UNIVERSITY OF NAPLES FEDERICO II

DEPARTMENT OF INDUSTRIAL ENGINEERING

PH.D. COURSE IN INDUSTRIAL ENGINEERING

COORDINATOR PROF. MICHELE GRASSI

XXXV CYCLE

PH.D. THESIS

GNC TECHNOLOGIES AND ALGORITHMS FOR CLOSE-PROXIMITY FLIGHT AND RE-ENTRY APPLICATIONS

By

Alessia Nocerino

Supervisors:

Prof. Roberto Opromolla

Prof. Giancarmine Fasano

Prof. Michele Grassi

Abstract

The objective of this Ph.D. thesis is the design, development and performance assessment of innovative Guidance Navigation and Control techniques enabling the autonomous execution of complex tasks required by future space missions, such as the close proximity maneuvering of a chaser spacecraft around a resident space object and the controlled de-orbiting of micro-satellite by means of aerodynamic drag.

Regarding close-proximity operations, both Active Debris Removal and In-Orbit Servicing missions requires an autonomous spacecraft (chaser) to safely monitor and then approach an active/inactive artificial space object (target) which may be or not equipped with artificial markers to aid the relative navigation task. In this framework, this thesis proposes two original relative navigation architecture to be applied in the monitoring and close-approach phase of an ADR/IOS mission. For the monitoring phase, an original multi-step architecture for the estimation of both relative motion parameters and inertia parameters of an uncooperative space target is proposed. Once the position and attitude (pose) parameters are initialized (first step), LIDAR-based pose measurements and a smoothing approach are used to retrieve accurate, linearly independent estimates of the target angular velocity. These estimates are then used to compute the target's moment of inertia ratios solving a linear system based on the conservation equation for the angular momentum. Once the inertia parameters are accurately estimated, the LIDAR-based pose measurements are used to feed a Kalman Filter to determine the full relative state according to a loosely coupled configuration.

In the final approach phase, when the chaser has to capture the target by means of a robotic arm, a second EO sensor (TOF camera) is installed on the tip of the end-effector in order to get direct pose measurements of the end effector with respect to the selected grasping point. The measurements of the two EO-sensors are integrated within two

different Kalman Filters aimed at the estimate of the target-chaser and end effector-robotic arm relative motion parameters.

Performance assessment is carried out through numerical simulations realistically reproducing close-range relative motion dynamics and LIDAR sensor operation, and considering targets characterized by highly variable size, shape, and orbital dynamics as test cases. The moments of inertia estimation algorithm has been validated experimentally within a set-up simulating the tumbling motion of an uncooperative space target.

Controlled reentry technologies also play a fundamental role to ensure future sustainability of the space environment. In this framework, the problem of aerodynamic re-entry is addressed in this thesis by designing and developing a control system aiming at modulating a deployable aerobrake to make the satellite follow nominal decay path using an umbrella-like actuator. Performance assessment of the proposed Linear Quadratic Regulator-based control techniques has been carried out within a numerical simulation environment which reproduces both the environmental perturbations and the actuators constraints.

KEYWORDS: autonomous GNC, spacecraft relative navigation, uncooperative space target, inertia parameters estimation, de-orbiting control, deployable aerobrake.

Table of Contents

List of Figures

List of Tables

List of Acronyms

ADR Active Debris Removal

CKF Cubature Kalman Filter

COG Center of Gravity

COM Center of Mass

COR Center of Rotation

CPOs Close Proximity Operations

CSEKF Constant State Extended Kalman Filter

CW Clohessy-Wiltshire

DH Denavit-Hartenberg

DOF Degree of Freedom

EKF Extended Kalam Filter

EO Electro-Optical

FOV Field of View

FFT Fast Fourier Transform

ICP Iterative Closest Point

InCKF Interaction CKF

IEKF Iterated Extended Kalman Filter

IOS In-Orbit Servicing

LEO Low Earth Orbit

LQR Linear Quadratic Regulator

GEO Geostationary Earth Orbit

GNC Guidance Navigation & Control

KF Kalman Filter

MIR Moment of Inertia Ratio

MPC Model Predictive Control

PCA Principal Component Analysis

RMS Root Mean Square

SLAM Simultaneous Localization and Mapping

SNR Signal to Noise Ratio

STM State Transition Matrix

TOF Time of Flight

TM Template Matching

UKF Unscented Kalman Filter

1. Introduction

This thesis focuses on the development and performance assessment of innovative Guidance Navigation and Control functions enabling the autonomous execution of close proximity maneuvers of a chaser spacecraft around a resident space object and the controlled de-orbiting of micro-satellite by means of aerodynamic drag.

1.1 Guidance, Navigation and Control Architecture for close-proximity operations

Since last decades, the space community is paying an increasing attention towards the theme of the sustainability of the space environment. Indeed, the feasibility of future space operations, especially in the most crowded regions such as Low Earth Orbit (LEO) and Geostationary Earth Orbit (GEO), is threatened by the presence of space debris, [1], [2]. They include spent rocket bodies, inactive satellites, mission-related objects and products of fragmentations caused by explosions and/or collisions. Their uncontrolled increase could lead to the triggering of a catastrophic chain of collisions, named Kessler effect, [3], [4]. Solutions proposed by the scientific community to address this problem include the introduction of passive mitigation [5], as well active remediation measures, [6]. Regarding passive mitigation measures, until 2022, it was recommended to satellite operators to ensure that their spacecraft decay and re-enter Earth atmosphere within no more than 25 years, [5]. Recently, the 25-year benchmark for post-mission disposal in LEO has been shortened to 5 years. However, many studies have highlighted the need of complementing passive measures with active ones [7]–[9]: they include Active Debris Removal (ADR), i.e., actively removing a defunct satellite from its orbit, and In-Orbit Servicing (IOS) operations which consist in performing refueling, repairing and maintenance operations to active spacecraft to extend its operative life. Additionally, the possibility to upgrade already in-orbit platforms by performing IOS operations plays a strategic role for space agencies and industries due to the potentially associated economic benefits. Anyway, carrying out these missions, require a servicing spacecraft performing high risk maneuvers in close proximity of a resident space object. In this framework the level of autonomy of the spacecraft, especially regarding their Guidance, Navigation and Control (GN&C) must be enhanced. The reason is twofold. On one side, the lack of coverage and communication delays may make tricky to rely on commands from ground stations; on the other hand, high degree of autonomy provides the possibility to increase frequency and reliability of future space missions. The main challenge comes from the fact that most of the potential ADR and IOS target lack of a dedicated radio-link to interact and exchange information with the servicing spacecraft. In this regard, a distinction can be made between semi-cooperative and non-cooperative targets: the formers are equipped with artificial markers and/or grappling fixtures to ease rendezvous and capture operations.

Regardless of the degree of cooperativeness of the target, a GN&C architecture for close-proximity operations must rely on the measurements provided by Electro-Optical (EO) sensors, either active (e.g., LIDAR system) or passive (e.g., monocular camera and stereovision system), [10] to accomplish the relative navigation task. Indeed, these sensors can image the target with adequate resolution to ensure the determination of the 6 Degree of Freedoms (DOFs) parameters describing the relative position and attitude (pose) between the two spacecraft. Clearly, the EO-based measurements must be integrated into a filtering architecture (also fed by chaser absolute state information) to fully characterize the relative motion with respect to the target (i.e., by also estimating the relative velocity and angular velocity information). All these relative state measurements are then required to compute the control actions to be exerted by the servicing spacecraft to carry out the planned mission.

Focusing on the relative navigation function, two major architectural choices can be made, thus allowing a distinction between tightly coupled or loosely coupled approaches. In the former, the raw data from EO sensors (i.e., images or point clouds) are processed to detects a set of features, i.e., fiducial markers in case or semi-cooperative target or selected geometric structures such as the launch adapter ring or the apogee motor when dealing with uncooperative target, whose position is included within the state vector of the relative navigation filter [11]. In the loosely coupled case, raw sensor data are processed by a separate pose determination block to obtain relative position and attitude observables which are used to update the navigation filter estimates [12]. Although the use of tightly coupled configurations can have some advantages when the target is uncooperative and its shape unknown, they require the development of feature detection algorithms robust enough to deal with recursive appearance/disappearance of features and a computational effort which increases with the number of features to be tracked. Thus, the use of loosely coupled architecture is generally preferred thanks to their robustness to self-occlusion phenomena and the fact that they allow realizing modular architecture in which different algorithmic approaches can be plugged-in to adapt it to the operational scenario without significantly affecting its structure.

In this context, this thesis addresses the relative navigation task of a GNC architecture enabling close-proximity operations with respect to both semi-cooperative (i.e., equipped with fiducial markers to ease the relative navigation function) and uncooperative target. Additionally, while the geometry of an uncooperative target is typically available, there could be a significant uncertainty in the knowledge of its inertia parameters due to the prolonged period spent in orbit (e.g., due to exhaustion of propellant, collisions or explosions) which are needed to estimate the rotational dynamics during approach and capture operations and to control the stack in the post-capture phase. In this regard, this thesis also presents an approach for the estimate of the inertia parameters exploiting the measurements of an EO-sensor and the conservation of the angular momentum law to estimate the moment of inertia ratios of a tumbling and uncooperative space target. The architecture presented in this thesis is tested within a numerical simulation environment which can simulate relevant ADR and IOS scenarios by realistically reproducing both the operation of passive and active EO sensors, i.e., monocular cameras and LIDAR systems, as well the relative motion between two spacecraft. Additionally, the performance of the inertia estimation algorithms has been assessed within an experimental set-up which simulates the tumbling motion of an uncooperative space target tracked by a solid-state LIDAR. This activity has been carried out during a visiting period at the ADAMUS laboratory of Embry-Riddle Aeronautical University (FL, USA) in the framework of *Programma Star – Linea 2* financially supported by the University of Naples "*Federico II*" and *Compagnia San Paolo*.

1.2 Trajectory control algorithms for re-entry applications

The aerodynamic forces experienced by a spacecraft in LEO can be controlled and exploited to achieve useful purposes such as planetary aerocapture [13], rendezvous with another vehicle [12]-[13], orbital transfer [16], constellation maintenance [17] and attitude control [18]. In this context, deployable aerobrake may offer many advantages in near future, including the opportunity to control the re-entry trajectory and correctly guide the spacecraft toward a selected landing site in order to recover payloads and samples from space with reduced risks and costs with respect to conventional systems. Indeed, the aerobrake can be deployed to modify the cross-sectional area of the spacecraft and, therefore, the ballistic coefficient, thus offering the opportunity to perform de-orbiting without the need of a dedicated propulsive subsystem and re-entry with reduced aero-thermal and mechanical loads.

The concept of mechanical deployable aerobrake has been proposed for the first time in the late eighties with ParaShield [19] and Bremsat [20]. Recently, NASA has tested this technology in a suborbital mission, ADEPT [21] while ESA is developing a mechanical deployable heat shield called IRENE [22] which will also be tested in a suborbital mission with a scaled-down prototype called Mini IRENE [23]. Also, the possibility to exploit inflatable system to deploy heat shields has been investigated and tested in the frame of IRDT ESA [24] and IRVE NASA programs [25].

Regardless of the system used to deploy the aerobrake, the de-orbiting and re-entry of a small spacecraft is a critical phase of the mission, where model uncertainties and environmental disturbances can cause a significant deviation of the descending path, and thus the landing point, from the reference ones. Therefore, the development of a control system able to cope with all the uncertainties is vital for the success of the mission.

In the scientific literature, the de-orbit and re-entry control problems have been addressed with several approaches. Typically, a nominal trajectory is designed, and then a trajectory tracking feed-back controller is used to counteract uncertainties and external disturbances.

In this frame, the second part of this thesis will focus on the development and performance assessment of a trajectory control algorithm for a micro-satellite equipped with a deployable and controllable aerobrake. This activity has been carried out in the frame of MISTRAL (Micro-Satellite with Air-Launchable Re-entry capabilities) project [26], under the supervision of the Campania Aerospace District (DAC) and in collaboration with a cluster of industries, research organization and universities, whose primary scope is the design of a multi-purpose air-launchable 50 kg class micro-platform with re-entry capability, equipped with a small recovery module.

The developed trajectory control algorithm has been tested in a numerical simulation scenario in which all the relevant perturbations for the scenario under study have been included along with the mechanical limitations and constraints posed by the presence of the umbrella like actuator.

1.3 Thesis organization

The rest of this thesis is organized as follows.

Chapter 2 presents the state of the art about relative navigation architectures for close-proximity applications. Also, the current existing techniques for the estimate of the inertia properties of uncooperative and tumbling targets are presented. This chapter also highlights the contribution of this thesis with respect to the state of the art. Chapter 3 presents in detail the GNC architecture with a particular focus on the relative navigation task. Specifically, two GNC modes are detailed: the monitoring, during which the servicer flies around the target following a trajectory which satisfy some safety constraints to inspect it and estimate its inertia properties and the final approach when the chaser approaches the target and, in case of berthing, the robotic arm grasp the selected grappling fixture. Chapter 4 describe the simulation environment built to assess the performance of the techniques discussed in Chapter 3 and presents the scenarios simulated in terms of sensors' characteristics, target geometries and relative trajectories. Also, the performance of the GNC architecture in both monitoring and close-range phase are presented and discussed. The experimental validation of the inertia parameters estimation algorithm with a detailed description of the experimental set-up is presented in Chapter 5. Chapter 6 focuses on the aerodynamic-based re-entry applications. It presents the state of the art about trajectory tracking algorithms for passive deorbiting of spacecraft exploiting aerodynamic drag. Then it presents the developed de-orbiting control algorithm and the simulations environment in which its performance are evaluated together with a discussion on the achieved results. Finally, Chapter 7 draws some conclusions on this work and discusses the possible future activities related to it.

2. Autonomous GN&C architecture for close-proximity operations: state of the art

2.1 Introduction

In the last decades, many solutions exploiting different technological architectures have been investigated to enable operations in close proximity of a resident space object, with a particular focus on the navigation function. In this regard, the sensor configuration can be passive (i.e., based on monocular camera [27] or stereo cameras [28]), active (i.e., LIDAR [29], [30] or Time of Flight cameras [31]) and hybrid [32], [33]. For instance, in the recent RemoveDEBRIS mission (i.e., a demonstration mission of many active debris removal technologies carried out using two CubeSats walking away from each other after the ejection from the ISS), one of the four experiments was aimed at testing different vision-based relative navigation technologies, namely a panchromatic camera, a color camera and a Flash LIDAR, [34], [35].

2.2 Relative navigation filters

Regardless of the selection of the most suitable sensor suite, the EO-based sensor measurements must be integrated within a filtering scheme to fully characterize the relative motion between the servicer and the target. Different solutions are possible regarding the filtering scheme suitable to estimate the full target-chaser relative navigation state, including Kalman filters, minimum energy filter and $H\infty$ filters. Along with the filtering scheme, a dynamic model describing the temporal evolution of the system state, and a measurement model which establishes the relation between the available measurements and the state variables, must be selected.

