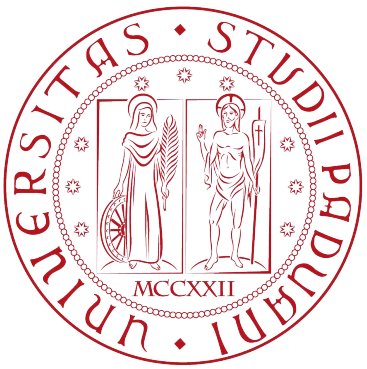


Università degli Studi di Padova
dipartimento di ingegneria industriale

Laurea magistrale in ingegneria aerospaziale

Autonomous rendezvous and docking maneuvers with Model Predictive Control strategies

Relatore: Dott. Valmorbida Andrea

Correlatore: Prof. Lorenzini Enrico

Laureando: Dennis Mantovani

Matricola: 1242271

Anno Accademico 2021-2022

Abstract

Since the beginning of space era, rendezvous and docking maneuvers have been of great importance for the success of numerous missions. The operation of meeting two or more space vehicles in orbit is a need for many missions which require the assembly or supply of orbital platforms, debris removal, or sample return for interplanetary missions. Such maneuvers can be performed automatically in a feedback way, with numerous advantages over manual control. The objective of this thesis is to present the Model Predictive Control (MPC), an advanced controller, for rendezvous and docking maneuvers between two cooperative satellites in orbit and compare its behaviors and performances with those of a classical PID controller. After the description of the operating principle of MPC and PID control strategies and the dynamics equations of the relative motion between satellites in orbit, a realistic rendezvous and docking scenario is considered. The scenario involves a 3U CubeSat performing autonomously the final approach to a target orbiting station along the V-bar direction, and includes some of the typical environmental disturbances at LEO orbits (differential drag and J2). A software developed in Matlab has been used to carry out the numerical simulations with the two kinds of controller. Once the optimal parameters of the controllers have been found and verified, a 1000-run Montecarlo simulation for both types of controller has been carried out and results have been compared in terms of quality of the trajectory inside the approach cones, respect of docking requirements, and use of delta-V.

Acknowledgments

I would like to express my sincere gratitude to Andrea Valmorbida for the precious advice and support during this thesis work, and Prof. Enrico Lorenzini for the passion transmitted over the years.

Contents

Abstract iii

Acknowledgments v

List of Figures xi

List of Tables xvi

1 Introduction 1

1.1 Rendezvous and docking overview 2

1.1.1 Launch and orbit injection 3

1.1.2 Orbit phasing 3

1.1.3 Far-range rendezvous operations 4

1.1.4 Close-range rendezvous operations 4

1.1.5 Mating 5

1.2 Motivation 6

1.3 Thesis outline 7

2 Spacecraft relative translational and attitude dynamics 8

2.1 Introduction 8
2.2 Reference frames 9
2.3 Translational dynamics 10
2.3.1 Nonlinear equations of relative motion 10
2.3.2 Linear equations of relative motion 12
2.3.3 Tschauner-Hempel equations 12
2.3.4 Clohessy-Wiltshire equations 15
2.4 Spacecraft attitude dynamics 20
2.4.1 Euler angles 20
2.4.2 Quaternions 21
2.4.3 Rotational dynamics 22
2.4.4 Euler equations with reaction wheels 24

3 Model Predictive Control **25**

3.1 Introduction 25
3.2 Basic formulation of Model Predictive Control 26
3.3 Formulation as a QP problem 29
3.4 Observer-based MPC 33
3.4.1 Kalman Filter 34
3.5 Measured disturbances and feed-forward 36
3.6 Explicit MPC 36

4 PID control **38**

4.1 Proportional action . . . 38

4.2 Derivative action . . . 39

4.3 Integral action . . . 40

4.4 Closed-loop stability . . . 41

5 Rendezvous and docking scenario definition **43**

5.1 Case studies . . . 43

5.1.1 Optical Communications and Sensors Demonstration Program . . . 43

5.1.2 Automated Transfer Vehicle . . . 44

5.1.3 CubeSat Proximity Operations Demonstration . . . 44

5.2 Chaser spacecraft . . . 45

5.2.1 Propulsion system . . . 46

5.2.2 Attitude control system . . . 47

5.2.3 Measurements . . . 48

5.3 Target station . . . 50

5.4 V-bar approach strategy . . . 51

5.5 Velocity reference . . . 52

5.6 Constraints . . . 53

5.7 Disturbances . . . 55

5.7.1 Differential drag . . . 55

5.7.2 J2 perturbation effects . . . 56

6 Controllers tuning and preliminary simulations results 58

6.1 Software simulator 58

6.2 State space formulation 59

6.3 Validation of MPC 60

6.4 Translational control with PID 65

6.5 MPC with disturbances 70

6.6 PID with disturbances 76

6.7 Attitude control 80

7 Montecarlo simulations 87

7.1 Introduction 87

7.2 MPC Montecarlo results 89

7.3 PID Montecarlo results 96

7.4 Attitude controller Montecarlo results 104

8 Conclusions 109

Bibliography 111

List of Figures

2.1 Coordinate frames in relative motion. 11

2.2 Case 1: $e = 0.001$. Propagation of the chaser position in Hill's frame with TH solutions (blue) and CW solutions (light blue, dashed). 17

2.3 Case 1: $e = 0.001$. 18

2.4 Case 2: $e = 0.1$. Propagation of the chaser position in Hill's frame with TH solutions (blue) and CW solutions (light blue, dashed). 18

2.5 Case 2: $e = 0.1$. 19

3.1 Receding horizon principle. 27

3.2 Block diagram of a state observer. 33

4.1 Block diagram of a PID controller with low-pass filter in the derivative term and anti-reset windup scheme in the integral term. 41

4.2 Stability region for a continuous-time system on the left, and a discrete-time system on the right. 42

4.3 Block diagram of a closed-loop system with discrete-time PID. 42

5.1 Chaser: 3U CubeSat. 45

5.2 Illustration of PWM and PWC modes in a time interval. 47

5.3 V-bar approach of the ATV. 51

5.4 (a) Along-track speed reference from S3 to S4; (b) Along-track velocity reference from S4 to docking. 52

5.5 Position constraints (not in scale). 54

6.1 Trajectory of the chaser with MPC in red; approach truncated pyramid in black; docking point starred. 62

6.2 Actual trajectory in solid blue line and reference trajectory in green dashed line. (a) In-plane relative position; (b) In-plane relative position from hold point S4 to docking; (c) Out-of-plane relative position; (d) Out-of-plane relative position from hold point S4 to docking. 62

6.3 Relative velocity of the chaser with MPC in blue; reference speed in green; speed constraints dashed in red. 64

6.4 Commanded accelerations in blue (MPC). Constraints dashed in red. . . . 65

6.5 Poles location. All points are inside the unit circle, thus ensuring stability. 67

6.6 Trajectory of the chaser with PID in red; approach truncated pyramid in black; docking point starred. 67

6.7 Actual trajectory in solid blue line and reference trajectory in green dashed line. (a) In-plane relative position; (b) In-plane relative position from hold point S4 to docking; (c) Out-of-plane relative position; (d) Out-of-plane relative position from hold point S4 to docking. 68

6.8 Relative velocity of the chaser with PID in blue; reference speed in green; speed constraints dashed in red. 69

6.9 Commanded accelerations in blue (PID). Constraints dashed in red. 69

6.10 Trajectory of the chaser with MPC in red; approach truncated pyramid in black; docking point starred. 71

6.11 Actual trajectory in solid blue line and reference trajectory in green dashed line. (a) In-plane relative position; (b) In-plane relative position from hold point S4 to docking; (c) Out-of-plane relative position; (d) Out-of-plane relative position from hold point S4 to docking. 72

6.12 Relative velocity of the chaser with MPC and disturbances in blue; reference speed in green; speed constraints dashed in red. 73

6.13 Position estimation error (MPC). 74

6.14 Velocity estimation errors (MPC). 74

6.15 Commanded accelerations in blue (MPC). Constraints dashed in red. . . . 75

6.16 Trajectory. 77

6.17 Actual trajectory in solid blue line and reference trajectory in green dashed line. (a) In-plane relative position; (b) In-plane relative position from hold point S4 to docking; (c) Out-of-plane relative position; (d) Out-of-plane relative position from hold point S4 to docking. 77

6.18 Relative velocity of the chaser with PID and disturbances in blue; reference speed in green; speed constraints dashed in red. 78

6.19 Position estimation error (PID). 79

6.20 Velocity estimation errors (PID). 79

6.21 Commanded accelerations in blue (PID). Constraints dashed in red. 80

6.22 Roll, pitch, and yaw angles. 82

6.23 Quaternion components of the attitude state. 83

6.24 Angular velocity of the chaser w.r.t. the LVLH frame. 83

6.25 Torques generated by the reaction wheels, acting on the body. Remember that the motor torque of each wheel is opposite in sign. 84

6.26 Speed of the reaction wheels w.r.t. the body frame. 84

6.27 Error of Euler angles. 85

6.28 Error of angular rates. 85

7.1 Trajectories of 1000 runs with MPC. 89

7.2 In-plane trajectory, from S3 to docking. 89

7.3 In-plane trajectory in the second approach cone, from S4 to docking. . . . 90

7.4 Zoom in the in-plane station keeping at point S4. 90

7.5 Out-of-plane trajectory from S3 to docking. 91

7.6 Out-of-plane trajectory in the second approach cone, from S4 to docking. . 91

7.7 Zoom in the out-of-plane station keeping at point S4. 92

7.8 Docking performances with MPC. 92

7.9 Histogram of docking performances with MPC. 93

7.10 Lateral misalignments (y, z) and lateral velocity errors at docking with MPC. All samples are within the limits. 94

7.11 Longitudinal velocity error at docking with MPC. 95

7.12 Delta-Vs with MPC (in body frame). (a) X-direction; (b) Y-direction; (c) Z-direction; (d) Sum of each axis' delta-V. 96

7.13 Trajectories of 1000 runs with PID. 97

7.14 In-plane trajectory, from S3 to docking. 97

7.15 In-plane trajectory in the second approach cone, from S4 to docking. . . . 98

7.16 Zoom in the in-plane station keeping at point S4. 98

7.17 Out-of-plane trajectory from S3 to docking. 99

7.18 Out-of-plane trajectory in the second approach cone, from S4 to docking. . 99

7.19 Zoom in the out-of-plane station keeping at point S4. 100

7.20 Docking performances with PID. 100

7.21 Histogram of docking performances with PID. 101

7.22 Lateral misalignments (y, z) and lateral velocity errors at docking with PID. Not all samples respect the limits. 102

7.23 Longitudinal velocity error at docking with PID. 102

7.24 Delta-Vs with PID (in body frame). (a) X-direction; (b) Y-direction; (c) Z-direction; (d) Sum of each axis' delta-V. 103

7.25 Comparison of the total delta-V distributions in the Montecarlo simulations for the MPC (in blue) and the PID (in red). 104

7.26 Roll, pitch, and yaw of the chaser. 105

7.27 Quaternions components of the chaser's attitude. 105

7.28 Torque generated by the reaction wheels on the spacecraft. 106

7.29 Angular misalignments ((a), (c), (e)) and angular rate errors ((b), (d), (f)) at docking. All samples are within the limits. 107

7.30 Maximum momenta accumulated by the reaction wheels. (a) X-axis; (b) Y-axis; (c) Z-axis. All samples are within the limit for this kind of RW. . . 108

List of Tables

2.1 Algorithm to solve the TH equations. 14

2.2 Simulation parameters. Position and velocity are expressed in Hill's frame. 17

5.1 Propulsion system of the chaser. 46

5.2 Characteristics of the reaction wheels. 47

5.3 Characteristics of the chaser. 48

5.4 Chaser's state at the docking point. 49

5.5 Requirements of the relative states of the chaser at docking. 50

6.1 Initial conditions. 60

6.2 MPC parameters. 61

6.3 Chaser's state at hold point S4 (MPC). 63

6.4 Chaser's state at the docking point (MPC). 63

6.5 Use of delta-V in the entire maneuver with MPC (in body axes). 65

6.6 Initial conditions. 66

6.7 PID gains for the translational motion. 66

6.8 Chaser's state at hold point S4 (PID). 68

6.9 Chaser's state at the docking point (PID). 68

6.10 Use of delta-V in the entire maneuver with PID (in body axes). 70

6.11 Weights of the MPC. 71

6.12 Chaser's state at hold point S4 (MPC). 72

6.13 Chaser's state at docking (MPC). 73

6.14 Use of delta-V in the entire maneuver with MPC and disturbances. 75

6.15 PID gains for the translational motion. 76

6.16 Chaser's state at arrival at hold point S4. 78

6.17 Chaser's state at docking. 78

6.18 Use of delta-V in the entire maneuver with PID and disturbances. 80

6.19 Initial conditions and setpoints for the attitude control. 81

6.20 Gains of the attitude PID controller. 82

6.21 Attitude states after 200 s. 86

7.1 Initial translational states at S3. 88

7.2 Initial attitude states at S3. 88

7.3 Lateral misalignments and velocity errors at docking with MPC. 94

7.4 Delta-V used with MPC (in body frame). 95

7.5 Lateral misalignments and velocity errors at docking with PID. 101

7.6 Delta-V used with PID (in body frame). 103

7.7 Attitude misalignments and rate errors at docking. 106

7.8 Average, minimum, and maximum value of the maximum momenta accumulated by the reaction wheels. 108

Chapter 1

Introduction

The process of rendezvous and docking (RVD) consists of a sequence of orbital maneuvers realized to bring an active vehicle (also referred to as *chaser*) close to or attached to a *target* vehicle. Rendezvous and docking comprise an important operation task for many space missions involving multiple spacecraft. Its function is to accomplish operations such as:

- assembly and supply of orbital platforms and stations;
- exchange of crew members in orbital stations;
- capture or retrieval of orbiting vehicles, such as for planetary return missions;
- debris removal.

Rendezvous and docking technologies arouse much interest during the 1960s, especially by the US Apollo Program when these techniques were necessary to implement the Lunar Orbit Rendezvous for the Moon landing missions.

The first-ever rendezvous maneuver was accomplished on December 15, 1965, by astronauts Walter Schirra and Thomas Stafford aboard Gemini VI conducting manually a series of orbital maneuvers to approach a distance of about 40 meters with zero relative velocity from Gemini VII's spacecraft. On March 16, 1966, the first docking between two spacecraft took place when Neil Armstrong and David Scott docked their Gemini VIII spacecraft with the Agena target vehicle.

The first automatic RVD was performed on October 30, 1967, between the Soviet vehicles Kosmos 186 and 188. Since then, rendezvous and docking operations have been performed in numerous programs, such as:

- US Apollo and Skylab programs;
- Soviet Salyut and Mir Space Station programs;
- US Space Shuttle missions to the space station Mir;
- assembly, crew exchange and re-supply of the International Space Station (ISS).

The ISS is one of the most important application of rendezvous and docking technologies since the beginning of the century. Several unmanned cargo ships have been launched and have successfully approached the ISS autonomously, to bring fuel and supplies. One of these was the Automated Transfer Vehicle (ATV), an unmanned cargo spacecraft developed and operated by the European Space Agency for supplying space cargo autonomously to the space station. Another example is the Russian Progress spacecraft, which versions have been used since 1978 by the Soviet Union for the space stations Salyut 6, Salyut 7 and Mir.

1.1 Rendezvous and docking overview

Rendezvous and docking process is a complex series of operations, each of which requires the fulfilment of numerous conditions and constraints. The whole process is characterized by five main phases [1, 2]:

1. Launch and orbit injection;
2. Orbit phasing;
3. Far-range rendezvous operations;
4. Close-range rendezvous operations;
5. Mating.

Each phase is briefly described in the following sections.

1.1.1 Launch and orbit injection

Due to the Earth rotation, there exist two time instants per day at which each point on the Earth surface intersects a given orbit plane. The launch window should be located at one of those time instants; however, because of the convenience of eastern launch directions and since only a limited sector of them are available at a launch site, there is normally only one compatible launch window per day.

Perturbations do not have to be neglected. The oblateness of the Earth causes a drift in the orbital plane, which rate depends on the orbital height. Therefore, the launch window has to be planned accounting for the orbital plane drift, so that when the chaser approaches the target, their orbital planes are as similar as possible and correction maneuvers may be avoided. The initial orbit of the chaser spacecraft is usually lower than the target's one, and the chaser will be located at a certain phase angle behind the target.

1.1.2 Orbit phasing

The objective of the next phase is reducing the phase angle between the chaser and the target. The insertion orbit may not be suitable to perform the orbit phasing, as its exact altitude is not known in advance due to some uncertainties of the rocket launch. Phasing depends also on the target's orbital position at the time of launch, so different strategies have to be employed for each rendezvous mission. Therefore, the chaser spacecraft needs to be brought into an appropriate phasing orbit. This can be accomplished by performing a Hohmann transfer or a Bielliptical maneuver.

Absolute navigation measurements are used during phasing maneuvers, which are provided by sensors aboard the chaser, such as GPS, or by ground measurements.

1.1.3 Far-range rendezvous operations

The main objective of this phase, also referred to as *homing*, is to reach the target's orbit and reduce approaching velocity. Typically, this phase starts when the chaser is located a few tens of kilometers away from the target and ends when it reaches a preselected point usually situated at a range of few kilometers.

In cases where the target is represented by a space station, safety concerns constrain the chaser vehicle to hold on outside a special approach ellipsoid located in the vicinity of the station. During this phase, relative navigation measurements, such as differential GPS between chaser and target or RADAR systems, are used. In cases of manned chaser and target vehicles, like a crewed spacecraft approaching a space station, a communication link between the two vehicles is established prior to or shortly after the beginning of the far rendezvous phase. This is necessary for safety considerations, as ground communication is more subjected to disturbances and black-outs.

1.1.4 Close-range rendezvous operations

This phase can be divided into two subphases: a preliminary phase leading to the final approach corridor (also called *closing*), and a final phase leading to the mating conditions.

The objective of the closing subphase is to further reduce the distance between the two spacecraft, leading the chaser to the approach corridor. This can be accomplished by a fly-around maneuver to acquire the approach axis, in case the direction of motion is not the same as the approach corridor. At the end of this phase, the chaser spacecraft is placed in the proper position with required attitude and velocity, ready to start the final approach. Because of the level of criticality during this phase, the navigation accuracy needs to be increased, and different types of sensors, such as Vision-Based-Navigation, may be adopted.

The typical final approach consists of a straight line trajectory starting from a few tens of meter from the docking port (or berthing box) lying within an approach-cone for observability and safety reasons. Another possibility is to follow a quasi-straight line trajectory realized by multiple hops, which is more convenient for man-controlled approaches.

An important issue which has not to be neglected during the final approach is the plume impingement. Hot combustion products of the thrusters can impact the target vehicle if the relative distance is short enough, with consequent negative effects on the spacecraft. Some of these regard the contamination of surfaces and sensitive elements, overheating of parts of the surface, and disturbance in attitude and position, depending on the masses involved. With the aim of minimizing these effects, constraints on velocity may be applied in the final approach.

1.1.5 Mating

Mating is the final phase, which ends with a stable connection of the two vehicles. Different tasks need to be fulfilled by the mating system during this operations, such as the achievement of capture, the attenuation of residual relative motions between the two vehicles, the achievement of rigid structural connection and others relating sealing and connection of data and power interfaces. Mating can be distinguished between docking and berthing. *Docking* operations consist of the realization of the connection between the chaser capture interfaces and the target ones. *Berthing* refers to the operation of approaching a determined box in space, where the spacecraft holds before being grappled by a robotic arm, or vice versa. This type of operation is less subjected to plume impingement effects, because the berthing box is usually placed at a proper distance from the target vehicle.

1.2 Motivation

Initially, rendezvous and proximity maneuvers have been performed manually, using open-loop techniques and error corrections based on sight, requiring high-end optical devices and professional training.

Space exploration has become a significant argument for many current and future missions, and autonomous RVD operations have turned into a growing necessity. In particular, several planned missions are conceived accounting for autonomous RVD operations in various phases of the mission. For example, the Mars Sample Return (MSR) program has been jointly studied by NASA and ESA for the next years, with the objective of acquire and return to Earth martian soil samples to be analyzed in terrestrial laboratories. The benefits of this program can lead to potential crucial discoveries in the history of Mars and life on another planet.

Recently, most docking missions have been adopting autonomous control. There are many advantages of using automated procedures, such as the introduction of feedback into the system. A typical control technique is the Proportional-Integral-Derivative (PID) controller, but advanced controllers like Linear Quadratic Regulators (LQR) or H-infinity methods are also applied.

During the final approach of a RVD, several constraints need to be respected. Among these there are safety constraints, such as the positioning of the spacecraft within the Line-of-Sight (LoS) cone in case of approaching a target platform, or constraints on the relative velocity and obstacle avoidance. A good controller should therefore be able to take these constraints into account, along with physical constraints, such as the thrust magnitude, to prevent saturation.

A solution to this matter could be adopting controllers based on Model Predictive Control (MPC) technique [3], which is able to effectively deal with constraints. This technology allows to achieve optimal trajectories for proximity maneuvering by solving an online optimization problem based on the minimization of a prescribed cost function.

The application of the MPC for rendezvous and docking maneuvers, which is the main argument of this work, will be tested with numerical simulations in a realistic scenario and the results will be discussed in terms of robustness and efficiency.

1.3 Thesis outline

The remaining part of this thesis can be divided into three main sections. At first, the mathematical models and the principle of operation of Model Predictive Control and PID control will be discussed (Chapter 2,3 and 4). Subsequently, the scenario of simulation will be presented in Chapter 5, along with some preliminary numerical simulation results used to prove the efficiency of the selected gains (Chapter 6). Finally, the results of a Monte-Carlo simulation for either controllers are shown and compared (Chapter 7).

Chapter 2

Spacecraft relative translational and attitude dynamics

2.1 Introduction

In this chapter, the equations of relative motion between spacecraft and attitude dynamics in unperturbed orbits are presented.

Relative motion is an important aspect to understand, in that several space missions involve multiple spacecraft or satellites. Spacecraft formation flying missions, for example, require maintenance of a desired relative separation among the vehicles, and rendezvous and docking maneuvers require appropriate relative speed and position to be achieved between the chaser and the target.

Equations of relative motion govern the relative dynamics of two orbiting objects, and constitute the dynamical model on which all the simulations are based.

In the following paragraphs, fully nonlinear equations of relative motion (NERM) are investigated, following which linear equations of relative motion (LERM) are described, along with the Tchauner-Hempel (TH) equations. Solution to the TH equations is obtained by a state-transition matrix as proposed by Yamanaka and Ankersen (YA) [4]. The simple case of circular orbit, described by the Clohessy-Wiltshire (CW) equations, and its solution are also presented.

Finally, the spacecraft attitude dynamics are investigated, where Euler's equations and quaternions are outlined.

2.2 Reference frames

The following reference frames are defined and used in this thesis.

Inertial reference frame $\mathcal{F}_{\mathcal{I}}$

The Earth-Centered-Inertial (ECI) coordinate system is centered at the Earth's center of mass and is defined by the triad of vectors $[\hat{\mathbf{I}}, \hat{\mathbf{J}}, \hat{\mathbf{K}}]$, where $\hat{\mathbf{I}}$ is aligned with the vernal point, $\hat{\mathbf{K}}$ is aligned with the Earth rotation axis and toward the North Pole, and $\hat{\mathbf{J}}$ completing the orthogonal triad. The inertial coordinates are herein defined as $[X, Y, Z]$.

Hill's reference frame $\mathcal{F}_{\mathcal{H}}$

Hill's coordinate system is an orbiting reference frame following the target orbital motion with origin in the target's center of mass. The unit vectors defining this system are denoted as $[\hat{\mathbf{i}}, \hat{\mathbf{j}}, \hat{\mathbf{k}}]$ and the coordinates are expressed as $[x, y, z]$. The x-axis is oriented along the radial axis, the z-axis is aligned with the angular momentum of the target's orbit, and the y-axis completes the right-handed orthogonal triad. Note that x and y lie on the orbital plane, with y directed toward the trajectory of the target (and parallel to its velocity only in circular orbits), whereas z corresponds to the out-of-plane direction.

Orbital reference frame $\mathcal{F}_{\mathcal{O}}$

The spacecraft local orbital frame $\mathcal{F}_{\mathcal{O}}$ is a coordinate system with origin in the center of mass of a spacecraft in circular orbit. This reference frame is also referred to as Local Vertical - Local Horizontal (LVLH) frame, and is widely used in flight mechanics. Z-axis points toward the center of the Earth, or Nadir, and for this reason it is often called R-bar. Y-axis is aligned opposite the orbit angular momentum, and finally X completes the

triad. The X-axis is also denoted as V-bar, as it points in the direction of the spacecraft's motion.

Body frame $\mathcal{F}_\mathcal{B}$

The spacecraft body frame, or simply body frame, is a reference frame with origin at the center of mass of the spacecraft and with axes fixed with the satellite. In this thesis, it is assumed that the body axes are parallel to the principal axes of inertia.

2.3 Translational dynamics

The translational relative motion between two orbiting vehicles can be analyzed with respect to the Hill's frame of reference attached to the target satellite.

2.3.1 Nonlinear equations of relative motion

To derive the nonlinear equations of relative motion, the description of the relative dynamics is firstly analyzed with respect to an inertial reference frame, and then transformed into the orbiting frame. The following definitions are given:

- $\boldsymbol{r_t}$ is the inertial position vector of the target;
- $\boldsymbol{r_c}$ is the inertial position vector of the chaser;
- $\boldsymbol{\rho} = \boldsymbol{r_c} - \boldsymbol{r_t}$ is the inertial relative position vector of the chaser w.r.t. the target.

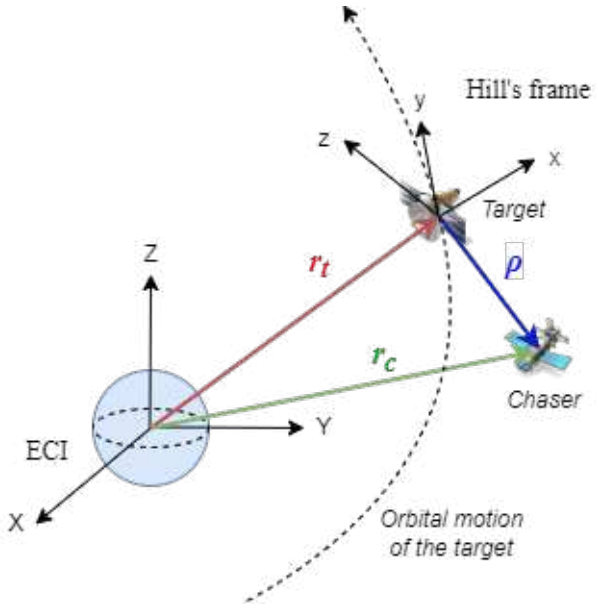


Figure 2.1: Coordinate frames in relative motion.

We want to express the relative acceleration $\ddot{\boldsymbol{\rho}}$ in Hill's coordinate system. To do so, we first decompose the acceleration as:

$$\ddot{\boldsymbol{\rho}} = \ddot{\boldsymbol{\rho}}_{rel} + 2\boldsymbol{\omega} \times \dot{\boldsymbol{\rho}}_{rel} + \dot{\boldsymbol{\omega}} \times \boldsymbol{\rho} + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{\rho}) \tag{2.1}$$

where $\ddot{\boldsymbol{\rho}}_{rel} = [\ddot{x} \quad \ddot{y} \quad \ddot{z}]^T$ is the relative acceleration vector, and $\boldsymbol{\omega} = [0 \quad 0 \quad \dot{\theta}]^T$ is the angular velocity vector of $\mathcal{F}_{\mathcal{H}}$ as seen in Hill's frame. Recalling that:

$$\begin{aligned} \ddot{\boldsymbol{\rho}} &= \ddot{\boldsymbol{r}}_c - \ddot{\boldsymbol{r}}_t \\ &= -\frac{\mu}{r^3}\boldsymbol{r}_c + \frac{\mu}{r_t^3}\boldsymbol{r}_t \end{aligned} \tag{2.2}$$

and that $\boldsymbol{r}_t = [r_t \quad 0 \quad 0]^T$, substituting Equation (2.2) into (2.1) yields the nonlinear equations of relative motion (NERM):

$$\begin{aligned} \ddot{x} &= 2\,\dot{\theta}\,\dot{y} + \ddot{\theta}\,y + \dot{\theta}^2 x + \frac{\mu}{r_t^2} - \frac{\mu\,(x + r_t)}{\left[\,(r_t + x)^{\,2} + y^2 + z^2\right]^{\frac{3}{2}}} \\ \ddot{y} &= -2\,\dot{\theta}\,\dot{x} - \ddot{\theta}\,x + \dot{\theta}^2 y - \frac{\mu\,y}{\left[\,(r_t + x)^{\,2} + y^2 + z^2\right]^{\frac{3}{2}}} \\ \ddot{z} &= -\frac{\mu\,z}{\left[\,(r_t + x)^{\,2} + y^2 + z^2\right]^{\frac{3}{2}}} \end{aligned} \tag{2.3}$$

2.3.2 Linear equations of relative motion

Assuming that the separation between the target and the chaser ρ is small if compared to the distance from the center of the planet r_t (from now on called r), the equations of relative motion (2.3) can be linearized about the origin of Hill's frame. The new equations are called linear equations of relative motion (LERM):

$$\begin{aligned} \ddot{x} &= 2\,\dot{\theta}\,\dot{y} + \left(\dot{\theta}^2 + 2\frac{\mu}{r^3}\right) x + \ddot{\theta}\, y \\ \ddot{y} &= -2\,\dot{\theta}\,\dot{x} - \ddot{\theta}\, x + \left(\dot{\theta}^2 - \frac{\mu}{r^3}\right) y \\ \ddot{z} &= -\frac{\mu}{r^3} z \end{aligned} \tag{2.4}$$

It is important to note that the in-plane dynamics described by the linearized equations are decoupled from the out-of-plane dynamics.

2.3.3 Tschauner-Hempel equations

The linearized equations can be written in state-space form, described by the state vector $\mathbf{x} = [x,\ y,\ z,\ \dot{x},\ \dot{y},\ \dot{z}]^T$, obtaining a linear time-variant system:

$$\dot{\mathbf{x}} = A(t)\mathbf{x} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ \left(\dot{\theta}^2 + 2\frac{\mu}{r^3}\right) & \ddot{\theta} & 0 & 0 & 2\dot{\theta} & 0 \\ -\ddot{\theta} & \left(\dot{\theta}^2 - \frac{\mu}{r^3}\right) & 0 & -2\dot{\theta} & 0 & 0 \\ 0 & 0 & -\frac{\mu}{r^3} & 0 & 0 & 0 \end{bmatrix} \boldsymbol{x} \tag{2.5}$$

The state matrix $A(t)$ is time-varying, as orbital radius r, the true anomaly θ, and their derivatives vary with time for a general elliptical orbit. The time t is the independent

variable, and θ is found by Kepler's equation

$$\begin{aligned} n\,(t - t_0) &= E - e\sin E \\ \tan\frac{\theta}{2} &= \sqrt{\frac{1+e}{1-e}}\tan\frac{E}{2} \end{aligned} \tag{2.6}$$

However, θ is a bijective function of time, and therefore it can be used instead of t as the independent variable. It is useful to introduce the following definitions:

- $k(\theta) \coloneqq 1 + e\cos\theta$;
- $\bar{x} \coloneqq k(\theta)\,x$ is a new variable scaled by the factor k;
- $\bar{x}' = \dfrac{d\bar{x}}{d\theta}$ is the derivative of $\bar{x}$ with respect to θ.

The change of variables can be summarized by a transformation matrix $T(\theta)$ such that:

$$\bar{\mathbf{x}} = T(\theta)\mathbf{x}$$

where:

$$T(\theta) = \begin{bmatrix} k(\theta)\,\mathbf{I_{3\times 3}} & \mathbf{0_{3\times 3}} \\ -e\sin\theta\,\mathbf{I_{3\times 3}} & \dfrac{p^2}{hk(\theta)}\mathbf{I_{3\times 3}} \end{bmatrix} \quad \text{and} \quad T^{-1}(\theta) = \begin{bmatrix} \dfrac{1}{k(\theta)}\mathbf{I_{3\times 3}} & \mathbf{0_{3\times 3}} \\ \dfrac{he}{p^2}\sin\theta\,\mathbf{I_{3\times 3}} & \dfrac{h}{p^2}k(\theta)\,\mathbf{I_{3\times 3}} \end{bmatrix}$$

By applying this transformation to the LERM (2.4), they can be rewritten in the following form known as the Tschauner-Hempel equations:

$$\begin{aligned} \bar{x}'' &= \frac{3}{k}\bar{x} + 2\bar{y}' \\ \bar{y}'' &= -2\bar{x}' \\ \bar{z}'' &= -\bar{z} \end{aligned} \tag{2.7}$$

It is to be noted that as the true anomaly is the independent variable, it is not limited to 2π, but instead increases with time.

Solution to the TH equations

The solution presented in this section refers to the work carried out by Yamanaka and Ankersen in [4] and also described by R. Sherrill et al. in [5]. The algorithm is showed in Table 2.1.

Table 2.1: Algorithm to solve the TH equations.

Solution algorithm to the TH equations		
1.	Given state vector at initial time	$x(t_0)$
2.	Rescale initial state vector	$\bar{x}_0 = T(\theta_0)x(t_0)$
3.	Build fundamental solution vectors	$\psi_1,\ \psi_2,\ \psi_3,\ \psi_4,\ \psi_5,\ \psi_6$
4.	Assemble solution matrix	$\Psi(\theta) = [\psi_1,\ \psi_2,\ \psi_3,\ \psi_4,\ \psi_5,\ \psi_6]$
5.	Create state transition matrix	$\Phi(\theta, \theta_0) = \Psi(\theta)\Psi^{-1}(\theta_0)$
6.	Compute desired state vector	$\bar{x}(\theta) = \Psi(\theta, \theta_0)\bar{x}_0$
7.	Return to time domain	$x(t) = T^{-1}(\theta)\bar{x}(\theta)$

The fundamental solutions ψ_i, $i = 1, \ldots, 6$, are:

$$\psi_1 = \begin{bmatrix} \sin\theta k(\theta) \\ 2\cos\theta - e\sin^2\theta \\ 0 \\ \cos\theta + e\cos 2\theta \\ -2\sin\theta k(\theta) \\ 0 \end{bmatrix}, \quad \psi_2 = \begin{bmatrix} \cos\theta k(\theta) \\ -2\sin\theta - e\sin\theta\cos\theta \\ 0 \\ -\sin\theta - e\sin 2\theta \\ e - 2\cos\theta k(\theta) \\ 0 \end{bmatrix}$$

$$\psi_3 = \begin{bmatrix} 1 - \dfrac{3e}{2}J\sin(\theta)k(\theta) \\ -\dfrac{3}{2}Jk(\theta)^2 \\ 0 \\ -\dfrac{3e}{2}J\left(\cos\theta + e\cos 2\theta\right) - \dfrac{3e\sin\theta}{2k(\theta)} \\ 3eJ\sin\theta k(\theta) - \dfrac{3}{2} \\ 0 \end{bmatrix}, \quad \psi_4 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\psi_5 = \begin{bmatrix} 0 \\ 0 \\ \sin\theta \\ 0 \\ 0 \\ \cos\theta \end{bmatrix}, \quad \psi_6 = \begin{bmatrix} 0 \\ 0 \\ \cos\theta \\ 0 \\ 0 \\ -\sin\theta \end{bmatrix}$$

where $J = \dfrac{h}{p^2}(t - t_0)$ and recall $k(\theta) = 1 + e\cos\theta$.

This algorithm is particularly useful because it avoids the need of numerical integration of the TH equations.

