TÉCNICO LISBOA

Attitude Determination and Control System for the QuantSat-PT project

João Nunes Revés

Thesis to obtain the Master of Science Degree in

Electrical and Computer Engineering

Supervisors: Prof. Rita Maria Mendes de Almeida Correia da Cunha
Prof. Rui Manuel Rodrigues Rocha

Examination Committee

Chairperson: Prof. João Manuel de Freitas Xavier
Supervisor: Prof. Rita Maria Mendes de Almeida Correia da Cunha
Member of the Committee: Prof. David Alexandre Cabecinhas

June 2022

In memory of Odete.

Declaration

I declare that this document is an original work of my own authorship and that it fulfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.

Acknowledgments

I would like to thank my supervisors, Prof. Rita Cunha and Prof. Rui Rocha, for sharing their knowledge and guidance throughout this work. Additionally, I would like to thank the ISTSat-1 and QuantSat-PT teams, especially Prof. João Paulo Monteiro, Rodrigo Ramos and Inês Viveiros, for the opportunity to research this interesting topic. I wish you the best of luck with future steps in these exciting projects.

I would also like to thank my colleagues that made my journey throughout University an even more enjoyable experience, especially Gonçalo Chambel, Filipe Oliveira and Pedro Pereira.

The FST Lisboa team helped me find a purpose for my future engineering career, as well as gave me memories I will never forget. Thank you all.

Finally, I would like to thank my family, friends and girlfriend, Inês, for their encouragement throughout difficult times and support to follow my ambitions. Without you, this wouldn't be possible. I am truly grateful.

Resumo

O Sistema de Determinação e Controlo de Atitude de um satélite é o subsistema responsável por fornecer informação de atitude e manter o direção do ”payload” da missão. O objetivo deste trabalho é fornecer um ADCS funcional ao projeto QuantSat-PT, um CubeSat desenvolvido para testar a implementação de comunicações quânticas. A missão requer alta precisão de direção, sendo limitada por limites severos de volume, bem como várias manobras além de apontar para a Terra. Tendo isso em conta, a detecção do ”beacon” do ”payload” é reaproveitada como um sensor, para fornecer uma referência da Terra, a fim de melhorar a estimativa de atitude e cumprir a precisão de apontamento combinada de $0,1^{\circ}$. A medição do ”beacon” tem disponibilidade limitada, por isso o conjunto de sensores é complementado por um magnetómetro, sensores de direção solar e um giroscópio. Embora as rodas de reação sejam o atuador mais comum para altas precisões de apontamento, ”magnetorquers” são usados para manobras adicionais. Uma comparação entre os algoritmos de ”multiplicative extended Kalman filter” (MEKF), ”complementary filter” (CF) e TRIAD é realizada para a determinação da atitude. Para o controlo de atitude, são estudados um ”sliding controller”, um ”global finite-time attitude tracking controller ”(GFTAC) e um controlador ”proportional derivative”(PD) simples. A lei de controle do ”B-dot” foi empregue para a manobra de ”detumbling”. Um ambiente de simulação permitiu uma comparação precisa das diferentes opções. O ADCS proposto manteve um erro de apontamento inferior a $0,1^{\circ}$ por $28,29\%$ do tempo em que o satélite estava a sobrevoar a estação terrestre (GS), com linha direta de visão (LOS).

Palavras-chave: Cubesat, filtro de Kalman, controlo não linear, estimação de atitude, controlo de atitude, QuantSat-PT

Abstract

The Attitude Determination and Control System (ADCS) of a satellite is the subsystem responsible for providing attitude knowledge and maintaining accurate pointing of the mission's payload. The objective of this work is to provide a functional ADCS to the QuantSat-PT project, a CubeSat developed to test the implementation of quantum communications. The mission requires high pointing accuracy, whilst being constrained by severe volume limits, as well as various manoeuvres besides Earth pointing. With this in mind, the payload's beacon detection is repurposed as a sensor, to provide an Earth's reference, in order to improve attitude estimation and comply with the combined pointing accuracy of 0.1°. The beacon measurement is limited in its availability, so the sensors suite is complemented by a magnetometer, sun sensors and a gyroscope. Although reaction wheels are the most common actuator for high pointing accuracies, magnetorquers are used for further manoeuvres. A comparison between the multiplicative extended Kalman filter (MEKF), Complementary Filter (CF) and TRIAD algorithms is performed for the attitude determination. For the attitude control, a sliding controller, a global finite-time attitude tracking controller (GFTAC) and a simple proportional derivative (PD) controller are studied. The B-dot control law was employed for the detumbling manoeuvre. A simulation environment allowed for an accurate comparison of the different options. The proposed ADCS maintained a pointing error lower than 0.1° for 28.29% of the time the satellite was flying over the ground station (GS), with a direct line of sight (LOS).

Keywords: Cubesat, Kalman filter, nonlinear control, attitude determination, QuantSat-PT

Contents

List of Tables

List of Figures

Nomenclature

Greek symbols

$\boldsymbol{\beta}$	Bias.
δ, Δ	Interval or error.
$\boldsymbol{\epsilon}$	Residual error.
Γ	Discrete input matrix.
$\boldsymbol{\nu}$	Noise vector.
ω	Angular velocity.
Φ	State transition matrix.
ω	Athmospheric density.
ϑ	Rotation vector parameterization.
θ, ϕ, ψ	Euler angle.
ϑ	Rotation axis parameterization.
Υ	Discrete process distribution matrix.
$\boldsymbol{\xi}$	Other variables to estimate.

Operations

$diag([\square, \square, ...])$	Diagonal matrix.
$E\{\square\}$	Expected value.
$\square \times \square$	External product.
$N(\square, \square)$	Gaussian white noise.
$\lim \square$	Limit function.
$\exp(\square)$	Exponential function.
$\otimes$, $\Omega(\square)$, $\Xi(\square)$	Quaternion functions.

$[\square\times]$ Skew-symmetric.

sat $(\square)$ Saturation function.

sign $(\square)$ Signum function.

Roman symbols

$\mathbf{0}$ Zero vector.

$\mathbf{a}$ Acceleration.

$\mathbf{b}$ Measurement in the body frame.

$\mathcal{B}$ Body frame.

$\mathbf{B}$ True magnetic field.

B Input matrix.

D Direct transmission matrix.

$\mathbf{e}$ Rotation axis.

$\mathcal{E}$ Earth-centered/Earth-fixed frame.

$\mathbf{F}$ Force.

f Sufficiently differentiable function.

F State matrix.

M Mass of the largest celestial body.

G Process noise distribution matrix.

K Kalman gain.

μ Height.

H Angular momentum.

H Observation matrix.

$\mathcal{I}$ Earth-centered inertial frame.

A Attitude matrix.

I identity matrix.

I_q Unit quaternion.

J Moment of inertia tensor.

J Moment of inertia tensor with reaction wheels.

$\mathbf{m}$	Commanded magnetic dipole.
m	Mass of a certain point.
μ	Earth's gravitational constant.
P	Covariance.
$\mathbf{q}$	Quaternion.
P	Covariance matrix.
$\mathbf{r}$	Measurement in the reference frame.
$\mathbf{r}$	Position.
$\mathcal{R}$	Attitude matrix in the Complementary filter.
t	Time.
$\mathbf{u}$	Input vector.
$\mathbf{v}$	Linear velocity.
$\mathbf{v}$	Measurement noise.
V	Lyapunov function.
$\mathbf{w}$	Process noise.
$\mathbf{x}$	State vector.
$\mathbf{h}$	Measurement vector.
$\mathbf{u}$	Observations vector.

Subscripts

$\square^{1,2,3,\ldots,i}$	i-th component.
$\square_{1,2,3,\ldots,i}$	i-th component.
$\square_{\mathcal{B}}$	Represented in the body frame.
$\square_{beacon}$	Beacon.
$\square_{mag}$	Magnetometer.
$\square_{d}$	Aerodynamic drag.
$\square_{\mathcal{E}}$	Represented in the Earth-centered/Earth-fixed frame.
$\square_{fw}$	Flywheel.
$\square_{G}$	Gravity.

$\square_p$ Perturbation.

$\square_g$ Gyroscope.

$\square_{GMST}$ Greenwich Mean Sidereal Time.

$\square_{\mathcal{I}}$ Represented in the Earth-centered inertial frame.

$\square_0$ Inital condition.

$\square_{moon}$ Moon.

$\square_{mtq}$ Magnetorquer.

$\square_{sun}$ Sun.

$\square_{sph}$ Spherical perturbation.

$\square_{TRIAD}$ TRIAD.

$\square_{x,y,z}$ Orthogonal axis.

Superscripts

$\square^c$ Centre of mass.

$\dot{\square}$ Derivative.

$\square^{ext}$ Exterior.

$\hat{\square}$ Estimate.

$\square^+$ *A posteriori* estimate.

$\square^-$ *A prori* estimate.

$\square^T$ Transpose.

Glossary

ADCS	Attitude Determination and Control System.
CF	Complementary filter.
COTS	Commercial off-the-shelf.
DCM	Direction cosine matrix.
ECEF	Earth-centered/Earth-fixed.
ECI	Earth-centered inertial.
EGM2008	Earth gravitational model 2008.
EKF	Extended Kalman filter.
FOG	Fiber-optic gyro.
FOV	Field-of-view.
FSM	Fast-steering mirrors.
GFTAC	Global finite-time attitude tracking controller.
GMST	Greenwich Mean Sidereal Time.
GNSS	Global Navigation Satellite System.
GS	Ground station.
IGRF	International Geomagnetic Reference Field.
IMU	Inertial measurement unit.
KF	Kalman filter.
LEO	Low Earth orbit.
LOS	Line-of-sight.
MCU	Microcontroller unit.
MEKF	Multiplicative extended Kalman filter.
MEMS	Micro-Electro-Mechanical Systems.
PD	Proportional derivatice.
QKD	Quantum Key Distribution.
RAAN	Right ascension of the ascending node.
RLG	Ring laser gyro.
RPM	Revolutions per minute.
SO(3)	Special orthogonal group of dimension.
SWaP	Size, weight, and power.

TRIAD TRIaxial Attitude Determination.

Chapter 1

Introduction

1.1 Motivation

The first satellite to be propelled into space was the Sputnik 1 in 1957 by the USSR, with the goal of having the upper hand in the race to Space with USA. In this period, the study of all subjects regarding space had a rise in popularity and established the foundations for the work that is done to this date. Since then, the satellites' missions have diversified, transforming from proof of concept to essential commodities in today's world. Since the attitude determination and control system (ADCS) is the subsystem responsible for pointing the satellite to a specific location, advancements in this area is what allows the diversification of missions. The pointing accuracy of the satellite is directly correlated to the imaging accuracy or capacity of communication, some of the most common applications for satellites.

The deployment of a satellite is costly and is directly related to its size and mass, so the trend in technology investments goes towards smaller satellites. The SmallSat category[1] has up to $180kg$ and includes various sub-divisions, such as the Nanosatellite, which weight ranges from $1kg$ to $10kg$. CubeSats are a class of nanosatellites composed of one or more standardized units (U), a cube with $10cm$ edges and mass up to $1.33kg$, and have access to commercial launchers. This category was created by California Polytechnic State University at San Luis Obispo (Cal Poly) and Stanford University in 1999 to provide a cheap and accessible platform for education and space exploration. In order to stimulate students to learn the techniques and methods professionals use, the *Fly Your Satellite!* [2] programme was created. The teams are guided through the typical development cycle of a space mission of a CubeSat and undergo accurate verification, with project reviews at certain phases.

When the CubeSat platform is chosen for implementation, it comes with constraints in terms of size, weight, and power (SWaP), in addition to the severe environmental conditions during launch and in-orbit that already posed a challenge regarding the miniaturization and ruggedization of the payload. This has stimulated the development of commercial-off-the-shelf (COTS) solutions, that with the technological advancements are able to comply not only with the harsh performance requirements but also withstand the space environmental challenges. However, in order to meet the SWaP demands of missions, the cost is steep, reaching 3300 USD for some of the most precise sensors [3], therefore there is an incentive

to find innovative solutions.

This work is focused on designing and evaluating an ADCS, while addressing these issues. The subsystem is integrated in the QuantSat-PT satellite, developed by Instituto Superior Técnico.

1.2 QuantSat-PT project

In the information age, it is crucial to create and maintain secure communications. One way of improving our current security standards is to use Quantum Key Distribution (QKD) methods, which allows two parties to exchange encryption keys with absolute confidence that any eavesdropping by a third party will be detected [4]. Given that this property of QKD stems from fundamental quantum mechanics laws, it is theoretically impossible to intercept the encryption keys without this interference being detected and will remain so even considering future technological developments. QKD is based on photonic communication links. Ground-based links typically rely on optical fibers, which have non-negligible losses, thus limiting transmission distances to a few hundreds of kilometers. As such, satellite-based QKD is a promising approach to establish a global quantum network. The development of a reliable and efficient space-to-ground link is an important first step, which has been demonstrated by the Micius satellite [5]. Implementing this technology in a CubeSat is a further step in the deployment of QKD technology, allowing for the low-cost deployment of large constellations. Given the limited quantum link budget, extremely narrow optical beams must be used. This places strict requirements on the attitude determination and control system to guarantee pointing accuracy in the range of tens of microradians [6]. Furthermore, due to the fact that space-based QKD can only be performed while the spacecraft is in eclipse, significant temperature variations can be expected and present a challenge for the internal alignment of optical components. These constraints are addressed using a three-level pointing system which allows for internal angular corrections using fast-steering mirrors (FSMs) and for spacecraft-to-ground locking using a laser beacon.

The QuantSat-PT project is the first step of a larger and longer-term vision of developing quantum communication satellites and ground stations in Portugal, making the country autonomous in such sovereignty technologies, and integrating Portugal in the future European Quantum Communication Infrastructure. The aim is to begin the development of a 3U CubeSat for a quantum downlink. Namely, the goals of project QuantSat-PT are:

- To develop a quantum payload for space-earth quantum key distribution in 2U,
- To test the quantum payload on earth over a distance of several kilometres,
- To develop the preliminary design of the space segment.

To attain these goals, the project QuantSat-PT brings together a unique and multidisciplinary team. It explores the complementary expertise of IT researchers from:

- The Physics of Information and Quantum Technologies Group responsible for the development of the first free-space quantum key distribution system in Portugal,

- The Network Architecture and Protocols Group and the Antennas and Propagation Group, both involved in the ISTSat One, a Portuguese classical communications CubeSat to be sent to space in 2022 through the ESA program *Fly Your Satellite!*,
- The Optical Networking Group, expert in classical and quantum optical communications.

1.3 Problem Statement

From the mission objectives, the attitude requirements are defined. In order to ensure Quantum communications, an attitude error in the range of tens of microradians must be obtained whilst in eclipse. Due to restrictions imposed by the payload size, the volume available for sensors and actuators is reduced, which often leads to an increase in the cost of the satellite, if no performance cutbacks are possible. With this in mind, the telescope from the payload will be used not only for communications but also to provide a ground reference, which replaces other sensors. Other CubeSats with optical communications missions have been designed to meet the attitude accuracy requirements through star trackers [7]. In [8], a staged approach is proposed for the attitude determination, where the beacon, when received, is used to improve pointing accuracy in the fine stage.

The proposed architecture for the attitude determination system can be seen in Fig. 1.1. The pointing requirements increase when more accurate sensors and actuators can be employed. The first stage consists of using the common ADCS sensors to receive a signal from the beacon being transmitted by the ground station (GS). The attitude required for this stage is where the payload is pointed at the GS with an attitude error of 10°, equivalent to the field of view (FOV) of the telescope that perceives the beacon. The second stage uses the Earth's reference from the beacon to improve the attitude estimation, with the goal of obtaining a pointing error of 0.1°. This value is defined by the amplitude of actuation of the FSM. These two stages are treated as one when it comes to the control and estimation algorithms and, when the required attitude is achieved, the following stage is employed.

A fine optical pointing stage is necessary to obtain the pointing accuracy required for quantum communications. The optical pointing system within the payload is in charge of this part of the mission. The laser beacon coming from the ground enters through the telescope and passes through an assembly of mirrors and lenses, guiding it to a quad-cell. The error signal generated from the difference between the measured and desired position of the beacon on the quad-cell is employed to drive a FSM to actuate on the beacon attitude, augmenting the pointing accuracy without the need to move the satellite structure. This fine pointing process is executed continuously throughout the mission, designed as a closed control loop that runs in parallel with the other stages managed by the ADCS. The process, architecture, and implementation of this section of the pipeline for the Quantum-Sat project are illustrated in [9].

1.3.1 ADCS Requirements

Although accomplishing the mission pointing requirements are the main objective of the ADCS, there are some other requirements it needs to comply with:

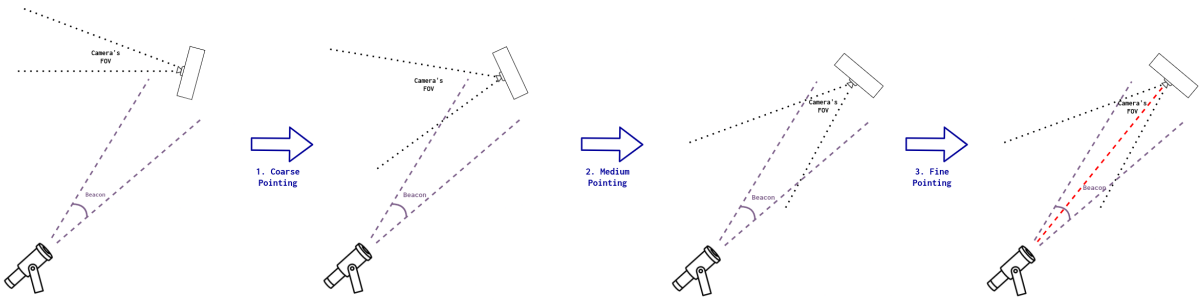

Figure 1.1: Pointing Stages.

- **Detumbling:** The satellite must reduce its angular velocity in after deployment in less than one orbit,
- **Communications:** The satellite must be able to point to nadir with an accuracy of 20 degree in order to allow communications,
- **Off-nadir pointing:** The satellite must be able to point towards the GS while passing over it.

1.4 Thesis Outline

The thesis is organized in the following chapters:

- **Chapter 2** gives a summary of the background concepts, such as attitude representations, different frames and spacecraft mechanics;
- **Chapter 3** selects the sensors that are used in the implementation and exposes the attitude estimation and determination algorithms;
- **Chapter 4** analyses the actuators for attitude control, as well as the algorithms for the different manoeuvres of the satellite;
- **Chapter 5** describes the simulation environment used to validate and compare the different methods. It also provides the parameters for the orbit, environment and sensor/actuators parameters;
- **Chapter 6** displays the results of the simulations and reflects on their performance;
- **Chapter 7** draws conclusions on the work done and suggests future work.

Chapter 2

Background

In this chapter, the concepts needed to develop the estimation and control algorithms are presented. In Section 2.1, the frames of reference for used to describe the relation between the satellite and the Earth are presented. In Section 2.2, the various attitude representations are presented and their benefits and shortcomings analyzed. The attitude error representations are also derived. In Section 2.3, the laws of motion that describe the behaviour of the satellite are presented. The kinematics, i.e., the variations of orientation when only the satellite is considered, are studied. The attitude dynamics relate torques with the inertia of the satellite, which generates angular accelerations. In Section 2.4, the orbital motion of the satellite is described, along with the parameters to describe it. Finally, in Section 2.5, the influence of external torques and forces by the space environment on the satellite are studied.

2.1 Reference Frames

Different three-dimensional reference frames are useful when exploring an attitude analysis problem. These are specified by the origin's location and the orientation of their coordinate axis. Different frames are used to model different behaviours of the satellite, as well as the environment that encloses it. This section introduces these useful frames and is based on [10].

2.1.1 Body Frame

The body frame of the satellite, designated as $\mathcal{B}$, has its origin in the estimated centre of mass and its axis rigidly connected to the satellite. It is composed by three unit vectors $\mathcal{B} = \{b_1, b_2, b_3\}$, such that b_3 is aligned with the telescope of the payload, b_1 is aligned with 3U length of the satellite and b_2 completes the right-hand coordinate system.

2.1.2 Inertial Reference Frame

In an inertial frame, all Newton's laws of motion are valid. Any frame moving at constant velocity and without rotation with respect to an inertial frame is also inertial. The frame of choice for near-Earth

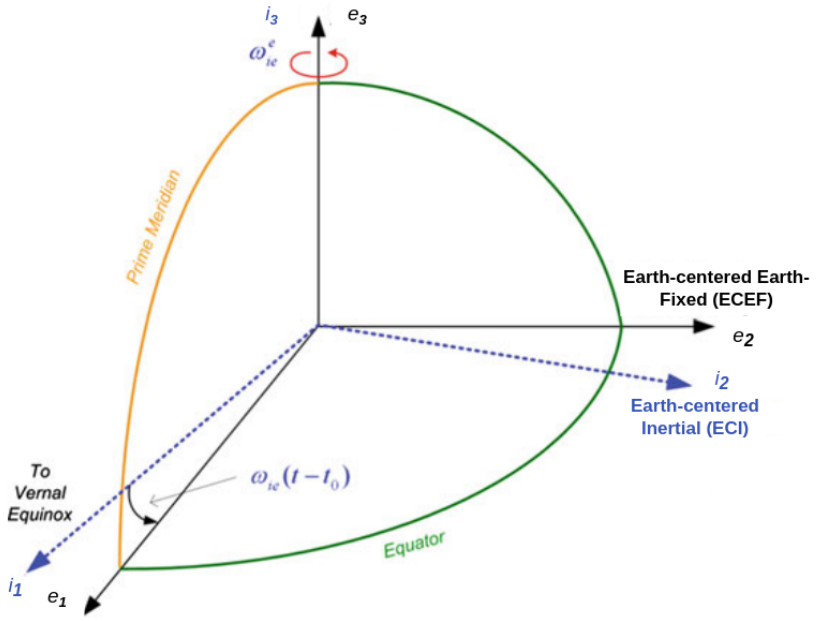

Figure 2.1: ECI (black) and ECEF (blue) frames, adapted from [10]

environments is the Earth-centered Inertial (ECI) frame, designated as $\mathcal{I} = \{i_1, i_2, i_3\}$. It has as origin the centre of mass of the Earth, with i_3 aligned with the Earth's rotation axis through the conventional terrestrial pole (defined as an average of the poles from 1900 to 1905), i_1 aligned with the equatorial plane, pointing towards the vernal equinox and i_2 completes the right-hand coordinate system. The ECI frame is represented in Fig. 2.1.

2.1.3 Earth-Centered/Earth-Fixed Frame

The Earth-Centered/Earth-Fixed frame (ECEF) is defined as $\mathcal{E} = \{e_1, e_2, e_3\}$. The axis e_3 is aligned with i_3, however e_3 points in the direction of Earth's prime meridian, e_2 completes the right-hand coordinate system. The rotation angle between $\mathcal{E}$ and $\mathcal{I}$ is known as Greenwich Mean Sidereal Time (GMST) angle, represented as θ_{GMST}, and follows the Earth's rotation. It is calculated based on that instant time a number of Julian centuries elapsed from the J2000 epoch. The ECEF frame is represented in Fig. 2.1.

The transformation of a position vector $\mathbf{r}$ from its ECI representation $\mathbf{r}_\mathcal{I}$ to its ECEF representation $\mathbf{r}_\mathcal{E}$ follows:

$$\mathbf{r}_\mathcal{E} = \begin{bmatrix} \cos\theta_{GMST} & \sin\theta_{GMST} & 0 \\ -\sin\theta_{GMST} & \cos\theta_{GMST} & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{r}_\mathcal{I} \tag{2.1}$$

The comparison of both frames can also be seen in Fig. 2.1.

2.2 Attitude Representations

The attitude of a satellite is defined as the orientation of the satellite's frame $\mathcal{B}$ with respect to the reference system $\mathcal{I}$. There are a few different ways of representing the orientation, each with its advantages and shortcomings. The most significant ones for this work will be analyzed. This section represents the useful representations for this work and is based on [11].

2.2.1 Rotation Matrix Representation

The rotation (or attitude) matrix, denoted by the letter A, is the fundamental representation to express a rotation between two axes. The attitude matrix A_{GF} represents the transformation from a reference frame F to a reference frame G and is given by

$$A_{GF}\mathbf{x}_F = \mathbf{x}_G \tag{2.2}$$

The matrix A_{GF} is a proper orthogonal 3×3 matrix, as all matrices in the special orthogonal group of dimension 3 ($SO(3)$) group, and is often designated as a Direction Cosine Matrix (DCM). Although it does not suffer from singularities and does not require large computation efforts, it is composed of 9 parameters, whereas in reality it has 6 scalar constraints. This is due to the orthogonality constraint $AA^T = I$ and symmetry requirement.

2.2.2 Euler Axis/Angle Representation

Euler's Theorem states that every rotation matrix consists of a rotation around a fixed axis, the $\mathbf{e}$ unit vector, that has the same representation in both frames. This statement is equivalent to substituting $\mathbf{x}_F$ and $\mathbf{x}_G$ by $\mathbf{e}$ in (2.2). It can be proven that the rotation axis $\mathbf{e}$ is the eigenvector, with a corresponding eigenvalue of 1.

To completely define the matrix, the angle of rotation about the rotation axis, ϑ, should be specified. The corresponding attitude matrix is as follows:

$$A(\mathbf{e}, \vartheta) = \mathit{I} - \sin\vartheta[\mathbf{e}\times] + (1 - \cos\vartheta)[\mathbf{e}\times]^2 \tag{2.3}$$

where $[\square\times]$ is the skew-symmetric matrix, defined in Appendix A.1.

This representation seems to be dependent on four parameters ($\mathbf{e}$ and ϑ) but when the constraint $||\mathbf{e}|| = 1$ is imposed there are only three independent parameters. From (2.3), it can be said that the representation is a periodic function of the rotation angle over an unlimited range, with a period of 2π.