2.2.1 Filtering schemes

Kalman filter

Kalman Filters (KF) are used whenever there are noisy information coming from different sources to get an estimate of the system state by minimizing the mean square errors between the estimated and the true state. One of the prerequisites to obtain an optimal solution is that the dynamics of the observed system must be governed by a linear stochastic equation [36], as shown in Eq. (1)

$$\boldsymbol{x}_k = \underline{\underline{A}}_{k-1}\boldsymbol{x}_{k-1} + \underline{\underline{B}}\boldsymbol{u}_{k-1} + \boldsymbol{w}_{k-1} \tag{1}$$

where k, is the time instant, $\boldsymbol{x}$ represents the state of the system, $\underline{\underline{A}}$ is the state transition matrix (STM), $\underline{\underline{B}}$ is the input-to-state matrix, $\boldsymbol{u}$ is the input of the system and $\boldsymbol{w}$ represents the process noise. Also, the available measurements, $\boldsymbol{z}$, must be related to the system state through a linear relation (the observation equation), as in Eq. (2)

$$\boldsymbol{z}_k = \underline{\underline{H}}_k\boldsymbol{x}_k + \boldsymbol{v}_k \tag{2}$$

where $\underline{\underline{H}}$ is the measurement matrix and $\boldsymbol{v}$ represents the measurement noise.

Under the assumption that both the process and the measurement noise are characterized by zero-mean white Gaussian distributions, as shown in Eq. (3) and (4), the solution provided by the Kalman filter is the optimal one.

$$p(\boldsymbol{w}) \sim N\left(0, \underline{\underline{Q}}\right) \tag{3}$$

$$p(\boldsymbol{v}) \sim N\left(0, \underline{\underline{R}}\right) \tag{4}$$

In Eq. (3) and (4) the matrices $\underline{\underline{Q}}$ and $\underline{\underline{R}}$ are the covariances of the process and measurement noise, respectively.

Given the knowledge of the system state at the time instant *k-1*, i.e., an *a-posteriori* state and error covariance estimate, $\hat{\boldsymbol{x}}_{k-1}^{+}$ and $\underline{\underline{P}}_{k-1}^{+}$, an *a-priori* state and error covariance estimate at time *k* is obtained through the time-update equation, shown in Eq. (5) and (6).

$$\hat{\boldsymbol{x}}_{k}^{-} = \underline{\underline{A}}\hat{\boldsymbol{x}}_{k-1}^{+} + \underline{\underline{B}}\boldsymbol{u}_{k-1} \tag{5}$$

$$\underline{\underline{P}}_{k}^{-} = \underline{\underline{A}}\underline{\underline{P}}_{k-1}^{+}\underline{\underline{A}}^{T} + \underline{\underline{Q}}_{k} \tag{6}$$

Finally, the *a-posteriori* state and error covariance estimate at time *k*, are computed as a linear combination of the a priori-state estimate, $\hat{\boldsymbol{x}}_{k}^{-}$ and a weighted difference between the actual measurement vector, $\boldsymbol{z}_{k}$, and the measurement prediction $\underline{\underline{H}}_{k}\hat{\boldsymbol{x}}_{k}^{-}$, as shown in Eq. (7) and (8).

$$\hat{\boldsymbol{x}}_{k}^{+} = \hat{\boldsymbol{x}}_{k}^{-} + \underline{\underline{K}}_{k}(\boldsymbol{z}_{k} - \underline{\underline{H}}_{k}\hat{\boldsymbol{x}}_{k}^{-}) \tag{7}$$

$$\underline{\underline{P}}_{k}^{+} = \left(\underline{\underline{I}} - \underline{\underline{K}}_{k}\underline{\underline{H}}_{k}\right)\underline{\underline{P}}_{k}^{-}\left(I - \underline{\underline{K}}_{k}\underline{\underline{H}}_{k}\right) + \underline{\underline{K}}_{k}\underline{\underline{R}}_{k}\underline{\underline{K}}_{k}^{T} \tag{8}$$

in which $\underline{\underline{K}}_{k}$ is the Kalman gain, chosen as the factor which minimizes the *a-posteriori* error covariance. The Kalman gain is compute through Eq. (9).

$$\underline{\underline{K}}_{k} = \underline{\underline{P}}_{k}^{-}\underline{\underline{H}}_{k}^{T}\left(\underline{\underline{H}}_{k}\,\underline{\underline{P}}_{k}^{-}\underline{\underline{H}}_{k}^{T} + \underline{\underline{R}}_{k}\right)^{-1} \tag{9}$$

Extended Kalman filter

However, most of the systems are not governed by non-linear process. When dealing with non-linear systems, modified version of the classical KF can be adopted. For instance, the Extended Kalman Filter (EKF) linearizes both the time-update equation and the observation equation around the current estimate of the state and covariance with a first-order Taylor approximation, [37]. Thus, if the governing equation is nonlinear, the process equation has the following form:

$$\boldsymbol{x}_k = f(\boldsymbol{x}_{k-1}, \boldsymbol{u}_{k-1}, \boldsymbol{w}_{k-1}) \tag{10}$$

Moreover, if the observation equation is nonlinear, Eq. (2) becomes:

$$\boldsymbol{z}_k = h(\boldsymbol{x}_k, \boldsymbol{v}_k) \tag{11}$$

The *a-priori* estimates of the state and error covariance are given by Eq. (12) and (13).

$$\hat{\boldsymbol{x}}_k^- = f(\hat{\boldsymbol{x}}_{k-1}^+, \boldsymbol{u}_{k-1}) \tag{12}$$

$$\underline{\underline{P}}_k^- = \underline{\underline{\Phi}}_k \underline{\underline{P}}_{k-1}^+ \underline{\underline{\Phi}}_k^T + \underline{\underline{Q}}_k \tag{13}$$

where $\underline{\underline{\Phi}}$ is the STM derived as a first-order Taylor expansion of the time update equation, and it can be computed as in Eq. (14).

$$\underline{\underline{\Phi}}_k = I + \left.\frac{\partial f}{\partial \boldsymbol{x}}\right|_{\hat{\boldsymbol{x}}_{k-1}^+, \boldsymbol{u}_{k-1}} (t_k - t_{k-1}) \tag{14}$$

Also, the equations providing the *a-posteriori* state and error covariance estimates are modified with respect to Eq. (7) and (8) as follows:

$$\hat{\boldsymbol{x}}_k^+ = \hat{\boldsymbol{x}}_k^- + \underline{\underline{K}}_k(\boldsymbol{z}_k - h(\hat{\boldsymbol{x}}_k^-)) \tag{15}$$

$$\underline{\underline{P}}_k^+ = \left(\underline{\underline{I}} - \underline{\underline{K}}_k \underline{\underline{H}}_k\right) \underline{\underline{P}}_k^- \left(I - \underline{\underline{K}}_k \underline{\underline{H}}_k\right) + \underline{\underline{K}}_k \underline{\underline{R}}_k \underline{\underline{K}}_k^T \tag{16}$$

where the Kalman gain is computed according to Eq. (9) and $\underline{\underline{H}}$, called sensitivity matrix, is computed as the partial derivative of the measurement function $\boldsymbol{h}$ with respect to the state evaluated at the current *a-posteriori* estimate, $\hat{\boldsymbol{x}}_{k-1}^+$,as shown in Eq. (17).

$$\underline{\underline{H}}_k = \left.\frac{\partial h}{\partial \boldsymbol{x}}\right|_{\hat{\boldsymbol{x}}_{k-1}^{+}} \tag{17}$$

However, the performance of the EKF scheme can deteriorate if the measurement equations are strongly non-linear. In this scenario, the Iterated Extended Kalman Filter (IEKF) can improve the estimation performance by means of local iteration of the update step. As in many iterative methods, the iteration procedure stops when there is no significant difference in consecutive iterations or when a user-defined maximum number of iterations is reached. At each iteration, the sensitivity matrix, the Kalman gain and thus the *a-posteriori* state estimates are re-computed as follows:

$$\underline{\underline{H}}_k^i = \left.\frac{\partial h}{\partial \boldsymbol{x}}\right|_{\hat{\boldsymbol{x}}_{k,i}^{+}} \tag{18}$$

$$\underline{\underline{K}}_k = \underline{\underline{P}}_k^- \underline{\underline{H}}_k^{i\,T} \left(\underline{\underline{H}}_k^i \underline{\underline{P}}_k^- \underline{\underline{H}}_k^{i\,T} + \underline{\underline{R}}_k \right)^{-1} \tag{19}$$

$$\hat{\boldsymbol{x}}_{k,i}^{+} = \hat{\boldsymbol{x}}_k^{-} + \underline{\underline{K}}_k (\boldsymbol{z}_k - h(\hat{\boldsymbol{x}}_k^{-}) - \underline{\underline{H}}_k^i (\hat{\boldsymbol{x}}_k^{-} - \hat{\boldsymbol{x}}_{k,i}^{+}) \tag{20}$$

In Eq. (18)-(20), i is the iteration number. Once one of the convergence criteria is met, the procedure is ended, and the results of the last iteration are regarded as *a-posteriori* estimates of the mean and error covariance.

Both the EKF and the IEKF may introduce large errors in the state estimation due to the first-order linearization approximation.

Unscented Kalman filter

To deal with highly nonlinear systems, a variant of the KF scheme, namely the Unscented Kalman Filter has been introduced, [38]. It exploits a deterministic approach to sample the state space, and it allows capturing the posterior mean and covariance accurately to the 3rd order. The UKF scheme is based on the concept of the unscented transform, which allows avoiding the derivation of the Jacobian matrices required in the EKF formulation. The key aspect of the UKF scheme is the selection of the so-called sigma-points (χ), i.e., a set of state vectors chosen so that their sample mean and covariance are equal to the mean and covariance of the current state. The number of sigma-points is related to the dimension of the state vector. If L is the number of parameters chosen to represent the state of the system, $2L+1$ sigma-points must be defined according to the following relations:

$$\boldsymbol{\chi}_{k-1}^{1} = \widehat{\boldsymbol{x}}_{k-1}^{+} \tag{21}$$

$$\boldsymbol{\chi}_{k-1}^{i} = \widehat{\boldsymbol{x}}_{k-1}^{+} + \sqrt{L+\lambda}\boldsymbol{W}_i, \qquad i = 2, \dots . L+1 \tag{22}$$

$$\boldsymbol{\chi}_{k-1}^{i} = \widehat{\boldsymbol{x}}_{k-1}^{+} - \sqrt{L+\lambda}\boldsymbol{W}_i, \qquad i = L+2, \dots .2L+1 \tag{23}$$

In Eq. (22) $\boldsymbol{W}_i$ is the i^th^ column vector of the Cholesky decomposition of $\underline{\underline{P}}_{k-1}^{+}$, the *a-posteriori* estimate of the covariance matrix, and λ is computed as follows:

$$\lambda = \alpha^2(L+c) - L \tag{24}$$

where L is the state vector dimension, a and c are two user-define parameters.

Specifically, the parameter a (selectable from 0 to 1) is related to the spread of the sigma-points around the mean of the statistical contribution, while c is typically set to 0. In the UKF scheme, the time-update equation is exploited to project ahead in time the sigma-points, and an *a-priori* estimate of the state and error covariance is obtained by properly weighting the predicted sigma-points with the assumption that they sample a Gaussian distribution as follows,

$$\widehat{\boldsymbol{x}}_{\boldsymbol{k}}^{-} = \sum_{i=1}^{2L} w_m^i f(\boldsymbol{\chi}_{\boldsymbol{k-1}}^{\boldsymbol{i}}) \tag{25}$$

$$\underline{\underline{P}}_{xx,k} = \sum_{i=1}^{2L} w_c^i \left(\boldsymbol{\chi}_k^i - \hat{\boldsymbol{x}}_k^-\right)\left(\boldsymbol{\chi}_k^i - \hat{\boldsymbol{x}}_k^-\right)^T + \underline{\underline{Q}} \tag{26}$$

where the weight w_m^i and $w_{\boldsymbol{c}}^{\boldsymbol{i}}$ are defined in Eq. (27)

$$w_m^1 = \frac{\lambda}{L+\lambda}$$

$$w_c^1 = \frac{\lambda}{L+\lambda} + (3 - \alpha^2) \tag{27}$$

$$w_{\boldsymbol{m}}^{\boldsymbol{i}} = w_{\boldsymbol{c}}^{\boldsymbol{i}} = \frac{1}{2(L+\lambda)}, \qquad i = 2,..,2L+1$$

Measurements and measurement covariance predictions are also obtained by projecting the sigma-points into the measurement space as in Eq. (28)-(30):

$$\boldsymbol{\gamma}_k^i = h(\boldsymbol{\chi}_k^i) \tag{28}$$

$$\hat{\boldsymbol{z}}_k = \sum_{i=1}^{2L} w_m^i \boldsymbol{\gamma}_i \tag{29}$$

$$\underline{\underline{P}}_{zz} = \sum_{i=1}^{2L} w_c^i \left(\boldsymbol{\gamma}_k^i - \hat{\boldsymbol{z}}_k\right)\left(\boldsymbol{\gamma}_k^i - \hat{\boldsymbol{z}}_k\right)^T + \underline{\underline{R}} \tag{30}$$

In the UKF scheme, a cross covariance matrix is introduced, computed as in Eq. (31), to determine the Kalman gain (Eq. (32)):

$$\underline{\underline{P}}_{xz} = \sum_{i=1}^{2L} w_c^i \left(\boldsymbol{\chi}_k^i - \hat{\boldsymbol{x}}_k\right)\left(\boldsymbol{\gamma}_k^i - \hat{\boldsymbol{z}}_k\right)^T \tag{31}$$

$$\underline{\underline{K}}_k = \underline{\underline{P}}_{xz}\underline{\underline{P}}_{zz}^{-1} \tag{32}$$

Once the Kalman gain has been computed, the *a-posteriori* estimate of the state and error covariance are given by:

$$\hat{\boldsymbol{x}}_k^+ = \hat{\boldsymbol{x}}_k^- + \underline{\underline{K}}_k\left(\boldsymbol{z}_k - h(\hat{\boldsymbol{x}}_k^-)\right) \tag{33}$$

$$\underline{\underline{P}}_k^+ = \underline{\underline{P}}_{k-1}^+ - \underline{\underline{K}}_k \underline{\underline{P}}_{zz} \underline{\underline{K}}_k^T \tag{34}$$

Anyway, all the KF variants rely on some assumptions on both the process and measurement noise.

Minimum energy filter

The adoption of a filtering scheme which relies on the minimum energy formulation allows removing some hypothesis on both the process and measurement noise. According to this scheme, the error signals are assumed square integrable. The objective of the minimum energy filters id to find, at each time *t*, the estimate of the state, consistently with the system model, which minimized a cost function on the error signals, given the actual measurement $\left\{\boldsymbol{y}_i|_{[0,t]}\right\}$, [39]. The error function can be written as follows:

$$J_t\left(\boldsymbol{x}(0), \boldsymbol{w}_{[0,t]}, \boldsymbol{v}_{[0,t]}\right) = \frac{1}{2}|\boldsymbol{x}(0) - \hat{\boldsymbol{x}}_0|^2_{\underline{\underline{k_0}}} + \int_0^t \frac{1}{2}|\boldsymbol{w}(\tau)|^2_{\underline{\underline{Q}}} + \frac{1}{2}|\boldsymbol{v}(\tau)|^2_{\underline{\underline{R}}} d\tau \tag{35}$$

with $\underline{\underline{Q}}$, $\underline{\underline{R}}$ and $\underline{\underline{k_0}}$ weighting matrices and $|\boldsymbol{b}|^2_{\underline{\underline{A}}} = \boldsymbol{b}^T \underline{\underline{A}} \boldsymbol{b}$.

The argument of the integral defines the cost on the energy of the two error signals, from which the expression "minimum energy" derives. Also, the subscript [0,t] indicates that the optimization takes place on that interval.

