


Politecnico di Torino

**Politecnico di Torino**

Master's Degree Course in Aerospace Engineering

# Final Approach Simulation and Strategies for Small Satellites involved in Promixity Operations

Supervisor:
**Prof. Fabrizio Stesina**

Co-Supervisor:
**Luca Niero**

Candidate:
**Alessandro Della Libera**
mat. 343413

Academic Year 2025/2026

# Abstract

The advancement in modern technology has transformed the dynamics within the space environment, making the so called *New Space Economy* an absolute protagonist in the international scenario. Particularly, the CubeSat standard has gained more and more popularity due to its versatility which makes it a very competitive format, together with the growing field of applications that they can realize: indeed, this type of spacecraft is capable of obtaining a high performance over cost ratio, enhancing the interest in space research.

The following work focuses on the mission Space Rider Observer Cube (SROC), promoted by the European Space Agency (ESA) and developed by a consortium composed of relevant actors such as Tyvak International, responsible for the satellite design, AIT and operations, Università degli Studi di Padova and Stellar Project, whose role is to test the docking system, Politecnico di Milano, supporting GNC verification and validation, and Politecnico di Torino, dedicated to the Mission Analysis field. This project represents an outstanding example of an academic collaboration which reaches a practical outcome, placing Italian athenaeums in the spotlight of the European horizon. The objective of the mission is to demonstrate the successful execution of rendezvous and docking operations with CubeSat technologies. Although initially the mission architecture involved the Space Rider platform as one of the two players, further risk analysis brought the necessity of developing a second twin CubeSat, namely SROC-B, thus requiring a complete redefinition of mission strategy.

More in detail, the thesis analyzes two different aspects regarding the mission: the Final Approach and the design of Collision Avoidance Manoeuvre (CAM) strategies. The work investigates more in detail the results proposed by the candidate Annamaria Sortino in her thesis *Development of guidance and control strategies for a small satellite involved in proximity operations* presented in December 2025 at Politecnico di Torino.
Thus, the first part explores different scenarios for the Final Approach, with the variation of the initial conditions starting from the latest Hold Point before entering the Approach Cone. The results are then validated using a statistical approach with MonteCarlo analysis. The second part focuses on the development of a proper Fault Detection, Isolation and Recovery (FDIR) strategy for the execution of an emergency CAM. Starting from an analytical approach about the selection of the type of manoeuvre, the final aim is to define a particular sequence of actions in order to gain

control over the Chaser satellite to return to a nominal condition when the emergency state is concluded.

The work was developed using a MATLAB/Simulink simulator which implements the Guidance Navigation and Control (GNC) functions. The accuracy of the final approach phase is guaranteed using a combined implementation of Port-to-Port and Hill-Clohessy-Wiltshire equations, while the controller adopted is a robust Non-linear Model Predictive Controller (NMPC). This thesis aims at representing a solid basis for future development of FDIR strategies, in order to satisfy the imposed safety requirements and, at the same time, introduce a critical analysis of recovery strategies to allow the mission to continue with its purpose.

# Contents


**1 Introduction** **1**

**2 Mission Description** **3**
2.1 State of the Art . . . 3
2.2 Characteristics of RVD missions . . . 4
2.3 SROC Mission . . . 8
2.3.1 Concept of Operations . . . 8
2.3.2 Spacecraft System Design and Docking System . . . 9
2.3.3 Mission Requirements . . . 10
2.4 Close range rendezvous operations - Final Approach to Contact . . . 12
2.5 Trajectory Safety . . . 13
2.5.1 Design rules . . . 13
2.5.2 Causes of violation of the approach corridor . . . 15
2.5.3 Countermeasures against violation of the approach corridor . . . 17

**3 Model Implementation** **18**
3.1 Dynamics & Kinematics . . . 18
3.2 Guidance & Control . . . 25
3.2.1 Sliding Mode Control . . . 27
3.2.2 Non-linear Model Predictive Control . . . 28
3.3 Navigation . . . 30
3.4 Environmental Disturbances . . . 31

**4 Final Approach** **32**
4.1 Nominal profile analysis . . . 32
4.1.1 First scenario . . . 33
4.1.2 Second scenario . . . 36
4.1.3 Third scenario . . . 36
4.1.4 Fourth scenario . . . 36
4.1.5 Fifth scenario . . . 43
4.1.6 Sixth scenario . . . 43
4.1.7 Final Comparison . . . 48
4.2 Monte Carlo Analysis . . . 49

**5 CAM: Strategies and Implementation** **53**
5.1 State of the Art . . . 53
5.1.1 General FDIR Architectures and Strategies . . . 53
5.1.2 CAM Planning Strategies . . . 58
5.2 Methodologies and Results . . . 61
5.2.1 First Iteration . . . 62
5.2.2 Second Iteration . . . 67
5.2.3 Third Iteration . . . 78
5.2.4 Final Comparison . . . 83
5.3 Monte Carlo Robustness Analysis . . . 85
5.3.1 Study on Location of the Way Point . . . 88

**6 Conclusions** **90**

**A Mathematical Formulation of Disturbances** **94**
A.1 Drag due to residual atmosphere . . . 94
A.2 Disturbances due to geopotential anomaly . . . 94
A.3 Solar pressure . . . 95
A.4 Effects of thruster plumes . . . 96
A.5 Navigation errors . . . 96
A.6 Thrust errors deviations . . . 97

**B Reference Frames** **98**
B.1 Earth-centred equatorial frame $F_{eq}$ . . . 98
B.2 Orbital plane frame $F_{op}$ . . . 98
B.3 Spacecraft local orbital frame $F_{lo}$ . . . 99

**C Simulink Code** **101**

# List of Tables


2.1 Logical flow of the nominal Rendezvous and Docking (RVD) mission profile and associated FDIR strategies. The block diagram illustrates the progression through mission phases, highlighting the transition from absolute to relative navigation regimes as the chaser approaches the target. Safety-critical nodes, such as Hold Points and Keep-Out Zones (KOZ), are integrated to depict the triggering conditions for Collision Avoidance manoeuvres (CAM) and breakaway procedures during the mid- and close-range operations. [1] . . . . . 6

2.2 Summary of the primary and secondary mission objectives for the SROC project. The primary goals focus on the in-orbit demonstration of autonomous proximity operations, while the secondary objectives address the on-ground qualification of critical hardware for future Space Rider integration. . . . . 8

2.3 Detailed system architecture of the SROC space segment, comparing the subsystems of the Chaser and Target 12U CubeSats. While sharing a common platform bus, propulsion system, and communication infrastructure, the vehicles feature asymmetric payloads tailored to their respective roles, including the active (DOCKS-A) and passive (DOCKS-B) variants of the docking mechanism and distinct optical sensor suites [2]. . . . . 10

2.4 Requirements defining the constraints and safety regulations governing the Final Approach phase of the rendezvous and docking operation between SROC-A and SROC-B. . . . . 11

2.5 Overview of the main disturbance sources causing trajectory deviations during rendezvous and proximity operations. The phenomena are categorized into environmental orbital disturbances (e.g., atmospheric drag, Earth's oblateness, solar radiation pressure, and plume interactions) and onboard system anomalies (e.g., navigation inaccuracies, thrust vector errors, and hardware failures) [1]. . . . . 15

4.1 Comparison of key-parameters between different scenarios . . . . . 48

4.2 Minimum number of Monte Carlo simulations required to demonstrate, with zero failures observed, a given reliability $R$ at a specified confidence level $C$ . . . . . 50

4.3 Ranges of the parameters dispersed in the Monte Carlo simulations, applied around their nominal values to account for uncertainties in the initial state, spacecraft mass, actuator performance and aerodynamic drag. . . . . 50

5.1 Comparison of hierarchical FDIR frameworks across different missions, highlighting the detection approach and the distinctive strategy adopted by each. . . . . 59
5.2 Summary comparison of the CAM strategies developed across the three iterations, including internal sub-cases (variations on the same strategy). . . . . 83
5.3 Ranges of the parameters dispersed in the Monte Carlo simulations, applied around their nominal values to account for uncertainties in the initial state, spacecraft mass, actuator performance and aerodynamic drag, with an extra variability around the location where the emergency state is activated. . . . . 85

# List of Figures

2.1 Examples of Rendezvous, Proximity Operations and Docking (RPOD) in different scale scenarios. **(a)** Rendering of the two 3U CubeSats developed by Tyvak/Terran Orbital, showing docking mechanisms aligned along the approach axis. The mission represents the first attempt to demonstrate autonomous RPOD between identical CubeSat platforms. **(b)** Uncrewed Crew Dragon on approach to the ISS Harmony module. The vehicle's forward docking mechanism is exposed in preparation for the automated docking sequence. Credit: NASA (Image ID: iss058e027464). . . . . 5

2.2 Concept of Operations (ConOps) of the SROC mission scenario. The sequence illustrates the progression of the two 12U CubeSats through the main mission phases: Launch and Early Operations Phase (LEOP), Proximity Operations Phase (POP), Docking Phase (DKP), Opportunity Extended Phase (OEP), and End of Mission Phase (EMP). The diagram highlights critical operations such as the initial deployment and commissioning, relative navigation during rendezvous, target observation, and the final mating and un-mating sequences prior to disposal [2]. . . . . 9

2.3 Schematic representation of V-bar final approach trajectory types during rendezvous operations. Profile (a) depicts a quasi-straight line, or hopping, trajectory approaching from a +V-bar hold point, which is realized by a multitude of small hops. Profile (b) illustrates a continuous, Closed-Loop controlled straight line trajectory approaching from a -V-bar hold point [1]. . . . . 12

2.4 Schematic illustration of the acquisition of the instantaneous docking axis during the final approach phase. The actual docking axis frequently deviates from the nominal direction due to factors such as the target vehicle's attitude bias, attitude control motions, and structural bending. Consequently, the chaser vehicle's approach trajectory must be actively controlled to acquire and follow this dynamic instantaneous docking axis within the designated approach corridor [1]. . . . . 13

2.5 Free drift trajectories of a chaser relative to a target vehicle, illustrating the effects of differential atmospheric drag over three orbital revolutions. The trajectories are plotted in the target's local orbital frame, centered in its centre of mass. The three distinct curves correspond to varying ballistic coefficient ratios between the chaser and the target ($CB_c/CB_t = 1.5, 3$, and 6) [1]. . . . . 14

3.1 Vector diagram detailing the relative navigation geometry. The illustration defines the primary reference frames and key position vectors required to formulate the relative state between two vehicles. Depicted are the Inertial frame (I), the Target frame (T), and the Chaser frame (C) [5]. . . . . 21
3.2 Block diagram illustrating the architecture of an automatic onboard rendezvous (RV) control system. The schematic details the hierarchical interaction between remote operator monitoring (via Telecommands and Telemetry), the core autonomous onboard systems (FDIR, MVM, and GNC), and the physical spacecraft plant (the chaser's dynamics and kinematics) [1]. . . . . 25
3.3 Block diagram of a Closed-Loop spacecraft Guidance, Navigation, and Control (GNC) system. It illustrates the continuous data flow between four main components: the sensor suite, the core GNC processing functions, the hardware actuators, and the physical spacecraft dynamics and environment [1]. . . . . 26
3.4 Functional block diagram of a Closed-Loop spacecraft Guidance, Navigation, and Control (GNC) architecture. The schematic details the signal flow and interactions between guidance generation, measurement and navigation functions, controller operations, and spacecraft body dynamics, including provisions for feed-forward commands and dynamic disturbance rejection [1]. . . . . 27
3.5 Two-dimensional phase plane portrait of a Sliding Mode Controller. The diagram highlights the transition from the initial reaching phase to the sliding phase along the surface $s = 0$, eventually converging to the target state $x(t_1)$ under discontinuous control efforts (chattering) [13]. . . . . 28
3.6 Computational flow of a single discrete-time Kalman filter iteration. This diagram illustrates the recursive algorithmic structure for one time step $(k + 1)$. System inputs and measurement data are processed through prediction (state and covariance propagation) and correction (measurement update via Kalman gain $K_{k+1}$) phases to calculate the estimated state vector $\hat{x}_{k+1}$ and the updated error covariance matrix $P_{k+1}$. The updated values are then fed back for use in the subsequent iteration [1]. . . . . 30

4.1 *Nominal Velocity vs Position: First Scenario* . . . . . 34
4.2 *Nominal Velocity vs Time: First scenario* . . . . . 34

4.3 *Thrusters control force: First Scenario* . . . 35
4.4 *Relative velocity - nominal conditions: First Scenario* . . . 35
4.5 *Nominal Velocity vs Position: Second Scenario* . . . 37
4.6 *Nominal Velocity vs Time: Second Scenario* . . . 37
4.7 *Thrusters control force: Second Scenario* . . . 38
4.8 *Relative velocity - nominal conditions: Second Scenario* . . . 38
4.9 *Nominal Velocity vs Position: Third Scenario* . . . 39
4.10 *Nominal Velocity vs Time: Third Scenario* . . . 39
4.11 *Thrusters control force: Third Scenario* . . . 40
4.12 *Relative velocity - nominal conditions: Third Scenario* . . . 40
4.13 *Nominal Velocity vs Position: Fourth Scenario* . . . 41
4.14 *Nominal Velocity vs Time: Fourth Scenario* . . . 41
4.15 *Thrusters control force: Fourth Scenario* . . . 42
4.16 *Relative velocity - nominal conditions: Fourth Scenario* . . . 42
4.17 *Nominal Velocity vs Position: Fifth Scenario* . . . 44
4.18 *Nominal Velocity vs Time: Fifth Scenario* . . . 44
4.19 *Thrusters control force: Fifth Scenario* . . . 45
4.20 *Relative velocity - nominal conditions: Fifth Scenario* . . . 45
4.21 *Nominal Velocity vs Position: Sixth Scenario* . . . 46
4.22 *Nominal Velocity vs Time: Sixth Scenario* . . . 46
4.23 *Thrusters control force: Sixth Scenario* . . . 47
4.24 *Relative velocity - nominal conditions: Sixth Scenario* . . . 47
4.25 Standard normal (Gaussian) distribution showing the fraction of the population contained within 1, 2, and 3 standard deviations ($\sigma$) from the mean ($\mu$): approximately 68.0%, 95.0%, and 99.73%, respectively. . 49
4.26 *Nominal Velocity vs Position: First Scenario* . . . 50
4.27 Nominal Monte Carlo analysis: velocity and position profiles. Left: time history of the relative velocity components for all Monte Carlo runs during the final approach phase. Right: corresponding relative position trajectories in the target frame, shown together with the approach cone . . . 51
4.28 Final position dispersion at mating conditions. Scatter plot of the chaser's final relative position at the instant of mechanical mating, for all Monte Carlo samples, compared with a predefined region of 1.2 cm diameter. . . . 51

5.1 Conceptual interaction between the Fault Detection and Diagnosis (FDD), Fault Tolerant Control (FTC), and Fault Tolerant Guidance (FTG) functions within the on-board Supervision, Health Management, and Self-Protection framework. FDD constitutes the initial stage of the chain, providing fault information to both FTC and FTG. FTC is invoked first to preserve system stability and achievable performance through the healthy control resources; if the available control authority proves insufficient, the process escalates to FTG, which accommodates the fault by reshaping the mission objectives (e.g., through trajectory replanning). Adapted from Zolghadri [22]. . . . . 55
5.2 Control stack architecture of the VISORS manoeuvre planner, illustrating how each functional layer — Guidance & Prediction, Decision Logic, Control Algorithms, and manoeuvre Buffer — is specialized per mission mode (Standby, Transfer, Science, Escape). A top-level state machine coordinates state estimate querying, manoeuvre emission, mode switching, and telemetry logging across the stack. Each mode pairs a dedicated guidance/prediction step and decision criterion (e.g., station-keeping dead-band violation, Mahalanobis distance to the alignment target, allowed time to escape) with the corresponding control algorithm (Algorithms 1–4), before planned manoeuvres are quantized and buffered to comply with actuation constraints. Adapted from Guffanti et al. [23]. . . . . 57
5.3 Hierarchical structure of the Masat-1 FDIR framework, organized across three levels (Units, Subsystems, Satellite/Mission). For each level, the table reports the typical failure impact, the fault detection techniques employed (ranging from local watchdog-based checks to system-level aggregated alarms), and the corresponding recovery strategies, from local correction up to safe-mode entry and subsystem reconfiguration. Adapted from Latachi et al. (2020) [26]. . . . . 58
5.4 Relative motion trajectories in the V-bar/R-bar plane resulting from impulsive boosts of increasing magnitude, computed from the Clohessy-Wiltshire equations. (Right) Tangential (In-Track) boosts produce a periodic, looping drift pattern along the V-bar direction, with each successive loop displaced further along-track while maintaining a similar radial amplitude. (Left) Radial boosts produce open, teardrop-shaped trajectories whose along-track (V-bar) extent and radial excursion both grow with increasing boost magnitude. The boost is 0.3 m/s in both scenarios. Figure considers increasing initial radial coordinates, $z$ = 20,50,100 m . . . . . 62

5.5 Required $\Delta V$ for a tangential escape manoeuvre as a function of the initial position within the approach cone. The color scale indicates the corresponding time required to exit the approach cone. As the initial position moves closer to the cone boundary, the required $\Delta V$ decreases monotonically, while the exit time increases correspondingly, reflecting the reduced propellant cost, at the expense of a longer escape duration, for manoeuvres initiated closer to the safety boundary. . . . 63
5.6 Total $\Delta V$ required for a combined (Radial + In-Track) escape manoeuvre, as a function of the initial position within the approach cone and the imposed time to exit the cone. As expected, the required $\Delta V$ decreases as the initial position approaches the cone boundary and as the allowed exit time increases, with the overlaid contour lines highlighting combinations of position and exit time that yield equal propellant cost. . . . . 64
5.7 Trade-off between the combined and purely tangential escape strategies discussed above, selected at each initial position. Two constant plateaus are the result of an imposed maximum cost for the manoeuvre (upper one) or a maximum exit time interval . . . . . 65
5.8 Open-Loop propagation of the guidance trajectory over one orbital day (86400 s) in the In-Track/Radial plane, starting from the escape manoeuvre near the origin. (a) Tangential-only manoeuvre: the chaser drifts in the negative V-bar direction, exhibiting a qualitatively similar growth in oscillation amplitude with increasing distance. (b) Combined manoeuvre: the chaser drifts monotonically in the positive V-bar direction, with the radial oscillation amplitude growing steadily as the along-track distance increases. . . . . 66
5.9 Detail of the chaser's trajectory in Open-Loop configuration with a combined thrust, whose complete evolution in 86400 s is shown in Figure (5.8b) . . . . . 67
5.10 Relative trajectories of the chaser in the InTrack-Radial plane for a set of simulation runs in which the CAM is triggered at a random InTrack position within the 6-25 m range. The required impulsive $\Delta V$ is computed directly within the Stateflow logic; the subsequent evolution of each trajectory is then obtained by propagating the post-burn state through the Orbit Propagator Numerical (high precision) block. The resulting spread of curves illustrates how the trajectory shape and its distance from the approach cone boundary vary depending on the InTrack position at which the manoeuvre is triggered. . . . . 69
5.11 Escape trajectory with Closed-Loop strategy for Monte Carlo simulations: zoom on the Approach Cone to show the reduced radial excursion 71
5.12 Escape trajectory with Closed-Loop strategy for Monte Carlo simulations: view on the complete manoeuvre, with transit behind the target and arrival at (50,0) . . . . . 71

5.13 Total impulsive cost as a function of the InTrack position at which the CAM is triggered, for the full set of simulation runs. Two distinct regimes can be observed: for triggering positions between 25 m and 6 m InTrack, the total $\Delta V$ increases nearly linearly as the triggering point moves closer to the target. Below 6 m InTrack, however, the total cost drops and settles onto a nearly constant value across all runs. . . . . 72
5.14 Comparison between escape trajectories for ideal Closed-Loop manoeuvres having a second point of transit behind the target at different coordinates, respectively at (-50,0) for Figure (5.14a) and (-600,0) for Figure (5.14b) . . . . . 73
5.15 Comparison between costs for ideal Closed-Loop trajectories having a second point of transit behind the target at different coordinates, respectively at (-50,0) for Figure (5.14a) and (-600,0 for Figure (5.14b)) . 74
5.16 Decision hierarchy from Hold Point 4, branching from anomaly detection into a nominal path (safe docking) and an emergency path, the latter further split by GNC availability into Closed-Loop and Open-Loop escape strategies. . . . . 75
5.17 Comparison between the chaser's trajectories within a Closed-Loop versus Open-Loop architecture. The discrepancy follows a growing trend initially and reaches a constant value for the last part of the manoeuvre. . . . . 77
5.18 Comparison between the chaser's trajectories within a Closed-Loop versus Open-Loop architecture. As clearly visible, the uncontrolled scenario collides against the Keep-Out Zone, making the whole manoeuvre not suitable for a safe escape. . . . . 78
5.19 Comparison between Open-Loop and Closed-Loop escape trajectories. No transit around the target is imposed for safety reasons: on the contrary, a direct manoeuvre towards point (50,0) is adopted. . . . . 79
5.20 Comparison between Open-Loop and Closed-Loop escape trajectories with a new location for the Way Point. As before, a direct manoeuvre towards point (50,0) is adopted. . . . . 80
5.21 Comparison between state-based and time-based approaches for the same emergency state, with equal initial conditions and impulses. The time-based, more realiable, is also less faithful to the desired trajectory. 82
5.22 Long-term propagation of the chaser's relative trajectories in the InTrack-Radial plane for the full set of Monte Carlo runs, colored according to elapsed time since the start of the simulation. Each closed loop corresponds to one orbital period of the target. The trajectories span nearly 15 orbital revolutions and reach an InTrack separation of about 1800 m by the end of the propagation. . . . . 86
5.23 Close-up view of the trajectories in the vicinity of the target, colored according to elapsed time since the manoeuvre was triggered. The safety requirement is satisfied for all runs that are allowed to continue undisturbed to completion. . . . . 86

5.24 Extreme close-up view of the trajectories near the origin, colored according to elapsed time since the manoeuvre was triggered. Several trajectories are deliberately truncated at the instant they intersect the approach cone boundary (red line), allowing the corresponding In-Track position to be identified with high precision as a candidate point of no return. . . . . 87
5.25 Total impulsive $\Delta V$ cost as a function of the InTrack position at which the CAM is triggered, for the full set of Monte Carlo runs (each color corresponding to a different simulation instance). The cost initially grows as the triggering point moves closer to the target, reaching a maximum in the 5-10 m range, and subsequently decreases for the innermost trigger positions . . . . . 87
5.26 Relative trajectory in the InTrack-Radial plane for a trigger position equal to (8,0) with the ideal Way Point located at (30,0). The chaser approaches along a long CW arc after exiting the approach cone and completing the final transit toward the WP. . . . . 89
5.27 Relative trajectory in the InTrack-Radial plane for a trigger position equal to (8,0) with the ideal WP relocated to (30,-5). The chaser approaches along a long CW arc after exiting the approach cone and completing the final transit toward the WP. . . . . 89

A.1 Variation of Earth's atmospheric density as a function of altitude for different levels of solar activity. The divergence between solar minimum (F10.7 = 75) and solar maximum (F10.7 = 225) conditions expands significantly, spanning multiple orders of magnitude, at higher altitudes. This dynamic variation directly dictates the magnitude of the residual drag force experienced by the chaser and target vehicles [3]. 95

B.1 Earth-centred equatorial frame, with depiction of the three axes direction and the equatorial plane . . . . . 99
B.2 Orbital plane frame, with depiction of the three axes direction, the equatorial plane, the ascending node and the inclination of the orbital plane . . . . . 99
B.3 Spacecraft local orbital frame, with depiction of the three axes direction and the orbital plane and motion . . . . . 100

C.1 Simulink block dedicated to the handling of CAM strategies . . . . . 104

**ATV** Automated Transfer Vehicle

**CAM** Collision Avoidance Manoeuvre

**COTS** Components Off-The-Shelf

**CPOD** CubeSat Proximity Operations Demonstration

**DCM** Direction Cosine Matrices

**DOCKS** DOCKing System

**ECI** Earth-Centered Inertial

**ESA** European Space Agency

**FDD** Fault Detection & Diagnosis

**FDIR** Fault Detection, Isolation and Recovery

**FTC** Fault Tolerant Control

**FTG** Fault Tolerant Guidance

**GEO** Geostationary Earth Orbit

**GNC** Guidance Navigation and Control

**GNSS** Global Navigation Satellite System

**GPS** Global Positioning System

**HCW** Hill-Clohessy-Wiltshire

**HP** Hold Point

**HTV** H-II Transfer Vehicle

**IMU** Inertial Measurement Unit

**ISL** Inter-Satellite Link

**ISS** International Space Station

**KOZ** Keep-Out Zone

**LEO** Low Earth Orbit

**LIDAR** Light Detection and Ranging

**LVLH** Local-Vertical Local-Horizontal

**NEXUS** Numerical EXact Unified Simulator

**NFOV** Narrow Field Of View

**NMPC** Non-linear Model Predictive Controller

**P2P** Port-to-Port

**RAAN** Right Ascension of the Ascending Node

**RIC** Radial-In-Track-Cross-Track

**ROE** Relative Orbit Elements

**RPOD** Rendezvous, Proximity Operations and Docking

**RVD** Rendezvous and Docking

**RW** Reaction Wheels

**SMC** Sliding Mode Control

**SROC** Space Rider Observer Cube

**SSO** Sun-Synchronous Orbit

**TOF** Time Of Flight

**UWFOV** Ultra-Wide Field Of View

**VBN** Visual Based Navigation

**VSCS** Variable Structure Control Systems

**WFOV** Wide Field Of View

# Chapter 1

# Introduction

Rendezvous, proximity operations and docking (RPOD) represent one of the most safety-critical phases of any orbital mission, requiring high-precision relative navigation and control to guarantee a successful and safe approach between chaser and target. While this class of manoeuvres has been extensively demonstrated on large crewed and uncrewed platforms, its application to CubeSat missions introduces additional challenges: limited onboard computational and propulsive resources, reduced sensor redundancy, and consequently a narrower margin for error. As demonstrated by missions such as CubeSat Proximity Operations Demonstration (CPOD), autonomous RPOD between small platforms remains an open engineering problem, particularly as regards the identification of robust FDIR strategies capable of guaranteeing safety in the presence of unexpected anomalies during the final approach. This thesis addresses this gap by focusing on the SROC mission scenario, investigating both the robustness of the nominal approach trajectory and the design of feasible CAM strategies to be adopted in case of an emergency [1].
The first objective of this work is to analyse different scenarios applied to the nominal phase of approach by introducing variations on the initial conditions and the velocity profile adopted and to validate the robustness of the chosen architecture through a Monte Carlo campaign. After that, the discussion will focus on the study and application of feasible safety escape manoeuvres thanks to a deep analysis of the state of the art as regards FDIR strategies, which will serve as a basis to build various applications, with consequent advantages and disadvantages.
This document is composed as follows:

1. Chapter 2 will present the achievements so far reached within the framework of Rendez-Vous and Docking missions, with a general overview of past missions and specifically on CubeSat scenarios. The discussion will then introduce a general approach for this type of manoeuvres to move to the specific mission architecture of SROC. Finally, the final approach phase will be investigated more in detail, being a central topic for this thesis, together with the causes of perturbation that may lead to the failure of the mission;

[1] Note: Claude AI was employed exclusively for proofreading and stylistic adjustments. The intellectual and technical substance of this thesis is strictly the author's original contribution.

2. Chapter 3 will then describe the mathematical methods adopted to realistically describe the evolution of the mission, with particular attention to the so-called Port-To-Port equations, which are particularly useful in describing the final and most critical step of the docking process. The GNC subsystem is then fully described, since it represents the section of the simulator most involved for the development of both nominal and emergency trajectories;

3. Chapter 4 will discuss various velocity profiles adopted for the final approach and compare them, in order to state whether there is an actual advantage in varying the characteristics of the timing of the final deceleration or not. The final part of this chapter is then dedicated to investigating the statistical method adopted for the Monte Carlo campaign and its results;

4. Chapter 5 will explore the wide literature on FDIR strategies adopted for different missions in order to provide a logical and clear rationale, which will be helpful when addressing different escape strategies. These latter will be presented as *iterations* since they actually represent attempts to solve the problem portrayed by the emergency state. As it will be seen, various difficulties will arise while proposing new methods. For this reason, it becomes a real iterative process, where each cycle uncovers a weakness addressed in the next. Finally, a Monte Carlo campaign is proposed to validate one of the solutions previously proposed.

5. Chapter 6 will offer a general overview on the results obtained, thus representing a conclusion to the work, and introduce new perspectives on future works, analysing strengths and weaknesses of the strategies adopted along the whole process.

6. The final Appendices (A, B and C) will briefly analyse the numerical formulation of elements encountered along the discussion, which are fully explained at the end of the document in order to maintain the readability of the main text. These concepts regard the mathematical formulation of orbital perturbations and reference frames and a brief commentary on the code adopted within the Simulink environment for CAMs.

# Chapter 2

# Mission Description

## 2.1 State of the Art

Rendezvous and Docking (RVD) operations comprise a precise sequence of orbital manoeuvres in order to connect an active vehicle, called *chaser*, with a passive agent, namely *target*. The docking phase represents a highly critical part of the mission, due to the characteristic high-level requirements and the strict precision demands for the relative dynamics model, which shall guarantee reliability, robustness and safety.
This kind of manoeuvre allows a large number of critical operations, e.g. in-orbit assembly, reparation and re-supply for the extension of orbital operative life of spacecrafts, space debris capture and transit of crew between stations. Hereby, some missions are listed and briefly described in order to understand what the starting point for the discussion is.

1. **16th March 1966, Gemini 8**: the first unmanned docking manoeuvre executed by Neil Armstrong to connect with the *Agena Target Vehicle*;

2. **30th October 1967, Cosmos 186/188**: the first autonomous docking achieved by the Soviet Union;

3. **7th June 1971, Sojuz 11-Saljut 1**: the first manned docking manoeuvre towards a permanent orbital outpost;

4. **January 1978, Sojuz 26/27-Saljut 6**: the first multiple docking, with two spacecrafts reaching the same target at the same time;

5. **1981, Space Shuttle**: first docking with robotic arm (namely *berthing*) using Canadarm developed by the Canadian Space Agency;

6. **1995, Space Shuttle-Mir**: first docking between the American Space Shuttle and the Russian orbital station Mir, predecessor of the to-be collaboration thanks to which the International Space Station (ISS) will be designed;

7. **1998-2010, ISS**: assembling operations including dozens of RVD with Shuttle, Sojuz, Progress and, later, Automated Transfer Vehicle (ATV), H-II Transfer

Vehicle (HTV) and Dragon/Cygnus. The European ATV will introduce advanced autonomous rendezvous systems based on relative Global Positioning System (GPS) and video telemetry to enhance precision and safety of unmanned operations;

8. **2010-2020**: intervention of private agents, such as SpaceX Dragon in 2012 (rendezvous & berthing with ISS) and Tianzhou 1 in 2017 (first Chinese autonomous refueling towards Tiangong-2).

9. **2010s, CPOD**: as regards the CubeSat context, the reference project is CPOD (*CubeSat Proximity Operations Demonstration*) which aimed at demonstrating rendezvous, proximity and docking operations with two identical 3U CubeSats equipped with a specific relative-navitagion payload integrating differential GPS, intersatellite links and infrared-visible sensors. The entire sequence of manoeuvres could not be completed due to the insurgence of problematics in the GNC system which caused the burnout of propellant before final docking. Nonetheless, this project represents a robust application of RPOD strategies, still valid nowadays.

10. **2010-2020s, other projects**: in parallel with pure docking operations, some other proximity/inspection missions were developed. Some examples are MarCO (2018 - two 6U NASA CubeSats towards Mars), Milani and Juventas (from mission HERA in close proximity to binary system Didymos-Dimorphos). Both represent a very important example of application of FDIR and CAM strategies.

## 2.2 Characteristics of RVD missions

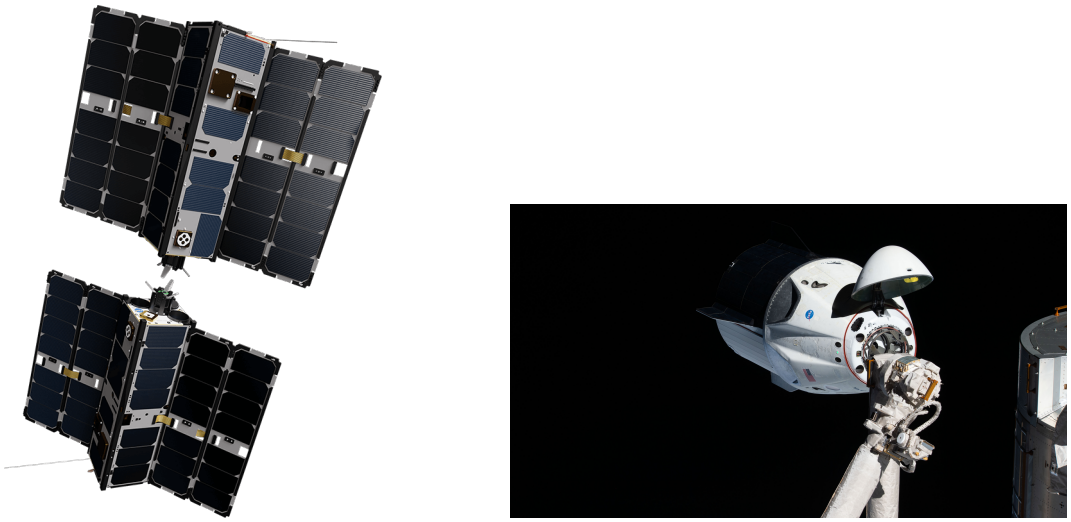
The characteristics of the target vehicle translate into different mission approaches: more in detail, this latter is usually classified based on how much the target can help, or on the contrary hinder, the chaser's relative navigation and control. This classification drives GNC design, sensor selection and FDIR strategy far more than the target's shape or size does. In this perspective, three types of satellite can be distinguished:

- **Cooperative**: the target is equipped with specific instrumentation in order to communicate with the chaser, e.g. GPS/Global Navigation Satellite System (GNSS) transmitters for relative navigation, docking- specific markers or even active control input such as attitude hold. This scenario allows state vector exchange between the two vehicles. An example is represented by ISS and approaching Dragon/Soyuz/Progress/Cygnus. Failure can often be mitigated thanks to this collaboration.

- **Semi-cooperative target**: the target consents approach even without active GNC role, carrying only passive aids. Uncrewed cargos belong to this category. Specifically, the chaser is completely responsible for the success of the mission.

- **Non-cooperative**: the target has no awareness of the chaser and cannot react to it. Its state is estimated purely from sensors aboard the chaser (vision-based navigation or Light Detection and Ranging (LIDAR) point clouds). In addition to this, the target's dynamics/attitude can be controlled or naturally dumped (as for defunct but still stabilized satellites) or unknown and freely evolving, typical of space debris or failed satellites. An example are Active Debris Removal missions such as ESA ClearSpace-1. This is also the worst scenario for FDIR implications since safe modes, CAM and abort logic must rely specifically on the chaser with no cooperative fallback at all. A particular variation of this kind of target are natural bodies, which present additional complications linked to their unknown gravity field and surface topology. In this case, the chaser's autonomy requirement is at its maximum.

Other classifications may be based on geometry/mass properties, dynamics and activeness of the target. The one above presented is usually the most decision-relevant one since it determines whether the chaser's FDIR strategy must rely on its own or not.



(a) NASA's CPOD mission. (b) SpaceX Crew Dragon.

Figure 2.1: Examples of RPOD in different scale scenarios. **(a)** Rendering of the two 3U CubeSats developed by Tyvak/Terran Orbital, showing docking mechanisms aligned along the approach axis. The mission represents the first attempt to demonstrate autonomous RPOD between identical CubeSat platforms. **(b)** Uncrewed Crew Dragon on approach to the ISS Harmony module. The vehicle's forward docking mechanism is exposed in preparation for the automated docking sequence. Credit: NASA (Image ID: iss058e027464).

Generally, the following mission phases can be identified:

Table 2.1: Logical flow of the nominal RVD mission profile and associated FDIR strategies. The block diagram illustrates the progression through mission phases, highlighting the transition from absolute to relative navigation regimes as the chaser approaches the target. Safety-critical nodes, such as Hold Points and Keep-Out Zones (KOZ), are integrated to depict the triggering conditions for Collision Avoidance manoeuvres (CAM) and breakaway procedures during the mid- and close-range operations. [1]

| Phase | Description |
|---|---|
| Launch & Insertion | The chaser is inserted into an orbit that lies on the same plane as the target's one but at a deliberately different altitude to avoid accidental collisions. Tipically, the chaser follows a lower orbit that generates a natural drift, which is taken advantage of in the following phases. After insertion, commissioning manoeuvre are executed, such aeventuales solar panels/antennas deployment, all subsystems checkout (from ADCS to Propulsion and Payload) and sensors calibration. Ground stations (GPS/GNSS) determine the initial orbit in the Earth-Centered Inertial (ECI) frame, giving orbital parameters and state in absolute terms. As regards FDIR strategy, no proximity logic is active yet so only the correct bus fuctioning is checked. |
| Phasing (Far-Range Rendezvous) | **a.** Drift orbit: the chaser remains in its initial orbit until the optimum phase angle with respect to the target is reached. Navigation is still absolute, no relative state estimation is activated yet. Possible correction manoeuvres are planned to refine the upcoming transfer<br>**b.** Transfer: once the separation is minimal, thanks to a specific sequence of Hohmann manoeuvres the chaser reaches a co-elliptic orbit (same shape but different altitude wrt the target's orbit to generate a controlled and predictable approaching rate). Possible failures can be recovered with subsequent dedicated manoeuvres. Collision risk is still negligible due to long distances. |

*Continues on the next page*

*From previous page*

| Phase | Description |
|---|---|
| Closing (Far- and Mid-Range Rendezvous) | The distance varies from 10 km to 1 km ca. The chaser activates the relative navigation, which consists in relative/differential GPS if the target is collaborative or radar determination on the contrary. Some transfer burns are executed in order to align with the nominal approach trajectory, particularly along the V-bar and R-bar direction. The first one coincides with the direction of the velocity vector while the second one points to the center of the Earth. The first Hold Point appears: it is equivalent to a predefined and safe location where the active vehicle intentionally stops or performs station-keeping relative to the passive vehicle before proceeding to the next phase. It allows ground operators or autonomous onboard systems to check sensor data, relative navigation and guidance systems. In the FDIR perspective, the first keep-out zone appears together with first CAM strategies which intervene whenever the approach corridor is accidentally abandoned. |
| Proximity Operations (Close-Range Rendezvous) | Typically defined between 1 km and a few meters from the target. Short-range sensors (LIDAR and visibile/IR cameras) are used and multiple Hold Points are defined, each of them with characteristis safety criteria, such as axes alignment error. Automatic controls are critical: every Hold Point (HP) has its own verification time interval within which every requirement needs to be checked, failing which a CAM intervenes. |
| Final Approach and Contact | During the last 30 meters, high precision is necessary, both in axes alignment and relative velocity. Control strategy occur in a pure Closed-Loop. This scenario takes into consideration a direct docking or berthing. FDIR strategy has only few seconds to determine whether a breakway is required or not. Soft-capture is characterized by first mechanical contact with dumping of residual bumps, while hard-capture concludes with structural blocking and closeness of transfer interfaces, together with verification and confirmation of the *mated* status. |
| Post-Docking Operations | Resources transfer and service operations distinguish this phase, during which the structural integrity is continuously checked. |

*Concludes from previous page*

## 2.3 SROC Mission

The objective of the SROC project is to demonstrate CubeSat capabilities and technologies required for a successful rendezvous and docking with a collaborative target. The mission concept includes a 12U CubeSat which will perform rendezvous, proximity operations, inspection and docking with Space Rider and concludes with a complete re-entry to Earth. Due to risks associated with the Space Rider mission scenario, the in-orbit demonstration involves, at the time being, two collaborative 12U CubeSats in the same scenario, represented by a Sun-Synchronous Orbit (SSO), firstly performing separation during the Commissioning phase and then reaching a docking condition followed by an un-mating sequence. A detailed design is expected for Q1 2027, to be followed by the demonstration phase in 2028. The project involves different actors, such as Tyvak International, Politecnico di Torino, Politecnico di Milano and Università degli Studi di Padova. The statement of the mission is hereby reported, for the sake of clarity [2]:
*"To operate a CubeSat in LEO to demonstrate capabilities in the close-proximity operations domain in a safety-critical context, including rendezvous and docking with another operational spacecraft."*
The mission objectives are listed in the following table:

| | |
|---|---|
| Primary mission objectives | To demonstrate safe CubeSat Rendezvous and Docking |
| | To demonstrate Target inspection |
| | To demonstrate autonomous and robust GNC & FDIR for close-proximity operations |
| Secondary mission objectives | To qualify on ground Multi-Purpose CubeSat Deployers |
| | To qualify on ground the SROC retractable solar panels |

Table 2.2: Summary of the primary and secondary mission objectives for the SROC project. The primary goals focus on the in-orbit demonstration of autonomous proximity operations, while the secondary objectives address the on-ground qualification of critical hardware for future Space Rider integration.

### 2.3.1 Concept of Operations

A new ConOps had to be defined considering the new mission concept involving the two 12U CubeSats. The two spacecrafts are released in a SSO at an altitude of 450-550 km and will autonomously perform all commissioning operations, including solar arrays deployment and attitude stabilization to guarantee the best lighting conditions. After separation and functional verification, the Chaser will reduce its phase angle wrt the target until relative navigation via intersatellite link is established. The proper rendezvous phase begins in this condition, reaching a second HP where Visual Based Navigation (VBN) is allowed via Narrow Field Of View (NFOV) cameras. Subsequently, an Observation phase is conducted to fulfill the inspection requirements.

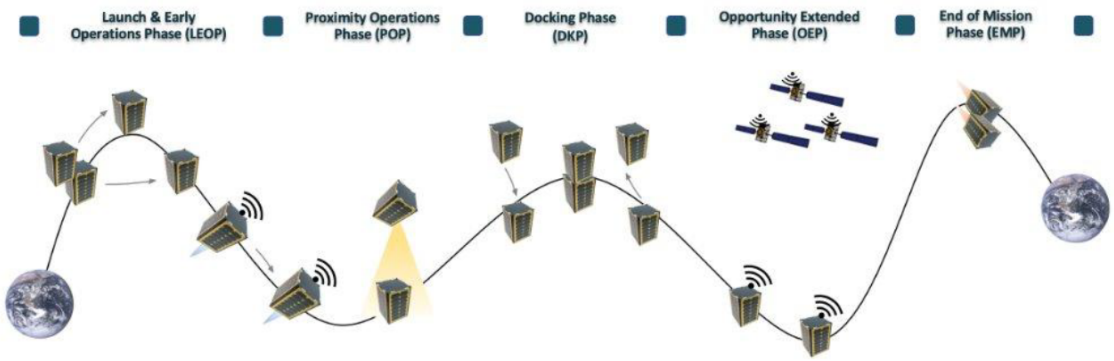



Figure 2.2: Concept of Operations (ConOps) of the SROC mission scenario. The sequence illustrates the progression of the two 12U CubeSats through the main mission phases: Launch and Early Operations Phase (LEOP), Proximity Operations Phase (POP), Docking Phase (DKP), Opportunity Extended Phase (OEP), and End of Mission Phase (EMP). The diagram highlights critical operations such as the initial deployment and commissioning, relative navigation during rendezvous, target observation, and the final mating and un-mating sequences prior to disposal [2].

During this phase, the Chaser maintains a safe and stable relative position while capturing images of the Target. After these images are transmitted to the ground, the Closing phase initiates, further reducing the distance to a final-approach gate. As the Chaser moves closer, the system performs a safe transition from the NFOV to the Wide Field Of View (WFOV) cameras. The Closing phase concludes when the Chaser reaches a final In-Track hold point to verify the accuracy and stability of its visual navigation system. Once the system functionalities are confirmed, the Chaser transitions to port-to-port guidance mode to execute a straight-line manoeuvreing sequence. Possible anomalies are detected by the FDIR logics based on navigation data, triggering a CAM and consequently the abort of the docking sequence. The final approach is completed once the two docking interfaces are in contact, performing a soft docking followed by a hard one, with the considerations previously mentioned.

