

Autonomous Control for Rendezvous and Docking of CubeSats

Master's Thesis

Aitor Ramírez Gómez

supervised by:

Dr. Jesper Abilgaard Larsen



AALBORG UNIVERSITY
DENMARK

**Department of Electronic Systems****Control Engineering**

Fredrik Bajers Vej 7

DK-9220 Aalborg Ø, Denmark

Telephone 99 40 86 00

<http://www.es.aau.dk>**Title:**

Autonomous Control for
Rendezvous and Docking
of CubeSats

Theme:

M.Sc. Thesis

Project period:

February 2019 - June 2019

Group number:

1033

Written by:

Aitor Ramírez Gómez

Supervisors:

Jesper A. Larsen

Copies: 3**Pages: 111****Appendices: 3****Finished on: 6-6-12****Abstract:**

Spacecraft rendezvous maneuvers have been recognised to be an important and challenging problem to tackle. Servicing, assembling, space debris removal or refueling powerless satellites in-orbit are some of the missions where rendezvous takes an important role. This problem, then, have been treated already by experts where different optimal and robust control strategies including Model Predictive Control (MPC), Sliding-Mode Control (SMC) or \mathcal{H}_∞ Control, have been analysed. Notwithstanding, fewer literature on this topic can be found for CubeSat rendezvous missions, which optimization problem adds an extra challenge due to the limited amount of fuel they can carry, as well as the lower performance of its miniaturized sensors and actuators compared to those of a large spacecraft. An important step towards achieving this has been done recently by [30]. The objective of this thesis is to contribute on this topic investigating other control and navigation solutions to perform RVD of micro-satellites.

“I... a universe of atoms, an atom in the universe.”

— Richard P. Feynman

Preface

This thesis has been developed during the period February to June 2019, as part of the M.Sc. in Control and Automation at Aalborg University (AAU), Denmark.

I would like to take this chance to express my gratitude towards the individuals that have helped me somehow, either from a personal or an academical perspective. Firstly, I would like to thank Jesper A. Larsen for his assistance and patience during this period. To Mohamad Al Ahdab for all the late nights in the library fighting against quaternions and filters, your support has been essential. Also to all my friends who dragged me outside to remember me that there is still life beyond the work.

Per últim, i no menys important, vull agrair de tot cor als meus pares per brindar-me l'oportunitat de créixer lluny de casa, on he après que els esdeveniments més insignificants poden devenir tant en horros suplici com en plaent goig. La seva virtud, que tant admiro, és estar present incondicionalment en qualsevol de les dues situacions. Gràcies per haver-me educat de la manera en que ho heu fet.

Aitor Ramírez Gómez

Table of Contents

| | | |
|----------|--|-----------|
| 1 | Introduction | 1 |
| 1.1 | Motivation | 3 |
| 1.2 | GNC Subsystems | 4 |
| 1.2.1 | Sensors | 5 |
| 1.2.2 | Actuators | 7 |
| 1.3 | Contributions | 8 |
| 1.4 | Structure of the Thesis | 9 |
| 2 | Problem Description | 11 |
| 2.1 | Reference Frames | 12 |
| 2.1.1 | Earth-Centred Inertial (ECI), \mathcal{F}_I | 12 |
| 2.1.2 | Earth-Centred, Earth-Fixed (ECEF), \mathcal{F}_E | 12 |
| 2.1.3 | Spacecraft Local Orbital, \mathcal{F}_O | 13 |
| 2.1.4 | Spacecraft Body, \mathcal{F}_B | 14 |
| 2.1.5 | Spacecraft Docking Port, \mathcal{F}_D | 14 |
| 2.1.6 | Spacecraft Navigation, \mathcal{F}_N | 15 |
| 2.1.7 | Sloshing Frame, \mathcal{F}_S | 15 |
| 2.2 | Orbital Parameters | 16 |
| 2.2.1 | Orbit Description | 16 |
| 2.2.2 | Circular Orbit Dynamics | 18 |
| 2.3 | RVD/B Phases | 19 |
| 2.3.1 | Phasing | 19 |
| 2.3.2 | Far-Range Rendezvous | 20 |
| 2.3.3 | Close-Range Rendezvous | 21 |
| 2.3.4 | Mating | 22 |
| 2.4 | Mission Requirements | 22 |
| 3 | Spacecraft Motion | 25 |
| 3.1 | Absolute Rotational Motion | 26 |
| 3.2 | Relative Rotational Motion | 29 |
| 3.3 | Relative Translational Motion | 31 |
| 3.4 | External Disturbances | 34 |
| 3.4.1 | Residual Magnetic Dipole | 34 |
| 3.4.2 | Gravity Gradient | 36 |
| 3.4.3 | Aerodynamic Drag | 36 |
| 3.4.4 | Thrust Missalignment | 37 |
| 3.5 | Internal Disturbances | 37 |
| 3.5.1 | Slosh Dynamics | 38 |

| | | |
|----------|---|------------|
| 4 | Control System | 41 |
| 4.1 | Linear Model | 42 |
| 4.2 | Relative Attitude Control | 43 |
| 4.2.1 | Sliding-Mode Control | 44 |
| 4.2.2 | Sliding Manifold | 44 |
| 4.2.3 | Reachability Condition | 46 |
| 4.3 | RW Momentum Dumping Control | 51 |
| 4.4 | Relative Position Control | 54 |
| 4.4.1 | Model Predictive Control | 55 |
| 4.4.2 | MPC Formulation | 56 |
| 4.4.3 | Motion Profiles | 64 |
| 5 | Navigation System | 67 |
| 5.1 | Measurement Equations | 68 |
| 5.1.1 | Star Tracker | 68 |
| 5.1.2 | Camera Sensor | 69 |
| 5.2 | Filtering | 75 |
| 5.2.1 | Nonlinear Discrete-Time Stochastic Model | 78 |
| 5.2.2 | Uncented Kalman Filter | 80 |
| 5.3 | Navigation Convergence Detection | 90 |
| 6 | Closure | 95 |
| 6.1 | Discussion | 95 |
| 6.2 | Future Work | 97 |
| | Bibliography | 99 |
| A | Dynamic Models | 103 |
| A.1 | Equation of Motion of the CoMs in the Orbital Frame | 103 |
| B | Sloshing Model | 107 |
| C | Uncented Kalman Filter | 109 |
| C.1 | Cholesky Decomposition | 109 |
| C.2 | UKF algorithm | 110 |

List of Figures

| | | |
|-----|--|----|
| 1.1 | Image of an animation of two 6U CubeSat performing the last stage of an autonomous rendezvous, and eventually docking. [Source]: http://www.esa.int/Our_Activities/Space_Engineering_Technology/How_to_dock_CubeSats [Accessed]: 23-2-19 | 2 |
| 1.2 | Design of the structure of the standardized small satellites. [Source]: http://www.talksatellite.com/Americas-A10085522.htm [Accessed]: 23-2-2019 | 3 |
| 1.3 | Accuracy variation on typical RVD sensors as range increases. [Source]: [14] | 6 |
| 2.1 | Position description of the CoM of the propellant w.r.t. the body frame. . | 16 |
| 2.2 | Keplerian parameters. [Source]: https://commons.wikimedia.org/w/index.php?curid=8971052 | 17 |
| 2.3 | Position description of the CoM of the propellant w.r.t. the body frame. . | 19 |
| 2.4 | Trajectory from far-range rendezvous to closing. [Source]: [30] | 20 |
| 3.1 | Representation of a translation induced in the chaser's docking port \mathcal{F}_{D_c} , w.r.t. the target's docking port, \mathcal{F}_{D_t} , due to a change in attitude. | 31 |
| 3.2 | Sloshing motion of the propellant in the tank modeled as a mass-spring-dampen system. | 39 |
| 3.3 | Force and torque disturbances originated by the propellant inside the tank assuming a mass-spring-dampen model. | 40 |
| 4.1 | Control torques derived using SMC, where the <i>chattering</i> effect can be witnessed. | 47 |
| 4.2 | Sigmoid function with different tuning parameters, $\lambda = 2, 3, 100$ | 48 |
| 4.3 | Control torques derived using the SMC with sigmoid function $\sigma(\cdot)$ to attenuate the <i>chattering</i> | 49 |
| 4.4 | Relative quaternion \mathbf{q} | 50 |
| 4.5 | Relative angular velocity $\boldsymbol{\omega}$ | 50 |
| 4.6 | Monte-Carlo simulation of the angular misalignment between the docking ports under uncertainties and disturbances. In total, 40 simulations are shown with inertia matrices randomly varying between -70 to 70%. | 51 |

| | | |
|------|---|----|
| 4.7 | Simulation showing the effect of the RW momentum dumping control. On the top, the evolution of the relative quaternion is shown using the attitude control and the RW momentum dumping control. In the middle, the angular momentum of the RWs while applying attitude and momentum dumping control is superposed on top of the angular momentum of the RWs while only applying attitude control. On the bottom, the magnetic dipole generated by the MTQs after a predetermined period of time of 6 s. | 53 |
| 4.8 | Position trajectories under uncertainties in the mass of the chaser ranging from -70% to 70%. Some solutions become unfeasible. The control requirements are denoted by the thick red cone. | 62 |
| 4.9 | Velocity trajectories under uncertainties in the mass of the chaser ranging from -70% to 70%. Some solutions become unfeasible. | 62 |
| 4.10 | Docking missalignments. The red continuous circle represent the margin that the docking port can physically handle. The red dashed circle represent the control requirements to be fulfilled. | 63 |
| 4.11 | 3D trajectories of the chaser towards the docking port. | 63 |
| 4.12 | Position and velocity profiles from SK ₀ to SK ₁ , assuming an under saturated thrust command of 20 [mN]. | 64 |
| 4.13 | Position and velocity profiles from SK ₁ to SK ₂ , assuming an under saturated thrust command of 20 [mN]. | 65 |
| 5.1 | Inner LED pattern. | 70 |
| 5.2 | Outer LED pattern. | 70 |
| 5.3 | Complete LED pattern. | 71 |
| 5.4 | Ideal pinhole camera model. | 72 |
| 5.5 | VBN measurement sigma points, when tracking the position of the 5-LED pattern from 2.5 [m] distance. | 84 |
| 5.6 | Estimation of the relative attitude \mathbf{q} and relative position \mathbf{r} , using the UKF as an state-estimator fusing measurements from the camera and the star-tracker. The simulation is then performed from the first to the second SK point. | 86 |
| 5.7 | Estimation of the relative angular velocity $\boldsymbol{\omega}$ and relative linear velocity \mathbf{v} , using the UKF as an state-estimator fusing measurements from the camera and the star-tracker. The simulation is then performed from the first to the second SK point. | 87 |
| 5.8 | Position and velocity estimation errors throughout the Final Approach. The navigation requirements —1% of the range— are superposed on top of the position errors in red. The navigation solution employed is denoted and separated by a vertical dashed line, as well as the different SK points. | 88 |
| 5.9 | Position and velocity estimation errors throughout the Final Approach. The navigation requirements —1% of the range— are superposed on top of the position errors in red. The navigation solution employed is denoted and separated by a vertical dashed line, as well as the different SK points. | 89 |
| 5.10 | Rotation-translation coupling in the VBN. | 90 |

| | |
|---|----|
| 5.11 CUMSUM in a simulation defined as both, S_k and G_k . The time $t = 0.61$ [s] denotes when S_k crosses the x -axis. | 93 |
| 5.12 The true quaternion —only the vector part $\mathbf{\epsilon}$ is shown— and the true position are represented with the continuous line. Their respective estimations are split in two segments: Not Converged (N.C.) and Converged (C.). The time when convergence is assumed is marked with an *. | 93 |

List of Tables

| | | |
|-----|---|-----|
| 1.1 | Star tracker's accuracy and mechanical specifications. [<i>Sources</i>]: [7, 30]. . | 5 |
| 1.2 | Camera Sensor's accuracy and mechanical specifications. [<i>Source</i>]: [30]. | 6 |
| 1.3 | Reaction wheels' accuracy and mechanical specifications. [<i>Source</i>]: [3]. . | 7 |
| 1.4 | Magnetorquers' performance and mechanical specifications. [<i>Source</i>]: [2]. | 7 |
| 1.5 | Propulsion module performance and mechanical specifications. [<i>Source</i>]: [1]. | 8 |
| 2.1 | Orbital Parameters. | 18 |
| 2.2 | Control requirements for the last stages of a CubeSat RVD mission. . . . | 22 |
| 4.1 | MPC guideline. It describes different optimization problems and indicates which is required to be solved depending on the range. | 61 |
| 4.2 | Mean and variance of the ΔV manoeuvres performed during the simulations. | 64 |
| 5.1 | Characteristics of the process and measurement noise \mathbf{w}_k and \mathbf{v}_k assumed for a Kalman filter. | 76 |
| 5.2 | Kalman filter algorithm for linear discrete-time systems. | 77 |
| C.1 | Uncented Kalman filter algorithm for nonlinear discrete-time systems with a state based on quaternions. | 111 |

Introduction

1

Rendezvous has been recognized to be an important and challenging problem to tackle. In general terms, to *rendezvous* is the action of two or more counterparts to meet in an agreed time and place. The solution to this problem is required in diverse applications of different fields, the space sector among them. In a wide variety of space missions, rendezvous turns out to be a key element. For instance, in servicing orbital platforms or stations, in-orbit assemblies, retrieval —capture and return to ground—, refueling or repairing spacecrafts in orbit or re-joining an orbiting vehicle [14, 25, 29].

In this context, it is sought for a more precise definition of *orbital rendezvous*, which can be stated as the sequence of maneuvers that bring two orbiting objects to a certain neighborhood in the same time, attaining the same velocity vector [20]. Generally, a rendezvous approach can be performed between two spacecrafts —known as *cooperative* rendezvous—, and between a spacecraft and a spacial station or a spacecraft and an uncontrolled object (e.g. debris or powerless satellites) —known as *non-cooperative* rendezvous. In the literature, some terminology is also introduced in order to make reference to the active spacecraft performing the rendezvous, usually named *chaser*, and the passive spacecraft, usually referred to as the *target* object.

The typical end of a rendezvous mission is to eventually get the chaser and the target into contact. This procedure is called *mating*. Although, there is also the possibility to perform close proximity operations, which includes formation-flying, where the Guidance, Navigation and Control (GNC) system controls the state parameters of the spacecraft in order to fly in a precise formation relative to the other spacecrafts [10]. The mating process can either be *docking* — where the the motion of the chaser is

controlled in order to come directly into contact with the docking port of the target spacecraft, or *berthing* — where the chaser maintains zero relative motion near the target, and a manipulator located in one of the spacecrafts grapples the other one in order to insert it into the berthing port.

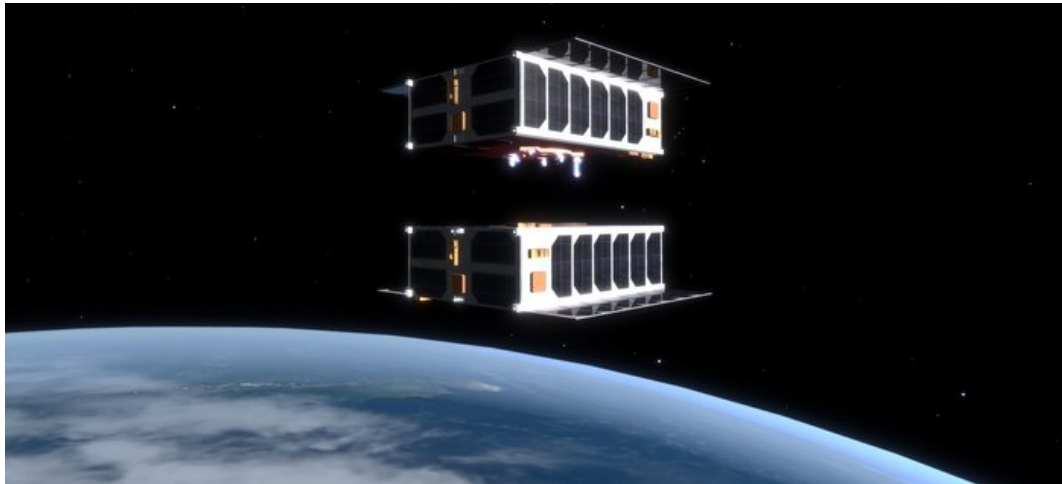


Figure 1.1: Image of an animation of two 6U CubeSat performing the last stage of an autonomous rendezvous, and eventually docking. [Source]: http://www.esa.int/Our_Activities/Space_Engineering_Technology/How_to_dock_CubeSats [Accessed]: 23-2-19

There are several stages in a Rendezvous and Docking/Berthing (RVD/B) mission. According to [14], once the spacecraft is injected on orbit, the mission can be partitioned in the following phases:

1. Phasing
2. Far-range rendezvous
3. Close-range rendezvous
4. Mating — docking or berthing

Each of these phases will be described shortly, where the tasks required in every process, as well as their complexity, will be emphasized.

Until today, several automated and manual RVD/B missions have been successfully performed. The very first RVD operation was performed manually in space by Neil Armstrong and Dave Scott, on board of the Gemini vehicle in 1966. This became the methodology for an extensive number of operations in the Apollo and MIR programmes. A year later, in 1967, the first autonomous rendezvous was accomplished between Cosmos 186 & 188, two Russian spacecrafts. After these events, NASA's Space Shuttle became the most well-known spacecraft docking with the International

Space Station (ISS). ESA's Automatic Transfer Vehicle (ATV), Russian's Soyuz and Progress, and others, followed the Space Shuttle, also rendezvousing and docking with the ISS. Currently, only Soyuz transports astronauts and cosmonauts to the ISS, and the station is resupplied by Progress — which docks with the ISS — and the Japanese HTV and SpaceX's Dragon cargo spacecrafts — which berth with the ISS. Rendezvous and docking maneuvers on board of the Soyuz spacecraft are performed manually by the astronauts and cosmonauts, while all the deliveries of cargo are performed autonomously.

1.1 Motivation

The RVD/B of satellites is a well-known problem that has been solved and analyzed by several experts, hence a wide diversity of optimal and robust control strategies, including Model Predictive Control (MPC), Sliding-Mode Control (SMC) and \mathcal{H}_∞ Control; already exist in the literature. Notwithstanding, fewer literature on this topic can be found for small satellites' rendezvous operations, which optimization problem adds extra challenges rising from the narrow constraints for docking and berthing. CubeSatellites, or simply CubeSats, are standardized pico-satellites based on cubic units, U, each having dimensions $10 \times 10 \times 10$ cm and weight less than 1.33 kg. Multiple units can be arranged together to expand the spacecraft capabilities, for example with 3U or 3U+ designs — where 3 units are stacked lengthwise, and an extra cylinder of 6.4 cm diameter extending 3.6 cm is added in the case of the 3U+ — or the largest structures 6U and 12U shown in Fig. 1.2. This reduction of scale is possible with the miniaturization of the essential components constituting a spacecraft, i.e. electronics, sensors and actuators. It is reasonable to assume that the performance and precision of the miniaturized components will be significantly reduced compared to those of large satellites, therefore, satisfying the required constraints for RVD/B operations can become a tough task.

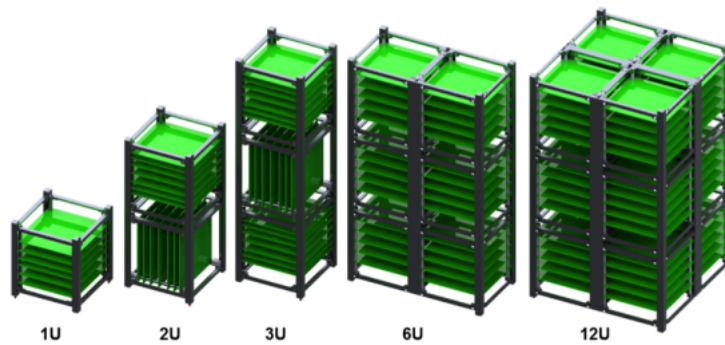


Figure 1.2: Design of the structure of the standardized small satellites. [Source]: <http://www.talksatellite.com/Americas-A10085522.htm> [Accessed]: 23-2-2019

The recent Ph.D thesis of Camille Sébastien Pirat [] provides the first step in the study of the feasibility of CubeSats rendezvous and docking in orbit. Pirat's dissertation is based on rendezvous and docking operations for a 6U CubeSat platform, i.e. (10×20×30 cm, ≤12 kg), and presents an extensive work with promising results yet to be implemented on real miniaturized satellites. Mathematical models for the relative motion of the two CubeSats, which includes a slosh¹ dynamics, have been derived in his work, as well as a comparison of different optimal —Linear Quadratic Regulator (LQR) and Linear Quadratic Integral (LQI)— and robust controllers — \mathcal{H}_∞ and μ -synthesis— performed by means of a μ -analysis and Monte-Carlo simulations. Moreover, a miniaturized vision-based navigation (VBN) and docking systems are proposed in his project. The VBN is a optical-based method to estimate both relative position and orientation of the spacecrafts with the help of LED patterns. In addition it has been tested under different illumination conditions to extract relevant information from the devices such as camera tuning parameters, accuracy or LED optimal wavelengths.

The author also emphasizes the importance of the study due to the possibility of assembly larger and heavier structures, such as telescopes or antennae, in-orbit by means of RVD of small satellites. This fact can be of great interest in the cases where the sizes are around the maximum limits of the payload fairing of the rockets. Moreover, docking two satellites also provides other beneficial possibilities, for instance, extending the life-time of certain missions or to capture and return non-working satellites.

Considering the advantages that RVD/B of CubeSats can offer, it seems worth it to analyze the problem once again, yet tailored to the required constraints and available levels of performance of the sensor and actuator systems for small satellites.

1.2 GNC Subsystems

CubeSats are build up on multiple systems, i.e. Power, Communication, Command & Data, GNC, etc. This thesis is mainly concerned with some of the subsystems belonging to the GNC system needed for the last manoeuvres of a RVD mission, where relative navigation is performed. They involve attitude and orbit actuators, as well as sensors for relative position and attitude tracking. Sensor equipment typically used for global navigation, like gyroscopes, magnetometers, sun sensors and GPS, are not specified in this section although they are needed for several stages of the RVD mission, prior to the stage accounted for in this work.

¹In this context, sloshing refer to the movement of propellant inside the spacecraft. In some cases it is important to take it into account in the derivations as it can introduce significant uncertainties of the position of the spacecrafts' center of mass.

Inter-Satellite Communication (ISC) is required during part of the rendezvous addressed in this project. The link between satellites in order to exchange information is based either on radio-frequency signals or on directed lasers [4], each of them presenting their own mechanical and control challenges that could be main topics of a thesis, and are beyond this work. It is then assumed that the micro-satellites involved in the RVD mission have on-board communication, and the exchange of information between chaser and target is performed ideally, i.e. constantly, instantly and uncorrupted.

The configuration of the CubeSat and the distribution of the subsystems are also not investigated, since the mass and volume budget are directly acquired from [30]. The mass of the chaser is then ≈ 10 [kg], plus ≈ 2 [kg] of propellant. The distribution of the subsystems affects the position of the CoM, which is introduced as an uncertainty based on the mechanical specification of a standard 6U. This is more detailed in the statement of the requirements Sec. 2.4.

In the subsequent sections, a brief technical description of commercial off-the-shelf (COTS) products considered for the last stage of the RVD is made in terms of physical properties and accuracy. The performance of the control and navigation systems are essentially dependent on the actuators and sensors specifications, thus it must be taken into account in the simulations.

1.2.1 Sensors

The spacecrafts are equipped with star trackers and a camera sensor with the most relevant features listed bellow. The choice of these kind of sensors are based on the accuracy they can provide.

- **Star Tracker** is a sensor device used for attitude determination based on the position of the stars in the firmament. The star tracker detects and recognize star patterns, which are later compared with star constellations in a database called *star catalog*. Star trackers outstrip all the other current sensors used for global attitude determination in terms of accuracy [7]. The detailing is listed in Tab. 1.1.

| Star Tracker | |
|---------------------------------------|---|
| Mass | 60 [g] |
| Volume | 62×56×68 [mm] — 0.24 [U] |
| Accuracy (σ) | 20 [arcsec] $\approx 5 \cdot 10^{-3}$ [deg] |

Table 1.1: Star tracker's accuracy and mechanical specifications. [Sources]: [7, 30].

- **Camera Sensor** is employed during the last stage of the RVD procedure, where relative models are used to control the spacecrafts. It will be seen in Sec. 2.3, that the part of the rendezvous tackled in this thesis starts from 10 [m] away from the target up to docking. This range distance is segmented into 3 smaller ranges where different sensors are used or combined. According to Fig. 1.3, camera type sensors are the most appealing, in terms of accuracy, from 10 [m] range.

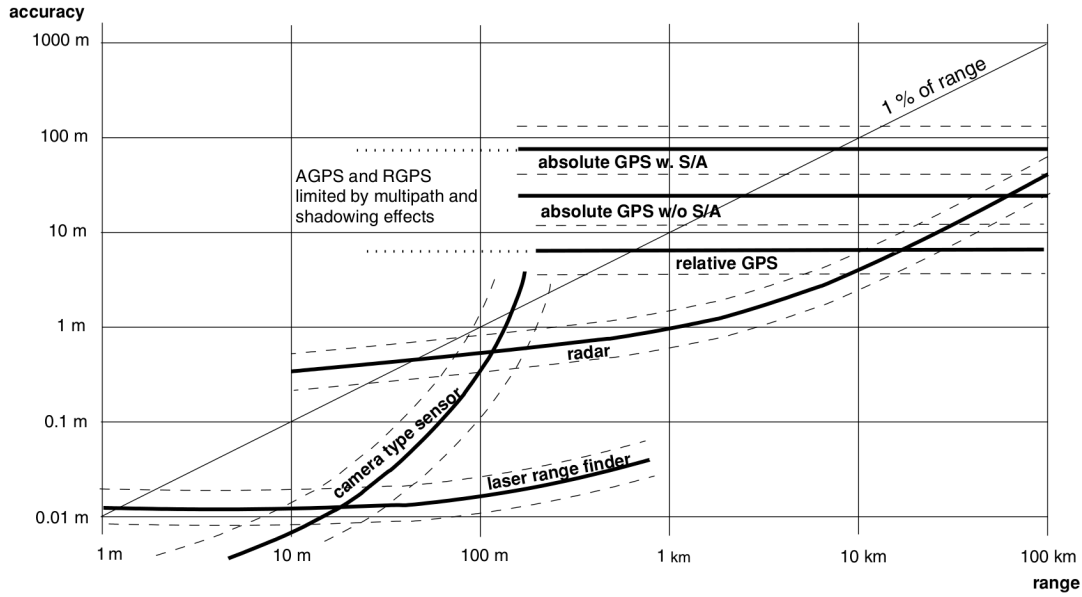


Figure 1.3: Accuracy variation on typical RVD sensors as range increases. [Source]: [14]

Therefore, the camera sensor is used to track the relative position during the first segment comprised from 10 [m] to 5 [m] range, and used to track both the relative position and attitude from 5 [m] until docking. This is further explained in Sec. 5. For this sensor, two sources of variation can be considered. Those are the misalignment of the camera axis and the noise density of the pixels², which are listed in Tab. 1.2, along with mass and volume specifications.

| Camera Sensor | |
|---------------------------------------|---|
| Mass | 900 [g] |
| Volume | 500 [cm ³] — 0.5 [U] |
| Accuracy (σ) | Misalignment of 0.1 [deg/s] $\approx 1.5 \cdot 10^{-4}$ [pixels] Pixel noise density 0.06 [pixels] |

Table 1.2: Camera Sensor's accuracy and mechanical specifications. [Source]: [30].

²The pixel noise density showed in Tab. 1.2 is an approximation of the true standard deviation, which is supposed to vary with the range. This is further explained in Sec. 5.2.1.

1.2.2 Actuators

The actuators considered to perform the RVD are reaction wheels (RW) to control the relative orientation, magnetorquers (MTQ) to control the RWs' angular momentum and Reaction Propulsion Control (RPC) to control the relative position. Their characteristics are based on GomSpace A/S current COTS, and the most relevant features are detailed below.

- **Reaction Wheels** are used to act on the relative orientation of the spacecraft. There exist several RW suppliers which performances can vary significantly from one to another. GomSpace provides a high-performance RW module for 6U and 12U satellites, the NanoTorque GSW-600, with an extensive datasheet describing the product. This module is also available in a tetrahedral —or pyramidal— configuration which increase the robustness and prevent the RWs from saturation. Although, a simpler configuration of 3 RWs is considered for the scope of this work. In the datasheet, specifications for one single flying wheel and for pyramidal configuration are provided and used to extract and summarize in the following table the important features of a 3 reaction wheel configuration.

| Reaction Wheels | |
|------------------------------|--------------------------------|
| Mass | $(\frac{940}{4}) \times 3$ [g] |
| Inertia (1 RW) | 300 [gm ²] |
| Volume | 0.4 [U] |
| Torque (Max./Min.) | 2/-2 [mNm] |
| Accuracy (σ) | 5 [rpm] |

Table 1.3: Reaction wheels' accuracy and mechanical specifications. [Source]: [3].

- **Magnetorquers** are a set of coils fixed to the spacecraft which can generate a magnetic dipole that interact with the present magnetic field. This interaction results in a torque around the CoM that allow the spacecraft orientation to be partially controlled. Performance and mechanical characteristics of the MTQs used are also provided by [2], and are summarized in Tab. 1.4.

| Magnetorquers | |
|------------------------|---|
| Mass | 156 [g] |
| Volume | 90.5×96.9×17.2 [mm] — 0.15 [U] |
| Magnetic Dipole | >300 [mAm ²] ≈ 0.4 [Am ²] |

Table 1.4: Magnetorquers' performance and mechanical specifications. [Source]: [2].

- **Reaction Control Propulsion** is the module in charge of providing linear momentum to the chaser, and the principal responsible of driving the chaser's docking port towards the target's docking port. It is required that the RCP provides, at least, 3 Degrees of Freedom (DoF) in order to perform the task. Although the COTS in [1] only accounts for 1 DoF, the thrust performance characteristics for a 3 or higher DoF version will remain similar. The RCP consist of 4 individual thrusters with closed-loop control on each of the axis, and butane as propellant. The benefits of using butane is that it can be liquefied and stored at lower pressures and lower volumes. Thus, no complex pressure regulation is needed for the tanks. Two important drawbacks, though, are that the specific impulse is slightly less than nitrogen, which is the propellant typically used; and that butane must be evaporated right before expelled. If some liquid phase is expelled instead, the performance of the butane thrusters gets significantly reduced [13]. The mass and volume are based on [30].

| Reaction Control Propulsion | |
|------------------------------|--------------------------------------|
| Mass (dry/wet) | 1/3 [kg] |
| Volume | 2 modules of 200×100×50 [mm] — 2 [U] |
| Thrust | 10×4 [mN] |
| Accuracy (σ) | 10 [μ N] |

Table 1.5: Propulsion module performance and mechanical specifications. [*Source*]: [1].

1.3 Contributions

The main contribution of this project is to continue on analyzing the feasibility of navigation and control strategies for the rendezvous and docking problem involving two CubeSats. The achievements accomplished in this thesis in order to fulfill the main objective are briefly summarized in this section.

Two dynamic models are derived in this thesis in order to use it as support in the navigation and control sections, a translational and a rotational model. The rotational model is upgraded to quaternions, in contrast to [30] where Euler angles are employed. A slightly less complex model is developed for the translational description, which model the relative position of the CoM of the spacecrafts, rather than the relative position of the docking ports.

A sliding-mode control based on quaternions is derived for relative attitude control, given the profitable nonlinear and robust properties that characterizes it. To act on the

spacecraft's orientation, a 3-reaction wheel formation have been chosen, accompanied with magnetorquers to prevent the angular momentum of the wheels to build up. For that, a momentum dumping control is additionally introduced in charge of computing the commands needed for the magnetorquer. A linear model predictive control is developed to optimize the fuel consumption while fulfilling predetermined constraints rising from the control requirements stated shortly. The controllers regarding the actuators are presupposed, namely they are assumed to provide the force and torques derived here. It is assumed that the available performances of the actuators are the ones stated in Sec. 1.2.

Moreover, only the special case of circular orbits is addressed since it significantly ease the calculations of the equations of motion, and it is considered a fair assumption as almost all the RVD/B missions are executed in Low-Earth Orbits (LEO) with small or zero eccentricity, namely, near-circular or circular orbits.

1.4 Structure of the Thesis

- **Chapter 2.** An overview of the reference frames required for the development of future chapters, as well as a brief description of the orbital parameters, which are needed to set up the simulations and for some reference transformation. This also ease the understanding of the RVD phases, described also in this chapter.
- **Chapter 3.** The modeling chapter. Relative and absolute attitude models are derived based on quaternions. A translational model is also derived, describing the CoM of the spacecrafts. Finally, the environmental disturbances are also derived.
- **Chapter 4.** The control chapter. An sliding-mode and a model predictive control are derived for attitude and position control. In addition, a reaction wheel momentum dumping control is introduced. Some results are shown and discussed.
- **Chapter 5.** The navigation chapter. The modeling of the measurements provided by the sensor equipment is derived. A nonlinear discrete stochastic model is introduced, which is used in the developement of an uncenced Kalman filter also based on quaternions. A basic navigation convergence detection algorithm is also derived based on the CUSUM algorithm.
- **Chapter 6.** The closure. Discussion of the overall thesis and future improvements.

Problem Description

2

A more accurate delineation of the problem is developed in this Chapter. The essential reference frames defined to solve the problem of CubeSats RVD are introduced firstly. It is followed by a brief explanation about how Keplerian orbits are characterized in order to gain some familiarization for the succeeding sections. The orbital dynamics for the special case of circular orbits is introduced right after, which is necessary to describe the motion of the local orbital reference. The orbital dynamics, combined with the spacecraft dynamics derived in Chap. 3, will parameterize the complete motion of the spacecrafts. The procedure of CubeSat RVD/B is later introduced in more detail, which includes the different phases involved and some of its complexities. A special interest is taken in the last phases, i.e. Final Approach and Docking, since they are the procedures addressed in this thesis. This, finally leads to the statement of the requirements for the GNC system of a CubeSat in order to perform the Final Approach of a RVD mission.

2.1 Reference Frames

In order to describe the problem mathematically, 6 different reference frames are defined. Each reference frame, denoted by \mathcal{F} and a subindex, will be given by an origin, O , and a set of three orthogonal vectors. More notation will be introduced shortly.

2.1.1 Earth-Centred Inertial (ECI), \mathcal{F}_I

The Earth-Centred Inertial (ECI) frame, \mathcal{F}_I , is used to describe the dynamics w.r.t. (with respect to) inertially fixed directions. The origin of the inertial reference frame, O_I , is placed in the Center of Mass (CoM) of the Earth. It is assumed that the Earth is completely spherical and the geometrical center coincides with the CoM and one of the foci of the orbital motions.

The z -axis is perpendicular to the equatorial plane and aligned with the direction of rotation of the Earth, the vector $\hat{\mathbf{z}}_I$ points towards the North Pole. The x -axis is in the equatorial plane, and $\hat{\mathbf{x}}_I$ points to the mean vernal equinox. The y -axis can be found straightforward as a vector product $\hat{\mathbf{y}}_I = \hat{\mathbf{z}}_I \times \hat{\mathbf{x}}_I$, completing the right-hand rule. Notice that the equatorial plane is spanned by $[\hat{\mathbf{x}}_I \ \hat{\mathbf{y}}_I]$, and the vernal equinox can be found as the intersection of the equatorial plane and the orbital plane of the Earth around the Sun. The ECI frame is, therefore, defined as $\mathcal{F}_I \triangleq \{O_I, \hat{\mathbf{x}}_I, \hat{\mathbf{y}}_I, \hat{\mathbf{z}}_I\}$.

2.1.2 Earth-Centred, Earth-Fixed (ECEF), \mathcal{F}_E

The Earth-Centred, Earth-Fixed (ECEF) frame is a rotating reference frame defined as $\mathcal{F}_E \triangleq \{O_E, \hat{\mathbf{x}}_E, \hat{\mathbf{y}}_E, \hat{\mathbf{z}}_E\}$. Its mainly purpose is to describe target points and models—such as the atmospheric model or the gravity field—fixed to the Earth's orientation.

The ECEF shares the origin and the z -axis with the ECI frame, i.e. $O_E = O_I$ and $\hat{\mathbf{z}}_E = \hat{\mathbf{z}}_I$. Vector $\hat{\mathbf{x}}_E$ points to the Earth's prime meridian, perpendicular to $\hat{\mathbf{z}}_E$. Vector $\hat{\mathbf{y}}_E$ completes the right hand coordinate system. Again, the equatorial plane is spanned by $[\hat{\mathbf{x}}_E \ \hat{\mathbf{y}}_E]$.

2.1.3 Spacecraft Local Orbital, \mathcal{F}_O

The Local Orbital frame, or simply orbital frame, is denoted by \mathcal{F}_O . Often, this reference frame is also referred to in the literature as Local-Vertical/Local-Horizontal (LVLH) frame, and appear to be useful to split the problem of parameterizing the spacecraft motion into two parts. On the one hand, the motion of the spacecraft while in nominal mode, i.e. *Nadir* or *Zenith*¹, is completely determined by means of the orbital dynamics [19]. On the other hand, determining the relative attitude motion between the spacecraft and \mathcal{F}_O is done, as in Sec. 3, studying the spacecraft dynamics. Other interests of defining this reference frame lies in representing the trajectory of the chaser in the local orbital frame of the target during rendezvous operations, since doing so is more suitable to analyze the relative motion between the spacecrafts.

The orbital frame for circular orbits is, then, $\mathcal{F}_O \triangleq \{O_O, \hat{\mathbf{x}}_O, \hat{\mathbf{y}}_O, \hat{\mathbf{z}}_O\}$. Its origin O_O is placed in the CoM of the spacecraft. Vector $\hat{\mathbf{z}}_O$ is pointing towards the center of the Earth, and can be mathematically determined as $\hat{\mathbf{z}}_O = -\frac{\mathbf{r}_I}{\|\mathbf{r}_I\|_2}$, being $\mathbf{r}_I \in \mathbb{R}^3$ the position of the CoM of the spacecraft expressed in the ECI frame and $\|\cdot\|_2$ the 2-norm function². Because of using \mathbf{r}_I to define $\hat{\mathbf{z}}_O$, the z -axis is also called \bar{R} in the literature. Vector $\hat{\mathbf{x}}_O$ points in the direction of the spacecraft's motion, and is mathematically defined as $\hat{\mathbf{x}}_O = \frac{\mathbf{v}_I}{\|\mathbf{v}_I\|_2}$, where $\mathbf{v}_I \in \mathbb{R}^3$ is the inertial linear velocity of the spacecraft. Again, the x -axis defined by $\hat{\mathbf{x}}_O$ can be found in the literature as \bar{V} for the same reason. Finally, vector $\hat{\mathbf{y}}_O$ is normal to the orbital plane, which is spanned by $[\hat{\mathbf{x}}_O \ \hat{\mathbf{z}}_O]$, and is calculated as $\hat{\mathbf{y}}_O = \hat{\mathbf{z}}_O \times \hat{\mathbf{x}}_O$. The direction of $\hat{\mathbf{y}}_O$ coincides with the direction of the orbital angular momentum, which is often denoted by the letter \mathbf{h} , and therefore, the y -axis can be found as \bar{H} in the literature.

During the final approach in a RVD mission, both satellites are close enough to consider that the orientation of the local orbital frame of the chaser, denoted by \mathcal{F}_{O_c} , and the local orbital frame of the target, denoted by \mathcal{F}_{O_t} , are practically the same. This can be a fair approximation given that relative velocities during RVD are significantly low compared to the inertial velocities. Thus, in the attitude modeling sections — Sec. 3.1 and Sec. 3.2— the subindices c and t for the local orbital frame will be dropped, and only a general orbital reference frame, \mathcal{F}_O , will be used.

¹*Nadir* and *Zenith* are pointing modes activated when the spacecraft is idle, meaning there are no objectives to fulfill and/or waiting for commands. In *Nadir*, the spacecraft points towards the centre of the Earth, while in *Zenith* it points to the opposite direction to *Nadir*.

