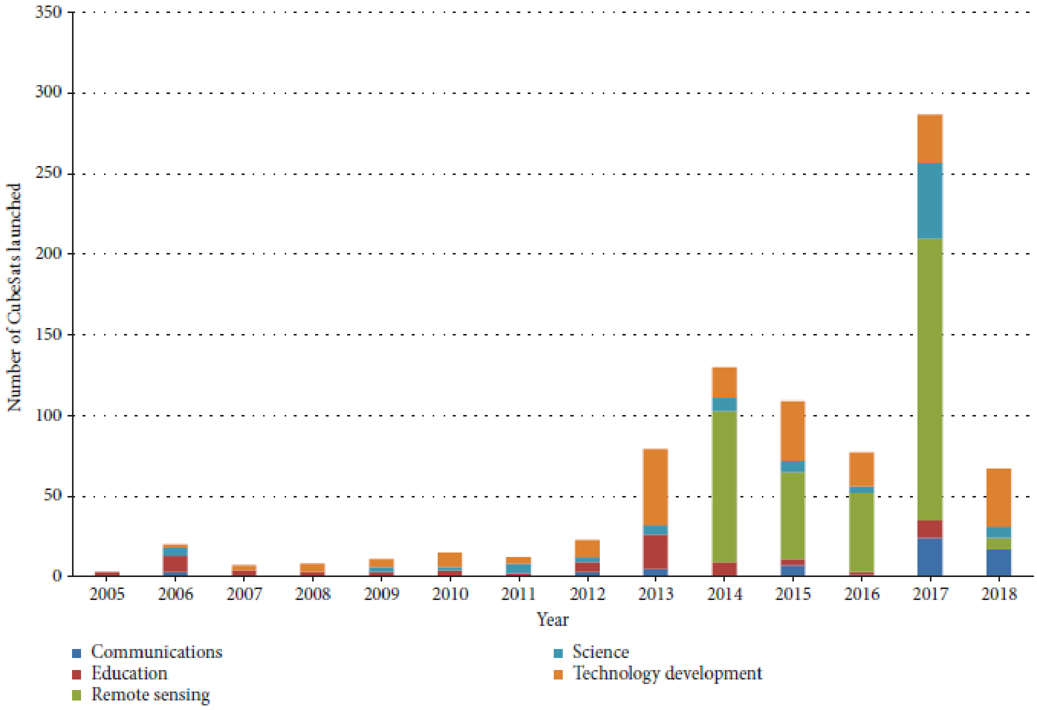


Figure 1.3: CubeSats launched per year and per application from May 2005 to May 2018 [4].

1.1.2 The ORCASat project

In April 2017 the Canadian Space Agency announced the Canadian CubeSat Project, with the purpose of developing highly qualified personnel in Canada and strengthen the future of the Canadian space industry. It gave the opportunity to Universities all over the country to obtain the funds to develop, design and build a nanosatellite. Fifteen proposals were chosen, among which was Optical and Radio CAlibration Satellite (ORCASat), the one from UVic, a render of which is shown in Fig. 1.4. The effort in the project's development is improved thanks to the collaboration with other institutions, namely the University of British Columbia (UBC), Simon Fraser University and, to a lower extent, the University of Lisbon, Harvard University, the National Research Council of Canada and Space System Loral[2].

Like most university projects, ORCASat serves both an educational and a scientific purpose. Designing and building a satellite is an exceptional challenge that can give very solid hands-on learning experience to graduate and undergraduate students, because it is a difficult process of integration of very diverse subsystems, which are usually fully designed and built by the students. The scientific mission objective is

[2] https://www.asc-csa.gc.ca/eng/satellites/cubesat/what-is-the-canadian-cubesat-project.asp

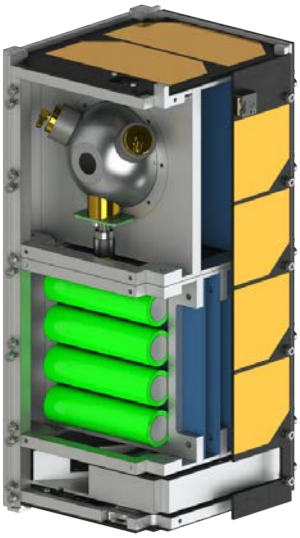


Figure 1.4: ORCASat render [12].

to demonstrate a novel methodology of calibrating optical telescopes by providing an in-situ-calibrated light source in LEO. Two already existing projects would benefit from the completion of this objective: ALTAIR and CHIME. They would acquire a platform to place calibration equipment in LEO, which will improve the measurements of their corresponding Earth based telescopes [12].

ALTAIR

The Airborne Laser for Telescopic Atmospheric Interference Reduction (ALTAIR) Project provides a precision photometric reference calibration above the atmosphere, in the optical and microwave spectra [13, 14].

The ALTAIR payload of ORCASat consists of a laser light source, an integrating sphere and a photodetector. It will allow to quantify the entity of the atmospheric attenuation by comparing the measurement of the emitted radiation done in orbit and the measurement performed by the ground observatory. This can be used to calibrate the telescope when it is observing a distant object, like a star, for which the absolute emission is unknown.

CHIME

The Canadian Hydrogen Intensity Mapping Experiment (CHIME) is a radio telescope designed to map the presence of hydrogen over a fraction of the observable universe. It was designed as an array of cylinder antennas with no moving parts and with a very large Field Of View (FOV) and a broad frequency range $(400-800$ MHz) [15].

The CHIME payload mounted on ORCASat is a small scale Ultra High Frequency (UHF) antenna operating on the frequency of observation of the telescope, which will help in the calibration of the telescope by transmitting up-chirp pulses[3].

[3] https://chime-experiment.ca

1.1.3 Thesis motivation and objectives

The Attitude Determination and Control Subsystem (ADCS) is a fundamental part of any satellite, because it assures that the spacecraft has a knowledge of its orientation in space with respect to known references and it allows the spacecraft to perform attitude maneuvers, i.e. rotations around its center of mass. Without it any satellite would just be like Sputnik 1 and the ground team would just have to hope for the communication antennas and the payloads to be pointed in the right direction in any instant.

Like in many CubeSat missions, the standard operational pointing mode of ORCASat is Nadir Pointing (NP): the satellite should always point the face where the payloads are mounted towards the center of the Earth. Initially the intention of the ORCASat team was to fully develop and build in house an ADCS that was designed to detumble the spacecraft and maintain it in Nadir Pointing mode. Later on in the development of the project, the decision was taken to purchase a COTS ADCS from CubeSpace: CubeADCS Y-Momentum. With respect to the ADCS that was originally designed in Lobo-Fernandes' Master's thesis [16] and later improved in Sabino's [17], the one from CubeSpace is a more reliable and easily implementable system. Hence it was decided to consider another possible attitude mode that would help improve significantly the power availability during the portions of the mission with high eclipse periods: the Pitch Sun Tracking (PST).

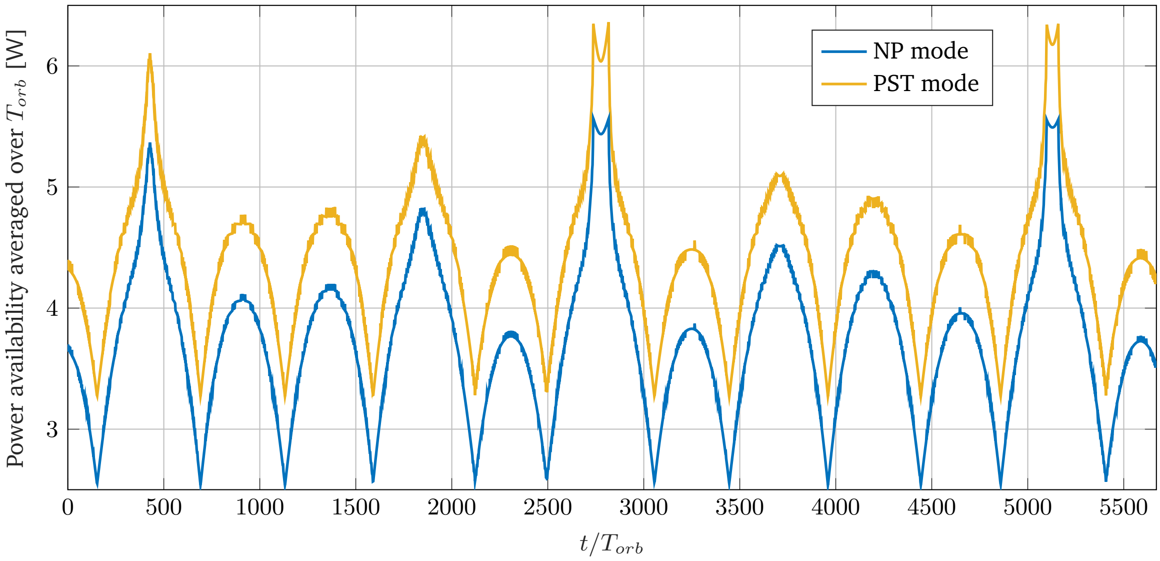


Figure 1.5: Power production comparison between the two considered attitude modes.

A big problem in the definition of the power budget of ORCASat is that while the power availability period is directly proportional to the eclipse period, the opposite is true for the power requirement. In fact, the longer the eclipse period is, the longer the battery has to provide energy that cannot be supplied by the solar panels. So the most critical periods of the mission are the orbits in which the eclipse period is longer. These represent the minima in the power production, which all of the power requirements have to be related to. This approach disregards the fact that in the vast majority of the life-cycle of the mission the margin between power requirement and power availability is much higher than the one the system is designed for. This is why the idea of the PST mode was taken into consideration. Fig. 1.5 is a first rough comparison between the power available in the two attitude modes that is useful in qualitative terms, but it is not accurate because it represents a very simplified situation. Nonetheless it shows that using a PST pointing algorithm it would be ideally possible to increase the absolute minimum in the power availability of around 0.7 W.

Consequently the objectives of this Master's thesis can be summarized as follows:

1. Perform a qualitative estimation of the benefits that the PST attitude mode could bring to the mission.
2. Develop on a high level a PST pointing algorithm.
3. Simulate the detumbling capability of CubeADCS Y-Momentum.
4. Simulate the behaviour of ORCASat under the nominal Nadir Pointing attitude mode.
5. Simulate the behaviour of ORCASat under the Pitch Sun Tracking attitude mode.
6. Compare the two pointing modes in terms of power budget and pointing accuracy.

1.2 Thesis Outline

The thesis work is divided into seven Chapters, excluding the Introduction:

- **Chapter 2** collects all the necessary background knowledge in terms of attitude representations, spacecraft kinematics and spacecraft mechanics that are necessary for the development of the successive work and its understanding.
- **Chapter 3** explains in detail what are the components of CubeSpace's CubeADCS; the sensors, the actuators and the software are presented and the estimation and control modes are introduced.
- **Chapter 4** introduces the concept of Attitude Determination; in this chapter the main estimation modes are described and they are rigorously derived in a mathematical sense.
- **Chapter 5** defines the concept of Attitude Control and derives the control algorithms used in the simulations.
- **Chapter 6** shows in a qualitative way the advantages of the Pitch Sun Tracking attitude mode with respect to the Nadir Pointing one.
- **Chapter 7** describes the simulation environment and subsequently presents a discussion of the main results that outcome from the numerous simulations that were performed.
- **Chapter 8** sums up the thesis conclusions and suggests future developments for this study.

Chapter 2

Spacecraft Mechanics

In this chapter an overview on the basic physical principles necessary to understand the thesis work is given. The motion of the spacecraft can be divided into translational motion, which is described by the equation of orbital mechanics, and rotational motion. The rotational motion is the basis for the study of the attitude, thus great focus will be put on it.

2.1 Orbital Mechanics

The restricted two body problem equation describes the motion of a small body, like a spacecraft, around a central body, like Earth, where the mass of the spacecraft is negligible with respect to that of the central body and Earth is treated as a perfectly spherical body

$$\ddot{\mathbf{r}} = \frac{\mu}{r^3}\mathbf{r} \tag{2.1}$$

where $\mathbf{r}$ is the vector that connects the center of mass of the Earth with that of the spacecraft and $\mu = 3.986004418 \times 10^{14}$ m^3/s^2 is the standard gravitational parameter of the Earth [18]. Given a set of 6 parameters for the initial conditions, Eq. (2.1) can be fully integrated over an indefinite time.

One of the possible parametrizations for the initial conditions involves the vectors of the position $\mathbf{r}$ and the velocity $\mathbf{v}$ of the spacecraft, while another involves the Keplerian parameters, namely the semi-major axis a, the eccentricity e, the inclination i, the longitude of the ascending node Ω, the argument of perigee ω and the true anomaly θ. Alternative representations substitute a and e with the radius of perigee r_p and the radius of apogee r_a, θ with the mean anomaly M or a with the orbital specific angular momentum h.

The solution to the differential equation of motion in terms of Keplerian parameters is the polar equation of a conic section. In fact, depending on the value of the eccentricity e, a trajectory can be any conic section: either a circular orbit ($e = 0$), an elliptic one ($0 < e < 1$), a parabolic trajectory ($e = 1$) or a hyperbolic one ($e > 1$) [19].

2.2 Spacecraft Attitude Parametrizations

Euler's theorem observes that the displacement of a rigid body with a fixed point P is a rotation around an axis that passes through that point, so the rotation itself can be described by four parameters: the three elements of the vector of the rotation axis $\mathbf{a}$ passing through P and the angle that represents the entity of the rotation ϕ. While there is one and only one rotation matrix $\boldsymbol{\mathcal{R}}$ for any given $(\mathbf{a}, \phi)$ pair, which in some occasion can be abbreviated as $\boldsymbol{\phi}$, the uniqueness does not work in the other direction, because one can define an infinite number of $(\mathbf{a}, \phi)$ pairs that lead to the same rotation matrix. To prove this it is sufficient to show that the same rotation matrix can be achieved with the pair $(\mathbf{a}, \phi)$ and $(-\mathbf{a}, 2\pi - \phi)$ or $(\mathbf{a}, \phi + 2m\pi)$, with $m \in \mathbb{N}$. For this reason it is necessary to define different unequivocal ways to parametrize the rotation, the most common of which are Direction Cosines Matrices, Euler Angles and Quaternions [20].

2.2.1 Direction Cosine Matrix

A possible description of the attitude of a satellite utilizes the rotation matrices. A vector can be expressed in an infinity of reference system; in this work, the subscript of the vector will indicate the reference system in which it is expressed. Given two different reference frames $\mathcal{F} = \{\hat{\mathbf{f}}_1, \hat{\mathbf{f}}_2, \hat{\mathbf{f}}_3\}$ and $\mathcal{G} = \{\hat{\mathbf{g}}_1, \hat{\mathbf{g}}_2, \hat{\mathbf{g}}_3\}$ there always exist a matrix $\boldsymbol{\mathcal{R}}_{G \leftarrow F}$, which for brevity will be indicated as $\boldsymbol{\mathcal{R}}_{GF}$, such that

$$\mathbf{u}_G = \boldsymbol{\mathcal{R}}_{GF} \mathbf{u}_F \tag{2.2}$$

$\boldsymbol{\mathcal{R}}_{GF}$ can also be called the Direction Cosine Matrix (DCM) of the basis unit vectors of $\mathcal{G}$ expressed in $\mathcal{F}$ coordinates. In other words, the DCM $\boldsymbol{\mathcal{R}}_{GF}$ contains as columns the unit vectors of $\mathcal{G}$ projected on $\mathcal{F}$

$$\boldsymbol{\mathcal{R}}_{GF} = \begin{bmatrix} \hat{\mathbf{g}}_1 \cdot \hat{\mathbf{f}}_1 & \hat{\mathbf{g}}_1 \cdot \hat{\mathbf{f}}_2 & \hat{\mathbf{g}}_1 \cdot \hat{\mathbf{f}}_3 \\ \hat{\mathbf{g}}_2 \cdot \hat{\mathbf{f}}_1 & \hat{\mathbf{g}}_2 \cdot \hat{\mathbf{f}}_2 & \hat{\mathbf{g}}_2 \cdot \hat{\mathbf{f}}_3 \\ \hat{\mathbf{g}}_3 \cdot \hat{\mathbf{f}}_1 & \hat{\mathbf{g}}_3 \cdot \hat{\mathbf{f}}_3 & \hat{\mathbf{g}}_3 \cdot \hat{\mathbf{f}}_3 \end{bmatrix} \tag{2.3}$$

The opposite transformation of $\boldsymbol{\mathcal{R}}_{GF}$ is the one that transform the frame $\mathcal{G}$ into $\mathcal{F}$ and its DCM is defined as the inverse of the former

$$\boldsymbol{\mathcal{R}}_{FG} = \boldsymbol{\mathcal{R}}_{GF}^{-1} \tag{2.4}$$

By definition any DCM of the type $\boldsymbol{\mathcal{R}}_{GF}$ is orthonormal, which means that $\boldsymbol{\mathcal{R}}_{GF} = \boldsymbol{\mathcal{R}}_{FG}^{T}$. Another important property of DCMs is that a sequence of $n \in \mathbb{N}$ rotations $R_1, R_2, ..., R_n$, characterized by the rotation matrices $\boldsymbol{\mathcal{R}}_1, \boldsymbol{\mathcal{R}}_2, ..., \boldsymbol{\mathcal{R}}_n$ are equivalent to a single rotation R_{1n} which is described by the DCM

$$\boldsymbol{\mathcal{R}}_{1n} = \boldsymbol{\mathcal{R}}_{12} \boldsymbol{\mathcal{R}}_{23} \cdots \boldsymbol{\mathcal{R}}_{(n-1)n} \tag{2.5}$$

Since the rotation matrix is the most intuitive way to define the attitude of the spacecraft, it will also be called attitude matrix and from now on it will be designated by $\mathbf{A}_{GF}$.

The main limitation of the DCM parametrization is that, in order to describe a rotation, where the necessary parameters, as expressed by Euler's theorem, are 4, it uses 9 parameters, so it implies a less effective use of the computational memory of any platform in which it is used. Nonetheless it is the most convenient way to transform vectors between reference systems, so it will be used in this work.

2.2.2 Euler Angles

The Euler Angles parametrization takes advantage of Eq. (2.5). Since a general angular displacement in space has three degrees of freedom and an axis of rotation has one degree of freedom, it is possible to describe any angular displacement as a combination of three principal rotations. Euler angles are commonly defined roll (ϕ), pitch (θ) and yaw (ψ). The order of the rotation influences the final result, because matrix multiplication is not commutative, so when describing a rotation using Euler Angles it is always necessary to also provide the sequence, i.e. 3-2-1, 1-2-3, 3-1-3, etc. One of the most frequently used is the yaw-pitch-roll (3-2-1) sequence, which yields the following matrix ($\mathrm{c}(\cdot)$ and $\mathrm{s}(\cdot)$ stand for $\cos(\cdot)$ and $\sin(\cdot)$ for brevity)

$$\mathbf{A}(\psi,\theta,\phi) = \begin{bmatrix} \mathrm{c}(\psi)\mathrm{c}(\theta) & \mathrm{c}(\psi)\mathrm{s}(\theta)\mathrm{s}(\phi)+\mathrm{s}(\psi)\mathrm{c}(\phi) & -\mathrm{c}(\psi)\mathrm{s}(\theta)\mathrm{c}(\phi)+\mathrm{s}(\psi)\mathrm{s}(\phi) \\ -\mathrm{s}(\psi)\mathrm{c}(\theta) & -\mathrm{s}(\psi)\mathrm{s}(\theta)\mathrm{s}(\phi)+\mathrm{c}(\psi)\mathrm{c}(\phi) & \mathrm{s}(\psi)\mathrm{s}(\theta)\mathrm{c}(\phi)+\mathrm{c}(\psi)\mathrm{s}(\phi) \\ \mathrm{s}(\theta) & -\mathrm{c}(\theta)\mathrm{s}(\phi) & \mathrm{c}(\theta)s(\phi) \end{bmatrix} \tag{2.6}$$

Although this parametrization may be the most intuitive and so the easiest to understand, it is affected by a fatal flaw: Gymbal Lock. The matrix in Eq. (2.6) has a singularity for $\theta = \pm 90°$, which means that when the spacecraft is operating on a pitch angle of ±90° a yaw rotation coincides with a roll rotation, so the sequence can no longer be considered valid. For this reason the Euler Angles parametrization can only be used in systems where the pitch angle never approaches ±90° and it is never used to describe the attitude of a spacecraft.

2.2.3 Quaternions

According to the definition that Hamilton originally gave of it, a quaternion is a four-dimensional entity constituted of a real number and three imaginary units [21]. In engineering applications, though, it is more convenient to define the quaternion as a 4×1 matrix consisting of a tridimensional vector part and a scalar part:

$$\mathbf{q} = \begin{bmatrix} \mathbf{q}_{1:3} \\ q_4 \end{bmatrix} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \tag{2.7}$$

A unique convention in the definition of quaternions does not exist, so in the literature they can also be defined with the scalar taking the first position in the vector, in which case it is called q_0. The following discussion is based on the definition with the scalar in the last position.

Quaternion Algebra

Quaternions have a slightly different algebra than conventional vectors [22]. The norm is the only operation that is identical for vector and quaternions algebra

$$q = ||\mathbf{q}|| = \sqrt{{q_1}^2 + {q_2}^2 + {q_3}^2 + {q_4}^2} \tag{2.8}$$

All of the quaternions used in this work will be unit quaternions, for which $||\mathbf{q}|| = 1$. The conjugate of a quaternion is defined as

$$\mathbf{q}^* = \begin{bmatrix} -\mathbf{q}_{1:3} \\ q_4 \end{bmatrix} \tag{2.9}$$

This allows to define also the inverse of a quaternion as

$$\mathbf{q}^{-1} = \frac{\mathbf{q}^*}{q} \tag{2.10}$$

Lastly it is important to define two product operations with the pair of general quaternions $\mathbf{q}$ and $\mathbf{p}$:

$$\mathbf{p} \otimes \mathbf{q} = \begin{bmatrix} q_4 \cdot \mathbf{p}_{1:3} + p_4 \cdot \mathbf{q}_{1:3} - \mathbf{p}_{1:3} \times \mathbf{q}_{1:3} \\ p_4 \cdot q_4 - \mathbf{p}_{1:3} \cdot \mathbf{q}_{1:3} \end{bmatrix} \tag{2.11a}$$

$$\mathbf{p} \odot \mathbf{q} = \begin{bmatrix} q_4 \cdot \mathbf{p}_{1:3} + p_4 \cdot \mathbf{q}_{1:3} + \mathbf{p}_{1:3} \times \mathbf{q}_{1:3} \\ p_4 \cdot q_4 - \mathbf{p}_{1:3} \cdot \mathbf{q}_{1:3} \end{bmatrix} \tag{2.11b}$$

It is straightforward that

$$\mathbf{q} \otimes \mathbf{p} = \mathbf{p} \odot \mathbf{q} \tag{2.12}$$

So both quaternion products, similarly to the external product for vectors, are non-commutative, but they are associative and distributive

$$\mathbf{q} \otimes \mathbf{p} \neq \mathbf{p} \otimes \mathbf{q} \tag{2.13a}$$

$$\mathbf{q} \otimes (\mathbf{p} \otimes \mathbf{w}) = (\mathbf{q} \otimes \mathbf{p}) \otimes \mathbf{w} \tag{2.13b}$$

$$\mathbf{q} \otimes (\mathbf{p} + \mathbf{w}) = \mathbf{q} \otimes \mathbf{p} + \mathbf{q} \otimes \mathbf{w} \tag{2.13c}$$

Analogous equations hold for the $\odot$ product. Note that if the convention with q_0 was taken, the operations would be different. Similarly to the cross product for vectors, quaternion products can be represented in a matrix form

$$[\mathbf{q}\otimes] = \begin{bmatrix} q_4 I_3 - [\mathbf{q}_{1:3}\times] & \mathbf{q}_{1:3} \\ -(\mathbf{q}_{1:3})^{\mathrm{T}} & q_4 \end{bmatrix} = \begin{bmatrix} q_4 & q_3 & -q_2 & q_1 \\ -q_3 & q_4 & -q_1 & q_2 \\ q_2 & -q_1 & q_4 & q_3 \\ -q_1 & -q_2 & -q_3 & q_4 \end{bmatrix} \tag{2.14}$$

Where I_3 is the 3×3 identity matrix and $[\mathbf{q}_{1:3}\times]$ is the matrix representation of the cross product.

A last useful property of quaternions is the transformation between reference frames: if the quaternion $\mathbf{q}_{GF}$ transforms reference frame $\mathcal{F}$ into $\mathcal{G}$, $\mathbf{q}_{FG} = \mathbf{q}_{GF}^{-1}$ is the opposite transformation. Moreover, if $\mathbf{q}_{HG}$ transforms $\mathcal{G}$ into $\mathcal{H}$, the following equation is valid

$$\mathbf{q}_{HF} = \mathbf{q}_{GF} \otimes \mathbf{q}_{HG} \tag{2.15}$$

Transformation from Quaternion to DCM

The quaternion representation of a rotational displacement is based on the previously mentioned Euler's theorem: the four scalar parameters that compose the quaternions can unequivocally parametrize any rotation defined by the pair $(\mathbf{a}, \phi)$. If the the rotation $(\mathbf{a}, \phi)$ transforms the reference frame $\mathcal{F}$ into $\mathcal{G}$, the equivalent quaternion representation is [22]

$$\mathbf{q} = \begin{bmatrix} \hat{\mathbf{a}} \sin\left(\frac{\phi}{2}\right) \\ \cos\left(\frac{\phi}{2}\right) \end{bmatrix} \tag{2.16}$$

So it is possible to define the rotation matrix in terms of the elements of the quaternion [23]

$$\mathbf{A}(\mathbf{q}) = \begin{bmatrix} q_1^2 - q_2^2 - q_3^2 + q_4^2 & 2(q_1q_2 + q_3q_4) & 2(q_1q_3 - q_2q_4) \\ 2(q_2q_1 - q_3q_4) & -q_1^2 + q_2^2 - q_3^2 + q_4^2 & 2(q_2q_3 + q_1q_4) \\ 2(q_3q_1 + q_2q_4) & 2(q_3q_2 - q_1q_4) & -q_1^2 - q_2^2 + q_3^2 + q_4^2 \end{bmatrix} \tag{2.17}$$

When the transformation to be represented is a null rotation the quaternion is the identity quaternion

$$\mathbf{I}_q = \begin{bmatrix} \mathbf{0}_3 \\ 1 \end{bmatrix} \tag{2.18}$$

This yields the identity rotation matrix $\mathbf{I_3}$.

The first very important perk of using this representation for attitude transformation is that two successive rotations represented by the quaternions $\mathbf{p}$ and $\mathbf{w}$ can be represented by the single quaternion $\mathbf{q}$ such that $\mathbf{q} = \mathbf{w} \otimes \mathbf{p}$, so that

$$\mathbf{A}(\mathbf{q}) = \mathbf{A}(\mathbf{w} \otimes \mathbf{p}) = \mathbf{A}(\mathbf{w})\mathbf{A}(\mathbf{p}) \tag{2.19}$$

Even if from a physical point of view the quaternion appears to not have any sense, it is a quicker computational tool than the DCMs. Moreover it does not present any singularities, so it can be used for absolutely any attitude transformation. For this reasons, even if it requires calculation that may seem more complex and less intuitive than the ones required by the two previously seen methods, in this work quaternions will be used in the attitude determination and control algorithms.

Transformation from DCM to Quaternion

Eq. (2.17) Transforms the attitude quaternion into the attitude DCM matrix, but also the opposite transformation is possible. The elements of a quaternion can be calculated from the main diagonal of the corresponding DCM.

$$q_1 = \frac{1}{2}\sqrt{1 + \mathbf{A}_{11} - \mathbf{A}_{22} - \mathbf{A}_{33}} \tag{2.20a}$$

$$q_2 = \frac{1}{2}\sqrt{1 - \mathbf{A}_{11} + \mathbf{A}_{22} - \mathbf{A}_{33}} \tag{2.20b}$$

$$q_3 = \frac{1}{2}\sqrt{1 - \mathbf{A}_{11} - \mathbf{A}_{22} + \mathbf{A}_{33}} \tag{2.20c}$$

$$q_4 = \frac{1}{2}\sqrt{1 + \mathbf{A}_{11} + \mathbf{A}_{22} + \mathbf{A}_{33}} \tag{2.20d}$$

2.3 Reference Frames

The accurate definition of different reference frames is essential to the proper functioning of an ADCS. A vector that indicates the position of an object, like the Sun, or the orientation of a face, like that of a solar panel, is always described in space in terms of magnitude and direction. For the purpose of Attitude Determination and Control, magnitude is superfluous, because only the orientation of the unit vectors is needed. This orientation can be described in different reference systems. The importance of defining many reference systems comes also from the fact that the attitude of a rigid body, like the spacecraft, can be defined as the orientation of a reference system that is embedded within it with respect to a second one [20].

An important distinction has to be made between inertial and non-inertial reference frames: in the former “the motion of a particle not subject to forces is in a straight line at constant speed” [24], so the reference frame itself moves at a constant speed. The latter are frames that are subjected to an external acceleration or to an angular rate with respect to an inertial frame, so any particle, or body, embedded in them undergoes a set of apparent accelerations.

The reference frames that will be used in this work are presented in the following discussion. All of them are defined by an origin and two orthogonal directions; the third direction is defined starting from the other two in order to form a right-handed orthonormal triad.

2.3.1 Body Frame

The spacecraft Body frame is centered at a chosen position inside the spacecraft's body, usually the center of mass, and its axis are most commonly taken as the three principal axis of inertia of the body [23]. The Body frame is important during the assembly of the satellite to align the various components, but it is even more important for Attitude Determination and Control, because when a certain attitude has to be reached, the typical command that the control algorithm performs is to align the Body frame with a second frame. The Body frame is designated by

$$\mathcal{B} = \{\hat{\mathbf{b}}_1, \hat{\mathbf{b}}_2, \hat{\mathbf{b}}_3\} \tag{2.21}$$

2.3.2 Earth Centered Inertial frame

Earth Centered Inertial (ECI) has its origin in the center of mass of the Earth. It will be indicated by

$$\mathcal{I} = \{\hat{\mathbf{i}}_1, \hat{\mathbf{i}}_2, \hat{\mathbf{i}}_3\} \tag{2.22}$$

The $\hat{\mathbf{i}}_1$ axis points to the direction of the vernal equinox, the $\hat{\mathbf{i}}_3$ axis is in the direction of the North pole of the planet and the $\hat{\mathbf{i}}_2$ is defined consequently to form a right handed triad [25].

Since the Earth is on a circular motion around the Sun the ECI frame cannot be considered absolutely inertial, but for the purpose of attitude control in Earth orbit its acceleration can be neglected [23].

2.3.3 Orbital Frame

The Orbital frame, also called Local Vertical Local Horizon (LVLH), is centered on the spacecraft center of mass, so it is a non-inertial frame because it rotates around the Earth. It is designated by

$$\mathcal{O} = \{\hat{\mathbf{o}}_1, \hat{\mathbf{o}}_2, \hat{\mathbf{o}}_3\} \tag{2.23}$$

The $\hat{\mathbf{o}}_3$ unit vector is parallel to the direction of the Nadir vector, the $\hat{\mathbf{o}}_2$ is parallel and opposite to the direction of the angular momentum of the orbit (the orbit's normal) and $\hat{\mathbf{o}}_1$, completes the right-handed triad.

The LVLH reference frame is of particular importance in Nadir Pointing spacecrafts, because in this case the command of the controller is to make the Body frame coincident with it.

2.3.4 Pitch Sun Tracking Frame

The Pitch Sun Tracking frame is defined appositely for the PST attitude mode. It will be defined by

$$\mathcal{S} = \{\hat{\mathbf{s}}_1, \hat{\mathbf{s}}_2, \hat{\mathbf{s}}_3\} \tag{2.24}$$

It is centered on the spacecraft center of mass and it can be thought of as a rotation of the orbital frame around the $\hat{\mathbf{o}}_2$ axis, so it is defined in terms of the orbital frame's unit vectors. The Sun unit vector can be described in the orbital frame as

$$\hat{\mathbf{S}}_O = S_1\hat{\mathbf{o}}_1 + S_2\hat{\mathbf{o}}_2 + S_3\hat{\mathbf{o}}_3 \tag{2.25}$$

The $\hat{\mathbf{s}}_2$ axis is always coincident with the $\hat{\mathbf{o}}_2$ of the orbital frame, whereas the other two axes result from a rotation around $\hat{\mathbf{o}}_2$:

$$\hat{\mathbf{s}}_1 := \cos(\sigma + \alpha)\hat{\mathbf{o}}_1 + \sin(\sigma + \alpha)\hat{\mathbf{o}}_3 \tag{2.26a}$$

$$\hat{\mathbf{s}}_2 := \hat{\mathbf{o}}_2 \tag{2.26b}$$

$$\hat{\mathbf{s}}_3 := \hat{\mathbf{s}}_1 \times \hat{\mathbf{s}}_2 \tag{2.26c}$$

Where σ is the rotation that aligns $\hat{\mathbf{s}}_1$ with the projection of the Sun vector on the orbital x-z plane

$$\sigma = \tan^{-1}(S_{3o}/S_{1o}) \tag{2.27}$$

α is an additional rotation that allows to orient the x and z faces of the body in order to maximize the energy production.

2.4 Spacecraft Attitude Kinematics

The Attitude Kinematics deals with the evolution of angular displacements in time. Let $\mathcal{F}$ and $\mathcal{G}$ be the usual pair of reference frames, if the relationship between the two frames is time dependent, the DCM that transforms the versors of $\mathcal{F}$ into those of $\mathcal{G}$ is also time dependent and it is possible to define the angular velocity vector of one with respect to the other one according to Newton's definition of derivative

$$\boldsymbol{\omega}_G^{GF}(t) = \lim_{\Delta t \to 0} \frac{\Delta\boldsymbol{\phi}_G^{GF}(t)}{\Delta t} \tag{2.28}$$

Where $\boldsymbol{\omega}_G^{GF}$ and $\Delta\boldsymbol{\phi}_G^{GF}$ are respectively the angular rate and the difference of angular displacement over the time Δt of $\mathcal{F}$ with respect to $\mathcal{G}$ expressed in $\mathcal{G}$ components. Using this definition the fundamental equation of Attitude Kinematics can be written

$$\dot{\mathbf{A}}_{GF}(t) = -\boldsymbol{\omega}_G^{GF}(t) \times \mathbf{A}_{GF}(t) \tag{2.29}$$

which is a linear differential equation with a time dependent coefficient, so it cannot be solved analytically except in particular cases. If the history of the angular velocity is known, the integration is possible, so that at every instant the attitude of $\mathcal{F}$ with respect to $\mathcal{G}$ is known. Alternatively, it is also possible to use Eq. (2.29) to compute the angular velocity if the history of the attitude is known (from now on the time dependency will be implied for clarity)

$$[\boldsymbol{\omega}_G^{GF}\times] = -\dot{\mathbf{A}}_{GF}\mathbf{A}_{GF}^T = -\dot{\mathbf{A}}_{GF}\mathbf{A}_{FG} \tag{2.30}$$

The angular rate presents a very useful property relating the angular rates of two systems with respect to each other [23]

$$\boldsymbol{\omega}^{GF} = -\boldsymbol{\omega}^{FG} \tag{2.31}$$

In many occasions, more than two reference frames are needed to perform determination and control operations; in these cases it is useful to have a formula that relates the angular rates of each frame with respect to the others. Let a new frame be introduced, $\mathcal{H}$; being careful as to representing each of the three vectors in the same reference frame, the angular rate vectors can be added as follows

$$\boldsymbol{\omega}^{HF} = \boldsymbol{\omega}^{HG} + \boldsymbol{\omega}^{GF} \tag{2.32}$$

Eq. (2.32) can be combined with Eq. (2.31) to obtain the angular rate of any frame knowing the angular rates of two other frames with respect to it.

2.4.1 Vector Kinematics

Given a vector $\mathbf{u}$, it can be represented in two reference frames $\mathcal{F}$ and $\mathcal{G}$ as expressed in Eq. (2.2), so the representation of its time derivative in the two frames can also be related via

$$\dot{\mathbf{u}}_G = \mathbf{A}_{GF}\dot{\mathbf{u}}_F - \boldsymbol{\omega}_G^{GF} \times \mathbf{u}_G \tag{2.33}$$

This is another fundamental and very useful equation of rotational mechanics. For Eq. (2.33), as for all the previous equation, there is no discrimination between inertial and non-inertial frames, because there is no indication on which of the frames is rotating in an absolute sense, but only the relative motion is of interest.

If Eq. (2.33) is differentiated, one can obtain the equation for the acceleration of the vector as expressed in the two reference frames

$$\ddot{\mathbf{u}}_G = \mathbf{A}_{GF}\ddot{\mathbf{u}}_F - \boldsymbol{\omega}_G^{GF} \times (\boldsymbol{\omega}_G^{GF} \times \mathbf{u}_G) - 2\boldsymbol{\omega}_G^{GF} \times \dot{\mathbf{u}}_G - \dot{\boldsymbol{\omega}}_G^{GF} \times \mathbf{u}_G \tag{2.34}$$

where, if $\mathbf{u}$ is a position vector, the terms on the right side have a special meaning: the first term is the relative acceleration, the second term is the centripetal acceleration, the third term is the Coriolis acceleration and the fourth one is the Euler acceleration.