### 2.3.2 Spacecraft System Design and Docking System

The main components for the two spacecraft are presented in table 2.3. The SROC spacecraft, based on a customized Tyvak 12U Renegade platform, integrates a Flight Computer Module for onboard processing, a sensor suite (star trackers, coarse sensors, Inertial Measurement Unit (IMU), GNSS) for attitude determination and reaction wheels with torque rods for attitude control, complemented by deployable solar arrays and redundant batteries for power, plus dedicated inter-satellite links enabling both Chaser-Target relative navigation via GNSS exchange and FDIR/CAM telemetry monitoring during docking, alongside an Iridium-relayed ground link. For close-proximity guidance, the platform relies on a visual-based navigation suite of three fixed-lens cameras (NFOV, WFOV, Ultra-Wide Field Of View (UWFOV)) built around the flight-proven Tyvak Visible Imager and processed by an NVIDIA Jetson

GPU, detecting spacecraft features and LED patterns continuously from acquisition up to first mechanical contact. The terminal docking phase is handled by DOCKing System (DOCKS), a self-contained, satellite-independent smart docking system split into a Chaser-side interface (DOCKS-A, hosting the processing intelligence) and a Target-side interface (DOCKS-B), which combines a multi-sensor package with a self-aligning centring-cone/drogue mechanism that first achieves electromagnet-assisted soft-docking and then, once alignment is confirmed by acknowledgement fork sensors, activates servomotor-driven claws for a final hard-docking capture.

| | |
|---|---|
| **Chaser CubeSat** | • Spacecraft Bus based on Tyvak Renegade 12U avionics<br>• Tyvak standard trifold deployable solar arrays<br>• Three optical cameras used for navigation and inspection, based on Tyvak heritage<br>• Payload Processing Unit<br>• DOCKS-A, active docking system developed by Università degli Studi di Padova and Stellar Project SRL<br>• A cold-gas 6DoF propulsion system Perseus, developed by T4i and Tyvak International<br>• Space-space Inter-Satellite Link (ISL) module and antenna to support R-GNSS navigation and safe docking sequence<br>• Space-space-ground ISL to extend communication beyond Ground station visibility windows via Iridium constellation |
| **Target CubeSat** | • Spacecraft Bus based on Tyvak Renegade 12U avionics<br>• Tyvak International standard trifold deployable solar arrays<br>• One optical camera<br>• Payload Processing Unit<br>• DOCKS-B, passive docking system developed by UniPD and Stellar Project SRL<br>• A cold-gas 6DoF propulsion system Perseus, developed by T4i and Tyvak International<br>• Space-space ISL module and antenna to support R-GNSS navigation and safe docking sequence<br>• Space-space-ground ISL module and antennas, based on Iridium constellation relay. |

Table 2.3: Detailed system architecture of the SROC space segment, comparing the subsystems of the Chaser and Target 12U CubeSats. While sharing a common platform bus, propulsion system, and communication infrastructure, the vehicles feature asymmetric payloads tailored to their respective roles, including the active (DOCKS-A) and passive (DOCKS-B) variants of the docking mechanism and distinct optical sensor suites [2].

### 2.3.3 Mission Requirements

In order to conduct a proper analysis for the final approach and the execution of emergency manoeuvres, some specific requirements must be considered: they will

guide the discussion of this thesis by providing numerical boundaries to respect. They are hereby listed in Table (2.4).

Table 2.4: Requirements defining the constraints and safety regulations governing the Final Approach phase of the rendezvous and docking operation between SROC-A and SROC-B.

| **Identifier** | **Description** |
|---|---|
| R-FA-010 | The duration of the final approach, from the last hold point to the mating condition, also referred to as *Time-To-Dock*, shall last no more than 30 minutes. |
| R-FA-020 | The kinematic conditions of SROC-A in the last hold point, with respect to SROC-B, shall be:<br>- Relative Position = [0, 25, 0] ± 2m<br>- Relative Velocity = [0, 0, 0] ± 0.02 m/s. |
| R-FA-030 | Relative attitude and Angular Velocity of SROC-A in the last hold point, with respect to SROC-B, shall be:<br>- Relative Attitude less than 1 degree, single axis, $2\sigma$;<br>- Relative Angular Velocity minor than 0.1 deg/s. |
| R-FA-040 | SROC-A maximum approach velocity along the docking axis shall not be greater than 0.1 m/s. |
| R-FA-050 | In the mating point, SROC-A shall have a relative velocity along the docking axis equal to 0.005 m/s, with misalignments on other axes no greater than 10 mm, with respect to SROC-B. |
| R-FA-060 | A Keep-Out-Zone is defined around the target with a radius of 20 m, centered on SROC-B center of mass. |
| R-FA-070 | The Approach Corridor is defined as a 10 degrees cone, symmetrical with respect to the docking port axis and centered on the docking point. |
| R-FA-080 | The Collision Avoidance manoeuvre capability of SROC-A shall put the chaser in a collision-free trajectory that does not cross the predefined Keep-Out-Zone around SROC-B. |
| R-FA-090 | At any point after the entry in the Approach Zone, if the on-orbit safety conditions can no longer be met, SROC-A shall abort the approach and notify the SROC-B Operator immediately while ensuring the capability of reaching a passively safe trajectory |

## 2.4 Close range rendezvous operations - Final Approach to Contact

Since this work focuses on this specific mission phase, a theoretical in-depth analysis is carried out to understand the environmental conditions in which the docking operations take place.
For a correct docking manoeuvre, both dynamics (id est, position and velocity) and relative attitude shall be investigated in order to deliver the chaser docking site into the reception range of the target docking mechanism. A non-negligible axial velocity is required for hard docking, while for soft docking the active latches are triggered by an external input. Two different approaches characterize the trajectory: Closed-Loop controlled straight line manoeuvres and quasi-straight line, composed of many small hops. Figure 2.3 helps compare the two scenarios.

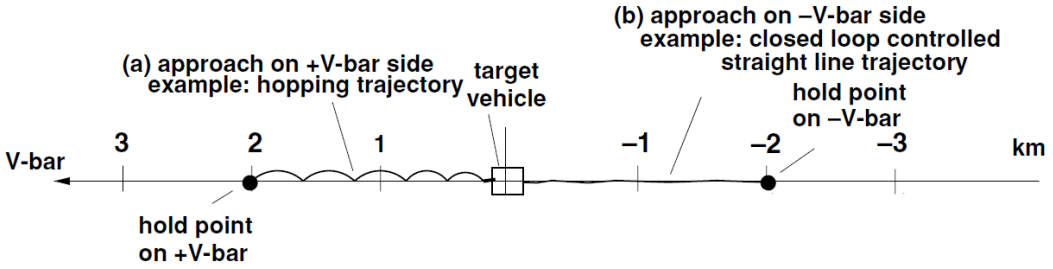



Figure 2.3: Schematic representation of V-bar final approach trajectory types during rendezvous operations. Profile (a) depicts a quasi-straight line, or hopping, trajectory approaching from a +V-bar hold point, which is realized by a multitude of small hops. Profile (b) illustrates a continuous, Closed-Loop controlled straight line trajectory approaching from a -V-bar hold point [1].

The first type of trajectory is preferred for automatic onboard control systems, as for the SROC project. Moreover, both types are suitable for this final phase due to the limited performance of cameras/sensors and because the docking interfaces must meet along their symmetry axes. Very strict requirements are identified as regards accuracy in order to execute a successful docking:

1. Lateral position: few centimetres;
2. Attitude: 1 deg;
3. Axial/Lateral rates: 1 cm/s;
4. Angular rate: 0.1 deg/s.

In addition to this, a very specific GNC condition is introduced only for docking. In fact, the chaser's navigation system shall track the center of the docking port and the direction of the docking axis, whose attitude is subjected to non-predictable variations. It is referred to as *instantaneous docking axis* [fig. 2.4]. The approach corridor observable in fig. 2.4 is defined for safety reasons to identify a cone-shaped region within which the trajectory must develop. Its centre coincides with the docking port

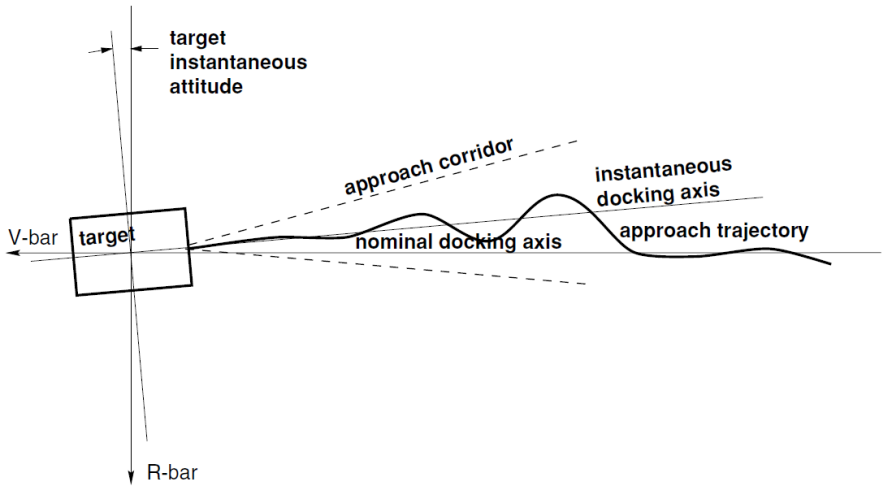



Figure 2.4: Schematic illustration of the acquisition of the instantaneous docking axis during the final approach phase. The actual docking axis frequently deviates from the nominal direction due to factors such as the target vehicle's attitude bias, attitude control motions, and structural bending. Consequently, the chaser vehicle's approach trajectory must be actively controlled to acquire and follow this dynamic instantaneous docking axis within the designated approach corridor [1].

and it has a half cone angle of 10 to 15 degrees (actually, 5 degrees specifically for SROC mission). When its boundaries are crossed, a series of emergency operations, such as retreat or CAM, are executed, depending on the FDIR strategy adopted.
In addition to this, thruster plumes and their effects must be taken into consideration. Particularly, forces exerted on the target vehicle by the plume pressure, the heat load on the structure of the target and the contamination of the target's surface by the combustion products. These causes produce, as effects, disturbances on the target's attitude, overheating of its structure and possible contamination of delicate elements, such as optical lenses. To avoid these problematics, usually the final contact velocity is reached at a specific distance from the target, so that the braking burn toward the target takes place at a sufficiently safe distance.

## 2.5 Trajectory Safety

### 2.5.1 Design rules

As anticipated in the previous chapter, any violation of the corridor boundaries inevitably leads to emergency operations in order to prevent either the loss of the docking opportunity or even the collision between the two spacecrafts. Safety requirements are usually expressed involving the following specific terms:

- *Single failures*, that is, the inability to perform the required function by either a system, subsystem or hardware/software;

- *Major consequence*, which translates within this context into the loss of the RVD mission;
- *Catastrophic failure*, an event resulting in irreversible and unacceptable consequences, such as the loss of crew life or, for this particular case, the destruction of a major space infrastructure, which dictates a strict *fail operational-fail safe* (FO/FS) design architecture.

Close tolerances have already been introduced regarding this scenario. They naturally translate into these design requirements for the approach trajectory strategy [1]:

1. Each trajectory element shall be collision-free for as long as possible, considering possible failures and dispersions;
2. Corrections shall be applied by the onboard system of the chaser in the last moments of approach whenever a non-negligible deviation is detected;
3. The FDIR strategies shall account for redundancies of the subsystems, functions and equipment as a back-up to address fault detection;
4. If all the previous requirements cannot be satisfied, collision avoidance shall be guaranteed, whether passively, by simply sticking to a collision-free trajectory, or actively, with the execution of a properly designed CAM.

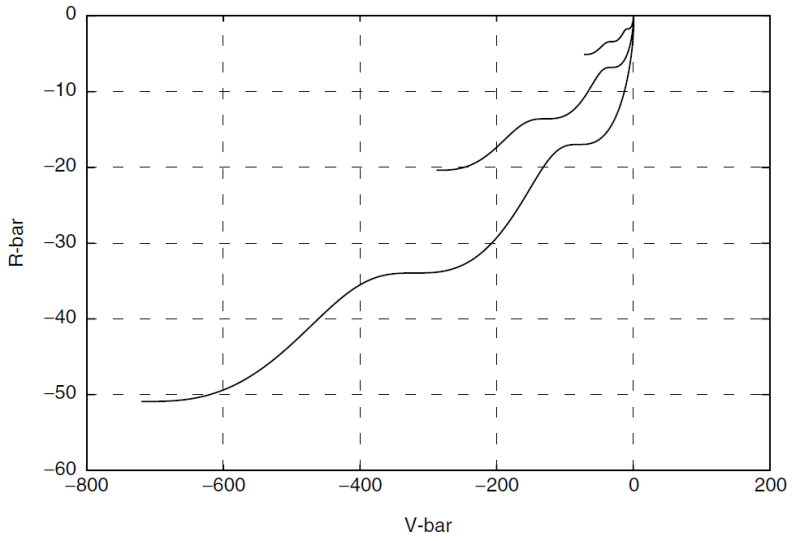



Figure 2.5: Free drift trajectories of a chaser relative to a target vehicle, illustrating the effects of differential atmospheric drag over three orbital revolutions. The trajectories are plotted in the target's local orbital frame, centered in its centre of mass. The three distinct curves correspond to varying ballistic coefficient ratios between the chaser and the target ($CB_c/CB_t$ = 1.5, 3, and 6) [1].

### 2.5.2 Causes of violation of the approach corridor

Deviations from the nominal trajectory have several distinct origins [1]. A comprehensive mathematical derivation quantifying these effects is provided in Appendix A.

Table 2.5: Overview of the main disturbance sources causing trajectory deviations during rendezvous and proximity operations. The phenomena are categorized into environmental orbital disturbances (e.g., atmospheric drag, Earth's oblateness, solar radiation pressure, and plume interactions) and onboard system anomalies (e.g., navigation inaccuracies, thrust vector errors, and hardware failures) [1].

| Cause | Description |
|---|---|
| Orbital disturbances | Forces of various nature, such as atmospheric drag, deviations from the Earth's gravitational potential from a sphere, solar pressure, thruster plume pressure, etc. As regards residual drag force per unit of mass, it is formulated as follows: $$\Delta\gamma_D = -\frac{\rho}{2}V_x^2\frac{1}{C_{Bc}}\left(1-\frac{C_{Bc}}{C_{Bt}}\right)$$ where $V_x$ is the orbital velocity, $C_B$ is the ballistic coefficient for chaser and target. It has been demonstrated that the density variation is mostly influenced by the solar activity cycle of 11 years [3]. Further considerations consent to conclude that when the target has a lower ballistic coefficient w.r.t. the chaser, it will decay faster, causing the dynamic shown in figure 2.5.<br>As regards the variations of the gravitational potential of the Earth, the most significant ones are caused by its second harmonic due to the planet's oblateness. In particular, it affects the positions of the ascending node and the perigee. Their rate of change can be calculated with the following formulas: $$\dot{\Omega}_{J2} = -\frac{3}{2}nJ_2\left(\frac{R_E^2}{a^2(1-e^2)^2}\right)\cos i$$ $$\dot{\omega}_{J2} = \frac{3}{4}nJ_2\left(\frac{R_E^2}{a^2(1-e^2)^2}\right)(5\cos^2 i - 1)$$ where $n$ is the mean orbital motion, $R_E$ is the Earth radius, $a$ is the semi-major axis, $e$ is the eccentricity and $i$ is the inclination. The difference between the alterations between the target's and the chaser's orbits gradually reduces with the approach. |

*Continues on the next page*

*From previous page*

| Cause | Description |
|---|---|
| Orbital disturbances | The solar pressure is about two to three orders of magnitude lower than the atmospheric drag in Low Earth Orbit (LEO). For the sake of clarity, the force produced on the spacecraft per unit of mass is $$\vec{\gamma_{SP}} = -p\frac{A}{m}\vec{u_S}$$ with $p$ being the radiation moment flux, $A$ the cross section of the body, $m$ its mass and $\vec{u_S}$ the Sun-pointing vector. Plume interaction becomes relevant on the short-range, assuming values that are 1-2 orders of magnitude greater than that of air drag. Thus, braking boost shall be avoided. |
| System disturbances | These alterations generally comprise navigation/control/thrust vector errors and thruster failures. Navigation errors are described by position and velocity measurement disalignments: along the In-Track/Cross-track direction, an error in the estimation of the position only reflects on the trajectory in the same direction while radial errors also translate into In-Track perturbations. Similar scenario for velocity measurement errors with the only difference being In-Track velocity errors influencing also the trajectory along the radial axis. Thrust errors are caused by a multitude of scenarios, from the magnitude of the thrust force to its direction. Alterations in the applied force (due to incorrect assumed mass or thrust force level) lead to directly proportional alterations in velocity and trajectory, while thrust duration influences them via more complex relations. Particularly, thruster-open failures represent a critical example of thrust duration errors since they cannot be predicted. Lastly, thruster failures comprise two complementary situations: the inability to close the valves at the end of the operation (*thruster-open failure*) and to open the valve for the execution of the manoeuvre (*thruster-closed failure*). The first ones need to be properly detected as early as possible, not to generate an off-nominal trajectory, while the second ones are easily solved with redundancy or, alternatively, the selection of a collision-free trajectory. |

*Concludes from previous page*

### 2.5.3 Countermeasures against violation of the approach corridor

In this section, only safety strategies covering the far- to close-range trajectory are examined. When in very close proximity, CAM are taken into consideration, as it will be discussed in the next chapter. Whenever adherence to the planned trajectory can be guaranteed, *active measures* are the obvious option, while *passive counteractions* become necessary when the spacecraft system is inevitably compromised.
Active protection is based on a Closed-Loop approach, which monitors the evolution of the trajectory within narrow margins in the so-called *controlled nominal trajectory scheme*. This continuous strategy may be replaced by an intermittent one, where specific impulses are applied when distinct checkpoints are reached. The selection depends on a critical analysis of the mission objectives and tolerance requirements: even though the second one may seem more convenient, it is less precise since only the final position is monitored rather than the whole attitude and state vector, with consequences also on the passive countermeasures. In addition to this, the margins to be checked for the Closed-Loop strategy represent the threshold beyond which an emergency CAM is triggered.

# Chapter 3

# Model Implementation

This chapter is dedicated to the description of the MATLAB/Simulink simulator to produce the final results, regarding both the final approach dynamics and the CAM analysis.
Three main sections characterize the structure of the mathematical model implemented for the mission:

- Dynamics & Kinematics
- Guidance & Control
- Navigation

The last segment will only be briefly mentioned since the actual modifications involve the other two blocks. Further details about the reference frames mentioned along this chapter are discussed in Appendix B.

## 3.1 Dynamics & Kinematics

To describe the relative motion between chaser and target, a local orbital frame $F_{lo}$ centered on the second one is adopted, together with the well-known *Hill equations*:

$$\begin{aligned} \ddot{x} - 2\omega\dot{z} &= \frac{1}{m_c}F_x \\ \ddot{y} + \omega^2 y &= \frac{1}{m_c}F_y \\ \ddot{z} + 2\omega\dot{x} - 3\omega^2 z &= \frac{1}{m_c}F_z \end{aligned} \tag{3.1}$$

where $\omega = \frac{2\pi}{T}$ is the angular frequency of the target orbit and $m_c$ is the chaser's mass. The second term of the equations, also summarily indicated with the term $\gamma$, represents external forces imposed on the system. Whenever the distance between the two spacecraft becomes negligible with respect to the radius of the orbit, Eqs. 3.1

can be linearised, as indeed it has been proposed by Clohessy & Wiltshire in 1960. In addition to this, in the presence of input pulses with constant amplitude, a further derivation of these equations can be found: this is particularly apt to describe motion after thrust manoeuvres, if they are accepted to be defined as impulsive. Thus, a new set of equations is introduced, with position errors that are considered negligible under one kilometer distance [1]

$$\begin{aligned}
x(t) &= \left(\frac{4\dot{x_0}}{\omega} - 6z_0\right)\sin(\omega t) - \frac{2\dot{z_0}}{\omega}\cos(\omega t) + (6\omega z_0 - 3\dot{x_0})t + \left(x_0 + \frac{2\dot{z_0}}{\omega}\right) + ... \\
&\quad + \frac{2}{\omega^2}\gamma_z(\omega t - \sin(\omega t)) + \gamma_x\left(\frac{4}{\omega^2}(1-\cos(\omega t)) - \frac{3}{2}t^2\right) \\
y(t) &= y_0\cos(\omega t) + \frac{\dot{y_0}}{\omega}\sin(\omega t) + \frac{\gamma_y}{\omega^2}(1-\cos(\omega t)) \\
z(t) &= \left(\frac{2\dot{x_0}}{\omega} - 3z_0\right)\cos(\omega t) + \frac{\dot{z_0}}{\omega}\sin(\omega t) + \left(4z_0 - \frac{2\dot{x_0}}{\omega}\right) + ... \\
&\quad + \frac{2}{\omega^2}\gamma_x(\sin(\omega t) - \omega t) + \frac{\gamma_z}{\omega^2}(1-\cos(\omega t))
\end{aligned} \tag{3.2}$$

Nonetheless, for the phase of Final Approach, Hill-Clohessy-Wiltshire (HCW) equations are not accurate enough and must be extended to the so-called *Port-to-Port* equations, which on the contrary take into consideration the relative distance between the two opposite docking ports, together with angular and translational motion of both satellites. Indeed, the target's rotation implies both the chaser's rotation and translation. This complex phenomenon has been deeply analyzed by C. Pirat in her Ph.D. thesis [4] and is here summarized in order to give a complete overview of the complexity of the dynamics [5].
Firstly, the absolute attitude is defined within the local orbital frame $F_{lo}$ (see Appendix B) thanks to the well-known Euler's Equation

$$\dot{\omega}_{lo}^{bI} = I_b^{-1}\left[T_b - T_b^{RW} - \omega_b^{bI} \times \left(I_b\omega_b^{bI} + H_b^{RW}\right)\right] \tag{3.3}$$

where

- $\omega_{lo}^{bI}$ is the body angular velocity with respect to the inertia frame;
- $I_b$ is the spacecraft inertia tensor;
- $T_b$ represents the input torques, as resulting from the action of both actuators and disturbances;
- $T_b^{RW}$ is the torque associated to the Reaction Wheels (RW) on board;
- $H_b^{RW}$ is the reaction wheels' angular momentum.

[1] This set of equations is the same one adopted to describe motion after the execution of a CAM: the hypothesis of impulsive thrust is still valid.

Every term is expressed in the local orbital frame $F_{lo}$. It is useful though to express the angular velocities in the body frame instead of the local orbital one ($F_{lo}$) by using the so-called Direction Cosine Matrices (DCM),

$$R_{b,lo} = R_{b,I} R_{I,lo} \tag{3.4}$$

in order to finally obtain this expression

$$\omega_b^{b,lo} = \omega_b^{b,I} - R_{b,lo}\omega_{lo}^{lo,I} \tag{3.5}$$

where $\omega_{lo}^{lo,I}$ is the angular velocity of the spacecraft with respect to the inertial frame, expressed in the local orbital frame. Namely,

$$\omega_{lo}^{lo,I} = \begin{bmatrix} 0 \\ -\omega_0 \\ 0 \end{bmatrix} \qquad \omega_0 = \sqrt{\frac{\mu_E}{r^3}} \tag{3.6}$$

where $\mu_E$ is the Earth gravitational parameter and $r$ is the radius of the orbit. Excluding the RW contribution, substituting (3.5) in (3.3) leads to

$$\dot{\omega}_b^{b,lo} = I_b^{-1}\left[T_b - \left(\omega_b^{b,lo} + R_{b,lo}\omega_{lo}^{lo,I}\right) \times \left(I_b(\omega_b^{b,lo} + R_{b,lo}\omega_{lo}^{lo,I})\right)\right] \tag{3.7}$$

For the sake of completeness, the relation between attitude variables and angular velocities is now illustrated. Generally, the attitude angles are defined by the symbols $\phi$, $\theta$ and $\psi$ such that the sequence 1-2-3 expressed with the DCM is

$$R(\alpha) = R_3(\psi)R_2(\theta)R_1(\phi) \tag{3.8}$$

This sequence is kinematically described by the following equation

$$\dot{\alpha}^{b,lo} = B(\theta,\psi)\omega_b^{b,lo} \qquad B(\theta,\psi) = \frac{1}{\cos\theta}\begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \cos\theta\sin\psi & \cos\theta\cos\psi & 0 \\ -\sin\theta\cos\psi & \sin\theta\sin\psi & \cos\theta \end{bmatrix} \tag{3.9}$$

The combination of both (3.7) and (3.9) leads to the nonlinear matricial function

$$\begin{bmatrix} \dot{\alpha}^{b,lo} \\ \dot{\omega}_b^{b,lo} \end{bmatrix} = f(\alpha^{b,lo}, \omega_b^{b,lo}, T_b) \tag{3.10}$$

For the target perspective, the assumption is to consider the three parameters null, that is, the target is perfectly aligned to its local orbital frame. The same hypothesis cannot be stated for the chaser, being too optimistic for a realistic scenario.
Firstly, the inertia tensor is expressed with respect to the docking part thanks to Steiner's theorem

$$I_d = R_{d,b}\left(I_b + m\left[r_{b,d}^2\mathbf{I} - r_{b,d}r_{b,d}^T\right]\right)R_{d,b}^T \tag{3.11}$$

so that Eq. (3.7) becomes

$$\dot{\omega}_d^{d,lo} = I_d^{-1}\left[T_d - \left(\omega_d^{d,lo} + R_{d,lo}\omega_{lo}^{lo,I} \times \left(I_d(\omega_d^{d,lo} + R_{d,lo}\omega_{lo}^{lo,I})\right)\right)\right] \tag{3.12}$$

describing the absolute spacecraft dynamics with respect to the docking port. The chaser's relative angular velocity, relative to the target, is

$$\omega_{d_c}^{d_c,d_t} = \omega_{d_c}^{d_c,lo} - R_{d_c,d_t}\omega_{d_t}^{d_t,lo} \tag{3.13}$$

where $d_c$ and $d_t$ respectively refer to the docking ports of the chaser and the target. By deriving the latter expression, the result is

$$\dot{\omega}_{d_c}^{d_c,d_t} = \omega_{d_c}^{\dot{d_c},lo} - R_{d_c,d_t}\omega_{d_t}^{\dot{d_t},lo} + [\omega_{d_c}^{d_c,d_t}\times](R_{d_c,d_t}\omega_{d_t}^{d_t,lo}) \tag{3.14}$$

where the derivatives of the angular velocities can be obtained with (3.12). The absolute attitude of the chaser is conveniently expressed with respect to the target by

$$\begin{aligned}\dot{\omega}_{d_c}^{d_c,lo} = I_{d_c}^{-1}\Bigg(&T_{d_c} - \left(\omega_{d_c}^{d_c,d_t} + R_{d_c,d_t}\omega_{d_t}^{d_t,lo} + R_{d_c,d_t}R_{d_t,lo}\omega_{lo}^{lo,I}\right)\times \\ &\times \left(I_{d_c}(\omega_{d_c}^{d_c,d_t} + R_{d_c,d_t}\omega_{d_t}^{d_t,lo} + R_{d_c,d_t}R_{d_t,lo}\omega_{lo}^{lo,I})\right)\Bigg)\end{aligned} \tag{3.15}$$

This expression can be inserted in (3.14), together with (3.12), to express it only in terms of absolute attitude. As in (3.9), the kinematics has the same form. The relative attitude model can now be written in a nonlinear state-space form as

$$\dot{x}_{att} = \begin{bmatrix}\dot{\alpha}^{d_c,d_t} \\ \dot{\omega}_{d_c}^{d_c,d_t} \\ \dot{\alpha}^{d_t,lo} \\ \dot{\omega}_{d_t}^{d_t,lo}\end{bmatrix} = f_{att}(x_{att}, u_{att}), \qquad x_{att} = \begin{bmatrix}\alpha^{d_c,d_t} \\ \omega_{d_c}^{d_c,d_t} \\ \alpha^{d_t,lo} \\ \omega_{d_t}^{d_t,lo}\end{bmatrix}, \qquad u_{att} = \begin{bmatrix}T_{d_c} \\ T_{d_t}\end{bmatrix} \tag{3.16}$$

Now, the Port-to-Port (P2P) position dynamics, which is a combination of the relative attitude dynamics just obtained with a modified version of Hill's Equations, will be derived.

The position of the docking port of the chaser with respect to the target's one, in the intertial frame, is given by

$$p_I^{d_c,d_t} = r_I^{b_c} + R_{I,b_c}r_{b_c}^{d_c} - r_I^{b_t} - R_{I,b_t}r_{b_t}^{d_t} = p_I^{b_c,b_t} + R_{I,b_c}r_{b_c}^{d_c} - R_{I,b_t}r_{b_t}^{d_t} \tag{3.17}$$

In the target docking frame, (3.17) becomes

$$p_{d_t}^{d_c,d_t} = \underbrace{R_{d_t,lo}R_{lo,I}p_I^{b_c,b_t}}_{p_{d_t}^{b_c,b_t}} + R_{d_t,d_c}\underbrace{R_{d_c,b_c}r_{b_c}^{d_c}}_{r_{d_c}^{d_c}} - \underbrace{R_{d_t,b_t}r_{b_t}^{d_t}}_{r_{d_t}^{d_t}} \tag{3.18}$$

where $R_{d_t,lo}$ is the rotation matrix from the local orbital frame to the target's docking one, $R_{d_t,d_c}$ expresses the rotation from the chaser's to the target's docking frames, $r_{d_c}^{d_c}$

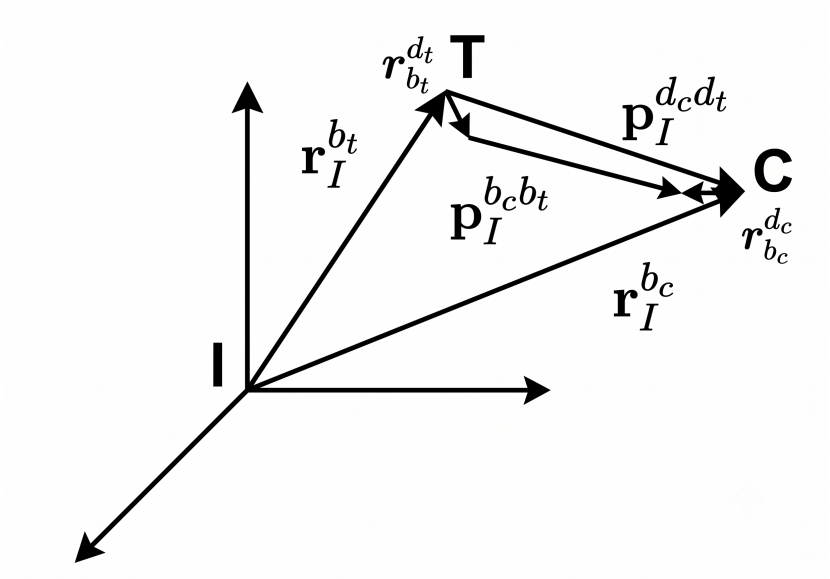



Figure 3.1: Vector diagram detailing the relative navigation geometry. The illustration defines the primary reference frames and key position vectors required to formulate the relative state between two vehicles. Depicted are the Inertial frame (I), the Target frame (T), and the Chaser frame (C) [5].

and $r_{d_t}^{d_t}$ are constant and $R_{lo,I}$ depends only on the target's orbit. By differentiating (3.18) two times, the following complex expression is obtained

$$\begin{aligned}
\ddot{p}_{d_t}^{d_c,d_t} = &- [\dot{\omega}_{d_t}^{d_t,lo}\times](p_{d_t}^{d_c,d_t} - R_{d_c,d_t}^T r_{d_c}^{d_c} + r_{d_t}^{d_t}) + \\
&- [\omega_{d_t}^{d_t,lo}\times][\omega_{d_t}^{d_t,lo}\times](p_{d_t}^{d_c,d_t} - R_{d_c,d_t}^T r_{d_c}^{d_c} + r_{d_t}^{d_t}) - 2[\omega_{d_t}^{d_t,lo}\times]\dot{p}_{d_t}^{d_c,d_t} + \\
&- [R_{d_t,lo}\omega_{lo}^{lo,I}\times][R_{d_t,lo}\omega_{lo}^{lo,I}\times](s_{d_t}^{d_c,d_t} - R_{d_c,d_t}^T r_{d_c}^{d_c} + r_{d_t}^{d_t}) + \\
&- 2[R_{d_t,lo}\omega_{lo}^{lo,I}\times]\dot{p}_{d_t}^{d_c,d_t} - 2[R_{d_t,lo}\omega_{lo}^{lo,I}\times][\omega_{d_t}^{d_t,lo}\times](p_{d_t}^{d_c,d_t} - R_{d_c,d_t}^T r_{d_c}^{d_c} + r_{d_t}^{d_t}) + \\
&+ 2[R_{d_t,lo}\omega_{lo}^{lo,I} + \omega_{d_t}^{d_t,lo}\times][R_{d_c,d_t}^T\omega_{d_c}^{d_c,d_t}\times]R_{d_c,d_t}^T r_{d_c}^{d_c} + \\
&+ [R_{d_c,d_t}^T\dot{\omega}_{d_c}^{d_c,d_t}\times]R_{d_c,d_t}^T r_{d_c}^{d_c} + 2[R_{d_c,d_t}^T\dot{\omega}_{d_c}^{d_c,d_t}\times][R_{d_c,d_t}^T\dot{\omega}_{d_c}^{d_c,d_t}\times]R_{d_c,d_t}^T r_{d_c}^{d_c} + \\
&+ \mu_E\frac{A_{d_t,lo}r_o^{b_t}}{\|r_o^t\|^3} - \mu_E\frac{A_{d_t,lo}r_o^{b_t} + p_{d_t}^{d_c,d_t} - R_{d_c,d_t}^T r_{d_c}^{d_c} + r_{d_t}^{d_t}}{\|A_{d_t,lo}r_o^{b_t} + p_{d_t}^{d_c,d_t} - R_{d_c,d_t}^T r_{d_c}^{d_c} + rd_t^{d_t}\|^3} + R_{d_t,lo}R_{lo,I}\frac{F_I}{m_c}
\end{aligned} \tag{3.19}$$

where $r_{lo}^t$ is the target's position with respect to the inertial frame but expressed in the local orbital frame and is a constant. Equation (3.19) is none other than the two-body problem in a non-inertial frame, referred to the docking ports.

- The quantity $(p_{d_t}^{d_c,d_t} - R_{d_c,d_t}^T r_{d_c}^{d_c} + r_{d_t}^{d_t})$ represents the relative positions of the centers of mass expressed in the target's docking frame;
- The term $2[R_{d_t,lo}\omega_{lo}^{lo,I}\times]\dot{p}_{d_t}^{d_c,d_t}$ and similar ones derive from Coriolis' acceleration;
- The term $[\omega_{d_t}^{d_t,lo}\times][\omega_{d_t}^{d_t,lo}\times](p_{d_t}^{d_c,d_t} - R_{d_c,d_t}^T r_{d_c}^{d_c} + r_{d_t}^{d_t})$ is the centripetal acceleration;

- The last line of the equation is the inertial gravitational term;
- The quantity $R_{d_t,lo}R_{lo,I}\frac{F_I}{m_c}$ is the chaser's control force in the chaser docking frame and is equivalent to

$$R^T_{d_c,d_t}\frac{F_{d_c}}{m_c}$$

in the target docking frame.

In a nonlinear matricial form, Eq. (3.19) becomes

$$\dot{x_{pos}} = \begin{bmatrix} \dot{p}^{d_c,d_t}_{d_t} \\ \dot{p}^{d_c,d_t}_{d_t} \end{bmatrix} = f_{pos}(x_{pos}, x_{att}, u_{pos}), quad x_{pos} = \begin{bmatrix} p^{d_t,d_c}_{d_t} \\ \dot{p}^{d_t,d_c}_{d_t} \end{bmatrix}, \qquad \left[u_{pos} = F_{d_c}\right] \tag{3.20}$$

The complete P2P dynamics is obtained by uniting (3.16) with (3.20) to obtain

$$\begin{gathered} \dot{x}_{full} = \begin{bmatrix} \dot{\alpha}^{d_c,d_t} \\ \dot{\omega}^{d_c,d_t}_{d_c} \\ \dot{\alpha}^{d_t,lo} \\ \dot{\omega}^{d_t,lo}_{d_t} \\ \dot{p}^{d_c,d_t}_{d_t} \\ \ddot{p}^{d_c,d_t}_{d_t} \end{bmatrix} = \begin{bmatrix} f_{att}(x_{att}, u_{att}) \\ f_{pos}(x_{pos}, x_{att}, u_{pos}) \end{bmatrix} = f_{P2P}(x_{full}, u_{full}) \\ x_{full} = \begin{bmatrix} x_{att} \\ x_{pos} \end{bmatrix} = \begin{bmatrix} \alpha^{d_c,d_t} \\ \omega^{d_c,d_t}_{d_c} \\ \alpha^{d_t,lo} \\ \omega^{d_t,lo}_{d_t} \\ p^{d_c,d_t}_{d_t} \\ \dot{p}^{d_c,d_t}_{d_t} \end{bmatrix}, \quad u_{full} = \begin{bmatrix} u_{att} \\ u_{pos} \end{bmatrix} = \begin{bmatrix} T_{d_c} \\ T_{d_t} \\ F_{d_c} \end{bmatrix} \end{gathered} \tag{3.21}$$

Thanks to some assumptions, various simplifications can be made to linearize this system of equations. For example

1. $\alpha^{d_t,lo}$ and $\omega^{d_t,lo}_{d_t}$ are null if the target is aligned with its local orbital frame;
2. If the chaser docking frame is aligned to target docking one, also $\alpha^{d_c,d_t}$ and $\omega^{d_c,d_t}_{d_c}$ become negligible;
3. when docking takes place, $p^{d_c,d_t}_{d_t} = \dot{p}^{d_c,d_t}_{d_t} = \mathbf{0}$;
4. control forces and torques are null, so that $T_{d_c} = T_{d_t} = F_{d_c} = \mathbf{0}$.

The final form of the linearization makes it possible to rewrite the model in the form

$$\begin{aligned} \dot{x}_{full} &= A_{full} x_{full} + B_{full} u_{full} \\ y_{full} &= C_{full} x_{full} + D_{full} u_{full} \end{aligned} \tag{3.22}$$

with

$$\begin{aligned} A_{full} &= \begin{bmatrix} 0 & A_{12} & 0 & 0 & 0 & 0 \\ A_{21} & A_{22} & A_{23} & A_{24} & 0 & 0 \\ 0 & 0 & 0 & A_{34} & 0 & 0 \\ 0 & 0 & A_{43} & A_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & A_{56} \\ A_{61} & A_{62} & A_{63} & A_{64} & A_{65} & A_{66} \end{bmatrix} \\ &\text{with } A_{ij} \in \mathbb{R}^{3\times3} \forall i, j = 1, ..., 6; \\ B_{full} &= \begin{bmatrix} 0 & 0 & 0 \\ B_{21} & B_{22} & 0 \\ 0 & 0 & 0 \\ 0 & B_{42} & 0 \\ 0 & 0 & 0 \\ B_{61} & B_{62} & B_{63} \end{bmatrix} \\ &\text{with } B_{ij} \in \mathbb{R}^{3\times3} \forall i = 1, ..., 6; \quad j = 1, ..., 3; \end{aligned} \tag{3.23}$$

Matrices $C$ and $D$ are fully described in Chapter 3 of [5].
Although the presence of term $T_{d_t}$ in Eq. (3.19) suggests the possibility of controlling both chaser and target simultaneously, their GNC systems shall preferably remain independent for the delicate phase of final approach to avoid technological challenges. The simplified model could be further reduced by removing the target relative attitude, to obtain

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \tag{3.24}$$

where

$$
\begin{aligned}
x &= \begin{bmatrix} \alpha^{d_c,d_t} \\ \omega_{d_c}^{d_c,d_t} \\ p_{d_t}^{d_c,d_t} \\ \dot{p}_{d_t}^{d_c,d_t} \end{bmatrix} \in \mathbb{R}^{12} \\
u &= \begin{bmatrix} T_{d_c} \\ F_{d_c} \end{bmatrix} \in \mathbb{R}^{6} \\
A &= \begin{bmatrix} 0 & A_{12} & 0 & 0 \\ A_{21} & A_{22} & 0 & 0 \\ 0 & 0 & 0 & A_{56} \\ A_{61} & A_{62} & A_{65} & A_{65} \end{bmatrix} \in \mathbb{R}^{12\times 12} \\
B &= \begin{bmatrix} 0 & 0 \\ B_{21} & 0 \\ 0 & 0 \\ B_{61} & B_{63} \end{bmatrix} \in \mathbb{R}^{12\times 6}
\end{aligned}
\tag{3.25}
$$

As regards specifically attitude, quaternions are used in the MATLAB/Simulink environment instead of Euler angles since they are computationally efficient and do not suffer from singularities. In detail, the equations adopted are

$$
\begin{aligned}
\dot{q_0} &= -\frac{1}{2}\vec{\omega}\dot{\vec{q}} \\
\dot{\vec{q_v}} &= \frac{1}{2}(\dot{q_0}\vec{\omega} - \vec{\omega} \times \vec{q_v})
\end{aligned}
\tag{3.26}
$$

with $q_0$ and $q_v$ being the scalar and vectorial components of the quaternion.

## 3.2 Guidance & Control

Figure 3.2 shows the typical control setup for RVD missions. The GNC function is divided into different constituents, such as sensors, GNC modes and actuators, whose role is defined and scheduled by the automatic Mission and Vehicle Management (MVM) unit. The automatic FDIR system interacts with both modules to define the onboard control system which communicates with the on-ground unit via Telemetry to receive Telecommand. In case of human presence to handle operations, another level above FDIR shall be considered. The GNC functions are on the contrary illustrated in the typical control loop of figure 3.3, to be implemented for every single degree of freedom. Particularly, depending on the phase of the mission, a specific GNC mode shall be selected, thus activating different algorithms and sensor types. The guidance function indicates the nominal trajectory values to be followed by the spacecraft while the control function is responsible for the activation of the proper torque or force commands in order to achieve and stick to the guidance instructions. Thruster management on the contrary translates these indications into real "on/off" commands for the selected thrusters.

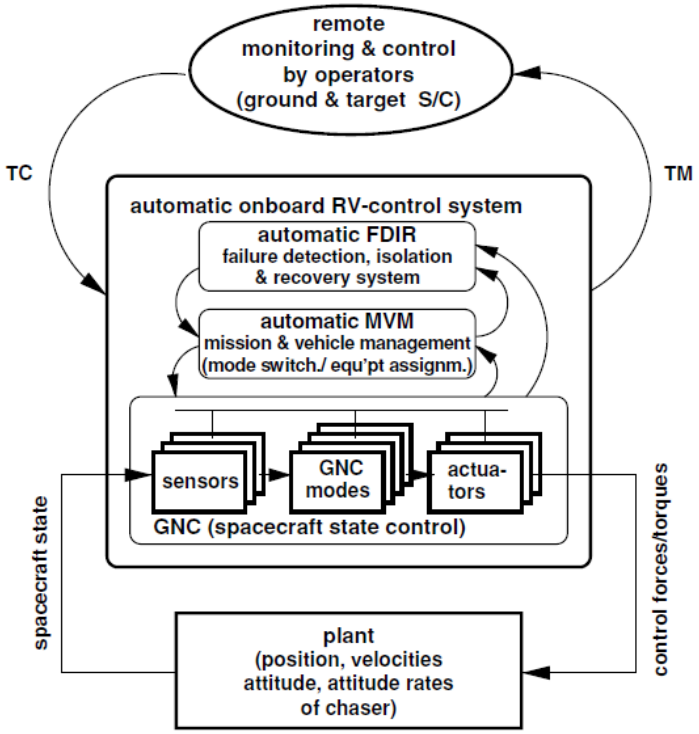



Figure 3.2: Block diagram illustrating the architecture of an automatic onboard rendezvous (RV) control system. The schematic details the hierarchical interaction between remote operator monitoring (via Telecommands and Telemetry), the core autonomous onboard systems (FDIR, MVM, and GNC), and the physical spacecraft plant (the chaser's dynamics and kinematics) [1].

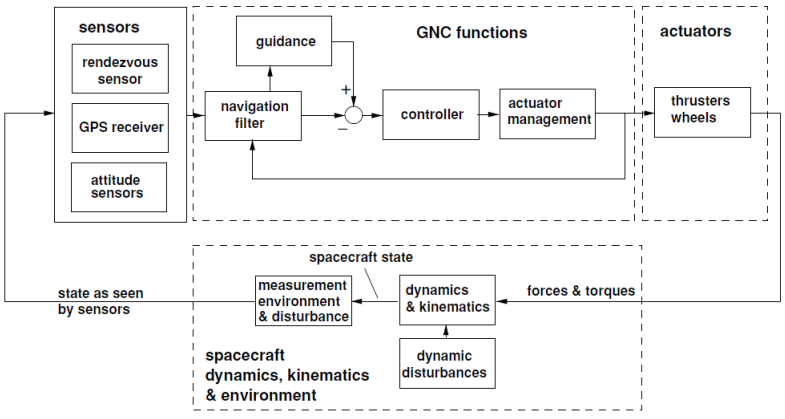



Figure 3.3: Block diagram of a Closed-Loop spacecraft Guidance, Navigation, and Control (GNC) system. It illustrates the continuous data flow between four main components: the sensor suite, the core GNC processing functions, the hardware actuators, and the physical spacecraft dynamics and environment [1].

More in detail, the guidance function pre-calculates boost manoeuvres, generates position, velocity and attitude profiles and propagates the instantaneous position of the center of mass during the mission. Guidance parameters cannot be anticipated and need to be computed starting from the actual state of the spacecraft. As regards boost manoeuvres, for example, different kinds of implementation may be taken into consideration: **Open-Loop**, where the necessary guidance parameters are calculated *a priori*, **Closed-Loop**, where the nominal trajectory is continuously checked, and manoeuvres with **mid-course corrections**, a slight modification from the Closed-Loop approach where possible corrections are applied only in specific checkpoints.
On the contrary, the control function conjugates the inputs coming from the ideal trajectory (from guidance) and the real one (from navigation), detects discrepancies between them and furnishes proper action to correct them.