²In general, the p -norm of a vector $\mathbf{x} = [x_1 \ x_2 \ \dots \ x_N]^T$ is defined as $(\sum_{i=1}^N |x_i|^p)^{1/p}$.

2.1.4 Spacecraft Body, $\mathcal{F}_{\mathcal{B}}$

The Body frame is denoted by $\mathcal{F}_{\mathcal{B}} \triangleq \{\mathcal{O}_{\mathcal{B}}, \hat{\mathbf{x}}_{\mathcal{B}}, \hat{\mathbf{y}}_{\mathcal{B}}, \hat{\mathbf{z}}_{\mathcal{B}}\}$. The subindex \mathcal{B} is replaced by \mathcal{C} when describing the chaser's body frame, and by \mathcal{T} when describing the target's body frame. The origin, $\mathcal{O}_{\mathcal{B}}$, is located in the CoM of the satellite, and the coordinate systems is usually aligned with the structure edges, which in this case coincides with directions of the principal moments of inertia. These reference frames are used to describe the attitude of the chaser and the target w.r.t. the local orbital frame. When the spacecraft is in nominal mode, $\mathcal{F}_{\mathcal{B}}$ and $\mathcal{F}_{\mathcal{O}}$ are aligned.

For the sake of simplicity, these frame is also used to describe the torques and forces applied to the spacecraft. It is thus considered that $\mathcal{F}_{\mathcal{B}}$ is aligned with the directions of the principal moments of inertia.

Technically, an intermediate geometrical reference frame fixed in the structure of the satellites should be previously defined. The reason lies in the fact that the position of the CoM of the spacecraft varies with time due to, for instance, sloshing dynamics or depletion of propellant. This generally forbid other upcoming reference frames—Docking Port and Navigation frames—to be defined relative to the *moving* body frame $\mathcal{F}_{\mathcal{B}}$, and require to be defined relative to a fixed geometrical reference frame. Instead, it will be assumed that the CoM of the spacecrafts are fixed in the structure of the satellites, and the uncertainty regarding the real position of the CoMs, along with its effects on the system—for example, of its misalignment with the thrusters—are considered as disturbances and dealt with it in the control design in Sec. 4.2.

2.1.5 Spacecraft Docking Port, $\mathcal{F}_{\mathcal{D}}$

The Docking Port frames, $\mathcal{F}_{\mathcal{D}_c}$ and $\mathcal{F}_{\mathcal{D}_t}$, are defined relative to their body frames $\mathcal{F}_{\mathcal{C}}$ and $\mathcal{F}_{\mathcal{T}}$. The origins, $\mathcal{O}_{\mathcal{D}_c}$ and $\mathcal{O}_{\mathcal{D}_t}$, are located in the lances of the docking ports of each spacecraft.

In general, the docking port—and therefore, the docking port frame—of a spacecraft can present any orientation relative to its body frame. Nevertheless, this generalization is not essential to prove the objectives of this thesis and, in order to simplify the problem further, the assumption that the body and the docking port frames of each spacecraft share the same orientation is made. This assumption allows, in Sec. 3.2, to determine the relative attitude motion of the docking port frame straightforward from the spacecrafts' body reference frame as both have the same orientation. It also ease the linearization process of the system's equations. Moreover, the misalignment of the chaser's and target's docking ports while performing the RVD can, thus, be calculated directly as the difference in angular position between $\mathcal{F}_{\mathcal{C}}$ and $\mathcal{F}_{\mathcal{T}}$.

The direction of the x -axis of $\mathcal{F}_{\mathcal{D}}$, define the *final approach axis*, where the chaser must be placed to align both docking ports before doing initiating the final maneuvers prior to docking.

2.1.6 Spacecraft Navigation, $\mathcal{F}_{\mathcal{N}}$

The navigation system is based on star trackers and vision-based sensors —this is introduced in Chap. 5. The navigation frame specifically describe the orientation and relative position of the vision-based navigation relative to the body frame, which includes a camera type sensor placed in the chaser and LED patterns placed in the target. The vision-based navigation and docking systems are mounted together in the chaser, as well as the LEDs patterns are mounted together with the docking system in the target. The LEDs and the camera are, although, displaced 7 [cm] inwards so that when dock is completed the camera can still detect the LEDs.

The navigation frames are, therefore, oriented like their respective docking port frames. The origin, $\mathcal{O}_{\mathcal{N}_c}$, denotes the center of the chaser's navigation frame and is placed 7 [cm] behind the origin of the chaser's docking frame, $\mathcal{O}_{\mathcal{D}_c}$. Its x -axis is aligned with the camera axis. The origin, $\mathcal{O}_{\mathcal{N}_t}$, on the other hand, denotes the center of the target's navigation frame and is placed 7 [cm] behind the origin of the target's docking frame, $\mathcal{O}_{\mathcal{D}_t}$. Its x -axis is aligned with the center of each pattern. The reference frame for both spacecrafts are depicted in Fig. 5.3 and Fig. 5.4.

2.1.7 Sloshing Frame, $\mathcal{F}_{\mathcal{S}}$

The sloshing frame is used to describe the motion of the CoM of the propellant inside the RCP module tank. The tank is fixed in the chaser's structure, and are easily located w.r.t the body frame. Given that a spherical tank can be considered —meaning that the orientation of this frame is irrelevant— the orientation of $\mathcal{F}_{\mathcal{S}}$ is thus assumed to be aligned with the body frame. This simplify the derivations in Sec. 3.5.1. The origin, $\mathcal{O}_{\mathcal{S}}$, is placed in the geometrical center of the tank. The position of the CoM of the propellant is then, described w.r.t. this frame, $\mathcal{F}_{\mathcal{S}}$ as shown in the Fig. 2.1.

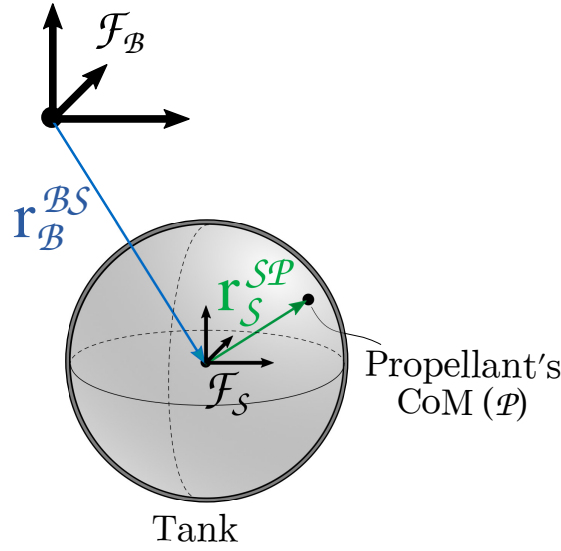


Figure 2.1: Position description of the CoM of the propellant w.r.t. the body frame.

2.2 Orbital Parameters

A brief introduction to the parameters that fully characterize a Keplerian orbit is believed to be of help at this point in order to become familiar with them before reaching the next sections. In addition, the orbital dynamics used to describe the motion of \mathcal{F}_O is performed for the special case of circular orbits.

2.2.1 Orbit Description

According to Kepler's Laws, an arbitrary orbit has an elliptic shape, which can be described using only 2 parameters. In a 3 dimensional space, the orientation of the orbit can be parameterized with 3 more parameters. Therefore, Keplerian orbits can be described by means of 5 parameters or Keplerian orbital elements. The following parameters

- a — the semi-major axis and
- e — the eccentricity,

describe the shape of the ellipse. The semi-major axis, a , is defined as the distance between the center and the perimeter of the ellipse passing through one of the foci. Being c the distance between the center and one of the foci of the ellipse, the

eccentricity, e , is then the ratio between c and a as shown in (2.1).

$$e = \frac{c}{a} = \frac{\sqrt{a^2 - b^2}}{a} = \sqrt{1 - \frac{b^2}{a^2}} \quad (2.1)$$

In order for it to be a closed orbit with negative total energy, the eccentricity must be $0 \leq e < 1$. As a special case, and the main focus of this thesis, it is referred to circular orbit whenever $e = 0$.

The last 3 parameters,

- i — the inclination,
- Ω — the Right Ascension of the Ascending Node (RAAN) and
- ω — the argument of periapsis

describe the orientation of the ellipse. The inclination i is defined as the angle between the equatorial and the orbital planes. In the intersection between the equatorial plane and the orbit itself, the Ascending and Descending nodes can be found. It is referred to as Ascending node if at that point the spacecraft is entering the northern hemisphere, and Descending node otherwise. Joining these two nodes a line is generated called *line of nodes* for obvious reasons, and the angle defined between this line and the vernal equinox is the RAAN, Ω .

An extra parameter is used to determine the position of an orbiting body along the ellipse, that is

- ν — the position of the orbiting body along the ellipse or *true anomaly*

The following figure graphically describes how they are defined. These parameters are needed to translate the disturbances from \mathcal{F}_E and \mathcal{F}_I to \mathcal{F}_O .

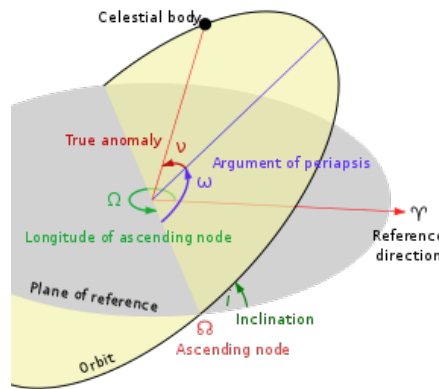


Figure 2.2: Keplerian parameters. [Source]: <https://commons.wikimedia.org/w/index.php?curid=8971052>

[Accessed]:12-3-2019.

The parameters choice of orbital parameters for the simulations are similar to the ones used in [30]. They are listed in the following table.

| a | e | i | Ω | ω | ν |
|-----------|---|------------|-------------|----------|-----------|
| 6741 [km] | 0 | 97.8 [deg] | 190.1 [deg] | 0 | 100 [deg] |

Table 2.1: Orbital Parameters.

2.2.2 Circular Orbit Dynamics

The motion of a particle orbiting the Earth can be described by means of the orbital dynamics. For the special case of circular orbits, the particle is performing a uniform circular motion under the action of a radial gravitational force $\mathbf{f}_I^g \in \mathbb{R}^3$ described in the inertial frame as

$$\mathbf{f}_I^g = -\mu \frac{m}{\|\mathbf{r}_I\|_2^2} \hat{\mathbf{r}}_I, \quad (2.2)$$

where $\mu \in \mathbb{R}$ is the Earth's gravitational constant, $m \in \mathbb{R}$ is the mass of the orbiting particle and $\mathbf{r}_I \in \mathbb{R}^3$ is the distance from the center of the Earth to the particle expressed in \mathcal{F}_I . For circular orbits, \mathbf{r}_I is constant, thus \mathbf{f}_I^g is also constant. According to Newton's second law,

$$\mathbf{f}_I^g = m\mathbf{a}_I, \quad (2.3)$$

being $\mathbf{a}_I \in \mathbb{R}^3$ the —centripetal— acceleration of the particle caused by the force \mathbf{f}_I^g , also expressed in the inertial frame. The magnitude of the centripetal acceleration is

$$a_I = \frac{\|\mathbf{v}_I\|_2^2}{\|\mathbf{r}_I\|_2}, \quad (2.4)$$

where $\mathbf{v}_I \in \mathbb{R}^3$ is the inertial velocity of the particle. The velocity changes constantly in direction, but not in magnitude, as vectors \mathbf{a}_I and \mathbf{v}_I are orthogonal in this particular case.

The orbital angular velocity represented in \mathcal{F}_I is $\omega_I^{IO} \in \mathbb{R}^3$, which is related to the linear velocity as

$$\omega_I^{IO} = \mathbf{r}_I \times \mathbf{v}_I. \quad (2.5)$$

Vectors ω_I^{IO} , \mathbf{r}_I , \mathbf{v}_I are orthogonal to each other, and the angular velocity is always perpendicular to the orbital plane. If these vectors are, then, expressed in \mathcal{F}_O , they appear to be $\omega_O^{IO} = [0 \ -\omega_o \ 0]^T$, $\mathbf{r}_O = [0 \ 0 \ -r_o]^T$ and $\mathbf{v}_O = [v_o \ 0 \ 0]^T$. The scope of this project is bounded only to small satellites, which are in general orbiting in

LEO. Besides, most of the RVD operations, including RVD/B with the ISS, are also performed in LEO. Hence, $r_o = r_{LEO} \approx 6871[km]$, where atmospheric and magnetic perturbations are notable. The angular velocity ω_o can be obtained merging equations (2.2), (2.3), (2.4) and (2.5), which yield to

$$\omega_o = -\sqrt{\frac{\mu}{r_{LEO}^3}}. \quad (2.6)$$

Finally, v_o can be found simply using relation (2.5).

2.3 RVD/B Phases

A typical rendezvous procedure is divided, as stated in the introduction, into four phases after orbit injection. A well detailed explanation of the different phases can be found in [14], however, this section presents a brief synthesis as it is hoped to be enough to understand the overall rendezvous process and its difficulties for the development of the future sections. The GNC system designed in this thesis will be, although, only focused on the Final Approach and Docking phases, i.e. the very last stages of a RVD operation.

2.3.1 Phasing

At the end of the launch phase, the spacecraft is placed orbiting a stable orbit below the target, i.e. with a smaller semi-major axis, and with an initial phase angle Φ . Considering that objects closer to Earth orbit faster, the phase angle between target and chaser will shrink without need of actuation. The main objective of this stage is to reduce the phase angle $\Delta\Phi$ while approaching to the neighbourhood of the target. During this process, the orbital parameters will be adjusted, and any error dragged from the launch injection will be corrected.

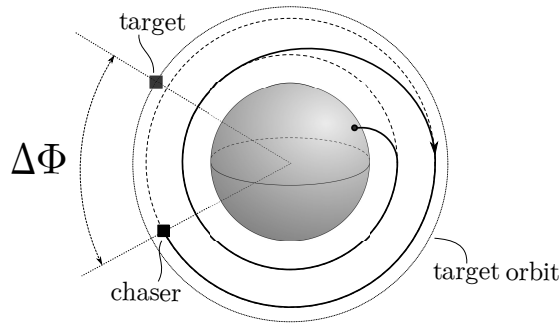


Figure 2.3: Position description of the CoM of the propellant w.r.t. the body frame.

Besides, the RAAN of the target and the chaser drift due to the non-spherical shape of the Earth. Since the difference of rates in the drifting of the RAANs depend on the altitude, the chaser spacecraft must be injected into a *virtual* orbit such that at the end of Phasing the target and the chaser share the same RAAN. The chaser is located behind the target, at a lower altitude when Phasing finishes. Close enough to establish ISC.

2.3.2 Far-Range Rendezvous

The objective of the far-range rendezvous, or Homing, is to place the chaser right behind the target, at the same altitude. This is usually performed with one manoeuvre, the so-called *Hohmann* transfer orbit. For the special case of a micro-satellite, Hohmann transfers can not be conducted due to the low thrust available [30]. Although, other approaches requiring the same amount of ΔV can be done instead. In the following figure it can be seen this stage in context, as well as part of the next stage, the close-range rendezvous.

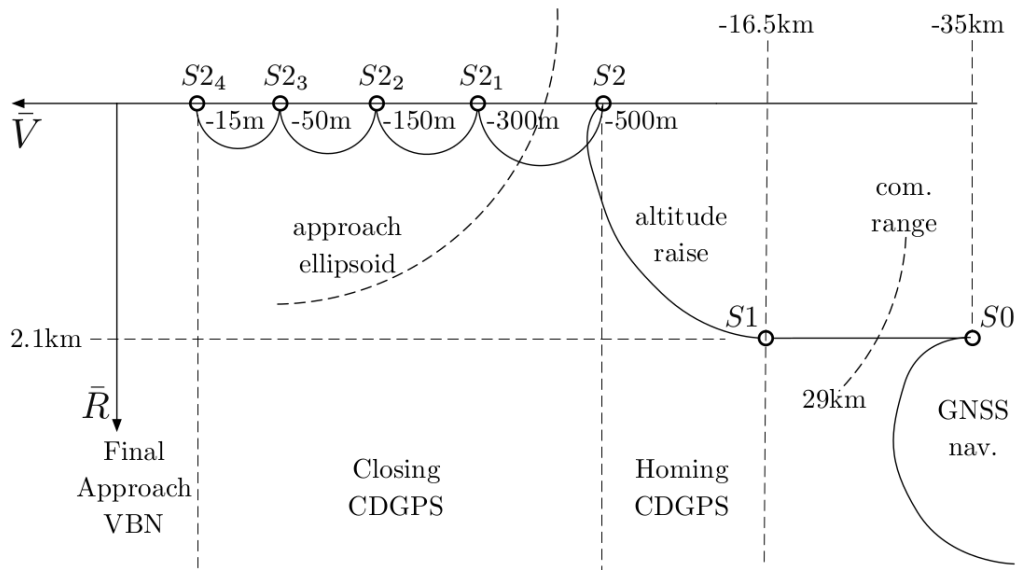


Figure 2.4: Trajectory from far-range rendezvous to closing. [Source]: [30]

2.3.3 Close-Range Rendezvous

Close-range rendezvous has the main objective of reducing the range to the target in a safety manner, which does not lead to danger of collision in case, for instance, of actuator failure. At the end of this phase, the chaser has achieved appropriate conditions to initiate the mating phase, i.e. docking or berthing. It can generally be divided into two stages usually referred to as *Closing* and *Final Approach*.

The Closing subphase is basically concerned with the reduction of range and the placement of the chaser into the final approach axis³ —or *corridor*— under certain safety constraints on position, velocity, attitude and angular velocities. As it is a possibility that the final approach axis is not aligned with \bar{V} , a *fly-around* maneuver might be required to place the chaser on the final approach axis before starting the Final Approach subphase. Due to an increase in the requirements for the navigation system's accuracy while reducing the range to the target, the Closing procedure must conclude with suitable conditions for a sensor change capable to meet the demanding accuracy requirements of the Final Approach.

When the chaser is placed in the final approach axis, at 10 meters from the target, the Final Approach takes place. This subphase addresses the last operations before docking or berthing. The objective is to slide through the final approach axis to finally obtain mating conditions. As previously mentioned, the final approach axis does not necessary have to be aligned with \bar{V} , which means that the trajectory imposed to the chaser until mating is not natural, and closed-loop strategies are required.

This approach is segmented into three stages FIGURE, which are separated by station-keeping (SK) points. The first approach, from $SK_0 = 10$ [m] to $SK_1 = 5$ [m] range, is performed using star trackers, camera-based navigation and ISC. Once at SK_1 point there is a handover of the sensors, relying only on camera-based navigation to perform the rest of the approaches. When the navigation has converged, the second approach takes place, from 5 [m] to 2.5 [m]. At SK_2 , a last check of the systems is done before the last approach is conducted until docking is achieved.

The analysis of the control and navigation strategies presented in this thesis start from the Final Approach subphase, assuming the uncertainties accumulated throughout the anterior phases, up to Docking.

³See Sec. 2.1.5

2.3.4 Mating

For a RVD mission, the mating phase starts when the chaser's docking port is in the range of that of the target. Relative velocities, and position and angular misalignments must lie within some predefined constraints in order to successfully achieve docking. Several tasks conform the mating phase, although, only capture will be tackled among them. One of the main issues during capture is the rebound of the bodies while the capture interface enters the docking port. This only allows the docking port to accomplish capture during a finite, relative small, amount of time before the chaser leave the docking port again. Docking is presupposed, although, once the docking ports achieve contact as this effect is not studied here.

2.4 Mission Requirements

The requirements to perform a CubeSat RVD are extracted from [30] where, as previously mentioned, an exhaust study on the feasibility of CubeSats RVD operations is made, as well as an intent on filling the technological gap, such as docking and navigation ports, adapted for small satellites. The scope of this thesis is to tackle other control and navigation strategies to perform the Final Approach and Docking with the available COTS presented in Sec. 1.2. Thus, only the requirements for the attitude and position control and navigation, during this last stage, are accounted for.

As stated by the author, the docking control requirements are derived from actual mechanisms, for example the MIT Universal Docking Port and a scaled-down docking system of the ATV. The author also concludes that the docking mechanism must be able to handle 1 [cm] and 2 [deg] misalignment in each axis. Therefore, in order to have 100% margin, the control accuracy must be of 5 [mm] and 1 [deg] in each axis when docking. During approach, a typically-used rule is to be within 5% of the range to minimize the fuel, and within 10% of the range while in a SK point.

The overall control requirements are gathered in the following table.

| Phase | Approaching | Station Keeping | Docking |
|-----------------|-----------------|------------------|---------|
| Attitude | 2 [deg] | 4 [deg] | 1 [deg] |
| Position | 5% of the range | 10% of the range | 5 [mm] |

Table 2.2: Control requirements for the last stages of a CubeSat RVD mission.

Any approach manoeuvre towards the target must be done no faster than 1 [cm/s].

The navigation system must have a performance such that it allows the control system to fulfill the requirements stated in Tab. 2.2. Therefore, another common rule-of-thumb for the navigation is to allow an error of 1% of the range at any point.

As it has been stated in GNC subsystems section, Sec. 1.2, an important design requirement for a 6U CubeSat is that the systems are as much uniformly distributed as possible across the whole structure of the satellite, such that its CoM is located within a specific region not far from the geometric center. The ideal scenario would be that the geometrical center of the structure and CoM were at the same point, however this is often not achieved. It is then required that the CoM lies within 2 [cm] in the short axis, 7 [cm] in the long axis and 4.5 [cm] in the intermediate axis from its geometrical center [34]. Despite the CubeSat design not being part of the scope of this thesis, this mismatch between the geometrical center and the CoM leads to more disturbances that need to be accounted for in the simulations of the control strategies.

Spacecraft Motion

3

The mathematical models required for simulation and design of the GNC system are derived in this chapter. Two models can be found: an *absolute* model that describes the motion of a spacecraft CoM w.r.t. the orbital frame, and a *relative* model that describes the motion of the CoM of one satellite —the chaser— w.r.t the CoM of the other satellite —the target. Each of these models can be essentially divided into two parts: the *dynamics*, which relate the underlying forces and torques and physical parameters to the motion quantities; and the *kinematics*, which describes how the motion quantities evolve with time.

Equations governing the attitude are based on quaternions. For convenience, the attitude equations are expressed in the body frame of the chaser, \mathcal{F}_C . In opposition, the equations describing the relative displacement of the CoMs are obtained in the target's body frame, \mathcal{F}_T . This places the *zero* on the target's CoM.

The most relevant external and internal disturbances are also modeled in this chapter. These include the disturbance forces and torques provided by the magnetic residual dipole, gravitational gradient and aerodynamic drag, as well as sloshing and thrust effects.

The equations of motion describing these models are key for the Navigation and Control systems, as they will be used to provide estimates of the states of motion and to calculate suitable input commands to control the 6 DoF of the chaser satellite in order to perform the RVD maneuver.

3.1 Absolute Rotational Motion

The attitude and rotational motion of the spacecraft w.r.t. the local orbital frame, \mathcal{F}_O , is determined in this section. The kinematic equations describing the rate of change in orientation are expressed in the spacecraft's body frame, \mathcal{F}_B . If unit quaternions are used to represent the attitude of the spacecraft in Euclidean space, the time-varying quaternion that describes the orientation of a spacecraft s w.r.t. \mathcal{F}_O is defined as

$$\mathbf{q}_O^B \triangleq \begin{bmatrix} \eta_s \\ \boldsymbol{\varepsilon}_s \end{bmatrix} \in \mathbb{H}, \quad (3.1)$$

where \mathbb{H} denotes the set of quaternions with $\eta_s \in \mathbb{R}$ and $\boldsymbol{\varepsilon}_s \triangleq [\varepsilon_x \ \varepsilon_y \ \varepsilon_z]^T \in \mathbb{R}^3$. In order to describe a pure rotation, these parameters must satisfy the constraint

$$\eta_s^2 + \boldsymbol{\varepsilon}_s^T \boldsymbol{\varepsilon}_s = 1. \quad (3.2)$$

Then $\mathbf{q}_O^B \in \mathbb{H}_1$, where $\mathbb{H}_1 = \{\mathbf{q} \triangleq [\eta \ \boldsymbol{\varepsilon}^T]^T \in \mathbb{H} : \eta^2 + \boldsymbol{\varepsilon}^T \boldsymbol{\varepsilon} = 1\}$. The time-derivative of \mathbf{q}_O^B can be expressed as [24]

$$\dot{\mathbf{q}}_O^B = \begin{bmatrix} \dot{\eta}_s \\ \dot{\boldsymbol{\varepsilon}}_s \end{bmatrix} = \frac{1}{2} \begin{bmatrix} -\boldsymbol{\varepsilon}_s^T \\ \eta_s \mathbf{I}_3 - [\boldsymbol{\varepsilon}_s \times] \end{bmatrix} \boldsymbol{\omega}_B^{OB}, \quad (3.3)$$

where \mathbf{I}_n is a $n \times n$ identity matrix, $\boldsymbol{\omega}_B^{OB} \in \mathbb{R}^3$ is the angular velocity of the spacecraft s relative to the orbital frame represented in the spacecraft's body frame, and $\mathbf{S}(\cdot)$ is a skew-symmetric matrix defined as

$$[\boldsymbol{\varepsilon}_s \times] \triangleq \begin{bmatrix} 0 & -\varepsilon_z & \varepsilon_y \\ \varepsilon_z & 0 & -\varepsilon_x \\ -\varepsilon_y & \varepsilon_x & 0 \end{bmatrix}, \quad (3.4)$$

which is used to represent cross products as matrix multiplication. Notice that $[\boldsymbol{\varepsilon}_s \times] \boldsymbol{\omega}_B^{OB} = -[\boldsymbol{\omega}_B^{OB} \times] \boldsymbol{\varepsilon}_s$ following the rules of the cross product.

A time-varying rotational matrix, $\mathbf{R}\{\mathbf{q}_O^B\}$, describes the same rotation as \mathbf{q}_O^B , i.e. from the local orbital frame to the spacecraft body frame, and can be expressed in terms of the parameters of the quaternion as

$$\mathbf{R}\{\mathbf{q}_O^B\} = \mathbf{I}_3 + 2\eta_s[\boldsymbol{\varepsilon}_s \times] + 2[\boldsymbol{\varepsilon}_s \times]^2. \quad (3.5)$$

Then, $\mathbf{R}\{\mathbf{q}_O^B\} \in SO(3)$ group, being $SO(3) = \{\mathbf{R} \in \mathbb{R}^{3 \times 3} : \mathbf{R}^T \mathbf{R} = \mathbf{I}, \det(\mathbf{R}) = 1\}$. To reduce notation, it is assumed that $\mathbf{R}_O^B \triangleq \mathbf{R}\{\mathbf{q}_O^B\}$. The time-derivative of \mathbf{R}_O^B , is then given by [24]

$$\dot{\mathbf{R}}_O^B = [\boldsymbol{\omega}_B^{BO} \times] \mathbf{R}_O^B = \mathbf{R}_O^B [\boldsymbol{\omega}_O^{BO} \times]. \quad (3.6)$$

The attitude dynamics of the spacecraft s are governed by its angular momentum, $\mathbf{h} \in \mathbb{R}^3$. Since the angular momentum is a relative quantity, it is required to mention that its value is derived w.r.t. the origin of the inertial reference frame, as it is the only fixed point in the system. Besides, a subindex will be used to identify the reference frame used to express it. Hence, if the satellite s is assumed to be a rigid body, its angular momentum expressed in the inertial reference is defined as

$$\mathbf{h}_I = \mathbf{J}_I \boldsymbol{\omega}_I^{IB}, \quad (3.7)$$

and $\mathbf{J}_I \in \mathbb{S}_{>0}^3$, where $\mathbb{S}_{>0}^n = \{\mathbf{A} \in \mathbb{R}^{n \times n} : \mathbf{A} = \mathbf{A}^T, \mathbf{x}^T \mathbf{A} \mathbf{x} > 0, \forall \mathbf{x} \in \mathbb{R}^n \neq \mathbf{0}\}$. Then, \mathbf{J}_I is a symmetric positive definite matrix describing, in \mathcal{F}_I , the *moment of inertia* of the spacecraft. Following the same logic as before, $\boldsymbol{\omega}_I^{IB}$ is the angular velocity of the rotating reference \mathcal{F}_B w.r.t. the fixed directions of \mathcal{F}_I , also expressed in \mathcal{F}_I . Being the reference \mathcal{F}_I fixed, and the spacecraft s an object in motion, expressing the moment of inertia \mathbf{J} in the spacecraft body frame seems more suitable as it remains constant and it is usually easier to calculate. To do so, the angular momentum is represented in the spacecraft body frame as

$$\mathbf{h}_B = \mathbf{R}_I^B \mathbf{h}_I = \mathbf{R}_I^B \mathbf{J}_I (\mathbf{R}_I^B)^T \mathbf{R}_I^B \boldsymbol{\omega}_I^{IB} = \mathbf{J}_B \boldsymbol{\omega}_B^{IB}. \quad (3.8)$$

Where \mathbf{R}_I^B is a rotation matrix that transforms elements from the inertial to the body coordinate system, and $(\mathbf{R}_I^B)^T$ is the transpose of \mathbf{R}_I^B . Since $\mathbf{R}_I^B \in \mathcal{SO}(3)$, the product $(\mathbf{R}_I^B)^T \mathbf{R}_I^B$ does not alter the equation in any sense and, by association, the following changes of reference can be noticed.

$$\mathbf{J}_B = \mathbf{R}_I^B \mathbf{J}_I (\mathbf{R}_I^B)^T, \quad (3.9a)$$

$$\boldsymbol{\omega}_B^{IB} = \mathbf{R}_I^B \boldsymbol{\omega}_I^{IB}. \quad (3.9b)$$

Since RWs are used as the attitude actuation system to perform RVD, an extra angular momentum must be added in equation (3.8) to attain for the moving parts of the system. \mathbf{h}_B^{RW} is, then, defined as the angular momentum of the RWs represented in the spacecraft body frame, which axes are assumed to be aligned with the angular momentums of each individual RW¹.

$$\mathbf{h}_B = \mathbf{J}_B \boldsymbol{\omega}_B^{IB} + \mathbf{h}_B^{RW} \quad (3.10)$$

According to Newton's second law, the change in *momentum* is equal to the external forces acting on the system. For rotations, that is

$$\frac{d\mathbf{h}_I}{dt} = \boldsymbol{\tau}_I^{ext}, \quad (3.11)$$

¹See Sec. 2.1

being $\boldsymbol{\tau}_I^{ext} \in \mathbb{R}^3$ the external torques acting on the system, represented in \mathcal{F}_I . Analyzing (3.11), it can be noticed that if no external torques are applied to the system, its angular momentum is conserved. This is known as the *principle of conservation of the angular momentum*. For a reference frame fixed on a rotating object, \mathcal{F}_B in this case, the change in angular momentum is slightly modified by taking equation (3.8) and applying the chain rule.

$$\frac{d\mathbf{h}_B}{dt} = \frac{d\mathbf{R}_I^B}{dt} \mathbf{h}_I + \mathbf{R}_I^B \frac{d\mathbf{h}_I}{dt}. \quad (3.12)$$

By taking the definitions in (3.6) and (3.11), equation (3.12) can be rewritten as

$$\frac{d\mathbf{h}_B}{dt} = -[\boldsymbol{\omega}_B^{IB} \times] \mathbf{R}_I^B \mathbf{h}_I + \mathbf{R}_I^B \boldsymbol{\tau}_I^{ext} = -[\boldsymbol{\omega}_B^{IB} \times] \mathbf{h}_B + \boldsymbol{\tau}_B^{ext}. \quad (3.13)$$

Taking the expression of \mathbf{h}_B from equation (3.10) and replacing in equation (3.13) yield to

$$\mathbf{J}_B \dot{\boldsymbol{\omega}}_B^{IB} + \dot{\mathbf{h}}_B^{RW} = -[\boldsymbol{\omega}_B^{IB} \times] \mathbf{J}_B \boldsymbol{\omega}_B^{IB} - [\boldsymbol{\omega}_B^{IB} \times] \mathbf{h}_B^{RW} + \boldsymbol{\tau}_B^{ext}. \quad (3.14)$$

Finally, replacing $\dot{\mathbf{h}}_B^{RW}$ by $\boldsymbol{\tau}_B^{RW}$ and rearranging the terms, equation (3.14) becomes

$$\dot{\boldsymbol{\omega}}_B^{IB} = \mathbf{J}_B^{-1} \left(-[\boldsymbol{\omega}_B^{IB} \times] \mathbf{J}_B \boldsymbol{\omega}_B^{IB} - [\boldsymbol{\omega}_B^{IB} \times] \mathbf{h}_B^{RW} - \boldsymbol{\tau}_B^{RW} + \boldsymbol{\tau}_B^{ext} \right), \quad (3.15)$$

where $[\boldsymbol{\omega}_B^{IB} \times] \mathbf{J}_B \boldsymbol{\omega}_B^{IB}$ and $[\boldsymbol{\omega}_B^{IB} \times] \mathbf{h}_B^{RW}$ are the cross coupling terms arising from the fact of describing the dynamics in a rotating reference frame. The terms $\boldsymbol{\tau}_B^{RW}$ and $\boldsymbol{\tau}_B^{ext}$ in (3.15) are the reaction torques from the RWs —the control inputs— and the external torques exerted to the body of the spacecraft —mainly disturbances—, respectively.

Because circular orbits are assumed, it is relatively easy to parameterize the angular velocity of the orbital reference frame as $\boldsymbol{\omega}_O^{IO} = [0 \ -\omega_o \ 0]^T$, being $\omega_o \in \mathbb{R}$ the mean angular velocity of the orbit defined in Sec. 2.2.2. Due to this fact, and that CubeSats are nominally in *Nadir* or *Zenith* modes, it seems more convenient to express the dynamics w.r.t. the local orbital frame \mathcal{F}_O instead, i.e. $\dot{\boldsymbol{\omega}}_B^{OB}$. Then, if

$$\boldsymbol{\omega}_B^{IB} = \boldsymbol{\omega}_B^{OB} + \mathbf{R}_O^B \boldsymbol{\omega}_O^{IO}, \quad (3.16)$$

the final equation describing the change in motion relative to the orbital frame can be simplified considering $\boldsymbol{\omega}_O^{IO}$ negligible compared to $\dot{\boldsymbol{\omega}}_B^{OB}$.

$$\dot{\boldsymbol{\omega}}_B^{OB} = \mathbf{J}_B^{-1} \left(-[\boldsymbol{\omega}_B^{OB} + \mathbf{R}_O^B \boldsymbol{\omega}_O^{IO} \times] \mathbf{J}_B \boldsymbol{\omega}_B^{IB} - [\boldsymbol{\omega}_B^{OB} + \mathbf{R}_O^B \boldsymbol{\omega}_O^{IO} \times] \mathbf{h}_B^{RW} - \boldsymbol{\tau}_B^{RW} + \boldsymbol{\tau}_B^{ext} \right) - \quad (3.17)$$

The equations governing the absolute attitude motion are equations (3.17) and (3.3). By replacing index B to C or \mathcal{T} , equations describe the chaser or the target spacecraft attitude motion respectively.

3.2 Relative Rotational Motion

In the last stages in a RVD/B operation, control is usually not performed from ground. In this last stage, the active spacecraft is close enough to takes advantage of relative measurements from the Navigation system². Consequently, both —Navigation and Control systems— are operating on board of the spacecraft, which require relative models for the design of filters and control strategies.

In this section, the relative attitude model is derived based also in unit quaternions. The definition of unit quaternion can be taken from the previous section, emphasizing whether it is describing the orientation of the chaser or the target body frames w.r.t. \mathcal{F}_O by replacing c or t where s . According to this, the attitude representations of the chaser and target are

$$\mathbf{q}_O^C \triangleq \begin{bmatrix} \eta_c \\ \boldsymbol{\varepsilon}_c \end{bmatrix} \in \mathbb{H}_1, \quad (3.18a)$$

$$\mathbf{q}_O^T \triangleq \begin{bmatrix} \eta_t \\ \boldsymbol{\varepsilon}_t \end{bmatrix} \in \mathbb{H}_1, \quad (3.18b)$$

respectively. If the relative attitude is defined as the rotation $\mathbf{q} \in \mathbb{H}_1$ left to align the moving references \mathcal{F}_c and \mathcal{F}_t , then

$$\mathbf{q}_O^C = \mathbf{q} \otimes \mathbf{q}_O^T, \quad (3.19)$$

where $\mathbf{q} \triangleq \mathbf{q}_{\mathcal{T}}^C = [\eta \ \boldsymbol{\varepsilon}^T]^T$ is the unit quaternion describing the relative attitude, and \otimes is the quaternion product. From equation (3.19), the relative attitude is found as follows.

$$\mathbf{q} = \mathbf{q}_O^C \otimes (\mathbf{q}_O^T)^{-1} = \begin{bmatrix} \eta_c \eta_t - \boldsymbol{\varepsilon}_c^T \boldsymbol{\varepsilon}_t \\ \eta_t \boldsymbol{\varepsilon}_c + \eta_c \boldsymbol{\varepsilon}_t + [\boldsymbol{\varepsilon}_c \times] \boldsymbol{\varepsilon}_t \end{bmatrix}, \quad (3.20)$$

being $(\mathbf{q}_O^T)^{-1} = [\eta_t \ -\boldsymbol{\varepsilon}_t^T]^T$ the inverse of \mathbf{q}_O^T , which coincides with the conjugate operation for unit quaternions.

Having the quaternion kinematics in equation (3.3), the time-derivative of the relative attitude \mathbf{q} can be analogously expressed as

$$\dot{\mathbf{q}} = \begin{bmatrix} \dot{\eta} \\ \dot{\boldsymbol{\varepsilon}} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} -\boldsymbol{\varepsilon}^T \\ \eta \mathbf{I}_3 - [\boldsymbol{\varepsilon} \times] \end{bmatrix} \boldsymbol{\omega}, \quad (3.21)$$

where $\boldsymbol{\omega} \triangleq \boldsymbol{\omega}_C^{\mathcal{T}} \in \mathbb{R}^3$ is the angular velocity of the chaser relative to the target, represented in \mathcal{F}_C . As it is the relative angular velocity, it can be written as the

²See Sec. 2.3

difference between the angular velocity of the two spacecrafts as follows.

$$\boldsymbol{\omega} = \boldsymbol{\omega}_C^{OC} - \mathbf{R}_T^C \boldsymbol{\omega}_T^{OT}. \quad (3.22)$$

The relative attitude dynamics are found by differentiating the relation in (3.22) w.r.t time, yielding to

$$\dot{\boldsymbol{\omega}} = \dot{\boldsymbol{\omega}}_C^{OC} - \dot{\mathbf{R}}_T^C \boldsymbol{\omega}_T^{OT} - [\boldsymbol{\omega}_C^{CT} \times] \mathbf{R}_T^C \boldsymbol{\omega}_T^{OT}. \quad (3.23)$$

Notice that $\boldsymbol{\omega}_C^{CT} = -\boldsymbol{\omega}_C^{TC} = -\boldsymbol{\omega}$. By means of the absolute attitude dynamics, terms $\dot{\boldsymbol{\omega}}_C^{OC}$ and $\dot{\boldsymbol{\omega}}_T^{OT}$ in equation (3.23) can be replaced.

$$\begin{aligned} \dot{\boldsymbol{\omega}} = & \mathbf{J}_C^{-1} \left(-[\boldsymbol{\omega}_C^{IC} \times] \mathbf{J}_C \boldsymbol{\omega}_C^{IC} - [\boldsymbol{\omega}_C^{IC} \times] \mathbf{h}_C^{RW} - \boldsymbol{\tau}_C^{RW} + \boldsymbol{\tau}_C^{ext} \right) \\ & + \mathbf{R}_T^C \mathbf{J}_T^{-1} \left([\boldsymbol{\omega}_T^{IT} \times] \mathbf{J}_T \boldsymbol{\omega}_T^{IT} + [\boldsymbol{\omega}_T^{IT} \times] \mathbf{h}_T^{RW} + \boldsymbol{\tau}_T^{RW} - \boldsymbol{\tau}_T^{ext} \right) \\ & + [\boldsymbol{\omega} \times] \mathbf{R}_T^C \boldsymbol{\omega}_T^{OT}, \end{aligned} \quad (3.24)$$

Since absolute and relative kinematics and dynamics have been derived at this point, it can be noticed that the attitude motion of one spacecraft can be parameterized using the absolute attitude kinematics and dynamics variables of the other spacecraft and the variables of the relative motion. To choose which option is the best —either describe the chaser variables in terms of the target's and the relative motions, or the other way around— consider the complete relative equations. These equations describe the 6 DoF of the relative motion between the CoMs of the chaser and target spacecraft, where the motion of one satellite can be interpreted as disturbances for the other satellite. Given that the motion of the target is usually known a priori in most of the RVD missions, it is more convenient to express the chaser's motion in terms of the target's and the relative motion variables. This can be done replacing the following equation in (3.24).

$$\boldsymbol{\omega}_C^{IC} = \boldsymbol{\omega} + \mathbf{R}_T^C \boldsymbol{\omega}_T^{OT} + \mathbf{R}_T^C \mathbf{R}_O^T \boldsymbol{\omega}_O^{IO}, \quad (3.25)$$

where $\mathbf{R}_O^T \triangleq \mathbf{R}\{\mathbf{q}_O^T\}$ and $\mathbf{R}_T^C \triangleq \mathbf{R}\{\mathbf{q}\}$ can be found using the absolute and relative attitude equations respectively, employing the relation in (3.5) to transform the unit quaternions to rotational matrices. $\boldsymbol{\omega}_T^{OT}$ and $\boldsymbol{\omega}$ can also be obtained from the absolute and relative attitude equations respectively, and again, $\boldsymbol{\omega}_O^{IO} = [0 \ -\omega_o \ 0]^T$ because circular orbits are assumed.