2.2.3 Rotation vector representation

A three-component representation can be computed from the combination of the Euler axis and angle, with $\boldsymbol{\vartheta} \equiv \vartheta\mathbf{e}$, such that

$$A(\mathbf{e}, \vartheta) = \exp(-[\boldsymbol{\vartheta}\times]) \tag{2.4}$$

where $\exp$ is the matrix exponential form. Although this representation is useful to analyze small rotations, when it comes to large displacements it obscures the periodicity of the attitude matrix as a function of ϑ.

2.2.4 Euler Angles

An Euler angle representation expresses a rotation by successive rotations around an initial frame, with constant column vectors and variable rotation angles. The vectors chosen in the classical Euler angle representation are the unit vectors for the orthogonal axis, such that

$$A_{zyx} = A(\mathbf{e}_z, \psi)A(\mathbf{e}_y, \theta)A(\mathbf{e}_x, \phi) \tag{2.5}$$

where

$$A(\mathbf{e}_z, \psi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.6a}$$

$$A(\mathbf{e}_y, \theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\in\theta & 0 & \cos\theta \end{bmatrix} \tag{2.6b}$$

$$A(\mathbf{e}_x, \phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix} \tag{2.6c}$$

Euler angles offer a transparent attitude representation due to their straightforward physical interpretation, in addition to not having redundant parameters. Due to the use of trigonometric functions and singularities for extreme values, their usefulness is limited.

2.2.5 Quaternion Representation

The quaternion concept was introduced as an extension to complex numbers, defined as

$$\mathbf{q}(\mathbf{e}, \vartheta) = \begin{bmatrix} \mathbf{e}sin(\vartheta/2) \\ cos(\vartheta/2) \end{bmatrix} \tag{2.7}$$

and obeys the parameterization and operations defined in Appendix A.2.

Unit quaternions, i.e. quaternions with unit norm, are used to parameterize rotations. The quaternion representation of the attitude matrix is defined as:

$$\mathbf{A}(\mathbf{q}) = (q_4^2 - ||\mathbf{q}_{1:3}||^2)I_3 - 2q_4[\mathbf{q}_{1:3}\times] + \mathbf{q}_{1:3}\mathbf{q}_{1:3}^T \tag{2.8}$$

In comparison to other attitude representations, the quaternion has fewer scalars (only 4), as well as

less computational intense functions for the useful operations. When compared to the attitude matrix, they have fewer parameters and only follow the unit norm constraint, as opposed to the six constraints. On the other hand, the quaternions representation's greatest disadvantage is that $\mathbf{q}$ and $-\mathbf{q}$ give the same attitude matrix. This can be avoided by restricting the quaternions to one hemisphere of the S^3, which is usually taken to be the $q_4 \geq 0$ hemisphere. However, this leads to the same type of discontinuities present in other attitude representations since it effectively diminishes the number of useful parameters.

Other methods to represent attitude, such as the (Modified) Rodrigues parameters, are not discussed in this work.

2.2.6 Attitude error representation

In order to evaluate estimation and control algorithms for the attitude, it is relevant to introduce attitude errors for the considered representations.

When it comes to the attitude matrix, the rotation between a reference frame $\mathcal{I}$ and a satellite body frame $\mathcal{B}$ can be represented as a small rotation $A_{\mathcal{B}\hat{\mathcal{B}}}$ between $\mathcal{B}$ and an estimated body frame $\hat{\mathcal{B}}$, such as

$$A_{\mathcal{B}\mathcal{I}} = A_{\mathcal{B}\hat{\mathcal{B}}} A_{\hat{\mathcal{B}}\mathcal{I}} \tag{2.9}$$

In the case of absence of errors, the matrix $A_{\mathcal{B}\mathcal{I}}$ is an identity matrix.

An attitude error represented by the rotation vector, with the small-angle approximation, is given by

$$A(\boldsymbol{\delta\vartheta}) = \exp(-[\boldsymbol{\delta\vartheta}\times]) \approx I_3 - [\boldsymbol{\delta\vartheta}\times] + \frac{1}{2}[\boldsymbol{\delta\vartheta}\times]^2 \tag{2.10}$$

The attitude error in terms of quaternions is given by

$$A(\boldsymbol{\delta}\mathbf{q}) \approx I_3 - 2[\boldsymbol{\delta}\mathbf{q}_{1:3}\times] + [\boldsymbol{\delta}\mathbf{q}_{1:3}\times]^2 \tag{2.11}$$

It is relevant to note that these expressions are equivalent through second-order considering

$$\boldsymbol{\delta\vartheta} = 2\boldsymbol{\delta}\mathbf{q}_{1:3} \tag{2.12}$$

2.3 Attitude Kinematics and Dynamics

In order to characterize the rotation of the satellite, the kinematics and dynamics must be defined. The kinematics show the rotational motion of the satellite body frame $\mathcal{B}$ with respect to the inertial frame $\mathcal{I}$. For this study, the point mass model is not satisfactory and a rigid body model is used instead. Again, this section is based on [11].

To describe attitude variations, the angular velocity $\boldsymbol{\omega}_{\mathcal{B}}^{\mathcal{B}\mathcal{I}}$ will be defined as:

$$\boldsymbol{\omega}_{\mathcal{B}}^{\mathcal{B}\mathcal{I}}(t) = \lim_{\Delta t \to 0} \frac{\Delta \boldsymbol{\vartheta}_{\mathcal{B}}^{\mathcal{B}\mathcal{I}}}{\Delta t} \tag{2.13}$$

The superscript $\mathcal{BI}$ means that it relates to the rotation from frame $\mathcal{I}$ to frame $\mathcal{B}$, and the subscript $\mathcal{B}$ means that the rotation vector is represented in frame $\mathcal{B}$. The attitude matrix is the fundamental representation of a rotation. The rotation fundamental equation of attitude kinematics, from the frame $\mathcal{I}$ to frame $\mathcal{B}$, is given by

$$\dot{\mathbf{A}}_{\mathcal{BI}} = -\mathbf{A}_{\mathcal{BI}} \left[\boldsymbol{\omega}_{\mathcal{I}}^{\mathcal{BI}}\times\right] = \mathbf{A}_{\mathcal{BI}} \left[\omega_{\mathcal{I}}^{\mathcal{IB}}\times\right] \tag{2.14}$$

The quaternion representation for attitude equivalent to $\mathbf{A}_{\mathcal{BI}}$ has the following equation for the kinematic motion:

$$\dot{\mathbf{q}}_{\mathcal{BI}} = \frac{1}{2}\boldsymbol{\omega}_{\mathcal{I}}^{\mathcal{IB}} \otimes \mathbf{q}_{\mathcal{BI}} = \frac{1}{2}\Omega(\boldsymbol{\omega}_{\mathcal{I}}^{\mathcal{IB}})\mathbf{q}_{\mathcal{BI}} = \frac{1}{2}\Xi(\mathbf{q})\omega_{\mathcal{I}}^{\mathcal{IB}} \tag{2.15}$$

where $\Xi(\mathbf{q})$ is computed according to Appendix A.2 and

$$\Omega(\boldsymbol{\omega}) = \begin{bmatrix} [\boldsymbol{\omega}\times] & -\boldsymbol{\omega} \\ \boldsymbol{\omega} & 0 \end{bmatrix} \tag{2.16}$$

In order to study how forces and torques are applied to the satellite and how they interact with its kinematics, the angular momentum must be taken into account. We will consider a satellite to be made of n point masses. The angular momentum with respect to the origin 0 of an inertial coordinate frame is defined in terms of the different masses m_i, positions $\mathbf{r}^{i0}$, and velocities $\mathbf{v}^{i0} = \dot{\mathbf{r}}^{i0}$ of the points relative to 0 by:

$$H^0 = \sum_{i=1}^{n} \mathbf{r}^{i0} \times m_i \mathbf{v}^{i0} \tag{2.17}$$

Newton's second law of motion states that $m_i \dot{\mathbf{v}}_{\mathcal{I}}^{i0} = F_{\mathcal{I}}^{i}$ in an inertial reference frame, and $\mathbf{v}^{i0} \times \mathbf{v}^{i0} = 0$, so the angular momentum obeys the equation:

$$\dot{H}_{\mathcal{I}}^{0} = \sum_{i=1}^{n} \mathbf{r}^{i0} \times F_{\mathcal{I}}^{iext} \equiv L_{\mathcal{I}}^{0} \tag{2.18}$$

where F^{iext} is the force exerted on i by everything external to the system of mass points and $L_{\mathcal{I}}^{0}$ is the net torque about 0 assuming that the force exerted by a mass point on itself is zero and the strong law of action and reaction [1].

By defining the centre of mass c as a collection of mass points and considering that the motion of the centre of mass and the motion of the mass points about their centre of mass are uncoupled, the fundamental equation of attitude dynamics is given as:

$$\dot{H}_{\mathcal{I}}^{c} \equiv L_{I}^{c} \tag{2.19}$$

By defining a rigid body, a body frame where the distance of any point mass to the centre of the frame, r^{ic} remains constant in time regardless of external forces or moments exerted on it, the moment of inertia (MOI) can be defined as:

$$J_{\mathcal{B}}^{c} = -\sum_{i=1}^{n} m_i [r_{\mathcal{B}}^{ic}\times]^2 \tag{2.20}$$

[1] the force between two mass points acts along the line between them

and can be used to describe the angular momentum in a body frame:

$$H_{\mathcal{B}}^{c} = J_{\mathcal{B}}^{c}\omega_{\mathcal{B}}^{\mathcal{BI}} \tag{2.21}$$

External torques $L_{\mathcal{B}}^{c}$ are often more easily computed in the body frame, so (2.19) can be replaced by:

$$\dot{H}_{\mathcal{B}}^{c} = L_{\mathcal{B}}^{c} - \omega_{\mathcal{B}}^{\mathcal{BI}} \times H_{\mathcal{B}}^{c} \tag{2.22}$$

2.4 Orbit Dynamics

Explaining the motion of celestial bodies has been subject of study for many centuries. One of the biggest breakthroughs was accomplished by Johannes Kepler, to be later refined by Isaac Newton. Kepler proposed three laws of planetary motion in the 17^{th} century based on observation by Tycho Brahe.

Newton's law of gravitation states that any two bodies attract each other with a force proportional to the product of their masses and inversely proportional to the square of the distance between them [12]. The equation for the force caused by gravity is

$$F_G = -\frac{GMm}{r^3}\mathbf{r} = -\frac{\mu m}{r^3}\mathbf{r} \tag{2.23}$$

where G is the universal constant of gravitation, M is the mass of the Earth, m is the mass of the satellite, $\mathbf{r}$ is the vector between them (where $r = ||\mathbf{r}||$) and μ is the Earth's gravitational constant.

Combining Newton's second law with his law of gravitation, it gives the equation for the acceleration of the satellite, such as

$$\ddot{\mathbf{r}} + (\mu r^{-3})\mathbf{r} = \mathbf{0} \tag{2.24}$$

The previous equation is the two-body equation of motion and gives the satellite's orbit. It assumes that gravity is the only force applied and that the Earth is spherically symmetric, as well as that the Earth and the satellite are the only two bodies in the system and can be treated as a point-mass. These concerns will be considered in Section 2.5, whilst (2.24) can be modified to account for perturbations, such as

$$\ddot{\mathbf{r}} = -(\mu r^{-3})\mathbf{r} + \mathbf{a}_p \tag{2.25}$$

The motion of a satellite could therefore be characterized by a fixed elliptical orbit in space with the Earth being at one of the foci. This orbit can be specified by the six elements of the satellite's position and velocity vectors at a specific epoch. However, an alternative representation that uses the six Keplerian elements was developed to provide a physical characterization of the orbit [10] and are the following:

- **Semimajor**: Longest dimension of the orbit;
- **Eccentricity**: Ovalness of the ellipse;

- **Inclination**: Angle of the orbital plane relative to the Earth's equatorial plane;
- **Right ascension of the ascending node (RAAN)**: Angle in the equatorial plane between the ascending node of the satellite's orbit and the vertical equinox;
- **Argument of perigee or periapsis**: Orientation of the elipse in orbital plane;
- **True anomaly**: Position of the satellite in the orbit.

2.5 Spacecraft perturbations

Spacecraft perturbations can be divided into perturbation forces, which affects accelerations, and perturbation torques. The forces have an impact in the orbital motion of the satellite and deviate it from the Keplerial moment, contributing to the $\mathbf{a}_p$ in (2.25). On the other hand, the torques cause changes in the angular rate of the satellite, depicted in the $L^c_{\mathcal{B}}$ in (2.22). This section models these perturbations, based on [11] [12] [13].

2.5.1 Perturbative forces

Perturbative forces are connected to (2.25) and impose unwanted accelerations that alter the orbital motion of the satellite. For satellite in LEO, the most significant forces are aerodynamic drag, non-spherical Earth perturbations and the gravitational influence from the Sun and Moon. These forces are depicted through

$$\mathbf{a}_p = \mathbf{a}_d + \mathbf{a}_{sph} + \mathbf{a}_{sun} + \mathbf{a}_{moon} \tag{2.26}$$

Aerodynamic drag

The atmospheric drag arises from the friction between the atmosphere and the satellite, causing a hinder force of the latter. This perturbation has the biggest impact for low-Earth orbits since the density of the atmosphere decreases exponentially with an increase in altitude.

In order to model the drag (and torque), the satellite is modelled as a collection of N flat plates of area S_i and outward normal unit normal vector $n^i_{\mathcal{B}}$ expressed in the satellite body fixed coordinate system. The drag and torque are dependent on velocity of the satellite concerning the atmosphere in the inertial frame $\mathbf{v}_{\mathbf{rel}\mathcal{I}}$ since the atmosphere is not stationary in that frame.

The inclination of the ith plate in relation to the relative velocity is given as

$$\cos\theta^i_d = (A^T n^i_{\mathcal{B}})^T (\frac{\mathbf{v}_{\mathbf{rel}\mathcal{I}}}{\mathbf{v}_{\mathbf{rel}\mathcal{I}}}) \tag{2.27}$$

where A is the attitude matrix that rotates the environment to the body frame.

The drag force is modelled through the following fluid mechanics equation

$$\mathbf{a}_d = -\frac{1}{2}\rho||\mathbf{v}_{\mathbf{rel}\mathcal{I}}||\mathbf{v}_{\mathbf{rel}\mathcal{B}}C_d\sum_{i=1}^{N}S_i\cos\theta_d^i \tag{2.28}$$

where ρ is the air density and C_d is the drag coefficient of the satellite. Only plates with $\cos\theta_d^i > 0$ are included in the summation.

There are various models to determine the air density but the simplest one in the Exponential decaying model atmosphere, given by

$$\rho = \rho_0\exp(-\frac{h-h_0}{H}) \tag{2.29}$$

where ρ_0 and h_0 are reference density and height respectively, h is the height and H is the scalar height, which is established according to the height. This model is fully static (independent of time) and doesn't take into account temperature changes.

Non-spherical Earth perturbation

The Earth is not a perfect sphere and mass is not distributed uniformly throughout it, ergo the gravity field will also be nonuniform. In order to translate the point-mass approach of Newton's law of universal gravitation to a more realistic approach models were created based on the spherical harmonic expansion, such as the Earth Gravitation Model 2008 (EGM2008).

Sun and Moon influence

Even though the Earth is the celestial body that the satellite orbits, every body with a mass has a gravitational field. Namely, in the Moon and Sun gravitational forces are classified as perturbations, though they are not subject to heterogeneous constitution since they are too distant to be noticeable. They can be modelled with Newton's universal gravitational law, such as

$$\mathbf{a}_{sun} = \frac{\mu_{sun}M_{sun}}{||\mathbf{r}_{sun}||^2}\frac{\mathbf{r}_{sun}}{|\mathbf{r}_{sun}|} \tag{2.30}$$

$$\mathbf{a}_{moon} = \frac{\mu_{moon}M_{moon}}{||\mathbf{r}_{moon}||^2}\frac{\mathbf{r}_{moon}}{|\mathbf{r}_{moon}|} \tag{2.31}$$

where μ are the gravitational constants of the celestial bodies, M the mass and $\mathbf{r}$ the distance between the centre of the body and the satellite point of mass.

2.5.2 Perturbative torques

Perturbative torques can be divided into internal or external. Internal perturbative torques include deployment of payload elements, propellant slosh and reaction wheel imbalances. These are often negligible in comparison to external torques and are not considered in this work. The external torques result from the interaction with the environment and include aerodynamic, gravitational and magnetic

torque. These torques are depicted through

$$L_p = L_p^c + L_d + L_m \tag{2.32}$$

Gravitational torque

The non-symmetry of a rigid body leads to nonuniform gravitational field over it, causing the appearance of a torque about its centre of mass. This torque can be computed as

$$L_g^c = 3(\frac{\mu}{r^3})\mathbf{o} \times J^c \cdot \mathbf{o} \tag{2.33}$$

where $\mathbf{r}$ is the distance between the Earth and the satellite (where $r = ||\mathbf{r}||$), $\mathbf{o}$ is the representation of a nadir-pointing unit vector in the body frame and J^c is the moment of inertia tensor about the centre of mass.

Aerodynamic torque

The aerodynamic torque translates how the aerodynamic forces instigate angular motion.

The inclination of the ith plate to the relative velocity is given by

$$\cos\theta_d^i = \frac{n_{\mathcal{B}}^i \cdot \mathbf{v_{rel\mathcal{B}}}}{||\mathbf{v_{rel\mathcal{I}}}||} \tag{2.34}$$

The aerodynamic force on the ith plane is

$$\mathbf{a}_d^i = -\frac{1}{2}\rho||\mathbf{v_{rel\mathcal{I}}}||\mathbf{v_{rel\mathcal{B}}}C_d\text{max}(\cos\theta_d^i, 0) \tag{2.35}$$

The aerodynamic torque on the satellite is then

$$L_d = \sum_{i=1}^{N} \mathbf{r}^i \times \mathbf{a}_d^i \tag{2.36}$$

where $\mathbf{r}^i$ is the vector from the satellite centre of mass to the centre of pressure of the ith plate.

Magnetic torque

When a satellite is immersed in a magnetic field $\mathbf{B}$ and has a magnetic dipole **m**, a torque is created. Permanent magnets and current loop contribute to the magnetic dipole, whilst it is possible to balance these elements in order to reduce their influence. The existence of a nonzero magnetic field in the neighbourhood of the Earth is what allows for magnetic torque control. This phenomenon follows

$$L_m = \mathbf{m} \times \mathbf{B} \tag{2.37}$$

Chapter 3

Estimation

Before attempting the pointing maneuver, attitude knowledge is essential. Sensors are used to convey information about the environment by providing measurements, such as the angular rate or the direction of the Sun. The technologies used have been subject of study recently and major improvements have been achieved, where SWaP has been minimized whilst maintaining (or even improving) their performance capabilities. However, the measurements provided by the sensors are corrupted with various types of noise and by themselves are not sufficient for attitude determination. Algorithms are used to fuse the sensors' measurements and filter the high and low-frequency noise. In this process, an estimate of the measurements, based on the satellite's location is necessary. The available sensors are presented in Section 3.1.

Multiple methods have been developed to give an estimate of attitude, divided in two approaches: static and recursive methods. The first provide an estimate when two measurements are available using deterministic methods. The latter use stochastic methods and combines the model of the systems with corrections through measurements. The inclusion of the measurements serves to mitigate the model uncertainties and to compensate for unmodelled disturbances. This coalition makes these types of methods more effective. In Section 3.2 the estimation methods are discussed and in Section 3.3 the attitude determination methods will be presented.

3.1 Estimation Sensors

The sensors used are the interface between the environment and the estimation algorithms, which through (noisy) measurements deliver information on the behaviour of the satellite or physical phenomena. Technological advancements have allowed for the use of a wider range of sensors to be installed in CubeSats, whilst still maintaining a high accuracy.

Each sensor possesses its advantages and shortcomings and before choosing the estimation hardware to be used in a specific mission, possible candidates should be reviewed. Sensors could be separated in two categories, regarding if they measure the motion of the satellite, called inertial, or if they are used to derive heading information from other sources, such as the sun or the Earth's magnetic

field.

3.1.1 Gyroscope

When it comes to inertial sensors, the most common is the gyroscope [14], used to measure angular velocity by angular displacement change over a period of time, in order to predict attitude changes between estimates. There are various types of gyroscopes available, ranging from the more accessible Micro-Electro-Mechanical Systems (MEMS) [15], to the mid-range fibre-optic gyros (FOGs) and most accurate ring laser gyros (RLGs). Knowledge of attitude variation is essential when using estimation algorithms that use the satellite dynamics. The selection criteria for this sensor are the required accuracy, bias and drift since they will a cause cumulative error that is not possible to ignore due to the absence of an absolute measurement.

3.1.2 Magnetometer

Regarding positioning sensors, which measure changes in the environment, on the lower side of the spectrum when it comes to price, there is the magnetometer. It can be used by itself [16] in order to estimate attitude and rate, but it is mostly used in complement with other sensors [17]. It provides a measurement of the magnetic field in the sensor's axis that must be compared to an estimate of the magnetic field which is obtained with estimates of International Geomagnetic Reference Field (IGRF), based on an estimation of instantaneous orbital position.

3.1.3 Sun Sensor

The Sun sensors take advantage of the brightest reference in space orbit to generate a sun vector in relation to the satellite. Since the measured Sun vector is the only point of reference, it doesn't provide complete information by itself for attitude estimation, and should be crossed with a non-collinear vector, as the Earth vector [18], obtained through other various types of sensors. The Earth's albedo can also be a source of measurement disturbance and must be taken into account [19].

There are different types of builds when it comes to power, whilst passive sensors use the irradiated energy to output the result, active sensors use multidetector arrays to process the digital image of incidence. The most relevant shortcoming of sun sensors is the unavailability of measurements whilst the sun is not in the satellite's FOV. The eclipse will hinder the performance of algorithms highly reliant on these sensors. Sun sensors can also be divided in fine or course based on their accuracy.

3.1.4 Star Tracker

Although the sun is the closest star to the Earth, it doesn't mean other can't be used as reference. Star sensors are cameras mounted on the satellite that compare the image being received with a reference map and estimate the attitude. Since the position of the stars can be measured with high accuracy,

it is possible to obtain sub-arcminute attitude estimation [3] and can be used as the only type of position-sensing sensor [20]. Star trackers use the 2D information of the stars within their FOV, which allows for absolute attitude estimation by itself and does not rely on other measurements as the Sun sensors.

The star imaging technology can also be used to create a stellar gyroscope. It compares two sequential images and by correlating them to the star maps, it is able to calculate a difference in rotation on the object on which it is mounted. This method is highly reliable and accurate since it is a differential method (removes noise or dead pixels) and is able to cancel accumulated bias while the body is detected as stationary.

3.1.5 Horizon Sensors

These sensors detect the Earth's horizon through infrared sensors. Although there are various methods of doing so, the most used is edge detection where the sensor is positioned to detect the edge of the earth. This information can be used to compute the nadir vector. There are some unavoidable errors in the horizon measurement due to the earth's atmosphere, which irradiates decreasing intensity from the true horizon.

3.1.6 Global Navigation Satellite System

Global Navigation Satellite System (GNSS) measurements cannot be used directly to estimate attitude but can be useful to have a very good estimate of the satellite position, which could be used in conjunction with other measurements. In the specific case of GNSS, the most utilized navigation system, the cost is high so it is not usually present in a Nano Satellite.

3.1.7 Beam detection

The laser beam emitted from the GS marks not only the desired orientation of the satellite, but also is another way to gain knowledge of the Earth's position. The satellite is composed by a telescope that receives and conditions the optical signal to a quad-cell that processes it. Although the QuantSat's optic payload is still under development, its performance can be estimated based on other implementations, as discussed in the section below. In order to use the Beam's attitude measurement, the optical signal must be within the FOV of the telescope lens, i.e. when the pointing accuracy is lower than the FOV.

3.1.8 Sensors Selection

ADCS systems are an essential component of a satellite so a lot of different combinations of sensors have been tested. In order to choose the appropriate hardware to suit the mission, not only the pointing requirements must be taken into account but also the 3U CubeSat restrictions, such as volume and weight. The optic payload occupies a large volume in the CubeSats. When it comes to available sensors, the use of a GNSS system and the Beam Detection have been established since they are included in the payload itself to provide timing and quantum communications respectively.

Although requirements compliance verification can only be done in simulation, literature provides a few guidelines that can be used. They are by no means an objective approach and are highly dependent on the sensors' relative quality and the algorithms used to process the data. Although pointing accuracy is the principal selection criteria, mission requirements (e.g. Earth pointing in eclipse) and other influences (e.g. redundancy, fault tolerance) must be taken into account. These guidelines are presented in table 3.1.

Required accuracy (°)	**Effect on ADCS**
>5	Major cost savings, sun sensors and magnetometers adequate
1-5	Sun sensors and horizon sensors adequate, magnetometers useful
0.1 - 1	Star tracker or horizon sensors, gyroscope, magnetometers for light satellite
<0.1	Star tracker needed

Table 3.1: Comparison of attitude sensors based on [12]

The table above does not include the beacon detection since it is not a conventional sensor. In [7], the attitude knowledge accuracy of a similar system is evaluated to be of 0.0015 $^\circ$ (mean), which is comparable to the typical accuracy of a Star tracker and superior to an Earth Sensor [3]. With this in mind, and considering that these are costly and redundant sensors, they are disregarded in the implementation.

Since the proposed ADCS works in two separate stages, two distinct analyses could be conducted. However, if an estimation error of less that 10° can be guaranteed, the first staged can be disregarded. This work will focus on this goal.

For the sensors, when the required accuracy is not high, sun sensors and magnetometers are enough to obtain all attitude information. The downfall of this sensor suite is the eclipse, which will be addressed further on. For more accurate systems, horizon sensors or even star trackers must be used, with the addition of the gyroscope for attitude propagation.