The final optimal state at time *t*, which represents the state estimate of the system, is the one which minimizes the following value function.

$$V(\boldsymbol{x}, t) = \min_{\boldsymbol{w}[0,t]} J_t(\boldsymbol{x}, w[0,t]) \tag{36}$$

$$\hat{\boldsymbol{x}}(t) = \arg\min_{\boldsymbol{x}} V(\boldsymbol{x}, t) \tag{37}$$

The filtering scheme is given by Eq. (38) and (39), derived by applying the Hamilton-Jacobi-Bellman (HJB) formulation,

$$\dot{\hat{\boldsymbol{x}}} = \underline{\underline{F}}\hat{\boldsymbol{x}} + \underline{\underline{B}}\boldsymbol{u} + \Sigma \underline{\underline{H}}^T \underline{\underline{R}} \left(\boldsymbol{y} - \underline{\underline{H}}\hat{\boldsymbol{x}}\right) \tag{38}$$

$$\dot{\underline{\underline{\Sigma}}} = \underline{\underline{Q}}^{-1} + \underline{\underline{\Sigma}}\,\underline{\underline{F}}^{T} + \underline{\underline{F}}\,\underline{\underline{\Sigma}} - \underline{\underline{\Sigma}\,\underline{\underline{H}}^{T}}\,\underline{\underline{R}}\,\underline{\underline{H}\,\underline{\underline{\Sigma}}} \tag{39}$$

with

- $\underline{\underline{\Sigma}}$ indicating the inverse of the Hessian of the value function,
- $\underline{\underline{F}}$ indicating the system model
- $\underline{\underline{B}}$ indicating the input matrix
- $\underline{\underline{H}}$ indicating the measurement model.

<u>H∞ filter</u>

On the other hand, the H∞ filter minimizes the ∞-norm of the estimation error rather than the mean-squared error and it does not make any assumption on the statistics of the process and measurement noise, [40]. However, as for the KF, both the system and measurement model must be governed by linear equations. Also, the filtering scheme is very similar to the KF: the main difference consists in the filter gain $\underline{\underline{K}}_k$, which is selected so that the following the relation in Eq. (40) is satisfied:

$$||\boldsymbol{T}_{ew}||_{\infty} < 1/\theta \tag{40}$$

where $||\boldsymbol{T}_{ew}||_{\infty}$ represents the difference between the predicted and real state, and θ is a tuning parameter. Eq. (40) leads to the following expression of the gain:

$$\underline{\underline{K}}_k = \underline{\underline{P}}_{k-1}^{+}\left[\underline{\underline{I}} - \theta\underline{\underline{P}}_{k-1}^{+} + \underline{\underline{H}}_k^{T}\underline{\underline{R}}_k^{-1}\underline{\underline{H}}_k\underline{\underline{P}}_{k-1}^{+}\right]^{-1}\underline{\underline{H}}_k^{T}\underline{\underline{R}}_k^{-1} \tag{41}$$

The correction equations are modified with respect to the ones of the classical KF as shown in Eq. (42) and (43).

$$\hat{\boldsymbol{x}}_k^{+} = \underline{\underline{F}}_k\hat{\boldsymbol{x}}_{k-1}^{+} + \underline{\underline{F}}_k\underline{\underline{K}}_k(\boldsymbol{z}_k - \underline{\underline{H}}_k\hat{\boldsymbol{x}}_k^{-}) \tag{42}$$

$$\underline{\underline{P}}_k^{+} = \underline{\underline{F}}_k\,\underline{\underline{P}}_{k-1}^{+}\left[\underline{\underline{I}} - \theta\underline{\underline{P}}_{k-1}^{+} + \underline{\underline{H}}_k^{T}\underline{\underline{R}}_k^{-1}\underline{\underline{H}}_k\underline{\underline{P}}_{k-1}^{+}\right]^{-1}\underline{\underline{F}}_k^{T} + \underline{\underline{Q}}_k \tag{43}$$

However, its performance results more sensitive to the tuning parameters that the one of the Kalman filters.

Focusing on space applications, Kalman filters, with all their variants, have a solid space heritage: they have been used for navigation purposes in the Apollo space program [41], and since then their use for aerospace applications has continuously grown ranging from Earth orbiting observation to space exploration missions, rendezvous, proximity operations and applications requiring precision formation flight. Many GNC architectures enabling proximity operations relies on Kalman filtering scheme to integrate within the relative navigation module the measurements provided by EO sensors whether according to loosely or tightly coupled configurations. For instance, some examples of tightly coupled schemes can be found in [11], [33], [42]. Specifically, the architectures presented in [11] and [42] relies on features' coordinates extracted from stereo images fed to an EKF and UKF, respectively, while the relative state between two spacecraft is estimated within a UKF exploiting 2D features extracted from monocular images and range measurements provided by a distance sensor (e.g., laser range finder) in [33]. In the framework of loosely coupled architectures, a dual inertial multiplicative EKF can be found in [29] in which the pose is measured by processing LIDAR data with the Oriented, Unique and Repeatable Clustered Viewpoint Feature Histograms (OUR-CVFH) algorithm. Also, the concept of dual quaternion, which as the twofold advantage of providing a concise and compact representations of the state parameters and taking the coupled rotational-translational motion effect into account, is exploited within an adaptive fading factor EKF to filter vision-based pose estimates, [43]. A detailed summary of the advantages and disadvantages related to the different kinds of filtering scheme is presented in Table 1.

Table 1 - Filtering schemes: advantages and disadvantages.

<table>
<tr><th>Filtering Scheme</th><th>Advantages</th><th colspan="2">Disadvantages</th></tr>
<tr><td>KF</td><td>• Very simple formulation
• Optimal solution (if all the hypotheses are satisfied)</td><td>• Only applicable to the case of linear models</td><td rowspan="3">Difficult to tune</td></tr>
<tr><td>EKF/IEKF</td><td>• Capability to handle Non-linear models</td><td>• Costly Jacobian matrices computation
• Based on local approximations</td></tr>
<tr><td>UKF</td><td>• Capability to capture the first three moments of an arbitrary PDF
• Jacobian-free</td><td>• Issues with nearly singular covariances
• Computationally expensive (Cholesky Factorization every time step)</td></tr>
<tr><td>Minimum Energy Filter</td><td>• No assumption on the PDFs of noise models</td><td colspan="2">• Complex mathematical formulation</td></tr>
<tr><td>H-∞ Filter</td><td>• No assumption on the PDFs of noise models</td><td colspan="2">• Only linear models</td></tr>
</table>

2.2.2 Process model for spacecraft dynamics

Regardless of the selected filtering scheme, a model describing the temporal evolution of the state variable must be identified. Generally, these models rely on the hypothesis that the translational and rotational dynamics are decoupled. While the standard model for the rotational dynamics is represented by the Euler equation coupled with the kinematics of the corresponding parameters representing the attitude (i.e., quaternions), different option for the relative translational model can be found in the open literature. In this scenario, the relative translational motion between two spacecraft can be modelled by a set of non-linear differential equations derived from the combination of the two-body problem written for both spacecraft under the assumption of Keplerian mechanics [44], as shown below

$$
\begin{aligned}
&\ddot{x} - \frac{\mu}{r_c^2} + \frac{\mu(r_T + x)}{[(r_T + x)^2 + y^2 + z^2]^{\frac{3}{2}}} - 2\dot{y}\dot{\theta} - y\ddot{\theta} - x\dot{\theta}^2 = f_{T,x} \\
&\ddot{y} + \frac{\mu y}{[(r_T + x)^2 + y^2 + z^2]^{\frac{3}{2}}} + 2\dot{x}\dot{\theta} - x\ddot{\theta} - y\dot{\theta}^2 = f_{T,y} \\
&\ddot{z} + \frac{\mu z}{[(r_T + x)^2 + y^2 + z^2]^{\frac{3}{2}}} = f_{T,z}
\end{aligned}
\tag{44}
$$

where x, y and z are the Cartesian components of the relative position vector expressed in a reference frame centered in the center of mass of the target in which the x-axis points outwards the Earth center, the z-axis is parallel to the angular momentum vector and the y-axis completes the right-handed triad, f_T is the specific thrust force acting on the chaser spacecraft, μ is the Earth gravitational constant, r_T is the norm of the position vector of the target position vector, $\dot{\theta}$ and $\ddot{\theta}$ are the orbital and angular velocity accelerations of the target spacecraft which can be expressed as in Eq. (45) and (46) with a_T, e_T and θ being the semi-major axis, eccentricity and true anomaly of the target.

$$
\dot{\theta} = \sqrt{\frac{\mu}{a_T^3(1 - e_T^2)^3}}(1 + e_T \cos\theta)^2 \tag{45}
$$

$$
\ddot{\theta} = -\frac{2\dot{r}_T\dot{\theta}}{r_T} \tag{46}
$$

The model reported in Eq. (44) can be linearized by applying the assumption that the two spacecraft fly in close proximity, obtaining the Yamanaka-Ankersen relative motion model [45], as shown in the following

$$
\ddot{x} - 2k\theta^{\frac{3}{2}}x - 2\theta\dot{y} - \dot{\theta}y - \theta^2 x = a_{T,x} \tag{47}
$$

$$\ddot{y} + k\theta^{\frac{3}{2}}y + 2\theta\dot{x} + \dot{\theta}x - \theta^2 y = a_{T,y}$$

$$\ddot{z} + 2k\theta^{\frac{3}{2}}z = a_{T,z}$$

where k is equal to $\mu/h^{3/2}$, being h the norm of the angular momentum vector of the target.

The model of Eq. (47) can be further simplified into the Clohessy-Wiltshire (CW) equations, shown in Eq. (48), where n represents the mean motion of the target spacecraft, by applying the further assumption that the target moves on a circular orbit, [46].

$$\begin{aligned} \ddot{x} - 2n\dot{y} - 3n^2 x &= f_{T,x} \\ \ddot{y} + 2n\dot{x} &= f_{T,y} \\ \ddot{z} + n^2 z &= f_{T,z} \end{aligned} \tag{48}$$

The hypothesis used to derive the relative motion models are summarized in Table 2. Despite the simplistic hypothesis on which the CW model relies, due to the time scales of the problem (the prediction is update with a frequency in the order of few Hz in close-proximity operations), it is the most used model to describe the relative translational motion between two spacecraft.

Table 2 - Relative translational motion models: hypothesis.

Relative translational motion model	**Hypothesis**		
	Two-body mechanics	*Close Proximity*	*Target on a circular orbit*
Gurfil et al. [44]	✓		
Yamanaka-Ankersen [45]	✓	✓	
CW-equations [46]	✓	✓	✓

Regarding the target-chaser rotational motion, a kinematics and dynamics model must be introduced. The attitude kinematics can be modelled using the quaternion parametrization of the attitude. Indeed, with respect to the Euler angles representation, the unit quaternion is not affected by singularity problem. Thus, the attitude kinematics is modelled through Eq. (49), where $\boldsymbol{q}$ is the unit quaternion representing the orientation of the target body frame with respect to a chaser body frame and $\boldsymbol{\omega}_{rel}$ is the angular velocity of the target with respect to the chaser and $\otimes$ represents the quaternion product.

$$\dot{\boldsymbol{q}} = \frac{1}{2}[0\ \boldsymbol{\omega}_{rel}] \otimes \boldsymbol{q} \tag{49}$$

The dynamic model is instead represented by the well-known Euler equation, under the free-body assumption,

$$\dot{\boldsymbol{\omega}}_{TRF/IRF} = -\underline{\underline{I}}_T^{-1}(\boldsymbol{\omega}_{TRF/IRF} \times \underline{\underline{I}}_T \boldsymbol{\omega}_{TRF/IRF}) \tag{50}$$

where $\boldsymbol{\omega}_{TRF/IRF}$ is the inertial angular velocity of the target and $\underline{\underline{I}}_T$ its inertia matrix.

2.3 Relative navigation architecture for state and target's inertia estimation

Since target inertia properties might have changed due to the long time spent in orbit, and since their knowledge is required to both allow the estimate of the relative rotational motion during the approach phase and to control the stack after the capture, the development of reliable and robust techniques to accurately reconstruct the inertia properties before performing the final approach and berthing or docking operations is a critical task. In this respect, the pose data retrieved by EO sensors can be exploited to estimate the inertia properties of the target. However, it is worth to highlight that without perturbing the rotational dynamics of the target with an external torque and observing the resulting change in its rotation [47]–[49], the inertia matrix can be estimated up to a scale factor. This torque can be generated by either physically impacting the target, [47], [49] or by getting close to it with a permanent magnet able to produce eddy currents [48]. Clearly, these procedures imply that the chaser approaches closely to the target to perturb it, thus risking unwanted collisions and possibly, fragmentations. Hence, safer, contactless approaches shall be preferred: estimate the moment of inertia ratios (MIRs) during the monitoring phase, when the two spacecraft are in a safe relative configuration, and then determine the scale factor when they are rigidly attached. In this regard, Setterfield et al. propose a method in which the principal axes and the inertia ratios are estimated by comparing an analytically predicted polhode (i.e., the curve traced by the angular velocity vector in the body reference frame of the target) with the one reconstructed by means of attitude and angular velocity data retrieved from visual sensor (in uncooperative case) or gyro data (in cooperative case), using a constrained optimization approach, [50]. The pipeline of this approach is depicted in Figure 1, where .

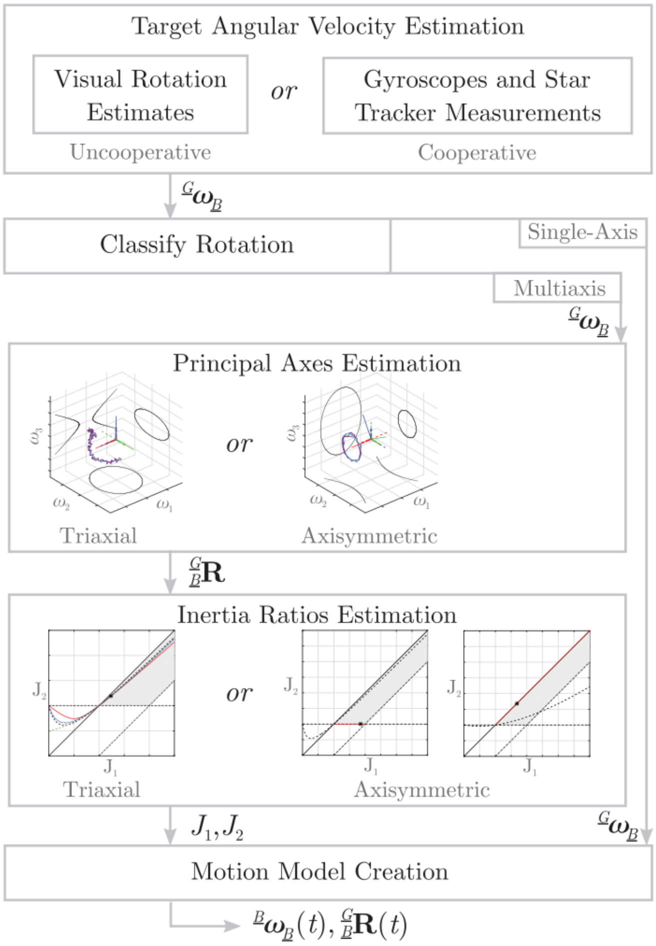


Figure 1 - Pipeline for the estimation of the principal axes and inertia ratios, [50]**.**

Once the target angular velocity has been estimated, the rotational motion is classified as single-axis or multi-axis rotation in order to determine if the inertial properties are observable: indeed, in case of single-axis rotation, the polhode curve collapses into a point which has no dependence on the direction of the principal axis or the MIRs. In case of multi-axis rotation, the direction of the principal axes with respect to the generic target fixed body frame is estimated by finding the rotation that correspond to ellipses and/or hyperbola that best fit the planar projection of the polhode curve. Then, the MIRs that fit the noisy polhode data are determined as the ones that minimize the Mahalanobis difference between the predicted and measured angular velocities. This method was tested aboard the International Space Station using visual and gyroscope data of the Synchronized Position Hold Engage and Reorient Experimental Satellites

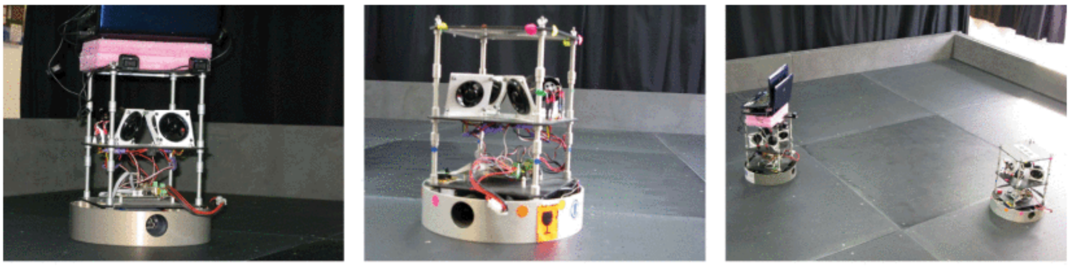
(SPHERES) and the Visual Estimation for Relative Tracking and Object Inspection (VERTIGO) and visual data from numerical simulation.

