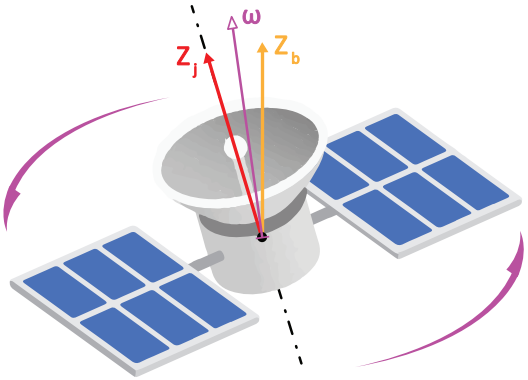
TÉCNICO LISBOA



Active Spin Stabilization for an Autonomous Spacecraft

Vítor Emanuel Andrade Narciso

Thesis to obtain the Master of Science Degree in

Aerospace Engineering

Supervisor(s): Prof. David Alexandre Cabecinhas
Dr. Pedro António Duarte Marques Lourenço

Examination Committee

Chairperson:	Prof. José Fernando Alves da Silva
Supervisor:	Prof. David Alexandre Cabecinhas
Member of the Committee:	Prof. Paulo Jorge Coelho Ramalho Oliveira

November 2022

To my dear Avó Manuela

Acknowledgments

During my studies and throughout the development of this master's thesis, I came across several people who helped me overcome many challenges and had a positive impact on my academic path. As such, I would like to acknowledge their importance in this brief section.

First and foremost, I wish to express my deepest gratitude to my supervisors, Professor David Cabecinhas and Doctor Pedro Lourenço, for their helpful insight and expertise, which were crucial for the research and writing of this document. I rejoice in all the support, criticism and wisdom shared that pushed me to deepen my thinking, enrich my knowledge, and elevate my work. It is with absolute conviction that I can conclude how much your guidance and advice were invaluable to endure this demanding task and complete my thesis.

I would also like to thank my family for withstanding this rough period with me. There are no words to describe my appreciation for your understanding, encouragement and unconditional love. To my parents António and Patrícia, I am grateful for your wise counsel, sympathetic ear and motivational support, and for providing me the opportunity to follow my dreams and study Aerospace Engineering. To my sisters Mariana and Marta, I cannot thank you enough for your friendship, comprehension, and the very welcome light-hearted distractions and amusing moments. Finally, to my grandmother Manuela, I am forever indebted to all your compassion, selflessness and never-ending concern. Your great tenderness and constant care for my well-being are more than I could have ever imagined and, for those reasons, you are the person to whom I dedicate this dissertation.

Furthermore, I wish to express my everlasting gratitude to my girlfriend Carolina. You have been my main pillar of strength during the last few years, and the backbone of my motivation to embrace this challenge. Your cheerful and supportive attitude, combined with your patience to tolerate my countless complaints and your many successful attempts to make me go outside and enjoy life beyond work were the reason why I could always keep myself calm, determined and focused while I faced this thesis and all the remaining obstacles during the last months. Thank you for everything from the bottom of my heart.

Last but not least, I would like to thank my friends and colleagues with whom I had the honor to share this long journey. To my closest friends Luís, Alice, Patrícia, João, Tiago and Mariana, thank you for the companionship, your kind advice and the entertaining moments accumulated over the last five years. I hope that you have collected innumerable beautiful memories as well from our time studying at Instituto Superior Técnico. To the fellow colleagues I had the pleasure to meet at AeroTéc, that helped me broaden my horizons and learn so much more about the most varied topics. And to you, the reader, who took some of your time to read this section and, hopefully, will also take advantage of the information gathered in this research study.

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.

Resumo

A estabilização passiva de atitude por rotação em torno de um eixo fixo no referencial do corpo é um método simples de manter o veículo espacial a apontar para uma determinada direção e reduzir o consumo de energia ou combustível, dado que um corpo rígido girando em torno do seu eixo principal de maior inércia encontra-se numa condição de rotação estável. De acordo com a álgebra linear, existe sempre um referencial centrado no corpo para o qual a matriz de inércia é diagonal, o que garante que os eixos associados ao maior e menor valores da diagonal correspondem aos eixos principais de maior e menor inércia, respetivamente. Contudo, a estimativa inicial dos principais momentos de inércia e respetivos eixos de referência pode estar incorreta, visto que é complicado determiná-los. Esta dissertação de mestrado aborda o problema de dimensionar um sistema ativo de estabilização por rotação, através do desenvolvimento de duas soluções de GNC que estabilizam a rotação do veículo em torno do eixo principal de maior inércia, mesmo quando este é incerto. Na primeira abordagem, um filtro de Kalman estendido é desenhado para estimar a rotação entre o referencial do corpo e os eixos principais de inércia, permitindo fornecer o vetor da velocidade angular devidamente alinhado como referência para o controlador. Alternativamente, uma lei de controlo não linear corrige a direção de rotação sem recorrer à complexidade do filtro. Estes algoritmos GNC são completados por um controlador de velocidade angular que garante a estabilidade global assimptótica para uma referência constante. As soluções propostas são testadas em simulações de Monte Carlo, resultando numa poupança de combustível média superior a 96%.

Palavras-chave: Estabilização por Rotação, Satélite, Filtro de Navegação, Filtro de Kalman Estendido, Controlador PI, Controlo Não Linear

Abstract

Single-spin passive attitude stabilization is a simple method of maintaining the spacecraft pointing in a certain direction and reduce power consumption, as a rigid body spinning about its principal major axis of inertia is in a stable rotation condition. According to linear algebra, there always exists a body reference frame where the inertia matrix is diagonal, which guarantees that the axes associated with the largest and smallest values of the diagonal correspond to the principal major and minor axes of inertia, respectively. However, the initial estimate of the principal moments of inertia and respective reference frame can be incorrect as they are notoriously difficult to determine. This master's dissertation addresses the problem of designing an active spin stabilization system, through the development of two GNC solutions that stabilize the spin motion of the vehicle about the principal major axis of inertia, even when it is uncertain. In the first approach, an extended Kalman filter is designed to estimate the rotation between the body-fixed frame and the principal axes of inertia, allowing then to provide the properly aligned angular velocity vector as a reference to the controller. As an alternative, a nonlinear control law is implemented to correct the spinning direction reference towards a principal axis avoiding the computational complexity of the filter. These GNC algorithms include an angular velocity controller to track the desired reference, with global asymptotic stability for a constant reference. The proposed solutions are extensively tested in Monte Carlo simulations, resulting in propellant savings over 96%.

Keywords: Spin Stabilization, Satellite, Navigation Filter, Extended Kalman Filter, PI Controller, Non-linear Control

Contents

List of Tables

List of Figures

Nomenclature

Greek symbols

α	Rotation angle.
$\alpha_{J_3\omega}$	Angle between the principal major axis of inertia and the angular velocity.
$\boldsymbol{\eta}$	Zero-mean Gaussian white-noise process.
$\boldsymbol{\Gamma}$	Discrete input matrix.
$\boldsymbol{\Lambda}$	Diagonal inertia matrix defined according to the principal axes of inertia.
$\boldsymbol{\mu}$	Magnetic dipole moment from the spacecraft.
$\boldsymbol{\omega_e}$	Angular velocity process error.
$\boldsymbol{\omega_I}$	Angular velocity of the spacecraft body frame.
$\boldsymbol{\omega_{\mathcal{B}}^{\mathcal{BI}}}$	Angular velocity of frame $\mathcal{B}$ with respect to frame $\mathcal{I}$, represented in frame $\mathcal{B}$.
$\boldsymbol{\Omega}$	Dynamics matrix.
$\boldsymbol{\omega}$	Angular velocity.
$\boldsymbol{\omega}_e$	Angular velocity error.
$\boldsymbol{\omega}_m$	Measured angular velocity.
$\boldsymbol{\omega}_r$	Angular velocity reference.
$\boldsymbol{\omega}_s$	Stationary angular velocity.
$\boldsymbol{\Phi}$	Discrete state matrix.
$\boldsymbol{\tau}$	External total torque acting on the body's center of mass.
$\boldsymbol{\tau}_a$	Aerodynamic torque acting on the body's center of mass.
$\boldsymbol{\tau}_d$	Disturbance torque acting on the body's center of mass.
$\boldsymbol{\tau}_m$	Magnetic torque acting on the body's center of mass.
$\boldsymbol{\tau}_n$	Control torque acting on the body's center of mass.

$\boldsymbol{\tau}_r$ Remaining disturbances torque acting on the body's center of mass.

$\boldsymbol{\tau}_s$ Solar radiation pressure torque acting on the body's center of mass.

$\boldsymbol{\tau}_{gg}$ Gravity-gradient torque acting on the body's center of mass.

$\boldsymbol{\tau}_{RW}$ Reaction wheel torque.

$\delta\boldsymbol{\omega}_r$ Variation of the angular velocity reference.

ϵ Small disturbance.

μ_E Earth gravitational parameter.

ν Orbit's true anomaly.

ω_n Angular velocity about n-axis.

ϕ Rotation angle about the $\boldsymbol{x}$-axis.

ρ Atmosphere density.

ρ_B Baumgarte stabilization parameter.

σ Standard deviation.

τ_n External total torque acting on the body's center of mass about n-axis.

θ Rotation angle about the $\boldsymbol{y}$-axis.

θ_{GMST} Greenwich Mean Sidereal Time (GMST) angle.

Roman symbols

$\boldsymbol{A_{\mathcal{BI}}}$ Attitude rotation matrix from $\mathcal{I}$ to $\mathcal{B}$ reference frame.

$\boldsymbol{A_{\mathcal{EI}}}$ Attitude rotation matrix from $\mathcal{I}$ to $\mathcal{E}$ reference frame.

$\boldsymbol{A}$ State matrix.

$\boldsymbol{B}$ Input matrix.

$\boldsymbol{b}$ Earth's magnetic field in the body reference frame.

$\mathcal{B}$ Spacecraft body reference frame.

$\boldsymbol{C}$ Observability matrix.

C_D Drag coefficient.

$\boldsymbol{D}$ Direct transmission matrix.

e Process error.

$\boldsymbol{F}$ Jacobian matrix for the nonlinear system evaluated at the current state estimate.

$\boldsymbol{f}$ Differentiable function.

$\boldsymbol{f}_a$ Aerodynamic force.

$\boldsymbol{h}$ Observation vector.

$\boldsymbol{h}$ Spacecraft angular momentum about its center of mass.

$\boldsymbol{I_n}$ Identity matrix $n \times n$.

$\boldsymbol{I}$ Identity matrix.

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

i Orbit's inclination.

$\boldsymbol{J}$ Moment of inertia tensor.

$\boldsymbol{J}_{ctr}$ Diagonal estimate for the inertia tensor provided to the controller.

J_n Moment of inertia tensor about n-axis.

J_x Principal moment of inertia about x-axis.

J_y Principal moment of inertia about y-axis.

J_z Principal moment of inertia about z-axis.

$\boldsymbol{K}$ Kalman filter gain.

$\boldsymbol{K}_i$ Integral controller gain.

$\boldsymbol{K}_p$ Proportional controller gain.

K Reflectivity.

m_c Spacecraft mass.

m_E Earth's mass.

$\boldsymbol{N}$ Unit vector cross-product matrix representation.

$\boldsymbol{n}$ Unit vector.

$\boldsymbol{P}$ Estimate error covariance matrix.

$\boldsymbol{P}_s$ Solar radiation pressure.

$\boldsymbol{Q_c}$ Noise covariance matrix associated to the prediction.

$\boldsymbol{q}$ Quaternion.

$\boldsymbol{R_c}$ Noise covariance matrix associated to the measurement.

$\boldsymbol{R}$ Euler angles' rotation matrix.

$\boldsymbol{r}$ Moment arm vector.

$\boldsymbol{r}_0$ Circular horizontal orbit radius in the inertial reference frame.

$\boldsymbol{r}_c$ Circular orbit radius in the body reference frame.

$\boldsymbol{r}_I$ Circular orbit radius in the inertial reference frame.

$\boldsymbol{R}_n$ Euler angles' rotation matrix about n-axis.

r Reference value.

r_c Spacecraft circular orbit radius.

r_E Earth's radius.

S Cross section area.

t Time.

T_{orb} Spacecract circular orbital period.

$\boldsymbol{u}$ Inputs vector.

$\boldsymbol{V}$ Lyapunov function.

$\boldsymbol{V}$ Spacecraft velocity.

v Zero-mean white-noise associated to the measurement.

w Zero-mean white-noise associated to the prediction.

$\boldsymbol{x}$ State vector.

Subscripts

0 Initial value.

ctr Control value.

est Estimated value.

k Variable evaluated at time k.

Superscripts

$+$ Updated estimate.

$-$ Propagated estimate.

T Transpose.

Glossary

ACS	Attitude Control System
ADCS	Attitude Determination and Control System
ADS	Attitude Determination System
AEKF	Additive Extended Kalman Filter
AIM	Asteroid Impact Mission
AOCS	Attitude and Orbit Control System
AUV	Autonomous Underwater Vehicle
CMG	Control Moment Gyro
CM	Center of Mass
CP	Center of Pressure
DCM	Direction Cosine Matrix
ECEF	Earth Centered Earth Fixed
ECI	Earth Centered Inertial
EKF	Extended Kalman Filter
ESA	European Space Agency
ESOQ	Estimator of the Optimal Quaternion
FOAM	Fast Optimal Attitude Matrix
FOG	Fiber Optic Gyro
FOV	Field of View
GAS	Global Asymptotic Stability
GMST	Greenwich Mean Sidereal Time
GNC	Guidance, Navigation and Control
ISS	International Space Station
KF	Kalman Filter
LEO	Low Earth Orbit
LQR	Linear Quadratic Regulator
LVLH	Local Vertical Local Horizontal
MEKF	Multiplicative Extended Kalman Filter
MEMS	Microelectromechanical Systems
MPC	Model Predictive Control

MarCO	Mars Cube One
NASA	National Aeronautics and Space Administration
ODE	Ordinary Differential Equation
PID	Proportional-Integral-Derivative
PI	Proportional-Integral
PSD	Positive Semi-Definite
QUEST	Quaternion Estimator
RLG	Ring Laser Gyro
RMSE	Root-Mean-Square Error
RW	Reaction Wheel
SMC	Sliding Mode Control
SRP	Solar Radiation Pressure
SVD	Singular Value Decomposition
TRIAD	Triaxial Attitude Determination
UAV	Unmanned Aerial Vehicle
UF	Unscented Filter
UKF	Unscented Kalman Filter

Chapter 1

Introduction

The Guidance, Navigation and Control (GNC) engineering systems are designed to manipulate the movement of vehicles, and can be found in essentially all autonomous or semi-autonomous systems, including driver-less automobiles, autonomous underwater vehicles (AUVs), unmanned aerial vehicles (UAVs) and spacecraft launch vehicles.

Guidance is the process of collecting and applying information for the purpose of generating maneuver commands to control vehicle movements, i.e., it refers to the determination of the necessary changes in velocity, rotation and acceleration, according to the desired trajectory from the vehicle's current location to a designated target [1]. Navigation refers to the determination of the vehicle's position, velocity, acceleration and attitude relative to a certain reference frame (i.e., its *state vector*) at a given time [2]. Control is the capacity to manipulate the forces and torques, by means of actuators (steering controls, thrusters, brakes, etc), required to bring the vehicle to the desired position, velocity, acceleration and attitude, while maintaining stability.

For spacecraft and satellite applications, the GNC system of a space vehicle is concerned with the orbital motion (trajectory) of the spacecraft's center of mass. The orbital GNC system is an area of space technology that plays a key role in the success of space missions that involve rendezvous, docking, and proximity operations [3]. On the other hand, an Attitude Determination and Control System (ADCS), which consists of an attitude determination system (ADS) and an attitude control system (ACS), should also be developed as a navigation solution to provide stabilization and control of the attitude (orientation) of a spacecraft using a variety of sensors and actuators in the presence of disturbance torques [3].

In the domain of space missions, the on-board ADCS/GNC systems are crucial in order for satellites or space vehicles to accomplish their missions. The tasks set for many spacecraft require them to maintain a specific absolute or relative pointing, position, velocity, acceleration and attitude (within acceptable errors), which can only be provided accurately with precise ADCS and GNC subsystems. For orbit control, the original ADCS/GNC systems can be extended to include an Attitude and Orbit Control System (AOCS). A typical complete GNC architecture is illustrated in Figure 1.1.

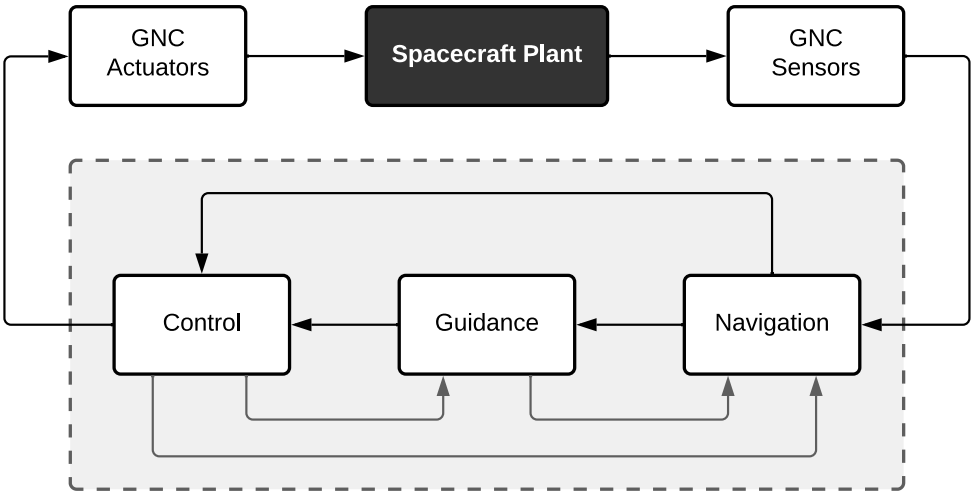


Figure 1.1: Spacecraft typical complete GNC architecture scheme

Presently, and for the last decades, the trend in satellites has been to do more for less cost, therefore, latest space missions tend to be cheaper and smaller, which led to the decrease of spacecraft sizes [4, 5]. The segments of nanosatellites (concerning masses ranging from 1 to $10\,\mathrm{kg}$ [6]) and, more specifically, CubeSats ($10\,\mathrm{cm}^3$ nanosatellites with a mass of up to $1.33\,\mathrm{kg}$) has gained popularity due to their reduced cost, development time and increased accessibility. Although education and technology demonstration are the primary objects in most CubeSat missions, some applications related with science domains have also appeared, with examples such as NASA's InSight/MarCO mission or ESA's AIM [7].

The decrease in satellite dimensions was accompanied by strict mass, volume, power and cost constraints, and taking into account the fact that the AOCS is an important subsystem for most spacecraft, control systems engineers should find low-cost and low-power solutions for AOCS subsystems [3, 6]. Hence, to avoid unnecessary energy usage, the ADCS subsystem can be composed of both active and passive stabilization schemes. The former require constant power and fuel consumption through active magnetic (low precision), momentum-based (medium precision) or reaction-wheels-based (high precision) attitude control. Conversely, the latter helps save propellant mass and reduce power usage taking advantage of physics and dynamics properties of the spacecraft and its surrounding environment, such as, spin stabilization, gravity gradient stabilization or passive magnetic stabilization [4].

This work focuses on developing a navigation and attitude control system for satellites dedicated to orbital phases in stabilized rotation (spin stabilized) in the context of space missions in projects of the European Space Agency (ESA). The reasons behind this choice and a brief description of what solving this problem entails are presented in Section 1.1. Then, the state of the art methods currently in use are reviewed during Section 1.2, and the contributions that this study provides to the scientific community are included in Section 1.3. Finally, the Section 1.4 enumerates the main objectives of this research, while an overview of the full document's structure is presented in Section 1.5.

1.1 Motivation

The GNC and AOCS systems are a continuous object of study in the scientific community, as well as in the space industry. Despite this, and given the conservative nature of the industry, many of the tools used are already classic and out of date with the state of the art both in terms of control theory and estimation systems, as well as current computing capabilities. For this reason, there are still huge engineering challenges associated with these two fundamental systems.

Spin stabilization is a method used to maintain the pointing direction of a spacecraft, while rotating around its own pointing axis. This is achieved using the gyroscopic effect - a principle that states, as explained and theoretically proved later in Section 3.3, that a rigid body spinning about its major or minor axis of inertia is in a stable rotation condition, i.e., it will keep the direction of its spinning axis fixed with respect to the inertial space reference frame, even in the presence of small disturbances [5, 8].

Single-spin attitude stabilization is a simple and passive way of keeping the spacecraft pointing in a certain direction. The dynamics and control of a spin stabilized spacecraft have been extensively researched, including NASA projects such as Pioneer 10 and 11 spacecraft, the Lunar Prospector and the Galileo Jupiter orbiter. Still, it has some crucial drawbacks with respect to communication efficiency, as it cannot be used to re-point the antennas or optical instruments [5, 8].

However, the main concern when tackling spin stabilized satellites is to ensure that the passive scheme method does not incur additional energy costs - it is indeed passive. To accomplish so, as stated before, the spacecraft must be rotating around its major or minor axis of inertia, in order to guarantee stability and avoid fuel consumption to thrust and keep the satellite spinning around a non-stable axis.

But how does one know a spacecraft's major and minor axis of inertia? According to linear algebra, there always exists a body reference frame where the inertia matrix is diagonal, which guarantees that the axes associated with the largest and smallest values of the diagonal correspond to the principal major and minor axes of inertia, respectively. However, the initial estimate of the principal axes of inertia and respective reference frame can be incorrect as they are notoriously difficult to determine and, as time passes, the satellite may change its mass and/or geometry, for instance opening solar panels or consuming fuel, causing the appearance of terms out of the main diagonal of the inertia tensor.

This situation leads to two possible scenarios. In the first case, the GNC system can ignore these displacements and neglect the existence of undesirable products of inertia, which causes an oscillatory condition as the satellite will be instructed to spin around a different axis of inertia, and also incurs in expenditure of propellant to keep the satellite spinning in the desired direction. On the other hand, the GNC system can estimate the angles between the current body reference frame and the coordinate system where the inertia matrix is diagonal and, therefore, guide the satellite to such attitude where the products of inertia will be truly negligible and the spacecraft will be spin stabilized, with minimal additional power consumption, around the real principal major or minor axes of inertia.

In summary, this dissertation addresses the problem of designing an active spin stabilization system, i.e., a GNC solution that provides the proper angular velocity vector as a reference to the controller, which is necessary to rotate the spacecraft and align its spin axis with the principal major inertia axis.

1.2 State of the Art

Attitude estimation and control is a problem with an extensive historical background that continues to attract intensive research, and several techniques to solve the inertia matrix estimation problem regarding spin stabilized satellites have been proposed in the literature. Authors such as J. R. Wertz, M.J. Sidi, Y. Yang, F. L. Markley, and J. L. Crassidis have published many articles and books exploring this subject [8–11], which will be taken into account during the Theoretical Background exposition in Chapter 2.

The first satellites, such as the Sputnik in 1957, did not have an ADCS, or relied on passive stabilization methods, while control torques were computed on the ground and transmitted to the spacecraft [11]. Some of the earliest proposals to address this problem were provided by algebraic solutions based on real-time measurements. One of the pioneering methods for a 3-axis attitude determination system was invented by Harold Black for the Transit satellite system and published in 1964. This method came to be known as the Triaxial Attitude Determination (TRIAD) algorithm, and made use of two and only two vector observations [12].

To overcome the limitations of the TRIAD algorithm, one year later, Grace Wahba formulated a general criterion for attitude determination involving any number of vector measurements [13]. The first successful applications of Wahba's problem was the Davenport's q-method algorithm, published in 1977, which provided a solution using the quaternion parameterization for the attitude [14]. In the following year, Shuster developed the Quaternion Estimator (QUEST) algorithm as an alternative to the q-method which avoided the eigenvalue decomposition and overcoming the computational power requirements of the former method [15, 16].

Later, other methods and algorithms came about to offer different solutions to Wahba's problem, such as the Estimator of the Optimal Quaternion (ESOQ), the Singular Value Decomposition (SVD) method and the Fast Optimal Attitude Matrix (FOAM) algorithm, which provided solutions in matrix form [11, 17]. These attitude determination methods mentioned above are known as the static attitude determination methods, since they combine two or more vector measurements taken at the same time; reference [17] provides a valuable summary of all those static methods, as well as a comparison between them.

Most recent solutions try to find the best estimate of the true attitude using dynamical models and measurements (both corrupted by noise and uncertainties), i.e., they retain information from a series of measurements taken over time. These type of approaches are called recursive attitude determination methods and have three main advantages over the aforementioned methods. First, the inclusion of memory from previous measurements provides more attitude accurate solutions. Second, they are able to provide estimates even when less than two measurements are available, which would be insufficient for the static attitude determination methods. Third, they can be used to simultaneously estimate the attitude and filter out noise and uncertainties [18].

One of the most popular recursive algorithms for attitude estimation is the Kalman Filter (KF), originally developed as a tool for linear estimation [18–20]. However, since the dynamics and measurements models involved are nonlinear, an extension of the KF is used to address nonlinear estimation problems - the Extended Kalman Filter (EKF). Both versions of the filter will be further discussed in Section 4.1.

The first public application of Kalman filtering to spacecraft attitude determination dates back to 1967, where James Farrell used an EKF to determine the attitude of a spacecraft for NASA's Apollo missions while establishing Euler angles as the attitude parameterization [21]. Still, in the mid-1970's, the quaternion parameterization was gaining popularity amongst the scientific community and, as the computational burden to do on-board real-time estimation decreased, more quaternion-based attitude estimation filters emerged, cementing the EKF as the "workhorse" of spacecraft attitude determination [18, 22].

For the last years, the EKF has been the prime choice in ADCS designs for CubeSats, becoming the most used method for spacecraft attitude determination. Although EKF solutions may come in several forms, such as the Additive EKF (AEKF), the Multiplicative EKF (MEKF) [23], and being a very powerful estimation technique, this filters do not guarantee optimality, stability or convergence. To overcome the poor performance or even divergence arising from the linearization implicit in the EKF, other nonlinear attitude estimation methods have been arising. Some examples are based on static attitude determination methods, such as the filter QUEST or the recursive QUEST, although they do not have an improved performance in comparison to the EKF [24, 25].

Other alternatives for attitude estimation are the Unscented Filter (UF) (exhibiting a lower expected error than the standard EKF) [26], the two-step optimal estimator (predictive and adaptive approaches) [27, 28], particle filters, and nonlinear observers. The latter method emerged recently as an attractive alternative given the possibility to establish convergence bounds and provide stability, since they often feature global stability proofs, i.e., they can converge from any initial condition. Although they are a relatively new topic in attitude estimation, the associated lower computational cost, and a more straightforward tuning of the design parameter, caused the nonlinear observers to already be considered for attitude estimation on CubeSats. Reference [18] provides an excellent review of several of the nonlinear attitude estimation methods discussed above.

While spacecraft attitude estimation development history is well documented, attitude control methods past is known to a lesser extent, partially due to the conservative nature of the space industry, where most documents were deemed classified by governments during the early days of spaceflight. Robert Roberson [29] claims that the first published works regarding spacecraft attitude control appeared during the mid 1950s. At the time, satellites were simply spheres without any kind of pointing requirements, while other spacecraft used passive control methods, i.e., spin stabilization, gravity-gradient stabilization or a momentum wheel providing a constant angular momentum bias [11].

As pointing requirements became more demanding and on-board computers have become more capable, spacecraft ACS designs have shifted towards active control methods, namely three-axis stabilization. Most actively-controlled satellites use thrusters, reaction wheels (RWs) or control moment gyros (CMGs) as their main actuators. For CubeSats and other resource-limited spacecraft, magnetorquers are often the primary actuators for attitude control due to their simplicity and low cost [11]. These actuators are commanded by a control method, the most common in spacecraft attitude control being the Proportional-Integral-Derivative (PID) controller and the Linear Quadratic Regulator (LQR).

Nowadays, still about 95% of aerospace industrial applications continue to be operated by PID controllers, just as they were back in the 1960s period. PID control action is based on past, present and prediction of the future relative errors between the desired position and the actual position of the spacecraft, obtained through a closed-loop transfer function. The proportional term acts on the rise time, but cannot completely eliminate the steady-state error. On the other hand, the integrative term gets rid of the steady-state error, but worsens the transitory response. At last, the derivative term decreases the overshoot, improving the transitory term. Since this work does not call for elaborate control requirements, a standard PI controller is the elected alternative, as described in Section 4.2.

However, as simple and practical as the PID controller can be to implement (as it relies on few parameters and offers some robustness), optimal control can lead to more efficient results, as it operates a dynamic system at minimum cost. One of the main techniques applied to space vehicles is the LQR, whose main goal is to find the control input that minimizes the cost function, an operational index that governs the trade-off between expended energy and performance of the closed-loop system.

In the last years, due to the digital revolution and exponential growth in computational power, modern control designs have been developed and adapted for the space industry to deal with the increasing parametric and non-parametric uncertainty present in the complex space missions underdevelopment, such as rendezvous orbital maneuvers and interplanetary spaceflights [30, 31]. These fault-tolerant controller designs include the so-called Robust Control methods, such as H-infinity, Linear Parameter-Varying (LPV) control, and Sliding Mode Control (SMC). The latter is a well-established control method for nonlinear systems, which is characterized by solid theoretical foundations and provides good robustness to the system. A sliding surface is defined, representing the desired trajectory and, once a feedback control law manages to bring the spacecraft trajectory to cross the sliding surface, the satellite's state motion will always be directed towards it - a "sliding mode" motion is enforced. More recently, Model Predictive Control (MPC) alternatives are being studied, as they can predict a dynamical model likely future and define a suitable control action for it. Nevertheless, it should be pointed out that MPC have an increased computational load when compared to the previously reviewed control methods, which can be a hassle even for current high-tech computational platforms.

Regarding inertia matrix estimation, as stated along the attitude estimation history exposure, the first satellites already relied on passive stabilization schemes like spin stabilized spacecraft. Such fact is indeed not surprising, since the concepts of moment of inertia and spin stabilization go far back in time. In 1673, Christian Huygens established the former parameter in his study of the oscillation of a body hanging from a pivot, known as a compound pendulum. However, the term "moment of inertia" was only introduced in 1765 by Leonhard Euler [32], and incorporated into Euler's second law. This formulation can be rewritten and adapted to form the Euler's moment equations, as later described in Section 2.3.

In the aerospace industry, moment of inertia theory was applied for the first time when the concept of spin stabilization was invented. In the middle of the 19th century, William Hale came up with a method of successfully eliminating the dead-weight of the flight-stabilizing guide stick. He designed jet vents at an angle, causing the rocket to spin, much like a bullet, and making them more stable and accurate, which represented a major improvement in performance and ease of handling.

Although those were huge steps in spacecraft stabilization, presently, the space industry relies more on the development of active stabilization schemes. However, one problem that still remains contemporary for passive stabilization schemes is the inertia matrix estimation. As described before, if a satellite's moments of inertia change over time, the once passive spin stabilization method becomes active and incurs in power consumption. According to the literature, in-flight estimates of the inertia tensor have been researched since 2002. The first control system that addressed inertia estimation used a least squares minimization problem [33]. Later, the constrained least squares [34] and EKF-based methods [35] were applied for inertia matrix estimation [34], to improve the accuracy and performance of the estimator. More recently, bolder approaches have emerged such as the implementation of an Unscented Kalman Filter (UKF) to process the light-curve and angles from on-board astronomic photographies and estimate orientation, position, velocity and inertia parameters [36], or even the design of a semi-adaptive filter based on the linear regression model, in order to achieve low variance estimates [37].

1.3 Contributions

In this dissertation, several different concepts, definitions and notations are presented. The first chapter introduces attitude representations, kinematics, dynamics, and other theory. Hence, it constitutes a practical guide for those uninitiated on spacecraft attitude determination and control.

On the other hand, all the contributions made to the research topic are incorporated in the design and implementation of the proposed GNC solution, which promotes a significant decrease in fuel consumption and, consequently, a remarkable improvement for spacecraft control system power efficiency. In addition, the angular velocity controller is theoretically proved to be globally asymptotically stable for a constant reference, which is an important step to validate this control system.

At last, the control-based guidance algorithm should be highlighted, as it proposes a novel simple and elegant approach to the initial problem. In this solution, dynamic variations are added to the initial reference value, with the objective of converging to the angular velocity reference obtained from the dynamics equation. Therefore, a non-linear control law is defined, which reduces the computational load of the control system, as it no longer relies on a navigation filter to achieve the desired results.

1.4 Dissertation Objective

The main goal of this study is to design, implement and test an active spin stabilization system, i.e, a GNC solution that aligns the spinning axis of the satellite to its principal major axis of inertia. The active spin stabilization solution can also be known as the Control System. Therefore, the spacecraft is guided to a spin stabilized condition that should drastically reduce fuel consumption.

To run the simulation tests, this Control System must be implemented in a realistic space environment that also includes the Spacecraft Plant model. Then, the obtained results from different algorithms are analyzed and compared in order to draw conclusions and determine future improvements.

1.5 Dissertation Outline

This dissertation is organized as follows:

Chapter 1 presents the motivation and objectives of the dissertation, along with the study and review of the existing state of the art related to the contents and topics covered in this study.

Chapter 2 provides an overview of the required theoretical background concepts for the current research, namely several attitude representations and frames of reference used throughout the work. Furthermore, the attitude dynamics and kinematics are introduced, including the external disturbances present in the space environment. Lastly, this Chapter also reviews and compares a number of actuators and sensors available.

Chapter 3 documents the required theory underlying the problem at hand, comprising inertia tensor definition, rotation matrix computation and with a concise description of the spin stability along the principal axes of inertia.

Chapter 4 introduces the design of an active spin stabilization system, including the navigation filter, guidance algorithms and control law used to achieve the proposed main objective. Lyapunov's direct method to evaluate the Control System's stability is also reviewed in this Chapter.

Chapter 5 describes the development of a simulation environment to test the Control System applied to a generic Spacecraft Plant, and implemented with a MATLAB®/Simulink® model. In addition, all the parameters and gains used in the performed simulations are also listed in a table.

Chapter 6 discusses and analyses the simulation case scenarios tested for both guidance algorithms, providing the initial setup and the obtained results, and later establishing a final comparison between the different guidance methods.

Chapter 7 sums up the developed study with a conclusion, and proposes some ideas for future work.