2.3.4 Clohessy-Wiltshire equations

The Clohessy-Wiltshire equations (CW), developed in the early 1960s, analyze spacecraft relative motion when the target's orbit is circular and the chaser is not too far from it. These equation can be obtained as a simplification of TH Equations (2.7), setting $e = 0$, and the result is a linear time-invariant system, for which a closed-form solution exists. CW equations take the following form:

$$\begin{aligned} \ddot{x} &= 3n^2 x + 2n\dot{y} \\ \ddot{y} &= -2n\dot{x} \\ \ddot{z} &= -n^2 z \end{aligned} \tag{2.8}$$

or, is state space formulation:

$$\dot{\mathbf{x}}(t) = A\mathbf{x}(t) = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 3n^2 & 0 & 0 & 0 & 2n & 0 \\ 0 & 0 & 0 & -2n & 0 & 0 \\ 0 & 0 & -n^2 & 0 & 0 & 0 \end{bmatrix} \mathbf{x}(t) \tag{2.9}$$

where n is the orbital angular velocity of the target spacecraft. The solution is easily obtained in terms of the transition matrix $e^{A(t-t_0)}$, that is (for $t_0 = 0$):

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} 4-3\cos nt & 0 & 0 & \dfrac{\sin nt}{n} & \dfrac{2}{n}(1-\cos nt) & 0 \\ -6nt+6\sin nt & 1 & 0 & -\dfrac{2}{n}(1-\cos nt) & \dfrac{4\sin nt}{n}-3t & 0 \\ 0 & 0 & \cos nt & 0 & 0 & \dfrac{\sin nt}{n} \\ 3n\sin nt & 0 & 0 & \cos nt & 2\sin nt & 0 \\ -6n\,(1-\cos nt) & 0 & 0 & -2\sin nt & -3+4\cos nt & 0 \\ 0 & 0 & -n\sin nt & 0 & 0 & \cos nt \end{bmatrix} \mathbf{x}(0) \tag{2.10}$$

An important observation is that the motion along y is unstable, because it contains secular terms which cause the spacecraft to drift along-track with time. Nevertheless, the motion will remain stable if this particular initial condition is verified:

$$\dot{y}(0) = -2nx(0) \tag{2.11}$$

In this case, the chaser relative motion describes a closed trajectory around the target (*fly-around*) generated by a section of an elliptic cylinder.

The example provided below presents the comparison of the CW solution with the TH solution, in two different cases. In case 1, the eccentricity of the target's orbit is set $e = 0.001$, whereas in case 2 it is $e = 0.1$. The initial position of the chaser with respect to the target-fixed coordinate frame (Hill's coordinate frame) is chosen such that $\rho \ll r$, thereby justifying the employment of these models.

Table 2.2: Simulation parameters. Position and velocity are expressed in Hill's frame.

Parameter	**Value**
Eccentricity	$e = 0.001$ and $e = 0.1$
Perigee height	450 km
Initial true anomaly	0°
Chaser initial position	$[10, -50, 20]$ m
Chaser initial velocity	$[0.1, -0.1, 0.1]$ m/s
Propagate duration	Two orbit revolutions

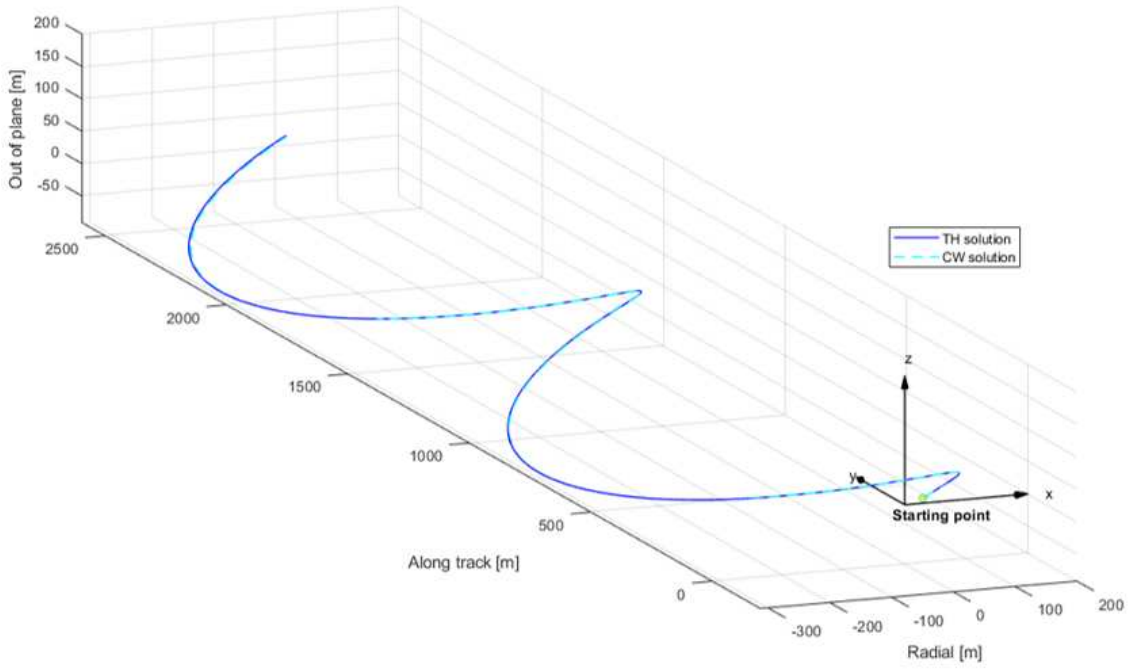
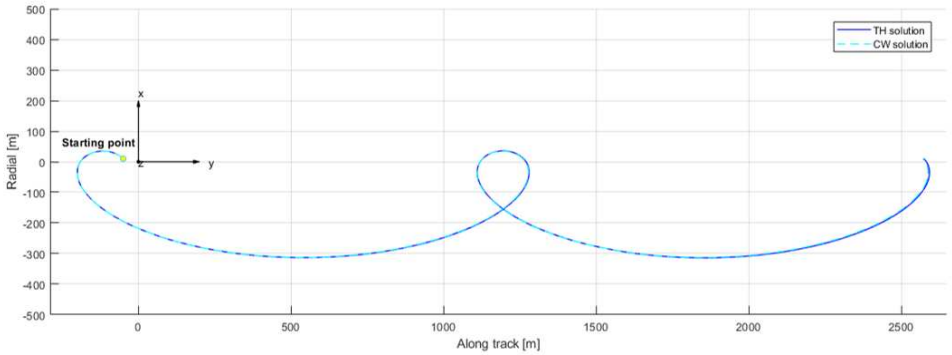
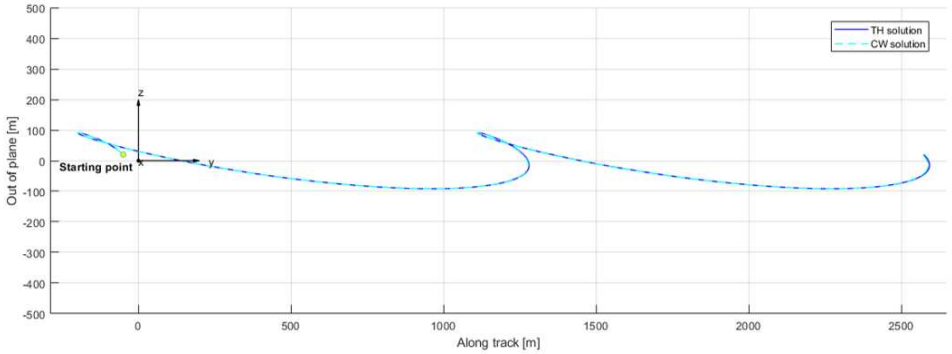


Figure 2.2: Case 1: $e = 0.001$. Propagation of the chaser position in Hill's frame with TH solutions (blue) and CW solutions (light blue, dashed).



(a) Plane X-Y.



(b) Plane Y-Z.

Figure 2.3: Case 1: $e = 0.001$.

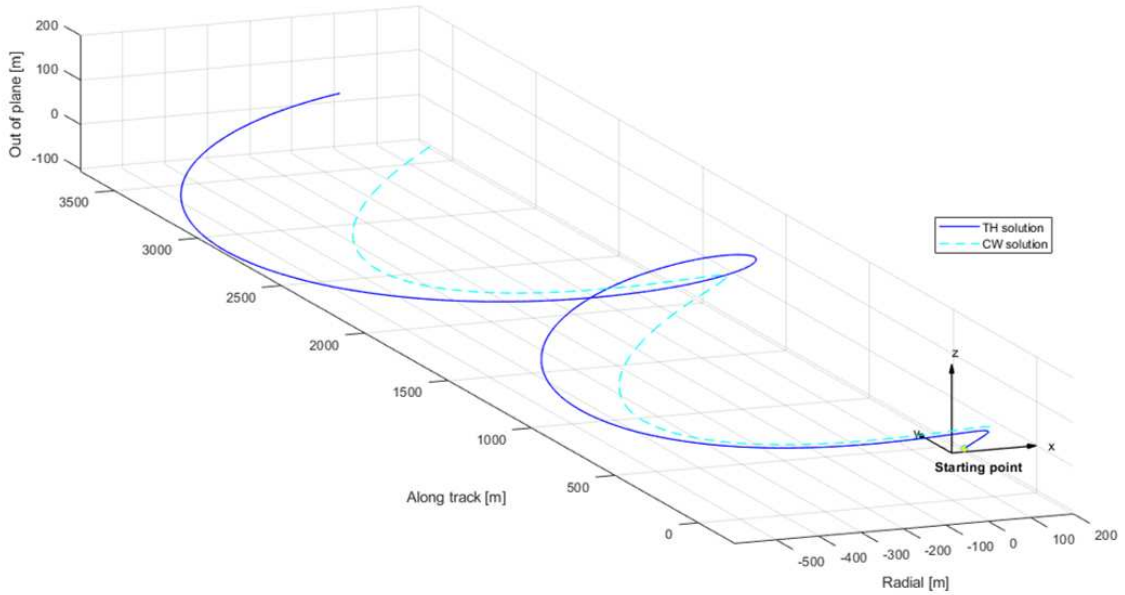
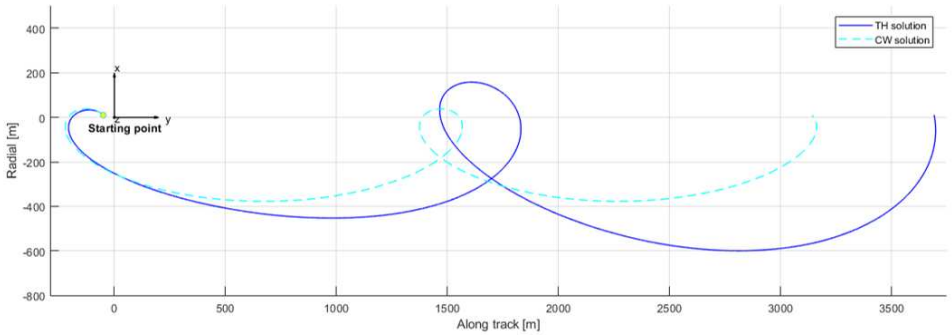
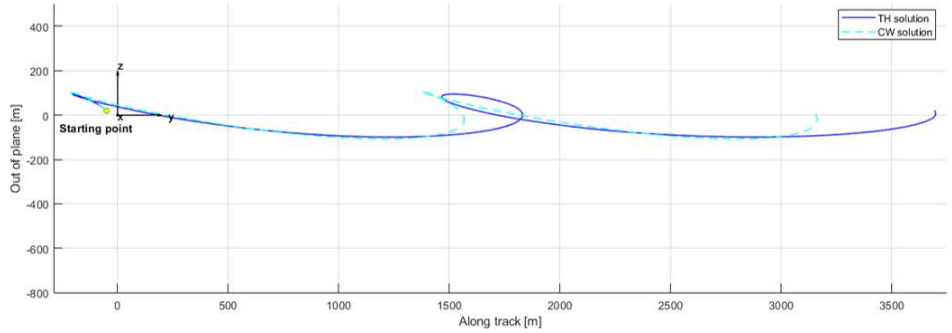


Figure 2.4: Case 2: $e = 0.1$. Propagation of the chaser position in Hill's frame with TH solutions (blue) and CW solutions (light blue, dashed).



(a) Plane X-Y.



(b) Plane Y-Z.

Figure 2.5: Case 2: $e = 0.1$.

Regarding case 1, we observe that the trajectory obtained by the CW solutions almost coincides with the one computed with the TH equations, as depicted in Figures 2.2 and 2.3. Therefore, for a scenario with a target's orbit eccentricity of $e = 0.001$ or less, the CW solutions provide a good model for the description of the dynamics.

Figures 2.4 and 2.5 show results for case 2, where the eccentricity of the target orbit is higher than case 1. In this second case, the CW solutions do not provide an accurate representation of the chaser's motion, as it deviates noticeably from the more accurate solution provided by the TH equations.

2.4 Spacecraft attitude dynamics

In this section, the mathematical models for the attitude dynamics and kinematics of a rigid body are investigated. First, the rotational dynamics equations are presented, along with the formulation that include reaction wheels for attitude control. Then the kinematics is discussed, with an overview on Euler angles and quaternions employed as attitude parameterization.

2.4.1 Euler angles

In order to fully describe the orientation of a rigid body with respect to an initial frame, three parameters are required. The conventional approach consists of finding three angles (Euler's angles) that describe a sequence of three rotations necessary to take the initial frame and make it coincide with the body frame. It should be noted that the initial frame does not require to be inertial. Indeed, for our purpose it is appropriate to describe the spacecraft attitude with respect to the local orbital frame.

There are many sequences of rotations; the one used in this work is the 3-2-1 sequence, in which the three angles, also known as Bryan's angles, correspond to yaw ψ, pitch θ, and roll ϕ.

We define $\boldsymbol{\omega}_B^{BO}$ as the angular velocity of the spacecraft body frame $\mathcal{F}_\mathcal{B}$ relative to the local orbital frame $\mathcal{F}_\mathcal{O}$ and expressed in the body frame. The relation between the derivatives of Euler's angles and the angular velocity of the spacecraft is given by the sequence of rotations 3-2-1 as follows:

$$\boldsymbol{\omega}_B^{BO} = \mathbf{R}_1(\phi) \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \mathbf{R}_1(\phi)\mathbf{R}_2(\theta) \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + \mathbf{R}_1(\phi)\mathbf{R}_2(\theta)\mathbf{R}_3(\psi) \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \tag{2.12}$$

where $\mathbf{R}_1(\phi)$, $\mathbf{R}_2(\theta)$, and $\mathbf{R}_3(\psi)$ are respectively the rotation matrices about the first, second and third axis resulting from intermediate transformations. Equations (2.12) can

be rewritten in the following compact form representing the attitude kinematics equations:

$$\boldsymbol{\omega}_B^{BO} = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\phi & \cos\phi\cos\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{2.13}$$

2.4.2 Quaternions

Euler's angles constitute a practical representation of a spacecraft's orientation, but they are also characterized by an unpleasant feature. The explicit form of the evolution of Euler angles as a function of the angular velocity, derived from equation (2.13), requires the inversion of a matrix, which determinant is, in our case, $\cos\phi$. Therefore, there exist particular situations which make the inversion impossible. These singularities can be avoided with the use of quaternions or Euler's parameters.

A quaternion $\mathbf{q} \in \mathbb{R}^4$ is a four-element entity which fully describes a rotation about an axis. The attitude matrix describing the transformation from the local orbital frame to the spacecraft body frame can thus be uniquely identified by a quaternion. The information regarding the angle of rotation and the direction of the axis of rotation are all contained in $\mathbf{q}$. In this work, the Hamilton formulation for quaternions is used, i.e. the scalar parameter $q_0 \in \mathbb{R}$ takes the first entry of $\mathbf{q}$:

$$\boldsymbol{q} \coloneqq \begin{bmatrix} q_0 \\ \tilde{\boldsymbol{q}} \end{bmatrix} \tag{2.14}$$

where $\tilde{\boldsymbol{q}} = [q_1,\ q_2,\ q_3]^T \in \mathbb{R}^3$.

The four elements must satisfy the constraint that normalizes the quaternion:

$$q_0^2 + \tilde{\boldsymbol{q}} \cdot \tilde{\boldsymbol{q}} = 1 \tag{2.15}$$

The time evolution of a quaternion can be expressed as

$$\dot{\mathbf{q}} = \frac{1}{2}\,\boldsymbol{\Omega}\,\mathbf{q} \tag{2.16}$$

where the matrix $\mathbf{\Omega}$ can be obtained from the components of the spacecraft's angular velocity in the body frame:

$$\mathbf{\Omega} = \begin{bmatrix} 0 & -\omega_1 & -\omega_2 & -\omega_3 \\ \omega_1 & 0 & \omega_3 & -\omega_2 \\ \omega_2 & -\omega_3 & 0 & \omega_1 \\ \omega_3 & \omega_2 & -\omega_1 & 0 \end{bmatrix} \tag{2.17}$$

The expression (2.16) is the analogous of (2.13), as both relate the attitude information with the angular velocity of the spacecraft, with the difference that the quaternion formulation lacks singularities and is more easily integrable.

However, it is not intuitive to understand the spacecraft attitude through the quaternions, instead it is more natural to represent it in terms of Euler angles. Given a quaternion, it is possible to deduce the Euler angles from it. As mentioned above, the function relating a quaternion with the corresponding rotation matrix is bijective, and therefore the rotation matrix can be computed as:

$$\mathbf{R}_{BO} = \left({q_0}^2 - \tilde{\boldsymbol{q}} \cdot \tilde{\boldsymbol{q}}\right) \mathbf{I}_{3\times 3} + 2\tilde{\boldsymbol{q}}\tilde{\boldsymbol{q}}^T - 2q_0\mathbf{Q} \tag{2.18}$$

where $\mathbf{I}_{3\times 3}$ is the 3×3 unit matrix and $\mathbf{Q}$ is a 3×3 matrix given by:

$$\mathbf{Q} = \begin{bmatrix} 0 & -q_3 & q_2 \\ q_3 & 0 & -q_1 \\ -q_2 & q_1 & 0 \end{bmatrix} \tag{2.19}$$

Finally, Euler angles can be obtained from the entries of the attitude matrix $\mathbf{R}_{BO}$ [6].

2.4.3 Rotational dynamics

The rotational dynamics of a rigid body are governed by the time-derivative of its angular momentum $\mathbf{H}$. If $\mathbf{H}$ is calculated with respect to the center of mass of the rigid body, then the dynamics equation is given by:

$$\frac{d\mathbf{H}}{dt} = \mathbf{T} \tag{2.20}$$

where $\mathbf{T}$ is the resultant of all external torques about the center of mass. Expressing $\mathbf{H}$ in the body frame, the time derivative of the angular momentum becomes:

$$\dot{\mathbf{H}}_B + \boldsymbol{\omega}_B^{BI} \times \mathbf{H}_B = \mathbf{T}_B \tag{2.21}$$

Furthermore, we can express $\mathbf{H}$ as the inertia tensor $\mathbf{J}_B$ of the rigid body multiplied by the absolute angular velocity:

$$\mathbf{H}_B = \mathbf{J}_B \, \boldsymbol{\omega}_B^{BI} \tag{2.22}$$

with

$$\mathbf{J}_B = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix} \tag{2.23}$$

Assuming that the body axes are parallel to the principal axes, the off-diagonal products of inertia cancel out, yielding:

$$\mathbf{J}_B = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \tag{2.24}$$

In the general case the inertia tensor will vary over time, specifically in the case of a spacecraft losing mass, therefore its derivative should not be neglected. Substituting (2.22) into (2.21) leads to the Euler's equation of rotational motion:

$$\mathbf{J}_B \, \dot{\boldsymbol{\omega}}_B^{BI} + \boldsymbol{\omega}_B^{BI} \times \mathbf{J}_B \, \boldsymbol{\omega}_B^{BI} + \dot{\mathbf{J}}_B \, \boldsymbol{\omega}_B^{BI} = \mathbf{T}_B \tag{2.25}$$

where $\dot{\boldsymbol{\omega}}_B^{BI}$ is the inertial angular acceleration of the spacecraft and $\mathbf{T}_B$ is the resultant of all the external torques. Rearranging the terms, the equation of rotational motion can be written in the way used for the integration, that is:

$$\dot{\boldsymbol{\omega}}_B^{BI} = \mathbf{J}_B^{-1} \left[\mathbf{T}_B - \boldsymbol{\omega}_B^{BI} \times \mathbf{J}_B \, \boldsymbol{\omega}_B^{BI} - \dot{\mathbf{J}}_B \, \boldsymbol{\omega}_B^{BI} \right] \tag{2.26}$$

For our purpose, it is better to consider the attitude angles as derived from the LVLH frame instead of an inertial frame. Using the theorem of the relative velocities we can

write:

$$\begin{aligned} \boldsymbol{\omega}_B^{BI} &= \boldsymbol{\omega}_B^{BO} + \boldsymbol{\omega}_B^{OI} \\ \boldsymbol{\omega}_B^{BI} &= \boldsymbol{\omega}_B^{BO} + \mathbf{R}_{BO}\, \boldsymbol{\omega}_O^{OI} \end{aligned} \tag{2.27}$$

where $\boldsymbol{\omega}_O^{OI} = [0 \ -n \ 0]^T$ is the orbit angular velocity of the local orbital frame, and $\mathbf{R}_{BO}$ is the rotation matrix, derived from the sequence of rotations 3-2-1, which maps a generic vector from the local orbital frame $\mathcal{F}_{\mathcal{O}}$ into the body frame $\mathcal{F}_{\mathcal{B}}$.

2.4.4 Euler equations with reaction wheels

Reaction wheels are typical actuators used for the attitude control of micro-satellites. They are flywheels used to exchange angular momentum to the spacecraft by the principle of action and reaction in order to either rotate the spacecraft in a desired orientation or to store angular momentum for compensating a disturbing torque acting on the system. A single reaction wheel can control the rotation along one axis, so at least three reaction wheels are necessary for a 3-axis attitude control. Sometimes, more than three wheels are used for redundancy, since the failure of one wheel would cause the loss of control along that axis, resulting in a potential failure of the mission.

We consider a spacecraft featuring three reaction wheels, each along an axis of the body frame, here defined as $\hat{\mathbf{e}}_1$, $\hat{\mathbf{e}}_2$ and $\hat{\mathbf{e}}_3$. The angular momenta of the wheels are added to the angular momentum of the spacecraft body, yielding the total angular momentum of the system:

$$\mathbf{H} = \mathbf{H}_B + \sum_{i=1}^{3} I_w\, \omega_i\, \hat{\mathbf{e}}_i \tag{2.28}$$

where I_w is the moment of inertia of each reaction wheel about its spinning axis, assumed to be the same for the three reaction wheels, and ω_i is its absolute rotation speed.

Inserting equations (2.22) into equation (2.28) and taking the time derivative, the new Euler's equation can be written as:

$$\mathbf{J}_B\, \dot{\boldsymbol{\omega}}_B^{BI} + \dot{\mathbf{J}}_B\, \boldsymbol{\omega}_B^{BI} + \boldsymbol{\omega}_B^{BI} \times \left[\mathbf{J}_B\, \boldsymbol{\omega}_B^{BI} + \sum_{i=1}^{3} I_w\, \omega_i\, \hat{\mathbf{e}}_i \right] = \mathbf{T}_B - \sum_{i=1}^{3} I_w\, \dot{\omega}_i \tag{2.29}$$

with $I_w\, \dot{\omega}_i$ being the torque generated by the i-th reaction wheel and $\mathbf{T}_B$ being the sum of all the external torques acting on the system. This term can be seen as an input torque commanded to compensate for disturbances or to redistribute the angular momentum of the system for attitude maneuvers.

Chapter 3

Model Predictive Control

3.1 Introduction

Model Predictive Control (MPC) is a modern control method developed in the 1970s widely used in industrial control applications, such as in chemical plants, power electronics and even agriculture.

This method solves an optimization problem by the minimization of a cost function to generate the control action, while satisfying a set of constraints [7, 8]. Its main difference from a Linear Quadratic controller is the possibility to account for constraints while computing the optimal control action, which makes it a powerful tool for aerospace applications, where limits on thrust magnitude of the actuators, as well as obstacle avoidance are critical aspects to consider. When compared to classical control strategies, the main advantages of the MPC are:

- ability to deal with MIMO problems naturally;
- use of a mathematical model of the system dynamics to predict and optimize its behaviour over a future time horizon;
- ability to handle constraints;
- management of priorities by tuning the cost function.

3.2 Basic formulation of Model Predictive Control

The objective of MPC is to provide at every time step the optimal control action computed to minimize a chosen cost function. The term *optimal* refers to the preferences that were made during the design of the MPC, which may regard the minimal usage of actuators as well as the fastest achievement of the target states.

A mathematical model of the evolution of the system is used to predict the system states over a finite prediction horizon. The more accurate the model, the more efficient the control actions sequence.

At each time instant k, the predicted states build up the cost function V_k with a sequence of control actions as unknown over a control horizon, and subject to a set of linear constraints. This optimization problem takes the form of a Quadratic Programming[1] problem which can be solved by numerous algorithms. The set of input actions that minimize V_k is the optimal input sequence to be applied. However, only the first term of the control sequence, i.e. the control action at the current time instant k, is applied to the system, whereas the rest of the control sequence is discarded.

When the state at the next time step $k+1$ is measured, a new optimization problem based on the new information is solved, and a new control sequence is obtained. This process is repeated *on-line* at every time step, thus receding the prediction horizon and including *feedback* in the system. For this reason, this method is also referred to as Receding Horizon Control. Figure 3.1 illustrates the work principle of the MPC. We denote by p the length of the prediction horizon, over which anticipations of the system state are calculated, and denote by m, with $m < p$, the length of the control horizon.

Consider the following linear time-invariant system in discrete time and assuming no feed-through (i.e. $D = 0$):

$$\begin{aligned} x_{k+1|k} &= Ax_{k|k} + Bu_{k|k} \\ y_{k|k} &= Cx_{k|k} \end{aligned} \tag{3.1}$$

where x is an n_x-dimensional state vector, u is an n_u-dimensional input vector, and y is an n_y-dimensional vector of measured outputs. The subscript of $x_{k+1|k}$ indicates that the

[1] A Quadratic Program is an optimization problem in which the cost function is quadratic and the constraint set is defined by linear inequalities.

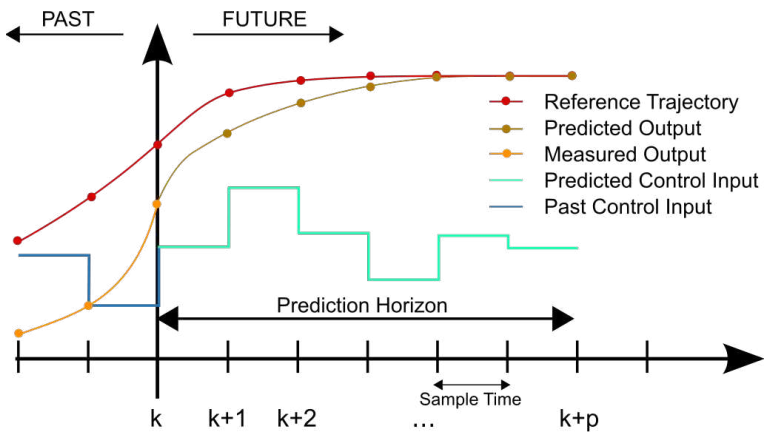


Figure 3.1: Receding horizon principle [9].

predicted value of the state vector at time $k+1$ is based on the value of x known (or observed) at instant k.

We denote by $r_{k+j|k}$ the reference output that the controlled variables $y_{k+j|k}$ are required to follow, such that $e_{k+j|k} = y_{k+j|k} - r_{k+j|k}$ is the tracking error vector. We also define $\Delta u_{k+j|k} = u_{k+j|k} - u_{k|k}$ as the increment of the control vectors between two time steps. We can now define a cost function as:

$$V_k = \sum_{j=0}^{p-1} (e_{k+j|k}^T \, Q \, e_{k+j|k}) + \sum_{j=0}^{m-1} (\Delta u_{k+j|k}^T \, R \, \Delta u_{k+j|k}) + e_{k+p|k}^T \, P \, e_{k+p|k} \tag{3.2}$$

where Q and P are symmetric semi-positive definite matrices and R is a positive definite matrix. These weight matrices allow us to penalize error in tracking and actuator usage by choosing larger values for a matrix with respect to the other one. Sometimes, for stability reasons, a different weight matrix P is considered for the last time step of the prediction horizon.

The goal is to find an optimal input control sequence $\Delta \mathcal{U}_k^* = [\, \Delta u_{k|k}^{*T}, \, \Delta u_{k+1|k}^{*T}, \, \ldots, \, \Delta u_{k+m-1}^{*T}]^T$ that minimizes the cost function V_k. It is an optimization problem, which can be formulated as follows:

$$V_k^* = \min_{\Delta u_k^*} \sum_{j=0}^{p-1} (e_{k+j|k}^T \, Q \, e_{k+j|k}) + \sum_{j=0}^{m-1} (\Delta u_{k+j|k}^T \, R \, \Delta u_{k+j|k}) + e_{k+p|k}^T \, P \, e_{k+p|k} \tag{3.3}$$

subject to

$$x_{k+j+1|k} = Ax_{k+j|k} + Bu_{k+j|k}$$

$$y_{k+j|k} = Cx_{k+j|k}$$

$$u_{k+j+1|k} = u_{k+j|k} + \Delta u_{k+j|k} \tag{3.4}$$

$$\Delta u_{k+i|k} = 0 \,, \ \forall i = m, \ldots, p-1 \tag{3.5}$$

$$E \begin{bmatrix} \Delta u_{k+j} \\ \\ 1 \end{bmatrix} \leq 0 \,, \ \forall j = 0, 1, \ldots, m-1 \tag{3.6}$$

$$F \begin{bmatrix} u_{k+j} \\ \\ 1 \end{bmatrix} \leq 0 \,, \ \forall j = 0, 1, \ldots, m-1 \tag{3.7}$$

$$G \begin{bmatrix} y_{k+j} \\ \\ 1 \end{bmatrix} \leq 0 \,, \ \forall j = 0, 1, \ldots, p \tag{3.8}$$

where the first two equations refer to the system model of the plant; Equation (3.5) states that the input control value does not change after the control horizon, remaining constant until the end of the prediction horizon; Equations (3.6), (3.7) and (3.8) are the constraints applied to the system in terms of actuators slew rate, magnitude and output limitations respectively.

Sometimes it is convenient to parameterize the model (3.1) using the increment vector $\Delta u_{k|k}$, as the cost function penalizes it, rather than $u_{k|k}$. We may therefore augment the state vector as follows:

$$\widetilde{x}_k = \begin{bmatrix} x_k \\ \\ u_{k-1} \end{bmatrix} \tag{3.9}$$

so that the input control values of the former time step is included in the state vector. Substituting (3.9) into (3.1) we obtain a new model for the plant:

$$\begin{bmatrix} x_{k+1|k} \\ u_{k|k} \end{bmatrix} = \begin{bmatrix} A & B \\ 0 & I \end{bmatrix} \begin{bmatrix} x_{k|k} \\ u_{k-1} \end{bmatrix} + \begin{bmatrix} B \\ I \end{bmatrix} \Delta u_{k|k} \tag{3.10}$$

$$y_{k|k} = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x_{k|k} \\ u_{k-1} \end{bmatrix} \tag{3.11}$$

3.3 Formulation as a QP problem

In order to solve the optimization problem, it is necessary to put it into the form of a Quadratic Programming (QP) problem [7], defined as:

$$\min_{\theta} \frac{1}{2}\theta^T \mathcal{H}\, \theta - \theta^T \mathcal{G} \qquad \text{subject to} \qquad \Omega\, \theta \leq \omega \tag{3.12}$$

It is proper to make the following definitions:

- $\mathcal{Y}_k = \begin{bmatrix} y_{k|k}^T, & y_{k+1|k}^T, & \dots, & y_{k+p|k}^T \end{bmatrix}^T$ is the sequence of predicted output vectors;
- $\Delta\mathcal{U}_k = \begin{bmatrix} \Delta u_{k|k}^T, & \Delta u_{k+1|k}^T, & \dots, & \Delta u_{k+m-1|k}^T \end{bmatrix}^T$ is the sequence of input changes within the control horizon;
- $\mathcal{U}_k = \begin{bmatrix} u_{k|k}^T, & u_{k+1|k}^T, & \dots, & u_{k+m-1|k}^T \end{bmatrix}^T$ is the sequence of input values within the control horizon;
- $\mathcal{T}_k = \begin{bmatrix} r_{k|k}^T, & r_{k+1|k}^T, & \dots, & r_{k+m-1|k}^T \end{bmatrix}^T$ is the sequence of reference vectors within the prediction horizon.