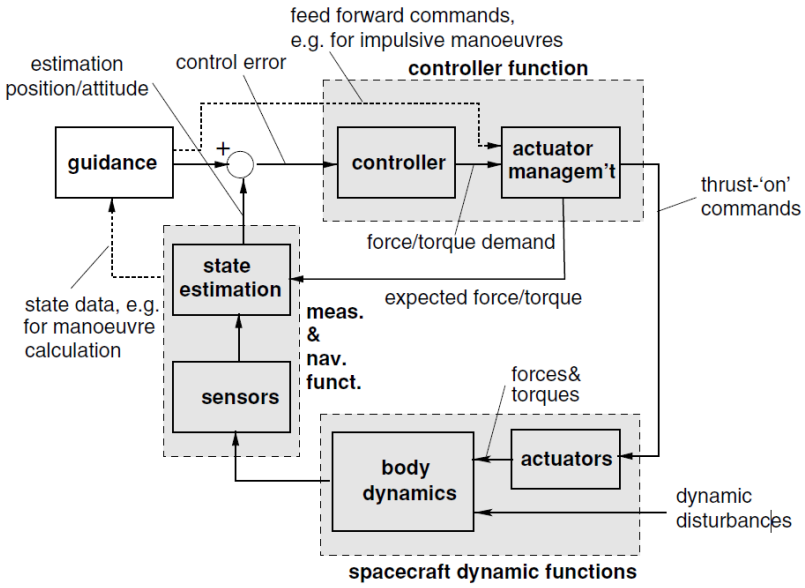



Figure 3.4: Functional block diagram of a Closed-Loop spacecraft Guidance, Navigation, and Control (GNC) architecture. The schematic details the signal flow and interactions between guidance generation, measurement and navigation functions, controller operations, and spacecraft body dynamics, including provisions for feed-forward commands and dynamic disturbance rejection [1].

Figure 3.4 illustrates the functional block diagram of a Closed-Loop controlled spacecraft. The *measurement and navigation function* is responsible for the state determination via sensors. The *controller function* exerts the commands on the propulsive system. The *actuation system* is represented by thrusters themselves.
For the last part of the trajectory, an Open-Loop implementation is no longer acceptable due to the high-level requirements imposed on the chaser's attitude and dynamics. The range of tolerability indeed depends on the guidance profile, the performance requirements and the type of disturbances. It is the result of a compromise between accuracy and fuel-saving. The control approach typically adopted tracks both the position and the relative attitude of the target docking port, thus coupling

the rotational and translational motions. This scenario has become the standard for all automatic GNC systems. A multi-variable design method is required to solve for this control problem.

For the Final Approach phase, a Sliding Mode Control was chosen for the attitude dynamics while a Non-linear Model Predictive Control was adopted for the position of the spacecraft.

### 3.2.1 Sliding Mode Control

Sliding Mode Control (SMC) is a non-linear variable structure control technique and it belongs to the class of Variable Structure Control Systems (VSCS), started with the work of Emel'yanov [6] [7] and Barbashin [8] [9] [10] in the 1960s. With $x(t)$ being the system state, this technique consists in defining a sliding variable $\sigma(x(t))$ which corresponds to a sliding surface (i.e. a subset of the state-space) when equal to zero [11]. In other words, the state error tends to zero when the system remains in the sliding surface. In addition to this, a feedback control law must be implemented to guarantee the nominal trajectory. Translated into analytical language [12]:

$$\begin{aligned} S(t) &\doteq \{x \in \mathbb{R}^n : \mathbf{s}(x,t) = 0\} \\ s(x,t) &= \tilde{y}^{(\gamma-1)} + k_\gamma \tilde{y}^{(\gamma-2)} + ... + k_2 \tilde{y} \end{aligned} \tag{3.27}$$

where $\tilde{y} = r - y$ is the tracking error as the difference between a reference state and a measured one, $\gamma$ defines the degree of the method, $k_i \in \mathbb{R}$ are chosen so that the characteristic roots of $P(\lambda) = \lambda^{\gamma-1} + k_\gamma \lambda^{\gamma-2} + ... + k_2$ have negative real part and $\mathbf{s} : \mathbb{R}^{n+1} \rightarrow \mathbb{R}$.

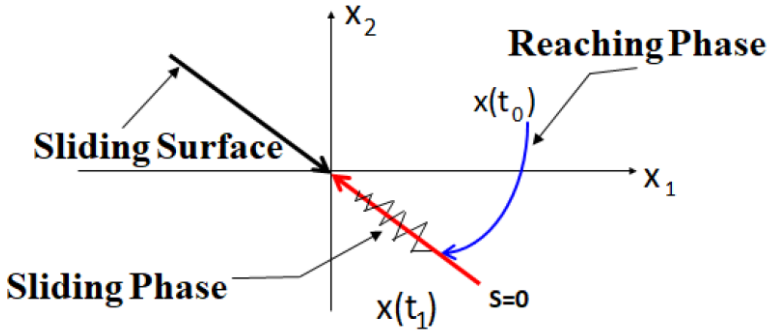



Figure 3.5: Two-dimensional phase plane portrait of a Sliding Mode Controller. The diagram highlights the transition from the initial reaching phase to the sliding phase along the surface $s = 0$, eventually converging to the target state $x(t_1)$ under discontinuous control efforts (chattering) [13].

Two phases may be identified: *reaching phase*, during which the state of the system moves towards the sliding surface, and *sliding phase* during which the system is kept on the sliding surface by the control input and the output error tends to zero. The

convergence on the sliding surface requires a finite amount of time, the equations involved are of reduced order and, above all, the Closed-Loop system is insensitive to disturbances during sliding phase, when adopting a first order SMC. The main drawback is the phenomenon of chattering for which the trajectory oscillates across the sliding surface. This could be mitigated by defining a boundary layer, that is, a corridor of tolerability around the sliding surface so that the error does not tend to zero anymore but to a set neighbouring zero. Besides, successive derivatives of the sliding variable could be equal to zero. Nonetheless, the chattering is only mitigated, not completely removed.

### 3.2.2 Non-linear Model Predictive Control

The NMPC is based on two different phases: firstly, the model of the plant is approximated at each step to make predictions on the future behaviour of the system over a given time horizon; secondly, the most adequate control sequence is selected based on the results of the previous phase. Similar strategies have already been consolidated in real CubeSat missions, such as CPOD, which used a linear MPC algorithm for ingress to a passively safe flying formation [14].
Three important definitions shall be introduced before continuing the discussion:

1. Prediction Horizon $H_p$: it is the temporal window within which the forecast is made by the controller together with the optimization of the system's dynamic response;

2. Sampling Time $T_s$: it is the interval between a control cycle and the following one. If too small, the computation cost becomes too high, causing instability phenomena, while on the contrary if too large the controller may not be able to compensate for rapid disturbances;

3. Control Horizon $H_c$: within the Prediction Horizon, it is the number of steps for which a variation of the input is taken into consideration. When small, only few initial controls are computed, thus dumping the transient response.

For each instant $t_k$ the system state is predicted at the time $t_k + T_p$. The predicted state is a function of the control inputs $U_k$, whose sequence is obtained by minimizing the following cost function:

$$J(u(t : t + T_p)) = \int_t^{t+T_p} (\|\tilde{y}_p(\tau)\|_Q^2 + \|u(\tau)\|_R^2) d\tau + \|\tilde{y}_p(t + T_p)\|_P^2 \tag{3.28}$$

where $\tilde{y}_p(\tau)$ is the predicted tracking error at time t, $u(\tau)$ is the input signal at time t and $\tilde{y}_p(\tau + T_p)$ is the predicted tracking error. The weighting matrices $Q$, $R$ and $P$ must be properly selected to guarantee the correct functioning of the NMPC. The matrices are typically diagonal with non-negative elements. $Q$ weights the state deviations from the first sampling instant up to the one preceding the end of the prediction

horizon, $R$ weights the input command and $P$ weights the state corresponding to the last time instant of the prediction horizon. An adaptive NMPC has been implemented within the simulator [15] and makes use of parameters, such as the chaser's mass, which are continuously updated. In particular, for the docking along the In-Track axes, the parameters required are the following ones:

1. Number of States $= 6$;
2. Sampling Time $T_s = 0.5$s;
3. Prediction Horizon $T_p = 3$s;
4. Tolerance $tol = 10^{-7}$

while the weighting matrices are

$$Q = \begin{bmatrix} 6\cdot10^4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2.6\cdot10^{-6} & 0 & 0 & 0 & 0 \\ 0 & 0 & 6\cdot10^4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4\cdot10^6 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.8\cdot10^6 & 0 \\ 0 & 0 & 0 & 0 & 0 & 6\cdot10^4 \end{bmatrix}$$

$$R = \begin{bmatrix} 10230 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1300 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10230 & 0 & 0 & 0 \\ 0 & 0 & 0 & 10230 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1300 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10230 \end{bmatrix}$$

$$P = \begin{bmatrix} 1.179\cdot10^6 & 0.0019 & 0 & 1.5016\cdot10^6 & 1.5608\cdot10^3 & 0 \\ 0.0019 & 4.3267 & 0 & -0.0169 & 4.0746 & 0 \\ 0 & 0 & 1.1795\cdot10^6 & 0 & 0 & 1.5015\cdot10^6 \\ 1.5016\cdot10^6 & -0.0169 & 0 & 1.4008\cdot10^7 & 1.2641\cdot10^4 & 0 \\ 1.5608\cdot10^3 & 4.0746 & 0 & 1.2641\cdot10^4 & 3.3903\cdot10^6 & 0 \\ 0 & 0 & 1.5015\cdot10^6 & 0 & 0 & 1.4008\cdot10^7 \end{bmatrix}$$

## 3.3 Navigation

The navigation system is hereby briefly presented. Its function is to provide the guidance and control functions with the necessary information regarding the state of the vehicle, via a digital filter which elaborates the inputs coming from different types of sensors. To this purpose, a Kalman filter is adopted: this instrument is capable of minimizing the estimation error in the least square sense and providing the actual state vector of the spacecraft. Its functionality is well documented in [16], [17], [18] and [1].
Figure 3.6 illustrates the principle of operation of a Kalman filter: the outputs of the state estimation and error covariance matrix will be used in the subsequent cycle while

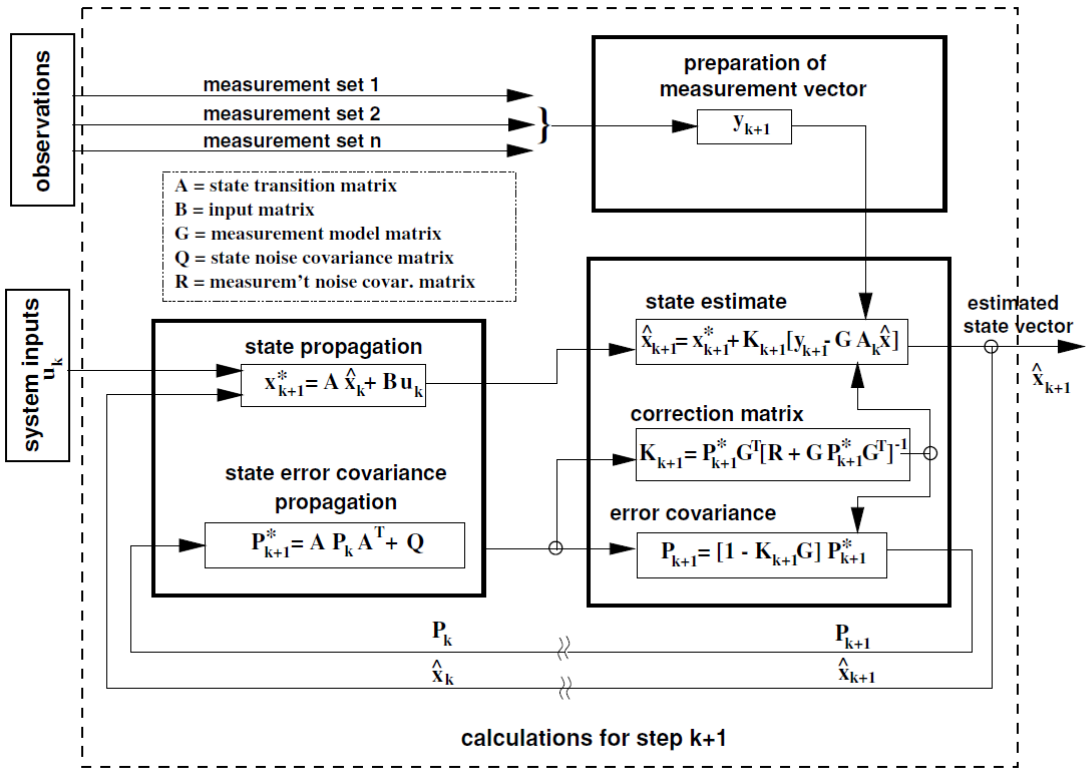



Figure 3.6: Computational flow of a single discrete-time Kalman filter iteration. This diagram illustrates the recursive algorithmic structure for one time step $(k+1)$. System inputs and measurement data are processed through prediction (state and covariance propagation) and correction (measurement update via Kalman gain $K_{k+1}$) phases to calculate the estimated state vector $\hat{x}_{k+1}$ and the updated error covariance matrix $P_{k+1}$. The updated values are then fed back for use in the subsequent iteration [1].

the block named "preparation of measurement vector" just refers to the possibility of the measurement to be translated in a form suitable for upcoming calculations. The filter is valid only for cases where prediction and measurement errors are equal to Gaussian random processes with zero mean. In addition to this, some terms may result particularly heavy from a computational point of view. Some solutions are proposed in Section 6.2.1 of [1].

## 3.4 Environmental Disturbances

These types of variations to the nominal orbit are discussed in Appendix A. In addition to them, it is realistic and more accurate to account for internal sources of interference, such as torques generated by mechanisms and moving parts and propellant sloshing. Together with the forces described in Appendix A, some additional torques, due to a realistic condition of non-equilibrium, shall be taken into consideration. Firstly, since the aerodynamic centre does not coincide with the centre of

mass, the resulting action is

$$\vec{T_D} = \vec{R_{cp,g}} \times \vec{F_D} \tag{3.29}$$

where $\vec{R_{cp,g}}$ is the distance between the two points in the body. In addition to this, the non-uniformity of the gravitational field along the structure leads to another type of torque:

$$\vec{T_g} = -3\omega^2 \vec{r} \times \vec{I} \cdot \vec{r} \tag{3.30}$$

where the position $\vec{r}$ is expressed in the $F_{eq}$ reference frame.

# Chapter 4

# Final Approach

The Final Approach represents the most critical part of the mission, since it is the last phase before mating. For this reason, the requirements imposed are particularly strict (see also Table (2.4)): indeed, docking is a sort of controlled impact and every alteration could provoke a fatal outcome. The objective of this chapter is to analyze different motion profiles to investigate how different the outcome could be. In this perspective, the analysis conducted may be defined as a two-step validation: firstly, different profiles are studied to evaluate and compare costs and duration of the manoeuvre; secondly, a MonteCarlo analysis will be introduced for one of these scenarios as an example of general procedure to verify the robustness of the method.

## 4.1 Nominal profile analysis

At first, the velocity profile followed by the chaser from HP#4 to the docking point is investigated under nominal operating conditions, i.e., in the absence of system failures or anomalies. The initial set-ups adopted for the simulations are defined within the Radial-In-Track-Cross-Track (RIC) coordinate system and are expressed as follows:

$$\vec{X}(0) = \begin{cases} X_R(0) = 0 \\ X_I(0) = 25 \\ X_C(0) = 0 \end{cases} \quad [\mathrm{m}] \qquad \vec{V}(0) = \begin{cases} V_R(0) = 0 \\ V_I(0) = v_0 \\ V_C(0) = 0 \end{cases} \quad [\mathrm{m/s}]$$

The constant $v_0$, representing the initial In-Track velocity component, is not defined *a priori* since one of the objectives of the simulations is to assess the sensitivity of the approach dynamics to varying initial velocity inputs along this axis. The profile chosen is trapezoidal because it is easily implementable and controlled decelerations reduce the risk of accidental collisions [[19]][[20]].
The instantaneous velocity of the chaser is governed by a piecewise function of the

distance $d$ from the docking point, comprising two distinct deceleration phases:

$$V_z = \begin{cases} -\frac{1}{2} V_{max} \cdot \left[ \frac{min(d, decel_pos)}{decel_pos} \right]^{1.7} & \text{if } d > \text{transition} \\ \\ -\frac{1}{2} V_{max,fine} \cdot \left[ \frac{min(d, decel_pos_fine}{decel_pos_fine} \right]^{1.1} & \text{if } d < \text{transition} \end{cases}$$

Once the value of $-0.004$ m/s is reached, it is automatically held constant until the end of the simulation. Only the exponents for the mathematical functions are fixed parameters since it has been demonstrated [20] that values below unity are found to compromise the execution of CAMs.

### 4.1.1 First scenario

The results obtained for the first simulation case, illustrated in figure 4.1, correspond to an initial In-Track relative velocity of $v_0 = 0.05$ m/s. As shown in figure 4.1, which reports the velocity as a function of the relative distance from the docking point, the first deceleration phase is initiated at 15 m from the target. During this phase, the chaser decelerates from -0.05 m/s to -0.02 m/s, covering a distance of approximately 7 m.
Figure 4.2, which presents the velocity profile as a function of time, indicates that this first deceleration phase is completed after approximately 200 seconds. The subsequent final deceleration phase requires a shorter spatial interval while the associated time duration remains comparable to that of the first phase. A terminal velocity of -0.004 m/s is reached and maintained until the simulation end condition is met, corresponding to a residual distance of 0.008 m from the docking point. The total duration of the mission for this approach scenario remains below 20 minutes.
Figure 4.3 reports the control forces generated by the thruster system along the three RIC axes throughout the manoeuvre. The In-Track component (green) exhibits the largest magnitude (about 15 $mN$) and the highest-frequency oscillations, which are consistent with the continuous adjustments required to track the prescribed velocity profile. The Radial and Cross-Track components (red and blue, respectively) remain of comparatively smaller amplitude, reflecting the predominantly one-dimensional nature of the approach trajectory.
Figure 4.4 provides the most direct assessment of Closed-Loop tracking performance: the Radial and Cross-Track components are kept close to zero throughout the entire manoeuvre, overlapping almost perfectly on the time axis. This result confirms that the controller successfully maintains lateral containment within the approach corridor: this represents a critical safety requirement for proximity operations.

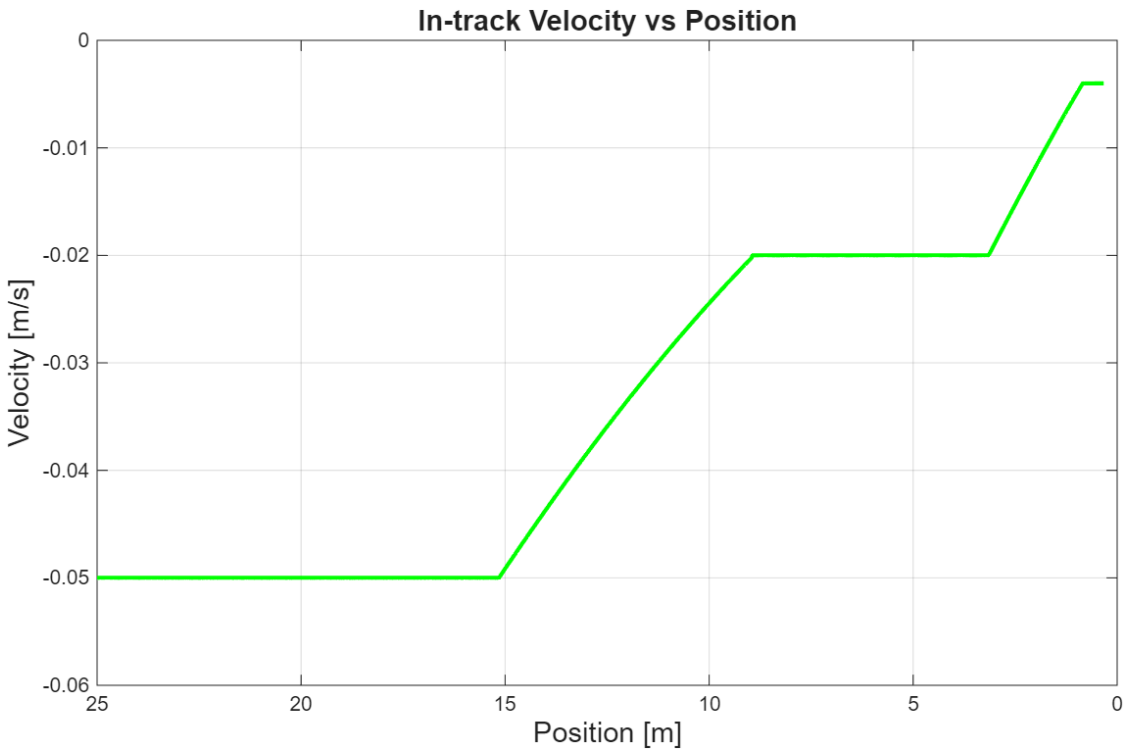



Figure 4.1: *Nominal Velocity vs Position: First Scenario*

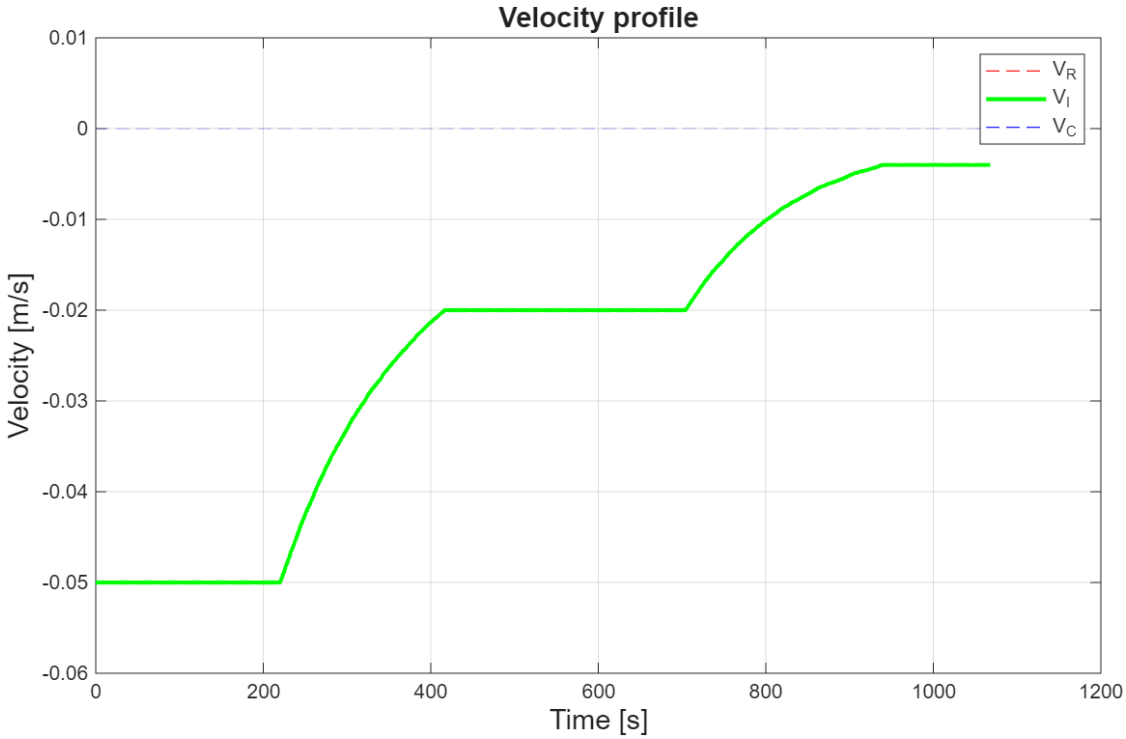



Figure 4.2: *Nominal Velocity vs Time: First scenario*

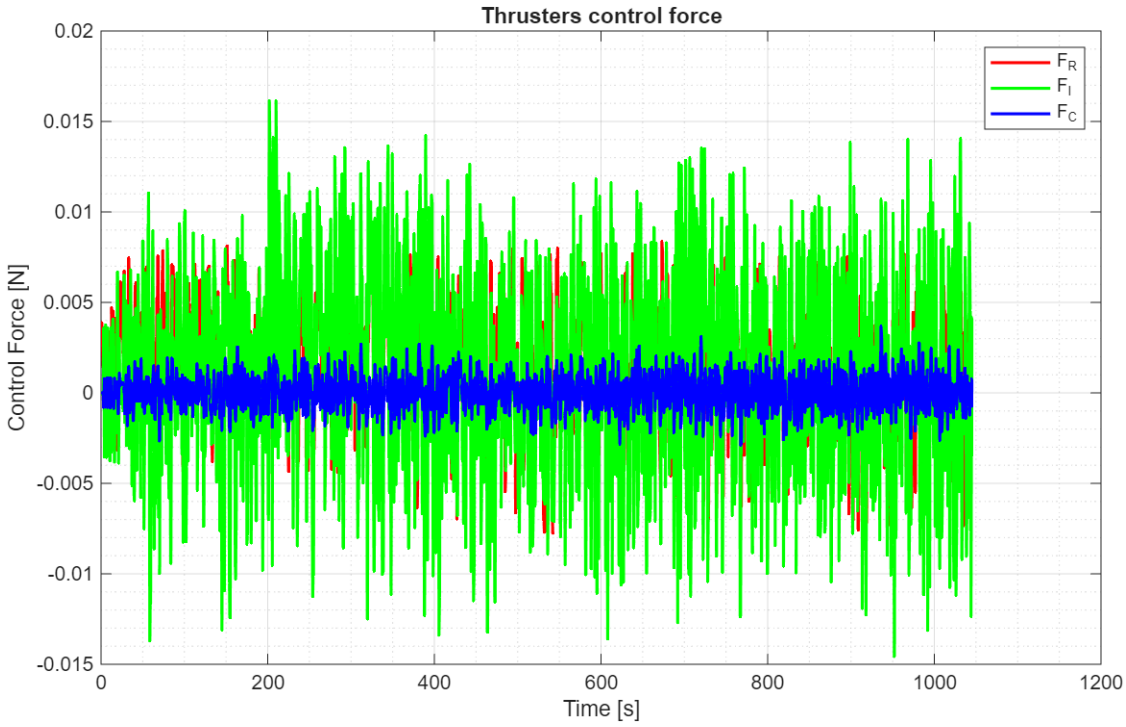



Figure 4.3: *Thrusters control force: First Scenario*

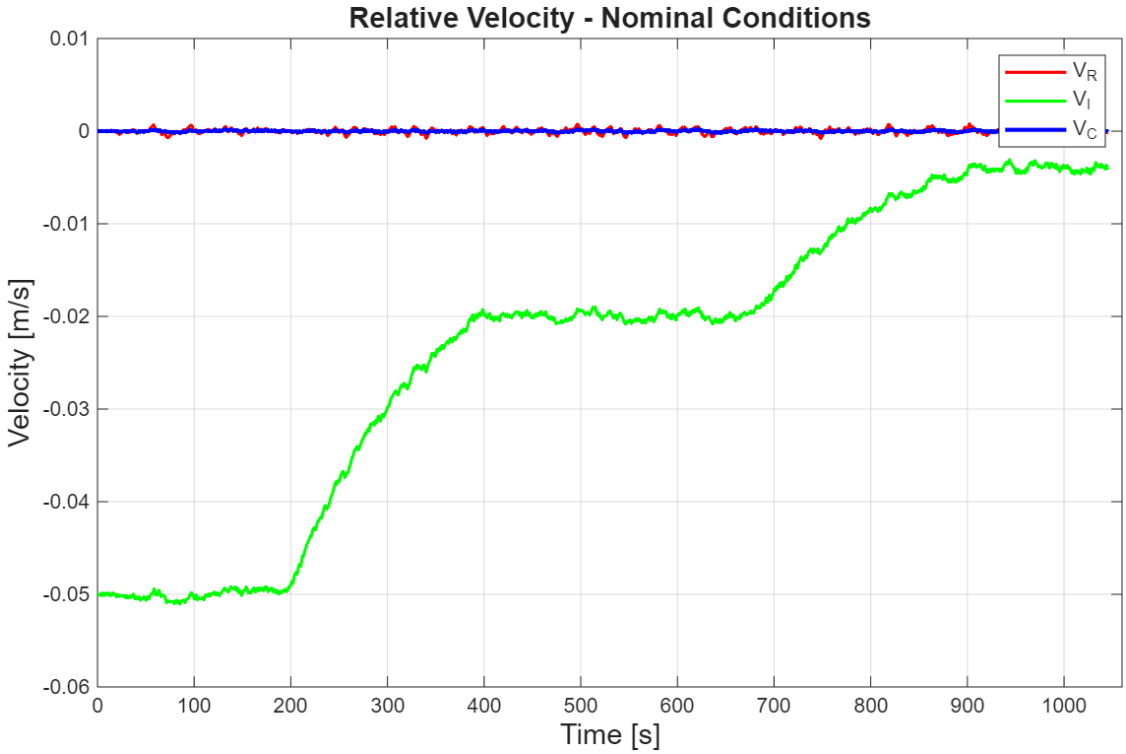



Figure 4.4: *Relative velocity - nominal conditions: First Scenario*

### 4.1.2 Second scenario

In the second case (figure 4.5), only a slight temporal delay is introduced for the first deceleration trigger, making the braking phase spatially closer to the target, compared to the first case. As a consequence, the manoeuvre is completed in a shorter total interval, as observable in figure 4.6. This kind of approach leads to tighter safety margins and a sharper command transition: a potential emergency operation (CAM) from within this gate is more demanding in terms of required $\Delta V$, making this configuration less robust to off-nominal events.
Regarding control forces on thrusters (figure 4.7), no significant differences in the amplitude can be highlighted with respect to case 1: only a reduced frequency along the In-Track axis is reported.

### 4.1.3 Third scenario

The duration of the manoeuvre is further reduced by starting the first deceleration phase when the chaser is 10 meters from the target (as in figure 4.10), i.e., about 300 seconds after the beginning of the final docking operation. By postponing the deceleration, the system can complete the operation in almost 16 minutes, with no significant alterations for the control forces, whose peaks still reach $\pm 0.015\ N$.

### 4.1.4 Fourth scenario

Alternatively, an anticipation of the first deceleration is suggested, in order to evaluate the effects of a less premature transition, as in case 4. The more conservative strategy, potentially beneficial from a safety point of view (particularly in the presence of navigation uncertainties or thruster response delays) impacts negatively on the total duration, as confirmed by the corresponding velocity profile over time (4.14): the whole manoeuvre runs for about 19.5 minutes, making it the longest so far. Nonetheless, the fourth scenario displays the most gradual and continuous transitions between velocity plateaus, with well-rounded inflection points suggesting a guidance law with higher-order continuity constraints or stronger smoothing filters. In addition to this, control forces adjusting the velocity profile present recurring peaks, as depicted in figure 4.15, with lower intensity, about $\pm 0.014\ N$, suggesting that the smoother guidance command in this case mirrors a marginally reduced control effort, though the difference is not dramatic. This result carries an important design implication: the relationship between the deceleration point and propellant efficiency is non-linear. Anticipating the braking phase beyond a certain threshold does not continue to reduce fuel consumption; rather, it increases the cumulative $\Delta V$ by extending the time over which the thruster system must actively compensate for environmental disturbances. From a mission operations perspective, the fourth case is the configuration best suited when the priority is placed on abort capability and robustness rather than on timing efficiency. However, its adoption as the nominal approach profile would require a justification of the additional propellant expenditure, approximately 44.5% more than Case 6, as it will be discussed further on.