Chapter 2

Theoretical Background

Spacecraft attitude can be defined as its orientation in space with respect to a specific reference frame. For most space applications, attitude knowledge is crucial, since the motion of a rigid body is determined not only by its position and velocity, but also by its rotation motion around its center of mass. The area of attitude determination and control systems (ADCS) consists fundamentally in the study of methods for estimating the relative spacecraft orientation and design of algorithms for attitude control.

This Chapter provides an overview of the underlying theory about the topic of spacecraft attitude determination and control, and some other mathematical concepts that must be outlined in order to formulate the main problem that the current dissertation aims to solve.

Initially, the most relevant attitude representations are presented in Section 2.1. Since some representations allow for multiple possible definitions, namely the Euler angles and the quaternions, the adopted conventions are defined. In Section 2.2, an exposition of the reference frames used throughout the work is carried out. The topic of attitude dynamics and kinematics is briefly reviewed in Section 2.3, including the enumeration of several external disturbances. Finally, the Section 2.4 provides a terse summary of the many sensors and actuators available, highlighting the applied solutions.

Before addressing this Chapter, it is suggested to read Appendix A, as it presents a review of some mathematical notions and conventions adopted in this work.

2.1 Attitude Representations

In geometry, it is mathematically possible to represent a rotation in three dimensions using a number of different formalisms. This concept is applied to classical mechanics regarding the angular kinematics, where the orientation of a rigid body at a given instant is described with the same tools. Considering $\mathcal{B} = \{b_1, b_2, b_3\}$, an orthogonal, right-handed basis whose axis and origin are fixed in the spacecraft body frame and the inertial reference frame $\mathcal{I} = \{i_1, i_2, i_3\}$, also orthogonal and right-handed, the spacecraft attitude is the orientation of the first basis $\mathcal{B}$, with respect to the reference system $\mathcal{I}$.

The various forms to represent this orientation, including matrices and three-component vectors, are the so-called attitude representations or parameterizations. This section covers the formalisms used in

this study, discussing overall advantages and disadvantages based on references [10, 11, 38, 39].

2.1.1 Direction Cosine Matrix (DCM)

The most general form of attitude representation is the direction cosine matrix (DCM) (or rotation matrix), a matrix whose multiplication with a vector results in its rotation while preserving its length. For an arbitrary vector $\boldsymbol{r}$ considered in the inertial reference frame $\mathcal{I}$, $\boldsymbol{r}_{\mathcal{I}}$, and to represent it in the body reference frame $\mathcal{B}$, $\boldsymbol{r}_{\mathcal{B}}$, a DCM is defined as giving the relative orientation of frame $\mathcal{B}$ w.r.t. frame $\mathcal{I}$, rotating vector $\boldsymbol{r}$ according to

$$\boldsymbol{r}_{\mathcal{B}} = \boldsymbol{A}_{\mathcal{BI}} \boldsymbol{r}_{\mathcal{I}} \tag{2.1}$$

where $\boldsymbol{A}_{\mathcal{BI}}$ represents the rotation matrix from $\mathcal{I}$ to $\mathcal{B}$.

To form this 3×3 matrix, nine parameters are required to completely describe the relative orientation between the two coordinate frames unambiguously. Considering the basis vectors for both reference frames, defined as $\{b_1, b_2, b_3\}$ and $\{i_1, i_2, i_3\}$, the rotation matrix can be algebraically given by:

$$\boldsymbol{A}_{\mathcal{BI}} = \begin{bmatrix} b_1 \cdot i_1 & b_1 \cdot i_2 & b_1 \cdot i_3 \\ b_2 \cdot i_1 & b_2 \cdot i_2 & b_2 \cdot i_3 \\ b_3 \cdot i_1 & b_3 \cdot i_2 & b_3 \cdot i_3 \end{bmatrix} \tag{2.2}$$

As it is a DCM, $\boldsymbol{A}_{\mathcal{BI}}$ is a proper real orthogonal rotation matrix that belongs to the *special orthogonal group* denoted by $SO(3)$, and defined as

$$SO(3) := \{\boldsymbol{A} \in \mathbb{R}^{3\times 3} : \det(\boldsymbol{A}) = 1 \wedge \boldsymbol{A}\boldsymbol{A}^T = \boldsymbol{A}^T\boldsymbol{A} = \boldsymbol{I}\} \tag{2.3}$$

Regarding to the attitude kinematics in rotation matrix representation, it is described as the relation between the angular velocity $\boldsymbol{\omega} \in \mathbb{R}^3$ in the body coordinate system and the DCM time-derivative $\dot{\boldsymbol{A}}_{\mathcal{BI}}$, which can be represented by

$$\dot{\boldsymbol{A}}_{\mathcal{BI}} = \boldsymbol{A}_{\mathcal{BI}} [\boldsymbol{\omega}\times] \tag{2.4}$$

The attitude matrix or rotation matrix $\boldsymbol{A}_{\mathcal{BI}}$ is one of the most popular formalisms and considered the fundamental representation of the spacecraft's attitude. Its simplicity and applicability, together with the fact that it is uniquely determined for any given configurations, means that there are no singularities or ambiguities. Furthermore, this representation is continuously differentiable and does not require evaluating trigonometric functions.

However, one drawback is that it includes nine different parameters, six of which are redundant due to the orthogonality and symmetry requirements. Other representations, such as the Euler angles or quaternion parameterization, require substantially fewer parameters, decreasing the computational load.

2.1.2 Euler Angles

The most common way of representing a rigid body attitude is a set of three successive rotations about the x, y and z axes, defined according to the respective Euler angles (ϕ, θ and ψ). In aerospace engineering applications, the three Euler angles are often referred to as roll, pitch and yaw, and each of these rotations about a single coordinate axis are called coordinate rotations.

Considering the generic X_1, Y_2 and Z_3 axes, a rotation of $\alpha \in \mathbb{R}$ about each axis corresponds to the following coordinate rotation matrices ($R_i : \mathbb{R} \longrightarrow SO(3)$, for $i \in \{1, 2, 3\}$):

$$\boldsymbol{R}_1(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix} \tag{2.5}$$

$$\boldsymbol{R}_2(\alpha) = \begin{bmatrix} \cos(\alpha) & 0 & -\sin(\alpha) \\ 0 & 1 & 0 \\ \sin(\alpha) & 0 & \cos(\alpha) \end{bmatrix} \tag{2.6}$$

$$\boldsymbol{R}_3(\alpha) = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.7}$$

Consequently, the expression that transforms an Euler angles vector ($[\phi, \theta, \psi]^T$) in its corresponding rotation matrix ($\boldsymbol{R}_{ijk} : \mathbb{R}^3 \longrightarrow SO(3)$, for $i, j, k \in \{1, 2, 3\}$) is the product of the coordinate matrices representing each individual rotation, as given by

$$\boldsymbol{R}_{ijk}(\phi, \theta, \psi) = \boldsymbol{R}_i(\phi)\boldsymbol{R}_j(\theta)\boldsymbol{R}_k(\psi) \tag{2.8}$$

To represent the attitude kinematics using Euler angles, a function $\boldsymbol{E} : \mathbb{R}^3 \to \mathbb{R}^{3\times3}$ must be developed such that

$$\begin{bmatrix} \dot{\phi} & \dot{\theta} & \dot{\psi} \end{bmatrix}^T = \boldsymbol{E}(\phi, \theta, \psi)\boldsymbol{\omega} \tag{2.9}$$

The Euler angles representation and its function depend vastly on the rotation sequence, that is, the coordinate rotations order. This freedom of choice regarding which axes to rotate and in which succession raise several possibilities. Out of the 12 possible unique combinations, the rotation matrices sequence chosen is the asymmetric set $(1, 2, 3)$, as it is the most commonly used for aerospace engineering applications. The attitude matrix for this sequence results in

$$\boldsymbol{R}_{123}(\phi, \theta, \psi) = \begin{bmatrix} c_\theta c_\psi & c_\theta s_\psi & -s_\theta \\ s_\phi s_\theta c_\psi - c_\phi s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & s_\phi c_\theta \\ c_\phi s_\theta c_\psi + s_\phi s_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi & c_\phi c_\theta \end{bmatrix} \tag{2.10}$$

and the attitude kinematics function from (2.9) is given by

$$\boldsymbol{E}_{123}(\phi,\theta,\psi) = \frac{1}{c_\theta}\begin{bmatrix} c_\theta & s_\phi s_\theta & c_\phi s_\theta \\ 0 & c_\phi c_\theta & -s_\phi c_\theta \\ 0 & s_\phi & c_\phi \end{bmatrix} \tag{2.11}$$

Its clear and convenient physical interpretation, along with the non-existence of redundant parameters, explain why the Euler angles representation is commonly used to describe the attitude. However, they can become troublesome to manage either in simulations or real life applications because singularities can occur at particular orientations. These singularities arise from the gimbal lock, resulting in indistinguishable changes between the first and third angles, when the second is at a critical value. Therefore, this disadvantage must be taken into account if the attitude of the body is unpredictable or can be set arbitrarily, in which case the most suitable approach would be to use a quaternion representation.

2.1.3 Unit Quaternions

Unlike the Euler angles, which represent the orientation as a series of rotations around X, Y or Z axes, a quaternion represents attitude as a rotation according to a rotational angle $\alpha \in \mathbb{R}$ around a rotational axis $\hat{\boldsymbol{e}} \in \mathbb{R}^3$.

According to the *Euler's rotation theorem*, any displacement of a rigid body such that a point on the rigid body remains fixed, is equivalent to a single rotation about some axis that runs through the fixed point. It also means that the composition of two or more rotations is also a rotation. Therefore, any rotation in three dimensions can be represented as a combination of a scalar α and an unit vector $\hat{\boldsymbol{e}}$.

In conformance with the quaternion definition, properties and operations discussed in Annex A.2, and as usual in space applications, a normalized quaternion may be defined as

$$\boldsymbol{q}(\hat{\boldsymbol{e}},\alpha) = \begin{bmatrix} \hat{\boldsymbol{e}}\sin(\alpha/2) \\ \cos(\alpha/2) \end{bmatrix} \tag{2.12}$$

The quaternion multiplication can also be used to prove that the product of two quaternions representing two consecutive rotations, can be summed up in a single rotation. Let $\boldsymbol{p} = \cos(\alpha/2) + \hat{\boldsymbol{e}}\alpha$ and $\boldsymbol{q} = \cos(\beta/2) + \hat{\boldsymbol{e}}\beta$ be two consecutive rotations, where $\boldsymbol{p}$ is the first rotation and $\boldsymbol{q}$ is the second rotation, the composed rotation $\boldsymbol{r}$ is computed as

$$\boldsymbol{r} = \boldsymbol{p} \otimes \boldsymbol{q} = \cos\left(\frac{\boldsymbol{\alpha}+\boldsymbol{\beta}}{2}\right) + \hat{\boldsymbol{e}}\sin\left(\frac{\boldsymbol{\alpha}+\boldsymbol{\beta}}{2}\right) = \cos(\boldsymbol{\gamma}) + \hat{\boldsymbol{e}}\sin(\boldsymbol{\gamma}) \tag{2.13}$$

The attitude kinematics in unit quaternions is represented using the matrix expression for the quaternion product (A.7). Considering $\boldsymbol{\omega} \in \mathbb{R}^3$ as the angular velocity in the spacecraft body frame and $\dot{\boldsymbol{q}}$ as the quaternion derivative, the relation is described as

$$\dot{\boldsymbol{q}} = \frac{1}{2}\boldsymbol{Q}(\boldsymbol{q})\begin{bmatrix} 0 \\ \boldsymbol{\omega} \end{bmatrix} \tag{2.14}$$

Alternatively, applying the distributive property in the previous expression and rearranging the matrix in order to isolate the quaternion, the nonlinear spacecraft kinematics equations of motion can be represented by the quaternion as follows:

$$\dot{\boldsymbol{q}} = \frac{1}{2}\boldsymbol{\Omega}(\boldsymbol{\omega})\boldsymbol{q} \tag{2.15}$$

where $\boldsymbol{\Omega}(\boldsymbol{\omega}) \in \mathbb{R}^{4\times4}$ is the dynamics matrix, defined as

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

The quaternion has the smallest dimension possible for a global non-singular representation of the $SO(3)$, also without the need to evaluate trigonometric or other complicated functions. Additionally, this four-element parameterization is more efficient to express rotations than the attitude matrix itself, and is bound to a single constraint - the unity norm. Still, the quaternions' most significant advantages are the absence of singularities and their suitability to use the angular velocity in the kinematics equation (2.15).

Nevertheless, there are a few disadvantages to the quaternion parameterization, such as not having an intuitive physical interpretation and including one redundant degree of freedom. Another major drawback present in attitude estimation for both quaternions and axis-angle representations is the unwinding phenomenon, i.e., the estimate may rest arbitrarily close to the true attitude and yet rotate through large angles before achieving the final desired attitude. This situation derives from the ambiguity in the representations, since $\boldsymbol{q}$ and $-\boldsymbol{q}$ describe the same rotation.

2.1.4 Axis-Angle

In mathematics, the axis-angle representation defines a three-dimensional rotation about a fixed axis, specified by a unit vector $\hat{e}$, according to a rotation angle α, and is a convenient parameterization when dealing with rigid body dynamics. As stated in the previous subsection (2.1.3), this two-parameter convention is valid to represent any rotation, in agreement with *Euler's rotation theorem*.

An attitude matrix $\boldsymbol{A}$, conforming to definition (2.3) and to the axis-angle representation, can be provided to express the relation between two vectors as

$$\boldsymbol{A}(\hat{\boldsymbol{e}}, \alpha) = (\cos(\alpha))\boldsymbol{I}_3 + \sin(\alpha)[\hat{\boldsymbol{e}}\times] + (1 - \cos(\alpha))\hat{\boldsymbol{e}}\hat{\boldsymbol{e}}^T \tag{2.17}$$

where $[\hat{e}\times]$ is the skew-symmetric cross product matrix formed from vector $\hat{e}$ as follows from Annex A.1.

Despite having an intuitive physical interpretation, this rotation formalism requires four parameters and a single quadratic constraint. To retrieve the axis-angle representation parameters from the rotation matrix $\boldsymbol{A}$, the rotation angle is calculated as

$$\alpha = \arccos\left(\frac{\boldsymbol{A}_{11} + \boldsymbol{A}_{22} + \boldsymbol{A}_{33} - 1}{2}\right) \tag{2.18}$$

and then, if $\sin(\alpha) \neq 0$, that value is used to compute the normalized axis

$$\hat{\boldsymbol{e}} = \frac{1}{2\sin(\alpha)} \begin{bmatrix} \boldsymbol{A}_{32} - \boldsymbol{A}_{23} \\ \boldsymbol{A}_{13} - \boldsymbol{A}_{31} \\ \boldsymbol{A}_{21} - \boldsymbol{A}_{12} \end{bmatrix} \tag{2.19}$$

Alternatively, as presented in equation (2.12), the axis-angle parameters can also be handled as variables to define the normalized quaternion. Then, from equation (2.15), the attitude kinematics are available for computation.

It is important to note that equation (2.18) has two symmetric solutions, thus implying that a rotation of α about $\hat{\boldsymbol{e}}$ is equivalent to a rotation of $-\alpha$ about $-\hat{\boldsymbol{e}}$. Therefore, as discussed in the quaternions subsection (2.1.3), the axis-angle representation may cause the unwinding phenomenon.

Furthermore, this parameterization requires leading with trigonometric functions, a singularity for the rotation axis $\hat{\boldsymbol{e}}$ when $\sin(\alpha) = 0$, and one redundant parameter. In fact, its main advantage relies on a convenient and simple definition that provides an understandable representation of the attitude.

2.2 Reference Frames

In physics and astronomy, a reference frame is an abstract coordinate system specified by its origin and the orientation of its coordinate axes. The attitude parameterizations described in Section 2.1 represent the spacecraft's orientation with respect to a particular coordinate frame. Some of the most frequently used reference frames are Earth Centered Inertial (ECI), Earth Centered Earth Fixed (ECEF), Local Vertical Local Horizontal (LVLH), and Spacecraft Body Frame [10, 11, 40].

2.2.1 Earth Centered Inertial (ECI)

To study the attitude of a spacecraft orbiting the Earth, the approximately inertial ECI frame is the go-to coordinate system. It has its origin at the center of mass of the Earth and three orthogonal axes fixed relative to it. The Z-axis is aligned with the mean North Pole, the X-axis points to the Vernal Equinox direction, and the Y-axis completes the right-handed system, as seen in Figure 2.1.

Although the ECI has a linear acceleration due to Earth's translation movement around the Sun, that effect may be neglected for attitude analysis and it can be considered inertial with respect to the stars. Hence, this coordinate system is used in this dissertation to define the *inertial reference frame*.

2.2.2 Earth Centered Earth Fixed (ECEF)

The ECEF is a global reference frame coincident with the ECI, as it shares the same origin - the center of mass of the Earth. Although it also has the Z-axis aligned with the mean North Pole, this coordinate system rotates with the Earth, differing from the inertial reference frame. The X-axis is determined according to the intersection of a reference Meridian and the Equator, and the Y-axis completes the right-handed system, as seen in Figure 2.2.

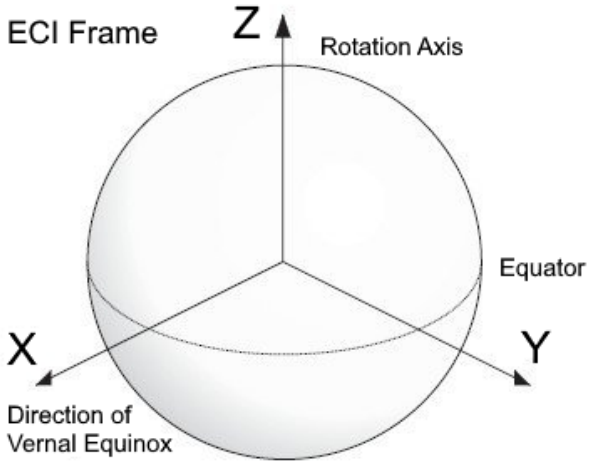


Figure 2.1: ECI reference frame (adapted from [41])

As the first pair of reference frames only differ one another by a rotation angle about the Z-axis, the transformation of a vector in the ECI frame to its ECEF representation (denoted by $\mathcal{I} = \{i_1, i_2, i_3\}$ and $\mathcal{E} = \{e_1, e_2, e_3\}$, respectively) can be mapped by:

$$\boldsymbol{A}_{\mathcal{E}\mathcal{I}} = \begin{bmatrix} \cos(\theta_{GMST}) & \sin(\theta_{GMST}) & 0 \\ -\sin(\theta_{GMST}) & \cos(\theta_{GMST}) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.20}$$

where $\boldsymbol{A}_{\mathcal{E}\mathcal{I}}$ is the rotation matrix defined according to equation (2.7) and θ_{GMST} is the Greenwich Mean Sidereal Time (GMST) angle.

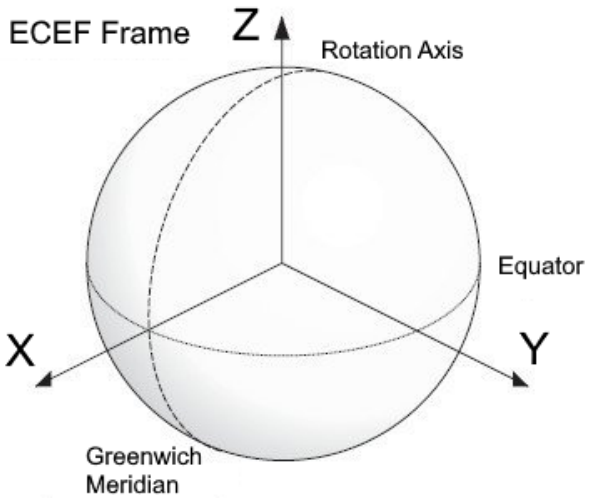


Figure 2.2: ECEF reference frame (adapted from [41])

2.2.3 Local Vertical Local Horizontal (LVLH)

For Earth-pointing satellites, it is often convenient to define a coordinate system referenced to the spacecraft's orbit. The predominant option is the LVLH orbit frame shown in Figure 2.3. Its Z-axis points along the nadir vector, directly towards the center of the Earth, and its Y-axis is coincident with the negative orbit normal, i.e., in the opposite direction of the spacecarft's orbital angular velocity. The X-axis is tangent to the trajectory, pointing in the direction of the linear velocity vector, and perpendicular to the remaining axes, completing the right-handed triad.

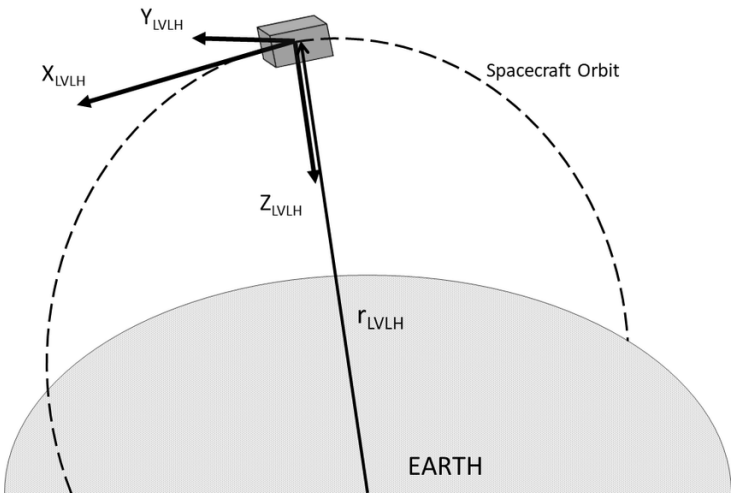


Figure 2.3: LVLH reference frame (from [42])

The LVLH reference frame is mostly used for representing nadir and zenith pointing, which are two opposite orientations distinguished for aligning the satellite with the Earth's radius. However, it can also be convenient to describe external disturbances, such as the gravity gradient torque.

2.2.4 Spacecraft Body Frame

The body coordinate system has its origin at the center of mass of the satellite and is vehicle-carried, i.e., its axes rotate with the spacecraft and are usually fixed to the principal axes of inertia (3.2). In this reference frame, which is denoted by $\mathcal{B} = \{b_1, b_2, b_3\}$, the Z_b-axis is determined according to the major principal axis of inertia direction, the Y_b-axis is aligned with the minor principal axis of inertia, and the X_b-axis points to the intermediate principal axis of inertia, complying with the right-hand rule. An axes representation of the reference frame is provided by Figure 2.4.

However, this dissertation introduces a nuance to the spacecraft body frame $\mathcal{B}$. In reality, it is slightly rotated w.r.t. the principal axes of inertia, as a means to reproduce the misalignment caused by an initial computation error. Even so, all equations use this unaligned reference frame when referring to the body coordinate system to prevent any ambiguity. On the other hand, the principal axes of inertia now constitute a body-centered reference frame denoted by $\mathcal{J} = \{j_1, j_2, j_3\}$. In Chapter 4, a representation of both coordinate systems can be seen in Figure 4.1.

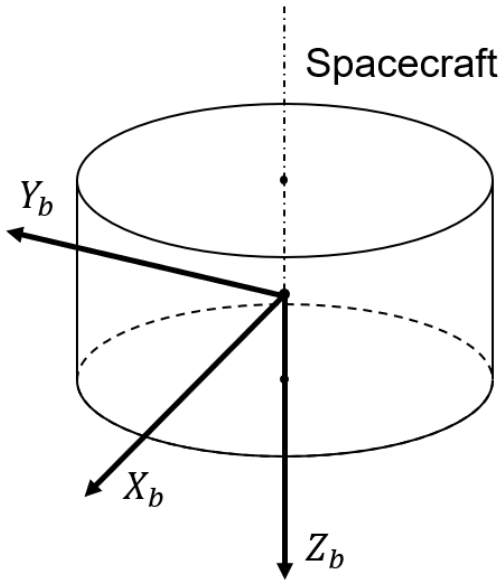


Figure 2.4: Spacecraft body frame

2.3 Attitude Kinematics and Dynamics

Rigid body dynamics studies the movement of systems under the action of external forces, and are described by the laws of kinematics. The distinction between kinematics and dynamics is that the former covers aspects of motion that can be analyzed without consideration of forces or torques. When those are introduced, it enters the realm of dynamics [11].

This section presents the fundamentals of a spacecraft's rotational motion, providing the solution adopted for the equations of motion, which define the relations between angular position, angular velocity, and angular momentum. Additionally, external disturbances that perturb the spacecraft orientation are described to conclude the attitude kinematics and dynamics.

2.3.1 Kinematics

Attitude kinematics analyzes how the orientation of the spacecraft changes in time according to the angular velocity, $\boldsymbol{\omega} \in \mathbb{R}^3$. In this case, the object of study is the attitude relating the spacecraft body frame, $\mathcal{B}$, to the inertial reference frame, $\mathcal{I}$, and the kinematic relations for the attitude matrix and its respective quaternions representation.

Denoting $\boldsymbol{A}_{\mathcal{BI}}$ as the rotation matrix from the frame $\mathcal{I}$ to frame $\mathcal{B}$ (2.2), $\boldsymbol{\omega}_{\mathcal{B}}^{\mathcal{BI}}$ as the angular velocity of frame $\mathcal{B}$ with respect to frame $\mathcal{I}$, represented in frame $\mathcal{B}$, then, the fundamental equation of attitude kinematics, as seen in equation (2.4), is given by

$$\dot{\boldsymbol{A}}_{\mathcal{BI}} = \boldsymbol{A}_{\mathcal{BI}} \left[\boldsymbol{\omega}_{\mathcal{B}}^{\mathcal{BI}} \times\right] \tag{2.21}$$

where $\dot{\boldsymbol{A}}_{\mathcal{BI}}$ is the time derivative of $\boldsymbol{A}_{\mathcal{BI}}$, and $[\boldsymbol{\omega}_{\mathcal{B}}^{\mathcal{BI}} \times]$ is a skew-symmetric matrix computed according

to Annex A.1.

This kinematic relation in the previous formula can also be written in terms of quaternions [10, 11]. From the derivative of the quaternion defined in Section 2.1, the nonlinear spacecraft kinematics equations of motion can be represented by the quaternion as follows:

$$\dot{\boldsymbol{q}}_{\mathcal{BI}} = \frac{1}{2}\boldsymbol{\omega}_{\mathcal{B}}^{\mathcal{BI}} \otimes \boldsymbol{q}_{\mathcal{BI}} \equiv \frac{1}{2}\boldsymbol{\Omega}(\boldsymbol{\omega}_{\mathcal{B}}^{\mathcal{BI}})\boldsymbol{q}_{\mathcal{BI}} \tag{2.22}$$

where $\boldsymbol{q}_{\mathcal{BI}}$ is the quaternion representation of the rotation $\boldsymbol{A}_{\mathcal{BI}}$, and $\boldsymbol{\Omega}(\boldsymbol{\omega}_{\mathcal{B}}^{\mathcal{BI}})$ is the dynamics matrix. Both $\dot{\boldsymbol{q}}$ and $\boldsymbol{\Omega}(\boldsymbol{\omega})$ have been previously defined in equations (2.15) and (2.16), respectively. A significant advantage of the quaternion parameterization is that the kinematic equation becomes linear, hence it is the attitude kinematics representation implemented in this dissertation.

2.3.2 Dynamics

In the previous subsection, the attitude motion was described in terms of the angular velocity, $\boldsymbol{\omega} \in \mathbb{R}^3$, of the body frame $\mathcal{B}$ fixed to the spacecraft with respect to an inertial reference frame, $\mathcal{I}$. As previously stated, attitude dynamics studies the forces and torques applied to the spacecraft in order to determine its angular velocity, which is necessary to compute the spacecraft attitude.

Euler's second law of motion [43] states

$$\dot{\boldsymbol{h}} + \boldsymbol{\omega} \times \boldsymbol{h} = \boldsymbol{\tau} \tag{2.23}$$

where $\boldsymbol{h} \in \mathbb{R}^3$ is the spacecraft's angular momentum about its center of mass, $\dot{\boldsymbol{h}} \in \mathbb{R}^3$ is its respective time derivative, and $\boldsymbol{\tau} \in \mathbb{R}^3$ is the total torque acting on the body about its center of mass. The angular momentum vector $\boldsymbol{h}$, represented in the body frame $\mathcal{B}$, can be given by the following simple expression:

$$\boldsymbol{h} = \boldsymbol{J}\boldsymbol{\omega}_{\boldsymbol{I}} \tag{2.24}$$

with $\boldsymbol{\omega}_{\boldsymbol{I}} = [\omega_{I1}, \omega_{I2}, \omega_{I3}]^T$ being the angular velocity vector of the spacecraft body frame w.r.t. the inertial frame and expressed in the body frame, and $\boldsymbol{J} \in \mathbb{R}^{3\times3}$ being the moment of *inertia tensor* or *inertia matrix* of a spacecraft [10], defined as

$$\boldsymbol{J} = \begin{bmatrix} J_{11} & -J_{12} & -J_{13} \\ -J_{21} & J_{22} & -J_{23} \\ -J_{31} & -J_{32} & J_{33} \end{bmatrix} \tag{2.25}$$

Then, from [44], equation (2.23) can be rewritten as

$$\boldsymbol{\tau} = \left(\frac{d\boldsymbol{h}_{\boldsymbol{I}}}{dt}\right)\Big|_{\mathcal{B}} = \left(\frac{d\boldsymbol{h}}{dt}\right) + \boldsymbol{\omega}_{\boldsymbol{I}} \times \boldsymbol{h} \tag{2.26}$$

and, assuming a constant inertia tensor in the body frame $\mathcal{B}$, this yields

$$\left(\frac{d\boldsymbol{h}}{dt}\right) = \boldsymbol{J}\dot{\boldsymbol{\omega}}_{\boldsymbol{I}} = -\boldsymbol{\omega}_{\boldsymbol{I}} \times \boldsymbol{J}\boldsymbol{\omega}_{\boldsymbol{I}} + \boldsymbol{\tau} \tag{2.27}$$

These are called the *Euler's moment equations*, which describe the rotation of a rigid body, using a rotating reference frame with its axes fixed to the body and parallel to the principal axes of inertia [8, 43].

In three-dimensional principal orthogonal coordinates, they become:

$$\boldsymbol{J}\dot{\boldsymbol{\omega}}_{\boldsymbol{I}} + \boldsymbol{\omega}_{\boldsymbol{I}} \times \boldsymbol{J}\boldsymbol{\omega}_{\boldsymbol{I}} = \boldsymbol{\tau} \Longrightarrow \begin{cases} J_1\dot{\omega}_1 + (J_3 - J_2)\omega_2\omega_3 = \tau_1 \\ J_2\dot{\omega}_2 + (J_1 - J_3)\omega_3\omega_1 = \tau_2 \\ J_3\dot{\omega}_3 + (J_2 - J_1)\omega_1\omega_2 = \tau_3 \end{cases} \tag{2.28}$$

where $\tau_k \in \mathbb{R}$ are the components of the torque, $J_k \in \mathbb{R}$ are the principal moments of inertia, and $\omega_k \in \mathbb{R}$ are the components of the angular velocity, all about the principal axes.

Likewise, these expressions can also be written in matrix form, resulting in

$$\dot{\boldsymbol{\omega}} = \begin{bmatrix} \dot{\omega}_1 \\ \dot{\omega}_2 \\ \dot{\omega}_3 \end{bmatrix} = \begin{bmatrix} 0 & \frac{J_2}{J_1}\omega_3 & -\frac{J_3}{J_1}\omega_2 \\ -\frac{J_1}{J_2}\omega_3 & 0 & \frac{J_3}{J_2}\omega_1 \\ \frac{J_1}{J_3}\omega_2 & -\frac{J_2}{J_3}\omega_1 & 0 \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} + \begin{bmatrix} \frac{1}{J_1} & 0 & 0 \\ 0 & \frac{1}{J_2} & 0 \\ 0 & 0 & \frac{1}{J_3} \end{bmatrix} \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} \tag{2.29}$$

where all principal products of inertia are positive values ($J_i > 0$), thus assuring that there are no zero-division singularities. At last, it should be recognized that the external torque $\boldsymbol{\tau}$ is composed of two parts - the control torque $\boldsymbol{\tau}_u \in \mathbb{R}^3$, that is, the input commanded by the Control System, and the disturbances torque $\boldsymbol{\tau}_d \in \mathbb{R}^3$, which represents the torques acting on a spacecraft subject to the surrounding environment. Thus, the dynamics equation (2.27) can be extended to

$$\boldsymbol{J}\dot{\boldsymbol{\omega}}_{\boldsymbol{I}} + \boldsymbol{\omega}_{\boldsymbol{I}} \times \boldsymbol{J}\boldsymbol{\omega}_{\boldsymbol{I}} = \boldsymbol{\tau}_u + \boldsymbol{\tau}_d \tag{2.30}$$

2.3.3 External Disturbances

In space, there are no large torques applied to the satellites, and hence minor influences integrated over time play major roles in governing the attitude dynamics of spacecraft. External torques arise from the interaction of a space vehicle with its environment, and require a specification of both vehicle properties and spacial environment to be calculated. Indeed, these torques are difficult to predict and can significantly affect the spacecraft's orientation if the ADCS is poorly designed. All disturbance torques are induced by external forces acting outside the center of mass (CM) [10].

This subsection describes the most representative perturbation torques (and their respective perturbation forces), explaining their origin and providing simple models for their computation. As the predominant disturbance, the gravity-gradient is tackled separately in more detail. Other spacecraft disturbance, the J_2 effect, is also addressed to complete the main external disturbances.

Disturbance Torques

For objects in lower altitudes, atmospheric drag is still a significant source of perturbing torque, as the remaining atmosphere present at the satellite's orbit altitude will disturb its path [10, 11]. The aerodynamic center of pressure (CP) is the exact point at which the aerodynamic force is applied, and depends on the satellite's form, velocity, and outward normal. Considering the drag equation definition, the **aerodynamic force** $\boldsymbol{f}_a \in \mathbb{R}^3$ w.r.t. the $\mathcal{B}$ reference frame can be computed as

$$\boldsymbol{f}_a = \frac{1}{2}\rho||\boldsymbol{V}||\boldsymbol{V}SC_D \tag{2.31}$$

where ρ is the atmosphere density, $\boldsymbol{V}$ is the spacecraft velocity, S is the cross section area and C_D is the drag coefficient. Consequently, the **aerodynamic torque** $\boldsymbol{\tau}_a \in \mathbb{R}^3$ can be evaluated by

$$\boldsymbol{\tau}_a = \boldsymbol{r} \times \boldsymbol{f}_a \tag{2.32}$$

with $\boldsymbol{r}$ being the moment arm vector from the center of mass to the aerodynamic center of pressure [10].