On the other hand, kinematic equations of motion and the principle of the conservation of the angular momentum vector in the inertial frame can be exploited for MIRs estimation purposes. For instance, Sheinfeld et al. describe a strategy for the estimation of both center of mass location and MIRs of a tumbling space target which exploits an unconstrained least square approach. Moreover, it identifies the condition to be met which guarantees a physical solution: the angular velocity measurements involved in the MIRs determination procedure must be linearly independent, [51]. This approach has been extended in [52] to take the inequality constraints on the inertia parameters into account and to ensure positive diagonal entries in the estimated MIRs matrix by applying the active set method for convex quadratic programming. Also, the authors provide a discussion on the observability of the inertial properties. It is underlined that they can be fully estimated only when the nutation angle, i.e., the angle between the angular momentum vector and the angular velocity vector, is nonzero; whilst in case of pure rotation (i.e., when the target rotates around one of its principal axes) the MIRs matrix is not fully observable.

Thus, the determination of the inertia properties of a non-cooperative space target cannot prescind from its motion parameters estimation.

However, most of the state-of-the-art GNC architecture enabling close proximity operations which address the target's inertia and motion state determination task relies on tightly coupled architectures and stereovision system as main relative navigation sensor. For instance, [28] presents a relative navigation architecture which enables close proximity operations with non-cooperative space object coping with uncertainties on the inertia properties of the target. The main filtering scheme is an

IEFK aiming at estimating the position of the tracked feature along with the relative motion parameters. The navigation filter is coupled with an inertia estimation algorithm based on the evaluation of a likelihood score over a set of probable inertia tensors. The resulting approach consists into different IEKF assuming different probable inertia tensor. Each filter determines the state vector assuming a hypothetical target inertia tensor. The most probable inertia tensor is then determined as the one corresponding to the IEFK with the lowest innovation variance using a Maximum A Posterior (MAP) estimator. The performance assessment is carried out within a numerical simulation environment in which a set of features on the target surface is assumed to be perfectly tracked from a sequence of stereo-images and within an experimental set-up with a 3-DOF (2 translational and 1 rotational) shown in Figure 2. The experimental results show a 20-cm and 5 cm error (1σ) in the range and features' positions, respectively and a steady-state estimation error of 0.5° for the relative attitude. Finally, the only observable MIR is estimated with a 10% accuracy.



(a) (b) (c)

Figure 2 - Experimental 3-DOF set-up to test stereo-based motion and inertia parameters estimation algorithm, [28]**.**

A similar approach, in which the IEKF is substituted with an Interaction Cubature KF (InCKF) is proposed in [53]. Indeed, the CKF guarantees improved performance with respect to conventional nonlinear filters when dealing with high dimensional nonlinear systems. However, the presence of several estimation filters working simultaneously requires high computational power. Numerical simulations which simulate the measurement error on the tracked features with Gaussian noise distribution show more accurate results than the one achieved with the IEKF.

A different parametrization of the inertia properties corresponding to the logarithmic algorithm of the principal MIRs is introduced in [54]: this allows to solve the estimation problem without considering additional constraints, having the logarithm the same validity domain of the inertia ratios, i.e.,]0, +∞[. This work relies on the development of a probabilistic factor graph process model based on both rigid-body kinematics and dynamics, i.e., Newton's second law and Euler's equation. Figure 3 shows a generic pose graph model in which the nodes of the graph, represented as circle, contains the variable to be estimated (i.e., the state $\boldsymbol{x}$ and the position of the features $\boldsymbol{l}$) while the rectangles represent the factors modeling the joint probability distribution between the nodes which represents the error between the variables in the nodes that must be minimized. The pose-graph optimization problem is then solved with the Incremental Smoothing and Mapping (iSAM) which, with respect to classical approach for Simultaneous Localization and Mapping (SLAM), allows updating only those entries of the information matrix that change within a time step.

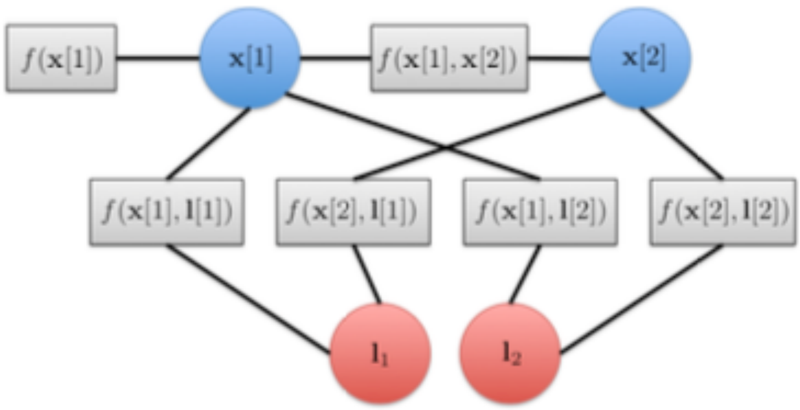


Figure 3 - Generic Pose Graph Model, [54].

This method has been validated on-board the International Space Station using SPHERES and the VERTIGO test platform.

The same logarithmic parametrization of the MIRs by Tweddle has been adopted in [11] for the simultaneous estimation of the state and inertia parameters of an uncooperative target within a filtering procedure, which is more efficient than a smoothing algorithm from a computational point of view, in which the MIRs are addressed as time-independent variables. Specifically, the authors compare the performance of a EKF and an IEKF in which the MIRs are addressed as time-independent variable and a pseudo-measurement constraint is added to force the inertia rations to converge to the correct value: this pseudo measurement is the classical Euler equation for the rotational dynamics of free-rigid bodies. Thus, a null output is considered in the measurement vector, whilst the Euler equation is added to the observation model. The main drawback of this approach comes from the need of angular acceleration measurements in pseudo-measurement equation which can be only retrieved by numerical differentiating the angular velocity measurements. The performance assessment is carried out within a numerical simulation environment which simulates the performance of a stereo-vision system by adding a Gaussian error to the reference position of the tracked features.

An adaptive UKF and the classic MIRs inertia parametrization is instead adopted in [42]. The authors integrate a classical UKF scheme for state and inertia parameters estimation with an adaptive algorithm which aims at modifying the covariance of the MIRs according to the measurement updating errors of the relative state obtaining a two-stage state and MIRs estimation procedure. First, the covariance of the MIRs is set large to ensure a sufficient degree of variability of the relative state parameters. Then, when the measurement updating error of the relative state reaches a certain threshold, the covariance of the MIRs is decreased so to ensure the convergence of both relative state and MIRs estimation errors. The validity of the proposed method has been verified within a numerical simulation environment simulating the noise of the tracked feature position with a Gaussian distribution. Stereo-vision measurements are exploited also in [55] in which the authors propose a method that allows determining the motion of a freely tumbling rigid body, including the MIRs, using a set of tracked feature by solving an optimization problem in which the objective function is the difference between the observed and predicted motion, thus without relying on a filtering scheme.

However, the use of stereovision systems can be effective only at very short distance (e.g., up to ten meters) due to the limits posed on the achievable baseline by installation constraints. Alternative technological solutions to stereo-vision systems have also been considered. In [56], the relative motion and the target inertia parameters are estimated in a two-stage algorithm which exploits the measurement provided by a monocular camera. In the first stage the initial condition for the motion propagation and the target's tensor of inertia are estimated by minimizing the difference between the points' pixel coordinate of a set of features tracked by a passive monocular camera and the corresponding ones on the target model through a continuous time interval, in a least square sense. In the second stage, the pose is tracked in real-time by minimizing the reprojection error of a set of features extracted in the monocular image. The algorithm has been tested using the frames acquired during the separation of the satellite "Chibis-M" from the cargo vehicle "Progress-13M". The results underline the major drawbacks related to the use of passive systems which is the presence of the Sun that can cause overexposure effect. On the contrary, active vision system guarantees robustness against the harsh lighting condition that a spacecraft may encounter on orbit.

Active systems are often integrated within loosely coupled architectures. In this regards, Aghili et Parsa propose an EKF which estimates the full relative motion state and the inertia parameters of the target satellite as well as the covariance of the measurement noise by exploiting the pose measured with data provided by a Laser Camera System (LCS), [57]. Experimental results demonstrate the possibility to implement the presented scheme in real-time and also its robustness against the occlusion of the sensor for a limited time interval (e.g., 10 seconds). The use of active systems is also foreseen in [58] where range images acquired by a set of cooperating 3D sensors are exploited for the estimate of the motion state, shape and inertia parameters of uncooperative and unknown target. The proposed architecture, shown in Figure 4, estimate the pose parameters (i.e., relative position and attitude) by performing the kinematic data fusion of the range images which are fed to a KF to determine the full motion state and target's inertia parameters. At the same time, the range data are exploited to reconstruct the shape of the target using a stochastic map-building approach. However, the numerical simulations carried out for performance assessment relies on the hypothesis that the range sensors are perfectly synchronized with respect to each other.

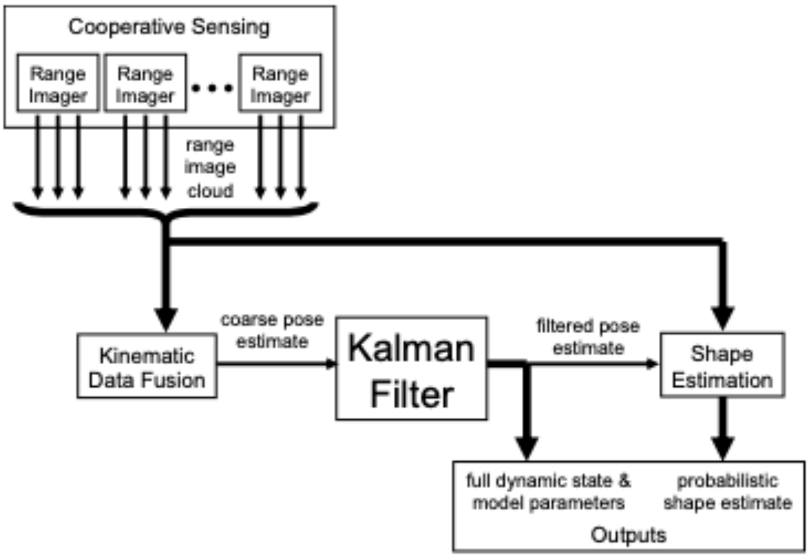


Figure 4 - Pipeline for the estimation of state, shape and inertia of uncooperative space target using range images, [58]**.**

The possibility to deal with an unknown target is addressed also in [59] where the authors use the Iterative Closest Point (ICP) algorithm to track the pose of an uncooperative target by registering consecutive point clouds measured with a LIDAR system and estimate the target MIRs within an EKF settings. The performance of the proposed scheme are assessed within an experimental set-up which simulates both the rotational and translational motion of the chaser, while the target simulator is fixed on a 3-axis platform. Despite the high accuracy achieved in the estimate of the relative motion parameters (cm-level and degree error level for the relative position and attitude), the inertia parameters converge to constant biases due to the weak observability of the system caused by the low angular velocities in play. More in general, some limitations arise from the use of KF for inertia properties estimation due to the strong influence on the achievable results of the filter tuning parameters and to the strong sensitivity to non-Gaussian noise which usually characterize the vision-based measurements.

To conclude, Table 3 provides a summary of the reviewed architectures for motion and inertia parameters estimation of non-cooperative space object.

Table 3 - Survey of approaches for motion and inertia parameters determination of uncooperative space target.

Reference	Relative navigation sensor	Algorithmic solutions	Performance Assessment criterion
[28]	Stereo-vision system	IEKF + MAP estimator	Ground-based experimental test
[53]	Stereo-vision system	InCKF + MAP estimator	Numerical simulations
[54]	Stereo-vision system	Factor graph model-based smoothing algorithm	Experimental test in micro-gravity condition
[11]	Stereo-vision system	EKF/IEKF	Numerical simulations
[42]	Stereo-vision system	AUKF	Numerical simulations
[55]	Stereo-vision system	LSO	Numerical simulations
[56]	Monocular camera	LSO	Numerical simulations using real images
[57]	Active sensor	EKF	Ground-based experimental test
[58]	Multiple active sensors	EKF	Numerical simulations
[59]	LIDAR	EKF	Numerical simulations

2.4 GNC architecture for close approach phase

As stated in Chapter 1, the knowledge of the inertia properties of the target is vital to plan the final phase of a rendezvous mission (namely, reach and capture phase) and it must be carried out during the monitoring phase. If the target is tumbling, or does not have a proper docking interface, the best way to capture it is by means of a robotic arm. It stays in a stowed configuration until the start of the reach and capture phase, when it is deployed so to reach a pre-defined grasping point on the target surface. In this phase, it is mandatory to have an accurate knowledge of the position of the end effector placed at the end of the robotic arm with respect to the target. In this case, the relative navigation subsystem may include (if necessary) an active or passive EO sensor on the tip of the robotic arm end effector. In the following, the main architectural solutions proposed in the recent literature are recalled, highlighting their main advantages and drawbacks.