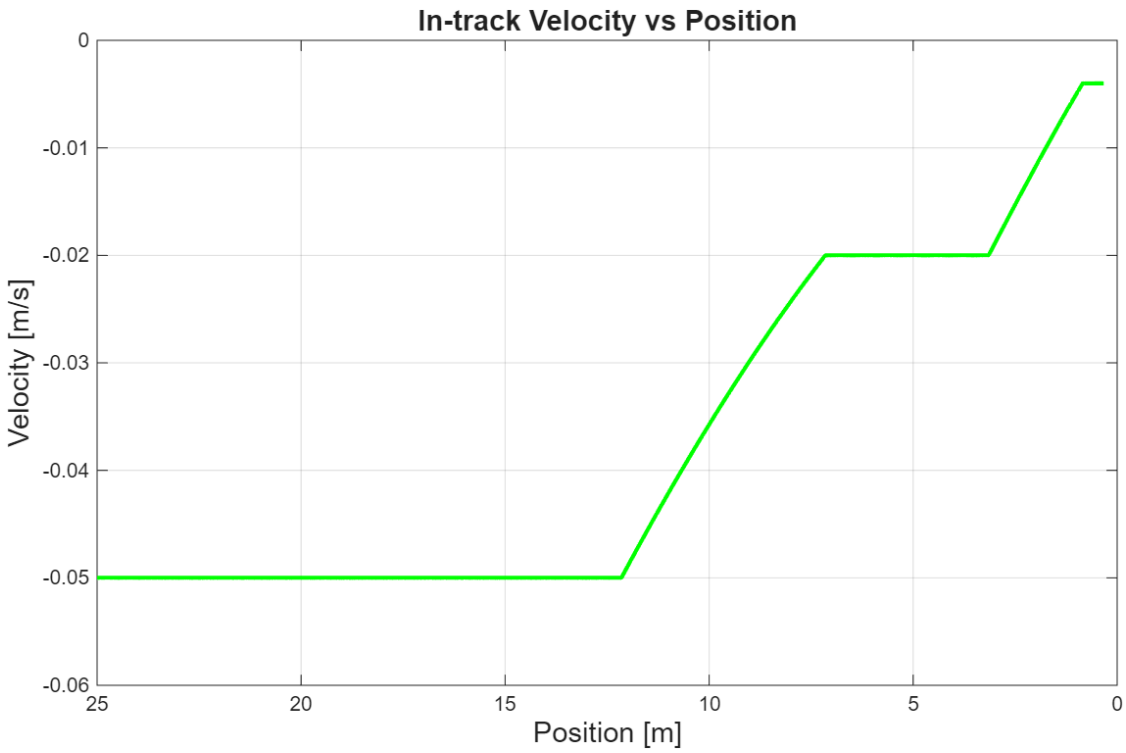



Figure 4.5: *Nominal Velocity vs Position: Second Scenario*

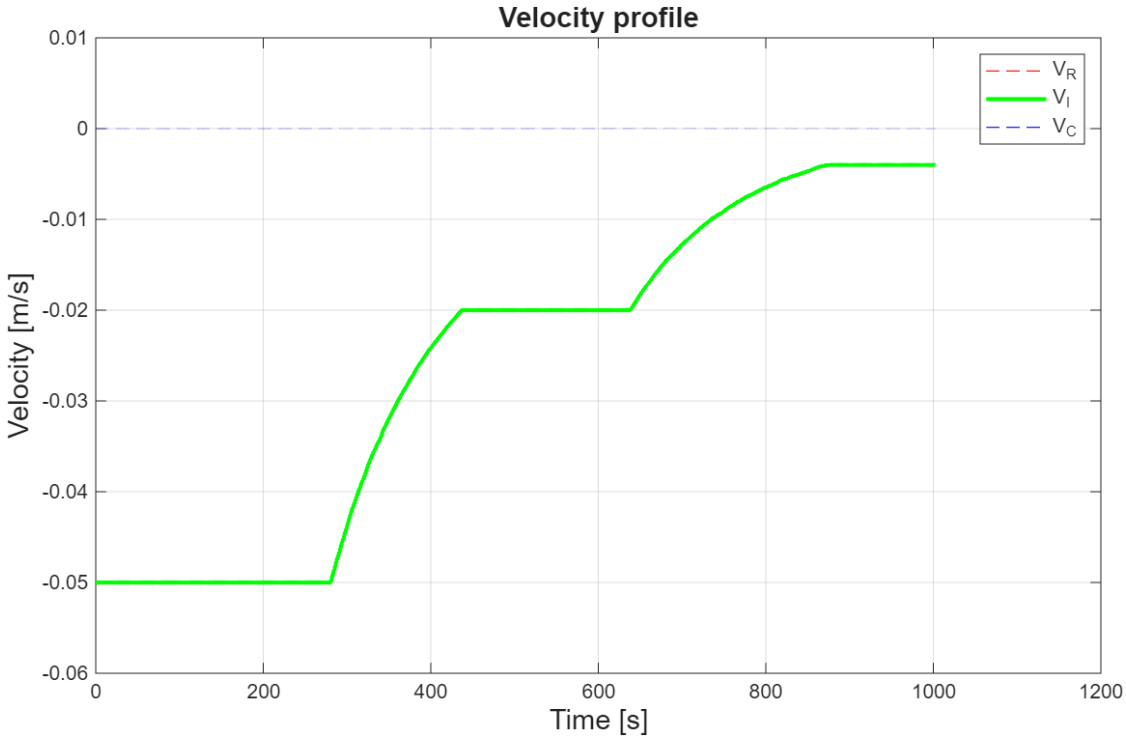



Figure 4.6: *Nominal Velocity vs Time: Second Scenario*

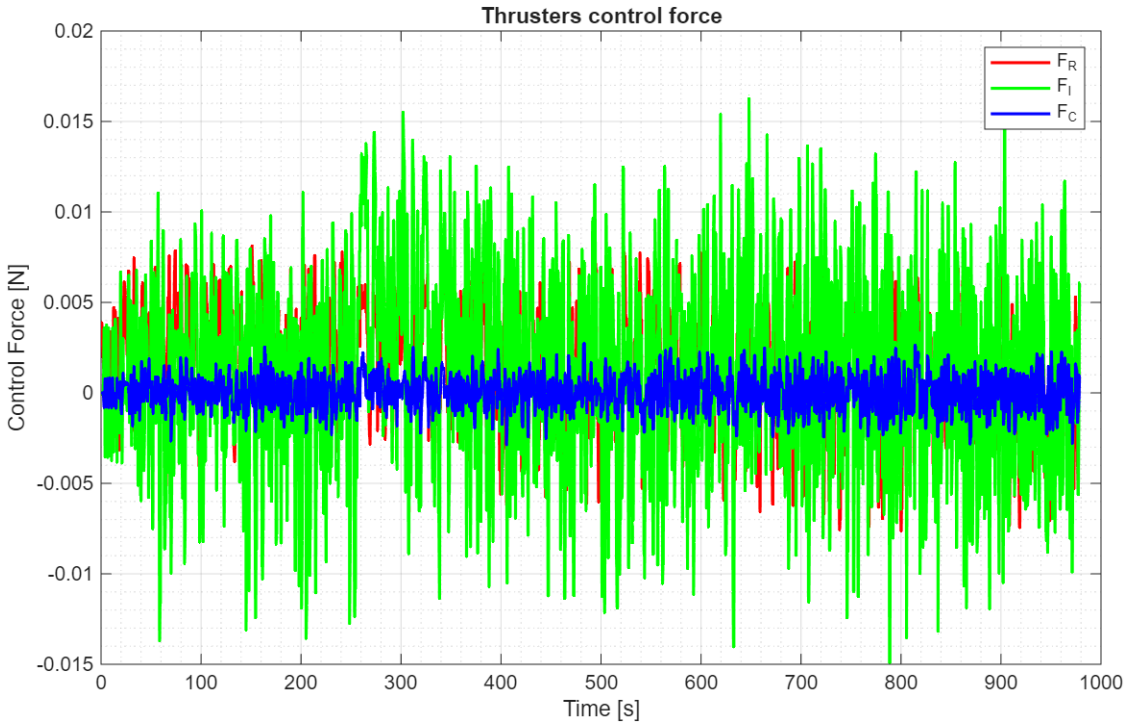



Figure 4.7: *Thrusters control force: Second Scenario*

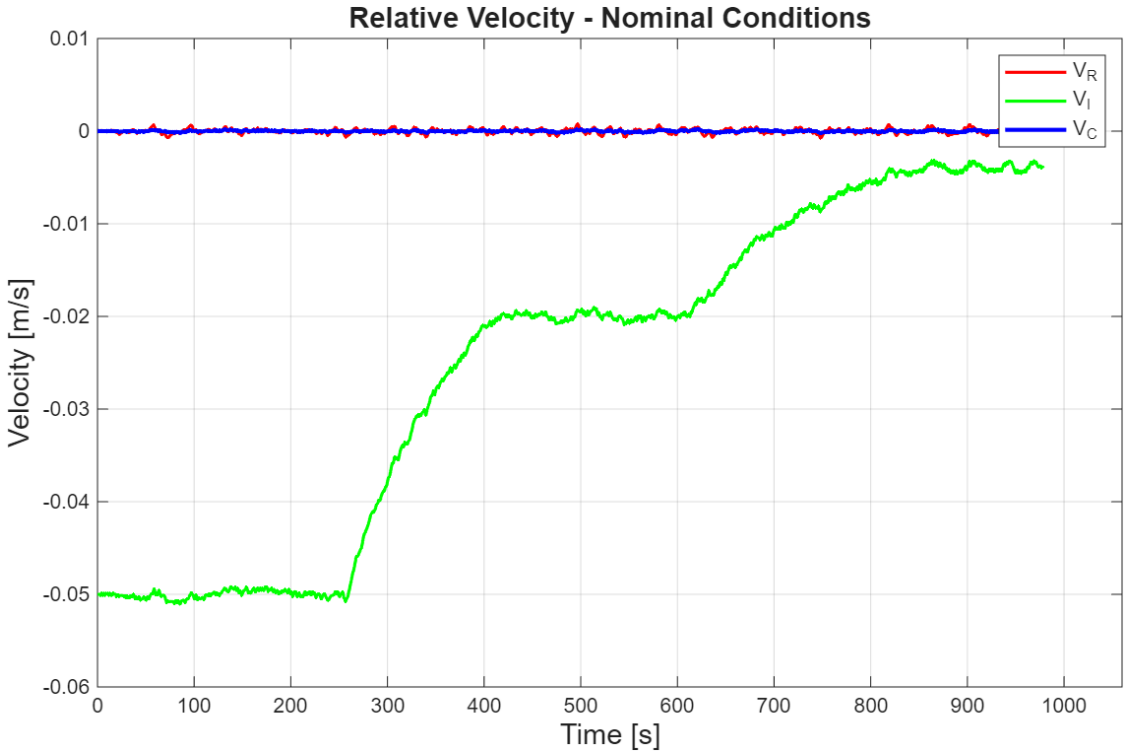



Figure 4.8: *Relative velocity - nominal conditions: Second Scenario*

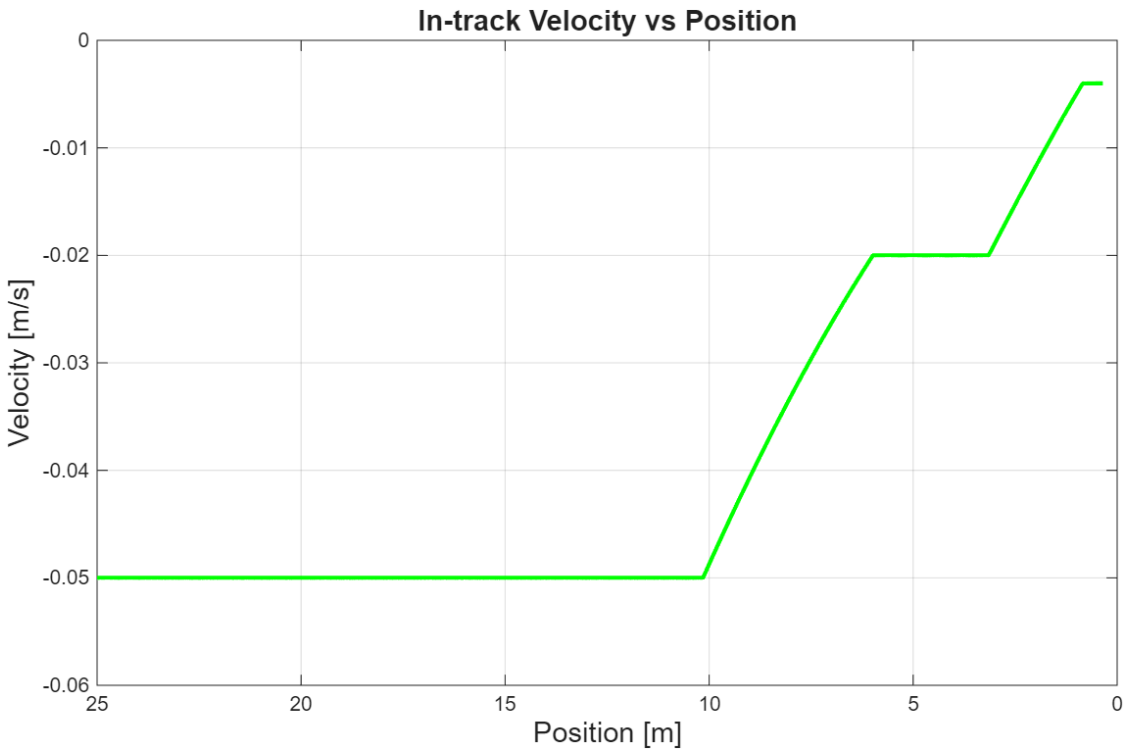



Figure 4.9: *Nominal Velocity vs Position: Third Scenario*

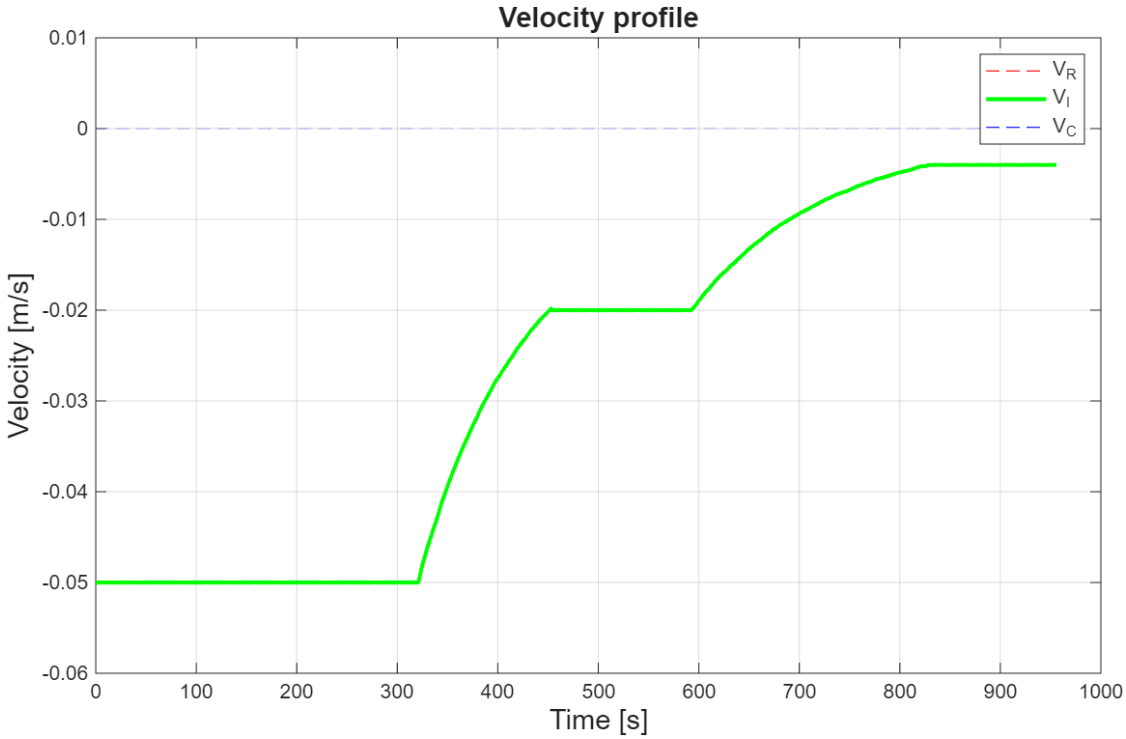



Figure 4.10: *Nominal Velocity vs Time: Third Scenario*

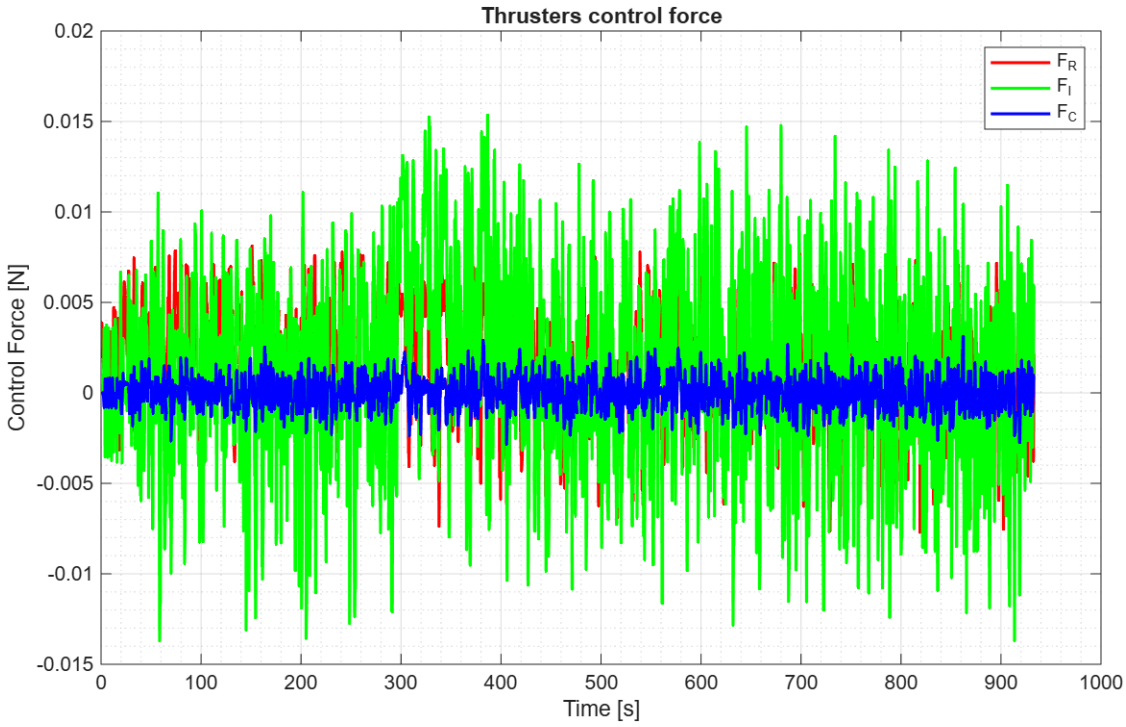



Figure 4.11: *Thrusters control force: Third Scenario*

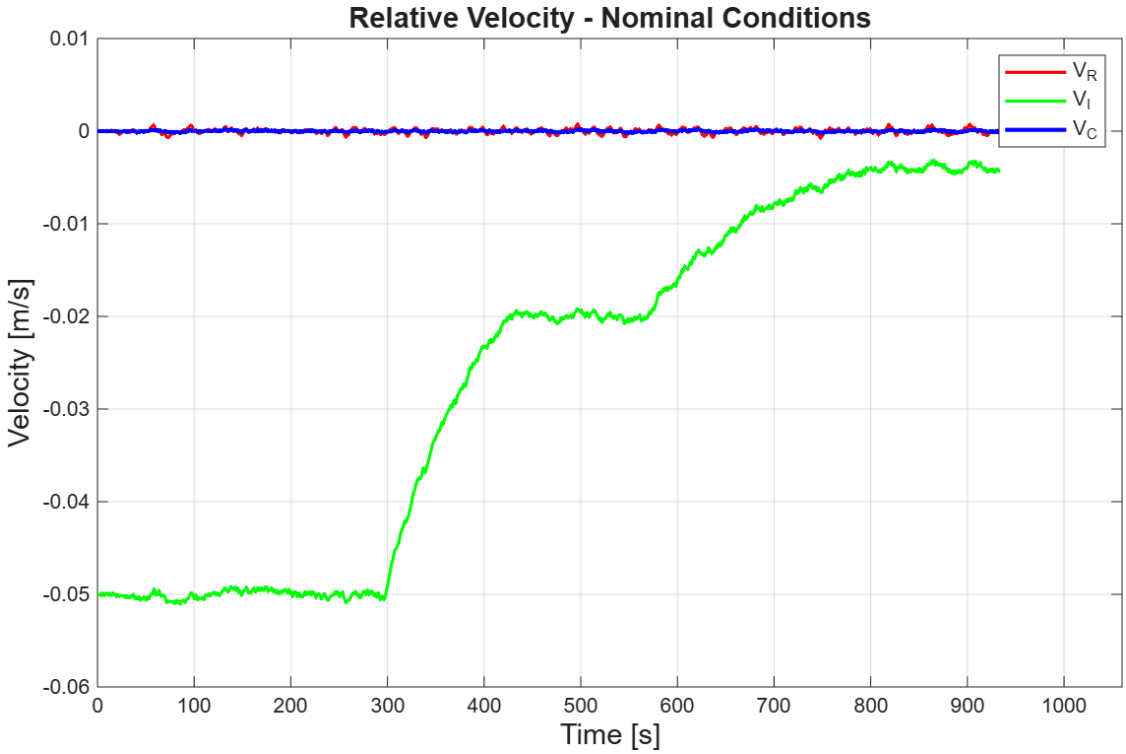



Figure 4.12: *Relative velocity - nominal conditions: Third Scenario*

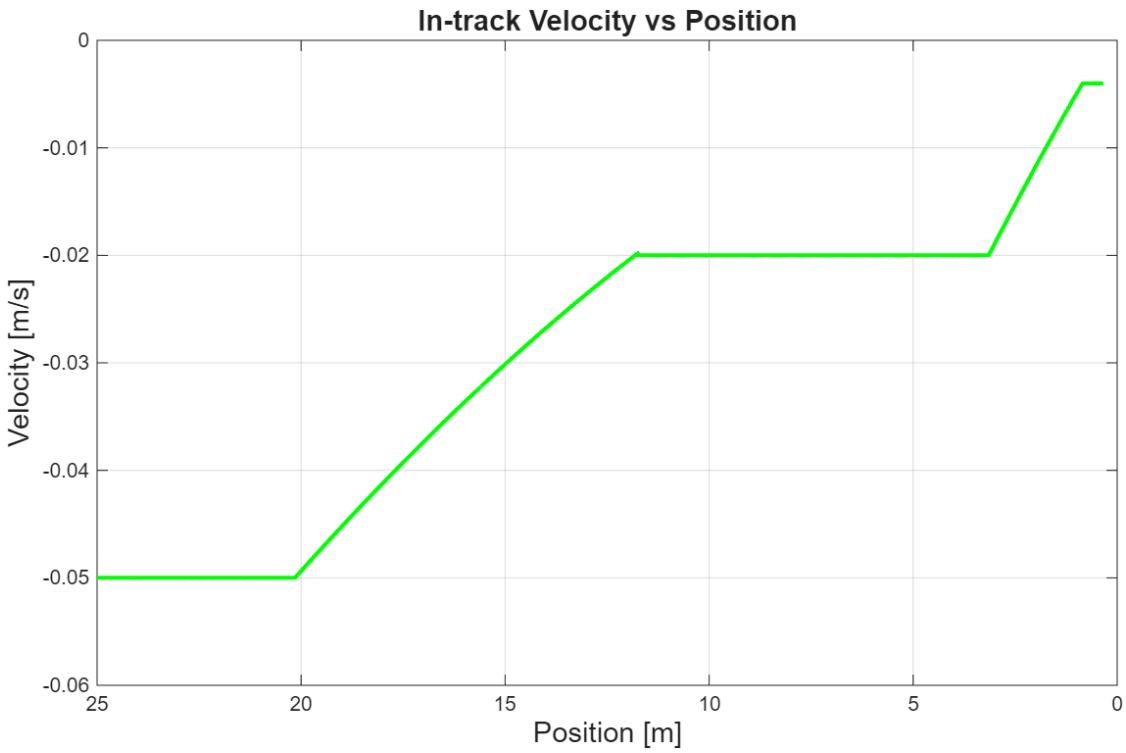



Figure 4.13: *Nominal Velocity vs Position: Fourth Scenario*

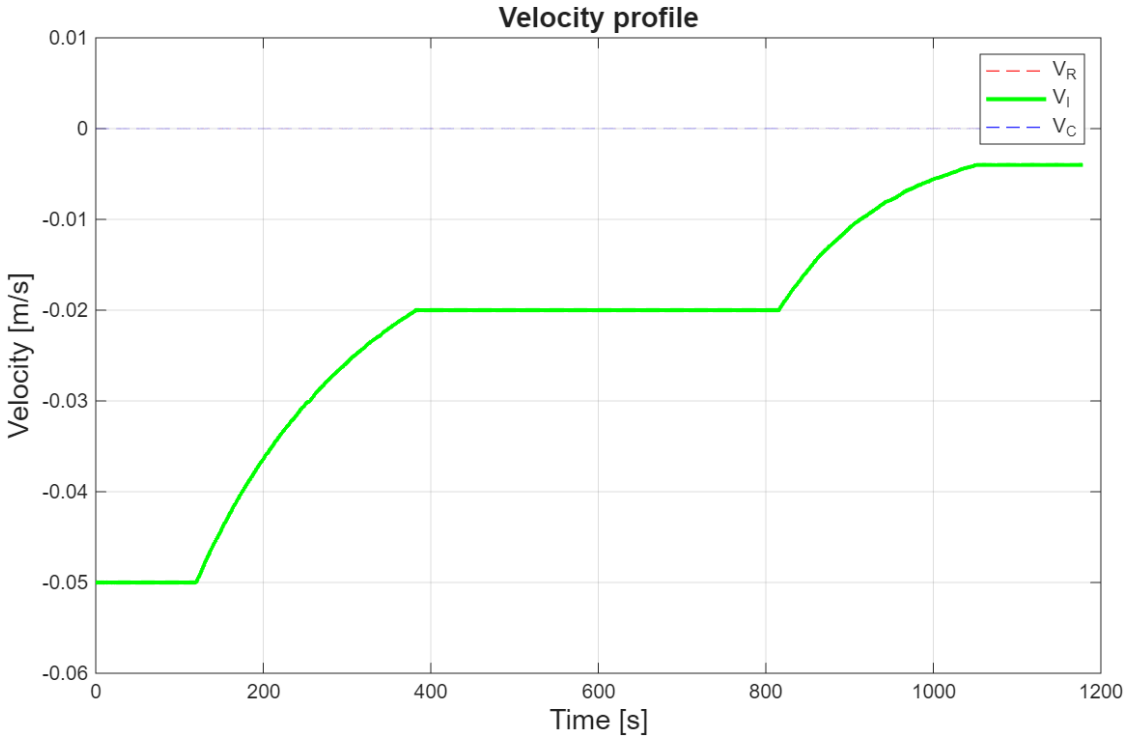



Figure 4.14: *Nominal Velocity vs Time: Fourth Scenario*

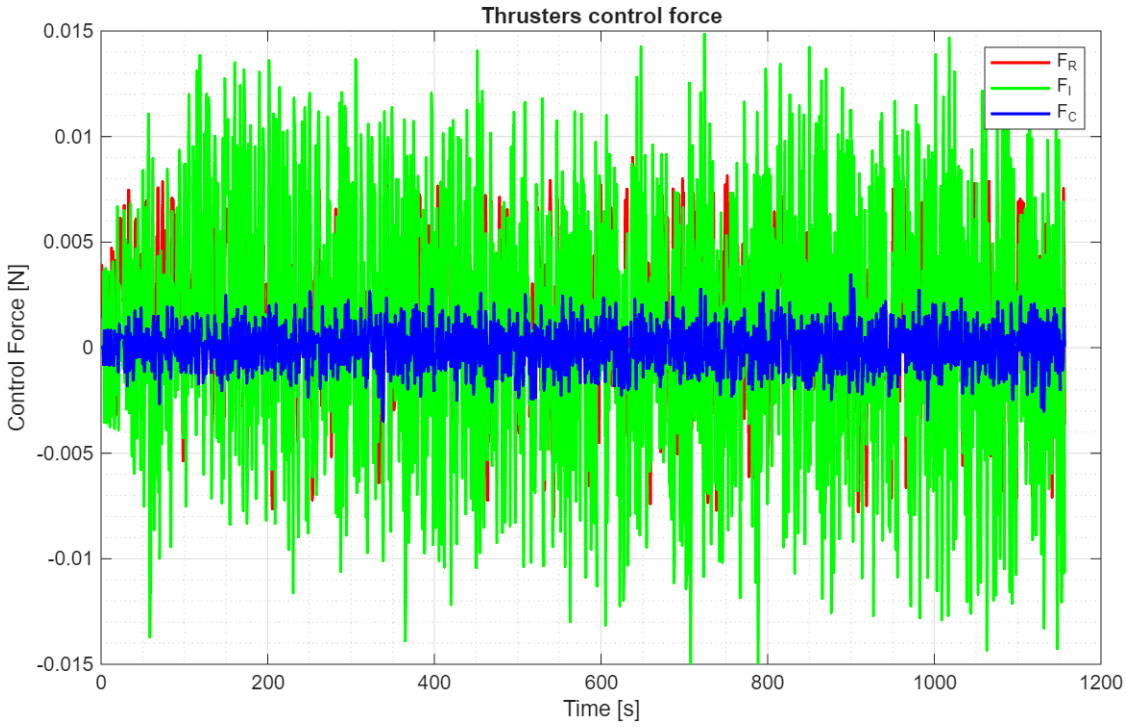



Figure 4.15: *Thrusters control force: Fourth Scenario*

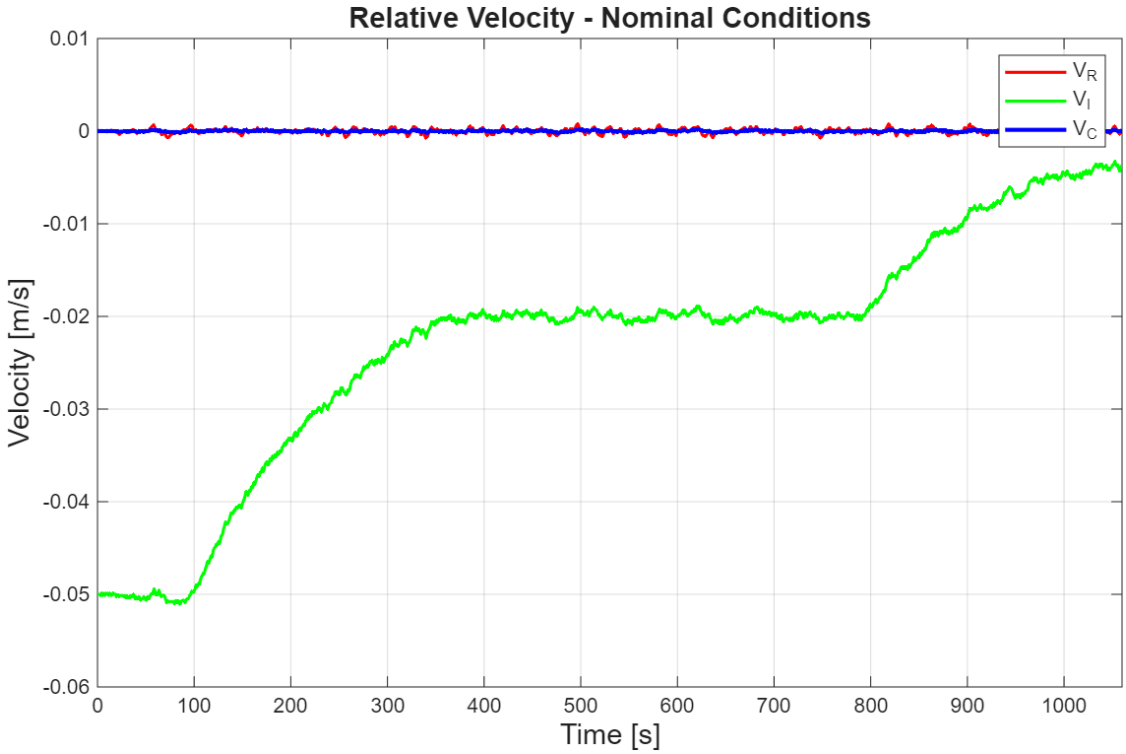



Figure 4.16: *Relative velocity - nominal conditions: Fourth Scenario*

### 4.1.5 Fifth scenario

At this point, two different scenarios offer an alternative perspective. The fifth simulation starts with an initial In-Track relative velocity $v_0 = -0.075$ m/s in order to investigate the dynamic and propulsive implications of a higher initial speed. This choice is motivated by the interest in reducing the total duration of the simulation at the cost of a more demanding deceleration phase. The velocity profile over time (figure 4.18) confirms that this configuration allows the docking to end within 15 minutes.
However, the most critical aspect emerges from figure 4.19, which shows that control force peaks exceed +0.020 $N$ along the In-Track direction during the first deceleration phase. Furthermore, the higher initial velocity increases the kinetic energy of the chaser at the beginning of the approach, which directly impacts the severity of a potential collision scenario in the event of a guidance or navigation failure prior to the first deceleration step. Nonetheless, the total $\Delta V$ expenditure of 0.450 m/s and the fuel consumption of 0.0263 kg represent a favorable outcome from a propellant efficiency perspective, second only to Case 6. This result reflects the fact that the higher initial velocity reduces the time spent under active control.

### 4.1.6 Sixth scenario

For the last scenario (figure 4.21), only the intermediate velocity is modified to 0.03 m/s, as distinct from all the previous cases (0.02 m/s). This configuration further reduces the temporal interval for the final manoeuvre (less than 850 seconds, as demonstrated in figure 4.22), representing a reduction of nearly 25% with respect to the baseline Case 1 and approximately 30.8% with respect to Case 4. This scenario is the most efficient in terms of propellant expenditure, achieving the lowest $\Delta V$ and fuel consumption so far, a non-negligible margin in the context of small satellite mission design where propellant mass directly affects launch cost and mission lifetime. In addition to this, control forces present peaks of about $\pm 0.015$ $N$, confirming that the higher intermediate hold velocity does not impose additional peak force requirements on the actuation system. However, the adoption of a higher intermediate hold velocity introduces specific safety concerns that must be carefully weighed: the elevated speed through the intermediate corridor reduces the time available for the guidance system to detect and respond to anomalies and the initiation of a CAM from this region imposes a greater cost, potentially reducing the abort success probability within the allocated propellant margin.

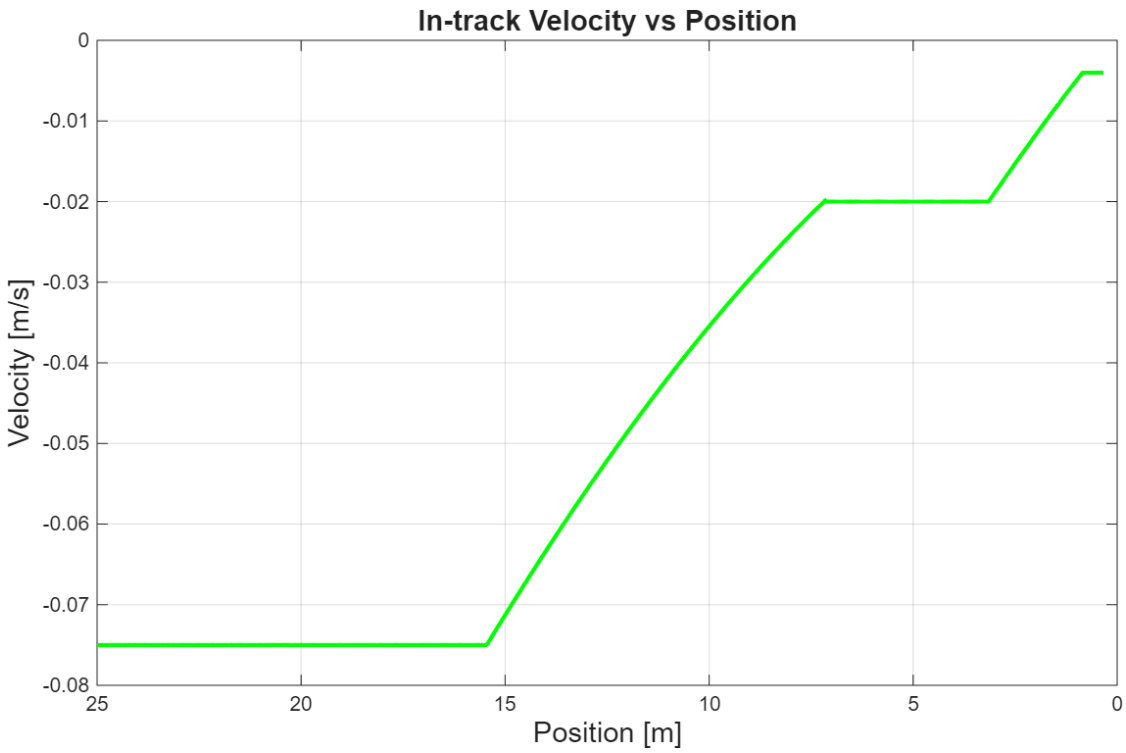



Figure 4.17: *Nominal Velocity vs Position: Fifth Scenario*

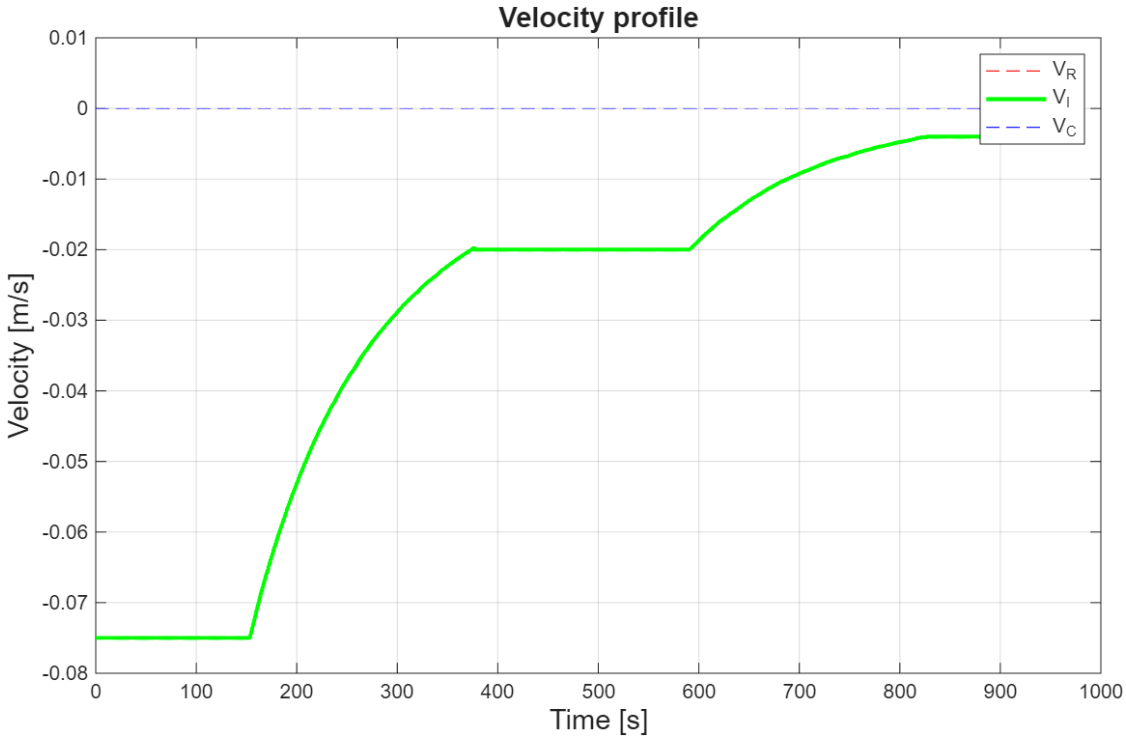



Figure 4.18: *Nominal Velocity vs Time: Fifth Scenario*

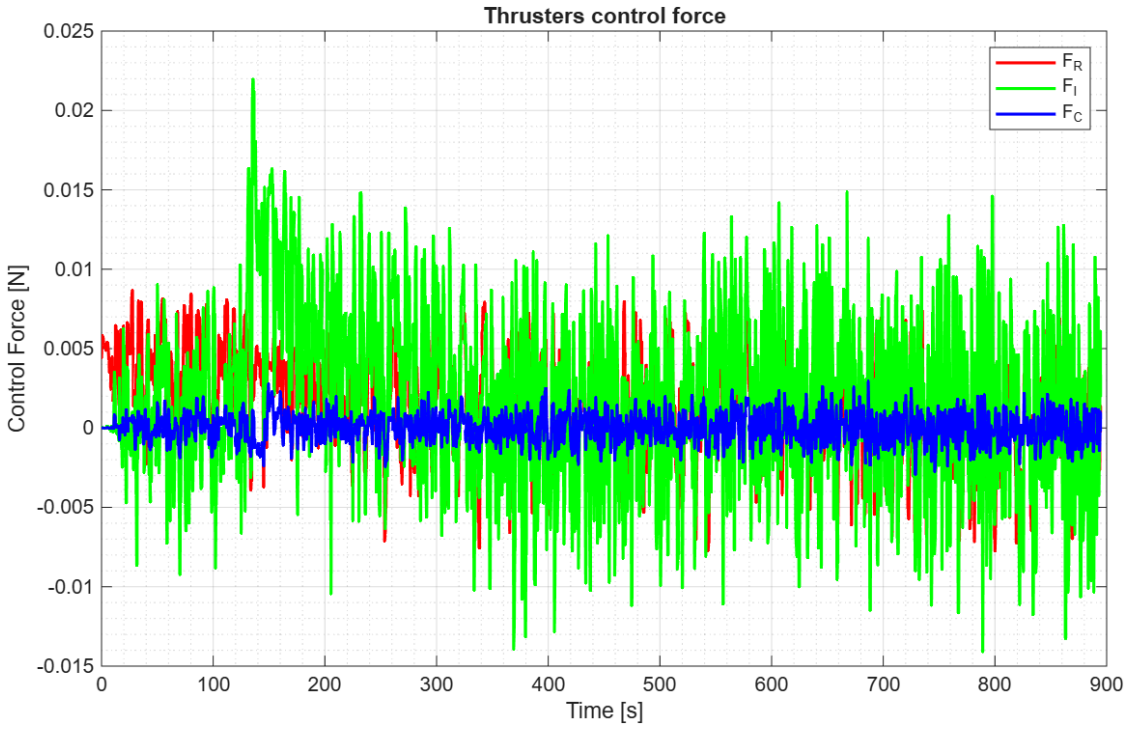



Figure 4.19: *Thrusters control force: Fifth Scenario*

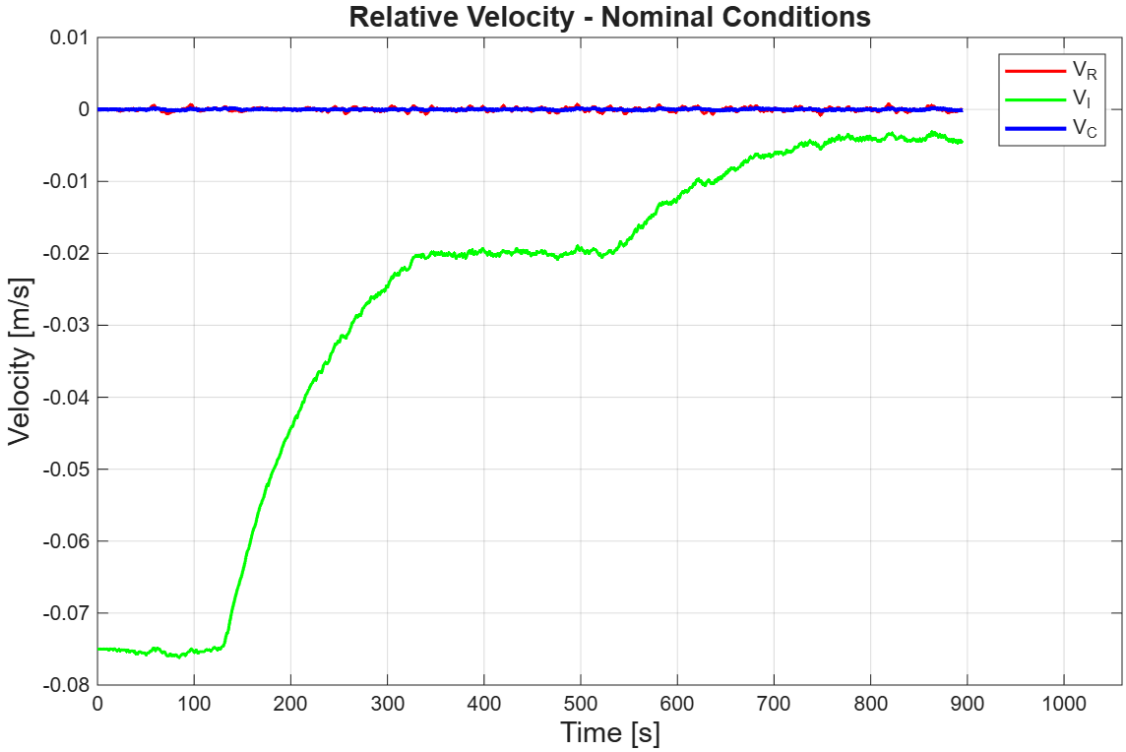



Figure 4.20: *Relative velocity - nominal conditions: Fifth Scenario*

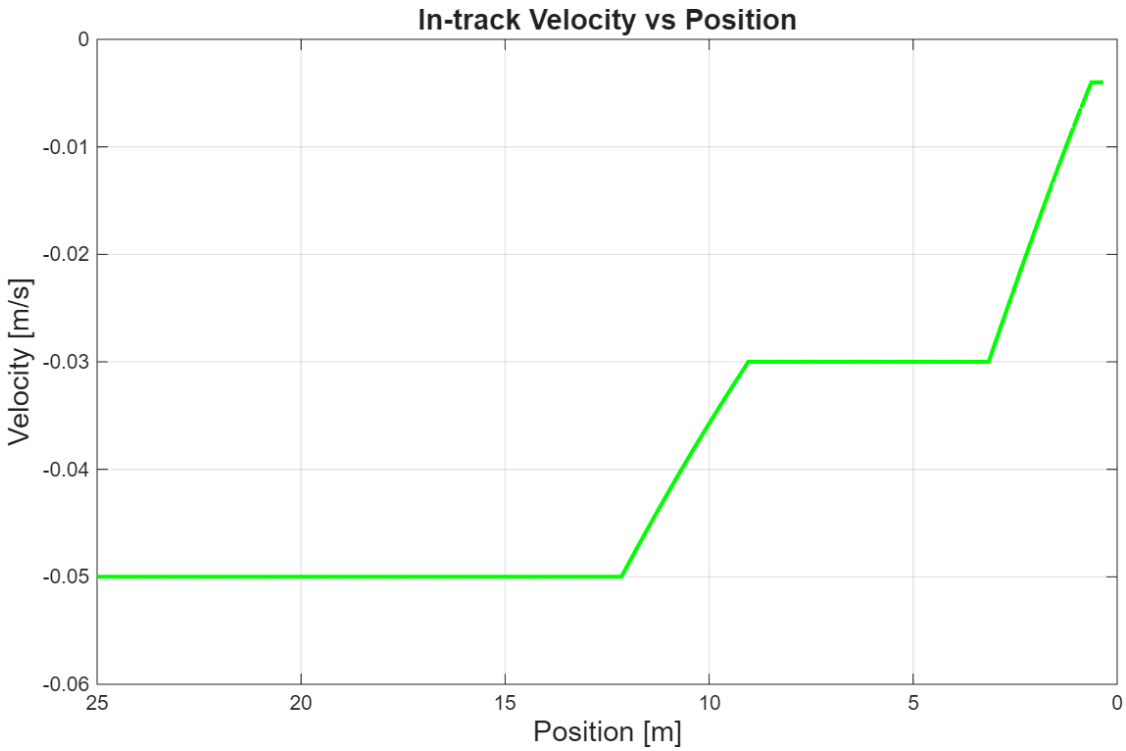



Figure 4.21: *Nominal Velocity vs Position: Sixth Scenario*

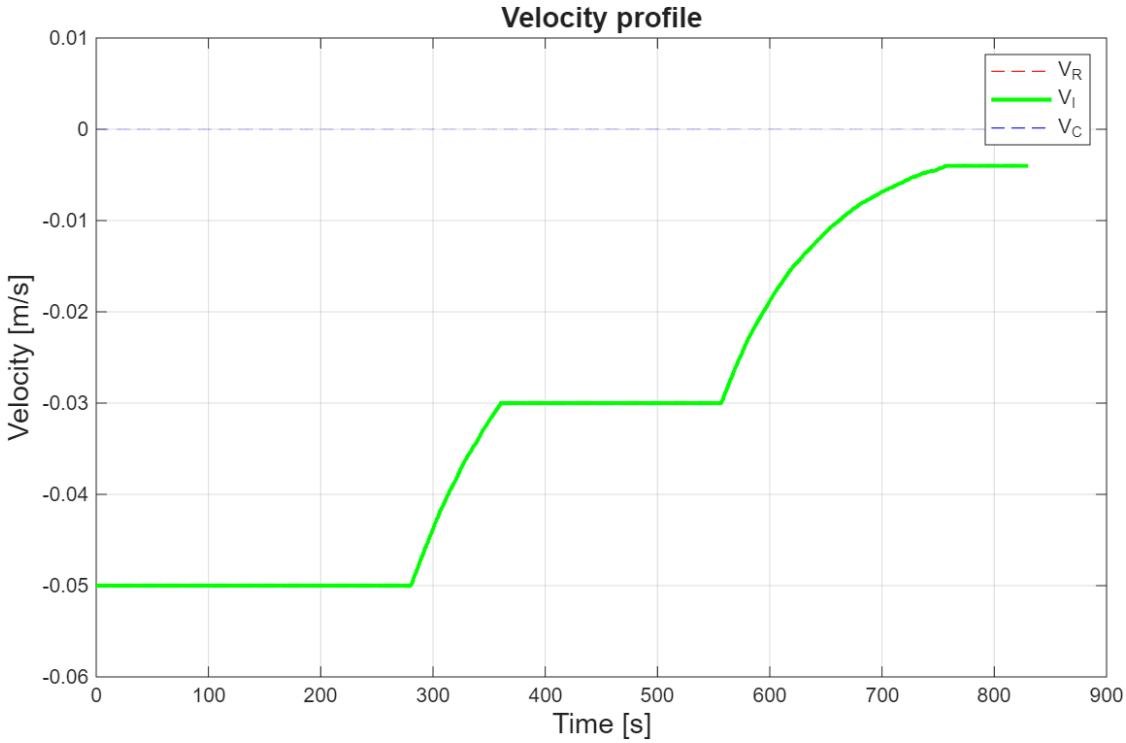



Figure 4.22: *Nominal Velocity vs Time: Sixth Scenario*

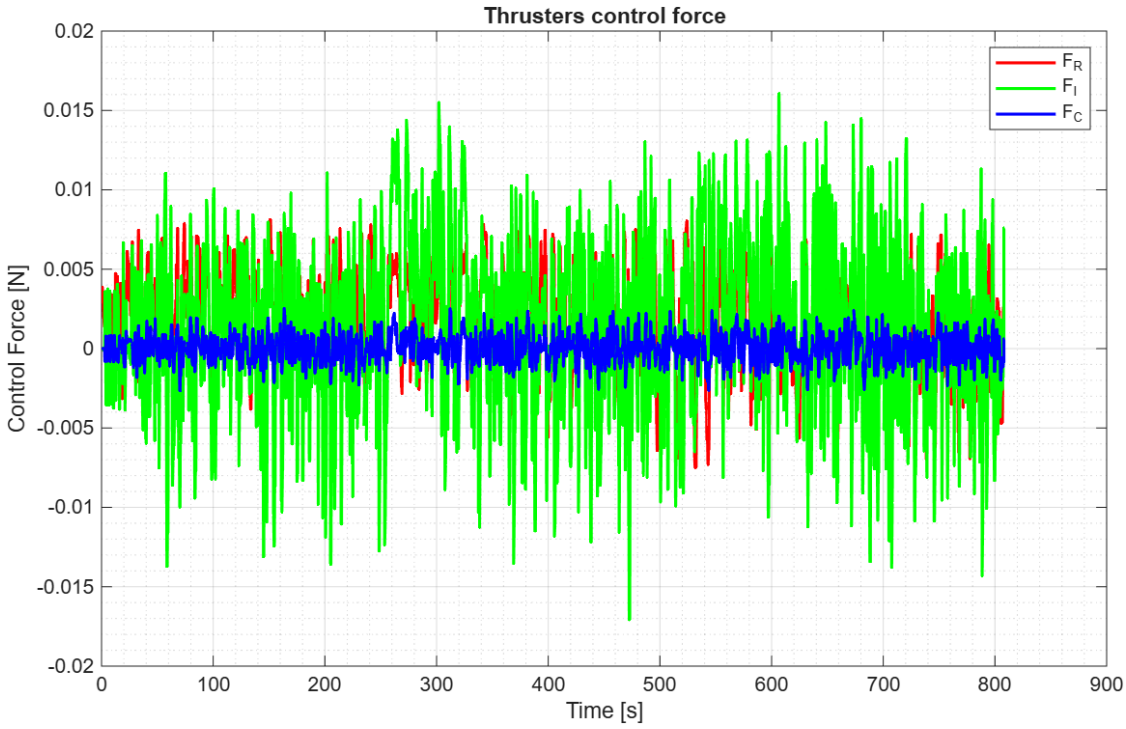



Figure 4.23: *Thrusters control force: Sixth Scenario*

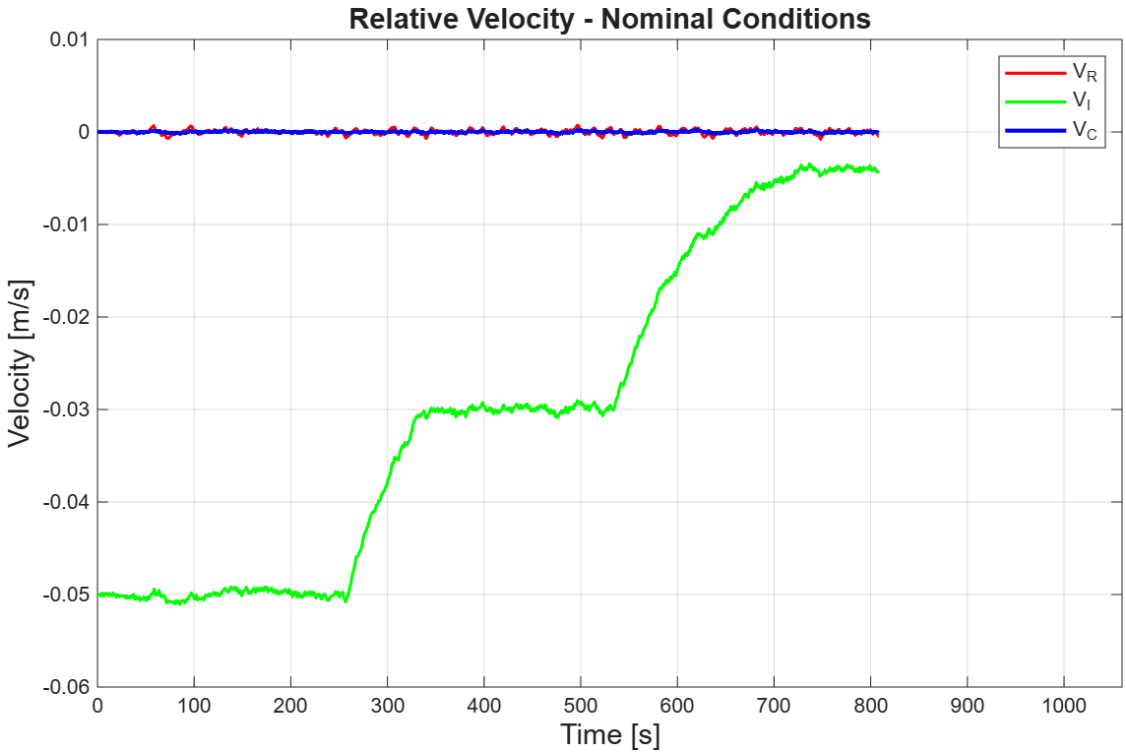



Figure 4.24: *Relative velocity - nominal conditions: Sixth Scenario*

### 4.1.7 Final Comparison

| Case | $\Delta V$ (m/s) | Fuel Consumption (kg) | Duration (s) |
|---|---|---|---|
| 1 | 0.519 | 0.0303 | ~ 1100 |
| 2 | 0.482 | 0.0281 | 1000 |
| 3 | 0.462 | 0.0269 | 950 |
| 4 | 0.584 | 0.0341 | ~ 1200 |
| 5 | 0.450 | 0.0263 | 900 |
| 6 | 0.404 | 0.0236 | ~ 830 |

Table 4.1: Comparison of key-parameters between different scenarios

The most propellant-intensive case is Case 4, which corresponds to the configuration with the earliest deceleration gate among those sharing the same initial In-Track velocity ($v_0$ = 0.05 m/s), combined with the longest manoeuvre duration (~ 1200 s). This result shows that excessive conservatism in the beginning of the deceleration phase does not necessarily translate into control efficiency gains: the extended time spent under active Closed-Loop control against orbital perturbations increases the cumulative $\Delta V$ expenditure despite the smoother command profile.
Cases 5 and 6 achieve the lowest cost (0.450 and 0.404 m/s, respectively) and the shortest durations (900 and 830 s), due to their higher initial or intermediate velocity. However, these configurations introduce distinct critical aspects: the fifth scenario produces In-Track control force peaks exceeding ±0.020 $N$, the highest among all cases. In the sixth case, by modifying the intermediate velocity, the mission time is effectively reduced but on the contrary narrows the velocity margin at the intermediate gate, potentially reducing the robustness of CAM initiation from this phase.
Within the subset of cases sharing identical initial conditions, a trade-off may be found: earlier decelerations (particularly, in case 4) match with higher cost, fuel consumption and duration but enabling possible safety manoeuvres with lower risk.
In conclusion, the parametric study demonstrates that no single configuration dominates across all performance dimensions simultaneously. Case 6 achieves the best propellant efficiency and shortest duration but at the expense of a higher intermediate hold velocity and negative implications, by potentially reducing the abort success probability. Case 3 represents the best compromise, offering the lowest fuel consumption with enhanced safety margins, albeit with the longest duration.

## 4.2 Monte Carlo Analysis

Monte Carlo simulations are used to apply a statistical approach within the study and analysis of a system, in order to validate the results obtained by introducing a small realistic perturbation in the initial conditions. This exact philosophy is used for this thesis to understand the characteristic degree of robustness of the motion profile adopted for the Final Approach phase. The success of the Monte Carlo campaign is deduced by the presence of "failures", that is, circumstances in which the requirements defined within Table (2.4) are not satisfied. Two statistical concepts must be introduced to understand how Monte Carlo simulations have been defined: reliability and confidence level. **Reliability** refers to *the consistency of a measurement, that is, the closeness of the initial estimated values to the subsequent estimated results*. The **Confidence Interval** is *a range of values which is likely to contain the true value of an unknown statistical parameter, such as a population mean* and the **Confidence Level** is *the probability that a calculated confidence interval will contain the true unknown population parameter if a simulation is repeated over time*.