Solar radiation pressure (SRP) is another cause of perturbances, as the photons from the Sun contain momentum, which produces an effective pressure on spacecraft surfaces. Although the SRP effect is dominated by aerodynamics at low-Earth orbits, its cumulative impact over sufficiently long periods may be significant in higher altitude orbits ($\geq 800\,\mathrm{km}$) [10, 11]. There are three distinct modes of interaction between solar radiation and the satellite surface - diffuse reflection, specular reflection, and total absorption. Even though it depends on the spacecraft surface properties, the latter is the most prevalent.

The optical center of pressure constitutes the effective location of SRP application, and is dependent on factors such as the total lit portion of the spacecraft surface, the location of the projected area of surface elements, and their unit outward normal. Considering the different modes of interaction, the total **SRP force** $\boldsymbol{f}_s \in \mathbb{R}^3$ represented in the body frame can be approximated by

$$\boldsymbol{f}_s = (1+K)S\boldsymbol{P}_s \tag{2.33}$$

where K is the reflectivity (from 0 to 1), S is the cross section area and $\boldsymbol{P}_s$ is the solar pressure. Consequently, the **SRP torque** $\boldsymbol{\tau}_s \in \mathbb{R}^3$ is given by

$$\boldsymbol{\tau}_s = \boldsymbol{r} \times \boldsymbol{f}_s \tag{2.34}$$

with $\boldsymbol{r}$ being the moment arm vector from the center of mass to the optical center of pressure [10].

Earth's magnetic field does not influence a satellite's orbit, but affects its attitude by generating torques from the interaction with currents inside the spacecraft. Presently, the majority of space vehicles are designed to reduce the effect of this perturbance by minimizing the use of magnetic materials,

a concept known as *magnetic cleanliness*. Still, the Earth's magnetic field effect is more often than not negligible, especially for high altitude orbits [11].

Considering the magnetic field of the Earth $\boldsymbol{b} \in \mathbb{R}^3$ described in body fixed coordinates and the magnetic dipole moment from the spacecraft $\boldsymbol{\mu} \in \mathbb{R}^3$, then the **magnetic torque** on the spacecraft is computed as

$$\boldsymbol{\tau}_m = \boldsymbol{\mu} \times \boldsymbol{b} \tag{2.35}$$

Gravity-Gradient Torque

Gravitational torques are fundamental to the attitude dynamics of spacecraft and are the most important external disturbances, exceeding the magnitude of the next-largest perturbation by up to six times. However, this disparity between the various disturbance torques may differ according to the position, attitude, altitude and shape of the spacecraft.

In Earth's orbit, Newton's law of universal gravitation applies and the gravitational field cannot be considered uniform, decreasing in quadratic proportion with distance to the planet. Consequently, these variations in the specific gravitational force over a rigid body lead to a torque about its body mass center. Thus, any non-symmetrical satellite in Earth's gravity field is subject to a gravity-gradient torque [11].

To simplify the gravitational torque expressions, some assumptions are usually made regarding both the gravitational source and the spacecraft itself. First, only one celestial primary needs to be considered for computations and it possesses a spherically symmetrical mass distribution. Additionally, the satellite is relatively small when compared to its distance from the primary's center of mass and it consists of a single body.

From Newton's law of universal gravitation and following the previous assumptions, the gravity-gradient disturbance torque $\boldsymbol{\tau}_{gg} \in \mathbb{R}^3$ can be defined as

$$\boldsymbol{\tau}_{gg} = \frac{3\mu_E}{r_c^5} \boldsymbol{r}_c \times \boldsymbol{J}\boldsymbol{r}_c \tag{2.36}$$

where μ_E is the Earth gravitational parameter, $r_c = ||\boldsymbol{r}_c||$ is the orbit radius norm (i.e., the distance from Earth's center of mass to the satellite), $\boldsymbol{r}_c$ is the orbit radius vector expressed in the spacecraft body frame, and $\boldsymbol{J}$ is the satellite's inertia matrix [10, 40].

For spacecraft in a circular orbit, the distance from Earth's center of mass to the satellite is constant and there is no need to apply Kepler's first law of motion to calculate its value over time. Hence, the only variable left to compute is the orbit radius $\boldsymbol{r}_c$, a zenith-pointing vector with origin at Earth's mass center. Assuming an orbit described on the equatorial plane according to the ECI reference frame, the horizontal orbit radius vector $\boldsymbol{r}_0 \in \mathbb{R}^3$ is given by

$$\boldsymbol{r}_0 = \begin{bmatrix} r_c \cos(\frac{2\pi t}{T_{orb}}) \\ r_c \sin(\frac{2\pi t}{T_{orb}}) \\ 0 \end{bmatrix} \tag{2.37}$$

with T_{orb} being the orbital period obtained from the expression below:

$$T_{orb} = 2\pi\sqrt{\frac{r_c^3}{\mu_E}} \tag{2.38}$$

In case the orbital plane is inclined according to an angle i, the inertial orbit radius vector $\boldsymbol{r}_I \in \mathbb{R}^3$ can be computed as the product of a rotation matrix $\boldsymbol{R}_2(i)$, defined conforming to equation (2.6), and the horizontal orbit radius vector $\boldsymbol{r}_0$, that is

$$\boldsymbol{r}_I = \boldsymbol{R}_2(i)\boldsymbol{r}_0 \tag{2.39}$$

At last, the orbit radius vector can be transformed to the body coordinate system through a rotation by the unit quaternion stemming from the attitude kinematics. Taking the quaternion multiplication definition (A.5) and extending the inertial orbit radius to a four-element vector, the orbit radius vector expressed in the spacecraft body frame $\boldsymbol{r}_c \in \mathbb{R}^3$ is given by

$$\boldsymbol{r}_c = \boldsymbol{q} \otimes \boldsymbol{r_I} \otimes \boldsymbol{q}^* \tag{2.40}$$

where $\boldsymbol{q}$ and $\boldsymbol{q}^*$ represent the attitude kinematics quaternion and its complex conjugate, respectively.

J2 Effect

Because the Earth is not a perfect sphere, the non-uniformity of its mass distribution is reflected in the gravity field. The term J_2 comes from the infinite mathematical series commonly implemented to describe the perturbational effects of oblation on the gravity of a planet [11].

This 6th-order spherical harmonic expansion model includes zonal harmonics, each one associated to a zonal coefficient J_n. However, J_2 is over 1000 times larger than the remaining higher order terms, hence the force produced by this effect can be approximated to

$$\boldsymbol{f}_{J_2} = -m_c \frac{3J_2\mu_E {r_E}^2}{2{r_c}^4} \begin{bmatrix} 1 - 3\sin^2(i)\sin(\nu) \\ 2\sin^2(i)\sin(\nu)\cos(\nu) \\ 2\sin(i)\cos(i)\sin(\nu) \end{bmatrix} \tag{2.41}$$

where m_c is the mass of the spacecraft, J_2 is a constant value, μ_E is the Earth gravitational parameter, r_E is the Earth radius, r_c is the orbit radius, i is the orbit inclination, and ν is the true anomaly.

As this disturbance is evaluated as a low-magnitude force acting only on the satellite's position, it is not included in the simulation environment developed for this study.

2.4 Sensors and Actuators

To determine and control a spacecraft's orientation, several attitude hardware components must be installed in the satellite. For attitude determination, there needs to be measurement equipment to

ascertain the orientation or at least provide measurements to be used for attitude estimation - these are called sensors. On the other hand, spacecraft actuators are the components responsible for producing the control torques necessary to achieve the desired attitude [9, 10].

Since the primary subject of this work is to design a navigation system to estimate the rotation angles between the spacecraft's spinning axis and its principal major axis of inertia, and the implementation only requires angular velocity measurements, this section exclusively delivers a brief review of the most popular attitude sensors and actuators, highlighting the applied solutions.

2.4.1 Sensors Overview

During the last decades, attitude sensor development has led to increased resolution and accuracy, in addition to a decrease in size, weight, and power requirements. Thus, most sensors have been adapted for small satellite applications, as its the case for sun sensors, magnetometers, star trackers, and gyroscopes [11].

Sun Sensors

A sun sensor is a device that detects the direction to the Sun. Depending on mission requirements, it can be as straightforward as some solar cells and shades, or as complex as a steerable telescope. The two major types of sun sensors are analog and digital.

Analog sensors are frequently called cosine detectors because they operate as solar cells whose output current is proportional to the cosine of the angle between the sensor normal and the incident solar radiation. Therefore, considering that a sun sensor only measures the incidence angle, all sun-vector possible directions define the so-called sun-cone. One solution is to simply include additional measurements from two other sun sensors to find the three sun-cones intersection and determine the true sun-vector [9].

Digital sensors are significantly more accurate, delivering a full sun-vector in sensor coordinates (which can be promptly rotated to the body reference frame), but also more intricate and expensive. For many applications, analog sun sensors must be sufficient though.

The main drawback of sun sensors lies in its restricted useful field of view (FOV), demanding the intervention of multiple sensors to keep the Sun in line of sight and achieve the necessary coverage. Additionally, there may be spacecraft angular rate limitations, as these sensors cannot provide the full three axes attitude of the satellite. Thus, sun sensors are more suitable for safe-mode operation, as contrary to more complex sensors, they have low power requirements and can be kept on along side other spacecraft essential functions.

Magnetometers

The three-axes magnetometer measures the components of the local magnetic field vector in sensor coordinates, providing the magnitude and direction of the magnetic field. This information can be readily converted to the spacecraft body frame, assuming sensor orientation knowledge.

Although they are sensitive to disturbances and have low accuracy, magnetometers are widely used as spacecraft attitude sensors as they are reliable, lightweight, and have low power requirements, and are particularly useful for initial attitude determination, before the more precise measurements provided by different sensors become available [9].

However, its dependence on the magnetic field of the Earth to provide measurements is a limitation since it implies that they can only be used in low Earth orbits (LEOs). In addition, the magnetometer requires further measurements if used with magnetic torques, to ensure there is no influence between them. Therefore, the sensor position inside the spacecraft is crucial to minimize corrupted values.

Star Trackers

Stars yield the most precise references for attitude determination, because they are inertially fixed objects and seem very small when compared to the apparent size of the Sun or Earth, hence the direction of a certain star can be measured very accurately.

Considering Earth orbiting spacecraft, both the sun-vector and the magnetic field vector depend on the orbital position of the Earth around the Sun, as well as the orbital position of the satellite around the Earth. Therefore, the Sun and Earth's magnetic field directions must be calculated in real-time, and there are errors associated with these computations. On the contrary, relying on stars' positions does not require real-time computations, as their inertial position and associated reference vectors may be cataloged and stored onboard.

Simple star trackers have the ability to lock-on, identify and track a single star at the time, outputting the measured unit star pointing vector in the star tracker frame. Then, using knowledge of the sensor orientation relative to the satellite and combining with other star tracker measurements, the spacecraft attitude can be determined. Modern star trackers can lock-on, identify and track multiple stars at once, thus processing all the star vector measurements to directly output the full three-axes attitude [11].

Although considered the most accurate device for a spacecraft's position evaluation, capable of attitude determination to within arc-second accuracy, the star tracker is extremely complex and expensive. They are also limited to a sunless FOV, as the Sun could shine down the tracker's optical boresight and interfere in attitude computation. Moreover, star trackers generally have to be specifically designed for each mission as the number of individual stars available in the sensor's FOV changes according to the trajectory, and otherwise cataloging every single star would call for an extensive memory capacity. Finally, there may be a noticeable time delay in the output due to its significant computational load.

Gyroscopes

A gyroscope (or *gyro* for short) is a device that can measure the angular velocity, with very low noise interference, which can be numerically integrated by an onboard computer to provide an estimate for the attitude, or even directly used in attitude control. These rate sensors are constituted by a rotor that rotates around its spin axis and maintains its spin direction, regardless of the outer frame orientation and opposing any attempts to change its attitude, specifically, exhibiting precession and/or nutation motions.

The main gyro types used in small spacecraft are fiber optic gyros (FOGs), microelectromechanical systems (MEMS) gyros and ring laser gyros (RLGs), with the latter offering better performance at a mass and cost penalty. An RLG relies on the Sagnac effect to evaluate angular rotation, i.e., when the laser beam frequencies differ, a fringe pattern of alternate dark and light stripes is created. Thus, a beam split into two separate beams which travel around the ring in opposite directions forms a shifting interference pattern, and measuring its frequency and relative phase indicates the magnitude and direction of the gyroscope's rotation.

One of the principal drawbacks regarding gyroscopes is that they hold an inherent bias that results in a constant drift, that needs to be compensated. Nonetheless, rate sensors are still useful to obtain filtered estimates of the attitude when combined with other sensors. In fact, the sensor simulated in this work is a gyroscope that provides direct angular velocity measurements for attitude determination, corrupted by a band-limited white noise, as to simulate the disturbance present in these sensors.

2.4.2 Actuators Overview

Actuator technologies have also been scaled down over time to be appropriate for microsatellites and cubesats. These equipments can be divided into two separate classes - momentum exchange devices, that generate torques that can be considered internal, and reaction-type actuators, that generate torques that can be considered external to the spacecraft. The most commonly used momentum actuators are reaction wheels (RWs) and control moment gyros (CMGs), while the most widely implemented reaction actuators are magnetic torque rods and thrusters. Some of them, such as RWs and thrusters, can easily provide the desired torques, but magnetic torque rods and CMGs sometimes require alternative design methods to reach the intended control input [10], as discussed in this subsection.

Reaction Wheels (RWs)

Reaction wheels (RWs) are actuators used primarily by spacecraft for three axis attitude control, which do not require rockets or external applicators of torque. They are composed of brushless motors attached to high-inertia flywheels, each free to spin along a fixed spacecraft axis. This device provides a high pointing accuracy, and is particularly useful when the satellite must be rotated by very small amounts, such as keeping a telescope pointed at a star.

The acceleration and deceleration of the flywheels generate the full torque of the reaction wheel $\boldsymbol{\tau}_{RW} \in \mathbb{R}^3$, which is given by

$$\boldsymbol{\tau}_{RW} = -\dot{\boldsymbol{h}}_{\boldsymbol{RW}} = -\boldsymbol{J}_{RW}\dot{\boldsymbol{\omega}}_{\boldsymbol{fw}} \tag{2.42}$$

where $\boldsymbol{h}_{RW}$ is the angular momentum vector of all the flywheels, $\boldsymbol{J}_{RW}$ is the inertia matrix about the flywheel rotation axis, and $\boldsymbol{\omega}_{fw}$ is the angular velocity of the flywheels. It is important to notice that by rotating a flywheel in one direction about its spin-axis, the wheel applies a reaction torque to the platform in the opposite direction, thus respecting the principle of conservation of angular momentum.

These properties illustrate why a RW is considered a momentum exchange device, i.e., angular momentum is transferred between the wheels and the platform in order to control spacecraft attitude.

However, this torque swapping mechanism is limited by flywheels maximum speed and, as such, the stored angular momentum cannot be kept indefinitely to avoid saturation. In this regard, while reaction wheels provide the most precise attitude control, there needs to be a momentum management control system that augments the ACS with reaction-type actuators capable of generating an external torque on the spacecraft to de-load the built up angular momentum on the wheels [10, 11].

The main advantage in using reaction wheels relies in the fact that the actuator is aligned with the body axis. On the other hand, the maximum achievable speed in the flywheels requires special attention from the controller, often requiring a second actuator to compensate. On another note, if a reaction wheel is operated at a near-constant rotation speed, creating a momentum bias to imbue a satellite with a large amount of stored angular momentum, it is referred to as a *momentum wheel* [10].

Control Moment Gyros (CMGs)

Similar to momentum wheels, a control moment gyroscope (CMG) has flywheels rotating with a constant high speed, except that the spin-axis is not fixed, but moves as the wheels are suspended in a gimbal. Another significant distinction lies in the desired torque, which is produced by altering the gimbal orientation rather than the flywheels speed.

For a gimbal vector $\boldsymbol{\varsigma} \in \mathbb{R}^3$, and an angular momentum vector of the flywheel $\boldsymbol{h}_{fw} \in \mathbb{R}^3$, the normalized CMG torque vector $\boldsymbol{\tau}_{CMG} \in \mathbb{R}^3$ is generated perpendicular to the gimbal axis, as defined by

$$\boldsymbol{\tau}_{CMG} = \boldsymbol{\varsigma} \times \boldsymbol{h}_{fw} \tag{2.43}$$

with the three vectors being orthonormal, and $\boldsymbol{\varsigma}$ and $\boldsymbol{h}_{fw}$ aligned with their respective rotation axis [10].

These actuators have a rapid slew capacity and high pointing accuracy, since they generate a continuous torque by changing the CMG angular momentum. Compared to RWs, control moment gyros are more effective and efficient, as they can produce high torques due to the amplification property caused by the gimbal axis rate. Indeed they are used for attitude control of the ISS!

However, as seen for the Euler angles in subsection (2.1.2), these gyroscopes possess an inherent geometric singularity, also known as *gimbal lock*, making it impossible to exert a control torque parallel to the singularity direction. Usually, most maneuvers are thought out before hand to avoid this undesired condition, but in case a singularity occurs, then an external torque provided by a reaction-type actuator is required to reorient the satellite and restore CMGs function [10, 11].

Magnetic Torque Rods

For most LEO satellites, magnetic torque rods or magnetorquers are the standard actuating solution. They consist of planar coils of uniform wire attached to the satellite along its body axes, and by passing a current through the coils, a magnetic dipole is created. The interaction between the coil dipole $\boldsymbol{\mu}_{mtr} \in \mathbb{R}^3$ and Earth's magnetic field $\boldsymbol{b} \in \mathbb{R}^3$ generates a torque $\boldsymbol{\tau}_{mtr} \in \mathbb{R}^3$ according to the law:

$$\boldsymbol{\tau}_{mtr} = \boldsymbol{\mu}_{mtr} \times \boldsymbol{b} \tag{2.44}$$

This momentum triggered by the magnetorquers attempts to align the coils' magnetic field in the opposite direction to the Earth's. Hence, the magnetic torques can only be generated perpendicular to the instantaneous Earth's magnetic field vector $\boldsymbol{b}$. This implies that it is impossible to generate a torque about an arbitrary axis using magnetic torquers alone, so a spacecraft equipped only with them is instantaneously underactuated.

Although the Earth's magnetic field direction varies with the orbital position, it is really weak, thus the magnitude of the torque created by the magnetorquers is also very small, suggesting that magnetic torque rods have limited control authority. When fine attitude control is required, the actuators of choice are typically a set of reaction wheels, coupled with magnetic torquers for dumping, i.e., to unload excess momentum from the RWs, which builds up due to external disturbances acting on the spacecraft [10].

However, magnetorquers still have some advantages over other actuators, such as their lightweight and energy-efficient. In addition, magnetic torque rods can also be more reliable than RWs and CMGs as they have no moving parts.

Thrusters

Thrusters are spacecraft propulsion devices that use propellants to generate both forces and torques, so they can be used for trajectory and attitude control. The force $\boldsymbol{f}_{prop} \in \mathbb{R}^3$ and torque $\boldsymbol{\tau}_{prop} \in \mathbb{R}^3$ provided by a thruster are defined as

$$\boldsymbol{f}_{prop} = -\dot{m}\boldsymbol{v}_m \tag{2.45}$$

$$\boldsymbol{\tau}_{prop} = \boldsymbol{r} \times \boldsymbol{f}_{prop} \tag{2.46}$$

where $\dot{m}$ is the expelled mass flow rate, $\boldsymbol{v}_m$ is the velocity of the expelled mass relative to the spacecraft, and $\boldsymbol{r}$ is the moment arm vector from the center of mass to the thruster.

There are three types of thrusters that are commonly used for attitude control - cold gas, monopropellant and electric thrusters. A cold gas thruster uses a non-reacting gas (typically nitrogen or helium) stored in a high pressure container. Helium provides a higher specific impulse and saves mass when compared to nitrogen, but is more expensive and prone to leakage because of its reduced molecular dimension. Monopropellant thrusters take advantage of a catalytically decomposed propellant, the most standard option being hydrazine. Monopropellant systems deliver greater specific impulses and do not require the high pressures needed for cold gas, but hydrazine is an exceptionally toxic substance. In the last years, greener monopropellants have been proposed to diminish operational hazards, increase fuel efficiency and reduce environmental impact. Finally, an electric thruster manipulates electric fields to accelerate expelled mass to an extreme velocity, increasing the specific impulse and thus saving in propellant mass.

This actuator stands out for its high specific energy, efficiency, small volume and mass, and low power consumption. Thrusters also have the advantage of being independent from the ambient magnetic and gravitational fields, so they can be used for control or momentum unloading in any orbit. In contrast, they have a major drawback due to the expendable propellant requirement, which restricts the duration of a thruster-dependent mission according to the propellant supply [11].

Therefore, they are frequently used to complement the main attitude control system actuators, as they are more suitable to be implemented as a correction actuator due to the budget limit related to fuel consumption. In fact, as the scope of this study consists in correcting the alignment between the satellite's spin-axis and its principal major axis of inertia, it constitutes a common secondary application, thus, an energy-efficient actuator like the thrusters can be used to effectively reach the desired goal, without compromising the main control system intended for the extensive space mission objectives.

Chapter 3

Passive Spin Stabilization

Almost all spacecraft employ the spin stabilization effect during part or all of their lifetime in space. Many satellites are spin stabilized to take advantage of the gyroscopic effect and, therefore, maintain their spinning axis stable and avoid unnecessary power consumption. The dynamic attributes of spinning bodies are used to stabilize satellites' attitude within both the orbital maneuvering stage and the final mission orbit. However, it is important to notice that only a body spinning about its major or minor axes of inertia will keep the direction of its spin axis fixed with respect to the inertial space.

In this regard, an overview of the principal axes of inertia is initially provided in Section 3.1, including the inertia tensor decomposition in eigenvalues and eigenvectors. Afterwards, the Section 3.2 transforms and maps the obtained rotation matrix according to the Euler angles. At last, the conditions for spin stability about any principal axis with no external moments acting on the body are studied in Section 3.3.

3.1 Principal Axes of Inertia

The inertia matrix $\boldsymbol{J}$ (2.25) is composed of diagonal elements or *moments of inertia*, and non-diagonal elements or *products of inertia*. In case the inertia tensor is diagonal, i.e., all the products of inertia are null and the main diagonal has positive values, then the inertia matrix is considered to be defined according to the *principal axes of inertia* ($\boldsymbol{X}_b$, $\boldsymbol{Y}_b$, $\boldsymbol{Z}_b$). Hence, the larger and smaller values of the main diagonal correspond to the spin-stable major and minor axis of inertia, respectively [8].

The problem at hand is to determine the orientation of the principal axes of inertia and adopt them as the spacecraft body frame. Thus, to define a new reference frame in which the inertia tensor is diagonal, an axis rotation must be performed. This matrix diagonalization process to convert a non-diagonal real square symmetric matrix into a diagonal one is a common procedure treated in linear algebra. For this case, the eigendecomposition into the product of a rotation matrix $\boldsymbol{R} \in SO(3)$ and a diagonal matrix $\boldsymbol{\Lambda} \in SO(3)$ is given by

$$\boldsymbol{J} = \boldsymbol{R}\boldsymbol{\Lambda}\boldsymbol{R}^T \tag{3.1}$$

with

$$\boldsymbol{\Lambda} = \begin{bmatrix} J_x & 0 & 0 \\ 0 & J_y & 0 \\ 0 & 0 & J_z \end{bmatrix} \tag{3.2}$$

The columns of the rotation matrix $\boldsymbol{R}$ are the inertia matrix eigenvectors and define the directions of the principal axes of inertia in the body frame, while the constants J_x, J_y and J_z are called the *principal moments of inertia* and are the inertia tensor eigenvalues. This result was first shown by J. J. Sylvester [45], and it is a form of Sylvester's law of inertia.

3.2 Rotation Matrices

As seen in Section 2.1, there are several methods to define a rotation and, thus, multiple approaches to obtain the rotation matrix $\boldsymbol{R}$ presented in equation (3.2). However, before addressing the representation problem, it should be noted that the initial rotation matrix can be split into a product of three rotation matrices, each one representing a rotation around axes of a certain coordinate system $\boldsymbol{x}$, $\boldsymbol{y}$ and $\boldsymbol{z}$, i.e.

$$\boldsymbol{R}(\phi, \theta, \psi) = \boldsymbol{R}_x(\phi)\boldsymbol{R}_y(\theta)\boldsymbol{R}_z(\psi) \tag{3.3}$$

One method to effect a transformation from the axis-angle representation (ϕ along x, θ along y, and ψ along z) to their respective rotation matrices ($\boldsymbol{R}_x$, $\boldsymbol{R}_y$ and $\boldsymbol{R}_z$) specified by the Euler angles is recurring to an exponential map. Essentially, given an angle α and a unit vector $\boldsymbol{n}$ representing the unit rotation axis, the equivalent rotation matrix $\boldsymbol{R}$ is obtained from a Taylor expansion as

$$\boldsymbol{R} = \exp(\alpha \boldsymbol{n}) = \sum_{k=0}^{\infty} \frac{(\alpha \boldsymbol{N})^k}{k!} = \boldsymbol{I} + \alpha \boldsymbol{N} + \frac{1}{2!}(\alpha \boldsymbol{N})^2 + \frac{1}{3!}(\alpha \boldsymbol{N})^3 + ... \tag{3.4}$$

where $\boldsymbol{I}$ is the identity matrix and $\boldsymbol{N}$ is a skew-symmetric matrix which represents the cross-product of the unit vector $\boldsymbol{n}$, i.e., $\boldsymbol{N} = [\boldsymbol{n}\times]$, as defined according to Annex A.1.

After some algebraic manipulation and taking into account the Taylor series formulas for trigonometric functions, equation (3.4) can be rewritten simply as

$$\boldsymbol{R} = \boldsymbol{I} + (\sin\alpha)\boldsymbol{N} + (1 - \cos\alpha)\boldsymbol{N^2} \tag{3.5}$$

This expression is known as the *Rodrigues' rotation formula*, and is the applied solution to compute the rotation matrices in this work.

3.3 Spin Stability about the Principal Axes

For a spacecraft orbiting the Earth, according to Euler's moment equations of angular motion (2.27), its spin direction only changes if external moments are applied about its center of mass and perpendicu-

lar to the spin axis [8, 11]. However, it is important to notice that even in the absence of external torques, the dynamics equation still holds true. Recalling the Euler's moment equations along the principal axes of inertia (2.28), they can be rewritten considering free dynamics ($\tau = 0$)

$$\begin{cases} J_x\dot{\omega}_x + (J_z - J_y)\omega_y\omega_z = 0 \\ J_y\dot{\omega}_y + (J_x - J_z)\omega_z\omega_x = 0 \\ J_z\dot{\omega}_z + (J_y - J_x)\omega_x\omega_y = 0 \end{cases} \tag{3.6}$$

which yields the following 3-state nonlinear ordinary differential equations (ODEs):

$$\begin{cases} \dot{\omega}_x = -\frac{(J_z - J_y)}{J_x}\omega_y\omega_z \\ \dot{\omega}_y = -\frac{(J_x - J_z)}{J_y}\omega_x\omega_z \\ \dot{\omega}_z = -\frac{(J_y - J_x)}{J_z}\omega_x\omega_y \end{cases} \tag{3.7}$$

From these ODEs, the spin stabilization conditions can be analyzed at once. These homogeneous equations can be solved under two different assumptions. The first case is to consider that the rotating body is axisymmetric, with, for instance, $J_x = J_y$. Suppose also that the body is spinning about the $\boldsymbol{Z}_b$ axis, which is the axis of symmetry. From the third free dynamics equation (3.6),

$$\dot{\omega}_z = -\frac{(J_x - J_x)}{J_z}\omega_x\omega_y = 0 \tag{3.8}$$

thus ω_z is constant. The remaining equations for ω_x and ω_y stay the same, except that ω_z is time-invariant. The first two equations become linear and, let $\lambda = \omega_z(J_z - J_x)/J_x$, they can be put in the following form:

$$\begin{cases} \dot{\omega}_x = -\lambda\omega_y \\ \dot{\omega}_y = \lambda\omega_x \end{cases} \tag{3.9}$$

If the first equation is multiplied by ω_x and the second equation is multiplied by ω_y, adding the first to the second results in $\omega_x\dot{\omega}_x + \omega_y\dot{\omega}_y = 0$, from which follows the important conclusion

$$\omega_{xy}^2 = \omega_x^2 + \omega_y^2 = \text{const} \tag{3.10}$$

where ω_{xy} is the component of angular velocity in the $\boldsymbol{X}_b$-$\boldsymbol{Y}_b$ plane of the rotating body, in this case, perpendicular to the axis of symmetry. This type of motion is often called *precession* [8].

In the previous discussion, it was assumed that the body was axisymmetric, with $J_x = J_y$. Now this assumption is relieved, as it is wished to find conditions for stability about any principal axis with no external moments acting on the body. Supposing again that stability conditions for rotation about the $\boldsymbol{Z}_b$ body axis are sought. Introducing a small disturbance $\epsilon \in \mathbb{R}$ in the angular velocity so that $\omega_\ddagger = \omega_z + \epsilon$

and, consequently, $\dot{\omega}_{\ddagger} = \dot{\omega}_z + \dot{\epsilon} = \dot{\epsilon}$. If $\epsilon \Rightarrow 0$ then equations (3.6) become

$$\begin{cases} J_x\dot{\omega}_x + (J_z - J_y)\omega_y\omega_z = 0 \\ J_y\dot{\omega}_y + (J_x - J_z)\omega_z\omega_x = 0 \\ J_z\dot{\epsilon} + (J_y - J_x)\omega_x\omega_y = 0 \end{cases} \tag{3.11}$$

The first two equations are linear, and a second differentiation of the first using the second yields

$$\ddot{\omega}_x + \frac{J_z - J_y}{J_y}\frac{J_z - J_x}{J_x}\omega_z^2\omega_x = 0 \tag{3.12}$$

and taking its Laplace transform results in the characteristic equation $\lambda(s)$ below:

$$\lambda(s) = \left[s^2 - \frac{J_z - J_y}{J_y}\frac{J_x - J_z}{J_x}\omega_z^2\right]\omega_x(s) = 0 \Longrightarrow \lambda(s) = s^2 - \gamma^2 = 0 \tag{3.13}$$

where $s = \pm\gamma$ are the roots of the characteristic equation and with $\gamma = \sqrt{(1 - J_z/J_x)(J_z/J_y - 1)\omega_z^2}$.

For stability, all roots of $\lambda(s)$ must have a negative real part ($\mathrm{Real}(s) < 0$). However, that is not possible for this system, as γ can only be a real number or a pure imaginary. Thus, to aim for neutral stability and at least prevent increased disturbances, all roots of $\lambda(s)$ should be purely imaginary ($\mathrm{Real}(s) = 0$). Therefore, γ needs to be an imaginary number, which means that the conditions for stability are

$$J_z > J_x, J_y \quad \text{or} \quad J_z < J_x, J_y$$

In words, if a body is spinning about its axis of minimum or maximum moment of inertia, i.e., if γ is a pure imaginary, then the angular motion is described by an harmonic oscillator, implying neutral stability along these axes. But suppose that the body spins about the axis with intermediate value of moment of inertia and so the previous condition is not satisfied. This means that

$$J_x > J_z > J_y \quad \text{or} \quad J_x < J_z < J_y$$

In this case, γ is a real number, hence one of the roots of $\lambda(s)$ is a positive real pole ($\mathrm{Real}(s) > 0$), thus implying instability for this specific condition.

These conclusions hold for the trivial case of free dynamics ($\boldsymbol{\tau} = 0$), where the body momentum and the rotational kinetic energy are both constant. In reality, these assumptions do not generally hold true, as there are parasitic and non-modeled torques that dissipate energy, possibly leading to instability along the minor axis of inertia [8]. Even so, the principle of spin stability is robust if considering the principal major axis of inertia and still holds true for this situation. Therefore, the major axis solution is applied in this dissertation.

Chapter 4

Active Spin Stabilization

All the preceding chapters laid out the fundamental theory required to define the spin stabilization problem. Now, it is necessary to find an appropriate control solution for this unique space application. To develop the GNC system, both a navigation filter for estimation, a guidance algorithm to determine the reference, and a control law to command the spacecraft to the desired reference must be designed.

In the particular case in study, the state estimate given by the navigation filter should also include the rotation angles from the initial spin axis to the major inertia axis, and the principal moments of inertia. On the other hand, the relation between Z_b and Z_j axes can be deduced from the spacecraft dynamics properties. From that evaluation, the controller is designed to provide the required control torques to orient the spacecraft according to the principal major axis of inertia. For this specific GNC system, the proposed solution is also known as *active spin stabilization*.

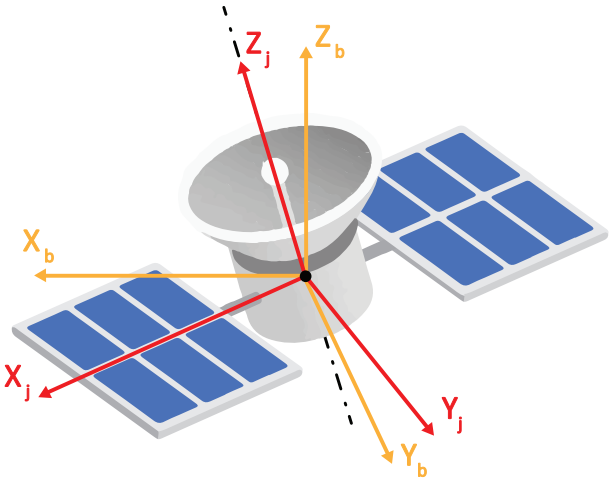


Figure 4.1: Spacecraft body frame $\mathcal{B}$ and principal axes of inertia reference frame $\mathcal{J}$

Figure 4.1 illustrates two body-centered and vehicle-carried coordinate systems - the spacecraft body frame $\mathcal{B}$ (in yellow) and the principal axes of inertia reference frame $\mathcal{J}$ (in red). Pragmatically, the active spin stabilization system is a GNC solution that takes an initial angular velocity coincident with the Z_b-axis, computes the two angles between the Z-axes or, alternatively, finds a relation between the relevant axes, and uses them to rotate the angular velocity vector and aligned it with the Z_j-axis.