2.4.2 Kinematic of Quaternions

The previous discussion is valid when a rotation matrix, or DCM, is used to describe the attitude orientation. If a quaternion is used, a different equation has to be used to define the angular rate. The time derivative of the quaternion can be written as

$$\dot{\mathbf{q}} = \frac{1}{2}\mathbf{q} \odot \boldsymbol{\omega} \tag{2.35}$$

Where $\mathbf{q} = \mathbf{q}_{GF}$ and $\omega = \boldsymbol{\omega}_G^{GF}$ and the subscripts have been neglected for clarity. It is convenient to write Eq. (2.36) in a matrix fashion

$$\dot{\mathbf{q}} = \frac{1}{2}\Xi(\mathbf{q})\boldsymbol{\omega} \tag{2.36}$$

where

$$\Xi(\mathbf{q}) = \begin{bmatrix} q_4 & -q_3 & q_2 \\ q_3 & q_4 & -q_1 \\ -q_2 & q_1 & q_4 \\ -q_1 & -q_2 & -q_3 \end{bmatrix} \tag{2.37}$$

So it is also possible to write the angular rate in terms of the quaternion and its derivative

$$\boldsymbol{\omega} = 2\Xi^{\mathrm{T}}(\mathbf{q})\dot{\mathbf{q}} \tag{2.38}$$

2.5 Spacecraft Attitude Dynamics

The rotational dynamics deal with the torques acting on the spacecraft and the consequent evolution of the angular momentum and the kinetic energy. All of the discussion of this section involves vectors represented in the body frame, so, unless there is a risk of ambiguity, the subscript B will be omitted [26].

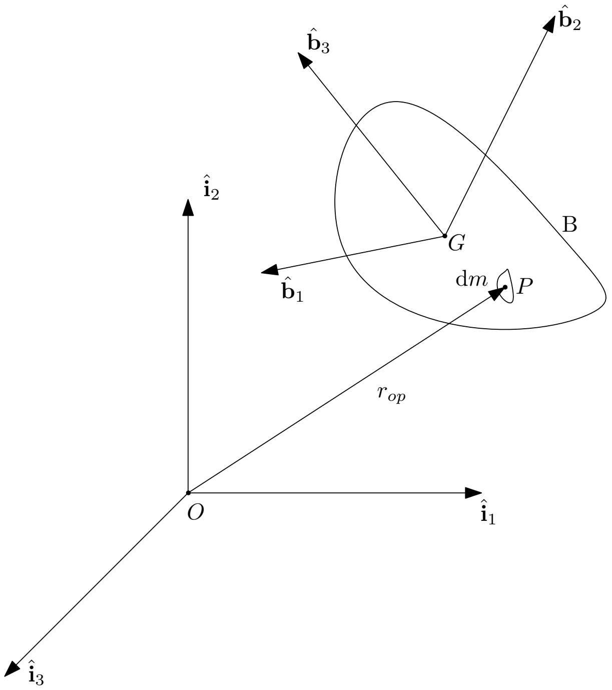


Figure 2.1: Inertial and Body reference frames.

Taking as a reference the representation of Fig. 2.1, the starting point is the definition of the angular momentum of the body B with respect to a generic pole O

$$\mathbf{h}_o = \int_B \mathbf{r}_{op} \times \mathbf{v}_p \mathrm{d}m \tag{2.39}$$

where the integration is performed over the whole volume of the rigid body of the spacecraft B with $\mathrm{d}m$ the infinitesimal mass of the generic point P of the body; $\mathbf{r}_{op}$ and $\mathbf{v}_p$ are respectively the distance of P from the point O and the absolute velocity of P.

The velocity $\mathbf{v}_p$ can be expressed as the sum of the velocity of the pole, the relative velocity of P with respect to O and a term dependent on the rotational velocity of the reference frame

$$\mathbf{v}_p = \frac{\mathrm{d}\mathbf{r}_{op}}{\mathrm{d}t} = \mathbf{v}_o + \dot{\mathbf{r}}_{op} + \boldsymbol{\omega} \times \mathbf{r}_{op} \tag{2.40}$$

Since the choice of the pole in Eq. (2.39) is arbitrary, one can chose it to be coincident with the center of mass of the spacecraft G without loss of generality ($O = G$). By hypothesis the spacecraft is rigid, so there is no relative motion with respect to the center of mass

$$\dot{\mathbf{r}}_{op} = 0 \tag{2.41}$$

Moreover, by definition of center of mass it is true that

$$\left(\int_B \mathbf{r}_{op} \mathrm{d}m \right) \times \mathbf{v}_o = 0 \tag{2.42}$$

Now one can substitute Eq. (2.40), simplified using Eq. (2.41) and Eq. (2.42), into Eq. (2.39) to obtain

$$\mathbf{h} = \int_B \mathbf{r}_p \times (\boldsymbol{\omega} \times \mathbf{r}_p) \mathrm{d}m \tag{2.43}$$

Following the convention, the subscript is omitted because the pole is the center of mass. Using the cross product properties it is possible to show that the integral in Eq. (2.43) is equivalent to

$$\mathbf{h} = \mathbf{J}\boldsymbol{\omega} \tag{2.44}$$

$\mathbf{J}$ is the symmetric matrix called the moment of inertia tensor, defined as

$$\mathbf{J} = \begin{bmatrix} \int_B (y_p^2 + z_p^2) \mathrm{d}m & \int_B -(x_p y_p) \mathrm{d}m & \int_B -(x_p z_p) \mathrm{d}m \\ \int_B -(y_p x_p) \mathrm{d}m & \int_B (z_p^2 + x_p^2) \mathrm{d}m & \int_B -(y_p z_p) \mathrm{d}m \\ \int_B -(z_p x_p) \mathrm{d}m & \int_B -(z_p y_p) \mathrm{d}m & \int_B (x_p^2 + y_p^2) \mathrm{d}m \end{bmatrix} \tag{2.45}$$

where $\mathbf{r}_p = \begin{bmatrix} x_p & y_p & z_p \end{bmatrix}^{\mathrm{T}}$ in Body coordinates.

Newton's Second Law in rotational dynamics asserts that the derivative of the angular momentum with respect to the center of mass is equivalent to the sum of the external torques acting on the body

$$\frac{\mathrm{d}\mathbf{h}}{\mathrm{d}t} = \boldsymbol{\tau}_{ext} \tag{2.46}$$

Eq. (2.46) is the fundamental equation of rotational dynamics for a rigid body.

Since Eq. (2.46) is expressed in the Body frame, it is possible to decompose the derivative of the angular momentum into two terms: one representing the variation of the quantity in the Body frame, the other representing the variation dependent on the rotation rate of the reference frame with respect to an inertial frame, such as the ECI frame:

$$\frac{\mathrm{d}\mathbf{h}}{\mathrm{d}t} = \dot{\mathbf{h}} + \boldsymbol{\omega}_B^{BI} \times \mathbf{h} \tag{2.47}$$

Combining Eq. (2.46) and Eq. (2.47), one can obtain the final form of the equation of dynamics

$$\dot{\mathbf{h}} = \boldsymbol{\tau}_{ext} - \boldsymbol{\omega}_B^{BI} \times \mathbf{h} \tag{2.48}$$

2.6 Orbital Perturbations in LEO

The restricted two body problem presented in Section 2.1 is a very simplified way of studying the orbit of a spacecraft. No actual object follows a Keplerian orbit because it is subjected in every instant to secondary accelerations that have to be accounted for in the differential equation of motion. So in reality the equation of motion does not have an analytical solution, contrarily to what happens with Eq. (2.1). For this reason complex models have to be develop to keep track of all the perturbative forces acting on the spacecraft, in order to write the equation of motion in the most accurate way possible and integrate it to find a trajectory that is as close as possible to the actual trajectory described by the satellite. Moreover, some of these forces do not act on an uniform way on the satellite, thus they impose a torque around the center of mass, causing the spacecraft to rotate [19]. The most important of these perturbations are presented in the following discussion.

Perturbative forces can be divided into four categories: high order gravitational harmonics caused by non-spherical mass distribution of the Earth, solar radiation pressure acceleration, atmospheric aerodynamic drag and third body accelerations of the Moon and the Sun. Since the spacecraft is in an ISS orbit, the last one of these perturbations will be neglected.

2.6.1 High Order Gravitational Harmonics

A more accurate model of the shape of the Earth includes the bulge at the equator and the flattening at the poles: in fact our planet has a mean equator radius of 6378 km and a mean polar radius of 6356 km.

The acceleration of a satellite can be found using the gradient of the gravitational potential function

$$\Phi(r,\phi,\psi) = \frac{\mu}{r}\Big[1 - \sum_{n=2}^{\infty} J_n \Big(\frac{R_E}{r}\Big)^n P_n \sin\phi + \sum_{n=2}^{\infty}\sum_{m=1}^{n}\Big(\frac{R_E}{r}\Big)^n (C_{nm}\cos m\psi + S_{nm}\sin m\psi)P_{mn}(sin\phi)\Big] \tag{2.49}$$

where r, ϕ and ψ are spherical coordinates of the Earth, μ is as usual the gravitational constant of the planet, R_E is its equatorial radius, P_n are the Legendre polynomials and C_{nm}, S_{nm} and J_n are the zonal coefficients, dimensionless constants which decrease in value with the indexes n and m.

The zonal coefficients can be separated into tesseral terms (in the second summation of Eq. (2.49)), which divide the Earth into sections defined by longitude and latitude, and sectoral (in the first summation of Eq. (2.49)), which divide the Earth into slices and only depend on latitude. A geopotential model includes a certain number of zonal coefficients, each of which corresponds to a gravitational harmonics of the geopotential function. The more terms are included in the model, the more accurate it is. In this work, the model developed in [27], including harmonics up to the 120th, order is employed.

Since the acceleration caused by high order gravitational harmonics is derived from a potential function, it is conservative, thus it does not modify the total energy of the orbit, but it only causes a periodic variation of all the orbital elements. The most important of these variations are caused by the J_2 harmonics and involve the argument of periapsis and the longitude of the ascending node of the orbit, which undergo a significant periodic change [18].

2.6.2 Gravity Gradient Torque

The gravitational field is not uniform over a body, because the entity of the acceleration depends of the position of the single infinitesimal element P that is being attracted, thus, if only the first order of the gravitational potential function is considered, the acceleration acting on each of these elements obeys to the gravitational law

$$\mathrm{d}\mathbf{a} = -\frac{\mu}{||\mathbf{r}_p||^3}\mathbf{r}_p \mathrm{d}m \tag{2.50}$$

and the torque produced by this acceleration, integrated over the whole body, is

$$\mathrm{d}\boldsymbol{\tau}_{gg} = \int_B \mathbf{r}_{cp} \times \mathrm{d}\mathbf{f} \tag{2.51}$$

where r_{cp} is the distance of the infinitesimal point P from the center of mass C. Choosing the principal axes of inertia to represent the body, Eq. (2.51) can be written in its final form

$$\boldsymbol{\tau}_{gg} = \frac{3}{2}\frac{\mu}{r_c^3}\begin{bmatrix} (I_3 - I_2)\sin(2\phi)\cos^2(\theta) \\ (I_1 - I_3)\sin(2\theta)\cos(\phi) \\ (I_2 - I_1)\sin(2\theta)\sin(\phi) \end{bmatrix} \tag{2.52}$$

where r_c is the distance of the center of mass of the spacecraft from the center of mass of the Earth and the torque was written in function of the Yaw, Pitch and Roll angles for easiness of exposition.

2.6.3 Aerodynamic Drag

In LEO orbits the Aerodynamic drag is the most important non-gravitational perturbative force. It is a decelerating force, because it acts in the direction opposite to the spacecraft velocity, thus it removes energy from the orbit and it cannot be modeled as a conservative phenomenon [20].

Acceleration

Let α be the angle of attack of a single air particle with respect to the infinitesimal portion of area of the spacecraft $\mathrm{d}A$, the total force acting on the spacecraft due to atmospheric drag is the integral of the contribution of every single particle impacting onto every single infinitesimal portion of the contact area:

$$\mathbf{f}_d = \int_A \mathrm{H}\big(\cos(\alpha)\big)\cos(\alpha)\rho||\mathbf{u}_r||\mathbf{u}_r \mathrm{d}A \tag{2.53}$$

where $\mathrm{H}(\cdot)$ is the Heaviside function, $\mathbf{u}_r$ is the velocity of the particle of air relative to the spacecraft body and ρ is the density of the atmosphere. If the assumption is made that the spacecraft is fast enough that its velocity is much higher than the average thermal motion of the atmosphere, the particles hitting it all have the same velocity equal and opposite to the velocity of the spacecraft $\mathbf{u}_r = -\mathbf{v}$. The speed of the spacecraft and the density of the air do not depend on the contact area, so it is possible to write the integral in a compact form as the product of the frontal area and a drag coefficient C_D:

$$C_D A = \int_A H\big(\cos(\alpha)\big)\cos(\alpha)\mathrm{d}A \tag{2.54}$$

With this notation and recalling Newton's Second Law ($\mathbf{f} = m\mathbf{a}$), Eq. (2.53) becomes

$$\mathbf{a}_d = -\frac{1}{2}\rho B||\mathbf{v}||\mathbf{v} \tag{2.55}$$

where B takes the name of ballistic coefficient

$$B = 2\frac{C_D A}{m} \tag{2.56}$$

The drag coefficient C_D for a small CubeSat is usually included in a range such as $1.8 < C_D < 2.4$ [28]. In this work a value of $C_D = 2$ has been chosen.

The main issue in modelling the drag perturbation is that a precise model is needed for the atmospheric density ρ, if an accurate result is sought. A variety of models are available in the literature; in this work the NRLMSISE-00 atmosphere model has been used [29].

Torque

It is possible to express the torque caused by the aerodynamic drag perturbation on the spacecraft body using the definition of center of pressure

$$\mathbf{c}_p = \frac{1}{A}\int_A H\big(\cos(\alpha)\big)\cos(\alpha)\mathbf{r}_{cp}\mathrm{d}A \tag{2.57}$$

where $\mathbf{r}_{cp}$ is the distance of the infinitesimal particle hitting the surface from the center of mass and A is computed using Eq. (2.54). By definition of center of pressure, the torque is now computed as

$$\boldsymbol{\tau}_d = \mathbf{c}_p \times \mathbf{f}_d \tag{2.58}$$

which can be written in extended form as

$$\boldsymbol{\tau}_d = -\frac{1}{2} m \rho B ||\mathbf{v}|| \mathbf{c}_p \times \mathbf{v} \tag{2.59}$$

2.6.4 Solar Radiation Pressure

Another dissipative perturbative force that acts on every spacecraft, even though only when it is not in eclipse, is the solar radiation pressure. Although in engineering application usually radiation is only intended as a wave, one should always remember its double nature of wave-corpuscular entity. The radiation incoming from the Sun includes a very large quantity of photons, which possess an infinitesimal mass, and a small number of more energetic and heavier particles. Since this flux has a certain mass, it carries a momentum, which is transferred to the foreign body that intercepts it, e.g. the spacecraft.

Acceleration

The exchange of momentum between one particle and the spacecraft can also be expressed in the form of a pressure multiplied by the area that is illuminated

$$\mathrm{d}\mathbf{f}_{sp} = H(\cos(\alpha)) \cos(\alpha) p \hat{\mathbf{s}} \mathrm{d}A \tag{2.60}$$

where $\hat{\mathbf{s}}$ is the unit vector in the Poynting vector direction, given by the cross product of the local electric field and the local magnetic field of the wave ($\mathbf{s} = \mathbf{e} \times \mathbf{h}$), and p is the pressure of radiation on a totally absorbing surface. Since the power density of the Sun is more or less constant in any Low Earth Orbit, p is a constant

$$p = \frac{||\mathbf{s}||}{c} \tag{2.61}$$

The angle α is formed by the intersection of the Poynting vector with the inward normal surface of unit magnitude

$$\cos(\alpha) = \hat{\mathbf{s}} \cdot \hat{\mathbf{n}} \tag{2.62}$$

Integrating Eq. (2.61) over all the contact area and remembering Newton's Second Law

$$\mathbf{a}_{sp} = \frac{pA}{m} \hat{\mathbf{s}} \tag{2.63}$$

where once again, similarly to the case of the atmospheric drag, the contact area is

$$A = \int_A H(\cos(\alpha)) \cos(\alpha) \mathrm{d}A \tag{2.64}$$

Torque

By definition, the torque is the integral of the contribution of each infinitesimal force multiplied by the distance of the infinitesimal surface illuminated by the photon from the center of pressure of the spacecraft

$$\boldsymbol{\tau}_{sp} = \mathbf{c}_p \times \mathbf{f}_{sp} \tag{2.65}$$

which ultimately results in

$$\boldsymbol{\tau}_{sp} = pA\mathbf{c}_p \times \hat{\mathbf{s}} \tag{2.66}$$

where $\mathbf{c}_p$ is the position of the center of pressure, defined in a similar fashion to what was done for the aerodynamic drag in Eq. (2.57).

2.6.5 Magnetic Torque

The spacecraft internal circuits and electrical devices cause it to be affected by a parasitic magnetic dipole moment $\mathbf{m}_{par}$, which, via the interaction with the magnetic field of the Earth $\mathbf{b}$ imposes a torque on the spacecraft:

$$\boldsymbol{\tau}_m = \mathbf{m}_{par} \times \mathbf{b} \tag{2.67}$$

Similarly to the gravitational field, the magnetic field of the Earth can be modeled using a potential function with an infinite number of harmonics

$$V(r, \phi, \psi) = R_E \Big[\sum_{n=1}^{\infty} \sum_{m=0}^{n} (g_{nm} \cos m\psi + h_{nm} \sin m\psi) \Big(\frac{R_E}{r} \Big)^{n+1} P_{nm}(\cos \phi) \Big] \tag{2.68}$$

Where r, ϕ and ψ are the spherical coordinates, R_E is the mean equatorial radius of the Earth, g_{nm} and h_{nm} are Gauss coefficients and $P_{nm}(x)$ is the Legendre function. The gradient of this potential function yields the magnetic field

$$\mathbf{b} = -\vec{\nabla} V \tag{2.69}$$

Thus a model is needed for computation of the harmonics up to a certain order of degree n and m. In this work the World Magnetic Model (WMM) 2020 was used[1].

[1] https://www.ngdc.noaa.gov/geomag/WMM/DoDWMM.shtml

Chapter 3

Hardware Configuration

ORCASat is a 2U CubeSat, as it can be seen in Fig. 3.1. At the time this document is being written, its development is still not concluded, but the maximum estimated mass of the satellite is 3.6 kg [30]. In Fig. 3.2 the accommodations for the payload and all the subsystems are shown. One of the focuses of the design is to keep the principal axes of inertia of the spacecraft as close as possible to the Body frame axis as represented in Fig. 3.1, so that in the attitude control there is as little coupling as possible between the control torques. The estimation for the tensor of inertia is [16]

$$\mathbf{J} = \begin{bmatrix} 0.003 & 0 & 0 \\ 0 & 0.007 & 0 \\ 0 & 0 & 0.008 \end{bmatrix} \text{kg m}^2 \tag{3.1}$$

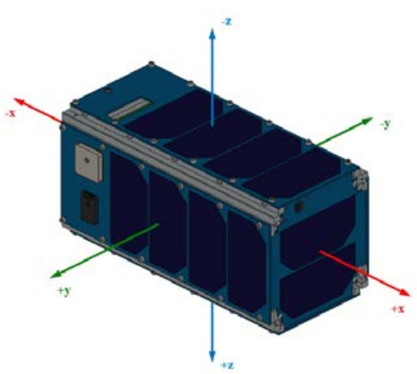


Figure 3.1: Body Axes configuration [30].

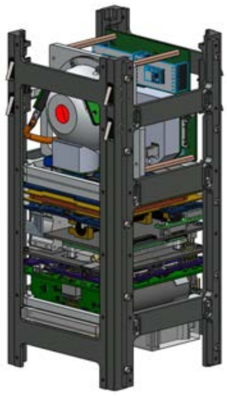


Figure 3.2: ORCASat's internal structure [30].

3.1 Solar Arrays

The CubeSat is powered by a set of body mounted solar panels, produced by Azur Space, which have a BOL efficiency of 29.6% [31], for which the configuration is presented in Table 3.1. The power produced by them is computed considering a constant value of solar power density (1350 W/m^2)

$$P = P_{peak} \cos(\theta) \tag{3.2}$$

where θ is the angle between the normal to the panel and the Sun vector.

Table 3.1: Solar Panels configuration.

Face	Number of panels	P_{peak} [W]
$\pm x$	2	2.4
$\pm y$	4	4.8
$+z$	0	0
$-z$	4	4.8

3.2 CubeADCS Y-Momentum

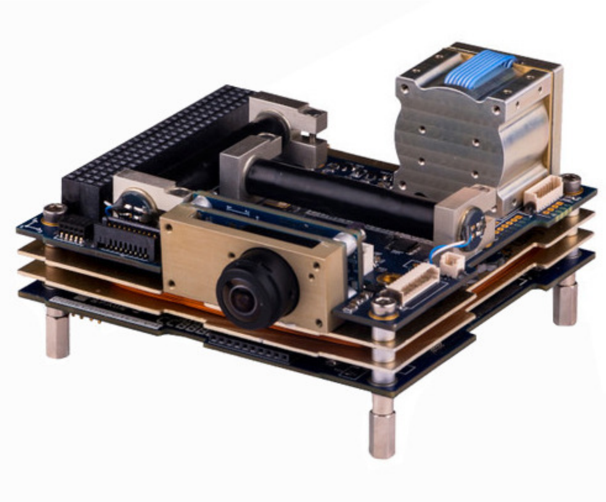


Figure 3.3: CubeADCS Y-Momentum[1].

The ADCS that will be implemented in ORCASat was produced by CubeSpace, a company that has been designing and building Attitude Determination and Control Subsystems for CubeSats since 2014 and which sells systems and components with reliable flight heritage[2]. In particular, the full ADCS solutions (called CubeADCS) proposed by CubeSpace are three: Magnetic, Y-Momentum and 3-Axes. The first one only comprises magnetic actuators, so it is mainly used on 1U CubeSats to detumble the spacecraft and put it into a controlled slow spin. The third one presents three momentum wheels that are able to accurately orient the CubeSat in Space on three axis, so it is used in mission that have restrictive inertial pointing requirements, which are typically performed by CubeSats made of 3U or more. The Y-momentum ADCS can be considered a middle ground between the two previously presented systems: it is

[1] https://www.cubespace.co.za/products/integrated-adcs/y-momentum/

[2] https://www.cubespace.co.za

used in missions where the spacecraft needs to be stabilized on three axes, but the pointing requirements are not extremely demanding, so it is perfect if the pointing requirement is higher than 1°. Contrarily to the Magnetic configuration, the Y-momentum wheel configuration, which can be seen in Fig. 3.3, allows to dissipate the rotation of the satellite after detumbling and to perform pitch maneuvers, but it is not able to perform precise maneuvers on the yaw and roll planes [32].

The MATLAB® Simulink® models for the sensors and actuators, here described qualitatively, and their parameters are presented in Appendix B.

3.2.1 Sensors

The attitude sensors are vital hardware components that feed the ADCS OBC with the environment vectors that are necessary to determine the attitude of the spacecraft. In order of increasing accuracy, the most commonly employed sensors in spacecrafts are coarse sun sensors, magnetometers, horizon sensors, fine sun sensors, and star trackers. In addition to these, gyroscopes are used to determine the rotational attitude of the satellite [26]. The sensors suite of CubeADCS Y-Momentum includes 3 MEMS Gyros, 10 Coarse Sun Sensors, one Tri-axial Magnetometer and one Fine Sun Sensor.

MEMS Gyros

Micro-Electro-Mechanical System (MEMS) gyros are the most commonly used devices for the measurement of the angular rate of CubeSats [33], due to their low cost, low mass, low power requirement and their inherent immunity from shock and vibration [34].

A MEMS gyro can be modeled as a two-fold orthogonal spring-mass-damper with stiffness k and resonant frequency ω_r [35]. The basic principle behind the functioning of MEMS gyros is that a vibrating object keeps on vibrating in the same plane even if its support rotates. So the proof mass has to be continuously kept in a vibrating mode. In the rotational reference frame of the support, the proof mass undergoes a Coriolis acceleration out of the plane

$$\mathbf{a_c} = 2\mathbf{v} \times \mathbf{\Omega} \tag{3.3}$$

where $\mathbf{\Omega}$ is the rotational rate of the support, which has to be measured and $\mathbf{v}$ is the velocity of the vibration of the mass. The vibrating velocity can be expressed a follows:

$$v(t) = X_i \omega_r \cos(\omega_r t) \tag{3.4}$$

where $X_i \sin(\omega_r t)$ is the in-plane position of the mass, under the hypothesis of a sinusoidal vibration.

Given that the suspensions of the mass provide elastic stiffness k, this acceleration is proportional to the displacement out of the plane:

$$\mathbf{x} = \frac{m\,\mathbf{a_c}}{k} \tag{3.5}$$

Combining Eq. (3.3), Eq. (3.4) and Eq. (3.5), the angular rate on the axis of the gyro can be calculated knowing the physical properties of the gyro and the out of plane displacement [36]:

$$\Omega = \frac{kx}{2mX_i\omega_r \cos(\omega_r t)} \tag{3.6}$$

Any gyroscope presents two sources of noise: the Rate Random Walk (RRW), which has the units of a rate [rad/s] and the Angular Random Walk (ARW) which appears with the unit [rad/s$^{0.5}$] [37].

Coarse Sun Sensors

Sun sensors provide an estimate of the position of the Sun with respect to the spacecraft's Body frame [33]. The coarse Sun sensors of CubeADCS are simple photo-diodes mounted on a small PCB, which has to be epoxied to the outer surface of the satellite. The configuration proposed by CubeSpace comprises 10 of these sensors, 5 of which are redundant, because it is sufficient to mount one on each face of the satellite, except the -z face, where the fine Sun sensor is. The basic idea behind the functioning of these Sun Sensors is that the electrical current produced on the photodiode is directly proportional to the cosine of the angle of incidence of the sun rays on its surface, so the angle can be easily deduced from the current measurement:

$$I = I_{max} \cos(\theta + \nu) \tag{3.7}$$

where the term ν represents the uncertainty of the measured angle in the form of a Gaussian white noise. These sensors are used to give an initial rough estimate of the satellite attitude, so they are quite inaccurate, as the name suggests: their measurement accuracy, which is the standard deviation of ν, is approximately 10° [38].

CubeMag

CubeMag is the tri-axial magnetometer provided by CubeSpace. Magnetometers are used to give an estimation of the local magnetic field in the body frame. This can be used to estimate the attitude of the satellite [39] and even to give an estimate of the position of the CubeSat [40]. The spacecraft's own circuits and electrical devices inherently produce magnetic fields that can be a great disturbance on the magnetometer's measurement [26], so a very important feature that makes the measurements from the magnetometer much more reliable is the ability to deploy outside of the CubeSat's structure. For a 2U CubeSat like ORCASat, a safe distance from the spacecraft center is around 50 mm, which is approximately the length of the deploying arm of CubeMag.

CubeSense V3

CubeSense V3 is a small camera operating in the range of the visible radiation which can be configured both as a fine horizon sensor and as a fine Sun sensor. If it is used as a Sun Sensor, as is the case for ORCASat, a filter is placed in front of the camera to allow only intense light to be detected. It is the sensor that requires the most energy (100 mW in nominal mode), but it can provide a measurement for the Sun vector with an accuracy of 0.2°, thanks to its very wide detection FOV of 180°. The main limitation of this sensor is obviously that it can only see than half of the space around the satellite. This is why the Coarse Sun sensors are very important in the side of the satellite where CubeSense is not present: if the spacecraft is oriented with this side towards the Sun the measurement of the Sun vector is much less accurate, but it can still be used to estimate the attitude [41, 38]. Nonetheless, it was decided by the team to mount CubeSense V3 on the Zenith face of the satellite, which is the one that is expected to be illuminated for most of the time.

3.2.2 Actuators

Actuators are the hardware devices that produce a torque, which has to be as close as possible to the one commanded by the controller at each instant in time, to make the satellite actively achieve, or maintain, the desired attitude [26]. The most commonly used actuators on spacecrafts are magnetorquers, reaction/momentum wheels and thrusters. These last ones are used only on medium size or large satellites, rarely on CubeSats smaller than 6U. The configuration proposed in CubeADCS includes three magnetorquers and a momentum wheel.

CubeRod and CubeCoil

Magnetorquers are the most commonly used actuators in small CubeSat because of their simplicity, low cost and effectiveness in LEO orbits, where most of these nanosatellites operate. They can also be used to desaturate the reaction wheels [23].

A magnetorquer is basically a solenoid, characterized by N turns of wire in a loop of area A and a direction $\hat{\mathbf{n}}$ perpendicular to the surface A. According to Ampere's law, when electric current flows inside the wire a magnetic dipole moment $\mathbf{m}$ parallel to $\hat{\mathbf{n}}$ and proportional to the current intensity, the area and the number of turns is produced

$$\mathbf{m} = NIA\hat{\mathbf{n}} \tag{3.8}$$

Denominating the local magnetic field of the planet $\mathbf{b}$, the torque that the magnetorquer imposes on the spacecraft is:

$$\boldsymbol{\tau} = \mathbf{m} \times \mathbf{b} \tag{3.9}$$

So, by modulating the electric current intensity flowing inside each of the three actuators it is possible to produce the desired torque in terms of intensity and direction.

To compute the power required by each magnetorquer, the voltage supply V of the actuator has to be known, so that the calculation is simply performed using Ohm's law

$$P = VI \tag{3.10}$$

The main limitation of magnetic actuation is that it is not auto-consistent: it is impossible to produce a torque parallel to the direction of the local magnetic field, hence to achieve instantaneous full control at least another type of actuator is needed. Nonetheless many studies point out how tri-axial control can be achieved using magnetic actuators in an averaged sense, because the change of the magnetic field in the Body frame over the course of an orbit allows the satellite to apply torques in any desired direction [42, 43, 44].

CubeADCS Y-Momentum presents three magnetorquers, two of which are called CubeRods, whereas the last one is called CubeCoil. Due to volumetric limitations it is impossible to have three equal actuators, like three rods, so one actuator is a coil, which allows to achieve a lower maximum magnetic dipole moment than the rods. In fact, the rods reach saturation at 0.48 A m^2, whereas the coil at 0.13 A m^2.

CubeWheel

CubeWheel can be used as a momentum wheel or as a reaction wheel. The basic difference between these two operating modes is that in the first case it is used to contrast the environmental disturbance torques, whereas in the second it is used to perform an attitude maneuver. If the wheel is used as a momentum wheel it is typically spun up to the maximum speed to provide gyroscopic stiffness to the satellite, via the increment of the total angular momentum. In fact, increasing the angular momentum makes the satellite less sensitive to external torques that are perpendicular to the direction of the rotation. This robustness towards perturbing torques is a fundamental requirement for a good pointing accuracy. The maximum momentum storage that can be achieved by the small CubeWheel present in CubeADCS is 1.7 mN m s. The operational mode of a momentum wheel is called speed mode, as opposed to the one the reaction wheel, namely the torque mode.

The wheel electric circuit can be modeled as the MISO closed loop system of Fig. 3.4, where the inputs are the commanded angular rate ω_c and the angular rate of the satellite ω_s and the output is the angular rate of the wheel ω_w. Typically small wheels are more subject than bigger ones to the influence of friction, which tends to slow them down. Hence, to keep the angular rate constant a significant power has to be fed to the CubeWheel. A momentum wheel undergoes the deceleration caused by two types of dissipative actions: the Coulomb friction and the viscous friction. The former is the constant torque that is needed to put the system in motion, while the latter is a deceleration proportional to the angular rate of the system [45]. The objective of the momentum wheel is to have a commanded torque τ_c as small as possible, in order to keep the angular rate constant, but the friction terms cause the rate to decrease, thus an accelerating command is always needed. Bearing this in mind, the dynamics of the momentum wheel can be model as a function of its moment of inertia J_w and the two friction parameters $c > 0$ and $B > 0$, which respectively model the Coulomb and the viscous frictions

$$J_w \dot{\omega}_w = \tau_w - B\omega_{rel} - c\,\mathrm{sgn}(\omega_w) - \tau_e \tag{3.11}$$

where $\omega_{rel} = \omega_w - \omega_s$ is the angular rate of the wheel relative to the pitch rate of the satellite, τ_w is the torque that it exerts and τ_e is a white Gaussian noise that models the disturbances caused by the eccentricity of the wheel.

The commanded angular velocity is compared to the one that has been reached by the wheel and

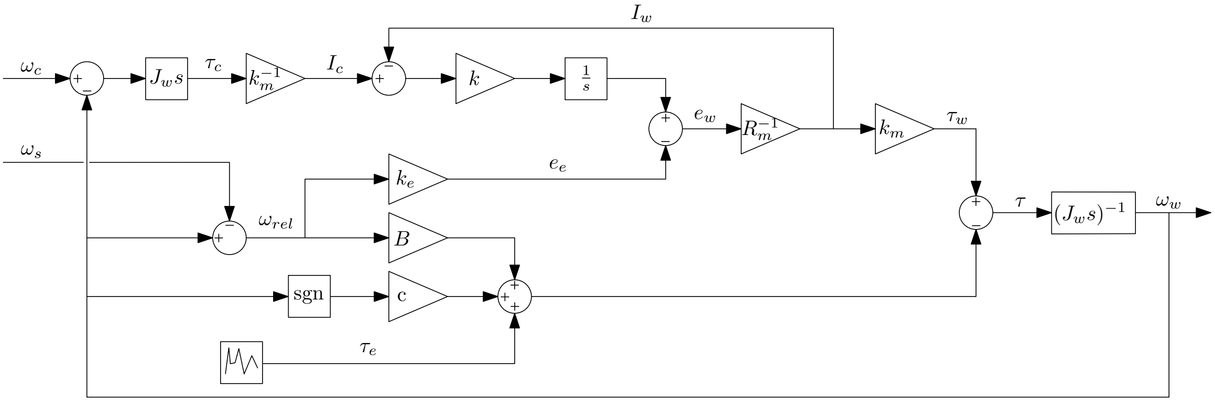


Figure 3.4: Momentum wheel electric and mechanical model (for the complete model see Appendix B).

then transformed into a commanded torque τ_c via the derivative operation and the multiplication by the wheel moment of inertia. The commanded torque is then transformed into an electric current command I_c, which is compared to the feedback value of the actual current that is circulating in the wheel I_w. The entity of the difference, multiplied by the retroactive constant k and integrated, returns the value of the executed voltage e_w. An important source of disturbance to consider in this node of the wheel electric system is the counter-electromotive force, which reduces the voltage commanded by the controller

$$e_e = k_e \omega_{rel} \tag{3.12}$$

where k_e has the same numerical value as k_m.

The torque that would be produced in the absence of dissipative forces is related to the current I_w that is fed to the wheel by the torque constant k_m

$$\tau_w = k_m I_w \tag{3.13}$$

The total torque, which is obtained according to Eq. (3.11), is at last integrated and divided by the moment of inertia to obtain the output value of the wheel angular velocity.

In the literature many empiric studies are found about methods of estimation of the wheel parameters (c, B, k_m, k, k_e), when it is possible to perform physical tests on the wheel [46, 45, 47], and many more on numerical simulations, when the viscous parameters are known [48, 49, 50, 51]. In this work, since the physical system was not available to perform tests with, the viscous coefficients were estimated based on the data-sheet provided by CubeSpace [38].

If R_m is the electric resistance of the wheel, the power needed by the system can be computed as

$$P = R_m I_w^2 \tag{3.14}$$

3.2.3 CubeComputer and Software

The CubeComputer is a low power OBC that functions as a control unit for the CubeADCS. It is able to run estimator and control algorithms and manage communications between the ADCS modules and the main OBC of the spacecraft via I2C, CAN and UART interfaces.

Two software programs are installed in the CubeComputer: a bootloader and an Attitude Control Program (ACP). The bootloader performs several important tasks such as providing status telemetry, enabling watchdogs to prevent faulty or damaged applications from halting the CPU and allowing reprogramming of applications via UART or I2C. The ACP is the main application launched by the bootloader which allows the CubeComputer to interface with every subsystem of the CubeADCS. It also coordinates the sampling and calibration of sensors, the execution of estimation modes and the transmission of control commands to the actuators [38]. It comprises several standard estimation modes and control modes, that are here listed and will be explained in Chapter 4 and Chapter 5. The standard estimation modes are:

- Gyro rate measurement: provides the measurement of the angular rates as read by the gyros.
- Magnetic Rate Extended Kalman Filter (MREKF): uses a filtering process to yield the angular rates starting from the measurements of the magnetometer.
- TRIaxial Attitude Determination (TRIAD): employs an algebraic method to provide an estimation of the quaternion attitude using two sensor measurements.
- Gyro (or Multiplicative) EKF: Uses an Extended Kalman Filter to compute the attitude quaternion and the gyro bias from the two sensor measurements and the gyro measurements.
- Full-state (or Additive) EKF: Uses an extended Kalman Filter to compute the attitude quaternion and the angular rates from the two sensor measurements.

The standard control modes are:

- X-Z Detumbling
- Y-Thomson Detumbling
- Fast Detumbling
- Very fast Detumbling
- Y-momentum

Chapter 4

Attitude Determination

The attitude determination modes that are implemented in the CubeSpace OBC are exposed in this section. The first algorithm barely uses the gyroscopes sensed measurements as an estimate for the angular rate and no information is given on the attitude. For the other four modes a more detailed discussion is necessary.