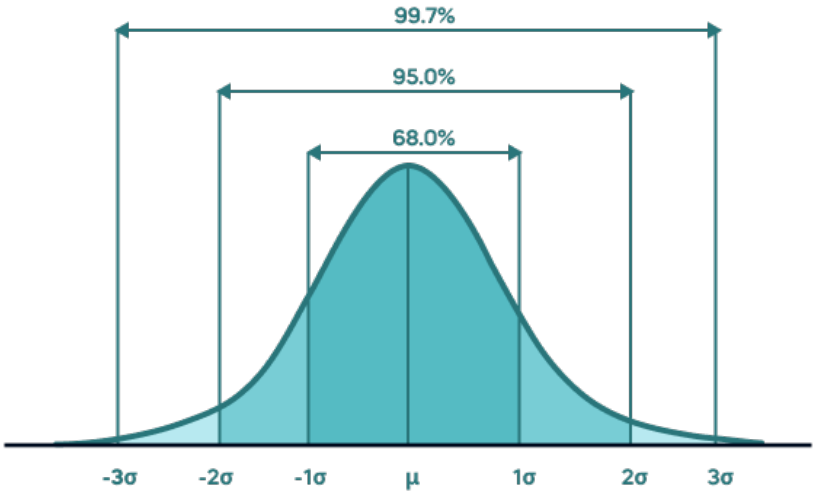



Figure 4.25: Standard normal (Gaussian) distribution showing the fraction of the population contained within 1, 2, and 3 standard deviations ($\sigma$) from the mean ($\mu$): approximately 68.0%, 95.0%, and 99.73%, respectively.

The confidence level may be chosen according to a desired standard deviation, as shown in Figure (4.25). By knowing reliability and confidence level, it is possible to calculate the required amount of simulations to be run thanks to Wilks' Theorem:

$$N = \frac{ln(1-C)}{ln(P)} \tag{4.1}$$

For the purposes of this thesis, a value of 95% for both reliability and confidence level has been considered appropriate. Thus, 58 simulations have been performed with varying initial conditions. In detail, the chosen parameters are the initial position,

| Reliability $R$ | Confidence Level $C$ | Number of Simulations |
|---|---|---|
| 68% | 95% | 8 |
| 95% | 95% | 58 |
| 99.73% | 95% | 1108 |

Table 4.2: Minimum number of Monte Carlo simulations required to demonstrate, with zero failures observed, a given reliability $R$ at a specified confidence level $C$

initial velocity, mass of the chaser, maximum thrust and drag coefficient, as indicated in Table (4.3) below.

| Parameter | Range |
|---|---|
| Initial Position | $p_0 \pm 0.5$ m |
| Initial Velocity | $V_0 \pm 0.02$ m/s |
| Chaser Mass | $20 < m_c < 24$ kg |
| Maximum Thrust | $7.5mN - 10\% < F_{max} < 7.5mN$ |
| Drag Coefficient | $2 < C_D < 2.5$ |

Table 4.3: Ranges of the parameters dispersed in the Monte Carlo simulations, applied around their nominal values to account for uncertainties in the initial state, spacecraft mass, actuator performance and aerodynamic drag.

The velocity profile chosen is the one represented in Figure (4.1) as a generic example of the validity of the method. Since it has been concluded that all the profiles described do not introduce considerable variations in the performances, different Monte Carlo campaigns may be conducted also for the other scenarios.

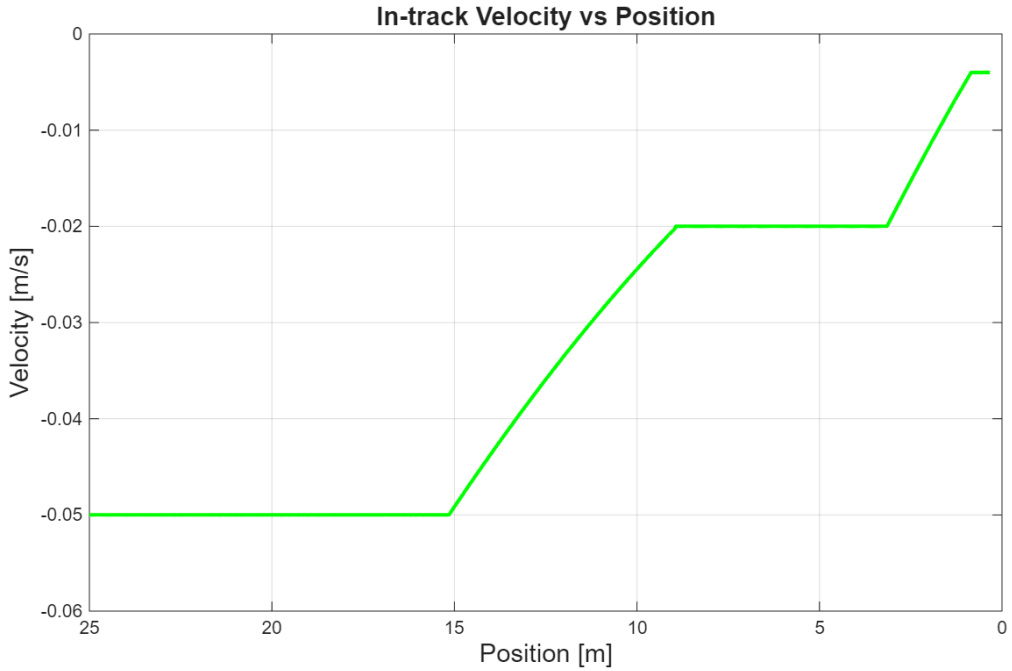



Figure 4.26: *Nominal Velocity vs Position: First Scenario*

The output from the 58 simulations may be appreciated in Figures (4.27) and (4.28).

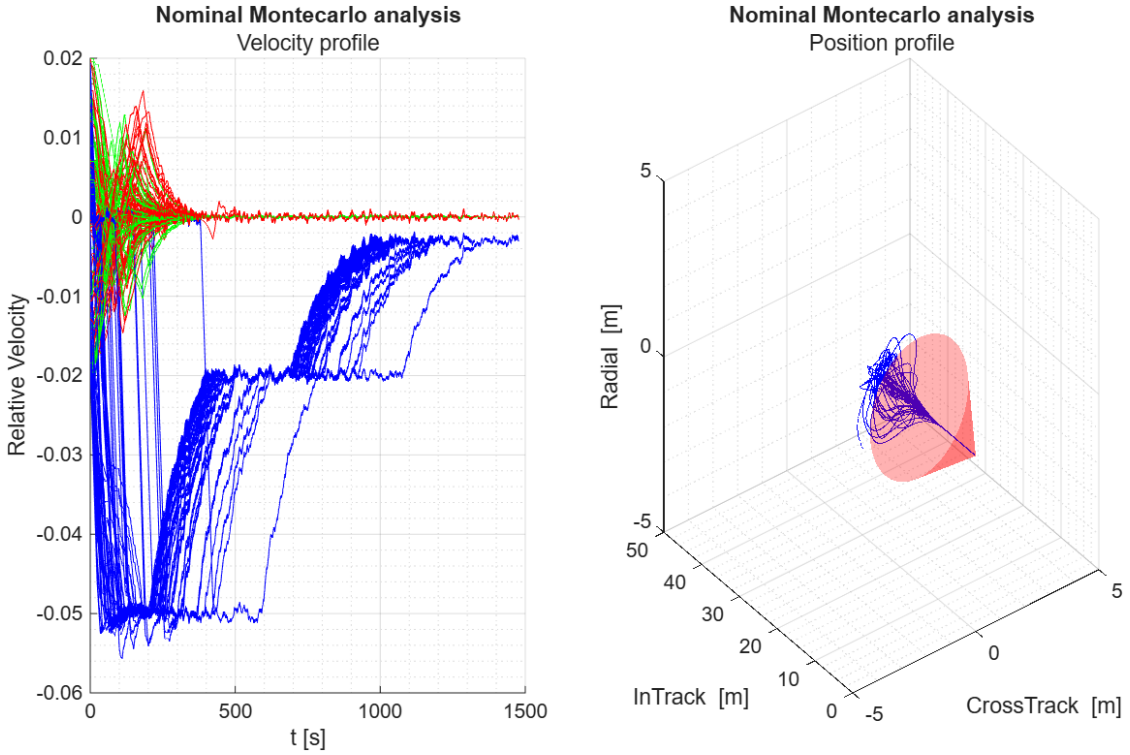



Figure 4.27: Nominal Monte Carlo analysis: velocity and position profiles.
Left: time history of the relative velocity components for all Monte Carlo runs during the final approach phase. Right: corresponding relative position trajectories in the target frame, shown together with the approach cone

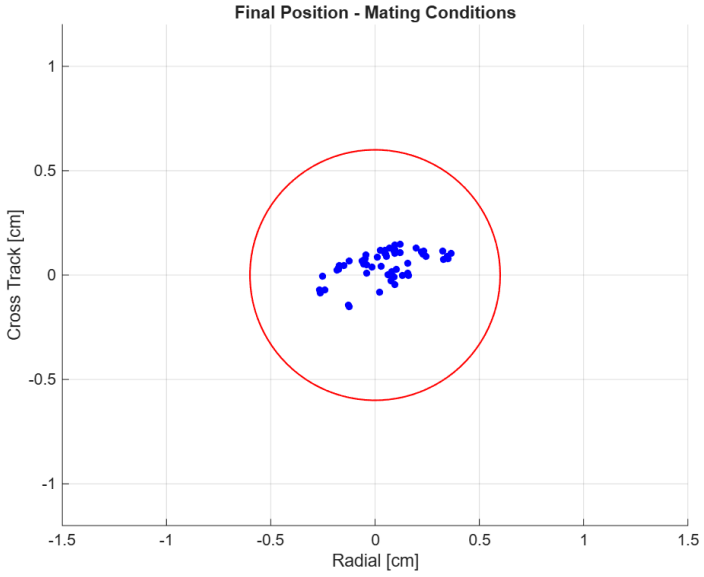



Figure 4.28: Final position dispersion at mating conditions. Scatter plot of the chaser's final relative position at the instant of mechanical mating, for all Monte Carlo samples, compared with a predefined region of 1.2 cm diameter.

The left panel of Figure (4.27) shows the relative velocity components converging toward the origin as the range closes, with the dispersion band narrowing over time. This represents evidence that the control loop is actively damping the precomputed uncertainties. The corresponding position profile, on the right, confirms that, despite this dispersion, all sampled trajectories remain confined within the approach cone, represented by the red surface, which is the primary safety requirement for this phase as stated in Table (2.4): the analyses demonstrate that the GNC subsystem provides adequate margin against corridor violation even under statistically dispersed conditions and not only for the nominal case. The second figure shows the ultimate point of the manoeuvre, i.e., the relative state at the instant of mechanical mating. The red circle represents the maximum acceptable lateral and radial misalignment. All samples falling within this region constitutes further empirical evidence that the GNC system satisfies the docking interface requirement under the assumed uncertainty model.The mean time of the simulation to approach is 1083.26 s, that is, about 18 minutes, in full accordance with the 30-minutes requirement presented in Table (2.4). Certainly, selecting a higher value of reliability will imply a very high number of simulations (about 1100) but will confirm the robustness of the method or otherwise show a non-predictable failure, neglected with only 58 runs. Future works may focus on a wide campaign aimed at testing every velocity profile discussed in Chapter 4.

# Chapter 5

# CAM: Strategies and Implementation

## 5.1 State of the Art

### 5.1.1 General FDIR Architectures and Strategies

Before discussing how CAM may be practically executed, a proper introduction to the subject and all the considerations regarding it is needed.
As already stated, a CAM belongs to the general field of FDIR manoeuvres, id est, the actions that are executed once a fault or an equivalent state of emergency is detected. The literature about it is very broad and, for the purposes of this thesis, a synthesis between the different parts is realized or, at least, is aimed at.
To begin, an interesting perspective proposed in [1] is to firstly understand *how* a fault can be detected and what the exact definition of *fault* is. For sure, a deviation from the planned trajectory and attitude is considered a failure. Together with it, the anomalous behaviour of subsystems and equipment may be regarded as a symptom of failure. Three different levels of malfunction detection are proposed:

- Check of all equipment in terms of temperature, pressure, voltage, et cetera, at the lowest level;
- Check of different criteria GNC-related, e.g. its modes and manoeuvres;
- Check of position, velocity, attitude and angular rates not to exceed safety margins, at the highest level.

The recovery strategy shall begin at the lowest possible level and, if necessary, continue towards an increasingly higher one, with a CAM being the last chance. Their implementation will be discussed further on, while low level strategies are here only presented but not effectively analysed. Figure (3.2) is not actually precise, since the FDIR system is not independent from the equipment and GNC functions, but they mutually interact in order to execute safety actions and realize the different detections above listed. Various options are possible, as regards failure detection within the GNC subsystem. For example, discrepancies between measured and propagated states are considered a major problem, as well as divergence of the navigation filter. At thruster level, possible failures already mentioned regard anomalies with

the boosts duration. Another formal distinction is made between 'soft' and 'hard' failures, the first ones being connected with misleading outputs from sensors, with critical deviations on the long term, and the second ones including serious malfunctions and cease of operations. Soft failures are easily managed considering redundant independent equipment: nonetheless, technical and economical limits do not allow reaching complete coverage of this type of failures. It becomes particularly efficient only in the case of equipment failures, where health status information is available and detection is immediate. Whenever isolation is not possible, redundant processors may be taken into consideration. If none of these previous strategies is possible, only then the interruption of the mission becomes necessary: the different options include the execution of a CAM which increases the risk of a collision, the retreat to a hold point or the shift towards a drift orbit.
Another relevant overview is offered in [21], discussing the challenges encountered while designing the FDIR strategy in Hera's mission by ESA. As before, three steps are identified:

- Detection of *a* failure;
- Identification of the origin of the failure and its isolation, to prevent a chain-effect;
- Recover from the emergency, with different approaches, such as waiting for ground control or actively allowing the autonomous on-board system to find a solution.

Within the mission architecture, a hierarchical FDIR model is proposed as a very common baseline in different spacecraft missions.

- Level 0: Internal failure which does not impact on the global system and is easily recoverable;
- Level 1: Interruption of proper functionality of a component, recovered thanks to a reset or a redundant unit;
- Level 2: Failure of a whole subsystem, not due to a single component, with a consequent reconfiguration needed;
- Level 3: Irrecoverable failure of the central computer, which requires a switch-over to the redundant lane;
- Level 4: Abnormal operation due to a major failure, with loss of at least part of the spacecraft.

Apart from the typical operations expected to be covered by the FDIR architecture, as regards specifically the GNC and propulsion subsystem, the tasks required consist in keeping the attitude of the spacecraft in different operational modes by implementing different cascade of failure detection and reconfiguration. In addition to this, attitude errors are recovered with a mode reconfiguration if the origin of the failure cannot be

properly identified. A thruster failure is recovered by aborting the manoeuvre and by switching into safe mode to ensure the attitude can be properly controlled after the insurgence of the emergency.
A challenge faced while designing the FDIR architecture regards how to properly handle all the inputs coming from different subsystems: the recovery is suggested to be handled only by the central software to avoid having multiple operations executed simultaneously. Different services are part of the recovery operations: *Event Reporting*, notifying the ground station about the anomaly taking place; *Monitoring Service*, which actively detects the failure by monitoring changes in a specific parameter; *Event and Action Service*, providing a proper action as a response for the specific emergency event; *Request Sequence*, when the previous service cannot be executed with a single telecommand.
Zolghadri's article [22] proposes a deep analysis of the state of the art, comparing model-based solutions with actual space applications, and describes three different units within the mission architecture, the so-called Fault Detection & Diagnosis (FDD), Fault Tolerant Control (FTC) and Fault Tolerant Guidance (FTG). The first system should detect abnormal situations with a high rate and discriminate between different emergencies, without being misled by false alarms. On the contrary, both FTC and FTG provide actions for recovery, by continuing the mission with a degraded level of performance or by reshaping its objectives, respectively. As regards space missions, a hierarchical approach is discussed, as in [21]. FDIR strategy is divided in different levels, as well: detection, local or global reconfiguration and prevention. Validation becomes complex since it requires considering all possible emergency situations but, at the time being, satellites are capable of reacting to some known events and possess a certain degree of autonomy, which may be helpful in the case of LEO orbits to reduce ground operations, for example.
Figure (5.1) illustrates the interaction between FDD, FTC and FTG. FTG intervenes only when FTC actions are not sufficient for a safe recovery. All functions collaborate with the GNC subsystem, using data provided from sensors, and generate the necessary control actions.
An appropriate recovery strategy, thus involving the FDD unit, should rely on the existing actuators and sensors and keep stability, with a degraded level of performance during the emergency, if possible. The detection and diagnosis shall happen much before the actual emergency manifests, in order to set up appropriate recovery actions. New control laws may be pre-computed or synthesized aboard. The correctness of the FDD system typically relies on Monte Carlo campaigns which simulate different operative conditions to test the robustness of the recovery strategy. The whole verification process becomes more and more costly with increasing GNC subsystem complexity. An explicit guideline to extract the most suitable FDD and FTC system has not been defined yet, while early attempts of FTG applications show promising results, providing for flexibility and robustness.
FDD techniques are the most consolidated ones and do not affect the system's stability. Moreover, advancement in on-board equipment is a great stimulus for new advanced strategies, which become necessary to detect more subtle type of failures.

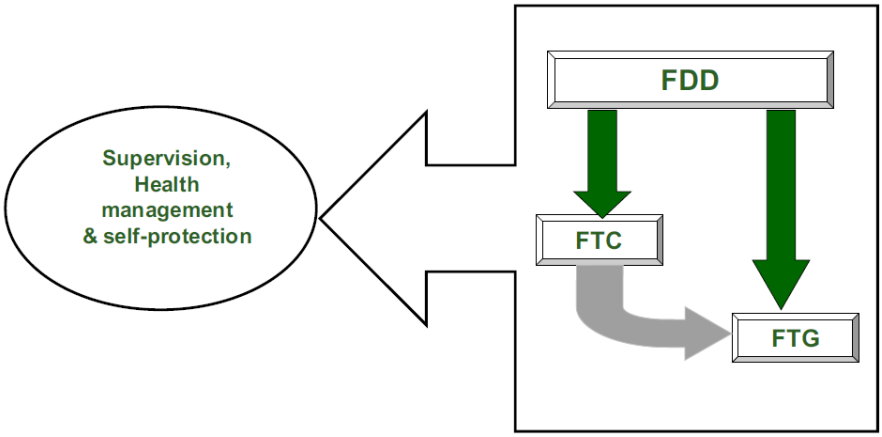



Figure 5.1: Conceptual interaction between the Fault Detection and Diagnosis (FDD), Fault Tolerant Control (FTC), and Fault Tolerant Guidance (FTG) functions within the on-board Supervision, Health Management, and Self-Protection framework. FDD constitutes the initial stage of the chain, providing fault information to both FTC and FTG. FTC is invoked first to preserve system stability and achievable performance through the healthy control resources; if the available control authority proves insufficient, the process escalates to FTG, which accommodates the fault by reshaping the mission objectives (e.g., through trajectory replanning). Adapted from Zolghadri [22].

Certainly, the best solution is one combining in the best way FDD specifications, computational costs and design complexity. On the contrary, FTC systems are seen with skepticisms by modern industry, due to the very long validation process, although many positive results have been provided by the academic community. FTG is yet quite unknown and needs further study, but at the same time represents the scenario with the highest potential.
VISORS [23] presents another example of FDIR implementation within a CubeSat mission in LEO orbit. In this particular context, every fault (GNC and non-GNC related) triggers a response, which can be a role switch between active and passive mode, a transfer to standby or an escape in case of safety violations detected by the GNC safety module. A manoeuvre planner shows the different approaches involved for each mission mode, as illustrated in figure (5.2).
Three types of input determine the response: outputs from navigation sensors, ground commands and FDIR feedback. A manoeuvre buffer allows it to store previously-computed manoeuvres until actual execution, with the possibility of introducing proper modifications or corrections. For the purposes of this thesis, the fourth algorithm is the most relevant one. When safety requirements are violated, the manoeuvre executed aims at increasing the distance along the In-Track direction while reducing the radial gap by analytical conditions on the Relative Orbit Elements (ROE), with just one single impulse. As a consequence, it is hard to find a good target point to be reached without interfering with safety requirements with a single thrust.
Another perspective is offered by Cervone et al. [24]: the fundamental idea behind

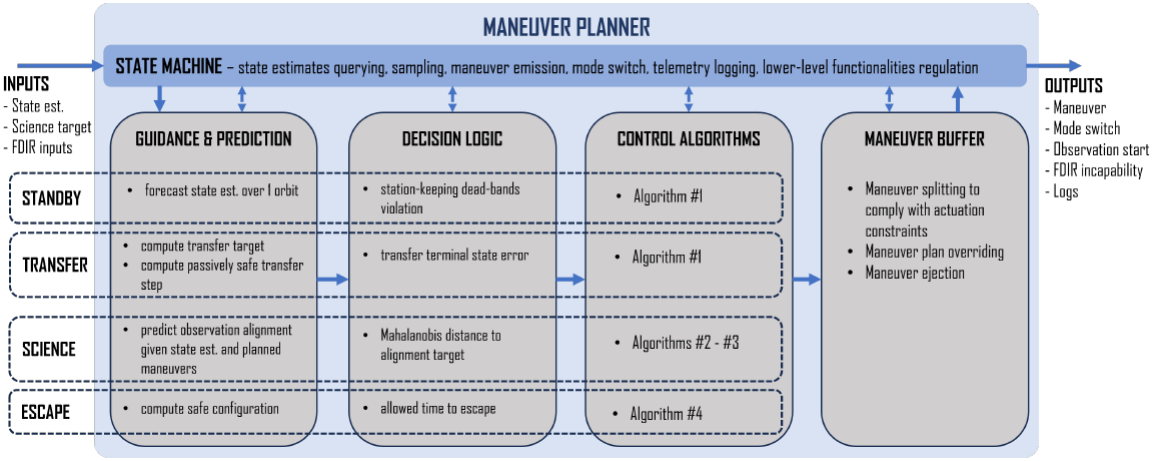



Figure 5.2: Control stack architecture of the VISORS manoeuvre planner, illustrating how each functional layer — Guidance & Prediction, Decision Logic, Control Algorithms, and manoeuvre Buffer — is specialized per mission mode (Standby, Transfer, Science, Escape). A top-level state machine coordinates state estimate querying, manoeuvre emission, mode switching, and telemetry logging across the stack. Each mode pairs a dedicated guidance/prediction step and decision criterion (e.g., station-keeping dead-band violation, Mahalanobis distance to the alignment target, allowed time to escape) with the corresponding control algorithm (Algorithms 1–4), before planned manoeuvres are quantized and buffered to comply with actuation constraints. Adapted from Guffanti et al. [23].

the LUMIO mission architecture is to identify and address every unexpected behaviour with a layered FDIR strategy which consists of five different levels which cover a gradually wider part of the system, from the unit element to the whole spacecraft. One of the most known methodologies is the so-called GAFE (Generics AOCS/GNC Techniques & Design Framework for FDIR), adopted in complex CubeSat missions. Its inputs come from the FDIR requirements and the nominal GNC design and is considered to be robust against uncertainties. The Structural Analysis Tool becomes particularly useful for CubeSat missions: a purely topological analysis makes it possible to build a structural model to connect the existing redundancies with the modelled faults, in order to reach an optimal solution without negatively impacting on the total costs. Table 3 from the same article shows an overview of modern applications highlighting the mission's level of autonomy and the type of FDIR strategy implemented.
A hierarchical strategy is adopted also for the AIM mission (Asteroid Impact Mission) by ESA [25]: from local level, checking the reliability of sensors' outputs, to subsystem level, evaluating data consistency and estimating the collision risk, up to recovery level with the performance of a CAM. For that particular case, the key parameters under investigation are the velocity margin on the escape velocity, the distance after one hour of propagation and the angular margin between the estimated velocity direction and the direction towards the binary system border.
Latachi's et al. [26] discuss an alternative to the hierarchical model, in a way similar to

the strategy already discussed, that is, the *half-satellite strategy* in which no isolation is performed when the emergency state is activated, but on the contrary the redundant units gain authority while waiting for ground control to fix the anomalies detected. Even though this concept is very simple and easily validated, for economical reasons the best option is always the hierarchical subdivision, as it was established for Masat-1 mission. In that particular case the FDIR analysis could only rely on software procedures since hardware redesign was not an option, the mission being based on Components Off-The-Shelf (COTS).

| | Impact of Failure per Level | Fault Detection Techniques per Level | Fault Recovery Techniques per Level |
|---|---|---|---|
| *Level 0: Units* | Unit faults: short currents, overvoltage, over temperature, etc. No impact on subsystem performance. | Local in unit: safety measures and watchdogs; log failures to inform ground. | SW/HW built-in safety functions; local correction. |
| ***Level 1: Subsystems*** | Minor failures: errors/faults of units or communication interfaces. Degraded subsystem performance. | In respective subsystem level: limit checking; diagnosis of housekeeping data, critical parameters control | Command retry, log failures to inform ground, and isolate problem; switch to safe mode. |
| ***Level 2 and 3: Satellite and Mission*** | Major SW/HW subsystem failures: one or more subsystems are affected; partial/total loss of vital subsystems' services. Anomalies of the OBC module or SW. Mission is partially compromised in terms of performance or limited operations. | Several alarms from level 0 and 1, faults on FDIR units, hardware alarms or control software alarms. | Switch to safe mode; subsystems reboots.Subsystem reconfiguration. |

Figure 5.3: Hierarchical structure of the Masat-1 FDIR framework, organized across three levels (Units, Subsystems, Satellite/Mission). For each level, the table reports the typical failure impact, the fault detection techniques employed (ranging from local watchdog-based checks to system-level aggregated alarms), and the corresponding recovery strategies, from local correction up to safe-mode entry and subsystem reconfiguration. Adapted from Latachi et al. (2020) [26].

Figure (5.3) shows some of the mitigation techniques, without considering unit level, since recovery for those scenarios relies on the built-in detection functions. Also for Masat-1 mission, the hierarchical architecture is divided into different levels, the highest one responsible for the spacecraft integrity and safety. It is both a centralized and distributed system, since the decisions are made by the flight planner but the FDIR functions are conjugated with the safety functions at software level. Every component possesses an internal health status check, to be used locally or on ground, and even the data transmitted are verified against anomalies. As previously mentioned, sensor's outputs are checked with a threshold value (highest or lowest acceptable limit). Command consistency, that is, the repetition of the same command, is another strategy if an initially-false alarm is received. When several attempts fail, the alarm reaches higher FDIR levels. When faults involve a subsystem or the whole system, the spacecraft automatically switches to safe mode until next contact with the ground is reached. Table (5.1) summarizes and compares the hierarchical FDIR frameworks discussed above, highlighting for each mission the adopted structure, detection approach, and distinctive strategy. Taken together, these frameworks converge on a common architectural pattern, i.e, a hierarchical or mode-based escalation from local fault detection to system-wide recovery, with a CAM (or an equivalent escape manoeuvre) reserved as the last-chance action. However, each of them addresses the problem from a different angle: Hera, LUMIO and AIM emphasize the architecture of the FDIR chain itself, VISORS focuses on a specific single-impulse escape law constrained by the difficulty of selecting a safe target point, and Masat-1 explores an alternative, non-isolating recovery philosophy driven by cost constraints.

Table 5.1: Comparison of hierarchical FDIR frameworks across different missions, highlighting the detection approach and the distinctive strategy adopted by each.

| **Mission / Ref.** | **Hierarchical structure** | **Detection approach** | **Distinctive strategy** |
|---|---|---|---|
| Hera [21] | Levels 0–4 (from internal recoverable failure up to loss of part of the spacecraft) | Detection → isolation → recovery, managed by central software to avoid concurrent actions | Generic hierarchical model, adopted as a common baseline across several missions |
| VISORS [23] | Non-hierarchical, mode-based (Standby, Transfer, Science, Escape) | Inputs from navigation sensors, ground commands, and FDIR feedback | Single-impulse escape based on Relative Orbit Elements (ROE); difficulty in identifying an effective target point with a single burn |
| LUMIO [24] | 5 levels, from single component up to the whole spacecraft | GAFE (Generics AOCS/GNC Techniques & Design Framework) methodology, with a Structural Analysis Tool to model redundancies | Framework specifically structured for complex CubeSat missions |
| AIM [25] | 3 levels: local (sensors) → subsystem (data consistency, collision risk) → recovery (CAM) | Escape velocity margin, distance after 1h propagation, angular margin | CAM as the last level of the hierarchy, with explicit quantitative criteria |
| Masat-1 [26] | 3 levels: Unit → Subsystem → Satellite/Mission | Internal status checks, transmitted data verification, sensor thresholds | Also proposes the "half-satellite strategy": no isolation, redundant units gain authority while awaiting ground control |

None of these works, however, couples a fully linear, cone-shaped keep-out geometry with an explicit fuel-optimality analysis of the escape manoeuvre itself, nor do they extend the discussion to the subsequent recovery sequence required to bring the chaser back to a nominal operative condition once the emergency is resolved. This thesis aims to address this gap by building on the analytical approach demonstrated for ellipsoidal avoidance regions (as it will be soon discussed for mission PRISMA) and adapting it to the conical geometry of the SROC approach corridor, while additionally defining the sequence of actions required for a safe return to nominal operations.

### 5.1.2 CAM Planning Strategies

Robust optimization has been obtained also for the CAM planning of mission PRISMA developed by the Swedish Space Corporation [27]. It consists of two different techniques, the *Separation Guidance* and the *Nominal Guidance*, which are selected depending on whether the main spacecraft enters an avoidance region centered in the target or not. The first one approximates the scenario of SROC mission since, as it will be explained further on, the emergency manoeuvre shall at first bring the chaser outside the approach cone without intersecting it. The requirements for mission PRISMA are to exit the avoidance region within a specific time, to ensure increasing separation distance while inside the region and, above all, to achieve a passively safe trajectory, due to which the main spacecraft will not re-enter the avoidance region. The computation of the necessary boost involves the initial position of the satellite, the extent of the avoidance region with a safety margin and a pre-established exit time. The effectiveness of the method is reached when different conditions are satisfied:

1. $|D| > 2d$;
2. $\dfrac{D}{|D|} = \dfrac{x_c}{x_c}$;
3. $|\dfrac{A}{D}| < \dfrac{1}{2}$;

where $D$ is the drift, $d$ is the semi-major axis of the region, $x_c$ is the initial centre of motion and $A$ is the amplitude of the oscillation of motion. With a separation time of 1500 seconds, that is, almost a quarter of an orbit, the cost of the manoeuvre was about 0.1 m/s on average and 0.23 m/s at maximum. Chapter 3 from [27] illustrates the analytical formulation of the CAM planning problem as a sequence of linear programs, with a very compact expression in which the dynamics at any future time is a function of the initial state. The three above mentioned requirements are analytically translated into relations which involve the dimension of the avoidance region or the orbital period itself. For a more in-depth discussion, see [27].
An alternative, purely passive approach to collision safety is introduced with the CPOD (CubeSat Proximity Operations Demonstration) mission [28]. Rather than relying on a triggered avoidance manoeuvre, CPOD achieves collision-free operations only with design: small selected differences between the chaser and target orbital

elements naturally produce a relative motion — a so-called "safety ellipse" — that causes the chaser not to cross the target's orbit, even in the event of a missed or failed manoeuvre. When the chaser is actively approaching, this same principle is extended into a "walking safety ellipse," in which the safe ellipse itself changes along the approach direction while still maintaining its collision-free property at every point of the trajectory. With safety being considered directly in the choice of orbital parameters, this approach provides inherent fault tolerance. This passive strategy is conceptually complementary to the active ones previously mentioned. Nonetheless, scenarios involving non-cooperative targets, close-proximity operations, or system faults that displace the vehicle *outside* the nominal safe geometry still require an active FDIR strategy to guarantee safety. The CPOD mission remains a relevant example of passive collision-safety design, and as motivation for why an active CAM capability remains necessary even in systems that already incorporate passively safe orbital design.

A different formalization of a cone-shaped safety region is proposed by D'Ortona et al. [15], within an Adaptive Model Predictive Control framework developed for spacecraft proximity operations. Rather than an ellipsoidal avoidance region as in PRISMA, the chaser is required to remain inside a cone-shaped approach corridor centered on the target, with the constraint expressed as

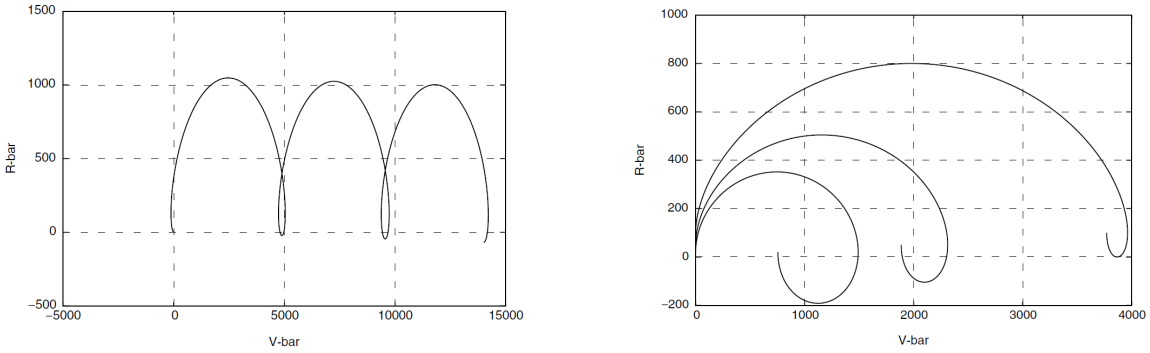
$$\sqrt{p_{R,k}^2 + p_{C,k}^2} < p_{I,k} \cdot \tan\left(\frac{\theta}{2}\right)$$

where $p_R$, $p_I$ and $p_C$ denote the Radial, In-Track and Cross-Track components of the relative position in the RIC frame, and $\theta$ is the cone angle. This constraint originates from the same research group and simulator environment used in this thesis. It is worth noting, however, that the constraint above is not linear: it defines a second-order cone (SOC) region, which remains convex but requires a quadratically-constrained formulation to be enforced within the optimization problem. This differs from the approach adopted in this thesis, where the CAM planning problem requires the chaser to move outside the approach cone rather than remain within it; in this case, the same conical geometry can be expressed directly through linear inequality constraints on the relative state, without resorting to a second-order cone formulation.

Other passive techniques are taken into consideration in [29], [30], [31]. Lastly, a further variation on collision avoidance strategy is presented by González-Rodríguez et al. [32], who propose an active, propellant-less methodology for CubeSats lacking a propulsion system: upon receiving a Conjunction Data Message, the satellite's reaction wheel changes its attitude, thus modifying its cross-sectional area and, consequently, the atmospheric drag force, to induce a small variation in orbital altitude and to avoid a predicted debris conjunction. Unlike the passive CPOD approach, this strategy still relies on an actively triggered and planned manoeuvre; however, since it acts on the ballistic coefficient rather than executing a direct boost, it is only applicable to slow, altitude-based avoidance against cataloged debris over a multi-day window, and is not suited to the short-timescale, propulsive CAM required for close-range operations such as those addressed in this work.

## 5.2 Methodologies and Results

Section (4.5) from [1] offers an overview about the typical type of manoeuvres adopted when executing a CAM. As previously discussed, an active manoeuvre of this type is the necessary solution only when all types of redundancies and passive strategies fail. With this in consideration, the hereby presented study focuses on the feasibility of a CAM within the SROC mission and the robustness of the method applied.
Typically, three different applications are taken into consideration: a tangential boost along the In-Track direction, a radial boost along the R-bar axis and a combined one. For all cases, the parameters on which the manoeuvre depends are the initial relative velocity between chaser and target and the time interval for which the escape trajectory must be collision-free. The latter requirement is easily satisfied considering very small boosts.




(a) *CAM with Tangential Boost* (b) *CAM with Radial Boost*

Figure 5.4: Relative motion trajectories in the V-bar/R-bar plane resulting from impulsive boosts of increasing magnitude, computed from the Clohessy-Wiltshire equations. (Right) Tangential (In-Track) boosts produce a periodic, looping drift pattern along the V-bar direction, with each successive loop displaced further along-track while maintaining a similar radial amplitude. (Left) Radial boosts produce open, teardrop-shaped trajectories whose along-track (V-bar) extent and radial excursion both grow with increasing boost magnitude. The boost is 0.3 m/s in both scenarios. Figure considers increasing initial radial coordinates, $z$ = 20,50,100 m

What makes radial boost particularly unsuitable for the practical case discussed in this thesis is that the advance in the In-Track direction only depends on the initial radial position. Since the chaser is approaching the target along the In-Track axis, the radial gap between the two spacecrafts is minimum. Thus, a radial boost represents a particularly dangerous approach in a perturbed environment because the safety zone around the target may be violated when the chaser completes a revolution. A combination of both boosts may bring positive results but an appropriate combination of thrusts must be evaluated in order to reach a passively safe trajectory.

### 5.2.1 First Iteration

$$\begin{aligned} x(t) &= \left(\frac{4\dot{x_0}}{\omega} - 6z_0\right)\sin(\omega t) - \frac{2\dot{z_0}}{\omega}\cos(\omega t) + (6\omega z_0 - 3\dot{x_0})t + \left(x_0 + \frac{2\dot{z_0}}{\omega}\right) \\ z(t) &= \left(\frac{2\dot{x_0}}{\omega} - 3z_0\right)\cos(\omega t) + \frac{\dot{z_0}}{\omega}\sin(\omega t) + \left(4z_0 - \frac{2\dot{x_0}}{\omega}\right) \end{aligned} \tag{5.1}$$

Equations (5.1) are derived from Eqs. (3.2) without considering disturbances and forces acting on the spacecraft. They represent the first approach adopted in the analysis of CAM manoeuvres. Considering a tangential boost, the following hypotheses are considered:

1. $y_0 = 0$, $z_0 = 0$;
2. $\dot{y}_0 = 0$, $\dot{z}_0 = 0$.

so that Eqs. (5.1) simplify in this form:

$$\begin{aligned} x(t) &= \frac{1}{\omega}\dot{x}_0(4\sin(\omega t) - 3\omega t) + x0 \\ z(t) &= \frac{2}{\omega}\dot{x}_0(\cos(\omega t) - 1) \end{aligned} \tag{5.2}$$

In these equations, $x(t)$ and $z(t)$ represent a specific target point to be reached: ideally, it has been placed in front of the approach cone at $(25, -25 \cdot tan(5°))$ and its function is to guide the chaser away from the target by not crossing the cone as imposed by requirement. For the tangential manoeuvre, the solution is uniquely identified since the only unknown terms are the initial velocity to apply when executing the CAM and the time to reach the target point. $x_0$ varies within a specified range of [25,0] meters, to cover all the potential starting points within the approach cone. The solution is presented in figure (5.5). There are two different behaviours to examine, which in a way can be considered as specular. While approaching the target, the cost of the manoeuvre starts increasing towards a maximum of about 0.325 m/s. Simultaneously, the time required to exit the approach cone is very little, about less than 100 seconds. The opposite scenario manifests when the emergency arise away from the target: the boost required is minimum and thus the temporal interval to reach the point is extremely large, up to more than 800 seconds. Unfortunately, there is no other option than contemplate the results because, as seen in Eqs. (5.2) the solution is defined. In order to add a degree of freedom, a radial boost may be considered together with the tangential one. Only the last hypothesis is changed, so that now $\dot{z}_0 \neq 0$. The Clohessy-Whiltshire equations modify in this other way:

$$\begin{aligned} x(t) &= \frac{1}{\omega}\dot{x}_0(4\sin(\omega t) - 3\omega t) + \frac{2}{\omega}\dot{z}_0(1 - \cos(\omega t)) + x0 \\ z(t) &= \frac{2}{\omega}\dot{x}_0(\cos(\omega t) - 1) + \frac{1}{\omega}\dot{z}_0\sin(\omega t) \end{aligned} \tag{5.3}$$

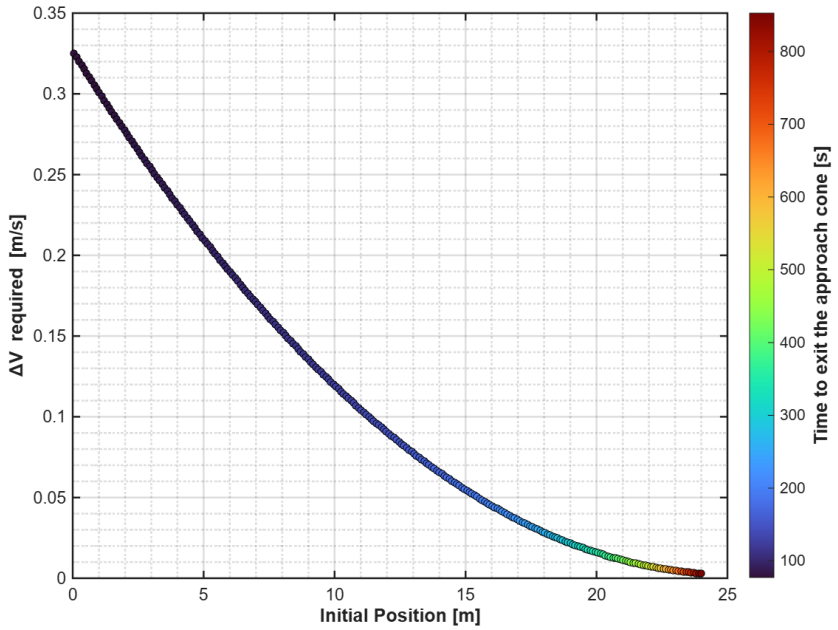



Figure 5.5: Required $\Delta V$ for a tangential escape manoeuvre as a function of the initial position within the approach cone. The color scale indicates the corresponding time required to exit the approach cone. As the initial position moves closer to the cone boundary, the required $\Delta V$ decreases monotonically, while the exit time increases correspondingly, reflecting the reduced propellant cost, at the expense of a longer escape duration, for manoeuvres initiated closer to the safety boundary.

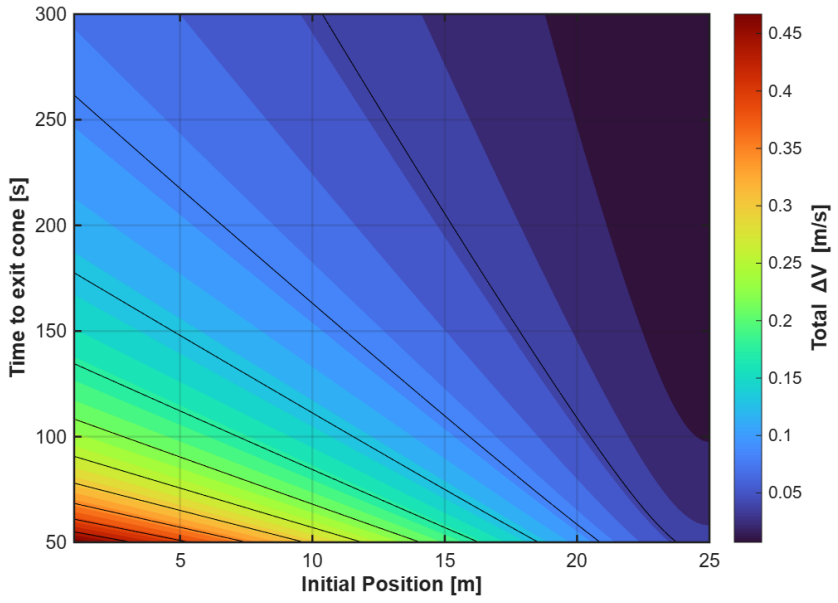



Figure 5.6: Total $\Delta V$ required for a combined (Radial + In-Track) escape manoeuvre, as a function of the initial position within the approach cone and the imposed time to exit the cone. As expected, the required $\Delta V$ decreases as the initial position approaches the cone boundary and as the allowed exit time increases, with the overlaid contour lines highlighting combinations of position and exit time that yield equal propellant cost.

A new unknown term is introduced and, consequently, there are infinite solutions to the problem. This aspect becomes particularly relevant when considering the idea of combining time and propellant costs in order to obtain a final manoeuvre which is the best compromise between the two. For this particular case, the time to exit the cones is pre-selected in an interval from 50 to 300 seconds and the two boosts are immediately obtained. The results are shown in the 3D plot of Figure (5.6). The *Total* $\Delta V$ bar is a sum of both impulses and ranges from 0 m/s to 0.45 m/s, so the upper limit is much higher than the one in Figure (5.5): this unfortunate circumstance arises when very close to the target and with an imposed very short time interval at disposition. Nonetheless, thanks to the combined approach, cheaper solutions are present when considering more minutes to allow the exit. Thus, the high upper limit is just a very particular scenario which of course will not be chosen, not only for the exorbitant costs but also because of the thruster plumes that may cause damage on the target's surface. Figure (5.7) shows a hypothetical trade-off between the two techniques above presented. Depending on the position within the approach cone, a manoeuvre is selected rather than the other one. In details, the constraints regard the total cost of the manoeuvre. For the central part, the tangential manoeuvre is adopted until an upper limit of 0.15 m/s or a lower one coinciding with a maximum interval of 200 seconds are reached. Then, the combined manoeuvre is adopted maintaining the same $\Delta V$ reached with the tangential boost, that is, 0.15 m/s for the upper plateau and 0.06 m/s for the lower one. This represents a practical example of a possible trade-off between cost and time where both parameters do not exceed a fixed value in order to make the manoeuvre as smooth as possible. Of course, this comes only as a proposal and will not be considered as an official choice. The relevant part is understanding how both techniques may be conjugated into one.