This Chapter documents the theory underlying the active spin stabilization method developed in this study. In Section 4.1, the Kalman filter is introduced, including the linear Kalman filter deduction and, subsequently, the design of an Extended Kalman Filter (EKF). Finally, Section 4.2 presents the torque proportional-integral controller and two alternatives for dynamic reference guidance, with the first controller being later analyzed for Lyapunov stability.

4.1 State Estimation Methods

State estimation refers to the process of finding the best estimate of the true system state from a dynamic model and a set of measured observations, both corrupted with random noise of known statistics. The variables to be estimated form the so-called *state vector*, which typically includes the attitude amongst other variables [11]. As introduced in Chapter 1, the methods for determining the orientation of a space vehicle can be divided into static attitude determination methods, which include deterministic mechanisms using at least two sensor measurements at any given time; and recursive attitude determination methods, which are able to provide information when less than two observations are available using stochastic mechanisms.

There are essentially three main reasons to propose stochastic system models over deterministic ones. Firstly, no mathematical model is perfect. Hence, every model includes only the characteristics that are of interest for the application, where several parameters are left unmodeled, leading to sources of uncertainty. Secondly, dynamic systems are driven not only by control inputs but also by disturbances which cannot be controlled nor modeled deterministically. Lastly, sensors do not provide perfect and complete data about a system [46].

For these reasons, frequently, a filter is implemented to infer the state of a system that cannot be measured directly. Filtering can generally provide a more accurate attitude estimate than static methods as it is a recursive method and incorporates memory of past observations.

This section describes Kalman filter theory, which is the basis for the estimation algorithm considered in this work, discussing the concepts of state and observation models, and the formulation of the linear discrete-time and continuous-time Kalman filters. Afterwards, the Extended Kalman filter (EKF) is presented, which is the most prominent application of the Kalman filter to nonlinear systems and later implemented as part of the control system in Chapter 5.

4.1.1 Kalman Filtering

State and Observation Dynamic Models

In order to determine and analyze the behavior of a system, it is fundamental to design a static or dynamic model on which estimation algorithms can be implemented. Dynamic models are defined by differential equations and the system's condition is described by the state vector $\boldsymbol{x} \in \mathbb{R}^n$. This vector is composed by the state variables, whose number defines the order of the system. The general structure

of an n^{th}-order dynamic model is provided below.

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) \quad , \quad \boldsymbol{x}(t_0) = \boldsymbol{x}_0 \tag{4.1}$$

where $\boldsymbol{f}$ is sufficiently differentiable and $\boldsymbol{u} \in \mathbb{R}^m$ denotes the input to the system independent from the state elements.

These dynamic models can be classified as linear systems or nonlinear systems. In case the system is linear, it follows the superposition principle and can be represented as:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}(t)\boldsymbol{x}(t) + \boldsymbol{B}(t)\boldsymbol{u}(t) \quad , \quad \boldsymbol{x}(t_0) = \boldsymbol{x}_0 \tag{4.2}$$

where the matrices $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ and $\boldsymbol{B} \in \mathbb{R}^{n\times m}$ are known as the state matrix and the input matrix, respectively.

On the other hand, observations are gathered as outputs from the sensors. The general form of an observation vector $\boldsymbol{y} \in \mathbb{R}^p$, used to show how a sensor relates to various state quantities and inputs, is given by:

$$\boldsymbol{y}(t) = \boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) \tag{4.3}$$

Likewise, linear observation models follow the same structure as linear state models:

$$\boldsymbol{y}(t) = \boldsymbol{C}(t)\boldsymbol{x}(t) + \boldsymbol{D}(t)\boldsymbol{u}(t) \tag{4.4}$$

where $\boldsymbol{C} \in \mathbb{R}^{p\times n}$ is the observability matrix and $\boldsymbol{D} \in \mathbb{R}^{p\times m}$ is called the direct transmission matrix. The set of equations (4.2) and (4.4) form the state space description of a continuous-time linear model [11].

However, nowadays, discrete-time systems have become the standard in most dynamic applications with the use of digital computers. Thus, a discrete-time representation for the dynamic state and observation models is derived. For that, first the system must be sampled to a constant time interval Δt. As $\boldsymbol{A}$ and $\boldsymbol{B}$ stay constant (or slowly varying), and setting the time for the first sample interval, the equation (4.2) becomes

$$\boldsymbol{x}(\Delta t) = \boldsymbol{\Phi}\boldsymbol{x}(0) + \boldsymbol{\Gamma}\boldsymbol{u}(0) \tag{4.5}$$

where

$$\boldsymbol{\Phi} \equiv e^{A\Delta t} \qquad \text{and} \qquad \boldsymbol{\Gamma} \equiv \left[\int_0^{\Delta t} e^{\boldsymbol{F}(t)}dt\right]\boldsymbol{B}$$

Expanding the equation (4.5) for $k+1$ samples, in order to include the entire discrete state space, yields

$$\boldsymbol{x}_{k+1} = \boldsymbol{\Phi}\boldsymbol{x}_k + \boldsymbol{\Gamma}\boldsymbol{u}_k \tag{4.6}$$

with $\boldsymbol{x}_k$ being the state vector at time t_k.

Equation (4.4) may also be represented in discrete time as

$$\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k + \boldsymbol{D}\boldsymbol{u}_k \tag{4.7}$$

Derivation of the Discrete-time Kalman Filter

A Kalman filter (KF) is an optimal recursive data processing algorithm [46]. It is recursive in the sense that it does not require all previous data to keep running, only the previous state and measurement, and operates by propagating the mean and covariance of the state through time. The approach to derive the Kalman filter involves the following steps.

1. Start with a mathematical description of a dynamic system whose states are to be estimated (i.e., dynamic model).

2. Implement the equations that describe how the mean of the state and the covariance of the state propagate with time.

3. Take the dynamic system that describes the propagation of the state mean and covariance, and implement the equations on a computer.

4. Every time that a new measurement is obtained, update the mean and covariance of the state.

These instructions form the basis for the derivation of the Kalman filter because the mean of the state is the Kalman filter estimate of the state, and the covariance of the state is the covariance of the Kalman filter state estimate. It should also be noted that both the discrete-time and continous-time KFs derivations are based on the formulations described by Dan Simon [47].

Supposing that a linear discrete-time system is given as follows:

$$\begin{cases} \boldsymbol{x}_k = \boldsymbol{F}_{k-1}\boldsymbol{x}_{k-1} + \boldsymbol{G}_{k-1}\boldsymbol{u}_{k-1} + \boldsymbol{w}_{k-1} \\ \boldsymbol{y}_k = \boldsymbol{H}_k\boldsymbol{x}_k + \boldsymbol{v}_k \end{cases} \tag{4.8}$$

The noise processes $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$ are white, zero-mean, uncorrelated, and have known covariance matrices $\boldsymbol{Q}_k$ and $\boldsymbol{R}_k$, respectively:

$$\begin{cases} \boldsymbol{w}_k \sim (0, \boldsymbol{Q}_k) \\ \boldsymbol{v}_k \sim (0, \boldsymbol{R}_k) \\ E[\boldsymbol{w}_k \boldsymbol{w}_j^T] = \boldsymbol{Q}_k \delta_{k-j} \\ E[\boldsymbol{v}_k \boldsymbol{v}_j^T] = \boldsymbol{R}_k \delta_{k-j} \\ E[\boldsymbol{w}_k \boldsymbol{v}_j^T] = 0 \end{cases} \tag{4.9}$$

where δ_{k-j} is the Kronecker delta function ($\delta_{k-j} = 1$, if $k = j$, and $\delta_{k-j} = 0$, if $k \neq j$). The goal is to estimate the state $\boldsymbol{x}_k$ based on knowledge of the system dynamics and the availability of the measurements $\boldsymbol{y}_k$.

The amount of information that is available for the state estimate varies depending on the particular problem at hand. If all of the measurements up to and including time k are available for use in the estimate $\boldsymbol{x}_k$, then it can form an *a posteriori* estimate, which is denoted as $\hat{\boldsymbol{x}}_k^+$. One way to set the *a posteriori* state estimate is to compute the expected value of $\boldsymbol{x}_k$ conditioned on all of the measurements

up to and including time k:

$$\hat{\boldsymbol{x}}_k^+ = E[\boldsymbol{x}_k|\boldsymbol{y}_1, \boldsymbol{y}_2, ..., \boldsymbol{y}_k] \tag{4.10}$$

If all of the measurements before (but not including) time k are available for use in the estimate of x_k, then it can form an *a priori* estimate, which is denoted as $\hat{x}_k^-$. An option to set the *a priori* state estimate is to compute the expected value of $\boldsymbol{x}_k$ conditioned on all of the measurements before (but not including) time k:

$$\hat{\boldsymbol{x}}_k^- = E[\boldsymbol{x}_k|\boldsymbol{y}_1, \boldsymbol{y}_2, ..., \boldsymbol{y}_{k-1}] \tag{4.11}$$

It is important to note that $\hat{\boldsymbol{x}}_k^+$ and $\hat{\boldsymbol{x}}_k^-$ are both estimates of the same quantity - $\boldsymbol{x}_k$. However, $\hat{\boldsymbol{x}}_k^-$ is an estimate of $\boldsymbol{x}_k$ *before* the measurement $\boldsymbol{y}_k$ is taken into account, while $\hat{\boldsymbol{x}}_k^+$ is an estimate of $\boldsymbol{x}_k$ *after* the measurement $\boldsymbol{y}_k$ is taken into account. Therefore, it is naturally expected that $\hat{\boldsymbol{x}}_k^+$ will be a better estimate than $\hat{\boldsymbol{x}}_k^-$, because more information is used to compute it.

The first measurement is taken at time $k = 1$. Since no measurements are available to estimate $\boldsymbol{x}_0$, it is reasonable to form $\hat{\boldsymbol{x}}_0^+$ as the expected value of the initial state $\boldsymbol{x}_0$, i.e.:

$$\hat{\boldsymbol{x}}_0^+ = E[\boldsymbol{x}_0] \tag{4.12}$$

The term $\boldsymbol{P}_k$ is used to denote the covariance of the estimation error at time t_k. The covariance of the estimation error of $\hat{\boldsymbol{x}}_k^-$ is denoted as $\boldsymbol{P}_k^-$, and the covariance of the estimation error of $\hat{\boldsymbol{x}}_k^+$ is denoted as $\boldsymbol{P}_k^+$, and given by

$$\begin{cases} \boldsymbol{P}_k^- = E[(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^-)(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^-)^T] \\ \boldsymbol{P}_k^+ = E[(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^+)(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^+)^T] \end{cases} \tag{4.13}$$

The estimation process begins with $\hat{\boldsymbol{x}}_0^+$, considered the best estimate of the initial state $\boldsymbol{x}_0$. Given $\hat{\boldsymbol{x}}_0^+$, now $\hat{\boldsymbol{x}}_1^-$ can be computed to refine the estimate. Recalling from equation (4.8) how the mean of $\boldsymbol{x}$ propagates with time, yields

$$\hat{\boldsymbol{x}}_1^- = \boldsymbol{F}_0\hat{\boldsymbol{x}}_0^+ + \boldsymbol{G}_0\boldsymbol{u}_0 \tag{4.14}$$

This is a specific equation that shows how to obtain $\hat{x}_1^-$ from $\hat{x}_0^+$. However, the reasoning can be extended to obtain the following general equation:

$$\hat{\boldsymbol{x}}_k^- = \boldsymbol{F}_{k-1}\hat{\boldsymbol{x}}_{k-1}^+ + \boldsymbol{G}_{k-1}\boldsymbol{u}_{k-1} \tag{4.15}$$

This is called the *time-update equation for* $\hat{\boldsymbol{x}}$. At this point, no additional measurements are available to help update the state estimate between time $(k-1)^+$ and $(k)^-$, so the state estimate should be updated based on the system dynamics.

Promptly, the time-update equation for the covariance of the state estimation error $\boldsymbol{P}$ should be computed, starting from $\boldsymbol{P}_0^+$, which is the covariance of the initial estimate of $\boldsymbol{x}_0$. If the initial state is perfectly known, then $\boldsymbol{P}_0^+ = 0$. If the value of $\boldsymbol{x}_0$ is unknown, then a much larger value is chosen for $\boldsymbol{P}_0^+$.

In general, $\boldsymbol{P_0^+}$ represents the uncertainty in the initial estimate of $\boldsymbol{x}_0$:

$$\boldsymbol{P}_0^+ = E[(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0^+)(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0^+)^T] \tag{4.16}$$

Given $\boldsymbol{P}_0^+$, now $\boldsymbol{P}_1^-$ can be computed according to the expression below

$$\boldsymbol{P}_1^- = \boldsymbol{F}_0 \boldsymbol{P}_0^+ \boldsymbol{F}_0^T + \boldsymbol{Q}_0 \tag{4.17}$$

This is a specific equation that shows how to obtain $\boldsymbol{P}_1^-$ from $\boldsymbol{P}_0^+$. However, the reasoning can be extended to obtain the following general equation:

$$\boldsymbol{P}_k^- = \boldsymbol{F}_{k-1} \boldsymbol{P}_{k-1}^+ \boldsymbol{F}_{k-1}^T + \boldsymbol{Q}_{k-1} \tag{4.18}$$

This is called the *time-update equation for* $\boldsymbol{P}$.

At the moment, both the time-update equations for $\hat{\boldsymbol{x}}$ and $\boldsymbol{P}$ have been obtained. Now the measurement-update equations for $\hat{\boldsymbol{x}}$ and $\boldsymbol{P}$ must be derived. The only difference between $\hat{\boldsymbol{x}}_k^-$ and $\hat{\boldsymbol{x}}_k^+$ is that the latter takes the measurement $\boldsymbol{y}_k$ into account. From the recursive least squares method, while adapting and relating the estimates and covariance to Kalman filtering, the availability of the measurement $\boldsymbol{y}_k$ changes the estimate of a constant $\boldsymbol{x}$ as follows:

$$\begin{cases} \boldsymbol{K}_k = \boldsymbol{P}_k^- \boldsymbol{H}_k^T (\boldsymbol{H}_k \boldsymbol{P}_k^- \boldsymbol{H}_k^T + \boldsymbol{R}_k)^{-1} \\ \hat{\boldsymbol{x}}_k^+ = \hat{\boldsymbol{x}}_k^- + \boldsymbol{K}_k (\boldsymbol{y}_k - \boldsymbol{H}_k \hat{\boldsymbol{x}}_k^-) \\ \boldsymbol{P}_k^+ = (\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{H}_k) \boldsymbol{P}_k^- \end{cases} \tag{4.19}$$

These are the *measurement-update equations for* $\hat{\boldsymbol{x}}_k$ *and* $\boldsymbol{P}_k$. The matrix $\boldsymbol{K}_k$ in the equations above is called *Kalman filter gain* and is determined by minimizing the trace of the updated covariance $\boldsymbol{P}_k^+$.

Table 4.1 summarizes the discrete-time Kalman filter formulation.

Table 4.1: Discrete-time Kalman Filter formulation

Discrete-time KF		
Model	$\boldsymbol{x}_k = \boldsymbol{F}_{k-1}\boldsymbol{x}_{k-1} + \boldsymbol{G}_{k-1}\boldsymbol{u}_{k-1} + \boldsymbol{w}_{k-1},$ $\boldsymbol{y}_k = \boldsymbol{H}_k \boldsymbol{x}_k + \boldsymbol{v}_k,$	with $\boldsymbol{w}_k \sim (0, \boldsymbol{Q}_k)$ with $\boldsymbol{v}_k \sim (0, \boldsymbol{R}_k)$
Initial condition	$\hat{\boldsymbol{x}}_0^+ = E[\boldsymbol{x}_0]$ $\boldsymbol{P}_0^+ = E[(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0^+)(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0^+)^T]$	
Propagation	$\hat{\boldsymbol{x}}_k^- = \boldsymbol{F}_{k-1}\hat{\boldsymbol{x}}_{k-1}^+ + \boldsymbol{G}_{k-1}\boldsymbol{u}_{k-1}$ $\boldsymbol{P}_k^- = \boldsymbol{F}_{k-1}\boldsymbol{P}_{k-1}^+\boldsymbol{F}_{k-1}^T + \boldsymbol{Q}_{k-1}$	
Gain	$\boldsymbol{K}_k = \boldsymbol{P}_k^- \boldsymbol{H}_k^T (\boldsymbol{H}_k \boldsymbol{P}_k^- \boldsymbol{H}_k^T + \boldsymbol{R}_k)^{-1}$	
Update	$\hat{\boldsymbol{x}}_k^+ = \hat{\boldsymbol{x}}_k^- + \boldsymbol{K}_k(\boldsymbol{y}_k - \boldsymbol{H}_k\hat{\boldsymbol{x}}_k^-)$ $\boldsymbol{P}_k^+ = (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H}_k)\boldsymbol{P}_k^-$	

Derivation of the Continuous-time Kalman Filter

At once, assuming that a linear continuous-time system is supposedly given as

$$\begin{cases} \dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} + \boldsymbol{w} \\ \boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{v} \\ \boldsymbol{w} \sim (0, \boldsymbol{Q}_c) \\ \boldsymbol{v} \sim (0, \boldsymbol{R}_c) \end{cases} \tag{4.20}$$

If the system were to be discretized with a sample of time t in order to obtain the discrete-time space state description provided in equation (4.8), the matrices in that system would be computed as follows:

$$\begin{cases} \boldsymbol{F} = \exp(\boldsymbol{A}t) \approx (\boldsymbol{I} + \boldsymbol{A}t) & \text{for small } t \\ \boldsymbol{G} = (\exp(\boldsymbol{A}t) - \boldsymbol{I})\boldsymbol{A}^{-1}\boldsymbol{B} \approx \boldsymbol{B}t & \text{for small } t \\ \boldsymbol{H} = \boldsymbol{C} \\ \boldsymbol{w}_k \sim (0, \boldsymbol{Q}), \quad \boldsymbol{Q} = \boldsymbol{Q}_c t \\ \boldsymbol{v}_k \sim (0, \boldsymbol{R}), \quad \boldsymbol{R} = \boldsymbol{R}_c / t \end{cases} \tag{4.21}$$

from the derivative formula and where $\boldsymbol{I}$ is the identity matrix.

Recalling the discrete-time Kalman filter gain (first expression in equation (4.19)), the continuous-time Kalman filter gain can be derived as

$$\begin{aligned} \boldsymbol{K} &= \lim_{t \to 0} \frac{\boldsymbol{K}_k}{t} \\ &= \lim_{t \to 0} \boldsymbol{P}_k^- \boldsymbol{C}^T (\boldsymbol{C}\boldsymbol{P}_k^- \boldsymbol{C}^T t + \boldsymbol{R}_c)^{-1} \\ &= \boldsymbol{P}_k^- \boldsymbol{C}^T \boldsymbol{R}_c^{-1} \end{aligned} \tag{4.22}$$

Similarly, the estimation-error covariance update equation (4.18) can be defined for small values of t

$$\begin{aligned} \boldsymbol{P}_{k+1}^- &= (\boldsymbol{I} + \boldsymbol{A}t)\boldsymbol{P}_k^+ (\boldsymbol{I} + \boldsymbol{A}t)^T + \boldsymbol{Q}_c t \\ &= \boldsymbol{P}_k^+ + (\boldsymbol{A}\boldsymbol{P}_k^+ + \boldsymbol{P}_k^+ \boldsymbol{A}^T + \boldsymbol{Q}_c)t + \boldsymbol{A}\boldsymbol{P}_k^+ \boldsymbol{A}^T t^2 \end{aligned} \tag{4.23}$$

Substituting for $\boldsymbol{P}_k^+$ (last expression in equation (4.19)), subtracting $\boldsymbol{P}_k^-$ from both sides and then dividing by t gives

$$\begin{aligned} \frac{\boldsymbol{P}_{k+1}^- - \boldsymbol{P}_k^-}{t} &= \frac{(\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{C})\boldsymbol{P}_k^- + \boldsymbol{A}\boldsymbol{P}_k^+ \boldsymbol{A}^T t^2}{t} \\ &\quad + \frac{[\boldsymbol{A}(\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{C})\boldsymbol{P}_k^- + (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{C})\boldsymbol{P}_k^- \boldsymbol{A}^T + \boldsymbol{Q}_c]t}{t} - \boldsymbol{P}_k^- \\ &= \frac{-\boldsymbol{K}_k\boldsymbol{C}\boldsymbol{P}_k^-}{t} + \boldsymbol{A}\boldsymbol{P}_k^+ \boldsymbol{A}^T t \\ &\quad + (\boldsymbol{A}\boldsymbol{P}_k^- + \boldsymbol{A}\boldsymbol{K}_k\boldsymbol{C}\boldsymbol{P}_k^- + \boldsymbol{P}_k^- \boldsymbol{A}^T - \boldsymbol{K}_k\boldsymbol{C}\boldsymbol{P}_k^- \boldsymbol{A}^T + \boldsymbol{Q}_c) \end{aligned} \tag{4.24}$$

Taking the limit $t \to 0$ and using equation (4.22), yields

$$\begin{aligned}\dot{\boldsymbol{P}} &= \lim_{t\to 0} \frac{\boldsymbol{P}_{k+1}^{-} - \boldsymbol{P}_{k}^{-}}{t} \\ &= -\boldsymbol{P}\boldsymbol{C}^T\boldsymbol{R}_c^{-1}\boldsymbol{C}\boldsymbol{P} + \boldsymbol{A}\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A}^T + \boldsymbol{Q}_c\end{aligned} \tag{4.25}$$

which is known as a *differential Riccati equation* and can be used to compute the estimation-error covariance for the continuous-time Kalman filter.

In the previous subsection, the Kalman filter equations for $\hat{\boldsymbol{x}}$ were derived. Now, assuming that t is small, equations (4.15) and (4.19) can be used to rewrite the measurement-update equation as

$$\begin{aligned}\hat{\boldsymbol{x}}_k^{+} &= \boldsymbol{F}\hat{\boldsymbol{x}}_{k+1}^{+} + \boldsymbol{G}\boldsymbol{u}_{k-1} + \boldsymbol{K}_k(\boldsymbol{y}_k - \boldsymbol{H}\boldsymbol{F}\hat{\boldsymbol{x}}_{k+1}^{+} - \boldsymbol{H}\boldsymbol{G}\boldsymbol{u}_{k-1}) \\ &= (\boldsymbol{I} + \boldsymbol{A}t)\hat{\boldsymbol{x}}_{k-1}^{+} + \boldsymbol{B}t\boldsymbol{u}_{k-1} + \boldsymbol{K}_k[\boldsymbol{y}_k - \boldsymbol{C}(\boldsymbol{I} + \boldsymbol{A}t)\hat{\boldsymbol{x}}_{k-1}^{+} - \boldsymbol{C}\boldsymbol{B}t\boldsymbol{u}_{k-1}]\end{aligned} \tag{4.26}$$

Now substituting for $\boldsymbol{K}_k$ from equation (4.22) and subtracting $\hat{\boldsymbol{x}}_{k-1}^{+}$, it gives

$$\begin{aligned}\hat{\boldsymbol{x}}_k^{+} - \hat{\boldsymbol{x}}_{k-1}^{+} = \boldsymbol{A}t\hat{\boldsymbol{x}}_{k-1}^{+} + \boldsymbol{B}t\boldsymbol{u}_{k-1} \\ + \boldsymbol{P}\boldsymbol{C}^T\boldsymbol{R}_c^{-1}t(\boldsymbol{y}_k - \boldsymbol{C}\hat{\boldsymbol{x}}_{k-1}^{+} - \boldsymbol{C}\boldsymbol{A}t\hat{\boldsymbol{x}}_{k-1}^{+} - \boldsymbol{C}\boldsymbol{B}t\boldsymbol{u}_{k-1})\end{aligned} \tag{4.27}$$

Finally, dividing by t and solving for the limit $t \to 0$, results in

$$\begin{aligned}\dot{\hat{\boldsymbol{x}}} &= \lim_{t\to 0} \frac{\hat{\boldsymbol{x}}_k^{+} - \hat{\boldsymbol{x}}_{k-1}^{+}}{t} \\ &= \boldsymbol{A}\hat{\boldsymbol{x}} + \boldsymbol{B}\boldsymbol{u} + \boldsymbol{K}(\boldsymbol{y} - \boldsymbol{C}\hat{\boldsymbol{x}})\end{aligned} \tag{4.28}$$

where $\boldsymbol{K}$ is defined as the Kalman gain in equation (4.22) and the differential equation can be used to integrate the state estimate [48]. These continuous-time Kalman filter equations are summarized in Table 4.2.

Table 4.2: Continuous-time Kalman Filter formulation

Continuous-time KF		
Model	$\dot{\boldsymbol{x}} = \boldsymbol{A}x + \boldsymbol{B}\boldsymbol{u} + \boldsymbol{w}$, $\boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{v}$,	with $\boldsymbol{w} \sim (0, \boldsymbol{Q}_k)$ with $\boldsymbol{v} \sim (0, \boldsymbol{R}_k)$
Initial condition	$\hat{\boldsymbol{x}}(0) = E[\boldsymbol{x}(0)]$ $\boldsymbol{P}(0) = E[(\boldsymbol{x}(0) - \hat{\boldsymbol{x}}(0))(\boldsymbol{x}(0) - \hat{\boldsymbol{x}}(0))^T]$	
Gain	$\boldsymbol{K} = \boldsymbol{P}_k^{-}\boldsymbol{C}^T\boldsymbol{R}_c^{-1}$	
Propagation	$\dot{\hat{\boldsymbol{x}}} = \boldsymbol{A}\hat{\boldsymbol{x}} + \boldsymbol{B}\boldsymbol{u} + \boldsymbol{K}(\boldsymbol{y} - \boldsymbol{C}\hat{\boldsymbol{x}})$ $\dot{\boldsymbol{P}} = -\boldsymbol{P}\boldsymbol{C}^T\boldsymbol{R}_c^{-1}\boldsymbol{C}\boldsymbol{P} + \boldsymbol{A}\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A}^T + \boldsymbol{Q}_c$	

4.1.2 Extended Kalman Filter

One of the assumptions made for the development of the Kalman filter in the previous section was that the system could be described by a linear model. However, a large class of estimation problems involve non-linearities and, if a system is modeled such that it takes them into account, then the state transition function becomes nonlinear. Thus, it implies that the probability density function of the noise is altered as it is transmitted through the nonlinear element, i.e., a Gaussian input does not guarantee a Gaussian output, as in the linear case.

For these situations, a nonlinear state estimator needs to be used instead of a Kalman filter, as Kalman filters are only defined for linear systems. Fortunately, many nonlinear systems can be approximated locally by a linear one. That is, the functional dependence of the state dynamics and/or the measurements are nonlinear, but approximately linear for small disturbances.

In this case, the most appropriate solution can be an Extended Kalman filter (EKF), which linearizes the nonlinear function around the mean of the current state estimate. At each time step, the linearization is performed locally and the resulting Jacobian matrices are then used in the propagation and update states of the Kalman filter algorithm [48].

Considering the general continuous-time nonlinear model described as

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) + \boldsymbol{w}(t) \quad , \quad \boldsymbol{w}(t) \sim (0, \boldsymbol{Q}(t)) \\ \boldsymbol{y}(t) &= \boldsymbol{h}(\boldsymbol{x}(t), t) + \boldsymbol{v}(t) \quad , \quad \boldsymbol{v}(t) \sim (0, \boldsymbol{R}(t)) \end{aligned} \tag{4.29}$$

where $\boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t)$ and $\boldsymbol{h}(\boldsymbol{x}(t), t)$ are continuously differentiable nonlinear functions.

The EKF's underlying premise regarding the non-linearity problem is to assume the true state is sufficiently close to the estimate state, hence the error dynamics can be represented by a linearized first-order Taylor series expansion [48].

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

$$\begin{aligned} \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) &\approx \boldsymbol{f}(\bar{\boldsymbol{x}}(t), \boldsymbol{u}(t), t) + \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\Big|_{\bar{\boldsymbol{x}}(t), \boldsymbol{u}(t)} [\boldsymbol{x}(t) - \bar{\boldsymbol{x}}(t)] \\ \boldsymbol{h}(\boldsymbol{x}(t), t) &\approx \boldsymbol{h}(\bar{\boldsymbol{x}}(t), t) + \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}\Big|_{\bar{\boldsymbol{x}}(t)} [\boldsymbol{x}(t) - \bar{\boldsymbol{x}}(t)] \end{aligned} \tag{4.30}$$

where the denoted partial derivative with respect to $\boldsymbol{x}$ is evaluated at $\bar{\boldsymbol{x}}(t)$.

In the Extended Kalman Filter, the current estimate $\hat{\boldsymbol{x}}$ is used for the nominal state estimate $\bar{\boldsymbol{x}}$, so that $\hat{\boldsymbol{x}} = \bar{\boldsymbol{x}}$. Thus, rewriting the equations for their expected value on both sides, with $\bar{\boldsymbol{x}} = \hat{\boldsymbol{x}}$, yields

$$\begin{aligned} E[\boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t)] &= \boldsymbol{f}(\bar{\boldsymbol{x}}(t), \boldsymbol{u}(t), t) \\ E[\boldsymbol{h}(\boldsymbol{x}(t), t)] &= \boldsymbol{h}(\bar{\boldsymbol{x}}(t), t) \end{aligned} \tag{4.31}$$

Therefore, recalling the discrete-time Kalman filter propagation (4.28), the state and output estimate

equations for the EKF are given by

$$\begin{aligned} \dot{\hat{\boldsymbol{x}}} &= \boldsymbol{f}(\hat{\boldsymbol{x}}, u, t) + \boldsymbol{K}[\boldsymbol{y} - \boldsymbol{h}(\hat{\boldsymbol{x}}, t)] \\ \boldsymbol{y} &= \boldsymbol{h}(\hat{\boldsymbol{x}}, t) \end{aligned} \tag{4.32}$$

where $\boldsymbol{K}$ is the Kalman filter gain (4.22), derived in the previous subsection.

Regarding the estimation-error covariance equation, likewise, equation (4.25) from the continuous-time Kalman filter holds true, except for matrix $\boldsymbol{A}$, since it is replaced by the jacobian matrix $\boldsymbol{F}$, obtained from the partial derivative evaluated at the current state estimate as

$$\boldsymbol{F} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\Big|_{\hat{\boldsymbol{x}}} \tag{4.33}$$

At last, Table 4.3 summarizes the equations presented above, as part of the EKF formulation. It is interesting to note that the formulation for both the continuous-time Kalman filter and Extended Kalman filter conform to the same structure and expressions, understandably since the EKF is simply a local linearization of a nonlinear system, followed by the KF algorithm. On the other hand, it is also important to note that, contrarily to the KF for linear systems, the Extended Kalman Filter does not guarantee stability, but is still widely implemented in the scientific community, as it yields precise, accurate and stable estimates for a great majority of systems.

Table 4.3: Continuous-time Extended Kalman Filter formulation

Continuous-time EKF		
Model	$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{w}, t)$, $\boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{v}, t)$,	with $\boldsymbol{w} \sim (0, \boldsymbol{Q}_k)$ with $\boldsymbol{v} \sim (0, \boldsymbol{R}_k)$
Initial condition	$\hat{\boldsymbol{x}}(0) = E[\boldsymbol{x}(0)]$ $\boldsymbol{P}(0) = E[(\boldsymbol{x}(0) - \hat{\boldsymbol{x}}(0))(\boldsymbol{x}(0) - \hat{\boldsymbol{x}}(0))^T]$	
Gain	$\boldsymbol{K} = \boldsymbol{P}\boldsymbol{C}^T\boldsymbol{R}_c^{-1}$	
Propagation	$\dot{\hat{\boldsymbol{x}}} = \boldsymbol{f}(\hat{\boldsymbol{x}}, \boldsymbol{u}, \boldsymbol{w}_0, t) + \boldsymbol{K}(\boldsymbol{y} - \boldsymbol{h}(\hat{\boldsymbol{x}}, \boldsymbol{v}_0, t))$ $\dot{\boldsymbol{P}} = -\boldsymbol{P}\boldsymbol{C}^T\boldsymbol{R}_c^{-1}\boldsymbol{C}\boldsymbol{P} + \boldsymbol{F}\boldsymbol{P} + \boldsymbol{P}\boldsymbol{F}^T + \boldsymbol{Q}_c$	with $\boldsymbol{w}_0 = 0$ and $\boldsymbol{v}_0 = 0$

4.2 Control Design Methods

Spacecraft attitude control is essential for a satellite to comply with mission pointing criteria, including position and attitude requirements. As stated in Chapter 1, early spacecraft designs used passive control methods, i.e., spin stabilization, gravity-gradient stabilization or a momentum wheel, which were mostly implemented due to their limited control actuation and lack of sophisticated computer technology for more complex control laws.

In recent times, control theory has been extensively studied and advanced, with multiple control algorithms present in current literature, such as the Linear Quadratic Regulator, Sliding Mode Control

and Model Predictive Control. Although there are now many different purposes for attitude controllers, adapting the most elemental passive solutions to be part of a modern active control method can still promote great advantages in any guidance, navigation and control (GNC) system. That is the case for spin stabilization GNC systems, as commanding the spacecraft to rotate around its principal major axis of inertia can reduce fuel consumption and, thus, increase its efficiency and mission time duration [11].

For active spin stabilization, there is no need to include an elaborate control law, as a simple proportional-integral controller can take the angular velocity estimate and compute the demanded control torque. However, the desired angular velocity reference can be considered more difficult to calculate, as addressing this elaborate task requires deducing an expression to relate the estimated rotation angles to the angular velocity.

This section defines the proportional-integral control law for the command torque, that is later transmitted to the spacecraft actuators. Additionally, two different approaches for the guidance algorithm are presented, providing both an estimate-based and a control-based dynamic reference. At last, the fundamentals of Lyapunov theory are introduced, whose conditions are applied to evaluate the nonlinear control functions stability.

4.2.1 Torque Proportional-Integral Controller

Proportional-Integral (PI) control is a type of linear feedback control system in which the process error, i.e., the difference between the desired value and the measured value, is continuously calculated and applies a correction based on proportional and integral terms to the controlled variable.