4.1 TRIAD

TRIaxial Attitude Determination (TRIAD) is a simple determination algorithm invented by Harold D. Black [52]. It makes use of only two instantaneous vector measurements and it does not require the knowledge of the history of the attitude of the spacecraft. Thanks to its elegance it has been the object of many studies and it can be found in many different varieties that offer different deterministic constructions for the attitude matrix [53, 54, 55]. In this section only the standard TRIAD will be presented. The starting point for the algorithm is the measurement of two vectors in the Body frame. Since TRIAD is a deterministic approach to attitude determination, the Body vectors expressed in the Body frame have to be coupled with the vectors as expressed in a second known reference frame, such as the ECI frame, at the same time t. The Body vectors will be called $\mathbf{b}_1$ and $\mathbf{b}_2$, while the reference vectors are $\mathbf{r}_1$ and $\mathbf{r}_2$. Typically the two vectors used are the Sun vector and the vector of the local magnetic field, as is the case for ORCASat. The final aim is to find an attitude matrix such that

$$\mathbf{A}\mathbf{r}_1 = \mathbf{b}_1 \tag{4.1a}$$

$$\mathbf{A}\mathbf{r}_2 = \mathbf{b}_2 \tag{4.1b}$$

It is impossible to satisfy at the same time both Eq. (4.1a) and Eq. (4.1b), because in reality the Body vectors are measurements which are affected by errors, so that, if the sensor measurements are used, Eq. (4.1) becomes

$$\mathbf{A}_1\mathbf{r}_1 = \tilde{\mathbf{b}}_1 = \mathbf{b}_1 + \boldsymbol{\nu}_1 \tag{4.2a}$$

$$\mathbf{A}_2\mathbf{r}_2 = \tilde{\mathbf{b}}_2 = \mathbf{b}_2 + \boldsymbol{\nu}_2 \tag{4.2b}$$

where $\boldsymbol{\nu}_1$ and $\boldsymbol{\nu}_2$ are the Gaussian white noises of the two measurements and $\mathbf{A}_1 \neq \mathbf{A}_2$.

The idea behind this algorithm is that given an orthonormal TRIAD of unit vectors $\{\hat{\mathbf{v}}_1, \hat{\mathbf{v}}_2, \hat{\mathbf{v}}_3\}$ in the reference frame and the corresponding TRIAD expressed in the Body frame $\{\hat{\mathbf{u}}_1, \hat{\mathbf{u}}_2, \hat{\mathbf{u}}_3\}$, the attitude matrix is

$$\mathbf{A} = \begin{bmatrix}\hat{\mathbf{u}}_1 & \hat{\mathbf{u}}_2 & \hat{\mathbf{u}}_3\end{bmatrix} \begin{bmatrix}\hat{\mathbf{v}}_1 & \hat{\mathbf{v}}_2 & \hat{\mathbf{v}}_3\end{bmatrix}^{\mathrm{T}} \tag{4.3}$$

The algorithm forms the first TRIAD from $\mathbf{r}_1$ and $\mathbf{r}_2$ and the second one from $\mathbf{b}_1$ and $\mathbf{b}_2$. This is possible only if the two vector are not parallel, situation in which the singularity of the solution would not allow for the problem to be solved. The two triads are simply built as follows

$$\hat{\mathbf{u}}_1 = \frac{\mathbf{b}_1}{||\mathbf{b}_1||} \tag{4.4a}$$

$$\hat{\mathbf{u}}_2 = \mathbf{b}_\times = \frac{\mathbf{b}_1 \times \mathbf{b}_2}{||\mathbf{b}_1 \times \mathbf{b}_2||} \tag{4.4b}$$

$$\hat{\mathbf{u}}_3 = \hat{\mathbf{u}}_1 \times \hat{\mathbf{u}}_2 \tag{4.4c}$$

The exact same procedure is used for the reference vectors. It is clear that in Eq. (4.4a) all the information included in the $\tilde{\mathbf{b}}_1$ measurement is preserved, whereas in Eq. (4.4b) only two components of $\tilde{\mathbf{b}}_2$ are used, since $\hat{\mathbf{u}}_2$ can be the result of the cross product of $\tilde{\mathbf{b}}_1$ with an infinity of vectors that lay on the plane parallel to $\tilde{\mathbf{b}}_2$ and orthogonal to $\tilde{\mathbf{b}}_1$. Since one of the measured unit vectors is usually more accurate than the other ($\nu_1 < \nu_2$), it is convenient to compute the matrix $\mathbf{A}$ such that Eq. (4.2a) is exactly satisfied and Eq. (4.2b) only approximately. In this way the measured scalar component that is not utilized is the third component of the less accurate vector. Finally the estimated attitude matrix is obtained substituting Eq. (4.4) into Eq. (4.3) and using the sensed vectors defined in Eq. (4.2)

$$\mathbf{A}_{est} = \tilde{\mathbf{b}}_1\mathbf{r}_1^{\mathrm{T}} + (\tilde{\mathbf{b}}_1 \times \tilde{\mathbf{b}}_\times)(\mathbf{r}_1 \times \mathbf{r}_\times)^{\mathrm{T}} + \tilde{\mathbf{b}}_\times\mathbf{r}_\times^{\mathrm{T}} \tag{4.5}$$

4.2 Extended Kalman Filters

In many situations static attitude determination techniques like TRIAD cannot provide an accurate solution to the determination problem, i.e. when the spacecraft is in eclipse, or when a singularity is present in the problem. In this regard, stochastic (or recursive) algorithms can be more reliable, because they combine the measurements coming from the sensors with a propagation step derived from the spacecraft kinematics [56]. For systems that are described with sufficient accuracy by a linear model, the Kalman Filter is a very useful tool of recursive estimation [57]. In the case of a more complex mechanics, described by non-linear differential equations, the Extended Kalman Filter (EKF) is one of the most commonly used filtering methods because of its simplicity and effectiveness [58]. In Fig. 4.1 a qualitative indication of the operation of the EKF is given: the propagation using the kinematic equation is corrected at each time step by a measurement update, characterized by a covariance matrix which models the uncertainties of the measurements. In this chapter, the subscript in x_k represents the time step in the propagation, whereas the superscript in x^- (x^+) indicates if the variable is estimated before or after the measurement update. The hat symbol $\hat{x}$ stands for the estimation of a variable, as opposed to the measured variable described by the tilde $\tilde{x}$ and the true value x^{true}.

A general nonlinear system can be described by the following differential equation

$$\dot{\mathbf{x}}^{\text{true}} = \mathbf{f}(\mathbf{x}^{\text{true}}, \mathbf{u}, \mathbf{w}, t) \tag{4.6}$$

where $\mathbf{x}$ is the physical state $n \times 1$ vector (typically position and its derivatives), $\mathbf{u}$ is the $m \times 1$ vector of the known input and $\mathbf{w}$ is a zero-mean Gaussian white-noise process vector with spectral density matrix $\mathbf{Q}$. Also the measurement model has to be considered:

$$\tilde{\mathbf{y}}_k = \mathbf{h}(\mathbf{x}_k^{\text{true}}) + \mathbf{v}_k \tag{4.7}$$

where $\mathbf{y}_k$ is the measured vector at the time step k, $\mathbf{h}(\cdot)$ is the measurement function and $\mathbf{v}_k$ is the Gaussian white noise measurement vector with covariance matrix $\mathbf{R}$.

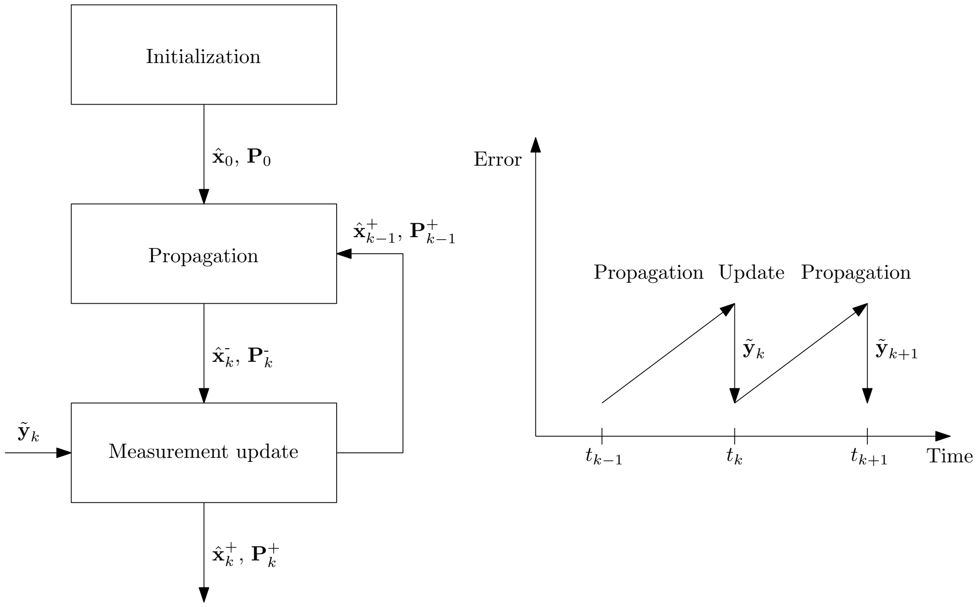

Initialization
$\hat{\mathbf{x}}_0$, $\mathbf{P}_0$
Propagation
$\hat{\mathbf{x}}_{k-1}^+$, $\mathbf{P}_{k-1}^+$
$\hat{\mathbf{x}}_k^-$, $\mathbf{P}_k^-$
$\tilde{\mathbf{y}}_k$
Measurement update
$\hat{\mathbf{x}}_k^+$, $\mathbf{P}_k^+$
Error
Propagation
Update
Propagation
$\tilde{\mathbf{y}}_k$
$\tilde{\mathbf{y}}_{k+1}$
t_{k-1}
t_k
t_{k+1}
Time

Figure 4.1: EKF formulation (adapted from [16]).

The filter propagates two variables: the state vector and the covariance matrix, that measures the uncertainty of the estimation. In the propagation stage Eq. (4.6) is discretized making use of a finite differences method and the state is propagated from the time step $k-1$ to k.

$$\hat{\mathbf{x}}_k^- = \mathbf{F}(\hat{\mathbf{x}}_{k-1}^+)\hat{\mathbf{x}}_{k-1}^+ + \mathbf{B}\mathbf{u}_{k-1} \tag{4.8}$$

where $\mathbf{F}$ is the $n \times n$ error-state transition matrix and $\mathbf{B}$ is the matrix that linearizes the input vector

$$\mathbf{F} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}\bigg|_{\hat{\mathbf{x}}} \tag{4.9a}$$

$$\mathbf{B} = \frac{\partial \mathbf{f}}{\partial \mathbf{u}}\bigg|_{\hat{\mathbf{x}}} \tag{4.9b}$$

A fundamental hypothesis of this filter is that the true state is sufficiently close to the estimated one, so that error dynamics can be represented quite accurately by a first order Taylor expansion

$$\mathbf{G} = \frac{\partial \mathbf{f}}{\partial \mathbf{w}}\bigg|_{\hat{\mathbf{x}}} \tag{4.10}$$

In this way, the noise term of Eq. (4.6) can be ignored in the propagation of the state and included only in the propagation of the covariance matrix. The covariance matrix is propagated using the discretized Riccati equation

$$\mathbf{P}_k^- = \mathbf{F}_{k-1}^+ \mathbf{P}_{k-1}^+ (\mathbf{F}_{k-1}^+)^{\mathrm{T}} + \mathbf{G}\mathbf{Q}\mathbf{G}^{\mathrm{T}} \tag{4.11}$$

In the measurement update step, the measurement sensitivity matrix is defined as

$$\mathbf{H} = \frac{\partial \mathbf{h}}{\partial \mathbf{x}}\bigg|_{\hat{\mathbf{x}}} \tag{4.12}$$

Using this information the Kalman Gain matrix can be computed

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^{\mathrm{T}} \big(\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^{\mathrm{T}} + \mathbf{R}\big)^{-1} \tag{4.13}$$

The estimated state vector is updated with the following equation:

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \mathbf{K}_k \big(\tilde{\mathbf{y}}_k - \mathbf{H}_k \hat{\mathbf{x}}_k^-\big) \tag{4.14}$$

And the covariance matrix is updated as well

$$\mathbf{P}_k^+ = \big(\mathbf{I}_n - \mathbf{K}_k \mathbf{H}_k\big)\mathbf{P}_k^- \tag{4.15}$$

4.2.1 Magnetometer Rate Extendend Kalman Filter

In the situation in which the MEMS gyro measurement is not available, it is useful to be able to determine with a certain degree of accuracy the angular rates using only the measurement from the magnetometer. Many studies present a way to utilize the EKF formulation to filter the magnetic field measurement and use it in this regard [59, 60, 61, 62, 63]. The filtering process of the Magnetic Rate Extended Kalman Filter (MREKF) only yields valid results if the angular rate of the spacecraft is below 35°/s [32], because it relies on the finite difference derivative of the sensed magnetic field, that can be very inaccurate at high rate.

Referring to Fig. 4.1, the state to be estimated is a three-dimensional vector equal to the angular velocity of the spacecraft expressed in the body reference frame (where $\boldsymbol{\omega} = \boldsymbol{\omega}_B^{BI}$ for brevity)

$$\mathbf{x}(t) = \boldsymbol{\omega}(t) \tag{4.16}$$

Eq. (4.6) is translated into the equation of motion of the spacecraft:

$$\mathbf{J}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times (\mathbf{J}\boldsymbol{\omega} + \mathbf{h}) = \boldsymbol{\tau} - \dot{\mathbf{h}} \tag{4.17}$$

where $\mathbf{J}$ is the tensor of inertia of the satellite, $\mathbf{h}$ is the angular momentum of the reaction wheel and $\boldsymbol{\tau}$ is the total external torque (disturbances and control) acting on the spacecraft. So, in terms of the state vector and the input vector, Eq. (4.17) can be written as

$$\dot{\mathbf{x}}(t) = \mathbf{F}(t)\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \tag{4.18}$$

The sensor model from Eq. (4.7) is based on the equation

$$\frac{\mathrm{d}\mathbf{b}_I}{\mathrm{d}t} = \frac{\mathrm{d}\mathbf{b}_B}{\mathrm{d}t} + \boldsymbol{\omega} \times \mathbf{b}_B \tag{4.19}$$

where $\mathbf{b}_I$ and $\mathbf{b}_B$ respectively stand for the local magnetic field vector in the ECI and in the Body frames. It is assumed that the magnetic field vector does not vary with respect to the inertial frame during the sampling interval, thus $\mathrm{d}\mathbf{b}_I/\mathrm{d}t = 0$. So Eq. (4.19) becomes

$$\frac{\mathrm{d}\mathbf{b}_B}{\mathrm{d}t} = \mathbf{b}_B \times \boldsymbol{\omega} \tag{4.20}$$

The measured magnetic field vector is

$$\tilde{\mathbf{b}}(t) = \mathbf{b}_B(t) + \mathbf{v}(t) \tag{4.21}$$

where $\mathbf{v}(t)$ is a three-dimensional noise process vector with standard deviation σ and covariance matrix $\mathbf{R} = \sigma^2\mathbf{I}_3$.

Propagation

The kinematic equation in Eq. (4.18) is discretized as expressed in Eq. (4.8)

$$\hat{\mathbf{x}}_k^- = \mathbf{F}(\mathbf{x}_{k-1}^+)\mathbf{x}_{k-1}^+ + \mathbf{B}\mathbf{u}_{k-1} \tag{4.22}$$

where $\mathbf{F}(\mathbf{x}_{k-1}^+)$ is the 3×3 state-error transition matrix and $\mathbf{B}\mathbf{u}_{k-1}$ is the known term; they are given as follows

$$\mathbf{F}_{k-1} = \mathbf{I}_3\Delta t \begin{bmatrix} 0 & \frac{0.5(\omega_z)_{k-1}^+(I_y-I_z)-(h_z)_{k-1}^+}{I_x} & \frac{0.5(\omega_y)_k^+(I_y-I_z)+(h_y)_{k-1}^+-H_0}{I_x} \\ \frac{0.5(\omega_z)_{k-1}^+(I_z-I_x)+(h_z)_{k-1}^+}{I_y} & 0 & \frac{0.5(\omega_x)_k^+(I_z-I_x)-(h_x)_{k-1}^+}{I_y} \\ \frac{0.5(\omega_y)_{k-1}^+(I_x-I_y)-(h_y)_{k-1}^++H_0}{I_z} & \frac{0.5(\omega_x)_k^+(I_x-I_y)+(h_x)_{k-1}^+}{I_z} & 0 \end{bmatrix} \tag{4.23a}$$

$$\mathbf{B}\mathbf{u}_{k-1} = \mathbf{J}^{-1} \cdot (\boldsymbol{\tau}^{+}_{k-1} - \dot{\mathbf{h}}^{+}_{k-1})\Delta t \tag{4.23b}$$

where H_0 is the constant angular momentum given by the orbital motion around the Earth on the negative pitch axis [64]. It is not necessary to include the three components of the angular momentum of the spacecraft in the state vector because they can simply be obtained multiplying the constant matrix of inertia and the angular rate vector ($\mathbf{h} = \mathbf{J}\boldsymbol{\omega}$). The covariance matrix propagation is carried out as indicated in Eq. (4.11), with $\mathbf{G} = \mathbf{I}_3$ and $\mathbf{Q} = \sigma_p^2\mathbf{I}_3$, where σ_p is the standard deviation of the process noise, which has to be tuned between 10^{-8} and 10^{-6}.

Measurement Update

The discrete measurement sensitivity matrix is defined as:

$$\mathbf{H}_k = [\tilde{\mathbf{b}}_k \times]\Delta t \tag{4.24}$$

The first derivative of the magnetic field measurement is estimated using a backward finite difference

$$\hat{\mathbf{y}}_k = \frac{\tilde{\mathbf{b}}_k - \tilde{\mathbf{b}}_{k-1}}{\Delta t} \tag{4.25}$$

Now the Kalman Gain is computed as suggested in Eq. (4.13) and the state estimate can be updated according to Eq. (4.14)

$$\hat{\boldsymbol{\omega}}^{+}_k = \hat{\boldsymbol{\omega}}^{-}_k + \mathbf{K}_k\big(\hat{\mathbf{y}}_k - \mathbf{H}_k\hat{\boldsymbol{\omega}}^{-}_k\big) \tag{4.26}$$

Also the covariance matrix is updated using Eq. (4.15).

4.2.2 Additive Extended Kalman Filter

As it was seen in Section 2.2.3, the quaternion is the most efficient way to estimate the attitude of the spacecraft, but its use in an EKF algorithm can be problematic, because in the standard formulation there is no constraint for the first element of the state variable $\mathbf{x}$ to be of unitary norm [65]. One of the ways to deal with this problem is to impose a brute force normalization on the updated quaternion. The material exposed in this section was collected from various sources [66, 67, 68, 69, 70]; the algorithm is alternatively called Gyroless EKF, Seven-state EKF, Full state EKF or Additive EKF (AEKF). The variables to be estimated are the quaternion $\mathbf{q}$ and the angular velocity $\boldsymbol{\omega}$. They are collected in a single vector $\mathbf{x}$

$$\mathbf{x}(t) = \begin{bmatrix} \mathbf{q}(t) \\ \boldsymbol{\omega}(t) \end{bmatrix} \tag{4.27}$$

where the quaternion $\mathbf{q} = \begin{bmatrix} \mathbf{q}^{\mathrm{T}}_{1:3} & q_4 \end{bmatrix}^{\mathrm{T}}$ obeys the normalization constraint $\mathbf{q}^{\mathrm{T}}\mathbf{q} = 1$. Eq. (4.6) consists of two parts: the kinematics equations expressed in terms of quaternion and angular velocity (from

Eq. (2.36)) and the dynamics equations

$$\dot{\mathbf{q}} = \frac{1}{2}\Xi(\mathbf{q})\boldsymbol{\omega} \tag{4.28a}$$

$$\mathbf{J}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times (\mathbf{J}\boldsymbol{\omega} + \mathbf{h}) = \boldsymbol{\tau} - \dot{\mathbf{h}} \tag{4.28b}$$

Similarly to what happens for a static method like TRIAD, a vector measured by a sensor in the body frame is compared with the same vector as expressed in the ECI frame to obtain an estimation of the attitude. If a reference vector, e.g. the Sun vector, is represented in the ECI and body frame respectively by $\mathbf{y}_I^{\text{true}}$ and $\mathbf{y}_B^{\text{true}}$, then the following relations hold

$$\tilde{\mathbf{y}}_k = \mathbf{A}(\hat{\mathbf{q}}_k)\mathbf{y}_I^{\text{true}} + \mathbf{e} = \mathbf{h}(\hat{\mathbf{x}}_k) + \mathbf{v} \tag{4.29}$$

where $\mathbf{v}$ is a vector of Gaussian white noise processes of standard deviation σ_s (equal for the three components) that model the error of the measurement with covariance matrix $\mathbf{R} = \sigma_s^2\mathbf{I}_3$.

Propagation

The discretized equation of motion can be written as in Eq. (4.8)

$$\hat{\mathbf{x}}_k^- = \mathbf{F}(\mathbf{x}_{k-1}^+)\mathbf{x}_{k-1}^+ + \mathbf{B}\mathbf{u}_{k-1} \tag{4.30}$$

The state error transition matrix is defined using the same process that was used for the MREKF

$$\mathbf{F}_{k-1} = \mathbf{I}_7 + \begin{bmatrix} \mathbf{F}_{11} & \mathbf{F}_{12} \\ \mathbf{F}_{21} & \mathbf{F}_{22} \end{bmatrix} \tag{4.31a}$$

$$\mathbf{F}_{11} = \frac{\Delta t}{4}\begin{bmatrix} 0 & (\omega_z)_{k-1}^+ & -(\omega_y)_{k-1}^+ & (\omega_x)_{k-1}^+ \\ -(\omega_z)_{k-1}^+ & 0 & (\omega_x)_{k-1}^+ & (\omega_y)_{k-1}^+ \\ (\omega_y)_{k-1}^+ & -(\omega_x)_{k-1}^+ & 0 & (\omega_z)_{k-1}^+ \\ -(\omega_x)_{k-1}^+ & -(\omega_y)_{k-1}^+ & -(\omega_z)_{k-1}^+ & 0 \end{bmatrix} \tag{4.31b}$$

$$\mathbf{F}_{12} = \frac{\Delta t}{4}\begin{bmatrix} (q_4)_{k-1}^+ & -(q_3)_{k-1}^+ & (q_2)_{k-1}^+ \\ (q_3)_{k-1}^+ & (q_4)_{k-1}^+ & -(q_1)_{k-1}^+ \\ -(q_2)_{k-1}^+ & (q_1)_{k-1}^+ & (q_4)_{k-1}^+ \\ -(q_1)_{k-1}^+ & -(q_2)_{k-1}^+ & -(q_3)_{k-1}^+ \end{bmatrix} \tag{4.31c}$$

$$\mathbf{F}_{21} = \mathbf{0}_{3\times 4} \tag{4.31d}$$

$$\mathbf{F}_{22} = \Delta t \begin{bmatrix} 0 & \frac{0.5(\omega_z)^+_{k-1}(I_y - I_z) - (h_z)^+_{k-1}}{I_x} & \frac{0.5(\omega_y)^+_{k-1}(I_y - I_z) + (h_y)^+_{k-1} - H_0}{I_x} \\ \frac{0.5(\omega_z)^+_{k-1}(I_z - I_x) + (h_z)^+_{k-1}}{I_y} & 0 & \frac{0.5(\omega_x)^+_{k-1}(I_z - I_x) - (h_x)^+_{k-1}}{I_y} \\ \frac{0.5(\omega_y)^+_{k-1}(I_x - I_y) - (h_y)^+_{k-1} + H_0}{I_z} & \frac{0.5(\omega_x)^+_{k-1}(I_x - I_y) + (h_x)^+_{k-1}}{I_z} & 0 \end{bmatrix} \tag{4.31e}$$

as in Section 4.2.1, there is no need to include the angular momentum of the spacecraft $\mathbf{h}$ in the state vector. The vector of the known term has to be included in the equation

$$\mathbf{Bu}_{k-1} = \begin{bmatrix} \mathbf{0}_{4\times 1} \\ J^{-1}(\boldsymbol{\tau}^+_{k-1} - \dot{\mathbf{h}}^+_{k-1})\Delta t \end{bmatrix} \tag{4.32}$$

Now the propagation equation can be written in the same way as it was done in the Section 4.2.1

$$\hat{\mathbf{x}}^-_k = \mathbf{F}(\hat{\boldsymbol{\omega}}^+_{k-1})\hat{\mathbf{x}}^+_{k-1} + \mathbf{Bu}_{k-1} \tag{4.33}$$

Since AEKF needs to propagate the full equation of motion, if an accurate result is wanted the algorithm has to be aided by a very robust set of environmental models and a good estimation for the satellite physical parameters. If this is not the case, the known term of Eq. (4.32) might substantially differ from the real torques acting on the spacecraft, thus making the estimation of the attitude very inaccurate.

Measurement Update

According to the definition of the measurement matrix as expressed in Eq. (4.12), the Jacobian of the vector equation that transforms the reference vector from the ECI frame to the Body frame has to be computed with respect to the quaternion elements. The equation to be derived with respect to the quaternion elements is

$$\hat{\mathbf{y}}_b = \mathbf{A}(\hat{\mathbf{q}}^-_k)\mathbf{y}_I \tag{4.34}$$

where the attitude matrix $\mathbf{A}$ is expressed in terms of the quaternion according to Eq. (2.17). So the sensitivity matrix is defined as follows

$$\mathbf{H}_k = \begin{bmatrix} \mathbf{H}_1 & \mathbf{H}_2 & \mathbf{H}_3 & \mathbf{H}_4 & \mathbf{H}_5 \end{bmatrix} \tag{4.35a}$$

$$\mathbf{H}_1 = \frac{\partial \hat{\mathbf{y}}_b}{\partial q_1}\bigg|_{\hat{\mathbf{x}}} = 2 \begin{bmatrix} (q_1)^-_k & (q_2)^-_k & (q_3)^-_k \\ (q_2)^-_k & -(q_1)^-_k & (q_4)^-_k \\ (q_3)^-_k & -(q_4)^-_k & -(q_1)^-_k \end{bmatrix} \mathbf{y}_I \tag{4.35b}$$

$$\mathbf{H}_2 = \frac{\partial \hat{\mathbf{y}}_b}{\partial q_2}\bigg|_{\hat{\mathbf{x}}} = 2 \begin{bmatrix} -(q_2)^-_k & (q_1)^-_k & -(q_4)^-_k \\ (q_1)^-_k & (q_2)^-_k & (q_3)^-_k \\ (q_4)^-_k & (q_3)^-_k & -(q_2)^-_k \end{bmatrix} \mathbf{y}_I \tag{4.35c}$$

$$\mathbf{H}_3 = \frac{\partial \hat{\mathbf{y}}_b}{\partial q_3}\bigg|_{\hat{\mathbf{x}}} = 2 \begin{bmatrix} -(q_3)_k^- & (q_4)_k^- & (q_1)_k^- \\ -(q_4)_k^- & -(q_3)_k^- & (q_2)_k^- \\ (q_1)_k^- & (q_2)_k^- & (q_3)_k^- \end{bmatrix} \mathbf{y}_I \tag{4.35d}$$

$$\mathbf{H}_4 = \frac{\partial \hat{\mathbf{y}}_b}{\partial q_4}\bigg|_{\hat{\mathbf{x}}} = 2 \begin{bmatrix} (q_4)_k^- & (q_3)_k^- & -(q_2)_k^- \\ -(q_3)_k^- & (q_4)_k^- & (q_1)_k^- \\ (q_2)_k^- & -(q_1)_k^- & (q_4)_k^- \end{bmatrix} \mathbf{y}_I \tag{4.35e}$$

$$\mathbf{H}_5 = \frac{\partial \hat{\mathbf{y}}_b}{\partial \boldsymbol{\omega}}\bigg|_{\hat{\mathbf{x}}} = \mathbf{0}_{3\times 3} \tag{4.35f}$$

The Kalman Gain is computed at this point using Eq. (4.13). The covariance matrix is updated according to Eq. (4.15) and the update term is calculated using the Kalman Gain

$$\Delta \hat{\mathbf{x}}_k^+ = \mathbf{K}_k(\tilde{\mathbf{y}}_b - \hat{\mathbf{y}}_b) \tag{4.36a}$$

$$\Delta \hat{\mathbf{x}}_k^+ = \begin{bmatrix} \Delta \hat{\mathbf{q}}_k^+ \\ \Delta \hat{\boldsymbol{\omega}}_k^+ \end{bmatrix} \tag{4.36b}$$

At this point a brute force normalization becomes necessary to ensure that the quaternion does not exceed the unit norm, so the update has to be done separately for the quaternion and the angular rate

$$\hat{\mathbf{q}}_k' = \hat{\mathbf{q}}_k^- + \Delta \hat{\mathbf{q}}_k^+ \tag{4.37a}$$

$$\hat{\mathbf{q}}_k^+ = \frac{\hat{\mathbf{q}}_k'}{||\hat{\mathbf{q}}_k'||} \tag{4.37b}$$

$$\hat{\boldsymbol{\omega}}_k^+ = \hat{\boldsymbol{\omega}}_k^- + \Delta \hat{\boldsymbol{\omega}}_k^+ \tag{4.37c}$$

From this point on, the $k+1$ step takes place in the propagation and the process repeats itself.

4.2.3 Multiplicative Extended Kalman Filter

Another way to overcome the problem of the quaternion normalization in the EKF is the use of the multiplicative approach in the computation of the quaternion error. The Multiplicative Extended Kalman Filter (MEKF) is one of the most commonly used estimation algorithms in CubeSats, thanks to its efficiency and precision [65, 71, 72, 73, 74]. In this work the version that was developed in Rondão's Master's Thesis [75] is employed. It takes advantage of the measurements of the attitude sensors (Sun sensors and magnetometer) combined with the rate sensor (MEMS gyroscopes) which is used in the state propagation step.

The formulation of the EKF is advantageous because it involves the estimation of only 6 variables, instead of 7, making the algorithm faster, and it allows to estimate both the attitude quaternion and the dynamic bias of the gyroscopes measurements, which is needed to obtain a better estimate of the angular velocity of the spacecraft.

As in the AEKF (Eq. (4.27)), the state vector is 7×1 vector made of a quaternion and an angular velocity part. The true quaternion can be written as the product of the error quaternion and the estimate according to Eq. (2.15) [16]:

$$\mathbf{q}^{\text{true}} = \delta\mathbf{q}(\delta\boldsymbol{\theta}) \otimes \hat{\mathbf{q}} \tag{4.38}$$

where $\delta\boldsymbol{\theta}$ is a three component state vector for the local representation of the error, which can be expressed as two times the vector part of the quaternion, so that

$$\delta\mathbf{q} = \begin{bmatrix} \delta\boldsymbol{\theta}/2 \\ 1 \end{bmatrix} \tag{4.39}$$

This last equation is only valid when the attitude error $\delta\boldsymbol{\theta}$ is small, because the relation between the quaternion error and the local attitude error is only a first order approximation.

A propagation equation is needed also for the angular velocity of the spacecraft. To this scope, a model of the noise of the angular rate measurement and its bias is needed: the true angular velocity can be written as

$$\boldsymbol{\omega}^{\text{true}} = \tilde{\boldsymbol{\omega}} - \boldsymbol{\beta}^{\text{true}} + \boldsymbol{\eta}_v \tag{4.40a}$$

$$\dot{\boldsymbol{\beta}}^{\text{true}} = \boldsymbol{\eta}_u \tag{4.40b}$$

where $\boldsymbol{\eta}_v$ and $\boldsymbol{\eta}_u$ are two zero mean white Gaussian noise processes with spectral densities $\sigma_v^2\mathbf{I}_3$ and $\sigma_u^2\mathbf{I}_3$ and they model the noise of the measurement and of its bias respectively. Since the parameter used in the estimation is the gyro bias vector $\boldsymbol{\beta}$, which only depends on the measurement and has no explicit dependence on time, the differential equation that governs its behaviour is

$$\dot{\boldsymbol{\beta}} = \mathbf{0}_3 \tag{4.41}$$

The observation model is the same as expressed in Eq. (4.29), because the observation process is the same in the MEKF and in the AEKF.

Propagation

The propagation step of the MEKF algorithm is derived from the kinematic equation, which is valid both for the true states and for the estimated ones

$$\dot{\mathbf{q}} = \frac{1}{2}\begin{bmatrix} \boldsymbol{\omega} \\ 0 \end{bmatrix} \otimes \mathbf{q} \tag{4.42}$$

Deriving Eq. (4.38) with respect to time

$$\dot{\mathbf{q}} = \delta\dot{\mathbf{q}} \otimes \hat{\mathbf{q}} + \delta\mathbf{q} \otimes \dot{\mathbf{q}}_{est} \tag{4.43}$$

Substituting Eq. (4.42) into Eq. (4.43) and excluding the terms where two errors are multiplied, because they are second-order infinitesimals, the following expression is obtained:

$$\delta\dot{\mathbf{q}} = -\begin{bmatrix} \hat{\boldsymbol{\omega}} \times \delta\mathbf{q}_{1:3} + \frac{1}{2}\delta\boldsymbol{\omega} \\ 0 \end{bmatrix} \tag{4.44}$$

Eq. (4.44) has to be used in a discretized form to obtain the propagation, which is found using

$$\hat{\mathbf{q}}_k^{-} = \boldsymbol{\Theta}(\hat{\boldsymbol{\omega}}_{k-1}^{+})\hat{\mathbf{q}}_{k-1}^{+} \tag{4.45}$$

were $\boldsymbol{\Theta}(\omega_{k-1}^{+})$ is a 4×4 propagation matrix defined as

$$\boldsymbol{\Theta}(\omega_{k-1}^{+}) = \begin{bmatrix} \cos\left(\frac{1}{2}||\hat{\boldsymbol{\omega}}_{k-1}^{+}||\Delta t\right)\mathbf{I}_3 - \left[\hat{\boldsymbol{\psi}}_{k-1}^{+} \times\right] & \hat{\boldsymbol{\psi}}_{k-1}^{+} \\ -\left(\hat{\boldsymbol{\psi}}_{k-1}^{+}\right)^{\mathrm{T}} & \cos\left(\frac{1}{2}||\hat{\boldsymbol{\omega}}_{k-1}^{+}||\Delta t\right) \end{bmatrix} \tag{4.46}$$

with

$$\hat{\boldsymbol{\psi}}_{k-1}^{+} = \frac{\sin\left(\frac{1}{2}||\hat{\boldsymbol{\omega}}_{k-1}^{+}||\Delta t\right)\hat{\boldsymbol{\omega}}_{k-1}^{+}}{||\hat{\boldsymbol{\omega}}_{k-1}^{+}||} \tag{4.47}$$

where Δt is the time step of the propagation.

The estimated attitude matrix, in the form of a DCM ($\mathbf{A}(\hat{\mathbf{q}}_k^{-})$), which will be used in the measurement update step, can be written now according to Eq. (2.17). The discretized propagation equation for the gyro bias is

$$\hat{\boldsymbol{\beta}}_k^{-} = \hat{\boldsymbol{\beta}}_k^{+} \tag{4.48}$$

The equation for the propagation of the angular rate is, then

$$\hat{\boldsymbol{\omega}}_k^{-} = \tilde{\boldsymbol{\omega}}_{k-1}^{+} - \hat{\boldsymbol{\beta}}_k^{-} \tag{4.49}$$

The last parameters that need to be propagated are the 36 elements of the covariance matrix of the measurements. Now the discrete process noise 6×6 covariance matrix can be defined

$$\mathbf{Q} = \begin{bmatrix} \left(\sigma_v^2\Delta t + \frac{1}{3}\sigma_u^2\Delta t^3\right)\mathbf{I}_3 & -\left(\frac{1}{2}\sigma_u^2\Delta t^2\right)\mathbf{I}_3 \\ -\left(\frac{1}{2}\sigma_u^2\Delta t^2\right)\mathbf{I}_3 & \left(\sigma_u^2\Delta t\right)\mathbf{I}_3 \end{bmatrix} \tag{4.50}$$

The covariance matrix is be propagated using Eq. (4.11) with $\mathbf{G} = \mathbf{I}_{6\times 6}$ and the error-state transition matrix $\mathbf{F}$ is computed using a power series approach

$$\mathbf{F}_{k-1} = \begin{bmatrix} \mathbf{F}_{11} & \mathbf{F}_{12} \\ \mathbf{F}_{21} & \mathbf{F}_{22} \end{bmatrix} \tag{4.51a}$$

$$\mathbf{F}_{11} = \mathbf{I}_3 - \left[\hat{\boldsymbol{\omega}}_{k-1}^{+} \times\right]\frac{\sin\left(||\hat{\boldsymbol{\omega}}_{k-1}^{+}||\Delta t\right)}{||\hat{\boldsymbol{\omega}}_{k-1}^{+}||} + \left[\hat{\boldsymbol{\omega}}_{k-1}^{+} \times\right]^2\frac{1 - \cos\left(||\hat{\boldsymbol{\omega}}_{k-1}^{+}||\Delta t\right)}{||\hat{\boldsymbol{\omega}}_{k-1}^{+}||^2} \tag{4.51b}$$

$$\mathbf{F}_{12} = \left[\hat{\boldsymbol{\omega}}_{k-1}^{+} \times\right] \frac{1-\cos\left(||\hat{\boldsymbol{\omega}}_{k-1}^{+}||\Delta t\right)}{||\hat{\boldsymbol{\omega}}_{k-1}^{+}||} - \mathbf{I}_3 \Delta t - \left[\hat{\boldsymbol{\omega}}_{k-1}^{+} \times\right]^2 \frac{||\hat{\boldsymbol{\omega}}_{k-1}^{+}||\Delta t - \sin\left(||\hat{\boldsymbol{\omega}}_{k-1}^{+}||\Delta t\right)}{||\hat{\boldsymbol{\omega}}_{k-1}^{+}||^3} \tag{4.51c}$$

$$\mathbf{F}_{21} = \mathbf{0}_{3\times 3} \tag{4.51d}$$

$$\mathbf{F}_{22} = \mathbf{I}_3 \tag{4.51e}$$

Contrarily to what happens for the AEKF algorithm, MEKF is able to partially overcome the problem of having a very good model for the environmental phenomena. In fact, the only equation to be propagated is the kinematic equation Eq. (4.42), because the angular rate knowledge is provided by the gyro measurements. In this manner it is possible to eliminate the need of propagating the equation of dynamics and a good estimate of the environmental torques is not necessary.