The mathematical model for the relative attitude contains equations (3.21), (3.24) with (3.25). The model describing the absolute attitude motion of the target are equations (3.3) and (3.17).

3.3 Relative Translational Motion

A nonlinear model describing the relative position of the chaser's docking port w.r.t the target's docking port is derived in [30]. The special characteristic of this model is the coupling between the attitude and the position states, given that rotations of the spacecraft induces, not only rotations, but also translations of the docking port frame. This effect can be seen in Fig. 3.1.

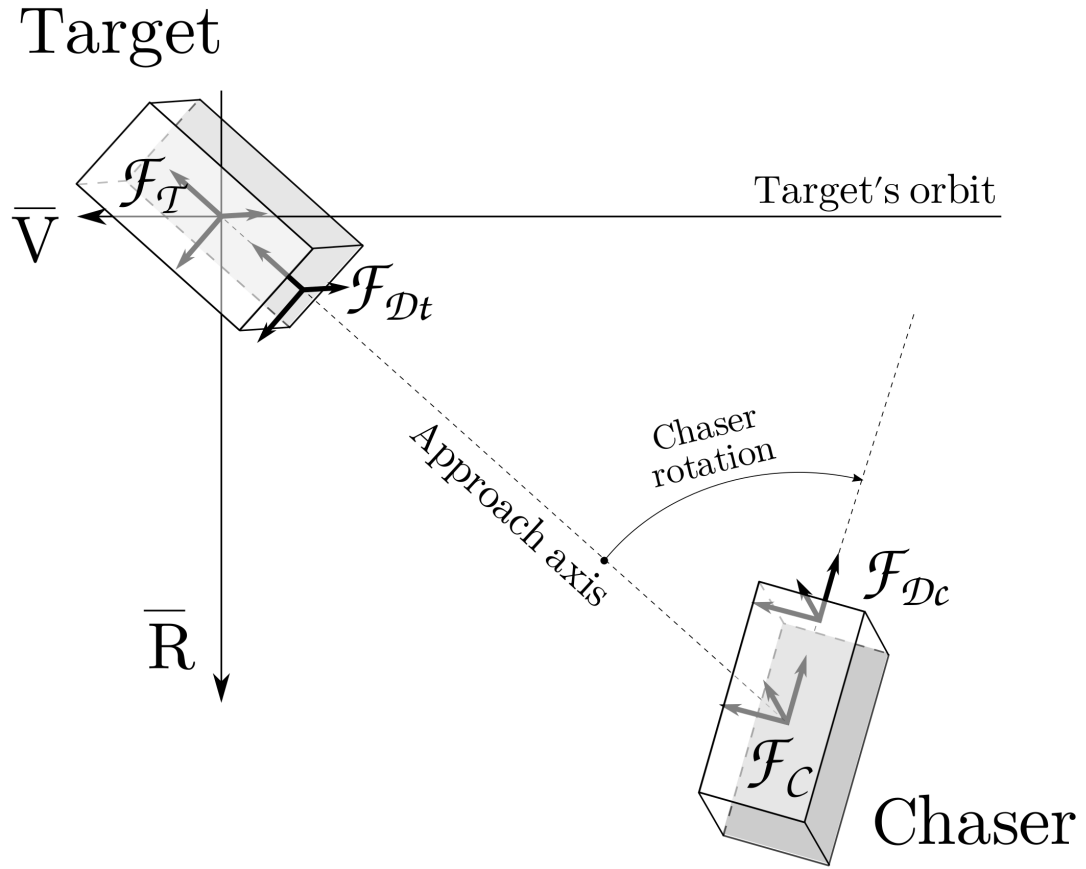


Figure 3.1: Representation of a translation induced in the chaser's docking port \mathcal{F}_{Dc} , w.r.t. the target's docking port, \mathcal{F}_{Dt} , due to a change in attitude.

In a real scenario, small deformations in the structure of the spacecraft might occur—although generally negligible for small satellites—, for instance, due to expansion and contraction of the materials exposed to extreme changing temperatures. Thus, it is coherent to search for a model describing directly the relative motion of the docking ports. However, these equations of motion appear to be highly nonlinear and extremely coupled, as mentioned before, turning the linearization process a tough task.

Despite the real interest lying on the relative motion of the docking ports, simpler

equations can be formulated for the CoM. Subsequently, the attitude motion equations derived above can be combined with this model to find the relative position of the docking ports. This can only be done under the assumption of rigid body for the spacecraft.

In practice, torsion and deflection deformations of the solid structure of a 6U CubeSat—not including appendages—are not considered in CubeSats RVD missions, due to the fact of being negligible; and deformations due to vibrations are mainly of concern during the launching phase. In-orbit, though, it is more likely that uneven expansion and contraction of the structure of the small satellite can take place caused by a non-uniform heating. This deformation can provoke a misalignment between the modeled and the true orientation and position of the components fixed in the structure. In [17], however, a maximum deviation have been estimated, for two different 3U CubeSats, of 0.03 [deg], which is not commensurable with the order of magnitude of the RVD requirements. It is fair, thus, to assume that the CubeSat structure behaves strictly as a rigid body.

In the following chapter, i.e. Chap. 4, control strategies are investigated to perform CubeSat RVD. In particular, a linear MPC strategy is introduced for the relative position control. This type of controller is based on solving multiple optimization problems, and is further detailed in the Sec. 4.4.1. However, an important aspect to advance is the capacity of this strategy to handle constraints in the states and inputs, since there lies the motivation for the choices made in this section.

Despite this controller having this appealing feature, constraints play an important role in the solution of the optimization problems, and must be look thorough. In this specific case, constraints arise from the control requirements stated in Sec. 2.4, where velocity and position error requirements must be continuously fulfilled. In order to avoid coupling of the states when defining some of the constraints for the MPC problem—this occurs with the model derived in Appendix A.1, describing the relative position in the orbital frame of the target—which can lead to minimum solutions that are not global, a model that describes the position of the chaser from the body frame of the target is considered. It is seen in Sec. 4.4.1 that a particular kind of nonlinear constraints still appear, however, not presenting important issues for the optimization problem.

Let $\mathbf{r}_{\mathcal{T}}^{\mathcal{T}C} \in \mathbb{R}^3$ be defined as the distance from the origin of the target body frame to the origin of the chaser body frame, i.e. from the CoM of the target to the CoM of the chaser, expressed in the target body frame. And $\dot{\mathbf{r}}_{\mathcal{T}}^{\mathcal{T}C} \in \mathbb{R}^3$ the time-derivative of $\mathbf{r}_{\mathcal{T}}^{\mathcal{T}C}$. These two vectors are the controlled variables, along with $\mathbf{q} \triangleq \mathbf{q}_{\mathcal{T}}^C$ and $\boldsymbol{\omega} \triangleq \boldsymbol{\omega}_C^{\mathcal{T}C}$, in the subsequent chapter.

In the inertial frame \mathcal{F}_I , the relative position vector $\mathbf{r}_I^{\mathcal{T}C} \in \mathbb{R}^3$ is

$$\mathbf{r}_I^{\mathcal{T}C} = \mathbf{r}_I^{IC} - \mathbf{r}_I^{IT}, \quad (3.26)$$

where $\mathbf{r}_I^{IC}, \mathbf{r}_I^{IT} \in \mathbb{R}^3$ are the distance from the CoM of the Earth to the CoM of the chaser and the target respectively. The relative acceleration vector in the inertial frame are straightforward, according to Newton's gravitational law.

$$\ddot{\mathbf{r}}_I^{TC} = \ddot{\mathbf{r}}_I^{IC} - \ddot{\mathbf{r}}_I^{IT} = -\mu \frac{\mathbf{r}_I^{IC}}{\|\mathbf{r}_I^{IC}\|_2^3} + \mu \frac{\mathbf{r}_I^{IT}}{\|\mathbf{r}_I^{IT}\|_2^3} + \frac{\mathbf{f}_I^{tot}}{m_c}. \quad (3.27)$$

In (3.27), $m_c \in \mathbb{R}$ is the mass of the chaser and $\mathbf{f}_I^{tot} \in \mathbb{R}^3$ is a sum of the forces applied to the chaser, expressed in the inertial frame. These forces are, the thrust reaction of the RCP, $\mathbf{f}_I^{th} \in \mathbb{R}^3$, and the disturbance forces, $\mathbf{f}_I^{dist} \in \mathbb{R}^3$ derived in the next sections.

It is desired to express the relative dynamics in the target body frame, \mathcal{F}_T , in which equation (3.26) transforms to

$$\mathbf{r}_T^{TC} = \mathbf{R}_O^T \mathbf{R}_I^O \mathbf{r}_I^{TC} = \mathbf{R}_I^T \mathbf{r}_I^{TC} \quad (3.28)$$

Differentiating w.r.t. time yields to the velocity vector, which is expressed as

$$\dot{\mathbf{r}}_T^{TC} = -[\omega_T^{IT} \times] \mathbf{R}_I^T \mathbf{r}_I^{TC} + \mathbf{R}_I^T \dot{\mathbf{r}}_I^{TC}, \quad (3.29)$$

and differentiating once again to obtain the acceleration,

$$\begin{aligned} \ddot{\mathbf{r}}_T^{TC} = & -[\dot{\omega}_T^{IT} \times] \mathbf{R}_I^T \mathbf{r}_I^{TC} + [\omega_T^{IT} \times][\omega_T^{IT} \times] \mathbf{R}_I^T \mathbf{r}_I^{TC} \\ & - 2[\omega_T^{IT} \times] \mathbf{R}_I^T \dot{\mathbf{r}}_I^{TC} + \mathbf{R}_I^T \ddot{\mathbf{r}}_I^{TC}. \end{aligned} \quad (3.30)$$

Following the same procedure as in the previous section, the term $\mathbf{R}_I^T \dot{\mathbf{r}}_I^{TC}$ in (3.29) can be isolated and replaced in equation (3.30). Rearranging the terms, this leads to

$$\begin{aligned} \ddot{\mathbf{r}}_T^{TC} = & -[\dot{\omega}_T^{IT} \times] \mathbf{R}_I^T \mathbf{r}_I^{TC} - [\omega_T^{IT} \times][\omega_T^{IT} \times] \mathbf{R}_I^T \mathbf{r}_I^{TC} \\ & - 2[\omega_T^{IT} \times] \dot{\mathbf{r}}_T^{TC} + \mathbf{R}_I^T \ddot{\mathbf{r}}_I^{TC}. \end{aligned} \quad (3.31)$$

Finally, the expression for $\ddot{\mathbf{r}}_I^{TC}$ obtained in (3.27) can be again replaced to achieve the last equation of motion.

$$\begin{aligned} \ddot{\mathbf{r}}_T^{TC} = & -[\dot{\omega}_T^{IT} \times] \mathbf{R}_I^T \mathbf{r}_I^{TC} - [\omega_T^{IT} \times][\omega_T^{IT} \times] \mathbf{R}_I^T \mathbf{r}_I^{TC} \\ & - 2[\omega_T^{IT} \times] \dot{\mathbf{r}}_T^{TC} + \mathbf{R}_I^T \left(-\mu \frac{\mathbf{r}_I^{IC}}{\|\mathbf{r}_I^{IC}\|_2^3} + \mu \frac{\mathbf{r}_I^{IT}}{\|\mathbf{r}_I^{IT}\|_2^3} + \frac{\mathbf{f}_I^{tot}}{m_c} \right). \end{aligned} \quad (3.32)$$

By rearranging the terms and recalling the transformations $\omega_T^{IT} = \omega_T^{OT} + \mathbf{R}_I^O \omega_O^{IO}$ and $\mathbf{r}_I^{IC} = \mathbf{r}_T^{TC} + \mathbf{R}_O^T \mathbf{r}_O^{OI}$, the previous equation can be written in a way that it only depends

on the relative states.

$$\begin{aligned} \ddot{\mathbf{r}}_{\mathcal{T}}^{\mathcal{T}C} = & -[\dot{\omega}_{\mathcal{T}}^{\mathcal{OT}} \times] \mathbf{r}_{\mathcal{T}}^{\mathcal{T}C} - [\omega_{\mathcal{T}}^{\mathcal{OT}} + \mathbf{R}_I^O \omega_O^{\mathcal{IO}} \times][\omega_{\mathcal{T}}^{\mathcal{OT}} + \mathbf{R}_I^O \omega_O^{\mathcal{IO}} \times] \mathbf{r}_{\mathcal{T}}^{\mathcal{T}C} \\ & - 2[\omega_{\mathcal{T}}^{\mathcal{OT}} + \mathbf{R}_I^O \omega_O^{\mathcal{IO}} \times] \dot{\mathbf{r}}_{\mathcal{T}}^{\mathcal{T}C} - \mu \frac{\mathbf{r}_{\mathcal{T}}^{\mathcal{T}C} + \mathbf{R}_O^{\mathcal{T}} \mathbf{r}_O^{\mathcal{OI}}}{\|\mathbf{r}_{\mathcal{T}}^{\mathcal{T}C} + \mathbf{R}_O^{\mathcal{T}} \mathbf{r}_O^{\mathcal{OI}}\|_2^3} + \mu \frac{\mathbf{R}_O^{\mathcal{T}} \mathbf{r}_O^{\mathcal{IO}}}{\|\mathbf{R}_O^{\mathcal{T}} \mathbf{r}_O^{\mathcal{IO}}\|_2^3} + \frac{\mathbf{f}_{\mathcal{T}}^{\text{tot}}}{m_c}. \end{aligned} \quad (3.33)$$

In order to express the force, $\mathbf{f}_C^{\text{tot}}$, in the chaser body frame, the following transformation can be applied, which adds a small coupling between attitude and position states.

$$\mathbf{f}_{\mathcal{T}}^{\text{tot}} = (\mathbf{R}_{\mathcal{T}}^C)^T \mathbf{f}_C^{\text{tot}}, \quad (3.34)$$

with $\mathbf{R}_{\mathcal{T}}^C \triangleq \mathbf{R}\{\mathbf{q}\}$. Following the notation used to denote the relative attitude, the relative position vector $\mathbf{r}_{\mathcal{T}}^{\mathcal{T}C}$ will be denoted as \mathbf{r} , and its time-derivative as \mathbf{v} . The relative coupled model for the CoM expressed in the target's body frame is

$$\dot{\mathbf{r}} = \mathbf{v} \quad (3.35a)$$

$$\begin{aligned} \dot{\mathbf{v}} = & -[\dot{\omega}_{\mathcal{T}}^{\mathcal{OT}} \times] \mathbf{r} - 2[\omega_{\mathcal{T}}^{\mathcal{OT}} + \mathbf{R}_I^O \omega_O^{\mathcal{IO}} \times] \mathbf{v} \\ & - [\omega_{\mathcal{T}}^{\mathcal{OT}} + \mathbf{R}_I^O \omega_O^{\mathcal{IO}} \times][\omega_{\mathcal{T}}^{\mathcal{OT}} + \mathbf{R}_I^O \omega_O^{\mathcal{IO}} \times] \mathbf{r} \\ & - \mu \frac{\mathbf{r} + \mathbf{R}_O^{\mathcal{T}} \mathbf{r}_O^{\mathcal{OI}}}{\|\mathbf{r} + \mathbf{R}_O^{\mathcal{T}} \mathbf{r}_O^{\mathcal{OI}}\|_2^3} + \mu \frac{\mathbf{R}_O^{\mathcal{T}} \mathbf{r}_O^{\mathcal{IO}}}{\|\mathbf{R}_O^{\mathcal{T}} \mathbf{r}_O^{\mathcal{IO}}\|_2^3} + \frac{1}{m_c} (\mathbf{R}_{\mathcal{T}}^C)^T \mathbf{f}_C^{\text{tot}}. \end{aligned} \quad (3.35b)$$

3.4 External Disturbances

This section addresses the modeling of the interaction of the spacecraft with the environment and other sources of disturbance. Only the most significant disturbances during the Final Approach have been taken into account.

3.4.1 Residual Magnetic Dipole

Interaction with the magnetic field of the Earth can produce a change in the momentum of the spacecraft, precisely in the angular momentum. This change in the angular momentum is exclusively accounted for in LEO, where the magnitude of the field is large enough to produce significant torques. In some cases, though, magnetic control has been used in higher orbits.

An important drawback of employing the geomagnetic field as a way to control the spacecraft's attitude is that only 2 out of the 3 DoF are controllable at any specific time.

The Earth's rotation, however, can cause that the direction of the 2 controllable DoF change with time achieving full controllability for determined orbits [27]. Generally, this is why RWs are employed for attitude control, instead of MTQs which are reserved for angular momentum dumping of the spacecraft and the RWs.

Another drawback is that magnetic torques can also be generated from undesired magnetic dipoles that originate in the spacecraft, called residual magnetic dipoles. This residual dipole for a 6U was approximated for the ARPAIMA satellite and given in [21] to have a magnitude of 0.1 [Am²], and to be aligned with one of the body axes. For the simulations, the direction of the magnetic dipole is assumed random for a robustness check.

To model the geomagnetic field, a truncated version of the International Geomagnetic Reference Field-12 (IGRF-12) [35] is derived. Since the magnetic field is a conservative quantity, the geomagnetic potential $V_b \in \mathbb{R}$ can be used to obtain the magnetic field vector $\mathbf{b} \in \mathbb{R}^3$. The magnetic potential is approximated in [35] as a series of spherical harmonics.

$$V_b = a \sum_{n=1}^N \sum_{m=0}^n \left(\frac{a}{r}\right)^{n+1} [g_n^m(t) \cos(m\Phi) + h_n^m(t) \sin(m\Phi)] P_n^m(\cos \theta), \quad (3.36)$$

where $a = 6371.2$ [km], which is the magnetic reference radius, r the magnitude of the distance between the center of the Earth and the spacecraft's CoM, $g_n^m(t)$ and $h_n^m(t)$ are coefficients tabulated in [35], and $P_n^m(\cos \theta)$ are the well-known associated *Legendre* functions. Currently, IGRF-12 has a precision up to $N = 13$ which makes the model differ 1% from the true magnetic field. For the sake of simplicity, it is only considered until $n = 1$, which from (3.36) gives the following potential [27]

$$V_b = \frac{a^3}{r^2} [g_1^0 \sin(\theta) + g_1^1 \cos \theta \cos \Phi + h_1^1 \cos \theta \sin \Phi] = \frac{\mathbf{m}^T \mathbf{r}}{r^3}, \quad (3.37)$$

which can be used to approximate the magnetic dipole of the Earth, $\mathbf{m} \in \mathbb{R}^3$. Arranging the terms in (3.37),

$$\mathbf{m}_E = a^3 \begin{bmatrix} g_1^1 \\ h_1^1 \\ g_1^0 \end{bmatrix} \quad (3.38)$$

and replacing the coefficients by the last values provided in [35], those are $g_1^1 = -1501.0$ [nT], $h_1^1 = 4797.1$ [nT], $g_1^0 = -29442.0$ [nT]. Note, that the magnetic dipole is expressed in the ECEF reference, given that the magnetic field is moving with this frame. Finally, the magnetic field is obtained from the potential in (3.37) with the Earth's magnetic dipole \mathbf{m} .

$$\mathbf{b}_E = -\nabla V_b = \frac{3(\mathbf{m}_E^T \mathbf{r}_E) \mathbf{r}_E - r^2 \mathbf{m}_E}{r^5}. \quad (3.39)$$

And the torque provided by the residual magnetic dipole interacting with \mathbf{b} is

$$\tau_B = \mathbf{m}_B \times \mathbf{b}_B. \quad (3.40)$$

3.4.2 Gravity Gradient

The orientation of non-symmetric spacecrafts are conditioned by the gravitational field, as well as any spacecraft orbiting in non-circular orbits. In both cases, there is an uneven contribution of the gravitational force on the spacecraft body that results in a torque that tends to align the principal axis of inertia of the spacecraft with $\bar{\mathbf{R}}$.

For the purpose of modeling this torque, it is adequate to consider the gravity field spherically, assuming an spherically-symmetric and homogeneous Earth [27]. Then, the gravity-gradient torque, $\boldsymbol{\tau}_{\mathcal{B}}^{gg} \in \mathbb{R}^3$, can be defined as

$$\boldsymbol{\tau}_{\mathcal{B}}^{gg} = \frac{3\mu}{\|\mathbf{r}_{\mathcal{B}}^{BI}\|^3} [\hat{\mathbf{r}}_{\mathcal{B}}^{BI} \times (\mathbf{J}_{\mathcal{B}} \hat{\mathbf{r}}_{\mathcal{B}}^{BI})] \quad (3.41)$$

where $\mathbf{r}_{\mathcal{B}}^{BI} \in \mathbb{R}^3$ is the position vector from the spacecraft's CoM to the Earth's CoM, expressed in the body frame —also called *Nadir* pointing vector. Then, $\hat{\mathbf{r}}_{\mathcal{B}}^{BI}$ is the same vector normalized. The latter vector is easily expressed in the orbit frame as $\hat{\mathbf{r}}_{\mathcal{O}}^{BI} = [0 \ 0 \ 1]^T$. Thus, a simple transformation can be used relating the attitude quaternions to transform the reference frame.

$$\hat{\mathbf{r}}_{\mathcal{B}}^{BI} = \mathbf{R}_{\mathcal{O}}^{\mathcal{B}} \hat{\mathbf{r}}_{\mathcal{O}}^{BI}, \quad (3.42)$$

where $\mathbf{R}_{\mathcal{O}}^{\mathcal{B}} \in SO(3)$ is defined using, as previously mentioned, the attitude quaternions. Since in this case the gravity-gradient torque is applied to the chaser, the rotation matrix is defined as $\mathbf{R}_{\mathcal{O}}^{\mathcal{C}} \triangleq \mathbf{R}\{\mathbf{q}\}\mathbf{R}\{\mathbf{q}_0^T\}$. Moreover, note that the distribution of the mass affects the torque by means of the inertia tensor $\mathbf{J}_{\mathcal{B}}$.

3.4.3 Aerodynamic Drag

Spacecrafts at LEO are subject to disturbances originated by the displacement of the object through the remaining particles of the atmosphere. The collisions with these particles produce a force opposite to the direction of motion, which cause a drag on the spacecraft, eventually losing altitude due to a reduction of its velocity. This force can be modeled by the aerodynamic force

$$\mathbf{f}_{\mathcal{I}}^{drag} = -\frac{1}{2}\varrho C_D A_s \|\mathbf{v}_{\mathcal{I}}^{rel}\|_2 \mathbf{v}_{\mathcal{I}}^{rel}. \quad (3.43)$$

In (3.43), $C_D = 2$ for CubeSats and it is the drag coefficient, $A_s \in \mathbb{R}$ is the incident area, $\mathbf{v}_{\mathcal{B}}^{rel} \in \mathbb{R}^3$ is the relative velocity between the atmosphere and the spacecraft and $\varrho \in \mathbb{R}$ is the atmospheric density.

The atmospheric density can simply be modeled using the exponentially decaying model [27]

$$\varrho = \varrho_0 \exp\left(-\frac{h - h_0}{H}\right), \quad (3.44)$$

with $\varrho_0, h_0 \in \mathbb{R}$ being the reference density and altitude respectively. On the other hand, $h \in \mathbb{R}$ is the altitude from the Earth's surface, and $H \in \mathbb{R}$ is a scaling factor. All three parameters in the exponential model, i.e. ϱ_0, h_0 and H , can be found tabulated.

The maximal area has been selected for A_s in order to represent the worst case.

Since the drag force is applied on the structure of the spacecraft, a torque is also originated. This torque is

$$\boldsymbol{\tau}_{\mathcal{B}}^{drag} = \mathbf{r}_{\mathcal{B}}^{drag} \times \mathbf{R}_I^{\mathcal{B}} \mathbf{f}_I^{drag}, \quad (3.45)$$

with $\mathbf{r}_{\mathcal{B}}^{drag} \in \mathbb{R}^3$ being the distance from the CoM to the furthest point in the spacecraft to keep attaining for the worst case.

3.4.4 Thrust Missalignment

Due to missalignments between the thrusters axis and the CoM of the satellite, a torque can be originated as a result of the force applied by the RCP when expelling the gas. The resultant torque is defined as

$$\boldsymbol{\tau}_{\mathcal{B}}^{th} = \mathbf{r}_{\mathcal{B}}^{CoM} \times \mathbf{f}_{\mathcal{B}}^{th}, \quad (3.46)$$

where $\mathbf{f}_{\mathcal{B}}^{th}$ is the thrust provided by the RCP, and $\mathbf{r}_{\mathcal{B}}^{CoM}$ a vector from the CoM to the geometrical center of the spacecraft.

3.5 Internal Disturbances

Given that a RCP system is required in a RVD mission, an important kind of disturbances originates as an interaction of the propellant with the walls of the tank, which modify the dynamics of the spacecraft. This problem is referred in the literature as *slosh dynamics*. This is classified as an internal disturbance, and there exist other types, such as the effects of flexible appendages —solar arrays or antennae. However, they have not been included in this work nor the simulations.

3.5.1 Slosh Dynamics

In order to model the behaviour of a fluid substance inside the tank, the best approach would be to take the *Navier-Stokes* equations. Nevertheless, these equations are highly nonlinear and are needed to be added together with the dynamic model in Sec. 3, which would add a lot of complexity to the model. As in [30], a simpler approach is attempted by assuming that the moment applied by the propellant to the spacecraft behaves as a *mass-spring-damper* system.

Consider the propellant with time-varying mass $m_s \in \mathbb{R}_{\geq 0}$, which represents the mass left in the tank, and the position of its center of mass $\mathbf{x}_B^{sl} \in \mathbb{R}^3$ expressed in the body frame. The *mass-spring-damper* system in \mathcal{F}_B is governed by the next equation.

$$m_s \ddot{\mathbf{x}}_B^{sl} = -k_s \mathbf{x}_B^{sl} - c_s \dot{\mathbf{x}}_B^{sl} + m_s \mathbf{a}_B, \quad (3.47)$$

where $k_s \in \mathbb{R}$ and $c_s \in \mathbb{R}$ are the stiffness and damping coefficients, which are uncertain but bounded. The estimation of these coefficients have been done for the ESA's spacecraft ATV, which lead to low frequency responses of the system (3.47), as it will be seen shortly. For small satellites the frequency is expected to be higher due to the difference in size and mass [30]. In addition, this will increase the conservativeness of the results given that the satellite model is a second order system, and higher-frequency disturbances would be more attenuated by the system's dynamics. The term $\mathbf{a}_B \in \mathbb{R}^3$ is the acceleration of the spacecraft affecting the tank described in the body frame, which from (3.31) can be related to the inertial acceleration \mathbf{a}_I as

$$\mathbf{a}_B = -[\mathbf{J}_B^{-1} \boldsymbol{\tau}_B^{tot} \times] \mathbf{r}_B^{BP} - 2[\boldsymbol{\omega}_B^{IB} \times] \dot{\mathbf{r}}_B^{BP} - [\boldsymbol{\omega}_B^{IB} \times][\boldsymbol{\omega}_B^{IB} \times] \mathbf{r}_B^{BP} + \mathbf{R}_I^B \mathbf{a}_I, \quad (3.48)$$

with \mathbf{a}_I equal to (3.27).

Also recall from the definition of the reference frames in Sec. 2.1.7, that $\mathbf{r}_B^{BP} \in \mathbb{R}^3$ is the position from the spacecraft's CoM to the propellant's CoM, which can be expressed as $\mathbf{r}_B^{BP} = \mathbf{r}_B^{BS} + \mathbf{R}_S^B \mathbf{x}_S^{sl}$, where $\mathbf{r}_B^{BS} \in \mathbb{R}^3$ is the constant position from the spacecraft's CoM to the sloshing frame—the geometrical center of the tank—and $\mathbf{x}_S^{sl} \triangleq \mathbf{r}_S^{SP}$ is the position of the CoM of the propellant from the sloshing frame. Given that \mathcal{F}_B and \mathcal{F}_S are assumed equally oriented, then $\mathbf{x}_S^{sl} = \mathbf{x}_B^{sl}$ and the expression reduces to $\mathbf{r}_B^{BP} = \mathbf{r}_B^{BS} + \mathbf{x}_B^{sl}$. Then, $\dot{\mathbf{r}}_B^{BP}$ is simply $\dot{\mathbf{x}}_B^{sl}$. The term $\boldsymbol{\tau}_B^{tot}$ is the resultant torque affecting the spacecraft, i.e. $\boldsymbol{\tau}_B^{tot} = \boldsymbol{\tau}_B^{ext} + \boldsymbol{\tau}_B^{MTQ} - \boldsymbol{\tau}_B^{RW}$, and the acceleration \mathbf{a}_I acting on the spacecraft is (3.27), expressed in the inertial frame.

Then, the mass m_s applies a force $\mathbf{f}_B^{sl} \in \mathbb{R}^3$ equal to

$$\mathbf{f}_B^{sl} = k_s \mathbf{x}_B^{sl} + c_s \dot{\mathbf{x}}_B^{sl}, \quad (3.49)$$

according to the mass-spring-damper equation. The CoM of the mass m_s is not centered in CoM of the spacecraft, hence the force \mathbf{f}_B^{sl} also originates a torque $\boldsymbol{\tau}_B^{sl} \in \mathbb{R}^3$ as follows.

$$\boldsymbol{\tau}_B^{sl} = \mathbf{r}_B^{\mathcal{BP}} \times \mathbf{f}_B^{sl}. \quad (3.50)$$

The sloshing dynamics can be linearized and represented in state-space form, with state equation (3.47), and output equations (3.49) and (3.50). The state-space form looks like follows.

$$\begin{bmatrix} \dot{\mathbf{x}}_B^{sl} \\ \ddot{\mathbf{x}}_B^{sl} \end{bmatrix} = \mathbf{A}^{sl} \begin{bmatrix} \mathbf{x}_B^{sl} \\ \dot{\mathbf{x}}_B^{sl} \end{bmatrix} + \mathbf{B}^{sl} \begin{bmatrix} \mathbf{a}_B \\ \mathbf{J}_B^{-1} \boldsymbol{\tau}_B^{tot} \end{bmatrix}, \quad (3.51a)$$

$$\begin{bmatrix} \mathbf{f}_B^{sl} \\ \boldsymbol{\tau}_B^{sl} \end{bmatrix} = \mathbf{C}^{sl} \begin{bmatrix} \mathbf{x}_B^{sl} \\ \dot{\mathbf{x}}_B^{sl} \end{bmatrix}. \quad (3.51b)$$

and the matrices $\mathbf{A}^{sl} \in \mathbb{R}^{6 \times 6}$, $\mathbf{B}^{sl} \in \mathbb{R}^{6 \times 6}$ and $\mathbf{C}^{sl} \in \mathbb{R}^{6 \times 6}$ are explicitly given in [30], and can be found in Appendix B. A simulation has been executed to see the behaviour of these equations and the resultant forces and torques. This can be seen in Fig. 3.2.

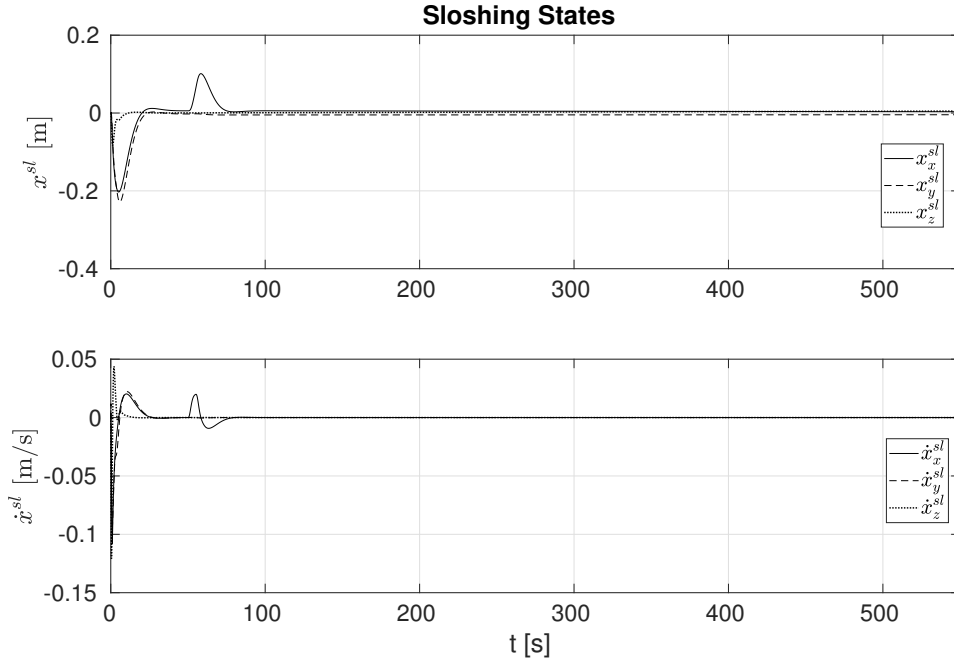


Figure 3.2: Sloshing motion of the propellant in the tank modeled as a mass-spring-damper system.

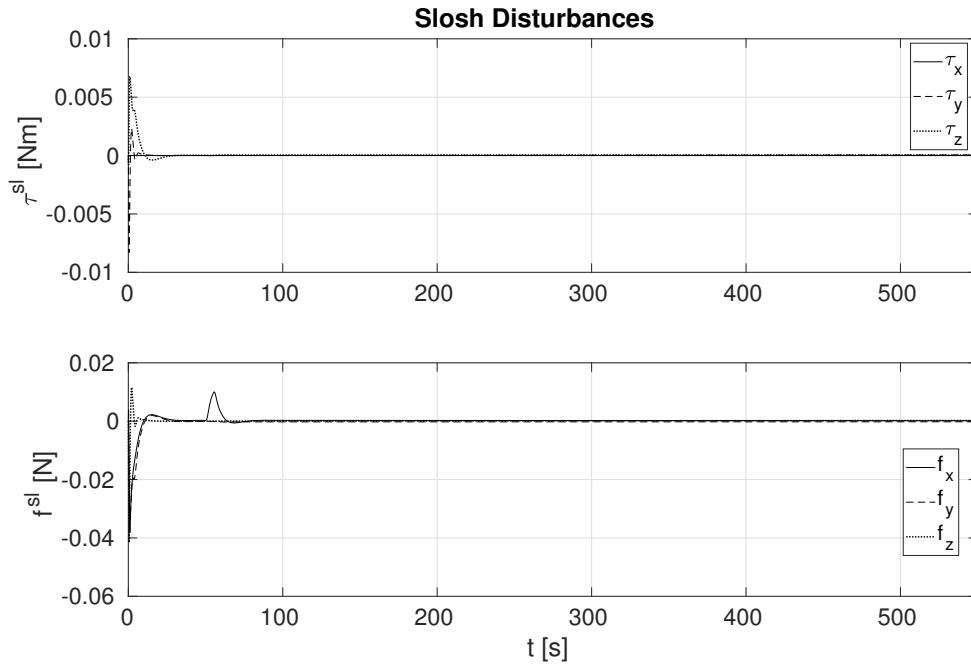


Figure 3.3: Force and torque disturbances originated by the propellant inside the tank assuming a mass-spring-dampen model.

These disturbances can be included both, in the chaser and the target. However, in this work the sloshing disturbance is only included in the chaser. This means that all \mathcal{B} indices must be replaced by \mathcal{C} .

Control System

4

This chapter covers one of the cores of this thesis, i.e. the control system design. A brief review of how the model has been linearized is initially given, which is a necessary step for some of the control strategies addressed. Later, the steps required to formulate the chosen strategies are detailed, being the choices a quaternion-based Sliding-Mode for relative attitude control and MPC for relative position control. Results showing the robustness of the controllers is subsequently provided at the end of each section. In addition, a reaction wheel momentum dumping control is also derived to prevent the RWs from saturation. For these various tasks, the models derived in Sec. 3 are constantly employed in the following sections.

4.1 Linear Model

A nonlinear model for the spacecraft motions have been obtained throughout Chap. 3. If all the variables in the model are stacked in a vector $\mathbf{x} \in \mathbb{R}^{26}$ and the control inputs in a vector $\mathbf{u} \in \mathbb{R}^{15}$, they would look like follow.

$$\mathbf{x} \triangleq [\mathbf{q}^T \quad \boldsymbol{\omega}^T \quad \mathbf{h}_C^{RW\ T} \quad \mathbf{q}_O^{\mathcal{T}\ T} \quad \boldsymbol{\omega}_{\mathcal{T}}^{O\mathcal{T}\ T} \quad \mathbf{h}_{\mathcal{T}}^{RW\ T} \quad \mathbf{r}^T \quad \mathbf{v}^T]^T, \quad (4.1a)$$

$$\mathbf{u} \triangleq [\boldsymbol{\tau}_{\mathcal{T}}^{MTQ\ T} \quad \boldsymbol{\tau}_C^{MTQ\ T} \quad \boldsymbol{\tau}_{\mathcal{T}}^{RW\ T} \quad \boldsymbol{\tau}_C^{RW\ T} \quad \mathbf{f}_C^{th\ T}]^T. \quad (4.1b)$$

Then, the nonlinear model can be shortly expressed as $\dot{\mathbf{x}} = \mathbf{f}(t, \mathbf{x}, \mathbf{u})^1$, where the function $\mathbf{f}(\cdot)$ refers to all equations conforming the model, those are equations (3.21), (3.24), (3.3), (3.17) and (3.35).