3.1.9 Sensors Model

Gyroscope

The angular velocity measurement $\hat{\boldsymbol{\omega}}$ from the gyroscope is considered to be corrupted by noise, η_{arw}, as well as a slow-varying bias, $\boldsymbol{\beta}_g$. As such, the gyroscope model is given as

$$\hat{\boldsymbol{\omega}} = \boldsymbol{\omega} + \boldsymbol{\beta}_g + \eta_{arw} \tag{3.1}$$

$$\dot{\boldsymbol{\beta}}_g = 0. \tag{3.2}$$

The sensor perturbation η_{arw} can be interpreted as an independent zero-mean Gaussian white-noise process.

Magnetometer

The magnetic field measurement is defined as the true magnetic field vectors, corrupted by a bias $\boldsymbol{\beta}_B$ and Gaussian white-noise η_B, such as

$$\hat{\mathbf{B}} = \mathbf{B} + \boldsymbol{\beta}_B + \eta_B \tag{3.3}$$

Beacon

The optical measurement for the GS laser beacon is given as

$$\hat{b}_b = A_{\eta_b} b_b = A(e_1, \phi_b) A(e_2, \theta_b) A(e_3, \xi_b) b_b \tag{3.4}$$

where b_b is the normalized direction vector of the GS, in the body frame. The euler angles used to generate the noise rotation matrix A_{η_b} can be characterized as Gaussian white-noise. This measurement model is based on Star Trackers, converted from quaternions to rotation matrix [21].

Sun sensors

Similarly to the Beacon, the Sun sensor measurements are given as

$$\hat{b}_s = A_{\eta_s} b_s = A(e_1, \phi_s) A(e_2, \theta_s) A(e_3, \xi_s) b_s \tag{3.5}$$

where b_{sun} is the normalized Sun vector, in the body frame. The euler angles used to generate the noise rotation matrix $A_{\eta_{sun}}$ can be characterized as Gaussian white-noise, with different variances for boresight and cross-boresight axis.

GNSS

The GNSS measurement is defined in the Inertial frame, given by the true position $\mathbf{p}_{sat}$ of the sensor with a sphere of uncertainty, such as:

$$\hat{\mathbf{r}}_{sat} = \mathbf{r}_{sat} + \begin{bmatrix} r\sin(\phi)\cos(\theta) \\ r\sin(\phi)\sin(\theta) \\ r\cos(\phi) \end{bmatrix} \tag{3.6}$$

3.2 State Estimation Algorithms

State estimation refers to the process of estimating the state of a system, based on a set of measurements corrupted by random noise, with known or estimated statistics. The dynamic model of a system is used in conjunction with measurements to find the best estimation. This type of approach also employs sensor fusion, since low-frequency measurements can be combined with kinematic or dynamic models, and high-frequency sensor measurements allow the state to be propagated. It also allows for a representation of a state that is not directly measurable through the use of sensors and models, as well as filtering of noisy data. In comparison with static methods, the main advantage is that it incorporates

memory of previous observations.

This section reviews the concepts of state and observation models, establishes the fundamentals of the Kalman Filter (KF), introducing then the formulation of linear and non-linear KF. Then, it is applied to attitude estimation and an onboard filter is formulated. Other estimation methods will also be discussed.

3.2.1 State and Observation Models

The description of a system is based on the state variables, which usually refer to a physical concept, directly related to the storage or dissipation of energy and are collected in the state vector. The size of this vector is known as the order of the system. When a dynamic system is being modelled, differential equations are used to characterize the interactions between the different states and their number is connected to the order of the system. In mechanical systems, the state usually includes the linear or angular position, as well as the corresponding velocity.

A n-th order system can be written as n first order equations that form the following system of state equations:

$$\dot{\mathbf{x}}(t) = F(\mathbf{x}(t), \mathbf{u}(t), t), \quad \mathbf{x}(t_0) = \mathbf{x}_0 \tag{3.7}$$

where F is an $n \times 1$ sufficiently differentiable vector, $\mathbf{x}$ is an $n \times 1$ state vector, and $\mathbf{u}$ is a $q \times 1$ vector that represents any input that does not depend on the state elements.

If (3.7) is particularized for linear systems, where the superposition principle is satisfied, the following system is obtained:

$$\dot{\mathbf{x}}(t) = F(t)\mathbf{x}(t) + B(t)\mathbf{u}(t), \quad \mathbf{x}(t_0) = \mathbf{x}_0 \tag{3.8}$$

where F is the $n \times n$ state matrix and B is the $n \times q$ input matrix. The solution of (3.8) is given by:

$$\mathbf{x} = \Phi(t, t_0)\mathbf{x}_0 + \int_{t_0}^{t} \Phi(t, \tau)B(\tau)\mathbf{u}(\tau)d\tau \tag{3.9}$$

where $\Phi(t, t_0)$ is known as the state transition matrix, which has the following properties:

$$\Phi(t_0, t_0) = I \tag{3.10a}$$

$$\Phi(t_0, t) = \Phi^{-1}(t, t_0) \tag{3.10b}$$

$$\Phi(t_2, t_0) = \Phi(t_2, t_1)\Phi(t_1, t_0) \tag{3.10c}$$

$$\dot{\Phi}(t, t_0) = F(t)\Phi(t, t_0). \tag{3.10d}$$

Observation vectors are used to disclose how the sensor measurements relate to the systems' states. The number of elements in an observation sensor is usually determined by the number of types of sensors and their multiplicity. The general form of an observation vector is given by

$$\mathbf{y}(t) = h(\mathbf{x}(t), \mathbf{u}(t), t) \tag{3.11}$$

where h is a $p \times 1$ observation. In linear systems, the previous equation can be written as:

$$\mathbf{y}(t) = H(t)\mathbf{x}(t) + D(t)\mathbf{u}(t) \tag{3.12}$$

where H is the $p \times n$ observation/sensitivity matrix and D is the $p \times q$ direct transmission matrix.

Since sensors and actuators do not work in continuous-time, it is relevant to introduce a discrete-time representation of the dynamic systems and observations. The following considerations are based in a zero-order hold sample-type since it is the most common. As the sampling interval Δt decreases, the sampled signal resembles more precisely the real signal.

When considering the first sample interval Δt, F and B constant, (3.9) simplifies to:

$$\Delta\mathbf{x} = \Phi\mathbf{x}(0) + \Gamma\mathbf{u}(0) \tag{3.13}$$

with $\Phi \equiv e^{F\Delta t}$ and $\Gamma \equiv [\int_0^{\Delta t} e^{Ft} dt]B$.

Extending (3.13) for $k+1$ samples:

$$\mathbf{x}[(k+1)\Delta t] = \Phi\mathbf{x}(k\Delta t) + \Gamma\mathbf{u}(k\Delta t)) \tag{3.14}$$

The common notation for (3.14) is:

$$\mathbf{x}_{k+1} = \Phi\mathbf{x}_k + \Gamma u_k \tag{3.15}$$

and (3.12) can also be written in discrete-time to complete the state-space description such as:

$$\mathbf{y}_k = H\mathbf{x}_k + D\mathbf{u}_k \tag{3.16}$$

3.2.2 Kalman Filter

The Kalman filter was proposed in the 1950s by Rudolph Emil Kalman, as a technique for filtering and prediction in linear systems [22]. The filter uses knowledge of the system, along with its related uncertainties, and fuses any available measurements, with corresponding errors, to create an optimum estimate of the systems' variables/states. When applied to a linear system where the process noise and the measurement noise follow statistical properties, the KF satisfies the optimality criterion that minimizes the trace of the covariance of the estimation error between the true state and the estimated one. The filter is recursive since it has memory of previous measurements, but it only needs to process the previous iteration's results.

The KF finds the optimal estimate of $\mathbf{x}$, given by $\hat{\mathbf{x}}$. A residual error is defined as the difference between the true value and the estimate

$$\boldsymbol{\epsilon} = \mathbf{x} - \hat{\mathbf{x}} \tag{3.17}$$

with covariance

$$P = E\{\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T\} \tag{3.18}$$

The three steps of the KF are shown in Alg. 1. The Initialization step specifies the initial values for $\hat{\mathbf{x}}_0$ and P_0. The following two steps alternate recursively, minimizing the trace of P. The Propagation step is based on the system kinematics and dynamics and estimates the state $\hat{\mathbf{x}}_{k+1}^-$ and covariance P_{k+1}^- at from t_k to time t_{k+1}. These are known as the *a priori* estimates and are denoted as $\square^-$, since they do not include the observations. The Update step utilizes the observations at $\mathbf{y}_{k+1}$ to correct the *a priori* estimates, providing the *a posteriori* estimates, $\hat{\mathbf{x}}_{k+1}^+$ P_{k+1}^+, denoted by $\square^+$. These values are used in the following iteration of the Propagation step.

Algorithm 1 Linear Kalman Filter

```
Initialization: x̂_0, P_0
loop
    Propagation: x̂⁻_{k+1}, P⁻_{k+1}
    if observations ready: y_{k+1} then
        Update: x̂⁺_{k+1}, P⁺_{k+1}
    end if
     return x̂⁺_k, P⁺_k
end loop
```

The KF algorithm starts by expanding the linear dynamic model in (3.7) with noise terms

$$\dot{\mathbf{x}}(t) = F(t)\mathbf{x}(t) + B(t)\mathbf{u}(t) + G(t)\mathbf{w}(t) \tag{3.19}$$

where $\mathbf{w}$ is a zero-mean $p \times 1$ Gaussian white-noise process with covariance matrix Q, and G is $n\times$ a process noise distribution matrix.

According to (3.9) and assuming $\mathbf{u}(t)$ is a piece-wise constant, the solution of (3.19) at discrete time instants can be written as

$$\mathbf{x}_{k+1} = \Phi_k\hat{\mathbf{x}}_k + \Gamma_k\mathbf{u}_k + \Upsilon_k\mathbf{w}_k \tag{3.20}$$

where $\mathbf{w}_k$ is a zero-mean $p \times 1$ Gaussian white-noise process with covariance Q, which results from sampling a continuous-time random process. The state estimate has the following continuous-time expression

$$\dot{\hat{\mathbf{x}}}(t) = F(t)\mathbf{x}(t) + B(t)\mathbf{u}(t) \tag{3.21}$$

and the discrete-time expression

$$\hat{\mathbf{x}}_{k+1} = \Phi\hat{\mathbf{x}}_k + \Gamma u_k \tag{3.22}$$

The previously defined covariance obeys the equation

$$\dot{P} = E\{\dot{\boldsymbol{\epsilon}}\boldsymbol{\epsilon}^T\} + E\{\boldsymbol{\epsilon}\dot{\boldsymbol{\epsilon}}^T\} = FP + PF^T + GQG^T \tag{3.23}$$

which is solved for $P(t)$ with initial condition P(t_0) = P_0. It can also be shown that the covariance evolves according to [23]

$$P_{k+1} = \Phi_k P_k \Phi_k^T + \Upsilon Q_k \Upsilon^T \tag{3.24}$$

Since the measurements are performed in discrete-time due to the sensors being analogue, they are modelled by

$$\mathbf{y}_k = H_k \mathbf{x}_k + \mathbf{v}_k \tag{3.25}$$

where $\mathbf{v}_k$ zero-mean $m \times 1$ Gaussian white-noise process vector with covariance R_k.

Propagation Step

The propagation step uses the previously defined equations for $\hat{\mathbf{x}}$ and P to transport the *a posteriori* estimates at time t_k to the *a priori* estimates at time t_{k+1}

Update Step

When a measurement is $\mathbf{y}_k$ is available at time t_k. The linear discrete-time update equation is given by

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + K_k[\mathbf{y}_k - H_k \hat{\mathbf{x}}_k^-] \tag{3.26}$$

The updated covariance expression is given by computing

$$P_k^+ = E\{(\mathbf{x} - \hat{\mathbf{x}}^+)(\mathbf{x} - \hat{\mathbf{x}}^+)^T\} \tag{3.27}$$

where I is a $n \times n$ identity matrix. Substituting (3.25) and (3.26) into (3.27) leads to tolerance

$$P_k^+ = [I - K_k H_k] P_k^- [I - K_k H_k]^T + K_k R_k K_k^t \tag{3.28}$$

where P_k^- is given from the propagated system. The optimal K_k gain is determined by minimizing the trace of the updated covariance, such as

$$J_k = tr P_k^+ \tag{3.29}$$

that leads to

$$K_k = P_k^- H_k^T [H_k P_k^- H_k^T + R_k]^T \tag{3.30}$$

Substituting (3.30) into (3.28) gives

$$P_k^+ = [I - K_k H_k] P_k^- \tag{3.31}$$

This is the most useful covariance update, but (3.28) can also be used when numerical instabilities are a concern.

The discrete-time KF is summarized in Tab. 3.2, where $N(\mathbf{0}, R_k)$ denotes a Gaussian distribution with zero mean and covariance R_k.

Model	$\mathbf{x}_{k+1} = \Phi_k \mathbf{x}_k + \Gamma_k \mathbf{u}_k + \Upsilon_k \mathbf{w}_k,$	with $\mathbf{w}_k \sim N(0, \mathbf{Q}_k)$
	$\mathbf{y}_k = H_k \mathbf{x}_k + \mathbf{v}_k,$	with $\mathbf{v}_k \sim N(0, R_k)$
Initial Condition	$\hat{\mathbf{x}}(t_0) = \hat{\mathbf{x}}_0$	
	$P(t_0) = P_0$	
Propagation	$\hat{\mathbf{x}}_{k+1}^- = \Phi_k \hat{\mathbf{x}}_k^+$	
	$P_{k+1}^- = \Phi_k P_k^+ \Phi_k^T + \Upsilon_k Q_k \Upsilon_k^T$	
Gain	$K_k = P_k^- H_k^T \left[H_k P_k^- H_k^T + R_k \right]^{-1}$	
Update	$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + K_k \left[\mathbf{y}_k - H_k \hat{\mathbf{x}}_k^- \right]$	
	$P_k^+ = \left[I - K_k H_k \right] P_k^-$	

Table 3.2: Summary of KF equations.

3.2.3 Extended Kalman Filter

Many problems, including attitude estimation, involve nonlinear models. To tackle these cases, the Extended Kalman Filter (EKF) was created, which revolves around linearization about the current best estimate and using the methods defined in Section 3.2.2. This section is based on [24].

Consider the following general continuous nonlinear model:

$$\mathbf{x}(t) = f(\mathbf{x}(t), \mathbf{u}(t), t)) + G(t)\mathbf{w}(t), \mathbf{w}(t) \sim N(\mathbf{0}, Q(t)) \tag{3.32}$$

$$\mathbf{y}(t) = h(\mathbf{x}(t, t)) + \mathbf{v}(t), \mathbf{v}(t) \sim N(\mathbf{0}, R(t)) \tag{3.33}$$

and the discrete-time equivalent

$$\mathbf{x}_{k+1} = f_k(\mathbf{x}_k, \mathbf{u}_k) + \Upsilon_k \mathbf{w}_k, \mathbf{w}_k \sim N(\mathbf{0}, Q(t)) \tag{3.34}$$

$$\mathbf{y}_k = h_k(\mathbf{x}_k) + \mathbf{v}_k, \mathbf{v}_k \sim N(\mathbf{0}, R(t)) \tag{3.35}$$

where $f(\mathbf{x}(t), \mathbf{u}(t), t)$ and $h(\mathbf{x}(t), t)$ are continuously differentiable nonlinear functions, and $f_k(\mathbf{x}_k, \mathbf{u}_k)$ and $h_k(\mathbf{x}_k)$ are discrete-time differentiable nonlinear functions and the other variables follow the definition in Section 3.2.2.

The probability density functions of the noise terms are altered when transmitted through nonlinear elements, which means that a Gaussian input doesn't inherently cause a non-Gaussian response. In order to overcome this problem, nonlinear systems are approximated by a linear one, through the first-order Taylor expansion, by assuming that the true state is sufficiently close to the estimate and therefore are valid for small perturbations.

The first-order expansion for $f(\mathbf{x}(t), \mathbf{u}(t), t)$ and $h(\mathbf{x}(t), t)$, about a nominal state $\bar{\mathbf{x}}(t)$ is given by

$$f(\mathbf{x}(t), \mathbf{u}(t), t)) \simeq f(\bar{\mathbf{x}}(t), \mathbf{u}(t), t)) + \frac{\partial f}{\partial \mathbf{x}}|_{\bar{\mathbf{x}}(t)} [\mathbf{x}(t) - \bar{\mathbf{x}}(t)] \tag{3.36}$$

$$h(\mathbf{x}(t, t)) = h(\bar{\mathbf{x}}(t, t)) + \frac{\partial h}{\partial \mathbf{x}}|_{\bar{\mathbf{x}}(t)} [\mathbf{x}(t) - \bar{\mathbf{x}}(t)] \tag{3.37}$$

where $\bar{\mathbf{x}}(t) = \hat{\mathbf{x}}(t)$ in the case of the EKF. This lead to the following estimation error dynamics

$$\Delta\dot{\mathbf{x}}(t) = [F(t) - K(t)H(t)]\Delta\mathbf{x}(t) - G(t)\mathbf{w}(t) + K(t)\mathbf{v}(t) \tag{3.38}$$

where $\Delta\dot{\mathbf{x}}(t) = \hat{\mathbf{x}} - \mathbf{x}(t)$, $H(t)$ is the Kalman gain matrix and

$$F(t) = \frac{\partial f(t)}{\partial \mathbf{x}}|_{\hat{\mathbf{x}}(t),\mathbf{u}(t)} \tag{3.39a}$$

$$H(t) = \frac{\partial h(t)}{\partial \mathbf{x}}|_{\hat{\mathbf{x}}(t)} \tag{3.39b}$$

$$\tag{3.39c}$$

As (3.38) is linear, the EKF can be formulated based on error estimation.

The filter structure can be summarized in Alg. 2 and yields the following equations in Tab. 3.3.

Model	$\mathbf{x}_{k+1} = f_k(\mathbf{x}_k, \mathbf{u}_k) + \Upsilon_k \mathbf{w}_k,$	$\mathbf{w}_k \sim N(0, Q_k)$
	$\mathbf{y}_k = h_k(\mathbf{x}_k) + \mathbf{v}_k,$	$\mathbf{v}_k \sim N(0, R_k)$
Initial condition	$\hat{\mathbf{x}}(t_0) = \hat{\mathbf{x}}_0$	
	$P(t_0) = P_0$	
Propagation	$\hat{\mathbf{x}}_{k+1}^- = f_k\left(\hat{\mathbf{x}}_k^+\right)$	
	$P_{k+1}^- = \Phi_k P_k^+ \Phi_k^T + \Upsilon_k Q_k \Upsilon_k^T,$	$\Phi_k \equiv \frac{\delta f_k}{\delta \mathbf{x}}\Big\vert_{\hat{\mathbf{x}}_k^+}$
Gain	$K_k = P_k^- H_k^T \left[H_k P_k^- H_k^T + R_k\right]^{-1},$	$H_k \equiv \frac{\delta h_k}{\delta \mathbf{x}}\Big\vert_{\hat{\mathbf{x}}_k^-}$
Update	$\Delta\hat{\mathbf{x}}_k^+ = K_k\left[\mathbf{y}_k - h_k\left(\hat{\mathbf{x}}_k^-\right)\right]$	
	$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \Delta\hat{\mathbf{x}}_k^+$	
	$P_k^+ = [I - K_k H_k] P_k^-$	

Table 3.3: Summary of EKF equations.

Algorithm 2 Extended Kalman Filter

```
Initialization: x̂_0, P_0
loop
    Propagation: x̂⁻_{k+1}, P⁻_{k+1}
    if observations ready: y_{k+1} then
        calculate kalman gain: K_k
        Calculate uncertainty: δx̂_{k+1}
        Update: x̂⁺_{k+1}, P⁺_{k+1}
        Reset: δx̂_{k+1} = 0
    end if
     return x̂⁺_k, P⁺_k
end loop
```

For filter performance, non-Gaussian measurement and process noise errors are a major source of concern, since its most real-world applications the sources are non-Gaussian. One alternative is to use a coloured-noise filter [23].

The main concern when it comes to the EKF is the accuracy of the linearized model, which directly influences the accuracy of the filter. For example, the quaternion kinematic model loses normalization

in a straightforward linearization.

3.2.4 Extended Kalman Filter applied to Spacecraft Attitude Estimation

In Chapter 2, multiple attitude representations were presented. Although all are useful in some ways throughout this work, some are more suitable for filtering applications than others. The Rotation Matrix is composed by 9 parameters and subject to 6 constraints, meaning that it can be described by three degrees of freedom locally, as seen in 2. With this in mind, a lower parameterization is more useful for filtering.

Since all three-component representations of the rotation group have discontinuities or singularities, a filter must provide a guarantee to avoid them. Although early Kalman Filters used Euler angles, which are intuitive for small angular displacements, they are not usually used due to requiring a fair number of trigonometric function evaluations, which increases computing power demands and the presence of singularities. The rotation vector $\boldsymbol{\vartheta}$ has singularities for 360° rotations. This is not useful for a global attitude representation due to the freedom of heading of the satellite.

The quaternion becomes the representation of choice since it is the lowest-dimensional parameterization that is free from singularities. On the other hand, the quaternion creates the challenge of maintaining the normalization constraint when, for example, the update equation is considered. To tackle this problem, distinct solutions are proposed.

Additive quaternion representation

The Additive Kalman Filter handles each of the quaternion components as independent parameters and spoils the normalization. The quaternion estimate is given by

$$\hat{\mathbf{q}} = E\{\mathbf{q}\} \tag{3.40}$$

and the error of estimation is given by

$$\Delta\mathbf{q} = \mathbf{q} - \hat{\mathbf{q}} \tag{3.41}$$

Assuming that the true quaternion $\mathbf{q}$ has unit norm, this leads to

$$E\{||\hat{\mathbf{q}}||\} = E\{||\mathbf{q}||\} + E\{||\Delta\mathbf{q}||\} = 1 + E\{||\Delta\mathbf{q}||\} \tag{3.42}$$

must violate the unit norm constraint for $\hat{\mathbf{q}}$.

Several methods have been proposed to deal with this problem. One method uses brute force to normalize the estimate but is only a second-order correction to the quaternion error. Other method gives up enforcing the quaternion norm and defines an attitude matrix whilst introducing one unobservable degree of freedom, but has been used successfully [25].

Multiplicative quaternion representation

The basic idea of the multiplicative EKF (MEKF) is to use the quaternion as the ”global” attitude

representation and use a three-component state vector $\delta\boldsymbol{\vartheta}$ for the "local" representation of attitude errors. The true quaternion can be obtained based on the error quaternion and the estimated quaternion such as

$$\mathbf{q} = \delta\mathbf{q}(\delta\boldsymbol{\vartheta}) \otimes \hat{\mathbf{q}} \tag{3.43}$$

The representations $\mathbf{q}$, $\delta\mathbf{q}$ and $\hat{\mathbf{q}}$ are properly normalized quaternions. The attitude error $\delta\mathbf{q}$ is defined in the body reference frame. The three-component attitude error vector $\boldsymbol{\delta\vartheta}$ follows the definition in 2.12. The MEKF error state is then defined as

$$\Delta\mathbf{x} = \begin{bmatrix} \delta\boldsymbol{\vartheta} \\ \Delta\boldsymbol{\xi} \end{bmatrix} \tag{3.44}$$

where $\boldsymbol{\xi}$ is a vector of other variables to be estimated and follows $\boldsymbol{\xi} = \hat{\boldsymbol{\xi}} + \Delta\boldsymbol{\xi}$. Thus, the conventions of the EKF can be applied to compute an estimate of $\Delta\mathbf{x}$. With this structure, the quaternion estimate $\hat{\mathbf{q}}$ is not part of the EKF and a reset operation transports the uncertainty to the global variable, maintaining a small error in $\delta\boldsymbol{\vartheta}$, thus avoiding singularities.

In comparison with the Additive Kalman Filter, the main advantage is faster computation due to having a smaller local variable, resulting in smaller covariance matrices. On the other hand, the dimensional reduction more accurately depicts the actual degrees of freedom of the system [26] [27], as well as has a transparent physical interpretation for the error covariance.

Attitude Kalman filter formulation

The MEKF follows the same structure of 2, with the most significant steps being initialization, measurement update, state vector reset and propagation to the next iteration, where the latter three are iterated.