Measurement Update

Since there is no dependence of the measurement from the angular rate, the discrete measurement sensitivity matrix, which linearizes the second part of Eq. (4.29) is given by

$$\mathbf{H}_k = \left[\left[\mathbf{A}(\hat{\mathbf{q}}_k^{-})\mathbf{y}_I^{\text{true}} \times\right] \quad \mathbf{0}_{3\times 3}\right] \tag{4.52}$$

Using this information the Kalman Gain matrix can be computed as indicated in Eq. (4.13). The error state is now updated using

$$\Delta\hat{\mathbf{x}}_k^{+} = \mathbf{K}_k\left[\tilde{\mathbf{y}} - \mathbf{A}(\hat{\mathbf{q}}_k^{-})\mathbf{y}_I^{\text{true}}\right] \tag{4.53a}$$

$$\Delta\hat{\mathbf{x}}_k^{+} = \begin{bmatrix} \delta\hat{\boldsymbol{\theta}}_k^{+} \\ \Delta\hat{\boldsymbol{\beta}}_k^{+} \end{bmatrix} \tag{4.53b}$$

The covariance matrix is to be updated using Eq. (4.15). Finally, the quaternion and gyro bias are updated

$$\hat{\mathbf{q}}_k' = \begin{bmatrix} \frac{1}{2}\delta\hat{\boldsymbol{\theta}}_k^{+} \\ 1 \end{bmatrix} \otimes \hat{\mathbf{q}}_k^{-} \tag{4.54a}$$

$$\hat{\boldsymbol{\beta}}_k^{+} = \hat{\boldsymbol{\beta}}_k^{-} + \Delta\hat{\boldsymbol{\beta}}_k^{+} \tag{4.54b}$$

To ensure that the normalization constraint is respected, it is also good practice to impose a brute force normalization on the quaternion:

$$\hat{\mathbf{q}}_k^{+} = \frac{\hat{\mathbf{q}}_k'}{||\hat{\mathbf{q}}_k'||} \tag{4.55}$$

From this point on, the $k+1$ step takes place in the propagation and the process is reiterated.

Chapter 5

Attitude Control

CubeADCS offers several control modes: four different modes to detumble the spacecraft, and the nominal mode, which makes use of a Linear Quadratic Regulator (LQR) controller to point the satellite towards the Nadir direction. For this thesis a sixth mode has also been developed, which makes the satellite track the motion of the Sun on the Pitch plane.

5.1 B-dot Detumbling

The tumbling is the motion of free evolution of the attitude of the spacecraft when it is not controlled, i.e. right after deployment or when the communication with the ground station is lost for a significant amount of time [23]. In a tumbling spacecraft the angular velocity can be divided into two components, one at the low orbital frequency $\boldsymbol{\omega}_0$, the other one at the high tumbling frequency $\boldsymbol{\omega}_{tumb}$

$$\boldsymbol{\omega} = \boldsymbol{\omega_0} + \boldsymbol{\omega}_{tumb} \tag{5.1}$$

The objective of the B-dot algorithm is to detumble the spacecraft, i.e. to decrease the angular rate by eliminating the high frequency component, using the magnetorquers. The most efficient way to do so is to provide a torque that is in every moment proportional and inverse to the angular rate of the spacecraft. This torque can be provided by a magnetic dipole moment which has to be itself perpendicular to the local magnetic field.

The total derivative of the magnetic field in the Body frame can be written as

$$\frac{\mathrm{d}\mathbf{b}_B}{\mathrm{d}t} = \dot{\mathbf{b}}_B + \boldsymbol{\omega}_B^{BI} \times \mathbf{b}_B \tag{5.2}$$

From now on the indexes will be neglected for the sake of clarity. Considering the approximation $\mathrm{d}\mathbf{b}/\mathrm{d}t = 0$ because the magnetic field is almost constant in the time of the maneuver, it is possible to write

$$\dot{\mathbf{b}} = -\boldsymbol{\omega} \times \mathbf{b} \tag{5.3}$$

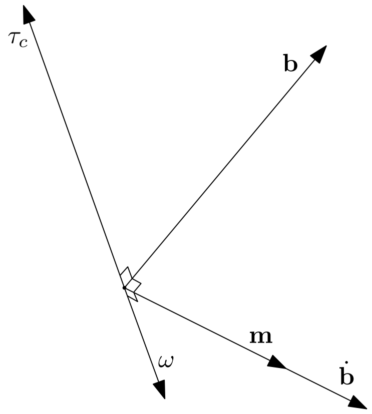


Figure 5.1: Vectorial representation of the involved variables.

Eq. (5.3) implies that $\mathbf{b} \perp \dot{\mathbf{b}}$. So the magnetic dipole moment $\mathbf{m}$ has to be parallel to the derivative of $\mathbf{b}$ in the body frame in order to obtain the maximum torque.

$$\mathbf{m} = k_d \dot{\mathbf{b}} \tag{5.4a}$$

$$\mathbf{m} = -k_d \boldsymbol{\omega} \times \mathbf{b} \tag{5.4b}$$

where k_d is a positive scalar gain. The vector of the local magnetic field in Body coordinates is measured by the magnetometer and derived with respect to time in the OBC and the magnetic dipole moment command is given accordingly to one of the forms of Eq. (5.4) [23]. The produced control torque is perpendicular to the magnetic dipole moment and the magnetic field, according to Eq. (3.9)

$$\boldsymbol{\tau}_c = \mathbf{m} \times \mathbf{b} \tag{5.5}$$

As illustrated in Fig. 5.1, the control torque vector is ideally in the opposite direction to the angular rate vector, thus maximizing the effectiveness of the deceleration.

According to Avanzini and Giulietti [76], a proper value for the detumbling gain k_d is given by the expression

$$k_d = \frac{4\pi}{T_{orb}}(1 + \sin\xi)J_{min} \tag{5.6}$$

where T_{orb} is the orbital period in seconds, ξ is the orbit inclination relative to the geomagnetic equatorial plane and J_{min} is the minimum principal axis of inertia of the spacecraft.

5.1.1 Detumbling controller stability

To study the stability of the algorithm, the expression for the rotational kinetic energy of the satellite is taken as a candidate Lyapunov's function [77]

$$V = \frac{1}{2}\boldsymbol{\omega}^{\mathrm{T}}\mathbf{J}\boldsymbol{\omega} \tag{5.7}$$

The time derivative of V is

$$\frac{\mathrm{d}V}{\mathrm{d}t} = \boldsymbol{\omega}^{\mathrm{T}}\mathbf{J}\frac{\mathrm{d}\boldsymbol{\omega}}{\mathrm{d}t} \tag{5.8}$$

The equation of dynamics (Eq. (2.46)) is substituted in Eq. (5.8), ignoring the disturbance torques

$$\frac{\mathrm{d}V}{\mathrm{d}t} = \boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{\tau}_c \tag{5.9}$$

where $\boldsymbol{\tau}_c$ is the control torque vector. Substituting Eq. (3.9) and Eq. (5.3) into Eq. (5.9)

$$\frac{\mathrm{d}V}{\mathrm{d}t} = -\mathbf{m}^{\mathrm{T}}(\boldsymbol{\omega} \times \mathbf{b}) \tag{5.10}$$

Finally, reminding the result from Eq. (5.4b)

$$\frac{\mathrm{d}V}{\mathrm{d}t} = -k_d(\boldsymbol{\omega} \times \mathbf{b})^{\mathrm{T}}(\boldsymbol{\omega} \times \mathbf{b}) \tag{5.11}$$

k_d is defined as a positive scalar gain, so it is clear that $\frac{\mathrm{d}V}{\mathrm{d}t} < 0$ and the controller is stable.

All of the four detumbling algorithms used by CubeADCS are based on the same principle that was here exposed and they are described in the following paragraphs. A main difference between them is the maximum angular rate that they are able to detumble. According to Fonod [77] the control sampling time can be loosely related to the maximum rate that can be detumbled in each component via the expression

$$T_s \leq \frac{\pi}{|\omega_{max}|} \tag{5.12}$$

5.1.2 Pitch-only and Y-Thomson Detumbling

The Pitch-only Detumbling mode is the most simple one. It only needs the instantaneous measurement of the magnetometers, which are derived in the Laplace domain in order to obtain the value of the first derivative of the magnetic field $\dot{\mathbf{b}}$. Only the Y-magnetometer is used in this mode, so it is only possible to reduce the components of the angular rate that are on the pitch plane (x and z). Since it does not use the measurement from the gyros to compute the derivative of the magnetic field, thus relying on Eq. (5.4a), this algorithm is simple and light, but it lacks in accuracy and it can only be used for very low angular rates, below 30°/s.

The second proposed algorithm is similar to the first one, but instead of using the y magnetorquer, it makes use of the x and z ones. It it is declared of being able to detumble angular rates below 30°/s, working at the control sampling frequency of 1 Hz [32].

5.1.3 Fast and Very Fast Detumbling

The Fast and the Very Fast Detumbling algorithms make use of the combination of the measurements of the rate gyros and of the magnetometer, so Eq. (5.4b) is employed. This last equation provides a more accurate estimate for the instantaneous rate of change of the magnetic dipole than Eq. (5.4a), because it makes use of the instantaneous measurements of two different sensors instead of two consecutive measurements of the same sensor. The difference between the two modes is that the first one works at a control sampling frequency of 1 Hz, whereas the second at 10 Hz. For this reason, while the Fast Detumbling mode is able to reduce initial rates up to 100°/s, the Very Fast can detumble up to 1000°/s [78].

5.2 Y-Momentum

The Y-Momentum control mode uses an LQR controller to 3-axes stabilize the CubeSat. The control torques are provided by the magnetorquers and the momentum wheel grants a maximum momentum bias of 1.7 mN m s to contrast the external torques. The default pointing objective of the Y-Momentum control mode is the Nadir direction, but the tracking of the Sun direction in the pitch plane is also investigated in this work.

In Fig. 5.2, the closed loop system that represents the quaternion feedback control (as presented in [26]) is reported.

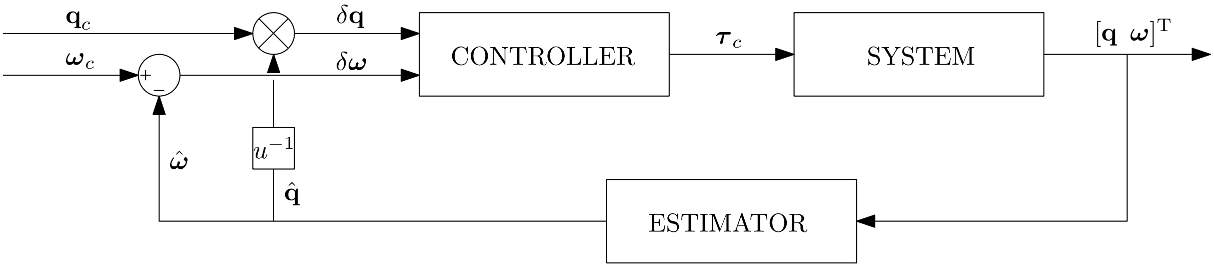


Figure 5.2: Closed loop system.

The non-linear system that represents the motion of the spacecraft, which was defined in Eq. (4.28), is included in the system block of Fig. 5.2,

$$\dot{\mathbf{q}} = \frac{1}{2}\Xi(\mathbf{q})\boldsymbol{\omega} \tag{5.13a}$$

$$\mathbf{J}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times (\mathbf{J}\boldsymbol{\omega} + \mathbf{h}) = \boldsymbol{\tau}_c \tag{5.13b}$$

Eq. (5.13b) is simplified considering only control torques and that the wheel angular momentum has reached the steady state value $\mathbf{h} = \begin{bmatrix} 0 & h_y & 0 \end{bmatrix}^{\mathrm{T}}$. The system composed by Eq. (5.13) is then linearized in Section 5.2.1 around the desired quaternion $\mathbf{q}_c$ and angular rate $\boldsymbol{\omega}_c$.

The state error can be defined as a combination of the quaternion error, as defined in Eq. (2.15), and the angular rate error

$$\delta\mathbf{q} = \mathbf{q}_c \otimes \mathbf{q}_{IB}^{-1} \tag{5.14a}$$

$$\delta\boldsymbol{\omega}_B = (\boldsymbol{\omega}_B)_c - \boldsymbol{\omega}_B^{IB} \tag{5.14b}$$

The output of the controller is the commanded control torque, defined using a control matrix $\mathbf{K}$

$$\boldsymbol{\tau}_c = -\mathbf{K}\begin{bmatrix}\delta\boldsymbol{\omega} & \delta\mathbf{q}_{1:3}\end{bmatrix}^{\mathrm{T}} \tag{5.15}$$

were the subscript B has been omitted for brevity. As shown in [23], for a Nadir Pointing spacecraft the commanded attitude quaternion is the one that makes the Body frame and the Orbital one coincident, so it is the quaternion that describes the rotation from the ECI frame to the Orbital frame

$$\mathbf{q}_c = \mathbf{q}_{OI} \tag{5.16}$$

The commanded angular velocity is the angular velocity of the ECI frame with respect to the Orbital frame [26]

$$\boldsymbol{\omega}_c = \boldsymbol{\omega}_B^{OI} = \begin{bmatrix} 0 \\ -\frac{||\mathbf{r}_I \times \mathbf{v}_I||}{||\mathbf{r}_I||^2} \\ \frac{||\mathbf{r}_I||(\hat{\mathbf{o}}_2)_I \cdot \dot{\mathbf{v}}_I}{||\mathbf{r}_I \times \mathbf{v}_I||} \end{bmatrix} \tag{5.17}$$

where $\mathbf{r}_I$, $\mathbf{v}_I$ and $(\hat{\mathbf{o}}_2)_I$ are respectively the position and velocity of the spacecraft in the ECI frame and the second column of the DCM that describes the transformation of the the Orbital frame into the ECI frame.

In the case of the Pitch Sun Tracking, the commanded quaternion is the one that transforms the ECI frame into the PST frame

$$\mathbf{q}_c = \mathbf{q}_{SI} \tag{5.18}$$

The commanded angular velocity is defined as the rate of rotation of the PST frame with respect to the ECI frame, which can be written using Eq. (2.5)

$$\boldsymbol{\omega}_B^{SI} = \boldsymbol{\omega}_B^{SO} + \boldsymbol{\omega}_B^{OI} \tag{5.19}$$

By definition of PST frame given in Eq. (2.26), the pitch axis of the PST frame coincides with the one of the Orbital frame, so the rotation $\boldsymbol{\omega}_B^{SO}$ only has a y non-zero component, that can be computed by the ADCS via a derivative operation on the rotation angle γ

$$\gamma = \sigma + \alpha \tag{5.20}$$

where σ is the angle defined in Eq. (2.27) and α is the additional constant angle needed to maximize the solar panels energy production, which only depends on the solar panels configuration.

The maximum illumination happens when the maximum available area on the pitch plane receives light. Fig. 5.3 might help to visualize the simple geometry to compute the optimal value of α: since the $-z$ face presents a solar panel twice the size of the $+x$ (or $-x$) face, the function to maximize to find the optimal angle α_{opt} is

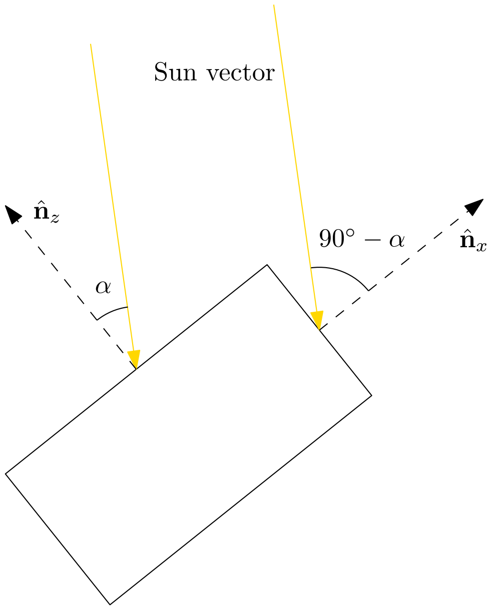


Figure 5.3: Definition of the α angle.

$$f(\alpha) = 2\cos(\alpha) + \sin(\alpha) \tag{5.21}$$

the derivative of which is

$$f'(\alpha) = -2\sin(\alpha) + \cos(\alpha) \tag{5.22}$$

In the range 0° – 90° the function presents a point of maximum in $\alpha_{opt} = \tan^{-1}(1/2) = 26.56°$.

Since α is a constant angle, the only term that appears in the commanded angular velocity equation is the derivative of the angle σ ($\dot{\sigma}$), which is computed by the OBC using two subsequent values of the angle in the finite difference formula

$$\boldsymbol{\omega}_c = \boldsymbol{\omega}_B^{IS} = \begin{bmatrix} 0 \\ -\frac{||\mathbf{r}_I \times \mathbf{v}_I||}{||\mathbf{r}_I||^2} + \dot{\sigma} \\ \frac{||\mathbf{r}_I||(\hat{\mathbf{o}}_2)_I \cdot \dot{\mathbf{v}}_I}{||\mathbf{r}_I \times \mathbf{v}_I||} \end{bmatrix} \tag{5.23}$$

5.2.1 Satellite Model Linearization

While the $\boldsymbol{\omega}_c$ for the Nadir Pointing mode has a dominant component in the y direction, that is four orders of magnitude higher than the one in the z direction, the commanded angular rate of the PST mode is generally always very close to $\mathbf{0}_3$ rad/s. This can be explained considering that in Eq. (5.19) the second term of $\boldsymbol{\omega}_B^{SO}$ and the one of $\boldsymbol{\omega}_B^{OI}$ are almost equal and opposite, because the Sun direction in the Orbital frame is quasi-constant throughout the orbital period. This results in an angular rate requirement very close to the inertial tracking case [23].

The controller is designed around the linearization point for the NP mode, so that the commanded angular rate is the one of Eq. (5.17). In order for the system to be linear it is required of it to be written in the form

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \tag{5.24}$$

It is well known that, in the words of Lustosa and al., the "nonlinear quaternion differential equations of motion linearization yields non-stabilizable systems" in a global sense [79]. A common way to address this problem is to only consider it locally, in the proximity of the objective state vector, which allows to express the quaternion in terms of three coordinates out of four. Thus, in this region the quaternion that transforms the body frame into the orbital frame can be written as [80]

$$\delta\mathbf{q} = \begin{bmatrix}\delta q_1 & \delta q_2 & \delta q_3 & 1\end{bmatrix}^{\mathrm{T}} \tag{5.25}$$

The state vector from Eq. (5.24) can be expressed in terms of quaternion and angular velocity:

$$\mathbf{x} = \begin{bmatrix}\delta\boldsymbol{\omega} & \delta\mathbf{q_{1:3}}\end{bmatrix}^{\mathrm{T}} \tag{5.26}$$

where $\boldsymbol{\omega} = \boldsymbol{\omega}_B^{OB}$.

Young [81] showed that, for any given $\delta\mathbf{q} \neq \begin{bmatrix}0 & 0 & 0 & -1\end{bmatrix}^{\mathrm{T}}$, there exists an univocal mapping between $\delta\boldsymbol{\omega}$ and $\delta\mathbf{q}$ and that the matrix of Eq. (2.38) can be written as

$$\boldsymbol{\Omega} = \begin{bmatrix} f(\mathbf{q}) & -\delta q_3 & \delta q_2 \\ \delta q_3 & f(\mathbf{q}) & -\delta q_1 \\ -\delta q_2 & \delta q_1 & f(\mathbf{q}) \end{bmatrix} \tag{5.27}$$

where $f(\mathbf{q}) := q_4 = \sqrt{1 - \delta q_1^2 - \delta q_2^2 - \delta q_3^2}$. So, Eq. (2.38) becomes

$$\delta\dot{\mathbf{q}}_{1:3} = \frac{1}{2}\boldsymbol{\Omega}\boldsymbol{\omega} \tag{5.28}$$

Because the hypothesis is that the first three components of the quaternion are almost equal to zero, it is common practice to approximate the matrix $\boldsymbol{\Omega}$ to the identity matrix $\mathbf{I}_3$. Eq. (5.28) consequently simplifies to

$$\delta\dot{\mathbf{q}}_{1:3} = \frac{1}{2}\boldsymbol{\omega} \tag{5.29}$$

Now it is convenient to recall the dynamics of the satellite from Eq. (2.48)

$$\mathbf{J}\dot{\boldsymbol{\omega}} = -\boldsymbol{\omega} \times (\mathbf{J}\boldsymbol{\omega} + \mathbf{h}) + \mathbf{u} \tag{5.30}$$

where $\mathbf{u}$ is the vector of the control torques. The environmental torques are excluded from the discussion.

A big advantage of using the mapping provided by Young is that the singularity $\delta\mathbf{q} \neq \begin{bmatrix}0 & 0 & 0 & -1\end{bmatrix}^{\mathrm{T}}$ is the farthest point from the operational point $\delta\mathbf{q} = \begin{bmatrix}0 & 0 & 0 & 1\end{bmatrix}^{\mathrm{T}}$, since a rotation of 180° separates them. This makes it possible to show the global stability of the system [82]. Disregarding the high order terms of the Taylor expansion around the stationary point, matrix $\mathbf{A}$ of Eq. (5.24) can be derived using Eq. (5.29) and Eq. (5.30)

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{0}_{3\times3} \\ \frac{1}{2}\mathbf{I}_3 & \mathbf{0}_{3\times3} \end{bmatrix} \tag{5.31a}$$

$$\mathbf{A}_{11} = \begin{bmatrix} 0 & 0 & -\frac{h_w}{J_1} \\ 0 & 0 & 0 \\ \frac{h_w}{J_3} & 0 & 0 \end{bmatrix} \tag{5.31b}$$

The matrix $\mathbf{B}$ is only dependant on the tensor of inertia

$$\mathbf{B} = \begin{bmatrix} \mathbf{J}^{-1} \\ \mathbf{0}_{3\times3} \end{bmatrix} \tag{5.32}$$

Lastly the input torque of the system is given by

$$\mathbf{u} = \mathbf{m} \times \mathbf{b}_B \tag{5.33}$$

where $\mathbf{m}$ is the control magnetic dipole and $\mathbf{b}_B$ is the magnetic field expressed in Body coordinates. Since it was assumed that the system is in close proximity to the operational point, the Body and the Orbital frame are almost coincident, thus Eq. (5.33) becomes

$$\mathbf{u} \simeq \mathbf{m} \times \mathbf{b}_O \tag{5.34}$$

5.2.2 Controller Design

The proposed attitude control algorithm is a Constant Gain Linear Quadratic Regulator (CGLQR), which was found by Sabino [17] to be the optimal control sequence among many others for this kind of mission. According to Wisniewski [83], it is convenient from a power efficiency point of view to introduce a new magnetic dipole moment $\mathbf{m}'$ defined as follows

$$\mathbf{m}'|\mathbf{m} = \frac{\mathbf{m}' \times \mathbf{b}_B}{||\mathbf{b}_B||} \tag{5.35}$$

Recalling from Eq. (5.34) that $\mathbf{b}_B \simeq \mathbf{b}_O$, the second term of the right side of Eq. (5.24) can now be written as

$$\mathbf{B}\mathbf{u} = \frac{\mathbf{J}^{-1}}{||\mathbf{b}_O||}\mathbf{b}_O \times \left(\mathbf{b}_O \times \mathbf{m}'\right) \tag{5.36}$$

So it is possible to rearrange the $\mathbf{B}$ matrix of Eq. (5.24) to be expressed in terms of the magnetic field $\mathbf{b}(t)$, which is time-variant, and the control vector to be equivalent to the new magnetic dipole moment defined in Eq. (5.35)

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t) \tag{5.37a}$$

$$\mathbf{B}(t) = \begin{bmatrix} \frac{\mathbf{J}^{-1}}{||\mathbf{b}_O(t)||} \big[\mathbf{b}_O(t) \times \big] \big[\mathbf{b}_O(t) \times \big] \\ \mathbf{0}_{3\times 3} \end{bmatrix} \tag{5.37b}$$

$$\mathbf{u}(t) = \mathbf{m}'(t) \tag{5.37c}$$

The problem is now to find a full-state feedback control law that minimizes the LQR cost function, which is defined as

$$J(\mathbf{u}) = \int_0^{15T_{orb}} [\mathbf{x}^{\mathrm{T}}(t)\mathbf{Q}\mathbf{x}(t) + \mathbf{u}^{\mathrm{T}}(t)\mathbf{R}\mathbf{u}(t)]\mathrm{d}t \tag{5.38}$$

where $\mathbf{Q}$ is a positive semi-definite matrix and $\mathbf{R}$ is a positive definite one. The first term inside the integral of Eq. (5.38) denotes the state space cost and the second one the energy cost [84].

To reduce the computational load of the algorithm, a constant gain approach is employed, so the matrix $\mathbf{B}(t)$ in Eq. (5.37a) is to be averaged over the longest period possible. In this work a time span of 1 day was taken, correspondent to slightly more than 15 orbital periods:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \Big(\frac{1}{15T_{orb}} \int_{\tau}^{\tau+15T_{orb}} \mathbf{B}(t)\mathrm{d}t\Big)\mathbf{u}(t) = \mathbf{A}\mathbf{x}(t) + \overline{\mathbf{B}}\mathbf{u}(t) \tag{5.39}$$

Since the system is Linear and Time Invariant (LTI) and controllable, a control law is yielded by any solution to the steady state algebraic Riccati equation.

$$\mathbf{P}^{\mathrm{T}}\mathbf{A} + \mathbf{P}\mathbf{A} - \mathbf{P}\overline{\mathbf{B}}\mathbf{R}^{-1}\overline{\mathbf{B}}^{\mathrm{T}}\mathbf{P} + \mathbf{Q} = 0 \tag{5.40}$$

The optimal control law is given by [85]

$$\mathbf{u}(t) = -\mathbf{R}^{-1}\overline{\mathbf{B}}^{\mathrm{T}}\mathbf{P}\mathbf{x}(t) = -\mathbf{K}\mathbf{x}(t) \tag{5.41}$$

where the matrix $\mathbf{K}$ is the control matrix. To conclude, the actual magnetic dipole moment $\mathbf{m}$ is computed according to Eq. (5.35).

5.2.3 LQR Controller Stability

In order to study the stability of the controller, a candidate Lyapunov's function has to be defined [84]

$$V = \frac{1}{2}\boldsymbol{\omega}_I^{\mathrm{T}}\mathbf{E}\boldsymbol{\omega}_I - 2q_0 \tag{5.42}$$

where ω_I is the rate vector in the inertial frame and $\mathbf{E}$ is a semi-positive definite matrix that represents the deviation from the state of stability. Only if $\frac{\mathrm{d}V}{\mathrm{d}t}$ is negative for the non-linear system dynamics will the controller be stable. The derivative of the Lyapunov function is

$$\frac{\mathrm{d}V}{\mathrm{d}t} = \boldsymbol{\omega}_I^{\mathrm{T}}\mathbf{E}\frac{\mathrm{d}\boldsymbol{\omega}_I}{\mathrm{d}t} - 2\frac{\mathrm{d}q_4}{\mathrm{d}t} \tag{5.43}$$

Substituting Eq. (5.29) and Eq. (5.30) into Eq. (5.43)

$$\frac{\mathrm{d}V}{\mathrm{d}t} = -\boldsymbol{\omega}_I^{\mathrm{T}}\mathbf{E}\big[\mathbf{J}^{-1}\boldsymbol{\omega}_I \times \big(\mathbf{J}\boldsymbol{\omega}_I + \mathbf{h}\big)\big] + \mathbf{J}^{-1}\mathbf{u} + \boldsymbol{\omega}_I^{\mathrm{T}}\mathbf{q}_{1:3} \tag{5.44}$$

Recalling the relation given by the gain matrix between the system input and the state vector from Eq. (5.41)

$$\frac{\mathrm{d}V}{\mathrm{d}t} = -\boldsymbol{\omega}_I^{\mathrm{T}}\mathbf{E}\big[\mathbf{J}^{-1}\boldsymbol{\omega}_I \times \big(\mathbf{J}\boldsymbol{\omega}_I + \mathbf{h}\big)\big] + \mathbf{J}^{-1}\mathbf{R}^{-1}\mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{x} + \boldsymbol{\omega}_I^{\mathrm{T}}\mathbf{q}_{1:3} \tag{5.45}$$

Substituting the values for the gain matrix

$$\frac{\mathrm{d}V}{\mathrm{d}t} = -\boldsymbol{\omega}_I^{\mathrm{T}}\mathbf{E}\big[\mathbf{J}^{-1}\boldsymbol{\omega}_I \times \big(\mathbf{J}\boldsymbol{\omega}_I + \mathbf{h}\big)\big] - \mathbf{J}^{-1}\mathbf{R}^{-1}\begin{bmatrix}\mathbf{J}^{-1}\\ \mathbf{0}_{3\times 3}\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}\mathbf{P}_{11} & \mathbf{P}_{12}\\ \mathbf{P}_{21} & \mathbf{P}_{22}\end{bmatrix}\begin{bmatrix}\boldsymbol{\omega}\\ \mathbf{q}_{1:3}\end{bmatrix} + \boldsymbol{\omega}_I^{\mathrm{T}}\mathbf{q}_{1:3} \tag{5.46}$$

Lastly, the equation can be simplified to show that the value of the derivative of the Lyapunov's function is always negative

$$\frac{\mathrm{d}V}{\mathrm{d}t} = -\boldsymbol{\omega}_I^{\mathrm{T}}\mathbf{E}\big[\mathbf{J}^{-1}\boldsymbol{\omega}_I \times \big(\mathbf{J}\boldsymbol{\omega}_I + \mathbf{h}\big)\big] - \boldsymbol{\omega}_I^{\mathrm{T}}\mathbf{E}\mathbf{J}^{-1}\mathbf{R}^{-1}\big(\mathbf{J}^{-1}\mathbf{P}_{11}^{\mathrm{T}}\boldsymbol{\omega} + \mathbf{J}^{-1}\mathbf{P}_{12}^{\mathrm{T}}\mathbf{q}_{1:3}\big) \tag{5.47}$$

Because $\mathbf{E}$, $\mathbf{P}$, $\mathbf{J}$ and $\mathbf{R}$ are all at least semi-positive definite matrices, all three terms of the right hand side of Eq. (5.47) are negative definite, thus the system is locally stable because $\frac{\mathrm{d}V}{\mathrm{d}t} < 0$. Moreover, it was shown by Yang [80] that the closed loop non-linear system described by Eq. (5.29) and Eq. (5.30) is asymptotically stable if either of the following conditions is respected

$$\mathbf{R} = c\mathbf{Q}_{22} \tag{5.48a}$$

$$\mathbf{R} = c\mathbf{Q}_{22}\mathbf{J} \tag{5.48b}$$

where $c > 0$ is a constant scalar and $\mathbf{Q}_{22}$ is the 3×3 sub-matrix on the bottom-right region of $\mathbf{Q}$.

Chapter 6

Pitch Sun Tracking Attitude Mode Feasibility Analysis

In the scientific literature it is very hard to find any study that highlights the advantages of adopting a PST attitude mode. Intuitively it appears to be an excellent method to improve the power availability of any CubeSat operating in LEO. In this chapter, simulation results are brought to establish in a qualitative way the advantages that this attitude mode would bring to the ORCASat mission.

6.1 ORCASat's Orbit

ORCASat is going to be launched on an ISS orbit, the orbital parameters of which are reported in Table 6.1

Table 6.1: ORCASat orbital parameters [16].

Parameter	**Value**
Radius of perigee	408 km
Radius of apogee	418 km
Inclination	51.64°
RAAN	117.76°
Argument of perigee	34.80°
Orbital period	5440 s

As it was explained in Section 2.6.1 the effect of the J2 harmonic of the gravitational field of the Earth makes the RAAN and the argument of perigee rotate, so the initial values are not of great importance in this preliminary analysis, because they periodically change with an almost linear behaviour.

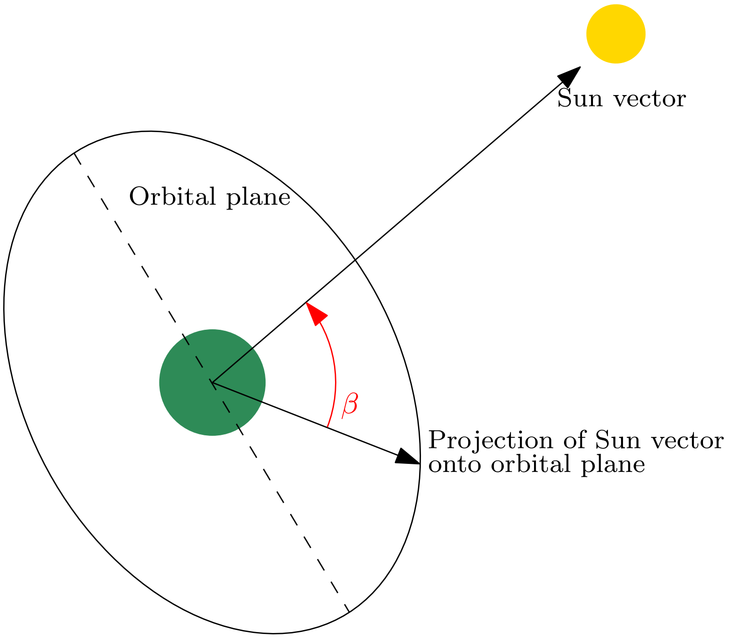


Figure 6.1: Definition of the β angle.

It is vital for the purpose of the feasibility analysis, to define two parameters that are closely related to the power budget analysis: the β angle and the eclipse period. The β angle is defined as the angle that the vector that links the center of the Earth with the Sun, which will be now called the Sun vector of the ECI frame, forms with its projection on the orbital plane.

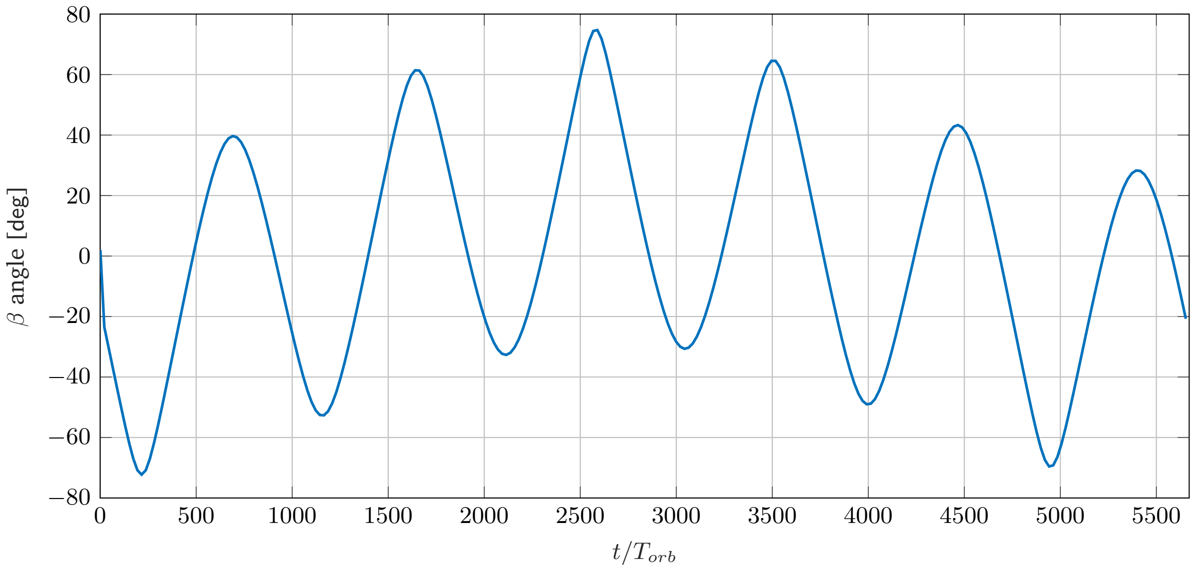


Figure 6.2: Evolution of the β angle for ORCASat's trajectory.

Fig. 6.2 shows that the β angle oscillates around the value of 0° with a double harmonic, reaching a maximum value of 75° and a minimum of $-72°$. The knowledge of this angle is important for two reasons: firstly it influences the entity of the eclipse period of the spacecraft, because it can be intended as a parametrization of the position of the Sun with respect to the satellite; secondly, the instantaneous power availability of the spacecraft is also dependent on it, because it gives an indication of where the

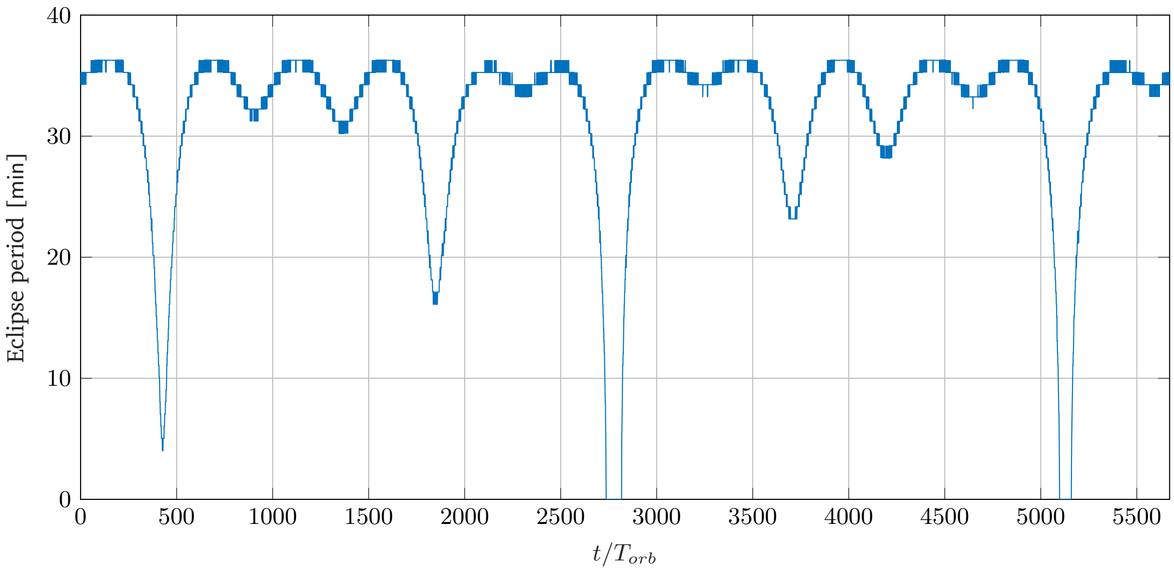


Figure 6.3: Eclipse period of ORCASat during its lifetime.