Linear and nonlinear control strategies are tackled soon in this chapter. Thus, a linearized version of the equations of motion is needed. In order to approximate the system to a linear model, the Taylor expansion of $\mathbf{f}(t, \mathbf{x}, \mathbf{u})$ about a suitable *operating* point of the system is taken, neglecting quadratic and higher order terms [23]. Lets consider the system moving around the vicinity of some operating point, i.e. $\mathbf{x} = \mathbf{x}_o + \bar{\mathbf{x}}$ and $\mathbf{u} = \mathbf{u}_o + \bar{\mathbf{u}}$, where $\mathbf{x}_o, \mathbf{u}_o \in \mathbb{R}^3$ are the operating points, and $\bar{\mathbf{x}}, \bar{\mathbf{u}} \in \mathbb{R}^3$ small deviations from these points. Then, the nonlinear equations in the proximity of $\mathbf{x}_o, \mathbf{u}_o$ can be approximated as

$$\dot{\mathbf{x}} = \mathbf{f}(t, \mathbf{x}, \mathbf{u}) \approx \mathbf{f}(t, \mathbf{x}_o, \mathbf{u}_o) + \left. \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right|_{(t, \mathbf{x}_o, \mathbf{u}_o)} \bar{\mathbf{x}} + \left. \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \right|_{(t, \mathbf{x}_o, \mathbf{u}_o)} \bar{\mathbf{u}} \quad (4.2)$$

where

$$\mathbf{A} \triangleq \left. \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right|_{(t, \mathbf{x}_o, \mathbf{u}_o)}, \quad \mathbf{B} \triangleq \left. \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \right|_{(t, \mathbf{x}_o, \mathbf{u}_o)} \quad (4.3)$$

are the partial derivatives of $\mathbf{f}(t, \mathbf{x}, \mathbf{u})$ w.r.t. \mathbf{x} and \mathbf{u} respectively —denoted by \mathbf{A} and \mathbf{B} — and evaluated in the operating point. These matrices are the so-called *Jacobian* matrices, or *state* and *input* matrices respectively. Notice that since $\mathbf{x}_o, \mathbf{u}_o$ are constant, it is possible to write $\dot{\mathbf{x}} = \dot{\bar{\mathbf{x}}}$. Moreover, if an equilibrium point of the system, if any, is taken as operation point for the linearization, then $\mathbf{f}(t, \mathbf{x}_o, \mathbf{u}_o) = \mathbf{0}$. If the elements in \mathbf{A} and \mathbf{B} do not depend on time, then the model is said to be linear time-invariant (LTI). Adding an output equation to model the measurements, the system can be represented in continuous state-space form as follows.

$$\dot{\bar{\mathbf{x}}} = \mathbf{A}\bar{\mathbf{x}} + \mathbf{B}\bar{\mathbf{u}} \quad (4.4a)$$

$$\bar{\mathbf{y}} = \mathbf{C}\bar{\mathbf{x}} + \mathbf{D}\bar{\mathbf{u}} \quad (4.4b)$$

¹Note that this is a simplification since it also depend on the dynamics $\dot{\mathbf{x}}$.

Equation (4.4a) is the state equation, with state matrix $\mathbf{A} \in \mathbb{R}^{26 \times 26}$ and input matrix $\mathbf{B} \in \mathbb{R}^{26 \times 9}$ according to dimensions of the state and input vectors. On the other side, (4.4b) is the output equation, which describes the deviation of the measurements taken by the spacecraft and the operational point, and it is represented by $\bar{\mathbf{y}} = \mathbf{y} - \mathbf{x}_o$, being \mathbf{y} the true measurement. Matrix \mathbf{C} , called output matrix, relates the deviations on the states to the output, whereas \mathbf{D} relates the inputs to the output and it is called direct transition matrix. For this application, these matrices are straightforward defined as $\mathbf{C} = \mathbf{I}_{26}$ and $\mathbf{D} = \mathbf{0}_{26 \times 9}$.

Assuming that the target spacecraft can present any arbitrary orientation in space, but it is not rotating or tumbling, the operating point is chosen to be

$$\mathbf{x}_o \triangleq [1 \quad \mathbf{0}_{1 \times 3} \quad \mathbf{0}_{1 \times 3} \quad \mathbf{h}_o \quad \eta_o \quad \boldsymbol{\varepsilon}_o^T \quad \mathbf{0}_{1 \times 3} \quad \mathbf{h}_o \quad \mathbf{0}_{1 \times 3} \quad \mathbf{0}_{1 \times 3}]^T, \quad (4.5a)$$

$$\mathbf{u}_o \triangleq [\mathbf{0}_{1 \times 3} \quad \mathbf{0}_{1 \times 3} \quad \mathbf{0}_{1 \times 3} \quad \mathbf{0}_{1 \times 3} \quad \mathbf{0}_{1 \times 3}]^T. \quad (4.5b)$$

where $\eta_o \in \mathbb{R}$ and $\boldsymbol{\varepsilon}_o \in \mathbb{R}^3$ are the elements of the target quaternion representing any possible operating orientation, and $\mathbf{h}_o \in \mathbb{R}^3$ is the nominal angular momentum of the reaction wheels. Since both spacecrafts are assumed to have the same equipment performance, this nominal value is assumed the same for both.

The linearization is made using the functions provided by the *Symbolic Math Toolbox*TM from *MATLAB*[®]. Once the equations are linearized, the system can be represented in state-space form as an LTI system.

It is important to remark that the coupling in (3.35b) is lost once the Jacobians are evaluated in the operating point. For this precise point choice, the chaser thrusters are considered constantly aligned with the target frame—which is the attitude equilibrium. One must be aware of this fact when utilizing the linearized model.

4.2 Relative Attitude Control

The goal of the relative attitude control system is to maintain the alignment between the chaser's and the target's docking ports, i.e. to force $\mathcal{F}_{\mathcal{D}_c} = \mathcal{F}_{\mathcal{D}_t}$. According to the requirements stated in Sec. 2.4, the attitude pointing accuracy must lie within ± 4 [deg] in SK mode, ± 2 [deg] during approach, and ± 1 [deg] while performing the docking. To achieve these performances, a nonlinear strategy has been chosen which, being redundant, has the advantage of taking the nonlinearities of the system into account. A nonlinear control strategy does not lose accuracy even taking the states of the system far from an ideal operating point and, thus, can be equally used for far- and close-range operations. Among the different nonlinear strategies, Sliding-Mode Control (SMC) is purposed for attitude control.

The motivation behind the selection of this type of controller is described in the following section. Advantages and disadvantages are also discussed, as well as the general procedure for the design of the SMC.

4.2.1 Sliding-Mode Control

The reason of such a choice lies in the robustness against *matched disturbances* and *matched uncertainties*. The term *matched* distinguish those disturbances and uncertainties present on the equations of the model that are directly affected by the control inputs from those that are not, where the term *unmatched* is used instead [31]. For the attitude model, the uncertainties are found in the physical parameters, i.e. the inertia tensors \mathbf{J}_C and \mathbf{J}_T , which are considered matched uncertainties as the control torque τ_C^{RW} appears in the same equation. The external torques acting on the spacecrafts, τ_C^{ext} and τ_T^{ext} , are also considered matched disturbances for the same reason. Given that all the disturbances and uncertainties are *matched*, a SMC for attitude control seems a reasonable choice.

This nonlinear strategy has, although, two major drawbacks. The first of them is the well-known *chattering* effect caused by delays in the system that characterizes the SMC, specially when dealing with big errors. The emergence of this phenomenon will be shown shortly in the following sections. The second drawback is the so-called *reaching phase*, where the system is vulnerable to the matched disturbances and uncertainties.

The general procedure to design a SMC consist of two steps. The first step is to define a suitable manifold, or hyperplane, with the special property that all trajectories lying on it converge to an imposed reference. Since the states are not necessarily lying on the sliding manifold initially, the control law obtained with the SMC must be designed in such a manner that all trajectories are driven to the hyperplane. This is the previously mentioned reaching phase, and conforms the second step in the SMC design. It will be clear that the derived control law is dependent on the manifold chosen. The output of the SMC are, then, the desired control torques to be applied to the chaser in order to align its docking port with that of the target.

4.2.2 Sliding Manifold

The states that are required to be controlled by the SMC are the relative quaternion, \mathbf{q} , and the relative angular velocity, ω . Assume in this section that both, \mathbf{q} and ω , are known or that an estimation of its values is available. In order to align the docking ports while maintaining zero relative motion, the relative quaternion and angular velocity

must reach the desired values $\mathbf{q}_d = [1 \ 0 \ 0 \ 0]^T$ and $\boldsymbol{\omega}_d = [0 \ 0 \ 0]^T$, respectively. Now, the error vectors, $\mathbf{q}_e \triangleq [\eta_e \ \boldsymbol{\varepsilon}_e^T]^T \in \mathbb{H}_1$ and $\boldsymbol{\omega}_e \in \mathbb{R}^3$, can be defined as follows.

$$\mathbf{q}_e = \mathbf{q}_d \otimes \mathbf{q}^{-1}, \quad (4.6a)$$

$$\boldsymbol{\omega}_e = \boldsymbol{\omega}_d - \boldsymbol{\omega}. \quad (4.6b)$$

The 6 dimensional error space, \mathcal{E} , is spanned by the angular velocity error $\boldsymbol{\omega}_e$ and the vector part of the quaternion error \mathbf{q}_e , i.e. vector $\boldsymbol{\varepsilon}_e$, which describes any error state of the system. Thus, $\mathcal{E} = \{\boldsymbol{\omega}_e, \boldsymbol{\varepsilon}_e\}$.

Consider a new variable $\mathbf{s}_C \triangleq [s_C^x \ s_C^y \ s_C^z]^T \in \mathbb{R}^3$, called *sliding variable*, defined as a linear combination of the vectors in \mathcal{E} [37].

$$\mathbf{s}_C \triangleq \boldsymbol{\omega}_e + \mathbf{K}\boldsymbol{\varepsilon}_e, \quad (4.7)$$

with $\mathbf{K} \in \mathbb{R}^{3 \times 3}$ a positive definite matrix. Given that $\boldsymbol{\omega}_e$ and \mathbf{q}_e are time-dependent, the sliding variable is also time-dependent, although it is not represented to relax notation. Notice that despite only using the vector part of the quaternion \mathbf{q}_e , the constraint in (3.2) links the vector to the scalar part.

The vectors of \mathcal{E} which make the sliding variable equal $\mathbf{0}$, are the ones that define the sliding manifold. These vectors generate a subspace, \mathcal{S} , that can be mathematically described as $\mathcal{S} \triangleq \{\boldsymbol{\omega}_e, \boldsymbol{\varepsilon}_e \in \mathcal{E} : \mathbf{s}_C = \mathbf{0}\}$.

In order to prove that trajectories lying in \mathcal{S} slide towards the origin $[\boldsymbol{\omega}_e^T \ \boldsymbol{\varepsilon}_e^T]^T = [\mathbf{0}^T \ \mathbf{0}^T]^T$ through the manifold, a Lyapunov candidate function $V_s : \mathbb{R}^4 \rightarrow \mathbb{R}$ is considered. Since the same rotation can be represented by two different quaternions, a Lyapunov candidate function that ensures that trajectories converge exactly to \mathbf{q}_d is [37]

$$V_s = \boldsymbol{\varepsilon}_e^T \boldsymbol{\varepsilon}_e + (1 - \eta_e)^2, \quad (4.8)$$

which can be rewritten using (3.2) simply as

$$V_s = 2(1 - \eta_e). \quad (4.9)$$

Equation (4.9) is a Lyapunov candidate given that V_s is a positive definite function in all $\mathbb{R}^4 - \{\mathbf{q}_d\}$, and equal to 0 at \mathbf{q}_d . The time-derivative of V_s is then

$$\dot{V}_s = 2\dot{\eta}_e = -\boldsymbol{\varepsilon}_e^T \boldsymbol{\omega}_e, \quad (4.10)$$

according to equations derived in Sec. 3.1. Replacing $\boldsymbol{\omega}_e$ from (4.7) —taking $\mathbf{s}_C = \mathbf{0}$ since trajectories are assumed to be confined in the manifold— to (4.10), leads to

$$\dot{V}_s = -\boldsymbol{\varepsilon}_e^T \mathbf{K} \boldsymbol{\varepsilon}_e, \quad (4.11)$$

which is a negative definite quadratic expression. This ensures that \mathbf{q}_e will exponentially converge exclusively towards \mathbf{q}_d as time increases. Besides, by examining equation (4.7) it can also be concluded that trajectories converging to \mathbf{q}_d in the sliding manifold $\mathbf{s}_C = \mathbf{0}$, will also converge to ω_d , which makes $\{\omega_d, \mathbf{q}_d\}$ an asymptotically stable equilibrium point.

4.2.3 Reachability Condition

This step addresses the derivation of the control law, which will drive trajectories that does not belong to \mathcal{S} directly to the sliding manifold, and slide until reaching the reference. The control law is found by examining the time-derivative of the sliding variable, and forcing that $\mathbf{s}_C = \mathbf{0}$ is an stable equilibrium. Thus, the time-derivative of \mathbf{s}_C is

$$\dot{\mathbf{s}}_C = \dot{\omega}_e + \mathbf{K}\dot{\mathbf{e}}_e, \quad (4.12)$$

where $\dot{\omega}_e = \dot{\omega}_d - \dot{\omega}$, being $\dot{\omega}_d = \mathbf{0}$ the desired angular acceleration. Terms $\dot{\omega}$ and $\dot{\mathbf{e}}_e$ can be obtained directly from the kinematics and dynamics derived in Sec. 3.1. When replaced, the derivative of the sliding variable become

$$\begin{aligned} \dot{\mathbf{s}}_C = & -\mathbf{J}_C^{-1} \left(-[\omega_C^{IC} \times] \mathbf{J}_C \omega_C^{IC} - [\omega_C^{IC} \times] \mathbf{h}_C^{RW} - \tau_C^{RW} + \tau_C^{ext} \right) \\ & - \mathbf{R}_C^C \mathbf{J}_T^{-1} \left([\omega_T^{IT} \times] \mathbf{J}_T \omega_T^{IT} + [\omega_T^{IT} \times] \mathbf{h}_T^{RW} + \tau_T^{RW} - \tau_T^{ext} \right) \\ & - [\omega \times] \mathbf{R}_C^C \omega_T^{OT} + \frac{1}{2} \mathbf{K} (\eta_e \mathbf{I}_3 - [\epsilon_e \times]) \omega_e, \end{aligned} \quad (4.13)$$

where the disturbance torques have been left in the equation since they could be measured or bounded, and thus there is the possibility of feed forwarding them in the control command. The motion of the target, which can be looked as a disturbance, is also left in the equation for the same reason.

In order to bring any trajectory of the spacecraft toward the manifold, the control torque τ_C^{RW} is split into an equivalent torque, $\tau_C^{eq} \in \mathbb{R}^3$, and a sliding-mode control torque, $\tau_C^{smc} \in \mathbb{R}^3$, that is [23]

$$\tau_C^{RW} \triangleq \tau_C^{eq} + \tau_C^{smc}. \quad (4.14)$$

The equivalent torque tries to cancel the nonlinear dynamics to maintain the motion in the sliding manifold, and can be derived from equation (4.13) equation by isolating τ_C^{RW} .

$$\begin{aligned} \tau_C^{eq} = & -[\omega_C^{IC} \times] \mathbf{J}_C \omega_C^{IC} - [\omega_C^{IC} \times] \mathbf{h}_C^{RW} + \tau_C^{ext} \\ & + \mathbf{J}_C \mathbf{R}_C^C \mathbf{J}_T^{-1} \left([\omega_T^{IT} \times] \mathbf{J}_T \omega_T^{IT} + [\omega_T^{IT} \times] \mathbf{h}_T^{RW} + \tau_T^{RW} - \tau_T^{ext} \right) \\ & + \mathbf{J}_C [\omega \times] \mathbf{R}_C^C \omega_T^{OT} - \frac{1}{2} \mathbf{J}_C \mathbf{K} (\eta_e \mathbf{I}_3 - [\epsilon_e \times]) \omega_e. \end{aligned} \quad (4.15)$$

On the other hand, the sliding-mode control torque bring the trajectories towards the hyperplane. It is simply

$$\tau_C^{smc} = -\beta \operatorname{sgn}(\mathbf{s}_C), \quad (4.16)$$

where $\beta \in \mathbb{R}_{>0}$ must be positive definite and is a tuning parameter, and $\operatorname{sgn}(\cdot) : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ is a discontinuous function defined as

$$\operatorname{sgn}(\mathbf{s}_C) \triangleq \begin{cases} \mathbf{0}, & \mathbf{s}_C = \mathbf{0} \\ \frac{\mathbf{s}_C}{\|\mathbf{s}_C\|_2}, & \text{otherwise} \end{cases} \quad (4.17)$$

Notwithstanding, when applying τ_C^{RW} , trajectories will tend to cross the surface $\mathbf{s}_C = \mathbf{0}$ once they reach it, due to inaccuracies in the model and inherent delays in the system. This triggers a sudden sign change in the control input which reverse the direction of the trajectories back to the manifold. Since the delay in the system will not disappear, this process will repeat endlessly leading to the undesired *chattering* or *zig-zag* effect, as shown in Fig. 4.1.

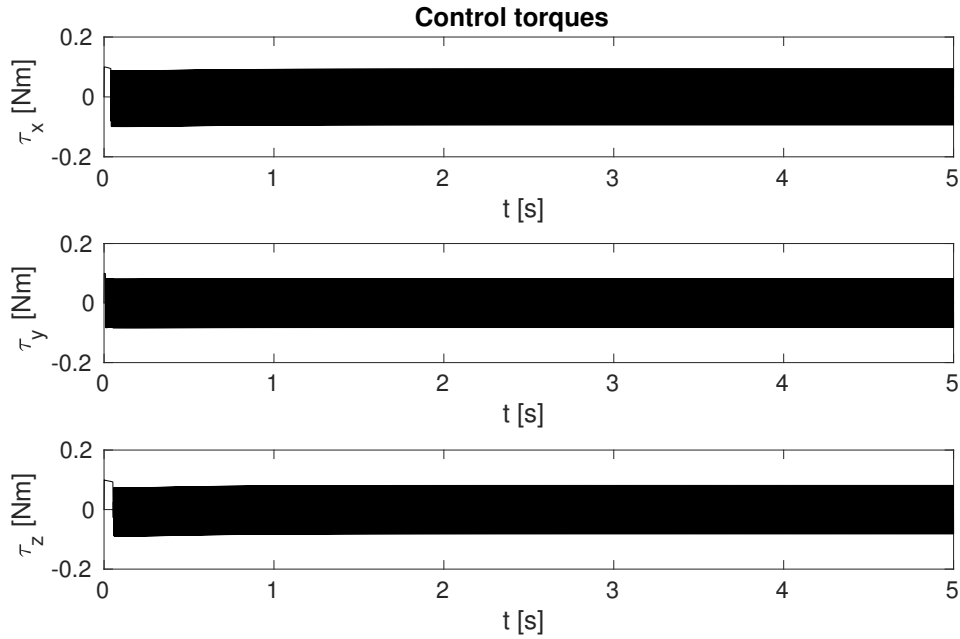


Figure 4.1: Control torques derived using SMC, where the *chattering* effect can be witnessed.

Control inputs with this behaviour are not implementable, since the high-frequency changes can easily damage the actuators and other systems. Besides, if actuator models are not included—as it is the case—the torque commands obtained by the SMC are derived without any consideration of the RWs dynamics, yielding to control torques

that might not be reached in an specific amount of time. This could have severe consequences such as instability of the system or detriment of the performance.

A general procedure to attenuate the *chattering* is to approximate the $\text{sgn}(\cdot)$ by means of a *saturation* function. In this project however, the *sigmoid* function, denoted $\text{sig}(\cdot)$, has been chosen instead. The reason lies in the fact that it is a continuous and continuously differentiable function with a tuning parameter that can approximate the $\text{sgn}(\cdot)$ function without discontinuities. The *sigmoid* function, $\text{sig}(\cdot) : \mathbb{R} \rightarrow \mathbb{R}$, is defined in this thesis as

$$\text{sig}(x) \triangleq \frac{1 - e^{-\lambda x}}{1 + e^{-\lambda x}}, \quad (4.18)$$

where $x \in \mathbb{R}$ represents one component of the state vector, and $\lambda \in \mathbb{R}$ is the tuning parameter with the property that, the largest its value, the closer $\text{sig}(\cdot)$ is to the $\text{sgn}(\cdot)$ function.

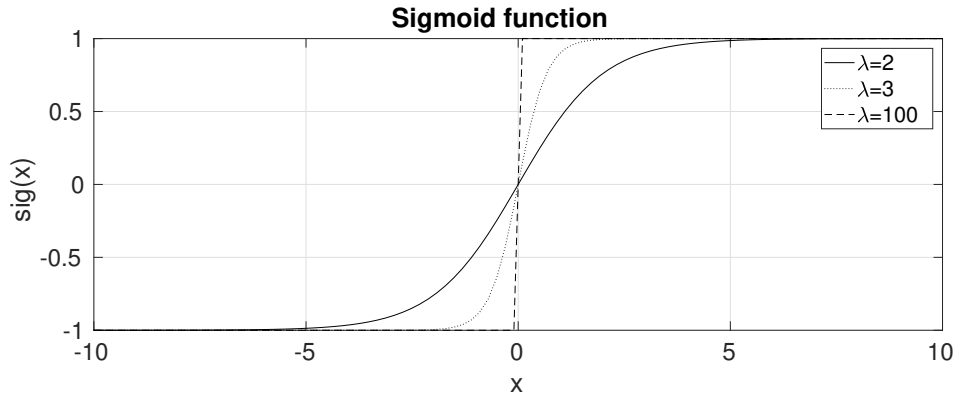


Figure 4.2: Sigmoid function with different tuning parameters, $\lambda = 2, 3, 100$.

Notice that $\text{sig}(\cdot)$ takes exactly one argument. Consider now the mapping function $\sigma(\cdot) : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ defined as follows.

$$\mathbf{s}_C = (s_C^x, s_C^y, s_C^z) \mapsto \sigma(\mathbf{s}_C) = (\text{sig}(s_C^x), \text{sig}(s_C^y), \text{sig}(s_C^z)). \quad (4.19)$$

Then, the sliding mode control τ_C^{smc} is redefined as

$$\tau_C^{smc} = -\beta \sigma(\mathbf{s}_C) \quad (4.20)$$

The torques obtained employing $\sigma(\cdot)$ are shown in the following figure.

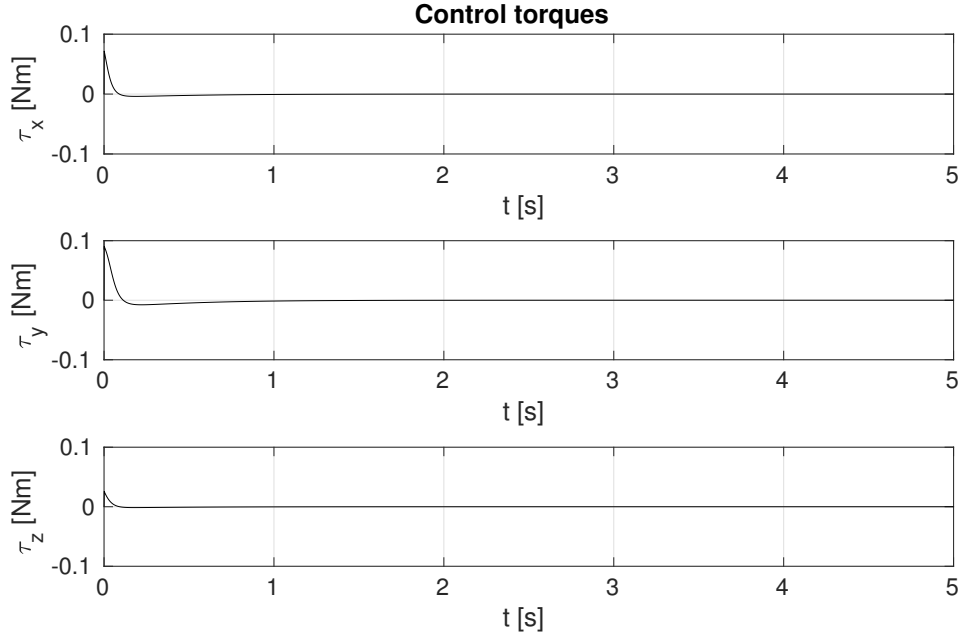


Figure 4.3: Control torques derived using the SMC with sigmoid function $\sigma(\cdot)$ to attenuate the *chattering*.

Another Lyapunov candidate function, $V_r : \mathbb{R}^3 \rightarrow \mathbb{R}$, is employed to check the stability of the reaching phase, being V_r defined as

$$V_r = \frac{1}{2} \mathbf{s}_C^T \mathbf{s}_C, \quad (4.21)$$

which, as before, is positive definite in $\mathbb{R}^3 - \{\mathbf{0}\}$, and 0 at $\mathbf{s}_C = \mathbf{0}$. The time-derivative, \dot{V}_r , is straightforward and has the following form.

$$\dot{V}_r = \mathbf{s}_C^T \dot{\mathbf{s}}_C. \quad (4.22)$$

Replacing equation (4.12) in (4.22), and utilizing the control torques derived in (4.15) and (4.20), the Lyapunov candidate reduces to

$$\dot{V}_r = -\beta \mathbf{s}_C^T \sigma(\mathbf{s}_C) = -\beta [s_C^x \ s_C^y \ s_C^z] \begin{bmatrix} \frac{1-e^{-\lambda s_C^x}}{1+e^{-\lambda s_C^x}} \\ \frac{1-e^{-\lambda s_C^y}}{1+e^{-\lambda s_C^y}} \\ \frac{1-e^{-\lambda s_C^z}}{1+e^{-\lambda s_C^z}} \end{bmatrix}. \quad (4.23)$$

In order to check the sign definiteness of \dot{V}_r , an element-wise inspection is carried out at this point.

$$\dot{V}_r = -\beta \left(s_C^x \frac{1 - e^{-\lambda s_C^x}}{1 + e^{-\lambda s_C^x}} + s_C^y \frac{1 - e^{-\lambda s_C^y}}{1 + e^{-\lambda s_C^y}} + s_C^z \frac{1 - e^{-\lambda s_C^z}}{1 + e^{-\lambda s_C^z}} \right). \quad (4.24)$$

For \dot{V}_r to be negative definite, each of the terms in the parenthesis must be positive definite. Lets consider, for instance, the first term, since conclusions can be extrapolated to the other two.

$$s_C^x \frac{1 - e^{-\lambda s_C^x}}{1 + e^{-\lambda s_C^x}} > 0 \Leftrightarrow s_C^x (1 - e^{-\lambda s_C^x}) > 0, \quad \forall s_C^x \neq 0. \quad (4.25)$$

Since it is clear that equation (4.25) is satisfied for all the terms in (4.24), it can be concluded that \dot{V}_r is negative definite for all $\mathbb{R}^3 - \{\mathbf{0}\}$, and (4.12) is stable. Therefore, the control torque (4.14) fulfills the reachability condition, and drives all trajectories outside \mathcal{S} to it.

Then, by applying the torque (4.14), the quaternion \mathbf{q} and angular velocity $\boldsymbol{\omega}$ are controlled. Fig. 4.4 and Fig. 4.5 show the evolution of these quantities in a 5 second simulation.

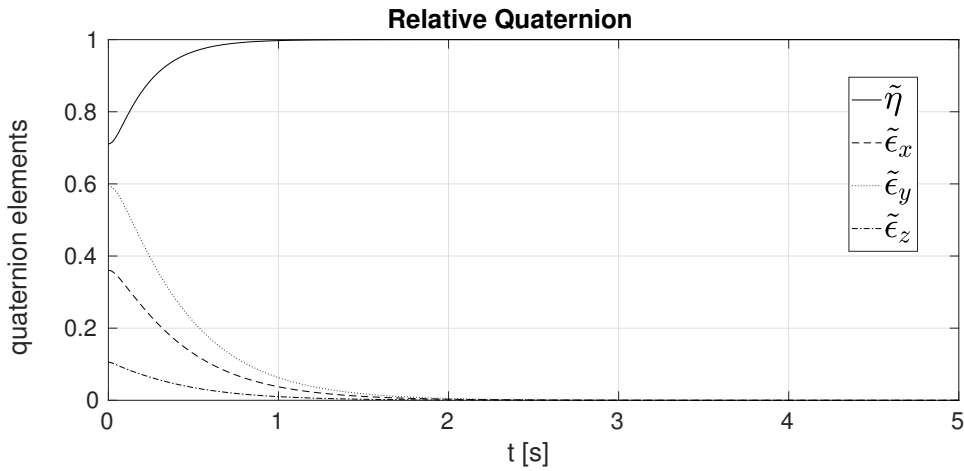


Figure 4.4: Relative quaternion \mathbf{q} .

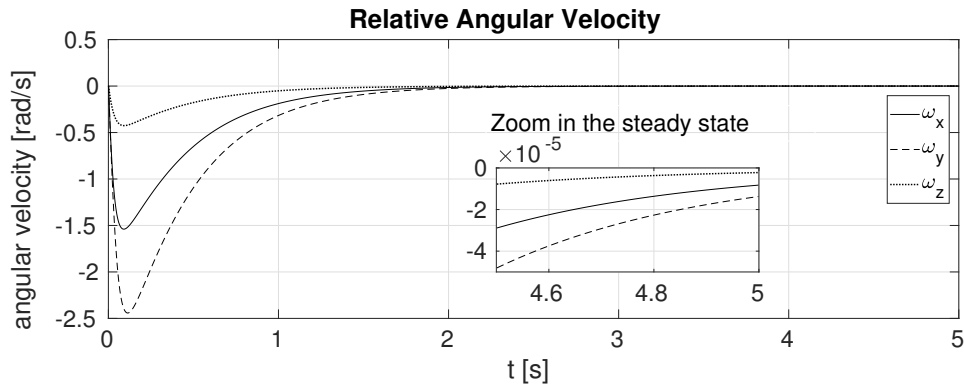


Figure 4.5: Relative angular velocity $\boldsymbol{\omega}$.

Finally, a Monte-Carlo simulation has been performed to prove the robustness of the SMC against disturbances and uncertainties. The initial conditions are the same for every simulation, and the uncertainties have been added in terms of changes in the inertia matrix \mathbf{J}_C , which has been set to vary between -70 to 70%. The results are in Fig. 4.6.

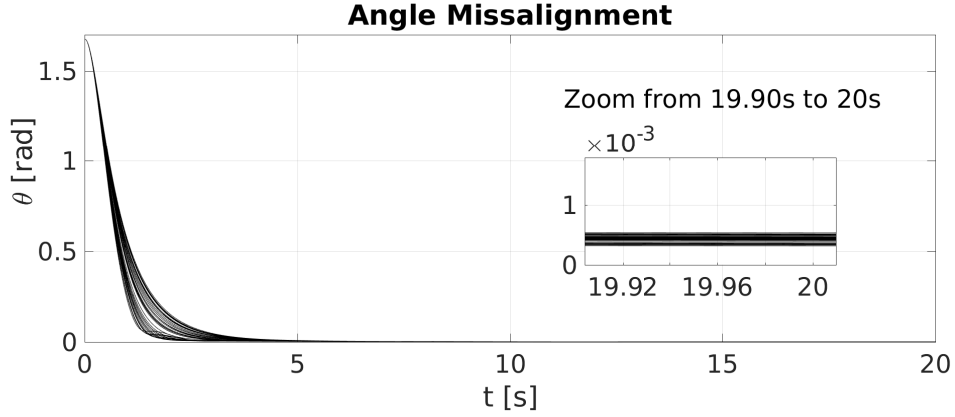


Figure 4.6: Monte-Carlo simulation of the angular misalignment between the docking ports under uncertainties and disturbances. In total, 40 simulations are shown with inertia matrices randomly varying between -70 to 70%.

It is concluded that SMC is robust against the disturbances modeled in Chap. 3, and against uncertainties in the physical parameters involved in the control law.

4.3 RW Momentum Dumping Control

This section addresses the control design for the MTQs. Secular disturbances acting on the spacecraft can cause a monotonously increasing angular momentum of the RW. As a consequence, the angular velocity of the RWs increases, resulting in an excess of momentum that saturates the RWs. In order to avoid reaching the saturation, a control law is presented to dampen the angular momentum of the RWs.

Taking a closer look at the attitude model of a single spacecraft, with only RW as attitude actuation and without disturbances,

$$\frac{d}{dt} \begin{bmatrix} \eta_s \\ \boldsymbol{\varepsilon}_s \\ \boldsymbol{\omega}_B^{IB} \\ \mathbf{h}_B^{RW} \end{bmatrix} = \begin{bmatrix} -\frac{1}{2} \boldsymbol{\varepsilon}_s^T \boldsymbol{\omega}_B^{OB} \\ \frac{1}{2} (\eta_s \mathbf{I}_3 - [\boldsymbol{\varepsilon}_s \times]) \boldsymbol{\omega}_B^{OB} \\ \mathbf{J}_B^{-1} \left(-[\boldsymbol{\omega}_B^{IB} \times] \mathbf{J}_B \boldsymbol{\omega}_B^{IB} - [\boldsymbol{\omega}_B^{IB} \times] \mathbf{h}_B^{RW} - \boldsymbol{\tau}_B^{RW} \right) \\ \boldsymbol{\tau}_B^{RW} \end{bmatrix}, \quad (4.26)$$

it is clear that in order to attain a certain angular velocity, $\omega_{\mathcal{B}}^{I\mathcal{B}}$, a certain torque $\tau_{\mathcal{B}}^{RW}$ needs to be applied. This will directly place the angular momentum of the RWs, $\mathbf{h}_{\mathcal{B}}^{RW}$, in a certain state that can not be chosen. This means that the system is not fully controllable only taking RWs as the actuation system for the attitude and that, in certain situations, some state can become unstable.

In this case, given that $\omega_{\mathcal{B}}^{I\mathcal{B}}$ is desired to be controlled, the angular momentum $\mathbf{h}_{\mathcal{B}}^{RW}$ remains uncontrolled, yielding to the situation described above. Extra actuation is required to make the system controllable, which can be chosen to be MTQs or RCPs. Since fuel is a limiting factor for a CubeSat, MTQs are used in order to dampen the angular momentum of the RW. This is also the most commonly utility of the MTQs, as well as satellite *detumbling*² after orbit injection.

The RW momentum dumping control has a simple design, and is based on the detumbling control design. A magnetic dipole, $\mathbf{m}_{\mathcal{B}} \in \mathbb{R}^3$, is derived using the following relation [5, 15].

$$\mathbf{m}_{\mathcal{B}} = \frac{k}{\|\mathbf{b}_{\mathcal{B}}\|_2} \left([\mathbf{h}_{\mathcal{B}}^{RW} - \mathbf{h}_{ref}] \times \frac{\mathbf{b}_{\mathcal{B}}}{\|\mathbf{b}_{\mathcal{B}}\|_2} \right) = \frac{k}{\|\mathbf{b}_{\mathcal{B}}\|_2} ([\mathbf{h}_{\mathcal{B}}^{RW} - \mathbf{h}_{ref}] \times \widehat{\mathbf{b}}_{\mathcal{B}}). \quad (4.27)$$

In (4.27), $k \in \mathbb{R}_{\geq 0}$ is a positive scalar tuning gain, $\mathbf{b}_{\mathcal{B}} \in \mathbb{R}^3$ is the local geomagnetic field vector expressed in the chaser body frame, $\widehat{\mathbf{b}}_{\mathcal{B}}$ the normalized geomagnetic field vector and \mathbf{h}_{ref} is the nominal angular momentum of the RWs. This control law has been chosen since the angular momentum of the RWs is generally available, although it has not been covered in this thesis.

The control torque, $\tau_{\mathcal{B}}^{MTQ} \in \mathbb{R}^3$, provided by the MTQs is obtained as the interaction between the magnetic dipole generated and the geomagnetic field. It is mathematically expressed as the cross product of both quantities.

$$\tau_{\mathcal{B}}^{MTQ} = \mathbf{m}_{\mathcal{B}} \times \mathbf{b}_{\mathcal{B}} = k ([\mathbf{h}_{\mathcal{B}}^{RW} - \mathbf{h}_{ref}] \times \widehat{\mathbf{b}}_{\mathcal{B}}) \times \widehat{\mathbf{b}}_{\mathcal{B}}. \quad (4.28)$$

It can be seen from the two previous equation that, in the ideal case of having vectors $\mathbf{h}_{\mathcal{B}}^{RW}$ and $\mathbf{b}_{\mathcal{B}}$ completely orthogonal, the control law is designed to apply a torque in the opposite direction of the angular momentum of the RWs. The worse case is, then, when both vectors are parallel, which torque derived is equal to 0. Thus, in the design of the CubeSat it is necessary to ensure that the reaction wheels are not constantly aligned with the geomagnetic field vector.

The same 20 seconds simulation is carried out with and without MTQ actuation to dampen the RW momentum under a constant disturbance. The results are shown

²When inserting a CubeSat in its orbit, it is generally spinning uncontrollably. Magnetorquers are used in this initial phase in order to null the angular velocity vector of the spacecraft.

in Fig. 4.7. The evolution of the quaternion elements is shown first only for the case where momentum dumping control is used —given that convergence of the relative quaternion to the reference only using RW is already shown in Fig. 4.4. A superposition of the magnitude of the RWs' angular momentum for both cases is presented below in order to compare the effect of the RW momentum dumping control. Finally, the magnetic dipoles derived using the control law in (4.27) are presented, which has been left offline during the first 6 seconds. The simulation has been performed with an improvised constant disturbance, which leads to a build up of the RWs' angular momentum.

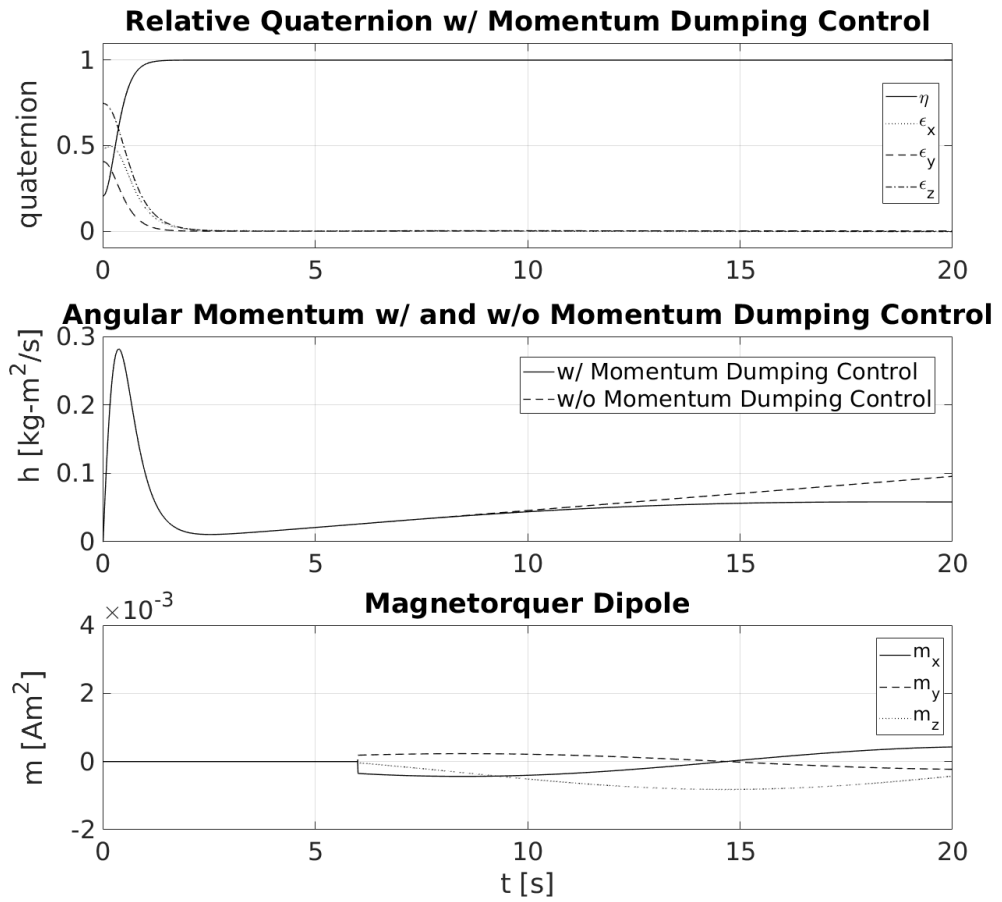


Figure 4.7: Simulation showing the effect of the RW momentum dumping control. On the top, the evolution of the relative quaternion is shown using the attitude control and the RW momentum dumping control. In the middle, the angular momentum of the RWs while applying attitude and momentum dumping control is superposed on top of the angular momentum of the RWs while only applying attitude control. On the bottom, the magnetic dipole generated by the MTQs after a predetermined period of time of 6 s.