The propagation step works in the global attitude representation, the quaternion, to propagate the state estimate and the covariance of the state. By differentiating the expression in (3.43), we obtain

$$\dot{\mathbf{q}} = \delta\dot{\mathbf{q}} \otimes \hat{\mathbf{q}} + (\delta\boldsymbol{\vartheta}) \otimes \dot{\hat{\mathbf{q}}} \tag{3.45}$$

The true and the estimate quaternion have the following kinematic equations

$$\dot{\mathbf{q}} = \frac{1}{2} \begin{bmatrix} \boldsymbol{\omega} \\ 0 \end{bmatrix} \otimes \mathbf{q} \tag{3.46}$$

$$\dot{\hat{\mathbf{q}}} = \frac{1}{2} \begin{bmatrix} \hat{\boldsymbol{\omega}} \\ 0 \end{bmatrix} \otimes \hat{\mathbf{q}} \tag{3.47}$$

where $\boldsymbol{\omega}$ and $\hat{\boldsymbol{\omega}}$ are the true and estimated angular velocities, respectively, and $\boldsymbol{\omega} = \hat{\boldsymbol{\omega}} + \delta\boldsymbol{\omega}$ Substituting (3.43), (3.46) and (3.47) into (3.45) gives

$$\delta\dot{\mathbf{q}} = \frac{1}{2} \left(\begin{bmatrix} \boldsymbol{\omega} \\ 0 \end{bmatrix} \otimes \delta\mathbf{q} - \delta\mathbf{q} \otimes \begin{bmatrix} \hat{\boldsymbol{\omega}} \\ 0 \end{bmatrix} \right) \tag{3.48}$$

Assuming that $\delta\boldsymbol{\omega}$ and $\delta\mathbf{q} - I_q$ are small, it leads to

$$\delta\dot{\mathbf{q}} = -\left(\begin{bmatrix}\hat{\boldsymbol{\omega}} \times \delta\mathbf{q}_{1:3} \\ 0\end{bmatrix} + \frac{1}{2}\begin{bmatrix}\delta\boldsymbol{\omega} \\ 0\end{bmatrix}\right) \tag{3.49}$$

An alternative representation for $\delta\mathbf{q}(\boldsymbol{\delta\vartheta})$ is

$$\delta\mathbf{q} \simeq \begin{bmatrix}\delta\boldsymbol{\vartheta}/2 \\ 1\end{bmatrix} = I_q + \frac{1}{2}\begin{bmatrix}\delta\boldsymbol{\vartheta} \\ 0\end{bmatrix} \tag{3.50}$$

By using (3.50) with (3.49) we obtain the value for the first three components, such as

$$\delta\dot{\boldsymbol{\vartheta}} = -\hat{\boldsymbol{\omega}} \times \delta\boldsymbol{\vartheta} + \delta\boldsymbol{\omega} \tag{3.51}$$

whilst the fourth component is $\delta\dot{q}_4 = 0$. The previous equation will be used to propagate the error-angle covariance. Its expected value expression is

$$\delta\dot{\hat{\boldsymbol{\vartheta}}} = -\hat{\boldsymbol{\omega}} \times \delta\hat{\boldsymbol{\vartheta}} \tag{3.52}$$

The measurement update step works within the error-state and partially in discrete-time.

The observation model follows the definition in (3.33), such that

$$\mathbf{y} = h(\mathbf{q}, \boldsymbol{\xi}) + \boldsymbol{\vartheta}, \boldsymbol{\vartheta} \sim N(\mathbf{0}, R) \tag{3.53}$$

that gives the following sensitivity matrix

$$H(\mathbf{q}, \boldsymbol{\xi}) = \frac{\partial h}{\partial(\Delta\mathbf{x})} = \begin{bmatrix}\frac{\partial h}{\partial(\delta\boldsymbol{\vartheta})} & \frac{\partial h}{\partial\boldsymbol{\xi}}\end{bmatrix} = \begin{bmatrix}H_{\boldsymbol{\vartheta}} & H_{\boldsymbol{\xi}}\end{bmatrix} \tag{3.54}$$

By substituting (3.50) and the quaternion operations in (3.45) it gives

$$\mathbf{q} = \left(I_q + \frac{1}{2}\begin{bmatrix}\boldsymbol{\delta\vartheta} \\ 0\end{bmatrix}\right) \otimes \hat{\mathbf{q}} = \hat{\mathbf{q}} + \frac{1}{2}\Xi(\hat{\mathbf{q}})\boldsymbol{\delta\vartheta} \tag{3.55}$$

From the previous equation, it follows that the attitude part of the measurement sensitivity matrix can be evaluated by using the chain rule

$$H_v = \frac{\partial h}{\partial\mathbf{q}}\frac{\partial\mathbf{q}}{\partial(\boldsymbol{\delta\vartheta})} = \frac{1}{2}\frac{\partial h}{\partial\mathbf{q}}\Xi(\hat{\mathbf{q}}) \tag{3.56}$$

In order to avoid recalculating the nonlinear function $h(\hat{\mathbf{q}}, \hat{\boldsymbol{\xi}})$ if batch measurements are processed, the first-order Taylor series is used to compute the expectation

$$E\{\ h(\mathbf{q}, \boldsymbol{\xi})\} \simeq h(\hat{\mathbf{q}}, \hat{\boldsymbol{\xi}}) + H(\hat{\mathbf{q}}, \hat{\boldsymbol{\xi}})\begin{bmatrix}\boldsymbol{\delta\vartheta} \\ \Delta\hat{\boldsymbol{\xi}}\end{bmatrix} \tag{3.57}$$

The state update for the k-th measurement is then

$$\begin{bmatrix} \boldsymbol{\delta\vartheta}_k^+ \\ \Delta\hat{\boldsymbol{\xi}}_k^+ \end{bmatrix} = \begin{bmatrix} \boldsymbol{\delta\vartheta}_k^- \\ \Delta\hat{\boldsymbol{\xi}}_k^- \end{bmatrix} + K_k \left\{ \mathbf{y}_k - h(\hat{\mathbf{q}}_k^-, \hat{\boldsymbol{\xi}}_k^-) - H(\hat{\mathbf{q}}_k^-, \hat{\boldsymbol{\xi}}_k^-) \begin{bmatrix} \boldsymbol{\delta\vartheta}_k^- \\ \Delta\hat{\boldsymbol{\xi}}_k^- \end{bmatrix} \right\} \tag{3.58}$$

where K_k is computed as defined in the previous section.

The reset step is responsible for converting the local $\Delta\mathbf{x}^+$ representation into the global representation $\mathbf{x}^-$ and resets the error state to zero, such as

$$\hat{\mathbf{q}}^+ = \mathbf{q}(\hat{\boldsymbol{\delta\vartheta}}^+) \otimes \hat{\mathbf{q}}^- = \frac{1}{\sqrt{1 + ||\hat{\boldsymbol{\delta\vartheta}}^+/2||^2}} \begin{bmatrix} \hat{\boldsymbol{\delta\vartheta}}^+/2 \\ 1 \end{bmatrix} \otimes \hat{\mathbf{q}}^- \tag{3.59}$$

$$\hat{\boldsymbol{\xi}}_k^+ = \hat{\boldsymbol{\xi}}_k^- + K_{\xi k}[\mathbf{y}_k - h_k(\hat{\mathbf{y}}_k^-, \hat{\boldsymbol{\xi}}_k^-)] \tag{3.60}$$

Mission mode Kalman filter

The mission mode EKF estimates the attitude and the gyroscope bias with a 6-state EKF and was first introduced in [28]. Precise angular rate information is vital to ensure precise attitude knowledge, filter noisy attitude data as well as to perform precise manoeuvres. In a pursuit of better accuracy, the gyroscope for dynamic model replacement method is employed. Instead of using the dynamic equations defined in 2 for the propagation step of the filter, information from modern accurate gyroscope is part of the dynamic model. The main advantages of this method are increased accuracy and decreased computational load. The rotational dynamic models are often inaccurate and torque measurements often unavailable, and would corrupt the information from the gyroscope. The computation of the dynamics equations is also much more cumbersome than the gyroscope alternative.

The filter assumes that misalignment, as well as scale factors, have already been determined. An update is performed at each time step.

The global state of the filter is given as

$$\mathbf{x} = \begin{bmatrix} \mathbf{q} \\ \boldsymbol{\beta} \end{bmatrix} \tag{3.61}$$

whilst the local error state is given as

$$\Delta\mathbf{x} = \begin{bmatrix} \delta\mathbf{q} \\ \Delta\boldsymbol{\beta} \end{bmatrix} \tag{3.62}$$

where $\Delta\boldsymbol{\beta} = \boldsymbol{\beta} - \hat{\boldsymbol{\beta}}$. From the gyroscope model presented in Section 3.1.9, both $\dot{\boldsymbol{\beta}}$ and $\dot{\hat{\boldsymbol{\beta}}}$ are 0.

The propagation step propagates the global variables, the quaternion and bias estimates. The discrete-time quaternion propagation follows

$$\hat{\mathbf{q}}_{k+1}^- = \exp[(\Delta\boldsymbol{\theta}/2)\otimes]\hat{\mathbf{q}}_k^+ \approx \bar{\Theta}\left(\hat{\boldsymbol{\omega}}_k^+\right)\hat{\mathbf{q}}_k^+ \tag{3.63}$$

with

$$\bar{\Theta}(\hat{\omega}_k^+) \equiv \begin{bmatrix} \cos\left(\frac{1}{2}\left\|\hat{\omega}_k^+\right\|\Delta t\right) I_3 - \left[\hat{\psi}_k^+\times\right] & \hat{\psi}_k^+ \\ -\hat{\psi}_k^{+T} & \cos\left(\frac{1}{2}\left\|\hat{\omega}_k^+\right\|\Delta t\right) \end{bmatrix} \tag{3.64}$$

where

$$\hat{\boldsymbol{\psi}}_k^+ \equiv \frac{\sin\left(\frac{1}{2}\left\|\hat{\boldsymbol{\omega}}_k^+\right\|\Delta t\right)\hat{\boldsymbol{\omega}}_k^+}{\left\|\hat{\boldsymbol{\omega}}_k^+\right\|} \tag{3.65}$$

In order to define the covariance propagation matrix, the error-state vector dynamics must be defined, such that

$$\Delta\dot{\mathbf{x}}(t) = F(t)\Delta\mathbf{x}(t) + G(t)\mathbf{w}(t) \tag{3.66}$$

where

$$\mathbf{w}(t) = \begin{bmatrix} \sigma_v(t) \\ \sigma_u(t) \end{bmatrix} \tag{3.67}$$

and the matrices $F(t)$ and $G(t)$ follow

$$F(t) = \begin{bmatrix} -[\hat{\boldsymbol{\omega}}(t)\times] & -I_3 \\ 0_{3\times3} & 0_{3\times3} \end{bmatrix} \tag{3.68}$$

$$G(t) = \begin{bmatrix} -I_3 & 0_{3\times3} \\ 0_{3\times3} & I_3 \end{bmatrix} \tag{3.69}$$

$$Q = \begin{bmatrix} \sigma_v^2 I_3 & 0_{3\times3} \\ 0_{3\times3} & \sigma_u^2 I_3 \end{bmatrix} \tag{3.70}$$

The discrete-time error state transition matrices for this case is given by

$$\Phi = \begin{bmatrix} \Phi_{11} & \Phi_{12} \\ \Phi_{21} & \Phi_{22} \end{bmatrix} \tag{3.71}$$

with

$$\Phi_{11} = I_3 - [\hat{\boldsymbol{\omega}}\times]\frac{sin(||\hat{\boldsymbol{\omega}}||\Delta t)}{||\hat{\boldsymbol{\omega}}||} + [\hat{\boldsymbol{\omega}}\times]^2\frac{\{1 - cos(||\hat{\boldsymbol{\omega}}||\Delta t)\}}{||\hat{\boldsymbol{\omega}}||^2} \tag{3.72a}$$

$$\Phi_{12} = [\hat{\boldsymbol{\omega}}\times]\frac{\{1 - cos(||\hat{\boldsymbol{\omega}}||\Delta t)\}}{||\hat{\boldsymbol{\omega}}||^2} - I_3\Delta t - [\hat{\boldsymbol{\omega}}\times]^2\frac{\{||\hat{\boldsymbol{\omega}}||\Delta t - sin(||\hat{\boldsymbol{\omega}}||\Delta t)\}}{||\hat{\boldsymbol{\omega}}||^3} \tag{3.72b}$$

$$\Phi_{21} = 0_{3\times3} \tag{3.72c}$$

$$\Phi_{22} = I_3 \tag{3.72d}$$

The formula for the covariance matrix Q in discrete-time is formulated in [24]. It assumes that $\hat{\boldsymbol{\omega}}$ is constant throughout the sampling time and that the sampling rate is below Nyquist's limit (for example, with a safety factor of 10, it is required that $||\hat{\boldsymbol{\omega}}||\Delta t < \pi/10$).

$$Q_k \approx \begin{bmatrix} (\sigma_v^2\Delta t + \frac{1}{3}\sigma_u^2\Delta t^3)I_3 & -(\frac{1}{2}\sigma_u^2\Delta t^2)I_3 \\ -(\frac{1}{2}\sigma_u^2\Delta t^2)I_3 & (\sigma_u^2\Delta t)I_3 \end{bmatrix} \tag{3.73}$$

The discrete-time covariance propagation follows the definition from Section 3.2.2 with

$$\Upsilon = \begin{bmatrix} -I_3 & 0_{3\times 3} \\ 0_{3\times 3} & I_3 \end{bmatrix} \tag{3.74}$$

The bias propagation is given by

$$\hat{\boldsymbol{\beta}}_{k+1}^{-} = \hat{\boldsymbol{\beta}}_{k}^{+} \tag{3.75}$$

The measurement step receives observations that can be derived from (3.54), for N vector observation at time t_k, given as

$$\mathbf{y}_k = \left[\begin{array}{c} A\left(\mathbf{q}\right)\mathbf{r}_1 \\ A\left(\mathbf{q}\right)\mathbf{r}_2 \\ \vdots \\ A\left(\mathbf{q}\right)\mathbf{r}_N \end{array} \right]\Bigg|_{t_k} + \left[\begin{array}{c} \boldsymbol{v}_1 \\ \boldsymbol{v}_2 \\ \vdots \\ \boldsymbol{v}_N \end{array} \right]\Bigg|_{t_k} \equiv \mathbf{h}_k\left(\mathbf{x}_k\right) + \mathbf{v}_k \tag{3.76}$$

$$R = diag([R_1 R_2 \dots R_N]) \tag{3.77}$$

where R_i is the covariance of $\boldsymbol{\vartheta}_i$. The covariance can either be simplified to $R_i = \sigma_i^2 I_3$ under the assumption that the measurement errors are isotropic or follow the QUEST measurement model [11], where the measurements are not isotropic but lead to computational disadvantages.

The process that leads to the observation sensitivity matrix $H_k(\hat{\mathbf{x}}_k^-)$ is presented. The true attitude matrix $A(\mathbf{q})$ is related to the *a priori* attitude matrix $A(\hat{\mathbf{q}}^-)$ through

$$A(\mathbf{q}) = A(\delta\mathbf{q})A(\hat{\mathbf{q}}^-) \tag{3.78}$$

The first-order approximation of the error-attitude matrix $A(\delta\mathbf{q})$ is given by

$$A(\delta\mathbf{q}) \simeq I_3 - [\delta\boldsymbol{\vartheta}\times] \tag{3.79}$$

For each sensor, the true $\mathbf{r}$, and estimated body vectors $\mathbf{b}$, are given by

$$\mathbf{b} = A(\mathbf{q})\mathbf{r} \tag{3.80}$$

$$\hat{\mathbf{b}} = A(\hat{\mathbf{q}}^-)\mathbf{r} \tag{3.81}$$

$$\Delta\mathbf{b} \equiv \mathbf{b} - \hat{\mathbf{b}}^- = -[\delta\boldsymbol{\vartheta}\times]A\left(\hat{\mathbf{q}}^-\right)\mathbf{r} = \left[\hat{\mathbf{b}}^-\times\right]\delta\boldsymbol{\vartheta} \tag{3.82}$$

Therefore the sensitivity matrix for all measurements is given by

$$H_k\left(\hat{\mathbf{x}}_k^-\right) = \left[\begin{array}{cc} \left[\hat{\mathbf{b}}_1^- \times\right] & 0_{3\times 3} \\ \left[\hat{\mathbf{b}}_2^- \times\right] & 0_{3\times 3} \\ \vdots & \vdots \\ \left[\hat{\mathbf{b}}_N^- \times\right] & 0_{3\times 3} \end{array}\right]\Bigg|_{t_k} \tag{3.83}$$

The filter is reset before every measurement update implicitly.

The mission mode attitude estimation algorithm is summarized in Tab. 3.4. The filter is first initialized with a fixed state and an error covariance matrix. The first three diagonal elements of the error covariance matrix correspond to attitude errors and the second three correspond to the bias. The estimated corrected angular velocity from the gyroscope is used to propagate the quaternion kinematic model and error covariance in the EKF. Then, the Kalman gain is computed using the measurement-error covariance R_k and sensitivity matrix in (3.83) The state error covariance and state error follow the standard EKF update. The quaternion is reset follows (3.59) and the bias follows (3.60).

Initial condition	$\hat{\mathbf{q}}(t_0) = \hat{\mathbf{q}}_0$
	$\hat{\boldsymbol{\beta}}(t_0) = \hat{\boldsymbol{\beta}}_0$
	$P(t_0) = P_0$
Gain	$K_k = P_k^- H_k^T\left(\hat{\mathbf{x}}_k^-\right)\left[H_k\left(\hat{\mathbf{x}}_k^-\right) P_k^- H_k^T\left(\hat{\mathbf{x}}_k^-\right) + H_k\right]^{-1}$
	$H_k\left(\hat{\mathbf{x}}_k^-\right) = \left[\begin{array}{cc} \left[\hat{\mathbf{b}}_1^- \times\right] & 0_{3\times 3} \\ \left[\hat{\mathbf{b}}_2^- \times\right] & 0_{3\times 3} \\ \vdots & \vdots \\ \left[\hat{\mathbf{b}}_N^- \times\right] & 0_{3\times 3} \end{array}\right]\Bigg\|_{t_k}$
Update	$P_k^+ = \left[I - K_k H_k\left(\hat{\mathbf{x}}_k^-\right)\right] P_k^-$
	$\Delta\hat{\mathbf{x}}_k^+ = K_k\left[\mathbf{y}_k - \mathbf{h}_k\left(\hat{\mathbf{x}}_k^-\right)\right]$
	$\Delta\hat{\mathbf{x}}_k^+ \equiv \left[\begin{array}{cc} \hat{\boldsymbol{\delta\vartheta}}_k^{+T} & \Delta\hat{\boldsymbol{\beta}}_k^{+T} \end{array}\right]^T$
	$\mathbf{h}_k\left(\hat{\mathbf{x}}_k^-\right) = \left[\begin{array}{c} A\left(\hat{\mathbf{q}}^-\right)\mathbf{r}_1 \\ A\left(\hat{\mathbf{q}}^-\right)\mathbf{r}_2 \\ \vdots \\ A\left(\hat{\mathbf{q}}^-\right)\mathbf{r}_N \end{array}\right]\Bigg\|_{t_k}$
	$\hat{\mathbf{q}}^* = \hat{\mathbf{q}}_k^- + \frac{1}{2}\Xi\left(\hat{\mathbf{q}}_k^-\right)\boldsymbol{\delta}\hat{\vartheta}_k^+$
	$\hat{\mathbf{q}}_k^+ = \mathbf{q}^* / \left\|\mathbf{q}^*\right\|$
	$\hat{\boldsymbol{\beta}}_k^+ = \hat{\boldsymbol{\beta}}_k^- + \Delta\hat{\boldsymbol{\beta}}_k^+$
Propagation	$\hat{\boldsymbol{\omega}}(t) = \boldsymbol{\omega}_g(t) - \hat{\boldsymbol{\beta}}(t)$
	$\dot{\hat{\mathbf{q}}}(t) = \frac{1}{2}\Xi(\hat{\mathbf{q}}(t))\hat{\boldsymbol{\omega}}(t)$
	$\dot{P}(t) = F(t)P(t) + P(t)F^T(t) + G(t)Q(t)G^T(t)$

Table 3.4: MEKF for mission mode - adapted from [11].

3.2.5 Complementary filter

The Complementary filter (CF) aims to estimate the attitude and gyro bias in a simple way - it was developed with the goal of run in low-power Microcontroller units (MCU). This filter was first developed in [29] and was later reformulated in [30] to address computational overhead due to online attitude reconstruction. This filter estimates the rotation from $\mathcal{B}$ to $\mathcal{I}$, as opposed to the previous designation, such that

$$\mathcal{R} = A^T = [R_{\mathcal{B}}^{\mathcal{I}}] \tag{3.84}$$

The same kinematics for the attitude matrix still apply, such that

$$\dot{\mathcal{R}} = \mathcal{R}[\boldsymbol{\omega}_{\mathcal{B}}]_\times = (\mathcal{R}\boldsymbol{\omega}_{\mathcal{B}})_\times \mathcal{R} \tag{3.85}$$

The filter creates an auxiliary frame $\hat{\mathcal{B}}$ to denote the estimation of the rotation between the body and the inertial frames.

The attitude estimation error matrix can be written as

$$\tilde{\mathcal{R}} = \hat{\mathcal{R}}^T \mathcal{R} \tag{3.86}$$

The goal of the attitude estimate is to drive $\hat{\mathcal{R}} \to \mathcal{R}$, so $\tilde{\mathcal{R}} \to I_3$. The estimator takes the dynamics in (3.87).

$$\dot{\hat{\mathcal{R}}} = \hat{\mathcal{R}}(\hat{\boldsymbol{\omega}}_{\mathcal{B}} - \boldsymbol{\beta}_g + k_p \boldsymbol{\gamma})_\times \tag{3.87}$$

where k_p is a positive scalar gain. The $\boldsymbol{\gamma}$ term can be computed in a number of ways: either reconstructing an attitude matrix $\mathcal{R}_y$ from the sensors to then compute $\tilde{\mathcal{R}}$ (direct approach) or the use of the estimated rotation $\hat{\mathcal{R}}$ to feedback and compute $\tilde{\mathcal{R}}$. This is the preferential approach since it doesn't propagate errors through the reconstruction of $\mathcal{R}_y$. The expression for $\boldsymbol{\gamma}$ is

$$\boldsymbol{\gamma} = \sum_{i=1}^{N} k_i [\mathbf{b}_i \times] A(\hat{\mathbf{q}}) \mathbf{r}_i \tag{3.88}$$

where k_i is a positive scalar gain for sensor i. It is possible to include a bias estimation for the filter using the correction factor $\boldsymbol{\gamma}$, such as

$$\dot{\hat{\boldsymbol{\beta}}}_g = -k_g \boldsymbol{\gamma} \tag{3.89}$$

Since quaternions are the preferential attitude representation, (3.87) can be rewritten as

$$\dot{\hat{\mathbf{q}}} = \Omega(\hat{\boldsymbol{\omega}}_{\mathcal{B}} - \boldsymbol{\beta} + k_p \boldsymbol{\gamma}) \otimes \hat{\mathbf{q}} \tag{3.90}$$

where $\mathbf{q}$ represents the rotation from $\mathcal{I}$ to $\mathcal{B}$.

3.3 Attitude determination methods

Even though these methods are not the main focus of this thesis and usually offer lower accuracy [13], an implementation is proposed to serve as a comparison for the method above. The TRIaxial Attitude Determination (TRIAD) method was chosen since it uses two measurements, the maximum simultaneous available measurements due to mission requirements.

The attitude matrix to be determined is the matrix that rotates vectors from the reference frame I to the satellite body frame B, such that

$$A\mathbf{r}_i = \mathbf{b}_i \quad \text{for} \quad i = 1, 2 \tag{3.91}$$

This is only true for error-free measurement, since (3.91) implies that

$$\mathbf{b}_1 \cdot \mathbf{b}_2 = (A\mathbf{r}_1) \cdot (A\mathbf{r}_2) = \mathbf{r}_1^T A^T A \mathbf{r}_2 = \mathbf{r}_1 \cdot \mathbf{r}_2 \tag{3.92}$$

The TRIAD algorithm assumes that one vector (e.g. $\mathbf{b}_1$) is more accurately determined than the other, such that $A\mathbf{r}_1 = \mathbf{b}_1$ exactly, but $A\mathbf{r}_2 = \mathbf{b}_2$ only approximately.

It is assumed that the orthonormal right-handed triad of vectors $\{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3\}$ in the reference frame has a corresponding orthonormal right-handed triad of vectors $\{\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3\}$ in the satellite body frame and form the attitude matrix

$$A = [\mathbf{w}_1 \mathbf{w}_2 \mathbf{w}_3][\mathbf{v}_1 \mathbf{v}_2 \mathbf{v}_3]^T = \sum_{i=1}^{3} \mathbf{w}_i \mathbf{v}_i^T \tag{3.93}$$

The algorithm forms each triad from the vectors in both frames, such that

$$\mathbf{v}_1 = \mathbf{r}_1, \quad \mathbf{v}_2 = \mathbf{r}_\times \equiv \frac{\mathbf{r}_1 \times \mathbf{r}_2}{||\mathbf{r}_1 \times \mathbf{r}_2||}, \quad \mathbf{v}_3 = \mathbf{r}_1 \times \mathbf{r}_\times \tag{3.94a}$$

$$\mathbf{w}_1 = \mathbf{b}_1, \quad \mathbf{w}_2 = \mathbf{b}_\times \equiv \frac{\mathbf{b}_1 \times \mathbf{b}_2}{||\mathbf{b}_1 \times \mathbf{b}_2||}, \quad \mathbf{w}_3 = \mathbf{b}_1 \times \mathbf{b}_\times \tag{3.94b}$$

The estimate of the attitude matrix is calculated as such

$$\hat{A}_{TRIAD} = \mathbf{b}_1 \mathbf{r}_1^T + (\mathbf{b}_1 \times \mathbf{b}_\times)(\mathbf{r}_1 \times \mathbf{r}_\times)^T + \mathbf{b}_\times \mathbf{r}_\times^T \tag{3.95}$$

This attitude matrix is invalid if the reference or the observed vectors are parallel or antiparallel.

Chapter 4

Control

Attitude control is essential to meet mission pointing requirements, as well as other satellite functions, such as detumbling and maintaining radio communications.

The actuators can be selected separately from the control algorithms, based on the mission's pointing accuracy requirements, whilst taking into account the SWaP and the required manoeuvres. The state-of-the-art actuators are presented and selected for mission use in Section 4.1. Although the maintenance of radio communications is a separate manoeuvre from off/nadir pointing, the control algorithm can be designed with the goal of performing both of the tasks, presented in Section 4.2. Lastly, the controller for the detumbling manoeuvre is presented in Section 4.3.

4.1 Actuators

Actuators are responsible for providing all necessary control actions. Since orbit correction equipment is not necessary for the considered mission, orbital actuators are not analyzed. Therefore, only attitude actuators are considered. Also, no passive solutions are considered due to the mission's high pointing precision and reduced SWaP capacity for redundancy. This section is based on [11] [31]. Fluid dynamic actuators are not considered due to the increased technological complexity.