The general equation for a proportional-integral controller in a three-dimensional system ($\boldsymbol{x}, \boldsymbol{u} \in \mathbb{R}^3$) is given by

$$\boldsymbol{u}(t) = \boldsymbol{K}_p \boldsymbol{e}(t) + \boldsymbol{K}_i \int_0^t e(\tau) d\tau \tag{4.34}$$

where $\boldsymbol{u}(t) \in \mathbb{R}^3$ is the controller output at time t, $\boldsymbol{K}_p \in \mathbb{R}^{3\times3}$ is the proportional gain, $\boldsymbol{K}_i \in \mathbb{R}^{3\times3}$ is the integral gain, and $\boldsymbol{e}(t) \in \mathbb{R}^3$ is the instantaneous process error at time t, i.e.,

$$\boldsymbol{e}(t) = \boldsymbol{r}(t) - \boldsymbol{y}(t) \tag{4.35}$$

with $\boldsymbol{r}(t) \in \mathbb{R}^3$ as the reference or desired value at time t, and $\boldsymbol{y}(t) \in \mathbb{R}^3$ as the measured value at time t.

This definition can be extended for the dynamics, and used to design a torque PI controller. Recalling equation (2.27), and solving in order to $\dot{\boldsymbol{\omega}}$, yields

$$\dot{\boldsymbol{\omega}} = -\boldsymbol{J}^{-1}(\boldsymbol{\omega} \times \boldsymbol{J}\boldsymbol{\omega}) + \boldsymbol{J}^{-1}\boldsymbol{\tau} \tag{4.36}$$

Then, to obtain an expression similar to equation (4.34), the external torque $\boldsymbol{\tau}$ should be defined as

$$\boldsymbol{\tau} = \boldsymbol{\omega} \times \boldsymbol{J}\boldsymbol{\omega} + \boldsymbol{K}_p \boldsymbol{\omega}_e + \boldsymbol{K}_i \int \boldsymbol{\omega}_e \tag{4.37}$$

and, substituting this result on equation (4.36), the spacecraft attitude dynamics can be described as

$$\begin{aligned}\dot{\boldsymbol{\omega}} &= -\boldsymbol{J}^{-1}(\boldsymbol{\omega}\times\boldsymbol{J}\boldsymbol{\omega}) + \boldsymbol{J}^{-1}\left[\boldsymbol{\omega}\times\boldsymbol{J}\boldsymbol{\omega} + \boldsymbol{K}_p\boldsymbol{\omega}_e + \boldsymbol{K}_i\int\boldsymbol{\omega}_e\right] \\ &= -\boldsymbol{J}^{-1}(\boldsymbol{\omega}\times\boldsymbol{J}\boldsymbol{\omega} - \hat{\boldsymbol{\omega}}\times\boldsymbol{J}\boldsymbol{\omega}) + \boldsymbol{J}^{-1}\boldsymbol{K}_p\boldsymbol{\omega}_e + \boldsymbol{J}^{-1}\boldsymbol{K}_i\int\boldsymbol{\omega}_e \\ &= \boldsymbol{K}_p\boldsymbol{J}^{-1}\boldsymbol{\omega}_e + \boldsymbol{K}_i\boldsymbol{J}^{-1}\int\boldsymbol{\omega}_e\end{aligned} \tag{4.38}$$

where $\int\boldsymbol{\omega}_e \equiv \int_0^t\boldsymbol{\omega}_e(\tau)d\tau$ and $\boldsymbol{\omega}_e = \boldsymbol{\omega}_r - \boldsymbol{\omega}$ is the angular velocity error, with $\boldsymbol{\omega}_r$ being the angular velocity reference of the spacecraft body frame w.r.t. the inertial reference frame and expressed in the body reference frame.

It should also be pointed out that the control law (4.37) is designed considering the real values for the angular velocity $\boldsymbol{\omega}$ and the inertia tensor $\boldsymbol{J}$. In fact, the true values should be the filtered angular velocity measurement $\hat{\boldsymbol{\omega}}$ and matrix $\boldsymbol{J}_{ctr}$, which is believed to represent the true diagonal inertia matrix Λ (3.2). Although these variables are technically different from the real values $\boldsymbol{\omega}$ and $\boldsymbol{J}$, they can be deemed sufficiently close to introduce the assumption that

$$\begin{cases}\boldsymbol{\omega} \cong \hat{\boldsymbol{\omega}} \\ \boldsymbol{J} \cong \boldsymbol{J}_{ctr}\end{cases} \tag{4.39}$$

In future work, other more complex and nonlinear control methods could be explored to later relief this conjecture. However, the control system developed under this assumption is proven capable to achieve its main objective, i.e., to lead the satellite to spin around its principal major axis of inertia, as documented in the obtained results in Chapter 6.

4.2.2 Dynamic Reference Guidance

As seen in the previous subsection, the torque proportional-integral controller (4.37) requires an angular velocity reference $\boldsymbol{\omega}_r \in \mathbb{R}^3$ to compute the control torque. Moreover, its value should be updated throughout the simulation to keep it aligned with the major inertia axis Z_j.

The expression that relates the filtered measurements and other variables to the angular velocity reference is part of the guidance algorithm. In fact, the inertia matrix $\boldsymbol{J}_{ctr}$ used for the controller can also be updated along time and included in this subsystem, but as its ultimate goal is to align the angular velocity reference with the principal major axis of inertia, the proposed applications constitute the so-called *dynamic reference guidance*.

For this problem, two different guidance solutions were studied and are now presented in this thesis. The first alternative takes advantage of the values provided by the navigation filter and is labeled *estimate-based guidance*. On the other hand, the second option follows a bolder approach and eliminates the need for a determination system, instead using another nonlinear control law to converge the angular velocity reference to its desired value, i.e. a vector aligned with J_z, hence being known as *control-based guidance*.

Estimate-Based Guidance

Usually, a navigation filter is responsible for providing values to the guidance algorithm. In this particular case, the estimated state vector includes the rotation angles about the x-axis ($\hat{\phi} \in \mathbb{R}$) and the y-axis ($\hat{\theta} \in \mathbb{R}$) for the frame transformation of $\mathcal{B}$ to $\mathcal{J}$, which represents a reference frame aligned with the principal moments of inertia, and that can be used to update the angular velocity reference.

The rotation angles describe the orientation of the major axis of inertia relative to the spacecraft body frame. Therefore, using the Euler angles matrices definition ((2.5), (2.6) and (2.7)), and the rotation matrix convention, as seen in equation (3.3), the angular velocity reference $\boldsymbol{\omega}_r$ can be updated according to

$$\boldsymbol{\omega}_r = \boldsymbol{R}(\hat{\phi}, \hat{\theta})\boldsymbol{\omega}_{r_0} \tag{4.40}$$

where $\boldsymbol{\omega}_{r_0}$ is the initially desired angular velocity of the spacecraft body frame w.r.t. the inertial reference frame and represented in the body coordinate system, and with $\boldsymbol{R}(\hat{\phi}, \hat{\theta})$ being defined as

$$\boldsymbol{R}(\hat{\phi}, \hat{\theta}) = \boldsymbol{R}_x(\hat{\phi})\boldsymbol{R}_y(\hat{\theta}) \tag{4.41}$$

Furthermore, the rotation matrix $\boldsymbol{R}(\hat{\phi}, \hat{\theta})$, together with Sylvester's law of inertia (3.1), can also be used to update the diagonal inertia tensor $\boldsymbol{J}_{ctr}$, as given by

$$\boldsymbol{J}_{ctr} = \boldsymbol{R}(\hat{\phi}, \hat{\theta})\boldsymbol{J}_{ctr_0}\boldsymbol{R}(\hat{\phi}, \hat{\theta})^T \tag{4.42}$$

where $\boldsymbol{J}_{ctr_0}$ is the initial estimate for the diagonal inertia matrix, which is considered to be the main diagonal of the inertia tensor.

Control-Based Guidance

Another way to address the problem at hand is to find a relation between the angular velocity reference $\boldsymbol{\omega}_r$ and the external torque $\boldsymbol{\tau}$, and thus introduce a novel guidance law that updates the angular velocity reference according to the current torque values. The intuition used to define this method is the fact that there is necessarily a commanded torque to maintain the satellite spinning around an unstable axis, i.e., when $\boldsymbol{\omega}_r$ is unaligned with the principal major axis of inertia. Additionally, this reasoning suggests that torque values are related to the difference between the angular velocity direction and the major inertia axis orientation.

Ideally, the stationary case should be a free dynamics situation as well, because in case the angular velocity reference has been promptly corrected and aligned with the principal major axis of inertia, then these directions overlap and there is no difference between them, meaning that the external torque should also converge to zero. In other words, if the dynamics equation (4.36) tends to the appropriate $\boldsymbol{\omega}_r$, it can be reduced to

$$\dot{\boldsymbol{\omega}}_r = 0 \iff \boldsymbol{\omega}_r \times \boldsymbol{J}_{ctr}\boldsymbol{\omega}_r = 0 \tag{4.43}$$

Therefore, the first step in this strategy is to solve the dynamics equation for the angular velocity reference $\boldsymbol{\omega}_r$, in order to obtain a result suitable for a controller design that leads the system to a torque-less dynamics state. Assuming that $\tau_1 \neq 0$ and $\tau_2 \neq 0$, and still neglecting the existence of products of inertia (i.e., using the diagonal inertia matrix $\boldsymbol{J}_{ctr}$), the dynamics equation (4.36) holds three possible solutions.

Solutions 1 & 2:

$$\begin{aligned}
\omega_{r_1} &= \pm\frac{\sqrt{(J_1-J_2)(J_1-J_3)(J_2-J_3)\tau_1\tau_2\tau_3}}{(J_1-J_2)(J_1-J_3)\tau_1} \\
\omega_{r_2} &= \mp\frac{\sqrt{(J_1-J_2)(J_1-J_3)(J_2-J_3)\tau_1\tau_2\tau_3}}{(J_1-J_2)(J_2-J_3)\tau_2} \quad , \quad J_1 \neq J_2, J_1 \neq J_3, J_2 \neq J_3, \tau_1 \neq 0, \tau_2 \neq 0, \tau_3 \neq 0 \\
\omega_{r_3} &= \pm\frac{\sqrt{(J_1-J_2)(J_1-J_3)(J_2-J_3)\tau_1\tau_2\tau_3}}{(J_1-J_3)(J_2-J_3)\tau_3}
\end{aligned} \tag{4.44}$$

Solution 3:

$$\begin{aligned}
\omega_{r_1} &= \frac{1}{(J_1-J_3)}\cdot\frac{\tau_2}{z} \\
\omega_{r_2} &= -\frac{1}{(J_2-J_3)}\cdot\frac{\tau_1}{z} \quad , \quad J_1 \neq J_3, J_2 \neq J_3, \tau_3 = 0 \\
\omega_{r_3} &= z \qquad\qquad\qquad , \quad z \in \mathbb{R}
\end{aligned} \tag{4.45}$$

As it can be seen above, the first two solutions are too algebraically complex and impose exceptionally restrictive conditions, which would be difficult to overcome in an effort to develop the new control law. Contrastingly, Solution 3 provides an elegant, simple and intuitive relation between the angular velocity reference and the external torque.

On that account, to obtain a stationary dynamics-free system ($\boldsymbol{\tau} = 0$), it is necessary to acknowledge that the torque proportional controller (in the inner loop) is significantly faster than the dynamic reference guidance algorithm (in the outer loop). Thus, since the initial angular velocity reference is already significantly close to the desired spinning axis, when the angular velocity converges to the initial reference, both τ_1 and τ_2 are the ones mostly responsible for maintaining the unstable spin axis. Hence, it can be inferred that $\tau_3 \approx 0$ after a certain settling time. Thereafter, in compliance with the remaining conditions and keeping $\omega_{r_3} = z$ constant, the angular velocity references ω_{r_1} and ω_{r_2} should evolve in the direction that draws them closer to the obtained solutions, i.e., according to the following equations:

$$\begin{cases}
\delta\dot{\omega}_{r_1} = -K_1 C_1 \tau_2 \quad , \quad C_1 = \frac{1}{(J_1-J_3)z} \\
\delta\dot{\omega}_{r_2} = -K_2 C_2 \tau_1 \quad , \quad C_2 = -\frac{1}{(J_2-J_3)z}
\end{cases} \tag{4.46}$$

where $K_1, K_2 \in \mathbb{R}$ are the proportional gains for the variations of the angular velocity references $\delta\omega_{r_1}$ and $\delta\omega_{r_2}$, respectively.

Therefore, the dynamic angular velocity reference $\boldsymbol{\omega}_r$ is updated by

$$\boldsymbol{\omega}_r = \boldsymbol{\omega}_{r_0} + \delta\boldsymbol{\omega}_r \tag{4.47}$$

where $\boldsymbol{\omega}_{r_0}$ is the initially desired angular velocity of the spacecraft body frame w.r.t. the inertial reference frame and represented in the body coordinate system, and with $\delta\boldsymbol{\omega}_r \in \mathbb{R}^3$ being defined as

$$\delta\boldsymbol{\omega}_r = \begin{bmatrix} \delta\omega_{r_1} & \delta\omega_{r_2} & 0 \end{bmatrix}^T \tag{4.48}$$

After a certain time, the angular velocity reference converges to the correct value and maintains a stationary behavior, which is represented from now on as $\boldsymbol{\omega}_s \in \mathbb{R}^3$. From the initial angular velocity reference $\boldsymbol{\omega}_{r_0}$ and projecting a new stationary angular velocity vector in the $y0z$-plane ($\boldsymbol{\omega}_{s_x} \in \mathbb{R}^3$) and in the $x0z$-plane ($\boldsymbol{\omega}_{s_y} \in \mathbb{R}^3$), it becomes possible to compute the rotation angles ϕ_{ctr} and θ_{ctr} using the arc tangent trigonometric notion.

$$\begin{cases} \phi_{ctr} = \operatorname{atan2}\left(\omega_{r_0}(3)\omega_{s_x}(1) - \omega_{r_0}(1)\omega_{s_x}(3) \ , \ \omega_{r_0}(1)\omega_{s_x}(1) + \omega_{r_0}(3)\omega_{s_x}(3)\right) \\ \theta_{ctr} = \operatorname{atan2}\left(\omega_{r_0}(2)\omega_{s_y}(3) - \omega_{r_0}(3)\omega_{s_y}(2) \ , \ \omega_{r_0}(2)\omega_{s_y}(2) + \omega_{r_0}(3)\omega_{s_y}(3)\right) \end{cases} \tag{4.49}$$

Furthermore, these angles can be used to compute the rotation matrix $\boldsymbol{R}(\phi_{ctr}, \theta_{ctr})$ and update the diagonal inertia matrix $\boldsymbol{J}_{ctr}$, as illustrated in equation (4.42) for the estimate-based guidance.

However, this approach is incomplete as it ignores that, since $\delta\omega_{r_3} = 0$ and, consequently, $\boldsymbol{\omega}_s(3) = \boldsymbol{\omega}_{r_0}(3)$ in the spacecraft body frame, it can be mathematically confirmed that $||\boldsymbol{\omega}_s|| > ||\boldsymbol{\omega}_{r_0}||$, which implies that, regardless of the angular velocity alignment with the major inertia axis, the satellite is spinning at a faster rate than intended.

To amend the control-based guidance algorithm, an equation for $\delta\omega_{r_3}$ must be added to guarantee that the norm of both the initial and the stationary angular velocity references have the same value. This condition can be ensured by the following expression:

$$\delta\omega_{r_3} = \sqrt{||\boldsymbol{\omega}_{r_0}||^2 - (\omega_{s_1}^2 + \omega_{s_2}^2)} - \omega_{r_{0_3}} \tag{4.50}$$

Finally, the dynamic reference guidance equation (4.47) is revised and the variation of the angular velocity reference is redefined to

$$\delta\boldsymbol{\omega}_r = \begin{bmatrix} \delta\omega_{r_1} & \delta\omega_{r_2} & \delta\omega_{r_3} \end{bmatrix}^T \tag{4.51}$$

4.2.3 Lyapunov's Direct Method

The basic philosophy of Lyapunov's direct method is the mathematical formalization of a fundamental physical observation - as the total energy of a linear or nonlinear system is continuously dissipated, then it must eventually settle down to an equilibrium point. Thus, the stability of a system can be determined by examining a single scalar function, as opposed to solving the complete differential equations.

According to Lyapunov's global asymptotic stability (GAS) theorem, if there exists a scalar function $\boldsymbol{V}(x) : \mathbb{R}^n \to \mathbb{R}$, with continuous first order derivatives such that

- $\boldsymbol{V}(x)$ is globally positive definite, i.e., $\boldsymbol{V}(0) = 0$ and $\boldsymbol{V}(x) > 0$, $\forall x \in \mathbb{R}\backslash\{0\}$
- $\dot{\boldsymbol{V}}(x)$ is globally negative definite, i.e., $\dot{\boldsymbol{V}}(0) = 0$ and $\dot{\boldsymbol{V}}(x) < 0$, $\forall x \in \mathbb{R}\backslash\{0\}$
- $\boldsymbol{V}(x)$ is radially unbounded, i.e., $\boldsymbol{V}(x) \to \infty$ as $||x|| \to \infty$

then the equilibrium point at the origin is globally asymptotically stable [49].

Promptly, the Lyapunov criteria for a nonlinear system can be put into practice to determine the stability of the designed controller.

Theorem 1 (Torque PI Control Global Asymptotic Stability)**.** *Let the spacecraft dynamics be given by*

$$\dot{\boldsymbol{\omega}} = -\boldsymbol{J}^{-1}(\boldsymbol{\omega} \times \boldsymbol{J}\boldsymbol{\omega}) + \boldsymbol{J}^{-1}\boldsymbol{\tau},$$

and the torque PI controller defined as

$$\boldsymbol{\tau} = \boldsymbol{\omega} \times \boldsymbol{J}\boldsymbol{\omega} + \boldsymbol{K}_p\boldsymbol{\omega}_e + \boldsymbol{K}_i \int \boldsymbol{\omega}_e \ \ , \ \ \text{with } \boldsymbol{K}_p, \boldsymbol{K}_i \in \mathbb{R}.$$

Then, for $\boldsymbol{K}_p > 0$, $\boldsymbol{K}_i > 0$, and PSD matrix $\boldsymbol{J}$, the closed-loop system is globally asymptotically stable.

Proof. Consider the Lyapunov candidate function is given by

$$\boldsymbol{V}(\boldsymbol{\omega}_e) = \frac{1}{2}\boldsymbol{\omega}_e^T\boldsymbol{J}\boldsymbol{\omega}_e + \frac{1}{2}\left(\int \boldsymbol{\omega}_e\right)^T \boldsymbol{K}_i \int \omega_e \tag{4.52}$$

which is a typical quadratic form polynomial, thus, as the Sylvester's criterion holds for the positive semi-definite (PSD) inertia tensor $\boldsymbol{J}$ and for $\boldsymbol{K}_i > 0$, it guarantees that $\boldsymbol{V}(\boldsymbol{\omega}_e)$ is globally positive definite.

From equation (4.38), the Lyapunov candidate function time-derivative $\dot{\boldsymbol{V}}(\boldsymbol{\omega}_e)$ can be calculated as

$$\begin{aligned}
\dot{\boldsymbol{V}}(\boldsymbol{\omega}_e) = \frac{\partial \boldsymbol{V}}{\partial \boldsymbol{\omega}_e} \cdot \frac{\partial \boldsymbol{\omega}_e}{\partial t} &= \boldsymbol{\omega}_e^T\boldsymbol{J}\dot{\boldsymbol{\omega}}_e + \left(\int \boldsymbol{\omega}_e\right)^T \boldsymbol{K}_i\boldsymbol{\omega}_e \\
&= -\boldsymbol{\omega}_e^T\boldsymbol{J}\left(\boldsymbol{K}_p\boldsymbol{J}^{-1}\boldsymbol{\omega}_e + \boldsymbol{K}_i\boldsymbol{J}^{-1}\int \boldsymbol{\omega}_e\right) + \boldsymbol{K}_i\left(\int \boldsymbol{\omega}_e\right)^T \boldsymbol{\omega}_e \\
&= -\boldsymbol{K}_p\boldsymbol{\omega}_e^2 - \boldsymbol{K}_i\boldsymbol{\omega}_e^T \int \boldsymbol{\omega}_e + \boldsymbol{K}_i\left(\int \boldsymbol{\omega}_e\right)^T \boldsymbol{\omega}_e \\
&= -\boldsymbol{K}_p\boldsymbol{\omega}_e^2 \leq 0 \ \ , \ \ \text{if } \boldsymbol{K}_p > 0
\end{aligned} \tag{4.53}$$

where the angular velocity error time-derivative $\dot{\boldsymbol{\omega}}_e$ is defined by

$$\dot{\boldsymbol{\omega}}_e = \dot{\boldsymbol{\omega}}_r - \dot{\hat{\boldsymbol{\omega}}} = -\dot{\hat{\boldsymbol{\omega}}} \tag{4.54}$$

and with the torque control law being deemed sufficiently fast to consider the angular velocity reference constant and, consequently, its derivative null.

Hence, as $\boldsymbol{K}_p > 0$ and $\dot{\boldsymbol{V}}(\boldsymbol{\omega}_e) = 0 \ \text{ iff } \ \boldsymbol{\omega}_e = 0$, then $\dot{\boldsymbol{V}}(\boldsymbol{\omega}_e)$ is globally negative definite. Therefore, since the previous Lyapunov function (4.52) is also radially unbounded, it is possible to conclude that the PI controller is globally asymptotically stable at the zero-error equilibrium point.

QED

Expanding Lyapunov's direct method to the guidance algorithm would require admitting reference variance, thus avoiding the simplification present in equation (4.54) and calling for a new Lyapunov candidate function $\boldsymbol{V}(\boldsymbol{\omega}_e, \boldsymbol{\omega}_r)$ that includes a term to deal with the angular velocity reference. Unfortunately, there is no mechanistic approach to create Lyapunov functions and, consequently, designing a new function to consider the angular velocity reference derivative can become an intricate task. For that reason, a theoretical formalization of the obtained results is deferred for future work.

Chapter 5

Implementation

For the previously introduced active spin stabilization problem, a navigation filter, a controller and two different guidance algorithms were proposed to control the system. At this point, it is essential to determine the feasibility and viability of the formerly presented solutions.

This chapter discusses the development of a simulation environment, as a practical and realistic approach to implement the active spin stabilization control system. To design and simulate the suggested model, the computational tools used were MATLAB® and Simulink®. The top-level architecture diagram of the model is depicted in Figure 5.1, which comprises two main blocks - the Spacecraft Plant (in blue) and the Control System (in red).

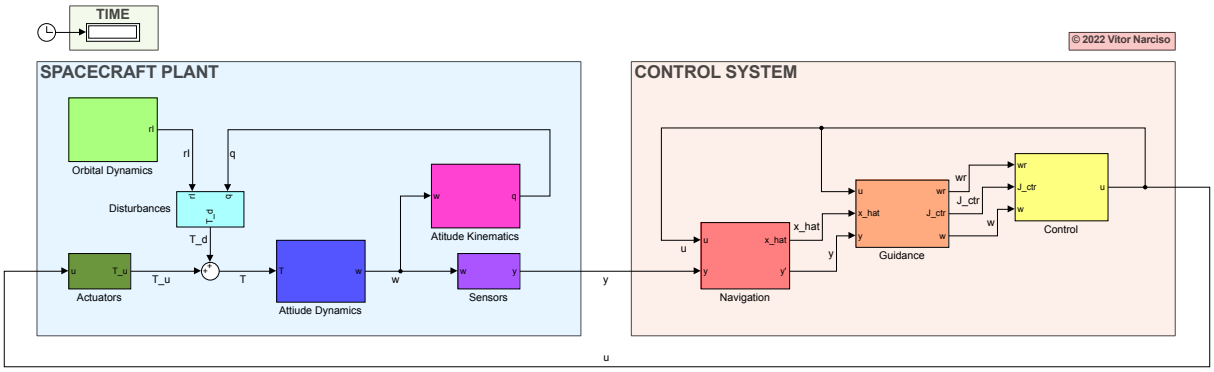


Figure 5.1: Simulink® top-level architecture block diagram

These two systems distinguish the real world environment from the guidance, navigation and control (GNC) system. In Section 5.1, the Spacecraft Plant is rigorously described, addressing the several subsystems necessary to simulate the position, attitude and physical properties of a satellite system in orbit around the Earth. Likewise, Section 5.2 presents in detail the Control System, tackling the design and implementation of the GNC solution to achieve active spin stabilization. Finally, a succinct list of all the simulation parameters used for the Simulink® model is provided in Section 5.3.

5.1 Spacecraft Plant

To recreate the space environment around a satellite, the Spacecraft Plant model must be implemented in the simulator. It is composed of an orbital dynamics model to define the satellite's position in a circular orbit around the Earth, a disturbances model that simulates the effects of external perturbances torques on the spacecraft, kinematics and dynamics models to propagate the satellite's attitude, and sensors and actuators models that reproduce their characteristic measurement outputs and control inputs, respectively.

Orbital Dynamics and Disturbances Models

As the satellite is considered to evolve along a circular Low Earth Orbit (LEO), the orbital dynamics model can take advantage of equations (2.37), (2.38) and (2.39) to define the inertial orbit radius $\boldsymbol{r}_I$ in the ECI reference frame. For this application, the selected LEO altitude is $h = 600\,\mathrm{km}$ and the orbit inclination is set to $i = 100\,\mathrm{deg}$.

In the disturbances model, the orbit radius vector expressed in the spacecraft body frame $\boldsymbol{r}_c$ is obtained from equation (2.40), using the quaternion provided by the attitude kinematics propagator. Then, the gravity-gradient disturbance torque can be computed according to equation (2.36).

However, all the disturbance torques models presented in subsection (2.3.3) should also be included in the attitude dynamics equation (2.30), with $\boldsymbol{\tau}_d$ representing the sum of the disturbance torques. Indeed, since the most relevant torque is caused by the gravity-gradient, it can be used to estimate and implement the remaining perturbances as a small percentage of its norm. Considering that

$$||\boldsymbol{\tau}_{gg}|| = \left|\left|\frac{3\mu_E}{r_c^5}\boldsymbol{r}_c \times \boldsymbol{J}\boldsymbol{r}_c\right|\right| \leq \left|\frac{3\mu_E}{r_c^5}\right| \cdot ||\boldsymbol{r}_c|| \cdot ||\boldsymbol{J}\boldsymbol{r}_c|| \leq \frac{3\mu_E}{r_c^3} \cdot \max(\boldsymbol{J}) \tag{5.1}$$

then the standard deviation for the gravity-gradient torque can be approximated as $3\sigma_{gg} \approx \frac{3\mu_E}{r_c^3} \cdot \max(\boldsymbol{J})$. Hence, the remaining disturbances torque $\boldsymbol{\tau}_r \in \mathbb{R}^3$ is modeled as a zero-mean Gaussian distribution, and defined by

$$\begin{cases} \boldsymbol{\tau}_d = \boldsymbol{\tau}_{gg} + \boldsymbol{\tau}_r \\ \boldsymbol{\tau}_r \sim (0, \sigma_r^2) \\ \sigma_r = 0.1\sigma_{gg} \end{cases} \tag{5.2}$$

Kinematics and Dynamics Models

The kinematics and dynamics models propagate the satellite's attitude, while taking into account the disturbance torques described above. In regard to attitude kinematics, as supported by the pros and cons theoretical review presented in Section 2.1, the quaternion parameterization is the most appropriate attitude representation for this application. Therefore, the model is implemented according to the quaternion derivative (2.15), but to avoid numerical integration errors that could result in non-unitary quaternion vectors, a simple correction known as Baumgarte stabilization [50] is introduced to

equation (2.22), yielding

$$\dot{\boldsymbol{q}} = \frac{1}{2}\boldsymbol{\Omega}(\boldsymbol{\omega})\boldsymbol{q} + \frac{1}{2}\rho_B \boldsymbol{q}\left((\boldsymbol{q}^T\boldsymbol{q})^{-1} - 1\right) \tag{5.3}$$

with the Baumgarte stabilization parameter set to $\rho_B = 100$.

On the other hand, the dynamics model consists of Euler's moment equations (2.27), applied to the real inertia matrix. However, it is important to notice that the external torque is provided by the sum of the control torque and the disturbances torque, as given by equation (2.30).

Sensors and Actuators Models

Both sensors and actuators are hardware models responsible for connecting the Spacecraft Plant to the Control System, and ensure that the transmitted signals are simulated as genuinely and faithful to the real-life hardware systems as possible.

In the sensors model, to reproduce the characteristic noise associated to rate gyroscopes measurements, the angular velocity is corrupted according to the following expression:

$$\boldsymbol{\omega}_m = \boldsymbol{\omega} + \boldsymbol{\eta}_v \tag{5.4}$$

where $\boldsymbol{\omega}_m$ is the measured rate, $\boldsymbol{\omega}$ is the true rate, and $\boldsymbol{\eta}_v$ are independent zero-mean Gaussian white-noise processes, with standard deviation $\sigma_\omega = 1 \times 10^{-6}\,\mathrm{rad/s}$.

Alternatively, the control inputs are simulated by three pairs of thrusters, lined up according to the body coordinate system, and each pair providing positive and negative torque about a single axis. To keep its implementation straightforward, the torques are simply restricted by a saturation block, whose limit value is $\tau_{max} = \pm 20\,\mathrm{N{\cdot}m}$, to replicate its typical physical constraints.

Furthermore, the actuators model also includes an estimation of the fuel consumption, based on the control torque. It constitutes a crucial indicator to the desired objective, i.e., to confirm the increased efficiency and reduced power requirements of spinning aligned to the principal major axis of inertia. To compute the fuel consumption estimate, the expelled propellant mass variation $\Delta M \in \mathbb{R}$ is calculated by

$$\Delta M = \int \frac{\sum_{n=1}^{3} |\tau_n|}{g_0 I_{sp} d_{CM}} \tag{5.5}$$

where $|\tau_n|$ is the torque absolute value in the n-axis, g_0 is Earth's gravitational acceleration, I_{sp} is the thrusters' specific impulse, and d_{CM} is the distance from the actuators to the satellite's center of mass.

5.2 Control System

The Control System is responsible for applying the active spin stabilization described in Chapter 4, namely, the navigation filter, the guidance algorithms and the control law. Its objective is to use the measurements provided by the rate gyros to determine the required torque that the thrusters must exert to achieve the desired spacecraft angular velocity.

Navigation Filter

The first subsystem integrated in the GNC/ADCS system is the navigation filter, which takes the angular velocity measurements and the control input provided by the controller to estimate the angular velocity vector ($\hat{\boldsymbol{\omega}}$), the rotation angles ($\hat{\phi}$ and $\hat{\theta}$) and the principal moments of inertia ($\hat{J}_1$, $\hat{J}_2$ and $\hat{J}_3$). These 8 variables constitute the state vector $\boldsymbol{x}$ and are all observable except for the moments of inertia.

Its implementation follows directly the continuous-time Extended Kalman filter (EKF) algorithm, as summed up in Table 4.3. The jacobian matrix $\boldsymbol{F}$ (4.33) and the nonlinear functions $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u})$ and $\boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u})$ are imported to the simulation environment as functions from MATLAB® files computed before execution, using the Symbolic Math Toolbox.

Regarding parameter tuning of the estimator, since the uncertainty of the initial estimate is different for every variable, the process noise matrix $\boldsymbol{Q}_c$ values are tuned independently. Likewise, the initial covariance matrix $\boldsymbol{P}_0$ is defined according to the confidence in each initial state estimate. Contrastingly, knowledge about the measurement process noise characteristics for the rate gyros can be exploited to achieve a balance between estimation error and convergence time for the measurement noise matrix $\boldsymbol{R}_c$.

Guidance Algorithm

From the angular velocity measurements, the guidance algorithms can update the inertia matrix and the angular velocity reference that must be provided to the controller. Both strategies for guidance were included in this subsystem, and can be adjusted manually in the Simulink® model, as seen in Figure 5.2.

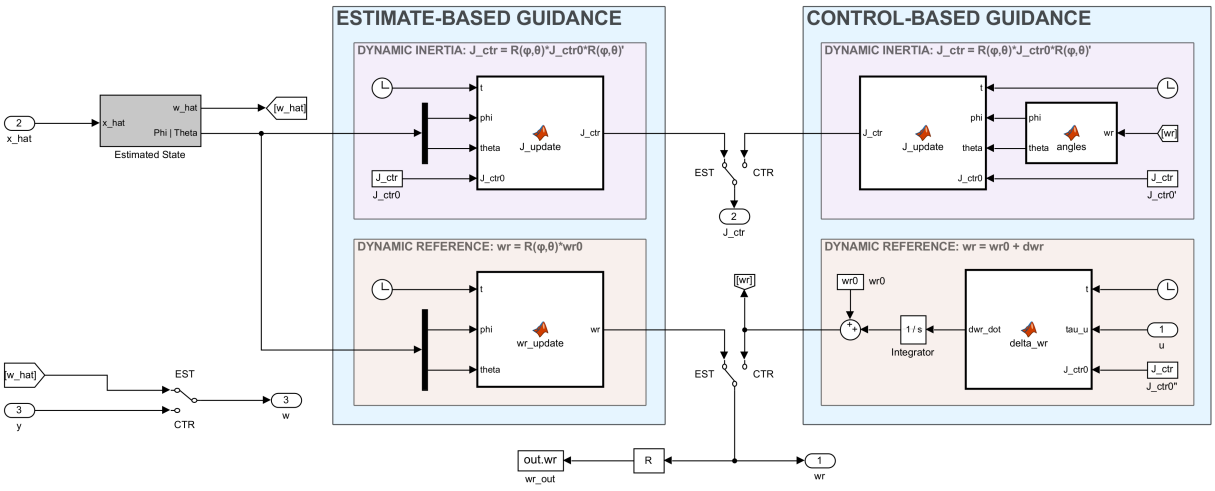


Figure 5.2: Simulink® guidance subsystem model

On the left side, the estimate-based guidance uses the filtered angular velocity $\hat{\boldsymbol{\omega}}$ and takes advantage of the rotation angles estimates to update the angular velocity reference $\boldsymbol{\omega}_r$ and the inertia matrix $\boldsymbol{J}_{ctr}$, which are computed as specified by equations (4.40) and (4.42), respectively. On the right side, the control-based guidance calculates the angular velocity reference $\boldsymbol{\omega}_r$ according to equations (4.47) and (4.51). Similarly, this algorithm updates the inertia matrix $\boldsymbol{J}_{ctr}$ using equation (4.42), but obtaining the rotation matrix $\boldsymbol{R}(\phi_{ctr}, \theta_{ctr})$ from the control angles equation (4.49).

Control Law

Lastly, the control law subsystem takes the inertia matrix, the angular velocity and its reference, and applies the torque controller given by equation (4.37), with a proportional gain $\boldsymbol{K}_p = 100$ and an integral gain $\boldsymbol{K}_i = 2$, to compute the adequate control torque. This value is transmitted to the Spacecraft Plant through the actuators model, and the feedback loop closes and repeats for the entire simulation time.