In the framework of the COMRADE project, two different options for the GNC architecture were proposed, [60]. The architectures were developed to be suitable to both ADR and OOS scenarios in which the chaser spacecraft is equipped with a robotic arm. According to the first option (*Option 1* in the following), the navigation function is entrusted to two different EO sensors, one mounted on the body of the chaser spacecraft, and the other mounted near the end-effector of the robotic arm. The measurements retrieved by the raw data, i.e., the target/chaser and end-effector/grasping point poses, are then processed by two different navigation filter, according to the scheme in Figure 5. The "*Navigation Filter Chaser*" fully characterizes the relative motion between the two spacecraft by exploiting the measured pose and chaser absolute navigation state, while the "*Navigation Filter*

Visual Servoing" provides refined estimate of the gripper/grasping point pose together with relative velocity and angular velocity information by exploiting the pose measured by means of the sensor fixed to the robotic arm, the kinematic model of the manipulator and also the output of the "*Navigation Filter Chaser*"

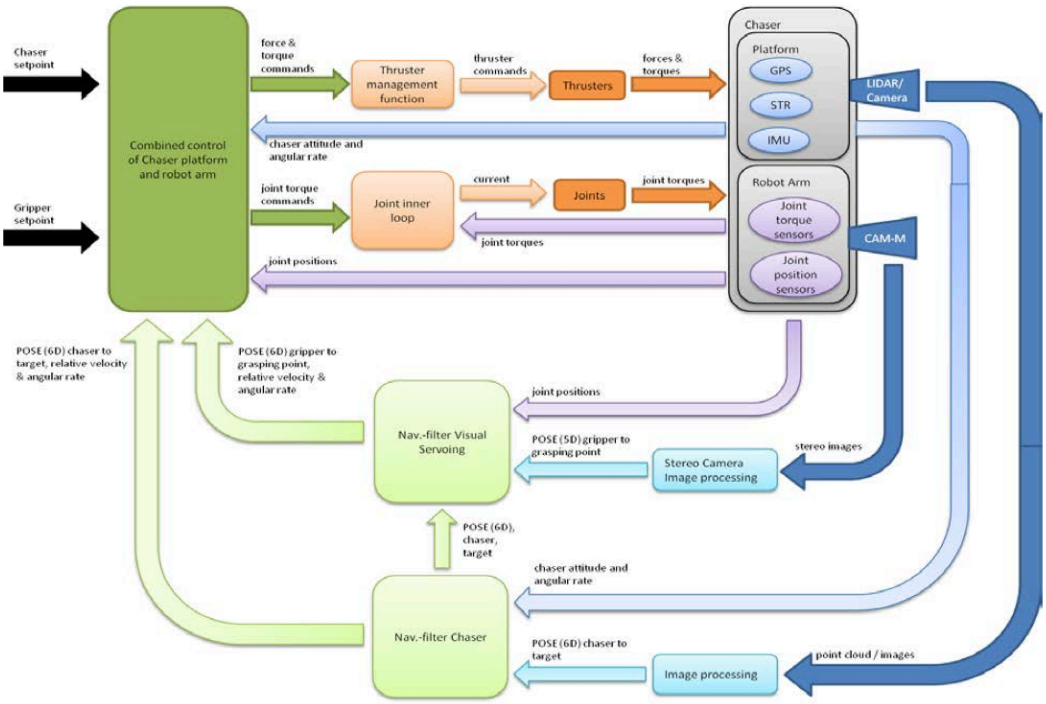


Figure 5 - GNC architecture COMRADE project, Option 1, [60]**.**

The second option, *Option 2* in the following, investigated in the framework of the COMRADE project is depicted in Figure 6. The relative navigation function is entrusted to an EO sensor mounted on the body of the chaser which provides the target/chaser pose measurement, while the gripper/grasping point pose information is retrieved by combining the joint positions, the robotic arm kinematic model and the target/chaser measured pose.

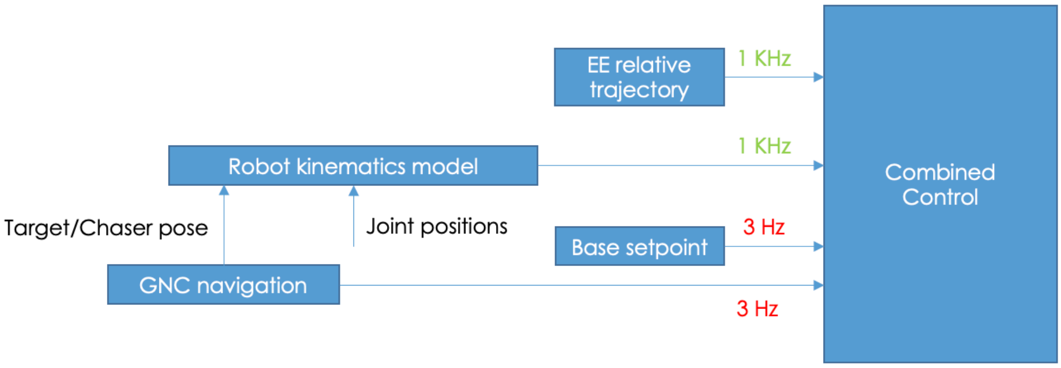


Figure 6 GNC architecture COMRADE project, Option 2, [60]**.**

A different architecture (*Option 3* in the following) relying on two EO sensors, one body mounted and one fixed to the robotic arm is presented in [61]. In this case the two EO sensors share the recognition task in a collaborative behavior, i.e., they trace different feature points on the target surface to measure the pose parameters according to the scheme provided in Figure 7. As in the *Option 2* proposed in the framework of the COMRADE project, the pose of the end effector with respect to the grasping point can be computed by combining the joint positions, the robotic arm kinematic model and the target/chaser measured pose.

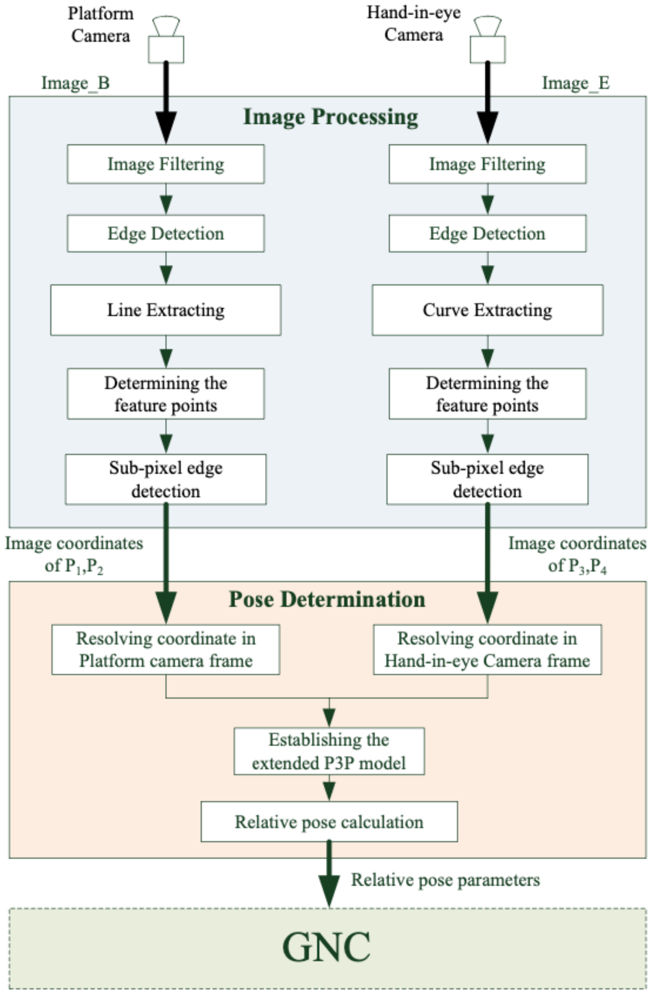


Figure 7 - Flow-chart for the pose estimation with collaborative cameras, [61]**.**

Some trade-off considerations about the described architectures are reported below.

- The main advantage of *Option 1* with respect to *Option 2* is the possibility to compensate the ego motion uncertainty of the robotic arm introduced by the kinematic model, i.e., by the uncertainty in the knowledge of the joint's rotation angles. Clearly, the update rate at which this compensation is carried out is limited by the measurement rate of the camera on the robotic arm.
- In scenarios in which the residual relative motion between the target and the chaser during the reach and capture phase is particularly slow, as well as when the target is collaborative (i.e., its attitude can be actively controlled to ease the

capture operations) or semi-collaborative (i.e., it keeps a stable attitude), *Option 2* can be also viable. In this respect, the main disadvantage of *Option 1* with respect to *Option 2* is given by the additional computational burden required to process the data acquired by the sensor attached to the end effector.

- *Option 3* can be applied to scenarios in which the body-fixed camera and the robotic arm are mounted on the chaser so that only the hand-in-eye camera looks at the target face hosting the selected grasping point. One major drawback related to this approach lies in the fact that the ego motion uncertainty of the robotic arm will affect not only the robotic arm/grasping point, but also the target/chaser pose estimation process. This occurs since the robotic arm kinematic model must be used to get the LOS of the feature points detected by the hand-in-eye camera with respect to the body-fixed camera reference frame. Moreover, this approach is only applicable if an initial guess for the length of the position vector of the above-mentioned feature with respect to the hand-in-eye camera is available.

3. Autonomous GNC architecture for close-proximity operations

This section describes the details of the developed GNC architecture for both the monitoring and reach and capture phase. The relative navigation architecture for the reach and capture phase has been developed in the framework of the GRACC (preparation of enabling space technologies and building blocks: Gnc and Robotic Arm Combined Control) study, conducted under ESA contract by a consortium of Italian universities composed by teams from Università degli Studi di Padova, Politecnico di Milano and Università degli Studi di Napoli "Federico II". Specifically, for the monitoring phase, an original LIDAR-based relative navigation architecture which relies on a multi-step strategy to estimate both the target-chaser relative motion parameters and the MIRs of the target. The focus of LIDAR systems is motivated by their capability to provide direct 3D data about the observed scene, unlike monocular cameras, with much larger operative range than stereovision systems. On the other hand, for the reach and capture phase, the relative navigation function relies on two sensors and two separate filtering schemes, inspired by *Option 1* of the COMRADE project. This approach has advantages in terms of flexibility and modularity with respect to designing a single filter since it allows to continuously update the state of the robotic arm with respect to the grasping point independently of the availability of measurements from the robotic arm sensor (e.g., when the robotic arm is being deployed from its stowed configuration and the target is not enclosed in the robotic arm sensors' FOV).

Before introducing the developed architectures for the monitoring and reach and capture phase, the mathematical notation as well a list of the reference frame adopted within this thesis is provided in the following sub-chapter.

3.1 Mathematical Preliminaries

The following mathematical notation are adopted.

- Plain italic letters (a) are used to indicate scalar quantities.
- Bold italic letters ($\boldsymbol{a}$) are used to indicate vector.
- Double underlined capital italic letters $\left(\underline{\underline{A}}\right)$ are used to indicate matrices.

Also, the position vector of a reference frame B with respect to a reference frame A in reference frame C is indicated as $\boldsymbol{t}_{A\rightarrow B}^{C}$, while $\boldsymbol{v}_{B/A}^{C}$ represents the velocity (both translational and rotational) of a frame A with respect to frame B in frame C. The superscript is omitted if C coincides with A. Finally, the rotation matrix from reference A to reference B is indicated as $\underline{\underline{R}}_{A}^{B}$ and its corresponding quaternion is denoted as $\boldsymbol{q}_{B/A}$.

The list of the relevant reference frames used within this thesis is here provided and their graphical representation is depicted in Figure 8.

- Inertial Reference Frame (IRF): inertial frame with the origin at the Earth center of mass (COM). The x axis points to the mean equinox of the year 2000, the third axis is aligned with the Earth rotation axis, and the second axis completes the right-hand triad.
- Hill's Reference Frame (HRF): reference system centered in the spacecraft COM. The x axis points outwards the Earth center, the z axis lies along the direction of the angular momentum vector and the y axis completes the right-handed triad. It is useful to describe the relative motion between two spacecraft.
- Chaser Reference Frame (CRF): centered in the center of mass of the chaser. The directions of the three axes are fixed with respect to the spacecraft body.

- Chaser Sensor Fixed Frame (CSFF): a reference frame with the origin in the optical center of the EO sensor with the *z* axis in the sensor boresight direction. The *x* and *y* axes lie in the image plane forming a right-hand triad. The suffix *arm* indicates the reference frame fixed to the EO-sensor mounted on the robotic arm.
- Robotic Arm Base Frame (RABF): reference frame with the origin at the nominal center of the base of the robotic arm.
- Target Reference Frame (TRF): centered in the center of mass of the target whose axes are fixed with respect to the target body.
- Target Attachment Point Frame (TAPF): reference with the origin at the nominal point of contact with the target.

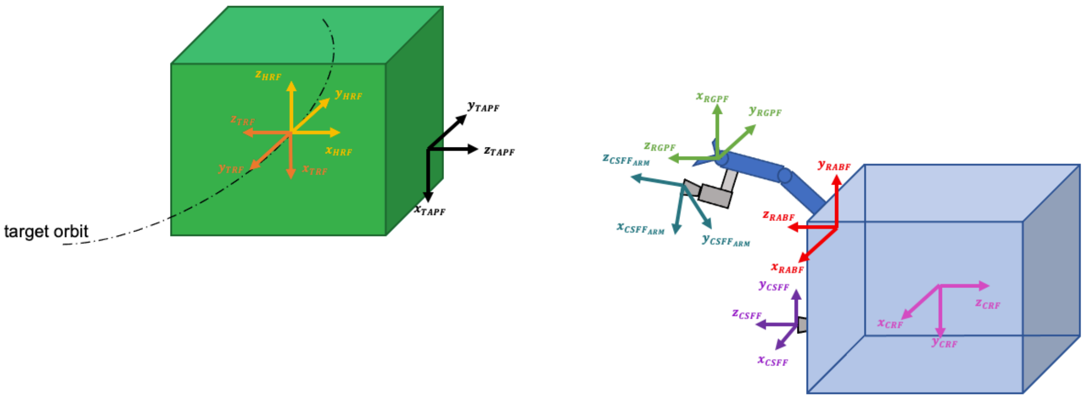


Figure 8 - Representation of the reference frames adopted.

3.2 Relative navigation approach for the monitoring phase

The multi-step approach adopted in the developed relative navigation architecture is highlighted in Figure 9. In the first step, the pose is initialized by means of a template matching technique, [62]. The second step includes the tracking of the pose parameters through an ICP-based algorithms. The pose estimates are also employed to determine the target absolute angular velocity. After a temporal interval large enough to ensure an adequate variation of the target angular velocity, e.g., one relative orbit, these measurements are processed to obtain an estimate of the target MIRs. Finally, in the third step, the full relative state estimation is entrusted to an UKF loosely coupled with the ICP-based pose tracker.

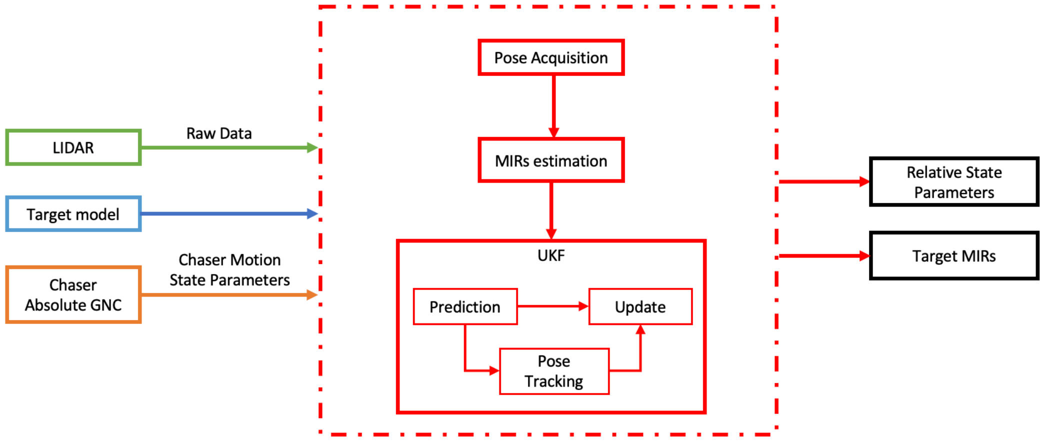


Figure 9 - Block diagram highlighting the multi-step approach of the developed relative navigation architecture for the monitoring phase. The processing steps are highlighted in red, the constant and time-varying input parameters are highlighted in blue and orange, respectively, the outputs in black.