Prediction of the output vector $y_{k+j|k}$ at a future time $k+j$ can be computed by applying the predictive model (3.1) recursively. This vector depends upon the current state $x_{k|k}$ at which the prediction is made and the assumed future input changes $\Delta\mathcal{U}_k$ that

will be applied (and that correspond to the degrees of freedom). Using these definitions we can write:

$$\mathcal{Y}_k = \Psi x_{k|k} + \Upsilon u_{k-1} + \Theta \Delta \mathcal{U}_k \tag{3.13}$$

where:

$$\Psi = \begin{bmatrix} CA \\ \vdots \\ CA^m \\ CA^{m+1} \\ \vdots \\ CA^p \end{bmatrix}, \ \Gamma = \begin{bmatrix} CB \\ \vdots \\ \sum_{i=0}^{m-1} CA^iB \\ \sum_{i=0}^{m} CA^iB \\ \vdots \\ \sum_{i=0}^{p-1} CA^iB \end{bmatrix}, \ \Theta = \begin{bmatrix} CB & \dots & 0 \\ CAB+CB & \dots & 0 \\ \vdots & \ddots & \vdots \\ \sum_{i=0}^{m-1} CA^iB & \dots & CB \\ \sum_{i=0}^{m} CA^iB & \dots & CAB+CB \\ \vdots & \vdots & \vdots \\ \sum_{i=0}^{p-1} CA^iB & \dots & \sum_{i=0}^{p-m} CA^iB \end{bmatrix}$$

First, we want to express the cost function V_k in the form of (3.12). We see that V_k can also be written with a different notation that emphasizes the quadratic problem:

$$V_k = \sum_{i=0}^{p} \|y_{k+i|k} - r_{k+i|k}\|^2_{Q_{(i)}} + \sum_{i=0}^{m-1} \|\Delta u_{k+i|k}\|^2_{R_{(i)}} \tag{3.14}$$

where the weight matrices $Q_{(i)}$ and $R_{(i)}$ refer to the time step i, so that they may penalize the error and the input control variables in different ways over the prediction horizon. Compacting this equation we get:

$$V_k = \|\mathcal{Y}_k - \mathcal{T}_k\|^2_{\mathcal{Q}} + \|\Delta \mathcal{U}_k\|^2_{\mathcal{R}} \tag{3.15}$$

where

$$\mathcal{Q} = \begin{bmatrix} Q_{(0)} & 0 & \cdots & 0 \\ 0 & Q_{(1)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & Q_{(p)} \end{bmatrix}, \ \mathcal{R} = \begin{bmatrix} R_{(0)} & 0 & \cdots & 0 \\ 0 & R_{(1)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & R_{(m-1)} \end{bmatrix}$$

Defining the vector:

$$\mathcal{E}_k \coloneqq \mathcal{T}_k - \mathcal{Y}_k \Big|_{\Delta \mathcal{U}_k = 0} = \mathcal{T}_k - \Psi x_k - \Upsilon u_{k-1} \tag{3.16}$$

as the tracking error vector in absence of a future control sequence, i.e., the error sequence that will occur if $\Delta\mathcal{U}_k = 0$, it is now possible to write the cost function in the form:

$$V_k = \Delta\mathcal{U}_k^T \, \mathcal{H} \, \Delta\mathcal{U}_k - \Delta\mathcal{U}_k^T \, \mathcal{G} + const \tag{3.17}$$

which is the standard formulation for a QP problem, where:

$$\mathcal{H} = \Theta^T \, \mathcal{Q} \, \Theta + \mathcal{R}$$

$$\mathcal{G} = 2\Theta^T \, \mathcal{Q} \, \mathcal{E}_k$$

At this point we need to put the constraint inequalities into the QP formulation (3.12). To begin, we see that inequalities (3.6), (3.7) and (3.8) can be written as

$$E \begin{bmatrix} \Delta\mathcal{U}_k \\ 1 \end{bmatrix} \leq 0 \tag{3.18}$$

$$F \begin{bmatrix} \mathcal{U}_k \\ 1 \end{bmatrix} \leq 0 \tag{3.19}$$

$$G \begin{bmatrix} \mathcal{Y}_k \\ 1 \end{bmatrix} \leq 0 \tag{3.20}$$

It is necessary to express all of them as a function of $\Delta\mathcal{U}_k$ in order to reach a more compact form. The first inequality can be easily expressed as:

$$\mathcal{W} \, \Delta\mathcal{U}_k \leq w \tag{3.21}$$

Under many circumstances the input changes are required not to exceed a maximum absolute value expressed by the simple inequalities $-\Delta u_{max,k+j} \leq \Delta u_{k+j|k} \leq \Delta u_{max,k+j}$. In this special case, the matrix $\mathcal{W}$ and the vector w are simply:

$$\mathcal{W} = \begin{bmatrix} I & 0 & \cdots & 0 \\ 0 & I & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & I \\ -I & 0 & \cdots & 0 \\ 0 & -I & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & -I \end{bmatrix}, \quad w = \begin{bmatrix} \Delta u_{max,k} \\ \vdots \\ \Delta u_{max,k+m-1} \\ \Delta u_{max,k} \\ \vdots \\ \Delta u_{max,k+m-1} \end{bmatrix}$$

With regard to the second inequality (3.19), some more manipulations are necessary, as it is not immediate to express it as a function of $\Delta\mathcal{U}_k$. Assuming that F has the form $F = \begin{bmatrix} F_1, & F_2, & \ldots, & F_m, & f \end{bmatrix}$, where f is the last column vector, then inequality (3.19) can be rewritten as:

$$\sum_{i=1}^{m} F_i \, u_{k+i-1|k} + f \leq 0$$

It is convenient to make the following definitions:

$$\mathcal{F}_i \coloneqq \sum_{j=i}^{m} F_j \quad \text{and} \quad \mathcal{F} \coloneqq \begin{bmatrix} \mathcal{F}_1, & \ldots, & \mathcal{F}_m \end{bmatrix}$$

and (3.19) can finally be written in the form:

$$\mathcal{F}\,\Delta\mathcal{U}_k \leq -\mathcal{F}_1 u_{k-1} - f \tag{3.22}$$

Inequality (3.20) can be rearranged by substituting Equation (3.13) into it and, assuming G is in the form $G = \begin{bmatrix} \Gamma\,, & g \end{bmatrix}$ so that the inequality becomes:

$$\Gamma\,\Theta\,\Delta\mathcal{U}_k \leq -\Gamma\left[\Psi x_{k|k} + \Upsilon u_{k-1}\right] - g \tag{3.23}$$

We have obtained three linear inequalities describing the constraints of the plant in the variable $\Delta\mathcal{U}_k$, so it is now possible to assemble them and obtain a single inequality:

$$\begin{bmatrix} \mathcal{W} \\ \mathcal{F} \\ \Gamma\Theta \end{bmatrix} \Delta\mathcal{U}_k \leq \begin{bmatrix} w \\ -\mathcal{F}_1 u_{k-1} - f \\ -\Gamma\left[\Psi x_{k|k} + \Upsilon u_{k-1}\right] - g \end{bmatrix} \tag{3.24}$$

that is the same formulation as in the QP problem (3.12). This optimization problem can be solved by several algorithms, as described in [7].

3.4 Observer-based MPC

In most practical systems it is impossible to measure the full state vector x_k directly. Indeed, it is only possible to measure the output vector y_k of the plant using some sensors, and an observer is employed to produce an estimation of the full state vector, $\hat{x}_k$. The working principle of a state observer is shown in Figure 3.2.

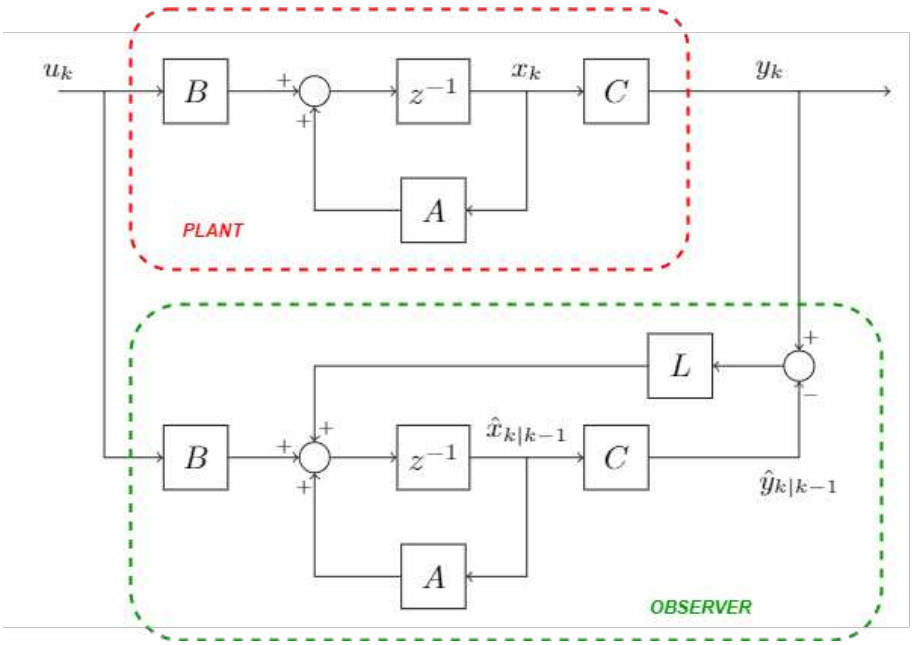


Figure 3.2: Block diagram of a state observer.

The structure of the observer is identical to the plant, with the addiction of the gain matrix L which is used as feedback to correct the state estimate $\hat{x}$. The equation of the

observer is:

$$\hat{x}_{k+1|k} = A\hat{x}_{k|k-1} + Bu_k + L\left(y_k - \hat{y}_{k|k-1}\right) \tag{3.25}$$

where the latter term is the correction due to the difference between the measured output and its prediction. This difference may be caused by unknown disturbances affecting the output of the plant.

In order to select an appropriate gain matrix L such that the state estimation converges to the real state, we first define the state estimation error $e_k = x_k - \hat{x}_{k|k-1}$. After some passages we get:

$$e_{k+1} = (A - LC)\, e_k \tag{3.26}$$

which shows that the estimation error will converge to zero if the eigenvalues of $A - LC$ lie inside the unit disk. If the pair (A, C) is observable, then a gain matrix L exists, such that the eigenvalues can be placed in a desired location.

Once a state estimator has been defined, the state vector x presents in Equation (3.1) can be substituted with its estimation $\hat{x}$, and the formulation of the MPC remains unchanged.

3.4.1 Kalman Filter

In this section a brief description of the working principle of the Kalman filter will be presented [10].

It is quite often in control systems that not all critical states of a system can be measured directly through sensors. In our case, for example, we have assumed that the chaser spacecraft is equipped with vision-based sensors capable of measuring the relative distance between the two satellites, but not the relative velocity. This is where a state observer comes in, feeding back the missing states (velocities) from the measured ones (distances). Moreover, measurements provided by sensors are in reality subjected to uncertainties, which are caused by mis-calibration of the instruments or small imperfections. In the former case, we have bias errors, whereas in the latter case we have random errors. Assuming that all sensors are well calibrated, the bias error can be cancelled, but the random noise will nevertheless be present.

A state observer should not only provide the lacking states, but also be able to distin-

guish real data from the noise perturbations. In the general case, the system is influenced by processing noise, which takes place as an additional input, and measurements noise, which adds to the output variables. The discrete-time system is then given by:

$$\begin{aligned} \mathrm{x}_{n+1} &= A\mathrm{x}_n + B\mathrm{u}_n + Gw_n \\ \mathrm{y}_n &= C\mathrm{x}_n + D\mathrm{u}_n + Hw_n + v_n \end{aligned} \tag{3.27}$$

where w_n and v_n are the processing and measurement noise signals, respectively, assumed to be Gaussian noise with zero mean. G and H are matrices of appropriate dimensions. Usually H is zero, specifying that there is no feed-through of the processing noise to the output. The noisy signals w and v satisfy the relations:

$$\begin{aligned} Q &= E\left\{w_n w_n^T\right\} \\ R &= E\left\{v_n v_n^T\right\} \\ N &= E\left\{w_n v_n^T\right\} \end{aligned} \tag{3.28}$$

where Q, R, and N are the process noise, measurement noise, and noise cross covariance matrices, respectively. The Kalman estimator computes the state estimates $\hat{\mathrm{x}}$ that minimizes the steady-state error covariances, given by:

$$\begin{aligned} P &= \lim_{n\to\infty} E\left\{\left(\mathrm{x}_n - \hat{\mathrm{x}}_{n|n-1}\right)\left(\mathrm{x}_n - \hat{\mathrm{x}}_{n|n-1}\right)^T\right\} \\ Z &= \lim_{n\to\infty} E\left\{\left(\mathrm{x}_n - \hat{\mathrm{x}}_{n|n}\right)\left(\mathrm{x}_n - \hat{\mathrm{x}}_{n|n}\right)^T\right\} \end{aligned} \tag{3.29}$$

by solving a discrete algebraic Riccati equation (DARE), thereby obtaining the filter gain L and the optimal innovation matrices M_x, M_y.

The Kalman filter takes as inputs the vector u and the measured variables y, and returns the estimates $\hat{\mathrm{y}}$ of the plant outputs and the state estimates $\hat{\mathrm{x}}$. The update equations:

$$\begin{aligned} \hat{\mathrm{x}}_{n|n} &= \hat{\mathrm{x}}_{n|n-1} + M_x\left(y_n - C\hat{\mathrm{x}}_{n|n-1} - Du_n\right) \\ \hat{\mathrm{y}}_{n|n} &= C\hat{\mathrm{x}}_{n|n-1} + Du_n + M_y\left(y_n - C\hat{\mathrm{x}}_{n|n-1} - Du_n\right) \end{aligned} \tag{3.30}$$

correct the current state and output estimates by means of the innovation matrices M_x, M_y, given knowledge of the *a priori* state estimates $\hat{\mathrm{x}}_{n|n-1}$ and the current measurements y_n.

Subsequently, the state equation of the Kalman filter, given by:

$$\hat{\mathrm{x}}_{n+1|n} = A\hat{\mathrm{x}}_{n|n-1} + Bu_n + L\left(y_n - C\hat{\mathrm{x}}_{n|n-1} - Du_n\right) \tag{3.31}$$

is used to predict the state estimate for the next time step.

3.5 Measured disturbances and feed-forward

As mentioned in Section 3.2, having an accurate predictive model is an ideal situation, in that the MPC is capable of predicting with high fidelity the state of the system over the prediction horizon. However, this is not feasible in reality because of many factors that come in during the design of an MPC, as for instance linearization of the dynamic equations in the predictive model, unknown disturbances affecting the system, measurement noise etc.

What can be done to improve the accuracy of the predictive model is take into account the effect of those disturbances for which a model is known. If this is possible, then the controller will be able to predict those effects in advance and compute a corrective control action to approximately cancel them out. This action is called *feedforward* control and its benefit is the ability to generate a suitable control action before the disturbance effects affect the system, rather than wait for the disturbances to produce an error in the output variables and take corrective actions thereafter, as in the feedback control scheme.

3.6 Explicit MPC

Solving an online optimization problem at every sample time may be very expensive for what concerns computational cost, and potentially unfeasible. This might be a problem for those low-end spacecraft (like Nanosatellites), which on-board computing capacity is more limited than others. In fact, it is more convenient to possess solutions which require the least computational effort and power consumption, as saved capacity can be used for additional functions and services.

However, there exists the possibility to radically reduce the computational costs of an MPC controller in cases where the plant can be approximated by a linear time-invariant (LTI) model. This is the case of an explicit MPC: if the model and the constraints are LTI and the quadratic cost is time invariant, a closed form solution to the optimization problem exists and can be calculated off-line.

Bemporad et al. showed in [11] that the optimization problem is a multi-parametric

quadratic program (mp-QP) and proposes an algorithm for its solution. Once the mp-QP problem has been solved off-line, the feedback control law is available explicitly and assumes the form of a piecewise affine function, which can be easily stored in a look-up table inside the memory of an on-board computer.

The choice of an explicit MPC is appropriate for systems featuring low computing power, although the allocation of the solutions inside the on-board computer may require significant storage. Therefore its selection over a traditional MPC must be a trade-off between the two options.

Chapter 4

PID control

The Proportional - Integral - Derivative (PID) controller is one of the most common control strategies. Its simplicity and high reliability make it a good choice for many applications and is therefore widely employed in industrial control systems. As the name suggests, the PID control action $u(t)$ is the sum of three distinct terms $u(t) = P(t) + I(t) + D(t)$, where $P(t)$ is the proportional term, $I(t)$ is the integral term and $D(t)$ is the derivative contribution. The following sections describes each contribution in details.

4.1 Proportional action

The proportional term is computed as the multiplication of the instantaneous error $e(t)$ between the reference state $r(t)$ and the current state $y(t)$ by a constant gain K_p. In continuous-time domain it is given by:

$$P(t) = K_p \left[r(t) - y(t)\right] = K_p\, e(t)$$

whereas for discrete time, it can be computed as:

$$P(k) = K_p\, e(k)$$

In Laplace domain, the transfer function of a proportional controller is simply given by:

$$H_P(s) = K_p$$

Similarly, for the discrete controller:

$$H_P(z) = K_p$$

The corrective action is therefore proportional to the amount of error at that instant. A high value of K_p can lead to very high control actions, despite the limitations in the actuators. A purely proportional controller is poor, because the step response shows a steady-state error that cannot be removed. Moreover, high values of the proportional gain can produce high oscillations and overshoot.

4.2 Derivative action

The derivative term is a control action proportional to the time derivative of the error:

$$D(t) = K_d \frac{de(t)}{dt} \tag{4.1}$$

The transfer function is:

$$H_D(s) = K_d\, s$$

In discrete-time formulation, the time derivative of the error can be approximated as:

$$\frac{de(t)}{dt} = \frac{e(k) - e(k-1)}{\Delta t} \tag{4.2}$$

and substituting into Equation (4.1) we can formulate the derivative action for a discrete-time controller as:

$$D(k) = K_d \frac{e(k) - e(k-1)}{\Delta t} \tag{4.3}$$

where Δt is the sample time interval. It is possible to write the discrete transfer function as:

$$H_D(z) = K_d \frac{1 - z^{-1}}{\Delta t} \tag{4.4}$$

The benefit of using a derivative correction consists of damping of oscillations; however, non-zero steady state error will be present if no integral action is included.

Such derivative action can improve the performances of an ideal controller. In practice, where measurements are affected by sensors noise, some precautions have to be taken

into account. The measured variable y_m can be written as the sum of the real controlled variable and an error ε introduced by the sensors:

$$y_m(t) = y(t) + \varepsilon(t)$$

Taking the time derivative of $y_m(t)$ may result in large values due to the high rate of change of $\varepsilon(t)$, especially if its spectrum presents high frequency components. This problem can be overcome by using also a low-pass filter (LPF) instead of a pure derivative contribution. The block diagram of the controller with a derivative term is visible in Figure 4.1, where the LPF transfer function is given by

$$H_{LPF}(s) = \frac{1}{\frac{T_d}{N}s + 1} \tag{4.5}$$

where $T_d = \dfrac{K_d}{K_p}$ and N is a parameter used to tune the filter.

Since the derivative term is proportional to the rate of change of the error, it may be possible to reformulate it as an action proportional to the velocity error. This is possible in our case, as the velocity reference and the current velocity of the chaser are both known, thus allowing to compute a derivative control action proportional to their difference e_{vel}:

$$D(k) = K_d\, e_{vel}(k) \tag{4.6}$$

In this case, the transfer function is simply given by the constant K_d, as in the proportional control.

4.3 Integral action

The integral term produces a control action proportional to the accumulated error over time. It can thus be expressed as:

$$I(t) = K_I \int_0^t e(\tau)\, d\tau \tag{4.7}$$

or, in discrete-time domain:

$$I(k) = I(k-1) + K_I\, e(k)\, \Delta t \tag{4.8}$$

The transfer functions in Laplace and Z domains are respectively given by:

$$\begin{aligned} H_I(s) &= \frac{K_I}{s} \\ H_I(z) &= \frac{\Delta t}{1 - z^{-1}} K_I \end{aligned} \tag{4.9}$$

The integral action brings an important advantage to the system, as it is responsible for the cancellation of the steady state error at a step response.

The drawback of an integral action is the problem known as *windup*, which occurs when the integral term accumulates a large error after a significant rise of the setpoint, and this may lead to the instability of the control. One possible solution to this problem is given by the anti-reset windup scheme as in Figure 4.1.

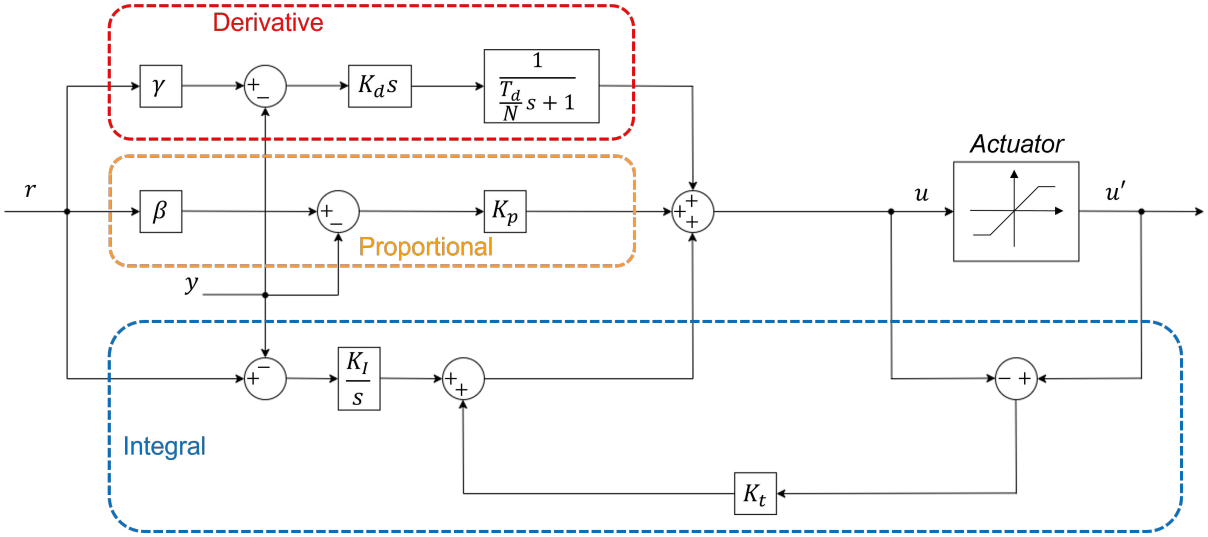


Figure 4.1: Block diagram of a PID controller with low-pass filter in the derivative term and anti-reset windup scheme in the integral term.

4.4 Closed-loop stability

When designing a controller it is important to analyze the stability of the closed-loop system and verify that the outputs meet the desired requirements.

For continuous-time systems, the stability is analyzed by the study of the poles of the closed-loop transfer function, which are the eigenvalues of the transition matrix of the closed-loop state space system [12]. If the real part of each pole is negative or null, then the closed-loop system is asymptotically stable or marginally stable, respectively; if the

real part of each pole is positive, then the system is unstable. For discrete-time systems, the left side of the real plane is mapped into a unit circle, and therefore the closed-loop poles need to lie inside this unit disk to guarantee stability to the system, as shown in Figure 4.2.

The block diagram of the translational dynamics with with a PID control is shown in Figure 4.3. Note that the PID block is made up of three PID controllers, one for each axis. Since the PID controllers present a discrete-time formulation, a discretization of the CW equations is necessary by means of a zero-order-hold (ZOH) block, with sampling time T_s. The sensor block represented by H has been considered unitary for simplicity in the closed-loop stability check.

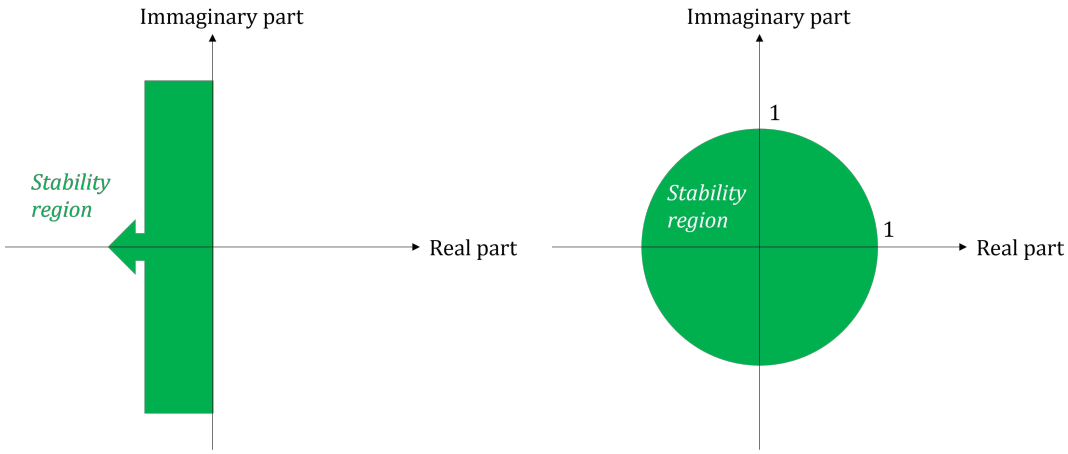


Figure 4.2: Stability region for a continuous-time system on the left, and a discrete-time system on the right.

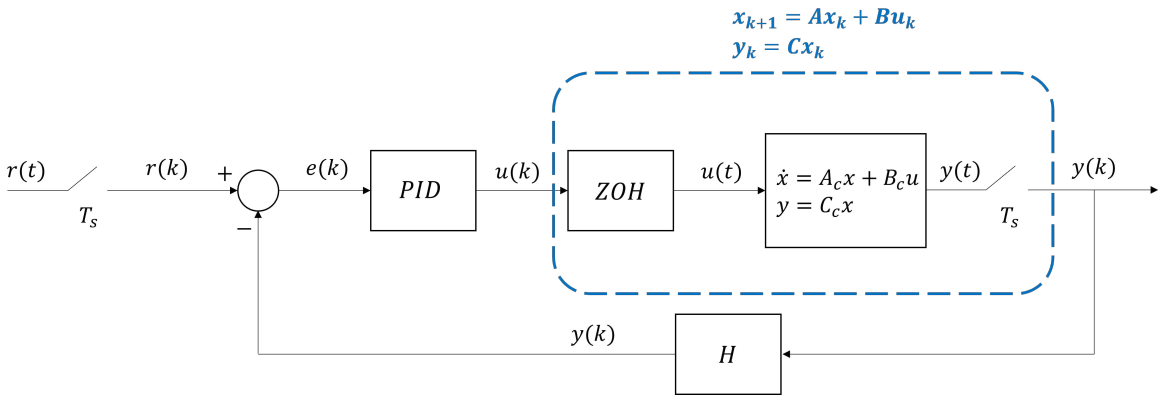


Figure 4.3: Block diagram of a closed-loop system with discrete-time PID.

Chapter 5

Rendezvous and docking scenario definition

In order to create a simulation representative of a realistic scenario for RVD operations, several conditions have been reproduced from real space missions realized in the last years or planned for the near future. The base idea of the simulation is to consider a CubeSat with controller based on MPC technique, and to simulate RVD maneuvers with a target station.

The following descriptions constitute examples of realistic scenarios which the numerical simulations refer to.

5.1 Case studies

5.1.1 Optical Communications and Sensors Demonstration Program

A mission of relevant interest for RVD operations involving CubeSats was the NASA Optical Communications and Sensors Demonstration (OCSD) program [13]. One of the goals of the mission was to realize proximity operations between AeroCube-OCSD-B and AeroCube-OCSD-C, two 1.5U CubeSats of 2.31 kg deployed from the ISS on December

7, 2017, into a circular orbit at an altitude of 450 km. Both CubeSats were equipped with GPS receivers and a three-axis attitude control system featuring two-axis sun sensors, Earth horizon sensors, an Earth nadir sensor, three-axis magnetometers, three-axis rate gyros, magnetic torque rods, reaction wheels, and star trackers, allowing a pointing accuracy better than 0.15°. A steam thruster was mounted on each satellite, which could generate a force of approximately 3 mN. The first objective of OCSD proximity operations was to insert AeroCube-OCSD-B into a co-orbital corkscrew orbit about the AeroCube-OCSD-C CubeSat with little risk of collision. This was achieved in May 2018, reaching a minimum range of 151 m.

5.1.2 Automated Transfer Vehicle

Another suitable example for RVD is the Automated Transfer Vehicle (ATV), the European space cargo ship operated from 2008 until 2015 to resupply the ISS. With its 4 main engines, 28 RCS thrusters and a mass of 19 600 kg, the ATV was designed to transport up to 6900 kg of cargo [1]. The main engines where capable of generating a thrust of 490 N, whereas the RCS thruster produced a level of thrust of 220 N. The ATV used to dock to the V-bar port on the Service Module of the ISS with a nominal contact velocity of $0.05 \div 0.10$ m/s.

5.1.3 CubeSat Proximity Operations Demonstration

The CubeSat Proximity Operations Demonstration (CPOD) mission is devised to demonstrate relative station keeping, precise rendezvous, proximity, and docking technologies between two identical 3U CubeSats [14]. The two spacecraft, each with a mass of roughly 5.5 kg, will be deployed into orbit together, and after a series of check-out procedures the proximity operations will begin. Some of the key technologies include miniaturized advanced avionics, miniaturized relative navigation sensors, and a multi-thruster cold propulsion system capable of supporting a total impulse of approximately 30 m/s. The ADCS subsystem can provide an attitude estimation better than 0.15° and a pointing

control better than 0.18° (3σ), featuring star trackers, MEMS IMU, reaction wheels, GPS receiver, sun sensors, magnetometers, and torque coils.

5.2 Chaser spacecraft

The spacecraft considered for the RVD simulations in this thesis is a 3U CubeSat. The reasons behind this choice are motivated by the increasing number of Nanosatellites put into orbit over the last years and the evolution of technology that will grant cutting-edge computational solutions even for small spacecraft.

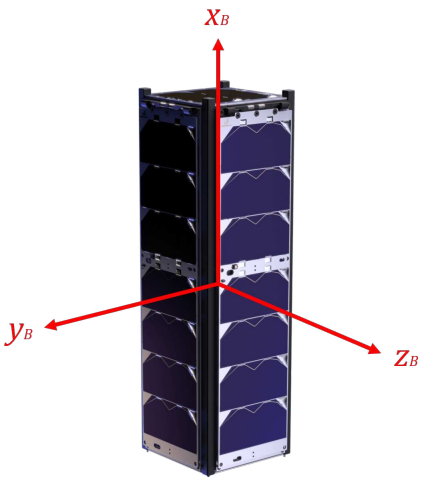


Figure 5.1: Chaser: 3U CubeSat.

The mass of the spacecraft is assumed to be $m = 4.0\,\text{kg}$ and equally distributed, and the three principal body axes oriented as in Figure 5.1, allowing an easy calculation of the moments of inertia in the body frame by the formulas:

$$\begin{aligned} I_x &= \frac{1}{12} m \left(w^2 + d^2\right) \\ I_y &= \frac{1}{12} m \left(w^2 + h^2\right) \\ I_z &= \frac{1}{12} m \left(d^2 + h^2\right) \end{aligned} \tag{5.1}$$

where w, d, and h for a 3U CubeSat are 10, 10, and 30 cm, respectively. The off-diagonal products of inertia I_{xy}, I_{xz}, and I_{yz} are null due to the symmetry of the spacecraft, so the

inertia tensor of the 3U CubeSat is:

$$\mathbf{J}_B = 10^{-3} \begin{bmatrix} 6.67 & 0 & 0 \\ 0 & 33.3 & 0 \\ 0 & 0 & 33.3 \end{bmatrix} \text{kg m}^2$$

5.2.1 Propulsion system

Typical propulsion systems employed on small satellites consist of cold gas thrusters, plasma thrusters and resistojets. Their authority is generally in the order of mN [15, 16], therefore a thrust of $F_{max} = 50\,\text{mN}$ enacting a cold gas propulsion system was considered for the chaser thrust system. Specifically, the parameters listed in Table 5.1 was deemed adequate for this scenario:

Table 5.1: Propulsion system of the chaser.

PROPULSION SYSTEM	
Maximum thrust	50 mN
Specific Impulse I_{sp}	50 s
Minimum Impulse Bit (MIB)	0.1 mN s

The minimum impulse bit translates into a minimum time of opening of the propulsion system solenoid valve, which for a thrust of 50 mN corresponds to 2 ms. This delay was taken into account in the simulations presented in the following chapters, assuring that every time the controller demanded a solenoid valve opening time shorter than the minimum value, the solenoid valve would remain shut and no force would be exerted.

The actuators of the spacecraft, besides, are assumed to work in Pulse-Width-Modulation (PWM) mode. This means that the thrusters cannot vary their thrust magnitude, which is equal to the maximum force they can produce. In order to produce an equivalent impulse, a duty cycle has to be derived at every time step to determine the fraction of time in which the thrusters have to be activated, as illustrated in Figure 5.2.

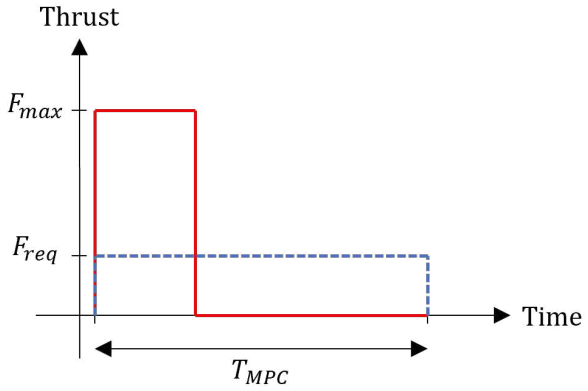


Figure 5.2: Illustration of PWM and PWC modes in a time interval.

This operating mode of the solenoid valves was accounted for in the integration of the dynamic equations.

In order to simulate real thrusters, a Gaussian percentage error of 10% (3σ) of the maximum authority was considered in the simulations.

5.2.2 Attitude control system

As for the attitude actuators, a 3-axis reaction wheel assembly seemed to be appropriate. The actuators selected are three BCT Micro Reaction Wheels, each one mounted along a principal body axis. Their characteristics are summarized in Table 5.2

Table 5.2: Characteristics of the reaction wheels.

REACTION WHEEL	
Maximum Torque	6 mN m
Maximum Speed	6500 rpm
Moment of Inertia	2.2036×10^{-5} kg m^2
Maximum Momentum	15 mN m s

Regarding the torque generated by each reaction wheel, a Gaussian error of up to 10% of the commanded magnitude was added to the control action in order to simulate

small imperfections and uncertainties. In addition, a minimum torque was considered, under which the actuator does not respond. This value of $2.3 \times 10^{-4}\,\mathrm{mN\,m}$ was chosen considering a wheel speed resolution of 0.1 rpm and multiplying it by its moment of inertia.

As done for the thrusters, the modeling of ideal reaction wheels may be approximated by adding a Gaussian error of 10% (3σ) of the commanded torque.

All parameters of the chaser are summarized in Table 5.3.

Table 5.3: Characteristics of the chaser.

CHASER SPACECRAFT	
Spacecraft type	CubeSat 3U
Dimensions	$10\,\mathrm{cm} \times 10\,\mathrm{cm} \times 30\,\mathrm{cm}$
Mass	$4.0\,\mathrm{kg}$
I_x	$6.67 \times 10^{-3}\,\mathrm{kg\,m^2}$
I_y	$3.33 \times 10^{-2}\,\mathrm{kg\,m^2}$
I_z	$3.33 \times 10^{-2}\,\mathrm{kg\,m^2}$
Thrust	$50.0\,\mathrm{mN}$
Torque	$6.0\,\mathrm{mN\,m}$

5.2.3 Measurements

The measurements generated by the on-board sensors have been modeled by adding appropriate Gaussian errors to the plant outputs. Specifically, it has been assumed that the on-board sensors can only measure relative position, as it is often the case, whilst the relative velocity needs to be estimated employing a Kalman filter.

During the first corridor, from S3 to S4, the relative position is altered with a noisy signals as in [17], which express the estimation accuracy achieved by the ATV during this transfer. These values are provided in Table 5.4.

Table 5.4: Chaser's state at the docking point.

Estimation error (3σ)			
Range $[m]$	x $[m]$	y $[m]$	z $[m]$
$-250 \div -100$	11.31	3.89	4.21
$-100 \div -20$	1.45	0.59	0.59

From point S4 the relative position measurements are obtained using the rule of thumb of considering 1% of the range [1], i.e. $\sim 10\,\text{cm}$. From there on, the accuracy is increased linearly up to the docking point, where it has been chosen to adopt an accuracy of one order of magnitude better than the maximum misalignment allowed by the docking mechanism (which is discussed later in Section 5.5), as suggested by Fehse in [1].

Concerning the relative velocity between the two spacecraft, according to [1] a value of a few mm/s has been considered adequate for the velocity accuracy at docking, being the optimal relative speed of $2\,\text{cm/s}$. The Kalman filters used to estimate the relative velocity from the relative position measurements have therefore been tuned such that a similar accuracy is obtained.

Relative position measurements can be obtained by Vision Based Navigation, which was used by the ATV, with various retroreflectors positioned on the ISS that were illuminated by LEDs on the chaser, and observed by a camera. This method is described in [18] showing that it is also a feasible solution for CubeSats.

Regarding attitude measurements, the rule of thumb of an attitude estimate of one order of magnitude better than the pointing accuracy has been applied, similarly to [18]. Therefore we have assumed that an ADCS subsystem like the one of the CPOD mission is adopted. Unlike translational velocity, the angular rate is simply estimated adding such an uncertainty to the actual value, thereby avoiding the use of an Extended Kalman Filter, guaranteeing a pointing accuracy of 0.15°.

5.3 Target station

In order to reproduce a well known orbiting station, the ISS parameters have been considered for the target spacecraft. The station is in circular orbit at an altitude of 450 km, with inclination of 51.6°. Other relevant parameters, are given by [19]:

- $C_D = 2.2$;
- $A = 130\,\mathrm{m}^2$;
- $m = 419\,455\,\mathrm{kg}$

where C_D is the drag coefficient, A is the frontal area in the direction of motion, m is the mass.

The docking port in consideration for our scope is the MIT Universal Docking Port which is suitable for CubeSats [20], and considered also in [18]. Along with other typical docking requirements provided by [1], it has been possible to derive a list of requirements for our scenario, as shown in the following table:

Table 5.5: Requirements of the relative states of the chaser at docking.

Docking requirements	
Lateral misalignment	$< 1\,\mathrm{cm}$
Longitudinal velocity	$1 \div 3\,\mathrm{cm/s}$
Lateral velocity	$< 1\,\mathrm{cm/s}$
Angular misalignment	< 2 deg
Angular rate	0.1 deg$/s$

The nominal docking velocity is set to 2 cm/s. Besides, the coordinates of the docking port are assumed to be known and placed at the origin of the LVLH frame, since the target is a cooperative station.