4.1.1 Reaction wheels

Reaction wheels accelerate flywheels with motors in order to achieve a reaction of the satellite in the opposite direction, following Newton's third law. A reaction wheel is usually defined by two attributes, torque (Nm) and stored Momentum (Nm/s). The first variable represents the amount of force that the wheel can apply on its axis and the latter represents the maximum energy that can be stored in the inertia of the wheel.

Reaction wheels can be either used to generate torque through the variation of the motor's angular velocity or ran at a constant high speed, generating a gyro stiffness across the wheel's axis - called the momentum wheel. These applications are usually employed as redundancy in order to stabilize an axis or to store energy [32].

4.1.2 Control momentum gyro

If a momentum wheel is mounted on a pivot, then when the pivot is driven to change the axis of the spinning wheel, a very high torque is generated on the satellite to resist the gyro stiffness of the wheel. This is the basic operating method of a Control Momentum gyro. Although these actuators are widely used in satellites [33], they do not scale to small sizes well and are less applicable to CubeSat applications, especially when there are tight volumetric requirements. Also, the control algorithms are complicated as there are certain positions where no torque is generated.

4.1.3 Magnetorquers

Magnetorquers are electromagnets that generate a large magnetic field ($A\text{m}^2$) when energized. This interacts with the local magnetic field and generates a torque across the satellite when their field is offset from the local Earth's magnetic vector. A wide variety of materials can be employed to generate the field, from coils of copper wire to rods. When not being used, they generate a small residual magnetic field that causes a disturbance in the satellite. Some rods fabricated from soft magnetic materials are often used as passive dampers.

4.1.4 Actuators comparision and selection

To select the necessary actuators, some analysis of the necessary manoeuvres' requirements must be conducted. For the pointing maneuver, considering both coarse and medium modes, the strictest requirement for accuracy is 0.1°. In table 4.1 literature is compiled.

Due to the stark requirements of the mission, the suggested pointing actuators are reaction wheels or control momentum gyro. As previously discussed, the latter is not adequate for CubeSats. The required slewing rate of the aforementioned desired frame in relation to the inertial frame is $2^\circ/s$, which in [12] also suggests the use of the control momentum gyros. However, as previously stated, this actuator is not applicable.

Required accuracy $^\circ$	**Effect on ADCS**
$>$ 5	Magnetorquers
1-5	Reaction wheels + magnetorquers (for momentum dumping)
0.1 - 1	Reaction wheels + magnetorquers (for momentum dumping)
$<$ 0.1	Reaction wheels + magnetorquers (for momentum dumping) but better class

Table 4.1: Comparison of attitude actuators for 3-axis stabilization, adapted from [12]

For the detumbling maneuver, the ADCS is responsible for reducing the spinning rate of the satellite until it gets lower than $5^\circ/s$. Such cannot be accomplished with the use of reaction wheels and are usually accomplished by magnetorquers. The momentum dumping maneuver, which involves reducing the reaction wheel's speed, thus creating torque, also needs this additional sensor in order to oppose

these forces.

4.1.5 Actuators model

The selected actuators must be modelled in order to be incorporated in the simulation environments. The model for the reaction wheel and magnetorquers is now presented.

Reaction wheels

The torque generated through a reaction wheel can be modelled through a second-order system of a DC-motor coupled to a flywheel [34]. This approach requires knowledge of the reaction wheel's motor parameters, such as armature inductance and resistance, as well as the flywheels parameters. None of these specifications are available in COTS reaction wheels, so a model was assumed based on a first-order system for the motor, maximum torque values and physical parameter, as discussed in 5.4. The highest angular velocity possible, positive or negative, that corresponds to maximum stored momentum, serves as saturation for the output torque. Jitter and control speed disturbances are also added to the wheel's angular velocity.

Magnetorquer

The torque generated by a magnetorquer is given by [35]

$$L_{mtq} = \mathbf{m} \times \mathbf{B} \tag{4.1}$$

where the $\mathbf{m}$ is the commanded magnetic dipole moment vector generated by the coils and $\mathbf{B}$ is the local geomagnetic field vector expressed in terms of the body-frame components.

4.2 Pointing

The mission requires the satellite to point to a specific location on the Earth, the GS. Whilst the satellite is not engaged in quantum communications, it still has to ensure radio communications. This will be handled by choosing between reference attitudes in different locations of the orbit for the pointing controller. Three different control architectures are proposed to accomplish this task.

Early missions designed spin-stabilized satellites due to the limited control actuation and lack of computing power to implement sophisticated control laws. Although these satellites are very stable, spin-stabilization is very difficult to achieve, since it imposes rigid constraints on component placement and design, which is an arduous task, although some spinners are still used to this day [36].

In the modern days, sensors, actuators and computer processors have advanced to allow for three-axis stabilization. These methods are supported as being effective for the mission's requirements, off-nadir pointing, as discussed in [12]. The control of satellite slews can be accomplished by either closed-loop or open-loop schemes. The latter usually requires pre-determined pointing manoeuvres that are typically determined using optimal control techniques [37]. These schemes are sensitive to satellite

parameter uncertainties and unexpected disturbances. Closed-loop schemes are more robust since they can compensate for these uncertainties and disturbances. Although some control schemes have been derived for various attitude representations [38], the quaternion is the chosen representation since it can provide effortless orientation feedback from the estimation, as discussed in Section 3.2. The angular rate is also available through the gyroscope, completing full-state feedback.

The desired attitude motion is generated by a frame $\mathcal{D}$ with its attitude and angular velocity denoted by q_d and w_d, creating a time-varying trajectory. It is defined in such a way that the payload is facing the GS when it is in LOS of the GS. The error quaternion and angular velocity error are defined as such

$$\boldsymbol{\delta q} = \begin{bmatrix} \boldsymbol{\delta q}_{1:3} \\ \delta q_4 \end{bmatrix} = \mathbf{q} \otimes \mathbf{q}_d^{-1} \tag{4.2}$$

$$\boldsymbol{\delta\omega} = \boldsymbol{\omega} - A(\boldsymbol{\delta q})\boldsymbol{\omega}_d = \boldsymbol{\omega} - \bar{\boldsymbol{\omega}}_d \tag{4.3}$$

where the operation $\otimes$ follows the definition in Appendix A and $A(\delta\mathbf{q})$ is the transformation matrix from $\mathcal{D}$ to $\mathcal{B}$. The goal of an attitude controller is to drive $\boldsymbol{\omega}$ to $\boldsymbol{\omega}_d$ and $\delta\mathbf{q}$ to the identity quaternion $\mathbf{I}_q = [0001]^T$.

The attitude kinematics and dynamics of a rigid body follow the definition in Section 2.3 (removing the superscripts for brevity):

$$\dot{\mathbf{q}} = \frac{1}{2}\Omega(\boldsymbol{\omega})\mathbf{q} \tag{4.4}$$

$$J\dot{\boldsymbol{\omega}} = -[\boldsymbol{\omega}\times]J\boldsymbol{\omega} + L \tag{4.5}$$

Since reaction wheels are used to control the spacecraft, it is useful to define $\bar{J}$ as the moment of inertia tensor of the satellite and reaction wheels, such that

$$\bar{J} = J + J_{wheels} \tag{4.6}$$

The equations of attitude motion relative to $\mathcal{D}$ can be written based on (4.4) and (4.5) as

$$\boldsymbol{\delta\dot{q}} = \frac{1}{2}\Omega(\boldsymbol{\delta\omega})\mathbf{q} \tag{4.7}$$

$$\bar{J}\boldsymbol{\delta\dot{\omega}} = (\bar{J}(\boldsymbol{\delta\omega} + \bar{\boldsymbol{\omega}}_d))^{\times} - \bar{\boldsymbol{\omega}}_d^{\times}\bar{J} - \bar{J}\bar{\boldsymbol{\omega}}_d^{\times} - \bar{\boldsymbol{\omega}}_d^{\times}\bar{J}\bar{\boldsymbol{\omega}}_d - \bar{J}A(\boldsymbol{\delta q})\dot{\boldsymbol{\omega}}_d + L \tag{4.8}$$

4.2.1 Global finite-time attitude tracking controller

This section follows the implementation in [39] for a global finite-time attitude tracking controller (GFTAC). The resulting control laws possess a simple, nonlinear, proportional-derivative (PD) structure that ensures bounded control torques.

From (4.8), the torque required to maintain the desired attitude motion with zero tracking error can be written as

$$L_d = \bar{\boldsymbol{\omega}}_d^X \bar{J}\bar{\boldsymbol{\omega}}_d + \bar{J}A(\boldsymbol{\delta q})\dot{\boldsymbol{\omega}}_d \tag{4.9}$$

In order to define the control law, a hysteretic switching law is employed [40] with the goal of guar-

anteeing global asymptotic stabilization. This is through the employment of a binary logic variable $h \in H = \{-1, 1\}$ as well as a constant hysteresis gap $\star \in (0, 1)$. The value of h is multiplied by the scalar component of the quaternion to define the equilibrium from ± 1. A hybrid system is described as

$$\mathcal{H} = \begin{cases} \dot{\mathbf{x}} = F(\mathbf{x}) & : \mathbf{x} \in C \\ \mathbf{x}^+ = G(\mathbf{x}) & : \mathbf{x} \in D \end{cases} \tag{4.10}$$

where the flow map F governs the continuous evolution of state $\mathbf{x}$ while on set C, while the jump map G governs the discrete dynamics over the jump D and $\mathbf{x}^+$ denotes the state value immediately after the jump.

In this case, the considered state is $\mathbf{x}_1 = (\boldsymbol{\delta q}, \boldsymbol{\delta \omega}, h)$. The flow and jump sets can be defined respectively as

$$C_1 = \{\mathbf{x}_1 : h\delta q_4 \geq -\star\} \tag{4.11}$$

$$D_1 = \{\mathbf{x}_1 : h\delta q_4 \leq -\star\} \tag{4.12}$$

where $C_1 \cup D_1 = S^3 \times R^n \times H$. A hybrid attitude controller is the designated as

$$L(\mathbf{x}_1) = L_d - k_1\kappa_1(h\boldsymbol{\delta q}, 1 - \alpha_1) - k_2 sat_{\alpha_2}\boldsymbol{\delta \omega}, \mathbf{x}_1 \in C_1 \tag{4.13}$$

$$\mathbf{x}_1^+ = G_1(\mathbf{x}_1) = (\boldsymbol{\delta q}, \boldsymbol{\delta \omega}, s\bar{g}n(\delta q_4)), \mathbf{x}_1 \in D_1 \tag{4.14}$$

where $k_1, k_2 > 0$, $0 < \alpha_1 < 1$ and $\alpha_2 = 2\alpha_1/(1 + \alpha_1)$. It is noteworthy that h reverses its sign only on the jump set D_1 in order to change the rotation signal. Otherwise, it remains constant and thus $\dot{h}$ = 0. The function κ_1 is defined as

$$\kappa_1 = \begin{cases} \frac{\mathbf{q}}{\sqrt{2(1-\delta q_4)}}, & \delta q_4 \neq 1 \\ 0, & : \delta q_4 = 1 \end{cases} \tag{4.15}$$

and $sat_\alpha(\mathbf{x}) = [sgn(x_1)min\{|x_1|^\alpha, 1\}, ..., sgn(x_n)min\{|x_n|^\alpha, 1\}]$ for **(**$x) \in R^N$, where $sgn()$ is the standard sign function and $s\bar{g}n$ is given as

$$s\bar{g}n(x) = \begin{cases} sgn(x) & : x \neq 0 \\ \{-1, 1\} & : x = 0 \end{cases} \tag{4.16}$$

4.2.2 Sliding Controller

A sliding mode controller, also referred to as variable structure control, varies its control law based on the position of the state trajectory. Once the state trajectories reach the sliding surface, referred to as the reaching phase, the discontinuous controller forces the states to slide towards the origin, referred to as the sliding phase. The basics of the sliding controller are presented in [41].

A sliding controller, as defined in [11], has the following sliding surface

$$\mathbf{s} = (\boldsymbol{\omega} - \boldsymbol{\omega}_d) + k\boldsymbol{\delta q}_{1:3} \tag{4.17}$$

where $\boldsymbol{\omega}_d$ and $\boldsymbol{\delta q}_{1:3}$ follow the definitions in Section 4.2.1. If $\mathbf{s} = 0$ then there will be no error whilst following the commanded trajectory. The time derivative of (4.17) is given as

$$\dot{\mathbf{s}} = (\dot{\boldsymbol{\omega}} - \dot{\boldsymbol{\omega}}_d) + k\boldsymbol{\delta}\dot{\mathbf{q}}_{1:3} \tag{4.18}$$

The error dynamics for $\boldsymbol{\delta q}_{1:3}$ are given as

$$\boldsymbol{\delta}\dot{\mathbf{q}}_{1:3} = \frac{1}{2}\delta q_4(\boldsymbol{\omega} - \boldsymbol{\omega}_d) + \frac{1}{2}\delta\mathbf{q}_{1:3} \times (\boldsymbol{\omega} + \boldsymbol{\omega}_d) \tag{4.19}$$

By substituting (4.5) and (4.19) into (4.18) gives

$$\dot{\mathbf{s}} = -\bar{J}^{-1}[\boldsymbol{\omega}\times]\bar{J}\boldsymbol{\omega} + \bar{J}^{-1}L_e - \dot{\boldsymbol{\omega}}_d + \frac{k}{2}[\delta q_4(\boldsymbol{\omega} - \boldsymbol{\omega}_d) + \delta\mathbf{q}_{1:3} \times (\boldsymbol{\omega} + \boldsymbol{\omega}_d)] \tag{4.20}$$

where L_e denotes the best estimate of the equivalent control. In order to account for model uncertainties a discontinuous term is added across the sliding surface, such as

$$\mathbf{s} = (\boldsymbol{\omega} - \boldsymbol{\omega}_d) + ksign(\delta q_4)\boldsymbol{\delta}\mathbf{q}_{1:3} \tag{4.21}$$

that gives the following control law that reorients the spacecraft in the shortest distance possible [42] [42]

$$L = \bar{J}\{\frac{k}{2}[|\delta q_4|(\boldsymbol{\omega}_d - \boldsymbol{\omega}) - sign(\delta q_4)\boldsymbol{\delta}\mathbf{q}_{1:3} \times (\boldsymbol{\omega} + \boldsymbol{\omega}_d)] + \dot{\boldsymbol{\omega}}_d - G\bar{\mathbf{s}}\} + [\boldsymbol{\omega}\times]\bar{J}\boldsymbol{\omega} \tag{4.22}$$

where G is a positive definite matrix and $\bar{\mathbf{s}} = [\bar{s}_1, \bar{s}_2, \bar{s}_3]$ and $\bar{s}_i$ are given as

$$\bar{s}_i = sat(s_i, \epsilon_i), i = 1, 2, 3 \tag{4.23}$$

where ϵ_i is a positive constant and s_i is the ith component of the sliding vector. The saturation function is defined by

$$sat(s_i, \epsilon_i) \equiv \begin{cases} 1 & \text{for } s_i > \epsilon_i \\ \frac{s_i}{\epsilon_i} & \text{for } |s_i| \leq \epsilon_i \\ -1 & \text{for } s_i < \epsilon_i \end{cases} \tag{4.24}$$

The stability of the closed-loop system of the controller defined above can be proven using the following candidate Lyapunov function [42]

$$V = \frac{1}{2}\mathbf{s}^T\mathbf{s} \tag{4.25}$$

Using (4.5), (4.21) and (4.22) the time derivative of (4.25) is given by

$$\dot{V} = -\mathbf{s}^T G\bar{\mathbf{s}} \tag{4.26}$$

which is always less than or equal to zero as long as G is positive definite.

4.2.3 PD controller - without feedforward

A considerable simplification to the tracking case could be done by assuming that the attitude variation is much smaller than the controller and system response and treating it as a fixed attitude with zero angular velocity. The performance of this method is expected to be lower but was implemented as another comparison for the methods above. In [11] different control laws are proposed, with corresponding stability analysis, but the one that demonstrated the best performance in the tracking case was

$$L = -k_p \text{sign}(\delta q_4)\boldsymbol{\delta}\mathbf{q}_{1:3} - k_d(1 \pm \boldsymbol{\delta}\mathbf{q}_{1:3}^T\boldsymbol{\delta}\mathbf{q}_{1:3})\boldsymbol{\omega} \tag{4.27}$$

4.3 Detumbling

The first maneuver to be engaged is detumbling control. As discussed in Section 4.1, these very high angular rates cannot be effectively controlled by the Reaction Wheels, therefore the Magnetometers are employed. For ISTsat-1, different detumbling algorithms were studied for the same actuators. It was concluded that the bag-bang B-dot controller would be the advisable controller when compared to the regular B-dot and the angular rate feedback due to robustness and less consumed energy [43].

The proposed B-dot controller generates a magnetic dipole based on the rate of change of the Earth's magnetic field measured such that [35]

$$\mathbf{m} = \frac{k}{||\mathbf{B}||}\boldsymbol{\omega} \times \mathbf{b} \tag{4.28}$$

where $\mathbf{b}$ is the unit vector parallel to the local geomagnetic vector. This gives the control torque

$$L = -k(I_3 - \mathbf{b}\mathbf{b}^T)\boldsymbol{\omega} \tag{4.29}$$

To prove the stability of this control law the following candidate Lyapunov function is considered:

$$V = \frac{1}{2}\boldsymbol{\omega}^T\bar{J}\boldsymbol{\omega} \tag{4.30}$$

Using equations (4.5) and (4.29), the derivative of the Lyapunov function is given by

$$\dot{V} = -k\boldsymbol{\omega}^T(I_3 - \mathbf{b}\mathbf{b}^T)\boldsymbol{\omega} \tag{4.31}$$

The function $\dot{V}$ is only negative semi-definite since when $\boldsymbol{\omega}$ is parallel to $\mathbf{b}$, $\dot{V} = 0$. In practical situations this condition is never met thus it can be considered asymptotically stable since $\boldsymbol{\omega}$ is not only decreasing but also converging to zero.

Chapter 5

Implementation

The simulation environment was developed in Matlab 2020b and Simulink and can be seen in Fig. 5.1. The model is divided into intuitive sections, similarly to this chapter. From left to right, the outputs of a block are used as the inputs to the following block. Each section has the modelling blocks, as well as separated data treatment blocks. This is done to expedite the swap of each separate block and reduce the amount of *Scopes*, *To Workspace* and other computational heavy blocks. Further data processing is done in separate scripts.

Individual models have been developed for pre-computation of the satellite, Sun and GS position and magnetic field. Although the simulation in usually ran with a 0.025 time-step, the previous data is calculated with a 0.5 second interval since it is enough to encompass the slow variations. A setup script also allows for quick changes into the simulation environment, such as sensor/actuators parameters, simulation time and initialization parameters for the real and estimate quaternion. A more detailed look into each of the subsystems is provided in Appendix C, that follows the same order as this section.

5.1 Environment

This section is responsible for the orbital motion, disturbances calculation and rotational dynamics.

The position motion of the satellite, Earth, Sun and Moon are pre-calculated in order to reduce computation time. The *planetEphemeris* function from the Aerospace toolbox gives the position and velocity of the Earth and Moon. These values are then propagated through Runge-Kutta 4^th order integration and the Sun position is calculated. The celestial bodies positions are used to calculated and propagate the satellite's position with resort to Runge-Kutta 8^th order integration. The magnetic field true model used is 12^{th} degree WMM2020 model, provided by Matlab.

The keplerian parameters for the orbit follow the Sun synchronous orbit and can be seen in Tab. 5.1. They were chosen to maximize the time that the satellite has LOS to the GS.

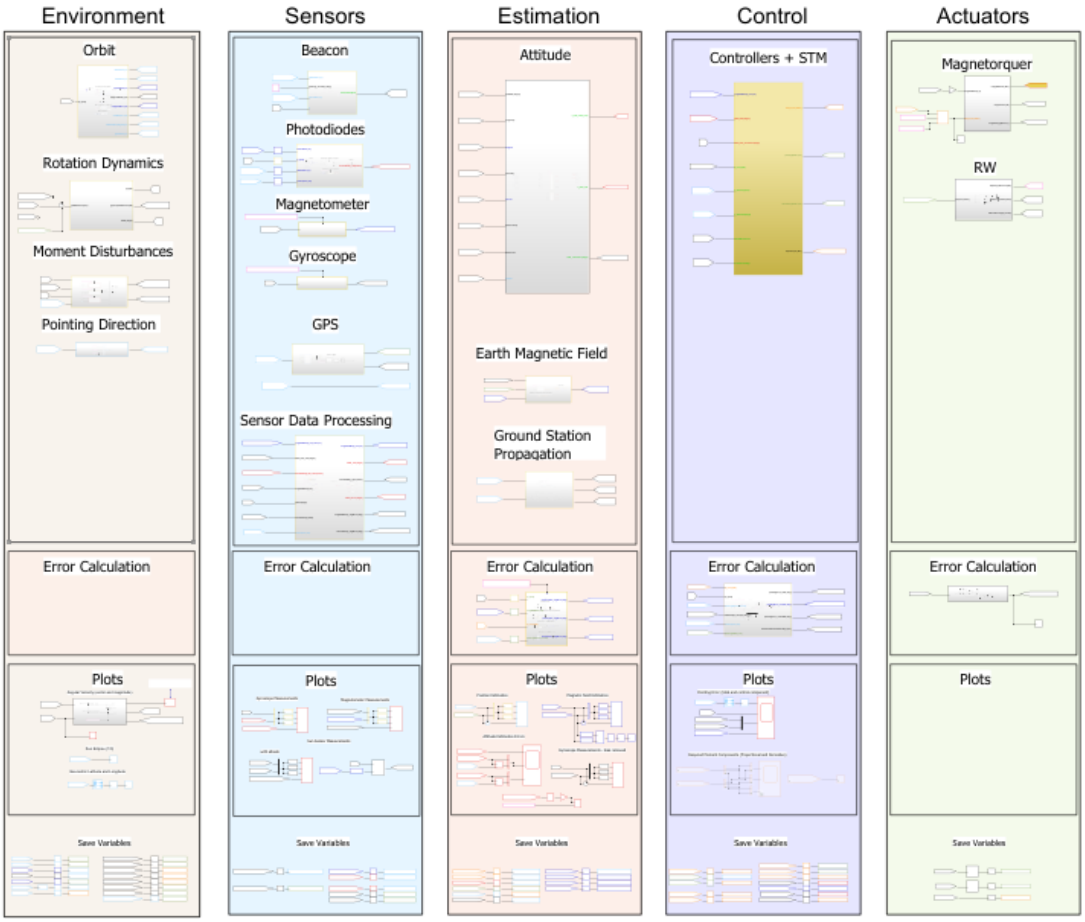

Figure 5.1: Full simulation model.

Semi-Major axis	$6871km$
Eccentricity	0
Inclination	$97.86°$
Right Ascending Node	$22.5°$

Table 5.1: Kepler parameters for Sun synchronous orbit.

The GS location is defined as has having coordinates $38°N\ 150°W$. Aiming to use them in a simulation environment, the coordinates are converted to ECI and propagated.

The rotational dynamics of the satellite follow (2.22), where the external torques are the sum of the disturbances.

5.2 Sensors

Since this project's goal is to provide an updated and improved approach to the ISTSat-1 ADCS's, the sensors same sensors are used. The simulation environment is designed to allow for testing of different sensors through a configuration file for further optimization and experimentation.

The satellite is fitted with and Inertial Measurement Unit (IMU), the TDK InvenSense MPU-9250, that

integrates a gyroscope with the parameters described in Tab. 5.2.

ADC	16 *bits*
Sampling frequency	$4 - 8000Hz$
Low-pass filter	$5 - 250Hz$
Noise RMS	$0.01°/s$

Table 5.2: Gyroscope parameters - Appendix B.1.

This leads to the a Gaussian white-noise $\boldsymbol{\eta}_{arw}$ with the following properties

$$E[\boldsymbol{\eta}_{arw}\boldsymbol{\eta}_{arw}^T] = \sigma_{arw}^2 I_3 \tag{5.1}$$

where σ_{arw} is the Allen Variation. It is assumed that the alignments and scaling factors have been calculated through ground tests.

The Sun sensor used is the Solar MEMS ISS-D25, one per face of the satellite. Some of its most relevant parameters are shown in table 5.3. It is modelled according to (3.5), where the values for the noise attitude matrices are derived from the datasheet (Appendix B.2) so that they have the same statistical properties of *Angle x* and *Angle y*.

Field of View (FOV)	$50°$
Accuracy (3 σ)	$<0,3°$
Precision	$<0,04°$
Angle resolution	$0,001$

Table 5.3: Sun sensor parameters - Appendix B.2.

Two magnetometers are available, the already mentioned MPU-9250 and the Honeywell HMC5983, in Appendix B.3. The latter is used due to its superior performance, whilst the former one can be used as a backup in future implementations. The model follows (3.3), where the noise component follows

$$E[\boldsymbol{\eta}_{mag}\boldsymbol{\eta}_{mag}^T] = \sigma_{mag}^2 I_3 \tag{5.2}$$

which has standard deviation σ_{mag} specified as $200nT$.

The beacon noise description follows the statistical characteristics from [7], that leads to a mean attitude error of $29.7\mu rad$. The GPS is modelled as having an uncertainty of $10m$ around the true satellite position, according to [10].

5.3 Estimation

The Earth magnetic field is estimated through the 5^{th} degree WMM2020, self-developed. The Sun vector true model and estimation use the same algorithm since it is suitable for onboard use.

The GS vector is calculated from the estimated position from the GPS and the known position of the GS. When the satellite is above the GS, the mission requires the payload to be pointed at it; otherwise, the payload should be pointed at nadir. A smooth transition was implemented between both cases, as can be seen in Fig. 5.2.