It can be seen that the quaternions are not affected by the activation of the momentum

dumping control, and the angular momentum of the RWs start decreasing after initializing the controller at second 6 of the simulation. The torques provided by the RWs have not been saturated in the simulation in order to see the effects clearly.

4.4 Relative Position Control

The relative position control is designed to maintain the position in the SK points, and to make the transition between them until contact of the docking ports is made. Unlike the RWs or the MTQs, which can be recharged using solar arrays, the amount of propellant is a limiting factor for the RCP system. Once the servicing spacecraft runs out of propellant, the RVD procedure can not be accomplished and the target's and other spacecrafts' safety can get endangered. Optimal strategies are then sought in order to reduce as much as possible the use of propellant during the operation, and *back-up* plans should be available at any time for undesired situations, such as thruster failure or running out of propellant.

Moreover, when approaching another object in-orbit extra care must be taken in order to avoid collision, which means that boundaries should be imposed to the motion — trajectory and velocity vectors— of the chaser to perform a safe RVD. This has been tackled through diverse methods based on e.g. pseudospectral methods [8] or artificial potential functions [16] among others, the former being widely utilized in combination with other control methods, such as adaptive time-varying control laws, to derive safe paths free of collisions.

Another extensively used method is the MPC approach, which is known as an optimal control strategy capable of handling constraints online. The optimization problem is based on minimizing a cost, or objective, function that can be directly related to the specifications or goals of the control system while satisfying some predefined constraints. According to these two facts, it is clearly a suitable strategy for the purpose of the control tasks detailed in this section. Given that unfeasible solutions can be reached —when the constraints can not be fulfilled in the next sampling time for any possible control command— extra *back-up* strategies should be added to the aforementioned ones.

It is worth it to notice that the usual *back-ups* employed during far-range and part of close-range rendezvous, are not available in this last stage of the mission since the chaser is imposed to follow a forced trajectory. Thus, some *back-ups* can be thought as control inputs for the chaser to stop the relative motion, or as action commands sent to the target via ISC to increase the relative distance if the chaser can not reduce the motion on time. Although, implementation of these strategies are beyond the scope of this project.

4.4.1 Model Predictive Control

Model Predictive Control (MPC) is an advanced *optimal*³ control strategy which control law is based on three keys

- a Prediction Model,
- a Cost Function and
- Constraints.

The prediction model is a mathematical description of the system, e.g. by means of the differential equations $\dot{\mathbf{x}} = \mathbf{f}(t, \mathbf{x}, \mathbf{u})$ derived in Sec. 3. This can be used to propagate the states in time if the input \mathbf{u} is known. Despite most of the models behaving in some nonlinear fashion, as it is the case of this thesis, simplifications of the model can be achieved by linearizing $\mathbf{f}(t, \mathbf{x}, \mathbf{u})$ around an equilibrium or an operating point, as it has also been done in Sec. 4.1. This leads to an LTI model, or to a linear time-varying (LTV) model if the operating point changes with time. One must be careful when applying this simplification, since control commands are being derived using the linear simplified model and subsequently applying them to the real system, which behaviour might differ greatly. The prediction model can further be treated by discretizing $\mathbf{f}(t, \mathbf{x}, \mathbf{u})$, or its linearized version, with a determined sampling time $T_s = \Delta t$ where the inputs are assumed constant. This yields to a discrete model $\mathbf{f}_d(k, \mathbf{x}_k, \mathbf{u}_k)$, where \mathbf{x} and \mathbf{u} are now functions of the discrete time $k = nT_s$, with $n \in \mathbb{N}$.

The cost function, on the other hand, is a mathematical function that generally depends on the controlled states, the inputs or both. It is used to penalize the deviation of the future predictions of these elements from their desired values. This means that time-domain objectives can be directly specified in the controller. Besides, the solution of the minimization of this function provides an optimal input command that can be applied to the system. The minimization problem is, in general, solved numerically until a minimum is reached, although, the minimum reached is not guaranteed to be a global minimum for an arbitrary cost function. A way to avoid being trapped in a local minimum is to solve a *convex* optimization problem instead, which is based on minimizing a convex cost function with a convex set. The convexity property ensures that there is only one global minimum—or a connected set of equal minima—and that it will eventually be reached in a finite time. Quadratic cost functions are the most commonly used convex functions, and solving this optimization problem is referred to as Quadratic Programming (QP). Linear Programming (LP) is a special case, where the quadratic cost function is replaced by a linear cost function. If the optimization problem is subject to linear inequalities as constraints, then the LP is also convex. Despite LP being faster to solve than QP, LP has the major drawback of always finding

³Emphasis is given to the term *optimal*, as no optimal control strategy exist in a strict sense due to the tuning parameters involved.

the global optimum on the constraints [26]. This is highly undesired, as it would mean placing the chaser constantly on the constraints and the limits of the control requirements. Even if the uncertainties and inaccuracies of the model were sufficiently handled, a minimum disturbance would take the motion of the chaser out of the control requirements, deteriorating greatly the performance of the GNC system. A quadratic cost function is, therefore, chosen for the MPC formulation introduced shortly.

Finally, constraints can be introduced to the MPC problem generally in the form of *linear inequalities*, which is an important fact to efficiently solve the constrained optimization problem. When solving the problem all the constraints are rewritten as a function of the control inputs, and as an LTI model is assumed, the rewritten constraints still linear and the solution of the constrained optimization problem reduces to a product of big matrices [26]. A variety of tools, for instance the *Shur complement* or the *S-procedure*, come into handy when dealing with specific nonlinear constraints. These linear inequalities are eventually expressed as linear matrix inequalities (LMIs), which are extensively used in the literature as the problem becomes reliably and efficiently solvable. Nevertheless, it is shortly seen that most of the constraints introduced have a particular nonlinear form, which are not as efficient as assuming LMIs, but still are solvable. Lastly, it is interesting to discuss two attributes regarding the constraint optimization problem. The first is that the control law derived using MPC behaves in some nonlinear fashion, despite assuming the model of the system to be LTI. These nonlinearities are, then, introduced by the presence of constraints in the optimization problem. The second, and most important, is that the optimization problem might become infeasible when adding constraints. This is a highly undesired situation for an online controller, and it is then when *back-up* strategies could come into play.

In the following section, the MPC formulation taken in this thesis is extended, introducing the new notation needed, and defining mathematically the cost function and the constraints utilized.

4.4.2 MPC Formulation

The basic formulation of MPC is firstly sought, where it is assumed a discrete LTI prediction model, a quadratic cost function and linear inequalities as constraints [26].

The LTI model is obtained in Sec. 4.1, where it is remarked that the rotational and translational models are decoupled after evaluating the Jacobians in the operating points. Thus, since \mathbf{r} and \mathbf{v} are the controlled states for the MPC, the vector \mathbf{x} is exclusively used only during this section to denote $\mathbf{x} = [\mathbf{r}^T \mathbf{v}^T]^T$, and then $\mathbf{u} = \mathbf{f}_C^{th}$. According to this, the linear model is discretized assuming zero-order hold, and the

following discrete LTI system is achieved.

$$\mathbf{x}_{k+1} = \mathbf{\Phi}\mathbf{x}_k + \mathbf{\Gamma}\mathbf{u}_k, \quad (4.29)$$

where $\mathbf{x}_k \triangleq [\mathbf{r}_k^T \mathbf{v}_k^T]^T$ and \mathbf{u}_k are the state vector and the control input at discrete time k respectively, and the matrices $\mathbf{\Phi} \in \mathbb{R}^{6 \times 6}$ and $\mathbf{\Gamma} \in \mathbb{R}^{6 \times 3}$ are obtained as

$$\mathbf{\Phi} = e^{\tilde{\mathbf{A}}T_s}, \quad (4.30a)$$

$$\mathbf{\Gamma} = \left(\int_0^{T_s} e^{\tilde{\mathbf{A}}\tau} d\tau \right) \tilde{\mathbf{B}}, \quad (4.30b)$$

being $\tilde{\mathbf{A}}$ the last 6×6 submatrix of \mathbf{A} , which only involves the state variables \mathbf{r} and \mathbf{v} , and $\tilde{\mathbf{B}}$ the last 3 columns of \mathbf{B} as it is the only part multiplying \mathbf{f}_C^{th} .

Now the model (4.29) can be used to make future predictions of the states. The estimates of the states at the present time, $\hat{\mathbf{x}}_{k|k}$, are assumed to be known throughout the design of the controller. Following the same notation, future predictions are derived using the model, which are denoted as $\hat{\mathbf{x}}_{k+i|k}$ for the prediction made at time k of the states at time $k+i$. Vector $\hat{\mathbf{x}}_{k|k}$ is thus propagated for $i \in \{1, 2, \dots, H_p\}$, being $H_p \in \mathbb{N}$ the finite prediction horizon, defined as a tuning parameter. Accordingly, $\hat{\mathbf{u}}_{k+j|k}$ with $j \in \{0, 1, \dots, H_p - 1\}$ is the input control derived at time k for the time step $k+j$. Once the whole propagation of the states is achieved, and the minimization problem solved, a sequence of optimal input vectors $\hat{\mathbf{u}}_{k+j|k}^*$ is obtained. However, only the first of them, namely $\hat{\mathbf{u}}_{k|k}^*$, is applied to the system. Then, a new estimate $\hat{\mathbf{x}}_{k+1|k+1}$ can be derived from the measurements, and the MPC problem is restarted. It is obvious, then, that the estimation of $\hat{\mathbf{x}}_{k|k}$ directly impacts on the calculation of $\hat{\mathbf{u}}_{k|k}^*$ and thus the performance of the controller, since the inaccuracies of the estimation are dragged through the predictions.

The cost function is defined, as previously mentioned, to penalize the deviation of the predicted controlled states from a desired predefined trajectory, denoted as $\mathbf{t}_k \in \mathbb{R}^6$. How the trajectories are defined is discussed in Sec. 4.4.3.

All the predictions of the states up to H_p can be stacked more conveniently in a vector, $\mathcal{X}_k \triangleq [\hat{\mathbf{x}}_{k+1|k}^T \hat{\mathbf{x}}_{k+2|k}^T \dots \hat{\mathbf{x}}_{k+H_p|k}^T]^T \in \mathbb{R}^{6H_p \times 1}$, as well as the reference vectors at each time step $\mathcal{T}_k \triangleq [\mathbf{t}_{k+1}^T \mathbf{t}_{k+2}^T \dots \mathbf{t}_{k+H_p}^T]^T \in \mathbb{R}^{6H_p}$ and the derived input vectors $\mathcal{U}_k \triangleq [\hat{\mathbf{u}}_{k|k}^T \hat{\mathbf{u}}_{k+1|k}^T \dots \hat{\mathbf{u}}_{k+H_p-1|k}^T]^T \in \mathbb{R}^{3H_p}$. Instead of penalizing the values of \mathcal{X}_k itself, which would be done in a regulating problem, the error between the states and the references is taken. Therefore, a new variable $\mathcal{E}_k \in \mathbb{R}^{6H_p}$ representing the error committed in each future prediction can be defined as $\mathcal{E}_k \triangleq \mathcal{X}_k - \mathcal{T}_k$. The quadratic cost function, denoted as $\mathcal{V}_k : \mathbb{R}^{6H_p} \rightarrow \mathbb{R}$, can finally be written as

$$\mathcal{V}_k = \mathcal{E}_k^T \mathbf{Q} \mathcal{E}_k + \mathcal{U}_k^T \mathbf{R} \mathcal{U}_k, \quad (4.31)$$

where $\mathbf{Q} \triangleq \text{diag}(\mathbf{Q}_1, \mathbf{Q}_2, \dots, \mathbf{Q}_{H_p})$ and $\mathbf{R} \triangleq \text{diag}(\mathbf{R}_0, \mathbf{R}_1, \dots, \mathbf{R}_{H_p-1})$ are fixed block diagonal matrices. Each submatrix $\mathbf{Q}_i \in \mathbb{R}_{>0}^{6 \times 6}$ and $\mathbf{R}_i \in \mathbb{R}_{\geq 0}^{3 \times 3}$ are weights used to penalize the error states and the input vector at each time step respectively. Generally, $\mathbf{R}_0 = \mathbf{R}_1 = \dots = \mathbf{R}_{H_p-1}$, which treats the input at each prediction equally. On the other hand, $\mathbf{Q}_1 = \mathbf{Q}_2 = \dots = \mathbf{Q}_{H_p-1}$, giving special attention to the last state by usually increasing \mathbf{Q}_{H_p} w.r.t. the previous ones. This is often done to enforce the stability of the system. Conclusively, \mathcal{V}_k is the sum of all the penalizations on the deviations of the controlled states and the input commands over all the prediction horizon, and the objective of the MPC controller is to minimize its value.

At each optimization step, it is also required that some state variables, and possibly the control input, satisfy a set of constraints. These constraints emerge from the control requirements and the actuators specification, and must be constantly fulfilled, since not doing so could imply catastrophic situations. As it has been pointed out previously, the constraints introduced hereafter would be more conveniently defined as LMIs. However, it will be shown shortly that it is not the case for most of them, but that it is not a major concern for the solvability of the constrained optimization problem. The expressions for the constraints for all the prediction horizon are derived, as it can be of help to solve the optimization problem manually. Although, the analytical solution is not found, instead, the *YALMIPTM Toolbox* for Modelling and Optimization is used to find the solutions numerically in *MATLAB[®]*.

Consider first the constant constraints imposed to the RCP system in Sec. 1.2, which bounds the thrust provided by the thrusters in each face of the CubeSat. If a vector $\mathbf{f}_{\max} = [f_{\max}^x \ f_{\max}^y \ f_{\max}^z]^T$ is used to describe the maximum thrust that can be provided by the RCP in each direction —being f_{\max}^i the maximum thrust allowed for the thruster pointing in the direction i . Given that thrusters are placed in each face of the CubeSat, three of them provide acceleration in the positive directions of x , y and z ; and three others provide acceleration in the negative directions of the same axis. Subsequently, the LMIs for the control input can be expressed for $j \in \{0, \dots, H_p - 1\}$ as

$$\hat{\mathbf{u}}_{k+j|k} \leq \mathbf{f}_{\max}, \quad (4.32a)$$

$$\hat{\mathbf{u}}_{k+j|k} \geq -\mathbf{f}_{\max}, \quad (4.32b)$$

being (4.32a) the linear inequality constraint for the three thrusters providing positive acceleration, and hence, (4.32b) the linear inequality constraint for the three thrusters providing negative acceleration.

To define the constraints for the motion, attention must be given to the requirements in Sec. 2.4. First, the constraint regarding the relative velocity is addressed. For safety reasons, the approach is intended to be effected at a low constant velocity, equal to 0.01 [m/s]. Thus, it is sought to bound the magnitude of the relative velocity at each time step. Since this is tackled in the design of the tracking reference, Sec. ??, by drawing the desired velocity profile to be tracked, a velocity constraint can be defined to bound

possible divergences from the reference. At an arbitrary prediction step $k + i|k$, this can be expressed as

$$\hat{\mathbf{v}}_{k+i|k}^T \hat{\mathbf{v}}_{k+i|k} \leq 0.02, \quad (4.33)$$

for $i \in \{1, \dots, H_p\}$. Notice that the constraints in (4.34) have a quadratic form, and are not defined as linear inequalities. When a QP is constrained with quadratic inequalities, in literature this receives the name of Quadratically Constrained Quadratic Programming (QCQP). Despite requiring more computational effort, these constraints can still be handled and the problem solved by transforming the quadratic constraints into second order cone (SOC) constraint, which is also convex and solvable. The addition of SOC constraints turns the problem into a Second Order Cone Programming (SOCP). Current solvers are capable of automatically recognize quadratic constraints and transforming them to SOCs. An alternative to avoid this is to constraint only the approaching velocity, that is

$$[1 \ 0 \ 0] \hat{\mathbf{v}}_{k+i|k} \leq 0.02, \quad (4.34)$$

Other pure SOC constraints rise from the relative position requirements, which need to be within 5% of the range during motion. This defines a cone which axis is aligned with the x -axis of the target's docking port reference frame, and is often called line of sight (LoS) constraint.

$$\hat{\mathbf{r}}_{k+i|k}^T \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \hat{\mathbf{r}}_{k+i|k} \leq 0.05 [1 \ 0 \ 0] \hat{\mathbf{r}}_{k+i|k}, \quad (4.35)$$

Lastly, an extra constraint for *soft-docking* is added for the last approach maneuver—from 2.5 [m] until docking. This constraint is introduced in [11], which utilizes the 1-norm of the position to bound the 1-norm of the velocity, so that the velocity decreases as the chaser approaches the docking port. This helps reducing the shock when dock happens and increasing the safeness of the docking procedure, since low velocities are involved. Using the 1-norm allows to define the constraints as linear inequalities, which is always desired for time efficiency. However, the set of feasible solutions become either more conservative or includes unwanted regions of solutions and further treatment is needed. Since QCQP are solvable, and this thesis is not focused on the computational effort and time required to solve the optimization problem, the 2-norm is used instead as it is more straightforward to implement and the magnitude of the distance can directly be related to the magnitude of the velocity. This constraint can be expressed for an arbitrary step time as

$$\hat{\mathbf{v}}_{k+i|k}^T \hat{\mathbf{v}}_{k+i|k} \leq \left(\frac{\hat{\mathbf{r}}_{k+i|k}^T \hat{\mathbf{r}}_{k+i|k} - \alpha}{\gamma} \right), \quad (4.36)$$

where $\alpha \in \mathbb{R}$ and $\gamma \in \mathbb{R}_{>0}$ are tuning parameters used to define the shape of the feasible solutions. Again, a simpler alternative to the previous constraint is to relate only the x component.

Equation (4.36) is considered a hard constraint, same as the previous ones, and increases the chances of reaching infeasible solutions [11]. Given that it is not considered a strict constraint to fulfill, since small violations would not put the mission on danger, the constraint is relaxed by means of a *slack* variable as shown below.

$$\hat{\mathbf{v}}_{k+i|k}^T \hat{\mathbf{v}}_{k+i|k} \leq \left(\frac{\hat{\mathbf{r}}_{k+i|k}^T \hat{\mathbf{r}}_{k+i|k} - \alpha}{\gamma} \right) + \sigma_{k+i|k}^{slack}, \quad (4.37a)$$

$$\sigma_{k+i|k}^{slack} \geq 0 \quad (4.37b)$$

where $\sigma_{k+i|k}^{slack} \in \mathbb{R}_{\geq 0}$ is the *slack* variable used to relax the constraint, making (4.37) a *soft*-constraint. The value of $\sigma_{k+i|k}^{slack}$ can be chosen to vary or to be constant throughout the whole prediction horizon, associating different meanings to the variable $\sigma_{k+i|k}^{slack}$.

Assuming $\sigma_{k+i|k}^{slack}$ as a time-varying variable, it can be used to penalize the violation committed at each time step k , if any appear to happen. This would, thus, add H_p new variables to the optimization problem, and now it would be solved for \mathbf{U}_k and a new vector $\mathbf{S}_k = [\sigma_{k+1|k}^{slack} \ \sigma_{k+2|k}^{slack} \ \dots \ \sigma_{k+H_p|k}^{slack}]^T$, being each $\sigma_{k+i|k}^{slack}$ for $i = 1, \dots, H_p$ different. Therefore, an extra penalization term depending on \mathbf{S}_k should be added to the cost function, otherwise $\sigma_{k+i|k}^{slack}$ could grow uncontrollably making the constraint useless. In case it is necessary for the solution of the problem to violate the hard constraint (4.36) at time $k + i|k$, then the slack variable $\sigma_{k+i|k}^{slack} \geq 0$, contributing to the cost function \mathcal{V}_k at that time step. This can be done, for instance, using the 1-norm of the slack vector as follows.

$$\mathcal{V}_k = \mathbf{E}_k^T \mathbf{Q} \mathbf{E}_k + \mathbf{U}_k^T \mathbf{R} \mathbf{U}_k + \varrho \|\mathbf{S}_k\|_1, \quad (4.38)$$

where $\varrho \in \mathbb{R}_{>0}$ is a strictly positive tuning value.

On the other hand, $\sigma_{k+i|k}^{slack}$ can also be assumed constant, and equal to the ∞ -norm of the slack vector, i.e. $\sigma_{k+i|k}^{slack} = \|\mathbf{S}_k\|_\infty$, penalizing its value in the cost function. By taking this approach, the maximum error committed is minimized, bounding the violations that might occur during the prediction horizon by the value of $\|\mathbf{S}_k\|_\infty$, for $i = 1, \dots, H_p$. This has the advantage that only one extra variable is added to the optimization problem, and therefore does not increase the computation effort as much as assuming a varying slack variable. The cost function, assuming a constant slack variable, has the following appearance.

$$\mathcal{V}_k = \mathbf{E}_k^T \mathbf{Q} \mathbf{E}_k + \mathbf{U}_k^T \mathbf{R} \mathbf{U}_k + \varrho \|\mathbf{S}_k\|_\infty, \quad (4.39)$$

As a remark, it is worth it to mention that either approach reach the optimal value for the slack variable for a big enough, value of ϱ [26].

Gathering all the constraints and cost functions defined in this section, a guideline describing the different optimization problems that are needed to be solved at each time

is proposed hereafter. It indicates, according to the state of the RVD procedure, what is the cost function to be minimized and the constraints considered. Not all constraints derived previously will be considered at every time, thus the cost function will vary according to that. The selection of the optimization problem—in other words, the selection of the cost function and constraints—that is needed to be solved at each step is done depending on the range.

The objective of the MPC controller is, then, to solve the corresponding optimization problem described in the following table at each time step k .

| Range [m] | Optimization Problem |
|-----------|--|
| | $\min_{\mathbf{u}_k} \mathcal{V}_k = \mathcal{E}_k^T \mathbf{Q} \mathcal{E}_k + \mathbf{u}_k^T \mathbf{R} \mathbf{u}_k$ |
| | s.t. |
| 10 - 2.5 | $\begin{aligned} \hat{\mathbf{u}}_{k+j k} &\leq \mathbf{f}_{max} & j = 0, \dots, H_p - 1 \\ \hat{\mathbf{u}}_{k+j k} &\geq -\mathbf{f}_{max} \\ \hat{r}_{Y, k+i k}^2 + \hat{r}_{Z, k+i k}^2 &\leq 0.05 \hat{r}_{X, k+i k} & i = 1, \dots, H_p \\ \hat{\mathbf{v}}_{k+i k}^T \hat{\mathbf{v}}_{k+i k} &\leq 0.02 \end{aligned}$ |
| | |
| | $\min_{\mathbf{u}_k, \ \mathbf{S}_k\ _\infty} \mathcal{V}_k = \mathcal{E}_k^T \mathbf{Q} \mathcal{E}_k + \mathbf{u}_k^T \mathbf{R} \mathbf{u}_k + \varrho \ \mathbf{S}_k\ _\infty$ |
| | s.t. |
| 2.5 - 0 | $\begin{aligned} \hat{\mathbf{u}}_{k+j k} &\leq \mathbf{f}_{max} & j = 0, \dots, H_p - 1 \\ \hat{\mathbf{u}}_{k+j k} &\geq -\mathbf{f}_{max} \\ \hat{r}_{Y, k+i k}^2 + \hat{r}_{Z, k+i k}^2 &\leq 0.05 \hat{r}_{X, k+i k} & i = 1, \dots, H_p \\ \hat{\mathbf{v}}_{k+i k}^T \hat{\mathbf{v}}_{k+i k} &\leq (\hat{\mathbf{r}}_{k+i k}^T \hat{\mathbf{r}}_{k+i k} - \alpha) / \gamma + \ \mathbf{S}_k\ _\infty \\ \ \mathbf{S}_k\ _\infty &\geq 0 \end{aligned}$ |

Table 4.1: MPC guideline. It describes different optimization problems and indicates which is required to be solved depending on the range.

The reference signal tracked at each step is defined in the next section.

Some results are shown hereafter, where it can be seen that the MPC performs well when low or no uncertainties are added. However, the problem becomes unfeasible, or even unstable, when larger uncertainties are introduced. The following simulations adds uncertainties in the physical parameters which ranging from -70% to 70%.

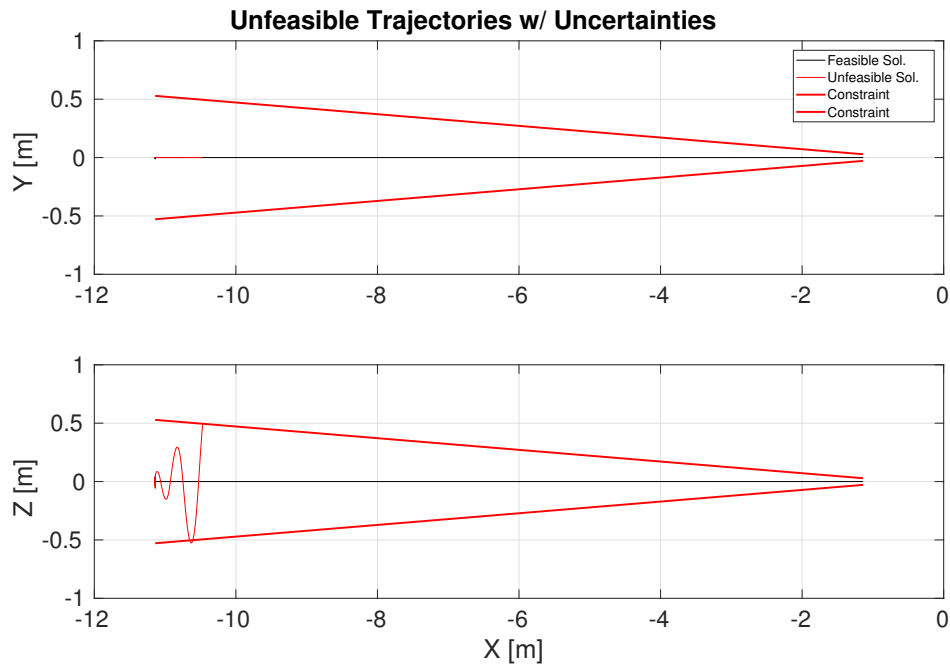


Figure 4.8: Position trajectories under uncertainties in the mass of the chaser ranging from -70% to 70%. Some solutions become unfeasible. The control requirements are denoted by the thick red cone.

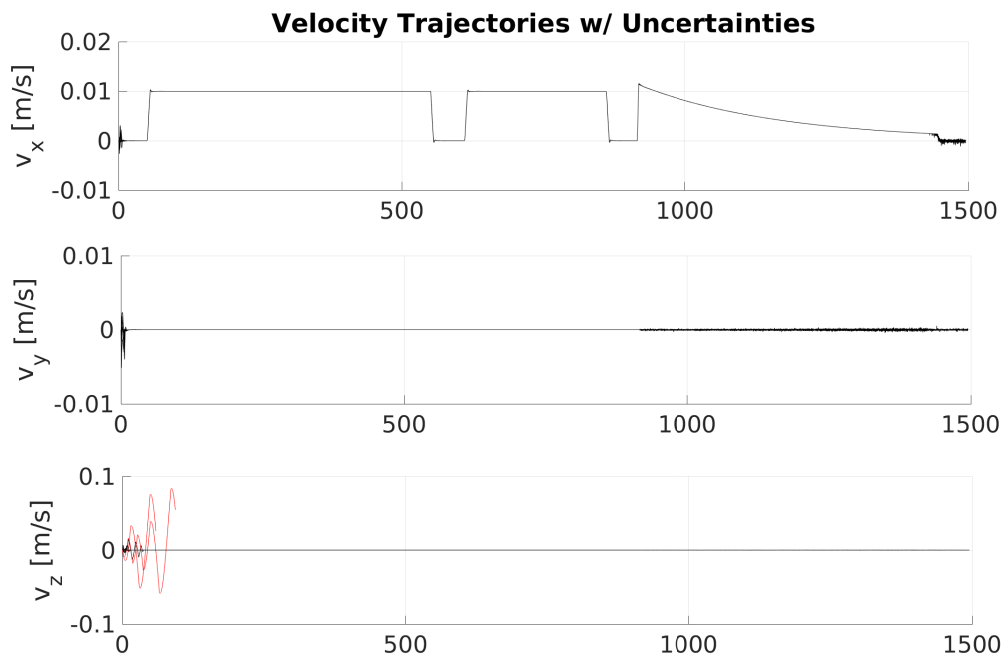


Figure 4.9: Velocity trajectories under uncertainties in the mass of the chaser ranging from -70% to 70%. Some solutions become unfeasible.

In spite of this, it can be seen from Fig. 4.9 that the velocity profile is perfectly tracked, and that the soft-docking constraint is successfully implemented, continuously decreasing the approaching velocity according to the range left until dock is completed.

The docking constraints are, then, only fulfilled when the MPC problem does not become unfeasible or unstable. However, 100% of those times the constraints are fulfilled. This can be seen in Fig. 4.10

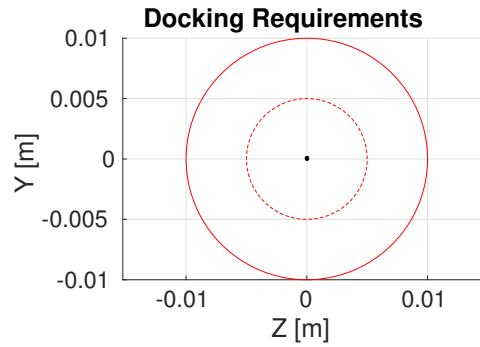


Figure 4.10: Docking missalignments. The red continuous circle represent the margin that the docking port can physically handle. The red dashed circle represent the control requirements to be fulfilled.

Finally, the 3D trajectories w.r.t. the orbital frame can be observed in Fig. 4.11. For these results, 10 simulations were performed, from which 2 became unfeasible.

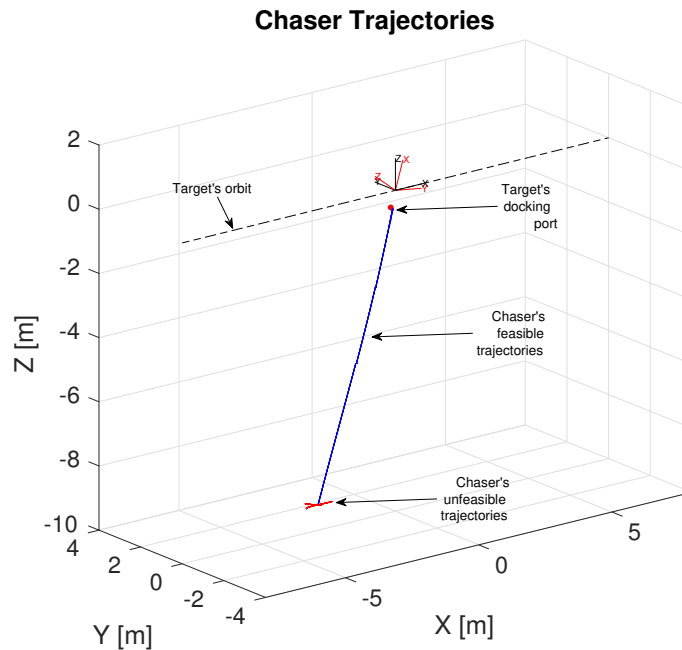


Figure 4.11: 3D trajectories of the chaser towards the docking port.

The ΔV maneuvers during the simulations have been summarized in the following table.

| $\overline{\Delta V}$ | $Var(\Delta V)$ |
|-----------------------|-----------------|
| 0.4642 [m/s] | 0.0079 [m/s] |

Table 4.2: Mean and variance of the ΔV manoeuvres performed during the simulations.

4.4.3 Motion Profiles

A guidance profile is provided in this section, which describes the tracking references, \mathbf{t}_k , given to the MPC controller. Velocity and position profiles of the chaser's CoM are separately derived for the first two transition between SK points. From SK_0 to SK_1 , the tracking references are shown in the figure below.

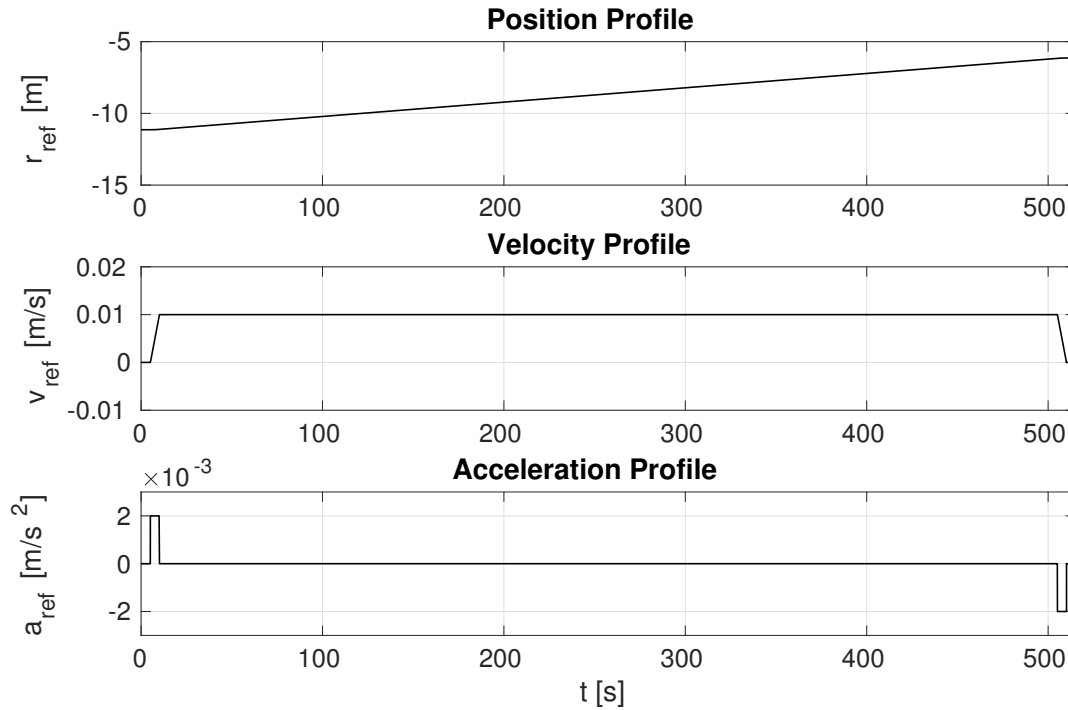


Figure 4.12: Position and velocity profiles from SK_0 to SK_1 , assuming an under saturated thrust command of 20 [mN].

Notice from Fig. 4.12 that the trajectory has been computed assuming a maximum thrust of 20 [mN], which prevents the thrust in the x -axis from saturation. For a satellite

of approximately 10 [kg], this thrust provides an acceleration of 2 [m/s²]. It is also worth it to remark the initial and final seconds remaining static, with 0 velocity and acceleration. This is done for the MPC to capture the change in motion, from being in rest to start the approach and viceversa, which allows the controller to be more optimal.

Similarly, the guidance profiles for the motion between SK₁ and SK₂ are shown in Fig. 4.13.

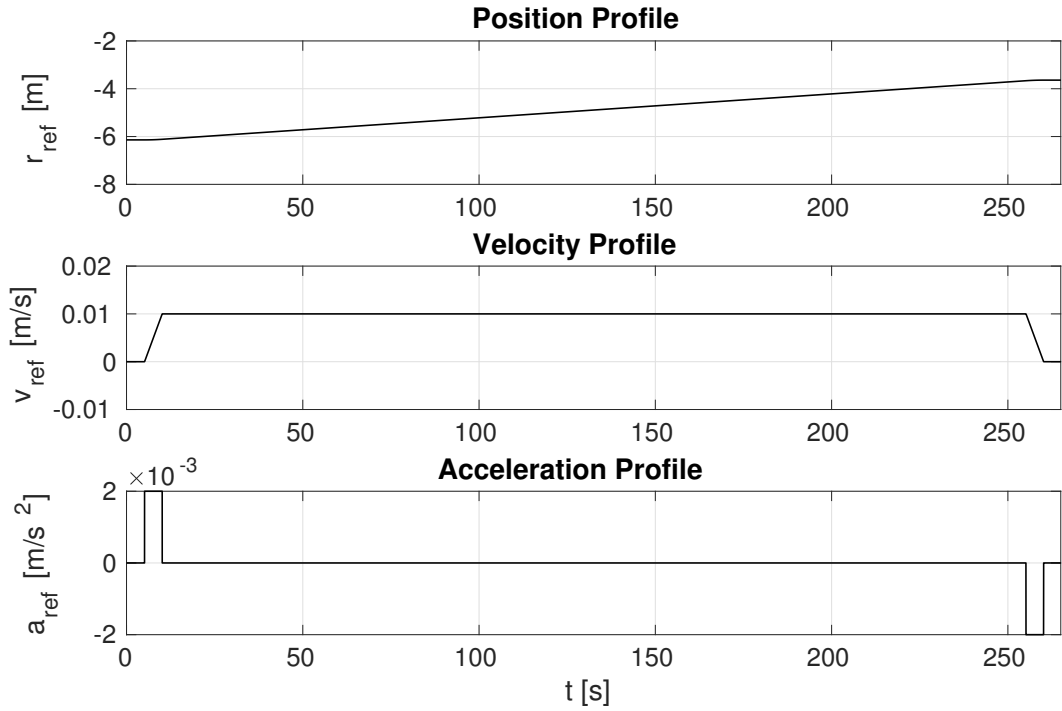


Figure 4.13: Position and velocity profiles from SK₁ to SK₂, assuming an under saturated thrust command of 20 [mN].

And the same reasoning as before applies to these reference trajectories.

For the last approach, between SK₂ and docking, no motion profiles are derived. The velocity and position references are set constant. The velocity reference to zero, and the position reference to the distance between the target's and the chaser's CoM when dock is achieved, i.e. the sum of the distances between the docking ports and their satellite's CoM. On the other hand, the velocity is additionally controlled by means of the soft-constraint defined in (4.37).

The reason why the profiles between SK points have not been defined together in one single motion profile lies in navigation system. While the chaser is in SK₀, the first MPC controller in Tab. 4.1 is used, with a constant reference trajectory to maintain the spacecraft in the SK point until the navigation system has converged. The time it

requires for the navigation to converge is not known a priori. Thus, it is convenient to remain in the SK point until navigation convergence is achieved. Subsequently, using the same controller, the motion profiles in Fig. 4.12 can be fed as reference trajectories to make the transition towards SK_1 . Once the second SK point is reached, a sensor handover takes place. Hence, the chaser must remain in SK_1 until the navigation has converged yet again. When the second navigation solution has converged, the motion profile in Fig. 4.13 can now be fed to the controller to make the transition to the last SK point. In SK_2 the process is repeated to remain in the position for a last sanity check of the different systems. Afterwards, the control system switches to the second MPC controller in Tab. 4.1, with the reference just discussed above to perform the last approach until docking.

Navigation System

5

The control system is build up on the premise that all the states of motion are well known. From there, control commands are derived in order to direct and correct the behaviour of the spacecraft. In reality, when the controller is on board, it has no certain knowledge of the true positions and velocities of the CubeSat.

Models could be used to propagate the states of motion according to a very well predefined initial condition. However, due to unmodeled uncertainties and disturbances—for the mere reason that they are hard or impossible to known—acting on the system, the error between the model predictions and the true states would escalate fast, and the problem would become intractable.

The navigation system is an essential part of the GNC system, given that it provides the required information of the actual states of the spacecraft to the control system. The controller performance will, therefore, depend on the precision of the state estimates, and therefore on the performance of the navigation system. The way these estimates are achieved is by filtering the information supplied by the different sensors on-board, or communication using ISL.

The reason why a filter is used lies on both, the ability of attenuating the sensor noise that corrupts the relevant information, and the possibility of estimating the states and extra parameters of the model. To do this, the sensors need to be modeled, such that the states of motion can be related to each of the sensors observations. This relation is called *measurement equation*, and will be found for all the sensors. The filter derived in this chapter for state-estimation is the well-known uncented Kalman filter (UKF).