3.2.1 Pose Acquisition

This sub-chapter aims at describing the first step of the relative navigation architecture corresponding to the pose acquisition, i.e., the determination of the pose parameters without any a-priori information on the relative motion state.

Once the first point cloud is acquired by the LIDAR sensor on board the chaser, an initial coarse estimate of the pose parameters is firstly obtained by applying the *on-line PCA-based Template Matching* (PCA-TM) algorithm, [62]. It consists of the following phases. First, a tentative solution for the relative position vector of the chaser with respect to the target is computed as the centroid of the point cloud. Second, according to the Principal Component Analysis [63], the direction of the target main axis is estimated as the eigenvector corresponding to the maximum eigenvalue associated to the point cloud covariance matrix. While two rotational DOFs can be directly derived from this direction, the remaining one is computed by applying a TM approach [62]. Since it is not possible to directly establish whether the target main axis is parallel or antiparallel to the direction estimated by the PCA, an ambiguity arises in the estimation of the main axis orientation, and, consequently in the relative attitude quaternion. This ambiguity in the pose vector is solved by applying twice a customized version of the ICP algorithm (whose main concept are recalled in the next sub-chapter) and choosing the solution characterized by the minimum value of the cost function. It is important to highlight that the initialization is considered successful if the ICP error metric function at convergence and the number of iterations are lower that the pre-defined thresholds. In fact, these parameters represent a measure of the pose estimation accuracy level. Hence, this approach ensures robustness against unfavorable observation conditions which may be encountered at the beginning of the monitoring phase. Clearly, if one of the two condition is not met, the initialization step is re-applied to a new point cloud.

It is worth outlining that the PCA-TM algorithm is tailored to target having a main geometric direction (indeed, most resident space objects, such as rocket bodies, have an elongated structure). However, in case the target does not have this characteristic, the initialization scheme can still be applied by substituting the PCA-TM algorithm with different state-of-the-art techniques able to provide an estimate of the pose parameters without relying on a prior-knowledge of the pose parameters [29], [64]–[67].

3.2.2 Pose tracking and MIRs estimation

Once the pose has been initialized, the target motion is monitored through an ICP-based pose determination strategy. By definition, the ICP is an iterative technique aiming at aligning two datasets by finding correspondences between them and minimizing an error metric function, [67]. In this case, the datasets to be aligned are a model point-cloud (built based in the available information about the target geometry) and the LIDAR point cloud. Hence, the following error metric function can be minimized to get the orientation and the translation vector aligning the two point clouds:

$$f_{ICP}(\boldsymbol{T}, q) = \frac{1}{N_p}\sum_{i=1}^{N_p}\left|\boldsymbol{P}_M^i - \underline{\underline{R}}(q)^T\left(\boldsymbol{P}_L^i + \boldsymbol{T}\right)\right|^2 \tag{51}$$

where $\boldsymbol{P}_M^i$ and $\boldsymbol{P}_L^i$ are the corresponding *i-th* points in the model and LIDAR point cloud, respectively, while N_p is the number of the correspondences, $\boldsymbol{T}$ is the relative position vector of a reference frame fixed to the sensor with respect to acquired point cloud and $\underline{\underline{R}}(\boldsymbol{q})$ is the rotation matrix corresponding to the relative attitude quaternion $\boldsymbol{q}$.

At the same time, also the angular velocity of the target is tracked by exploiting the ICP-based relative attitude knowledge and the data coming from the absolute navigation system of the chaser. The inertial attitude of TRF can be computed as in (59):

$$\boldsymbol{q}_{TRF/IRF}(t_0 + \Delta t) = \boldsymbol{q}_{TRF/IRF}(t_0) \otimes \Delta\boldsymbol{q}_{TRF/IRF} \tag{52}$$

where $\Delta\boldsymbol{q}_{TRF/IRF}$ is the quaternion that describes the target attitude variation between t_0 and $t_0 + \Delta t$. By applying the small angle assumption (which is verified if the pose measurement update rate is large enough with respect to the relative attitude variation rate), the term $\Delta\boldsymbol{q}_{TRF/IRF}$ can be approximated by considering the vector form of the rotation angle between two different time instants, as in Eq. (59).

$$\Delta\boldsymbol{q}_{T/I} \approx \begin{bmatrix} 1 \\ \frac{1}{2}\Delta\boldsymbol{\Phi} \end{bmatrix} \tag{53}$$

Once the rotation vector that describes the change in the attitude of the target is known, its absolute angular velocity can be computed under the small angle assumption as:

$$\boldsymbol{\omega}_{TRF/IRF}^{TRF} \approx \frac{d\boldsymbol{\Phi}}{dt} \tag{54}$$

The time derivative as shown in Eq. (**54**) is computed numerically by applying a first order finite difference method. Since the estimate of the absolute angular velocity of the target must be as accurate as possible in order to correctly compute the target MIRs, the time interval over which the rotation angle is computed is not constant, but it is chosen depending on the accuracy of the measured relative attitude. Indeed, as anticipated, the value of the error metric function at convergence of the ICP algorithm, f_{END}, can be used as a measure of the pose accuracy. The output of the iterative process is used to compute the numerical derivative only if the value of f_{END}, is lower than a selected threshold, τ_{ICP}. Also, once the first good estimate if the relative pose is obtained, the angular velocity is estimated by considering a time interval short enough to keep the small angle assumption valid. Specifically, the equivalent Euler angle which describes the attitude variation between the two time instants must be lower than 10°. After the chaser has collected enough information (e.g., after completing one relative orbit in LEO), the target MIRs can be estimated by exploiting the principle of the conservation of the angular momentum vector. This is possible since the target is a tumbling rigid body which can be assumed torque-free if the effects of the external forces are negligible in the short period. The principle states that the components of the angular momentum vector of a free rigid body in an inertial frame remains constant over time, leading to the following relation,

$$\boldsymbol{h}^{IRF} = \underline{\underline{R}}_{TRF}^{IRF}(t)\underline{\underline{I}}_T \boldsymbol{\omega}_{TRF/IRF}^{TRF}(t) \tag{55}$$

where $\boldsymbol{h}^{IRF}$ is the angular momentum vector in IRF and $\underline{\underline{I}}_T$ is the target inertia matrix. Equation (**6**) represents a linear system with 9 unknowns (i.e., three components of $\boldsymbol{h}^{IRF}$ and six elements of $\underline{\underline{I}}_T$) and can be written in the explicit form $\underline{\underline{A}}\boldsymbol{x} = \boldsymbol{b}$:

$$\begin{bmatrix} \underline{\underline{\Omega}} & -\underline{\underline{R}}_{IRF}^{TRF} \end{bmatrix} \begin{bmatrix} \boldsymbol{I}_v \\ \boldsymbol{h}^{IRF} \end{bmatrix} = [\boldsymbol{0}_{3x1}] \tag{56}$$

where

- $\underline{\underline{\Omega}}$ is a matrix containing the components of the angular velocity vector $\boldsymbol{\omega}_{TRF/IRF}^{TRF} = [\omega_x;\ \omega_y;\ \omega_z]$.

$$\underline{\underline{\Omega}} = \begin{bmatrix} \omega_x & -\omega_y & -\omega_z & 0 & 0 & 0 \\ 0 & -\omega_x & 0 & \omega_y & -\omega_z & 0 \\ 0 & 0 & -\omega_x & 0 & -\omega_y & \omega_z \end{bmatrix} \tag{57}$$

- $\boldsymbol{I}_v$ is a vector containing the six moments of inertia with respect to the TRF axes,

$$\boldsymbol{I}_v = [I_{xx} \quad I_{xy} \quad I_{xz} \quad I_{yy} \quad I_{yz} \quad I_{zz}]' \tag{58}$$

- $\boldsymbol{0}_{3x1}$ is a 3-by-1 null matrix.

It is worth noting that the number of rows in the linear system shown in Eq. (56) is lower than the number of unknows, thus three linearly dependent angular velocity vectors must be considered to find a solution.

Also, since (56) represents a homogeneous system, it cannot be completely solved, but its solution can be determined up to a scale factor. If the linear system is non-dimensional with respect to the term I_{xx}, a unique solution can be determined. Thus, the linear system to be solved becomes:

$$\begin{bmatrix} -\omega_y & -\omega_z & 0 & 0 & 0 & \\ -\omega_x & 0 & \omega_y & -\omega_z & 0 & -R_{IRF}^{BRF} \\ 0 & -\omega_x & 0 & -\omega_y & \omega_z & \end{bmatrix} \begin{bmatrix} \bar{\boldsymbol{I}}_v \\ \bar{\boldsymbol{h}}^{IRF} \end{bmatrix} = \begin{bmatrix} -\omega_x \\ 0 \\ 0 \end{bmatrix} \tag{59}$$

where $\bar{\boldsymbol{I}}_v$ contains the MIRs of the target while $\bar{\boldsymbol{h}}^{IRF}$ is the ration between the inertial components of the angular momentum vector and I_{xx}.

The whole process of the on-line MIRs estimation is summarized in Figure 10. Ideally, the target MIRs could be computed by considering three different angular velocity measured at different time instants. However, the estimated data about the absolute rotational dynamics of the target are affected by noise. So, more than three observations (*N*, in general) are needed to improve the accuracy of the results. Therefore, the system to solve is a 3*N*-by-8 linear system. Moreover, as mentioned before, the measurements used to define the linear system must be linearly independent.

The observation of the angular velocity vector collected while the chaser follows the monitoring trajectory, are not directly included in the linear system, but some post-processing operations are required. First, a Savitzky-Golay filter is applied to the measured time-history of the absolute angular velocity of the target to reduce the level of noise. It is a low pass filter widely used in literature to smooth and differentiate time series of data when dealing with noisy signal, [68].

Once the angular velocity data have been smoothed, a Fast Fourier Transform (FFT) is applied to find the period of the target attitude dynamics. This operation allows excluding repetitive data (which correspond to linearly dependent equations in Eq. (59)) from the observations. Then, a set of smoothed angular velocity measurements are uniformly sampled over the polhode period with a time span large enough to enhance the difference in the inherent information. Finally, a least square solver is applied to solve the system. The MIRs value obtained at the end of this procedure will be used within the next step to improve the accuracy of the dynamic model used to propagate the target rotational dynamics, as described in the next sub-chapter.

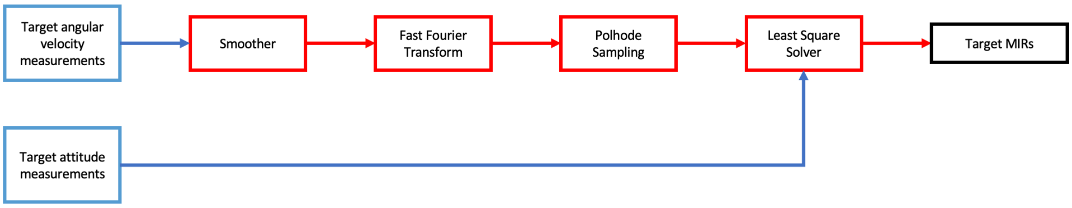


Figure 10 - Block diagram summarizing the MIRs estimation process. The processing steps are highlighted in red, while the inputs are enclosed in blue square-shaped boxes.

3.2.2 Relative navigation filter

The third step of the relative navigation architecture aims at estimating the relative motion parameters between the chaser and the target through a filtering scheme. Following the considerations of Section 2.2, a Kalman Filter has been selected as filtering scheme to estimate the relative velocity and absolute angular velocity of the target, as well as the relative position and attitude of the target with respect to the chaser spacecraft: both the EKF and UKF have been considered as viable option to be applied to the scenario under study.

As mentioned in Section 2.2., regardless of the selected version of the Kalman Filters, the state vector along with a plant and an observation model must be defined.

The state vector is defined as follows:

$$\boldsymbol{x} = [\boldsymbol{t}_{TRF\rightarrow CRF}^{HRF} \; \boldsymbol{v}^{HRF}{}_{CRF/TRF} \; \boldsymbol{q}_{TRF/CRF} \; \boldsymbol{\omega}_{TRF/ECI}] \tag{60}$$

The translational dynamics has been modelled by means of the CW equations (see Eq. (48)) , i.e., a set of differential equation derived from the two body mechanics and by applying a number of hypothesis listed in Table 2. As regards the rotational dynamics, the temporal evolution of the relative attitude quaternion is described by Eq. (49), while the absolute angular velocity of the target has been propagated by means of Euler equation with the hypothesis of a freely tumbling rigid body (i.e., neglecting the effect of external torque).

It is worth noting that if one uses the quaternion parametrization to describe the relative attitude, the covariance matrix associated to the state vector may become singular because of the unit norm constraint which introduces a linear dependence between the rows of the matrix. To overcome this issue, a three-parameters representation for the covariance of the relative attitude can be introduced under the assumption of small angular error, [69]. The Gibbs vector, defined as the ratio between vectorial and scalar part of the attitude quaternion, has been selected in this work as three-parameters representation for the relative attitude error.

In case of EFK use, the matrix $\underline{\underline{J}}$ whose elements contain the partial derivative of the dynamic model equation with respect to the state variable. Being the relative rotational kinematics described by a non-linear equation, a linear model describing the kinematics of the attitude error in terms of the Gibbs vector can be obtained, as in [70] from Eq. (61):

$$\boldsymbol{q} = \delta\boldsymbol{q} \otimes \widehat{\boldsymbol{q}} \tag{61}$$

where $\delta\boldsymbol{q}$ is the quaternion error, defined as the rotation from the estimated attitude quaternion, $\widehat{\boldsymbol{q}}$, to the true one $\boldsymbol{q}$. Taking the time derivative of Eq. (61), and combining it with (49), the kinematics of the quaternion error can be obtained, as shown in Eq. (62).

$$\frac{d}{dt}(\delta\boldsymbol{q}) = \frac{1}{2}\boldsymbol{\omega}_{rel}\otimes\delta\boldsymbol{q} - \frac{1}{2}\delta\boldsymbol{q}\otimes\boldsymbol{\omega}_{rel} \tag{62}$$

Then, by applying the definition of the Gibbs vector, the linearized kinematics of the relative attitude error can be obtained:

$$\delta\dot{\boldsymbol{g}} = \frac{1}{2}\delta\boldsymbol{\omega}_{\boldsymbol{rel}} - \frac{1}{2}\boldsymbol{\omega}_{rel} \times \delta\boldsymbol{g} - \frac{1}{2}\widehat{\boldsymbol{\omega}}_{rel} \times \delta\boldsymbol{g} \tag{63}$$

From Eq. (63), recalling that $\boldsymbol{\omega_{rel}}$ is defined as in Eq. (64) and being $\delta\boldsymbol{\omega_{rel}}$ the difference between the estimated and the true relative angular velocity, the partial derivative with respect to state vector components can be evaluated.

$$\boldsymbol{\omega}_{rel} = \boldsymbol{\omega}_{TROF/ECI} - \underline{\underline{R}}_{CROF\to TROF}\boldsymbol{\omega}_{CROF/ECI} \tag{64}$$

For all the other equations, it is straightforward to derive the partial derivatives with respect to the state variables. For the sake of clarity, the expressions of the components of the $\underline{\underline{J}}$ matric are reported below.