Sun is with respect to the spacecraft's solar panels.

Since the power availability is null during an eclipse, the orbits that present a higher eclipse period are also those in which the average power availability is expected to be lower. If one compares Fig. 6.3 with Fig. 6.2, what catches the eye is that the plot of the eclipse periods presents maxima in correspondence of the orbits where the β angle is close to zero, because in these orbits the ratio of eclipse time over orbital period is close to $\frac{1}{2}$.

In Fig. 6.3 it is shown that in certain moments the spacecraft undergoes a series of orbits where the eclipse period is null, but for most of the time the eclipse period oscillates between 28 min and 37 min. This plot is very important because it immediately gives an indication about the orbits where the power availability will be maximum and minimum.

6.2 Nadir Pointing vs Pitch Sun Tracking

As it was mentioned in the Introduction, the main objective of the PST mode would be to increase the minimum power availability during the mission lifetime. In order to do so, the satellite is set in an almost inertial attitude, which is simplified in Fig. 6.4, as was extensively explained in Section 5.2.

In order to compare the power availability of the two modes a set of simulations has been performed using AGI's System Tool Kit (STK). Since the intention is not to obtain exact results, but only a general idea of the viability of the proposed attitude mode, the simulations were run under very simplified conditions. The orbit of the spacecraft has been propagated using a J4 perturbation propagator[1], meaning that any perturbation different from the first two high order gravitational harmonics (atmospheric drag, solar pressure, third body) is not considered in the differential equation of motion. Moreover, attitude errors and delays related to the determination and control algorithms are disregarded and also environmental perturbing torques are not considered, so that the spacecraft is always pointing exactly where it

[1] https://help.agi.com/stk/index.htm

is supposed to, with a pointing error of 0°.

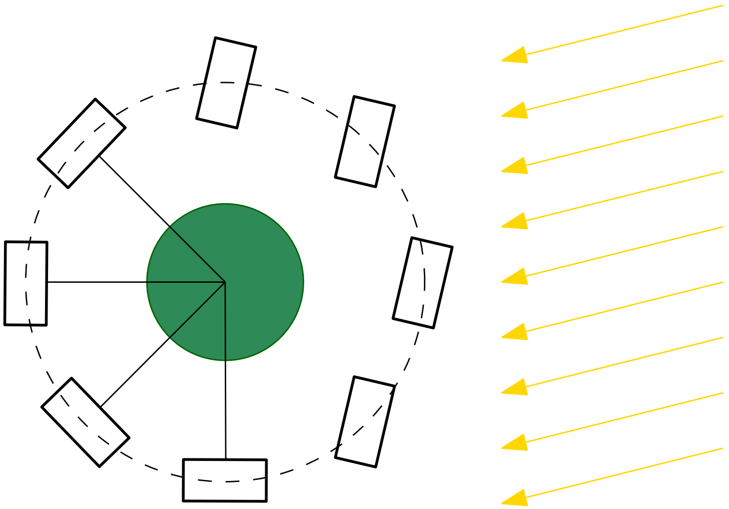


Figure 6.4: Nadir Pointing (on the left) and Pitch Sun Tracking (on the right): the direction of the Sun rays is indicated by the yellow vectors.

Two time scales were considered for the study: the first one comprehends the whole expected mission lifetime, whereas the second only includes one orbit, the one in which the power availability is expected to be lower. In the first case the comparison is done not on the instantaneous power availability, but on the power availability averaged over the orbital period, because this parameter gives a better idea of the evolution of the power available to the spacecraft during its lifetime. For the second pair of simulation, instead, since the time period is that of an orbit, the instantaneous power availability is considered.

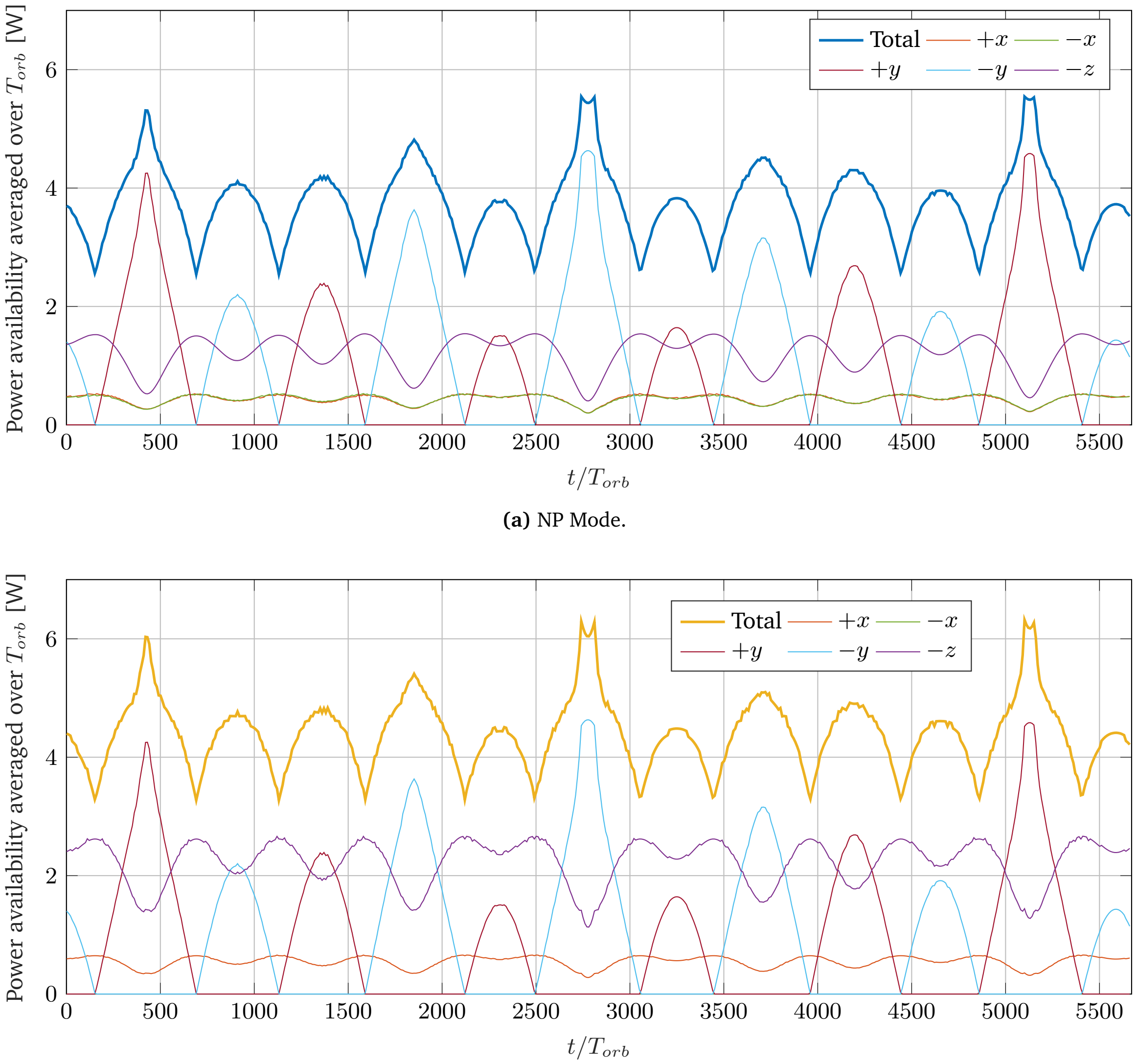
The first pair of simulations starts January 1st 2022 at 19:00:00 UTC. This is an arbitrary starting time, but it does not imply any loss of generality, because the relative motion of the spacecraft, the Sun and Earth periodically yields the same configuration every year. The time step adopted for the propagation of the orbit is 60 seconds, which is low enough to have very low errors dependent on the integration.

The second set of simulations starts on the 153rd orbit from Fig. 6.5 and is performed with a time step of 10 seconds, so that the plot is smoother, while the error in the integration is still low.

Table 6.2: Average power availability for the two attitude modes.

Mode	P_{in} [W]		
	max	mean	min
NP	5.56	3.74	2.54
PST	6.36	4.41	3.27
Difference	0.80	0.67	0.73

Table 6.2 shows that the PST attitude mode grants an increase both in the minimum power availability and in the maximum. Comparing Fig. 6.5b with Fig. 6.5a it is clear that this advantage is given mainly by an increase in the power availability of the $-z$ face, which on average produces almost 1 W more than in the NP mode. Also the $+x$ face has an increased availability, but this is counterbalanced by the fact that only one of the $\pm x$ faces is able to Sun-track, while the other stays on the shade for all the time. Using



(a) NP Mode.

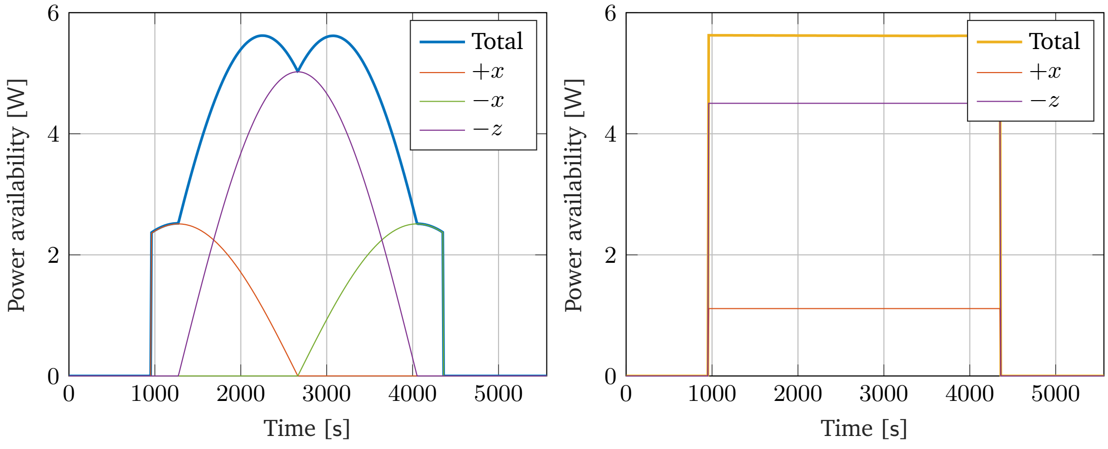
(b) PST Mode.

Figure 6.5: Comparison between the average power availability per orbit for the two attitude modes.

the NP mode, instead, the $+x$ and the $-x$ faces are alternatively illuminated. What obviously remains unchanged between the two plots is the availability of the $\pm y$ faces, because the Sun-tracking happens on the pitch plane, thus they are not affected by it.

The worst orbits for the power availability are those in which the $\pm y$ faces of the spacecraft are not illuminated. As Fig. 6.6 shows, the $+y$ and $-y$ power availability periodically go down to 0 W. This happens when the Sun, as expressed in the LVLH frame, does not have any component out of the pitch plane, so that it only illuminates the faces on the xz orbital plane. Table 6.3 highlights that in the single orbit period, the PST mode does not increase the maximum power availability the spacecraft is capable of, but it makes it possible for the power availability to stay almost constant. In fact, via tracking the Sun on the pitch plane, the power available on the $-z$ and $+x$ face remain constant, so that the average power produced by the solar panels is 0.76 W higher than the one that the NP mode grants. This is evident in a graphical way also from Fig. 6.6. The average power availability for the worst orbit is determined only by the total eclipse period for the PST mode, whereas for the NP mode also the variation of the illumination of the $\pm x$ and $-z$ faces plays an important role. In other words, in NP mode the power availability function is a sum of sinusoids, while in PST mode it is a sum of step functions.

The mean operation for the first pair of simulations is performed over the whole orbital period, including the eclipse periods, for which the power availability is null. For the second set of simulations, instead, the analysis is performed only on the time the satellite spends in the sunlight, so that also the minimum value of the available power can be compared. This is the reason why the mean values of Table 6.3 do not match the minima in the two graphics of Fig. 6.5.



(a) NP Mode.

(b) PST Mode.

Figure 6.6: Comparison between the instantaneous power availability for the two attitude modes for the worst orbit.

Table 6.3: Instantaneous power availability for the two attitude modes for the worst orbit.

Mode	P_{in} [W]		
	max	mean	min
NP	5.64	4.88	2.36
PST	5.64	5.64	5.64
Difference	0	0.76	3.28

Chapter 7

Numerical Simulations

In this chapter the results of the ADCS simulations are presented. In the first part a description of the simulation environment is provided. The second part is an assessment of the performances of the four detumbling modes. In the third one a brief comparison is established between the two angular rate estimation modes. The fourth and fifth parts focus on the estimation accuracy of the two EKF estimators and on the pointing accuracy of the two pointing modes, Nadir Pointing and Pitch Sun Tracking. In the fifth part, also a comparison regarding the power analysis of the two pointing modes is performed, in terms of availability vs consumption, with an eye towards the trade off between power consumption and pointing requirement, which is mainly determined by the adopted value of the momentum bias.

7.1 Simulation Environment

All the simulations were performed using a developed version of a MATLAB® Simulink® model that was previously created in the preliminary works to this thesis by Lobo-Fernandes [16] and Sabino [17]. A simplification of the top level version of the model is reported in Fig. 7.1. The ADCS block needs to provide an estimation for the quaternion and the angular rates using the information from the OBC and from the sensors; moreover it provides the commanded magnetic dipole moment to feed the actuators block and it toggles the activation of the propagator inside the OBC, the reset of the GPS measurement and the starting up of the momentum wheel. In order to have an estimate of the attitude state, in all the estimation algorithms (e.g. TRIAD) sensor measurements and inertial vectors are combined. Hence the ADCS needs a way to estimate the reference vectors (position, velocity and environmental torques). In the Spacecraft Mechanics Simulator block the physics of the the satellite is reproduced; assuming that this block provides the real evolution of the spacecraft motion, the OBC block provides the best estimate of all the kinematic and dynamic variables. Lastly, the Sensors and Actuators block simulates the physical behavior of the Sun sensors, the magnetometer, the gyros, the magnetorquers and the momentum wheel.

To have a knowledge of the various errors that result from the simulations, most importantly the estimation error and the pointing error, the actual attitude of the spacecraft is needed, which would be impossible to know in reality. Recalling Eq. (2.15) and calling the real quaternion $\mathbf{q}^{\text{true}}$ and the estimated

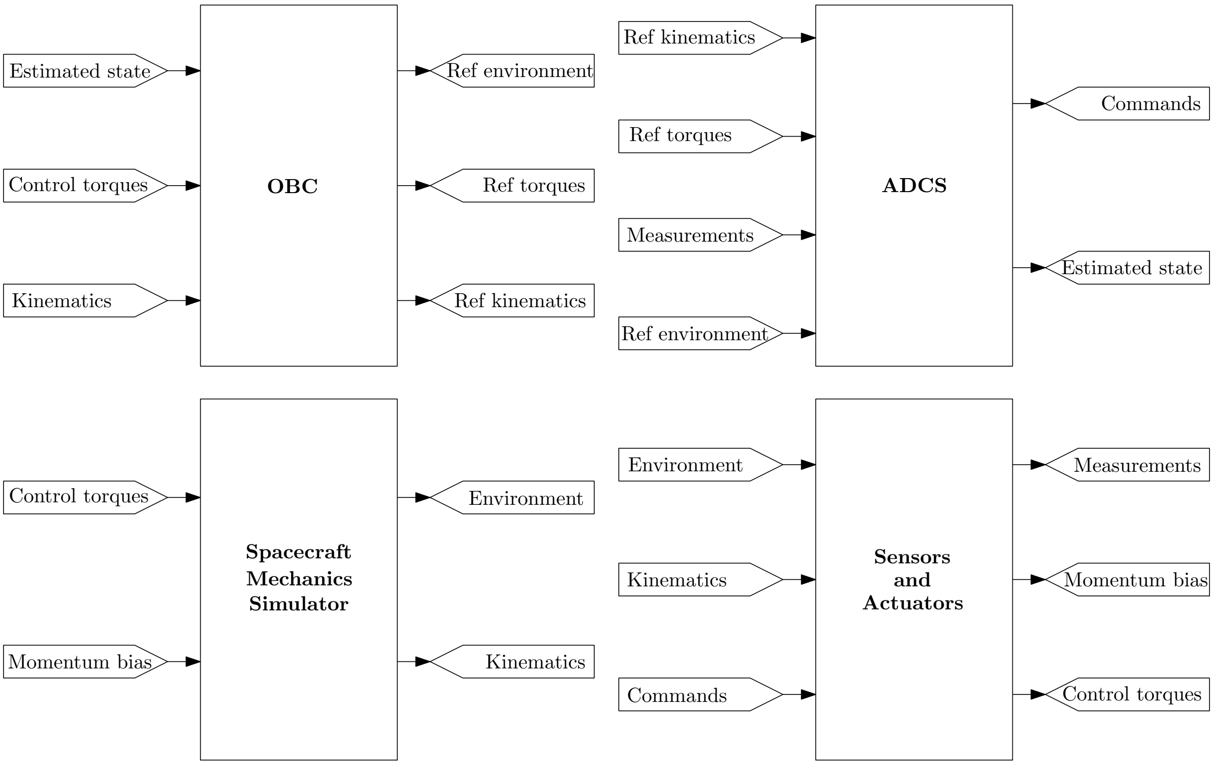


Figure 7.1: Top level model configuration.

one $\hat{\mathbf{q}}$, the quaternion of the estimation error is

$$\boldsymbol{\delta}\mathbf{q}_{est} = \mathbf{q}^{\text{true}} \otimes \hat{\mathbf{q}}^{-1} \tag{7.1}$$

If $\hat{\mathbf{q}} \simeq \mathbf{q}^{\text{true}}$, it follows that $\delta q_4 \simeq 1$ and $\boldsymbol{\delta}\mathbf{q}_{1:3} \simeq \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^{\text{T}}$ and it is possible to associate to δq_4 the angle $0° < \delta\phi < 180°$, which defines the estimation error

$$e_{est} = \delta\phi = 2\cos^{-1}(\delta q_{est})_4 \tag{7.2}$$

The same can be done for the pointing error e_{point}, if $\mathbf{q}_c$ is the quaternion of the commanded transformation defined in Section 5.2

$$\boldsymbol{\delta}\mathbf{q}_{point} = \mathbf{q}_c \otimes (\mathbf{q}^{\text{true}})^{-1} \tag{7.3}$$

$$e_{point} = 2\cos^{-1}(\delta q_{point})_4 \tag{7.4}$$

The calculation of all the other quantities necessary to evaluate the performance of the ADCS, like the angular rates, is simply done subtracting the measured, estimated or commanded vector (e.g. $\tilde{\boldsymbol{\omega}}$ or $\hat{\boldsymbol{\omega}}$, $\boldsymbol{\omega}_c$) to the real one (e.g. $\boldsymbol{\omega}^{\text{true}}$)

$$\Delta\boldsymbol{\omega}_{meas} = \boldsymbol{\omega}^{\text{true}} - \tilde{\boldsymbol{\omega}} \tag{7.5}$$

$$\Delta\boldsymbol{\omega}_{est} = \boldsymbol{\omega}^{\text{true}} - \hat{\boldsymbol{\omega}} \tag{7.6}$$

$$\Delta\boldsymbol{\omega}_{point} = \boldsymbol{\omega}_c - \boldsymbol{\omega}^{\text{true}} \tag{7.7}$$

In Table 7.1 the parameters selected in the base scenario are presented.

Table 7.1: Base simulation scenario.

Parameter	**Value**	**Unit**
Simulation time	86400	s
Initial epoch	19:00:00 01/01/2022	
Initial mean anomaly	$\pi/3$	rad
Initial attitude quaternion	$\begin{bmatrix}1 & 0 & 0 & 0\end{bmatrix}^{\mathrm{T}}$	
Mass	3.6	kg
Inertia matrix	$\begin{bmatrix}0.003 & 0 & 0 \\ 0 & 0.007 & 0 \\ 0 & 0 & 0.008\end{bmatrix}$	kg m^2
Parasitic dipole moment	$\begin{bmatrix}0.00707 \\ 0 \\ 0.00707\end{bmatrix}$	A m^2

7.2 Detumbling Modes Simulations

The simulations on the detumbling phase take into consideration all of the four control modes described in Section 5.1. Four different scenarios where considered for each of the control modes, which are presented in Table 7.2. Two cases are considered for the state of the momentum wheel: ON if it is accelerated until it reaches the maximum momentum bias, OFF if it is kept inactive. To demonstrate the robustness of the algorithms, two scenarios include an uncertainty in the direction of the principal axes of inertia, corresponding to a 123 Euler rotation of $\begin{bmatrix}-21^\circ & 15^\circ & 19^\circ\end{bmatrix}^{\mathrm{T}}$, and a significant variation in the value of the parasitic dipole moment. The detumbling gain for the Fast and the Very fast control modes were computed using Eq. (5.6) to be $k_d = 1.21 \times 10^{-5}$ N m s/T^2, whereas the gain for the other two control modes was selected via a trail and error process to be $k_d = 12.1$ N m s/T^2.

Each detumbling control mode was tested for a starting angular rate equivalent to the maximum one presented in Section 5.1. The detumble is considered successful once the absolute angular rate of the spacecraft goes below the threshold of $||\boldsymbol{\omega}|| = 0.02$ rad/s [17]. Once the absolute angular rate is reduced, each attitude mode is kept active until the base simulation period of 1 day was reached, in order to have an idea of the steady state behavior of each of the algorithms. In Table 7.3 the time that is taken to detumble the spacecraft in each scenario is presented, along with the maximum steady state absolute value of the angular rate and the mean value of the power required by the ADCS during the transient and the steady state. In this chapter only the plots of the evolution of the angular rate of the Very Fast and the XZ mode are shown in Fig. 7.2 and Fig. 7.3; in Appendix A the plots of the other two control modes

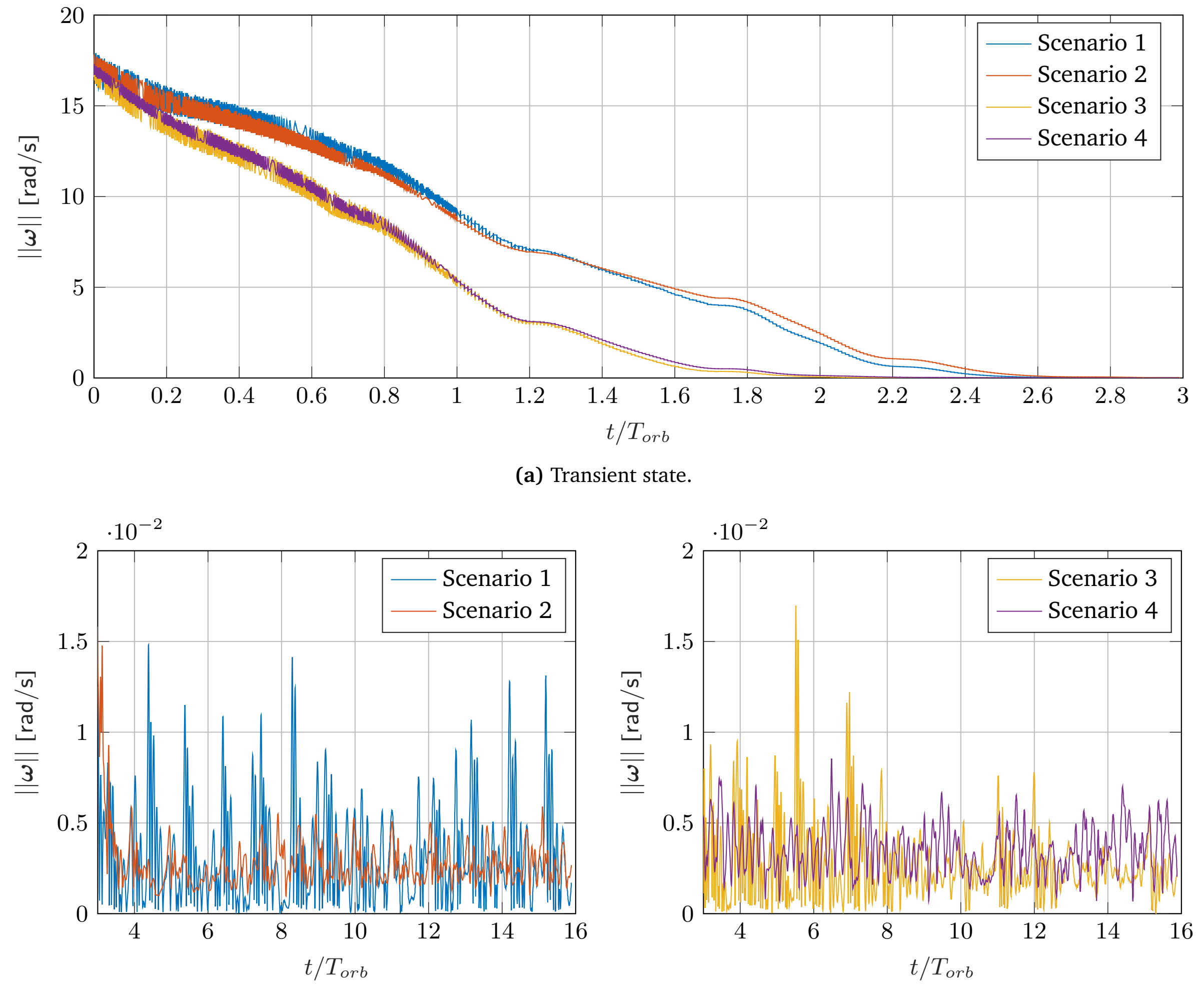
Table 7.2: Detumbling scenarios.

Scenario	Wheel state	Inertia Matrix [kg m^2]	Parasitic dipole moment [A m^2]
1	ON	$\begin{bmatrix} 0.003 & 0 & 0 \\ 0 & 0.007 & 0 \\ 0 & 0 & 0.008 \end{bmatrix}$	$\begin{bmatrix} 0.00707 \\ 0 \\ 0.00707 \end{bmatrix}$
2	OFF	$\begin{bmatrix} 0.003 & 0 & 0 \\ 0 & 0.007 & 0 \\ 0 & 0 & 0.008 \end{bmatrix}$	$\begin{bmatrix} 0.00707 \\ 0 \\ 0.00707 \end{bmatrix}$
3	ON	$\begin{bmatrix} 0.003814 & -0.001587 & -0.0006396 \\ -0.001587 & 0.006402 & 0.0001005 \\ -0.0006396 & 0.0001005 & 0.007784 \end{bmatrix}$	$\begin{bmatrix} 0.01514 \\ 0.01 \\ 0.01514 \end{bmatrix}$
4	OFF	$\begin{bmatrix} 0.003814 & -0.001587 & -0.0006396 \\ -0.001587 & 0.006402 & 0.0001005 \\ -0.0006396 & 0.0001005 & 0.007784 \end{bmatrix}$	$\begin{bmatrix} 0.01514 \\ 0.01 \\ 0.01514 \end{bmatrix}$

are presented for the sake of completeness.

The detumble times for the four modes are hard to compare, considering that each of them starts from a different initial rate, it is clear that the Very Fast mode takes almost three orbital periods to reduce the angular rate below 0.02 rad/s, whereas the Fast and the Y-Thomson take less than 1 orbital period. This difference is so evident because the first one starts from an absolute angular rate of 17.32 rad/s, which is ten times higher than the second one and more than thirty times higher than the third one. In general for the first two control modes, in scenarios 1 and 3, in which the momentum wheel is active, the detumble time is slightly lower than the scenarios in which it is inactive, respectively 2 and 4. For the Y-Thomson control mode, instead, the detumble time of scenario 4 is the lowest of the four. This is mainly due to the start-up torque of the momentum wheel that increases the angular rate, which makes the initial angular rate of scenarios 1 and 3 slightly higher than those of scenarios 2 and 4. In the case of the XZ control mode, theoretically the algorithm should not be able to detumble the spacecraft, because the pitch axis is uncontrolled. Nonetheless, since the parasitic magnetic dipole moment presents constant components on the X and Z directions, the detumble process can be achieved. In fact, the constant parasitic magnetic dipole moment can take the role of the magnetic dipole moment given by a permanent magnet. Some studies show how such permanent magnets can be used to passively detumble a spacecraft because over an orbital period on average they apply a torque that monotonically decreases the rotational kinetic energy of the satellite [86, 87]. In the two scenarios where the wheel is active, though, the detumble process either gets stuck on the initial value of the pitch rate (scenario 1) or decreases with an insufficient slope (scenario 3) as can be seen in Fig. 7.3.

The value of the maximum absolute angular rate reached in the steady state is mainly dependent on two factors: the state of the momentum wheel and the value of the detumble gain. It may surprise that among the first three control modes the Very Fast one, which is supposed to be the most precise, presents the highest values of maximum angular rate for almost all the scenarios. This can be explained by the



(a) Transient state.

(b) Steady state: scenarios 1 and 2.

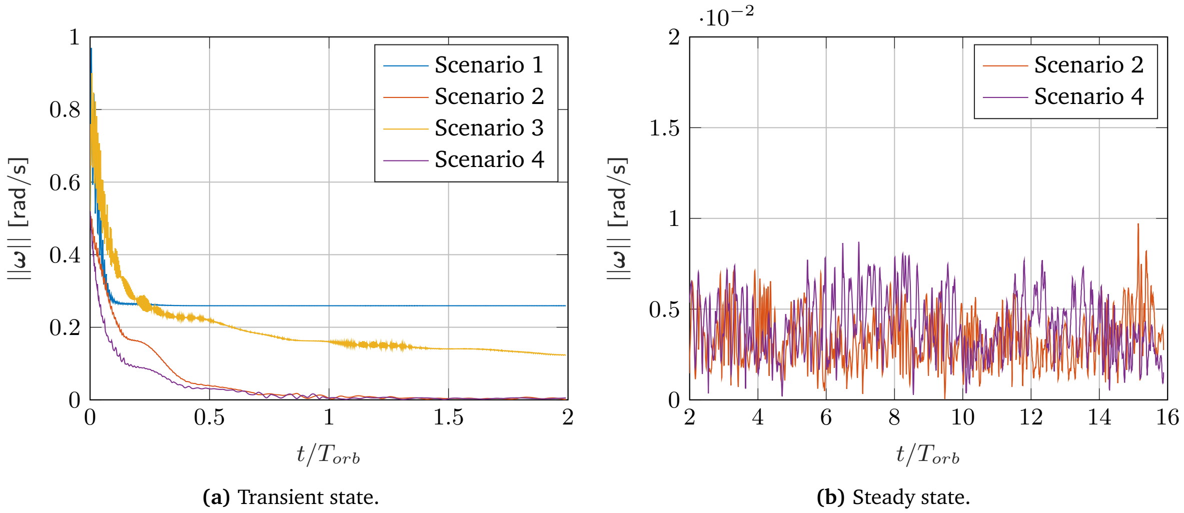
(c) Steady state: scenarios 3 and 4.

Figure 7.2: Very Fast detumbling mode angular rate evolution.

comparison of the sample time of each mode: the Very Fast is the only mode that works at 10 Hz, but the variation of the magnetic field during the steady state of the detumble mode are very small, so the detumble gain k_d tends to amplify them, because it is tuned for the transient state. In the case of the Fast mode, instead, since it works at 1 Hz, the provided control torque tends to be more constant, thus making the period of the oscillations of the angular velocity larger and their amplitude slightly lower.

Power-wise, since the Y-Thomson mode uses only two magnetorquers and the XZ mode only uses one, they are bound to require less power than the first two modes, both in the transient and in the steady state phase. For all of the modes scenarios 1 and 3 naturally require more power than the scenarios 2 and 4, in which the wheel is not working. This difference in power requirement oscillates around 330 mW, which is the maximum power required by the momentum wheel.

From the previous analysis, one can understand that, depending on the tumbling situation of the satellite, it might be convenient to use one detumbling mode rather than the others. If the satellite is tumbling with an absolute angular rate $||\boldsymbol{\omega}|| > 1.75$ rad/s the Very Fast control mode is the only one that can ensure that the satellite gets detumbled. If the angular rate is below the aforementioned threshold and the information from the gyroscopes is available, the Fast control mode performs better than the



(a) Transient state.

(b) Steady state.

Figure 7.3: XZ detumbling mode angular rate evolution.

others in terms of power efficiency. If, instead, the measurement from the gyroscopes are not available for whatever reason, the Y-Thomson control mode can detumble rates below 0.5 rad/s in an acceptable time. The XZ control mode should be used only as a last resort in extreme cases, i.e. if the power availability to the ADCS suddenly drops or if one of the magnetorquers undergoes a malfunctioning.

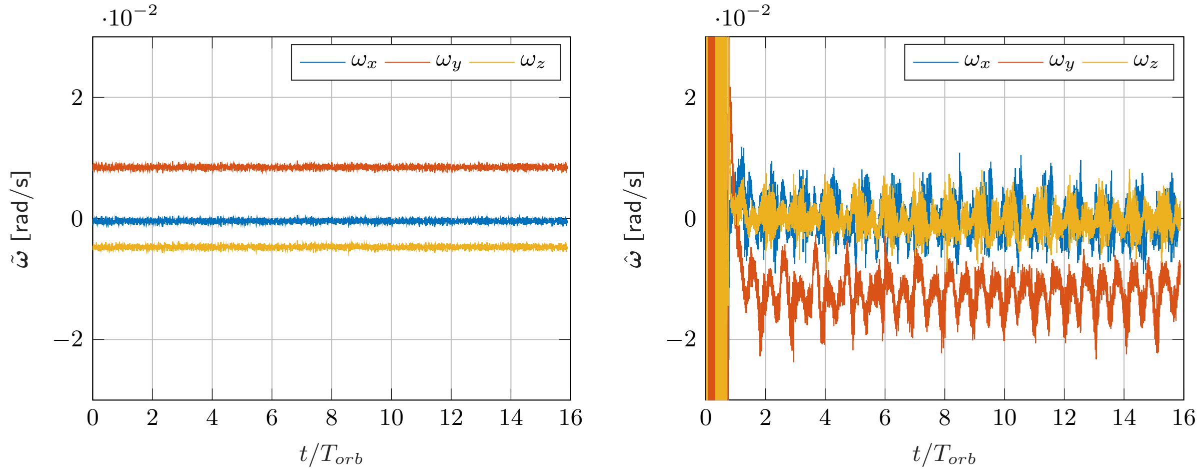
Table 7.3: Detumbling simulations results.

Mode	ω_0 [rad/s]	**Scenario**	$\mathbf{t_{det}}$ [s]	$\|\| \omega_{max} \|\|$ [rad/s]	$\overline{\mathbf{P}}$ **transient** [mW]	$\overline{\mathbf{P}}$ **steady** [mW]
Very Fast	$\begin{bmatrix}10\\10\\10\end{bmatrix}$	1	14607	1.50×10^{-2}	1336	453
		2	16013	0.93×10^{-2}	1000	111
		3	11434	1.89×10^{-2}	1356	423
		4	13137	0.86×10^{-2}	1008	89
Fast	$\begin{bmatrix}1\\1\\1\end{bmatrix}$	1	2572	1.11×10^{-2}	1385	358
		2	4661	0.77×10^{-2}	615	13
		3	3358	1.55×10^{-2}	1005	346
		4	4190	1.05×10^{-2}	602	9
Y-Thomson	$\begin{bmatrix}0.3\\0.3\\0.3\end{bmatrix}$	1	1052	0.96×10^{-2}	940	345
		2	2060	0.77×10^{-2}	460	9
		3	1139	1.65×10^{-2}	947	346
		4	709	0.69×10^{-2}	508	9
XZ	$\begin{bmatrix}0.3\\0.3\\0.3\end{bmatrix}$	1	∞	25.9×10^{-2}	338	nd
		2	3699	4.28×10^{-2}	136	3
		3	> 86400	1.33×10^{-2}	342	nd
		4	3567	1.65×10^{-2}	141	3

7.3 Rates Estimation Modes Simulations

As stated in Section 3.2.3, the first two estimation modes, Gyro rate measurement and Magnetic Rate Extended Kalman Filter (MREKF), are only able to estimate the angular rates of the satellite, so they have to be used in combination with another mode, like TRIAD, if the estimation of the full state of the satellite is needed. Nonetheless, in some occasions, such as during the end of the detumbling phase, only the knowledge of the angular rates might be required. In these cases it is preferable to use one of these modes instead of the more computationally heavy EKFs.

In Fig. 7.4, the behaviour of the two estimation modes during one orbit is presented. It appears that the estimation provided by the measurements of the gyroscopes, although inherently biased, tends to be almost one order of magnitude more accurate than the one provided by MREKF. The precision of the MREKF estimator is around 10^{-2} rad/s, so it is not advisable to use this estimation mode during either pointing mode, but only when the angular rate is inside the range 0.01 rad/s $< ||\boldsymbol{\omega}|| < 0.5$ rad/s. In fact, as demonstrated in Section 5.2, during the NP mode the absolute angular velocity of the satellite must equal the orbital rate $\omega_0 \simeq 2 \times 10^{-3}$ rad/s, whereas during the PST mode the angular rates vector should be always very close to $\mathbf{0}_3$. Moreover, when the angular rate is higher than the upper threshold, the magnetic field perceived by the magnetometer changes too quickly between two consecutive measurements, so that the computed value of its derivative, necessary for the knowledge of the angular velocity of the satellite, is impossible to determine. This is the reason why in the first orbital period of Fig. 7.4b, relative to the high rate detumbling phase, the estimation error is so much higher than in the rest of the simulation time, reaching a peak of 1.68 rad/s.