5.1 Measurement Equations

The measurement equations are derived to mathematically relate the state vector, $\mathbf{x} = [\mathbf{q}^T \ \boldsymbol{\omega}^T \ \mathbf{h}^T \ \mathbf{r}^T \ \mathbf{v}^T]^T$, to the observations made by the sensory equipment, which are denoted by a vector \mathbf{y} . The measurement vector \mathbf{y} is an estimation of the true observation derived using the dynamic models in Sec. 3 and new models describing the measurement equipment. These new models are addressed in the following sections for the star tracker and the camera sensor. The dimension of \mathbf{y} vary depending on the output of each sensor, therefore it is not yet stated here, but in each of the following sections separately.

The angular velocity of the reaction wheels is assumed to be known, although extra sensor equipment such as rotary encoders should be added to also include the angular momentum of the RWs \mathbf{h} in the estimation problem of the state vector \mathbf{x} .

5.1.1 Star Tracker

The star tracker is an optical sensor that can identify and measure the position of the stars in order to compare them with an internal database, a star catalog, placed on board of the satellite. The difference between the positions of the stars in the images obtained by the cameras in the star tracker, and the positions of the same stars in the database is used to make a prediction with high-degree of accuracy of the satellite's orientation.

The attitude prediction delivered by the star tracker is given as a unit quaternion, $\mathbf{q}^{ST} \in \mathbb{H}_1$ defined as

$$\mathbf{q}^{ST} = \begin{bmatrix} \eta_{ST} \\ \boldsymbol{\varepsilon}_{ST} \end{bmatrix}, \quad (5.1)$$

where as usual, $\eta_{ST} \in \mathbb{R}$ is the scalar part and $\boldsymbol{\varepsilon}_{ST} \in \mathbb{R}^3$ is the vector part of the quaternion, and both fulfill the constraint in (3.2).

Measurements from the star tracker will be corrupted with noise, which will be modeled as an additive random vector $\mathbf{v}^{ST} \in \mathbb{R}^3$. Since unit quaternions are not confined in an Euclidean space and addition is not well defined, \mathbf{q}^{ST} must be transformed into a vector version in \mathbb{R}^3 . This is the angle vector $\boldsymbol{\theta}^{ST} \in \mathbb{R}^3$ which

is obtained through the logarithmic map of the quaternion [12].

$$\boldsymbol{\vartheta}^{ST} = \mathbf{e}^{ST} \boldsymbol{\vartheta}^{ST}, \quad (5.2a)$$

$$\mathbf{e}^{ST} = \frac{\boldsymbol{\varepsilon}_{ST}}{\|\boldsymbol{\varepsilon}_{ST}\|_2}, \quad (5.2b)$$

$$\boldsymbol{\vartheta}^{ST} = 2 \arctan (\|\boldsymbol{\varepsilon}_{ST}\|_2, \eta_{ST}). \quad (5.2c)$$

All these elements are time-dependent, continuously provided by the star tracker device at a specific frequency. The measurement equation for the star tracker is, therefore

$$\mathbf{y}_k^{ST} = h_{ST}(\mathbf{x}_k) + \mathbf{v}_k^{ST} = \boldsymbol{\vartheta}_k^{ST} + \mathbf{v}_k^{ST}. \quad (5.3)$$

The subindex k now denotes the sample time when the elements described were given by the star tracker. How \mathbf{v}_k^{ST} are distributed, and other vector noises introduced shortly, is explained in Sec. 5.2.

5.1.2 Camera Sensor

During close-range, most of the measurements come from camera-type sensors, since they provide more accuracy than any other sensor below ≈ 10 [m] range¹. Different solutions for a camera-based navigation have been tried in order to perform RVD. For instance, one navigation solution was based on sticking passive fiducial markers [36] on the target, which makes it a very compact and versatile approach since only a camera is required to detect them. This, although, suffers from different drawbacks, the most important being the certain illumination conditions required for the camera in order to read the markers.

The camera-based, or vision-based, navigation (VBN) used in this thesis is grounded on the proposed solution by [30], where Light-Emitting Diodes (LEDs) forming a specific pattern are utilized, instead of the fiducial markers lately mentioned. This approach shows an improvement during eclipses and low illumination scenarios, but its performance gets reduced with certain illumination conditions. In addition, the ideal pinhole camera model is used to describe the projections of the LED patterns onto the camera plane. In the VBN, these projections are the measurement outputs \mathbf{y} , and are mathematically related to the vector states \mathbf{x} as shown shortly.

Two different patterns can be distinguished in the LED formation, an *inner* and an *outer* pattern. The inner pattern is formed by a total of 5 LEDs, 4 in a so-called *cross-shaped* pattern and one out of the plane generated by the previous 4, as shown in Fig. 5.1. With

¹ See Fig. 1.3 in Sec. 1.2

the addition of the LED out of the plane, the inner pattern is able to provide 6 DoF information. Although, the inherent symmetry in the form of the pattern introduces some ambiguity in rotations around the x axis for rotations bigger than $\pm\frac{\pi}{2}$ [rad]. This is not a big concern, as this ambiguity can be easily overcome adding a 6th LED that breaks the symmetry.

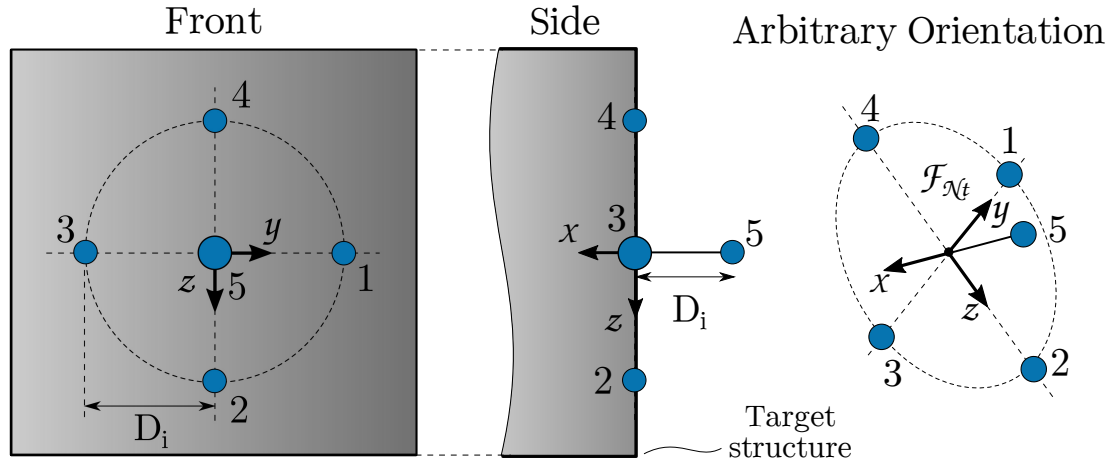


Figure 5.1: Inner LED pattern.

The outer pattern consist of 3 LEDs —a,b and 5— forming a “V” shape, as in Fig. 5.2. They are used to reach the first SK point in the final approach axis —from -10 [m] to -5 [m] range. According to [30], the estimation of the attitude using the outer pattern can be poor at these distances, depending on the visibility of LED 5. The outer pattern is thus used to only infer the relative distance, and the relative attitude is estimated with the help of the other sensors and ISC.

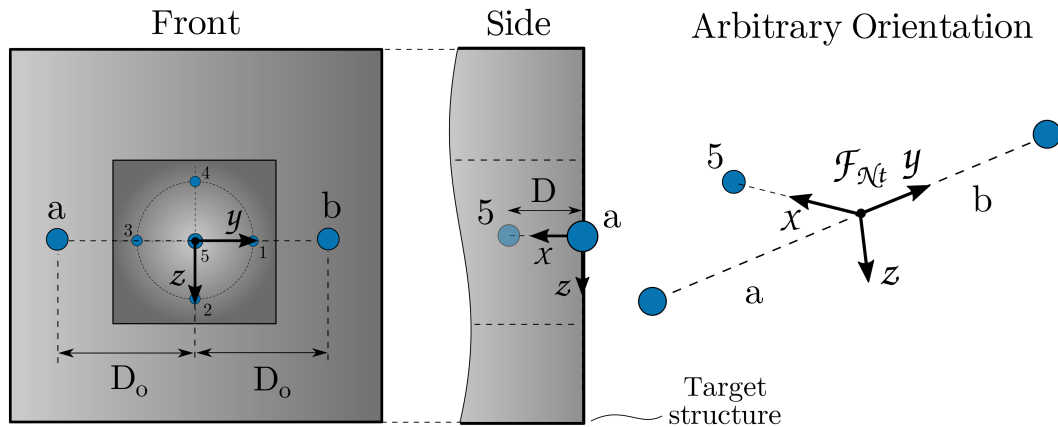


Figure 5.2: Outer LED pattern.

The complete pattern has the following structure, note that LED 5 is in both patterns.

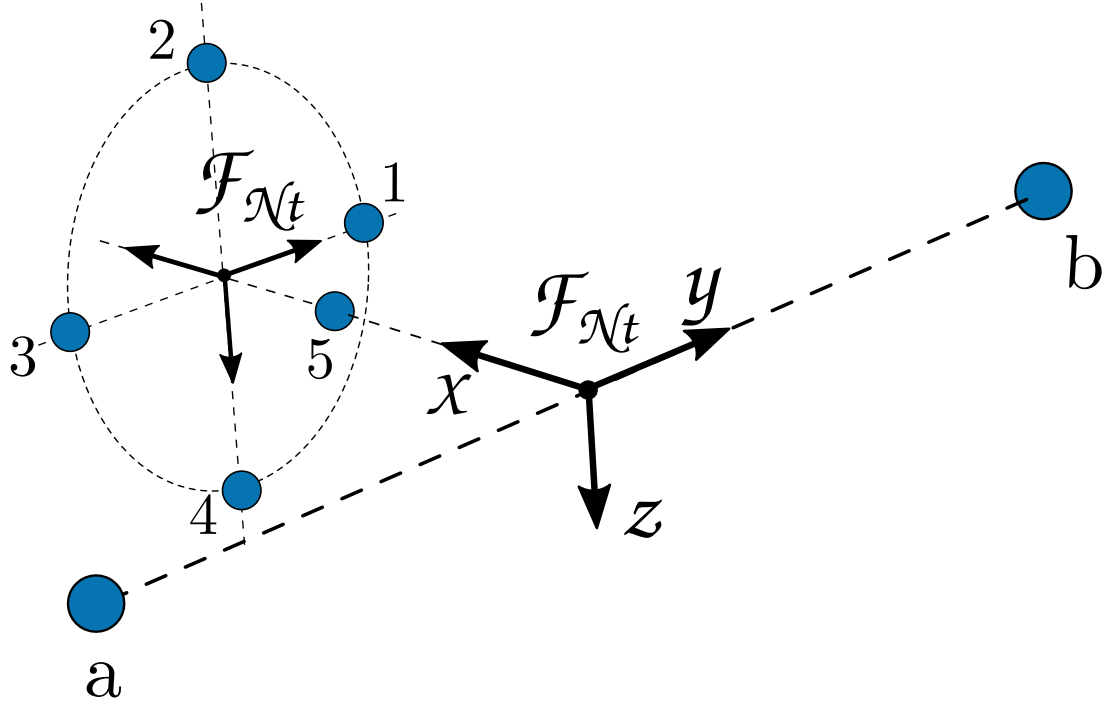


Figure 5.3: Complete LED pattern.

Notice that two navigation frames, \mathcal{F}_{N_t} , are used. One to locate the LEDs of the inner pattern, and another to locate the LEDs of the outer pattern, both denoted with a vector $\mathbf{d}_{N_t}^i \in \mathbb{R}^3$ with $i \in \{1, 2, 3, 4, 5, a, b\}$. Being the center of each pattern centered in the origin of each frame, the position of the LEDs of the inner pattern is well-known

$$\mathbf{d}_{N_t}^1 = [0 \quad \frac{D_i f}{R} \quad 0]^T \quad (5.4a)$$

$$\mathbf{d}_{N_t}^2 = [0 \quad 0 \quad \frac{D_i f}{R}]^T \quad (5.4b)$$

$$\mathbf{d}_{N_t}^3 = [0 \quad -\frac{D_i f}{R} \quad 0]^T \quad (5.4c)$$

$$\mathbf{d}_{N_t}^4 = [0 \quad 0 \quad -\frac{D_i f}{R}]^T \quad (5.4d)$$

$$\mathbf{d}_{N_t}^5 = [-\frac{D_i f}{R} \quad 0 \quad 0]^T, \quad (5.4e)$$

as well as the ones of the outer pattern in its respective frame

$$\mathbf{d}_{N_t}^a = [0 \quad \frac{D_o f}{R} \quad 0]^T \quad (5.5a)$$

$$\mathbf{d}_{N_t}^5 = [\frac{D f}{R} \quad 0 \quad 0]^T \quad (5.5b)$$

$$\mathbf{d}_{N_t}^b = [0 \quad -\frac{D_o f}{R} \quad 0]^T, \quad (5.5c)$$

where the positions $D_i = D = 2$ [cm] and $D_o = 5$ [cm] are scaled by the focal length, f , and the range, $R = \|\mathbf{r}_{\mathcal{N}_t}^{\mathcal{N}_c}\|_2 \in \mathbb{R}$. The vector $\mathbf{r}_{\mathcal{N}_t}^{\mathcal{N}_c} \in \mathbb{R}^3$ is the distance vector from the target's to the chaser's navigation reference frame, expressed in $\mathcal{F}_{\mathcal{N}_t}$. The focal length is the distance between the lens and the camera plane, and it is set to 0.04 [mm], as in [30]. The focal length affects the magnification and the field of view (FoV). This parameter, and others introduced hereafter, are depicted in Fig. 5.4

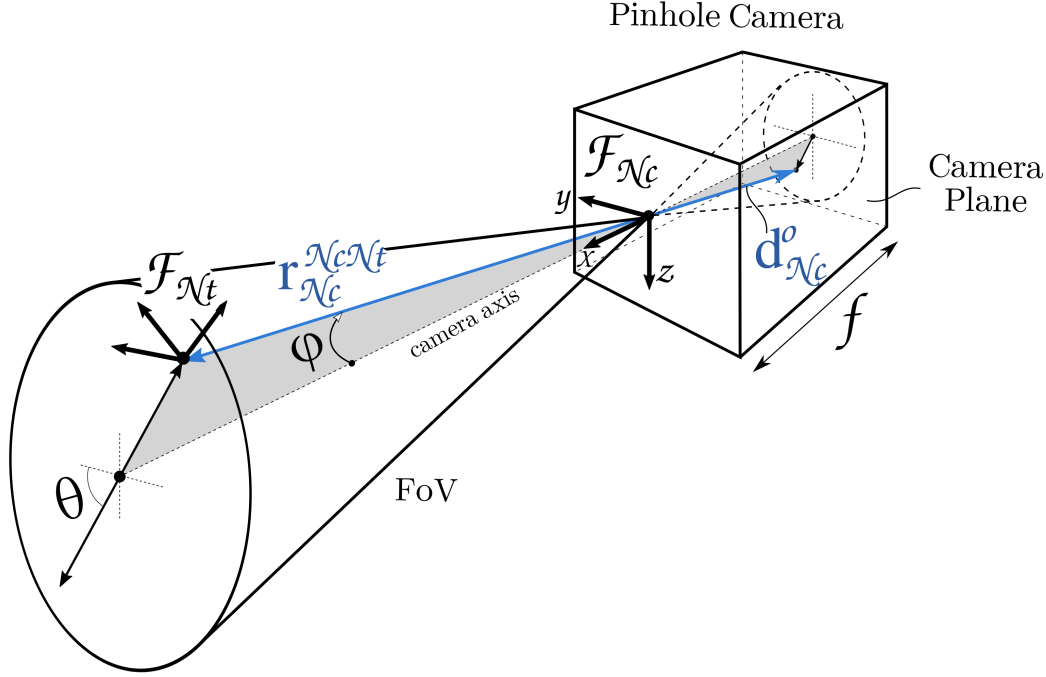


Figure 5.4: Ideal pinhole camera model.

To build the camera measurement equation, $\mathbf{y}_k^{VBN} = h_{VBN}(\mathbf{x}_k)$, it is required to relate the output of the camera, \mathbf{y}_k^{VBN} , to the motion states \mathbf{x}_k , where the measurement \mathbf{y}_k^{VBN} is the projection of the LEDs on the camera plane. Note, that the vectors are expressed in the chaser navigation frame $\mathcal{F}_{\mathcal{N}_c}$, and the projections on the camera plane correspond to the y and z components of these vectors.

According to this, during the first approach —from 10 to 5[m] range— the measurement output is $\mathbf{y}^{VBN_3} = [d_{\mathcal{N}_c,y}^a \ d_{\mathcal{N}_c,z}^a \ d_{\mathcal{N}_c,y}^5 \ d_{\mathcal{N}_c,z}^5 \ d_{\mathcal{N}_c,y}^b \ d_{\mathcal{N}_c,z}^b]^T$, whereas in the second and last approach before docking —from 5 to 0 [m] range— the measurement output is $\mathbf{y}^{VBN_5} = [d_{\mathcal{N}_c,y}^1 \ d_{\mathcal{N}_c,z}^1 \ d_{\mathcal{N}_c,y}^2 \ d_{\mathcal{N}_c,z}^2 \ \dots \ d_{\mathcal{N}_c,y}^5 \ d_{\mathcal{N}_c,z}^5]^T$, being $\mathbf{d}_{\mathcal{N}_c}^i \triangleq [d_{\mathcal{N}_c,x}^i \ d_{\mathcal{N}_c,y}^i \ d_{\mathcal{N}_c,z}^i]^T \in \mathbb{R}^3$ the position of the LED $i \in \{1, 2, 3, 4, 5, a, b\}$ expressed in the chaser's navigation reference frame $\mathcal{F}_{\mathcal{N}_c}$.

The next transformation can be employed in order to obtain $\mathbf{d}_{\mathcal{N}_c}^i$, assuming that the navigation frame and the body frame share the same orientation.

$$\begin{bmatrix} 0 \\ \mathbf{d}_{\mathcal{N}_c}^i \end{bmatrix} = (\mathbf{q}_{VBN} \otimes \mathbf{q})^* \otimes \begin{bmatrix} 0 \\ \mathbf{d}_{\mathcal{N}_t}^i \end{bmatrix} \otimes (\mathbf{q} \otimes \mathbf{q}_{VBN}) + \begin{bmatrix} 0 \\ \mathbf{d}_{\mathcal{N}_c}^o \end{bmatrix}, \quad i \in \{1, \dots, 5, a, b\}. \quad (5.6)$$

Or, equivalently

$$\mathbf{d}_{\mathcal{N}_c}^i = \mathbf{R}\{\mathbf{q}_{VBN}\}\mathbf{R}\{\mathbf{q}\}\mathbf{d}_{\mathcal{N}_t}^i + \mathbf{d}_{\mathcal{N}_c}^o, \quad i \in \{1, \dots, 5, a, b\}. \quad (5.7)$$

where $\mathbf{q}_{VBN} \in \mathbb{H}_1$ is the attitude transformation needed to align the center of the pattern with the camera axis, and $\mathbf{d}_{\mathcal{N}_c}^o \in \mathbb{R}^3$ denotes the position of the center of the pattern, i.e. the origin of $\mathcal{F}_{\mathcal{N}_t}$, expressed in $\mathcal{F}_{\mathcal{N}_c}$. Both elements need to be found. The vector $\mathbf{d}_{\mathcal{N}_c}^o$ can be related to the relative position vector $\mathbf{r}_{\mathcal{N}_c}^{\mathcal{N}_t} \in \mathbb{R}^3$ by means of trigonometry relations, as shown in Fig. 5.4.

Notice that the vector $\mathbf{r}_{\mathcal{N}_c}^{\mathcal{N}_t}$ describes the relative position of the navigation frames, and is not known a priori. It can be linked to the analogous vector $\mathbf{r}_{\mathcal{N}_t}^{\mathcal{N}_c}$ —which is easier to find— as the following simple transformation that only changes the frame of reference from the target's navigation frame to the chaser's, and the direction of the vector is inverted.

$$\mathbf{r}_{\mathcal{N}_c}^{\mathcal{N}_t} = -\mathbf{R}\{\mathbf{q}\}\mathbf{r}_{\mathcal{N}_t}^{\mathcal{N}_c}. \quad (5.8)$$

Considering $\mathbf{r}_{\mathcal{N}_t}^{\mathcal{N}_c}$, the relation with the state vectors $\mathbf{r} \triangleq \mathbf{r}_{\mathcal{T}}^{\mathcal{T}C} = \mathbf{r}_{\mathcal{N}_t}^{\mathcal{T}C}$ and $\mathbf{q} \triangleq \mathbf{q}_{\mathcal{T}}^C = \mathbf{q}_{\mathcal{N}_t}^{\mathcal{N}_c}$ is easier to establish.

$$\mathbf{r}_{\mathcal{N}_t}^{\mathcal{N}_c} = \mathbf{r} + \mathbf{R}\{\mathbf{q}\}\mathbf{p}_{\mathcal{N}_c}^{CN_c} - \mathbf{p}_{\mathcal{N}_t}^{\mathcal{T}N_t}, \quad (5.9)$$

where $\mathbf{p}_{\mathcal{N}_c}^{CN_c} \in \mathbb{R}^3$ and $\mathbf{p}_{\mathcal{N}_t}^{\mathcal{T}N_t} \in \mathbb{R}^3$ are the vectors describing the position of the origin of the chaser's and target's navigation frames respectively, w.r.t the origin of their body frame, and expressed in their navigation frames.

The angle φ in Fig. 5.4 can now be obtained using $\mathbf{r}_{\mathcal{N}_c}^{\mathcal{N}_t}$ as

$$\tan \varphi = \frac{\sqrt{(r_{\mathcal{N}_c, y}^{\mathcal{N}_t})^2 + (r_{\mathcal{N}_c, z}^{\mathcal{N}_t})^2}}{r_{\mathcal{N}_c, x}^{\mathcal{N}_t}}. \quad (5.10)$$

Through Fig. 5.4, it can be deduced that φ is actually the angle needed to align the center of the pattern with the chaser's camera axis, and therefore, will be used also to find \mathbf{q}_{VBN} . Knowing the focus length, f , of the camera, along with (5.10), it is possible to find the magnitude of $\mathbf{d}_{\mathcal{N}_c}^o$ as follows.

$$\|\mathbf{d}_{\mathcal{N}_c}^o\|_2 = f \tan \varphi. \quad (5.11)$$

To completely define $\mathbf{d}_{\mathcal{N}_c}^o$ only one extra angle, θ , is needed. Afterwards, $\mathbf{d}_{\mathcal{N}_c}^o$ can be defined as $\mathbf{d}_{\mathcal{N}_c}^o = [* \ f \tan \varphi \cos \theta \ f \tan \varphi \sin \theta]^T$. Unfortunately, when defining the $\sin(\cdot)$ and $\cos(\cdot)$ functions of θ using only the y and z components of $\mathbf{r}_{\mathcal{N}_c}^{\mathcal{N}_t}$, a $\frac{0}{0}$ indetermination is reached when the center of the pattern gets aligned with the camera axis, namely when $r_{\mathcal{N}_c,y}^{\mathcal{N}_t} = r_{\mathcal{N}_c,z}^{\mathcal{N}_t} = 0$, as it can be seen in (5.12).

$$\sin \theta = \frac{r_{\mathcal{N}_c,z}^{\mathcal{N}_t}}{\sqrt{\left(r_{\mathcal{N}_c,y}^{\mathcal{N}_t}\right)^2 + \left(r_{\mathcal{N}_c,z}^{\mathcal{N}_t}\right)^2}}, \quad (5.12a)$$

$$\cos \theta = \frac{r_{\mathcal{N}_c,y}^{\mathcal{N}_t}}{\sqrt{\left(r_{\mathcal{N}_c,y}^{\mathcal{N}_t}\right)^2 + \left(r_{\mathcal{N}_c,z}^{\mathcal{N}_t}\right)^2}}. \quad (5.12b)$$

In order to avoid mathematical indeterminations, the same procedure employed by [30] is enforced. Henceforth, two angles will be used to determine the orientation of the vector $\mathbf{d}_{\mathcal{N}_c}^o$ in the camera sensor plane, which will involve also the x component of $\mathbf{r}_{\mathcal{N}_c}^{\mathcal{N}_t}$. The advantage is that the minimum distance between the navigation frames is 7 [cm], which ensures that no indetermination is reached.

The two angles used eventually are the Az – *azimuth* and El – *elevation* angles, which are mathematically defined as follows.

$$\tan Az = \frac{r_{\mathcal{N}_c,y}^{\mathcal{N}_t}}{r_{\mathcal{N}_c,x}^{\mathcal{N}_t}}, \quad (5.13a)$$

$$\tan El = - \frac{r_{\mathcal{N}_c,z}^{\mathcal{N}_t}}{\sqrt{\left(r_{\mathcal{N}_c,x}^{\mathcal{N}_t}\right)^2 + \left(r_{\mathcal{N}_c,y}^{\mathcal{N}_t}\right)^2}}. \quad (5.13b)$$

Finally, the position vector locating the center of the cross-shaped patterns in the camera sensor plane is

$$\mathbf{d}_{\mathcal{N}_c}^o = \begin{bmatrix} * \\ f \tan \varphi \tan Az \\ f \tan \varphi \tan El \end{bmatrix}. \quad (5.14)$$

Since only the projection in the camera sensor is needed, which is in the $y - z$ plane of $\mathcal{F}_{\mathcal{N}_c}$, the x component is disregarded.

The last element required to utilize equation (5.7) is the unit quaternion $\mathbf{q}_{VBN} \triangleq [\eta_{VBN} \ \boldsymbol{\varepsilon}_{VBN}^T]^T$, with $\eta_{VBN} \in \mathbb{R}$ and $\boldsymbol{\varepsilon}_{VBN} \in \mathbb{R}^3$ and also fulfilling constraint (3.2). Given that the rotation angle φ have already been obtained in equation (5.10), according to the definition of quaternion, then the only part left to determine is the vector part $\boldsymbol{\varepsilon}_{VBN}$,

that is composed by the angle of rotation φ and an axis of rotation $\hat{\mathbf{e}} = [e_x \ e_y \ e_z]^T$ as shown below.

$$\eta_{VBN} = \cos \frac{\varphi}{2} \quad (5.15a)$$

$$\boldsymbol{\epsilon}_{VBN} = \begin{bmatrix} e_x \sin \frac{\varphi}{2} & e_y \sin \frac{\varphi}{2} & e_z \sin \frac{\varphi}{2} \end{bmatrix}^T \quad (5.15b)$$

The axis of rotation, $\hat{\mathbf{e}}$, can easily be obtained, for example, as

$$\hat{\mathbf{e}} = \frac{\mathbf{r}_{\mathcal{N}_c}^{\mathcal{N}_c \mathcal{N}_t}}{\|\mathbf{r}_{\mathcal{N}_c}^{\mathcal{N}_c \mathcal{N}_t}\|_2} \times \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}. \quad (5.16)$$

Employing relation (3.5), the quaternion \mathbf{q}_{VBN} can be expressed as a rotation matrix, i.e. $\mathbf{R}\{\mathbf{q}_{VBN}\}$, and the transformation in (5.7) can finally be applied to find the projection of all the LEDs in the camera sensor plane, i.e. $\mathbf{d}_{\mathcal{N}_c}^i$.

During the first approach maneuver until 5 [m] range, the measurement vector \mathbf{y}_{VBN_3} takes only the y and z components of $\mathbf{d}_{\mathcal{N}_c}^i$ for $i \in \{a, 5, b\}$; and is used in the next section mainly to estimate \mathbf{r} . From 5 [m] range until docking, the measurement vector \mathbf{y}_{VBN_5} takes the y and z components of $\mathbf{d}_{\mathcal{N}_c}^i$ for $i \in \{1, 2, 3, 4, 5\}$, and is used to estimate both, \mathbf{q} and \mathbf{r} . In summary, there are two VBN equations, those are

$$\mathbf{y}_k^{VBN_3} = h_{VBN_3}(\mathbf{x}) + \mathbf{v}_k^{VBN_3}, \quad (5.17a)$$

$$\mathbf{y}_k^{VBN_5} = h_{VBN_5}(\mathbf{x}) + \mathbf{v}_k^{VBN_5}. \quad (5.17b)$$

Notice that throughout this section, the centroid of the LEDs have been used to model the measurement equation. Of course, before being able to use the this equation, it is required to develop a pattern recognition algorithm to identify the LED patterns from the images and extract the centroid positions from them [30].

5.2 Filtering

A filter is a an algorithm used to attenuate the noise in a process measurement, e.g. in the outputs obtained from the sensors described above. Notwithstanding, noise attenuation is not the only application of a filter. A very essential framework of filters in control theory is state- and parameter-estimation. In the following section an extension of the well-known Kalman filter (KF) to nonlinear models for the plant and the measurements is formulated for state-estimation.

The linear KF is a simple algorithm based on the two first moments, i.e. the mean and the covariance, of the states and measurements. If a linear discrete-time stochastic

model is assumed, then it can be described by equations of the following form

$$\mathbf{x}_{k+1} = \Phi \mathbf{x}_k + \Gamma \mathbf{u}_k + \mathbf{w}_k, \quad (5.18a)$$

$$\mathbf{y}_k = \mathbf{H} \mathbf{x}_k + \mathbf{v}_k, \quad (5.18b)$$

with $\mathbf{w}_k \stackrel{\text{iid}}{\sim} \mathcal{N}(\mathbf{0}, \Sigma_w)$ and $\mathbf{v}_k \stackrel{\text{iid}}{\sim} \mathcal{N}(\mathbf{0}, \Sigma_v)$ as the process and the measurement noise respectively, having Σ_w and Σ_v as covariance matrices. The noise processes are assumed to be additive, *iid* —independent and identically distributed— and uncorrelated. This is summarized in the following table.

| Noise | \mathbf{w}_k | \mathbf{v}_k | |
|--------------------------------|--|--|-------------------------------|
| Identically Distributed | $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \Sigma_w)$ | $\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \Sigma_v)$ | $\forall k$ |
| Independent | $\mathbf{E}[\mathbf{w}_k \mathbf{w}_j^T] = \mathbf{0}$ $\mathbf{E}[\mathbf{w}_k \mathbf{w}_j^T] = \Sigma_w$ | $\mathbf{E}[\mathbf{v}_k \mathbf{v}_j^T] = \mathbf{0}$ $\mathbf{E}[\mathbf{v}_k \mathbf{v}_j^T] = \Sigma_v$ | $\forall k \neq j$ $k = j$ |
| Uncorrelated | $\mathbf{E}[\mathbf{w}_k \mathbf{v}_j^T] = \mathbf{0}$ | | $\forall k, j$ |

Table 5.1: Characteristics of the process and measurement noise \mathbf{w}_k and \mathbf{v}_k assumed for a Kalman filter.

where $\mathbf{E}[\cdot]$ is the expectation operator and $k, j \in \mathbb{N}$. Given the model in (5.18), a set of measurements and initial conditions $\mathbf{x}_0 \stackrel{\text{iid}}{\sim} \mathcal{N}(\hat{\mathbf{x}}_0, \mathbf{P}_0)$, where $\hat{\mathbf{x}}_0$ is the mean and \mathbf{P}_0 the covariance of the initial condition, then the linear KF can be stated as an optimization problem which aims to find $\hat{\mathbf{x}}$ that minimize the mean square error (MSE). This can be expressed as

$$\min_{\hat{\mathbf{x}}_k} \mathbf{E}[(\mathbf{x}_k - \hat{\mathbf{x}}_k)^T (\mathbf{x}_k - \hat{\mathbf{x}}_k)]. \quad (5.19)$$

The algorithm described in the following table solves the problem in (5.19) in a recursive manner. The terms with the superscript - denote the *a priori* —or predicted— values, whereas the superscript + designates the *a posteriori* —or corrected— values. The true state and true measurement coming from the sensor are denoted by \mathbf{x}_k and \mathbf{y}_k respectively, and their estimations as $\hat{\mathbf{x}}_k$ and $\hat{\mathbf{y}}_k$.

| Linear KF algorithm | |
|-----------------------|--|
| Initialization | $\hat{\mathbf{x}}_0 = \mathbf{E}[\mathbf{x}_0]$ $\mathbf{P}_0 = \mathbf{E}[(\mathbf{x}_0 - \hat{\mathbf{x}}_0)(\mathbf{x}_0 - \hat{\mathbf{x}}_0)^T]$ |
| Propagation | $\hat{\mathbf{x}}_{k+1}^- = \mathbf{E}[\mathbf{x}_{k+1} \mathcal{Y}_k] = \Phi \hat{\mathbf{x}}_k^+ + \Gamma \mathbf{u}_k$ $\mathbf{P}_{k+1}^- = \mathbf{E}[(\mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1}^-)(\mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1}^-)^T \mathcal{Y}_k] = \Phi \mathbf{P}_k^+ \Phi^T + \Sigma_w$ |
| Innovation | $\Delta \hat{\mathbf{y}}_{k+1} = \mathbf{y}_{k+1} - \hat{\mathbf{y}}_{k+1} = \mathbf{y}_{k+1} - \mathbf{H} \hat{\mathbf{x}}_{k+1}^-$ $\mathbf{P}_{k+1}^{yy} = \mathbf{E}[(\mathbf{y}_{k+1} - \hat{\mathbf{y}}_{k+1})(\mathbf{y}_{k+1} - \hat{\mathbf{y}}_{k+1})^T \mathcal{Y}_{k+1}] = \mathbf{H} \mathbf{P}_{k+1}^- \mathbf{H}^T + \Sigma_v$ |
| Kalman Gain | $\mathbf{K}_{k+1} = \mathbf{P}_{k+1}^- \mathbf{H}^T (\mathbf{P}_{k+1}^{yy})^{-1}$ |
| Update | $\hat{\mathbf{x}}_{k+1}^+ = \hat{\mathbf{x}}_{k+1}^- + \mathbf{K}_{k+1} \Delta \hat{\mathbf{y}}_{k+1}$ $\mathbf{P}_{k+1}^+ = [\mathbf{I} - \mathbf{K}_{k+1} \mathbf{H}] \mathbf{P}_{k+1}^-$ |

Table 5.2: Kalman filter algorithm for linear discrete-time systems.

The term $\mathbf{E}[\mathbf{x}_{k+1} | \mathcal{Y}_k]$ in table Tab. 5.2 is a conditional expectation, which reads as the expectation of \mathbf{x}_{k+1} given a set of $\mathcal{Y}_k = [\mathbf{y}_k \ \mathbf{y}_{k-1} \ \dots \ \mathbf{y}_0]$ observations up until time k .

If the plant behaves as a LTI system, then the KF is an *optimal* estimator. If that is not the case, and the plant behaves in some nonlinear fashion, then some approximations can still be done in order to keep using the algorithm in Tab. 5.2 as an state-estimator. However, the resulting algorithm is not *optimal* anymore, but sub-optimal. This is what happens with, probably, the most known and widely used version of the KF, the extended Kalman filter (EKF). For a nonlinear system, the value of the expectations of the propagation and innovation steps in table Tab. 5.2 are generally hard to determine given the nonlinear relation $\mathbf{f}_d(k, \mathbf{x}_k, \mathbf{u}_k)$ between $\hat{\mathbf{x}}_{k+1}$ and $\hat{\mathbf{x}}_k$, and the nonlinear relation $\mathbf{h}_d(\mathbf{x}_k)$ between \mathbf{y}_k and $\hat{\mathbf{x}}_k$. To overcome this complication, the EKF uses the Taylor expansion to approximate the functions \mathbf{f}_d and \mathbf{h}_d , and recalculate the expectations, which eventually look like

$$\mathbf{E}[\mathbf{x}_{k+1} | \mathcal{Y}_k] \approx \Phi_k \hat{\mathbf{x}}_k^+ + \Gamma_k \mathbf{u}_k, \quad (5.20a)$$

$$\mathbf{E}[(\mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1}^-)(\mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1}^-)^T | \mathcal{Y}_k] \approx \Phi_k \mathbf{P}_k^+ \Phi_k^T + \Sigma_w, \quad (5.20b)$$

$$\mathbf{E}[(\mathbf{y}_{k+1} - \mathbf{H} \hat{\mathbf{x}}_{k+1}^-)(\mathbf{y}_{k+1} - \mathbf{H} \hat{\mathbf{x}}_{k+1}^-)^T | \mathcal{Y}_{k+1}] \approx \mathbf{H}_k \mathbf{P}_{k+1}^- \mathbf{H}_k^T + \Sigma_v, \quad (5.20c)$$

where now Φ_k , Γ_k and \mathbf{H}_k vary with k , according to the following Jacobian matrices.

$$\Phi_k = \left. \frac{\delta f_d}{\delta \mathbf{x}} \right|_{\mathbf{x}=\hat{\mathbf{x}}_k^+}, \quad (5.21a)$$

$$\Gamma_k = \left. \frac{\delta f_d}{\delta \mathbf{u}} \right|_{\mathbf{u}=\mathbf{u}_k}, \quad (5.21b)$$

$$\mathbf{H}_k = \left. \frac{\delta h_d}{\delta \mathbf{x}} \right|_{\mathbf{x}=\hat{\mathbf{x}}_k^+}. \quad (5.21c)$$

Different reasons have lead to disregard the EKF as an option for state-estimation in this work. Firstly, the high nonlinear models $f_d(k, \mathbf{x}_k, \mathbf{u}_k)$ and $h_d(\hat{\mathbf{x}}_k)$ —which are the discrete versions of the models obtained in Sec. 3 and Sec. 5.1— complicate the derivation of the Jacobians in (5.21a). Moreover, after assuming these approximations in the EKF, it is known that the state-estimates can diverge in the worst cases where the model have a high order of nonlinearity [22]. These problems can be avoided with a different formulation of the KF, the so-called uncenced Kalman filter (UKF). Apart from relaxing the assumption of additive noise in the model, the UKF also offers better estimations than the EKF, or at least as good, and therefore faster convergence for large initialization errors [9].

5.2.1 Nonlinear Discrete-Time Stochastic Model

As for the case of the EKF, a nonlinear discrete-time stochastic model is considered to model uncertainties, which have the following form.

$$\mathbf{x}_{k+1} = f_d(k, \mathbf{x}_k, \mathbf{u}_k) + \Upsilon \mathbf{w}_k, \quad (5.22a)$$

$$\mathbf{y}_k = h_d(\mathbf{x}_k, \mathbf{v}_k). \quad (5.22b)$$

The model $f_d(\cdot)$ is the discrete version of $f(\cdot)$ derived in Sec. 3, and $h_d(\cdot)$ are the measurement equations derived in Sec. 5.1. The process noise $\mathbf{w}_k \stackrel{\text{iid}}{\sim} \mathcal{N}(\mathbf{0}, \Sigma_w)$ is assumed additive, and it is meant to model the sources of uncertainty coming from the RWs and the RCP module, and how the disturbances affect the system. Since this only affects the dynamic equations, then $\Sigma_w \in \mathbb{R}^{6 \times 6}$, and a matrix $\Upsilon \in \mathbb{R}^{16 \times 6}$ multiplies the noise vector $\mathbf{w}_k \in \mathbb{R}^6$ to only attain the dynamic equations. The matrix Υ is then

$$\Upsilon \triangleq \begin{bmatrix} \mathbf{0}_{4 \times 3} & \mathbf{0}_{4 \times 3} \\ \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix}. \quad (5.23)$$

The measurement noise, \mathbf{v}_k , is assumed uncorrelated from the process noise, \mathbf{w}_k , and introduces the uncertainties coming from the camera sensor and the star tracker. From 10-5 [m] range both sensors are used, then $\mathbf{v}_k \in \mathbb{R}^9$ since the output vector is $\mathbf{y}_k = [\mathbf{y}^{VBN_3} \ \mathbf{y}^{ST}]^T$, with the camera sensor detecting the outer pattern. Then

$$\mathbf{v}_k \triangleq \begin{bmatrix} \mathbf{v}_k^{VBN_3} \\ \mathbf{v}_k^{ST} \end{bmatrix}. \quad (5.24)$$

From 5[m] until docking the measurement output is $\mathbf{y}_k = \mathbf{y}^{VBN_5}$, detecting now the inner pattern. Then, the noise measurement vector is only

$$\mathbf{v}_k \triangleq \mathbf{v}_k^{VBN_5}. \quad (5.25)$$

The uncertainties on the camera sensor come from two uncorrelated sources, i.e. a scalar noise $v_k^o \in \mathbb{R}$ that affects all the LEDs equally —modeling the uncertainty of the camera position— and a vector noise $\mathbf{v}_k^i \in \mathbb{R}^p$ that affects each LED independently —modeling the pixel density noise. Then, either if the outer pattern is being observed, or it is the inner one, the camera noise is defined as

$$\mathbf{v}_k^{VBN_{p/2}} = \mathbf{v}_k^i + \mathbf{1}_p v_k^o, \quad (5.26)$$

where $\mathbf{1}_p$ is a column vector with p 1's, and $p = 6$ from 10-5 [m] range and $p = 10$ from 5 [m] range until docking. It is important to remark that \mathbf{v}_k^{VBN} should not be modeled as additive noise in the state-space model (5.22a) given that the inaccuracy of the VBN reduces with the range. This is one of the main distinctions with the EKF, which only can handle additive noise. However, a worse case scenario is considered with constant pixel noise, rather than variable. With this assumption, the variance of the two sources can directly be added, i.e. $\sigma_{VBN}^2 = \sigma_{LED}^2 + \sigma_{CAM}^2$. According to this, the VBN noise is distributed as $\mathbf{v}_k^{VBN_{p/2}} \stackrel{\text{iid}}{\sim} \mathcal{N}(\mathbf{0}, \Sigma_v^{VBN_{p/2}})$.