5.3 Simulation Parameters

For this study, the satellite model and its properties are considered to be constant. Thus, except for the real inertia matrix $\boldsymbol{J}$ in the body reference frame, the Spacecraft Plant is the same at all times. In fact, many parameters never change throughout the entire implementation and testing phase. Therefore, to provide a simple process to consult their values, every single one is classified and listed in Table 5.1. It should be noted that the $n \times n$ identity matrix is represented by $\boldsymbol{I}_n$.

Table 5.1: Simulation parameters

Topic	**Parameter**	**Notation**	**Value**
Spacecraft Model	Satellite Mass	m_{sat}	$400\,\mathrm{kg}$
	Moments of Inertia	J_{ii}	$[360 \quad 280 \quad 500]\,\mathrm{kg{\cdot}m^2}$
	Products of Inertia	J_{ij}	$[30 \quad -40 \quad 35]\,\mathrm{kg{\cdot}m^2}$
Orbital Dynamics	Circular Orbit Reference Altitude	h	$600\,\mathrm{km}$
	Orbit Inclination	i	$100\,\mathrm{deg}$
Disturbances	Gravity-Gradient Torque Std. Dev.	σ_{gg}	$5.605 \times 10^{-4}\,\mathrm{N{\cdot}m}$
	Remaining Disturbances Torque Std. Dev.	σ_r	$5.605 \times 10^{-5}\,\mathrm{N{\cdot}m}$
Sensors	Rate Gyroscopes Sample Frequency	f_ω	$100\,\mathrm{Hz}$
	Angular Velocity Noise Std. Dev.	σ_ω	$1 \times 10^{-6}\,\mathrm{rad/s}$
Actuators	Thrusters' Specific Impulse	I_{sp}	$290\,\mathrm{s}$
	Maximum control torque	τ_{max}	$20\,\mathrm{N{\cdot}m}$
	Distance from the Actuators to the CoM	d_{CM}	$\sqrt{3}\,\mathrm{m}$
Navigation	Angular Velocity Process Noise	$\boldsymbol{Q}_c^\omega$	$1 \times 10^{-2} \cdot \boldsymbol{I}_3$
	Rotation Angles Process Noise	$\boldsymbol{Q}_c^\alpha$	$1 \times 10^{-2} \cdot \boldsymbol{I}_2$
	Principal Axis of Inertia Process Noise	$\boldsymbol{Q}_c^J$	$1 \times 10^{0} \cdot \boldsymbol{I}_3$
	Measurement Noise	$\boldsymbol{R}_c$	$1 \times 10^{-3} \cdot \boldsymbol{I}_3$
	Angular Velocity Initial Covariance	$\boldsymbol{P}_0^\omega$	$1 \times 10^{-3} \cdot \boldsymbol{I}_3$
	Rotation Angles Initial Covariance	$\boldsymbol{P}_0^\alpha$	$1 \times 10^{0} \cdot \boldsymbol{I}_2$
	Principal Axis of Inertia Initial Covariance	$\boldsymbol{P}_0^J$	$1 \times 10^{3} \cdot \boldsymbol{I}_3$
Guidance	Proportional Gain for $\delta\omega_{r_1}$	K_1	1×10^{-2}
	Proportional Gain for $\delta\omega_{r_2}$	K_2	1×10^{-2}
Control	Proportional Gain for Torque Control	$\boldsymbol{K}_p$	100
	Integral Gain for Torque Control	$\boldsymbol{K}_i$	2

Chapter 6

Simulation Results

The final goal of this study consists in taking advantage of the simulation environment and evaluate the performance and robustness of the active spin stabilization control system. For that purpose, multiple simulations were conducted, in order to use the obtained results to compare their performances and draw conclusions.

Firstly, both the Spacecraft Plant and the Control System are simulated in continuous-time, and the selected solver is the one-step Dormand-Prince method (`ode45`), with a fixed time-step of $0.01\,\mathrm{s}$. For stiff problems, the Simulink® model automatically switches to multi-step numerical differentiation formulas (`ode15s`) in order to increase the solver's efficiency. On all scenarios and iterations evaluated, the simulation time is set to $t_{sim} = 3000\,\mathrm{s}$ for each run of the system, and has data collected at a sample frequency of $f_S = 20\,\mathrm{Hz}$. The remaining constant values used in the simulation environment are referenced in Table 5.1.

This Chapter addresses a couple of the most important simulated cases among the total amount of trials, using estimate-based and control-based guidance. Section 6.1 describes and evaluates one simple case using realistic predetermined values for all parameters. Later, to test the control robustness and effectiveness under different initial conditions, a Monte Carlo analysis is carried out, providing the obtained results and drawn conclusions in Section 6.2. Finally, all simulations included noise and disturbances, namely the gravity-gradient disturbance torque. As such, their respective plots and other less significant figures for the outcomes discussion are provided in Appendix B.

6.1 Selected Parameters Simulation

6.1.1 Setup

This first case considers the satellite properties and the selected parameters values made available in Table 5.1. Therefore, the only step left to run the simulation is to define the initial conditions for attitude $\boldsymbol{q}_0$, angular velocity reference $\boldsymbol{\omega}_{r_0}$, angular velocity reference variance $\delta\boldsymbol{\omega}_{r_0}$, state vector $\boldsymbol{x}_0$ (which includes the angular velocity $\boldsymbol{\omega}_0$, the rotation angles ϕ_0 and θ_0, and the principal moments of inertia $\boldsymbol{\Lambda}_0$), control torque $\boldsymbol{\tau}_{u_0}$, and the diagonal inertia matrix $\boldsymbol{J}_{ctr_0}$ for the Control System.

Table 6.1: Initial conditions for the selected parameters simulation

Topic	**Variable**	**Notation**	**Value**
Attitude	Unit Quaternion	$\boldsymbol{q}_0$	$[0 \quad 0 \quad 0 \quad 1]$
Reference	Angular Velocity	$\boldsymbol{\omega}_{r_0}$	$[0 \quad 0 \quad 10]\,\mathrm{deg/s}$
	Angular Velocity Variance	$\delta\boldsymbol{\omega}_{r_0}$	$[0 \quad 0 \quad 0]\,\mathrm{deg/s}$
State Vector	Angular Velocity	$\boldsymbol{\omega}_0$	$[1 \quad -1 \quad 8]\,\mathrm{deg/s}$
	Rotation Angle in x-Axis	ϕ_0	$0\,\mathrm{deg}$
	Rotation Angle in y-Axis	θ_0	$0\,\mathrm{deg}$
	Principal Moments of Inertia	$\boldsymbol{\Lambda}_0$	$[360 \quad 280 \quad 500]\,\mathrm{kg{\cdot}m^2}$
Control System	Control Inertia Matrix	$\boldsymbol{J}_{ctr_0}$	$\mathrm{diag}([360 \quad 280 \quad 500])\,\mathrm{kg{\cdot}m^2}$

Table 6.1 collects all the initial conditions for the above-mentioned variables. Each value has been picked beforehand to recreate a standard scenario where active spin stabilization could become advantageous for the space application. In particular, the initial inertia tensor provided to the controller $\boldsymbol{J}_{ctr_0}$ and the initial estimate for the principal moments of inertia $\boldsymbol{\Lambda}_0$ are simply the true inertia tensor diagonal, and the rotation angles initial estimates ϕ_0 and θ_0 are null, as the spacecraft body frame is initially believed to be aligned with the principal axes of inertia.

However, the remaining values have been selected using a relatively arbitrary approach. For instance, the initial angular velocity reference is defined as $\boldsymbol{\omega}_{r_0} = [0 \quad 0 \quad \omega_z]^T$ to guarantee that the angular velocity vector rotates and converges to the desired direction (since the Control System aligns the z-axis of the body coordinate frame to the principal major axis of inertia), but ω_z could have been defined by any other reasonable value and even switched to another axis without loss of generality (w.l.o.g.). The same reasoning may be applied to the initial angular velocity $\boldsymbol{\omega}_0$, which has been weighted in favor of the z-axis, yet can take any other value w.l.o.g., as long as it is sensible considering the space environment.

Lastly, the initial angular velocity reference variance $\delta\boldsymbol{\omega}_{r_0}$ is set to zero, as the Simulink® model starts from a no-control resting position. Additionally, the initial attitude $\boldsymbol{q}_0$ has been defined as the quaternion identity w.l.o.g., since the active spin stabilization system does not rely on direct attitude control and, in practice, the space mission AOCS/GNC systems would comply with any attitude control system requirement.

6.1.2 Results

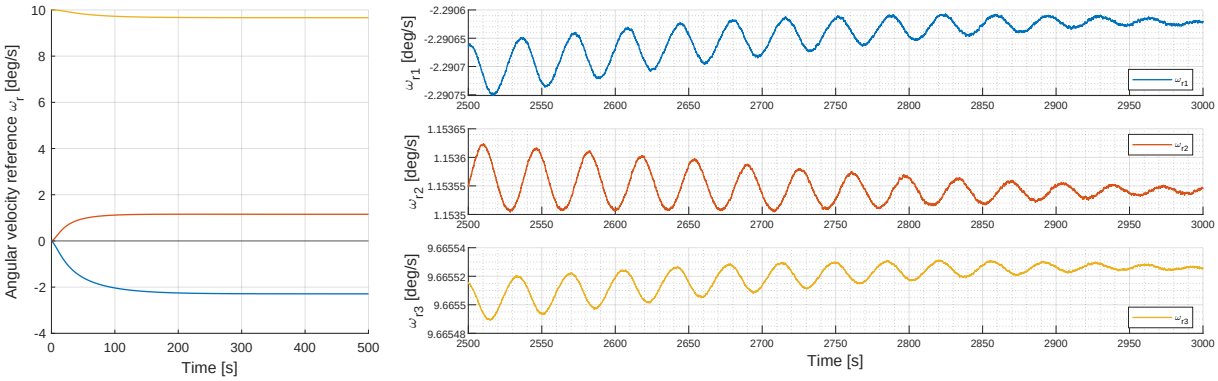
After performing the selected parameters simulation, the obtained results were compared to the real values. In this particular case, the true inertia matrix in Table 5.1 can be factorized according to Sylvester's law of inertia (3.1) using the MATLAB® function `eig()`. Therefore, the principal moments of inertia (J_1, J_2 and J_3) are directly provided by the eigenvalues, and the rotation angles (ϕ and θ) can be computed from the arc tangent trigonometric notion (4.49) applied to the principal major axis of inertia eigenvector. These parameters are displayed in Table 6.2.

Table 6.2: Real values for the selected parameters simulation

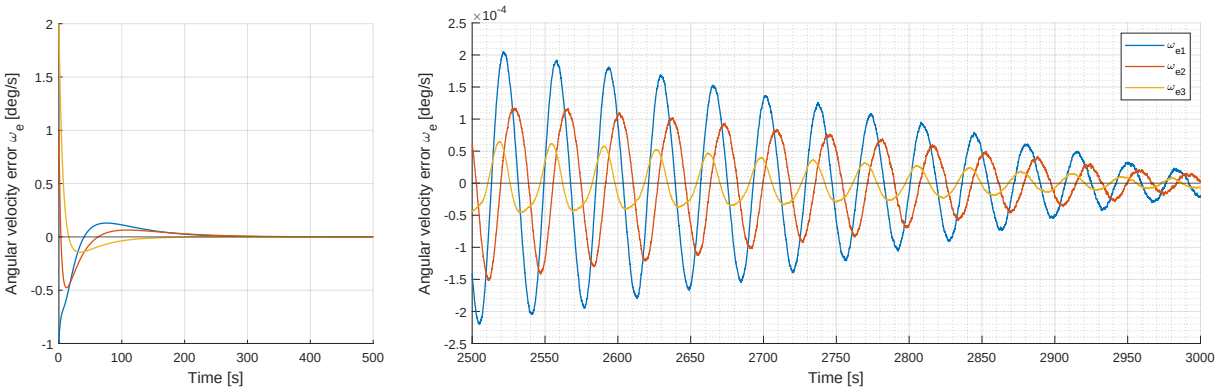
Parameter	J_1	J_2	J_3	ϕ	θ
Value	$365.3879\,\mathrm{kg\cdot m^2}$	$260.9539\,\mathrm{kg\cdot m^2}$	$513.6582\,\mathrm{kg\cdot m^2}$	$-6.8067\,\mathrm{deg}$	$-13.3339\,\mathrm{deg}$

Estimate-Based Guidance

In this scenario, the navigation filter has an active role in the guidance algorithm. Figure 6.1 presents the evolution of the angular velocity reference, provided by the estimate-based guidance, and the angular velocity tracking error, i.e., the difference between the angular velocity reference and the real angular velocity. Both units are measured in degrees per second since it is a more intuitive representation.



(a) Initial convergence and steady-state evolution of the angular velocity reference $\boldsymbol{\omega}_r$



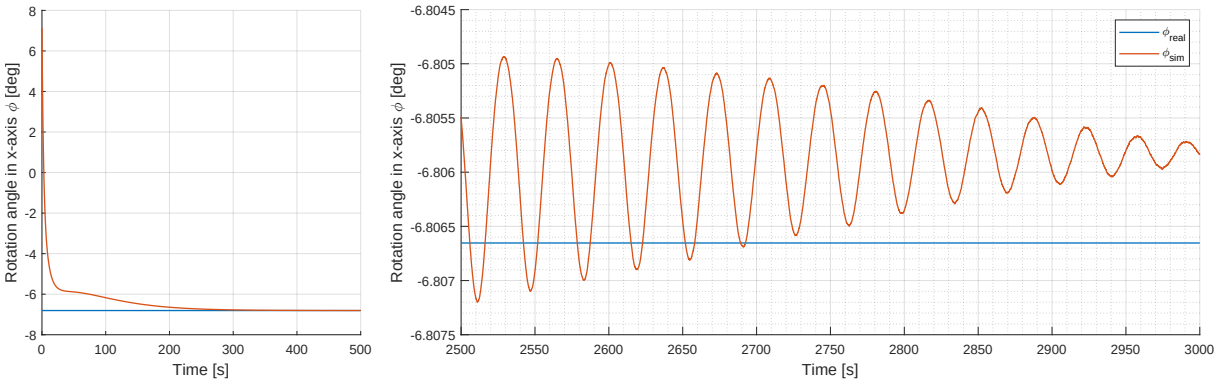
(b) Initial convergence and steady-state evolution of the angular velocity error $\boldsymbol{\omega}_e$

Figure 6.1: Angular velocity reference (6.1(a)) and error (6.1(b)) for estimate-based guidance

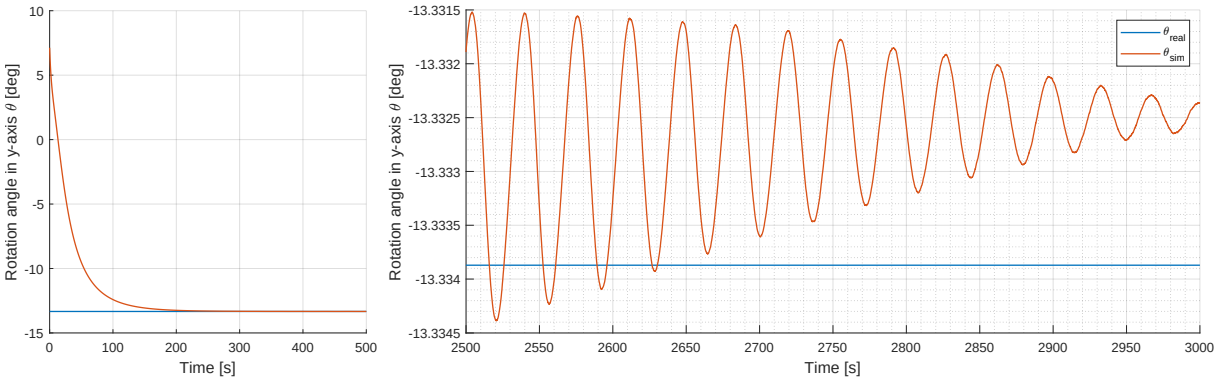
For the angular velocity reference, the end goal is to reach a stable value that is aligned with the principal major axis of inertia, while the angular velocity error should keep up with the reference and converge to zero. As can be observed in graphics 6.1(a) and 6.1(b), the guidance algorithm brings the angular velocity reference to a steady-state and the controller leads the angular velocity to the desired value in under $400\,\mathrm{s}$. There is a noticeable oscillatory behavior that is also present in the remaining graphs, caused by the gravity-gradient disturbance. In fact, although it seems the steady-state plots are still converging, the values are simply following the gravity-gradient amplitude, as shown in Figure B.2.

To update the angular velocity reference, the rotation angles estimates are applied to equation (4.40). Unfortunately, as the EKF estimates angles from the dynamics equation (4.36) with the rotation matrix

defined according to Rodrigues' rotation formula (3.5), it cannot disambiguate the angles individually and θ_{err} only converges to a $10^{-1}\,\mathrm{deg}$ steady-state error, as one can see in Figure B.4.



(a) Initial convergence and steady-state evolution of the simulated rotation angle in x-axis ϕ_{sim}



(b) Initial convergence and steady-state evolution of the simulated rotation angle in y-axis estimate θ_{sim}

Figure 6.2: Rotation angles simulated in x-axis (6.2(a)) and y-axis (6.2(b)) for estimate-based guidance

To determine if the angular velocity vector has the same direction as the major inertia axis, the rotation angles of the simulation are computed using arc tangent (4.49) and compared to the true rotation angles listed in Table 6.2. From graphs 6.2(a) and 6.2(b), it becomes clear that both angles are aligned within a margin error of $10^{-2}\,\mathrm{deg}$. This inaccuracy comes from the angles estimate error mentioned above but, in a real application, a $0.01\,\mathrm{deg}$ shift is insignificant and can be deemed essentially as zero.

Additionally, Figure 6.3 provides the generated control torque which converges to zero as the spinning axis approaches the major inertia axis, hence reaffirming the power efficiency of spin stabilization.

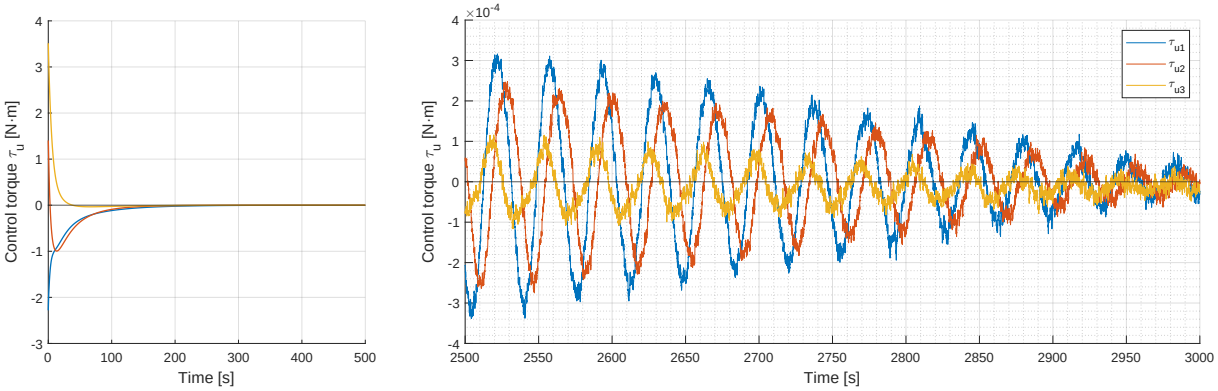
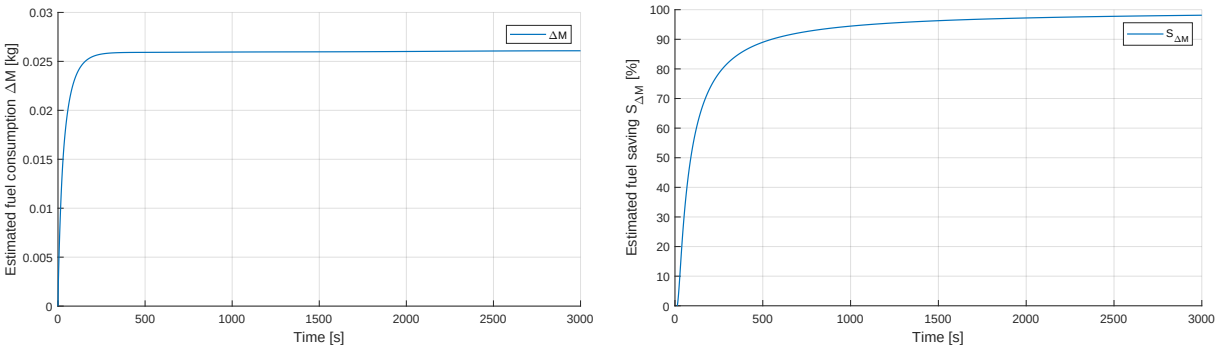


Figure 6.3: Initial convergence and steady-state evolution of control torque for estimate-based guidance

In fact, to further assess the impact of active spin stabilization in reducing propellant usage and increase the satellite's power autonomy, an estimate for fuel consumption is obtained from equation (5.5).



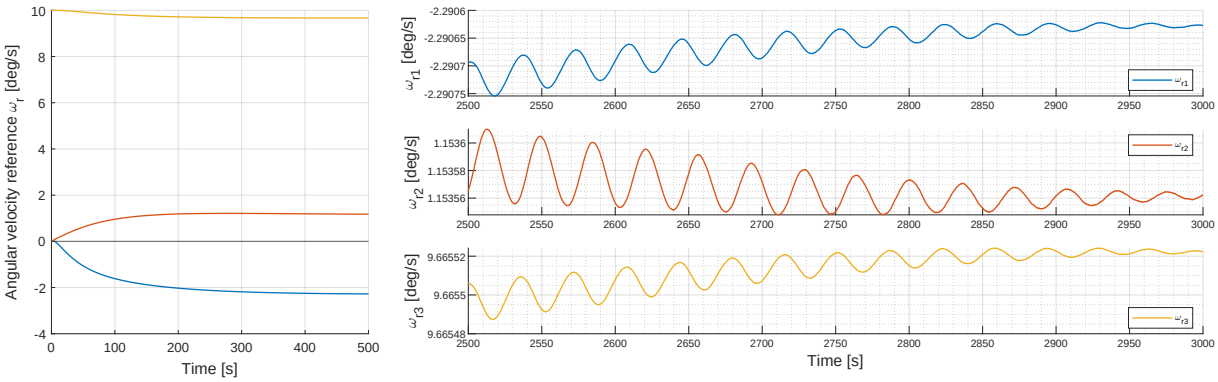
(a) Estimated fuel consumption ΔM

(b) Estimated fuel saving $S_{\Delta M}$

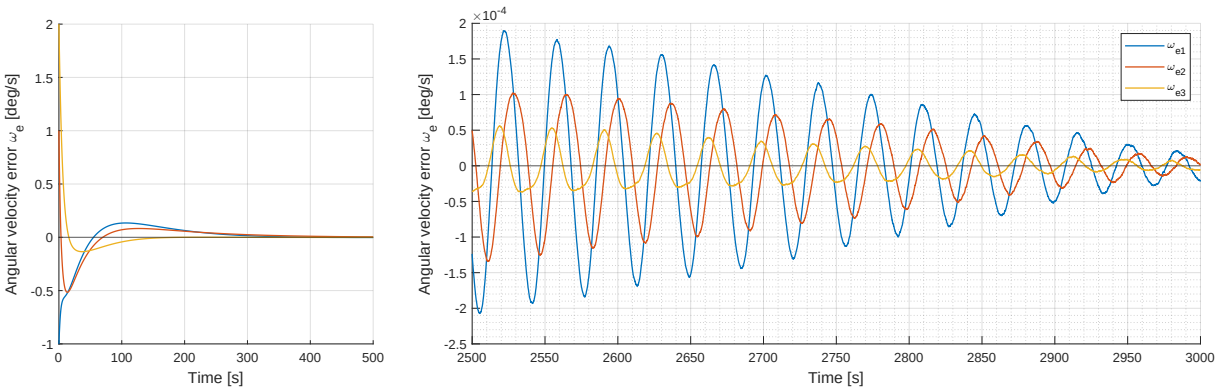
Figure 6.4: Estimated fuel consumption (6.4(a)) and saving (6.4(b)) for estimate-based guidance

The estimated fuel consumption has an initial boost related to the torque needed for the satellite to reach the desired angular velocity. However, as the relevant axes get increasingly aligned, this variable rapidly settles to an approximately constant value, with a final propellant consumption estimate of $0.0261\,\mathrm{kg}$. Comparing these value to a situation with no guidance to update the angular velocity reference leads to an estimated reduction of 98.1301% in fuel expending for the total simulation time.

Control-Based Guidance



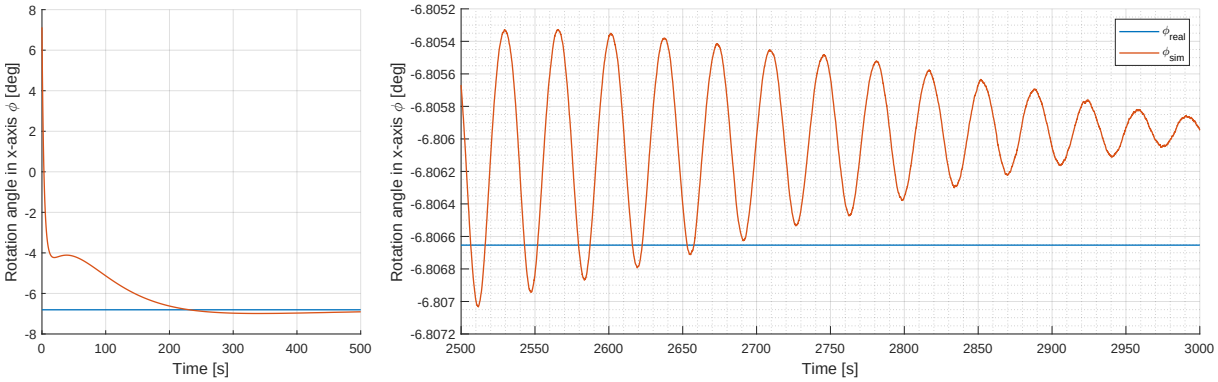
(a) Initial convergence and steady-state evolution of the angular velocity reference $\boldsymbol{\omega}_r$



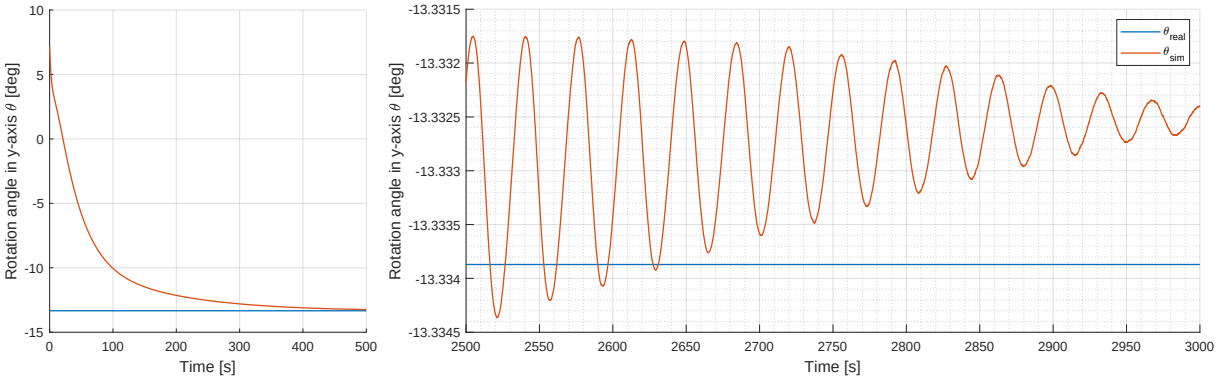
(b) Initial convergence and steady-state evolution of the angular velocity error $\boldsymbol{\omega}_e$

Figure 6.5: Angular velocity reference (6.5(a)) and error (6.5(b)) for control-based guidance

On the other hand, the control-based guidance does not require a navigation solution. In Figure 6.5, the angular velocity reference and the angular velocity tracking error exhibit a similar steady-state evolution when compared to the estimate-based case. Indeed, both variables rapidly come close to the desired values, revealing just a rather longer convergence time as they only settle after $500\,\mathrm{s}$.



(a) Initial convergence and steady-state evolution of the simulated rotation angle in x-axis ϕ_{sim}



(b) Initial convergence and steady-state evolution of the simulated rotation angle in y-axis estimate θ_{sim}

Figure 6.6: Rotation angles simulated in x-axis (6.6(a)) and y-axis (6.6(b)) for control-based guidance

The same behavior can be observed for both the simulated rotation angles in graphs 6.6(a) and 6.6(b), since the variables take more time to converge as well. Moreover, it should be noted that the control torque in Figure 6.7 has additional noise corruption due to the absence of an EKF, which filters and reduces the noise processes for the angular velocity measurements, as evident in Figure B.3.

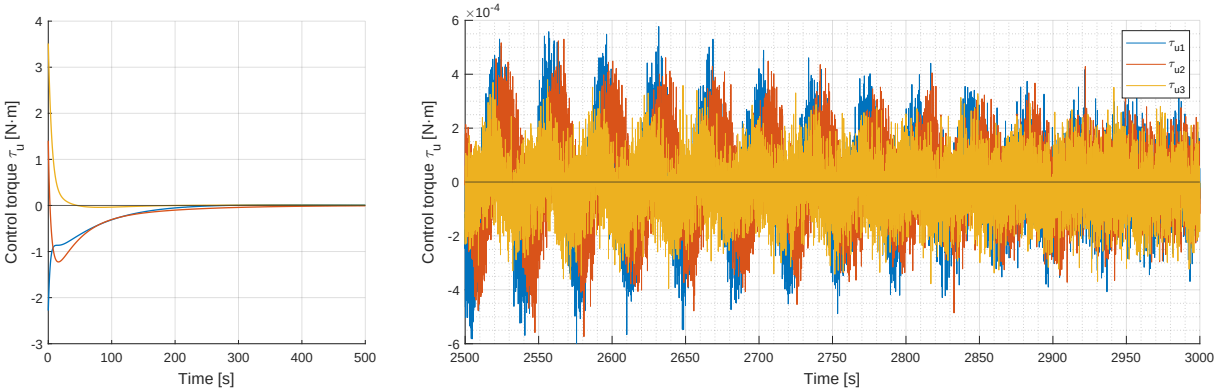
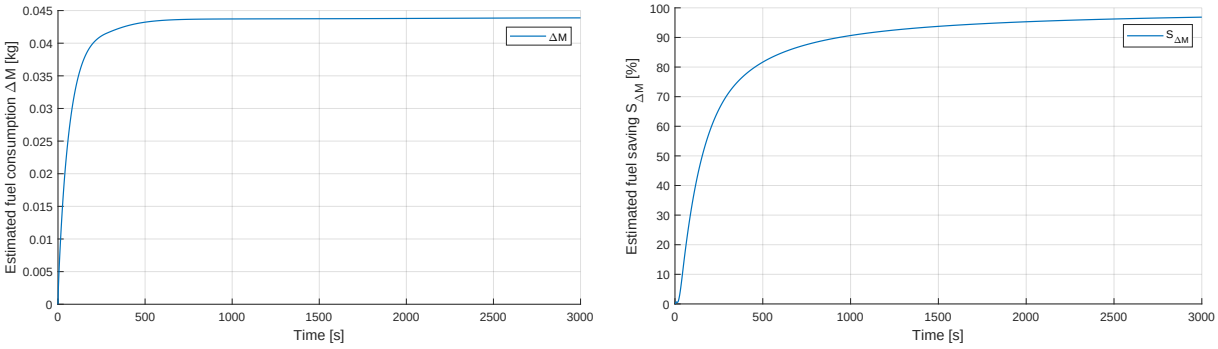


Figure 6.7: Initial convergence and steady-state evolution of control torque for control-based guidance

Although this guidance algorithm manages to achieve the desired outcome, the additional minutes spent to reach the appropriate steady-state values causes a slight increase in fuel consumption. Indeed, the total propellant usage is estimated to be $0.0439\,\mathrm{kg}$, dropping the fuel saving estimate to a still fairly good 96.8542% for the entire simulation time.



(a) Estimated fuel consumption ΔM

(b) Estimated fuel saving $S_{\Delta M}$

Figure 6.8: Estimated fuel consumption (6.8(a)) and saving (6.8(b)) for control-based guidance

Guidance Comparison and Discussion

In conclusion, both scenarios lead to significant improvements in power efficiency due to a sharply decreased fuel consumption. However, the estimate-based solution prevails as it converges to the desired angular velocity faster and overall uses less propellant. To establish a final results comparison and assess the active spin stabilization system performance, Table 6.3 gathers a set of mean steady-state values evaluated for the last $500\,\mathrm{s}$ of simulation time, with total propellant consumption and fuel saving.

Table 6.3: Guidance comparison for selected parameters simulation

Guidance Method	**Variables**				**Fuel**	
	$\|\|\bar{\boldsymbol{\omega}}_e\|\|\,[\mathrm{deg/s}]$	$\|\bar{\phi}_e\|\,[\mathrm{deg}]$	$\|\bar{\theta}_e\|\,[\mathrm{deg}]$	$\|\|\bar{\boldsymbol{\tau}}\|\|\,[\mathrm{N{\cdot}m}]$	$\Delta M\,[\mathrm{kg}]$	$S_{\Delta M}\,[\%]$
No Guidance	8.9489×10^{-5}	6.8067	13.3339	1.6191	1.3949	———
Estimate-Based	1.0032×10^{-4}	8.2106×10^{-4}	1.2594×10^{-3}	1.9728×10^{-4}	0.0261	98.1301
Control-Based	9.1768×10^{-5}	6.7889×10^{-4}	1.1795×10^{-3}	1.7550×10^{-4}	0.0439	96.8542

As can be seen above, either guidance algorithm drives the most important variables to mean values within reach of the gravity-gradient disturbance amplitude (i.e., near an order of magnitude of 10^{-4}). Furthermore, there seems to be a slightly better convergence for the control-based solution, but the increased settling time ends up in an overall performance detriment relative to the estimate-based guidance. Eventually, the control-based solution ought to provide better results for an extended simulation time, as the improved convergence should decrease the steady-state fuel consumption. Still, this difference between methods can be bridged by better tuning the control-based proportional gains. At last, the principal moments of inertia estimates are not discussed as the EKF uses angular velocity measurements produced by an equilibrium spin, which does not give sufficiently rich data to guarantee full-state observability [35], as noticeable in Figure B.5.