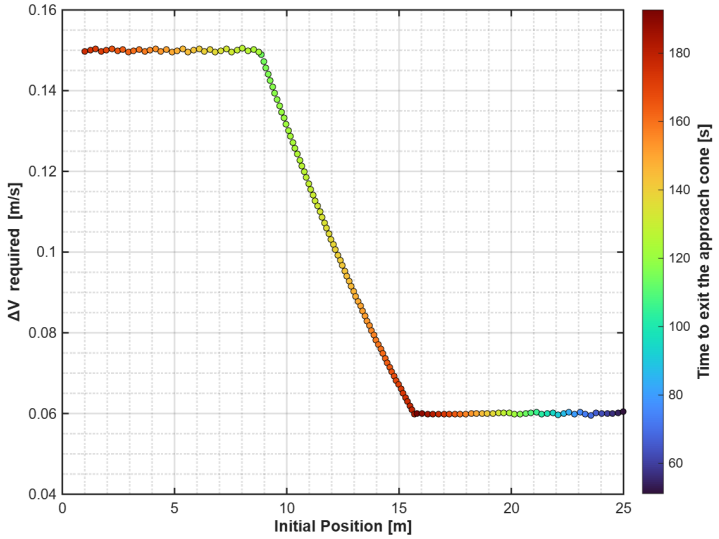
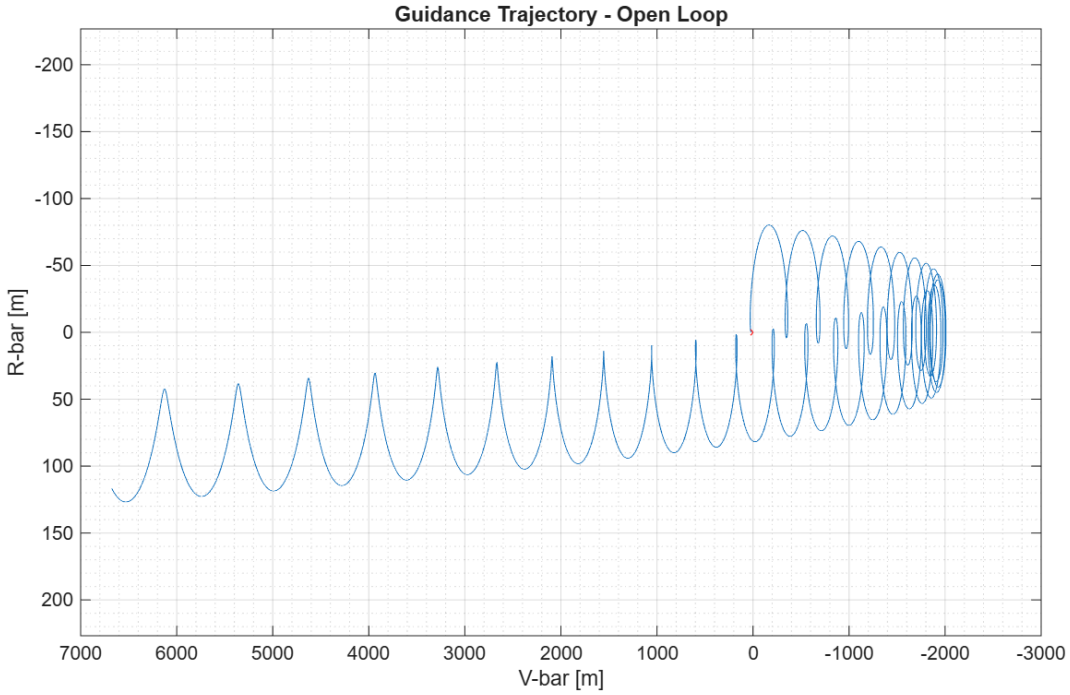
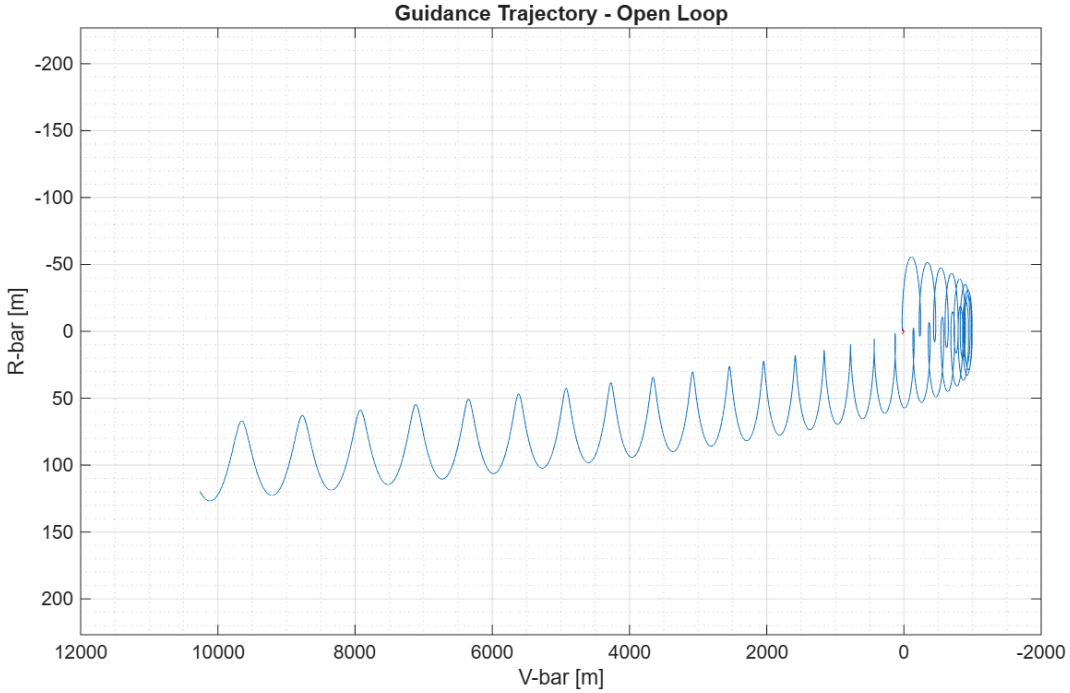



Figure 5.7: Trade-off between the combined and purely tangential escape strategies discussed above, selected at each initial position. Two constant plateaus are the result of an imposed maximum cost for the manoeuvre (upper one) or a maximum exit time interval




(a) *Tangential Boost - Trajectory after 86400 s*




(b) *Combined Boost - Trajectory after 86400 s*

Figure 5.8: Open-Loop propagation of the guidance trajectory over one orbital day (86400 s) in the In-Track/Radial plane, starting from the escape manoeuvre near the origin. (a) Tangential-only manoeuvre: the chaser drifts in the negative V-bar direction, exhibiting a qualitatively similar growth in oscillation amplitude with increasing distance. (b) Combined manoeuvre: the chaser drifts monotonically in the positive V-bar direction, with the radial oscillation amplitude growing steadily as the along-track distance increases.

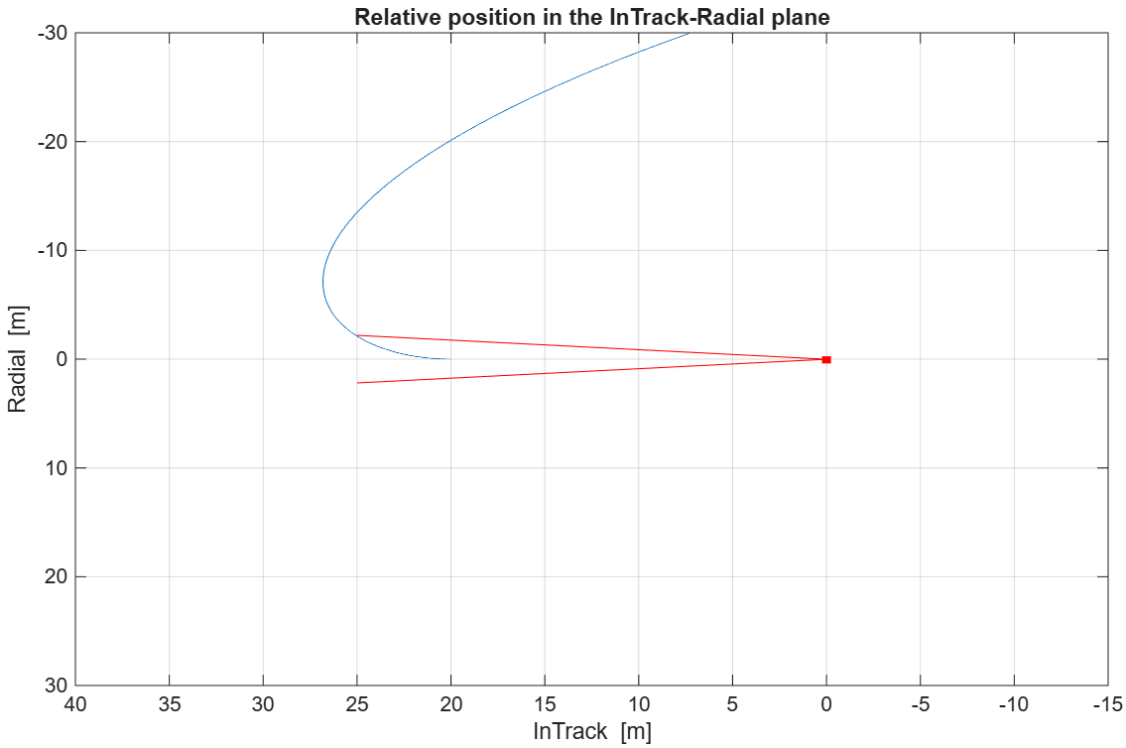



Figure 5.9: Detail of the chaser's trajectory in Open-Loop configuration with a combined thrust, whose complete evolution in 86400 s is shown in Figure (5.8b)

Figure (5.8) shows the long-term comparison between the chaser's trajectory after a tangential or combined boost. The addition of a radial component allows the spacecraft to reach further positive In-Track coordinates, while the negative drift is reduced with respect to the only tangential boost. The initial starting point is for both scenarios equal to [20, 0]. This simulation is helpful not only to compare the two approaches but also, and most importantly, to conclude that this type of manoeuvre does not guarantee that the Keep-Out Zone (KOZ) is not violated. The definition of KOZ around the target spacecraft constitutes a fundamental safety requirement in any close-proximity operation, as it establishes a protected volume that the chaser must not violate under any circumstance, whether during nominal rendezvous operations or especially in the presence of anomalies. The fact that the chaser returns towards the starting point is a very dangerous behaviour: the scenarios displayed in Figure (5.8) represent a very lucky circumstance but it cannot be generalized to every possible initial condition: a MonteCarlo campaign shall be carried out in order to test the robustness of the method. Nonetheless, the ideal hypotheses introduced with this analysis (first of all, the absence of perturbations) make the method not suitable for a real application. Nonetheless, it represents a very good starting point to examine the evolution of relative trajectories and handle with the equations of motion.

### 5.2.2 Second Iteration

The following step aims at introducing the same logic so far adopted *within* the Simulink code. The previous strategy, in fact, implied boosts pre-computed in the MATLAB Workspace and were considered not appropriate for a real application, not only because the actual starting point after the activation of the emergency state does not lie precisely on the In-Track axis (as presumed for hypothesis) but also because the manoeuvre itself and its evolution strongly depends on the initial conditions, as observable in Eqs. (5.1). Thus, the real and correct impulse is now computed in the Simulink environment within a so-called *Stateflow* [33]. Among the tools available, Stateflow proves particularly well-suited to modelling the logical and temporal behaviour of FDIR architectures. Rather than describing continuous dynamics, Stateflow allows it to represent a system as a finite set of discrete states, together with the transitions and conditions that rule the switching between them. This makes it particularly useful for capturing the decision-making strategy of a spacecraft, where the system must react to specific triggering events. In the context of this work, Stateflow was employed to distinguish between the nominal operational state and the emergency state associated with the activation of a CAM, allowing the model to clearly separate the two behavioral regimes and to define the conditions under which the transition between them occurs. Beyond this binary distinction, the tool's temporal logic, through parameters such as timers, event-driven transitions and time-dependent conditions, makes it possible to give precise and specific instructions within each state, for instance triggering manoeuvre commands only after a defined time interval has elapsed since the CAM was activated. This combination of discrete-state representation makes Stateflow especially effective for modeling CAM-related logic, where the correct sequencing and timing of actions is often what determines the safety and effectiveness of the manoeuvre.
Once the emergency is triggered, the necessary $\Delta V$ is computed within the emergency state itself by a simple inverse matrix computation, which provides for a tangential and radial impulses, given a 150 seconds time interval to reach the first obligatory Way-Point. Differently from the previous case, its location has been slightly changed from $(25, -25 \cdot tan(5°))$ to $(30, -30 \cdot tan(3°))$ as a first attempt to apply a conservative approach: further investigation will prove that the location of the Way-Point is not so relevant for the respect of the safety requirements, as it will be discussed for the following iterations. An Open-Loop approach is considered: the system furnishes a thrust whose duration is computed with the following simple formulas:

$$\begin{aligned} t^x_{burn} &= m_c \frac{|dVx|}{F_{max}} \\ t^z_{burn} &= m_c \frac{|dVz|}{F_{max}} \end{aligned} \tag{5.4}$$

Once the impulse is concluded, the system simply lets the chaser's state evolve following the output given by the *Orbit Propagator Numerical (High Precision)* Simulink block: as part of the Aerospace Blockset, it determines the spacecraft's position and velocity at each time step through direct numerical integration of the equations of motion,

rather than relying on the analytical Kepler universal variable formulation used by its faster, unperturbed counterpart. This approach allows the block to account for a more complete central-body gravity model, as well as additional perturbing effects such as atmospheric drag, third-body gravitational attraction, and solar radiation pressure, at the cost of increased computational load compared to the simplified Kepler propagation [34]. The High Precision propagator was secondly replaced by another one, the so-called Numerical EXact Unified Simulator (NEXUS) developed by Luca Niero. In detail, the model implements a numerical orbit propagator for a single spacecraft, integrating the equations of motion in the ECI frame starting from a set of initial conditions computed beforehand in MATLAB. The model is organized around a central block that aggregates the perturbing accelerations acting on the satellite, such as the Earth's non-spherical gravity field, atmospheric drag, solar radiation pressure, and third-body attraction from the Sun and Moon, together with an eclipse-conditioning logic and any external control input. The resulting total acceleration is enters a numerical integrator that propagates position and velocity over time. The model has already been solidly validated and its robustness is not object of this thesis. More pieces of information are presented in Appendix (C). A very brief Monte Carlo campaign considering 25 runs, each lasting 1000 seconds, shown in Figure (5.10), had the objective to show the evolution of the chaser's orbit. The blue dot represents the Way-Point with respect to which the ideal boost has been computed and it is clearly observable that in no circumstance the chaser managed to actually transit through that point. Secondly, and most importantly, a new hazard is introduced with the real-time approach: differently from Figure (5.9), in which the trajectory is initially tangent to the horizontal axis, Figure (5.10) exhibits a short transient radial excursion before diverging along the escape direction.
This behavior directly derives from the intrinsic In-Track/Radial coupling of the linearized relative motion described by the Clohessy-Wiltshire equations, which introduces an epicyclic component in addition to the secular In-Track drift induced by the impulsive manoeuvre. For this reason, it is necessary to pay attention not to violate the Approach Cone. It represents the only admissible corridor through which the chaser is allowed to approach the target, its geometry being specifically designed to constrain the relative trajectory and to guarantee that, in case of a missed approach or an aborted manoeuvre, the chaser's natural drift does not result in a collision. Ensuring that the relative trajectory remains within the boundaries of this cone is therefore not merely a matter of mission performance, but a critical safety constraint: any excursion outside the cone would place the chaser on a trajectory potentially intersecting the KOZ, with no guarantee of a safe recovery. Beyond the purely dynamical safety aspect, the approach cone is also typically aligned with the field of view of the target's relative navigation sensors: departing from this corridor means that the chaser may exit the region within which it can be tracked and observed by the target, effectively breaking the visibility link between the two spacecrafts. This directly compromises the relative navigation solution, since the sensor suite responsible for estimating the chaser's position and velocity relative to the target may lose contact or provide degraded measurements once the chaser leaves the target's sens-

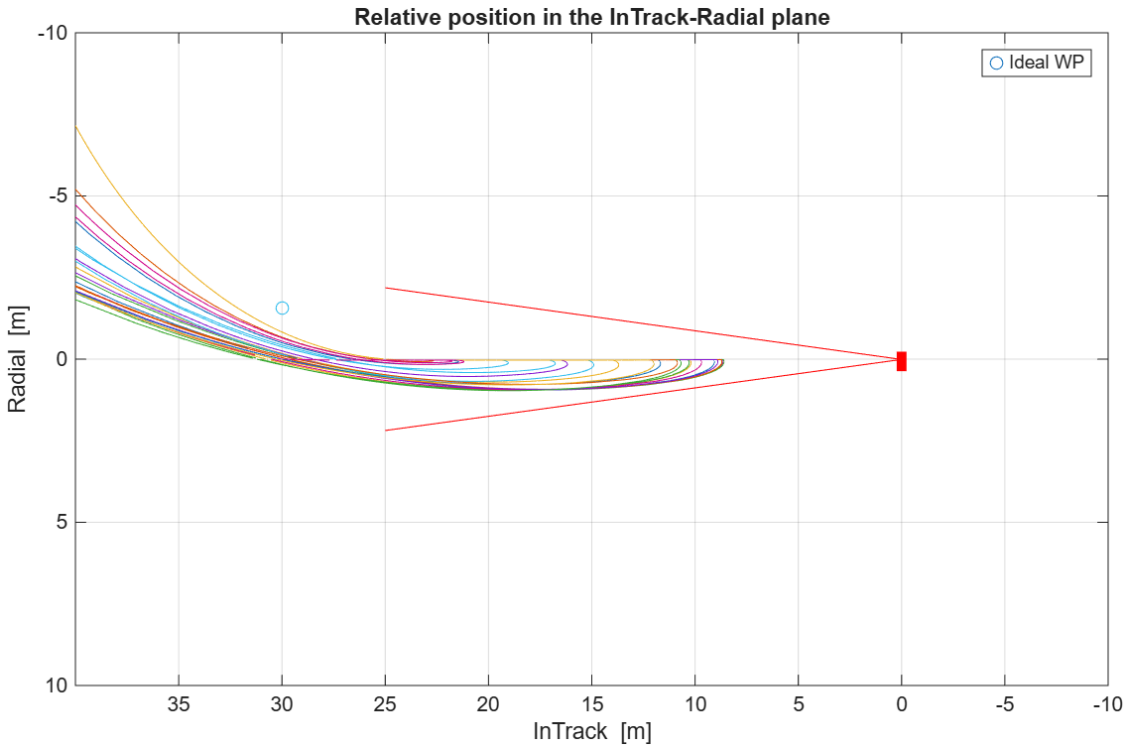



Figure 5.10: Relative trajectories of the chaser in the InTrack-Radial plane for a set of simulation runs in which the CAM is triggered at a random InTrack position within the 6-25 m range. The required impulsive $\Delta V$ is computed directly within the Stateflow logic; the subsequent evolution of each trajectory is then obtained by propagating the post-burn state through the Orbit Propagator Numerical (high precision) block. The resulting spread of curves illustrates how the trajectory shape and its distance from the approach cone boundary vary depending on the InTrack position at which the manoeuvre is triggered.

ing coverage. This requirement becomes particularly significant when considered in an FDIR perspective, since the very purpose of the CAM logics implemented in this work is to detect anomalous or hazardous conditions and to autonomously redirect the chaser toward a safe departure trajectory. Consequently, the escape manoeuvre must be designed not only to increase the relative distance from the target, but to do so while remaining consistent with the approach cone geometry, ensuring that the corrective action commanded does not itself introduce a violation of the safety boundaries it is meant to protect, nor a loss of the relative navigation capability on which the safety of the entire proximity operation depends.
Introducing a Closed-Loop control strategy may become a safer option since it is possible to effectively direct the chaser's trajectory towards the Way-Point: an on-board implemented function defines the ideal trajectory the chaser is constrained to follow. The position of the chaser reaches a point belonging to the negative horizontal axis (which can be defined as a second Way-Point), after which a second impulse is computed to return in front of the target, to a location arbitrarily placed at the coordinates (50,0): the reason is purely connected with the preliminary study and

analysis of the relative dynamics during the selection of the most appropriate CAM strategy. Secondary analyses may individuate the final destination point as one of the Hold Points.
The control approach may be appreciated in Figures (5.11) and (5.12). Particularly, Figure (5.11) illustrates the first phase of the escape manoeuvre: a controlled approach is capable of effectively reducing the radial excursion, thus satisfying the safety constraint so that the approach cone is not violated anymore. In addition to this, the suggested transit is now completely respected and no violation is observable. When observing the same image on the big scale, it is immediate to notice a similar pattern for all trajectories. The horizontal axis is crossed at an In-Track coordinate that widely varies from -300 to -1300 meters circa, covering more than 1 kilometer: this different behaviour is reasonably connected with the initial trigger, which may be activated very close to the target and thus require a higher impulse to exit the cone without touching it. The return trajectory to the point (50,0) is also influenced by the timing of the trigger.

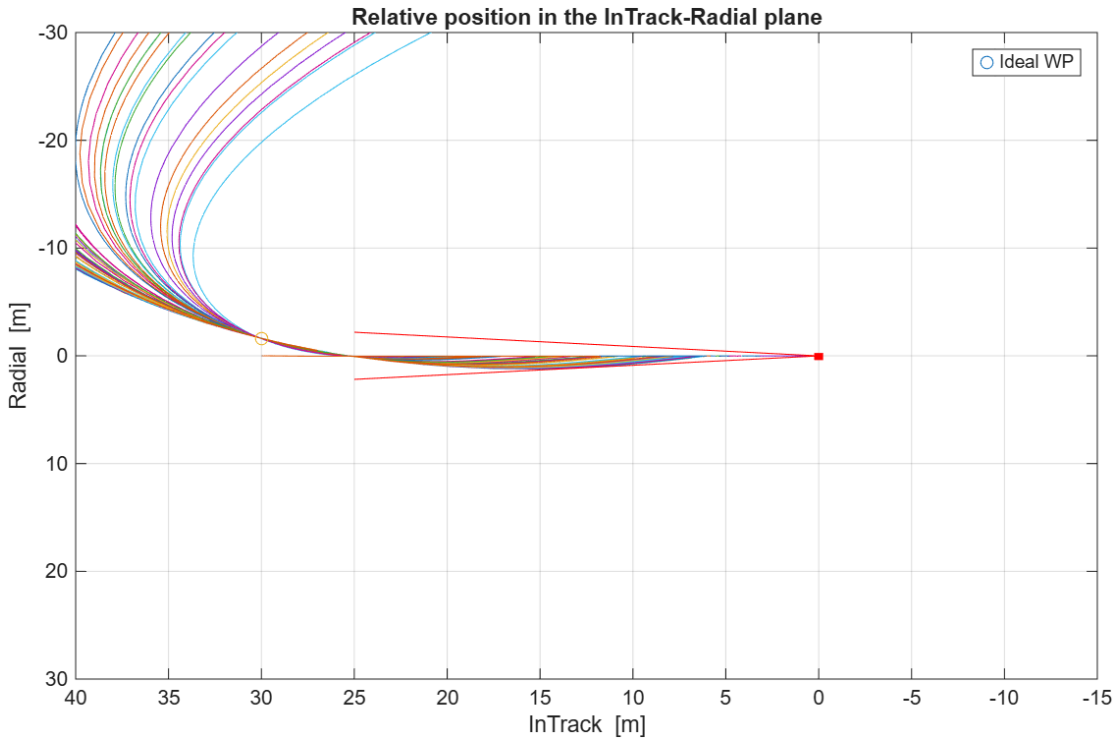



Figure 5.11: Escape trajectory with Closed-Loop strategy for Monte Carlo simulations: zoom on the Approach Cone to show the reduced radial excursion

Figure (5.13) shows the total cost of every single escape manoeuvre, showing an almost-linearly growing trend for initial In-Track positions ranging from 25 to 6 meters. This result mirrors what anticipated when discussing Figure (5.12), that is, very internal points require stronger impulses to escape the approach cone without intersecting it. This concept helps introducing a new parameter that will be one of the objects discussed in the remaining part of this chapter: the point of no return. With this expression, the aim is to define a precise location after which a CAM manoeuvre

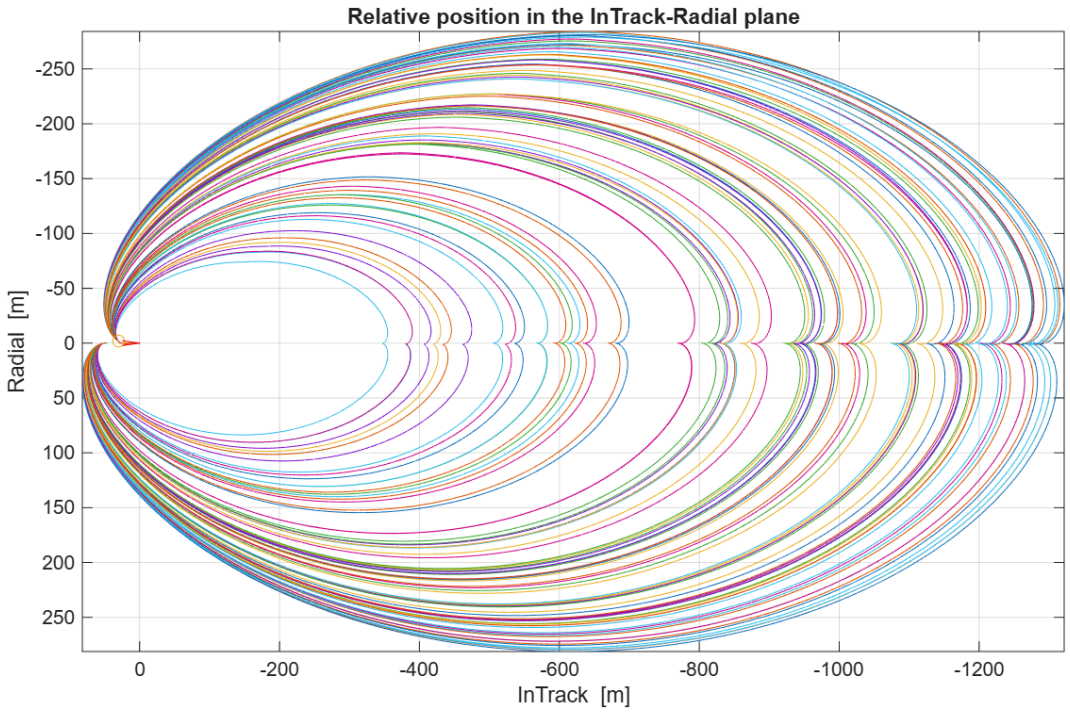



Figure 5.12: Escape trajectory with Closed-Loop strategy for Monte Carlo simulations: view on the complete manoeuvre, with transit behind the target and arrival at (50,0)

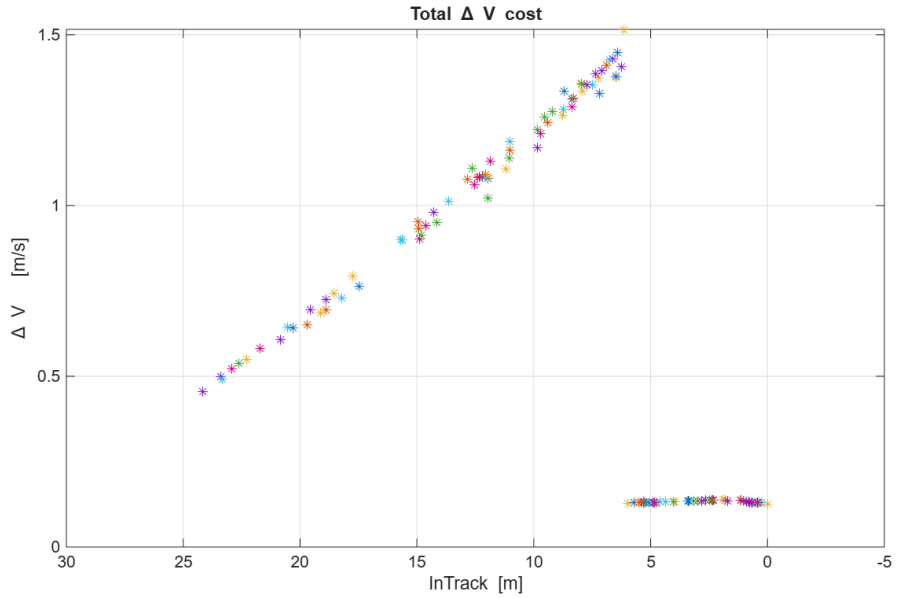
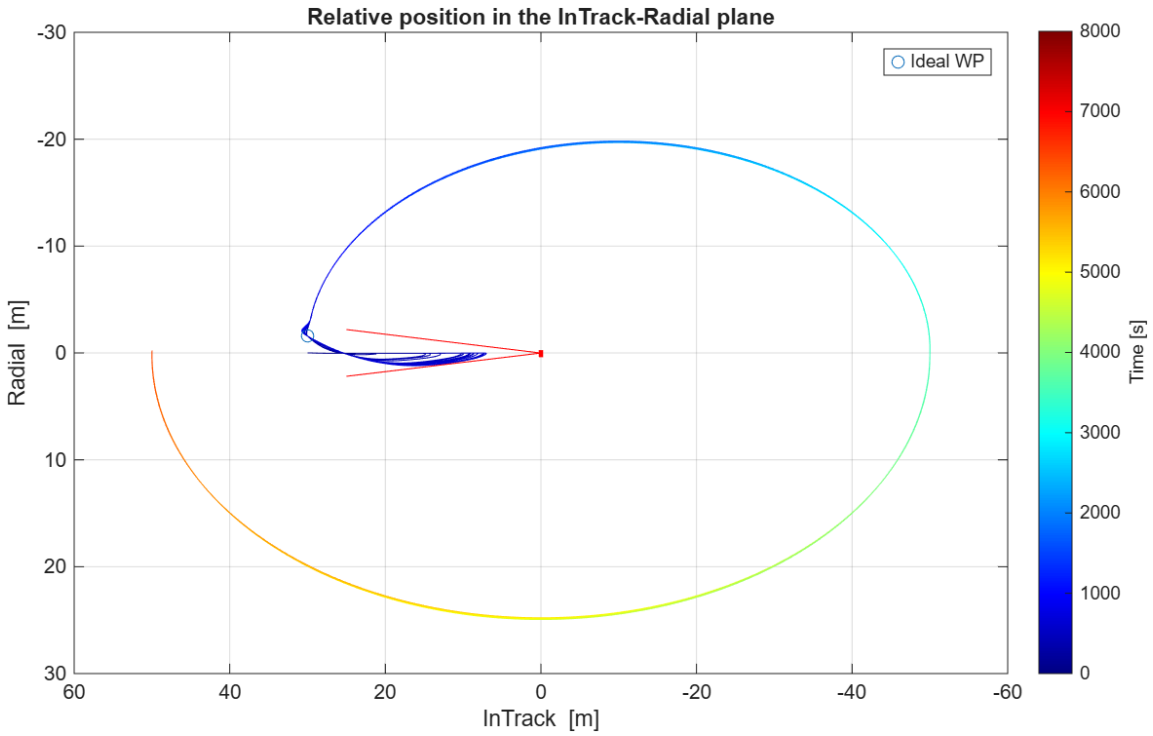
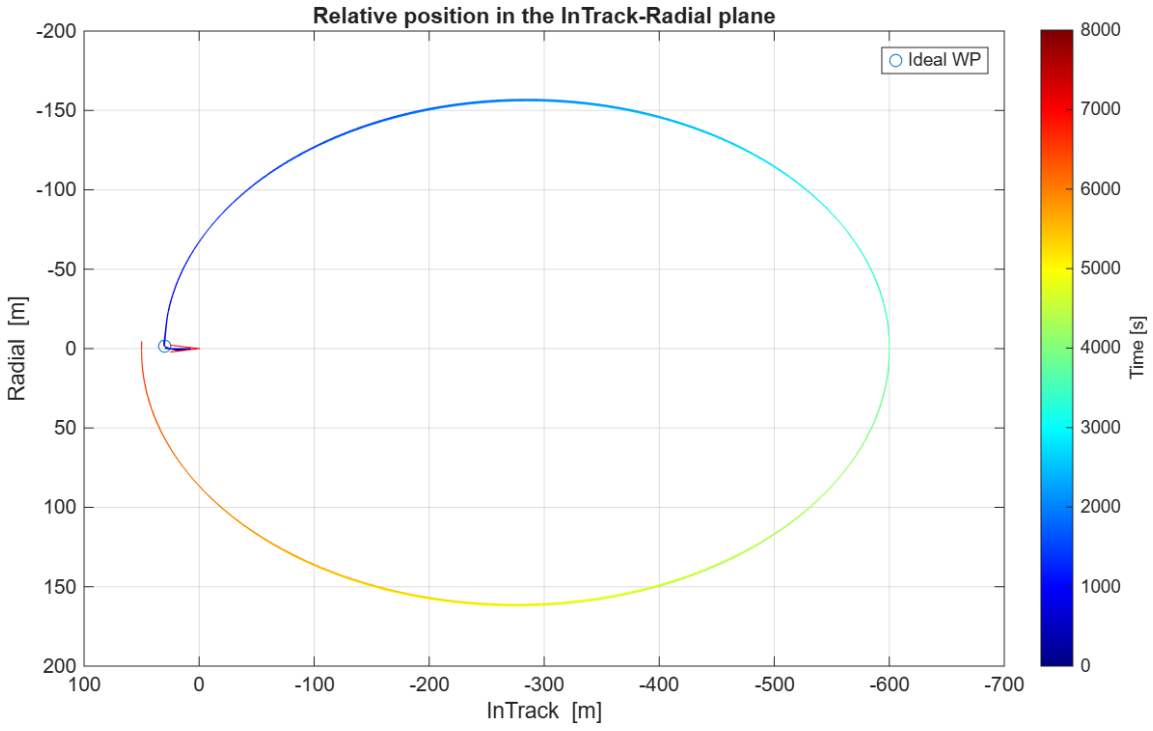



Figure 5.13: Total impulsive cost as a function of the InTrack position at which the CAM is triggered, for the full set of simulation runs. Two distinct regimes can be observed: for triggering positions between 25 m and 6 m InTrack, the total $\Delta V$ increases nearly linearly as the triggering point moves closer to the target. Below 6 m InTrack, however, the total cost drops and settles onto a nearly constant value across all runs.

is no longer executable. When introducing FDIR strategies and orbital disturbances, thruster plumes were discussed to state that thrusts from the chaser become dangerous for the target's integrity when commanded too close to the docking port. In a very naive way, a preliminary definition of point of no-return was already qualitatively given. Considering actual escape manoeuvres, a new perspective arises, moving this point even further to respect the constraint connected with the approach cone. Thus, not only is the location of the no-return point depending on the minimum distance for which thruster plumes become harmless, but also it is directly linked to the maximum admissible radial excursion of the escape trajectory. For this Monte Carlo campaign, the coordinates (6,0) represent only a preliminary suggestion and must not be considered as a general statement. Before indicating a definitive position for the no-return point, the most suitable CAM strategy shall be defined first.
An upgrade from the previous strategy includes the definition of a common point of transit behind the target. A simple example is illustrated in Figure (5.14), in which few simulations have been run to show the impact of selecting a close versus far transit point. As expected, even though the two plots appear qualitatively similar, the main difference arises for the total cost of the manoeuvre, which ranges from a maximum value of about 0.5 m/s to a peak of more than 1.2 m/s when the objective is reaching a further destination, as observable in Figure (5.15). As regards the total time of the simulations, both arcs are designed to be covered in about 2839 seconds each, that is, half of the target's orbital period, in order to perform a smooth transition when crossing the negative In-Track axis and giving the last impulse. Of course, this parameter may arbitrarily vary with new imposed requirements, thus it is not fixed. The new perspective presented with this comparison aims at offering a prelimary idea of expected costs and times of a potentially adoptable strategy for Closed-Loop architectures.
The discussion developed so far was implicitly founded on a rather strong assumption, that is, once a collision hazard is detected, the chaser retains the ability to close the control loop around its own relative state, continuously estimating its position and velocity with respect to the target and correcting its trajectory. This is precisely what makes a Closed-Loop CAM strategy attractive from a performance perspective: the manoeuvre can be indeed planned, monitored, and, if necessary, re-planned in real time on the basis of updated navigation information. However, this capability is not free: it presupposes that the spacecraft's GNC subsystem is, at the moment the emergency is triggered, still capable of producing a valid state estimate and of translating it into commands. In other words, a Closed-Loop architecture is only reliable as long as the GNC subsystem can sustain it. This assumption, while reasonable for a broad class of anomalies, becomes insufficient, precisely in the scenarios where the consequences of failure are most severe. It is useful, at this point, to return to the hierarchical view of spacecraft emergencies that has already been introduced in the context of the FDIR architecture at the beginning of this chapter.
Emergencies affecting a spacecraft's safety span a wide spectrum of severity, from transient sensor glitches or minor actuator degradations, which leave the GNC chain substantially intact and merely require a reconfiguration of the control authority or

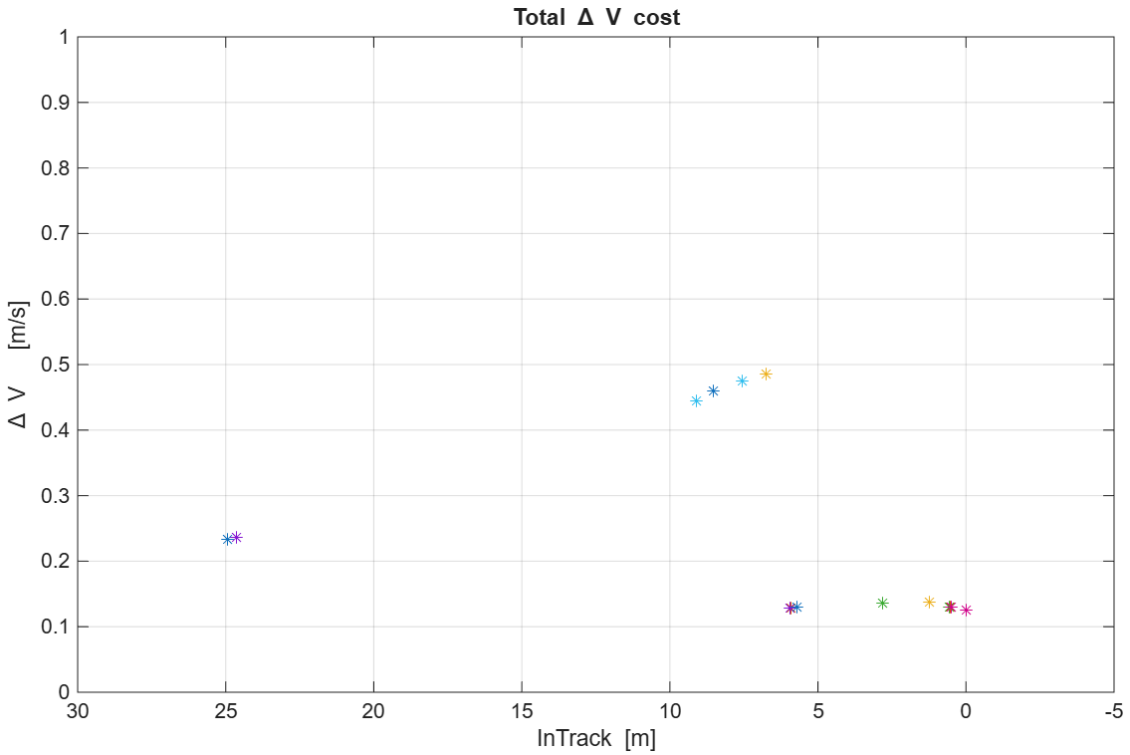



(a) *CAM trajectory with second Way Point at (-50,0)*

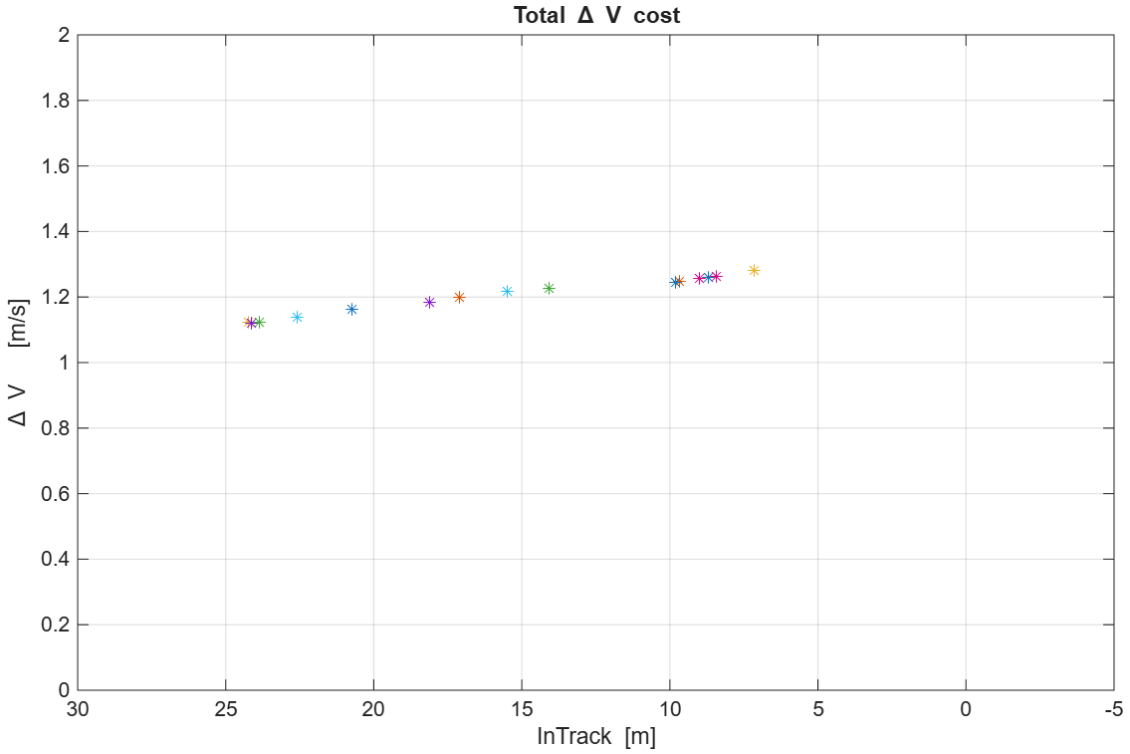



(b) *CAM trajectory with second Way Point at (-600,0)*

Figure 5.14: Comparison between escape trajectories for ideal Closed-Loop manoeuvres having a second point of transit behind the target at different coordinates, respectively at (-50,0) for Figure (5.14a) and (-600,0) for Figure (5.14b)




(a) *CAM trajectory with second Way Point at (-50,0)*




(b) *CAM trajectory with second Way Point at (-600,0)*

Figure 5.15: Comparison between costs for ideal Closed-Loop trajectories having a second point of transit behind the target at different coordinates, respectively at (-50,0) for Figure (5.14a) and (-600,0 for Figure (5.14b))

a temporary tightening of safety margins, to catastrophic failures in which the navigation function itself is lost, either because the relative sensor has failed, the onboard processing unit responsible for state estimation has become unavailable, or because the attitude control system can no longer guarantee the pointing accuracy that relative navigation depends upon. It is precisely this second class of anomalies, the ones in which the GNC subsystem itself is compromised, that represents the true worst-case scenario from a safety perspective, and it is exactly in this class that a Closed-Loop CAM strategy loses its meaning. This observation suggests a natural refinement of the FDIR architecture discussed earlier. Rather than treating "emergency" as a single branch of the fault-response hierarchy that triggers a uniform CAM logic, it is more representative of the underlying physical reality to introduce a further level of hierarchical distinction within the emergency branch itself, based on the availability of the GNC function at the moment the anomaly is detected, as illustrated in Figure (5.16). The diagram illustrates the decision logic after Hold Point 4 within the overall FDIR architecture. Starting from the nominal state, the spacecraft continuously runs the anomaly detection logic in the background; as long as the monitored health parameter remains within its nominal bounds, the system proceeds undisturbed toward the safe docking condition, which represents the successful completion of the nominal path. If, at any point, the anomaly detection logic identifies a violation the system transitions into the emergency state. The architecture introduces a further hierarchical distinction at this stage, based on the operational status of the GNC subsystem at the moment the anomaly is detected.