$$\underline{\underline{J}} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3n^2 & 0 & 0 & 0 & 2n & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2n & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -n^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \omega_z & -\omega_y & 1/2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\omega_z & 0 & \omega_x & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \omega_y & -\omega_x & 0 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & A_{11} & A_{12} & A_{13} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & A_{21} & A_{22} & A_{23} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & A_{31} & A_{32} & A_{33} \end{bmatrix} \tag{65}$$

where the matrix $\underline{\underline{A}}$ is:

$$\underline{\underline{A}} = \underline{\underline{I_t}}^{-1}[[\boldsymbol{\omega}_{TROF/ECI}\text{x}](\underline{\underline{I_t}}\,\boldsymbol{\omega}_{TROF/ECI} - [\boldsymbol{\omega}_{TROF/ECI}\text{x}]\underline{\underline{I_t}}$$

As stated in Section 2.2.1, the observation model is a system of equation which relates the available measurements, i.e., the output of the pose determination algorithm, with the system state and shown in the following:

$$\hat{\boldsymbol{z}} = \begin{cases} R_{CRF\to CSFF}(-\boldsymbol{t}_{CRF\to CSFF} - R_{HRF\to CRF}\boldsymbol{t}_{TRF\to CRF}\ ) \\ \boldsymbol{q}_{CSFF/CRF} \otimes \boldsymbol{q}_{CRF/TRF} \end{cases} \tag{66}$$

Since the observation model consists of non-linear equations, the sensitivity matrix $\underline{\underline{H}}$, must be computed as the Jacobian matrix of the observation model with respect to the state variables evaluated at the current state estimate. The expression of the components of the sensitivity matrix are shown in Eq. (67).

$$\underline{\underline{H}} = \begin{bmatrix} -\underline{\underline{R}}_{CRF\to CSFF}\underline{\underline{R}}_{HRF\to CRF} & 0_{3x3} & 0_{3x3} & 0_{3x3} \\ 0_{3x3} & 0_{3x3} & -\underline{\underline{R}}_{CRF\to CSFF}\underline{\underline{R}}_{TRF\to CRF} & 0_{3x3} \end{bmatrix} \quad (67)$$

It is worth noting that since in both the state and measurement vector, the attitude is parametrized by means of unit quaternion, the innovation term $\boldsymbol{\Delta z}$**,** conventionally computed as the difference between the measurement vector and the projection of the state in the measurement space, is computed as follows:

$$\Delta \boldsymbol{z} = \begin{bmatrix} \boldsymbol{t}_{CSFF\to TRF} - \hat{\boldsymbol{t}}_{CSFF\to TRF} \\ \boldsymbol{g}(\boldsymbol{q}_{CSFF/TRF} \otimes \hat{\boldsymbol{q}}_{CSFF/TRF}) \end{bmatrix} \quad (68)$$

where $\hat{\boldsymbol{t}}_{CSFF\to TRF}$ and $\hat{\boldsymbol{q}}_{CSFF/TRF}$ are the components of $\hat{\boldsymbol{z}}$, while $\boldsymbol{t}_{CSFF\to TRF}$ and $\boldsymbol{q}_{CSFF/TRF}$ are the measured pose parameters**.** The innovation term must be pre-multiplied by the Kalman gain to get the correction term, $\Delta\boldsymbol{x}_{up}$, as follows:

$$\Delta\boldsymbol{x}_{up} = \underline{\underline{K}}(\Delta\boldsymbol{z}) \quad (69)$$

The correction term is used to get the a-posteriori estimate of the state and covariance according to Eq. (70) and (**16**), respectively.

$$\hat{\boldsymbol{x}}_k^+ = \begin{bmatrix} \boldsymbol{t}^{HRF}{}_{TRF\to CRF}{}_k^+ \\ \boldsymbol{v}^{HRF}{}_{CRF/TRF}{}_k^+ \\ \boldsymbol{q}_{TRF/CRF}{}_k^+ \\ \boldsymbol{\omega}_{TRF/ECI}{}_k^+ \end{bmatrix} = \begin{bmatrix} \boldsymbol{t}^{HRF}{}_{TRF\to CRF}{}_k^- + \Delta\boldsymbol{t}^{HRF}{}_{TRF\to CRF} \\ \boldsymbol{v}^{HRF}{}_{CRF/TRF}{}_k^- \Delta\boldsymbol{v}^{HRF}{}_{CRF/TRF} \\ \dfrac{[1\ \boldsymbol{\Delta g}]}{\sqrt{1+\Delta\boldsymbol{g}(\boldsymbol{\Delta g})'}} \otimes \boldsymbol{q}_{TRF/CRF}{}_k^- \\ \boldsymbol{\omega}_{TRF/ECI}{}_k^- + \Delta\boldsymbol{\omega}_{TRF/ECI} \end{bmatrix} \quad (70)$$

3.3 Relative navigation approach for the final approach phase

In the final approach phase of an ADR/OOS mission in which the chaser must capture the target with a robotic arm, as stated in Section 2.4, the relative navigation function must be entrusted to two EO sensors, one mounted on the body of the chaser and the other on the end-effector of the robotic-arm. Based on the trade-off considerations presented in Section 2.4, the *Option 1* is the most convenient architectural choice to support the capture operations, as it can provide a solution for the gripper-grasping point relative state without including the robotic arm ego-motion uncertainty. A high-level block diagram for this architecture highlighting its main processing steps and interface is provided in Figure 11. The main blocks are highlighted in red, while the blocks providing constant and time-varying inputs are highlighted in orange and blue, respectively. The architecture is loosely coupled, i.e., the data collected by the body-mounted and robotic arm EO sensors, as well as by the joints' sensors, are processed within separate block to obtain measurements used by the filtering schemes. Specifically, the measurement set for the *T/C relative state estimation* block is given only by the T/C pose estimates (obtained processing the body mounted sensor data). Instead, the measurement set for the *G/E relative state estimation* block comprises the G/E pose estimates (obtained processing the robotic arm sensor data) and the C/E pose estimates (obtained by applying the robotic arm forward kinematic model which requires joints' sensors data). An important aspect to highlight is the fact that the robotic arm EO sensor is used to produce direct pose measurements of the end effector with respect to the grasping point only in the final portion of the reach and capture trajectory, i.e., when the robotic arm is fully deployed and pointed to the selected grasping point up to contact. However, the *G/E relative state estimation* block is adopted to measure end-effector-grasping point relative state estimate during the entire reach and capture phase, i.e., even in absence of these direct G/E pose measurements. As regards the choice of the robotic arm sensor, a Time of Flight (TOF) camera is selected instead of a LIDAR (selected as main body sensor) since it poses lower constraints to the allocation on the robotic arm end effector in terms of cost, weight, size and power consumption.

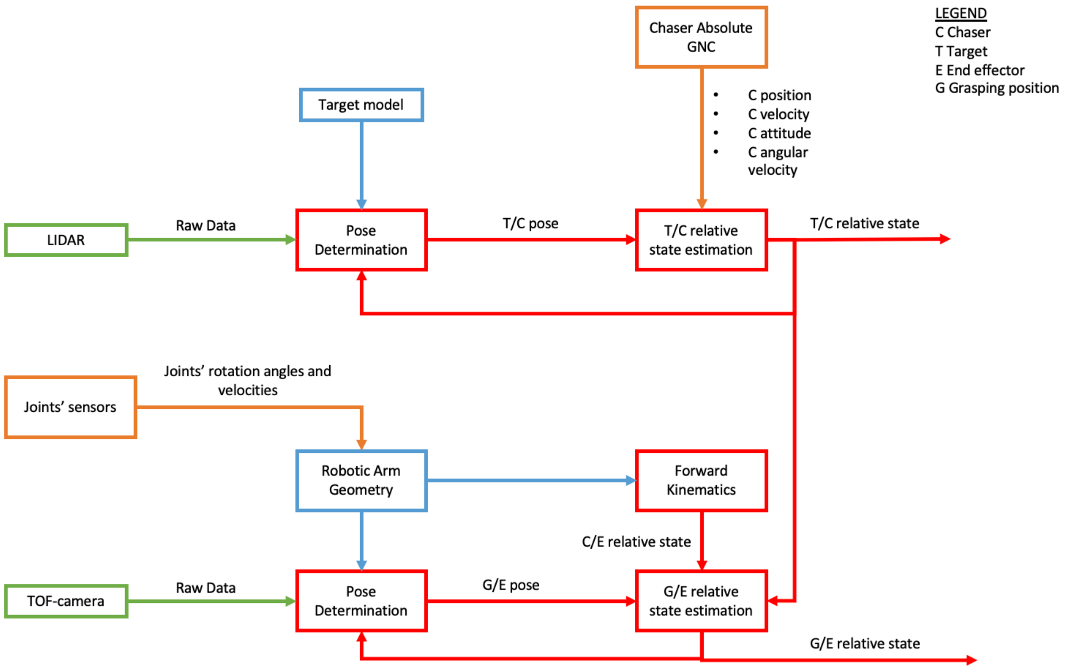


Figure 11 - Block diagram describing at high level the architectural choice for the relative navigation subsystem for the final approach phase of an ADR/OSS mission. The processing steps are highlighted in red, the constant and time-varying input parameters are highlighted in blue and orange, respectively.

While the pose determination and the T/C relative state estimation blocks are the same adopted in the relative navigation architecture developed for the monitoring phase, the remaining processing block are described in the following.

3.2.1 Robotic arm forward kinematics

The relative state of a reference frame attached to the end-effector of the robotic arm with respect to its base fixed to the chaser body can be determined as a function of the joints' angles by exploiting the forward kinematics model of the robotic arm. The most common approach for describing the robot kinematics is the Denavit-Hartenberg (DH) method which uses a four-parameters representation: the link length a_i, the link twist α_i, the link offset d_i and the joint angle θ_i, The homogeneous matrix of Eq. (71) defines the transformation between two frames attached to two consecutive links.

$$\underline{\underline{A}}_i^{i-1} = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & a_i \\ \sin\theta_i \cos\alpha_i & \cos\alpha_i \cos\theta_i & -\sin\alpha_i & -d_i \sin\alpha_i \\ \sin\theta_i \sin\alpha_i & \cos\theta_i \sin\alpha_i & \cos\alpha_i & d_i \cos\alpha_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{71}$$

For a n-DOF manipulator, the homogeneous matrix defining the transformation between the frame attached to the base of the robotic arm and the frame attached to the end effector, is given by Eq. (72).

$$\underline{\underline{A}} = \underline{\underline{A}}_1^0 \cdots \underline{\underline{A}}_{n-1}^{n-2} \underline{\underline{A}}_n^{n-1} = \underline{\underline{A}}_n^0 \tag{72}$$

Thus, knowing the joint variables, the position of the end-effector with respect to a frame fixed to the base is given by Eq. (73), while the rotation matrix is given by Eq. (74).

$$\boldsymbol{P_{ee}} = A(1{:}3,4) \tag{73}$$

$$\underline{\underline{\boldsymbol{R}}} = A(1{:}3,1{:}3) \tag{74}$$

The relationship between joint velocities and end-effector velocities is provided by the Jacobian matrix, which is computed with a geometric approach, i.e., by considering the contributions of each joint velocity to the components of the end-effector linear and angular velocities, as shown in Eq. (75),

$$\begin{bmatrix} \boldsymbol{v} \\ \boldsymbol{\omega} \end{bmatrix} = \mathrm{J}(\mathbf{q})\dot{\boldsymbol{q}}, \qquad J(q) = \begin{bmatrix} \boldsymbol{z_1} \times (\boldsymbol{p_{ee}} - \boldsymbol{p_1}) & \cdots & \boldsymbol{z_i} \times (\boldsymbol{p_{ee}} - \boldsymbol{p_i}) \\ \boldsymbol{z_1} & \cdots & \boldsymbol{z_i} \end{bmatrix} \tag{75}$$

where

- Z_i represents the axis of rotation of the ith joint expressed in the base frame. It is obtained from the third column of the homogeneous matrix,
- $\boldsymbol{p_{ee}}$ is the position of the end-effector in the base frame,
- $\boldsymbol{P_i}$ is the position of the ith joint expressed in the frame.

3.2.2 Grasping point-end effector relative state estimation

The grasping point-end effector relative state estimation is entrusted to an EKF which exploits as measurements (i) the output of the forward kinematics model and (ii) the grasping point-end effector pose obtained by processing the raw-data of the TOF camera (when available).

The state vector of the G/E filter is defined as follows:

$$\boldsymbol{x} = [\boldsymbol{t}_{RGPF \to TAPF} \;\; \boldsymbol{v}_{TAPF/RGPF} \;\; \boldsymbol{q}_{RGPF/TAPF} \;\; \boldsymbol{\omega}_{RGPF/TAPF}] \tag{76}$$

As for the target-chaser relative navigation filter, the covariance associated to the estimated relative attitude is expressed in terms of the Gibbs vector to avoid singularity problems. The process, in this thesis, is described by a constant velocity and angular velocity model, as follows:

$$\begin{cases} \dot{\boldsymbol{v}}_{TAPF/RGPF} = 0 \\ \dot{\boldsymbol{q}}_{RGPF/TAPF} = \frac{1}{2}[0 \; \boldsymbol{\omega}_{RGPF/TAPF}] \otimes \boldsymbol{q}_{RGPF/TAPF} \\ \dot{\boldsymbol{\omega}}_{RGPF/TAPF} = 0 \end{cases} \tag{77}$$

By applying the definition of the Gibbs vector to the model described in Eq. (77), it is straightforward to derive the state transition matrix according to its definition given in Eq. (14), whose components are shown below:

$$\underline{\underline{\Phi}} = \begin{bmatrix} I_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & I_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & I_{3\times3} & \frac{dt}{2} I_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & I_{3\times3} \end{bmatrix} \tag{78}$$

The a-priori state estimate has to be projected into the measurement space through the observation model of Eq. (79). Depending on whether the target falls in the FOV of the robotic arm TOF camera or not, the observation model can include or not the equations relating the state vector with the measured pose parameters.

$$
\hat{\boldsymbol{z}} = \begin{cases}
\boldsymbol{t}_{CRF \to RGPF} = -\underline{\underline{R}}_{HRF}^{CRF} \boldsymbol{t}_{TRF \to CRF}^{HRF} + \\
+\underline{\underline{R}}_{TRF}^{CRF} \left(\boldsymbol{t}_{TROF \to TAPF} - \underline{\underline{R}}_{TAPF}^{RGPF} \boldsymbol{t}_{RGPF \to TAPF} \right) \\
\boldsymbol{v}_{RGPF/CRF} = \frac{1}{2} \Big(-\underline{\underline{R}}_{TRF}^{CRF} \left(\underline{\underline{R}}_{RGPF}^{TAPF} \left(\boldsymbol{v}_{TAPF/RGPF} + \boldsymbol{\omega}_{RGPF/TAPF} \times \boldsymbol{t}_{RGPF \to TAPF} \right) + \right. \\
+\boldsymbol{\omega}_{TRF/CRF} \times \underline{\underline{R}}_{RGPF}^{TAPF} t_{RGPF \to TAPF} - \boldsymbol{\omega}_{TRF/CRF} \times \boldsymbol{t}_{TRF \to TAPF}) + \\
-2 \underline{\underline{R}}_{LVLH}^{CRF} \left(\boldsymbol{v}_{\frac{CRF}{TRF}}^{HRF} + \boldsymbol{\omega}_{\frac{HRF}{CRF}} \times \boldsymbol{t}_{TRF \to CRF}^{HRF} \right) \Big) \\
\boldsymbol{q}_{RGPF/CRF} = \boldsymbol{q}_{RGPF/TAPF} \otimes \boldsymbol{q}_{TRF/CRF} \\
\boldsymbol{\omega}_{RGPF/CRF} = \boldsymbol{\omega}_{RGPF/TAPF} + \underline{\underline{R}}_{TAPF}^{RGPF} \boldsymbol{\omega}_{TRF/CRF} \\
\boldsymbol{t}_{CSFFarm \to TAPF} = -\underline{\underline{R}}_{RGPF}^{CSFFarm} \boldsymbol{t}_{RGPF \to CSFFarm} + \underline{\underline{R}}_{RGPF}^{CSFFarm} \boldsymbol{t}_{RGPF \to TAPF} \\
\boldsymbol{q}_{CSFFarm/TAPF} = \boldsymbol{q}_{SFFarm/RGPF} \otimes \boldsymbol{q}_{RGPF/TAPF}
\end{cases} \tag{79}
$$