5.4 V-bar approach strategy

The V-bar approach has been used for docking of the ATV to the -V-bar port on the Service Module of the ISS [17]. The sequence of maneuvers and operations of the ATV are shown in Figure 5.3, which includes the far-rendezvous operations starting from the aim point S0, located nominally 3000 m below the target's orbit and 30 km behind. After the homing (from S1 to S2) and closing (from S2 to S3) phases, the ATV is positioned at hold point S3, at a distance of $250 \div 500$ m from the target docking port and inside the Approach Ellipsoid (AE). Finally, the final transfer begins, and the ATV follows a straight-line inside an approach cone through the Keep-Out-Sphere (KOS) up to the docking port, with a stop at hold point S4 located 20 m behind the target.

In this thesis it was considered a scenario in which a CubeSat follows the same operations of the ATV beginning from the hold point S3 and ending to the docking. Clearly, some adjustments must be done between the two scenarios, as different kinds of spacecraft imply different requirements.

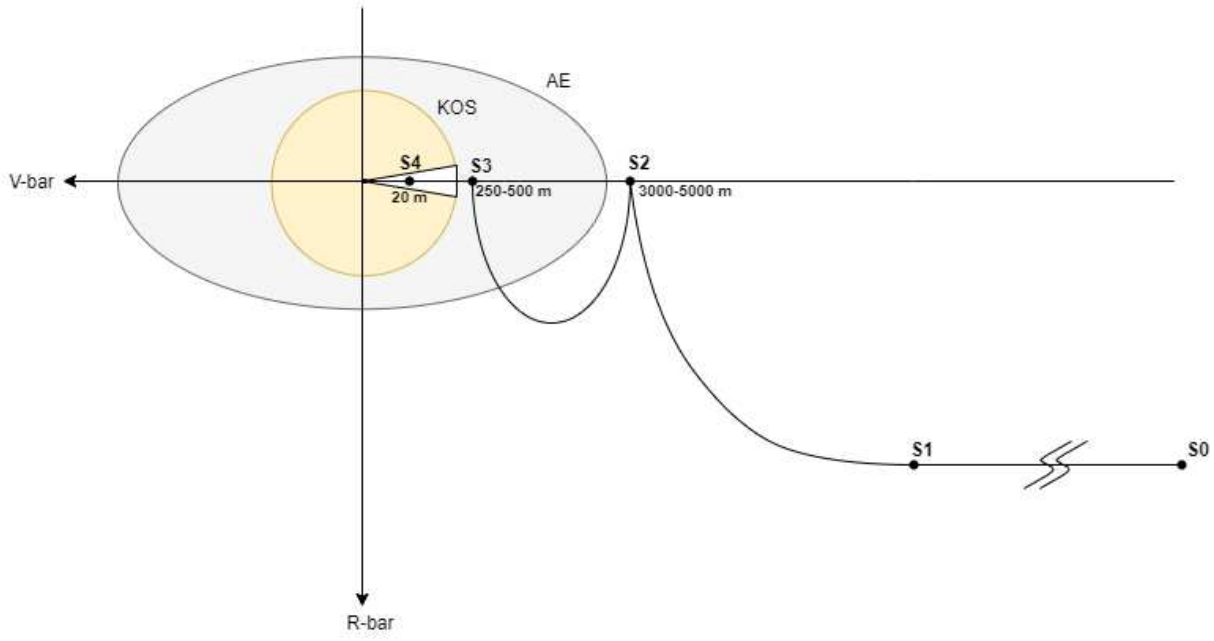


Figure 5.3: V-bar approach of the ATV.

5.5 Velocity reference

In order to achieve its objectives, the controller needs to be provided with a reference to follow. For this type of scenario, two different speed profiles have been adopted for the two phases of the transfer. Regarding the first phase, from initial point S3 to S4, the along-track velocity profile is composed of an accelerating phase, a coasting phase, and a decelerating phase. As for the second phase of the transfer, from S4 to docking port, a similar profile is provided, with an additional final coasting phase at the docking velocity (see Figure 5.4).

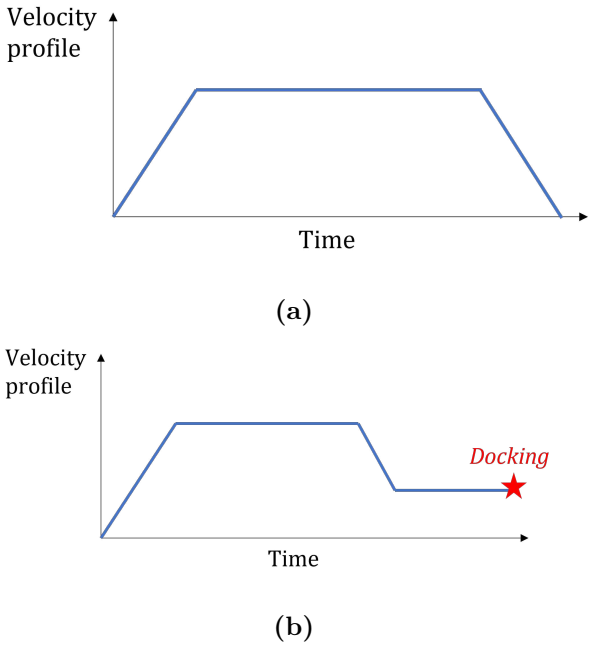


Figure 5.4: (a) Along-track speed reference from S3 to S4; (b) Along-track velocity reference from S4 to docking.

The magnitude of the acceleration and deceleration has been selected so as to be smaller than the maximum acceleration provided by the multi-thruster system. As for the coasting velocity, a value of 10 cm/s has been deemed appropriate for the first phase. No cross-track and out-of-plane velocities are required for this RVD maneuver, therefore their speed profiles are zero for the whole simulation.

Concerning the transition from S4 to the target's docking port, a re-scaled velocity

profile has been generated, requiring a coasting velocity of 5 cm/s. In addition, at the hold point S4, a waiting time of 500 s has been set.

5.6 Constraints

The advantage of MPC is its capability of handling constraints. However, too many constraints can lead to infeasibility in solving the optimization problem 3.3. For example, if the controlled variables have to be maintained within certain values and some disturbances move them outside the constraints, a large control action may be required, but since it is generally restricted by physical limits, the output variables would consequently violate the constraints, leading to infeasibility of the QP problem.

This issue can be overcome by the distinction between hard and soft constraints. Hard constraints represent physical limitations, such as thrust magnitude, which cannot be violated in the optimization problem; conversely, soft constraints allow to be violated for a short period because they are not representative of physical limitations of the system, or because their violation does not involve risks for the safety or functionality of the maneuver. To include constraint softening, the optimization problem needs to be reformulated with the use of a slack variable ε and its weight ρ in the constraints and the cost function [21].

Constraints on the control variables

Since the controller demands control actions expressed in the LVLH reference frame regardless of the spacecraft orientation, the needed acceleration may exceed the maximum value that a single thruster can actually generate. Specifically, the constraint on the maximum thrust acceleration is formulated as:

$$u_{x,k}^2 + u_{y,k}^2 + u_{z,k}^2 \leq u_{max}^2$$

where u_{max} is the maximum acceleration that each thruster can provide. As this expression is nonlinear and the standard MPC requires linear constraint inequalities, a conservative

approach can be followed [22] and the constraints on the control actions become:

$$\begin{aligned} \frac{u_{max}}{\sqrt{3}} \leq u_{x,k} \leq \frac{u_{max}}{\sqrt{3}} \\ \frac{u_{max}}{\sqrt{3}} \leq u_{y,k} \leq \frac{u_{max}}{\sqrt{3}} \\ \frac{u_{max}}{\sqrt{3}} \leq u_{z,k} \leq \frac{u_{max}}{\sqrt{3}} \end{aligned} \tag{5.2}$$

which is suitable for the MPC formulation.

Position constraints

Position constraints are necessary to keep the spacecraft inside a space region during the maneuvers. Similarly to the ATV final approach, the chaser needs to remain inside two truncated cones with different semi-amplitude angles for the two phases. However, both region would be described by nonlinear inequalities, which are not suitable for the MPC formulation. In order to impose linear inequalities, a truncated pyramid is considered instead of the cone. The semi-amplitude angle of the pyramid is set to 10° (less than that for the ATV), for the first phase; as for the second phase, the pyramid is obtained by shrinking linearly the maximum lateral misalignment allowed from point S4 to the docking point. This is done considering a tolerance of ±10 cm at hold point S4, and of ±1 cm at the docking port (Figure 5.5).

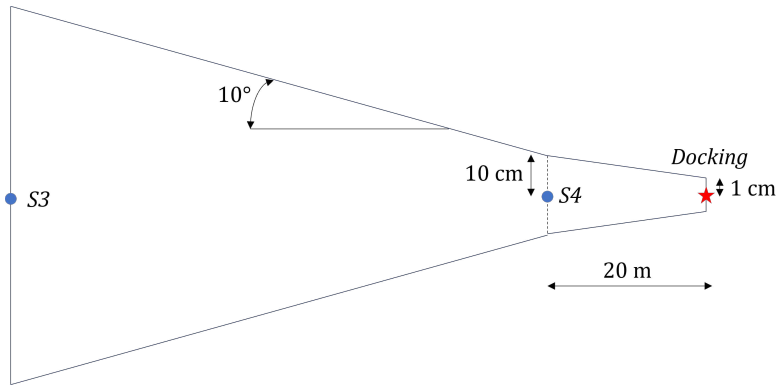


Figure 5.5: Position constraints (not in scale).

Velocity constraints

Velocity is constrained for safety reasons. Lateral velocity (i.e. $\dot{y}$ and $\dot{z}$) is bounded between $\pm 2\,\mathrm{cm/s}$ during the whole operation. As for the longitudinal direction the velocity is required to be at most $10\,\mathrm{cm/s}$ during the first phase, and $5\,\mathrm{cm/s}$ during the second phase.

5.7 Disturbances

In this section we present the environmental perturbations that have been considered in the plant of the system. In our scenario, the main disturbances considered are differential drag and relative J2 effects, since the spacecraft is in LEO. Other effects due to solar radiation pressure, third-body actions, magnetic torques or gravity gradient torques have been considered negligible.

5.7.1 Differential drag

Aerodynamics effects are particularly important in LEO orbits. Reid and Misra proposed in [23] a linear model of differential drag for orbits with small eccentricity, expressed in the Hill's reference. After a change of coordinates, the drag perturbation model in the LVLH frame assumes the form:

$$\mathbf{f}_{drag} = -\frac{1}{2}\left(\frac{1}{\beta_C} - \frac{1}{\beta_T}\right) r n^2 \tilde{\sigma}_2 \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} - \frac{1}{2}\frac{1}{\beta_C} n^2 \tilde{\sigma}_2 \begin{bmatrix} 0 & 0 & -2 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} - \frac{1}{2}\frac{1}{\beta_C} n \tilde{\sigma}_1 \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} \tag{5.3}$$

where r is the orbital radius of the target, n is the target orbit angular velocity, and $\tilde{\sigma}_1$, $\tilde{\sigma}_2$ are dimensionless quantities defined as:

$$\begin{aligned} \tilde{\sigma}_1 &= 1 - \frac{\omega_E}{n}\cos i \\ \tilde{\sigma}_2 &= 1 - 2\frac{\omega_E}{n}\cos i \\ \omega_E &= 86\,164\,\mathrm{s} \end{aligned} \tag{5.4}$$

β_C and β_T represent the dimensionless ballistic coefficients of the chaser and target spacecraft respectively, given by:

$$\beta = \left(\rho\frac{C_D A}{m} r\right)^{-1} \tag{5.5}$$

where the atmospheric density ρ is assumed to be constant over the whole orbit. In general, the atmospheric density profile for circular orbits is not constant because of several factors such as solar and geomagnetic activity levels, but these variations are quite small and are here neglected.

When the ballistic coefficients of the chaser and target are very different, the differential drag could introduce some perturbation effects. At the target's altitude, 450 km, the atmospheric density is $\rho = 1.184 \times 10^{-12}\,\mathrm{kg/m^3}$, according to the US Standard Atmosphere [24], and considering the target's characteristics (see Section 5.3) the dimensionless ballistic coefficient of the target is:

$$\beta_T = 1.841 \times 10^8$$

whereas for a CubeSat 3U, assuming the frontal area $A_c = 1 \times 10^{-2}\,\mathrm{m^2}$, $C_D = 2.2$, and $m = 4\,\mathrm{kg}$, Equation (5.5) yields:

$$\beta_C = 2.248 \times 10^7$$

It follows that the ballistic coefficient of the target is only one order of magnitude greater than that of the chaser, therefore not high perturbations will be expected.

5.7.2 J2 perturbation effects

The approximation of the Earth gravity field to the first harmonic introduces higher order errors, which are attributable to the non-spherical and unequal distribution of the Earth mass. The J2 perturbation effect is the most significant term among the approximation

errors that causes modifications in the orbital parameters. Specifically, prograde LEO orbits are subjected to a regression of the right ascension of ascending node (RAAN) and a variation in the argument of perigee.

In [23] a linear-time-variant model for the J2 perturbation in Hill's coordinate frame is described. Multiplying the gradient of the J2 potential field by the relative position vector, it is possible to calculate the linear relative J2 disturbances:

$$\mathbf{f}_{J_2} = \nabla \mathbf{J_2}(\mathbf{r})\mathbf{x} \tag{5.6}$$

where the gradient of the J2 potential field expressed in Hill's coordinate frame is given by:

$$\nabla \mathbf{J_2}(r,\theta,i) = \frac{6\mu J_2 R_E^2}{r^5} \begin{bmatrix} \left(1-3\sin^2 i \sin^2\theta\right) & \sin^2 i \sin 2\theta & \sin 2i \sin\theta \\ \sin^2 i \sin 2\theta & -\frac{1}{4} - \sin^2 i \left(\frac{1}{2} - \frac{7}{4}\sin^2\theta\right) & -\left(\sin 2i \cos\theta\right)/4 \\ \sin 2i \sin\theta & -\left(\sin 2i \cos\theta\right)/4 & -\frac{3}{4} + \sin^2 i \left(\frac{1}{2} + \frac{5}{4}\sin^2\theta\right) \end{bmatrix} \tag{5.7}$$

This expression was added to the plant dynamics of the simulator. However, since it is a LTV model, it cannot be included in the MPC formulation, and a proper LTI model should be defined. The approach followed is to take the average value of the J_2 gradient matrix over one orbit, and consider it in the prediction model of the controller. For circular orbits the result is as follows:

$$\frac{1}{2\pi}\int_0^{2\pi} \nabla \mathbf{J_2} d\theta = n^2 \begin{bmatrix} 4s & 0 & 0 \\ 0 & -s & 0 \\ 0 & 0 & -3s \end{bmatrix} \tag{5.8}$$

where s is a parameter defined as:

$$s = \frac{3}{8} J_2 \left(\frac{R_E}{r}\right)^2 (1 + 3\cos 2i) \tag{5.9}$$

Inserting Equation (5.8) into (5.6) we obtain the LTI model to be added to the prediction model of the MPC. Observe that the order of magnitude of the J2 relative acceleration is between 10^{-7} and $10^{-9}\,\mathrm{m/s^2}$, therefore the perturbation effects are very small.

Chapter 6

Controllers tuning and preliminary simulations results

6.1 Software simulator

Numerical simulations of autonomous rendezvous and docking maneuvers have been run on a software simulator developed in Matlab. A former version of the software, capable of simulating maneuvers along two translational axes and one rotational axis, has been upgraded to deal with six-degree-of-freedom maneuvers. The software, which is based on object-oriented programming, was conceived to test and simulate different types of controllers in various scenarios with the aim of validating the optimal strategies.

The analysis of the behaviour of the physical system under examination supplies important information that can be used as the basis for making design decisions such as to adjust the controller parameters in order to achieve better performances.

6.2 State space formulation

The translational state of the system contains position and velocity of the chaser in the orbital local frame. These variables are grouped to form the state vector $\mathbf{x}$ as following:

$$\mathbf{x} = [x,\, y,\, z,\, \dot{x},\, \dot{y},\, \dot{z}]^T \tag{6.1}$$

The control actions are the translational accelerations in the three directions and are grouped in the vector $\mathbf{u}$:

$$\mathbf{u} = [u_x,\, u_y,\, u_z]^T \tag{6.2}$$

Since the target's orbit is circular, or near-circular, and the range between the chaser and the target is relatively small compared to the target's orbital radius, the Clohessy-Wiltshire equations can be used to define the prediction model of the MCP controller. Note that the reference system used for the simulation is the LVLH system, whereas the system used for the CW equations in (2.9) is the Hill's frame. A change in coordinates needs to be done through a rotation matrix given by:

$$\mathbf{R}_{HO} = \begin{bmatrix} 0 & 0 & -1 \\ 1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix} \tag{6.3}$$

describing a generic transformation from $\mathcal{F}_{\mathcal{O}}$ to $\mathcal{F}_{\mathcal{H}}$. In order to transform the transition matrix (2.9) in $\mathcal{F}_{\mathcal{O}}$ frame, the following two rotations are necessary:

$$A_O = \mathbf{R}_{HO}^T A_H \mathbf{R}_{HO}$$

It is possible to write the state space formulation of the translational motion as:

$$\begin{aligned} \dot{\mathrm{x}}(t) &= A_c \mathrm{x}(t) + B_c \mathrm{u}(t) \\ \mathrm{y}(t) &= C_c \mathrm{x}(t) \end{aligned} \tag{6.4}$$

where:

$$A_c = \begin{bmatrix} & \mathbf{0}_{3\times3} & & & \mathbf{I}_{3\times3} & \\ 0 & 0 & 0 & 0 & 0 & 2n \\ 0 & -n^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3n^2 & -2n^2 & 0 & 0 \end{bmatrix}, \quad B_c = \begin{bmatrix} \mathbf{0}_{3\times3} \\ \mathbf{I}_{3\times3} \end{bmatrix} \tag{6.5}$$

and $C_c = \mathbf{I}_{6\times6}$, so that the measured variables coincide with the state vector. The subscript "c" indicates that the matrices of the model describe the evolution of the system in continuous time. To apply the MPC it is necessary to discretize these matrices considering a sampling time T_s, and this is done by using the function `c2d` of Matlab.

6.3 Validation of MPC

This section presents the tuning and some preliminary results of the translational control with MPC. The goal is also to show the characteristics of this modern control tool in terms of efficiency and reliability when applied to a RVD scenario. The scenario considers the final approach of the CubeSat from the hold point S3 to the docking port, as described in Section 5.2, along V-bar, with the initial conditions listed in Table 6.1.

Table 6.1: Initial conditions.

Initial conditions	**Value**
x	$-250\,\mathrm{m}$
y	$-10\,\mathrm{m}$
z	$20\,\mathrm{m}$
$\dot{x}$	$0\,\mathrm{m/s}$
$\dot{y}$	$0\,\mathrm{m/s}$
$\dot{z}$	$0\,\mathrm{m/s}$

The attitude dynamics are decoupled from the translational dynamics in our case, and its control is not object of discussion in this section.

Disturbances and measurement noises have been excluded for these preliminary simulations, but they are taken into account in the next simulations. Furthermore, the predictive model of the MPC has been set identical to the dynamics equations of the plant.

As for the thursters model, no uncertainties have been included, and the action mode

is set to Piece-Wise-Constant, as it is supposed to be in the MPC formulation.

The MPC parameters are given in the Table 6.2.

Table 6.2: MPC parameters.

MPC PARAMETERS	
Sample Time	1 s
Prediction Horizon	60
Control Horizon	15

Tuning weights for the optimization problem have been chosen with the aim of penalizing fuel consumption more than reference tracking. The corresponding weighing matrices are:

$$\begin{aligned} \mathbf{Q} &= \text{diag}\,(\,10,\,10,\,10,\,1,\,1,\,1\,) \\ \mathbf{R_u} &= \text{diag}\left(\,10^3,\,10^3,\,10^3\right) \\ \mathbf{R_{\Delta u}} &= \text{diag}\left(\,10^{-10},\,10^{-10},\,10^{-10}\right) \end{aligned} \tag{6.6}$$

where the first three entries of $\mathbf{Q}$ refer to the position errors and the last three entries refer to the velocity errors. As for the other two matrices, their entries correspond to each of the control variables. The choice of a very small weight on the rate of change of the control actions means that the controller may allow for large variations of thrust, if necessary.

Figures 6.1 and 6.2 display the trajectory of the chaser from the initial point up to docking.

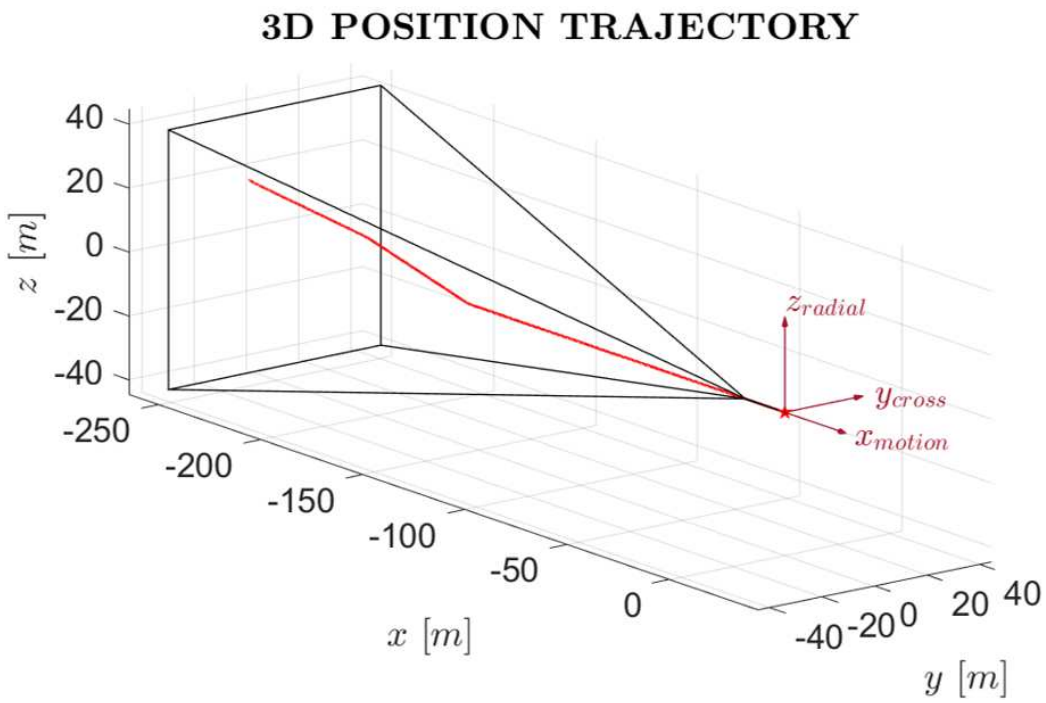


Figure 6.1: Trajectory of the chaser with MPC in red; approach truncated pyramid in black; docking point starred.

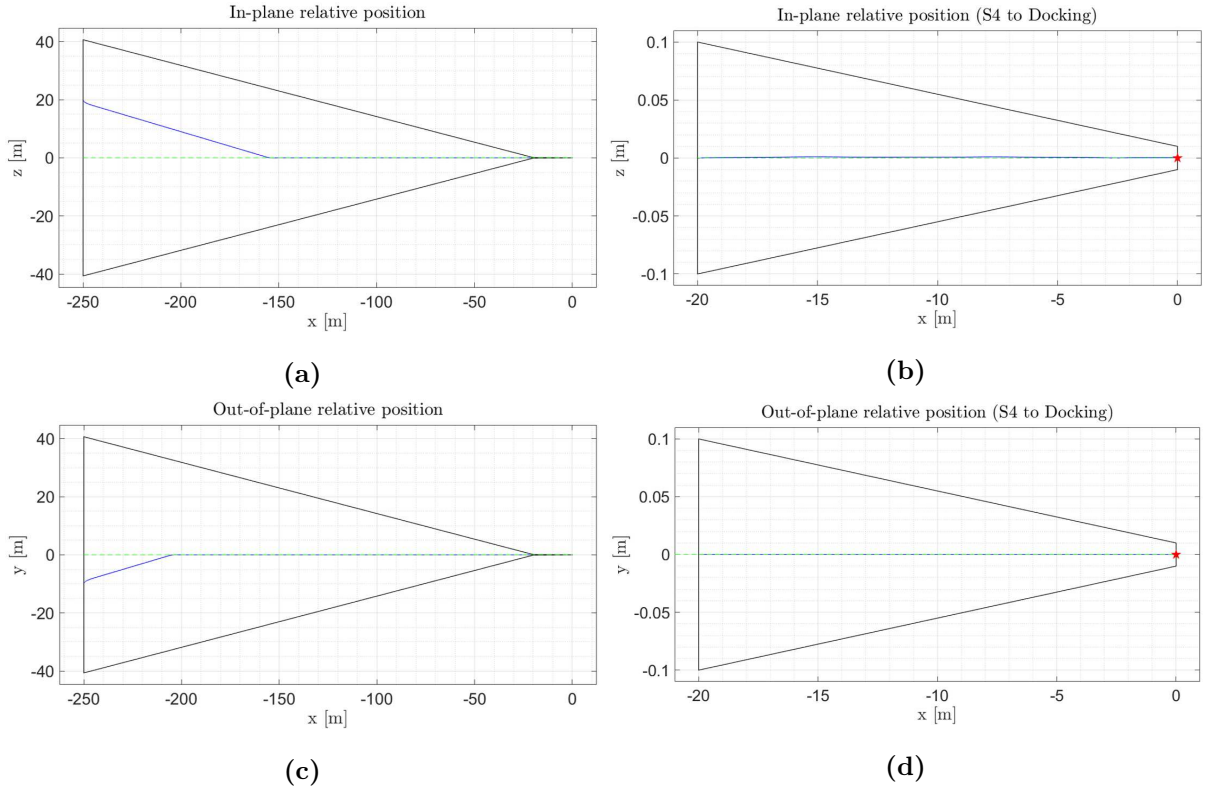


Figure 6.2: Actual trajectory in solid blue line and reference trajectory in green dashed line. (a) In-plane relative position; (b) In-plane relative position from hold point S4 to docking; (c) Out-of-plane relative position; (d) Out-of-plane relative position from hold point S4 to docking.

The initial lateral errors along Z and Y are cancelled out as the spacecraft smoothly approaches the axis of the approach corridor, until it reaches the hold point S4. After five hundred seconds of waiting time, the spacecraft enters the second approach corridor and reaches the docking port well within the allowed ranges. The state of the system in these two points is summarized in Tables 6.3 and 6.4.

Table 6.3: Chaser's state at hold point S4 (MPC).

S4 hold point			
	x	y	z
Position $[m]$	-20.005	-0.000	-0.001
Velocity $[m/s]$	0.002	-0.000	0.000

Table 6.4: Chaser's state at the docking point (MPC).

Docking point			
	x	y	z
Position $[m]$	-0.002	-0.000	0.000
Velocity $[m/s]$	0.020	0.000	0.000

With regard to the relative velocity, the profile and reference of the chaser's velocity are shown in Figure 6.3.

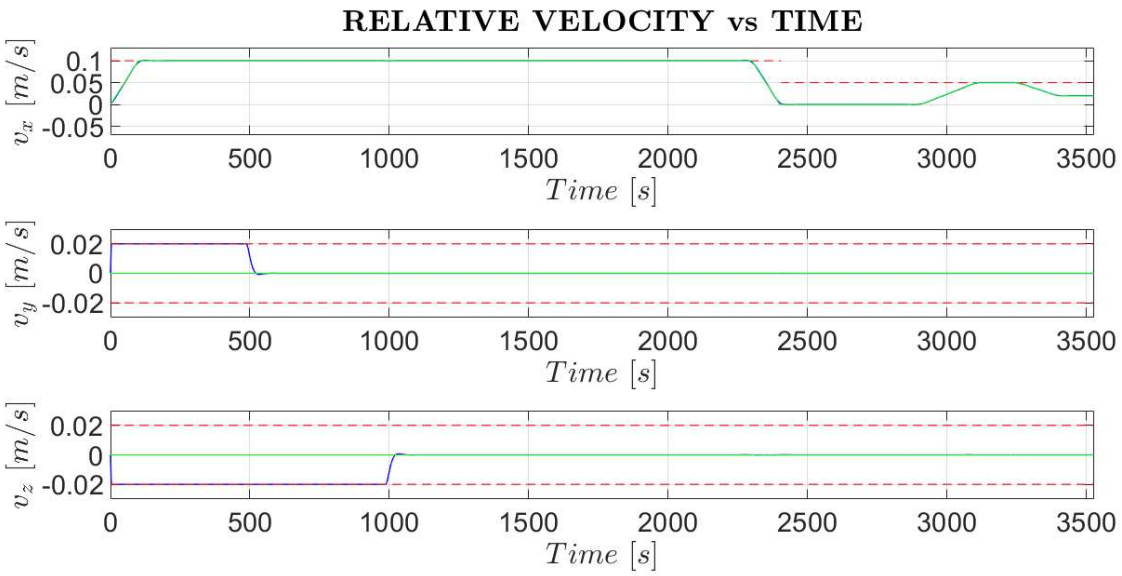


Figure 6.3: Relative velocity of the chaser with MPC in blue; reference speed in green; speed constraints dashed in red.

As we can see, the along-track velocity, in blue, barely shows any difference from the reference profile, in green, over the whole simulation. As for the out-of-plane and radial directions, the spacecraft needs to accelerate along both directions in order to close the gap originated by the initial conditions. It is important to note that these accelerations stop when the limit of lateral velocity is reached, which is possible thanks to the velocity constraints of the MPC. In this way, a smooth approach through the corridor is obtained, along with saving of fuel.

Concerning fuel consumption, results are expressed in terms of delta-V calculated by integration of the accelerations in the body frame provided by the thrusters over the whole simulation. The control actions are shown in Figure 6.4. Note that, as the attitude is nominal for the whole time, the LVLH frame and body frame coincide, and the magnitude of thrust is quite low compared to the maximum limit, which is a consequence of a high weight on the control actions weighing matrix.

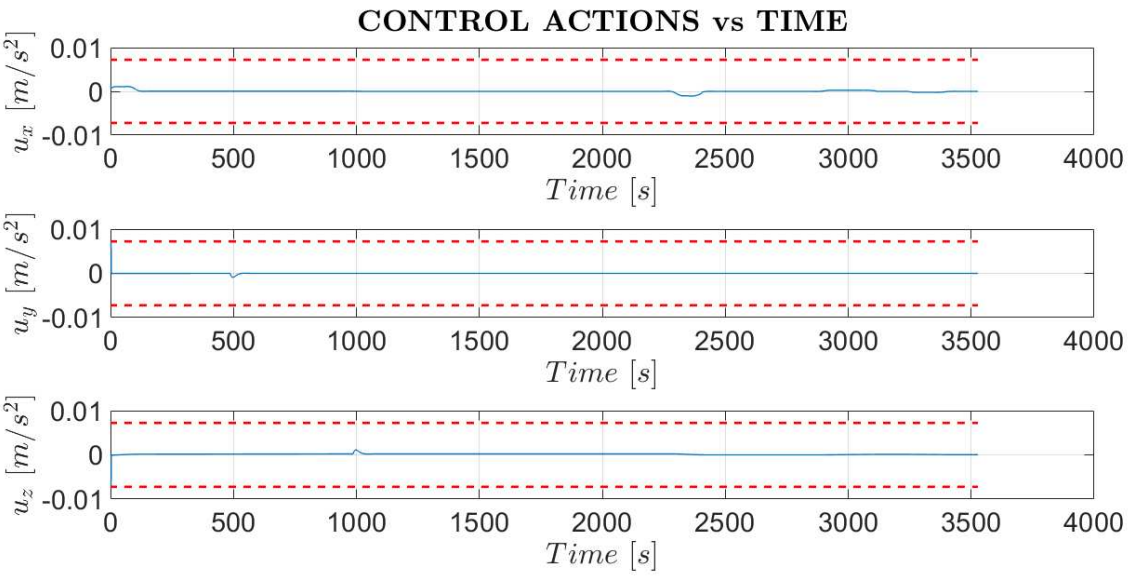


Figure 6.4: Commanded accelerations in blue (MPC). Constraints dashed in red.

Table 6.5: Use of delta-V in the entire maneuver with MPC (in body axes).

Delta V [m/s]	
ΔV_x	0.289533
ΔV_y	0.000000
ΔV_z	0.559854
ΔV_{total}	0.849386

Summarizing, the MPC translational controller has proven to guarantee a good performance for this ideal scenario.

6.4 Translational control with PID

This section presents the numerical results of a PID controller used for the translational control of the spacecraft during the RDV maneuver from point S3 to the docking port. As for the validation of the MPC controller, the attitude motion has been excluded in the

simulation. The initial conditions are shown in Table 6.6, while the PID controller gains are listed in Table 6.7.

Table 6.6: Initial conditions.

Initial conditions	**Value**
x	$-250\,\mathrm{m}$
y	$-10\,\mathrm{m}$
z	$20\,\mathrm{m}$
$\dot{x}$	$0\,\mathrm{m/s}$
$\dot{y}$	$0\,\mathrm{m/s}$
$\dot{z}$	$0\,\mathrm{m/s}$

Table 6.7: PID gains for the translational motion.

Translational PID Gains			
	x	y	z
K_p	3×10^{-2}	3×10^{-2}	3×10^{-2}
K_i	2×10^{-4}	2×10^{-4}	2×10^{-4}
K_d	5	5	5
K_t	1×10^{-4}	1×10^{-4}	1×10^{-4}
β	1	1	1
γ	1	1	1

The choice of the gains has been made by trial-and-error verifying the stability of the closed-loop system as described before. The discretization time is $T_s = 1\,\mathrm{s}$, as in the MPC control. A small value of K_t does not introduce much non-linearity to the system, and as long as the integral term does not require anti-reset windup action, the overall system is once again linear. Therefore, the check on the stability by observation of the closed-loop poles location is valid. Figure 6.5 displays the location of these poles.

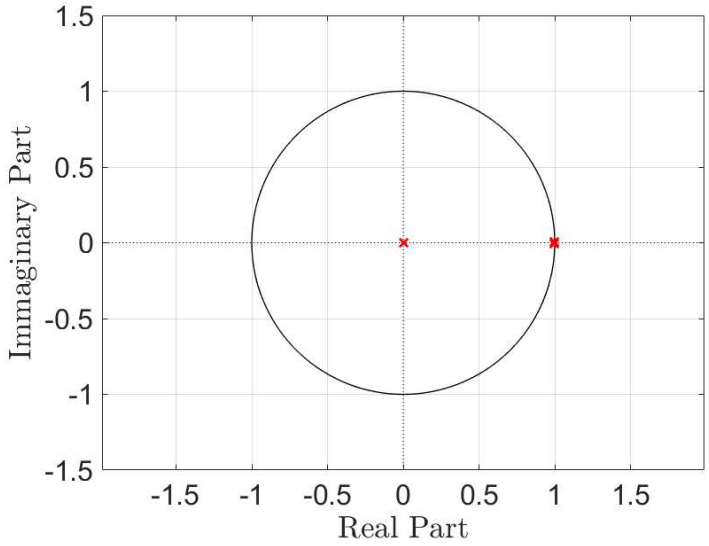


Figure 6.5: Poles location. All points are inside the unit circle, thus ensuring stability.

Figures 6.6 and 6.7 illustrate the trajectory of the chaser from the initial point to the docking point. Hold point S4 and docking point are reached with the states given in Tables 6.8 and 6.9.