In order to determine the desired quaternion and angular velocity a rotation matrix was calculated by adding constraints to the Rodrigues's rotation formula. This is done by restricting the desired attitude to have the y axis aligned with the orbit's axis of rotation and $-z$ axis aligned with the GS vector, whilst the GS is starting to be visible. The quaternion is recovered from the attitude matrix and the angular velocity is determined through derivation. To extract the quaternion from the attitude matrix the Modified Shepperd's algorithm is used [44]. This algorithm doesn't recover a continuos quaternion and may lead to issues in the calculation of the desired frame. To avoid it, a selection of only one of the estimates was selected which leads to a non-optimal estimate. The alternatives found [45] require a high computational effort, that is not suitable for onboard use.

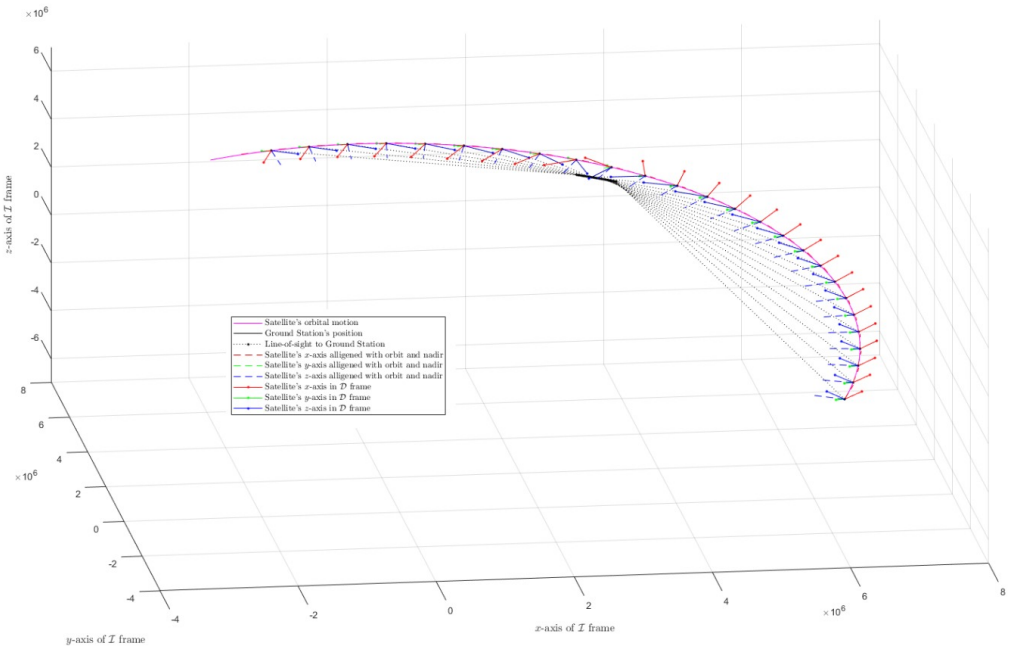

Figure 5.2: Orbit of the satellite in relation to the Ground Station.

For the MEKF, the initial parameters are shown in table 5.4. The quaternion initial estimation is varied to test for different situation.

$\mathbf{q}_0$	$[0.2608, -0.4776, 0.3602, 0.7577]$ ($5.3°$ of initial error)
	$[0.2951, -0.4104, 0.3900, 0.7697]$ ($9.8°$ of initial error)
	value from TRIAD
$\mathbf{b}_0$	$[0, 0, 0]$
P_0	$0.000001 \times I_{6\times6}$
propagation frequency	$40Hz$
update frequency	$10Hz$

Table 5.4: MEKF filter initialization parameters.

Depending on the availability of the sensors the matrix R_k changes, such that

Not in eclipse	$\begin{bmatrix} R_{mag} & 0 \\ 0 & R_{sun} \end{bmatrix}$
Eclipse, beacon not in LOS	R_{mag}
Eclipse and beacon in LOS	$\begin{bmatrix} R_{mag} & 0 \\ 0 & R_{beacon} \end{bmatrix}$

Table 5.5: MEKF measurement covariance matrix.

The values for each measurement covariance matrix were initially set to the sensors specifications and, by trial and error, were tuned to

R_{mag}	$0.0055 \times I_3$
R_{sun}	$0.01 \times diag(0.16, 0.16, 0.45)$
R_{beacon}	$10^{(-5)} \times diag(0.1, 0.1, 0.3)$

Table 5.6: Measurement covariance matrix values.

5.4 Control and Actuators

The reaction wheels used are the Cubespace Cubewheel Small, one per axis. The parameters for the reaction wheel can be seen in Tab. 5.7.

Max RPM	8000
Max. Torque	$0.23mT$
Max. stored Momentum	$1.77mT/s$
Inertia	3.12×10^{-6}

Table 5.7: Reaction wheel parameters - Appendix B.4.

The first-order system that models the reaction wheel is tuned to have the response according to the manufacturer's parameters, with saturation for the stored momentum/RPM (revolutions per minute). The

output torque is controlled by a reference torque provided by the controller. The step response for the maximum torque in the time domain can be seen in Fig. 5.3. It can be seen that the output torque is zero when the reaction wheel reaches its maximum stored momentum.

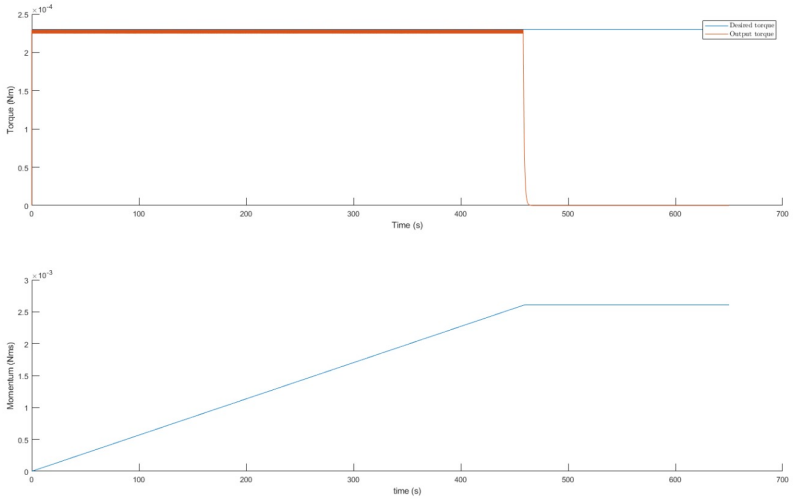

Figure 5.3: Stored momentum and torque response for a step of the maximum torque.

The second-order system was tuned by analyzing the manufacturer's properties and matching it to the general flywheel equations [46]

$$L_{fw} = J_{fw}\dot{\boldsymbol{\omega}}_{fw} = kmr^2\dot{\boldsymbol{\omega}}_{fw} \tag{5.3a}$$

$$H_{fw} = \frac{1}{2}J_{fw}\boldsymbol{\omega}_{fw}^2 \tag{5.3b}$$

where L_{fw} is the produced torque and H_{fw} the momentum stored by the flywheel, J_{fw} is the moment of inertia of the wheel, $\boldsymbol{\omega}_{fw}$ the angular velocity of the wheel, m the mass, r the radius and k the inertia constant [1].

The satellite will be powered by the EuduroSat solar panels that have an optional feature which includes a printed coil in the back of the panel to work as magnetorquers. Only 3 solar panels are fitted with the optional coil, one per axis. These magnetorquers provide a magnetic dipole of $131mAm^2$ and are controlled through PWM.

The sliding controller uses the parameters in Tab. 5.8.

[1] the parameters m, r and k where the main parameters used to tune the model and were extrapolated from the datasheet

Frequency	$1Hz$
K	0.026
ϵ	8×10^{-3}
G	$5 \times 10^{-4} I_3$
$\hat{J}$	J_{sat}

Table 5.8: Sliding Controller parameters.

The GFTAC controller uses the parameters in Tab. 5.9.

Frequency	$1Hz$
k_1	8.5×10^{-4}
k_2	5×10^{-4}
α_1	0.6
α_2	0.75
σ	0.4

Table 5.9: Sliding Controller parameters.

The PD controller uses the parameters in Tab. 5.10.

k_d	2.3×10^{-5}
k_p	8.4×10^{-5}

Table 5.10: PD Controller parameters.

Chapter 6

Testing

In order to select the appropriate combination of sensors, actuators and algorithms, different scenarios must be run. The simulation uses the parameters discussed in Chapter 5, with different angular rate and quaternion initializations. The detumbling controller can be tested separately since is the first maneuver to be performed and it does not use bias estimation to correct the gyroscope measurements and the reaction wheels are not used.

Since the beacon is not a regular sensor used in a CubeSats, a verification of its usability must be done before integrating into the mission implementation. The MEKF is used to compare the performance with and without the beacon due to the convenience of tuning it for both options.

A comparison between the different methods must be done in order to verify if the computational complexity that the filters add does in fact lead to better attitude and bias estimation. The estimation is evaluated separately with identical simulation environments for the different methods. The different control algorithms must also be tested without the influence of the estimators. The convergence time, the angular velocity and pointing error are evaluated. After the preferable method for estimation and control are chosen a simulation is made to verify wether the pointing requirements are met.

6.1 Detumbling

In order to test the detumbling maneuver a simulation is started with a $10^{\circ}/s$ angular rate, the expected tumbling rate after leaving the launcher. The results are presented in Fig. 6.1, detailing the angular velocity in the different axes. After 1.3474 orbits the angular relate crosses the detumbled threshold of $5^{\circ}/s$ and, with the use of the controllers, is slowed down further. Although the satellite is slowed down to the required threshold, it does so in more than one orbit.

6.2 Beacon effectiveness verification

In order to verify if the estimation accuracy is improved with the use of the beacon, a comparison of the two cases (with and without the beacon) is done using the MEKF. Two distinct scenarios were used

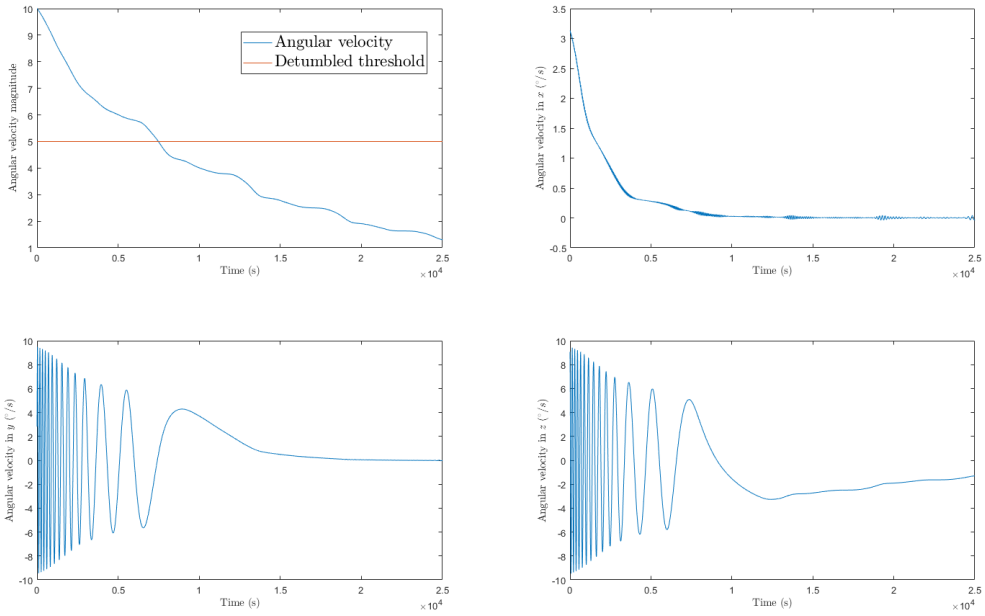

Figure 6.1: Detumbling controller with a $10°/s$ angular rate.

for the initial error estimation, with the results presented in Tab. 6.1 and in Fig. 6.2. It is assumed that when the beacon is in the LOS it is also within the FOV of the telescope.

The addition of the beacon decreases the attitude estimation error drastically, especially when it is only considered the period where the GS is in the LOS of the satellite and can be used to improve estimation. A decline in the error estimation can be observed in Fig. 6.2 as soon as the measurement is available. The estimation performance decreases in eclipse due to the unavailability of the Sun sensors and without the integration of the beacon in the filter it never meets the required accuracy of $0.1°$. The bias estimation without the use of the beacon results in a higher accuracy and a smoother conversion to the real value.

The initial estimate for the quaternion has a significant effect in the attitude estimation error and results in less accurate estimates in LOS due to a longer convergence period. It has little effect on the bias estimation. If the filter is run long enough, both of the scenarios will perform similarly.

It can be concluded that the beacon has a positive impact in the estimation and should be used in the attitude estimation. However, the availability is limited due to only being available in eclipse and in LOS.

Mean	**With beacon**		**Without beacon**	
Initial error (°)	5.3	9.8	5.3	9.8
Attitude estimation error (°)	0.7128	0.7974	1.3985	1.1611
Attitude estimation error in LOS (°)	0.1884	0.4429	1.4710	1.7852
Bias estimation error ($°/s$)	3×10^{-4}	2.6433×10^{-4}	2.9433×10^{-3}	2.6012×10^{-4}

Table 6.1: Attitude and bias estimation performace with and without beacon.

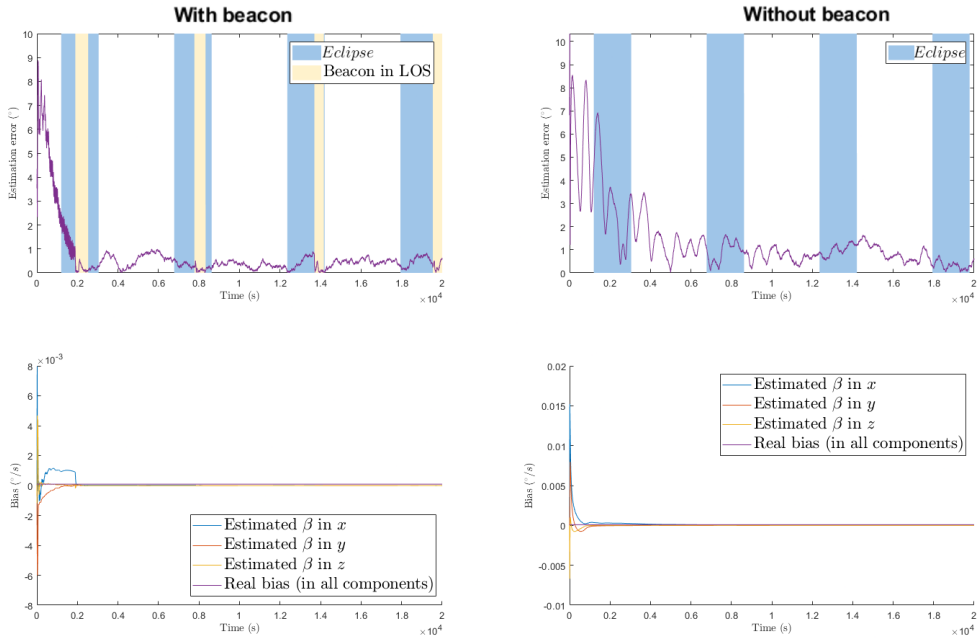

Figure 6.2: Attitude (top) and bias (bottom) estimation with (left) and without (right) beacon.

6.3 Comparison of estimation methods

In order to compare the attitude estimation methods scenario is simulated using the same initial values for quaternion and velocity. For the MEKF and CF, the initial quaternion estimate used illustrates $5.3°$ of initial error. The results of the simulation can be seen Fig. 6.3 and Tab. 6.3. The TRIAD method doesn't estimate the bias. As done in Section 6.2, it is assumed that when the beacon is in the LOS it is also within the FOV of the telescope. Without it, the CF wouldn't be comparable since it has an attitude error greater than the FOV of the telescope in eclipse.

A comparison of the advantages and disadvantages of the estimation methods can be seen in Tab. 6.3. Although the CF has a high mean attitude estimation error, it mainly results from the eclipse phase. When only the magnetometer and gyroscope are available the attitude error increases drastically. Its accuracy is not high enough in the mission area since it doesn't reach a $0.1°$ accuracy consistently. The bias estimate is also far from the real value.

The TRIAD method has a consistent estimate, that is only slightly improved with the addition of the beacon. Since it only uses 2 measurements, it doesn't provide an estimate in eclipse. As it is a deterministic method it doesn't need to converge, which is useful to provide an initial estimate for other methods considering it has a low computational demand.

The MEKF demonstrate an acceptable accuracy when the beacon is used, although it is not throughout the full duration of the LOS. This is due to the fact that it has a convergence period when the beacon available. In eclipse the estimation is also deteriorated, but not as much as the CF. This results in a better bias estimation, that is used to correct the angular velocity measurements in the propagation phase. Tests were done with only the propagation step in eclipse, but it was found that the update with only one

measurement was still beneficial for the attitude estimation. Hence, his method is the one chosen for the mission implementation.

Mean	MEKF	TRIAD	CF
Attitude estimation error (°)	0.9429	0.6775	25.1388
Attitude estimation error in LOS (°)	0.3780	0.5224	2.3361
Bias estimation error ($°/s$)	2.7×10^{-4}	N/A	0.4529

Table 6.2: Comparison of the attitude estimation methods.

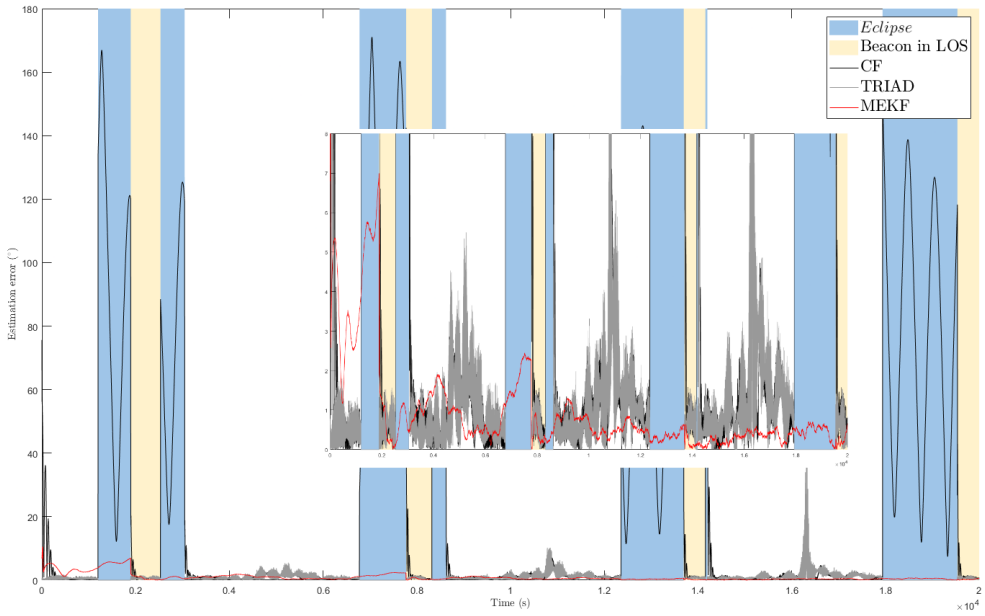

Figure 6.3: Comparison of the attitude estimation methods.

	Advantages	Disadvantages
TRIAD	- Consistent and low computational effort - Provides an estimation with only two samples, without the need to converge	- Does not use past information to refine estimation - Does not estimate gyroscope bias - Is limited by the sensors' performance
CF controller	- Low computational effort (in comparison to state estimation algorithms) - Good estimation when the beacon is available	- Bad estimation in eclipse - Accuracy bellow expectations
MEKF	- Very good attitude estimation after the filter converges, especially with the incorporation of the beacon	- High computational efforts - Unaccurate representation of the gyroscope bias

Table 6.3: Advantages and disadvantages of estimation algorithms.

Mean	GFTAC	Sliding	PD
Attitude control error (°)	0.2244	1.1888	3.0028
Angular velocity error ($^\circ/s$)	0.0689	0.0637	0.073

Table 6.4: Comparison of controllers.

6.4 Comparison of control methods

To test the different control algorithms, a scenario was simulated where there was a large initial error and a non-zero angular rate. This allowed for the controllers to be tested in how they converged to the desired frame. The controllers were tuned to don't surpass the detumbling threshold for the angular rate whilst converging or following the desired frame. To only evaluate the controllers' performance the real quaternion was used to calculate $\boldsymbol{\delta q}$ instead of the estimated quaternion. The results are presented in Tab. 6.4 and in Fig. 6.4.

A comparison of the advantages and disadvantages of the control methods can be seen in Tab. 6.5. As expected, the PD controller has a high angular velocity and angular error and is incapable to react to the fast attitude variations whilst overflying the GS. It also has a larger steady-state error, but converges to it quicker than the sliding controller.

The sliding controller has the longest convergence time but when the desired frame is reached it has a much better performance than the PD controller. Although the angular velocity error is comparable to the GFTAC controller, the angular error while overflying the GS is larger.

The GFTAC controller achieves an angular error below 0.1° throughout the simulation, after convergence. Due to having the fastest convergence times, it has a large angular rate in this period. Considering its performance isn't deteriorated while overflying the GS it is chosen for the mission implementation.

	Advantages	Disadvantages
PD Controller	- Simple implementation	- Unable to cope with fast attitude changes
Sliding controller	- Very good angular velocity tracking - Good angular tracking	- Very complicated implementation - Fast-switching control law
GFTAC controller	- Good angular velocity tracking - Very good angular tracking	- Complicated implementation - Hard to have a fast initial conversion without surpassing the detumbling threshold for the angular velocity

Table 6.5: Advantages and disadvantages of controllers.

6.5 Mission mode control and estimation

For the mission, the MEKF is selected as the estimation algorithm. In this scenario, the beacon is only available if the GS is in the LOS of the satellite and if it is within the FOV of the telescope. Namely, if the pointing error rises above 10° the sensor is not available. The TRIAD algorithm is run once for the initialization of the filter, out of eclipse so that it can use both the magnetometer and sun sensors. When it comes to the controller, the GFTAC was chosen for its consistency in pointing accuracy and low angular velocity error. The quaternion estimate was used to calculate $\boldsymbol{\delta q}$ and the bias used to correct

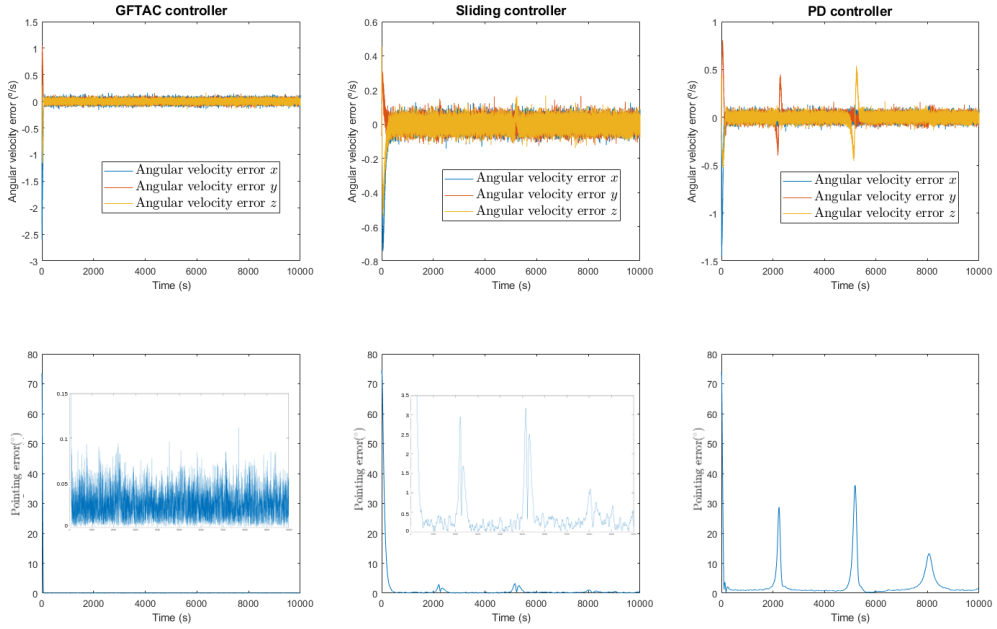

Figure 6.4: GFTAC (left), sliding (middle) and PD (right) controllers angular velocity (top) and pointing error (bottom) for the same initial attitude error.

the angular velocity measurements from the gyroscope.

Fig. 6.5 shows the pointing error of the complete system. As observed in Section 6.2 and Section 6.2, the highest values for the pointing error are observed in eclipse and are improved whilst the beacon is LOS. It can be said that the estimation is predominantly responsible for the errors then. The initialization of the filter with the TRIAD method resulted in lower overall errors throughout the simulation. A pointing error lower than $0.1°$ was accomplished for 28.29% of the time in LOS.

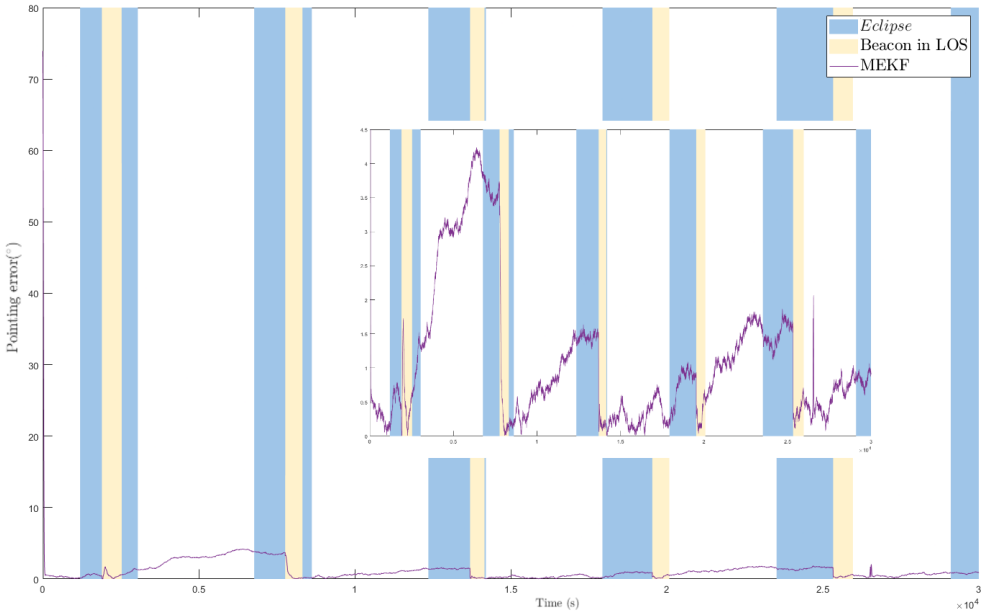

Figure 6.5: Complete mission mode simulation with MEFK estimation and GFTAC control.