(a) Gyro Rate estimation mode.

(b) MREKF estimation mode.

Figure 7.4: Angular rate estimation accuracy.

7.4 Nadir Pointing Mode Simulations

The behaviour of the two complete estimation algorithms, AEKF and the MEKF, are compared in this section. The pointing error is determined by the estimation mode that is adopted, because the estimation accuracy influences the pointing accuracy. For the sake of completeness also the results of the TRIAD estimation mode are reported.

Realistic initial conditions are considered, so all the simulations presented in this section start from the initial angular rate $\boldsymbol{\omega}_0 = \begin{bmatrix} 0.5 & 0.5 & 0.5 \end{bmatrix}^{\mathrm{T}}$ rad/s. The Fast detumbling mode is used to reduce the angular rate at least down to $||\boldsymbol{\omega}|| = 0.02$ rad/s. Basing on the value of the absolute angular rate, the controller block manages the toggling of a series of actions:

- $||\boldsymbol{\omega}|| \geq 0.15$ rad/s: The Fast detumbling mode is employed to reduce the angular rate of the satellite, while keeping the momentum wheel turned off. This ensures a faster convergence, as is demonstrated in Section 7.2. The Gyro rate measurement estimation mode is activated.

- $||\boldsymbol{\omega}|| < 0.15$ rad/s: The wheel is started up. This causes the angular rate to slightly increase and it makes the detumbling process a bit slower, because the gyroscopic stiffness gets higher.

- $||\boldsymbol{\omega}|| < 0.13$ rad/s: The propagator inside the OBC is toggled in order to start computing the necessary reference variables that are needed to feed the estimator and the controller. To avoid the divergence of the errors in position and velocity of the spacecraft (which would cause a divergence also in the error of the computed environmental torques), the propagator is reset every hour via a GPS measurement.

- $||\boldsymbol{\omega}|| < 0.11$ rad/s: The estimator is turned on. It is important that the estimator starts working before the nominal mode is activated, so that when the Y-Momentum mode is started a good knowledge of the attitude is available. Up to this point the estimated angular rates are always equal to the measured rates, until the estimation of the attitude is acquired.

- $||\boldsymbol{\omega}|| < 0.02$ rad/s: When the estimated attitude gives a pitch pointing error $e_\theta < 25°$ and an error between the angular velocity achieved and the one required by the pointing mode in the pitch axis $e_{\dot{\theta}} < 0.05$ rad/s the estimation knowledge is considered acquired and the pointing mode is activated.

7.4.1 TRIAD Estimation Mode

The TRIAD algorithm is an algebraic method and it can only be activated when both the Sun sensors and the magnetometer measurements are available. This method could be used in practice only in those orbits in which the eclipse period is null ($\beta = 90°$), because when the measured Sun vector is not available it cannot work. So in such orbits as those just mentioned, TRIAD might be preferable to the EKF modes, because it is a faster algorithm in terms of computational load. Nonetheless, as Fig. 7.5 shows, the typical orbit of the satellite does include a non-null eclipse period and TRIAD is not capable of keeping the estimation error below the 2° requirement, even when combined with a good estimation method for the angular rates (Gyro Rate estimation mode was employed for the result of Fig. 7.5).

The best usage which can be done of TRIAD is the initialization of the recursive attitude determination methods like the two EKFs of the next section. In fact, one of their problems is that they must be fed an initial state and an initial covariance matrix; hence, if they are initialized when the Sun is in view of the

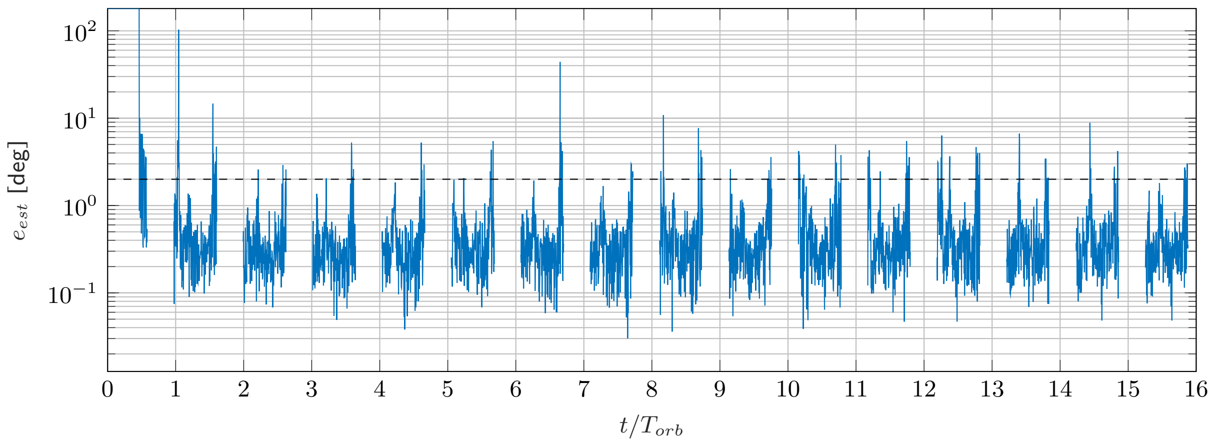


Figure 7.5: TRIAD estimation error.

sensors, TRIAD can provide a reasonable estimate (typically below 5°) for the state, which can be coupled with an initial diagonal covariance matrix with values below 10^{-6}. If the Sun sensors are not in view, instead, the initial state has to be guessed and the convergence tends to be much slower.

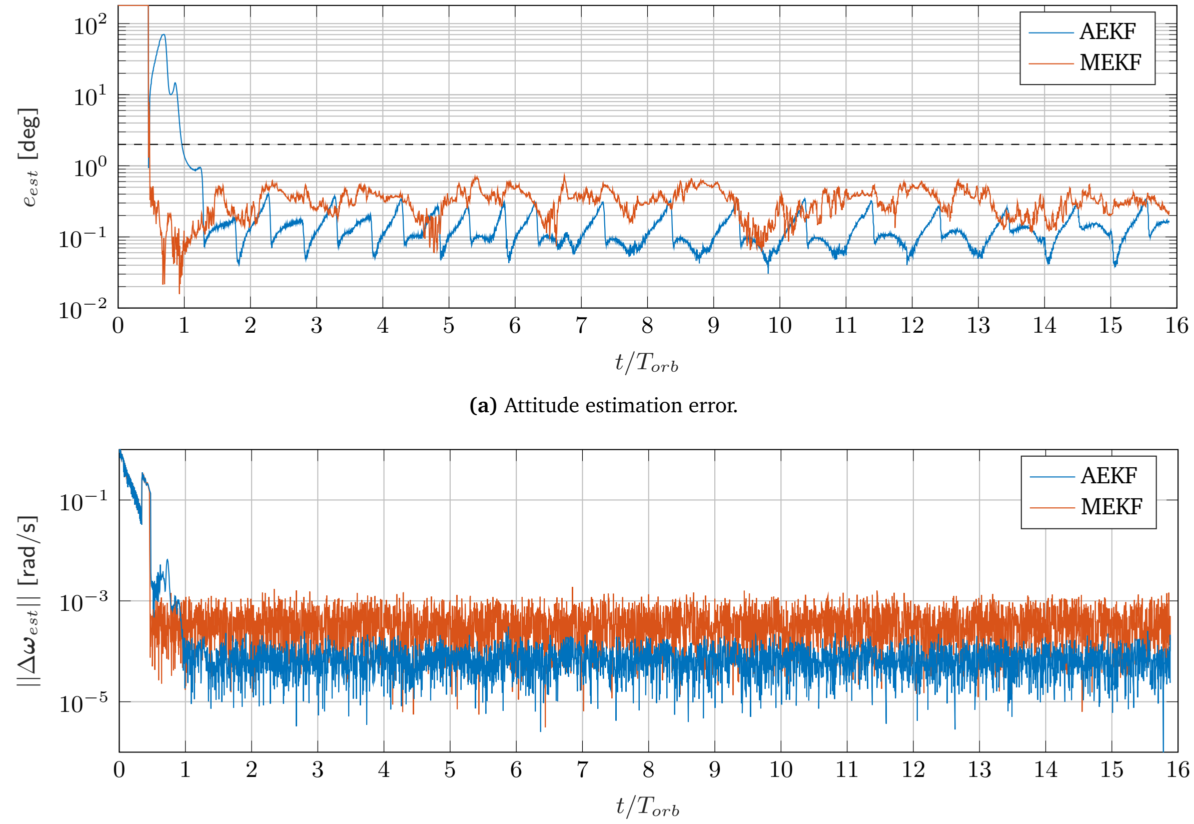
7.4.2 EKF Estimation Modes

The Additive EKF and the Multiplicative EKF are the only two estimation modes that can be used to support the two pointing modes. Similarly to what was done for the detumbling modes, five different scenarios, presented in Table 7.4, are considered to evaluate the robustness of the estimators and of the LQR controller.

Table 7.4: Pointing scenarios.

Scenario	**Mom. bias** [mN m s]	**Inertia matrix** [kg m^2]	**Par. mag. dipole mom.** [A m^2]
1	1.70	$\begin{bmatrix} 0.003 & 0 & 0 \\ 0 & 0.007 & 0 \\ 0 & 0 & 0.008 \end{bmatrix}$	$\begin{bmatrix} 0.00707 \\ 0 \\ 0.00707 \end{bmatrix}$
2	1.02	$\begin{bmatrix} 0.003 & 0 & 0 \\ 0 & 0.007 & 0 \\ 0 & 0 & 0.008 \end{bmatrix}$	$\begin{bmatrix} 0.00707 \\ 0 \\ 0.00707 \end{bmatrix}$
3	1.70	$\begin{bmatrix} 0.003 & 0 & 0 \\ 0 & 0.007 & 0 \\ 0 & 0 & 0.008 \end{bmatrix}$	$\begin{bmatrix} 0.00807 \\ 0.00500 \\ 0.00807 \end{bmatrix}$
4	1.70	$\begin{bmatrix} 0.003814 & -0.001587 & -0.0006396 \\ -0.001587 & 0.006402 & 0.0001005 \\ -0.0006396 & 0.0001005 & 0.007784 \end{bmatrix}$	$\begin{bmatrix} 0.00707 \\ 0 \\ 0.00707 \end{bmatrix}$
5	1.02	$\begin{bmatrix} 0.003814 & -0.001587 & -0.0006396 \\ -0.001587 & 0.006402 & 0.0001005 \\ -0.0006396 & 0.0001005 & 0.007784 \end{bmatrix}$	$\begin{bmatrix} 0.00807 \\ 0.00500 \\ 0.00807 \end{bmatrix}$

In Fig. 7.6 the estimation error of the pointing direction and of the angular rate of the two algorithms on scenario 1 shows that AEKF tends to behave slightly better than MEKF. In particular, the error on the pointing direction of the two modes is very similar, as AEKF presents a maximum estimation error of 0.410° and MEKF reaches 0.767°, but the pattern of AEKF is more stable than the one of MEKF. In terms of angular rate (Fig. 7.6b), instead, AEKF provides an estimation which is typically one order of magnitude lower than MEKF: the maximum rate estimation error of the former is 5.23×10^{-4} rad/s against 2.48×10^{-3} rad/s of the latter. So, in general, AEKF is expected to provide a lower pointing error than MEKF.



(a) Attitude estimation error.

(b) Rate estimation error.

Figure 7.6: Estimation performance of the EKF algorithms in Scenario 1 on NP mode.

It is important to notice, however, that the simulation of the AEKF mode presents some inherent flaws. As it was stated in Section 4.2.2, the Additive EKF strongly relies on a robust model of the environmental torques and for the propagation of the equations of motion. In the frame of this work, the model, which includes the geomagnetic, the atmospheric, the Sun pressure and the GNSS models, that computes the disturbances to propagate the motion of the satellite, is structurally identical to the one used to simulate the real environmental phenomena. This means that in the simulated world the OBC is provided with a model that, if given the perfect initial conditions, could predict with no error what the behaviour of the satellite would be. To overcome this issue, the GNSS initial position and velocity are corrupted by an overestimated source of error, so that, thinking to be in a slightly wrong position, the OBC would compute slightly wrong torques. As can be seen in Fig. 7.6a, though, the estimation is still really good and in Fig. 7.6b it appears odd that AEKF can provide a better estimation for the rates, via the simple integration of the equation of motion, than MEKF, which is supported by a sensor measurement. These concerns are confirmed by the study by Markley comparing the two quaternion EKFs [65], which highlights how MEKF is generally a more accurate estimator and there is no reason to use AEKF, unless the gyros measurements are not available.

Table 7.5: EKF estimators performance in Nadir Pointing mode.

Scenario	**Mode**	e_{est} [deg]		$\|\|\Delta\boldsymbol{\omega}_{est}\|\|$ [rad/s]		t_{est} [s]
		max	mean	max	mean	
1	AEKF	0.410	0.131	5.23×10^{-4}	7.33×10^{-5}	2754
	MEKF	0.767	0.324	2.48×10^{-3}	4.06×10^{-4}	16
2	AEKF	0.654	0.154	4.35×10^{-4}	7.01×10^{-5}	4887
	MEKF	0.840	0.313	2.58×10^{-3}	4.06×10^{-4}	3
3	AEKF	10.8	3.81	2.05×10^{-3}	1.96×10^{-4}	> 86400
	MEKF	0.816	0.324	2.56×10^{-3}	4.11×10^{-4}	41
4	AEKF	0.450	0.138	5.86×10^{-4}	7.72×10^{-5}	2546
	MEKF	1.15	0.329	2.46×10^{-3}	4.06×10^{-4}	14
5	AEKF	24.2	8.12	4.248×10^{-3}	3.71×10^{-4}	> 86400
	MEKF	0.783	0.325	2.528×10^{-3}	4.06×10^{-4}	7

To support this discussion it is sufficient to check the estimation performance of AEKF in scenarios 3 and 5 (presented in Table 7.5 and also in a graphic form in Appendix A), where the parasitic magnetic dipole moment $\mathbf{m}_{par}$ is modified. The result is completely unacceptable, since the maximum estimation error abundantly exceeds the 2° requirement.

Since the parasitic magnetic dipole moment is a parameter which is relatively hard to estimate, a set of thirteen simulations were run to illustrate the sensibility of the AEKF algorithm to it. As it appears from Fig. 7.7, both the mean and the maximum estimation errors increase almost linearly with the error between the value of the actual parasitic magnetic dipole moment and the one that is fed as a constant input to AEKF. This means that, while AEKF uses the value presented in Table 7.1, the real value provided in the Spacecraft Mechanics Simulator Block of Fig. 7.1 differs from it. Since the objective was not to have a fully quantitative estimation of the errors related to this aspect but just an idea of the entity of its influence in the estimation, in each simulation $\mathbf{m}_{par}$ was increased by the vector $\begin{bmatrix}1 & 1 & 1\end{bmatrix}^{\mathrm{T}}$ multiplied by a coefficient between 10^{-4} A m^2 and 10^{-2} A m^2. This error was found to be the parameter with the highest influence on the AEKF estimator.

Nonetheless AEKF is very robust to other sources of error, such as the perturbation of the inertia matrix. The comparison of the first and the fourth row in Table 7.5, leads to the conclusion that this parameter has little to no influence on the behaviour of the estimator.

Also, it was proven by another set of simulations (see Fig. 7.8) that even an increased initial error in the propagation of the equation of motion does not influence significantly the accuracy of AEKF. AEKF was always able to keep the maximum estimation error below 2° while the standard deviation of the position error of the GNSS was kept below 2.7×10^3 m. Fig. 7.8b, instead, illustrates how the estimator is much more sensible to the error of the orbital velocity measured by the GNSS: the maximum tolerable error is of 15 m/s, if the estimation error is to be kept below the requisite. Anyway, the typical error of a GNSS receiver is around the order of magnitude of 1 m for the position measurement and 10 cm/s for the velocity measurement, so these parameters should not cause any concern.

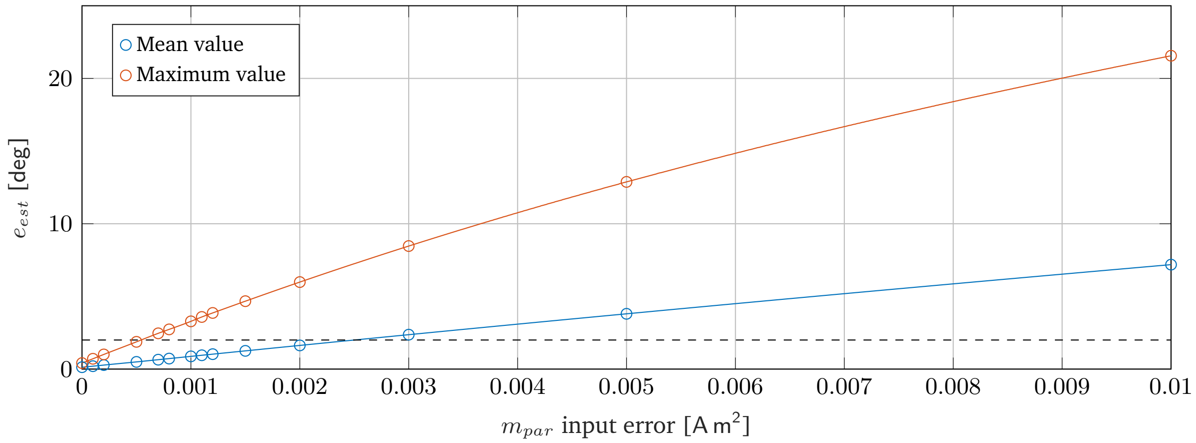
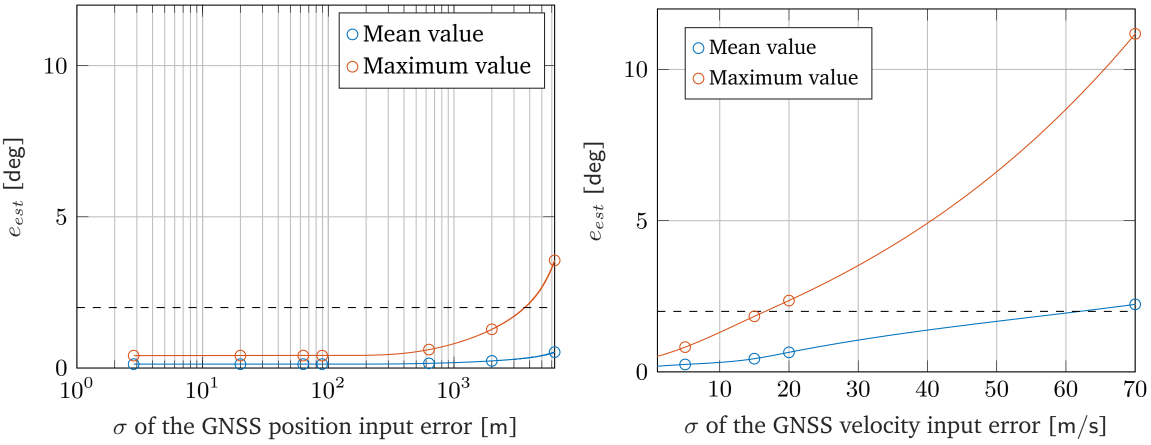


Figure 7.7: AEKF sensibility to the parasitic magnetic dipole moment input error.



(a) AEKF Sensibility to the position error.

(b) AEKF Sensibility to the velocity error.

Figure 7.8: AEKF sensibility to the GNSS initial error.

In scenario 2 AEKF and MEKF respectively increase the maximum attitude estimation error by 0.244° and 0.073° with respect to scenario 1, while the angular rate estimations do not present any significant difference, if not for some random effects. Therefore the value of the momentum bias does not have an noticeable influence on the behaviour of the estimators.

Lastly, it is significant to compare the time required by the estimators to converge to a solution below the requisite of 2° (last column of Table 7.5). MEKF is able to converge almost instantaneously in all the scenarios, while AEKF takes at least 2500 s. Both the algorithms use TRIAD for the initialization, as is explained in Section 7.4.1: for the first 30 s they are fed a constant covariance matrix and the quaternion estimated by the static method. Anyway, MEKF proves to be very robust to the transition between TRIAD and EKF, as the estimation error does not go above 2° for the whole time of the simulation (see Fig. 7.6a). The estimation error of AEKF, instead, after the first 30 s, spikes back to 70° and afterwards it slowly converges below the threshold. This means that MEKF is really fast in computing the right covariance matrix, whereas AEKF requires many iterations to do so.

7.4.3 NP Pointing Accuracy

From the previous discussion it emerges that the MEKF estimator is more reliable and more robust to uncertainties than AEKF, so the former was used to obtain the results of this section. In Fig. 7.9 the pointing error for the NP mode is shown in its evolution during the simulation period. The periodicity of the steady state is respected regardless of the considered scenario. It is interesting how the uncertainty of the parasitic magnetic dipole moment and the value of the momentum wheel play an interconnected role: in the first periodic peak scenario 2 presents the highest pointing error, while the one from scenario 5 is kept very low, even lower than scenario 1; anyhow in the second periodic peak scenario 5 always reaches higher peaks than scenario 2.

Table 7.6: MEKF pointing performance in Nadir Pointing mode.

Scenario	e_{point} [deg]			$\|\|\Delta\omega_{point}\|\|$ [rad/s]			t_{point} [s]
	Max	Mean	IAE [deg s]	Max	Mean	IAE [rad]	
1	9.77	4.58	3.79×10^5	4.66×10^{-4}	1.68×10^{-4}	13.92	3058
2	12.4	4.88	4.05×10^5	5.43×10^{-4}	2.01×10^{-4}	16.65	3784
3	8.42	5.26	4.35×10^5	4.71×10^{-4}	2.02×10^{-4}	16.78	2864
4	9.75	4.86	4.04×10^5	4.16×10^{-4}	1.78×10^{-4}	14.89	2864
5	9.44	5.77	4.82×10^5	4.35×10^{-4}	2.01×10^{-4}	16.76	1529

In scenarios 2 and 5, where the momentum bias is lower, the pointing accuracy seams generally worse, because it reaches higher peaks than in the other scenarios. The Integral Absolute Error (IAE) in Table 7.6, which is the best indicator of the error as a continuous entity, though, implies that scenarios 3 and 5 present the absolute worse performance. So, it can be concluded that the parasitic magnetic dipole moment is a parameter that can have a large influence on the pointing accuracy, even a more important one than the momentum bias value.

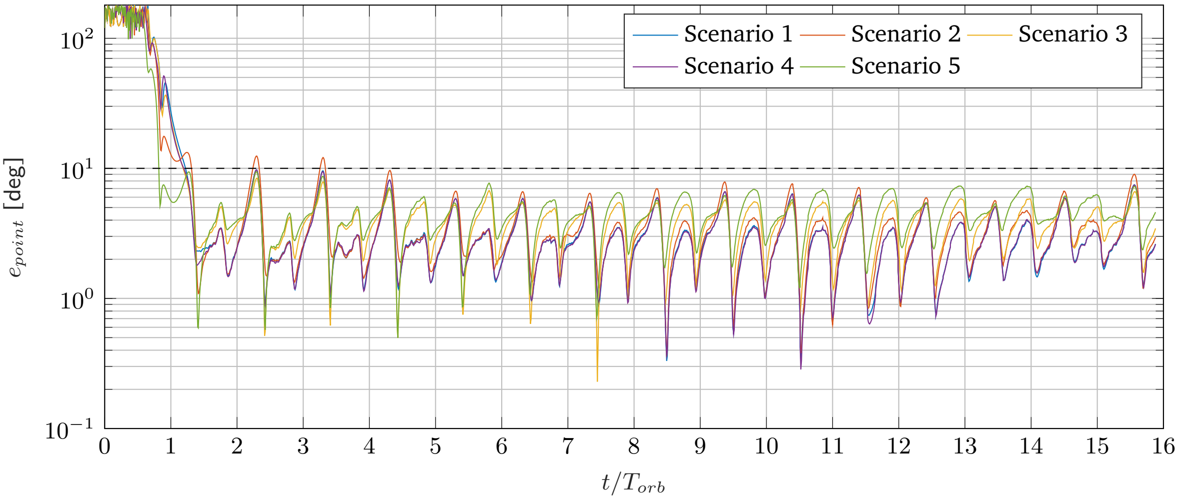


Figure 7.9: NP mode pointing accuracy.

As expected, the perturbation of the inertia matrix (scenario 4) does not have any noteworthy impact on the pointing accuracy, because it influences only the gravity gradient torque, which is not as significant as others (e.g. the parasitic magnetic torque).

The pointing error on the angular rate side is somehow less important than the attitude pointing error, but it can be indicative of the stability of its counterpart. Scenario 1 presents the lowest value for the mean error and for the IAE, whereas scenario 5 the highest ones. This is in accordance with the ranking of the scenarios in terms of attitude pointing error.

The last column of Table 7.6 reports the time that the controller needs to reach the threshold of 10°, since the pointing mode is activated. As expected, when the momentum wheel is running at low rate the convergence is lower, but in scenario 5, where all three perturbations are acting together, the convergence time is almost half that of the base simulation.

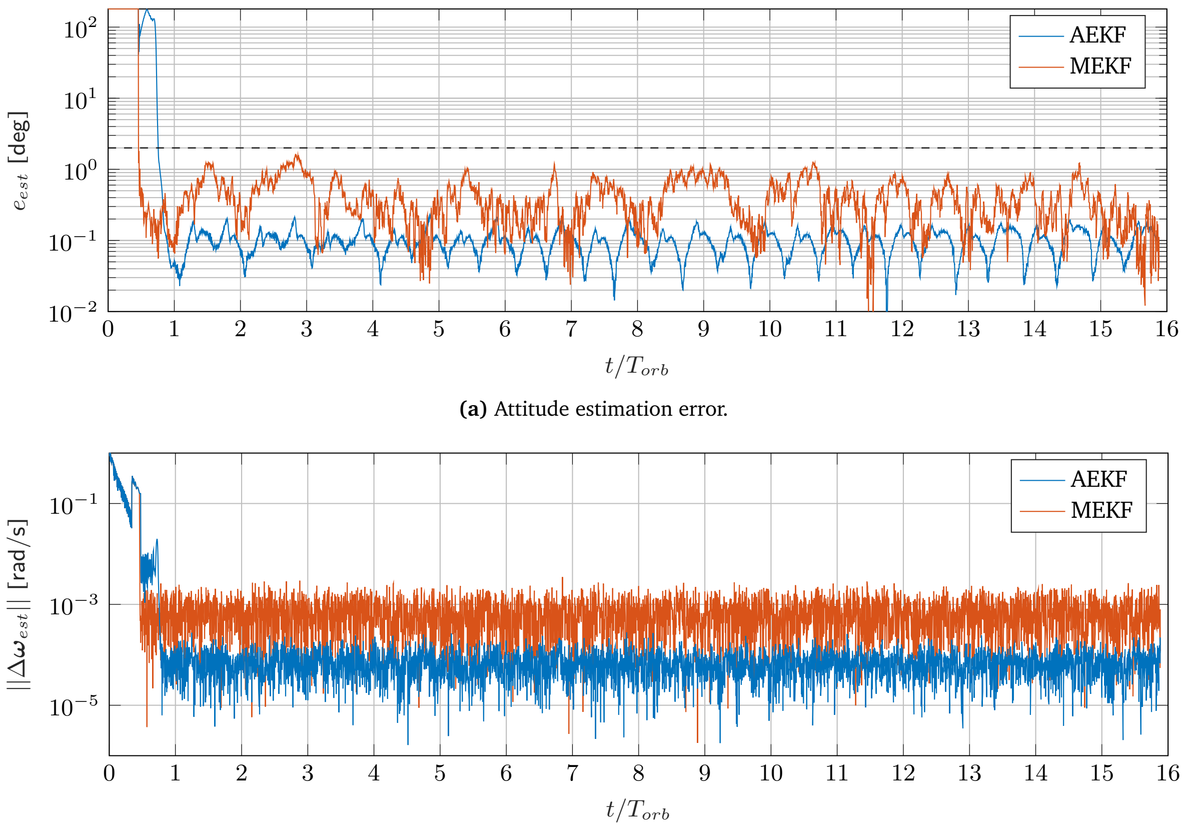
7.5 Pitch Sun Tracking Mode Simulations

The same methodology used in the previous section has been employed to study the behaviour of the estimators during the PST mode. It can be asserted, by looking at Table 7.7, that the behaviour of the estimators does not change with the change of pointing commands. It is clear, comparing Fig. 7.10 with Fig. 7.6, that the MEKF estimation is somewhat worse while the PST mode is functioning, but it is always kept below the requirement of 2°.

Table 7.7: EKF estimators performance in Pitch Sun Tracking mode.

Scenario	**Mode**	e_{est} [deg]		$\|\|\Delta\boldsymbol{\omega}_{est}\|\|$ [rad/s]		t_{est} [s]
		max	mean	max	mean	
1	AEKF	0.228	0.100	4.69×10^{-4}	7.15×10^{-5}	2754
	MEKF	1.67	0.418	4.51×10^{-3}	7.23×10^{-4}	16
2	AEKF	0.240	0.114	4.23×10^{-4}	6.79×10^{-5}	4887
	MEKF	1.58	0.422	4.44×10^{-3}	7.23×10^{-4}	3
3	AEKF	5.93	1.89	8.41×10^{-4}	1.33×10^{-4}	> 86400
	MEKF	0.891	0.243	2.48×10^{-3}	4.06×10^{-4}	41
4	AEKF	0.236	0.108	6.09×10^{-4}	7.51×10^{-5}	2546
	MEKF	1.606	0.427	4.55×10^{-3}	7.23×10^{-4}	14
5	AEKF	13.3	4.55	1.81×10^{-3}	2.38×10^{-4}	> 86400
	MEKF	1.52	0.428	4.41×10^{-3}	7.23×10^{-4}	7

The most noticeable difference between Fig. 7.10a and Fig. 7.6a is that in the former the estimation error of the AEKF mode is lower than in the latter, while for MEKF the opposite happens. AEKF behaves better because during the PST mode only the Zenith face is pointed towards the Sun, so that the Fine Sun Sensor is always used to compute the Sun vector. This results in an improved accuracy of the computed Sun vector with respect to the one of the NP mode, which eventually leads to a better estimation result. The same does not happen when the MEKF mode is active because, while also in this case only the fine Sun sensor is employed, the angular velocity is always very close to 0 rad/s and the performance of the gyros is worse than in the NP case (as can be seen comparing Fig. 7.10b with Fig. 7.6b). Since the estimation of the quaternion and the one of the angular rate vector are dependent from each other, it



(a) Attitude estimation error.

(b) Rate estimation error.

Figure 7.10: Estimation performance of the EKF algorithms in Scenario 1 on PST mode.

is logical to observe a slightly worse behaviour of the estimator in PST mode. Nonetheless, the same problematic exposed in Section 7.4.2 still apply in this case, so MEKF has to be considered more reliable than AEKF during the PST attitude mode as well.

7.5.1 PST Pointing Accuracy

The behaviour of the pointing accuracy for the different scenarios on PST mode is very similar to the one pointed out in Section 7.4.3.

Table 7.8: MEKF pointing performance in Pitch Sun Tracking mode.

Scenario	e_{point} [deg]			$\|\|\Delta\omega_{point}\|\|$ [rad/s]		
	Max	Mean	IAE [deg s]	Max	Mean	IAE [rad]
1	21.2	8.03	6.65×10^{5}	9.97×10^{-4}	3.23×10^{-4}	26.76
2	33.0	9.35	7.77×10^{5}	1.48×10^{-3}	3.48×10^{-4}	28.87
3	18.4	8.38	6.94×10^{5}	9.19×10^{-4}	3.18×10^{-4}	26.37
4	21.0	7.61	6.33×10^{5}	1.02×10^{-3}	2.74×10^{-4}	22.80
5	27.2	9.78	8.16×10^{5}	1.29×10^{-3}	3.76×10^{-4}	31.42

In this case, as expected, the pointing error is much higher, because of the more stringent requirement on angular velocity (see Section 5.2). Indeed, comparing the angular rate pointing error of Fig. 7.12b with the one of Fig. 7.12a, it emerges that in PST mode the controller is only able to maintain a rate pointing error that is twice the one of the NP mode. In fact, in Table 7.8 the IAE value of $||\Delta\boldsymbol{\omega}_{point}||$ for all the scenarios is more than 10 rad/s higher than in the NP scenarios (Table 7.6). Moreover, the periodicity is still respected, but with many more oscillations, in particular in scenarios 2, 3 and 5.

Because in PST mode there is no requirement on the pointing accuracy, it would be pointless to establish a convergence time of the controller. Nonetheless, as is visible is Fig. 7.11, in all the scenarios the pointing error seems to reach the first peak at a similar time, after less than 1.5 orbit periods.

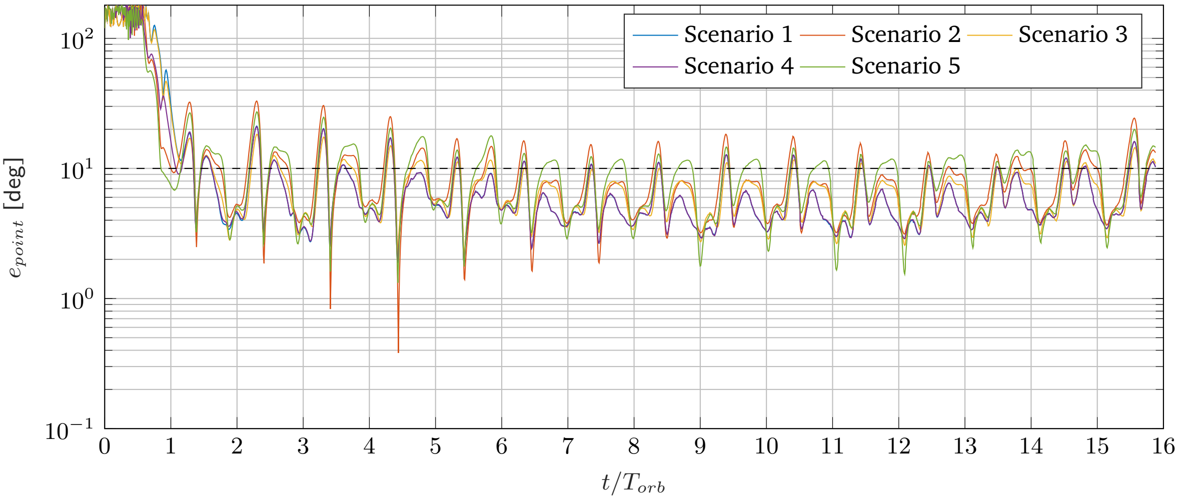
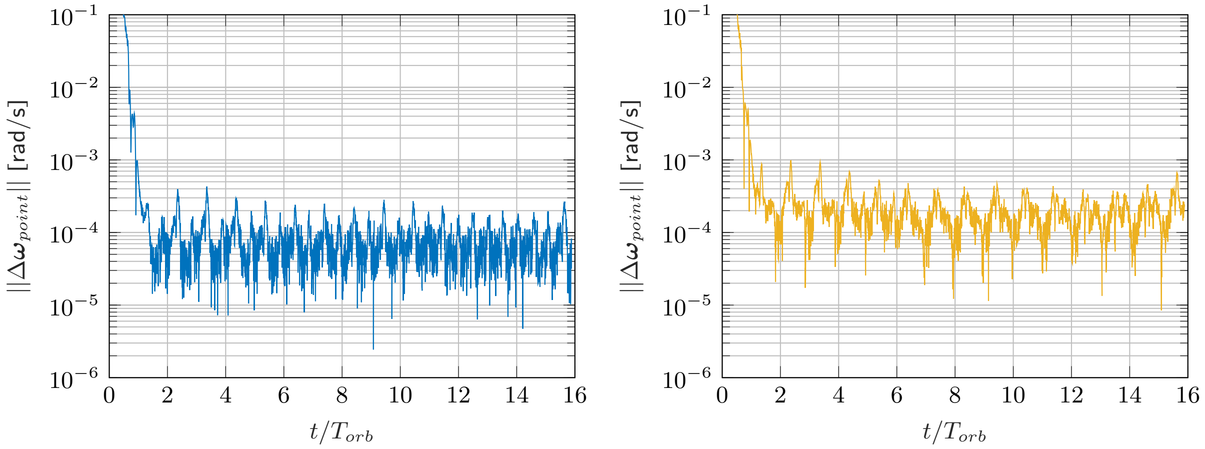


Figure 7.11: PST mode pointing accuracy.



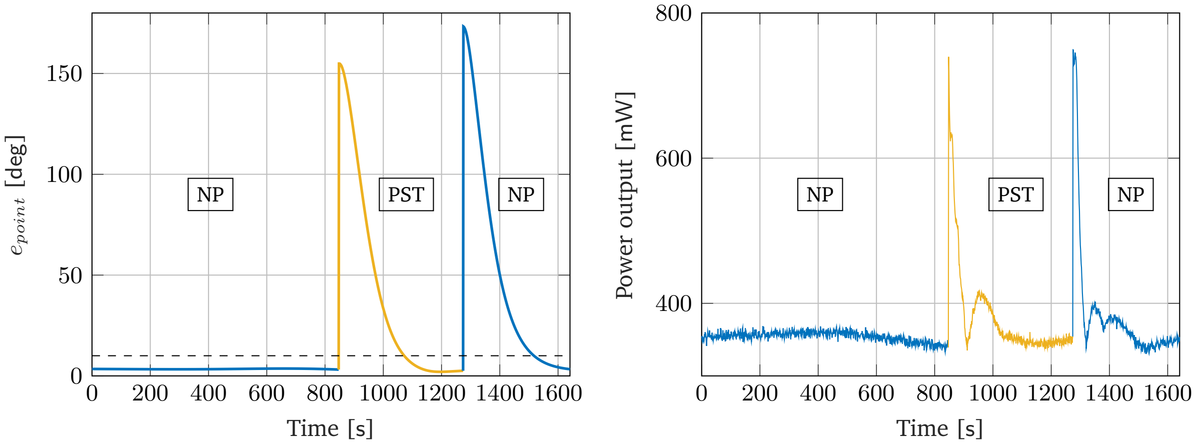
(a) NP mode.