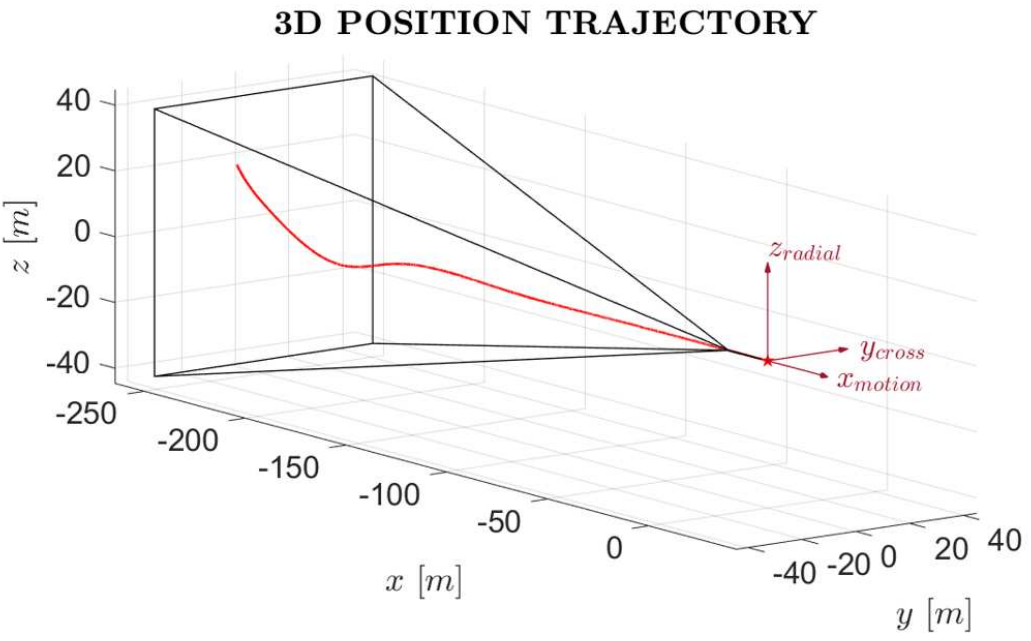


Figure 6.6: Trajectory of the chaser with PID in red; approach truncated pyramid in black; docking point starred.

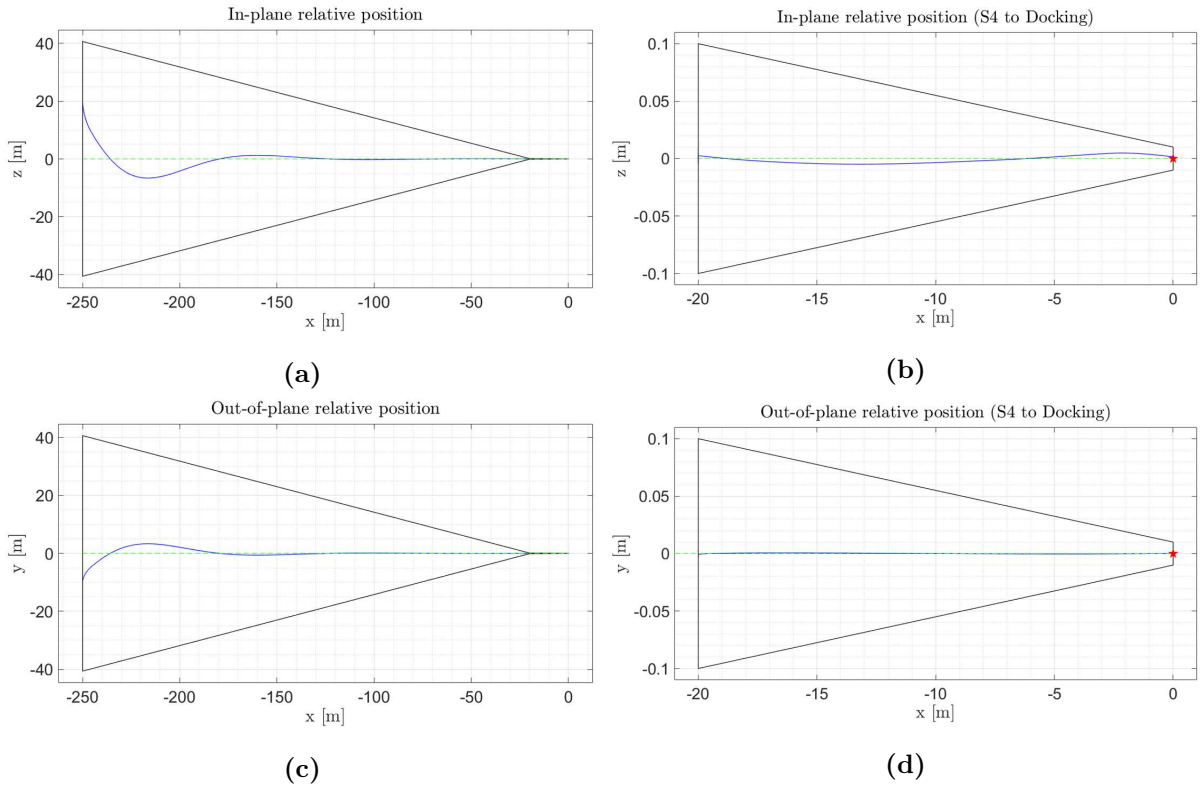


Figure 6.7: Actual trajectory in solid blue line and reference trajectory in green dashed line. (a) In-plane relative position; (b) In-plane relative position from hold point S4 to docking; (c) Out-of-plane relative position; (d) Out-of-plane relative position from hold point S4 to docking.

Table 6.8: Chaser's state at hold point S4 (PID).

S4 hold point

	x	y	z
Position $[m]$	-20.003	-0.003	0.002
Velocity $[m/s]$	0.011	0.000	0.000

Table 6.9: Chaser's state at the docking point (PID).

Docking point

	x	y	z
Position $[m]$	-0.005	0.000	0.002
Velocity $[m/s]$	0.020	0.000	-0.000

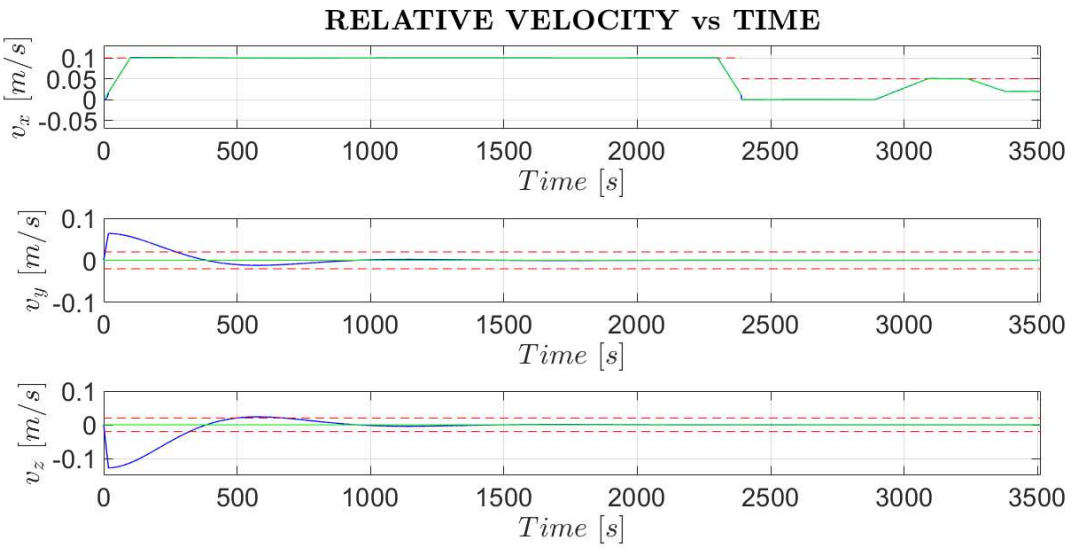


Figure 6.8: Relative velocity of the chaser with PID in blue; reference speed in green; speed constraints dashed in red.

Figure 6.8 shows the velocity profile of the chaser during the maneuver. It can be observed that the along-track velocity profile is well followed by the chaser, while the velocities along y and z axes violate the limits imposed by the mission (0.02 m/s) in the first phase. This inefficiency highlights one of the limitations of a classic PID controller with respect to a modern MPC. Regarding the usage of thrusters, the delta-V required is shown in Table 6.10. To summarize, the translational PID controller tested turns out to perform efficiently, despite not being capable of handling all constrained required.

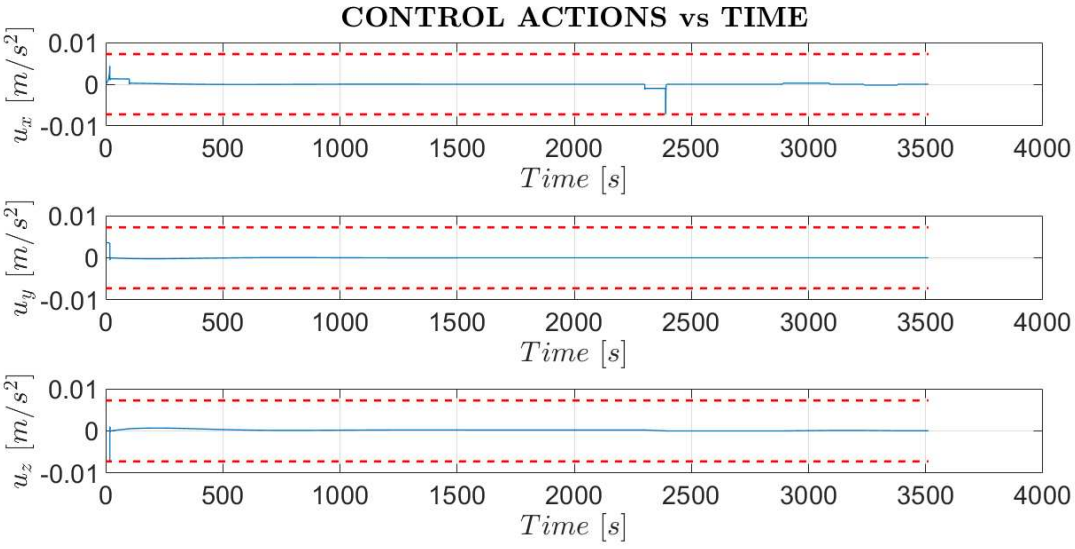


Figure 6.9: Commanded accelerations in blue (PID). Constraints dashed in red.

Table 6.10: Use of delta-V in the entire maneuver with PID (in body axes).

Delta V [m/s]	
ΔV_x	0.363530
ΔV_y	0.157266
ΔV_z	0.820215
ΔV_{total}	1.341011

6.5 MPC with disturbances

In Section 6.3 we showed the results of the translational MPC controller in absence of environment disturbances and sensor noise. Now, we present numerical results of a simulation which accounts for such factors. The same initial conditions of section 6.3 are used in order to give a comparison between the ideal case and the realistic one.

Some modeling errors have been considered in the MPC formulation as well. Regarding the atmospheric model, the knowledge of the atmospheric density is not usually known with high precision, therefore an uncertainty of at most $\pm 10\%$ of the actual density (which is the same as in Section 5.7.1) has been added to the value used by the MPC predictive model. Furthermore, an error in the orbit angular rate n of at most $\pm 1\%$ has been considered.

With regard to the actuators, a Gaussian percentage error of 10% (3σ) of the maximum authority has been applied, as anticipated.

In the first test, only one set of parameters for the MPC was used for the whole RDV maneuver. However, this configuration raised a problem in terms of performance, as high weights on the controlled variables were necessary in the most critical parts of the transfer, such as when reaching the point S4 and the docking port, but not in the initial phase, when the spacecraft is far enough to be controlled with less effort. The result was an intense use of actuators at the beginning, with consequent violation of the lateral velocity constraint. For this reason, it has been chosen to use different parameters of the MPC at various stages along the transfer, which are shown in Table 6.11 for completeness.

Table 6.11: Weights of the MPC.

MPC weights

Range $[m]$	**Q**						**R**			$\mathbf{R_{\Delta u}}$		
$-500 \div -100$	1.5	7.5e-1	7.5e-1	1.5	7.5e-1	7.5e-1	6e+2	6e+2	6e+2	1e+4	1e+4	1e+4
$-100 \div -50$	1e+1	1e+1	1e+1	1.5	1.5	1.5	1.8e+3	1.8e+3	1.8e+3	1e+5	1e+5	1e+5
$-50 \div -30$	1e+2	1e+2	1e+2	1.5	1.5	1.5	1.8e+3	1.8e+3	1.8e+3	1e+5	1e+5	1e+5
$-30 \div -25$	5e+2	2.5e+2	2.5e+2	1.5	7.5	7.5	1.4e+3	1.4e+3	1.4e+3	1e+4	1e+4	1e+4
$-25 \div -20$	3e+3	3e+4	3e+4	9	9e+1	9e+1	1.4e+3	1.4e+3	1.4e+3	1e+4	1e+4	1e+4
$-20 \div -15$	1e+3	1e+3	1e+3	2e+1	2e+1	2e+1	3.6e+2	3.6e+2	3.6e+2	1e+3	1e+3	1e+3
$-15 \div -10$	4.5e+4	4.5e+5	4.5e+5	2.5e+3	2.5e+4	2.5e+4	1e+3	1e+3	1e+3	1e+4	1e+4	1e+4
$-10 \div -5$	5.4e+5	5.4e+6	5.4e+6	3e+4	3e+5	3e+5	5e+2	5e+2	5e+2	1e+3	1e+3	1e+3
$-5 \div -3$	7.2e+5	7.2e+6	7.2e+6	4e+4	4e+5	4e+5	1.5e+2	1.5e+2	1.5e+2	1e+3	1e+3	1e+3
$-3 \div -1.5$	9e+5	9e+6	9e+6	5e+4	5e+5	5e+5	1e+2	1e+2	1e+2	1e+3	1e+3	1e+3
$-1.5 \div -0.5$	1.08e+6	1.08e+7	1.08e+7	6e+4	6e+5	6e+5	8e+1	8e+1	8e+1	1e+3	1e+3	1e+3
$-0.5 \div 0$	1.35e+6	1.35e+7	1.35e+7	7.5e+4	7.5e+5	7.5e+5	6e+1	6e+1	6e+1	1e+3	1e+3	1e+3

The following Figures 6.10 and 6.11 show the trajectory of the spacecraft.

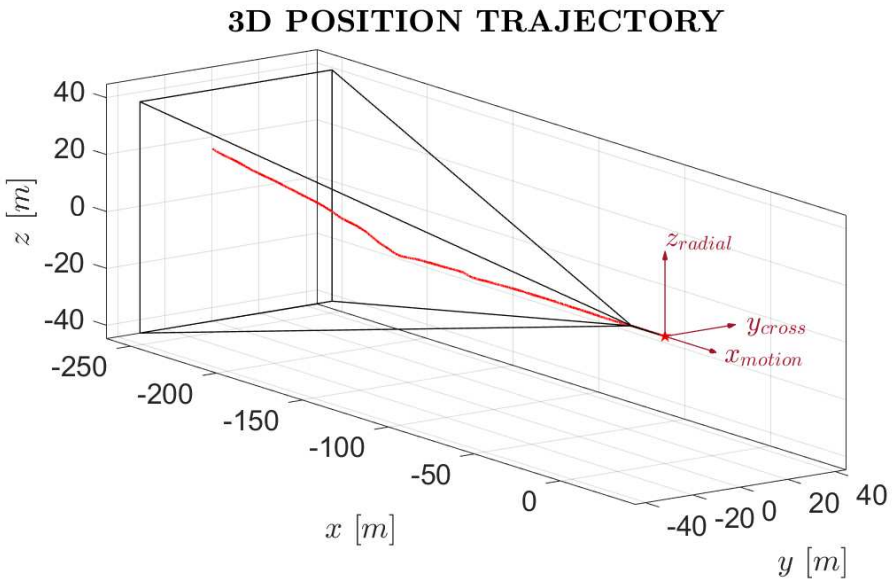


Figure 6.10: Trajectory of the chaser with MPC in red; approach truncated pyramid in black; docking point starred.

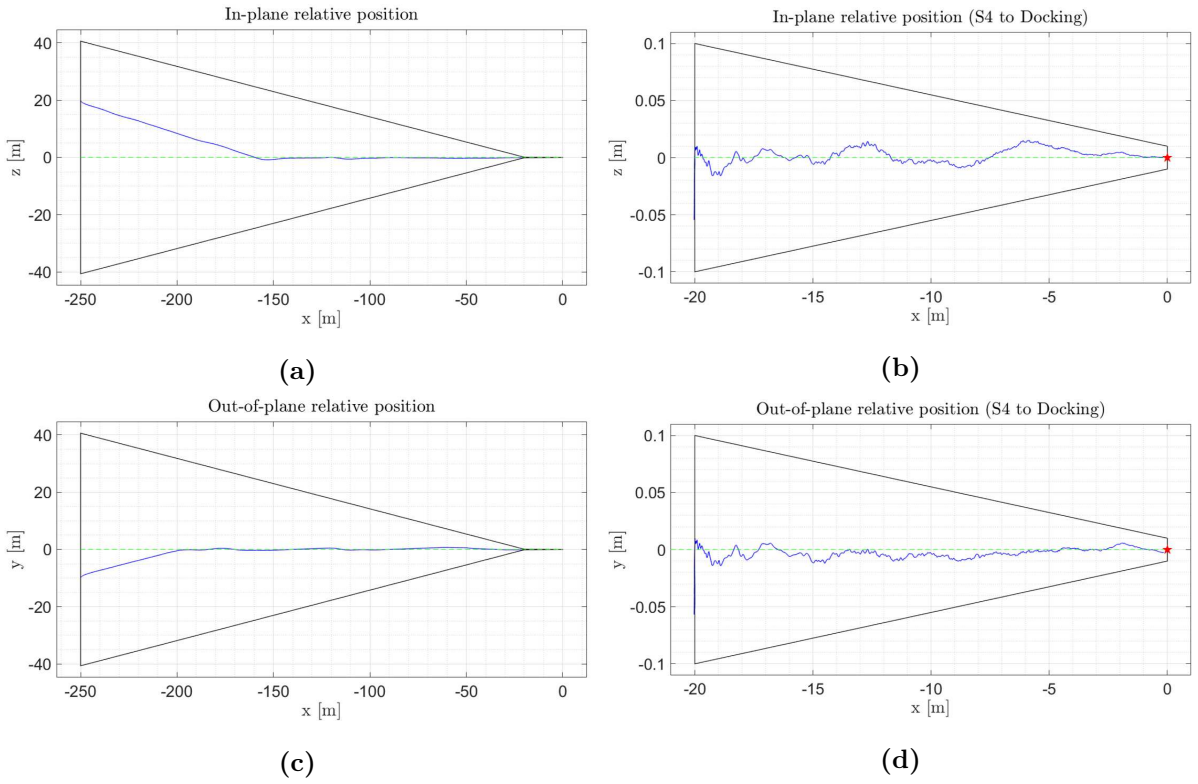


Figure 6.11: Actual trajectory in solid blue line and reference trajectory in green dashed line. (a) In-plane relative position; (b) In-plane relative position from hold point S4 to docking; (c) Out-of-plane relative position; (d) Out-of-plane relative position from hold point S4 to docking.

In order not to exceed the transversal boundaries especially in the critical phases such as at docking, it has been chosen to set an approach cone inside the MPC formulation slightly tighter than that specified by the mission. This is a conservative approach that helps improve the meeting of the requirements.

Besides, since the docking mechanism of the spacecraft is located 15 cm in front of the center of mass, the docking occurs when the chaser is in position $[-0.15,\ 0,\ 0]$.

Table 6.12: Chaser's state at hold point S4 (MPC).

S4 hold point			
	x	y	z
Position $[m]$	-20.008	-0.054	-0.002
Velocity $[m/s]$	0.014	0.005	0.005

Table 6.13: Chaser's state at docking (MPC).

Docking point			
	x	y	z
Position $[m]$	-0.150	-0.002	0.001
Velocity $[m/s]$	0.020	-0.000	0.000

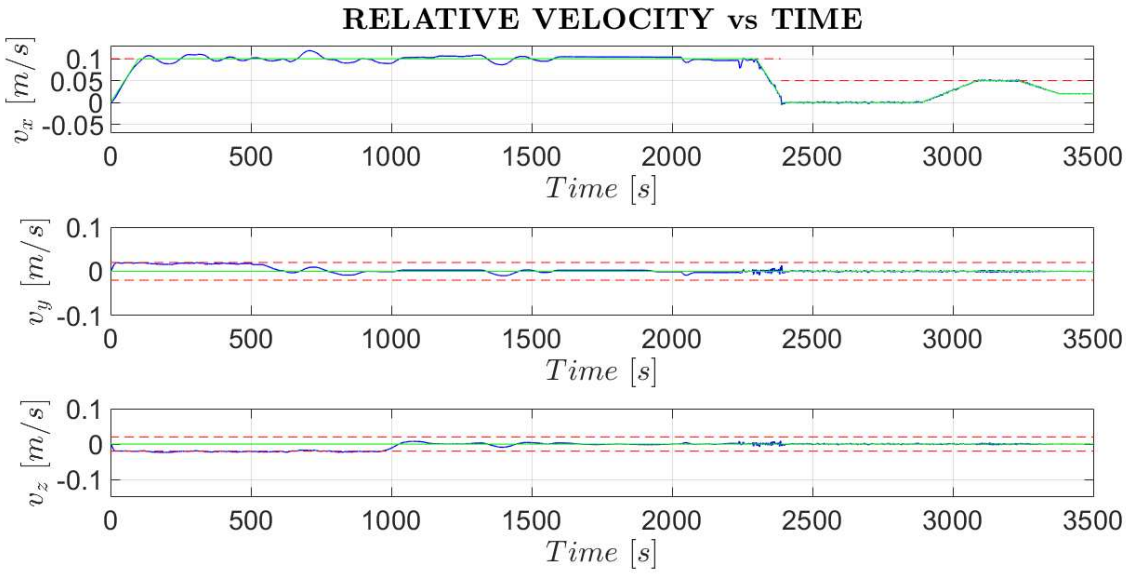


Figure 6.12: Relative velocity of the chaser with MPC and disturbances in blue; reference speed in green; speed constraints dashed in red.

Comparing the results given in Tables 6.12 and 6.13, and Figure 6.12, with those of Section 6.3 it is possible to observe some small deviations due to the external disturbances and measurement noise. The lateral velocity limits are still respected, despite very small violations which are allowed by the fact that speed constraints have been softened in the MPC formulation.

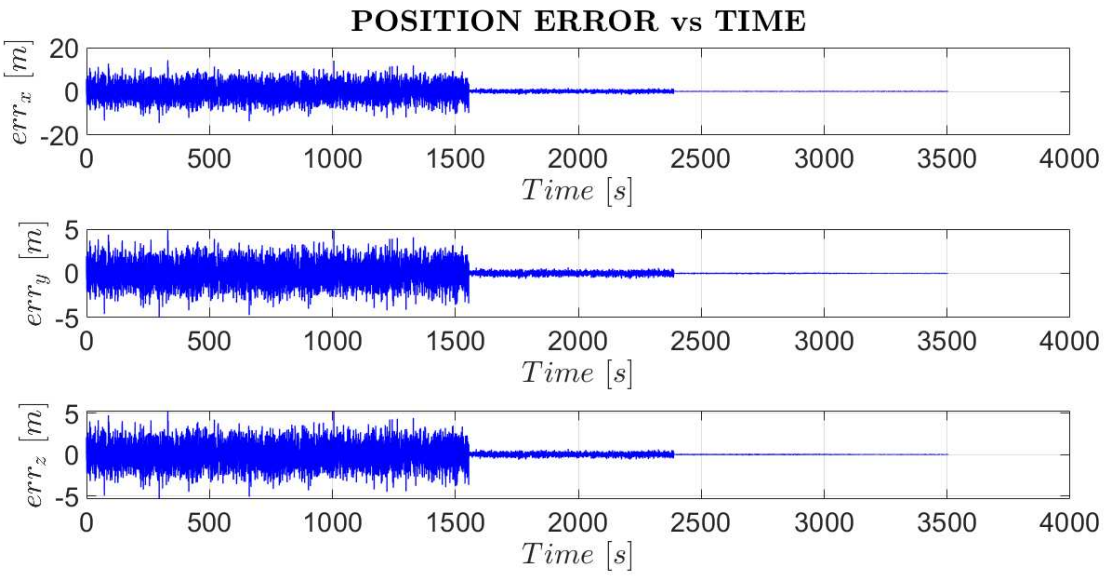


Figure 6.13: Position estimation error (MPC).

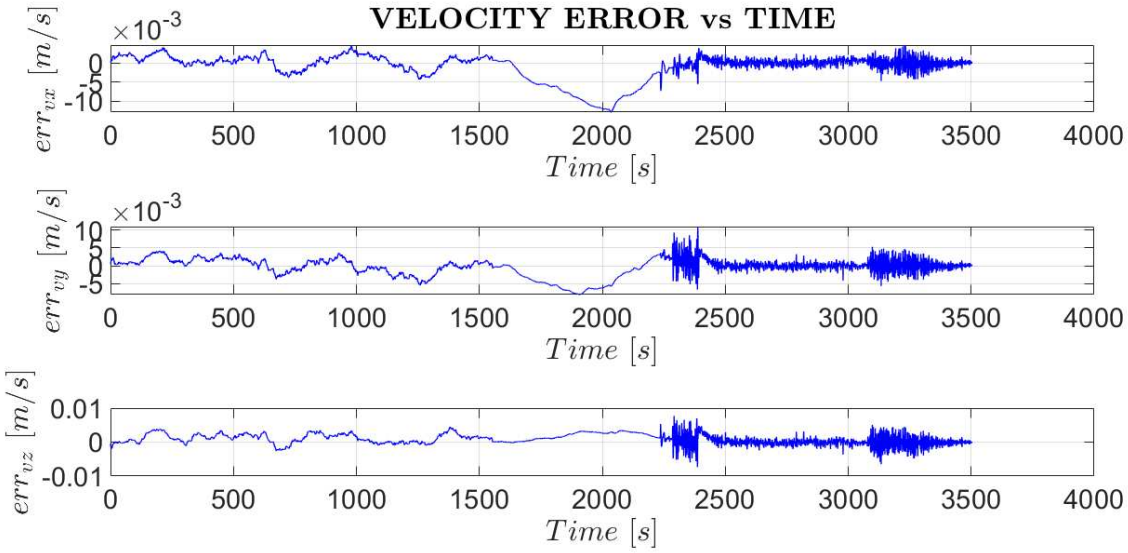


Figure 6.14: Velocity estimation errors (MPC).

Regarding measurements, it is quite visible from Figure 6.13 the increasing position estimation with decreasing range, as explained in Section 5.2.3. As for the velocity estimation, estimates are obtained by the built-in Kalman filter of the MPC Toolbox, which parameters were determined by trial and error verifying that the estimation accuracy was similar to that expected for this spacecraft. Particularly, two different Kalman filters are used during the maneuvers, one from the initial point to S4, and one from S4 to dock-

ing. The reason for this choice was due to the different measurements accuracy that the spacecraft experience in the maneuver, and hence different parameters for the filter.

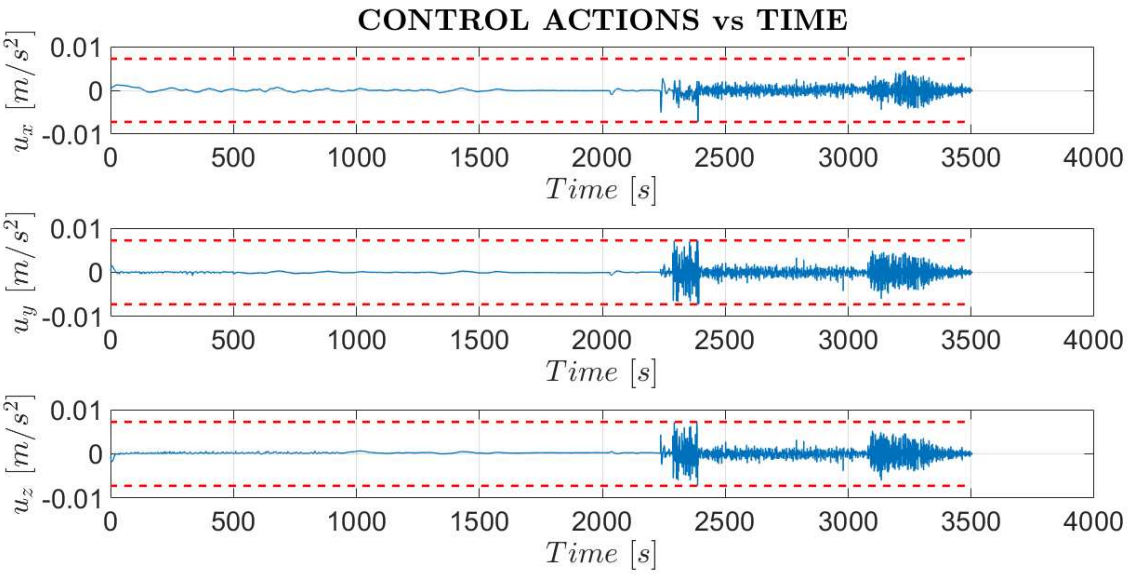


Figure 6.15: Commanded accelerations in blue (MPC). Constraints dashed in red.

The commanded accelerations are quite different from the previous case. Indeed, starting from point S4 a more intense use of thrusters occurs, as the weights on position and velocity errors are more pronounced. As a result, the delta-Vs are larger than the ideal case.

Table 6.14: Use of delta-V in the entire maneuver with MPC and disturbances.

Delta-V [m/s]	
ΔV_x	1.563638
ΔV_y	1.562791
ΔV_z	1.887996
ΔV_{total}	5.014426

This simulation proved that the MPC with the current parameters turns out to be a performing controller for these initial conditions. In the following chapter, a Montecarlo approach is used to test the MPC with many other possible initial conditions.

6.6 PID with disturbances

Similarly to what was done with the MPC test in the presence of disturbances and noise, the same approach has been carried out with the PID. The same initial conditions and disturbances used in the RDV scenario with MPC are applied. Moreover, the derivative term has been configured to avoiding the use of a low-pass filter since the relative velocity is estimated by a Kalman Filter and thus already filtered.

Since the MPC in Section 6.5 could use different parameters along the corridor, the PID has been featured with varying gains as well. In particular, the best results were obtained with the parameters shown in Table 6.15 found by trial and error.

Table 6.15: PID gains for the translational motion.

Translational PID Gains									
Range [m]	K_p			K_I			K_d		
	x	y	z	x	y	z	x	y	z
$-500 \div -100$	0.1	0.1	0.1	1e-4	1e-4	1e-4	3.15	1.46	1.46
$-100 \div -50$	2e-3	7.5e-3	7.5e-3	1e-4	1e-4	1e-4	3.5	1.7	1.7
$-50 \div -20$	3e-3	9e-3	9e-3	2e-4	2e-4	2e-4	5	5	5
$-20 \div 0$	3e-2	3e-2	3e-2	2e-4	2e-4	2e-4	5	5	5

The gain for the anti wind-up correction is set to Kt = 0.1 for all controllers, and so are $\beta = 1$ and $\gamma = 1$.

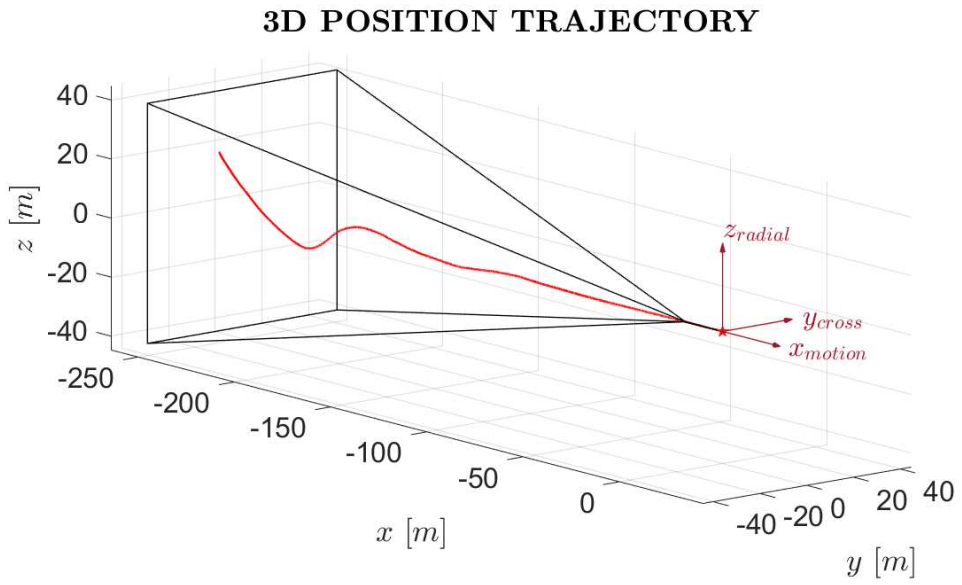


Figure 6.16: Trajectory.

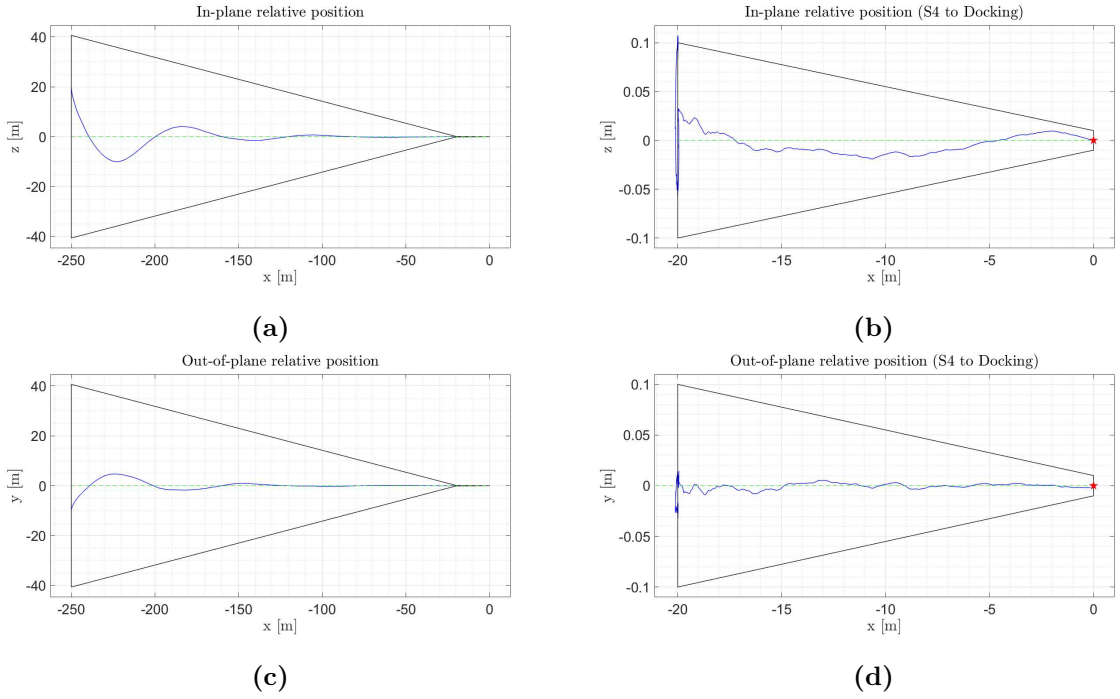


Figure 6.17: Actual trajectory in solid blue line and reference trajectory in green dashed line. (a) In-plane relative position; (b) In-plane relative position from hold point S4 to docking; (c) Out-of-plane relative position; (d) Out-of-plane relative position from hold point S4 to docking.

Table 6.16: Chaser's state at arrival at hold point S4.

S4 hold point			
	x	y	z
Position $[m]$	-20.010	-0.031	0.084
Velocity $[m/s]$	0.017	-0.001	0.000

Table 6.17: Chaser's state at docking.