On another note, the required time to simulate the different control systems varied depending on whether the navigation filter was included. This is an important factor regarding the guidance method selection for a real space application, as an increased computational load reduces the processing speed and, consequently, shortens the reaction time for a certain maneuver. In fact, as the control-based guidance has fewer mathematical operations than the estimate-based solution (that includes matrix inversion and multiplication), the former method can run the simulation $16.36\times$ faster than the real time, while the latter can only simulate up to $4.36\times$ the real time. Additionally, one should bear in mind that the angular velocity error converges also for no guidance scenarios, as there is still an active controller.

6.2 Monte Carlo Analysis

6.2.1 Setup

To determine if the Control System can converge to the desired values for different initial conditions, while also assessing the sensitivity and robustness of the implemented active spin stabilization solution, a Monte Carlo analysis is carried out.

Several parameters from the previous simulations are preserved in this novel scenario, and can be re-consulted in Table 6.1. However, there are some adjustments to the remaining parameters. For instance, regarding the satellite properties, the real inertia matrix $\boldsymbol{J}$ is now obtained from Sylvester's law of inertia (3.1), where the principal moments of inertia are still the same as in Table 5.1, but the rotation matrix $\boldsymbol{R}$ is computed using Rodrigues' rotation formula (3.5). This method provides the opportunity to precisely define the true rotation angles ϕ and θ within a certain range for each iteration.

Furthermore, the initial diagonal inertia matrix $\boldsymbol{J}_{ctr_0}$ is also defined by the principal moments of inertia, albeit with the introduction of an independent initial error to replicate the control inertia matrix uncertainty. On top of that, the initial angular velocity $\boldsymbol{\omega}_0$ is now arbitrary, i.e., it can take any value in a restricted interval. All the limit conditions for the parameters above are presented in Table 6.4.

Table 6.4: Initial conditions for the Monte Carlo analysis

Topic	**Variable**	**Notation**	**Value**
Satellite Properties	Principal Moments of Inertia	$\boldsymbol{\Lambda}$	$\mathrm{diag}([360 \quad 280 \quad 500])\,\mathrm{kg{\cdot}m^2}$
	Rotation Angle in x-Axis	ϕ	$\pm\,[0;30]\,\mathrm{deg}$
	Rotation Angle in y-Axis	θ	$\pm\,[0;30]\,\mathrm{deg}$
State Vector	Angular Velocity	$\boldsymbol{\omega}_0$	$\pm\,[_{[0;1]\ [0;1]\ [0;1]}]\,\mathrm{deg/s}$
Control System	Control Inertia Matrix	$\boldsymbol{J}_{ctr_0}$	$\pm\,[_{[0;10]\ [0;10]\ [0;10]}]\%\cdot\boldsymbol{\Lambda}\ \mathrm{kg{\cdot}m^2}$

For the initial parameters, the angular velocity $\boldsymbol{\omega}_0$ is determined by random values between -1 and $1\,\mathrm{deg/s}$ independently for all axes, to provide unbiased alignment conditions for each iteration. Both rotation angles ϕ and θ can range from -30 to $30\,\mathrm{deg}$, in order to collect a broad set of different orientations for the principal major axis of inertia. At last, the control inertia matrix $\boldsymbol{J}_{ctr_0}$ is assumed to be calculated accurately enough to simply introduce a maximum $\pm 10\%$ individual error for each moment of inertia.

6.2.2 Results

In agreement with the parameters and initial conditions established in the previous tables, 100 different iterations have been simulated for both guidance algorithms. In order to develop a clear and simple strategy to validate the Control System's reliability, the Monte Carlo analysis is evaluated according to three performance criteria.

Each criterion has been determined to address the most important factors in order to assure that the active spin stabilization solution is achieving the desired results. In specific, the performance criteria are defined by the root-mean-square error (RMSE) for the following variables:

1. Angle between the principal major axis of inertia and the angular velocity $\alpha_{J_3\omega}$
2. Vector norm of the angular velocity tracking error $||\boldsymbol{\omega}_e||$
3. Vector norm of the external torque $||\boldsymbol{\tau}||$

To decide whether the performance criteria are fulfilled, distinct thresholds have been defined for each one. These limit values are selected according to the expected results from several previous tests. For both the angular velocity norm error $||\boldsymbol{\omega}_e||$ and the torque norm $||\boldsymbol{\tau}||$, the maximum error value has been set to $10^{-3}\,\mathrm{deg/s}$ and $10^{-3}\,\mathrm{N{\cdot}m}$, respectively. On the other hand, as seen for the preceding case, the rotation angles do not converge as well as the remaining variables, hence the error for the angle $\alpha_{J_3\omega}$ is delimited by one order of magnitude higher ($10^{-2}\,\mathrm{deg}$).

In case the three performance criteria converge to a steady-state value under the defined thresholds, without ever breaching the convergence envelope, then the Control System can be deemed as stable and robust according to the Monte Carlo analysis. Additionally, other graphics related to the state estimate error are also presented as further proof of system stability in Annex B.2.

Estimate-Based Guidance

For this guidance method, all the performance criteria intersect the threshold and permanently stay inside the convergence envelope in about $500\,s$. There are still some accumulated oscillations for the lower orders of magnitude due to the noise and external disturbances affecting the system, but these do not compromise the main convergence objective, as it can be seen in Figures 6.9, 6.10 and 6.11.

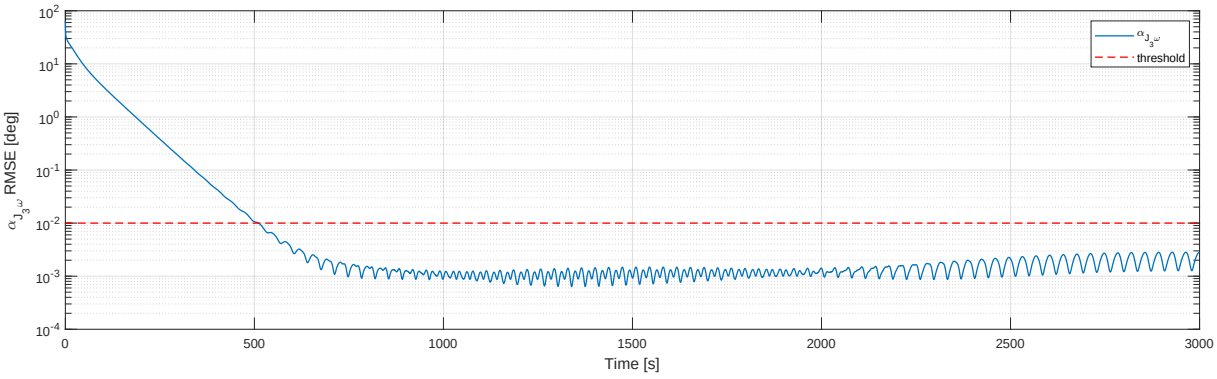


Figure 6.9: Performance criterion #1 - Angle between major inertia axis and angular velocity axis $\alpha_{J_3\omega}$ for estimate-based guidance

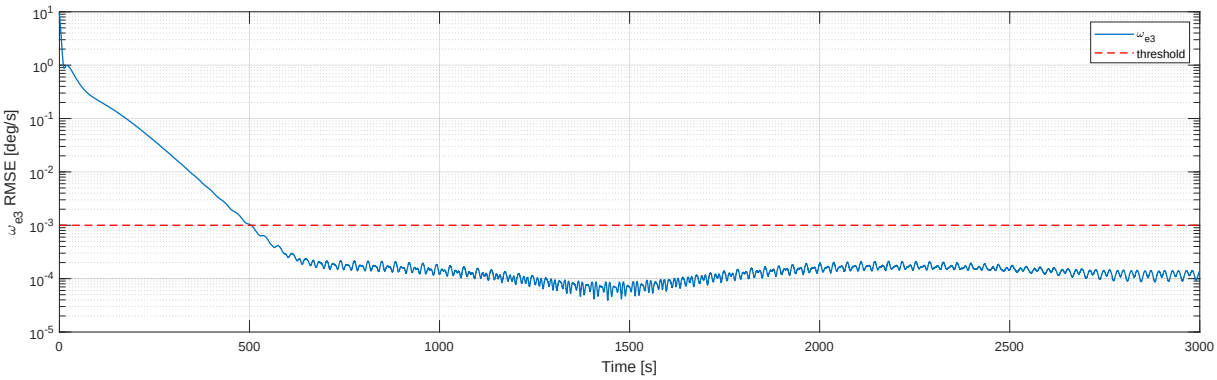


Figure 6.10: Performance criterion #2 - Angular velocity tracking error $||\boldsymbol{\omega}_e||$ for estimate-based guidance

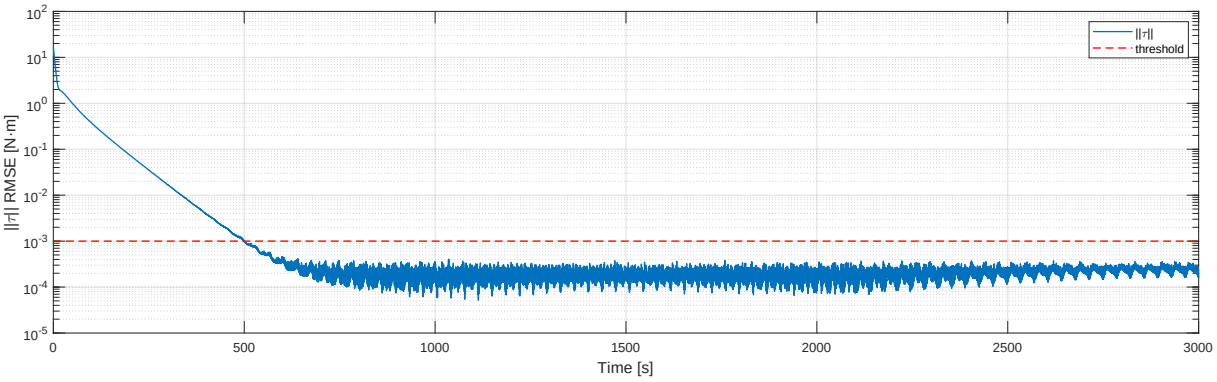
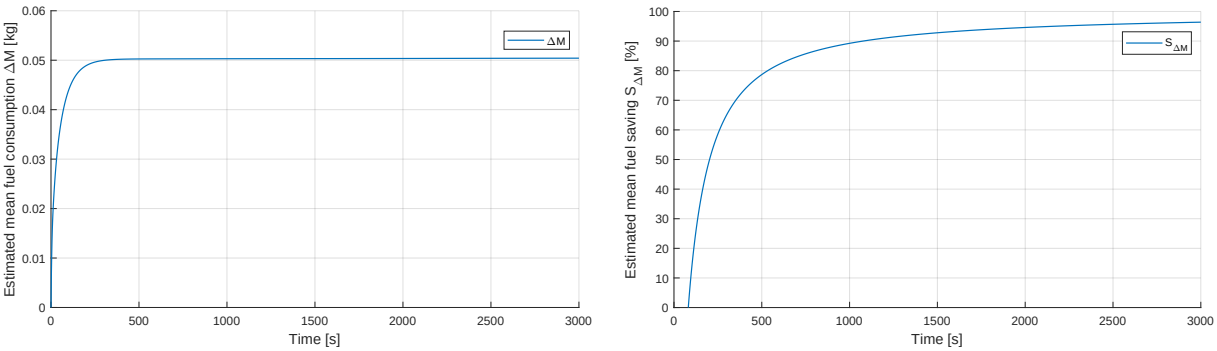


Figure 6.11: Performance criterion #3 - External torque norm $||\boldsymbol{\tau}||$ for estimate-based guidance

At last, the mean fuel consumption and the mean fuel saving estimates are included in this analysis to conclude that the Control System steadily reduces the propellant usage of the spacecraft. Indeed, Figure 6.12 illustrates that the mean total fuel consumption is estimated to be $0.0504\,\mathrm{kg}$, which follows the results obtained in the first scenario. As expected, the fuel saving estimate is 96.3859%, a similar reduction when compared to the estimate-based selected parameters simulation.



(a) Estimated mean fuel consumption $\overline{\Delta M}$

(b) Estimated mean fuel saving $\overline{S_{\Delta M}}$

Figure 6.12: Estimated fuel consumption (6.12(a)) and saving (6.12(b)) for estimate-based guidance

Control-Based Guidance

Similarly, the control-based solution also complies with the three performance criteria, but it takes more time to enter the convergence envelope and settle to a constant value. In fact, the average threshold crossing time for this guidance method is approximately double of the estimate-based one, i.e., over around $1000\,\mathrm{s}$.

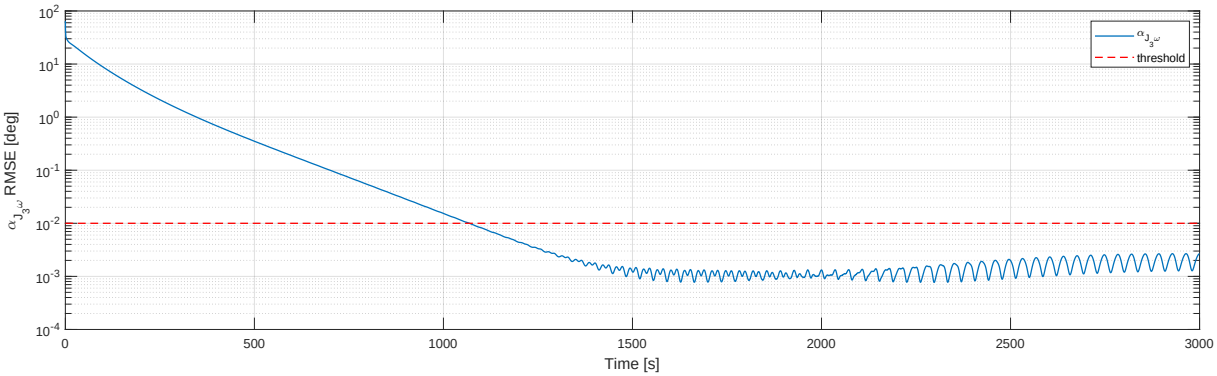


Figure 6.13: Performance criterion #1 - Angle between major inertia axis and angular velocity axis $\alpha_{J_3\omega}$ for control-based guidance

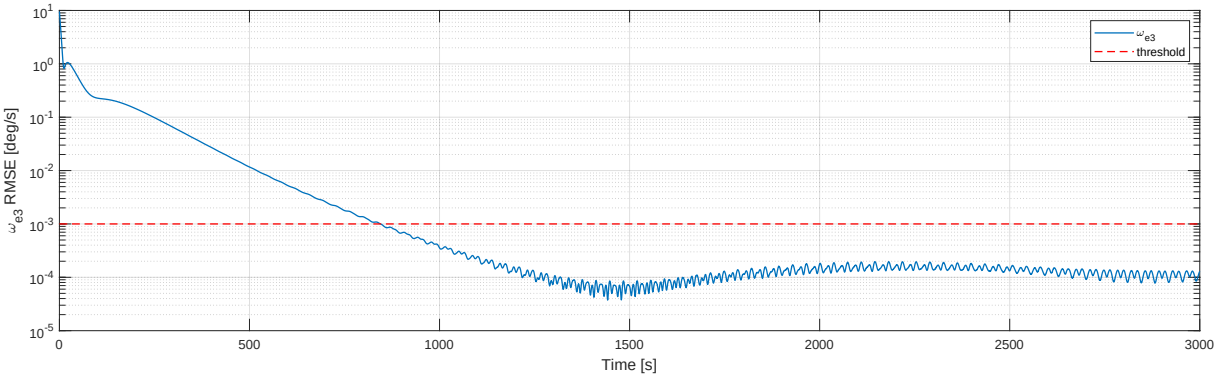


Figure 6.14: Performance criterion #2 - Angular velocity tracking error $||\boldsymbol{\omega}_e||$ for control-based guidance

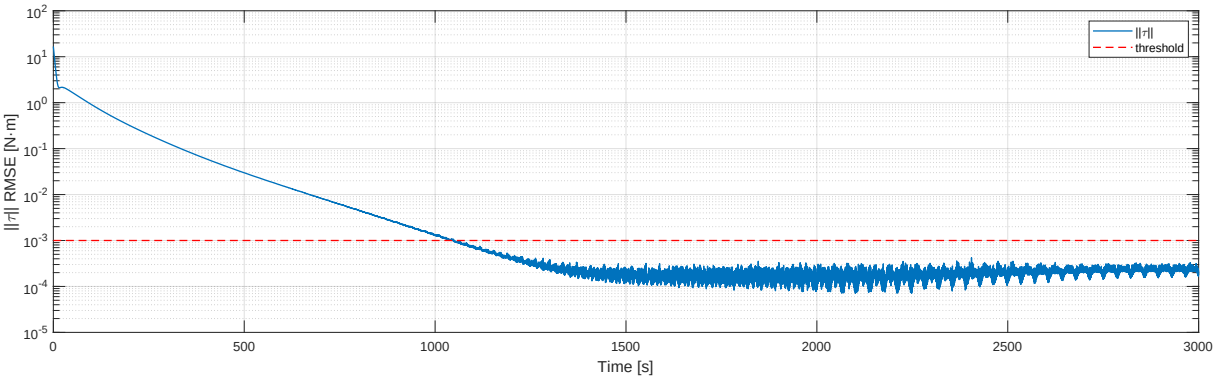
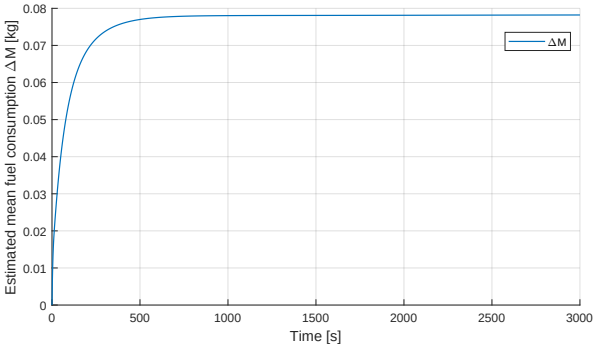
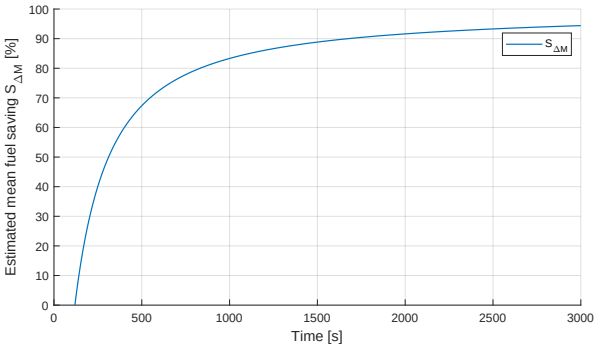


Figure 6.15: Performance criterion #3 - External torque $||\boldsymbol{\tau}||$ for control-based guidance

Once again, the increased time to reach the desired steady-state evolution translates to an increased fuel consumption. Indeed, the total mean propellant expelled mass estimate is $0.0782\,\mathrm{kg}$, thus resulting in an estimated reduction of 94.3933% in energy usage for the total simulation time.



(a) Estimated mean fuel consumption $\overline{\Delta M}$



(b) Estimated mean fuel saving $\overline{S_{\Delta M}}$

Figure 6.16: Estimated fuel consumption (6.16(a)) and saving (6.16(b)) for control-based guidance

Guidance Comparison and Discussion

To sum up, the Monte Carlo analysis provides stronger evidence to safely conclude that both guidance methods lead to major improvements in power efficiency for a great variety of initial conditions and parameters. In addition, the same reasoning for the best Control System still applies, as the introduction of an EKF seems to generally drive the spacecraft to converge twice as fast to the performance criteria, thus reducing propellant usage and increasing the space mission duration. However, it should be noted that these results may also simply reflect an inadequate tuning of the control-based proportional gains.

As a final analysis comparison, the mean RMSE values of the three performance criteria are evaluated for the last $1500\,\mathrm{s}$ of simulation time, as all variables exhibit a steady-state evolution in that time interval. Alongside the mean fuel consumption and saving, these mean values are included in Table 6.5.

Table 6.5: Guidance comparison for Monte Carlo analysis

Guidance Method	**Performance Criteria**			**Fuel**	
	$\|\|\bar{\boldsymbol{\omega}}_e\|\|\,[\mathrm{deg/s}]$	$\|\bar{\alpha}_{J_3\omega}\|\,[\mathrm{deg}]$	$\|\|\bar{\boldsymbol{\tau}}\|\|\,[\mathrm{N{\cdot}m}]$	$\Delta M\,[\mathrm{kg}]$	$S_{\Delta M}\,[\%]$
No Guidance	8.9489×10^{-5}	15.3385	1.6191	1.3949	———
Estimate-Based	1.3561×10^{-4}	1.5054×10^{-3}	1.9518×10^{-4}	0.0504	96.3859
Control-Based	1.2627×10^{-4}	1.4030×10^{-3}	1.8345×10^{-4}	0.0782	94.3933

From the listed variables, all performance criteria seem to follow the same trend as in the selected parameters simulation, since the torque norm and the angular velocity error converge to an order of magnitude of 10^{-4}, and the angle $\alpha_{J_3\omega}$ settles for a one-order-higher magnitude. On the contrary, the convergence difference between the guidance methods declines for the Monte Carlo analysis. In fact, it is also noticeable in the mean RMSEs, as they tend to settle for relatively higher values. This circumstance arises from the favorable initial conditions and real values of the first scenario, i.e. the initial angular velocity and rotation angles were near the desired outcome. Additionally, no "worst cases" are made available, since all iterations are close to the mean RMSEs, as can be seen in Annex B.2.

Therefore, the estimate-based guidance can be considered the best method for the tested parameters and gains, due to its reduced fuel usage and a non-significant performance difference w.r.t. the other solution. Nevertheless, the computational load imposed by the EKF is a considerable drawback.

Chapter 7

Conclusions

In this dissertation, the main objective was to study, design, implement and test an active spin stabilization system, able to align the satellite's spinning axis to the principal major axis of inertia, and therefore reduce the fuel consumption and increase the space mission time.

To accomplish that goal, the first step was to review the existing literature regarding attitude representations, reference frames, kinematics and dynamics. In addition, several external disturbances, sensors and actuators were also researched during this theoretical background analysis. Subsequently, the problem statement was addressed, presenting the principal axes of inertia definition and the inertia tensor computation, following with the mathematical argument for spin stability about the principal axes.

Afterwards, Chapter 4 introduced the active spin stabilization proposed solution, which consists of a navigation filter provided by a continuous-time EKF, a guidance algorithm implemented according to two alternatives - estimate-based and control-based guidance, and a control law defined by a PI controller. Later, the respective MATLAB®/Simulink® system implementation was also described in detail.

Ultimately, two different tests were developed for both guidance solutions to assess the Control System performance, while using the realistic simulation environment. In the first case, a single scenario with selected reasonable values for the initial conditions and parameters was simulated. On the other hand, the second case was a Monte Carlo analysis that addressed 100 different iterations to validate the robustness and steady-state convergence of the system under different circumstances.

For the selected parameters simulation, both guidance algorithms converged to the desired solution, resulting in a significant increase in power efficiency. The estimate-based solution leads to 98.1301% less propellant consumption while the control-based solution has a fuel saving percentage of 96.8542%. Therefore, although control-based guidance errors converge slightly closer to zero, the faster convergence and increased power saving distinguishes the estimate-based guidance as the better performing method. However, one important drawback is the increased computational load imposed by the EKF, which results in an extended time to simulate this solution.

In regard to the Monte Carlo analysis, the obtained results corroborate and serve as further evidence that both guidance methods are able to considerably reduce fuel consumption, even for different initial conditions and parameters. Similarly, the introduced performance criteria seem to follow the selected

parameters simulation and converge to values within the same order of magnitude. Therefore, the estimate-based guidance can be considered the preferred method, due to its reduced total propellant usage and no significant performance difference from the control-based guidance. Nevertheless, the former solution extended time to simulate must be looked upon as the computational efficiency is an important factor when deciding which guidance algorithm to implement in the spacecraft.

As a concluding remark, it should be noted that the obtained results are dependent on the parameters and gains previously defined and listed in Table 5.1, implying that other values could generate slightly different outcomes. In fact, after a more exhaustive analysis to fine-tune the guidance proportional gains, it should be possible to decrease the settling time for the control-based method variables. Moreover, the simulated cases may vary from the real scenarios, since the thrusters also have a minimum limit which is certainly higher than the simulated torque steady-state values, and the space environment would have several more external disturbances acting on the satellite.

7.1 Future Work

Further improvements under the scope of this dissertation include the possibility of testing the simulation environment for a non-constant inertia tensor, i.e., to include small variations in the inertia matrix and evaluate the Control System ability to adapt and still converge for this dynamic condition. In addition, an actual implementation has physical actuator limitations, determining that the thrusters must be turned on for a minimum time, which could become an issue when the commanded torque is too small. Therefore, a minimum limit value could be defined to incorporate a dead-zone into the actuation system and, consequently, represent more realistic actuators models, as described above.

Another important avenue of future development is the deduction of a new Lyapunov function that includes the dynamic angular velocity reference, in order to theoretically assess the Control System stability. Finally, depending on the space mission, the spinning axis orientation w.r.t. the inertial frame can be an important factor to control and, as such, one last step is to upgrade the current state vector to also include the attitude, in order to take advantage of the fuel saving provided by this GNC solution and replace the original main GNC/AOCS system.

Bibliography

[1] D. C. S. Draper, D. W. Wrighley, D. G. Hoag, D. R. H. Battin, J. E. Miller, D. A. Koso, D. A. L. Hopkins, and D. W. E. V. Velde. Apollo: Guidance and Navigation. Technical report, MIT, 1965.

[2] J. M. Dumoulin. NSTS Shuttle Reference Manual. Technical report, NASA, 1988.

[3] M. Malcolm and V. Badescu. *The International Handbook of Space Technology*. Springer, 1^{st} edition, 2014. ISBN:9783642411007.

[4] X. Xia, G. Sun, K. Zhang, S. Wu, T. Wang, L. Xia, and S. Liu. NanoSats/CubeSats ADCS survey. *29th Chinese Control and Decision Conference (CCDC)*, 2017.

[5] J. D. Biggs and N. Horri. Optimal geometric motion planning for a spin-stabilized spacecraft. *Systems & Control Letters*, 2012.

[6] J. Bouwmeester and J. Gu. Survey of worldwide pico- and nanosatellite missions, distributions and subsystem technology. *Acta Astronautica*, 2010.

[7] J. Guo and C. Han. Where is the limit: The analysis of CubeSat ADCS performance. *4S Symposium*, 2016.

[8] M. J. Sidi. *Spacecraft Dynamics & Control: A Practical Engineering Approach*. Cambridge University Press, 1^{st} edition, 1997. ISBN:9780511815652.

[9] J. W. Wertz. *Spacecraft Attitude Determination and Control*. Astrophysics and Space Science Library, 1^{st} edition, 1978. ISBN:9789027712042.

[10] Y. Yang. *Spacecraft Modelling, Attitude Determination, and Control: Quaternion-Based Approach*. CRC Press, 1^{st} edition, 2019. ISBN:9781138331501.

[11] F. L. Markley and J. L. Crassidis. *Fundamentals of Spacecraft Attitude Determination and Control*. Springer, 1^{st} edition, 2014. ISBN:9781493908011.

[12] H. D. Black. A passive system for determining the attitude of a satellite. *AIAA Journal, 2(7)*, 1964.

[13] G. Black. A Least Squares Estimate of Satellite Attitude. *SIAM Review, 7(3)*, 1965.

[14] P. B. Davenport. A vector approach to the algebra of rotations with applications. Technical report, NASA, 1968.

[15] M. D. Shuster. Approximate algorithms for fast optimal attitude computation. *Guidance and Control Conference*, 1978.

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

[17] F. L. Markley and D. Mortari. Quaternion Attitude Estimation Using Vector Observations. *The Journal of the Astronautical Sciences, 48(2-3)*, 2000.

[18] J. L. Crassidis, F. L. Markley, and Y. Cheng. Survey of Nonlinear Attitude Estimation Methods. *Journal of Guidance, Control, and Dynamics, 30(1)*, 2007.

[19] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering, 82(1)*, 1960.

[20] R. E. Kalman. New Results in Linear Filtering and Prediction Theory. *Journal of Basic Engineering, 83(1)*, 1961.

[21] J. L. Farrell. Attitude Determination by Kalman Filteringn. Technical report, NASA, 1967.

[22] E. Lefferts, F. Markley, and M. Shuster. Kalman Filtering for Spacecraft Attitude Estimation. *20th Aerospace Sciences Meeting*, 1982.

[23] F. L. Markley. Multiplicative vs. Additive Filtering for Spacecraft Attitude Estimation. Technical report, NASA, 2003.

[24] M. D. Shuster. A Simple Kalman Filter and Smoother for Spacecraft Attitude. *The Journal of the Astronautical Sciences*, 1989.

[25] I. Y. Bar-Itzhack. REQUEST: A Recursive QUEST Algorithm for Sequential Attitude Determination. *Journal of Guidance, Control, and Dynamics, 19(5)*, 1996.

[26] J. L. Crassidis and F. L. Markley. Unscented Filtering for Spacecraft Attitude Estimation. *Journal of Guidance, Control, and Dynamics, 26(4)*, 2003.

[27] G. Haupt, N. Kasdin, G. Keiser, and B. Parkinson. An Optimal Recursive Iterative Algorithm for Discrete Nonlinear Least Squares Estimation. *Guidance, Navigation and Control Conference*, 1995.

[28] N. J. Kasdin and T. Weaver. Recursive Satellite Attitude Estimation with the Two-Step Optimal Estimator. *AIAA Guidance, Navigation and Control Conference and Exhibit*, 2002.

[29] R. E. Roberson. Two Decades of Spacecraft Attitude Control. *Journal of Guidance, Control, and Dynamics, 2(1)*, 1979.

[30] A. Zhang, Y. Wang, Z. Zhang, and H. Karimi. Robust Control Allocation for Spacecraft Attitude Stabilization under Actuator Faults and Uncertainty. *Mathematical Problems in Engineering*, 2014.

[31] A. Safa, M. Baradarannia, H. Kharrati, and S. Khanmohammadi. Constrained Robust Control for Spacecraft Attitude Stabilization under Actuator Delays and Faults. *Journal of Dynamic Systems, Measurement, and Control, 139(5)*, 2017.

[32] L. Euler. *Theoria motus corporum solidorum seu rigidorum*. Novi Commentarii academiae scientiarum Petropolitanae, 1^{st} edition, 1765.

[33] A. Y. Lee and J. A. Wertz. In-Flight Estimation of the Cassini Spacecraft's Inertia Tensor. *Journal of Spacecraft and Rockets, 39(1)*, 2002.

[34] J. A. Keim, A. B. Aqikmese, and J. F. Shields. Spacecraft Inertia Estimation Via Constrained Least Squares. *2006 IEEE Aerospace Conference*, 2006.

[35] M. C. Norman and M. A. Peck. In-Orbit Estimation of Inertia and Momentum-Actuator Alignement Parameters. *Journal of Guidance, Control, and Dynamics, 34(6)*, 2011.

[36] R. Linares, F. A. Leve, M. K. Jah, and J. L. Crassidis. Space Object Mass-Specific Inertia Matrix Estimation from Photometric Data. *Advances in the Astronautical Sciences*, 2012.

[37] C. Nainer, H. Garnier, M. Gilson, H. Evain, and C. Pittet. In-Flight Inertia Matrix Estimation of a Gyroless Satellite. *2019 Conference in Guidance, Navigation & Control in Aerospace (EuroGNC19)*, 2019.

[38] M. D. Shuster. A Survey of Attitude Representations. *The Journal of Astronautical Sciences*, 1993.

[39] J. Diebel. Representing Attitude: Euler Angles, Unit Quaternions, and Rotation Vectors. Technical report, Stanford University, 2006.

[40] B. Wie. *Space Vehicle Dynamics and Control*. American Institute of Aeronautics and Astronautics, 2^{nd} edition, 2008. ISBN:9781563479533.

[41] G. Lavezzi. Image processing of multiclass satellite tracklets for initial orbit determination based on optical telescopes. Technical report, Politecnico di Milano, 2018.

[42] N. Bloise, E. Capello, M. Dentis, and E. Punta. Obstacle Avoidance with Potential Field Applied to a Rendezvous Maneuver. *Applied Sciences, 7(10)*, 2017.

[43] P. C. Hughes. *Spacecraft Attitude Dynamics*. Courier Corporation, 1^{st} edition, 2004. ISBN:9780486140131.

[44] R. A. Serway and J. W. Jewett. *Physics for Scientists and Engineers*. Books/Cole Thomson Learing, 1^{st} edition, 2004. ISBN:9781133954071.

[45] J. J. Sylvester. A demonstration of the theorem that every homogeneous quadratic polynomial is reducible by real orthogonal substitutions to the form of a sum of positive and negative squares. *Philosophical Magazine and Journal of Science, 4(23)*, 1852.

[46] P. S. Maybeck. Stochastic Models, Estimation, and Control. *Academic Press*, 1979.

[47] D. Simon. *Optimal State Estimation: Kalman, H_∞, and Nonlinear Approaches*. John Wiley & Sons, 1^{st} edition, 2006. ISBN:9780471708582.

[48] J. L. Crassidis and J. L. Junkins. *Optimal Estimation of Dynamic Systems*. CRC Press, 2^{nd} edition, 2012. ISBN:9781439839867.

[49] J. E. Slotine and W. Li. *Applied Nonlinear Control*. Prentice Hall, 1^{st} edition, 1991. ISBN:9780130408905.

[50] S. Gros, M. Zanon, and M. Diehl. Baumgarte stabilisation over the SO(3) rotation group for control. *54th IEEE Conference on Decision and Control (CDC)*, 2015.

Appendix A

Review of Notations

This appendix is intended to contextualize some of the notations and conventions adopted in this dissertation. Other mathematical concepts, definitions or operations are reviewed, as adequate.

A.1 Cross Product Representation

For two arbitrary vectors $\boldsymbol{u}$ and $\boldsymbol{v}$ ($\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^3$), the cross product is written as

$$\boldsymbol{u} \times \boldsymbol{v} = [\boldsymbol{u}\times] \cdot \boldsymbol{v} \tag{A.1}$$

with

$$[\boldsymbol{u}\times] = \begin{bmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{bmatrix} \tag{A.2}$$

In this work, $[\boldsymbol{u}\times]$ can also be represented by the notation $\boldsymbol{S}(\boldsymbol{u})$.