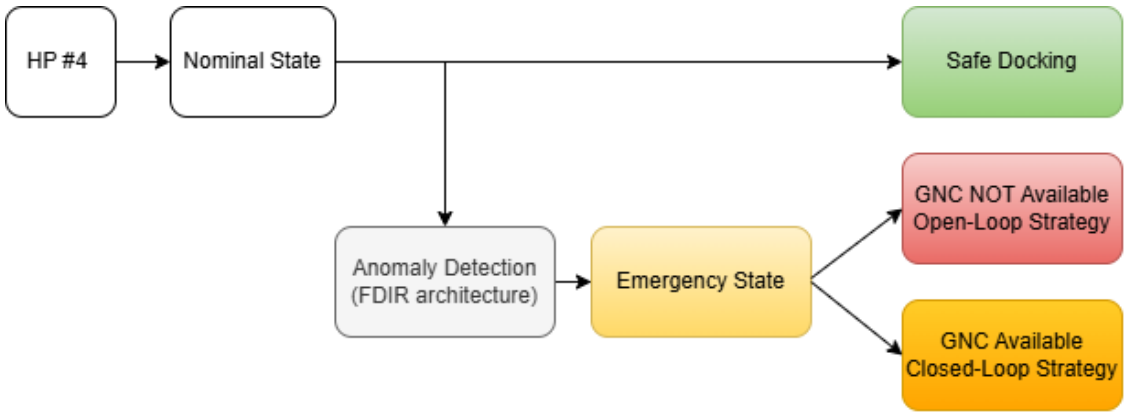



Figure 5.16: Decision hierarchy from Hold Point 4, branching from anomaly detection into a nominal path (safe docking) and an emergency path, the latter further split by GNC availability into Closed-Loop and Open-Loop escape strategies.

In the first sub-branch the GNC subsystem remains functional (even if degraded), and the spacecraft therefore retains the ability to estimate its relative state and to actuate corrective manoeuvres on the basis of that estimate. For this case a Closed-Loop CAM strategy, like those discussed so far, remains not only feasible but preferable, since it can exploit the available navigation information to optimize the escape trajectory, adapt to residual uncertainties, and potentially even abort and resume nominal operations if the anomaly is subsequently cleared.

In the second sub-branch, on the contrary, the GNC subsystem is no longer available, or its output can no longer be trusted with sufficient confidence to base a control decision upon, and for this exact reason any strategy relying on real-time feedback becomes not merely suboptimal, but structurally inapplicable: the spacecraft has no reliable means of knowing where it is with respect to the target. This is the scenario in which an Open-Loop CAM strategy becomes the only viable option.
This distinction finds direct support in the established literature on rendezvous and proximity operations safety. Fehse [1], in his treatment of collision avoidance and abort strategies for automated rendezvous and docking, repeatedly emphasizes the design principle according to which safety during close-range operations should not depend on the continued correct functioning of the very systems whose failure one is trying to protect against. This is the conceptual foundation behind the widespread use, in real rendezvous mission designs, of passively safe trajectories: these latter, once a corrective impulse has been applied, evolve under the natural relative dynamics in such a way that the chaser is guaranteed not to re-approach or collide with the target over at least one orbital period, without requiring any further control action, sensor measurement, or onboard decision. The very fact that such trajectories are computed to be safe in the absence of subsequent intervention is, in essence, the Open-Loop philosophy applied to collision avoidance: the manoeuvre is designed once, at the moment it is triggered, on the basis of the best available last-known relative state, and its safety must be intrinsic to the trajectory itself.
Naturally, this shift in philosophy comes with an unavoidable trade-off that needs to be stated explicitly rather than left implicit. A Closed-Loop strategy can in general achieve a more efficient escape in terms of precision in avoiding the KOZ, because of a real-time correction. An Open-Loop strategy, on the contrary, has no such fortune: in fact, the escape trajectory must be designed with sufficient margin to remain safe under the full envelope of uncertainty that could realistically apply at the trigger instant. This is, in essence, the price of robustness: an Open-Loop CAM must trade some performance for the unconditional guarantee that it does not rely on any capability that may no longer exist.
It is worth noting that if the decision to adopt a Closed-Loop or an Open-Loop CAM strategy is to be made autonomously and onboard, then the availability of the GNC function must itself become one of the conditions evaluated by the fault-detection logic before the appropriate escape strategy is selected. This means that the decision hierarchy implemented within the Stateflow chart should not simply branch into "nominal state" and "CAM state" as previously described, but should further partition the CAM state itself into (at least) two mutually exclusive sub-states, each associated with a distinct escape logic: a *GNC-available sub-state*, in which the Closed-Loop escape logic governs the manoeuvre and can be continuously informed by the relative navigation solution, and a *GNC-unavailable sub-state*, in which the Open-Loop escape logic takes over, computing a single, conservatively-margined impulsive manoeuvre on the basis of the last reliable relative state estimate, and then relying entirely on the natural propagation of the resulting trajectory.
Framed in these terms, the Open-Loop CAM strategy developed and analyzed in this

work should not be considered merely as a simplified or lower-fidelity alternative to a Closed-Loop approach. Rather, it should be understood as a more critical branch of the FDIR decision tree in which the spacecraft can no longer trust or access its own relative navigation function, and in which the only remaining safeguard against collision is a manoeuvre whose safety has been guaranteed a priori, independently of any capability the spacecraft may have already lost. It is precisely in this sense that an Open-Loop strategy constitutes the ultimate fallback layer of a well-designed FDIR architecture: the strategy of last resort, whose entire purpose is to remain effective exactly when every other, more capable, alternative has ceased to be.
With these considerations, an appropriate Open-Loop backup plan becomes necessary and, indeed, it has been applied to the previous case, maintaining the exact same logic with no active GNC functions. A clear difference is predictable, its qualitative representation being shown in Figure (5.17).

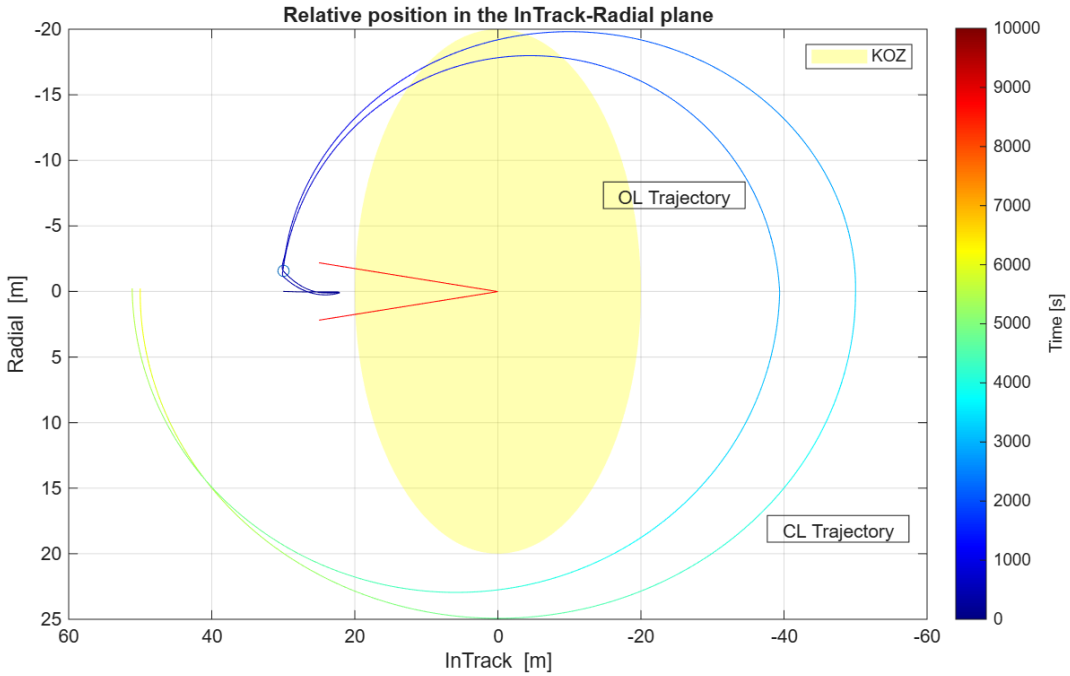



Figure 5.17: Comparison between the chaser's trajectories within a Closed-Loop versus Open-Loop architecture. The discrepancy follows a growing trend initially and reaches a constant value for the last part of the manoeuvre.

The results may look positive, since for a particular case no great difference is detectable between the two cases: at a first glance, the Open-Loop strategy seems a valid alternative. A very simple robustness analysis will prove the exact contrary. Figure (5.18) serves as evidence. For that particular circumstance, not only does the Open-Loop trajectory reach a final point very far from the desired one, but also the potential risk of a collision becomes higher and unacceptable, since the KOZ is clearly violated and the safety requirements are not met. In conclusion, the apparently trivial study requires a forward-thinking attitude: a new proposal shall substitute this second attempt, with the objective of finding a more robust solution.

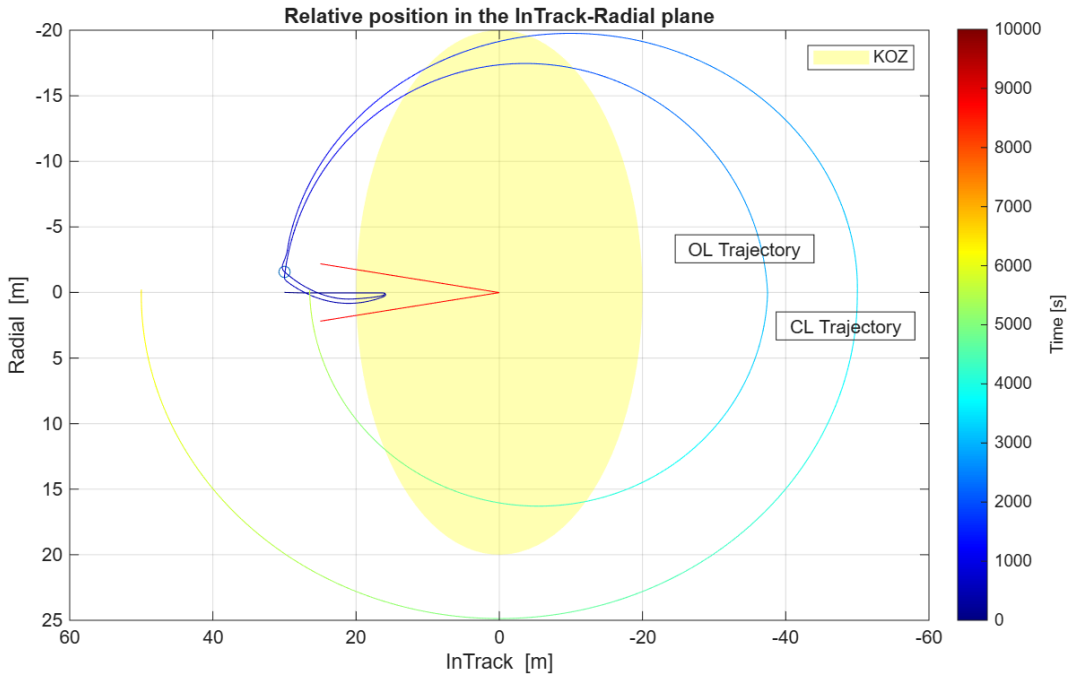



Figure 5.18: Comparison between the chaser's trajectories within a Closed-Loop versus Open-Loop architecture. As clearly visible, the uncontrolled scenario collides against the Keep-Out Zone, making the whole manoeuvre not suitable for a safe escape.

### 5.2.3 Third Iteration

Instead of a three-impulse strategy, as the ones described so far, a new suggestion is to make the chaser reach the final point *directly* after the first Way-Point has been touched, without any circular trajectory around the target. The objective is to find a more robust solution, capable of fitting into an Open-Loop architecture. Some examples are shown in the next Figure (5.19), which allows a direct comparison between the controlled approach and the Open-Loop architecture. The results can be considered very optimistic and have a great potential. Even though the evolution of the Open-Loop trajectory is not predictable, no risk of collision is detected and certainly the KOZ is not at risk. For these scenarios, the simulation is stopped after the horizontal axis has been touched again but it could potentially continue on. More detailed analyses regarding this long-term pattern are going to be presented further on. All simulations are computed considering a time of transfer of 0.25 $T$ for the second arch, corresponding to about 1420 seconds, plus 300 s for the first escape. As before, these values may freely vary since two impulses (Tangential and Radial) are combined, leaving the selection of the temporal interval to the user's discretion. In detail, a quarter of orbital period has been chosen just to avoid a trajectory that initially approaches the target before diverging away from it. Further studies may define more in detail this behaviour.
The three simulations shown provide a general overview on the third iteration, presented as alternative to the previous case. This time, the KOZ is perfectly avoided

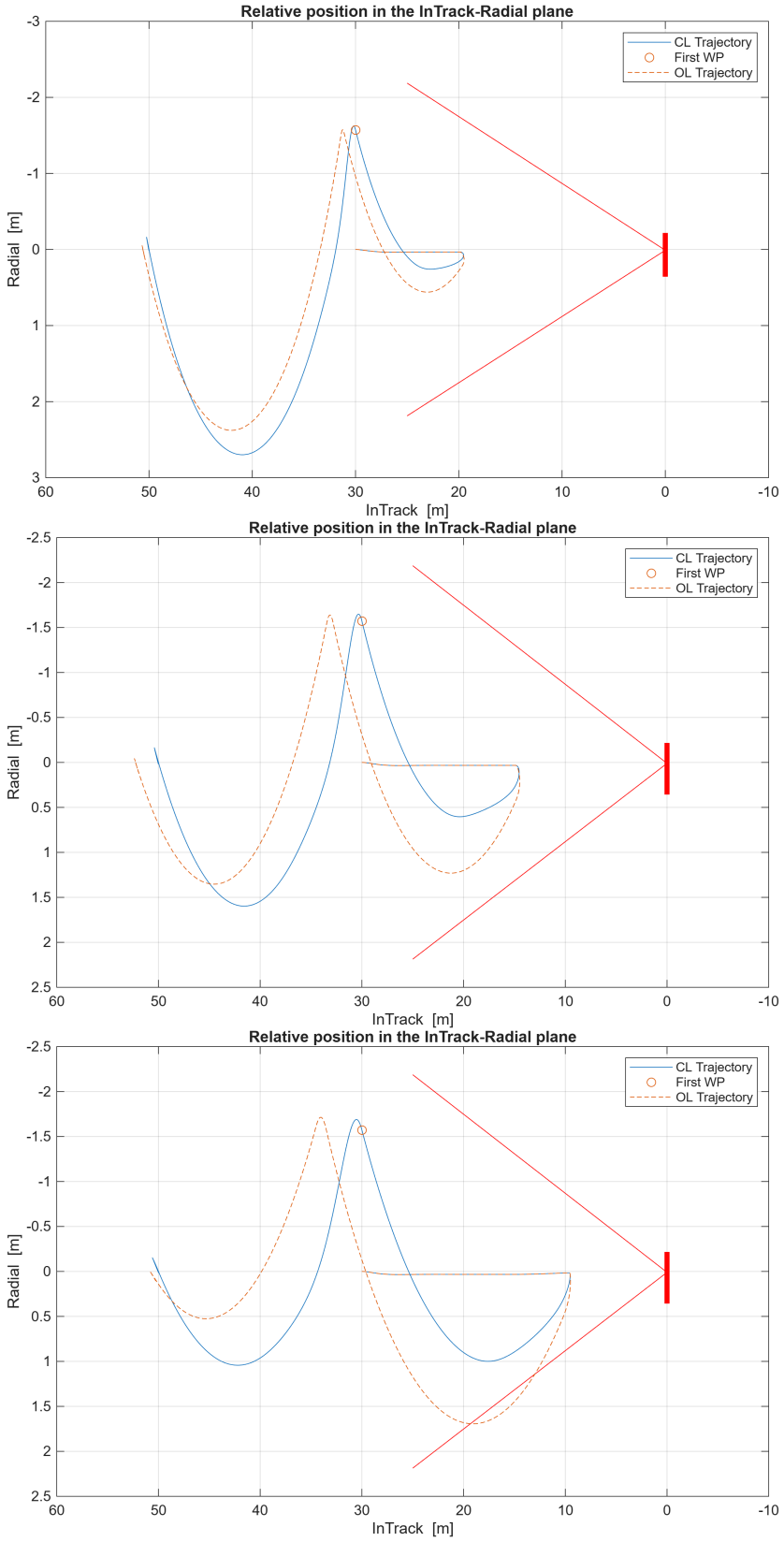



Figure 5.19: Comparison between Open-Loop and Closed-Loop escape trajectories. No transit around the target is imposed for safety reasons: on the contrary, a direct manoeuvre towards point (50,0) is adopted.

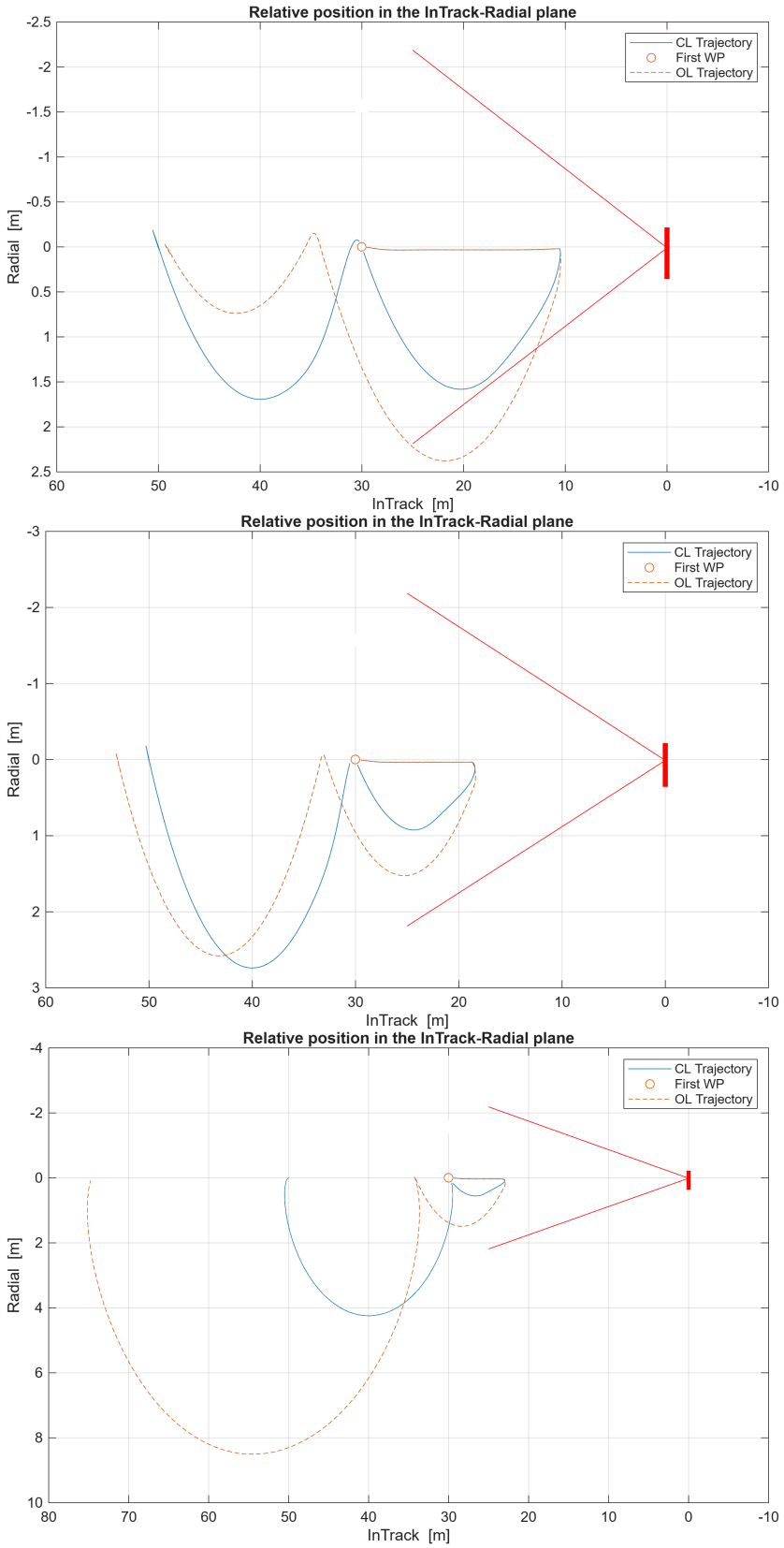



Figure 5.20: Comparison between Open-Loop and Closed-Loop escape trajectories with a new location for the Way Point. As before, a direct manoeuvre towards point (50,0) is adopted.

and the safety requirement regarding it is met without exceptions. However, the time for the escape must be correctly calibrated in order to perform a manoeuvre which will not result in the crossing of the approach cone, exactly as happens for the third simulation of Figure (5.19). This does not necessarily mean that the approach is not valid and another strategy must be defined: the idea is to identify a specific Time Of Flight (TOF) depending on the initial position during nominal procedures in correspondence of which the emergency is triggered. This is precisely what will be implemented for the robustness analysis presented at the end of this chapter.
Alternatively, the position of the Way-Point may be changed to study the consequences on the chaser's trajectory. As an example, the point was moved to the coordinates (30,0): this strategy could potentially be suitable for the mission, but some important observations shall be made. Firstly, the approach cone may be violated as well, and the reason clearly stands within the choice of the TOF, which shall be appropriately calibrated: at the time being, the two arcs are covered in the same temporal interval as those in Figure (5.19), that is, 300 and 1420 seconds respectively. In addition to this, the discrepancy between ideal and actual trajectories may become too wide, as it is shown in the third simulation of Figure (5.20). Even though it does not represent a hazard for safety requirements, a more precise study about this type of manoeuvre shall be conducted, just to observe the difference with the previous case.
Up to this point, the simulation logic relied on a state-based transition strategy: the Stateflow chart advanced to the next state only once a specific condition, typically a geometric threshold in the relative position, was satisfied. While intuitive from a control perspective, this approach implicitly assumes continuous, high-rate knowledge of the relative state, which may be compromised in an emergency situation, as stated previously. For this reason, a time-based transition strategy is considered more suitable: state transitions are triggered after a predefined time interval has elapsed, rather than being conditioned on instantaneous state knowledge. The next figure simply illustrates a comparison between state-based and time-based strategies, in order to identify potential improvements or defects.
Both trajectories are generated using the same nominal transfer time, imposed to compute the two required impulses along the tangential and radial directions. The difference between the two lies exclusively in how the second impulse is triggered. In the state-based case, the transition to the next Stateflow state is fired as soon as a specific geometric condition is satisfied, meaning the actual firing time depends on the real-time relative trajectory and may occur earlier or later than the nominal transfer time. In the time-based case, the second impulse is instead applied exactly at the nominal transfer time used to compute the $\Delta V$ components, regardless of the instantaneous relative state. As shown in the plot, this discrepancy produces visibly different trajectories: the time-based strategy anticipates the second burn relative to the state-based one, since the geometric threshold is met after the nominal transfer time elapses, leading to a more accurate approach toward the ideal Way Point but a less precise strategy according to real-time scenarios. This confirms that decoupling the manoeuvre timing from the true relative state does not preserve the consistency

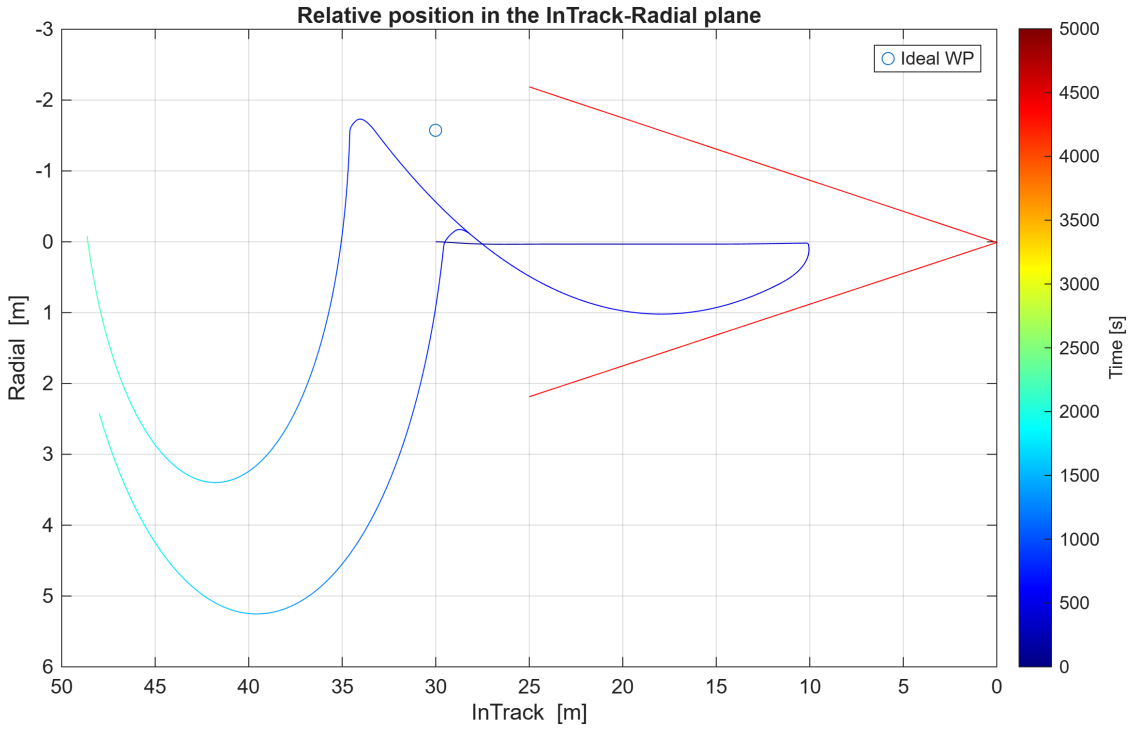



Figure 5.21: Comparison between state-based and time-based approaches for the same emergency state, with equal initial conditions and impulses. The time-based, more realiable, is also less faithful to the desired trajectory.

between the planned manoeuvre and the actual trajectory, which is a more realistic representation of an operational GNC logic. As regards costs, no particular difference is noticeable between the two approaches, which continue being suitable for the mission, from this particular perspective.
The third iteration presents very good results which, even if not perfect, are considered an optimal starting point for further discussions. The most important requirement to be met yet is the absence of violation of the approach cone: the problem may be easily solved with a proper selection of the TOF for the escape trajectories and for this reasons it is reputed easily solvable. A general comparison is hereby proposed to have an overview of pros and cons for each iteration, in order to easily operate a conclusive report of all the strategies presented so far.

### 5.2.4 Final Comparison

Table 5.2: Summary comparison of the CAM strategies developed across the three iterations, including internal sub-cases (variations on the same strategy).

| Iteration | Sub-case | TOF (s) | $\Delta V$ Cost (m/s) | Advantages | Disadvantages |
|---|---|---|---|---|---|
| **I** – Analytical, pre-Simulink (Eqs. 5.1–5.3) | 1.1 – Tangential boost only | [100; 800] | [0; 0.33] | Unique closed-form solution; straightforward derivation from Eq. 5.2; no optimization/DOF selection needed | Single free parameter ($x_0$) → no cost/time trade-off; $\Delta V$ up to 0.325 m/s near the target; exit time >800 s far from it |
| | 1.2 – Combined (Tangential + Radial) boost | [50; 300] | [0; 0.45] | Extra DOF (radial impulse) yields infinite solutions; enables cost/time trade-off over pre-selected TOF window (50–300 s); generally cheaper for a given time budget | Total $\Delta V$ up to 0.45 m/s for short imposed TOF near the target; requires a priori TOF window selection; higher risk from thruster plumes close to the target |
| | 1.3 – Cost/time trade-off (hybrid) | [50; 200] | [0.06; 0.15] | Practical compromise bounding both cost (≤0.15 m/s) and time (≤200 s); simple rule-based switching | Only a qualitative proposal, not validated/official; based on ideal unperturbed CW dynamics; KOZ avoidance not generalized (single favorable case, no MC) |
| **II** – Real-time Stateflow logic | 2.1 – Open-Loop, single impulse (1st WP at $(30, -30\tan 3°)$) | - | - | Impulse computed directly on-board (Stateflow, inverse-matrix); no dependency on continuous GNC feedback; simple real-time implementation | Chaser never exactly transits through WP; short transient radial excursion before diverging (approach-cone risk); strong sensitivity to CW In-Track/Radial coupling |
| | 2.2 – Closed-Loop toward WP + 2nd WP behind target | 2T (~ 11354) | [0.1; 1.5] | Effectively reduces radial excursion; approach cone fully respected in tested runs; 2nd transit point tunable for mission flexibility | Cost grows sharply with distance of 2nd WP (>1.2 m/s far vs. $(-50, 0)$); requires continuously available GNC; In-Track transit >1 km for far WPs |

*Continued on next page*

Table 5.2 – *continued from previous page*

| Iteration | Sub-case | TOF (s) | $\Delta V$ Cost (m/s) | Advantages | Disadvantages |
|---|---|---|---|---|---|
| | 2.3 – CL vs. OL comparison (GNC-availability architecture) | 1T (∼ 6000) | [0.2; 0.5] | Formalizes decision hierarchy on GNC availability (CL when available, OL as fallback); grounded in literature (Fehse) on passively-safe trajectories | OL variant fails robustness test in ≥1 configuration (KOZ violated); OL trades performance for robustness margin; not yet sufficient as sole safety guarantee |
| **III** – Direct two-impulse manoeuvre (no transit around target) | 3.1 – Original WP, direct to $(50, 0)$ | 300 + ∼ 1420 | [0.15 0.6] | Only two impulses (no circular transit); optimistic/promising overall results; KOZ never at risk even under OL uncertainty | Approach cone crossed in ≥1 tested case; fixed TOF not recalibrated for all trigger positions; position-dependent TOF still to be defined |
| | 3.2 – WP moved to $(30, 0)$ | 300 + ∼ 1420 | [0.15; 0.6] | Confirms strategy flexibility to WP relocation; no absolute KOZ violation in tested nominal case | Approach cone violated as well (TOF not recalibrated for new geometry); larger CL/OL discrepancy in ≥1 run; further dedicated study required |
| | 3.3 – State-based vs. time-based 2nd-impulse triggering (WP at $(30, -30 \tan 3)$) | 300 + ∼ 1420) | [0.15; 0.6] | Time-based firing decouples 2nd impulse from real-time relative-state knowledge, reflecting operational GNC under degraded/emergency navigation; no appreciable $\Delta V$ cost penalty wrt state-based logic | Time-based approach less faithful to the ideal trajectory shape, since firing time does not adapt to the actual relative state; discrepancy between planned nominal transfer time and true relative motion still present |

## 5.3 Monte Carlo Robustness Analysis

A proper study shall be confirmed with a robustness analysis which guarantees the success of the escape manoeuvre with varying initial conditions. As for the nominal scenario, the parameters selected and their respective range of variability are here reminded of. In addition to them, the starting point for the emergency trigger is randomly chosen within the approach cone with no limitations, in order to precisely identify the previously-mentioned Point of No Return.

| Parameter | Range |
|---|---|
| Initial Position | $p_0 \pm 0.5$ m |
| Initial Velocity | $V_0 \pm 0.02$ m/s |
| Chaser Mass | $20 < m_c < 24$ kg |
| Maximum Thrust | $7.5mN - 10\% < F_{max} < 7.5mN$ |
| Drag Coefficient | $2 < C_D < 2.5$ |
| Trigger In-Track Position | $0 < x_{CAM} < 25$ |

Table 5.3: Ranges of the parameters dispersed in the Monte Carlo simulations, applied around their nominal values to account for uncertainties in the initial state, spacecraft mass, actuator performance and aerodynamic drag, with an extra variability around the location where the emergency state is activated.

As before, 58 simulations have been run to reach a 95% of reliability and confidence level. All simulations last 87000 seconds to cover an entire day (86400 seconds) plus an extra 600 seconds for the nominal trajectory before the emergency is activated. The results are shown in Figure (5.22). The color gradient shows the evolution of the manoeuvre, which ultimately reaches a final distance of about 1700 m from the target. The safety of the long-term analysis is guaranteed since the KOZ is not violated. Figure (5.23) offers another perspective, showing in detail the approach cone which is successfully not crossed. Besides, Figure (5.24) presents an extreme close-up view to show trajectories which are deliberately interrupted once the approach cone is violated. The respective In-Track points in correspondance of which the emergency is activated become potential no return points. The furthest one from the target is then selected as the official candidate. Lastly, Figure (5.25) gives a general overview on costs for each manoeuvre, with a maximum value of 0.5 m/s circa before the No-Return Point is identified. With all these considerations, its location corresponds to the coordinates (6.89, 0) m. As a safety margin, the proposed definitive position is **(8, 0)** to make sure no crossing is possible. This optimal result is achieved by properly calibrating (within *ad hoc* simulations not reported here) the TOF, which may be visualized with these instructions:

$$\begin{aligned} TOF &= 400s \qquad 25m < x_{CAM} \leq 20m \\ TOF &= 300s \qquad 20m < x_{CAM} \leq 15m \\ TOF &= 200s \qquad otherwise \end{aligned} \tag{5.5}$$

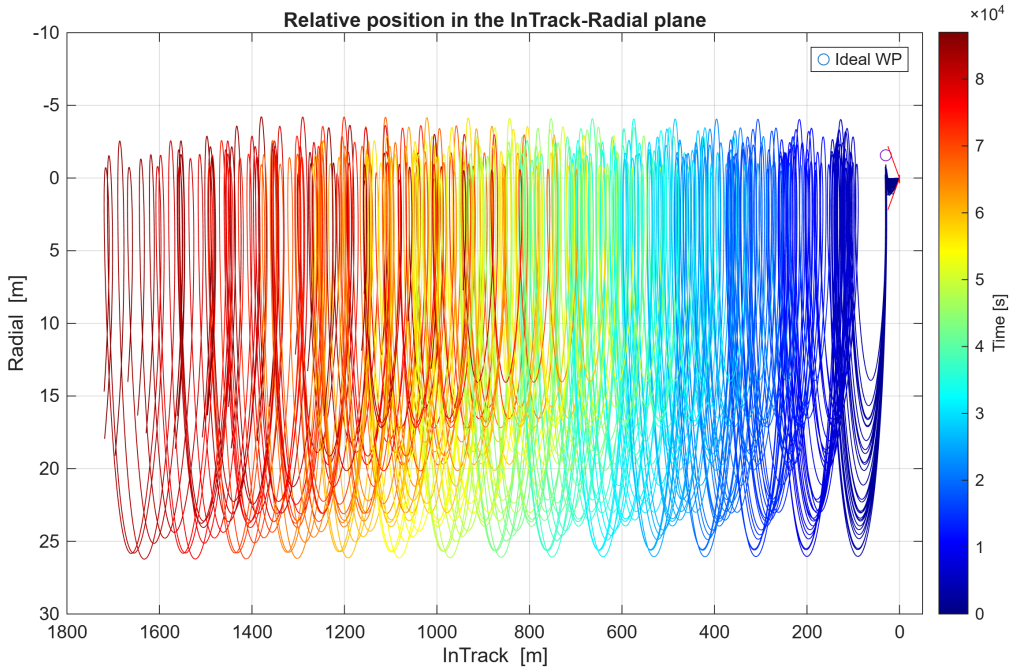



Figure 5.22: Long-term propagation of the chaser's relative trajectories in the InTrack-Radial plane for the full set of Monte Carlo runs, colored according to elapsed time since the start of the simulation. Each closed loop corresponds to one orbital period of the target. The trajectories span nearly 15 orbital revolutions and reach an InTrack separation of about 1800 m by the end of the propagation.

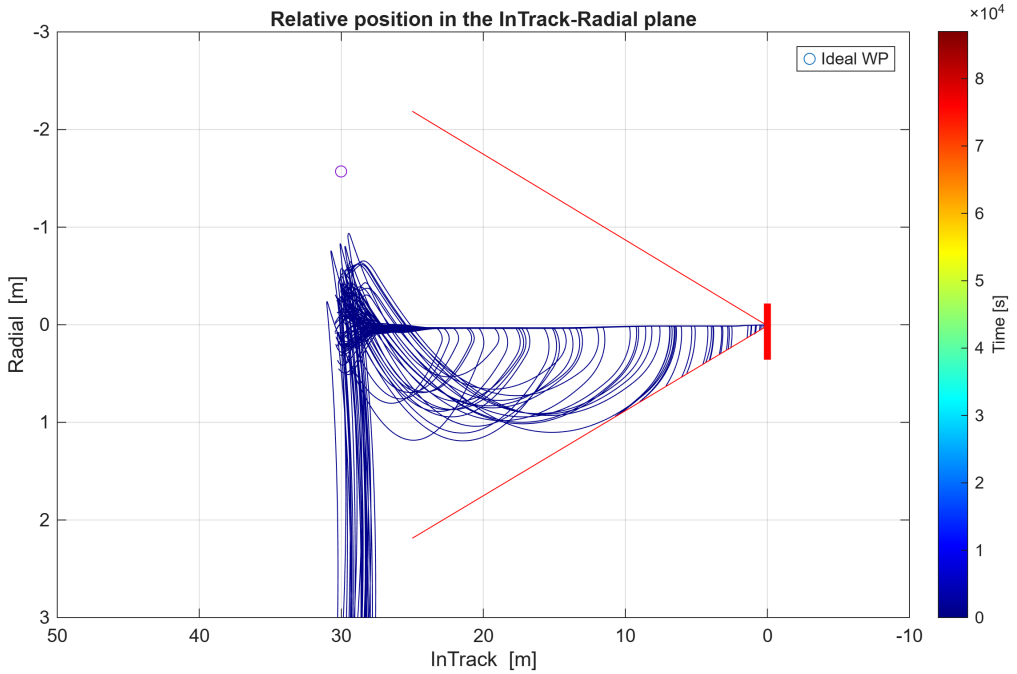



Figure 5.23: Close-up view of the trajectories in the vicinity of the target, colored according to elapsed time since the manoeuvre was triggered. The safety requirement is satisfied for all runs that are allowed to continue undisturbed to completion.

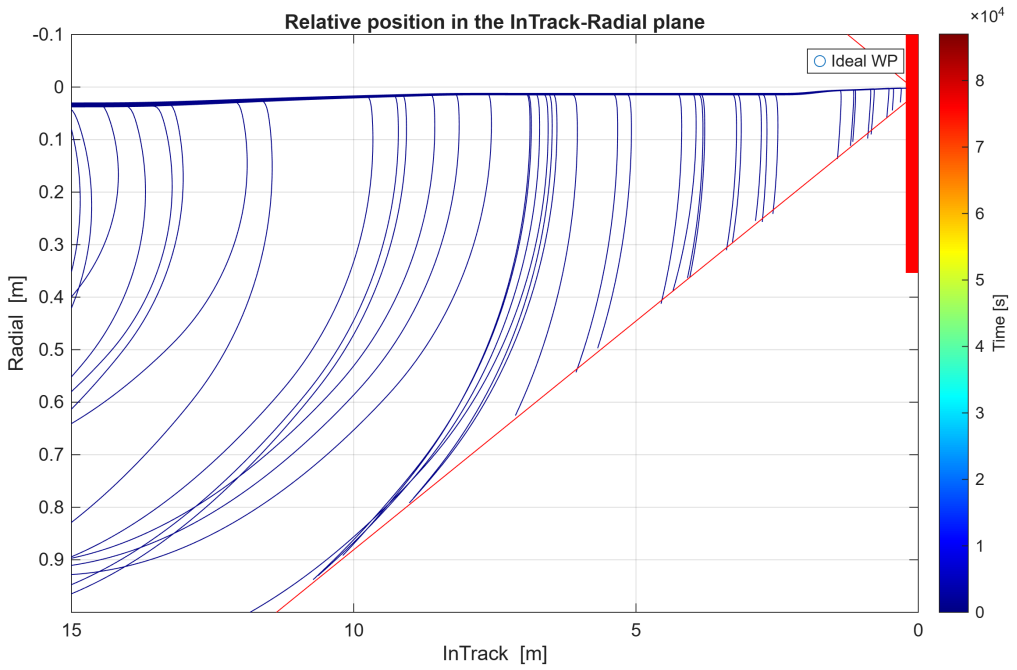



Figure 5.24: Extreme close-up view of the trajectories near the origin, colored according to elapsed time since the manoeuvre was triggered. Several trajectories are deliberately truncated at the instant they intersect the approach cone boundary (red line), allowing the corresponding InTrack position to be identified with high precision as a candidate point of no return.

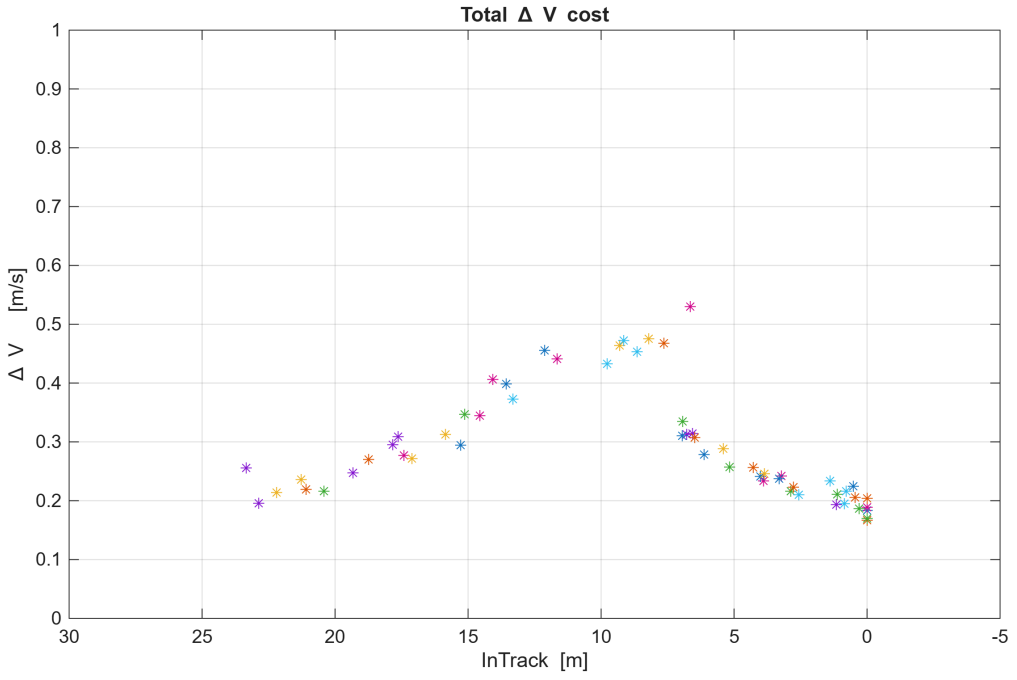



Figure 5.25: Total impulsive $\Delta V$ cost as a function of the InTrack position at which the CAM is triggered, for the full set of Monte Carlo runs (each color corresponding to a different simulation instance). The cost initially grows as the triggering point moves closer to the target, reaching a maximum in the 5-10 m range, and subsequently decreases for the innermost trigger positions

For these Monte Carlo simulations, a very important note must be reported. After the first 300-seconds-transfer, the second impulse computes the required $\Delta Vs$ to reach the point (100,0) (corresponding to HP#3) in 4258 seconds (~ 0.75T) and **NOT** 1420 seconds (~ 0.25T) for the simple reason that if the trajectory is left free to naturally evolve, and not stopped at the final point, as it always happened so far, the chaser dangerously starts heading towards the target and may eventually violate the KOZ. This phenomenon could not be observed without a long-term simulation as the one proposed in Figure (5.22). This statement is therefore fundamental to propose a final escape strategy.

### 5.3.1 Study on Location of the Way Point

To conclude the discussion about the Monte Carlo robustness process, another final campaign has been run, considering varying initial conditions, included the radial position of the Way Point, triggering the CAM in correspondance with the above defined No-Return point located at (8,0). The aim is to understand whether better positions for the WP are possible and how they will improve the results. Hereby are presented some of the 58 runs conducted as examples. The radial position of the Way Point varies between 0 and -5 meters (actually, a wide range as the results will demonstrate). For the first case in Figure (5.26) the Way Point is located too close to the horizontal axis, thus making the chaser's trajectory crossing the approach cone and failing the safety requirement. A mirrored behaviour is presented in Figure (5.27) where, on the contrary, the Way Point is too far from the InTrack axis and the chaser collides against the approach cone on the opposite side, towards negative radial coordinates. From this preliminary campaign, the ideal Way Point lies inside the radial interval [-0.98, -3.64]: since the initial coordinate was $-30 \tan 3° \simeq -1.5722m$, the admissible error on the radial position of the WP is asymmetric: an increase of up to +0.592 m (toward the upper limit) or a decrease of up to -2.068 m (toward the lower threshold) can be tolerated before the approach-cone constraint is violated. This asymmetry reflects the fact that the nominal position is not centered within the acceptable interval, leaving considerably more margin against an excessive radial excursion. Even though these preliminary results strongly depend on the InTrack coordinate as well, further analyses are suggested in order to identify more suitable locations. Nonetheless, the one here proposed is considered acceptable, with a margin error of about ±25% (corresponding ± 0.393 m) on the radial position of the Way Point. The simulations proposed in Figure (5.26) and (5.27) are intentionally interrupted after 5000 seconds because the real purpose of these runs is to study the initial variability on the chaser's trajectory rather than its natural evolution over time. A proper robustness analysis considering also a random radial location of the Way Point within the interval $[-30 \tan(3°) \pm 0.393]$ m may bring solid results and confirm the correctness of the strategy selected.