Chapter 7

Conclusions

The main focus of this dissertation was to provide a functional ADCS for QuantSat-PT, using the payload's beacon detection as a sensor. Additionally, there were strict restrictions, especially in the volume, that limited greatly the hardware available. A pair of estimation and control algorithms had to analyzed in such a way that could steer the satellite's design and provide a proof of concept for the architecture.

Before deciding the algorithms that would be used, focused was put on the sensors and actuators. A study was conducted, based on literature, in order to verify which suite combinations would be useful, as well as how the beacon could replace the more usual sensors. According to the mission's goal, a breakdown of performance objectives was listed, such as required accuracy and frequency of sensors. To complement the beacon, a gyroscope, a magnetometer and a Sun sensor were selected. The reaction wheels' cluster and a magnetorquer are responsible for attitude control. Modelling of each sensor and actuator processed their choice, with individual testing to corroborate their workings. Since the beacon is an uncommon sensor, verification of its effectiveness was made in Section 6.2.

The simulation environment was developed based on the concepts presented in Chapter 2 and allowed for complete simulations to be run. The attitude kinematics and dynamics were simulated, as well as the orbital motion of the satellite and the relevant celestial bodies.

After the simulation platform was finalized, the work in the algorithms began. The TRIAD algorithm has a very low complexity, resulting in a lower computational effort. This allows for higher frequency for attitude estimation, as long as the necessary measurements are available. It does not need initialization which allows for fairly accurate attitude estimate from the start. The major downfalls of this method lie in the eclipse phase and the lack of bias estimation. Only when two measurements that follow the rules i.e. are not parallel, are available, an attitude estimation is computed.

The recursive filter showed much better results, especially with the inclusion of the beacon, due to its accurate measurements. The attitude estimation error was decreased while in LOS, even with the relatively short duration of the period. The bias estimation is also a major advantage since it allows for a more accurate angular velocity measurement, which is not only useful for the propagation itself. When both methods are compared, it is clear that the MEKF has the upper hand when it comes to attitude and

bias estimation. This is mostly due to its robust response whilst in the eclipse stage, dealing with the lack of measurements better than the CF. However, it can be said that the CF converges to the lower boundary of estimate much faster when the beacon is available.

When it comes to control algorithms, it is clear that without velocity tracking it is not possible to have a correct heading above the GS. The PD controller, however simple, is not enough for this high slew-rate mission. With the introduction of the desired torque from the GFTAC controller as feedforward the performace would be improved, however it would increase its complexity, which was the main reason for its choice. The performance improvements of the sliding and GFTAC controllers come with heavy computational efforts. The sliding controller has a smoother conversion to the desired velocity, which results in lower angular rates. This might be beneficial if the complete satellite is prone to instabilities that might exceed the detumbled threshold. For the pointing error, the GFTAC controller has superior performance, even though the angular velocity tracking error is comparable to the sliding controller. The B-dot controller is robust and is able to perform the detumbling manoeuvre. However, it does so in more than one orbit, which isn't good enough. A selection of more appropriate solar panel with embedded magnetorquers and enough wattage must be performed or a standalone magnetorquers could be considered.

When the estimation and attitude control are paired together, the pointing error follows the same profile as the estimation error. This hints on the aspect that could use improvements, although meeting the performance requirements for 28.29% of the time. The TRIAD method for initialization is a great way to reduce converging time, but could be left out if constrains were put in the onboard computers' memory.

Although the percentage of time that the estimation error is maintained under the 0.1° mark while in the mission zone seems disappointing, the implementation is able to fulfill the mission requirements. The fine stage is able to complete the further alignments and the quantum communications can take place in the remainder of the time. Even though the designed ADCS is not displaying the best possible performance indicated in literature for comparable sensor suites, it is able to comply with the point the payload to the GS with acceptable performance.

7.1 Future Work

When it comes to improving the work done in this thesis, there are two main areas that could be improved: the fidelity of the simulation and the estimation and control methods. The simulation environment must integrate the fine pointing mode in order to prove that the entire system could work and solve the interaction constraints between the systems, that will undoubtedly arise. The reaction wheel's model could be improved with tests on the actual hardware or a better model, that would have to be provided by the manufacturer. The simulation for the beacon represents how the beacon was characterized at the time; when its architecture is better defined, the simulation should depict it. Other strategies for the desired frame could be explored, with the goal of minimizing attitude representations' conversion problems.

As discussed, the estimation seems to be the limiting factor in pointing accuracy. It was established that recursive methods have better results than static methods, so a greater emphasis could be put on them. Although a MEFK was implemented, other variations to the EKF could be studied, such as the UKF. The architecture for an estimator is defined and should be easier to implement. When it comes to control algorithms, other configurations of reaction wheels could be studied in order to provide redundancy. The momentum dumping maneuver is one of the basic manoeuvres left to be studied.

Bibliography

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

[2] About Fly Your Satellite! URL https://www.esa.int/Education/CubeSats_-_Fly_Your_Satellite/About_Fly_Your_Satellite!

[3] S. T. Gutierrez, C. I. Fuentes, and M. A. Diaz. Introducing SOST: An Ultra-Low-Cost Star Tracker Concept Based on a Raspberry Pi and Open-Source Astronomy Software. 8:166320–166334. ISSN 2169-3536. doi: 10.1109/ACCESS.2020.3020048. URL https://ieeexplore.ieee.org/document/9179736/.

[4] R. Renner. SECURITY OF QUANTUM KEY DISTRIBUTION. page 130.

[5] S.-K. Liao. Satellite-to-ground quantum key distribution. 549(7670):43–47. ISSN 0028-0836, 1476-4687. doi: 10.1038/nature23655. URL http://www.nature.com/articles/nature23655.

[6] P. Serra, O. Čierny, W. Kammerer, E. S. Douglas, D. W. Kim, J. N. Ashcraft, G. Smith, C. Guthery, T. Vergoossen, A. Lohrmann, R. Bedington, C. Perumangatt, A. Ling, and K. Cahoy. Optical front-end for a quantum key distribution cubesat. In Z. Sodnik, B. Cugny, and N. Karafolas, editors, International Conference on Space Optics — ICSO 2020, page 118. SPIE. ISBN 978-1-5106-4548-6 978-1-5106-4549-3. doi: 10.1117/12.2599542. URL https://www.spiedigitallibrary.org/conference-proceedings-of-spie/11852/2599542/Optical-front-end-for-a-quantum-key-distribution-cubesat/10.1117/12.2599542.full.

[7] T. Nguyen, K. Riesing, R. Kingsbury, and K. Cahoy. Development of a pointing, acquisition, and tracking system for a CubeSat optical communication module. page 93540O. doi: 10.1117/12.2080591. URL http://proceedings.spiedigitallibrary.org/proceeding.aspx?doi=10.1117/12.2080591.

[8] T. S. Rose, D. W. Rowen, S. D. LaLumondiere, N. I. Werner, R. Linares, A. C. Faler, J. M. Wicker, C. M. Coffman, G. A. Maul, D. H. Chien, A. C. Utter, R. P. Welle, and S. W. Janson. Optical communications downlink from a low-earth orbiting 15U CubeSat. 27(17):24382. ISSN 1094-4087. doi: 10.1364/OE.27.024382. URL https://www.osapublishing.org/abstract.cfm?URI=oe-27-17-24382.

[9] A. Borralho. Alignment Control of an Optical Link in a Turbulent Channel for Satellite Communication Systems.

[10] A. Noureldin, T. B. Karamat, and J. Georgy. Fundamentals of Inertial Navigation, Satellite-based Positioning and Their Integration. Springer Berlin Heidelberg. ISBN 978-3-642-30465-1 978-3-642-30466-8. doi: 10.1007/978-3-642-30466-8. URL http://link.springer.com/10.1007/978-3-642-30466-8.

[11] F. L. Markley and J. L. Crassidis. Fundamentals of Spacecraft Attitude Determination and Control. Springer New York. ISBN 978-1-4939-0801-1 978-1-4939-0802-8. doi: 10.1007/978-1-4939-0802-8. URL http://link.springer.com/10.1007/978-1-4939-0802-8.

[12] J. R. Wertz and W. J. Larson, editors. Space Mission Analysis and Design. Space Technology Library. Microcosm ; Kluwer, 3rd ed edition. ISBN 978-0-7923-5901-2 978-1-881883-10-4.

[13] C. S. Corporation and J. R. Wertz, editors. Spacecraft Attitude Determination and Control. Number v. 73 in Astrophysics and Space Science Library. Reidel. ISBN 978-90-277-0959-2.

[14] V. Carrara, R. B. Januzi, D. H. Makita, L. F. d. P. Santos, and L. S. Sato. The ITASAT CubeSat Development and Design. 9(2):147–156. ISSN 2175-9146. doi: 10.5028/jatm.v9i2.614. URL http://www.jatm.com.br/ojs/index.php/jatm/article/view/614.

[15] Y. Cheng, Z. Sun, D. Li, and D. Ye. The design of attitude measurement unit based on multiple MEMS gyros. In 2014 International Conference on Mechatronics and Control (ICMC), pages 1314–1318. IEEE. ISBN 978-1-4799-2538-4. doi: 10.1109/ICMC.2014.7231765. URL http://ieeexplore.ieee.org/document/7231765/.

[16] M. S. Challa. Magnetometer-only attitude and rate determination for a Gryo-less Spacecraft. page 8.

[17] A. Adnane, H. Jiang, M. A. S. Mohammed, A. Bellar, and Z. A. Foitih. Reliable Kalman Filtering for Satellite Attitude Estimation Under Gyroscope Partial Failure. In 2018 2nd IEEE Advanced Information Management,Communicates,Electronic and Automation Control Conference (IMCEC), pages 449–454. IEEE. ISBN 978-1-5386-1803-5. doi: 10.1109/IMCEC.2018.8469422. URL https://ieeexplore.ieee.org/document/8469422/.

[18] F. Wu, X. Cao, and J. Zhang. Nano satellite attitude estimation with magnetometer and photodiodes measurements. In 2016 Chinese Control and Decision Conference (CCDC), pages 6714–6718. IEEE. ISBN 978-1-4673-9714-8. doi: 10.1109/CCDC.2016.7532205. URL http://ieeexplore.ieee.org/document/7532205/.

[19] D. Cilden-Guler, H. Schaub, C. Hajiyev, and Z. Kaymaz. Attitude Estimation with Albedo Interference on Sun Sensor Measurements. 58(1):148–163. ISSN 0022-4650, 1533-6794. doi: 10.2514/1.A34814. URL https://arc.aiaa.org/doi/10.2514/1.A34814.

[20] H. Ma, Z. Lu, X. Zhang, W. Liao, and K. Briess. High-Accuracy and Low-Cost Attitude Measurement Unit of the CubeSat. 2020:1–13. ISSN 1687-5966, 1687-5974. doi: 10.1155/2020/4973970. URL https://www.hindawi.com/journals/ijae/2020/4973970/.

[21] J. Bae, Y. Kim, and H. S. Kim. Satellite Attitude Determination and Estimation using Two Star Trackers. In AIAA Guidance, Navigation, and Control Conference. American Institute of Aeronautics and Astronautics. ISBN 978-1-60086-962-4. doi: 10.2514/6.2010-8447. URL https://arc.aiaa.org/doi/10.2514/6.2010-8447.

[22] S. Thrun. Probabilistic robotics. 45(3):52–57. ISSN 0001-0782, 1557-7317. doi: 10.1145/504729.504754. URL https://dl.acm.org/doi/10.1145/504729.504754.

[23] F. L. Lewis. Optimal and Robust Estimation: With an Introduction to Stochastic Control Theory, Second Edition. page 547.

[24] J. L. Crassidis and J. L. Junkins. Optimal Estimation of Dynamic Systems. page 750.

[25] M. L. Psiaki, E. M. Klatt, P. M. Kintner, and S. P. Powell. Attitude Estimation for a Flexible Spacecraft in an Unstable Spin. 25(1):88–95. ISSN 0731-5090, 1533-3884. doi: 10.2514/2.4853. URL https://arc.aiaa.org/doi/10.2514/2.4853.

[26] F. L. Markley. Attitude Error Representations for Kalman Filtering. 26(2):311–317, . ISSN 0731-5090, 1533-3884. doi: 10.2514/2.5048. URL https://arc.aiaa.org/doi/10.2514/2.5048.

[27] F. L. Markley. Attitude Estimation or Quaternion Estimation? 52(1-2):221–238, . ISSN 0021-9142, 2195-0571. doi: 10.1007/BF03546430. URL https://link.springer.com/10.1007/BF03546430.

[28] E. Lefferts, F. Markley, and M. Shuster. Kalman Filtering for Spacecraft Attitude Estimation. 5(5): 417–429. ISSN 0731-5090, 1533-3884. doi: 10.2514/3.56190. URL https://arc.aiaa.org/doi/10.2514/3.56190.

[29] R. Mahony, T. Hamel, and J.-M. Pflimlin. Complementary filter design on the special orthogonal group SO(3). In Proceedings of the 44th IEEE Conference on Decision and Control, pages 1477–1484. IEEE, . ISBN 978-0-7803-9567-1. doi: 10.1109/CDC.2005.1582367. URL http://ieeexplore.ieee.org/document/1582367/.

[30] R. Mahony, T. Hamel, and J.-M. Pflimlin. Nonlinear Complementary Filters on the Special Orthogonal Group. 53(5):1203–1218, . ISSN 0018-9286. doi: 10.1109/TAC.2008.923738. URL http://ieeexplore.ieee.org/document/4608934/.

[31] M. Ovchinnikov and J. Barrington-Brown. Attitude determination and control systems. In Cubesat Handbook, pages 263–281. Elsevier. ISBN 978-0-12-817884-3. doi: 10.1016/B978-0-12-817884-3.00014-X. URL https://linkinghub.elsevier.com/retrieve/pii/B978012817884300014X.

[32] P. Tsiotras, H. Shen, and C. Hall. Satellite Attitude Control and Power Tracking with Momentum Wheels. page 13.

[33] C. Gurrisi, R. Seidel, S. Dickerson, S. Didziulis, P. Frantz, and K. Ferguson. Space Station Control Moment Gyroscope Lessons Learned. page 16.

[34] V. Carrara and H. K. Kuga. Torque and speed control loops of a reaction wheel. page 11.

[35] G. Avanzini and F. Giulietti. Magnetic Detumbling of a Rigid Spacecraft. 35(4):1326–1334. ISSN 0731-5090, 1533-3884. doi: 10.2514/1.53074. URL https://arc.aiaa.org/doi/10.2514/1.53074.

[36] A. J. Westfall. Design of an Attitude Control System for Spin-Axis Control of a 3U CubeSat. URL https://scholarworks.sjsu.edu/etd_theses/4672.

[37] S. L. Scrivener and R. C. Thompson. Survey of time-optimal attitude maneuvers. 17(2):225–233. ISSN 0731-5090, 1533-3884. doi: 10.2514/3.21187. URL https://arc.aiaa.org/doi/10.2514/3.21187.

[38] P. Tsiotras. Stabilization and optimality results for the attitude control problem. 19(4):772–779. ISSN 0731-5090, 1533-3884. doi: 10.2514/3.21698. URL https://arc.aiaa.org/doi/10.2514/3.21698.

[39] H. Gui and G. Vukovich. Global finite-time attitude tracking via quaternion feedback. 97:176–183. ISSN 01676911. doi: 10.1016/j.sysconle.2016.09.017. URL https://linkinghub.elsevier.com/retrieve/pii/S016769111630144X.

[40] C. G. Mayhew, R. G. Sanfelice, and A. R. Teel. Quaternion-Based Hybrid Control for Robust Global Attitude Tracking. 56(11):2555–2566. ISSN 0018-9286. doi: 10.1109/TAC.2011.2108490. URL http://ieeexplore.ieee.org/document/5701762/.

[41] J.-J. E. Slotine and W. Li. Applied Nonlinear Control. Prentice Hall. ISBN 978-0-13-040890-7.

[42] J. L. Crassidis, S. R. Vadali, and F. L. Markley. Optimal Variable-Structure Control Tracking of Spacecraft Maneuvers. 23(3):564–566. ISSN 0731-5090, 1533-3884. doi: 10.2514/2.4568. URL https://arc.aiaa.org/doi/10.2514/2.4568.

[43] D. Neves. Control Algorithm for the ISTsat-1.

[44] F. L. Markley. Unit Quaternion from Rotation Matrix. 31(2):440–442, . ISSN 0731-5090, 1533-3884. doi: 10.2514/1.31730. URL https://arc.aiaa.org/doi/10.2514/1.31730.

[45] J. Wu. Optimal Continuous Unit Quaternions from Rotation Matrices. 42(4):919–922. ISSN 0731-5090, 1533-3884. doi: 10.2514/1.G004043. URL https://arc.aiaa.org/doi/10.2514/1.G004043.

[46] P. Gamboa, S. F. Pinto, J. F. Silva, and E. Margato. A flywheel energy storage system with Matrix Converter controlled Permanent Magnet Synchronous Motor. In 2008 18th International Conference on Electrical Machines, pages 1–5. IEEE. ISBN 978-1-4244-1735-3. doi: 10.1109/ICELMACH.2008.4799861. URL http://ieeexplore.ieee.org/document/4799861/.

Appendix A

Review of notation

A.1 Cross product

The cross of two vector $\mathbf{u}$ and $\mathbf{v}$, can be written as $\mathbf{u} \times \mathbf{v}$. However, using the skew-symmetric matrix, it can be written as $[\mathbf{u}\times]\mathbf{v}$, with

$$[\mathbf{u}\times] = \begin{bmatrix} 0 & u_3 & -u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{bmatrix} \tag{A.1}$$

A.2 Quaternion operations

The quaternions $\mathbf{q}$ were introduced by William Rowan Hamilton in 1843 as an extension of the complex number. However, the notation used in this work follows [11], leading to a three-vector part $\mathbf{q}_{1-3}$ and a scalar part q_4, such that

$$\mathbf{q} = \begin{bmatrix} \mathbf{q}_{1:3} \\ q_4 \end{bmatrix} \tag{A.2}$$

We define two 4×4 matrices analogous to the 3×3 cross-product matrix:

$$[\mathbf{q}\otimes] \equiv \begin{bmatrix} q_4 I_3 - [\mathbf{q}_{1:3}\times] & \mathbf{q}_{1:3} \\ -\mathbf{q}_{1:3}^T & q_4 \end{bmatrix} = [\Psi(\mathbf{q})\mathbf{q}] \tag{A.3}$$

$$[\mathbf{q}\odot] \equiv \begin{bmatrix} q_4 I_3 + [\mathbf{q}_{1:3}\times] & \mathbf{q}_{1:3} \\ -\mathbf{q}_{1:3}^T & q_4 \end{bmatrix} = [\Xi(\mathbf{q})\mathbf{q}] \tag{A.4}$$

Two alternative products of a pair of quaternions $\bar{\mathbf{q}}$ and $\mathbf{q}$ are defined as:

$$\mathbf{q} \otimes \overline{\mathbf{q}} = [\mathbf{q}\otimes]\overline{\mathbf{q}} \tag{A.5}$$

$$\mathbf{q} \odot \overline{\mathbf{q}} = [\mathbf{q}\circ]\overline{\mathbf{q}} \tag{A.6}$$

The identity quaternion is defined as:

$$\mathbf{I}_q \equiv \begin{bmatrix} \mathbf{0}_3 \\ 1 \end{bmatrix} \tag{A.7}$$

obeying all the properties commonly know of the identity. The conjugate $\mathbf{q}^*$ of a quaternion is:

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

and the inverse of a quaternion having non-zero norm is:

$$\mathbf{q}^{-1} \equiv \mathbf{q}^* / \|\mathbf{q}\|^2$$

Appendix B

Technical Datasheets

Datasheets for relevant sensors and actuators are presented bellow.

B.1 TDK InvenSense MPU-9250

3 Electrical Characteristics

3.1 Gyroscope Specifications

Typical Operating Circuit of section 4.2, VDD = 2.5V, VDDIO = 2.5V, T_A=25°C, unless otherwise noted.

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS
Full-Scale Range	FS_SEL=0		±250		º/s
	FS_SEL=1		±500		º/s
	FS_SEL=2		±1000		º/s
	FS_SEL=3		±2000		º/s
Gyroscope ADC Word Length			16		bits
Sensitivity Scale Factor	FS_SEL=0		131		LSB/(º/s)
	FS_SEL=1		65.5		LSB/(º/s)
	FS_SEL=2		32.8		LSB/(º/s)
	FS_SEL=3		16.4		LSB/(º/s)
Sensitivity Scale Factor Tolerance	25°C		±3		%
Sensitivity Scale Factor Variation Over Temperature	-40°C to +85°C		±4		%
Nonlinearity	Best fit straight line; 25°C		±0.1		%
Cross-Axis Sensitivity			±2		%
Initial ZRO Tolerance	25°C		±5		º/s
ZRO Variation Over Temperature	-40°C to +85°C		±30		º/s
Total RMS Noise	DLPFCFG=2 (92 Hz)		0.1		º/s-rms
Rate Noise Spectral Density			0.01		º/s/√Hz
Gyroscope Mechanical Frequencies		25	27	29	KHz
Low Pass Filter Response	Programmable Range	5		250	Hz
Gyroscope Startup Time	From Sleep mode		35		ms
Output Data Rate	Programmable, Normal mode	4		8000	Hz

Table 1 Gyroscope Specifications

3.3 Magnetometer Specifications

Typical Operating Circuit of section 4.2, VDD = 2.5V, VDDIO = 2.5V, T_A=25°C, unless otherwise noted.

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS
MAGNETOMETER SENSITIVITY					
Full-Scale Range			±4800		µT
ADC Word Length			14		bits
Sensitivity Scale Factor			0.6		µT / LSB
ZERO-FIELD OUTPUT					
Initial Calibration Tolerance			±500		LSB

B.2 Solar MEMS ISS-D25

6. Characteristics of the ISS-DX

Parameter	D60	D25	D15	D5	Unit	Comments
Sensor type	2 axes	2 axes	2 axes	2 axes	-	Orthogonal
Field of view (FOV)	120	50	30	10	º	Aperture of the cone of view
Accuracy	< 0,4	< 0,3	< 0,2	< 0,1	º	3σ
Precision	< 0,06	< 0,04	< 0,02	< 0,005	º	Sensitivity
Angle resolution	0.01	0.001	0.001	0.001	º	
Radiation accuracy	< 10	< 10	< 10	< 10	%	As accurate as close to normal vector
Radiation resolution	1	1	1	1	W/m^2	
Max. radiation	1200	1200	1200	1200	W/m^2	
Temperature accuracy	2	2	2	2	ºC	
Temperature resolution	1	1	1	1	ºC	
Sampling frecuency	50	50	50	50	Hz	
Bandwidth	0,4	0,4	0,4	0,4	Hz	
T^a 25ºC, V_{DD} 5V, Radiation 900 W/m^2						
Expected life time: 10 years +						

Table 5. Characteristics of the sensor

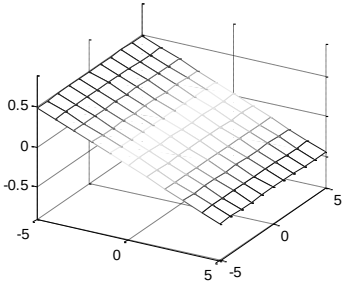

Fig 3. Sensor response ISSD5: axis x

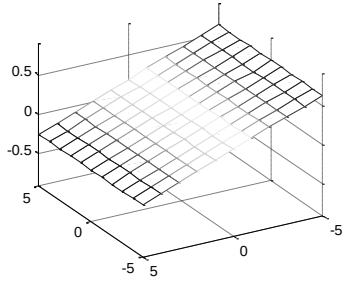

Fig 4. Sensor response ISSD5: axis y

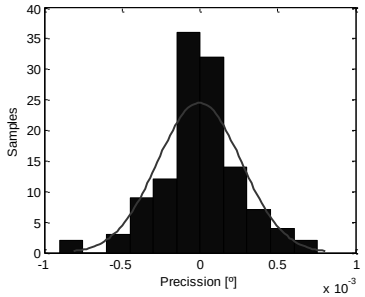

Fig 5. Statistics of accuracy ISSD5: axis x

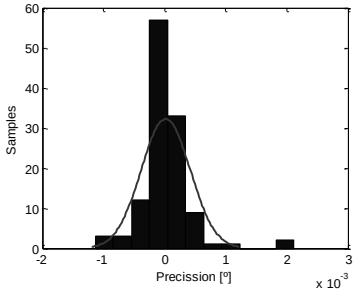

Fig 6. Statistics of accuracy ISSD5: axis y

7. Main operations

ISS-DX sensor measures the incidence angles of a solar radiation respect to its perpendicular. This information is provided through a RS485 UART channel (master-slave configuration).

7.1. ISS-DX parameters

7.1.1. Reference Axes

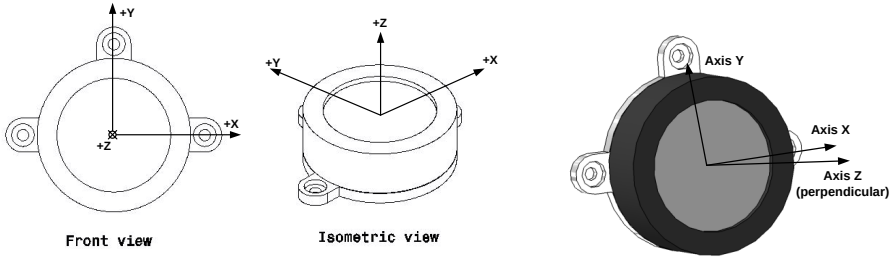

Fig 7. ISS-DX reference system

Z axis is perpendicular to the sensor base plane.