Docking point			
	x	y	z
Position $[m]$	-0.147	-0.002	0.000
Velocity $[m/s]$	0.020	-0.000	-0.000

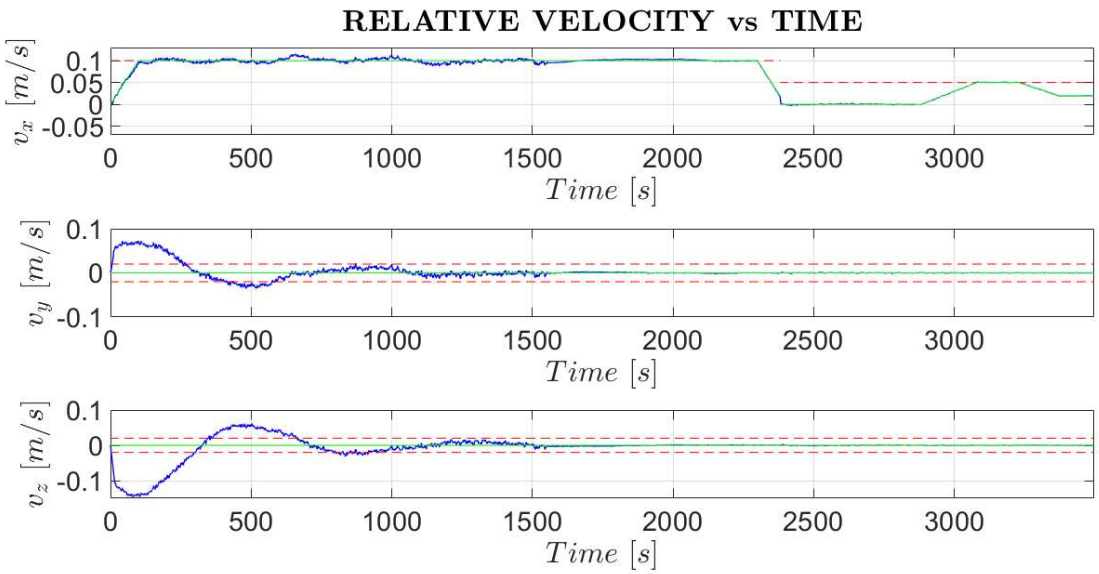


Figure 6.18: Relative velocity of the chaser with PID and disturbances in blue; reference speed in green; speed constraints dashed in red.

Relative position and velocity show a trend similar to Section 6.4, despite some more oscillations. However, the requirements are satisfied, so this controller is worth to be better analyzed.

The position estimates are obtained in the same way as the previous simulation, whereas velocity is estimated by two Kalman filters, one for each approach cone, designed as in Subsection 3.4.1.

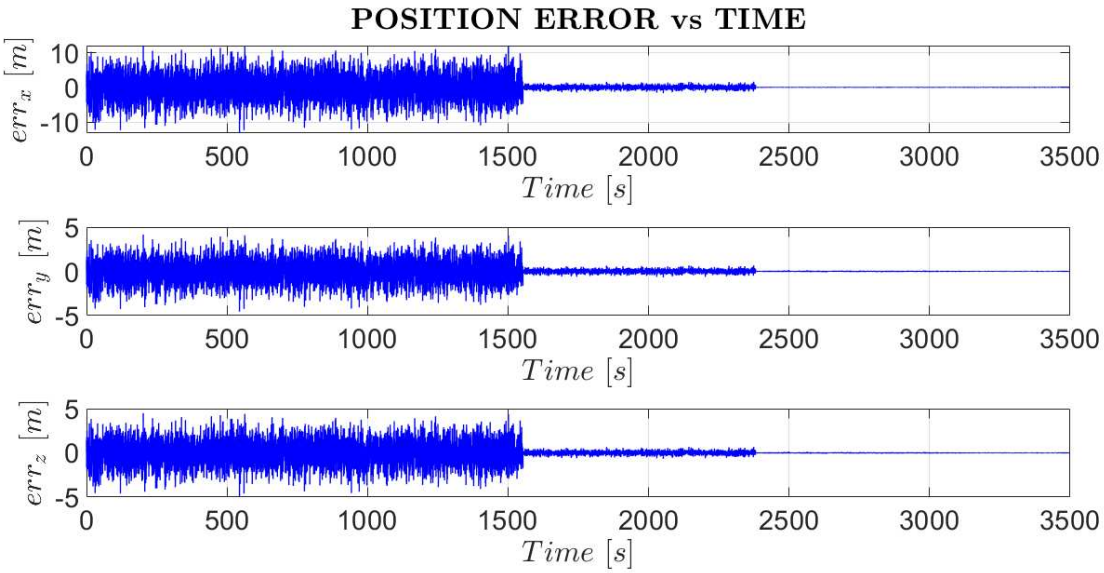


Figure 6.19: Position estimation error (PID).

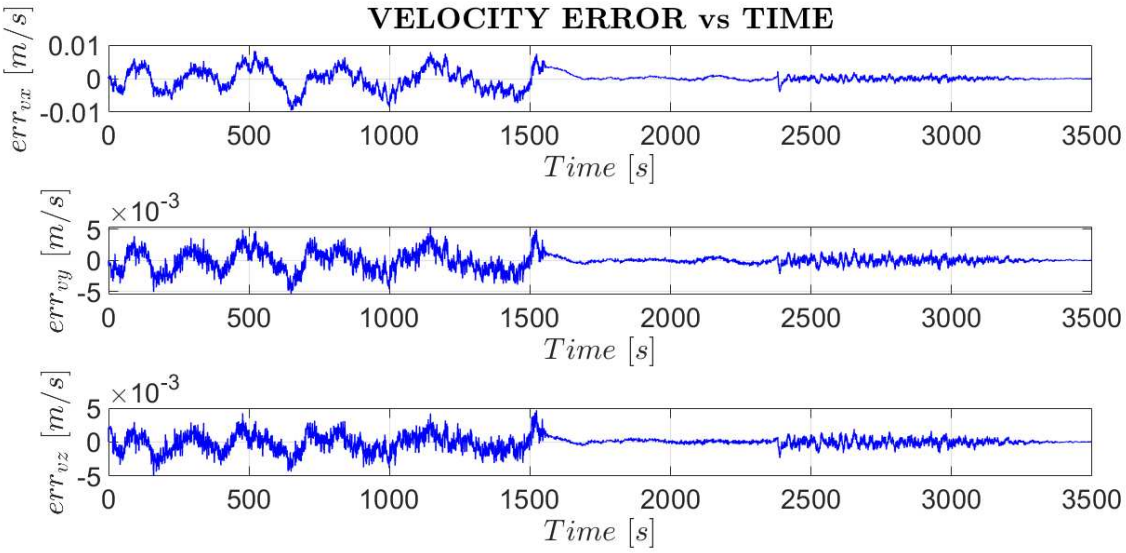


Figure 6.20: Velocity estimation errors (PID).

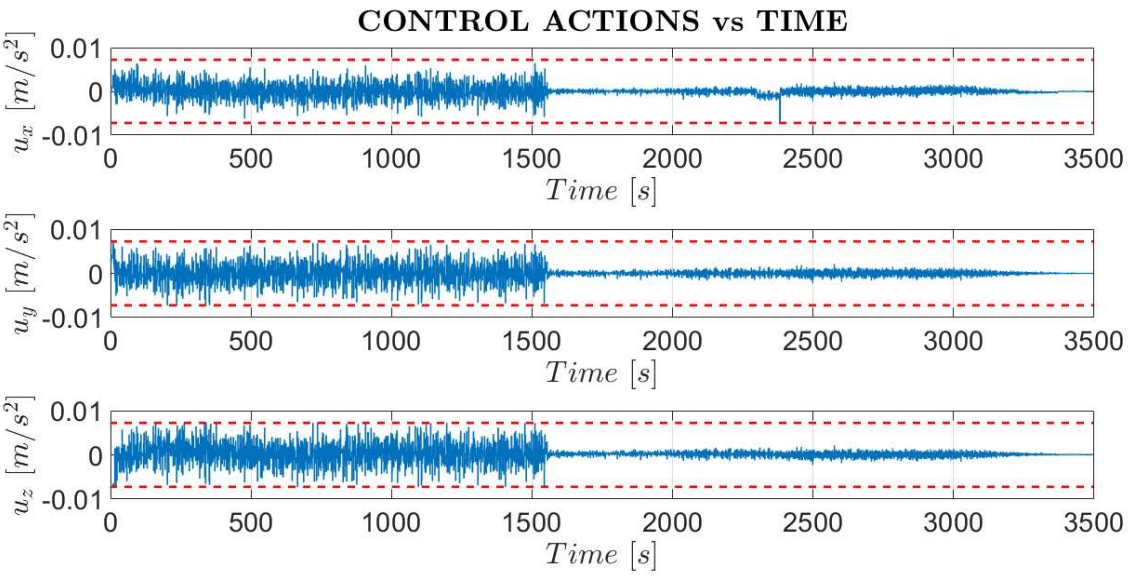


Figure 6.21: Commanded accelerations in blue (PID). Constraints dashed in red.

Table 6.18: Use of delta-V in the entire maneuver with PID and disturbances.

Delta V [m/s]	
ΔV_x	3.317521
ΔV_y	3.933014
ΔV_z	4.284259
ΔV_{total}	11.534794

With regard to the control actions, a higher use of thrusters is commanded by the PID. This is due to the measurement noise and disturbances, and therefore the delta-Vs result in larger values.

6.7 Attitude control

With regard to the spacecraft attitude control, the linear quadratic MPC cannot be adopted as the equations of dynamics are nonlinear, therefore the choice of another type of controller is necessary. The options were to use a nonlinear MPC controller or a PID.

The former had been used at the beginning, but high computational effort and thereby too long simulation times led to the choice of a PID control. Indeed, when the chaser is far away from the target, the attitude control accuracy is not as demanding as in the closer phase, so a complex controller such as a nonlinear MPC would be unnecessary.

In this section we will present the result of a PID controller implemented in the software simulator, in order to control the attitude of the chaser during the transfer maneuvers.

As mentioned before, the ADCS subsystem is assumed to guarantee an attitude estimation better than 0.15° and this is simulated by adding a Gaussian noise of 0.15° and 0.15 °/s (3σ) respectively to the actual values. The following results are obtained running a simulation with the objective of bringing the spacecraft to the null nominal conditions. The initial conditions are listed in Table 6.19, while the PID parameters are shown in Table 6.20. From Figure 6.22 to Figure 6.28 the performances of the attitude controller are illustrated.

Table 6.19: Initial conditions and setpoints for the attitude control.

	Initial	**Setpoint**
Roll (ϕ)	8 deg	0 deg
Pitch (θ)	5 deg	0 deg
Yaw (ψ)	-7 deg	0 deg
ω_x	0.5 deg/s	0 deg/s
ω_y	-0.8 deg/s	0 deg/s
ω_z	0.3 deg/s	0 deg/s

Table 6.20: Gains of the attitude PID controller.

Attitude PID Gains			
	x	y	z
Kp	8×10^{-4}	8×10^{-4}	8×10^{-4}
Ki	1×10^{-5}	1×10^{-5}	1×10^{-5}
Kd	3.5×10^{-3}	3.5×10^{-3}	3.5×10^{-3}
Kt	1×10^{-4}	1×10^{-4}	1×10^{-4}
N	90	90	90
β	1	1	1
γ	1	1	1

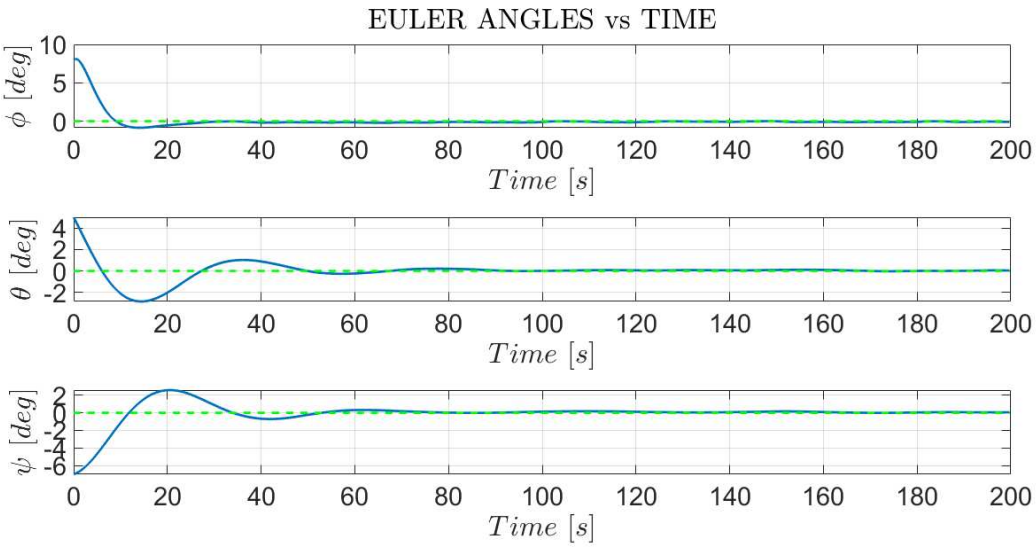


Figure 6.22: Roll, pitch, and yaw angles.

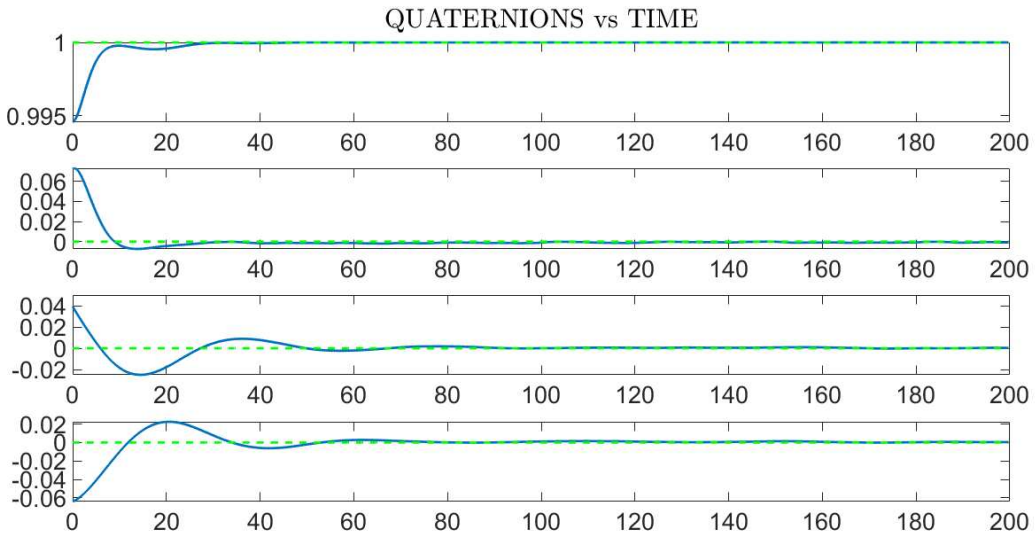


Figure 6.23: Quaternion components of the attitude state.

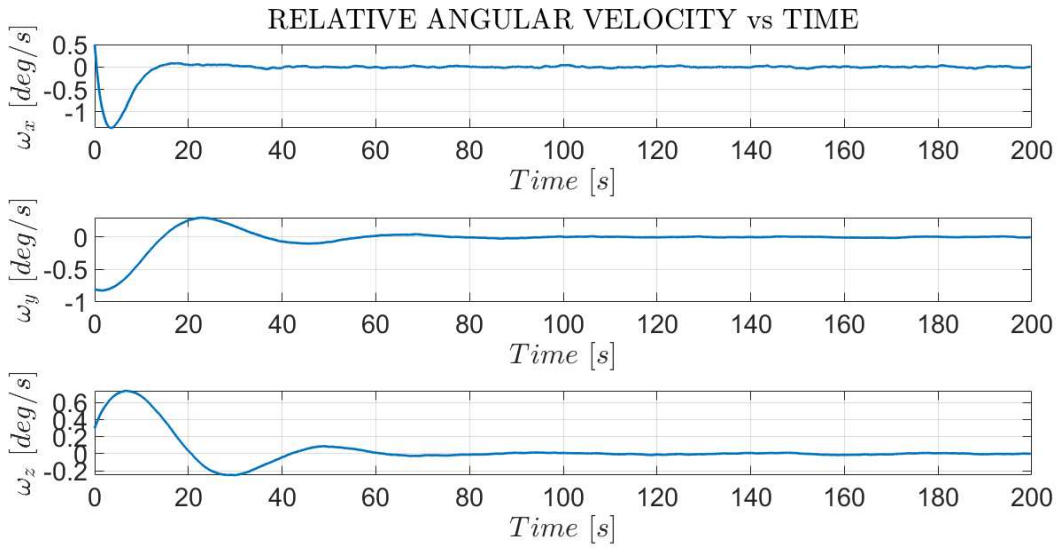


Figure 6.24: Angular velocity of the chaser w.r.t. the LVLH frame.

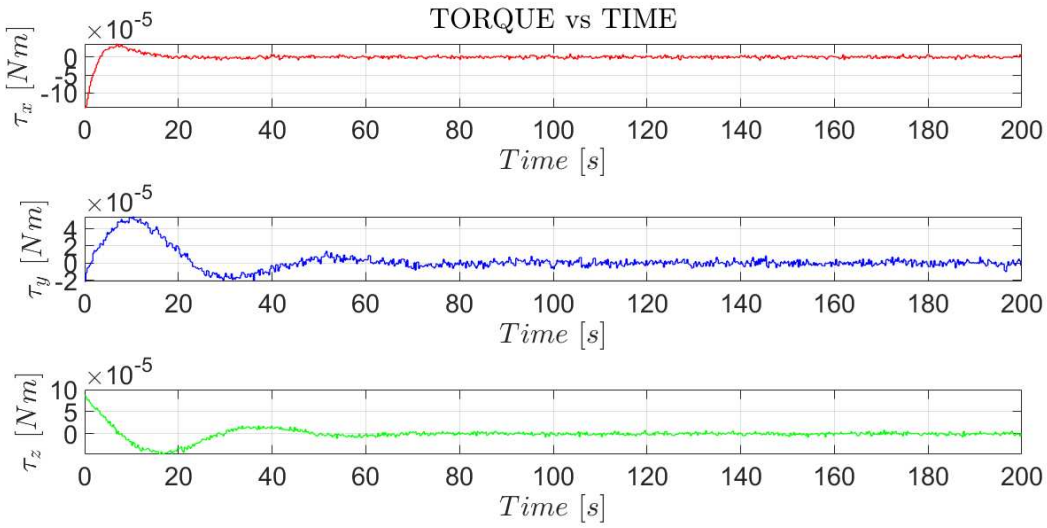


Figure 6.25: Torques generated by the reaction wheels, acting on the body. Remember that the motor torque of each wheel is opposite in sign.

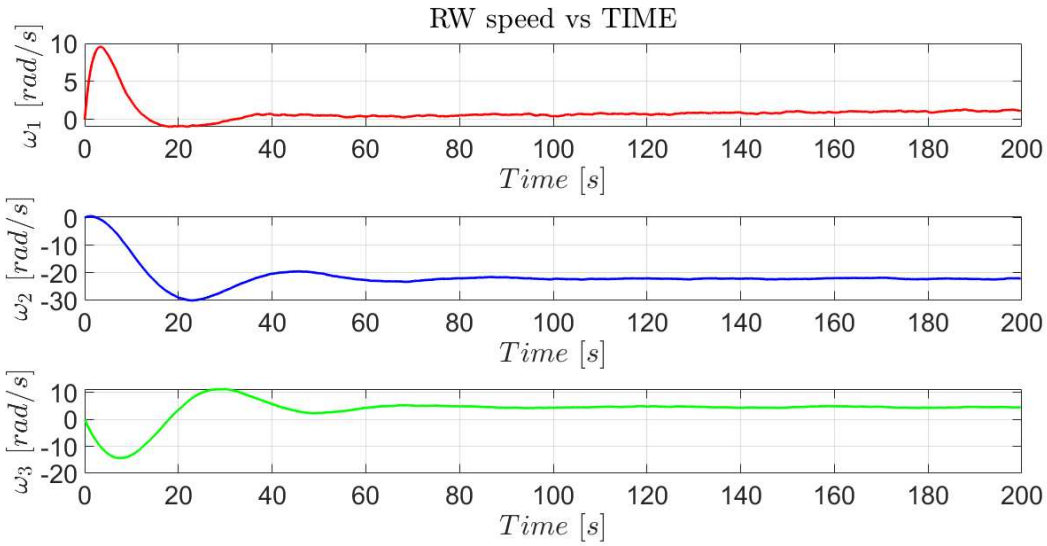


Figure 6.26: Speed of the reaction wheels w.r.t. the body frame.

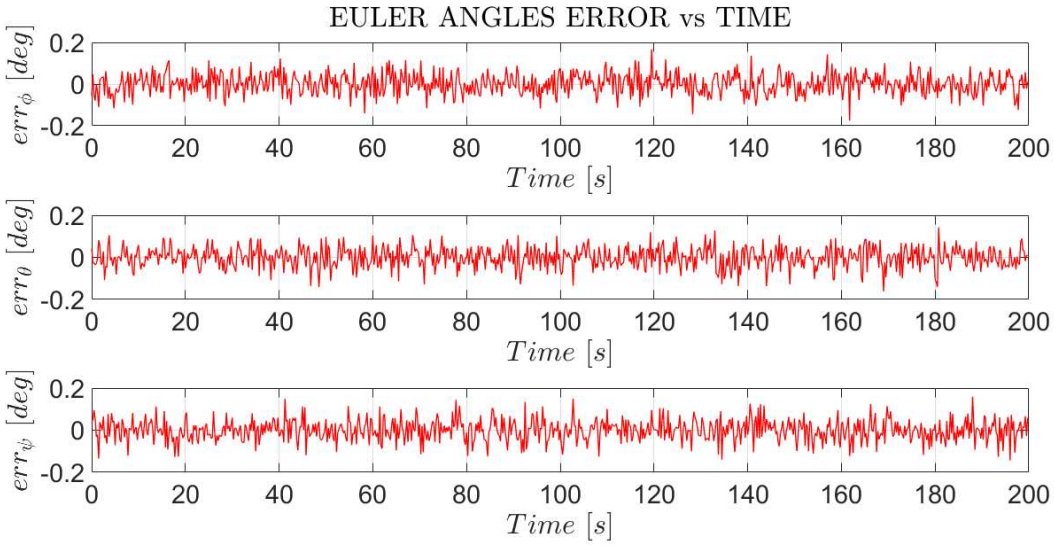


Figure 6.27: Error of Euler angles.

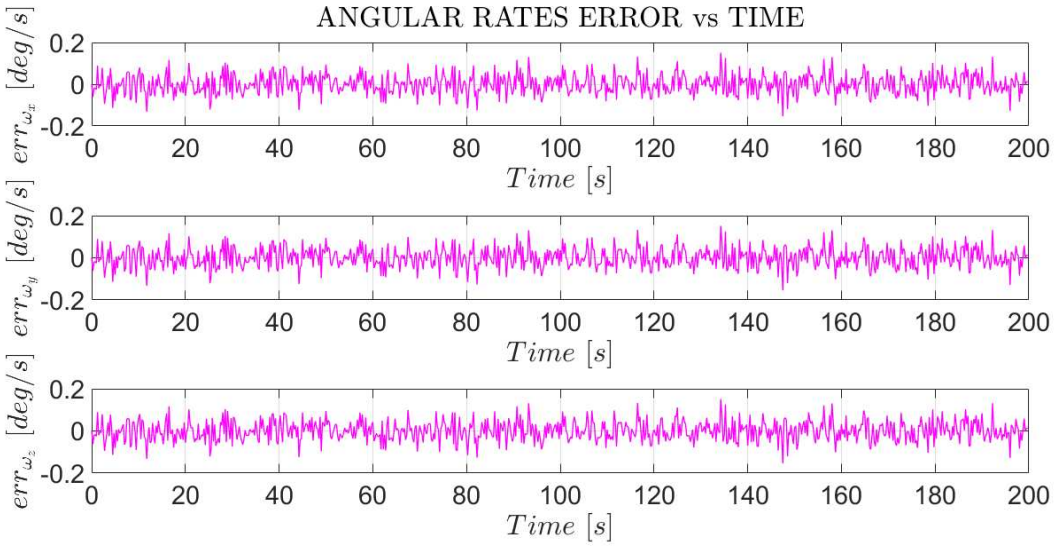


Figure 6.28: Error of angular rates.

The results prove that the PID controller chosen is effective and able to provide control accuracy within the range established at the beginning. After only 200 s, which is a short time compared to that of the approaching maneuver, the steady state values are reached and are all within the limits of the docking requirements, as shown in Table 6.21.

Table 6.21: Attitude states after 200 s.

Performances	
Roll	-3.03×10^{-2} deg
Pitch	6.90×10^{-2} deg
Yaw	4.65×10^{-2} deg
ω_x	2.55×10^{-3} deg/s
ω_y	5.92×10^{-2} deg/s
ω_z	4.54×10^{-3} deg/s

Chapter 7

Montecarlo simulations

7.1 Introduction

In this chapter, the analysis of a 1000-run Montecarlo simulation for the MPC and PID controllers is presented. The objective of the simulations is to verify whether the spacecraft can effectively reach the goals of the rendezvous and docking mission, independent of its initial conditions at hold point S3.

These initial states are randomly picked from a determined range of values that encompass all possible conditions expected to happen at the beginning of the final approach corridor at hold point S3. ATV's accuracy requirements at point S3, described in [25] and [26], have been used as an estimate of possible ranges for the initial conditions. To be more conservative, the maximum and minimum values for the initial translational and rotational states have been enlarged, so as to consider worst-case scenarios and verify robustness of the controller. The range of the initial conditions are described in Tables 7.1 and 7.2.

Table 7.1: Initial translational states at S3.

Translational initial state range		
State	Min.	Max.
$x\ [m]$	-500	-250
$y\ [m]$	-30	30
$z\ [m]$	-30	30
$\dot{x}\ [m/s]$	-0.20	0.20
$\dot{y}\ [m/s]$	-0.20	0.20
$\dot{z}\ [m/s]$	-0.20	0.20

Table 7.2: Initial attitude states at S3.

Rotational initial state range		
State	Min.	Max.
Roll $[deg]$	-30	30
Pitch $[deg]$	-30	30
Yaw $[deg]$	-30	30
$\dot{\omega}_x\ [deg/s]$	-10	10
$\dot{\omega}_y\ [deg/s]$	-10	10
$\dot{\omega}_z\ [deg/s]$	-10	10

It should be noted that the translational and rotational dynamics are decoupled in our case. Therefore, the results from the attitude PID controller will be presented only once, since the same attitude controller is used for both Montecarlo simulations.

7.2 MPC Montecarlo results

The results shown from Figure 7.1 to Figure 7.9 are valid for an MPC designed with the same characteristics of Section 6.5.

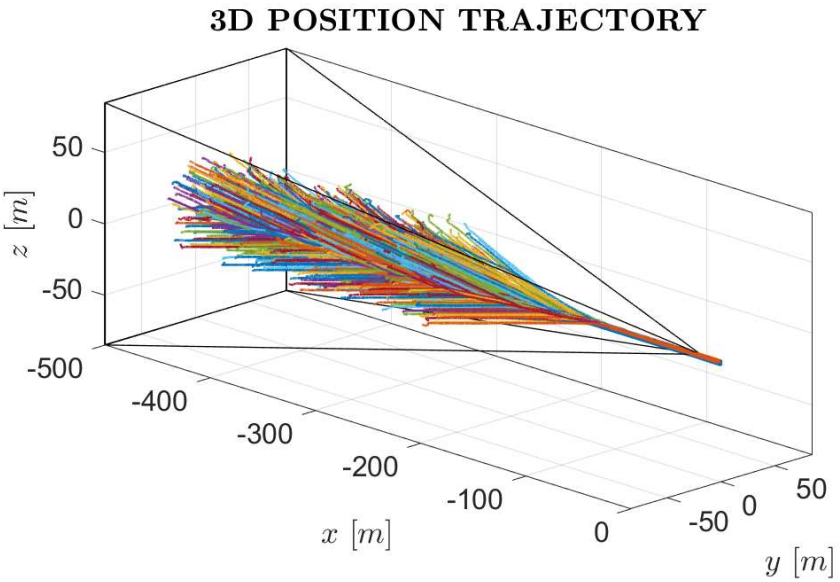


Figure 7.1: Trajectories of 1000 runs with MPC.

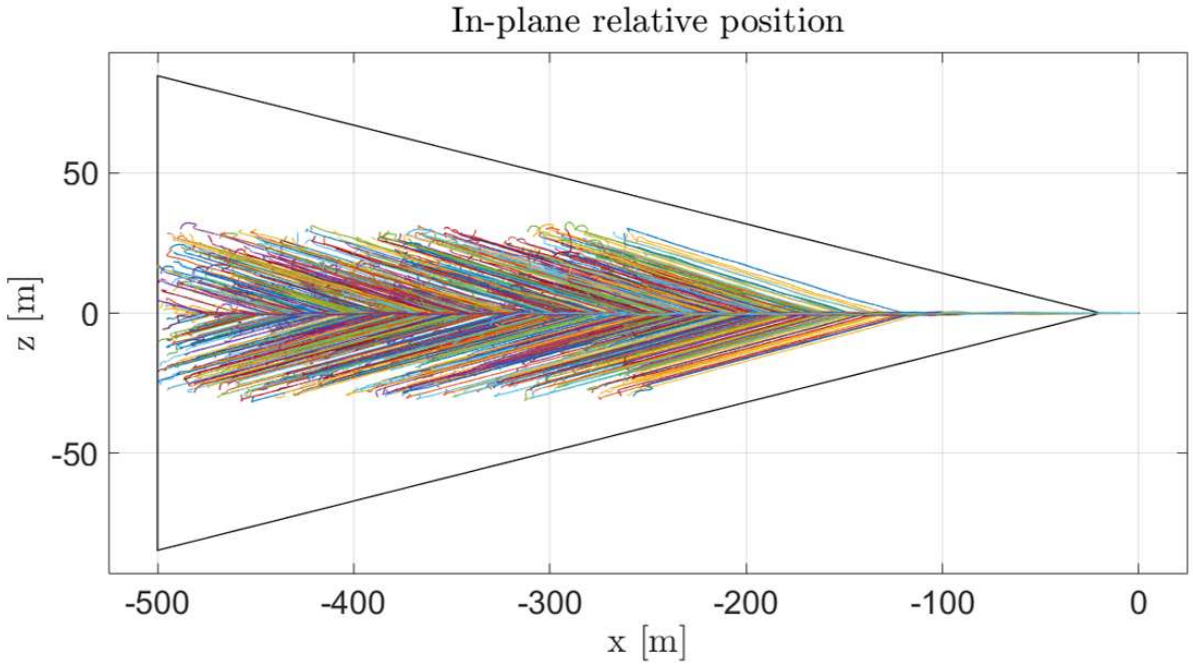


Figure 7.2: In-plane trajectory, from S3 to docking.

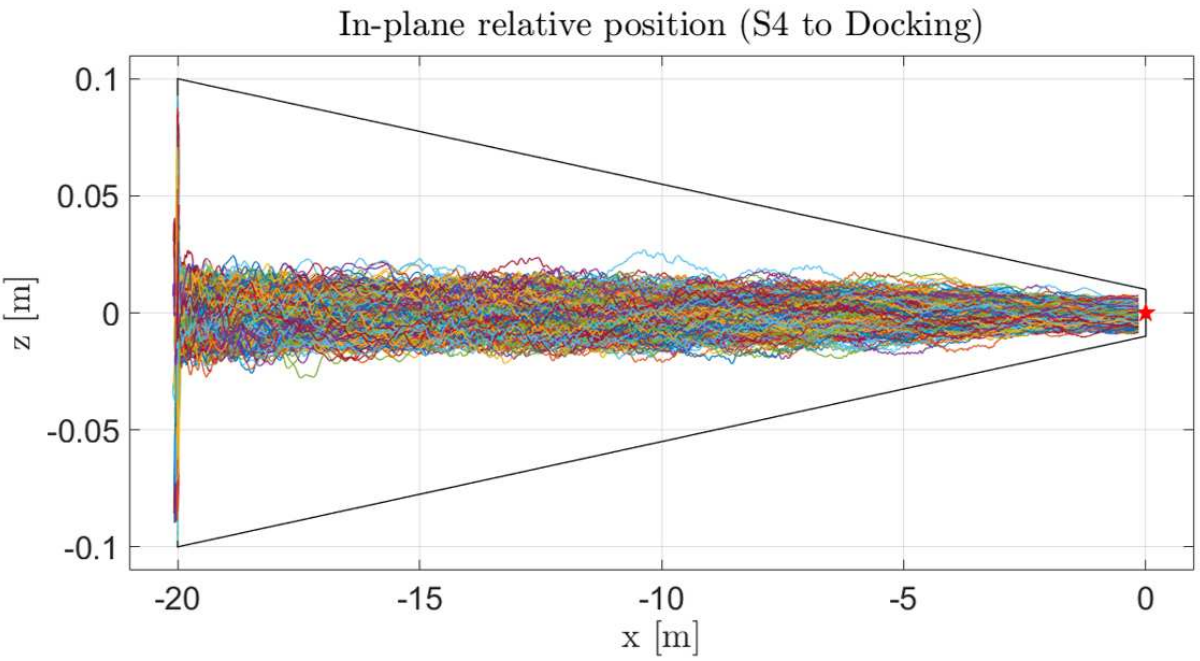


Figure 7.3: In-plane trajectory in the second approach cone, from S4 to docking.

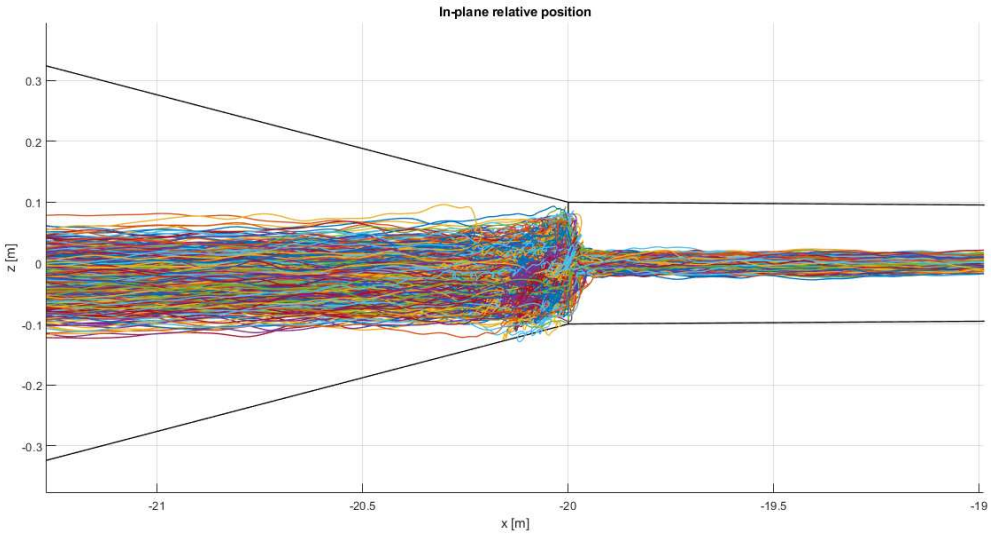


Figure 7.4: Zoom in the in-plane station keeping at point S4.