(b) PST mode.

Figure 7.12: Angular rate pointing error for the two attitude modes.

7.5.2 Transition Between Pointing Modes

To assess the practicality of executing the PST attitude mode, a simulation was run in which, after achieving Nadir Pointing, the commanded state is changed into the one required by the PST mode. Fig. 7.13a shows that the time required to acquire the PST mode with a pointing error below 10° is 223 s. Afterwards, the commanded state is changed back into the NP one and the controller requires another very short time period, i.e. 250 s, to reduce the pointing error down to 10°. In the particular scenario considered, the pointing direction of the NP mode (the Nadir direction) and the one of the PST mode (the projection of the Sun vector on the orbital plane) are almost 180° apart, so the two maneuvers can be considered to be happening in the worst case scenario.



(a) Pointing error in switching between modes.

(b) Power required in switching between modes.

Figure 7.13: Mode transition simulation.

In Fig. 7.13b the power required by the actuators is plotted for the desired maneuver. When either mode transition happens, a spike appears, making the power peak at 751 mW. This is the increased load that the magnetorquers require in order to quickly rotate the spacecraft from one direction to the other. The power required is quickly brought down to the nominal value of around 350 mW because the pointing stability is rapidly reached.

This result confirms the versatility of the designed CGLQR controller, which can easily function using the same gain matrix for both the NP and the PST mode. Moreover, it gives reassurances on the feasibility of execution of the PST mode, which can be quickly employed as a safe mode when a low power input is foreseen.

7.6 Power Budget Analysis

In this section the results of the simulations in terms of power budget of the mission are presented. In Fig. 7.14 the power availability to the solar panels is compared over four of the central orbits for the two attitude modes. The behaviour of the two power curves is periodic with period equal to the orbital period, so the central orbits are as good a sample as any to illustrate the comparison.

From the previous discussion it resulted that the MEKF estimator is more reliable than the AEKF, so the former was used to obtain the result of this section. Also MEKF is less sensible to the considered perturbation. For this reason, it is appropriate to assume that the power analysis is rather independent from the considered scenario. Thus, even if the results of Table 7.9 are obtained from scenario 1, they can be considered valid in a general sense.

The detailed simulations of this section confirm what was anticipated in Chapter 6. In Table 7.9 the

Table 7.9: Power budget comparison over 1 day of simulation.

Mode	P_{in} [W]			P_{out} [mW]	
	max	mean	min	max	mean
NP	6.940	5.785	3.886	420	354
PST	7.018	6.874	6.598	480	363
Difference	0.078	1.089	2.712	60	9

same type of results presented in Table 6.3 are shown, with the difference that in this case the simulation takes into account all the disturbances and imperfections that come from an ADCS subsystem. As in Section 6.2, the mean operation is performed only on the period the satellite spends in the sunlight. It is proven that the main goal of the PST mode, which is to increase the mean power availability of the solar panels, is fully achieved. In fact, one can appreciate an improvement in the mean power availability of more than 1 W and in its minimum value of 2.7 W.

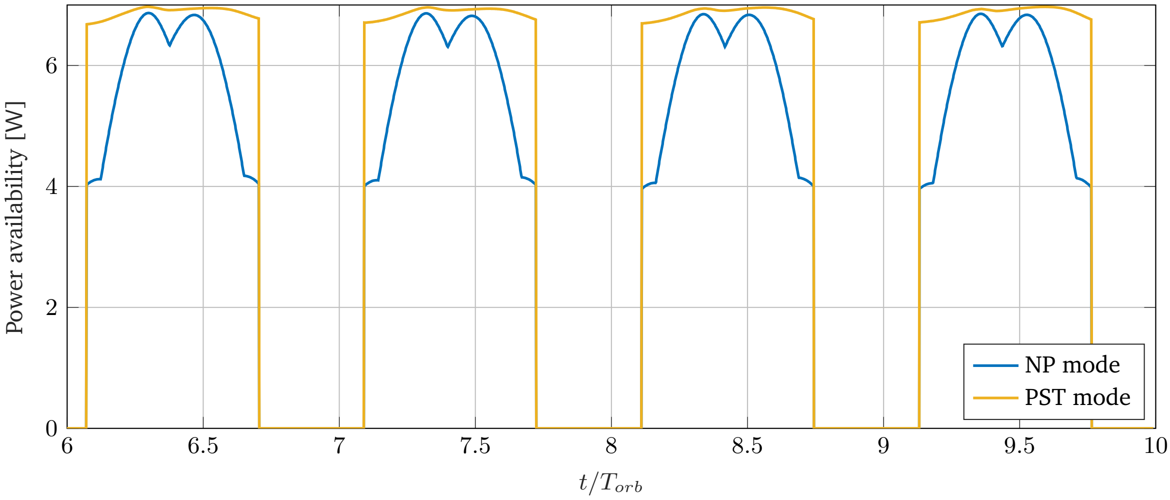


Figure 7.14: Comparison of the power availability for the two pointing modes.

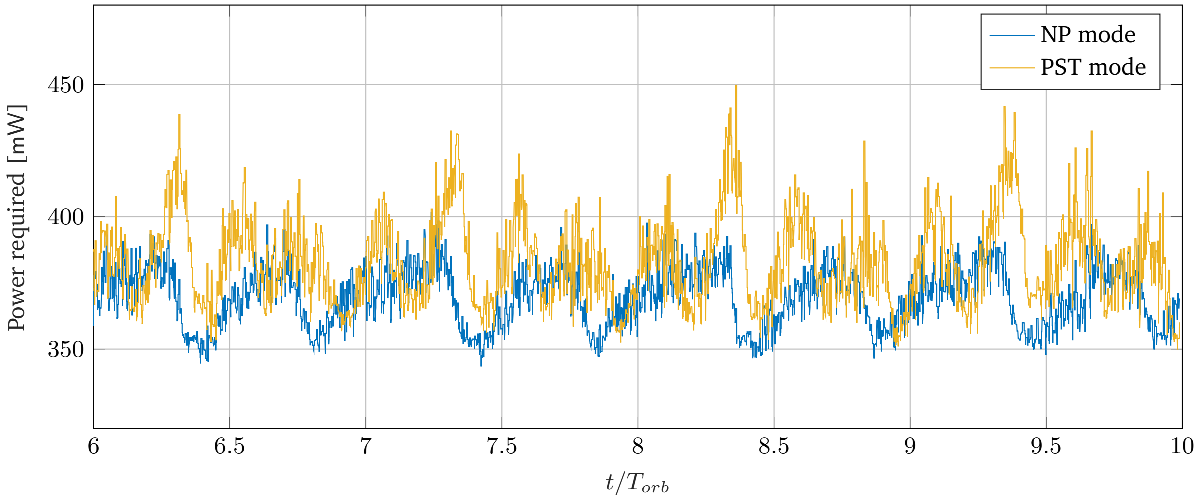


Figure 7.15: Comparison of the power required by the two pointing modes.

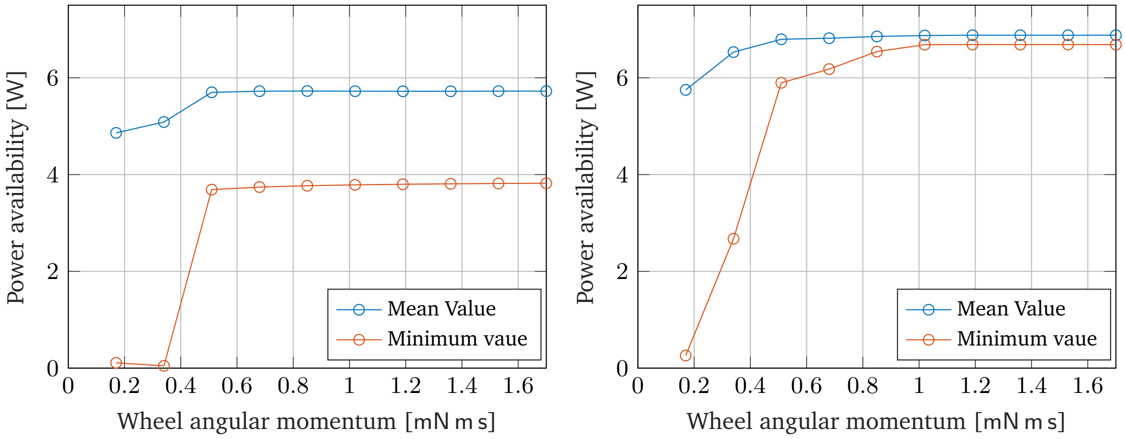
One aspect that could not be investigated in the preliminary analysis of Chapter 6 is the comparison of the power required by the actuators in the two cases. This parameter is at least as significant as the power production, because its study assures that there is no important increase in terms of energy requirements when actuating the Pitch Sun Tracking attitude mode. For the substantial increase in power availability described before, the mean power required by momentum wheel and magnetorquers increases by just 9 mW ($< 1\%$). It is true that the magnetorquers undergo an increased stress, because in their highest peaks, observable in Fig. 7.15, they need 60 mW more than what they need during the NP mode, but still this is a small price to pay for the advantage that can be achieved.

7.6.1 Momentum Bias Analysis

A parameter that significantly influences the power budget of the mission is the value of the momentum bias provided by the momentum wheel. As stated in Section 3.2.2, the CubeWheel maximum angular momentum is 1.7 mN m s, but the wheel can also operate at different angular rates.

The power available at the solar panels is plotted in Fig. 7.16 while the power consumed by the ADCS is presented in Fig. 7.17. The plots present the power values as a function of the momentum bias in both the considered operational attitude modes. The discussion of the results can be applied both to the PST and to the NP mode, because it is clear from the figures that the behaviour of the function is very similar for the two cases.

For the PST mode the mean value of the power available to the solar panels steeply increases up to the momentum bias point of 1 mN m s, above which it increases with a very small slope; the same can be said for the NP mode, for which the threshold value is around 0.7 mN m s. For both the attitude modes, instead, the value of the power consumed by the actuators increases linearly with the momentum bias, because the power needed by the wheel to win the viscous friction is proportional to its angular rate, as seen in Section 3.2.2.



(a) NP mode.

(b) PST mode.

Figure 7.16: Power availability sensibility to the momentum bias.

It is worth recalling that, when operating in PST attitude mode, the ADCS does not have any quantitative pointing requirement, but it only aims at maximizing the power budget. So, it is relevant to notice from Fig. 7.16b that the difference in mean power availability between the maximum operating point of the wheel (1.7 mN m s) and the one at 1 mN m s is 1.1 mW, whereas the difference in mean power consumed by the actuators is 80.3 mW, as seen in Fig. 7.17. This means that, in addition to the power budget improvement given by the PST mode with respect to the NP mode, operating the wheel at a lower speed ensures further savings in terms of power.

For the NP mode the pointing error requirement of 10° is more important than the optimization of the power budget. So, as clearly visible in Fig. 7.18a, it is sufficient to adopt a momentum bias of 0.83 mN m s, for which the maximum pointing error is 7.52°. The power available, as appears from Fig. 7.16a, is very little sensitive to the momentum bias after the threshold of 0.83 mN m s, so the choice of this value is justified[1].

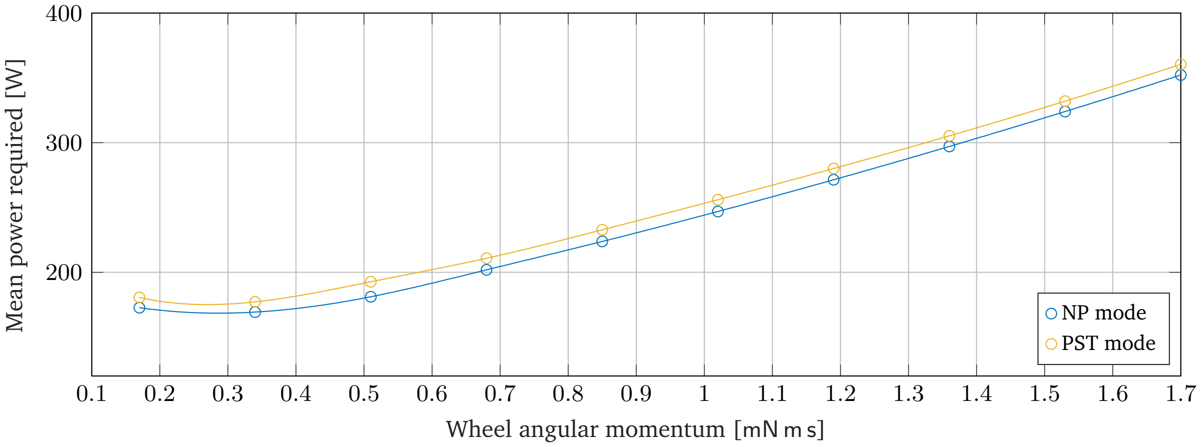
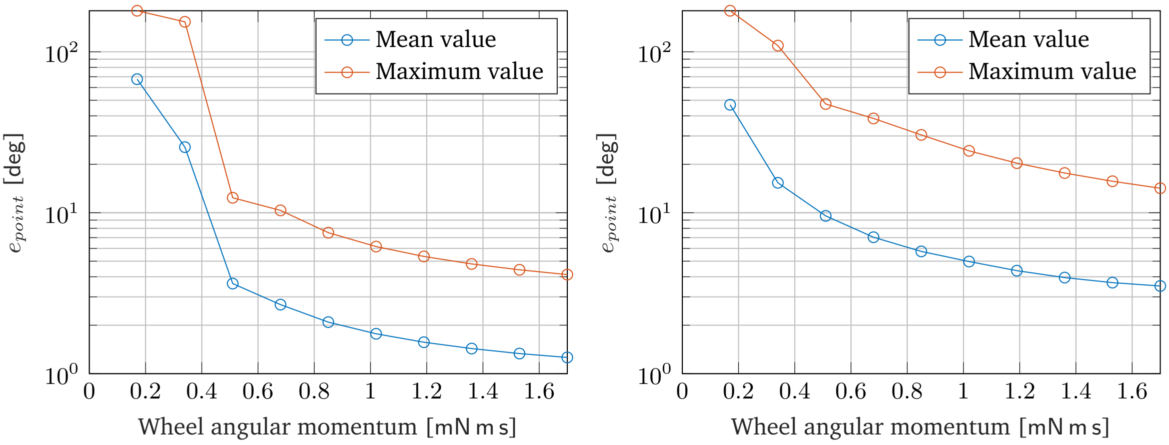


Figure 7.17: Mean power output sensibility to the momentum bias.



(a) NP mode.

(b) PST mode.

Figure 7.18: Pointing error sensibility to the momentum bias.

[1] The values for the pointing accuracy differ from the ones presented in Section 7.4.3 and Section 7.5.1 because in this section the value is taken after the pointing accuracy is already brought beneath the requirement (after some orbits), while in the previous ones the first peak of the pointing accuracy function was considered for the maximum value of e_{point}.

Chapter 8

Conclusions

This MSc thesis had two main goals: the numerical validation of CubeADCS Y-Momentum by CubeSpace and the development of a new Pitch Sun Tracking (PST) attitude mode for the 2U ORCASat. The ADCS needed to prove capable to encounter the mission requirements in terms of detumbling capability, estimation accuracy (2°) and Nadir Pointing (NP) accuracy (10°). Moreover the study wanted to assess the advantages that the PST mode would bring to the mission. A first qualitative analysis showed that this new attitude mode would be capable of increasing the minima in power production over the lifetime of the mission by more than 0.7 W (27%).

In order to study the advantages in a quantitative way and to ensure that the ADCS could respect the mission requirements, a model of the satellite and of the ADCS had to be developed. CubeADCS Y-Momentum, presents a typical set of sensors for a Nadir Pointing mission: ten coarse Sun sensors, a tri-axial magnetometer, three MEMS gyros and a fine Sun sensor. The actuation is performed by three magnetorquers, aided by a pitch momentum wheel that stabilizes the attitude motion in the orbital xz plane. For the power production, the satellite is covered with solar arrays made of AzurSpace cells (BOL efficiency of 29.6%), which can provide a peak power of 2.4 W per unit of CubeSat surface.

CubeADCS Y-Momentum offers by default five attitude estimation modes and two types of attitude modes. The first two estimation modes, Gyro Rates estimation and Magnetic Rate Extended Kalman Filter (MREKF), are only able to estimate the angular rates, so they are mainly used during the detumbling phase or if the attitude knowledge is not required. The third one, TRIAD, is computationally very fast, but it only functions when the satellite is in view of the Sun, so it cannot be safely employed during normal operation. The last two, AEKF and MEKF, are two Extended Kalman Filters. While both of them were proven to be able to maintain the attitude estimation error below the requirement, MEKF was found to be more robust than AEKF, thus the analysis of the pointing accuracy was performed utilizing the data from the simulations adopting the former.

The pointing accuracy was evaluated using five scenarios, which represent the uncertainty of some design parameters, i.e. the tensor of inertia, the parasitic magnetic dipole moment and the momentum bias given by the wheel. The employed Constant Gain Linear Quadratic Regulator controller was shown to be capable of keeping the NP pointing error below the requirement in all the scenarios, whereas in the PST scenarios it maintained a pointing accuracy more than 10° higher ($> 100\%$), due to the more demanding angular rate command.

The comparison of the results of the simulations on the power budget analysis for the two pointing modes confirmed what was initially demonstrated via simplified simulations: over the period of 1 day, the PST mode was shown to be able to increase the average power availability by more than 17% of the value of the NP mode (1 W), while it required a peak increase in the power needed by the actuators of just 60 mW (6% of the increase in power availability). Hence, it has been numerically shown that it is worth adopting the PST mode when a low power income is predicted.

Each of the four detumbling modes offered by the ADCS was found to perform acceptably when starting from the maximum angular rate they were declared to be able to detumble by CubeSpace.

8.1 Future Work

This thesis is the third of a series of works performed by students of Instituto Superior Técnico on the numerical simulation of the ADCS of ORCASat. The developed simulation models within the MATLAB® Simulink® environment achieved a high level of detail and proved to accurately simulate the dynamics. Since the ADCS was acquired Off-The-Shelf, the need to convert the software into a low-level programming language is significantly reduced with respect to the built-in ADCS case, because the code is already written inside the ACP of CubeComputer. Nonetheless, since only the NP mode is defined by default in the OBC, it would be necessary to add the definition of the pointing command relative to the PST, if the new attitude mode is to be tested and eventually employed during the operational life of ORCASat.

A second interesting development, successive to the implementation of the PST mode, would be to design an Hardware-In-the-Loop (HIL) test bed for the ADCS hardware and eventually perform HIL simulations. This process is not something that is usually performed on small CubeSats, but it would provide very useful data to the ORCASat team and it would start a valuable know-how for future UVic CubeSat teams.

Bibliography

[1] H. Heidt, J. Puig-suari, A. S. Moore, S. Nakasuka, and R. J. Twiggs, “CubeSat: A new Generation of Picosatellite for Education and Industry Low-Cost Space Experimentation,” in *Small Satellite Conference*, (Logan, Utah, USA), 2000.

[2] M. Swartwout, “The first one hundred CubeSats: A statistical look,” *Journal of Small Satellites*, vol. 2, no. 2, pp. 213–233, 2013.

[3] A. Poghosyan and A. Golkar, “CubeSat evolution: Analyzing CubeSat capabilities for conducting science missions,” *Progress in Aerospace Sciences*, vol. 88, pp. 59–83, 2017.

[4] T. Villela, C. A. Costa, A. M. Brandão, F. T. Bueno, and R. Leonardi, “Towards the thousandth CubeSat: A statistical overview,” *International Journal of Aerospace Engineering*, 2019.

[5] D. Selva and D. Krejci, “A survey and assessment of the capabilities of Cubesats for Earth observation,” *Acta Astronautica*, vol. 74, pp. 50–68, 2012.

[6] E. Peral, E. Im, L. Wye, S. Lee, S. Tanelli, Y. Rahmat-Samii, S. Horst, J. Hoffman, S. H. Yun, T. Imken, and D. Hawkins, “Radar Technologies for Earth Remote Sensing from CubeSat Platforms,” *Proceedings of the IEEE*, vol. 106, no. 3, pp. 404–418, 2018.

[7] W. A. Shiroma, L. K. Martin, J. M. Akagi, J. T. Akagi, B. L. Wolfe, B. A. Fewell, and A. T. Ohta, “CubeSats: A bright future for nanosatellites,” *Central European Journal of Engineering*, vol. 1, no. 1, pp. 9–15, 2011.

[8] H. R. Goldberg, O. Karatekin, B. Ritter, A. Herique, P. Tortora, C. Prioroc, B. G. Gutierrez, P. Martino, and I. Carnelli, “The Juventas CubeSat in Support of ESA’s Hera Mission to the Asteroid Didymos,” in *33rd Annual AIAA/USU Conference on Small Satellites*, (Logan, Utah, USA), 2019.

[9] T. Kohout and J.-E. Wahlund, “Asteroid Prospection Explorer (APEX) CubeSat for Hera mission Didymos Interior,” in *EPSC-DPS Joint Meeting*, (Geneva, Switzerland), 2019.

[10] M. N. Sweeting, “Modern Small Satellites-Changing the Economics of Space,” *Proceedings of the IEEE*, vol. 106, no. 3, pp. 343–361, 2018.

[11] C. Nieto-Peroy and M. R. Emami, “CubeSat mission: From design to operation,” *Applied Sciences (Switzerland)*, vol. 9, no. 15, pp. 1–24, 2019.

[12] A. Doknjas, R. Arthurs, and L. Buzas, "Concept of Operations for ORCASat," tech. rep., CfAR, UVic, 2019.

[13] J. E. Albert, M. H. Fagin, Y. J. Brown, C. W. Stubbs, N. A. Kuklev, and A. J. Conley, "Precision Calibration via Artificial Light Sources Above the Atmophere," in *ASP Conference*, (New York, NY, USA), 2012.

[14] J. E. Albert, "Satellite mounted light sources as photometric calibration standards for ground-based telescopes," *Astronomical Journal*, vol. 143, no. 1, 2012.

[15] G. Sterling, A. Cheng, M. Caverley, D. Sood, and D. Came, "Design of radio telescope calibration payload for a nano-satellite," *Proceedings of the 13th ASCE Aerospace Division Conference and the 5th NASA/ASCE Workshop on Granular Materials in Space Exploration*, pp. 355–364, 2012.

[16] B. Lobo-Fernandes, "Design and Analysis of an Attitude Determination and Control System for the ORCASat Nanosatellite," Master's thesis, Dept. Mech. Eng., IST, Lisbon, Portugal, 2019.

[17] B. Sabino, "Attitude Determination and Control System of the ORCASat," Master's thesis, Dept. Mech. Eng., IST, Lisbon, Portugal, 2020.

[18] H. D. Curtis, *Orbital Mechanics for Engineering Students*. Kundli, Haryana, India: Butterworth-Heinemann, third ed., 2014.

[19] W. J. Larson and J. R. Wertz, *Space mission analysis and design*. Space Techonology, El Segundo, CA, USA: Microcosm Press and Kluwer Academic Publishers, 1992.

[20] P. C. Hughes, *Spacecraft Attitude Dynamics*. Mineola, NY, USA: Dover Publications, 2004.

[21] W. R. Hamilton, "On Quaternions , or on a New System of Imaginaries in Algebra," *The London, Edinburgh and Dublin Philosophical Magazine and Journal of Science*, vol. 25, no. 3, pp. 489–495, 1844.

[22] K. Grossekatthöfer and Z. Yoon, "Introduction into quaternions for spacecraft attitude representation." TU Berlin Department of Astronautics and Aeronautics Berlin, Germany, 2012.

[23] F. L. Markley and J. L. Crassidis, *Fundamentals of spacecraft attitude determination and control*. El Segundo, CA, USA: Microcosm Press and Springer, 2014.

[24] R. Takwale, *Introduction to Classical Mechanics*. New Delhi, India: Tata McGraw-Hill, 1980.

[25] B. D. Tapley, B. E. Schutz, and G. H. Born, "The Orbit Problem," *Statistical Orbit Determination*, vol. 2, pp. 17–91, 2004.

[26] J. R. Wertz, *Spacecraft attitude determination and control*. Torrance, CA, USA: Kluwer Academic Publishers, 1990.

[27] N. K. Pavlis, S. A. Holmes, S. C. Kenyon, and J. K. Factor, "The development and evaluation of the Earth Gravitational Model 2008 (EGM2008)," *Journal of Geophysical Research: Solid Earth*, vol. 117, no. 4, pp. 1–38, 2012.

[28] D. L. Oltrogge and K. Leveque, "An evaluation of CubeSat orbital decay," in *25th Annual AIAA/USU Conference on Small Satellites*, (Logan, Utah, USA), 2011.

[29] J. M. Picone, A. E. Hedin, D. P. Drob, and A. C. Aikin, "NRLMSISE-00 empirical model of the atmosphere: Statistical comparisons and scientific issues," *Journal of Geophysical Research: Space Physics*, vol. 107, no. A12, pp. 1–16, 2002.

[30] A. Doknjas and B. De Lobo, "ORCASat Preliminary Design Review," tech. rep., CfAR, UVic, 2019.

[31] Azur Space, "30% triple junction gaas solar cell assembly," 2016.

[32] CubeSpace, *CubeADCS - Interface Control Document V3.15*, 2019.

[33] "State of the Art Small Spacecraft Technology.," Tech. Rep. December, NASA, 2018.

[34] S. Dussy, D. Durrant, T. Moy, N. Perriault, and C. Bruno, "MEMS gyro for space applications," in *AIAA Guidance, Navigation, and Control conference*, (San Francisco, CA, USA), pp. 1–12, 2005.

[35] H. Xie and G. K. Fedder, "Integrated microelectromechanical gyroscopes," *Journal of Aerospace Engineering*, vol. 16, no. 2, pp. 65–75, 2003.

[36] V. Apostolyuk, "Theory and design of micromechanical vibratory gyroscopes," in *MEMS/NEMS Handbook*, vol. 1, pp. 173–195, Springer, 08 2007.

[37] Q. M. Lam, N. Stamatakos, C. Woodruff, and S. Ashton, "Gyro modeling and estimation of its random noise sources," in *AIAA Guidance Navigation and Control Conference and Exhibit*, (Austin, TX, USA), 2003. ISBN: 9781563479786.

[38] CubeSpace, *CubeADCS - User Manual V3.08*, 2019.

[39] F. Martel, P. K. Pal, and M. Psiaki, "Three Axis Attitude Determination via Kalman Filtering of Magnetometer Data," in *Flight Mechanics/Estimation Theory Symposium*, (Greenbelt, MD, USA), 1988.

[40] M. Psiaki and F. Martel, "Autonomous Magnetic Navigation for Earth Orbiting Spacecraft," in *AIAA/USU Conference on Small Satellites*, (Logan, Utah, USA), 1989.

[41] CubeSpace, *CubeSense V3 - Interface Control Document V1.2*, 2019.

[42] P. Wang, Y. B. Shtessel, and Y.-q. Wang, "Satellite Attitutde Control Using only Magnetorquers," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (Boston, USA), 1998.

[43] A. R. Walker, P. T. Putman, and K. Cohen, "Solely magnetic genetic/fuzzy-attitude-control algorithm for a CubeSat," *Journal of Spacecraft and Rockets*, vol. 52, no. 6, pp. 1627–1639, 2015.

[44] D. M. Torczynski, R. Amini, and P. Massioni, "Magnetorquer based attitude control for a nanosatellite testplatform," *AIAA Infotech at Aerospace 2010*, no. April, 2010.

[45] V. Carrara, H. K. Kuga, and A. G. Silva, “A dynamic friction model for reaction wheels,” in *IAA Conference on Dynamics and Control of Space Systems*, (Porto, Portugal), pp. 1–12, 2012.

[46] R. Fulcher, “A brushless DC-torquer-driven reaction wheel for spacecraft attitude control,” Tech. Rep. July, Goddard Space Flight Center, Goddard Space Filght Center, MD, USA, 1969.

[47] V. Carrara and H. K. Kuga, “Estimating friction parameters in reaction wheels for attitude control,” *Mathematical Problems in Engineering*, 2013.

[48] Y. H. Jia, S. J. Xu, and L. Tang, “Bias momentum attitude control system using energy/momentum wheels,” *Chinese Journal of Aeronautics*, vol. 17, no. 4, pp. 193–199, 2004.

[49] K. C. Liu, P. Maghami, and C. Blaurock, “Reaction wheel disturbance modeling, jitter analysis, and validation tests for Solar Dynamics Observatory,” in *AIAA Guidance, Navigation and Control Conference and Exhibit*, (Honolulu, HI, USA), 2008.

[50] A. Shirazi and M. Mirshams, “Design and performance simulation of a satellite momentum exchange actuator,” *Australian Journal of Mechanical Engineering*, vol. 14, no. 1, pp. 1–9, 2016.

[51] Q. Wan, G. Liu, H. Shi, X. Zhang, and X. Ning, “Analysis of dynamic characteristics for momentum wheel assembly with joint clearance,” *Advances in Mechanical Engineering*, vol. 10, no. 12, pp. 1–12, 2018.

[52] H. D. Black, “A passive system for determining the attitude of a satellite,” *AIAA Journal*, vol. 2, no. 7, pp. 1350–1351, 1964.

[53] F. L. Markley, “Attitude determination using vector observations and the singular value decomposition,” *Journal of the Astronautical Sciences*, vol. 38, November 1987.

[54] S. Tanygin and M. D. Shuster, “The many TRIAD algorithms,” *Advances in the Astronautical Sciences*, vol. 127, pp. 81–100, 2007.

[55] M. D. Shuster, “The optimization of TRIAD,” *Journal of the Astronautical Sciences*, vol. 55, no. 2, pp. 245–257, 2007.

[56] M. Ovchinnikov and D. Ivanov, “Approach to study satellite attitude determination algorithms,” *Acta Astronautica*, vol. 98, no. 1, pp. 133–137, 2014.

[57] C. E. Hutchinson, “The Kalman Filter Applied to Aerospace and Electronic Systems,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-20, no. 4, pp. 500–504, 1984.

[58] S. M. Marques, “Small satellites attitude determination methods,” Master’s thesis, Dept. Mech. Eng., IST, Lisbon, Portugal, 2000.

[59] T. E. Humphreys, M. L. Psiaki, E. M. Klatt, S. P. Powell, and P. M. Kintner, “Magnetometer-based attitude and rate estimation for spacecraft with wire booms,” *Journal of Guidance, Control, and Dynamics*, vol. 28, no. 4, pp. 584–593, 2005.

[60] D. Li, P. Harkness, and T. Walkinshaw, “Design of a solar panel deployment and tracking system for PocketQube pico-satellite,” *IEEE Aerospace Conference Proceedings*, no. August, pp. 4–11, 2017.

[61] L. B. Li, M. X. Li, L. X. Jiang, D. Y. Wang, F. Zhan, and T. Sheng, “Angular rate estimation and damping control of satellite with magnetometer data,” *Optik*, vol. 180, no. November, pp. 1049–1055, 2019.

[62] G. A. Natanson, S. F. Mclaughlin, and R. C. Nicklas, “A method of determining attitude from magnetometer data only,” in *Flight Mechanics Estimation Theory Symposium*, (Greenbelt, MD, USA), pp. 359–378, 1990.

[63] G. a. Natanson, M. S. Challa, J. Deutschmann, and D. F. Baker, “Magnetometer Only Attitude and Rate Determination for a Gyro-Less Spacecraft,” *Space Mission Operations and Ground Data Systems*, vol. 1, pp. 791–798, 1994.

[64] B. Wie, *Space vehicle dynamics and control*. Ames, Iowa, USA: American Institute of Aeronautics and Astronautics, 2008.

[65] F. L. Markley, “Multiplicative vs. Additive Filtering for Spacecraft Attitude Determination Quaternion estimation,” *Journal of Guidance, Control, and Dynamics*, vol. 26, no. 2, pp. 311–317, 2003.

[66] C. Hajiyev, D. Cilden, and Y. Somov, “Gyro-free attitude and rate estimation for a small satellite using SVD and EKF,” *Aerospace Science and Technology*, vol. 55, pp. 324–331, 2016.

[67] C. Hajiyev and D. Cilden Guler, “Review on gyroless attitude determination methods for small satellites,” *Progress in Aerospace Sciences*, vol. 90, no. March, pp. 54–66, 2017.

[68] H. Nor Hazadura, Y. Sazali, M. Hariharan, and H. Norhizam, “Nonlinear observers for attitude estimation in gyroless spacecraft via Extended Kalman filter algorithm,” *International Journal of Scientific and Research Publications*, vol. 4, no. 10, pp. 1–9, 2014.

[69] A. M. S. I. Mohammed, M. Benyettou, A. Boudjemai, and Y. Hashida, “Seven State Kalman Filtering for LEO Microsatellite,” *Recent Advance in Signal Processing, Robotics and Automation*, no. January, 2010.

[70] J. Lousada, “Design and development of the ECOSat’s Attitude Determination and Control System (ADCS) onboard software,” Master’s thesis, Dept. Mech. Eng., IST, Lisbon, Portugal, 2013.

[71] F. Qin, L. Chang, S. Jiang, and F. Zha, “A sequential multiplicative extended kalman filter for attitude estimation using vector observations,” *Sensors (Switzerland)*, vol. 18, no. 5, 2018.

[72] M. D. Pham, K. S. Low, S. T. Goh, and S. Chen, “Gain-scheduled extended kalman filter for nanosatellite attitude determination system,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 2, pp. 1017–1028, 2015.

[73] O. Awad, “A Multiplicative Extended Kalman Filter for Low Earth Orbit Attitude Estimation Aboard a 0.5U SmallSat,” *AAS Journal*, no. January, 2020.

[74] N. Filipe, M. Kontitsis, and P. Tsiotras, “Extended Kalman Filter for spacecraft pose estimation using dual quaternions,” *Journal of Guidance, Control, and Dynamics*, vol. 38, no. 9, pp. 1625–1641, 2015.

[75] D. Rondão, “Modeling and Simulation of the ECOSat-III Attitude Determination and Control System,” Master’s thesis, Dept. Mech. Eng., IST, Lisbona, Portugal, 2016.

[76] G. Avanzini and F. Giulietti, “Magnetic detumbling of a rigid spacecraft,” *Journal of Guidance, Control, and Dynamics*, vol. 35, no. 4, pp. 1326–1334, 2012.

[77] R. Fonod and E. Gill, “Magnetic detumbling of fast-tumbling picosatellites,” *Proceedings of the International Astronautical Congress, IAC*, vol. October, 2018.

[78] CubeSpace, *CubeADCS - Firmware Reference Manual V3.12*, 2019.

[79] L. R. Lustosa, F. Cardoso-Ribeiro, F. Defay, and J. M. Moschetta, “A new look at the uncontrollable linearized quaternion dynamics with implications to LQR design in underactuated systems,” in *European Control Conference*, (Limassol, Cyprus), 2018.

[80] Y. Yang, “Analytic LQR Design for Spacecraft Control System Based on Quaternion Model,” *Journal of Aerospace Engineering*, vol. 25, no. 3, pp. 448–453, 2012.

[81] Y. Yang, “Quaternion based model for momentum biased nadir pointing spacecraft,” *Aerospace Science and Technology*, vol. 14, no. 3, pp. 199–202, 2010.

[82] Y. Yang, “Quaternion-Based LQR spacecraft control design is a robust pole assignment design,” *Journal of Aerospace Engineering*, vol. 27, no. 1, pp. 168–176, 2014.

[83] R. Wiśniewski, *Satellite Attitude Control Using Only Electromagnetic Actuation*. PhD Thesis, Dept. Contr. Eng., Aalborg University, Aalborg, Denmark, 1996.

[84] Salahudden, P. Kumar, V. S. Dwivedi, D. K. Giri, and A. K. Ghosh, “Quaternion Based Optimal Controller for Momentum Biased Nadir Pointing Satellite,” in *IEEE Aerospace Conference*, (Big Sky, MT, USA), 2020.

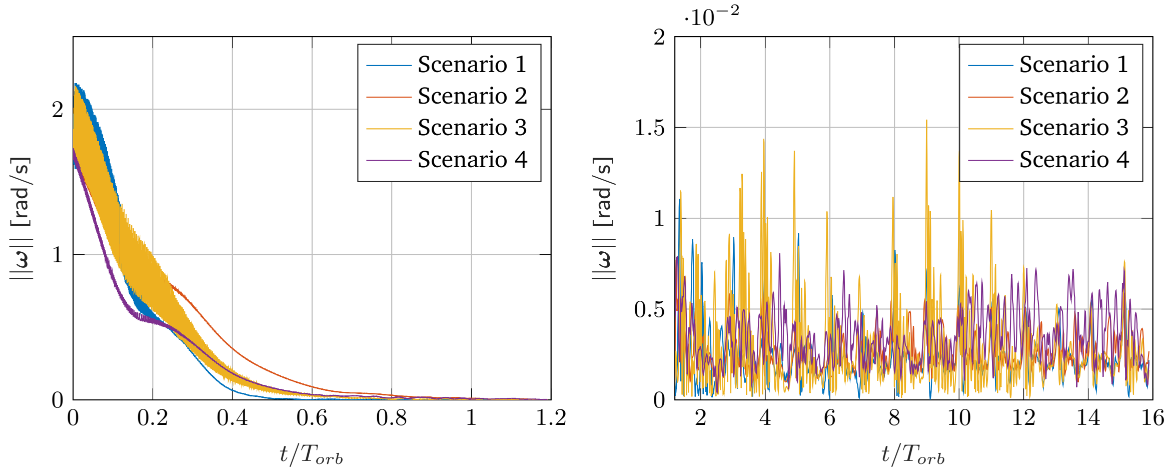
[85] M. Athans and P. L. Falb, *Optimal control: An introduction to the theory and its applications*. Mineola, NY, USA: Dover Publications, 2006.