7.1.2. Angles

The *angle x and angle y* specify the angular position of the incident sun ray inside the field of view of the ISS DX sensor. The accuracy of the sensor increases close to zero degrees (perpendicular). Both angles are provided in degrees.

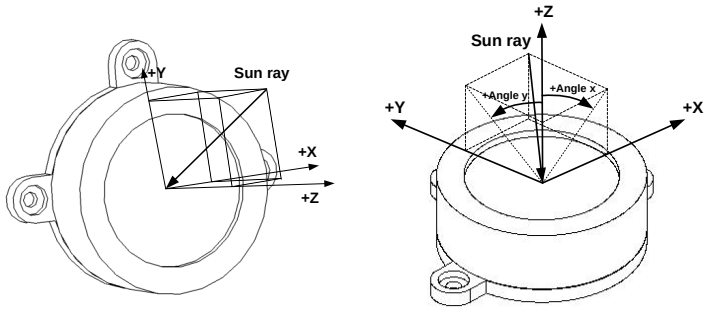

Fig 8. Reference for measured angles

The ISS-DX does an internal filtering processing to the angle measurements with the following features: third-order Butterworth filter with sampling frequency of 50 Hz and cutoff frequency of 0.4 Hz.

B.3 Honeywell HMC5983

3-Axis Digital Compass IC
HMC5983

Honeywell

Advanced Information

The Honeywell HMC5983 is a temperature compensated three-axis integrated circuit magnetometer. This surface-mount, multi-chip module is designed for low-field magnetic sensing for applications such as automotive and personal navigation, vehicle detection, and pointing.

The HMC5983 includes our state-of-the-art, high-resolution HMC118X series magnetoresistive sensors plus an ASIC containing amplification, automatic degaussing strap drivers, offset cancellation, and a 12-bit ADC that enables 1° to 2° compass heading accuracy. The I²C or SPI serial bus allows for easy interface. The HMC5983 is a 3.0x3.0x0.9mm surface mount 16-pin leadless chip carrier (LCC).

The HMC5983 utilizes Honeywell's Anisotropic Magnetoresistive (AMR) technology that provides advantages over other magnetic sensor technologies. Honeywell's anisotropic, directional sensors excel in linearity, low hysteresis, null output and scale factor stability over temperature, and with very low cross-axis sensitivity. These sensors' solid-state construction is designed to measure both the direction and the magnitude of magnetic fields, from milli-gauss to 8 gauss. Honeywell's Magnetic Sensors are among the most sensitive and reliable low-field sensors in the industry.

FEATURES	BENEFIT
3-Axis Magnetoresistive Sensors and ASIC in a 3.0x3.0x0.9mm LCC Surface Mount Package	Small size for Highly Integrated Products. Just add a micro-controller interface, plus two external SMT Capacitors designed for high volume, cost sensitive OEM Designs. Easy to assemble & compatible with High Speed SMT Assembly.
Temperature Compensated Data Output and Temperature Output	Automatically maintains sensor's sensitivity under wide operating temperature range
Automatic Offset Compensation	Maximizes sensor's full dynamic range and resolution
12-Bit ADC Coupled with Low Noise AMR Sensors Achieves 2 milli-gauss Field Resolution	Enables 1° to 2° degree compass heading accuracy
I²C (Standard, Fast, High-Speed modes) or SPI Digital Interface	High-speed interfaces for fast data communications. I²C up to 3.4 MHz and SPI up to 8.0 MHz
Fast 220 Hz Maximum Output Rate	Enables Pedestrian Navigation and LBS Applications
Built-in Self Test	Enables low-cost functionality test after assembly in production
Low Voltage Operations (2.16 to 3.6V) and Low Power Consumption (100 µA)	Compatible for battery powered applications
Built-In Strap Drive Circuits	Set/reset and offset strap drivers for degaussing, self test, and offset compensation. Eliminates sensor calibration necessary for other magnetic sensor technologies.
Lead Free Package Construction	RoHS Compliance
Wide Magnetic Field Range (+/-8 Oe)	Sensors can be used in strong magnetic field environments with a 1° to 2° degree compass heading accuracy
Software and Algorithm Support Available	Compassing Heading, Hard Iron, Soft Iron, and Auto Calibration libraries available

SPECIFICATIONS (* Tested and specified at 25°C except stated otherwise.)

Characteristics	Conditions*	Min	Typ	Max	Units
Power Supply					
Supply Voltage	VDD Referenced to AGND VDDIO Referenced to DGND	2.16 1.71	2.5 1.8	3.6 VDD+0.1	Volts Volts
Average Current Draw	Idle Mode Measurement Mode (7.5 Hz ODR; No measurement average, MA1:MA0 = 00) Specified at: VDD = 2.5V, VDDIO = 1.8V	- -	2 100	- -	µA µA
Performance					
Field Range	Full scale (FS)	-8		+8	gauss
Mag Dynamic Range	3-bit gain control	±1		±8	gauss
Sensitivity (Gain)	VDD=3.0V, GN=0 to 7, 12-bit ADC	230		1370	LSb/gauss
Digital Resolution	VDD=3.0V, GN=0 to 7, 1-LSb, 12-bit ADC	0.73		4.35	milli-gauss
Noise Floor (Field Resolution)	VDD=3.0V, GN=0, No measurement average, Standard Deviation 100 samples (See typical performance graphs below)		2		milli-gauss
Linearity	±2.0 gauss input range			0.1	±% FS
Hysteresis	±2.0 gauss input range		±25		ppm
Cross-Axis Sensitivity	Test Conditions: Cross field = 0.5 gauss, Happlied = ±3 gauss		±0.2%		%FS/gauss
Output Rate (ODR)	Continuous Measurment Mode Single Measurement Mode	0.75		220 160	Hz
Measurement Period	From receiving command to data ready		6		ms
Turn-on Time	Ready for I²C commands Analog Circuit Ready for Measurements		200 50		µs ms
Gain Tolerance	All gain/dynamic range settings		±5		%
I²C Address	8-bit read address 8-bit write address		0x3D 0x3C		hex
Clock Rate	Controlled by I²C /SPI Master I²C SPI			 3400 8000	kHz
I²C Hysteresis	Hysteresis of Schmitt trigger inputs on SCL and SDA - Fall (VDDIO=1.8V) Rise (VDDIO=1.8V)		 0.2*VDDIO 0.8*VDDIO		Volts
Self Test	X & Y Axes Z Axis		±1.16 ±1.08		gauss
	X & Y & Z Axes (GN=5) Positive Bias X & Y & Z Axes (GN=5) Negative Bias	243 -575		575 -243	LSb
Temperature Sensor Accuracy	3σ at T ≥ 0°C 3σ at T = -25°C 3σ at T = -40°C			7 11 14	°C
Sensitivity Tempco	T_A = -40 to 85°C, Compensated Output Temperature Sensor On		-0.03 (3σ =0.12)		%/°C

B.4 Cubespace Cubewheel Small

4. Mechanical Interface

This section shows the dimensions of the various sizes of CubeWheel units, as well as the mechanical definition of the mounting holes on the units.

4.1 Small CubeWheel

The Small CubeWheel unit has three sets of four M2 mounting holes on three different facets, as illustrated in Figure 7. The outer dimensions of the Small CubeWheel are 28 mm x 28 mm x 26.1 mm. Note that an additional 1.5 mm is required on one side of the CubeWheel for internal harnesses.

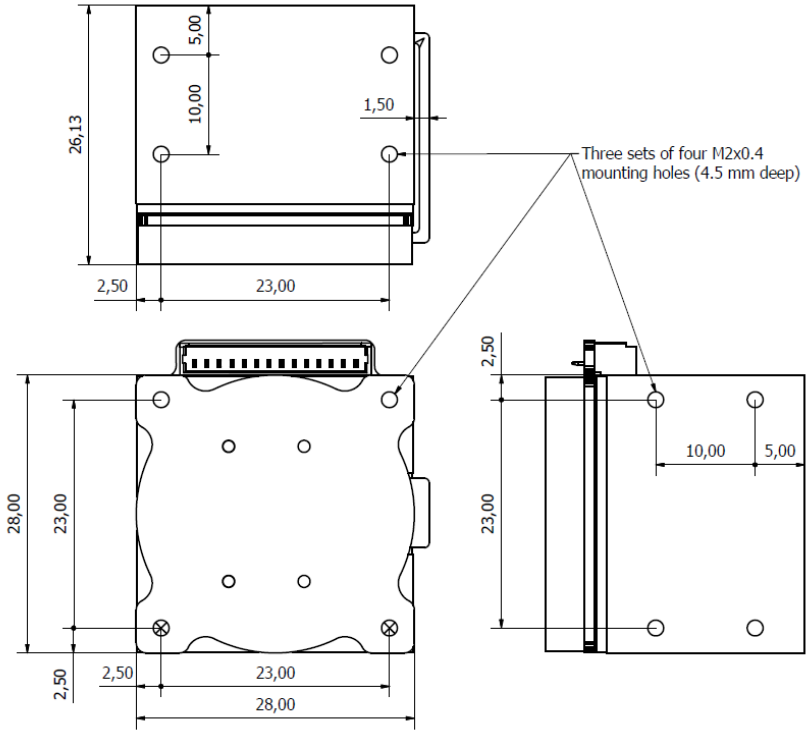

Figure 7 – Small CubeWheel mechanical interface

7. Specifications

The following tables lists the specifications of the various sizes of CubeWheels. Please note that all wheels draw 70mW of power from the 3V3 supply. The power usages listed in the tables are for the battery supply.

Table 7: CubeWheel Small Specifications

Supply voltage			
Digital supply voltage (3V3)	Min	Nom	Max
	3.0V	3.3V	3.5V
Power supply voltage (Battery)	Min	Nom	Max
	6.5V	N/A	16V

Supply power usage	**8V**	**12V**	**16V**
500 rpm	55mW (6.9mA)	84mW (7mA)	112mW (7mA)
2000 rpm	80mW (10mA)	120mW (10mA)	144mW (9mA)
4000 rpm	128mW (16mA)	168mW (14mA)	224mW (14mA)
6000 rpm	208mW (26mA)	264mW (22mA)	336mA (21mA)
Max acceleration	<512mW (64mA)	<480mW (40mA)	<464mW (29mA)

Momentum and speed				
Maximum momentum storage				1.77 mNms
Maximum wheel speed				± 8000 rpm
Torque constant				3.5 mNm/A
Maximum torque				0.23 mNm
Speed Control	**50rpm**	**500rpm**	**2000rpm**	**6000rpm**
Δ Speed(RMS)	±2rpm	±1rpm	±2rpm	±4rpm
Δ Speed(RMS) (Backup Mode)	-	±4rpm	±17rpm	±17rpm

Appendix C

Simulation models

Some subsystems of the simulation model are shown bellow. The simulation environment will be available after the thesis delivery in `https://gitlab.com/jnreves/thesis`.

C.1 Environment

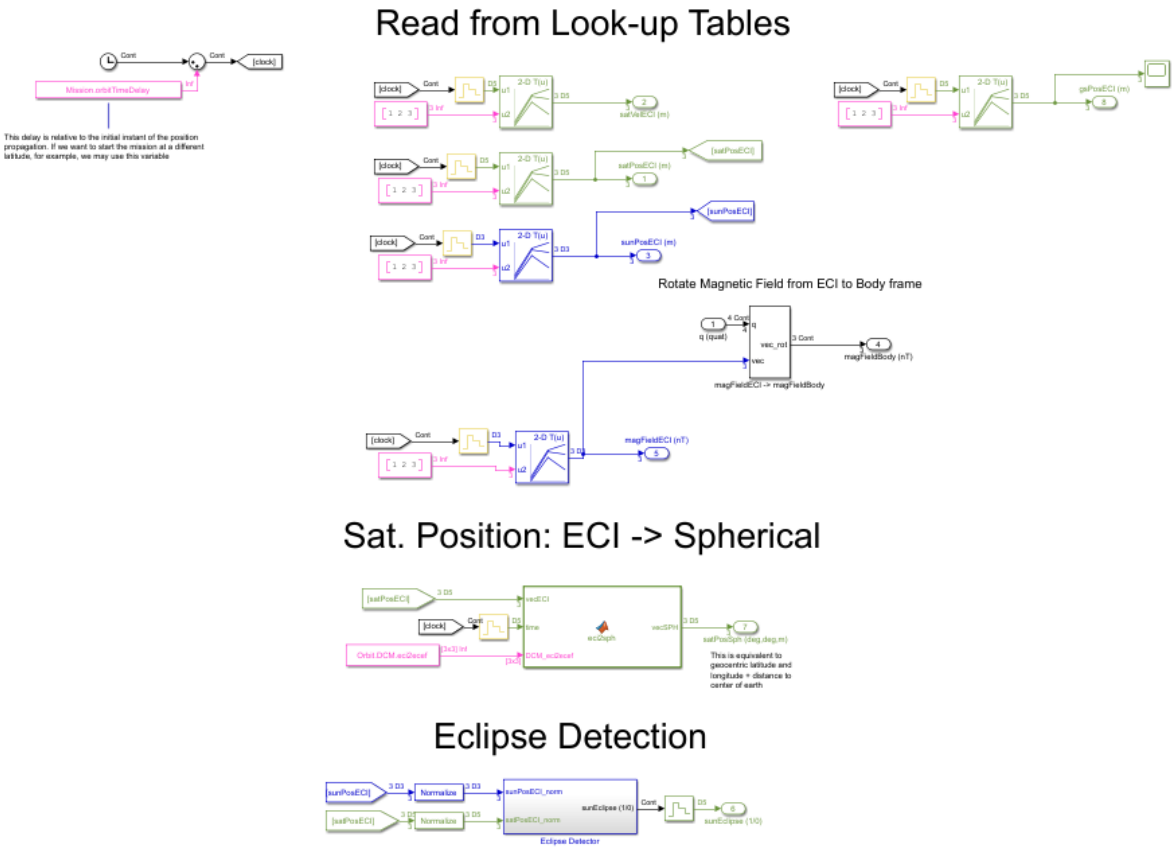

Figure C.1: Pre-processed data read and handled

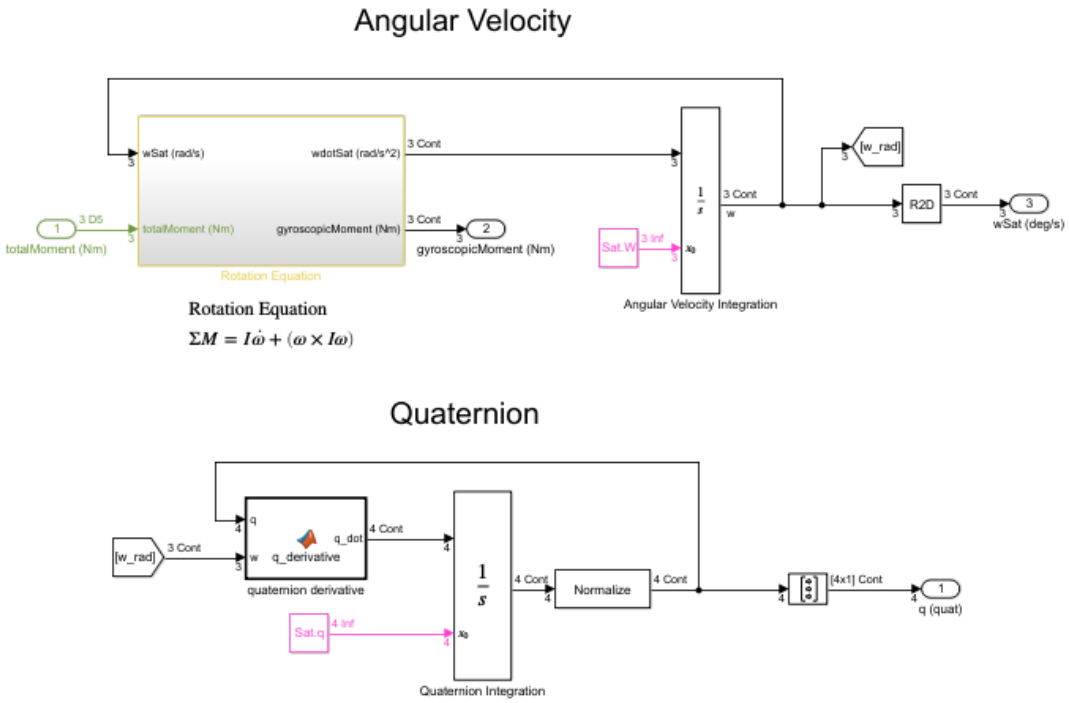

Figure C.2: Rotation dynamics

C.2 Sensors

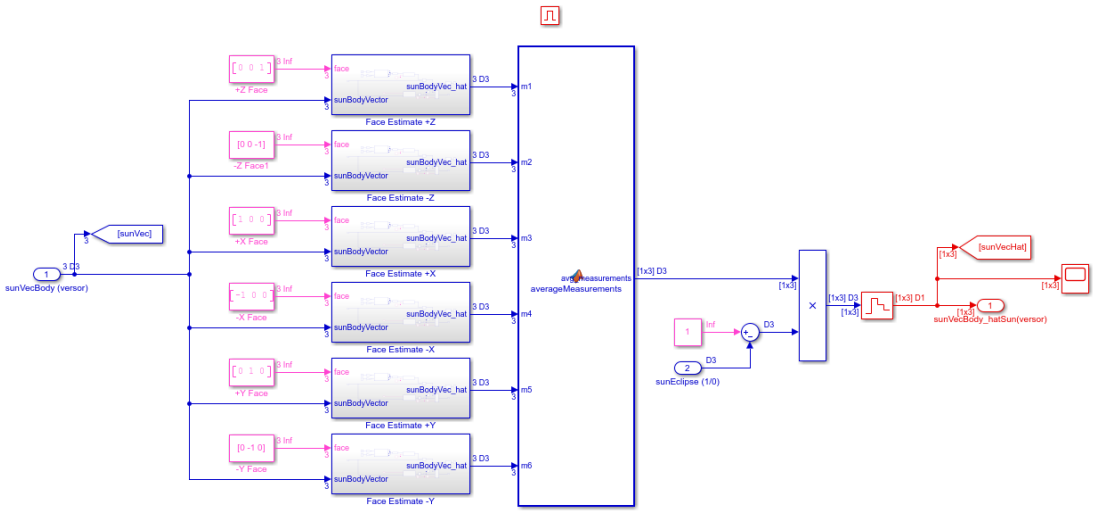

Figure C.3: Sun sensors overview

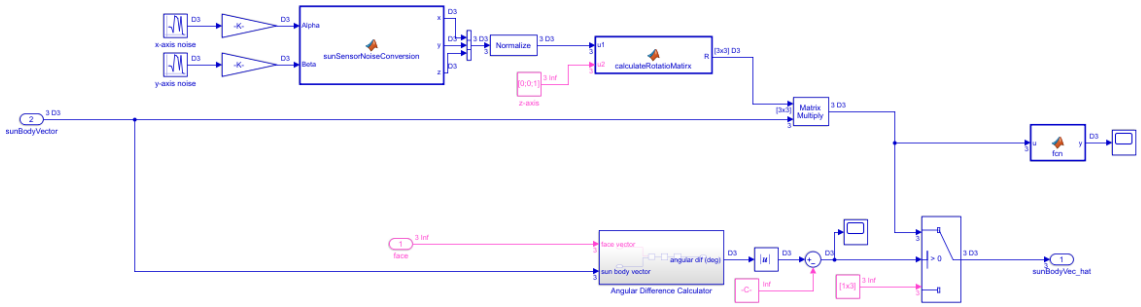

Figure C.4: Sun sensor model

C.3 Estimation

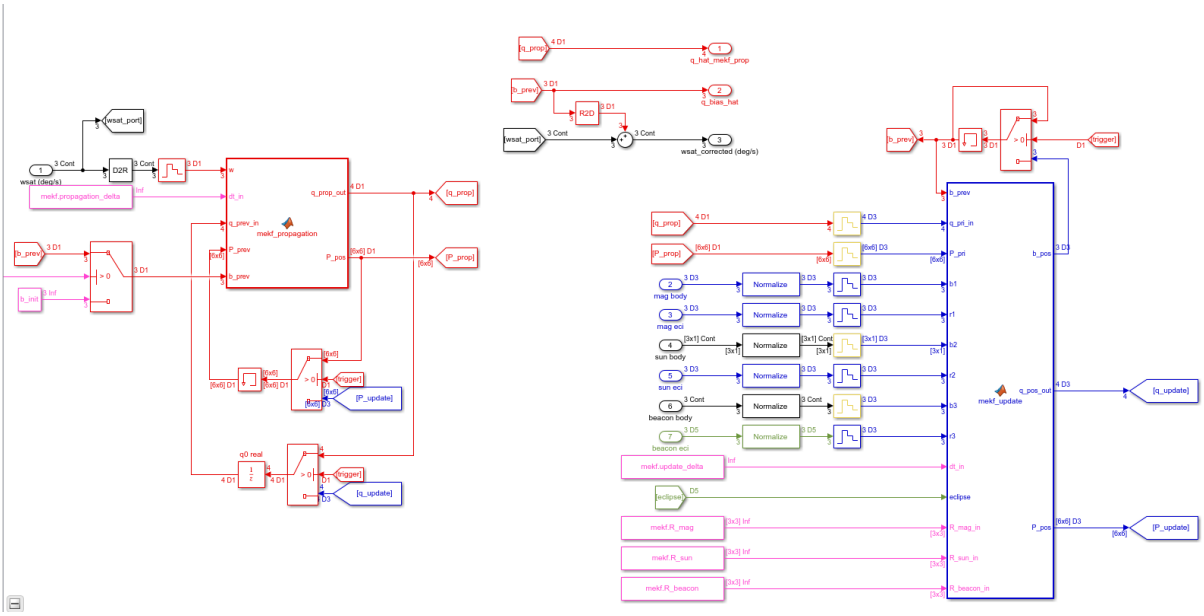

Figure C.5: MEKF architecture - propagation (left) and update (right)

C.4 Control

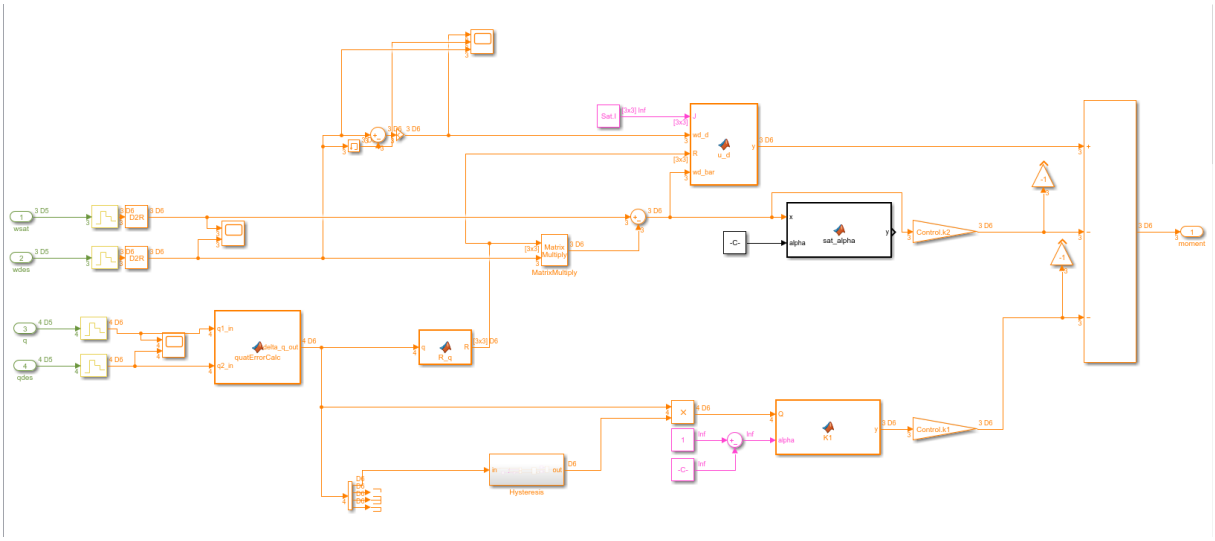

Figure C.6: GFTAC controller

C.5 Actuators

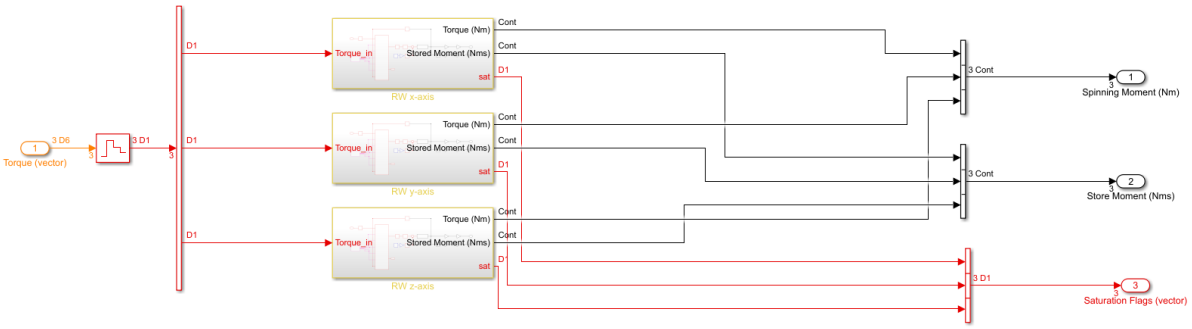

Figure C.7: Reaction wheels