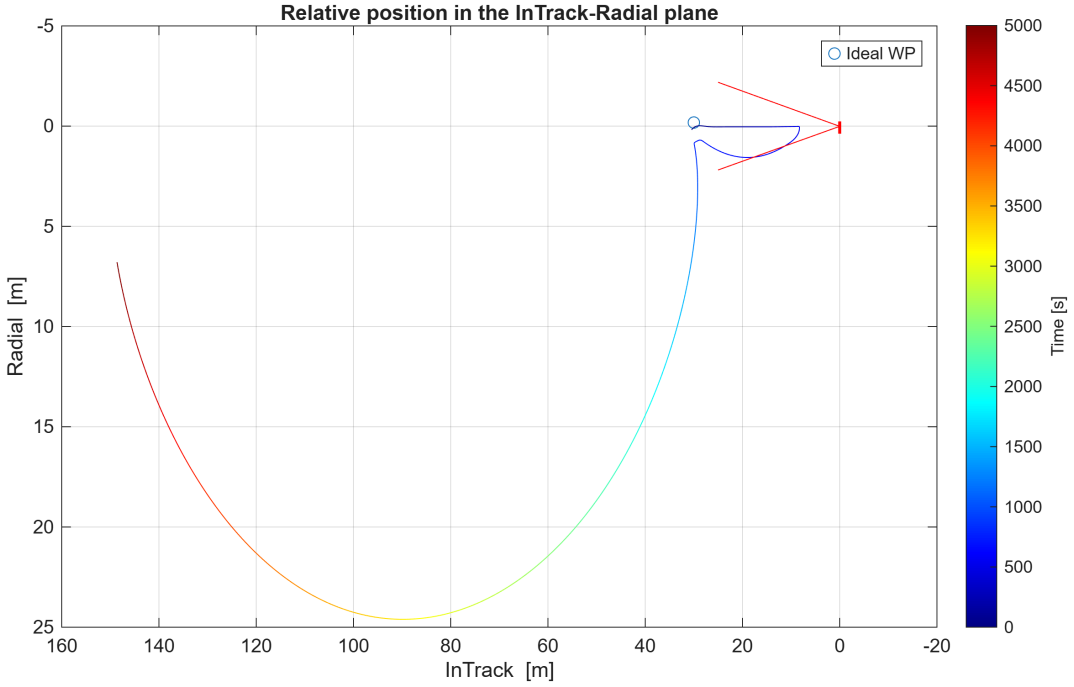



Figure 5.26: Relative trajectory in the InTrack-Radial plane for a trigger position equal to (8,0) with the ideal Way Point located at (30,0). The chaser approaches along a long CW arc after exiting the approach cone and completing the final transit toward the WP.

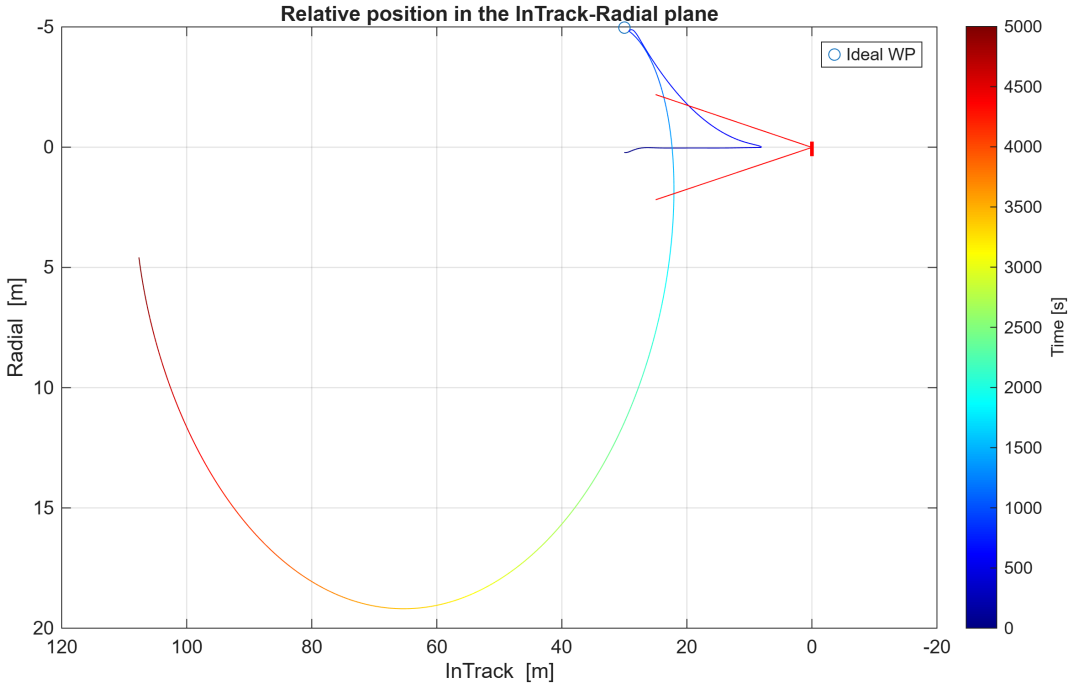



Figure 5.27: Relative trajectory in the InTrack-Radial plane for a trigger position equal to (8,0) with the ideal WP relocated to (30,-5). The chaser approaches along a long CW arc after exiting the approach cone and completing the final transit toward the WP.

# Chapter 6

# Conclusions

This thesis set out to pursue two main objectives: first, to assess the robustness of the nominal final approach phase against variations in the velocity profile and initial conditions; second, to investigate and design feasible CAM strategies to be triggered in case of an emergency state, building on a critical review of existing FDIR architectures. The following sections summarise the main results obtained with respect to these two objectives.

As regards the Final Approach, different velocity profiles have been investigated to carry out a comparison between them based upon specific judgement parameters such as total $\Delta V$, fuel consumption and duration of the manoeuvre. For the latter, an explicit requirement states that the maximum temporal interval admissible is 1800 seconds, that is, 30 minutes to complete docking. Among all the described cases, no scenario violates this requirement. When comparing the other parameters, which are directly interconnected, it can be seen that the cheapest solutions introduce potentially critical aspects as regards control thrusts and CAM feasibility. As stated, the hazards introduced are potential and not actually verified and for this exact reason a more detailed analysis is suggested. On the contrary, the most propellant-intensive cases present smoother command profiles but at the same time longer durations. Generally speaking, though, no scenario may be excluded *a priori* since the differences detected are not actually verified against safety. So far, the Final Approach is robust against variations in its velocity profile, making it very resilient to unpredictable perturbations. The Monte Carlo campaign conducted on the first profile of Figure (4.1) demonstrates the validity of the method with a 95% confidence level and a 95% reliability, confirming it as the most suitable profile among those listed. A higher number of successful runs will increase the reliability of the strategy. Nonetheless, the results obtained with 58 simulations are considered optimal for the purposes of this thesis.

As regards CAM strategies, a more complex situation arises: firstly, the proper architecture shall be identified. A common hierarchical structure is considered particularly suitable for this type of mission, since it provides for a very distinct subdivision of emergency levels depending on the components or subsystems involved. This thesis focused particularly on the highest level, where no other possibility is available than a CAM. Through different steps, it was possible to identify a proper way to address

the escape problem:

1. the computation of the manoeuvre must take place necessarily on board the spacecraft to account for real initial conditions of the chaser's state;

2. to protect the KOZ, no trajectory around the target is admissible, especially in Open-Loop when the GNC subsystem is not available to check the relative state of the two spacecraft;

3. when planning the escape trajectories, the appropriate timing of the transfer shall be investigated not to violate the approach corridor or the KOZ, and thus to meet the safety requirements imposed;

4. together with timing, the location of the so-called Way Point shall be identified, to respect the safety requirement regarding the violation of the approach cone;

5. the no-return point needs to be properly identified as the location, alongside the nominal approach trajectory, after which the CAM is no longer executable due to violation of safety requirements;

6. a time-based strategy becomes necessary especially if the GNC functions are compromised, as in typical emergency scenarios. The recovery is then complicated by the randomness of this approach, since it is very hard to match the expected and the real trajectories during the escape manoeuvre.

The following schedule describes the proper steps for a safe escape manoeuvre:

1. emergency state is detected inside the approach corridor, the In-Track component ranging from 25 to 8 meters from the target. For closer points, another strategy shall be implemented;

2. depending on the location in which the emergency is triggered, the appropriate TOF shall be imposed as follows

$$\begin{aligned} TOF &= 400s \qquad 25m < x_{CAM} \leq 20m \\ TOF &= 300s \qquad 20m < x_{CAM} \leq 15m \\ TOF &= 200s \qquad otherwise \end{aligned} \tag{6.1}$$

3. the Way Point is located at $(30, -30\tan 3°)$. The robustness of the selected point is confirmed through a Monte Carlo analysis;

4. the following impulse is computed to reach the previous hold point in $(100, 0)$, a practical solution since it represents a defined location already adopted for nominal procedures, with a TOF of 0.75T, corresponding to 4258 seconds, because shorter times may compromise the safety of the escape since the chaser's trajectory will head towards the target. This latter temporal interval was introduced only within the Monte Carlo campaign since previous simulations did not investigate the long-term behaviour of the escape manoeuvre.

5. the chaser's trajectory is left free to naturally evolve, with no safety hazards detected within a whole day.

Some limitations of the present work should be acknowledged. The proposed escape strategy has only been validated for emergency states triggered within the 8-25 m In-Track range of the approach corridor; scenarios occurring closer than 8 meters remain unaddressed. Furthermore, several potential hazards associated with the more propellant-efficient velocity profiles (e.g. control thrust peaks) were identified but not verified through dedicated simulations, and the analysis of the Way Point location, while indicative of a viable interval, is based on a preliminary campaign of 58 runs and would benefit from a broader statistical validation.
Future developments shall surely discuss which strategy to implement when the emergency state is activated closer than 8 meters, a concept that has remained uncovered in this thesis. A further study on the proper location of the Way Point shall be added as well in order to understand whether better transfers are possible or not. Lastly, a new method to completely recover the chaser after the escape trajectory has been commanded shall be studied to allow the mission to continue with its objectives. Overall, the results obtained provide a validated baseline for both the nominal approach phase and the emergency escape logic of the SROC mission, and offer a replicable methodology that can inform the design of FDIR architectures for future CubeSat RPOD missions, where onboard autonomy and resource constraints make robust, well-characterised emergency strategies particularly valuable.

# Appendix A

# Mathematical Formulation of Disturbances

This appendix is intended to clarify and justify the results produced in table 2.5. The logic flux follows the structure proposed in [1].

## A.1 Drag due to residual atmosphere

The drag force is defined as

$$F_D = -\frac{\rho}{2} V_x^2 C_D A \tag{A.1}$$

where $V_x = \omega r$ is the orbital velocity; $C_D$ is the drag coefficient; $A$ is the cross section of the body. The differential drag force per unit mass between chaser and target is

$$\begin{aligned} \Delta\gamma_D = \frac{F_{Dc}}{m_c} - \frac{F_{Dt}}{m_t} = \gamma_{Dc} - \gamma_{Dt} = -\frac{\rho}{2}\omega^2 r^2 \left( \frac{C_{Dc} A_c}{m_c} - \frac{C_{Dt} A_t}{m_t} \right) = \\ = -\frac{\rho}{2}\omega^2 r^2 \frac{1}{C_{Bc}} \left( 1 - \frac{C_{Bc}}{C_{Bt}} \right) \end{aligned} \tag{A.2}$$

with the introduction of the ballistic coefficient $C_B$ defined as $\frac{m}{C_D A}$. The highest uncertainty belongs to the definition of the atmospheric density $\rho$, defined as

$$\rho(z) = \rho(z_0) exp\left( \frac{z - z_0}{H(z_0)} \right) \tag{A.3}$$

in the neighbourhood of a known value for a certain orbital height. $H(z_0)$ is a scale height coefficient. Solar fluxes influence atmospheric density via temperature changes: under solar illumination, density will increase, vice versa on the opposite side. Solar flux is characterised by three different cyclic alterations with their own periods (27 days for solar rotation, 1 year for solar revolution and 11 years for solar activity). As already stated, the last effect is the preponderant one.

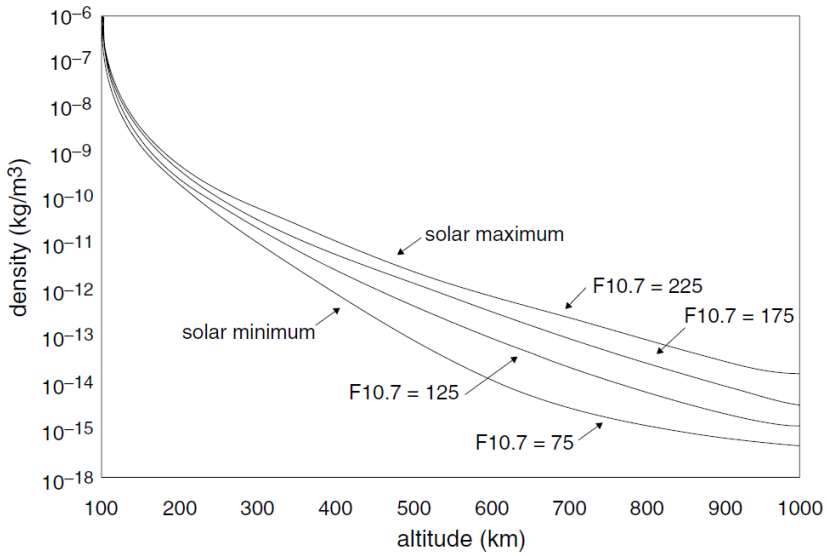



Figure A.1: Variation of Earth's atmospheric density as a function of altitude for different levels of solar activity. The divergence between solar minimum (F10.7 = 75) and solar maximum (F10.7 = 225) conditions expands significantly, spanning multiple orders of magnitude, at higher altitudes. This dynamic variation directly dictates the magnitude of the residual drag force experienced by the chaser and target vehicles [3].

## A.2 Disturbances due to geopotential anomaly

These alterations derive from our planet's deviations from an ideal sphere and the fact that gravitational forces are not necessarily directed towards its centre. The gravitational potential of the Earth can be approximated by the function

$$\Phi = \frac{\mu}{r}\left(1 - \sum_{n=2}^{\infty} J_n \left(\frac{R_E}{r}\right)^n P_n \sin\phi\right) \tag{A.4}$$

with $J_n$ being the harmonic coefficients of the potential; $R_E$ the Earth radius at the equator; $r$ the distance of the satellite from the centre of the Earth; $P_n$ Legendre polynomials; $\phi$ the latitude; and $\mu$ the gravitational constant. This model only accounts for zonal deviations which depend on latitude only. As already discussed, the most significant contribution derives from the second harmonic $J_2$ which produces relevant rotation of the line of apsides and regression of the ascending node.

## A.3 Solar pressure

As stated in table 2.5, the force produced by solar radiation, per unit mass, is

$$\vec{\gamma_{SP}} = -p\frac{A}{m}\vec{u_S} \tag{A.5}$$

Particularly, the radiation momentum flux $p$ varies from a maximum of $4.68 \prod 10^{-6} N/m^2$ at perihelion to a minimum of $4.38 \prod 10^{-6} N/m^2$ at aphelion. Solar pressure has effects on all orbital parameters, especially eccentricity and inclination, and naturally varies with alternation of sunlight/eclipse. The effects are enhanced in Geostationary Earth Orbit (GEO).

## A.4 Effects of thruster plumes

Obviously, these alterations become non-negligible at short-range distances. The force exerted by a thruster plume on a surface element dS is defined as

$$dF = -P(r, \theta) \cos \gamma dS \tag{A.6}$$

where $P(r, \theta)$ is the plume pressure and $\gamma$ is the angle between the surface normal vector and the gas flux vector. Desplats ([35]) suggests this modelization for the pressure $P$

$$P(r, \theta) = \frac{\phi_0}{r^2} exp\left(\frac{-\theta^2}{2\theta_0^2}\right) \tag{A.7}$$

where $\phi_0$ is a flux constant, $\theta_0$ is the half cone angle, typically 13 deg. Integrating Eq. (A.6) with the pressure profile of Eq. (A.7) brings this final result for the characteristic flux:

$$\phi_0 = \frac{F_0}{\pi} \frac{1}{\int_0^\pi exp(-\frac{\theta^2}{2\theta_0^2}) \sin(2\theta) d\theta} \tag{A.8}$$

## A.5 Navigation errors

Navigation errors involve both position and velocity deviations, which are quantified in the following table:

| **Measurement error** | **Trajectory error** |
|---|---|
| $\Delta x_m$ | $\Delta x_m$ |
| $\Delta V_{xm}$ | $\Delta x(t) = \Delta V_{xm}\left(\frac{4}{\omega}\sin(\omega t) - 3t\right)$<br>$\Delta z(t) = \frac{2}{\omega}\Delta V_{xm}\left(\cos(\omega t) - 1\right)$ |
| $\Delta y_m$ | $\Delta y(t) = \Delta y_m \cos(\omega t)$ |
| $\Delta V_{ym}$ | $\Delta y(t) = \frac{1}{\omega}\Delta V_{ym} \sin(\omega t)$ |

## A.6 Thrust errors deviations

Thrust errors can be divided into two categories, depending whether the alterations involve the thrust force and the actual mass of the spacecraft or the duration of the

| Different orbits | |
|---|---|
| **Measurement error** | **Trajectory error** |
| $\Delta z_m$ | $\Delta x(t) = \frac{3}{2}\omega \Delta z_m t$ |

| Same orbit | |
|---|---|
| **Measurement error** | **Trajectory error** |
| $\Delta z_m$ | $\Delta x(t) = 6\Delta z_m(\omega t - \sin(\omega t))$<br>$\Delta z(t) = \Delta z_m(4 - 3\cos(\omega t))$ |

| Measurement error | Trajectory error |
|---|---|
| $\Delta V_{zm}$ | $\Delta x(t) = \frac{2}{\omega}\Delta V_{zm}\big(1 - \cos(\omega t)\big)$<br>$\Delta z(t) = \frac{1}{\omega}\Delta V_{zm}\sin(\omega t)$ |

impulse. Thus, the nominal thrust force per unit mass is

$$\gamma_t = \frac{F_t}{m_c} \tag{A.9}$$

while the actual value is

$$\gamma = \gamma_t + \Delta\gamma = \gamma_t \epsilon_\gamma \tag{A.10}$$

The thrust error factor, depending on its origin, has different formulations:

$$\epsilon_\gamma = 1 + \frac{\Delta F}{F_t}; \qquad \epsilon_\gamma = 1 + \frac{m_c}{\Delta m} \tag{A.11}$$

or even

$$\epsilon_t = 1 + \frac{\Delta t}{t_t} \tag{A.12}$$

if alterations on the duration occur. The combination of both typologies produces the complex variations on trajectory and velocity hereby presented.

**Thrust in the InTrack direction**

$$x(t) = x_t(t)\epsilon_x = \gamma_x \epsilon_\gamma \left(\frac{4}{\omega^2}(1 - \cos(\omega t_t \epsilon_t)) - \frac{3}{2}t_t^2 \epsilon_t^2\right) z(t) = z_t(t)\epsilon_z = \frac{2}{\omega^2}\gamma_x \epsilon_\gamma(\sin(\omega t \epsilon_t)) - \omega t \epsilon_t) \tag{A.13}$$

$$\dot{x}(t) = \dot{x}_t(t)\epsilon_{V_x} = \gamma_x \epsilon_\gamma \left(\frac{4}{\omega}\sin(\omega t \epsilon_t) - 3t\epsilon_t\right) \dot{z}(t) = \dot{z}_t(t)\epsilon_{V_z} = \frac{2}{\omega}\gamma_x \epsilon_\gamma(\cos(\omega t \epsilon_t) - 1) \tag{A.14}$$

**Thrust in the Cross-Track direction**

$$y(t) = y_t(t)\epsilon_y = \frac{1}{\omega^2}\gamma_y \epsilon_\gamma(1 - \cos(\omega t \epsilon_t)) \tag{A.15}$$

$$\dot{y}(t) = \dot{y}_t(t)\epsilon_y = \frac{1}{\omega}\gamma_y \epsilon_\gamma \sin(\omega t \epsilon_t) \tag{A.16}$$

**Thrust in the Radial direction**

$$x(t) = x_t(t)\epsilon_x = \frac{2}{\omega^2}\gamma_z \epsilon_\gamma(\omega t \epsilon_t - \sin(\omega t \epsilon_t)) z(t) = z_t(t)\epsilon_z = \frac{1}{\omega^2}\gamma_z \epsilon_\gamma(1 - \cos(\omega t \epsilon_t)) \tag{A.17}$$

$$\dot{x}(t) = \dot{x}_t(t)\epsilon_{V_x} = \frac{2}{\omega}\gamma_z\epsilon_\gamma(1-\cos(\omega t\epsilon_t))\dot{z}(t) = \dot{z}_t(t)\epsilon_{V_z} = \frac{1}{\omega}\gamma_z\epsilon_\gamma\sin(\omega t\epsilon_t) \quad \text{(A.18)}$$

As regards thrust direction errors, different origins can explain their insurgence, such as attitude errors, geometric misalignment of the thruster hardware, etc. These kind of perturbations lead to a component of thrust in a perpendicular direction.

# Appendix B

# Reference Frames

Describing absolute/relative motion and attitude, three types of coordinate frames are needed according to [1]:

- Orbit reference frames, which describe the spacecraft orbit with respect to an inertial frame;
- Spacecraft local orbital reference frames, useful to describe the motion relative to another spacecraft;
- Spacecraft attitude and body frames, used when defining manoeuvres according the spacecraft's centre of mass' perspective.

The last type of frames will not be discussed since they have not been used during the analysis conducted for the purposes of this thesis.

## B.1 Earth-centred equatorial frame $F_{eq}$

With the assumption of truly spherical planet, this frame is used to describe the orbital motion around the centre of the Earth, which as a consequence represents the Origin of the reference system. The first axis points towards the vernal equinox, the third one towards north, normal to the equatorial plane, while the second one completes the triad (Fig. B.1).

## B.2 Orbital plane frame $F_{op}$

This reference frame describes the absolute motion with respect to the orbital plane. Its origin coinciding with the centre of the Earth, it is the result of two rotations in succession from the $F_{eq}$, that is, a first rotation around the third equatorial axis by the Right Ascension of the Ascending Node (RAAN) angle and second rotation around the first axis by the inclination angle $i$. In mathematical terms,

$$\begin{bmatrix} x_{op} \\ y_{op} \\ z_{op} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos i & \sin i \\ 0 & -\sin i & \cos i \end{bmatrix} \begin{bmatrix} \cos\Omega & \sin\Omega & 0 \\ -\sin\Omega & \cos\Omega & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{eq} \\ y_{eq} \\ z_{eq} \end{bmatrix}$$

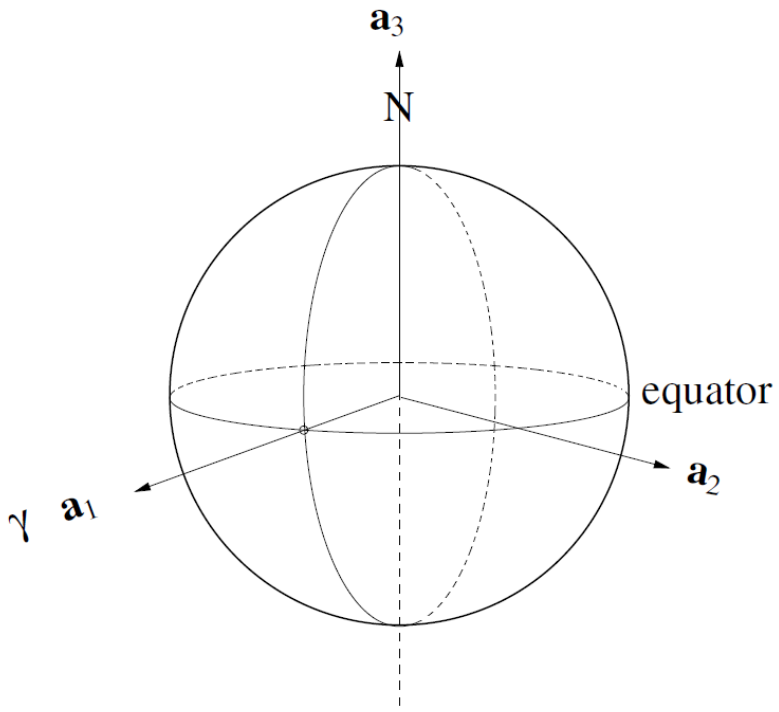



Figure B.1: Earth-centred equatorial frame, with depiction of the three axes direction and the equatorial plane

Thus, the first axis points towards the ascending node, the third one is normal to the orbital plane and the second one completes the triad (Fig. B.2).

## B.3 Spacecraft local orbital frame $F_{lo}$

This frame is centred on the spacecraft's centre of mass, with the second axis in the opposite direction of the angular momentum vector, the third one pointing to the Earth's centre and the first one, often but not necessarily aligned with the orbital velocity vector, completes the triad. The three axes are usually referred to respectively as **V-bar**, **H-bar** and **R-bar**. The target's local orbital frame, often called Local-Vertical Local-Horizontal (LVLH) frame, is the reference system in which the chaser's approach trajectory is described.

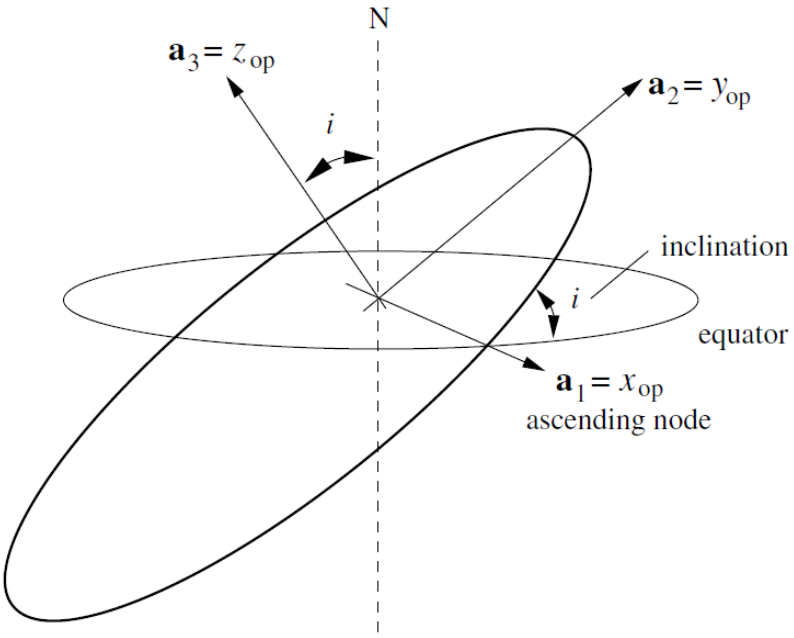



Figure B.2: Orbital plane frame, with depiction of the three axes direction, the equatorial plane, the ascending node and the inclination of the orbital plane

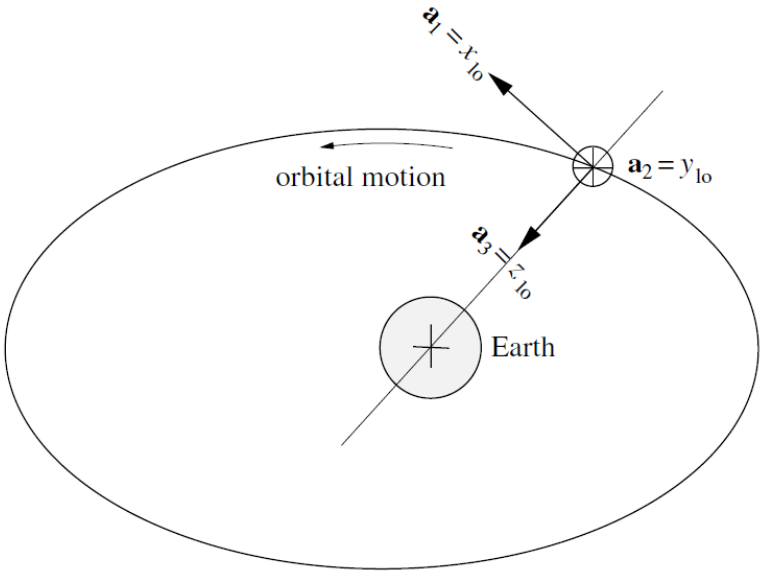



Figure B.3: Spacecraft local orbital frame, with depiction of the three axes direction and the orbital plane and motion

# Appendix C

# Simulink Code

The Simulink code implemented within Stateflow is hereby inserted and briefly commented, in order to help the reader familiarise with the logical flow adopted in this work. The code is the one reiterate for the Monte Carlo campaign described in Chapter 5 to validate the results.

```
1        EMERGENCY
2        entry:
```

The state is called EMERGENCY and the first instructions are initialised after calling the *entry:* command

```
1        hold=0;
2        coder.extrinsic('compute_impulse');
3        dx_req=0.0;
4        dz_req=0.0;
```

The parameter *hold* is an output which will determine different safety strategies adopted within the Simulink environment. The external function $compute_impulse$ is recalled at the beginning of the state and its outputs are given an initial value, necessary to make the simulation work.

```
1    t0=t;
2        x_curr=p(1);
3        z_curr=p(3);
4        dx_curr = dp(1);
5        dz_curr = dp(3);
```

The state captures the state of the chaser when the emergency state is activated. The In-Track and Radial position and velocity are respectively defined as $x_curr$,$z_curr$,$dx_curr$ and $dz_curr$.

```
1        if x_curr<25 && x_curr>=20
2        t_opt=400;
3        elseif x_curr<20 && x_curr >=15
4        t_opt=300;
5        else
6        t_opt=200;
```

```
        end
```

The TOF is defined with respect to the horizontal coordinate where the CAM is triggered.

```
        x_threshold=8;
        if x_curr >= x_threshold
        scenario=2;
        x_current = [x_curr; 0; z_curr; dx_curr; 0; dz_curr];
        %r_target = [30; 0; -30*tand(3)];
        r_target = [30; 0; WP];
        [dx_req, dz_req] = compute_impulse(x_current, r_target,
           omega,t_opt);
```

The threshold is imposed conservatively at 8 meters from the target for safety reasons, to avoid colliding against the approach cone. The parameter $scenario$ is another output useful within a Simulink block. $r_target$ may be varied, in this case the radial coordinate, called $WP$ was randomly computed in the Matlab workspace. The function initially recalled here computes the necessary $\Delta Vs$ to execute the manoeuvre, here called $dx_req$ and $dz_req$.

```
        dx0=dx_req+dx_curr;
        dz0=dz_req+dz_curr;
        t_burn_x = M0*abs(dx_req)/F_max;
        t_burn_z = M0*abs(dz_req)/F_max;
```

$dx0$ and $dz0$ on the contrary represent the total initial velocity after the impulse has been transmitted. These are parameters used in closed-loop architectures to define the ideal trajectory but become useless in an open-loop environment. $t_burn_x$ and $t_burn_z$ are the duration of the thrust respectively along the In-Track and Radial directions: they are going to be used in a time-based sub-state to command the thruster as long as necessary.

```
        else
        hold=1;
        scenario=1;
        dx0=0;
        dz0=0;
        t_opt=inf;
        t_burn_x=0;
        t_burn_z=0;
        end
```

Whenever the position of the chaser is under 8 meters from the target, another scenario is activated but does not have validity within the work of this thesis since it includes just the stop of the chaser and its confinement within the last known position.

```
        x_EMcy = x_curr;
        x_STOP=x_curr;
```

```
        z_STOP=z_curr;
        %during:
        %if abs(p(3)/p(1)) > tand(5)
        % stopFLAG=1;
        % fprintf('WP non accettabile = %.2f s\n', WP)
        %else
        % stopFLAG=0;
        %end
```

$x_EMcy$ is just a variable saved in the workspace for the construction of future plots, while $x_STOP$ and $z_STOP$ both were used in closed-loop architectures as well. The last part was used during the Monte Carlo simulations to stop the run whenever the approach cone was crossed.
The function recalled during the execution of the emergency state is here reported:

```
        function [dVx, dVz] = compute_impulse(x_current, r_target,
           omega,dt)

        r_0 = x_current(1:3);
        r_0 = r_0(:);

        v_0 = x_current(4:6);
        v_0 = v_0(:);

        r_target = r_target(:);


        sin_wt = sin(omega*dt);
        cos_wt = cos(omega*dt);

        Phi_rr = [ 1, 0, 6*(omega*dt - sin_wt);
        0, cos_wt, 0;
        0, 0, 4 - 3*cos_wt ];

        Phi_rv = [ (4/omega)*sin_wt - 3*dt, 0, (2/omega)*(1 -
           cos_wt);
        0, (1/omega)*sin_wt, 0;
        (2/omega)*(cos_wt - 1), 0, (1/omega)*sin_wt ];

        v_req_0 = Phi_rv\(r_target - Phi_rr * r_0);

        delta_V = v_req_0 - v_0;

        dVx = delta_V(1);
        dVz = delta_V(3);
```

The function is simply the expression of HCW equations in matrix form, based on section 22.4.1 of [[36]]. The equations are divided into two part, here reunited in the

matrices *Phi_rr* and *Phi_rv*, and the required impulses are simply computed with an inverse operation.
The emergency state is left after *t_opt* seconds, which are initially defined depending on the In-Track position of the chaser. A second impulse is therefore computed in another state:

```
SECONDIMPULSE
entry:
coder.extrinsic('compute_impulse');
t0=t;
dx_req=0.0;
dz_req=0.0;
scenario=2;
x_current = [p(1); 0; p(3); dp(1); 0; dp(3)];
r_target = [100; 0; 0];
t_opt=0.75*T;
[dx_req, dz_req] = compute_impulse(x_current, r_target,
   omega,t_opt);
dx0=dx_req+dp(1);
dz0=dz_req+dp(3);
x_STOP=p(1);
z_STOP=p(3);
t_burn_x = M0*abs(dx_req)/F_max;
t_burn_z = M0*abs(dz_req)/F_max;
```

The instructions are totally analogous to the ones above described. The trasfer time is now 0.75T due to safety reasons because the natural evolution of trajectories with short TOF, such as 0.25T, produce dangerous comebacks towards the target.

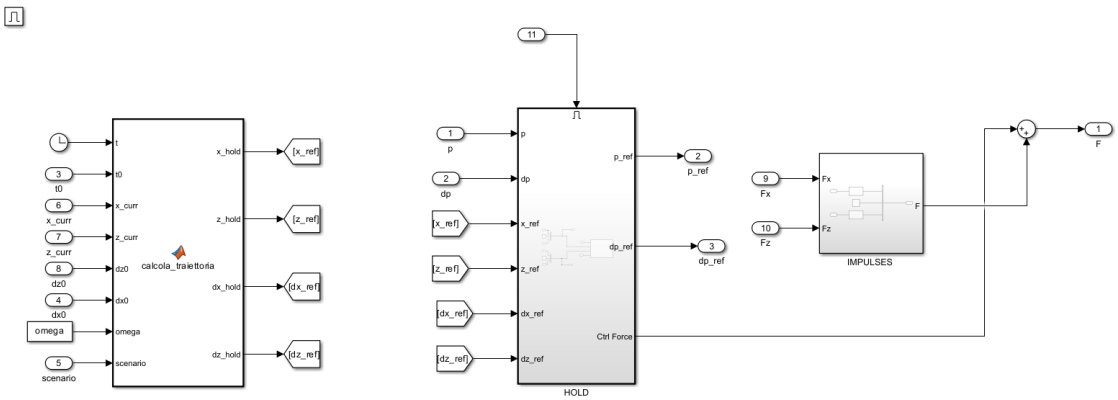



Figure C.1: Simulink block dedicated to the handling of CAM strategies

Figure (C.1) illustrates the blocks involved for the correct execution of the CAM. In Open-Loop, only the third block, called *IMPULSES* is actually used. Its function is to elaborate and merging the different thrusts into a single vector which is going to be used within the NEXUS propagator to obtain the chaser's relative state. The function *calcolo_traiettoria* adopts the HCW equations to propagate the ideal trajectory,

represented by its outputs $x_ref$, $z_ref$, $dx_ref$ and $dz_ref$ which are used in a controller block, here called $HOLD$ to maintain the chaser adherent to the imposed trajectory. Obviously, this strategy is valid only in Closed-Loop architectures.
Lastly, a more detailed comment on NEXUS is left for the reader. NEXUS implements a high-fidelity numerical propagator for the orbital state of a single spacecraft, expressed in the ECI frame. The model is initialised from a companion MATLAB script, which defines the physical constants (Earth gravitational parameter, equatorial radius, J2 coefficient, Earth rotation rate), the third-body gravitational parameters of the Sun and Moon, the epoch, the spacecraft's initial orbital elements and physical properties such as mass, drag and Solar Radiation Pressure coefficients and areas.
The top-level architecture is organised into four functional blocks. The Environment subsystem computes the instantaneous position of the Sun and the Moon in the ECI frame via two dedicated position-model subsystems, providing the ephemeris data required by the perturbation models. The Force Model subsystem aggregates all accelerations acting on the spacecraft and outputs their sum to the integrator. Within it, the gravitational acceleration is computed by a variant subsystem that switches, based on a workspace-defined flag, between a simple J2-only analytical formulation and a full spherical-harmonics expansion of the Earth's gravity field; both branches share the same interface so that the switch is transparent to the rest of the model. Atmospheric drag is computed from an atmospheric density model, parametrised by a solar-activity level (low/medium/high), combined with the spacecraft's relative velocity, drag coefficient, cross-sectional area and mass. Solar radiation pressure is computed from the Sun's relative position, the spacecraft's reflectivity coefficient and area, and an eclipse/shadow function that determines the illumination fraction of the spacecraft along its orbit. Third-body perturbations from the Sun and Moon are computed from their relative positions and gravitational parameters. All of these perturbation models are implemented as embedded MATLAB System blocks (S-Functions) wrapped with demultiplexer and terminator blocks to manage their vector outputs. The Integrator subsystem receives the total acceleration and numerically integrates it twice, propagating the spacecraft's position and velocity from the specified initial conditions. Finally, the Conversions subsystem post-processes the propagated state, computing the spacecraft's altitude above the Earth's surface and the corresponding set of classical orbital elements for monitoring and validation purposes.

# Bibliography


[1] W. Fehse, *Automated Rendezvous and Docking of Spacecraft* (Cambridge Aerospace Series). Cambridge University Press, 2003, vol. 16, ISBN: 978-0-521-82492-7.

[2] L. Deva, G. Ammirante, G. Taiano, et al., "Sroc: Space rider observer cube mission," in *Small Satellites Systems and Services Symposium*, Pula, Italy, 2026.

[3] W. J. Larson and J. R. Wertz, Eds., *Space Mission Analysis and Design*, 2nd. Dordrecht: Microcosm and Kluwer Academic Publishers, 1992.

[4] C. Pirat, "Guidance, navigation and control for autonomous rendezvous and docking of nano-satellites," Ph.D. Thesis, École Polytechnique Fédérale de Lausanne (EPFL), Dec. 2018.

[5] G. Ichino, "Performance evaluation of a vision based navigation algorithm for cubesat docking missions," Master's Thesis, Politecnico di Torino, 2021. [Online]. Available: `https://webthesis.biblio.polito.it/21142/1/tesi.pdf`

[6] S. V. Emelyanov, *Variable Structure Automatic Control Systems*. Moscow, Russia: Nauka, 1967.

[7] S. V. Emelyanov, *Theory of Variable Structure Systems*. Moscow, Russia: Nauka, 1970.

[8] E. A. Barbashin, *Introduction to the Theory of Stability*. Moscow, Russia: Nauka, 1967.

[9] E. A. Barbashin and E. I. Gerashchenko, "Forced sliding regimes in automatic control systems," *Differential Equations*, vol. 1, no. 1, pp. 16–20, 1965.

[10] E. A. Barbashin, A. Tabueva, and R. M. Eidinov, "On the stability of variable structure control systems when sliding conditions are violated," *Automation and Remote Control*, vol. 7, pp. 81–16, 1963.

[11] E. Capello, "Dynamics and control of space vehicles," Master's Degree course lectures notes, Politecnico di Torino, Academic Year 2025/2025, 2024.

[12] S. Russo, "Development of attitude and trajectory control strategies for small satellites involved in rendez-vous and docking operations," Master's Degree Thesis, Politecnico di Torino, 2021.

[13] I. Ahmad, M. Liaquat, F. Mumtaz, H. Ullah, and U. Ali, "Variants of the sliding mode control in presence of external disturbance for quadrotor," *IEEE Access*, vol. 8, pp. 227 810–227 824, Dec. 2020. DOI: `10.1109/ACCESS.2020.3041678`

[14] I. A. Spiegel, B. Zhou, R. Goodloe, B. Fox, and J. DiMatteo, "Cubesat proximity operations demonstration (cpod) mission results," in *37th Annual Small Satellite Conference*, Paper SSC23-XI-01, 2023.

[15] A. D'Ortona, L. Lovaglio, and F. Stesina, "Adaptive model predictive control with online parameter learning during spacecraft proximity operations," *IFAC PapersOnLine*, vol. 58, no. 16, pp. 235–240, 2024. DOI: 10.1016/j.ifacol.2024.08.492

[16] R. Issermann, *Digital Control Systems*. Berlin: Springer-Verlag, 1981.

[17] R. Brown and Y. Hwang, *Introduction to Random Signals and Applied Kalman Filtering*. New York: Wiley and Sons, 1992.

[18] G. Franklin, J. Powell, and M. Workman, *Digital Control of Dynamic Systems*, 3rd. Menlo Park, CA: Addison-Wesley, 1998.

[19] H.-J. Heo, Y. Son, and J.-M. Kim, "A trapezoidal velocity profile generator for position control using a feedback strategy," *Energies*, vol. 12, no. 7, p. 1222, 2019. DOI: 10.3390/en12071222

[20] A. Sortino, "Development of guidance and control strategies for a small satellite involved in proximity operations," Master's Thesis, Politecnico di Torino, 2025. [Online]. Available: https://webthesis.biblio.polito.it/id/eprint/38562

[21] Y. Di Crescenzio, "The challenges of the fdir design in hera spacecraft: Spacecraft autonomy and interaction with the cubesats," in *Small Satellites Systems and Services Symposium (4S 2024)*, M. Petrozzi-llstad, Ed., ser. Proceedings of SPIE, International Society for Optics and Photonics, vol. 13546, 2025, p. 1 354 638. DOI: 10.1117/12.3062643

[22] A. Zolghadri, "Advanced model-based fdir techniques for aerospace systems: Today challenges and opportunities," *Progress in Aerospace Sciences*, vol. 53, pp. 18–29, 2012. DOI: 10.1016/j.paerosci.2012.02.004

[23] T. Guffanti, T. Bell, S. Y. W. Low, M. Murray-Cooper, and S. D'Amico, "Autonomous guidance navigation and control of the visors formation-flying mission," *arXiv preprint arXiv:2309.16698*, 2023, Preprint AAS 23-149.

[24] A. Cervone et al., "The path towards increasing rams for novel complex missions based on cubesat technology," *CEAS Space Journal*, vol. 16, pp. 203–224, 2024. DOI: 10.1007/s12567-023-00517-9

[25] A. Pellacani, P. Kicman, M. Suatoni, M. Casasco, J. Gil, and I. Carnelli, "Design and validation of a gnc system for missions to asteroids: The aim scenario," *CEAS Space Journal*, vol. 10, pp. 555–566, 2018. DOI: 10.1007/s12567-017-0189-x

[26] I. Latachi, T. Rachidi, M. Karim, and A. Hanafi, "Reusable and reliable flight-control software for a fail-safe and cost-efficient cubesat mission: Design and implementation," *Aerospace*, vol. 7, no. 10, p. 146, 2020. DOI: 10.3390/aerospace7100146

[27] J. B. Mueller and R. Larsson, "Collision avoidance maneuver planning with robust optimization," in *7th International ESA Conference on Guidance, Navigation & Control Systems*, Tralee, County Kerry, Ireland, Jun. 2008.

[28] J. Bowen, M. Villa, and A. Williams, "Cubesat based rendezvous, proximity operations, and docking in the cpod mission," in *29th Annual AIAA/USU Conference on Small Satellites*, Paper SSC15-III-5, 2015.

[29] E. de Pasquale, "Atv jules verne: A step by step approach for in-orbit demonstration of new rendezvous technologies," in *American Institute of Aeronautics and Astronautics (AIAA) Conferences*, AIAA, 2012. DOI: 10.2514/6.2012-1314619

[30] B.-C. Lin, C.-W. Kong, S. Semeraro, and J. W. McMahon, "A control framework for cubesat rendezvous and proximity operations using electric propulsion," *arXiv preprint arXiv:2412.00827*, 2024, Preprint AAS 23-296.

[31] C. W. T. Roscoe, J. J. Westphal, and E. Mosleh, "Overview and gnc design of the cubesat proximity operations demonstration (cpod) mission," *Acta Astronautica*, vol. 153, pp. 410–421, 2018. DOI: 10.1016/j.actaastro.2018.03.033

[32] D. González-Rodríguez, P. Orgeira-Crespo, C. Cappelletti, and F. Aguado-Agelet, "Methodology for cubesat debris collision avoidance based on its active adcs system," *Applied Sciences*, vol. 13, no. 22, p. 12 388, 2023. DOI: 10.3390/app132212388

[33] The MathWorks, Inc., *Stateflow: Model and simulate decision logic using state machines and flow charts*, https://www.mathworks.com/products/stateflow.html, Accessed: 2026-06-20, 2026.

[34] The MathWorks, Inc., *Orbit propagator numerical (high precision)*, https://www.mathworks.com/help/aeroblks/orbitpropagatornumericalhighprecision.html, Accessed: 2026-09-07, 2026.

[35] E. Desplats, "Description of the refined plume impingement model, wp241 of esa contract rvd-gsp," MATRA, Technical Report S295/NT/39.88, 1988, Unpublished.

[36] C. Pirat, "Close-proximity operations, formation flying, and on-orbit servicing," in *Next Generation CubeSats and SmallSats*, Elsevier, 2023, ch. 22, pp. 543–602. DOI: 10.1016/B978-0-12-824541-5.00007-8