A.2 Quaternion Definition, Properties and Operations

In mathematics, the quaternion number system extends the complex numbers and is frequently used as a parameterization of a rigid body attitude. Initially devised by William Rowan Hamilton in 1844, now there are different conceptual notations for the quaternion definition, properties and operations [10, 11].

Hereinafter, a quaternion $\boldsymbol{q} \in \mathbb{H}$ will be considered a four-component vector with a three-vector part $\boldsymbol{q_v}$ and a scalar part q_4:

$$\boldsymbol{q} = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 \end{bmatrix}^T = \begin{bmatrix} \boldsymbol{q_v} \\ q_4 \end{bmatrix} \tag{A.3}$$

This definition is conceptually different from the one first introduced by Hamilton, resulting in distinct properties and operations used throughout this work.

Equality and Addition

Let

$$\boldsymbol{p} = \boldsymbol{i} \cdot p_1 + \boldsymbol{j} \cdot p_2 + \boldsymbol{k} \cdot p_3 + p_4$$

and

$$\boldsymbol{q} = \boldsymbol{i} \cdot q_1 + \boldsymbol{j} \cdot q_2 + \boldsymbol{k} \cdot q_3 + q_4$$

be two quaternions, then the two quaternions are equal if and only if

$$p_1 = q_1, \quad p_2 = q_2, \quad p_3 = q_3, \quad p_4 = q_4$$

The *sum of two quaternions* is defined as

$$\boldsymbol{p} + \boldsymbol{q} = \boldsymbol{i} \cdot (p_1 + q_1) + \boldsymbol{j} \cdot (p_2 + q_2) + \boldsymbol{k} \cdot (p_3 + q_3) + (p_4 + q_4) \tag{A.4}$$

Multiplication and the Identity

A striking feature of quaternions is that the product of two quaternions is associative, distributive, but non-commutative, meaning that it depends on which factor is to the left of the multiplication sign and which factor is to the right.

Let $\boldsymbol{p}$ and $\boldsymbol{q}$ be defined as before, the *multiplication of two quaternions* $\boldsymbol{p}$ and $\boldsymbol{q}$ is given by

$$\boldsymbol{p} \otimes \boldsymbol{q} = p_4 \cdot \boldsymbol{q_v} + q_4 \cdot \boldsymbol{p_v} + \boldsymbol{p} \times \boldsymbol{q} + p_4 \cdot q_4 - \boldsymbol{p_v} \cdot \boldsymbol{q_v} \tag{A.5}$$

with the vector part $p_4 \cdot \boldsymbol{q_v} + q_4 \cdot \boldsymbol{p_v} + \boldsymbol{p} \times \boldsymbol{q}$ and scalar part $p_4 \cdot q_4 - \boldsymbol{p_v} \cdot \boldsymbol{q_v}$.

Similarly to a rotation matrix and to the Euler angles, quaternion products can also be represented by matrix multiplication, very much like the cross product:

$$\boldsymbol{p} \otimes \boldsymbol{q} = \boldsymbol{Q}(\boldsymbol{p}) \cdot \boldsymbol{q} \tag{A.6}$$

where $\boldsymbol{Q} : \mathbb{H} \longrightarrow \mathbb{R}^{4\times4}$, known as the quaternion matrix function, is defined as

$$\boldsymbol{Q}(\boldsymbol{q}) = \begin{bmatrix} q_4 \cdot \boldsymbol{I_3} + [\boldsymbol{q_v}\times] & \boldsymbol{q_v} \\ -\boldsymbol{q_v}^T & q_4 \end{bmatrix} \tag{A.7}$$

The *identity quaternion* is defined as $\boldsymbol{I_q} = [0, 0, 0, 1]^T$, i.e., vector part zero and scalar part one, which obeys $\boldsymbol{I_q} \otimes \boldsymbol{q} = \boldsymbol{q} \otimes \boldsymbol{I_q}$, as required of the identity.

Complex Conjugate, Norm, and Inverse

The *complex conjugate* of quaternion $\boldsymbol{q}$ is denoted and obtained by

$$\boldsymbol{q}^* = \begin{bmatrix} \boldsymbol{q_v} \\ q_4 \end{bmatrix}^* = \begin{bmatrix} -\boldsymbol{q_v} \\ q_4 \end{bmatrix} \tag{A.8}$$

Then, it is straightforward to compute the sum of the quaternion and its conjugate as

$$\boldsymbol{q} + \boldsymbol{q}^* = (\boldsymbol{q_v} + q_4) + (-\boldsymbol{q_v} + q_4) = 2q_4 \tag{A.9}$$

The product of a quaternion with its conjugate is equal to the square of its norm times the identity quaternion

$$\boldsymbol{q} \otimes \boldsymbol{q}^* = \boldsymbol{q}^* \otimes \boldsymbol{q} = ||\boldsymbol{q}||^2 \cdot \boldsymbol{I}_q \tag{A.10}$$

The *conjugate of the product* of two quaternions $\boldsymbol{p}$ and $\boldsymbol{q}$ is the product of the conjugates in the opposite order $(\boldsymbol{p} \otimes \boldsymbol{q})^* = \boldsymbol{q}^* \otimes \boldsymbol{p}^*$. This relation, combined with equation (A.10) and the associativity of quaternion multiplication, can be used to infer that the *norm of the product* of two quaternions $\boldsymbol{p}$ and $\boldsymbol{q}$ is the product of the individual norms

$$||\boldsymbol{p} \otimes \boldsymbol{q}|| = ||\boldsymbol{p}|| \cdot ||\boldsymbol{q}|| \tag{A.11}$$

The *norm* of a quaternion is defined as $||\boldsymbol{q}|| = \sqrt{\boldsymbol{q}^* \otimes \boldsymbol{q}}$. It is also simple to verify that the norm satisfies $||\boldsymbol{q}|| = \sqrt{q_1^2 + q_2^2 + q_3^2 + q_4^2}$.

The *inverse* of a non-zero norm quaternion is defined by

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

so that $\boldsymbol{q} \otimes \boldsymbol{q}^{-1} = \boldsymbol{q}^{-1} \otimes \boldsymbol{q} = \boldsymbol{I_q}$, as required by the definition of an inverse.

Finally, the *inverse of the product* of two quaternions $\boldsymbol{p}$ and $\boldsymbol{q}$ is the product of the inverses in the opposite order

$$(\boldsymbol{p} \otimes \boldsymbol{q})^{-1} = \boldsymbol{q}^{-1} \otimes \boldsymbol{p}^{-1} \tag{A.13}$$

For a normalized quaternion, which satisfies $||\boldsymbol{q}|| = 1$, it is also true that $\boldsymbol{q}^{-1} = \boldsymbol{q}^*$.

Appendix B

Complementary Graphics

This appendix is intended to present additional graphics to complement the discussion of obtained results in Chapter 6. Other details and information are provided for both simulation cases, as adequate.

B.1 Selected Parameters Simulation

Figures B.1 and B.2 present the noise interference for the angular velocity measurements and the total external disturbances torque, respectively. These perturbations are added to all the simulations.

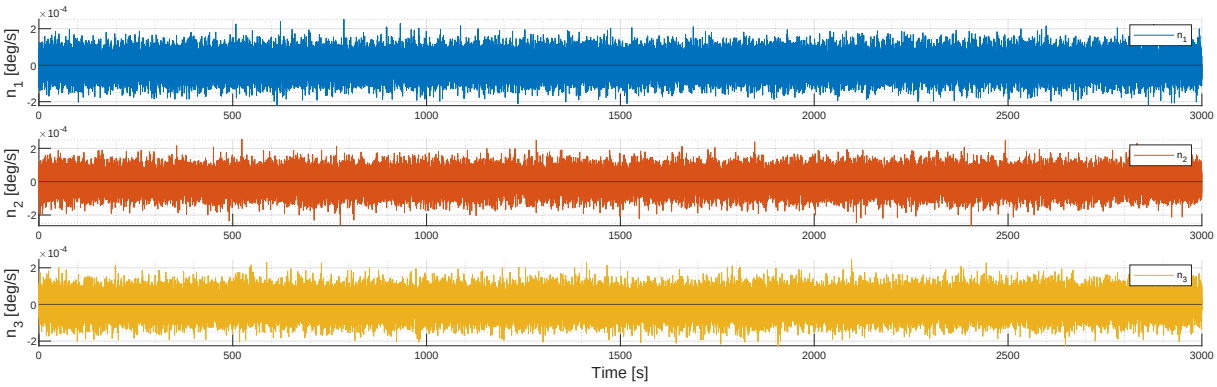


Figure B.1: Noise interference for angular velocity measurements in x-axis n_1, y-axis n_2 and z-axis n_3

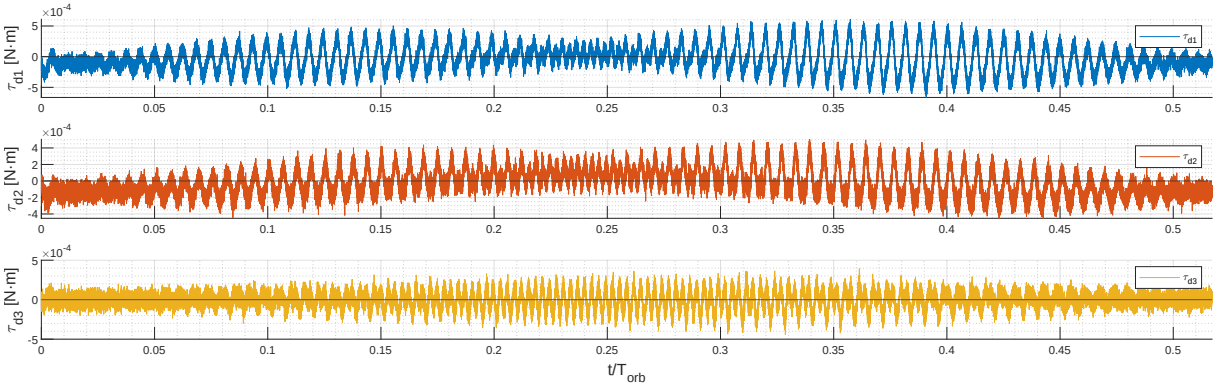


Figure B.2: External disturbances torque in x-axis τ_{d_1}, y-axis τ_{d_2} and z-axis τ_{d_3}

On the other hand, Figures B.3, B.4 and B.5 introduce the angular velocity, rotation angles and principal moments of inertia estimate errors, respectively. These variables form the state-vector used for the EKF implemented in the estimate-based guidance method.

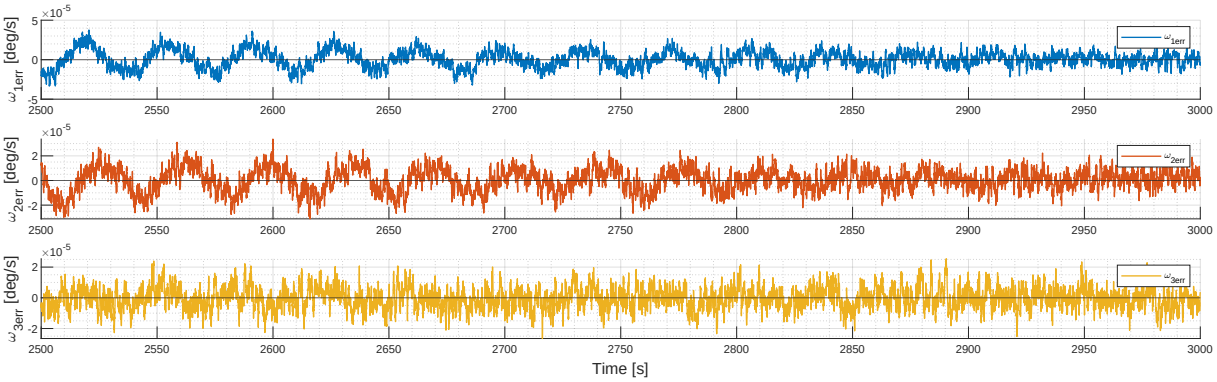


Figure B.3: Steady-state evolution of the angular velocity estimate error $\boldsymbol{\omega}_{err}$

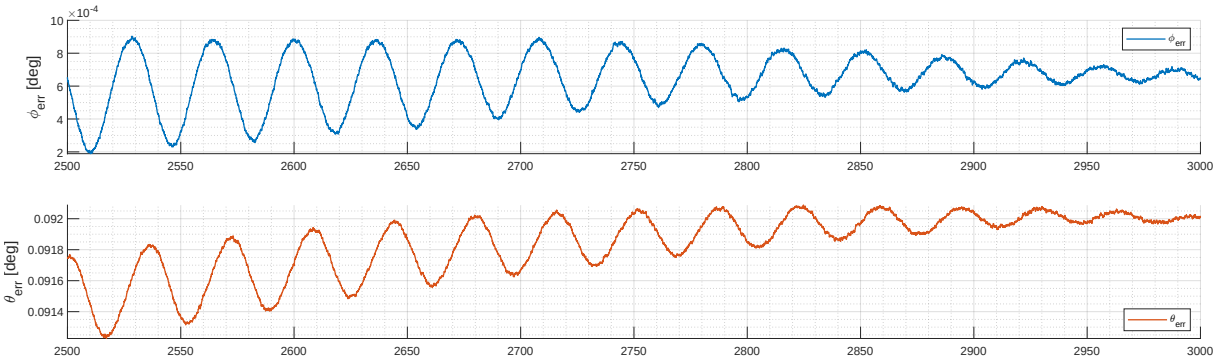


Figure B.4: Steady-state evolution of the rotation angles estimate error ϕ_{err} and θ_{err}

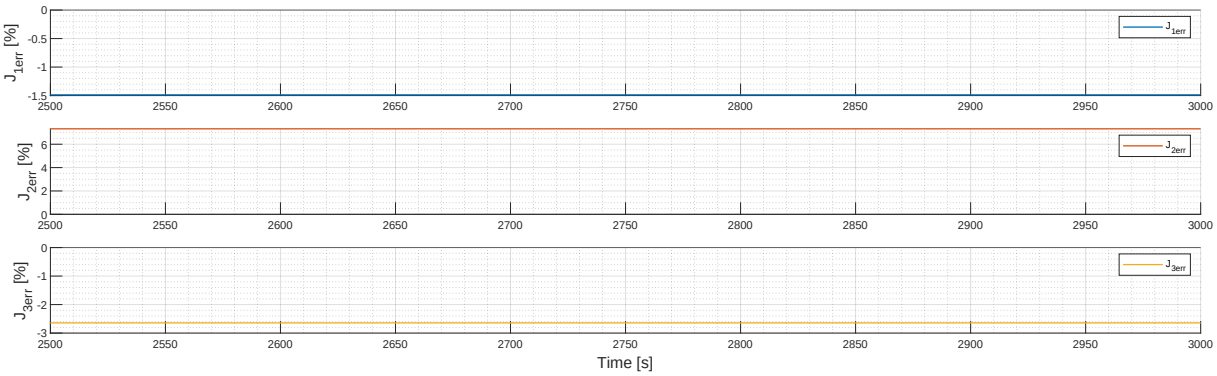


Figure B.5: Steady-state evolution of the principal inertia moments estimate error $J_{1_{err}}$, $J_{2_{err}}$ and $J_{3_{err}}$

B.2 Monte Carlo Analysis

In this section, supplementary graphs are included to further assure the Control System' robustness and effectiveness for the 100 different iterations. Indeed, to ensure that the system is converging to the desired angular velocity and, consequently, reducing fuel consumption, the best variable to highlight is the control torque, as its convergence to zero confirms alignment with the principal major axis of inertia.

Therefore, Figures B.6, B.7 and B.8 express all the simulated estimate-based values for the control torque in x-axis, y-axis and z-axis, respectively. As it can be observed, every single iteration converges within the amplitude of the most significant perturbation - the gravity-gradient disturbance torque.

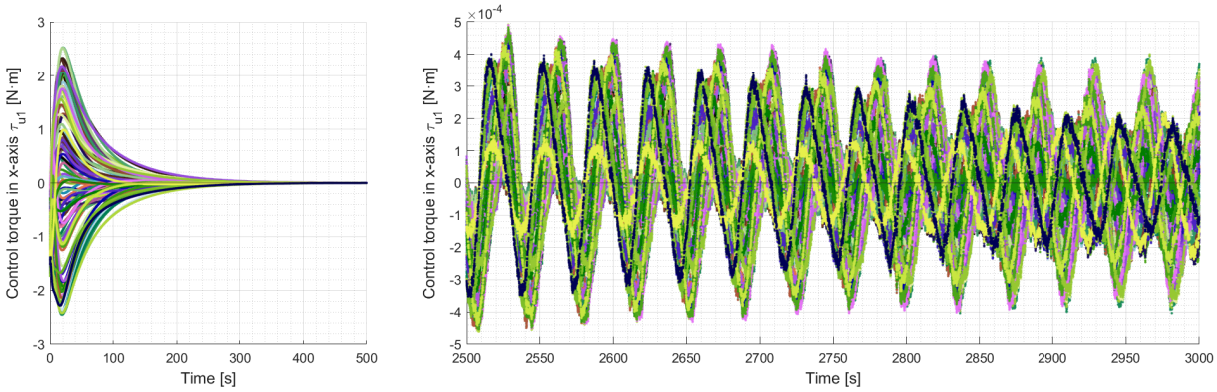


Figure B.6: Initial convergence and steady-state evolution of control torque x-axis τ_{u_1} for estimate-based guidance

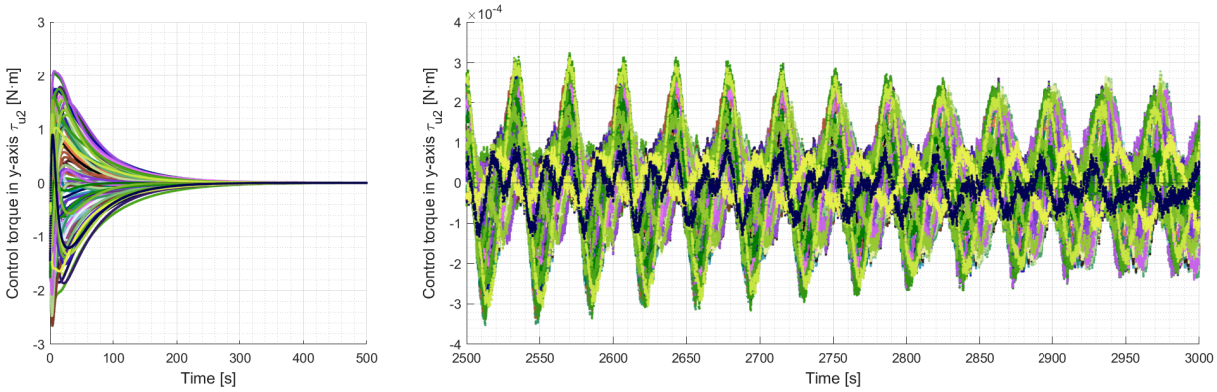


Figure B.7: Initial convergence and steady-state evolution of control torque in y-axis τ_{u_2} for estimate-based guidance

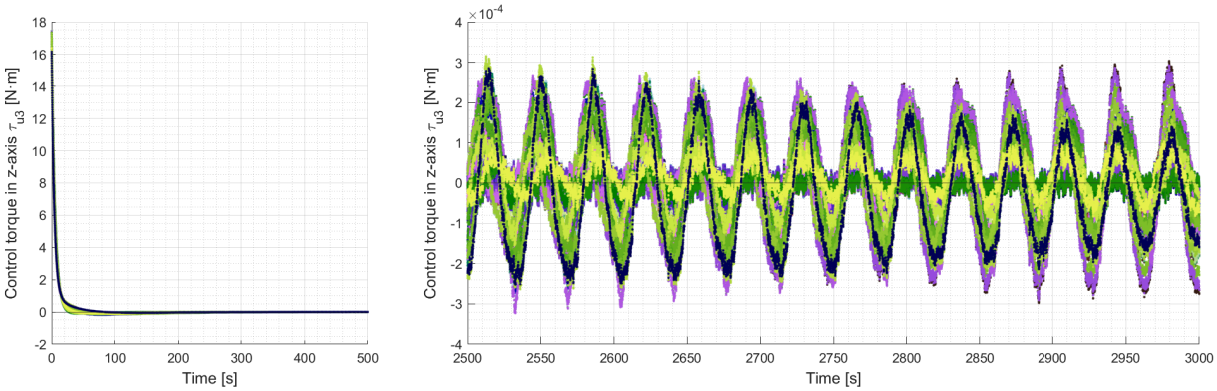


Figure B.8: Initial convergence and steady-state evolution of control torque z-axis τ_{u_3} for estimate-based guidance

Similarly, all the simulated control-based values for the control torque in x-axis, y-axis and z-axis are available below, as can be seen in Figures B.9, B.10 and B.11, respectively.

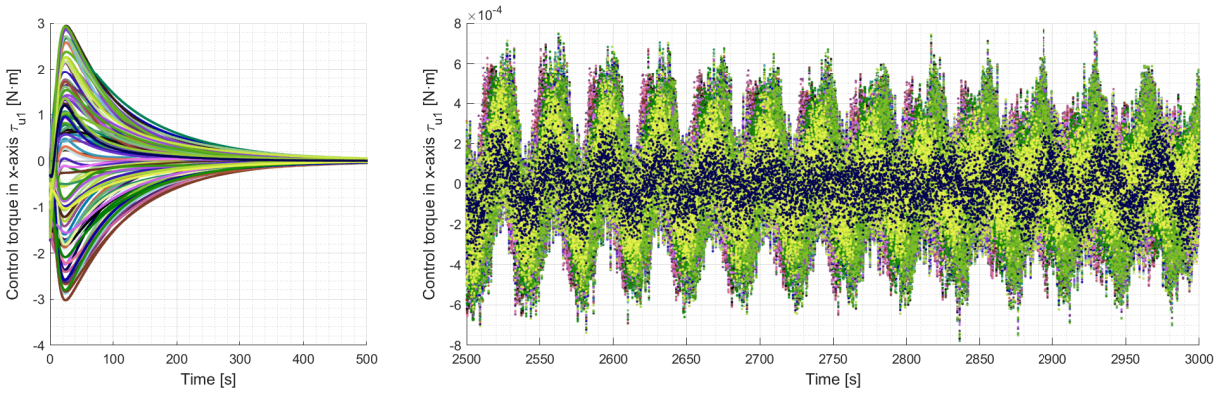


Figure B.9: Initial convergence and steady-state evolution of control torque x-axis τ_{u_1} for control-based guidance

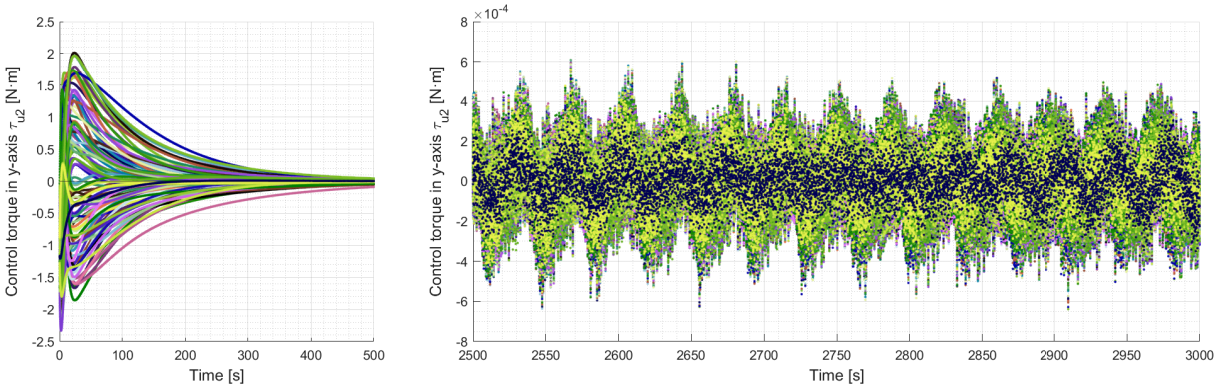


Figure B.10: Initial convergence and steady-state evolution of control torque y-axis τ_{u_2} for control-based guidance

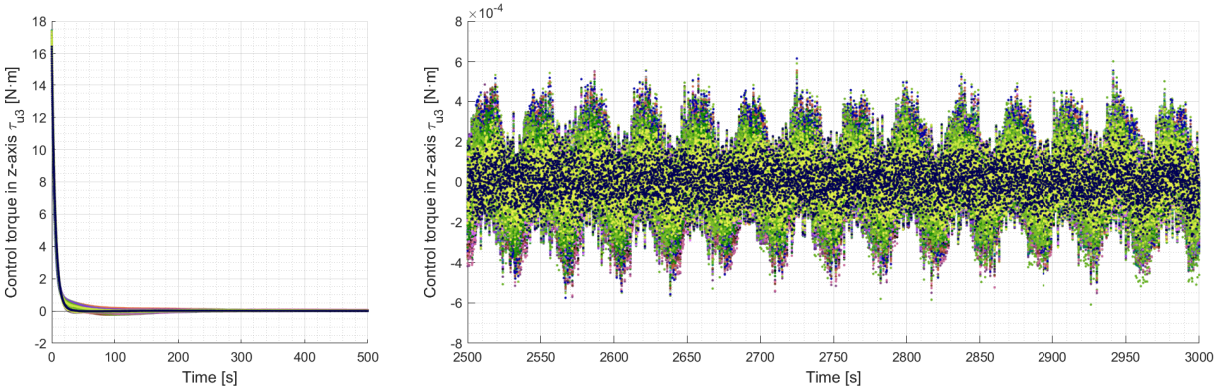


Figure B.11: Initial convergence and steady-state evolution of control torque z-axis τ_{u_3} for control-based guidance

In the last graphs for the control torque in the z-axis (B.8) and (B.11), it becomes evident that the initial torque is considerably larger than the others. This circumstance arises from the close proximity of this axis with the desired spinning axis, thus requiring a powerful initial boost to increase the angular velocity around the z-axis and take it to the desired value.

Finally, to demonstrate the EKF accuracy for state-vector estimation, the next figures provide the estimate error for the angular velocity in the x-axis (B.12), y-axis (B.13) and z-axis (B.14), and for the rotation angles in the x-axis (B.15) and y-axis (B.16). These plots contain all the obtained estimates for the 100 different simulated iterations.

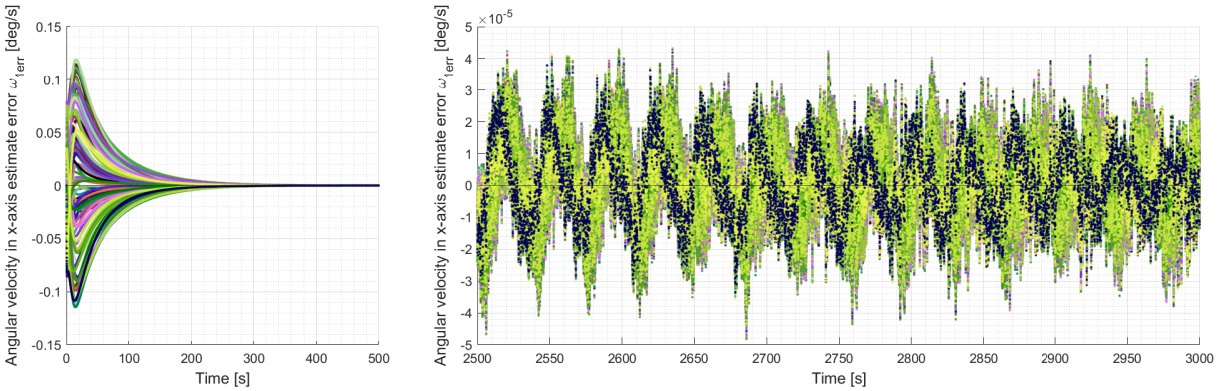


Figure B.12: Initial convergence and steady-state evolution of angular velocity x-axis estimate error $\omega_{1_{err}}$

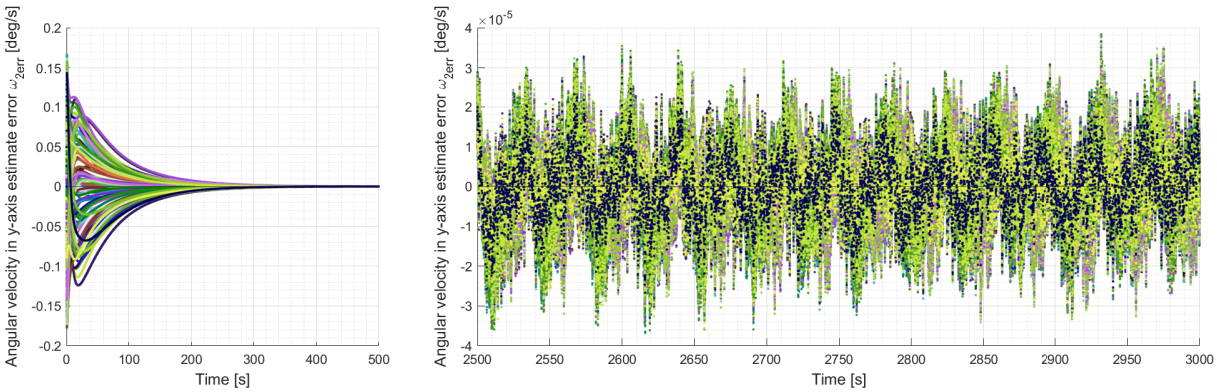


Figure B.13: Initial convergence and steady-state evolution of angular velocity y-axis estimate error $\omega_{2_{err}}$

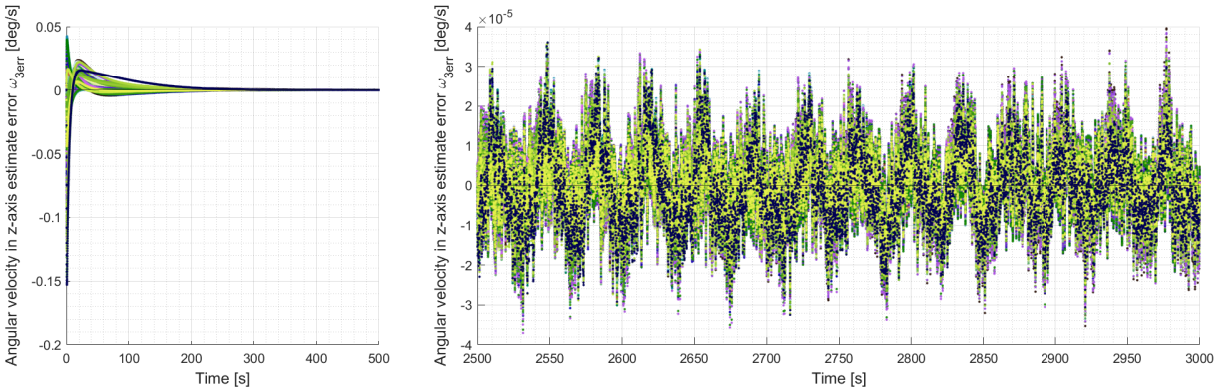


Figure B.14: Initial convergence and steady-state evolution of angular velocity z-axis estimate error $\omega_{3_{err}}$

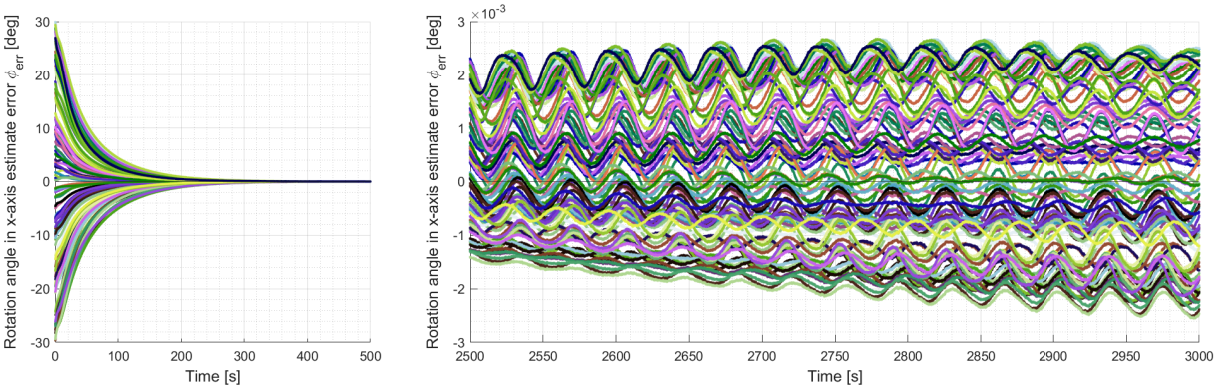


Figure B.15: Initial convergence and steady-state evolution of rotation angle in x-axis estimate error ϕ_{err}

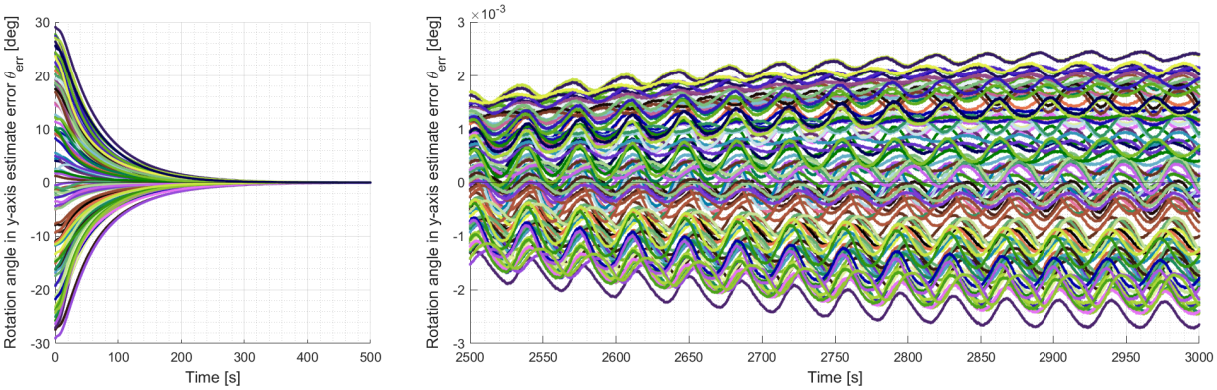


Figure B.16: Initial convergence and steady-state evolution of rotation angle in y-axis estimate error θ_{err}