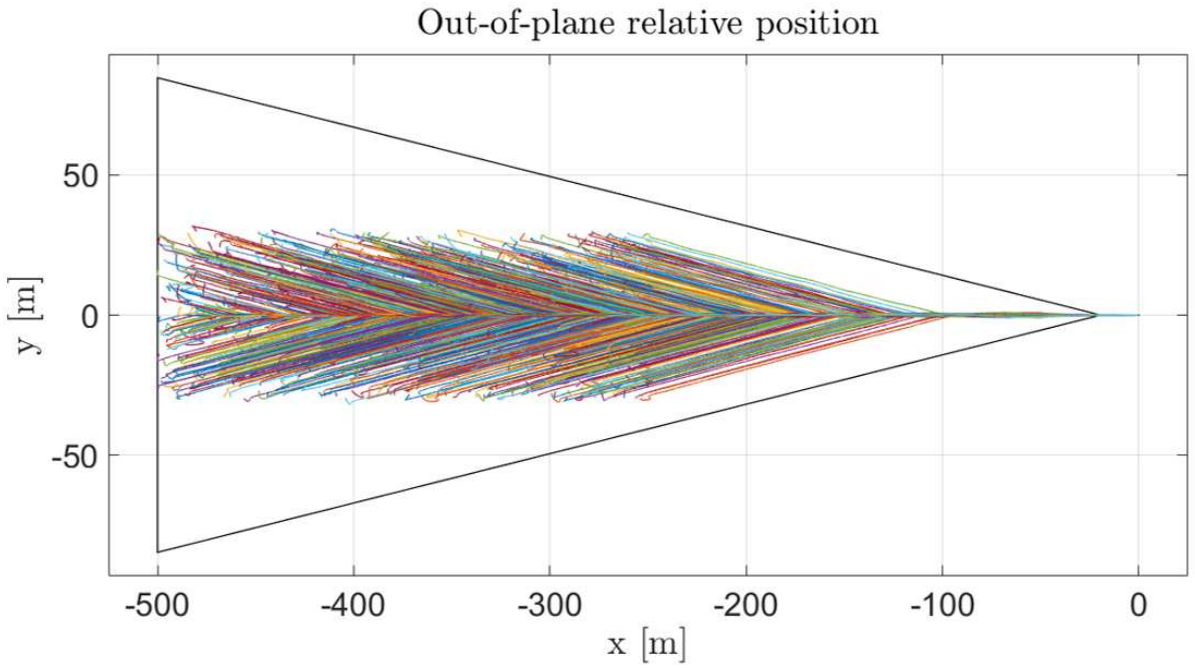


Figure 7.5: Out-of-plane trajectory from S3 to docking.

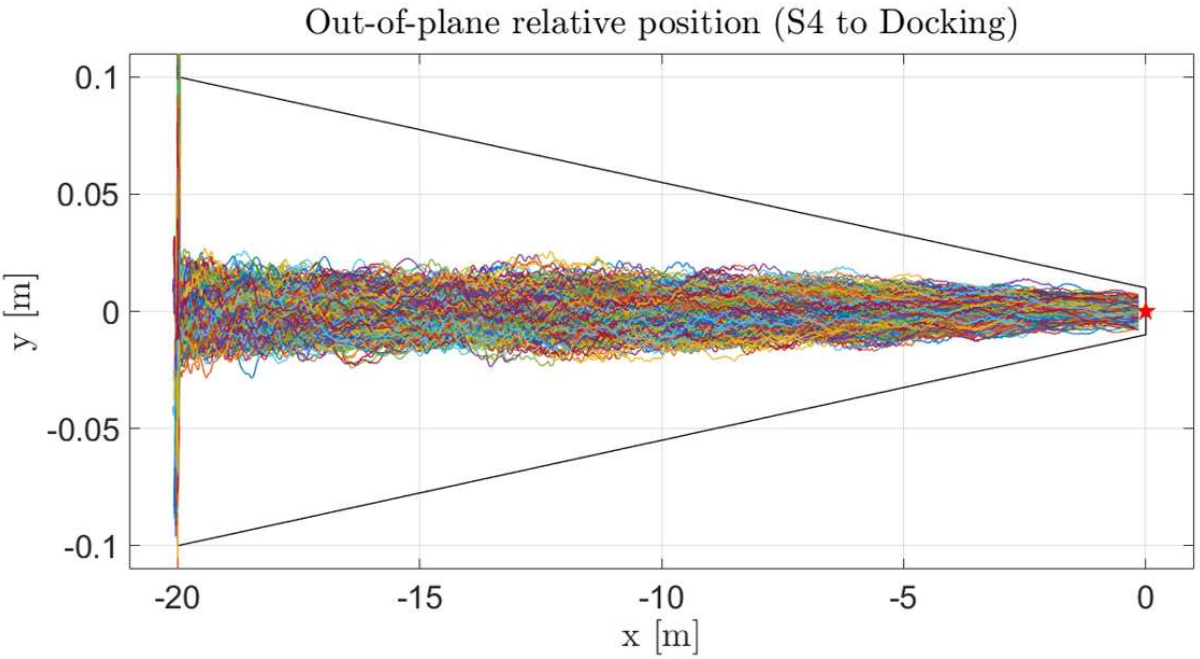


Figure 7.6: Out-of-plane trajectory in the second approach cone, from S4 to docking.

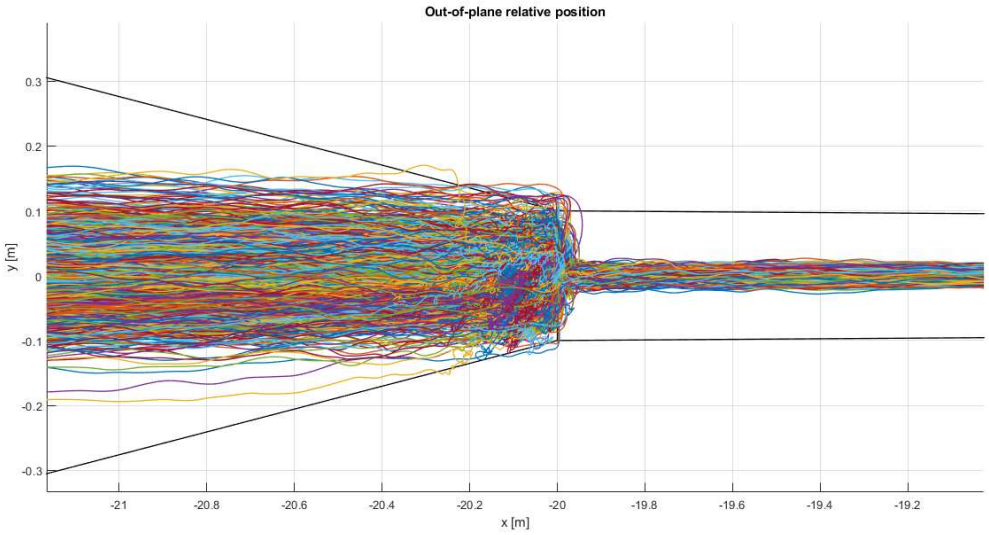


Figure 7.7: Zoom in the out-of-plane station keeping at point S4.

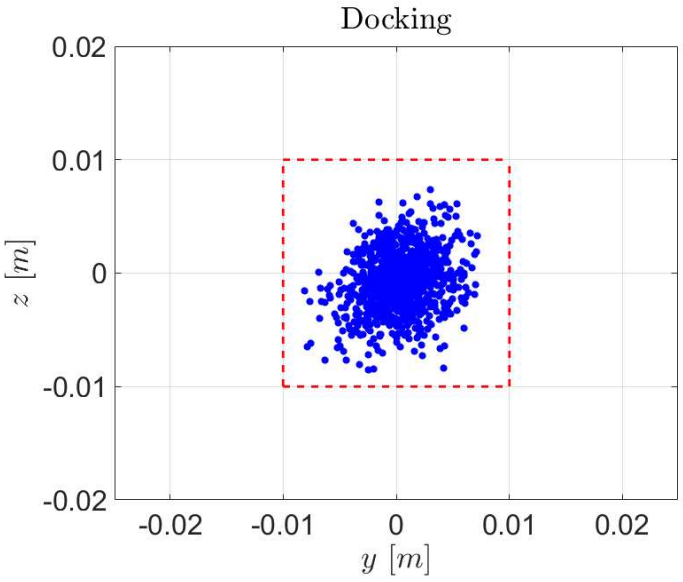


Figure 7.8: Docking performances with MPC.

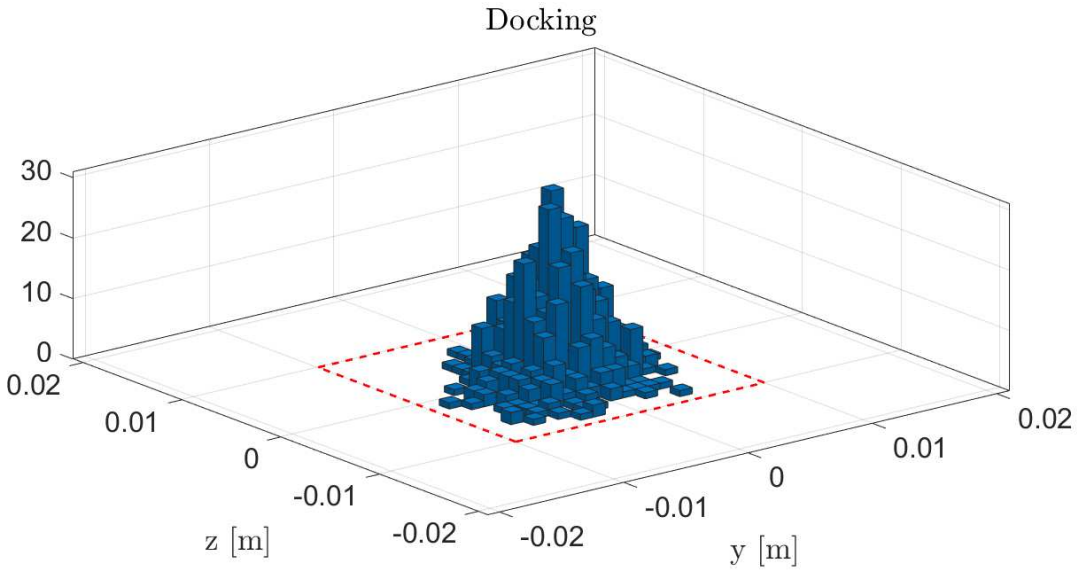


Figure 7.9: Histogram of docking performances with MPC.

All 1000 trajectories reach the final point at docking respecting the docking port constraints. At hold point S4, a few trajectories violate the boundaries of the corridors, as shown by Figures 7.4 and 7.7, but this infringement is of only a few centimeters, which may be tolerated.

To quantify the performance of the MPC, it has been computed the average values of the misalignments and velocity errors at docking, along with their standard deviation. The histograms in the Figures 7.10 and 7.11 are also a good interpretations of these results.

Table 7.3: Lateral misalignments and velocity errors at docking with MPC.

Lateral misalignment [m]

	Avg.	3σ	Min.	Max.	Limit
y	3.68e-4	7.71e-3	-8.06e-3	7.15e-3	1e-2
z	-7.69e-4	7.65e-3	-8.53e-3	7.39e-3	1e-2

Velocity error [m/s]

v_x	-1.51e-5	3.66e-4	-4.09e-4	3.74e-4	1e-2
v_y	-6.82e-6	3.45e-4	-3.74e-4	3.50e-4	1e-2
v_z	-2.51e-5	3.69e-4	-5.18e-4	3.32e-4	1e-2

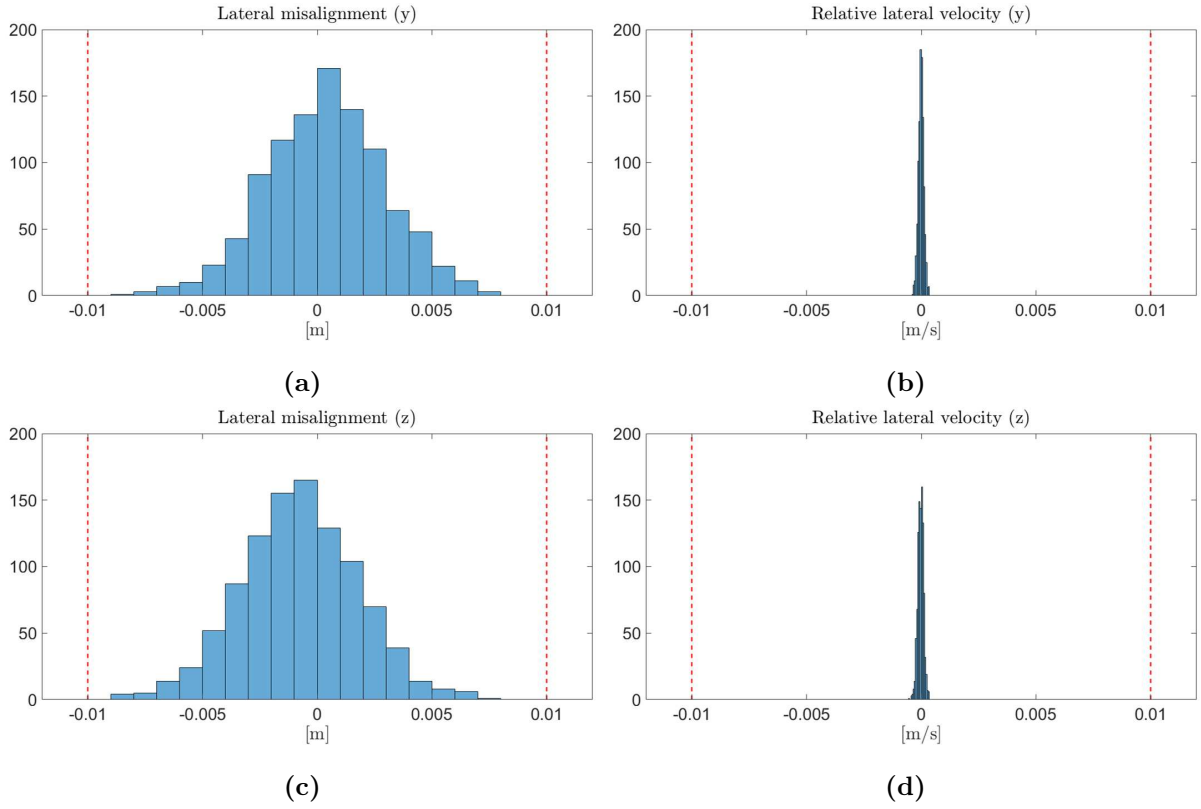


Figure 7.10: Lateral misalignments (y, z) and lateral velocity errors at docking with MPC. All samples are within the limits.

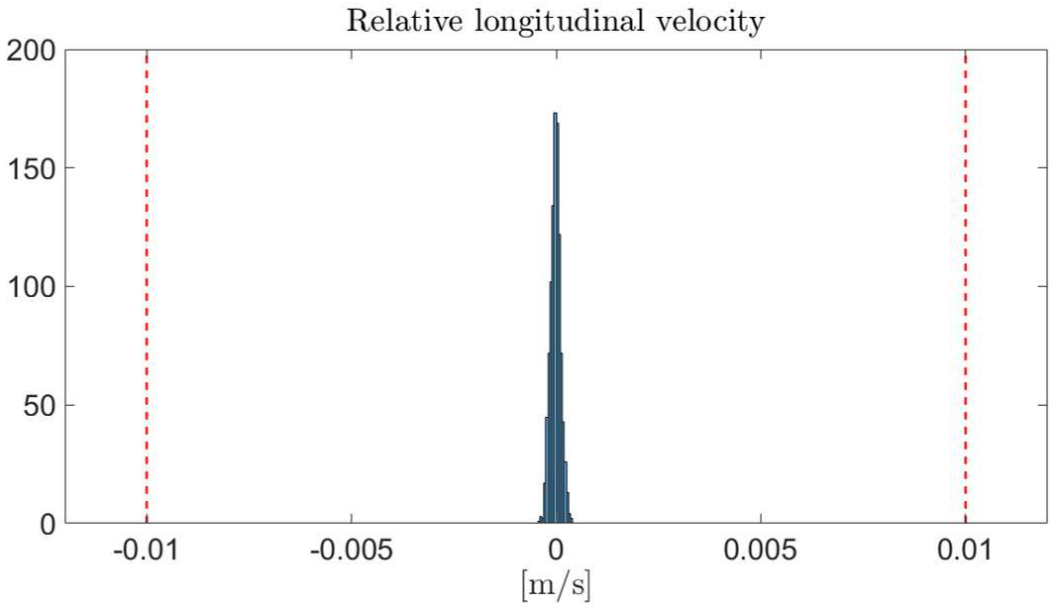


Figure 7.11: Longitudinal velocity error at docking with MPC.

The usage of thrusters depends clearly from the initial position and velocity of the vehicle, which span over long ranges especially for the longitudinal direction. Hence, it is expected that some simulations required more use of fuel than others. To give an idea of the delta-Vs required, Table 7.4 shows the average, minimum, and maximum delta-V obtained from the Montecarlo simulation.

Table 7.4: Delta-V used with MPC (in body frame).

Delta-V [m/s]

	Avg.	Min.	Max.
x	1.986	1.521	4.021
y	1.911	1.484	4.786
z	2.476	1.947	5.115
Total	6.373	5.059	13.921

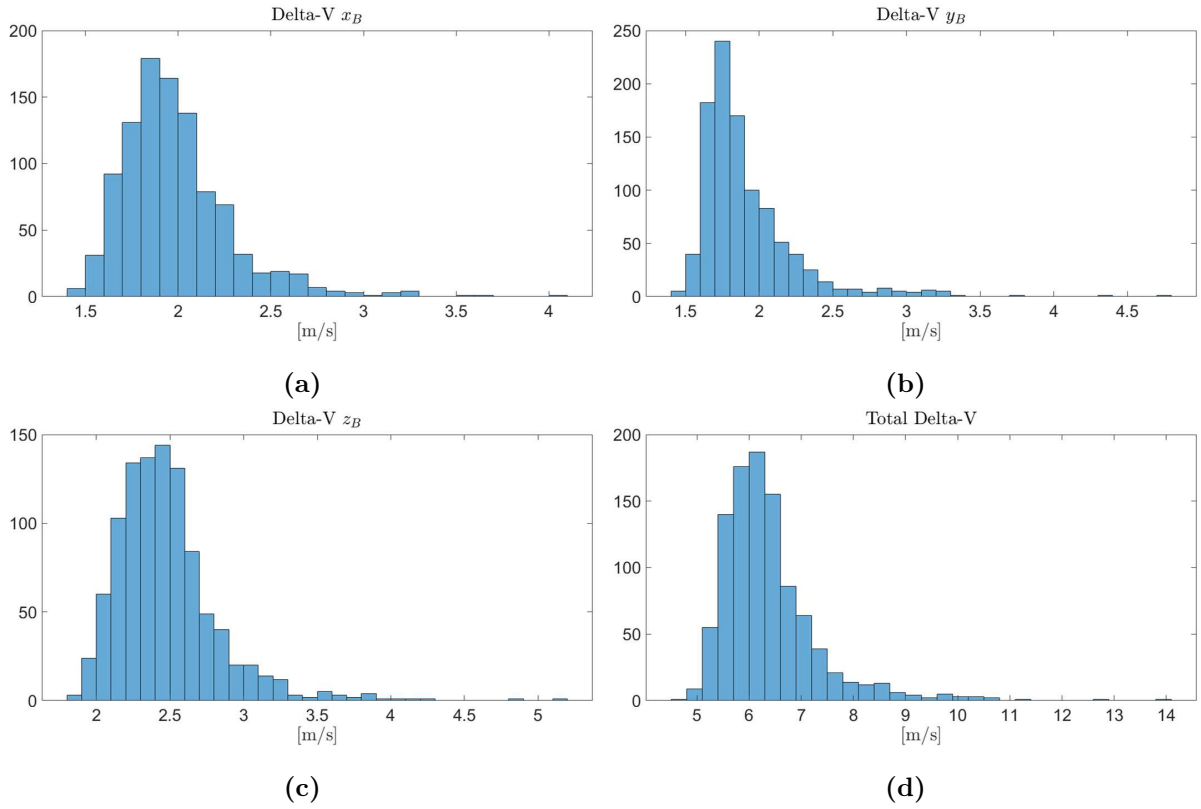


Figure 7.12: Delta-Vs with MPC (in body frame). (a) X-direction; (b) Y-direction; (c) Z-direction; (d) Sum of each axis' delta-V.

7.3 PID Montecarlo results

In this section we present the results of the Montecarlo simulation carried out with the PID control, as designed in Section 6.6. The results are illustrated from Figures 7.13 to 7.21.

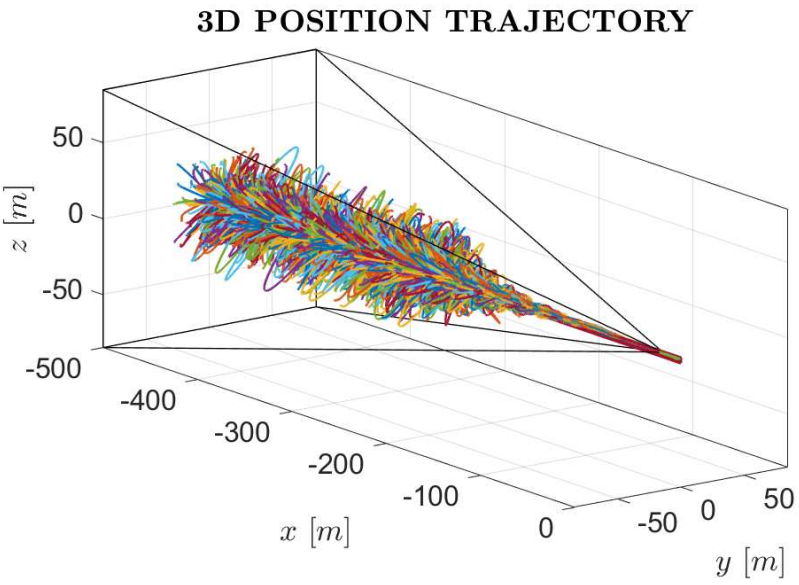


Figure 7.13: Trajectories of 1000 runs with PID.

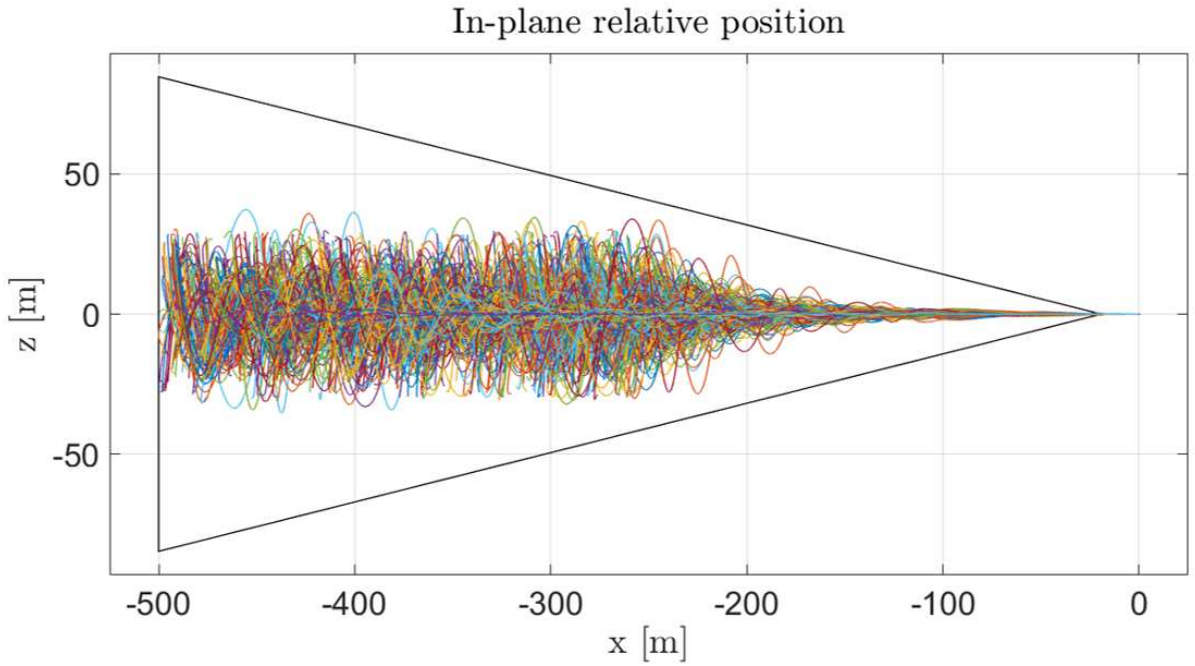


Figure 7.14: In-plane trajectory, from S3 to docking.

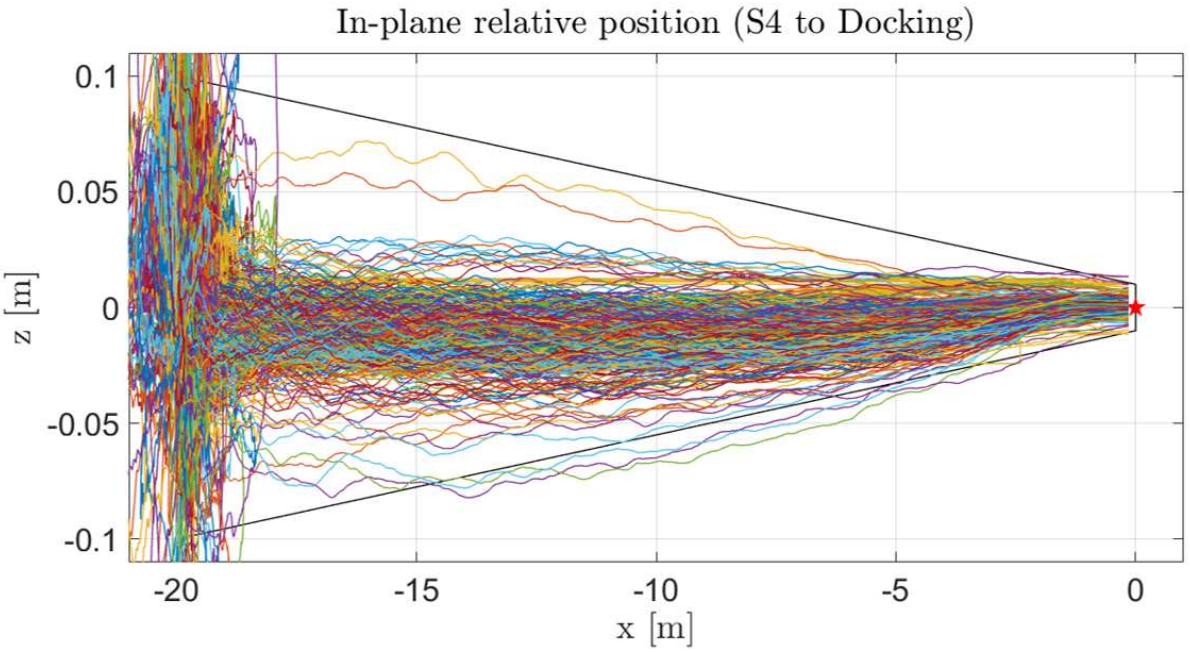


Figure 7.15: In-plane trajectory in the second approach cone, from S4 to docking.

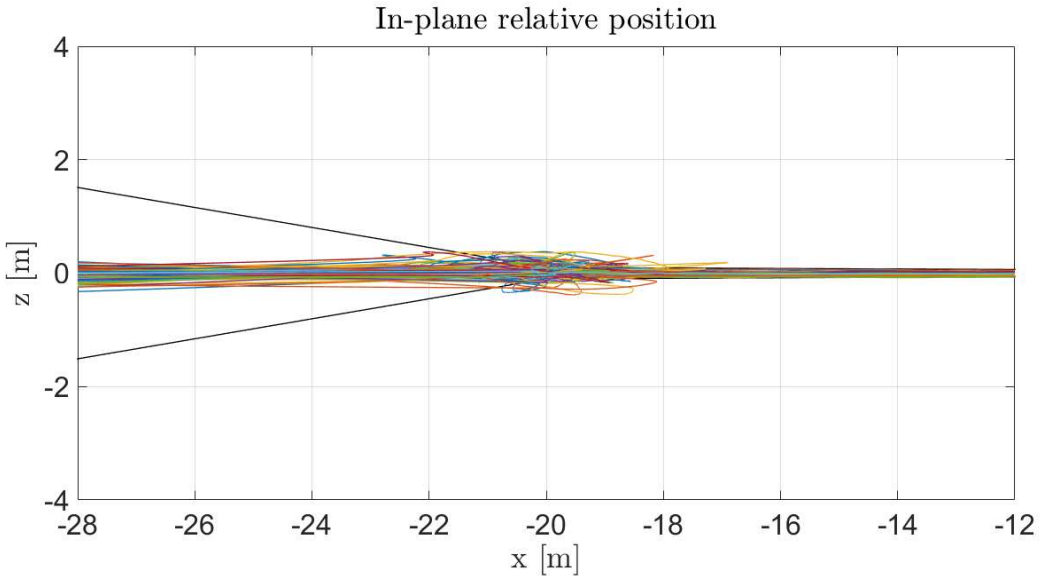


Figure 7.16: Zoom in the in-plane station keeping at point S4.

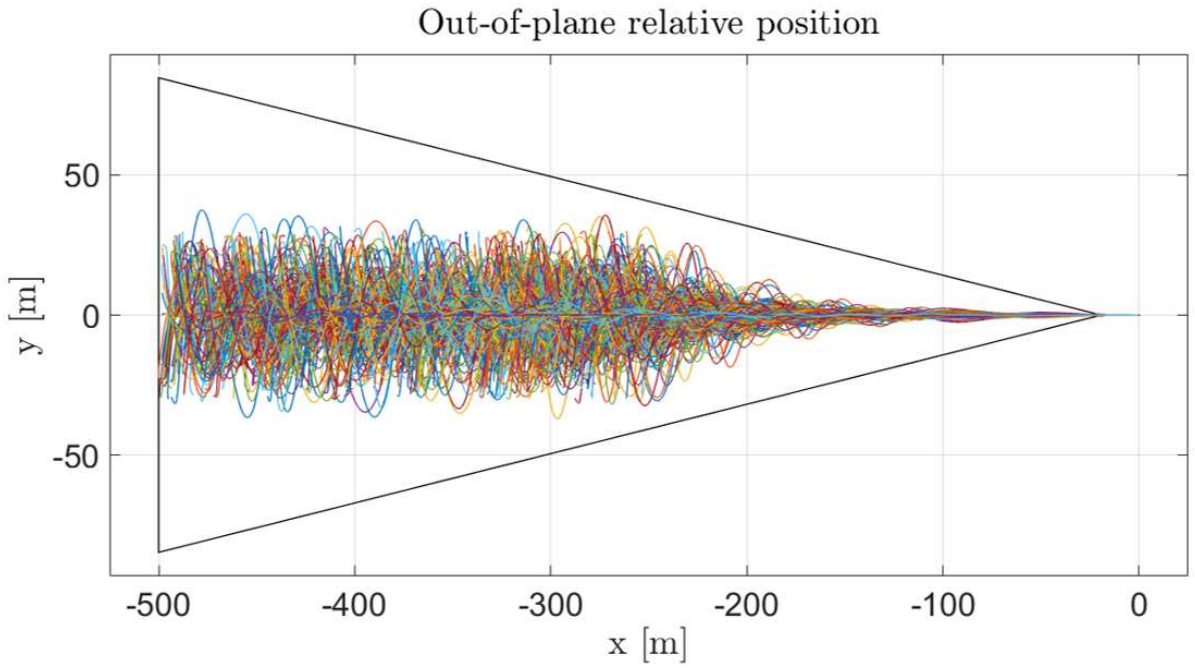


Figure 7.17: Out-of-plane trajectory from S3 to docking.

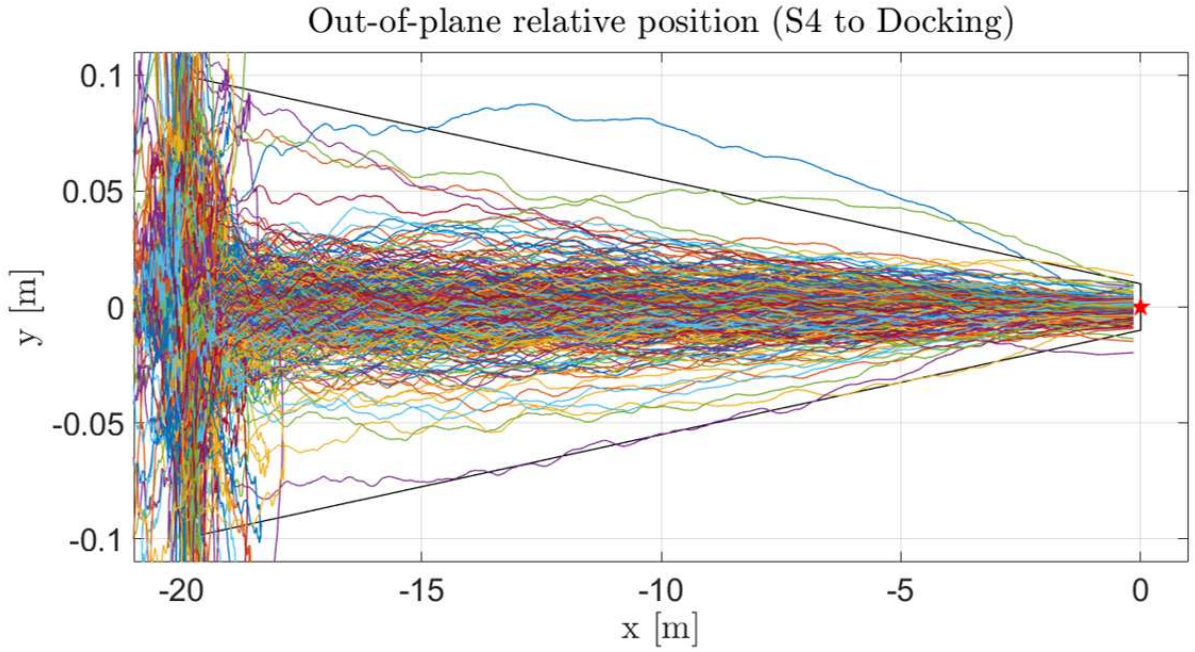


Figure 7.18: Out-of-plane trajectory in the second approach cone, from S4 to docking.

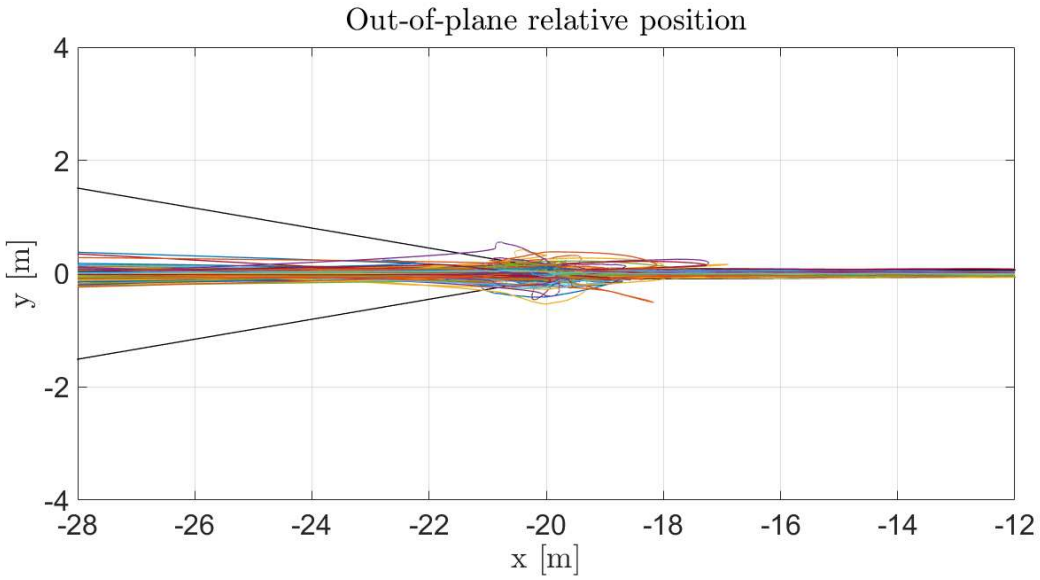


Figure 7.19: Zoom in the out-of-plane station keeping at point S4.