[86] X. Liu, Y. Lu, and Y. Yin, “Conceptual design of an electromagnetic detumbling system made by permanent magnet array,” *IEEE Aerospace Conference Proceedings*, vol. March, pp. 1–7, 2018.

[87] Z. Xie, P. Gong, M. Liang, Q. Zhang, and H. Liu, “Using permanent magnetic array device to detumble and despin defunct spacecraft,” *IOP Conference Series: Materials Science and Engineering*, vol. 631, 2019.

Appendix A

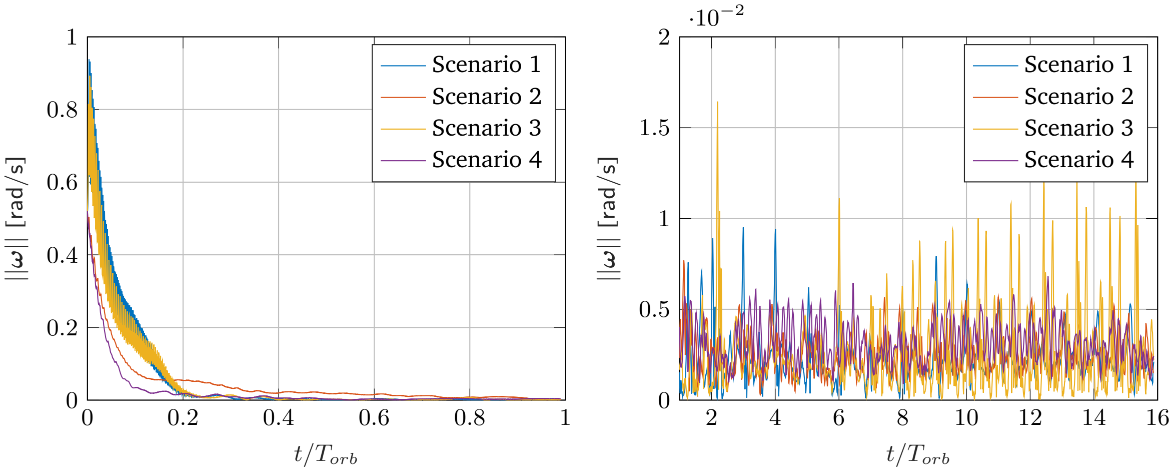
Additional Simulation Plots



(a) Transient state.

(b) Steady state.

Figure A.1: From Section 7.2: Fast detumble mode angular rate evolution.



(a) Transient state.

(b) Steady state.

Figure A.2: From Section 7.2: Y-Thomson detumble mode angular rate evolution.

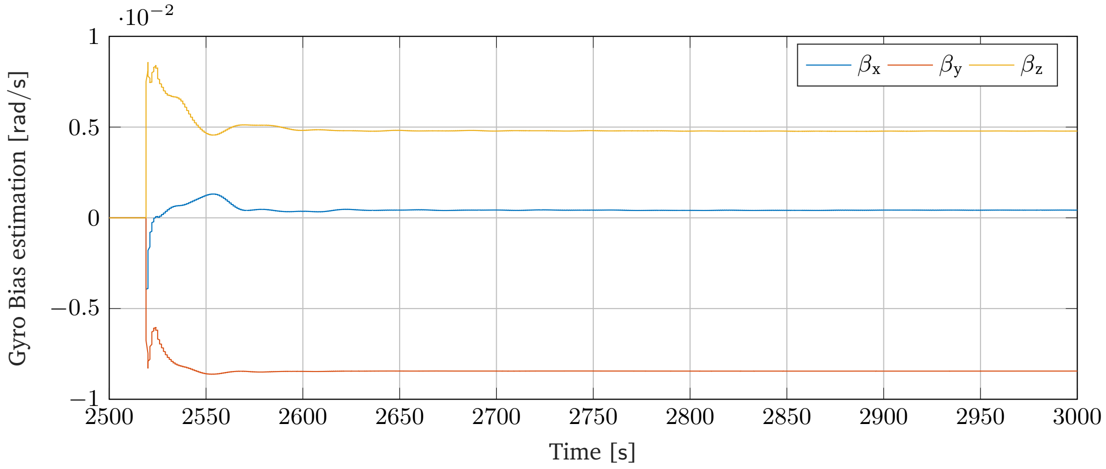
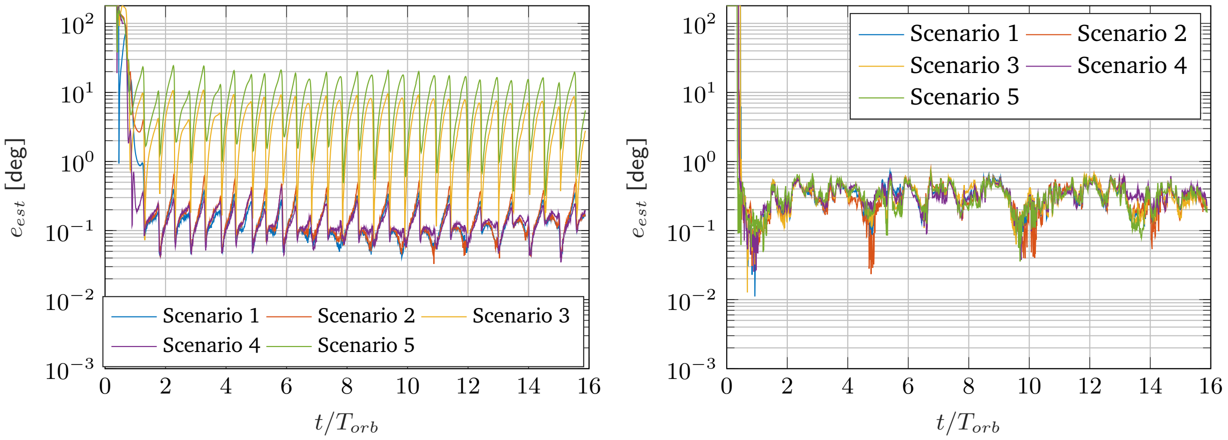


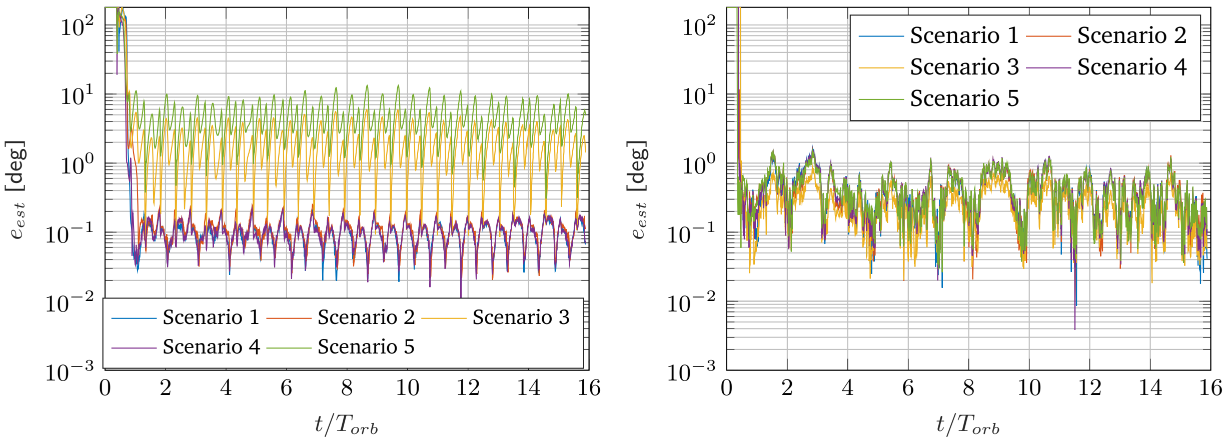
Figure A.3: From Section 7.4.2: Estimation of the Gyro bias during the MEKF estimation mode. The algorithm is very fast in stabilizing the estimation of the bias on the three axes around the correct value.



(a) AEKF: the estimation of scenarios 3 and 5 is not acceptable.

(b) MEKF: the estimation accuracy is rather independent from the scenario

Figure A.4: From Section 7.4.2: NP mode attitude estimation error.



(a) AEKF

(b) MEKF

Figure A.5: From Section 7.5: PST mode attitude estimation error.

Appendix B

Hardware Models

In this appendix, the MATLAB® Simulink® models for the hardware of the spacecraft are presented. Also, for each model the values of the parameters that influence the behaviour of the simulated hardware are included in a table.

Fine Sun Sensor

Table B.1: Fine Sun Sensor parameters.

Parameter	**Unit**	**Value**
Update rate	[s]	0.1
Noise standard deviation	[deg]	0.2
FOV	[deg]	180
Sensitivity	[LSB/deg]	8

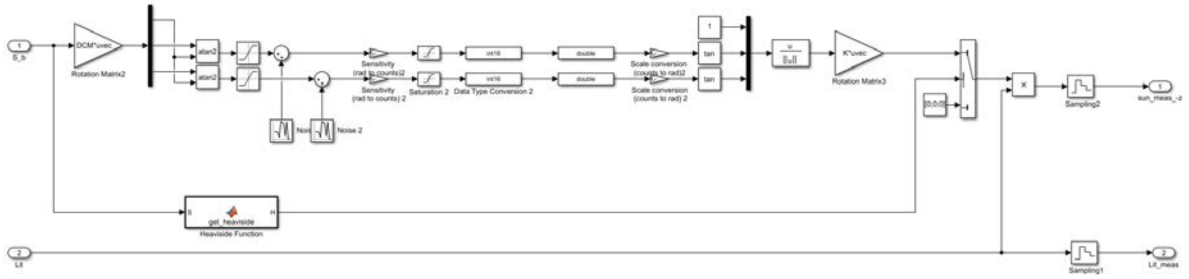
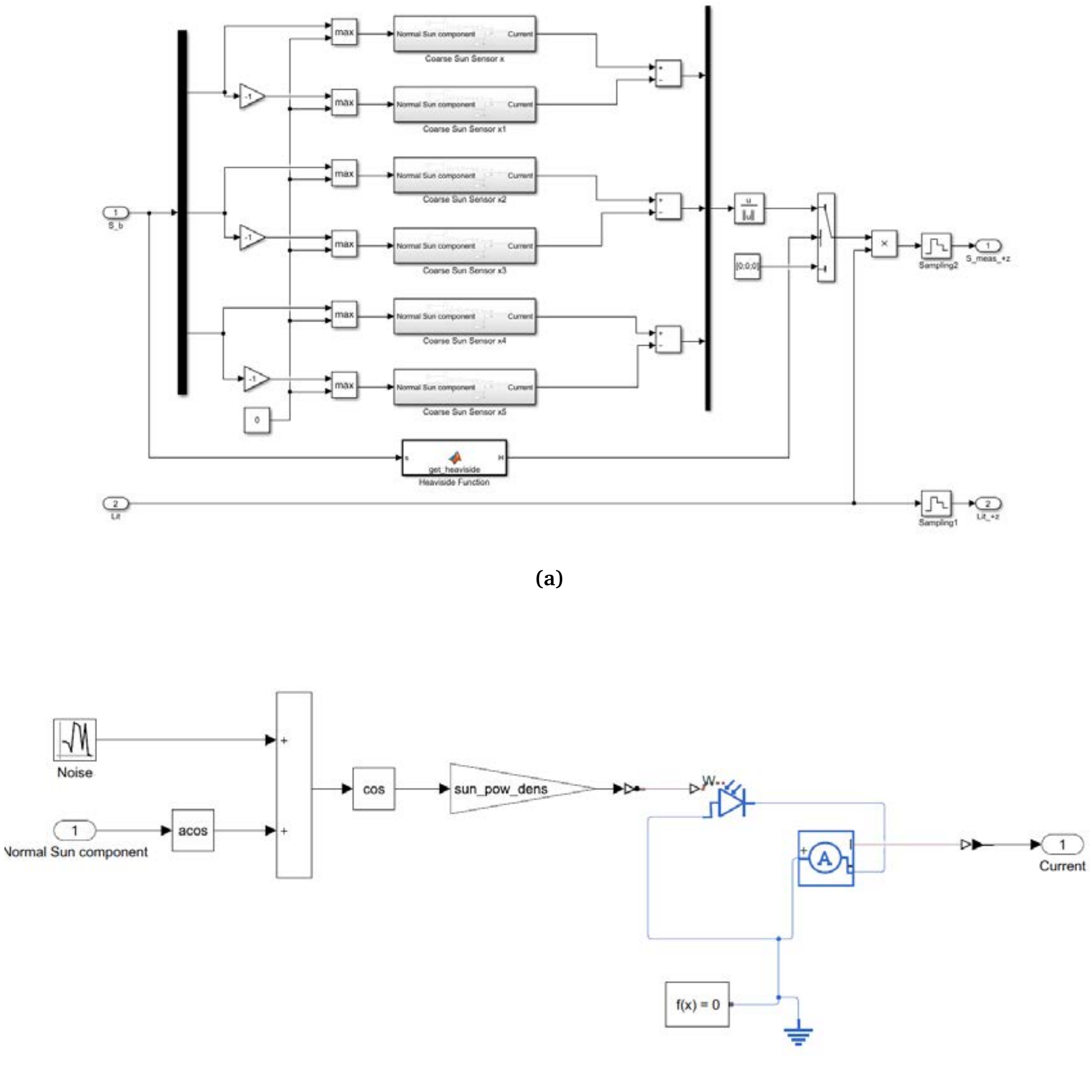


Figure B.1: Fine Sun sensor model.

Coarse Sun Sensors

Table B.2: Coarse Sun Sensor parameters.

Parameter	Unit	Value
Average Sun power Density	[W]	1396
Update rate	[s]	0.1
Noise standard deviation	[deg]	10
FOV	[deg]	180



(a)

(b)

Figure B.2: Coarse Sun sensors model.

Magnetometer

Table B.3: Magnetometer parameters.

Parameter	Unit	Value
Static Bias	[nT]	$\begin{bmatrix} 13.2234 \\ 46.323 \\ 1.5323 \end{bmatrix}$
Update rate	[s]	0.1
Noise standard deviation	[nT]	5
Sensitivity	[LSB/nT]	0.071428
Scale factors and cross-couplings		$\begin{bmatrix} 1.00058175 & -0.00001825 & -0.00095329 \\ -0.000608663 & 1.000493663 & 0.00084429 \\ -0.00020263 & 0 & 1.0009998 \end{bmatrix}$

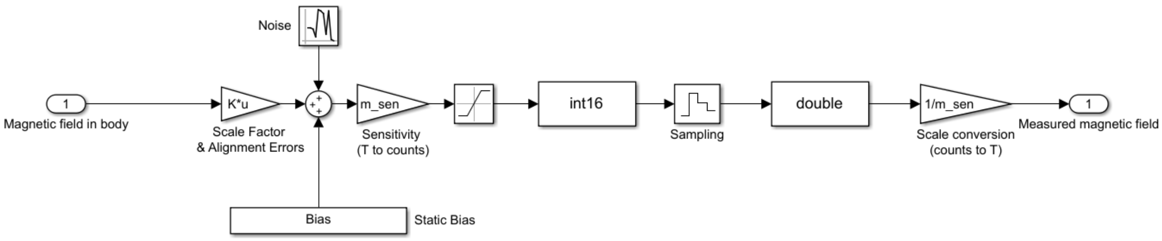


Figure B.3: Magnetometer model.

Gyroscope

Table B.4: Gyroscope parameters.

Parameter	Unit	Value
Static Bias	[rad/s]	$\begin{bmatrix} 0.52885 \\ -8.37277 \\ 4.81255 \end{bmatrix} \times 10^{-3}$
ARW standard deviation	[rad/s$^{0.5}$]	8.145×10^{-5}
RRW standard deviation	[rad/s]	2.618×10^{-4}
Update rate	[s]	0.1
Sensitivity	[LSB s/rad]	1.53×10^{4}
Scale factors and cross-couplings		$\begin{bmatrix} 1.0001499899 & 0.0001000150 & -0.0001000150 \\ -0.0001000150 & 1.000149989 & -0.0001000149 \\ 0.0001000150 & 0.000000010 & 1.0001499950 \end{bmatrix}$

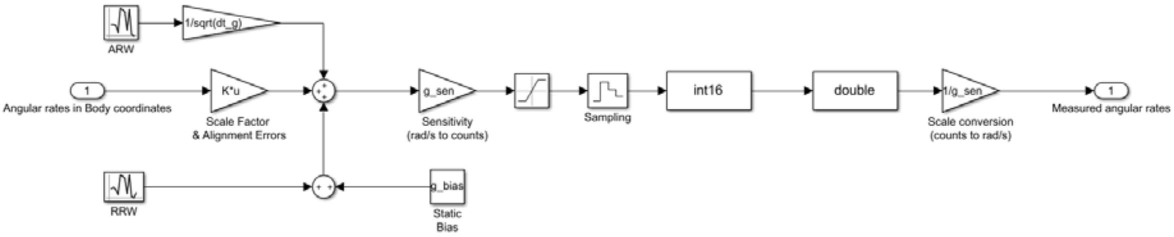
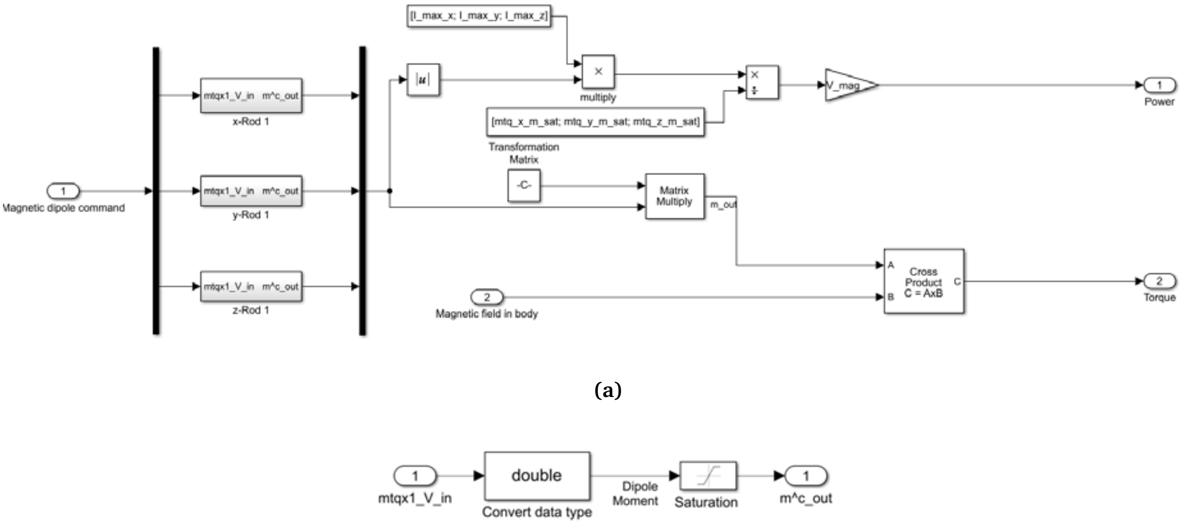


Figure B.4: MEMS Gyroscopes model.

Magnetorquers

Table B.5: Magnetorquers parameters.

Parameter	**Unit**	**Value**
Voltage	[V]	5
Magnetic dipole moment saturation limit (x)	[A m^2]	0.13
Magnetic dipole moment saturation limit (y & z)	[A m^2]	0.48
Electric current saturation limit (x)	[A]	0.064
Electric current saturation limit (y & z)	[A]	0.080



(a)

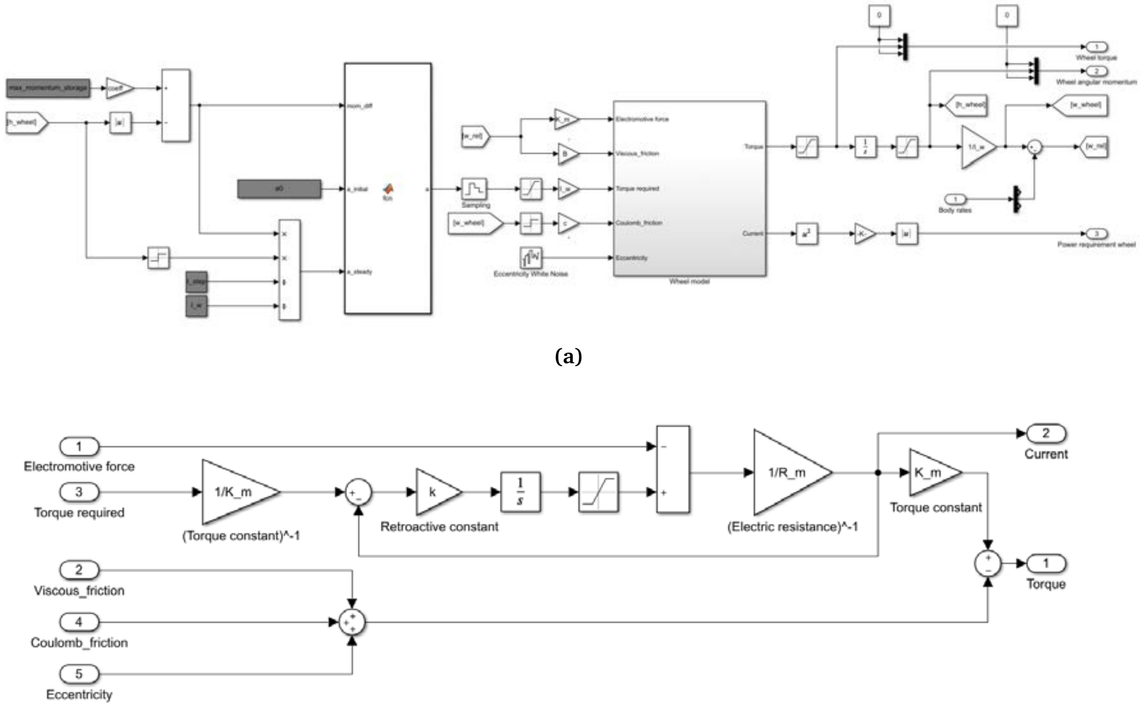
(b)

Figure B.5: Magnetorquers model.

Momentum Wheel

Table B.6: Momentum Wheel parameters.

Parameter	Unit	Value
Moment of Inertia	[kg m^2]	2.02922×10^{-6}
Maximum momentum storage	[N ms]	0.0017
Maximum torque	[N m]	0.00023
Maximum speed	[rev/min]	8000
Coefficient of viscous friction	[N m]	10^{-7}
Coefficient of Coulomb friction	[N m]	1.11×10^{-4}
Electric resistance	[Ω]	150
Torque constant	[N m/A]	0.0041
Retroactive constant	[Ω/s]	869.3



(a)

(b)

Figure B.6: Momentum wheel model.

Appendix C

Estimation Algorithms Scripts

In this appendix the MATLAB® scripts for the processing of the estimation algorithms developed in this work are presented.

MREKF

```
function [w_po, P_po] = mekf(w_pr, P_pr, b_m, z, sigma_m, dt, w_0, J_0, T, H_w, T_w, clock)
    %INPUTS:
    %w_pr    = A priori estimated rates                         [3x1]
    %P_pr    = A priori covariance matrix                       [3x3]
    %sigma_m = Standard deviation opf the magnetometer          [1x1]
    %dt      = Propagation time interval                        [1x1]
    %w_0     = orbital angular rate                             [1x1]
    %J_0     = Matrix of inertia                                [3x3]
    %b_m     = Measured magnetic field vector                   [3x1]
    %z       = Measured derivative of the magnetic field vector  [3x1]
    %T       = Environmental torque                             [3x1]
    %H_w     = Momentum bias                                    [3x1]
    %T_w     = Wheel torque                                     [3x1]
    %clock   = Measured time                                    [1x1]
    %OUTPUTS:
    %x_po    = A posteriori estimated state                     [3x1]
    %P_po    = A posteriori covariance matrix                   [3x3]

    J = diag(J_0);
    if clock < 10
        w_po = [0 0 0]';
        P_po = 0.0001*eye(3);
    else

    % Propagation stage of state vector and covariance matrix
    [w_p, P_p] = rate_prop(w_pr, P_pr, dt, w_0, T, H_w, T_w, J);
    % Update stage of state vector and covariance matrix
    [w_po, P_po] = rate_upd(w_p, P_p, b_m, sigma_m, z, dt);
    end
end

function [w_po, P_po] = rate_prop(w, P_pr, dt, w_0, T, H_w, T_w, J)
    H_0 = J(2)*w_0;
    Phi = eye(3)...
       + dt*[0 (0.5*w(3)*(J(2)-J(3))-H_w(3))/J(1) (0.5*w(2)*(J(2)-J(3))+H_w(2)-H_0)/J(1);
            (0.5*w(3)*(J(3)-J(1))+H_w(3))/J(2) 0 (0.5*w(1)*(J(3)-J(1))-H_w(1))/J(2);
            (0.5*w(2)*(J(1)-J(2))-H_w(2)+H_0)/J(3) (0.5*w(1)*(J(1)-J(2))+H_w(1))/J(3) 0];
    W    = (T-T_w)*dt./J;
    P_po = Phi*P_pr*Phi'+1e-6*eye(3)*dt;
    w_po = Phi*w + W;
 end

function [w_po, P_po] = rate_upd(w_pr, P_pr, b, sigma, z, dt)
    H    = skew(b)*dt;
    R    = sigma^2*eye(3);
    K    = P_pr*H'*(H*P_pr*H' + R)^(-1);
    P_po = (eye(3) - K*H)*P_pr*(eye(3) - K*H)'+K*R*K';
    w_po = w_pr + K*(z-H*w_pr);
end

function Xi = getXi(q)
    e = q(1:3);
    skew_e = skew(e);
    Xi = [q(4)*eye(3) + skew_e; -e'];
end
```

MEKF

```
function [x_po, P_po] = fcn(x_pr, P_pr, dt, sigmas, gains, w_meas, b_m, b_s, r_m, r_s, lit, mekf_on, q_triad)
    %INPUTS:
    %x_pr    = A priori estimated state                        [6x1]
    %P_pr    = A priori covariance matrix                      [6x6]
    %dt      = Propagation time interval                       [1x1]
    %ss      = vector of the sigmas of the sensors             [3x1]
    %gs      = vector of the gains of the CGGAO                [4x1]
    %w_meas  = Measured angular rates                          [3x1]
    %b_m     = Measured magnetic field vector                  [3x1]
    %b_s     = Measured Sun vector                             [3x1]
    %r_m     = Inertial magnetic field vector                  [3x1]
    %r_s     = Inertial Sun vector                             [3x1]
    %lit     = Illumination state (Boolean)                    [1x1]
    %mekf_on = flag for the activation of the algorithm        [1x1]
    %q_triad = quaternion from the TRIAD algorithm             [4x1]
    %OUTPUTS:
    %x_po    = A posteriori estimated state                    [6x1]
    %P_po    = A posteriori covariance matrix                  [6x6]

    sigma_u = sigmas(1);
    sigma_v = sigmas(2);
    sigma_m = sigmas(3);
    if b_s<0
       sigma_s = sigmas(4);
    else
       sigma_s = sigmas(5);
    end
    k_m       = gains(1);
    k_s       = gains(2);
    k_i       = gains(3);
    k_p       = gains(4);
    norm_b_m  = norm(b_m);
    b_m       = b_m/norm_b_m;

    norm_b_s  = norm(b_s);
    b_s       = b_s/norm_b_s;
    sigma_m_n = sigma_m/norm_b_m;
    norm_r_m  = norm(r_m);
    r_m       = r_m/norm_r_m;
    norm_r_s  = norm(r_s);
    r_s       = r_s/norm_r_s;
    if mekf_on == 0
        if lit == 0
            [x_po, P_po] = cggao(x_pr, P_pr, w_meas, dt, r_m, b_m, r_s,...
                                b_s, k_m, k_s, k_i, k_p, lit);
        else
            x_po = [q_triad; zeros(3,1)];
            P_po = 1e-3*eye(6);
        end
    else
    [x_p, P_p]   = mekf_prop(w_meas, x_pr, P_pr, sigma_u, sigma_v, dt);
    [x_po, P_po] = mekf_upd(x_p, P_p, r_m, b_m, sigma_m_n);
        if lit == 1
             [x_po, P_po] = mekf_upd(x_po, P_po, r_s, b_s, sigma_s);
        end
    end
end

function [x_po, P_po] = mekf_prop(omega_m, x_pr, P_pr, sigma_u,sigma_v, dt)
    q_pr    = x_pr(1:4);
    beta_pr = x_pr(5:7);
    omega   = omega_m - beta_pr;
    beta_po = beta_pr;
    Omega   = getOmega_mekf(omega, dt);
    q_po    = Omega*q_pr;
    gamma   = [-eye(3)    zeros(3,3);
                zeros(3,3) eye(3)];
    Phi     = getPhi(omega, dt);
    Q       = getQ_mekf(sigma_v, sigma_u, dt);
    P_po    = Phi*P_pr*Phi' + gamma*Q*gamma';
    x_po    = [q_po; beta_po];
 end

function [x_po, P_po] = mekf_upd(x_pr, P_pr, r, b, sigma_s)
    q_pr     = x_pr(1:4);
    beta_pr  = x_pr(5:7);
    Xi       = getXi(q_pr);
    Pi       = getPi(q_pr);
    A_pr     = Xi'*Pi;
    h        = A_pr*r;
    H        = [skew(h) zeros(3,3)];
    R        = sigma_s^2*eye(3);
    K        = P_pr*H'*(H*P_pr*H' + R)^(-1);
    P_po     = (eye(6) - K*H)*P_pr*(eye(6) - K*H)'+K*R*K';
    dx_po    = K*(b - h);
    dphi_po  = dx_po(1:3);
    dbeta_po = dx_po(4:6);
    q_po     = q_pr + 0.5*Xi*dphi_po;
    q_po     = q_po/norm(q_po);
    beta_po  = beta_pr + dbeta_po;
    x_po     = [q_po; beta_po];
end

```

```
function [Omega] = getOmega_mekf(omega, dt)
    omega_norm = norm(omega);
    psi = omega*sin(0.5*omega_norm*dt)/omega_norm;
    o1 = cos(0.5*omega_norm*dt)*eye(3) - skew(psi);
    o2 = psi;
    o3 = -psi';
    o4 = cos(0.5*omega_norm*dt);
    Omega = [o1     o2;
             o3     o4];
end

function Q = getQ_mekf(sigma_v, sigma_u, dt)
    Q11 = eye(3)*(sigma_v^2*dt + 1/3*sigma_u^2*dt^3);
    Q12 = eye(3)*(1/2*sigma_u^2*dt^2);
    Q21 = Q12;
    Q22 = eye(3)*(sigma_u^2*dt);
    Q = [Q11    Q12;
         Q21    Q22];
end
```

AEKF

```
function [x_po, P_po] = fcn(x_pr, P_pr, dt, ss, gs, J_0, b_m, b_s, r_m, r_s, lit, T, H_w, T_w, w_0, aekf_on)
        %INPUTS:
        %x_pr    = A priori estimated state                         [7x1]
        %P_pr    = A priori covariance matrix                       [7x7]
        %dt      = Propagation time interval                        [1x1]
        %ss      = vector of the sigmas of the sensors              [3x1]
        %gs      = vector of the gains of the CGGAO                 [4x1]
        %J_0     = matrix of inertia                                [3x3]
        %b_m     = Measured magnetic field vector                   [3x1]
        %b_s     = Measured Sun vector                              [3x1]
        %r_m     = Inertial magnetic field vector                   [3x1]
        %r_s     = Inertial Sun vector                              [3x1]
        %lit     = Illumination state (Boolean)                     [1x1]
        %T       = Environmental torque                             [3x1]
        %H_w     = Momentum bias                                    [3x1]
        %T_w     = Wheel torque                                     [3x1]
        %w_0     = orbital angular rate                             [1x1]
        %aekf_on = flag for the activation of the algorithm         [1x1]
        %OUTPUTS:
        %x_po    = A posteriori estimated state                     [7x1]
        %P_po    = A posteriori covariance matrix                   [7x7]

        % Definition of the constants
        sigma_m = ss(1);
        if b_s<0
            sigma_s = ss(2);
        else
            sigma_s = ss(3);
        end
        J   = diag(J_0);
        k_m = gs(1);
        k_s = gs(2);
        k_i = gs(3);
        k_p = gs(4);
        % Reorganization of the data from the sensors
        norm_b_m = norm(b_m);
        b_m      = b_m/norm_b_m;
        norm_b_s = norm(b_s);
        b_s      = b_s/norm_b_s;
        % Normalization of magnetometer Standard Deviation
        sigma_m_n = sigma_m/norm_b_m;
        % >Reorgazization of the inertial vectors
        norm_r_m = norm(r_m);
        r_m      = r_m/norm_r_m;
        norm_r_s = norm(r_s);
        r_s      = r_s/norm_r_s;
        %Initialization
        if aekf_on == 0
            if lit == 0
                [x_po, P_po] = cggao(x_pr, P_pr, dt, r_m, b_m, r_s, b_s, k_m, k_s, k_i, k_p, lit);
            else
                x_po = [q_triad; zeros(3,1)];
                P_po = 1e-6*eye(7);
        end
        else
        %propagation step
        [x_p, P_p] = aekf_prop(x_pr, P_pr, dt, T, H_w, T_w, w_0, J);
        %Magnetometer measurement update
        [x_po, P_po] = aekf_upd(x_p, P_p, r_m, b_m, sigma_m_n);
        %Sun sensor measurement update (only if the satellite is illuminated)
        if lit == 1
            [x_po, P_po] = aekf_upd(x_po, P_po, r_s, b_s, sigma_s);
        end
        end
end

function [x_po, P_po] = aekf_prop(x, P, dt, T, H_w, T_w, w_0, J)
    q     = x(1:4);
    w     = x(5:7);
    H_0   = J(2)*w_0;
    Phiq  = 1/4*[0     w(3) -w(2) w(1)    q(4) -q(3)  q(2);
                 -w(3)  0     w(1) w(2)    q(3)  q(4) -q(1);
                  w(2) -w(1)  0    w(3)   -q(2)  q(1)  q(4);
                 -w(1) -w(2) -w(3) 0      -q(1) -q(2) -q(3)];
    Phiw  = [0 0 0 0 0 (0.5*w(3)*(J(2)-J(3))-H_w(3))/J(1) (0.5*w(2)*(J(2)-J(3))+H_w(2)-H_0)/J(1);
            0 0 0 0 (0.5*w(3)*(J(3)-J(1))+H_w(3))/J(2) 0 (0.5*w(1)*(J(3)-J(1))-H_w(1))/J(2);
            0 0 0 0 (0.5*w(2)*(J(1)-J(2))-H_w(2)+H_0)/J(3) (0.5*w(1)*(J(1)-J(2))+H_w(1))/J(3) 0];
    Phi   = eye(7) + [Phiq; Phiw]*dt;
    W     = [zeros(4,1); (T-T_w)./J*dt];
    S     = 0.6e-8*eye(3);
    Q     = [zeros(4,7);
             zeros(3,4), S];
    P_po  = Phi*P*Phi' + Q*dt;
    x_po  = Phi*x + W;
end

function [x_po, P_po] = aekf_upd(x, P, r, b, sigma)
    q     = x(1:4);
    w     = x(5:7);
    Xi    = getXi(q);
    Pi    = getPi(q);
    A     = Xi'*Pi;
    h     = A*r;
    H1    = 2*[ q(1)    q(2)    q(3);
                q(2)   -q(1)    q(4);
                q(3)   -q(4)   -q(1)]*r;
    H2    = 2*[-q(2)    q(1)   -q(4);
                q(1)    q(2)    q(3);
                q(4)    q(3)   -q(2)]*r;
    H3    = 2*[-q(3)    q(4)    q(1);
               -q(4)   -q(3)    q(2);
                q(1)    q(2)    q(3)]*r;
    H4    = 2*[ q(4)    q(3)   -q(2);
               -q(3)    q(4)    q(1);
                q(2)   -q(1)    q(4)]*r;
    H     = [H1 H2 H3 H4 zeros(3,3)];
    R     = sigma^2*eye(3);
    K     = P*H'*(H*P*H' + R)^(-1);
    P_po  = (eye(7) - K*H)*P;
    dx_po = K*(b - h);
    dq_po = dx_po(1:4);
    dw_po = dx_po(5:7);
    q_po  = q + dq_po;
    q_po  = q_po/norm(q_po);
    w_po  = w + dw_po;
    x_po  = [q_po; w_po];
end

function A = getA(q)
    q_v = q(1:3);
    q_0 = q(4);
    skew_q_v = skew(q_v);
    A = eye(3) - 2*q_0*skew_q_v + 2*skew_q_v^2;
end
function alpha = getalpha(A, r, b, k)
    b_hat = A*r;
    alpha = k*cross(b, b_hat);
end
function Pi = getPi(q)
    e = q(1:3);
    skew_e = skew(e);
    Pi = [q(4)*eye(3) - skew_e; -e'];
end
function skew = skew(v)
    skew = [0      -v(3)     v(2);
            v(3)    0       -v(1);
           -v(2)    v(1)     0];
end
```