The uncertainties on the star tracker, on the other hand, are additive and distributed as $\mathbf{v}_k^{ST} \stackrel{\text{iid}}{\sim} \mathcal{N}(\mathbf{0}, \Sigma_v^{ST})$. The total standard deviation of the star tracker is given in Tab. 1.1. It is fair to assume that the standard deviation regarding each axis is equal, and thus

$$\sigma_{ST} = \sqrt{\sigma_{ST_x}^2 + \sigma_{ST_y}^2 + \sigma_{ST_z}^2} = \sqrt{3}\sigma_{ST_x}. \quad (5.27)$$

The covariance matrices used to model uncertainties are extracted from Sec. 1.2.

$$\Sigma_w = \begin{bmatrix} 5 \cdot 10^{-5} \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & 10^{-6} \mathbf{I}_3 \end{bmatrix}, \quad (5.28a)$$

$$\Sigma_v^{VBN_3} = (0.06^2 + (1.5 \cdot 10^{-4})^2) \mathbf{I}_6, \quad (5.28b)$$

$$\Sigma_v^{VBN_5} = (0.06^2 + (1.5 \cdot 10^{-4})^2) \mathbf{I}_{10}, \quad (5.28c)$$

$$\Sigma_v^{ST} = \left(\frac{5 \cdot 10^{-3}}{\sqrt{3}} \right)^2 \mathbf{I}_3 \quad (5.28d)$$

$$(5.28e)$$

5.2.2 Uncented Kalman Filter

The uncenced Kalman filter (UKF) follows the same principle as the linear KF, where time propagation is made and then the prediction is corrected in the update step using the measurements. The key difference between UKF and the other KF formulations is that UKF use a strategic minimal set of sample points —called *sigma* points and denoted as χ_k — to describe and propagate the probability distribution of the measurements and states. This is, of course, a more accurate approximation to capture the probability distributions in nonlinear systems than using the Jacobian matrices.

The set of sigma points are calculated as

$$\chi_k^0 = \hat{\mathbf{x}}_k^+, \quad (5.29a)$$

$$\chi_k^i = \hat{\mathbf{x}}_k^+ + \delta\chi_k^i, \quad i \in \{1, 2, \dots, n\} \quad (5.29b)$$

$$\chi_k^i = \hat{\mathbf{x}}_k^+ - \delta\chi_k^i, \quad i \in \{n+1, n+2, \dots, 2n\}. \quad (5.29c)$$

being n the dimension of the state vector and $\delta\chi_k^i$ the *error sigma points*, which are determined by the following expression.

$$\delta\chi_k^i = \left(\sqrt{(n+\lambda)(\mathbf{P}_k^+ + \Sigma_w)} \right)_i, \quad i \in \{1, 2, \dots, 2n\}. \quad (5.30)$$

The subindex i in $\left(\sqrt{(n+\lambda)(\mathbf{P}_k^+ + \Sigma_w)} \right)_i$ refers to the column of the resultant matrix and k is the time step. The constant $\lambda = \alpha^2(n+\kappa) - n$ is a scaling parameter, that depends on α , which describes how much spread are the sigma points and κ which is an additional scaling parameter [6]. The square-root of a symmetric positive definite matrix can be computed using the *Cholesky decomposition*, Appendix C.1, which problem is to find a matrix \mathbf{L} such that

$$\mathbf{P} = \mathbf{L}\mathbf{L}^T. \quad (5.31)$$

Given that $(\mathbf{P}_k + \Sigma_w)$ is a symmetric positive definite matrix, the Cholesky decomposition can be applied to find the *square-root matrices* in (5.30).

At this point, a special distinction between some of the state variables must be pointed out. It is important to notice again that one of the state variables, the relative unit quaternion \mathbf{q} , do not belong to an Euclidean space, as it has been discussed before. Consequently, addition and subtraction are not well defined, thus the equations in (5.29) can not be used to obtain the sigma points —at least not for all the states. Moreover, a covariance matrix for a complete quaternion can not be obtained in the classical way, since it also involves additions and subtractions.

To avoid these problems, the inverse trick used for the star tracker measurement equation is attempted. Instead of defining an error covariance matrix for \mathbf{q} to directly

find the quaternion sigma points, a covariance matrix for the angle vector of the quaternion, $\boldsymbol{\vartheta} \in \mathbb{R}^3$, which does belong to the Euclidean space, is defined. This means that the error covariance in (5.30) used to find the error sigma points do not represent the quaternion itself—and the other states—but the angle vector. Then, a whole error sigma point obtained from (5.30) have the following form.

$$\delta\chi_k^i = \begin{bmatrix} \delta\boldsymbol{\vartheta}_k^i \\ \delta\boldsymbol{\omega}_k^i \\ \delta\mathbf{r}_k^i \\ \delta\mathbf{v}_k^i \end{bmatrix}, \quad i \in \{1, 2, \dots, 2n\} \quad (5.32)$$

Note that $\chi_k^i \in \mathbb{R}^n$ and $\delta\chi_k^i \in \mathbb{R}^{n-1}$. Now, $\delta\boldsymbol{\vartheta}_k^i$ can be transformed into an error quaternion $\delta\mathbf{q}_k^i$ using the exponential map [12].

$$\delta\mathbf{q}_k^i = e^{\delta\boldsymbol{\vartheta}_k^i} = \begin{bmatrix} \cos(\|\delta\boldsymbol{\vartheta}_k^i\|_2/2) \\ \frac{\delta\boldsymbol{\vartheta}_k^i}{\|\delta\boldsymbol{\vartheta}_k^i\|_2} \sin(\|\delta\boldsymbol{\vartheta}_k^i\|_2/2) \end{bmatrix} \quad (5.33)$$

The set of sigma points $\chi_k \in \mathbb{R}^{n \times (2n+1)}$, with $n = 13$, and using the notation in (5.30) and (5.34), is then built as

$$\chi_k = [\chi_k^0 \ \chi_k^1 \ \dots \ \chi_k^n \ \chi_k^{n+1} \ \dots \ \chi_k^{2n}], \quad (5.34a)$$

$$\chi_k^0 = \begin{bmatrix} \hat{\mathbf{q}}_k^+ \\ \hat{\boldsymbol{\omega}}_k^+ \\ \hat{\mathbf{r}}_k^+ \\ \hat{\mathbf{v}}_k^+ \end{bmatrix}, \quad (5.34b)$$

$$\chi_k^i = \begin{bmatrix} \delta\mathbf{q}_k^i \otimes \hat{\mathbf{q}}_k^+ \\ \hat{\boldsymbol{\omega}}_k^+ + \delta\boldsymbol{\omega}_k^i \\ \hat{\mathbf{r}}_k^+ + \delta\mathbf{r}_k^i \\ \hat{\mathbf{v}}_k^+ + \delta\mathbf{v}_k^i \end{bmatrix}, \quad i \in \{1, \dots, n\}, \quad (5.34c)$$

$$\chi_k^i = \begin{bmatrix} (\delta\mathbf{q}_k^i)^* \otimes \hat{\mathbf{q}}_k^+ \\ \hat{\boldsymbol{\omega}}_k^+ - \delta\boldsymbol{\omega}_k^i \\ \hat{\mathbf{r}}_k^+ - \delta\mathbf{r}_k^i \\ \hat{\mathbf{v}}_k^+ - \delta\mathbf{v}_k^i \end{bmatrix}, \quad i \in \{n+1, \dots, 2n\}, \quad (5.34d)$$

with $\hat{\mathbf{x}}_k^+ = [\hat{\mathbf{q}}_k^+ \ \hat{\boldsymbol{\omega}}_k^+ \ \hat{\mathbf{r}}_k^+ \ \hat{\mathbf{v}}_k^+]^T$. Each of the columns of the previous matrix represent a sigma point, and once obtained they are propagated in time by means of the discrete version of the dynamic models derived in Sec. 3, i.e. $\mathbf{x}_{k+1} = \mathbf{f}_d(k, \mathbf{x}_k, \mathbf{u}_k)$. Then the propagated sigma points χ_{k+1} are obtained.

$$\chi_{k+1}^i = \mathbf{f}_d(k, \chi_k^i, \mathbf{u}_k), \quad i \in \{0, 1, 2, \dots, 2n\} \quad (5.35)$$

In order to find the propagated prediction state $\hat{\mathbf{x}}_{k+1}^-$ from the propagated sigma points χ_{k+1} , a weighed average known as the uncenced transform (UT) is generally calculated as follows.

$$\hat{\mathbf{x}}_{k+1}^- = \sum_{i=0}^{2n} \bar{W}_i \chi_{k+1}^i, \quad (5.36)$$

and for the error covariance as

$$\mathbf{P}_{k+1}^- = \sum_{i=0}^{2n} W_i (\chi_{k+1}^i - \hat{\mathbf{x}}_{k+1}^-) (\chi_{k+1}^i - \hat{\mathbf{x}}_{k+1}^-)^T, \quad (5.37)$$

being the weight \bar{W}_i for the mean, and the weight W_i for the covariance. They are computed as shown below CITATION.

$$\bar{W}_0 = \frac{\lambda}{n + \lambda}, \quad (5.38a)$$

$$W_0 = \bar{W}_0 + (1 - \alpha^2 - \beta), \quad (5.38b)$$

$$\bar{W}_i = W_i = \frac{1}{2(n + \lambda)}, \quad i \in \{1, 2, \dots, 2n\}, \quad (5.38c)$$

The new scaling parameter β is used to introduce prior knowledge of how the state vector \mathbf{x}_k is distributed. Choosing $\beta = 2$ is optimal for Gaussian distributions [6].

Some issues arise, once again, while computing the sample mean (5.36) and covariance (5.37) due to the unit quaternion. The part of the sigma points χ_k involving the relative quaternion can not, at first, be added in the summation, as the property of unity is lost after the addition. Besides, it has been decided at the begining of the section that the covariance matrix \mathbf{P}_k is related to the angle vector $\boldsymbol{\theta}$, instead of the unit quaternion \mathbf{q} . Therefore, the covariance prediction \mathbf{P}_{k+1}^- should be derived with angle vectors sigma points instead of quaternion sigma points.

In [6], it is sought for an approximation to perform the UT for the sample mean using directly the unit quaternion, which solves the first problem. This approximation is found as a solution of a true optimization problem, and not as an *ad hoc* solution. The solution is

$$\hat{\mathbf{q}}_{k+1}^- = \frac{\sum_{i=0}^{2n} \bar{W}_i [\chi_{k+1}^i]_{\mathbf{q}}}{\left\| \sum_{i=0}^{2n} \bar{W}_i [\chi_{k+1}^i]_{\mathbf{q}} \right\|_2}, \quad (5.39)$$

where $[\chi_{k+1}^i]_{\mathbf{q}}$ takes only the quaternion part of each propagated sigma point. Note then, that the predicted states prior to the correction step is $\hat{\mathbf{x}}_{k+1}^- = [\hat{\mathbf{q}}_{k+1}^- \hat{\boldsymbol{\omega}}_{k+1}^- \hat{\mathbf{r}}_{k+1}^- \hat{\mathbf{r}}_{k+1}^-]^T$.

For the computation of the error covariance, the propagation of the error sigma points (5.30) —which are in terms of $\delta\boldsymbol{\vartheta}_k$ — is needed. Given that both, $\hat{\mathbf{x}}_{k+1}^+$ and $\boldsymbol{\chi}_{k+1}$, are now known, the error sigma points can be computed. If the propagated error sigma points are denoted as

$$\delta\boldsymbol{\chi}_{k+1}^i = \begin{bmatrix} \delta\boldsymbol{\vartheta}_{k+1}^i \\ \delta\boldsymbol{\omega}_{k+1}^i \\ \delta\mathbf{r}_{k+1}^i \\ \delta\mathbf{v}_{k+1}^i \end{bmatrix}, \quad i \in \{1, 2, \dots, 2n\}, \quad (5.40)$$

then each error state in the error sigma point can be computed as described hereafter. For the error angle $\delta\boldsymbol{\vartheta}_{k+1}^i$, the logarithmic map must be applied to the propagated error quaternion, $\delta\mathbf{q}_{k+1}^i$, which is derived as

$$\delta\mathbf{q}_{k+1}^i = [\boldsymbol{\chi}_{k+1}^i]_{\mathbf{q}} \otimes (\hat{\mathbf{q}}_{k+1}^-)^*, \quad i \in \{1, 2, \dots, 2n\}. \quad (5.41)$$

Then, the logarithmic map can be applied to $\delta\mathbf{q}_{k+1}^i$ as

$$\delta\boldsymbol{\vartheta}_{k+1}^i = \delta\mathbf{e}^i \delta\boldsymbol{\vartheta}^i, \quad (5.42)$$

where $\delta\mathbf{e}^i$ and $\delta\boldsymbol{\vartheta}^i$ are calculated as for \mathbf{e}^{ST} and $\boldsymbol{\vartheta}^{ST}$ in the star tracker's measurement equation (5.2), using $\delta\mathbf{q}_{k+1}^i$ instead of \mathbf{q}^{ST} . For the rest of the states, the error sigma point is basically

$$\delta\boldsymbol{\omega}_{k+1}^i = \hat{\boldsymbol{\omega}}_{k+1}^- - [\boldsymbol{\chi}_{k+1}^i]_{\boldsymbol{\omega}}, \quad (5.43a)$$

$$\delta\mathbf{r}_{k+1}^i = \hat{\mathbf{r}}_{k+1}^- - [\boldsymbol{\chi}_{k+1}^i]_{\mathbf{r}}, \quad i \in \{1, 2, \dots, 2n\}, \quad (5.43b)$$

$$\delta\mathbf{v}_{k+1}^i = \hat{\mathbf{v}}_{k+1}^- - [\boldsymbol{\chi}_{k+1}^i]_{\mathbf{v}}, \quad (5.43c)$$

where, as for $[\boldsymbol{\chi}_{k+1}^i]_{\mathbf{q}}$, the terms $[\boldsymbol{\chi}_{k+1}^i]_{\boldsymbol{\omega}}$, $[\boldsymbol{\chi}_{k+1}^i]_{\mathbf{r}}$ and $[\boldsymbol{\chi}_{k+1}^i]_{\mathbf{v}}$ denote the relative angular velocity, relative position and relative linear velocity parts of the error sigma points respectively. Now, the error sigma points (5.40) can be built, and with it the prediction of the error covariance matrix.

$$\mathbf{P}_{k+1}^- = \sum_{i=0}^{2n} W_i (\delta\boldsymbol{\chi}_{k+1}^i) (\delta\boldsymbol{\chi}_{k+1}^i)^T, \quad (5.44)$$

All the aforementioned procedure constitutes the *propagation* step in the UKF. Following the linear KF algorithm in Tab. 5.2, the subsequent step is to compute the *innovation* term $\Delta\hat{\mathbf{x}}_{k+1}$ which depends on the true measurement \mathbf{y}_{k+1} and the predicted measurement $\hat{\mathbf{y}}_{k+1}$. The predicted measurement is also computed using the sigma points, which are evaluated in the measurement equations obtaining the points $\mathbf{Y}_{k+1} \in \mathbb{R}^{p \times (2n+1)}$ and then using the UT to find $\hat{\mathbf{y}}_{k+1}$. The points \mathbf{Y}_{k+1} are derived as

$$\mathbf{Y}_{k+1}^i = h_d(\boldsymbol{\chi}_{k+1}^i), \quad i \in \{0, 1, 2, \dots, 2n\}. \quad (5.45)$$

where $h_d(\cdot)$ is the measurement equation derived at the beginning of this chapter. To find the measurement estimation $\hat{\mathbf{y}}_{k+1}$, the UT can be applied as in (5.36).

$$\hat{\mathbf{y}}_{k+1} = \sum_{i=0}^{2n} \bar{W}_i \mathbf{Y}_{k+1}^i. \quad (5.46)$$

When the satellite only relies on VBN, the values in \mathbf{Y}_k^i represent the projection of the 5 inner LEDs into the camera sensor according to the sigma points χ_{k+1}^i . Since the sigma points are deviations from the mean prediction, in one arbitrary step k the projection of the 5 inner LEDs, namely the vectors \mathbf{Y}_k^i , would look like follows.

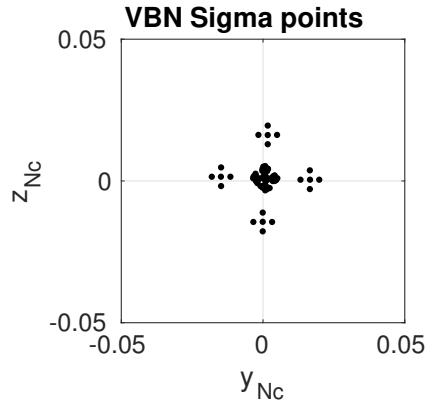


Figure 5.5: VBN measurement sigma points, when tracking the position of the 5-LED pattern from 2.5 [m] distance.

Gathering the true and the estimated measurement, the innovation term $\Delta\hat{\mathbf{y}}_{k+1}$ can be computed simply as their subtraction. That is,

$$\Delta\hat{\mathbf{y}}_{k+1} = \mathbf{y}_{k+1} - \hat{\mathbf{y}}_{k+1}. \quad (5.47)$$

The innovation term is then left multiplied by the *Kalman gain*, \mathbf{K}_{k+1} , which is computed based on the knowledge of the uncertainty at that time. Given that the uncertainty is modeled as covariance matrices, the following covariances are calculated.

$$\mathbf{P}_{k+1}^{yy} = \sum_{i=0}^{2n} W_i (\mathbf{Y}_{k+1}^i - \hat{\mathbf{y}}_{k+1}) (\mathbf{Y}_{k+1}^i - \hat{\mathbf{y}}_{k+1})^T + \Sigma_v \quad (5.48a)$$

$$\mathbf{P}_{k+1}^{xy} = \sum_{i=0}^{2n} W_i (\delta\chi_{k+1}^i) (\mathbf{Y}_{k+1}^i - \hat{\mathbf{y}}_{k+1})^T \quad (5.48b)$$

where $\mathbf{P}_k^{yy} \in \mathbb{R}^{p \times p}$ denote the measurement e covariance, and $\mathbf{P}_k^{xy} \in \mathbb{R}^{(n-1) \times p}$ denote the cross correlation between the states and the measurements. Joining these two pieces of information, the value of $\mathbf{K}_{k+1} \in \mathbb{R}^{(n-1) \times p}$ can be found with the following formula.

$$\mathbf{K}_{k+1} = \mathbf{P}_k^{xy} (\mathbf{P}_k^{yy})^{-1}, \quad (5.49)$$

Subsequently, the *measurement update* can finally be achieved using (5.49) and (5.47) as below,

$$\Delta \hat{\mathbf{x}}_{k+1} = \mathbf{K}_{k+1} \Delta \hat{\mathbf{y}}_{k+1} = \begin{bmatrix} \Delta \boldsymbol{\theta}_{k+1} \\ \Delta \boldsymbol{\omega}_{k+1} \\ \Delta \mathbf{r}_{k+1} \\ \Delta \mathbf{v}_{k+1} \end{bmatrix}. \quad (5.50)$$

Once again, notice that $\Delta \hat{\mathbf{x}}_{k+1}$ contains angle vectors $\Delta \boldsymbol{\theta}$, instead of quaternions $\Delta \mathbf{q}_{k+1}$. This leads to a last transformation using the exponential map, as in (5.33).

$$\Delta \mathbf{q}_{k+1} = e^{\Delta \boldsymbol{\theta}_{k+1}}. \quad (5.51)$$

Finally, the updated prediction can be written as

$$\hat{\mathbf{x}}_{k+1}^+ = \hat{\mathbf{x}}_{k+1}^- + \begin{bmatrix} \Delta \mathbf{q}_{k+1} \\ \Delta \boldsymbol{\omega}_{k+1} \\ \Delta \mathbf{r}_{k+1} \\ \Delta \mathbf{v}_{k+1} \end{bmatrix}, \quad (5.52)$$

and the updated error covariance can be computed as

$$\mathbf{P}_{k+1}^+ = \mathbf{P}_{k+1}^- - \mathbf{K}_{k+1} \mathbf{P}_{k+1}^{yy} \mathbf{K}_{k+1}^T. \quad (5.53)$$

A table is provided in the appendix, Tab. C.1, summarizing the complete UKF algorithm described above.

The following figures, Fig. 5.6 and Fig. 5.7, show a short simulation on how the estimation of the states provided by the UKF downgrades the performance of the controller before the navigation has converged. The simulation is conducted from the SK_0 to the SK_1 . This means, that both sensors, the camera and the star tracker, are available and fused in the UKF.

The estimations are initialized at random positions, which make the controllers to initially react in order to correct the deviations between the estimations and the references. This creates some initial disturbances that can be witnessed in the y and z components of the position \mathbf{r} . Therefore, it is interesting to introduce some method to detect when the convergence of the estimations have been achieved. After detecting it, then the motion in the x component of \mathbf{r} could be restarted in order not to compromise the mission.

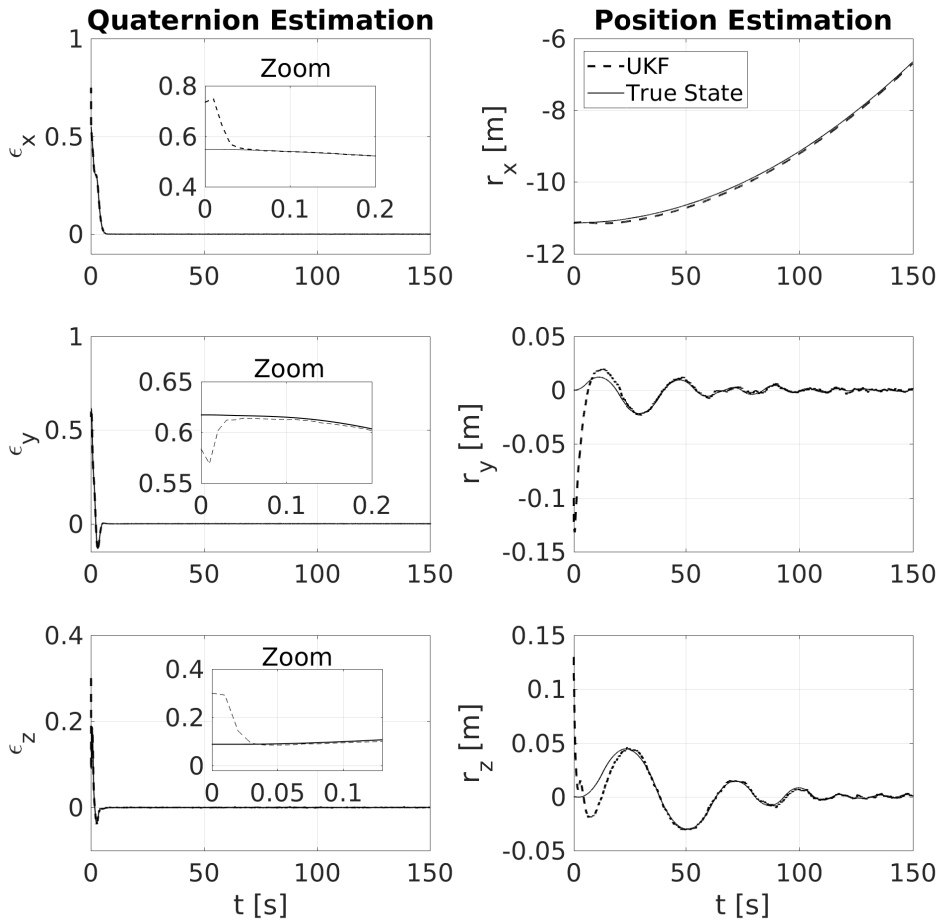


Figure 5.6: Estimation of the relative attitude \mathbf{q} and relative position \mathbf{r} , using the UKF as an state-estimator fusing measurements from the camera and the star-tracker. The simulation is then performed from the first to the second SK point.

Another fact that can be seen in Fig. 5.6 is the overshoot in the vector part of the quaternion, right before reaching the reference $[0 \ 0 \ 0]^T$. The explanation for this needs the help of the following figure, which shows the estimations of the states that are not directly measured, i.e. the relative angular velocity $\boldsymbol{\omega}$ and the relative linear velocity \mathbf{v} .

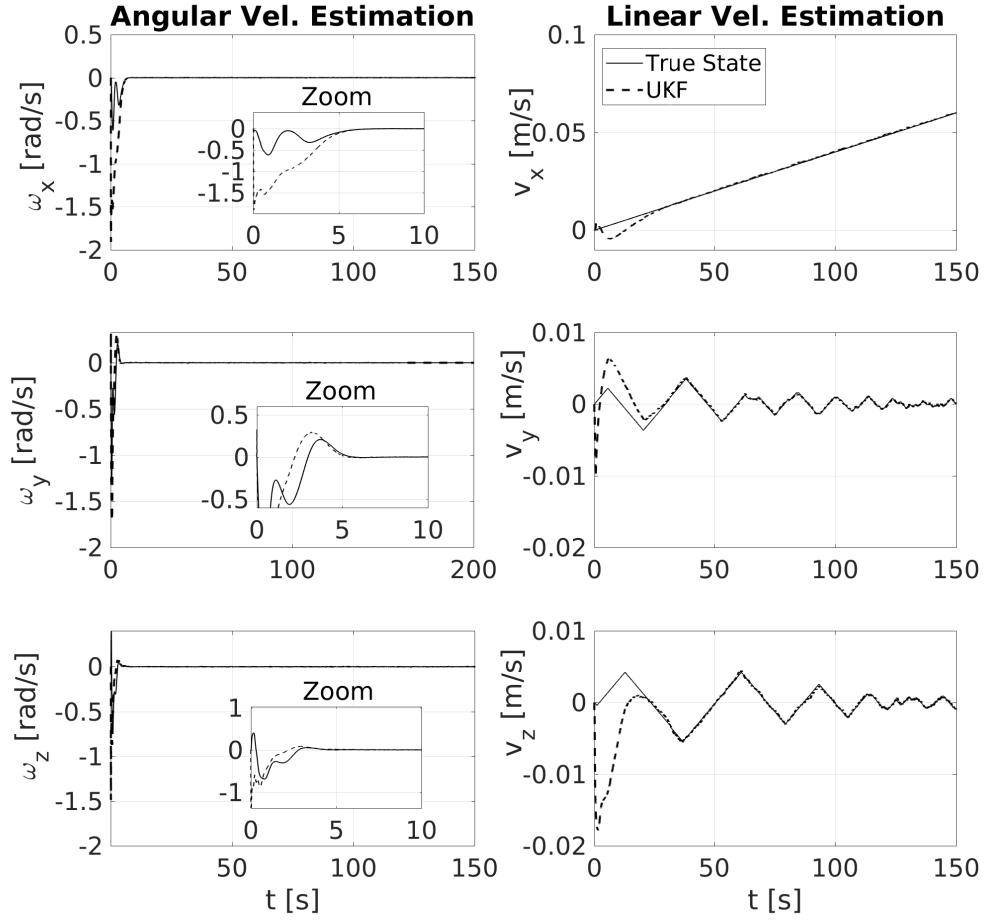


Figure 5.7: Estimation of the relative angular velocity ω and relative linear velocity \mathbf{v} , using the UKF as a state-estimator fusing measurements from the camera and the star-tracker. The simulation is then performed from the first to the second SK point.

The attitude controller behaves according to the sliding manifold defined in (4.7), which explicitly depends on the quaternion \mathbf{q} and the angular velocity ω . The overshoot occurs then, due to the error in the estimation at the initialization, before the navigation converges. The deviated UKF estimations are sent to the controllers, which derive a torque—in the case of the SMC—to bring the states to the reference. However, since the estimations and the true states are not the same at the beginning, the torques derived might deviate the true states even further. The reference can be achieved, obviously, once the navigation has converged, that is at ≈ 5 [s].

For a general overview of the performance of this navigation solution, a final simulation is conducted from SK_0 until docking is achieved. The figures presented hereafter show the errors of the estimations during the simulation. For the particular case of the position estimation, the navigation requirements are superposed.

This is shown in Fig. 5.8.

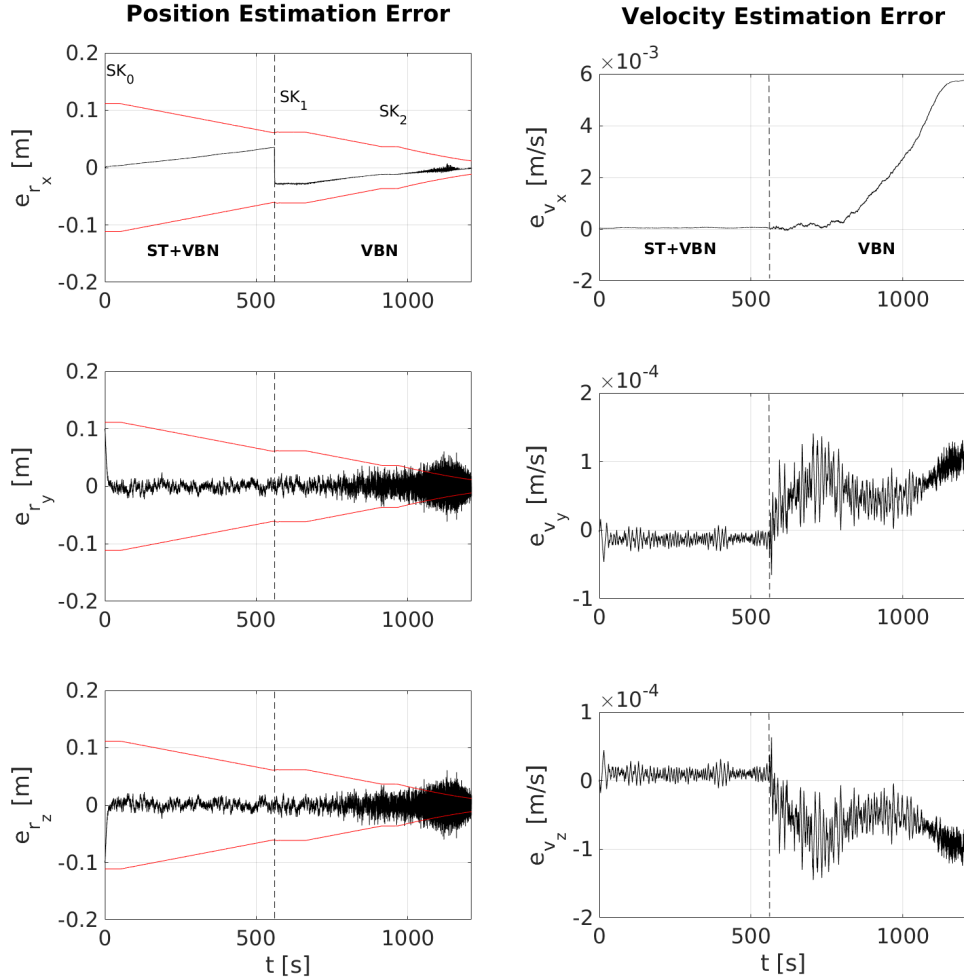


Figure 5.8: Position and velocity estimation errors throughout the Final Approach. The navigation requirements —1% of the range— are superposed on top of the position errors in red. The navigation solution employed is denoted and separated by a vertical dashed line, as well as the different SK points.

It has been noticed that the performance of the UKF is extremely sensitive to the covariance matrices Σ_w and Σ_v , which has made the tuning process a tedious task. Several tuning matrices have been tested, and the ones employed in this final simulation showed the best results. Although several things need to be remarked.

First, the x component of the position error in Fig. 5.8 slightly diverge with time during the initial approach. A similar behaviour can be seen in the components of the error velocity while relying only on VBN. This effect is thought to be caused by a poor tuning of the UKF, which might rely more on an inaccurate model rather than the

measurements. The inaccuracies of the model in the UKF are likely originated due to the ordinary differential equation (ODE) solver that has been employed, which differ from the solution given by the ODE solver used to simulate the *true* model. For the continuous *true* model, *ode45* has been used, whereas for the discrete propagation step in the UKF, it has been used *ode4*.

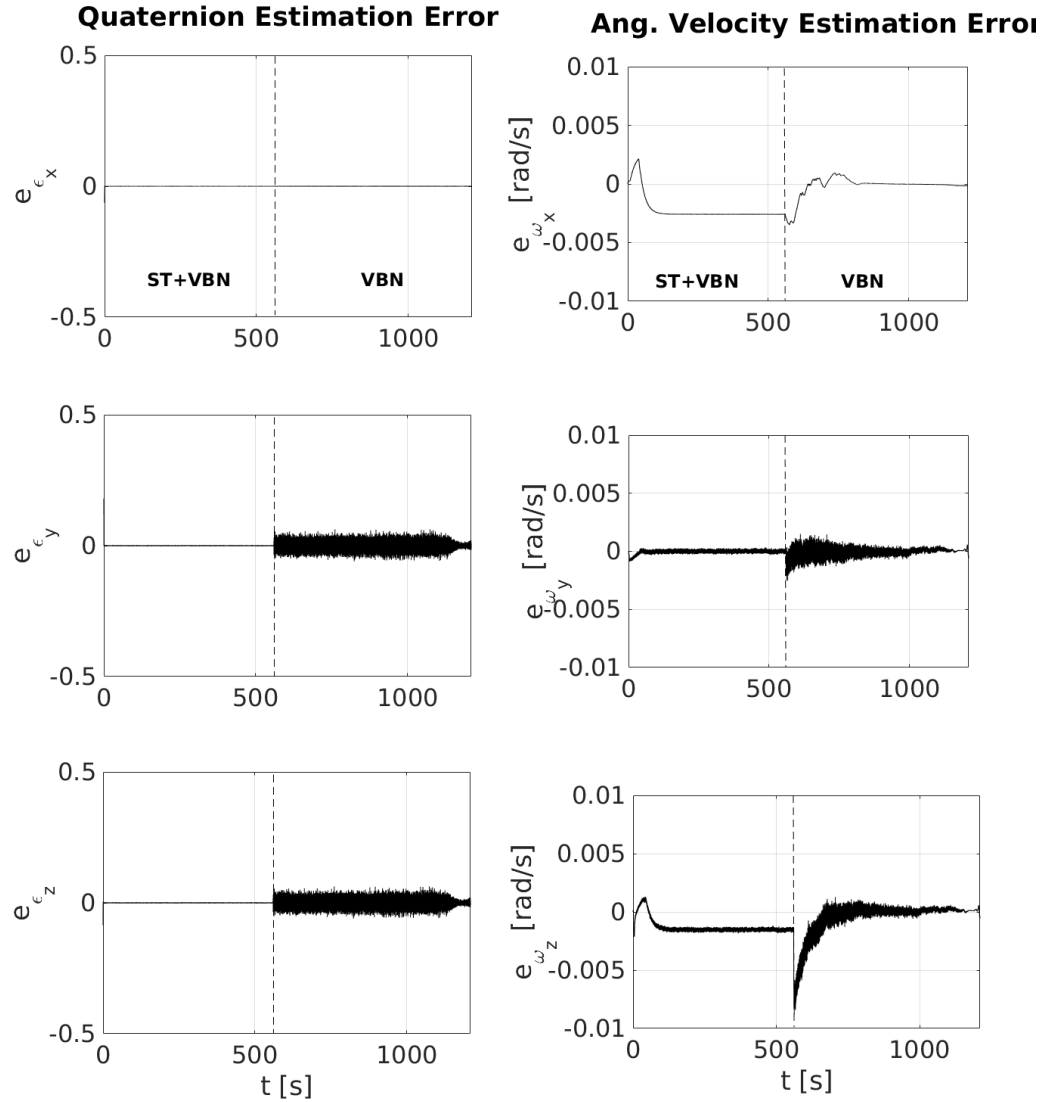


Figure 5.9: Position and velocity estimation errors throughout the Final Approach. The navigation requirements —1% of the range— are superposed on top of the position errors in red. The navigation solution employed is denoted and separated by a vertical dashed line, as well as the different SK points.

The most substantial feature from Fig. 5.8 though, is that the navigation requirements are always fulfilled. Despite the last seconds of simulation having an increase of noise in the y and z components of error position, the mean is clearly centered at 0. Moreover, the fact that the error in the positions is significantly low, it can be concluded that the velocity during approach maneuvers is very close to the required 1 [cm/s].

The increase of noise in the y and z components of error position can be read as an increase on the uncertainty. Notice, although, that the uncertainty of the attitude in Fig. 5.9 is reduced at the very last. This two facts might be related, given that the position and the attitude of the navigation port is coupled. This can be easily seen in Fig. 5.10

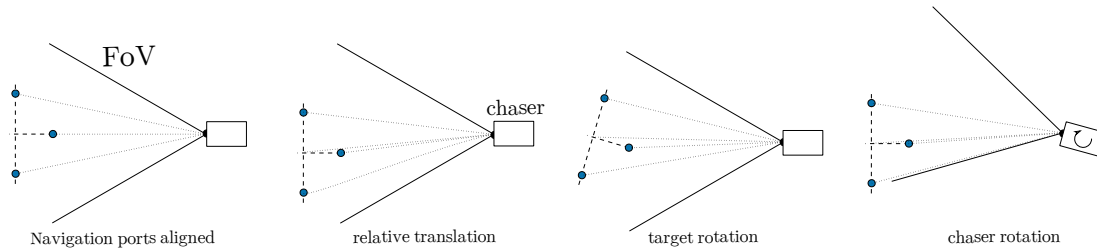


Figure 5.10: Rotation-translation coupling in the VBN.

Moreover, the uncertainty in the y and z components of the vector quaternion present more uncertainty than the x component. This makes sense, since most of the rotation motion takes place in the y and z axis.

One final remark is that this navigation solution does not provide the possibility of parameter-estimation, due to the fact that positions and orientations are measured, instead of the states related to the physical parameters worth to estimate their uncertainties.

5.3 Navigation Convergence Detection

During the final approach manoeuvre, two sensor handovers take place. The first one is right after the fly-around manoeuvre that places the chaser in the final approach axis, at 10 [m] from the target. This is the first SK point, where the handover to the vision —detecting the outer LED pattern— and star tracker navigation happens. The chaser must remain in the SK point until the camera's and star tracker's estimations converge. Once the estimations have converged, the motion towards the second SK point, which is placed on the final approach axis at 5 [m] from the target, starts. In the second SK point, the second sensor handover occurs, relying only on the camera sensor detecting now the inner LED pattern. When the VBN has converged, or it is made

sure that the estimations are not diverging as a consequence of the sensor transition, the motion towards the third SK is initiated. Recall that the third and last SK point is only placed as a last sanity check for the different subsystems, and there are no more sensor transitions from the third SK point until docking.