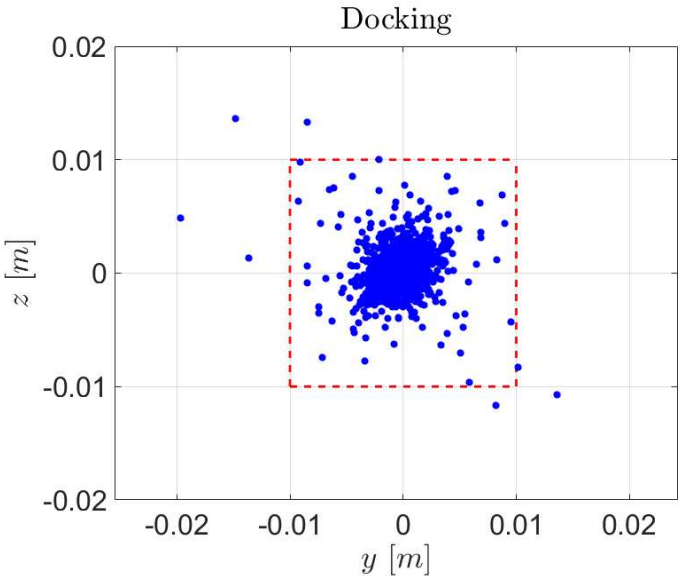


Figure 7.20: Docking performances with PID.

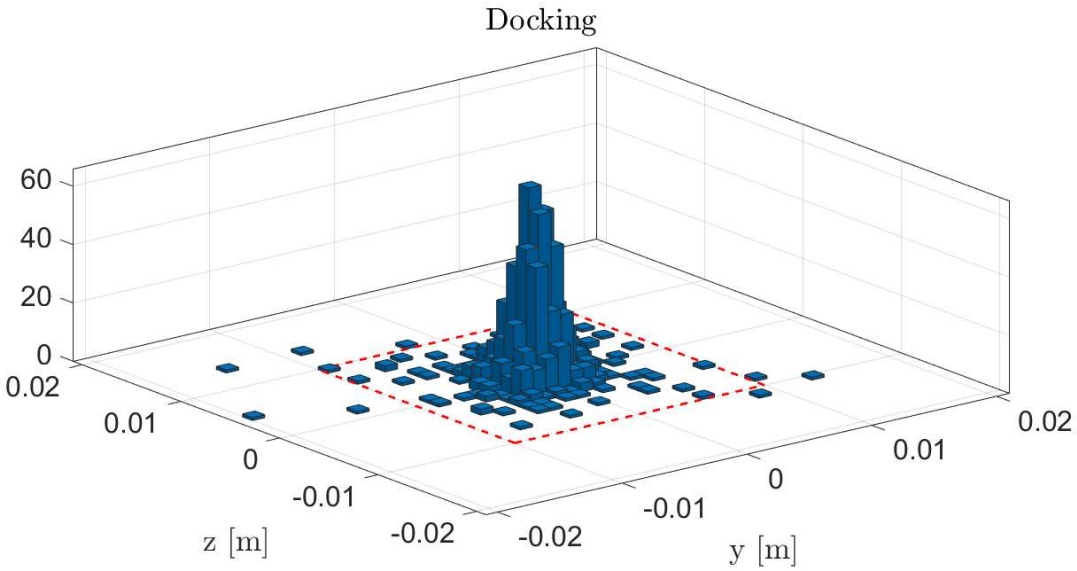


Figure 7.21: Histogram of docking performances with PID.

Unlike the previous simulation, this controller seems not to perform as well. In Figures 7.15 and 7.18 it is evident that the chaser violates the constraints of the second approach cone in several instances, and so happens during the station-keeping at point S4. Moreover, high oscillations are present in the final part of the approach.

As a result, not all the trajectories reach the docking point within the limits, as depicted in Figures 7.20 and 7.21.

Table 7.5: Lateral misalignments and velocity errors at docking with PID.

Lateral misalignment [m]

	Avg.	3σ	Min.	Max.	Limit
y	-3.48e-4	7.05e-3	-1.96e-2	1.36e-2	1e-2
z	3.26e-4	6.72e-3	-1.16e-2	1.36e-2	1e-2

Velocity error [m/s]

v_x	-6.04e-5	6.97e-4	-8.55e-4	7.87e-4	1e-2
v_y	-1.83e-7	8.13e-5	-1.60e-4	1.17e-4	1e-2
v_z	-8.56e-6	1.10e-4	-1.46e-4	1.18e-4	1e-2

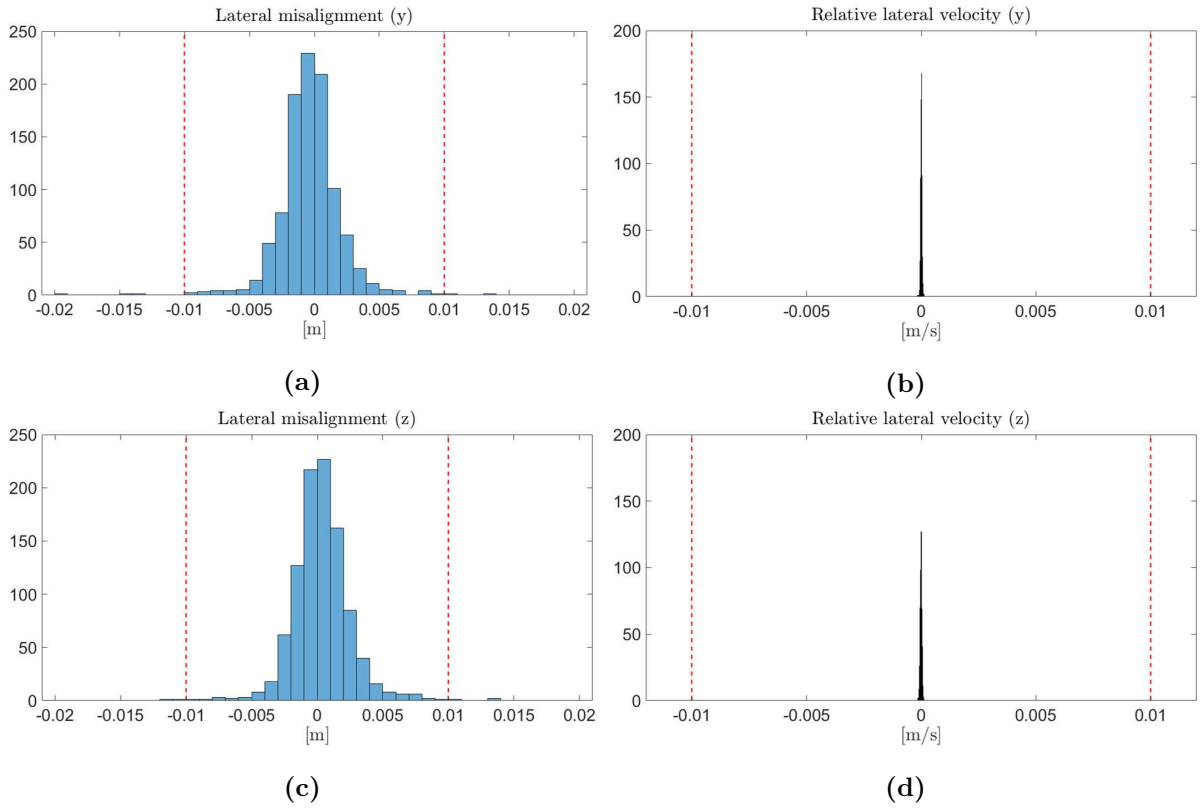


Figure 7.22: Lateral misalignments (y, z) and lateral velocity errors at docking with PID. Not all samples respect the limits.

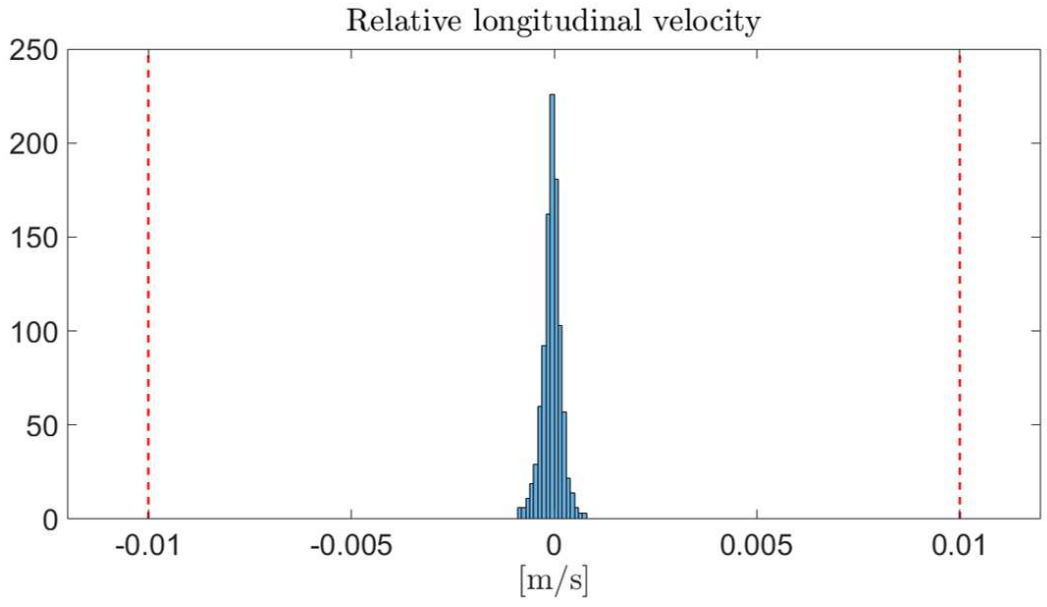


Figure 7.23: Longitudinal velocity error at docking with PID.

Concerning relative velocity, high accuracy is nevertheless reached at the docking point, but the same cannot be said for the initial approach, when the velocity constraints

cannot be respected by this kind of controller.

The delta-Vs are higher than in the MPC case, as shown in the Table 7.6 and Figure 7.24.

Table 7.6: Delta-V used with PID (in body frame).

Delta-V [m/s]

	Avg.	Min.	Max.
x	4.781	2.866	6.770
y	6.610	3.596	13.137
z	7.060	3.885	12.937
Total	18.451	11.108	31.056

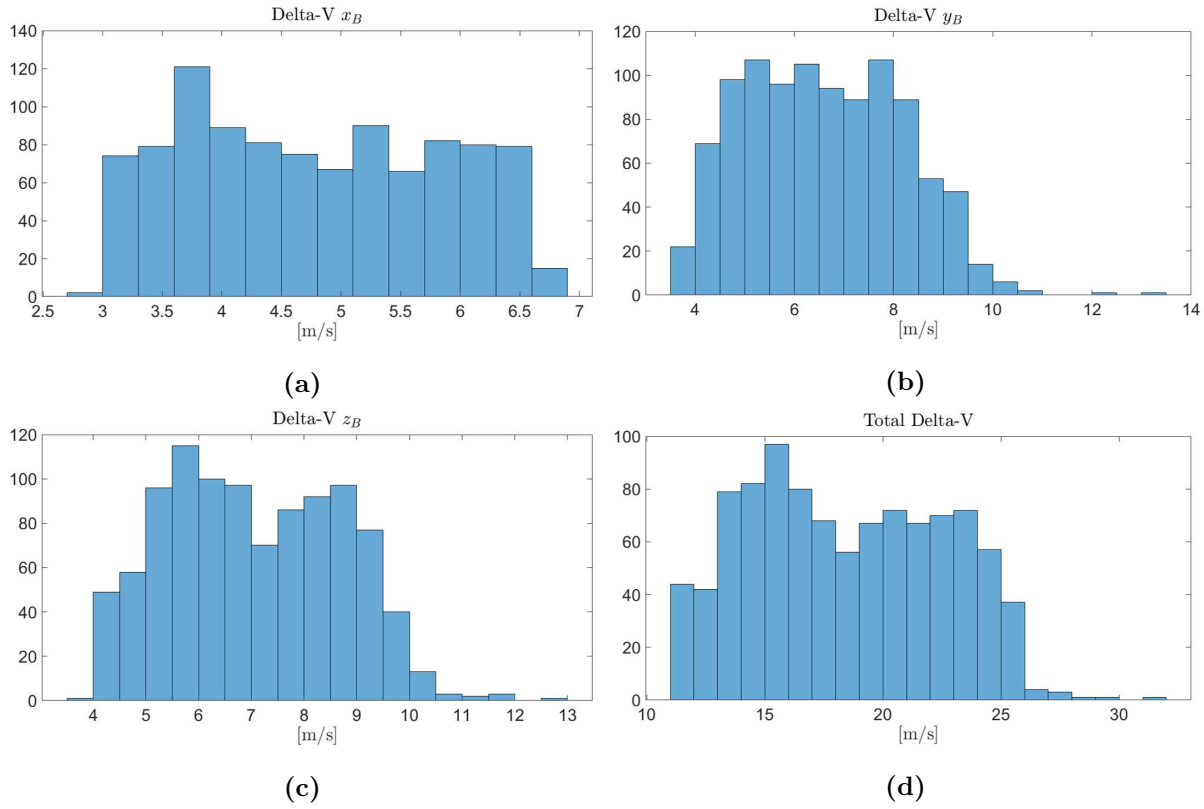


Figure 7.24: Delta-Vs with PID (in body frame). (a) X-direction; (b) Y-direction; (c) Z-direction; (d) Sum of each axis' delta-V.

Figure 7.25 shows a comparison of the total delta-V histograms between MPC and PID.

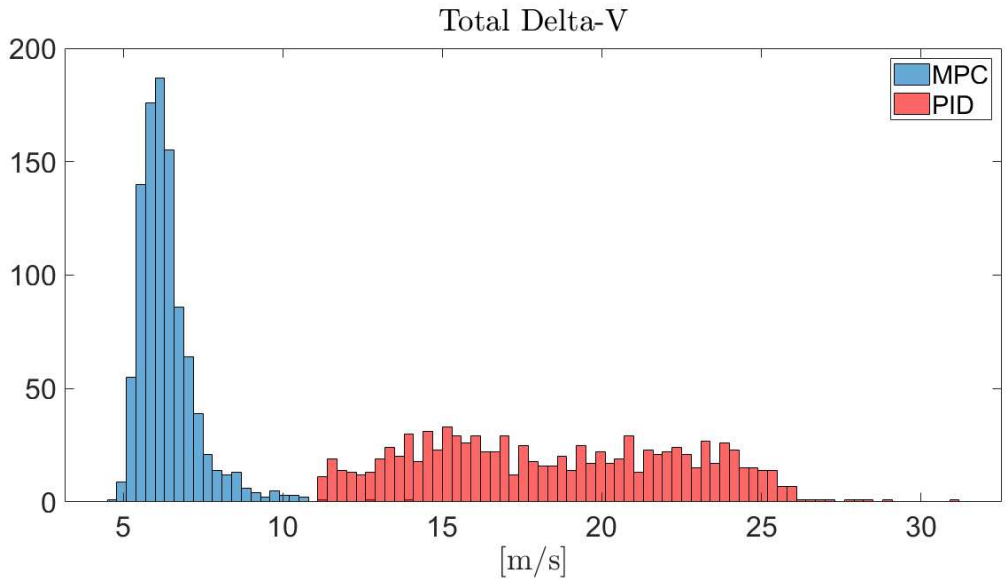


Figure 7.25: Comparison of the total delta-V distributions in the Montecarlo simulations for the MPC (in blue) and the PID (in red).

The PID control is less performing than the MPC and is not suitable for a RDV scenario. Indeed, during the tuning of the PID it has been noticed that the control actions were very high and brought the thrusters to saturation many times during the simulations. Trying to decrease the PID gains, not good results are obtained as well, because a less use of thrusters produced divergence of the simulation.

7.4 Attitude controller Montecarlo results

The attitude controller of Section 6.7 is well designed since all setpoints of the simulations are reached long before the arrival at the target point, and the usage of actuators is always within the manufacturing limits. Figures 7.26, 7.27 and 7.28 show the performances of the attitude controller and the torques applied.

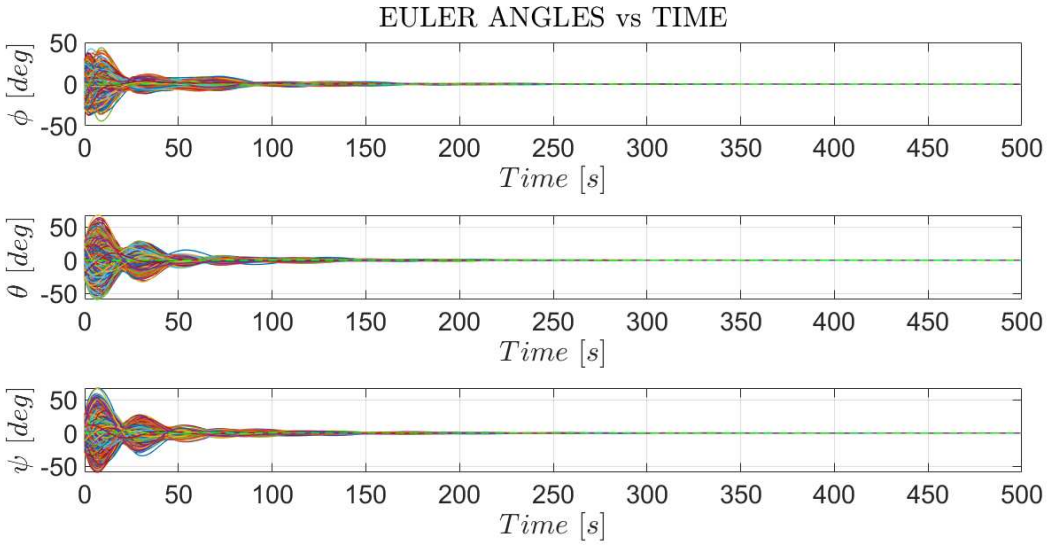


Figure 7.26: Roll, pitch, and yaw of the chaser.

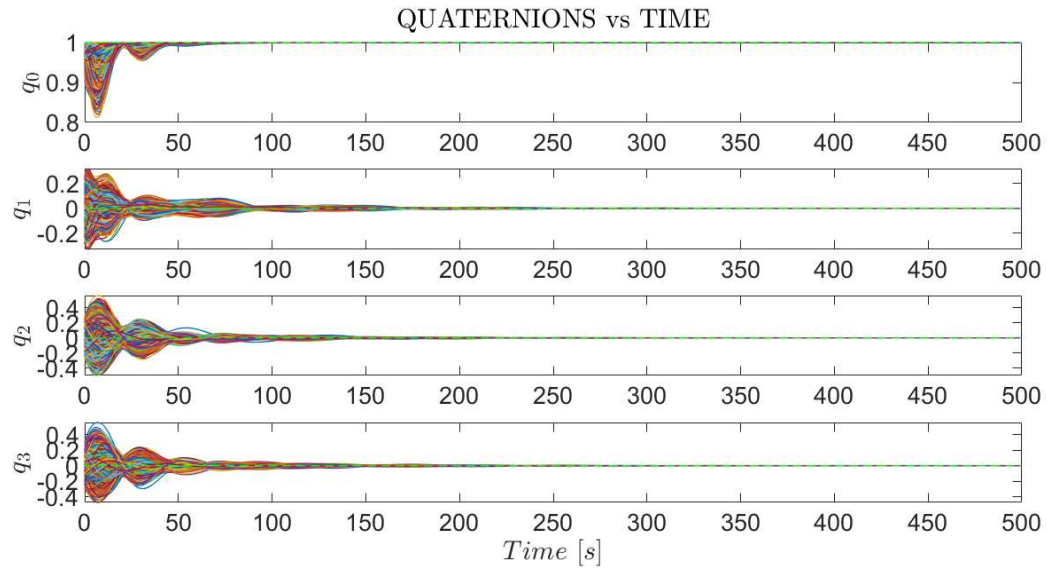


Figure 7.27: Quaternions components of the chaser's attitude.

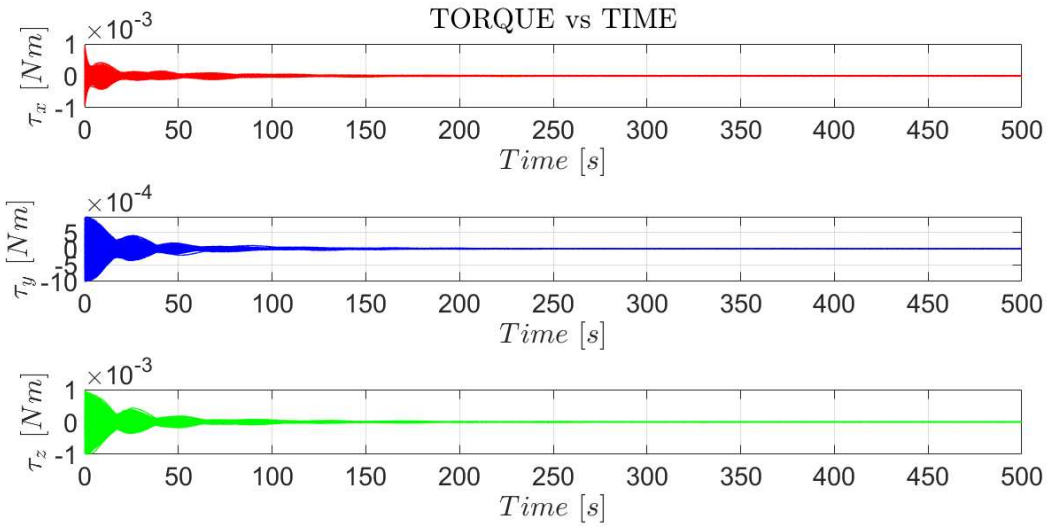


Figure 7.28: Torque generated by the reaction wheels on the spacecraft.

Quantitatively, in Table 7.7 and Figure 7.29 the angular misalignemnts and rate errors with respect to the docking constraints are shown.

Table 7.7: Attitude misalignments and rate errors at docking.

Angular misalignment [deg]

	Avg.	3σ	Min.	Max.	Limit
Roll	4.68e-4	1.61e-1	-1.67e-1	1.81e-1	2.0
Pitch	-1.20e-3	1.23e-1	-1.26e-1	1.38e-1	2.0
Yaw	-4.68e-3	1.53e-1	-1.58e-1	1.52e-1	2.0

Angular rate error [deg/s]

ω_x	4.86e-6	4.09e-2	-4.2e-2	5.1e-2	0.1
ω_y	-2.16e-6	1.84e-2	-3.74e-4	4.14e-4	0.1
ω_z	5.56e-6	1.96e-2	-1.9e-2	2.0e-2	0.1

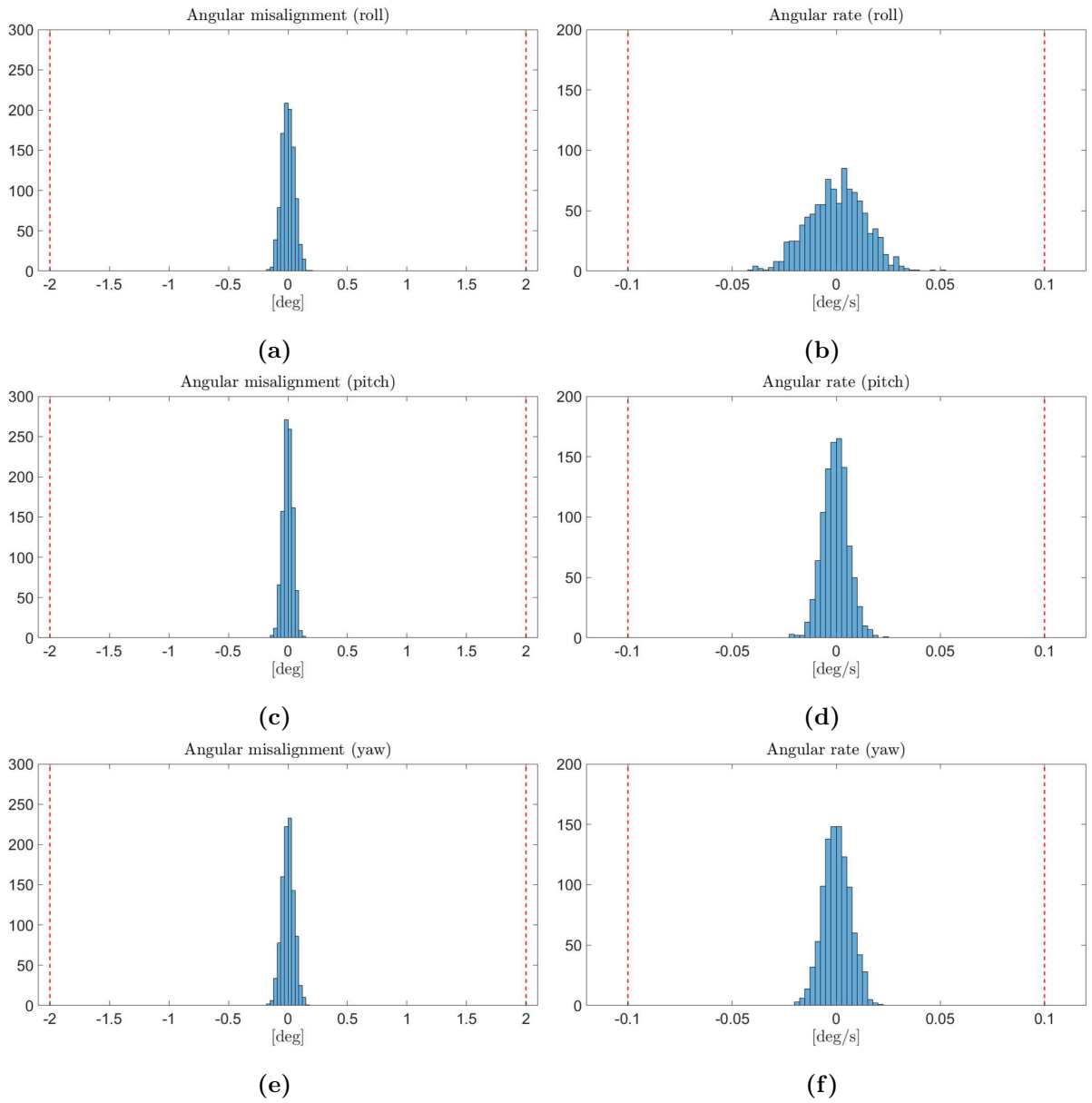


Figure 7.29: Angular misalignments ((a), (c), (e)) and angular rate errors ((b), (d), (f)) at docking. All samples are within the limits.

Table 7.8: Average, minimum, and maximum value of the maximum momenta accumulated by the reaction wheels.

Reaction wheels max. momentum [mNm s]

	Avg.	Min.	Max.	Limit
x	3.153	0.098	7.088	0.02
y	4.311	0.178	11.117	0.02
z	4.243	0.242	10.985	0.02

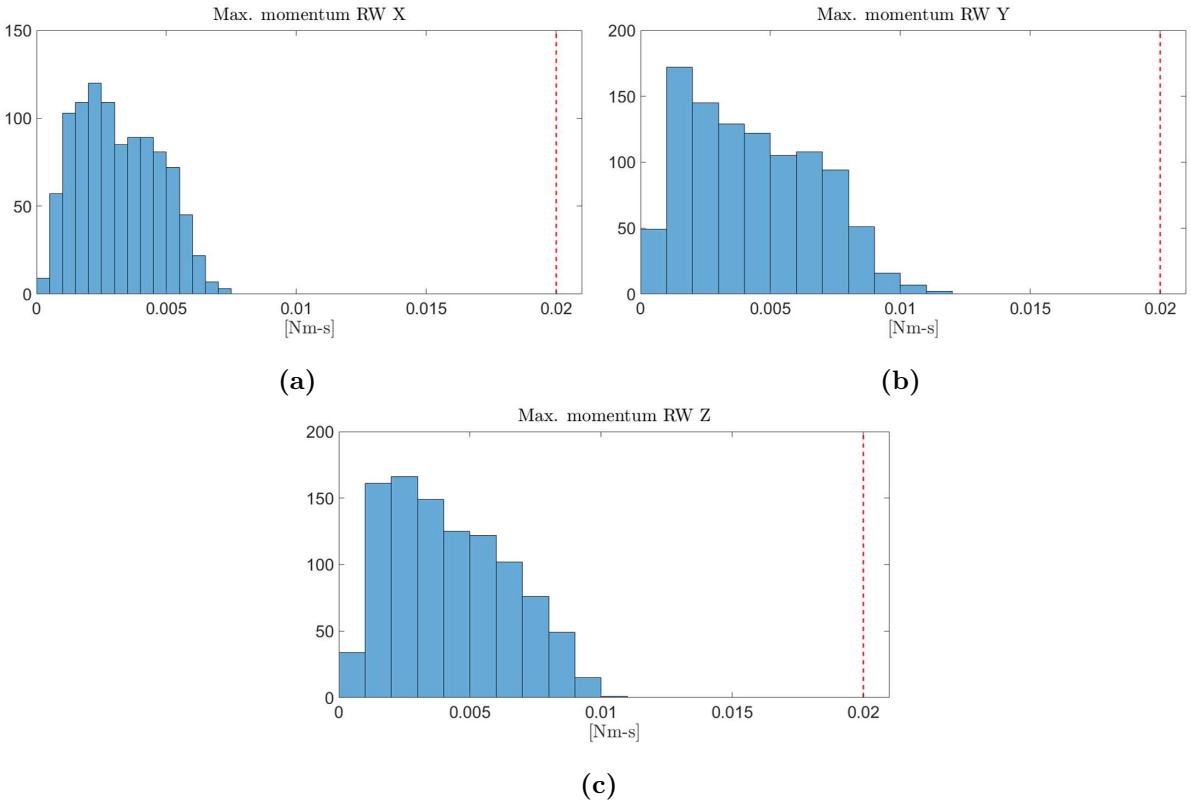


Figure 7.30: Maximum momenta accumulated by the reaction wheels. (a) X-axis; (b) Y-axis; (c) Z-axis. All samples are within the limit for this kind of RW.

Chapter 8

Conclusions

The aim of this thesis is to show the usage and the performances of Model Predictive Control strategies in carrying out autonomous maneuvers of rendezvous and docking between a CubeSat and an orbiting station in circular orbit along V-bar, emphasizing also its advantages and better performances with respect to a classical PID controller. Model Predictive Control is a modern control strategy that is gaining popularity thanks to its advantages over classical controllers for its predictive nature and optimal control. During the work, several numerical simulations have been performed to validate the proposed MPC controller.

In the first part of the work, the mathematical models of the relative dynamics between the chaser and target spacecraft have been investigated. The translational dynamics model, in particular, is used by the MPC as predictive model of the system. After a thorough understanding of the operating principle of the two types of controller, they have been implemented in the software.

The software simulator has been developed in Matlab to emulate the 6-degree-of-freedom relative motion between the two spacecraft, taking into account some external perturbations, such as differential drag and J2, measurement noise and modeling errors, and with the possibility to select the kind of controller to be employed. Some preliminary tests have been carried out to tune the controllers efficiently in terms of trajectory, fuel consumption (delta-Vs), and successful docking.

After a proper tuning of both MPC and PID parameters, a 1000-run Montecarlo sim-

ulation for both controllers has been carried out. The analysis of the results showed that the MPC is more performing than the PID controller in the scenario investigated. Regarding the approach trajectory, the MPC results guaranteed good stability and robustness, whereas the PID control proved not to be suitable for this scenario, since in various Montecarlo runs it was unable to fulfill the approach trajectory requirements. Moreover, the MPC control could well respect the lateral velocity limits of 2 cm/s especially in the initial approach, unlike the PID control. In terms of delta-Vs, the predictive and optimal control nature of the MPC proved to require less actuation of thrusters, with an average use of 6.37 m/s against 18.45 m/s required for the PID control. As for the docking arrival, all MPC simulations were capable of reaching the target within the constraint of 1 cm for lateral misalignment, unlike the PID simulations, in which the docking requirements were not always satisfied.

The work carried out in this thesis has proven that MPC strategies are a valid option to perform rendezvous and docking satisfying all the constraints imposed by the approach corridor for safety reasons and those at the docking point for the docking mechanism to properly work. A possible future development on this topic could be to embed an MPC controller on the on-board computer of a ground simulator in laboratory and perform testing of docking maneuvers.

Bibliography

[1] W. Fehse, *Automated rendezvous and docking of spacecraft*, vol. 16. Cambridge University Press, 2003.

[2] R. Duch, "Phases and manoeuvres involved in orbital rendezvous missions," *International journal for research and development in technology*, vol. 6, no. 2, pp. 265–267, 2016.

[3] 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*, vol. 22, no. 12, pp. 1398–1427, 2012.

[4] K. Yamanaka and F. Ankersen, "New state transition matrix for relative motion on an arbitrary elliptical orbit," *Journal of guidance, control, and dynamics*, vol. 25, no. 1, pp. 60–66, 2002.

[5] R. E. Sherrill, A. J. Sinclair, S. Sinha, and T. A. Lovell, "Time-varying transformations for hill–clohessy–wiltshire solutions in elliptic orbits," *Celestial Mechanics and Dynamical Astronomy*, vol. 119, no. 1, pp. 55–73, 2014.

[6] H. Curtis, *Orbital mechanics for engineering students*. Butterworth-Heinemann, 2013.

[7] J. M. Maciejowski, *Predictive control: with constraints*. Pearson education, 2002.

[8] A. Valmorbida, "Development and testing of model predictive control strategies for spacecraft formation flying," 2014.

[9] M. Behrendt. `https://en.wikipedia.org/wiki/Model_predictive_control#/media/File:MPC_scheme_basic.svg`.

[10] MathWorks, "Kalman filtering." `http://mathworks.com/help/control/ug/kalman-filtering.html`.

[11] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," *Automatica*, vol. 38, no. 1, pp. 3–20, 2002.

[12] D. K. Anand, *Introduction to control systems*, vol. 8. Elsevier, 2013.

[13] D. Rowen, B. Hardy, C. Coffman, D. Hinkley, R. Welle, and S. Janson, "The nasa optical communications and sensor demonstration program: Proximity operations," 2018.

[14] J. Bowen, M. Villa, and A. Williams, "Cubesat based rendezvous, proximity operations, and docking in the cpod mission," 2015.

[15] W. Storck, O. Billett, M. Jambusaria, A. Sadhwani, P. Jammes, and J. Cutler, "A survey of micropropulsion for small satellites," 2006.

[16] S. Mauthe, F. Pranajaya, and R. Zee, "The design and test of a compact propulsion system for canx nanosatellite formation flying," 2005.

[17] D. Pinard, S. Reynaud, P. Delpy, and S. E. Strandmoe, "Accurate and autonomous navigation for the atv," *Aerospace Science and Technology*, vol. 11, no. 6, pp. 490–498, 2007.

[18] C. Pirat, F. Ankersen, R. Walker, and V. Gass, "Vision based navigation for autonomous cooperative docking of cubesats," *Acta Astronautica*, vol. 146, pp. 418–434, 2018.

[19] V. U. Nwankwo and S. K. Chakrabarti, "Theoretical model of drag force impact on a model international space station satellite due to solar activity," *TRANSACTIONS OF THE JAPAN SOCIETY FOR AERONAUTICAL AND SPACE SCIENCES, AEROSPACE TECHNOLOGY JAPAN*, vol. 12, pp. 47–53, 2014.

[20] D. L. Miller, *Development of resource-constrained sensors and actuators for in-space satellite docking and servicing.* PhD thesis, Massachusetts Institute of Technology, 2015.

[21] J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model predictive control: theory, computation, and design*, vol. 2. Nob Hill Publishing Madison, 2017.

[22] S. 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*, vol. 22, no. 12, pp. 1398–1427, 2012.

[23] T. Reid and A. K. Misra, "Formation flight of satellites in the presence of atmospheric drag," *Journal of Aerospace Engineering*, vol. 3, no. 1, p. 64, 2011.

[24] U. S. Atmosphere, *US standard atmosphere*. National Oceanic and Atmospheric Administration, 1976.

[25] J. Gonnaud and V. Pascal, "Atv guidance, navigation and control for rendezvous with iss," in *Spacecraft Guidance, Navigation and Control Systems*, vol. 425, p. 501, 2000.

[26] E. De Pasquale, "Atv jules verne: in-flight demonstration of new rdv technology,"