There are some facts, however, that need to be tackled. Firstly, it is obvious that the system has an endogenous input, i.e. the torque from the RW and the thrust from the RCP, which values are obtained by the control system according to the *known* —estimated— values of the states. The known and the true value differ, as it has been seen in the previous section, and the more they differ the worse are the inputs commanded by the control system to the actuators. In the initialization of the navigation system, the state-estimates can be significantly different from the true values momentarily, and commanding the control system to execute any manoeuvre before the navigation converge can be dangerous given that the behaviour can be totally unexpected. Then, it is desired to initialize the state-estimates in the desired values so that the control system acts minimally until navigation convergence is achieved.

In order to measure convergence of the estimations, the cumulative sum (CUSUM) algorithm is employed. This algorithm is vastly used in the literature as a fault-detection method [28, 32, 33]. Notwithstanding, in this work the CUSUM algorithm will be used to detect the opposite scenario, i.e. when the mean estimator of the state-estimates is not changing. Then, it can be concluded that the navigation system has converged in a more quantitative manner.

The CUSUM algorithm is based on the residual output signal, $\tilde{\mathbf{y}}_k$, which is obtained by subtracting the estimated output of the UKF to the real measurement, i.e. $\tilde{\mathbf{y}}_k \triangleq \mathbf{y}_k - \hat{\mathbf{y}}_k$. Then, two different situations can be hypothesized regarding the distribution of the residual output signal. These two hypothesis are

$$\tilde{\mathbf{y}}_k \stackrel{\text{iid}}{\sim} \begin{cases} \mathcal{N}(\mathbf{0}, \mathbf{P}_k^{\text{yy}}) & \mathcal{H}_0 : \text{Converged} \\ \mathcal{N}(\boldsymbol{\mu}_1, \mathbf{P}_k^{\text{yy}}) & \mathcal{H}_1 : \text{Not converged} \end{cases}, \quad (5.54)$$

which are based on the assumption that the residual output signal is Gaussian distributed, given that \mathbf{y}_k and $\hat{\mathbf{y}}_k$ are also Gaussian. Ideally, the residual output signal should have a $\mathbf{0}$ mean, which indicates that the expected value of the estimation error is $\mathbf{0}$. On the other side, if the expectation of the measurement estimation is $\boldsymbol{\mu}_1 \neq \mathbf{0}$, then it is fair to consider that the estimation of the states still have a significantly large error. Notice that the parameter $\boldsymbol{\mu}_1$ is a priori unknown.

Let the *log-likelihood* ratio at time k be define by the following expression [18].

$$s_k = \ln \left(\frac{p[\tilde{\mathbf{y}}_k | \mathcal{H}_1]}{p[\tilde{\mathbf{y}}_k | \mathcal{H}_0]} \right), \quad (5.55)$$

where $p[\tilde{\mathbf{y}}_k|\mathcal{H}_i]$ is the conditional probability of $\tilde{\mathbf{y}}_k$ given that the hypothesis \mathcal{H}_i is true, for $i = 0, 1$. The special characteristic of s_k is that, for an arbitrary $\tilde{\mathbf{y}}_k$, $s_k > 0$ if the hypothesis \mathcal{H}_1 is more likely to be true, and $s_k < 0$ otherwise. In case both hypothesis are equally probable to be true, i.e. $p[\tilde{\mathbf{y}}_k|\mathcal{H}_1] = p[\tilde{\mathbf{y}}_k|\mathcal{H}_0]$, then $s_k = 0$. If the log-likelihood ratio is consecutively added on its previous value, the cumulative sum S_k is defined as follows.

$$S_k = \sum_{i=0}^k s_i = S_{k-1} + s_k, \quad (5.56)$$

which can either increase, decrease or stay still. For the detection function not to constantly diverge, it is rewritten as

$$G_k = \max\{0, G_{k-1} + s_k\}. \quad (5.57)$$

With the CUSUM defined as in (5.57), it can be seen that G_k either increases —if \mathcal{H}_1 is more probable— or decreases until 0 —if \mathcal{H}_0 is more probable. This means that G_k can be used as a detector given that it only increases if the estimations have not converged, and decreases until 0 or converges asymptotically to a certain positive value if the estimations have converged.

To address the value of μ_1 , two alternatives can be tackled. The first one, used employed in this project, is to use the mean estimator. That is,

$$\mu_1 = \frac{1}{n} \sum_{i=0}^n \tilde{\mathbf{y}}_i. \quad (5.58)$$

Another approach, is to place two detectors with $\mu_1 = \pm\varepsilon$. Where ε is a small enough number let for tuning. The increase or decrease of its value determine how fast the convergence is detected. Although it can lead to early false detections.

In Fig. 5.11 it is shown how S_k increases at the beginning of the simulation due to the high inaccuracy of the navigation system. At the time the estimations start approaching the true states, S_k starts decreasing endlessly. Notice that the fact that S_k —and therefore, possibly also G_k — converge asymptotically to a value is due to the estimate of the mean in (5.58). Recall that if $\mu_1 = \mathbf{0}$, then the hypothesis \mathcal{H}_1 is equivalent to \mathcal{H}_0 and as a consequence $s_k = 0$. However, μ_1 will never be $\mathbf{0}$ in a finite amount of time, instead it will asymptotically tend to $\mathbf{0}$. To see the results, figure Fig. 5.12 is presented, where the states and their respective estimations are shown, as well as the moment when navigation convergence is assumed.

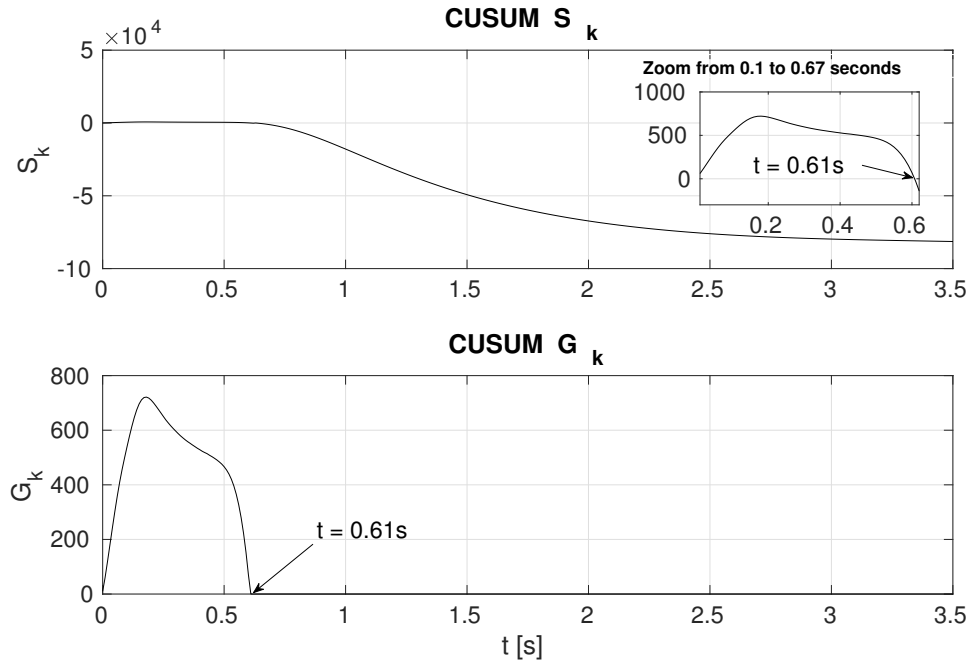


Figure 5.11: CUMSUM in a simulation defined as both, S_k and G_k . The time $t = 0.61$ [s] denotes when S_k crosses the x -axis.

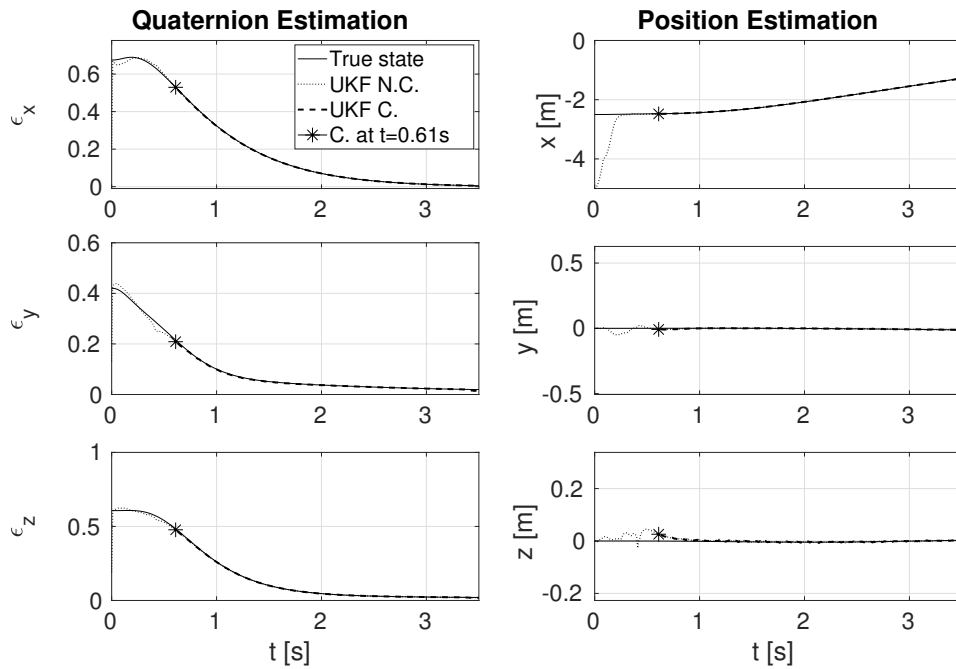


Figure 5.12: The true quaternion —only the vector part ϵ is shown— and the true position are represented with the continuous line. Their respective estimations are split in two segments: Not Converged (N.C.) and Converged (C.). The time when convergence is assumed is marked with an *.

As a special remark, note that transition towards the next SK point is not initiated until convergence is achieved. This can be seen in the motion of x in Fig. 5.12. Since the orientation does not add any risk while staying in the two first SK points, the attitude controller is allowed to give commands despite the navigation not having totally converged.

This method can be more conservative, if instead of using G_k as a detection function, S_k is used instead. Once S_k is stabilized, then it can be ensured with more confidence that the navigation has converged.

Closure

6

The final remarks regarding the results of this thesis is presented in this chapter. To conclude, possible future work and enhancements of this thesis are detailed.

6.1 Discussion

Several tasks of the GNC system have been tackled throughout this work, with the objective of contributing in the investigation of possible control and navigation solutions to perform the Final Approach of a RVD mission for micro-satellites. This has been done under some assumptions and simplifications that require to be revisited for the true implementation of the results presented here.

The RVD of CubeSats has not been attempted, yet further investigation and technology is required to be standardized. This is the miniaturized docking port and the vision-based navigation. Besides of this, the real challenge lies on the actual performance of the actuator and sensor devices on board of small satellites. However, the simulations show that RVD missions involving CubeSats might be feasible.

To carry out the simulations, a relative model have been successfully derived for the attitude and translation. The attitude representation has been based on quaternions, given the possibility of the chaser to dock with the target in any orientation. A translational model is derived for the relative position of the CoMs of the chaser and

target. Despite the interest lying on the relative position of the docking ports, it has been chosen to describe the CoMs instead due to the reduction of the state couplings and the decrease of computational complexity when deriving the linearized model. This could be done under the assumption of rigid body for the spacecraft and the moving components.

Two main cores of the GNC system have been deeply investigated, those are the control and the navigation systems. Robust and optimal strategies were sought for the control system. On the one hand, a quaternion-based sliding-mode control has proven to be a reliable choice given the robustness that has been shown. On the other hand, a model predictive control has been derived for its optimal characteristics. Fuel is a limiting factor, and optimal strategies such as MPC are highly appealing. However, it has not presented the level of robustness desired for this kind of mission where the control requirements are tight and a high degree of performance is expected. In addition, a simple reaction wheel momentum dumping control has also been derived to control the speed of the reaction wheels.

The second core, the navigation system, is grounded on strategies for relative navigation, rather than absolute. For this, a camera-type sensor and a set of LEDs have been placed in the chaser and the target respectively. This solution is founded on the one presented in [30], where two different LED patterns are strategically placed in the target's navigation port. In this work, the camera-type sensor on the chaser's navigation port tracks the centroid of the LEDs, and measurement equations are upgraded with the quaternion representation and used to infer the relative states. Despite the performance of this strategy lying within the navigation requirements, it is thought that the performance could be enhanced with proper tuning. Moreover, a navigation convergence detection has been also derived based on the CUSUM algorithm. This has been done given the disturbances produced by the controller itself acting when the navigation has not yet converged.

Finally, the strategies for the GNC system just described have been tested on simulation under the most relevant internal and external disturbances during Final Approach, and uncertainties regarding physical parameters and CoM location. However, it has been left for future work to test the performance of both systems working simultaneously.

6.2 Future Work

Different manners to improve and extend the work presented in this thesis, that have been thought by the author during its development.

- **Mathematical Model.** The model used and derived describes, as has been mentioned several times, to describe the relative position of the CoMs of the spacecrafts. This is, however, an alternative that has been chosen due to the lack of computational resources to calculate the Jacobians needed for the linearization of the model. It is, then, more accurate and correct to use a model that describe the relative position of the docking ports directly. More disturbances should also be added in order to complete the simulation environment, for instance the gravitational potential, solar radiation and disturbances produced by appendages. This would allow other stages of the RVD mission could be tackled. Another possibility is to study the RVD problem for elliptic orbits. Finally, modeling of the actuator systems are required, which would introduce more disturbances and another control layer. For example, replacing the 3-RW formation by a tetrahedral or pyramidal configuration which add failure robustness, and prevention against saturation.
- **Control System.** A more robust approach would be sought for the relative position control. Possibly a robust formulation of the MPC, but other options are also appealing. For instance, utilizing the MPC for path-planning instead, which can then be forwarded to another SMC track it as the reference. This would take the most powerful features of both controllers, by computing the optimal path while achieving robustness against disturbances and uncertainties. It is also interesting to introduce the attitude in the MPC problem, which would allow the add constraints to the angular position and velocity. Furthermore, extra control strategies are needed for the aforementioned model of the actuators, such as reaction wheels, the propulsion system and the magnetorquers. In addition, fault-detection and compensation algorithms for the sensors and actuators could also be tackled.
- **Navigation System.** The navigation solution presented is quite complex, and only a slight part have been investigated in this thesis. This strategy requires several previous work that have been taken for granted, due to the lack of time to develop all these required tasks. In [30], however, most of this prior work has already been done. This includes patten recognition and tracking to follow the LED formation. Calibration of the VBN. Assessment of the noise under different illumination conditions. Optimal LED wavelength, etc. Regarding the work presented here, the UKF show good results, although further tuning should be attempted. An important point left to test, is the performance of both systems combined—control and navigation—in the simulation, given that the navigation introduces disturbances to the control system.

Bibliography

- [1] NanoProp 6U — GomSpace. [Online]: https://gomspace.com/UserFiles/Subsystems/flyer/gomspace_nanoprop_cgp6_flyer.pdf, [Accessed: 2019-4-12].
- [2] NanoTorque GST-600 — GomSpace. [Online]: <https://gomspace.com/UserFiles/Subsystems/datasheet/gs-ds-nanotorque-gst-600-13.pdf>, [Accessed: 2019-4-12].
- [3] NanoTorque GSW-600 — GomSpace. [Online]: <https://gomspace.com/UserFiles/Subsystems/datasheet/gs-ds-nanotorque-gsw-600-20.pdf>, [Accessed: 2019-4-12].
- [4] D. N. Amanor, W. W. Edmonson, and F. Afghah. Intersatellite communication system based on visible light. *IEEE Transactions on Aerospace and Electronic Systems*, 54(6):2888–2899, Dec 2018.
- [5] G. Avanzini and F. Giulietti. Magnetic detumbling of a rigid spacecraft. *Journal of Guidance, Control, and Dynamics*, 35:1326–1334, 07 2012.
- [6] Y.-J. Cheon and J.-H. Kim. Unscented filtering in a unit quaternion space for spacecraft attitude estimation. pages 66 – 71, 07 2007.
- [7] C. Christian Liebe. Star trackers for attitude determination. *Aerospace and Electronic Systems Magazine, IEEE*, 10:10 – 16, 07 1995.
- [8] X. Chu, J. Zhang, S. Lu, Y. Zhang, and Y. Sun. Optimised collision avoidance for an ultra-close rendezvous with a failed satellite based on the gauss pseudospectral method. *Acta Astronautica*, 128:363 – 376, 2016.
- [9] J. L. Crassidis and F. L. Markley. Unscented filtering for spacecraft attitude estimation. *Journal of Guidance, Control, and Dynamics*, 26(4):536–542, 2003.
- [10] S. D’Amico and O. Montenbruck. Proximity operations of formation flying spacecraft using an eccentricity/inclination vector separation. *AIAA Journal of Guidance, Control and Dynamics*, 01 2005.
- [11] S. Di Cairano, H. Park, and I. Kolmanovsky. Model predictive control approach for guidance of spacecraft rendezvous and proximity maneuvering. *International Journal of Robust and Nonlinear Control*, 22(12):1398–1427, 2012.
- [12] J. Diebel. Representing attitude: Euler angles, unit quaternions, and rotation vectors. *Stanford University*, 2006.

- [13] D.M.Gibbon and C. Underwood. Low cost butane propulsion systems for small spacecraft. *15th AIAA, USU Conference on Small Satellites*.
- [14] W. Fehse. *Automated Rendezvous and Docking of Spacecraft*. Cambridge Aerospace Series. Cambridge University Press, 2003.
- [15] J. R. Forbes. Fundamentals of spacecraft attitude determination and control [bookshelf]. *IEEE Control Systems Magazine*, 35(4):56–58, Aug 2015.
- [16] J. Ge, J. Zhao, and J. Yuan. A novel guidance strategy for autonomously approaching a tumbling target. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 232(5):861–871, 2018.
- [17] V. Gorev, A. Pelemeshko, A. M. Zadorozhny, and A. Sidorchuk. Thermal deformation of 3u cubesat in low earth orbit. *MATEC Web of Conferences*, 158:01013, 01 2018.
- [18] P. Granjon. The cusum algorithm - a small review. 06 2013.
- [19] G. Hintz. *Orbital Mechanics and Astrodynamics: Techniques and Tools for Space Missions*. Springer International Publishing, 2015.
- [20] G. R. Hintz. *Spacecraft Rendezvous*, pages 223–241. Springer International Publishing, Cham, 2015.
- [21] F. J. Franquiz, P. Edwards, B. Udrea, M. Nayak, and T. Pueschl. Attitude determination and control system design for a 6u cubesat for proximity operations and rendezvous. 08 2014.
- [22] S. J. Julier and J. K. Uhlmann. Unscented filtering and nonlinear estimation. In *PROCEEDINGS OF THE IEEE*, pages 401–422, 2004.
- [23] H. K. Khalil. *Nonlinear systems; 3rd ed.* Prentice-Hall, Upper Saddle River, NJ, 2002.
- [24] R. Kristiansen, P. J. Nicklasson, and J. T. Gravdahl. Quaternion-based backstepping control of relative attitude in a spacecraft formation. In *Proceedings of the 45th IEEE Conference on Decision and Control*, pages 5724–5729, Dec 2006.
- [25] F. Liao, H. Ji, and Y. Xie. A nearly optimal control for spacecraft rendezvous with constrained controls. *Transactions of the Institute of Measurement and Control*, 38(7):832–845, 2016.
- [26] J. Maciejowski. *Predictive Control with Constraints*. Prentice Hall, England., 2002.
- [27] F. L. Markley and J. L. Crassidis. *Fundamentals of Spacecraft Attitude Determination and Control*. Springer, 2014.

- [28] V. M. D. Oca, D. R. Jeske, Q. Zhang, C. Rendon, and M. Marvasti. A cusum change-point detection algorithm for non-stationary sequences with application to data network surveillance. *Journal of Systems and Software*, 83(7):1288 – 1297, 2010. SPLC 2008.
- [29] M. Okasha and B. Newman. Relative motion guidance, navigation and control for autonomous orbital rendezvous. *Journal of Aerospace Technology and Management*, 6:301–318, 09 2014.
- [30] C. S. PIRAT. *Guidance, Navigation and Control for Autonomous Rendezvous and Docking of Nano-Satellites*. PhD thesis, 2018.
- [31] M. Rubagotti, A. Estrada, F. Castanos, A. Ferrara, and L. Fridman. Integral sliding mode control for nonlinear systems with matched and unmatched perturbations. *IEEE Transactions on Automatic Control*, 56(11):2699–2704, Nov 2011.
- [32] M. Severo and J. Gama. Change detection with kalman filter and cusum. In L. Todorovski, N. Lavrač, and K. P. Jantke, editors, *Discovery Science*, pages 243–254, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.
- [33] M. B. Shams, H. Budman, and T. Duever. Fault detection using cusum based techniques with application to the tennessee eastman process. *IFAC Proceedings Volumes*, 43(5):109 – 114, 2010. 9th IFAC Symposium on Dynamics and Control of Process Systems.
- [34] The CubeSat Program, Cal Poly SLO. 6U CubeSat Design Specification PROVISIONAL. 2016.
- [35] E. Thébault, C. C. Finlay, C. D. Beggan, P. Alken, J. Aubert, O. Barrois, F. Bertrand, T. Bondar, A. Boness, L. Brocco, E. Canet, A. Chambodut, A. Chulliat, P. Coïsson, F. Civet, A. Du, A. Fournier, I. Fratter, N. Gillet, B. Hamilton, M. Hamoudi, G. Hulot, T. Jager, M. Korte, W. Kuang, X. Lalanne, B. Langlais, J.-M. Léger, V. Lesur, F. J. Lowes, S. Macmillan, M. Manda, C. Manoj, S. Maus, N. Olsen, V. Petrov, V. Ridley, M. Rother, T. J. Sabaka, D. Saturnino, R. Schachtschneider, O. Sirol, A. Tangborn, A. Thomson, L. Tøffner-Clausen, P. Vigneron, I. Wardinski, and T. Zvereva. International geomagnetic reference field: the 12th generation. *Earth, Planets and Space*, 67(1):79, May 2015.
- [36] B. Tweddle and A. Saenz-Otero. Relative computer vision-based navigation for small inspection spacecraft. *Journal of Guidance, Control, and Dynamics*, 38:969–977, 01 2015.
- [37] R. Wisniewski. *Satellite Attitude Control Using Only Electromagnetic Actuation*. PhD thesis, 1997.

Dynamic Models



A.1 Equation of Motion of the CoMs in the Obital Frame

Let $\mathbf{r}_O^{\mathcal{T}C} \in \mathbb{R}^3$ be defined as the distance from the origin of the target body frame to the origin of the chaser body frame, i.e. from the CoM of the target to the CoM of the chaser, expressed in the target orbital frame. And $\dot{\mathbf{r}}_O^{\mathcal{T}C} \in \mathbb{R}^3$ the time-derivative of $\mathbf{r}_O^{\mathcal{T}C}$. These two vectors are the controlled variables, along with $\mathbf{q}_{\mathcal{T}}^C$ and $\omega_C^{\mathcal{T}C}$, in the subsequent chapter.

In the inertial frame \mathcal{F}_I , the relative position vector $\mathbf{r}_I^{\mathcal{T}C} \in \mathbb{R}^3$ is

$$\mathbf{r}_I^{\mathcal{T}C} = \mathbf{r}_I^{IC} - \mathbf{r}_I^{IT}, \quad (\text{A.1})$$

where $\mathbf{r}_I^{IC}, \mathbf{r}_I^{IT} \in \mathbb{R}^3$ are the distance from the CoM of the Earth to the CoM of the chaser and the target respectively. The acceleration vectors in the inertial frame are straightforward, according to Newton's gravitational law.

$$\ddot{\mathbf{r}}_I^{\mathcal{T}C} = \ddot{\mathbf{r}}_I^{IC} - \ddot{\mathbf{r}}_I^{IT} = -\mu \frac{\mathbf{r}_I^{IC}}{\|\mathbf{r}_I^{IC}\|_2^3} + \mu \frac{\mathbf{r}_I^{IT}}{\|\mathbf{r}_I^{IT}\|_2^3} + \frac{\mathbf{f}_I^{th}}{m_c}. \quad (\text{A.2})$$

In (A.2), $m_c \in \mathbb{R}$ is the mass of the chaser and $\mathbf{f}_I^{th} \in \mathbb{R}^3$ is the reaction force that the thrusters apply to the chaser, expressed in the inertial frame.

It is desired to express the relative dynamics in the target orbital frame, \mathcal{F}_O , in which equation (A.1) transforms to

$$\mathbf{r}_O^{\mathcal{T}C} = \mathbf{R}_I^O \mathbf{r}_I^{\mathcal{T}C}, \quad (\text{A.3})$$

where the matrix \mathbf{R}_I^O is defined as the transformation matrix from \mathcal{F}_I to \mathcal{F}_O , and can be found using the orbital parameters, although it is not needed here. The time-derivative of $\mathbf{r}_O^{\mathcal{T}C}$ can then be obtained by differentiating the previous equation w.r.t. time. Following the chain rule, that is,

$$\dot{\mathbf{r}}_O^{\mathcal{T}C} = -[\omega_O^{IO} \times] \mathbf{R}_I^O \mathbf{r}_I^{\mathcal{T}C} + \mathbf{R}_I^O \dot{\mathbf{r}}_I^{\mathcal{T}C} \quad (\text{A.4})$$

The dynamics are obtained by differentiating one more time, being then

$$\ddot{\mathbf{r}}_O^{\mathcal{T}C} = [\omega_O^{IO} \times][\omega_O^{IO} \times] \mathbf{R}_I^O \mathbf{r}_I^{\mathcal{T}C} - 2[\omega_O^{IO} \times] \mathbf{R}_I^O \dot{\mathbf{r}}_I^{\mathcal{T}C} + \mathbf{R}_I^O \ddot{\mathbf{r}}_I^{\mathcal{T}C}. \quad (\text{A.5})$$

From equation (A.4), term $\mathbf{R}_I^O \dot{\mathbf{r}}_I^{\mathcal{T}C}$ can be isolated and replaced in (A.5), which leads to

$$\ddot{\mathbf{r}}_O^{\mathcal{T}C} = -[\omega_O^{IO} \times][\omega_O^{IO} \times] \mathbf{R}_I^O \mathbf{r}_I^{\mathcal{T}C} - 2[\omega_O^{IO} \times] \dot{\mathbf{r}}_O^{\mathcal{T}C} + \mathbf{R}_I^O \ddot{\mathbf{r}}_I^{\mathcal{T}C}. \quad (\text{A.6})$$

Replacing equation (A.2) into (A.6), finally yields to the dynamic equation.

$$\begin{aligned} \ddot{\mathbf{r}}_O^{\mathcal{T}C} = & -[\omega_O^{IO} \times][\omega_O^{IO} \times] \mathbf{R}_I^O \mathbf{r}_I^{\mathcal{T}C} - 2[\omega_O^{IO} \times] \dot{\mathbf{r}}_O^{\mathcal{T}C} \\ & + \mathbf{R}_I^O \left(-\mu \frac{\mathbf{r}_I^{\mathcal{T}C}}{\|\mathbf{r}_I^{\mathcal{T}C}\|_2^3} + \mu \frac{\mathbf{r}_I^{\mathcal{IT}}}{\|\mathbf{r}_I^{\mathcal{IT}}\|_2^3} + \frac{\mathbf{f}_I^{th}}{m_c} \right). \end{aligned} \quad (\text{A.7})$$

Arranging the terms, the previous equation can be expressed as a function of the relative position vector $\mathbf{r}_O^{\mathcal{T}C}$. Notice that since the target orbital frame, \mathcal{F}_O , and the target body frame, \mathcal{F}_T , are centered in the CoM of the target, this means that $\mathbf{r}_O^{\mathcal{IT}} = \mathbf{r}_O^{IO} = [0 \ 0 \ -r_{LEO}]^T$, and is constant as circular orbits are assumed.

$$\begin{aligned} \ddot{\mathbf{r}}_O^{\mathcal{T}C} = & -[\omega_O^{IO} \times][\omega_O^{IO} \times] \mathbf{r}_O^{\mathcal{T}C} - 2[\omega_O^{IO} \times] \dot{\mathbf{r}}_O^{\mathcal{T}C} \\ & - \mu \frac{\mathbf{r}_O^{\mathcal{T}C} + \mathbf{r}_O^{IO}}{\|\mathbf{r}_O^{\mathcal{T}C} + \mathbf{r}_O^{IO}\|_2^3} + \mu \frac{\mathbf{r}_O^{IO}}{\|\mathbf{r}_O^{IO}\|_2^3} + \frac{\mathbf{f}_O^{th}}{m_c}. \end{aligned} \quad (\text{A.8})$$

Since the thruster reference frame is assumed to be aligned with the chaser body frame, it is of interest to change the coordinates of \mathbf{f}_O^{th} . In order to express the thrust in the chaser body frame, the following transformation should be applied.

$$\mathbf{f}_O^{th} = (\mathbf{R}_T^C \mathbf{R}_O^T)^T \mathbf{f}_C^{th}, \quad (\text{A.9})$$

where \mathbf{f}_C^{th} is the thrust force expressed in the chaser body frame. The rotational matrices \mathbf{R}_T^C and \mathbf{R}_O^T are obtained from the relative and absolute attitude motion respectively. Following the notation for the relative attitude, the controlled variable \mathbf{r}_O^{TC} will be denoted as \mathbf{r} , and its time derivative as \mathbf{v} . Therefore, the relative model for the CoM in \mathcal{F}_O is

$$\dot{\mathbf{r}} = \mathbf{v} \quad (\text{A.10a})$$

$$\begin{aligned} \dot{\mathbf{v}} = & -[\boldsymbol{\omega}_O^{IO} \times][\boldsymbol{\omega}_O^{IO} \times]\mathbf{r} - 2[\boldsymbol{\omega}_O^{IO} \times]\mathbf{v} \\ & - \mu \frac{\mathbf{r} + \mathbf{r}_O^{IO}}{\|\mathbf{r} + \mathbf{r}_O^{IO}\|_2^3} + \mu \frac{\mathbf{r}_O^{IO}}{\|\mathbf{r}_O^{IO}\|_2^3} + \frac{1}{m_c} (\mathbf{R}_T^C \mathbf{R}_O^T)^T \mathbf{f}_C^{th}. \end{aligned} \quad (\text{A.10b})$$

Sloshing Model

B

The *Jacobian* matrices, \mathbf{A}^{sl} , \mathbf{B}^{sl} and \mathbf{C}^{sl} , derived for the linearized sloshing model are provided by [30].

$$\mathbf{A}^{sl} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ \omega_y^2 - \frac{k_s}{m_s} + \omega_z^2 & -\omega_x\omega_y & -\omega_x\omega_z & -\frac{c_s}{m_s} & 2\omega_z & -2\omega_y \\ -\omega_x\omega_y & \omega_x^2 - \frac{k_s}{m_s} + \omega_z^2 & -\omega_y\omega_z & -2\omega_z & -\frac{c_s}{m_s} & 2\omega_x \\ -\omega_x\omega_z & -\omega_y\omega_z & \omega_x^2 - \frac{k_s}{m_s} + \omega_y^2 & 2\omega_y & -2\omega_x & -\frac{c_s}{m_s} \end{bmatrix} \quad (\text{B.1a})$$

$$\mathbf{B}^{sl} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & -r_{\mathcal{B},z}^{\mathcal{BS}} & r_{\mathcal{B},y}^{\mathcal{BS}} \\ 0 & 1 & 0 & r_{\mathcal{B},z}^{\mathcal{BS}} & 0 & -r_{\mathcal{B},x}^{\mathcal{BS}} \\ 0 & 0 & 1 & -r_{\mathcal{B},y}^{\mathcal{BS}} & r_{\mathcal{B},x}^{\mathcal{BS}} & 0 \end{bmatrix} \quad (\text{B.1b})$$

$$\mathbf{C}^{sl} = \begin{bmatrix} k_s & 0 & 0 & c_s & 0 & 0 \\ 0 & k_s & 0 & 0 & c_s & 0 \\ 0 & 0 & k_s & 0 & 0 & c_s \\ 0 & -k_s r_{\mathcal{B},z}^{\mathcal{BS}} & k_s r_{\mathcal{B},y}^{\mathcal{BS}} & 0 & -c_s r_{\mathcal{B},z}^{\mathcal{BS}} & c_s r_{\mathcal{B},y}^{\mathcal{BS}} \\ k_s r_{\mathcal{B},z}^{\mathcal{BS}} & 0 & -k_s r_{\mathcal{B},x}^{\mathcal{BS}} & c_s r_{\mathcal{B},z}^{\mathcal{BS}} & 0 & -c_s r_{\mathcal{B},x}^{\mathcal{BS}} \\ -k_s r_{\mathcal{B},y}^{\mathcal{BS}} & k_s r_{\mathcal{B},x}^{\mathcal{BS}} & 0 & -c_s r_{\mathcal{B},y}^{\mathcal{BS}} & c_s r_{\mathcal{B},x}^{\mathcal{BS}} & 0 \end{bmatrix} \quad (\text{B.1c})$$

Uncented Kalman Filter C

C.1 Cholesky Decomposition

As stated in the main report, given a matrix $\mathbf{P} \in \mathbb{S}_{>0}$, the Cholesky decomposition aims to find a matrix \mathbf{L} of the same dimensions such that

$$\mathbf{P} = \mathbf{L}\mathbf{L}^T. \quad (\text{C.1})$$

It is denoted as \mathbf{L} since it is a lower-triangular matrix of the following form.

$$\mathbf{L} = \begin{bmatrix} l_{11} & 0 & 0 & \dots & 0 \\ l_{21} & l_{22} & 0 & \dots & 0 \\ l_{31} & l_{32} & l_{33} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & l_{n3} & \dots & l_{nn} \end{bmatrix}. \quad (\text{C.2})$$

If the given matrix \mathbf{P} is defined as

$$\mathbf{P} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & \dots & p_{1n} \\ p_{21} & p_{22} & p_{23} & \dots & p_{2n} \\ p_{31} & p_{32} & p_{33} & \dots & p_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p_{n1} & p_{n2} & p_{n3} & \dots & p_{nn} \end{bmatrix}, \quad (\text{C.3})$$

Then the elements of the lower-triangular matrix can be obtained with the following summations.

$$l_{ki} = p_{ki} - \sum_{j=1}^{i-1} l_{ij} l_{kj}, \quad k, i \in \{1, \dots, n\}, \quad k \neq i \quad (\text{C.4a})$$

$$l_{kk} = \sqrt{p_{kk} - \sum_{j=1}^{k-1} l_{kj}^2}, \quad k \in \{1, \dots, n\} \quad (\text{C.4b})$$

C.2 UKF algorithm

Uncented KF algorithm *Part 1*

Initialization $\hat{\mathbf{x}}_0 = \mathbf{E}[\mathbf{x}_0]$
 $\mathbf{P}_0 = \mathbf{E}[(\mathbf{x}_0 - \hat{\mathbf{x}}_0)(\mathbf{x}_0 - \hat{\mathbf{x}}_0)^T]$
 $\bar{W}_0 = \frac{\lambda}{n+\lambda},$
 $W_0 = \bar{W}_0 + (1 - \alpha^2 - \beta),$
 $\bar{W}_i = W_i = \frac{1}{2(n+\lambda)}, \quad i \in \{1, 2, \dots, 2n\}$

Propagation $\delta \chi_k^i = \left(\sqrt{(n+\lambda)(\mathbf{P}_k^+ + \Sigma_w)} \right)_i, \quad i \in \{1, \dots, 2n\}.$
 $\chi_k = [\chi_k^0 \ \chi_k^1 \ \dots \ \chi_k^n \ \chi_k^{n+1} \ \dots \ \chi_k^{2n}],$ with (5.34b), (5.34c) and (5.34d).
 $\chi_{k+1}^i = f_d(k, \chi_k^i, \mathbf{u}_k), \quad i \in \{0, 1, \dots, 2n\}.$
 $\mathbf{q}_{k+1}^- = \frac{\sum_{i=0}^{2n} \bar{W}_i [\chi_{k+1}^i]_{\mathbf{q}}}{\left\| \sum_{i=0}^{2n} \bar{W}_i [\chi_{k+1}^i]_{\mathbf{q}} \right\|}, \quad \omega_{k+1}^- = \sum_{i=0}^{2n} \bar{W}_i [\chi_{k+1}^i]_{\omega}$
 $\mathbf{r}_{k+1}^- = \sum_{i=0}^{2n} \bar{W}_i [\chi_{k+1}^i]_{\mathbf{r}}, \quad \mathbf{v}_{k+1}^- = \sum_{i=0}^{2n} \bar{W}_i [\chi_{k+1}^i]_{\mathbf{v}}$
 $\hat{\mathbf{x}}_{k+1}^- = [\mathbf{q}_{k+1}^- \ \omega_{k+1}^- \ \mathbf{r}_{k+1}^- \ \mathbf{v}_{k+1}^-]^T.$
 $\delta \chi_{k+1}^i = [\delta \mathbf{q}_{k+1}^i \ \delta \omega_{k+1}^i \ \delta \mathbf{r}_{k+1}^i \ \delta \mathbf{v}_{k+1}^i]^T, \text{ with (5.41), (5.42) and (5.43),}$
 $\mathbf{P}_{k+1}^- = \sum_{i=0}^{2n} W_i (\delta \chi_{k+1}^i) (\delta \chi_{k+1}^i)^T$

| Uncented KF algorithm <i>Part 2</i> | |
|-------------------------------------|---|
| Innovation | $\mathbf{Y}_{k+1}^i = h_d(\chi_{k+1}^i), \quad i \in \{0, 1, 2, \dots, 2n\}$ $\hat{\mathbf{y}}_{k+1} = \sum_{i=0}^{2n} \bar{W}_i \mathbf{Y}_{k+1}^i$ $\Delta \hat{\mathbf{y}}_{k+1} = \mathbf{y}_{k+1} - \hat{\mathbf{y}}_{k+1}$ |
| Kalman Gain | $\mathbf{P}_{k+1}^{yy} = \sum_{i=0}^{2n} W_i (\mathbf{Y}_{k+1}^i - \hat{\mathbf{y}}_{k+1}) (\mathbf{Y}_{k+1}^i - \hat{\mathbf{y}}_{k+1})^T + \Sigma_v$ $\mathbf{P}_{k+1}^{xy} = \sum_{i=0}^{2n} W_i (\delta \chi_{k+1}^i) (\mathbf{Y}_{k+1}^i - \hat{\mathbf{y}}_{k+1})^T$ $\mathbf{K}_{k+1} = \mathbf{P}_{k+1}^{xy} (\mathbf{P}_{k+1}^{yy})^{-1}$ |
| Update | $\Delta \hat{\mathbf{x}}_{k+1} = \mathbf{K}_{k+1} \Delta \hat{\mathbf{y}}_{k+1} = [\Delta \boldsymbol{\theta}_{k+1} \Delta \boldsymbol{\omega}_{k+1} \Delta \mathbf{r}_{k+1} \Delta \mathbf{v}_{k+1}]^T,$ $\hat{\mathbf{x}}_{k+1}^+ = \hat{\mathbf{x}}_{k+1}^- + \Delta \hat{\mathbf{x}}, \text{ with } \Delta \boldsymbol{\theta}_{k+1} \rightarrow \Delta \mathbf{q}_{k+1} \text{ using (5.2)}$ $\mathbf{P}_{k+1}^+ = \mathbf{P}_{k+1}^- - \mathbf{K}_{k+1} \mathbf{P}_{k+1}^{yy} \mathbf{K}_{k+1}^T$ |

Table C.1: Uncented Kalman filter algorithm for nonlinear discrete-time systems with a state based on quaternions.