As for the T/C relative navigation filter, the observation model is not linear and the sensitivity matrix $\underline{\underline{H}}$ has to be computed as the partial derivatives of Eq. (79) with respect to the state variables, as follows:

$$
\underline{\underline{H}} = \begin{bmatrix}
-\underline{\underline{R}}_{TROF}^{CROF} \underline{\underline{R}}_{RGPF}^{TAPF} & 0_{3x3} & 2\underline{\underline{R}}_{TROF}^{CROF} \underline{\underline{R}}_{RGPF}^{TAPF} [\boldsymbol{t}_{RGPF \to TAPF} \times] & 0_{3x3} \\
A & -\frac{1}{2} \underline{\underline{R}}_{TROF}^{CROF} \underline{\underline{R}}_{RGPF}^{TAPF} & B & 0_{3x3} \\
0_{3x3} & 0_{3x3} & I_{3x3} & 0_{3x3} \\
0_{3x3} & 0_{3x3} & 2\underline{\underline{R}}_{TAPF}^{RGPF} [\boldsymbol{\omega}_{TROF/CROF} \times] & I_{3x3} \\
\underline{\underline{R}}_{CSFFarm}^{RGPF} & 0_{3x3} & 0_{3x3} & 0_{3x3} \\
0_{3x3} & 0_{3x3} & \underline{\underline{R}}_{CSFFarm}^{RGPF} & 0_{3x3}
\end{bmatrix} \tag{80}
$$

with

$$
A = -\frac{1}{2} \underline{\underline{R}}_{TRF}^{CRF} \underline{\underline{R}}_{RGPF}^{TAPF} ([\boldsymbol{\omega}_{RGPF/TAPF} \times] \left[\left(\underline{\underline{R}}_{RGPF}^{RGPF} \boldsymbol{\omega}_{TRF/CRF} \right) \times \right]
$$

$$
\boldsymbol{B} = \underline{\underline{R}}_{TRF}^{CRF} \left(\underline{\underline{R}}_{RGPF}^{TAPF} \left([v_{TAPF/RPGF} \times] + [(\omega_{RPGF/TAPF} \times t_{RGPF \to TAPF}) \times] \right) \right.
$$

$$
\left. + [\omega_{TRF/CRF} \times] \underline{\underline{R}}_{RGPF}^{TAPF} [t_{RGPF \to TAPF} \times] \right)
$$

Similarly to the T/C relative state filter, the Kalman gain pre-multiplies the innovation vector $\boldsymbol{\Delta z}$ (see Eq. (81)) in order to get the correction term $\boldsymbol{\Delta x}_{up}$. Finally, the a-posteriori state and covariance estimate can be computed according to Eq. (82) and Eq. (16), respectively.

$$\Delta \boldsymbol{z} = \begin{bmatrix} \boldsymbol{t}_{RGPF \to TAPF} - \hat{\boldsymbol{t}}_{RGPF \to TAPF} \\ \boldsymbol{v}_{TAPF/RGPF} - \hat{\boldsymbol{v}}_{TAPF/RGPF} \\ \boldsymbol{g}(\boldsymbol{q}_{RGPF/TAPF} \otimes \hat{\boldsymbol{q}}_{RGPF/TAPF}) \\ \boldsymbol{\omega}_{RGPF/TAPF} - \hat{\boldsymbol{\omega}}_{RGPF/TAPF} \\ \boldsymbol{t}_{CROF \to RGPF} - \hat{\boldsymbol{t}}_{CROF \to RGPF} \\ \boldsymbol{g}(\boldsymbol{q}_{RGPF/CROF} \otimes \hat{\boldsymbol{q}}_{RGPF/CROF}) \end{bmatrix} \tag{81}$$

$$\hat{\boldsymbol{x}}_{\boldsymbol{k}}^{+} = \begin{bmatrix} \boldsymbol{t}_{RGPF \to TAPF}{}_k^+ \\ \boldsymbol{v}_{TAPF/RGPF}{}_k^+ \\ \boldsymbol{q}_{RGPF/TAPF}{}_k^+ \\ \boldsymbol{\omega}_{RGPF/TAPF}{}_k^+ \end{bmatrix} = \begin{bmatrix} \boldsymbol{t}_{RGPF \to TAPF}{}_k^- + \Delta \boldsymbol{t}_{RGPF \to TAPF} \\ \boldsymbol{v}_{TAPF/RGPF}{}_k^- \Delta \boldsymbol{v}_{TAPF/RGPF} \\ \dfrac{[1\ \boldsymbol{\Delta g}]}{\sqrt{1 + \Delta \boldsymbol{g}(\boldsymbol{\Delta g})'}} \otimes \boldsymbol{q}_{RGPF/TAPF}{}_k^- \\ \boldsymbol{\omega}_{RGPF/TAPF}{}_k^- + \Delta \boldsymbol{\omega}_{RGPF/TAPF} \end{bmatrix} \tag{82}$$

4. GNC Architecture Performance Assessment

This chapter describes the numerical simulation environment and the performance assessment of the developed relative navigation architectures described in Chapter 3.

4.1 Numerical simulation environment

The numerical simulation environment developed in MATLAB/Simulink reproduces both the target-chaser relative dynamics and the operation of a scanning LIDAR. A block diagram showing all the inputs needed to perform the numerical simulations and how they are collected is depicted in Figure 12.

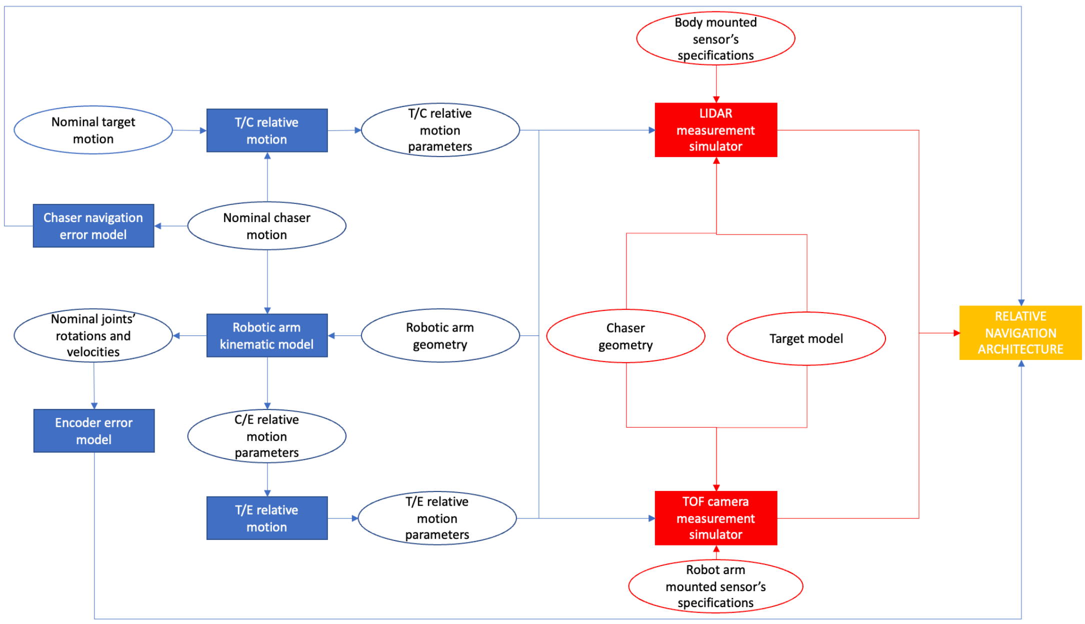


Figure 12 - Block diagram of the numerical simulation environment.

The functioning of the simulator can be summarized as follows.

- The nominal trajectories of the target and the chaser are required in input by the *T/C relative motion* block to obtain the nominal relative motion parameters required in input by the *LIDAR simulator* block to simulate the output of the body mounted LIDAR system. Clearly, to this aim, the *LIDAR/TOF simulator* block also require in input the target model, the sensors specifications, as well as the chaser and robotic arm (when applicable) geometries. It is worth underlying that the TOF camera and LIDAR measurements are simulated with the same process.

- The *T/E relative motion* block is entrusted to simulate the motion of the robotic arm. It requires in input the nominal trajectories of the target and the chaser as well as the nominal joint' rotations. Then, the output of this block is fed to the *LIDAR simulator* block to simulate the output of the TOF camera installed on the end-effector. To this aim, also the target model along with the sensor's intrinsic parameters and the chaser and robotic arm geometries are required.
- The simulated absolute trajectory of the chaser, the joint's rotations and the synthetic generated point cloud are used as input by the *Relative Navigation Architecture* block. Clearly, this block also requires in input a set of specific target geometric information required for the application of the pose determination algorithm.
- The *encoder error model* and *chaser navigation error* model are used to simulate the joint's rotations and chaser absolute navigation state measurements from their nominal values.

4.1.1 Nominal relative motion parameter generation

The goal of this paragraph is to describe the operations carried out within the *T/C relative motion* and *T/E relative motion* blocks.

First, both the target and chaser absolute translational motion have been propagated using the General Mission Analysis Tool (GMAT) starting from the instantaneous orbit parameters which satisfy the initial relative geometry of the scenario under study. For the propagation of the nominal trajectory the main orbital perturbations (i.e., aerodynamic drag, gravitational harmonics up to the fourth order and the solar radiation pressure) have been included.

With regards to the rotational motion, the time variation of the attitude parameters and angular velocity of the target have been obtained by numerically propagating the quaternion kinematic equation and the Euler equation including the gravity-gradient torque and a disturbance torque modelled with an harmonic law as in [71], as shown in Eq. (83)

$$M = [10^{-4}(1 - \sin(\omega_{ORB}t)) \quad 10^{-4}\cos(\omega_{ORB}t) \quad -10^{-4}\cos(\omega_{ORB}t)] \tag{83}$$

where ω_{ORB} is the orbital angular velocity. On the other hand, for the rotational motion of the chaser, it has been assumed that the chaser is controlled so that the LIDAR's boresight axis is always pointed towards the center of mass of the target. Then, the target-chaser motion parameters have been computed by combining the absolute motion parameters of both spacecraft.

The gripper-grasping point relative trajectory is computed by solving the inverse kinematics of the manipulator in two different instants: at an intermediate instant and at the final one so to guarantee that RGPF and TAPF are aligned and grasping point to be tracked is enclosed in the FOV of the TOF camera in its operative range. The nominal joint rotations and velocities are obtained over the entire time frame through a linear interpolation.

4.1.2 LIDAR measurement simulator

The *LIDAR measurement* block requires in input the sensor's specification, e.g., the FOV and the angular resolution (δ_{LOS}). This block comprises three modules, as described in detail in [72]. The first block generates a purely geometric point cloud by applying a ray-tracing algorithm (no source of noise considered). The ray tracing algorithm finds the range of interception between each transmitted laser beam according to the defined scan pattern and the closest surface of the geometric model of the target.

Once the ideal point cloud is computed, the LIDAR detection process is simulated in order to establish whether the backscattered laser beams are detected or not. This is done by evaluating the probability of detection (P_D) of each received echo as a function of the probability of false alarm (P_{FA}) and the Signal to Noise Ratio (SNR), as in [73].

Finally, the detected point cloud is modified considering the sensor measurement uncertainties, which are simulated as a Gaussian White Noise on the measured range (σ_ρ) and laser beam direction (σ_{LOS}). The possibility to produce outliers as a percentage of the detected points (*O%*) is also considered. Specifically, they are randomly extracted among the elements of the measured point cloud and their range uncertainty is set to four times σ_ρ.

4.1.3 Error models

The *Chaser navigation error model* and *encoder error model* blocks allows adding a time correlated error (s_p) to a true signal (s_t). this error is obtained by adding a white Gaussian noise (with zero mean and σ standard deviation) through a discrete-time lowpass filter, whose input-output relation is as follows:

$$y(t) = a\, y(t-1) + b\, x(t), \qquad t = nT, \qquad n = 1,2,\dots \tag{84}$$

where y is the output signal, x is the input one and T is the sampling period. This means that, for a given input signal, the corresponding output at each time step is a linear combination of the output at the previous time step and of the input at the current one. The values that have been selected for the a and b coefficients of the filter are the following: $a = 0.99\ and\ \ b = 1$.

4.2 Error metrics definition

The error metrics adopted to describe the performance of the relative navigation architectures are defined in the following.

First, the error level for each component of the relative position, velocity and angular velocity vectors can be defined. As an example, considering a generic vector $\boldsymbol{a}$, the following error metrics can be introduced:

$$e(\boldsymbol{a}) = \boldsymbol{a}_{est} - \boldsymbol{a}_{true} \tag{85}$$

where the subscript "*est*"/"*true*" indicates the estimated/true quantity. Also, a more synthetic metric can be introduced by considering the error in the norm of the relative position, velocity and angular velocity vectors: for the generic vector $\boldsymbol{a}$ is defined as:

$$|\boldsymbol{a}|_{ERR} = |\boldsymbol{a}_{est}| - |\boldsymbol{a}_{true}| \tag{86}$$

where |.| indicates the Euclidean norm operator.

With regards to the relative attitude, while typically the output of the navigation system is represented using the quaternion parametrization, it is more convenient to use angular parameters as performance metrics since its interpretation is more straightforward with respect to the quaternion error. Thus, the attitude estimation error can be evaluated as the error in each Euler angles by introducing the following error metrics:

$$\begin{aligned} \Delta\alpha &= \alpha_{est} - \alpha_{true} \\ \Delta\beta &= \beta_{est} - \beta_{true} \\ \Delta\gamma &= \gamma_{est} - \gamma_{true} \end{aligned} \tag{87}$$

However, to avoid ambiguities in the error calculation when Euler angles are close to their singular sequences (e.g., if β = 90°), the relative attitude error can be also evaluated by computing the angular deviations between the true and estimated direction of a generic reference frame A in a generic reference frame B, as follows:

$$e_{dir_A} = \begin{bmatrix} \cos^{-1}(\underline{\underline{R}}_{1,est} \cdot \underline{\underline{R}}_{1,true}) \\ \cos^{-1}(\underline{\underline{R}}_{2,est} \cdot \underline{\underline{R}}_{2,true}) \\ \cos^{-1}(\underline{\underline{R}}_{3,est} \cdot \underline{\underline{R}}_{3,true}) \end{bmatrix} \tag{88}$$

Also, a synthetic metrics for the attitude error can be introduced with the Equivalent Euler angle error:

$$\Phi_{ERR} = 2\cos^{-1}(q_{0,ERR}) \tag{89}$$

where $q_{0,ERR}$ is the scalar component of the quaternion error $\boldsymbol{q}$ computed as:

$$\boldsymbol{q}_{ERR} = \boldsymbol{q}_{true} \otimes \boldsymbol{q}_{est}^{-1} \tag{90}$$

4.3 Relative Navigation for state and target inertia estimations: numerical results

The relative navigation architecture presented in paragraph 3.1 has been tested in the simulation environment described in paragraph 4.1 considering four different target geometries placed on both LEO and GEO orbits, as shown in the following. The UKF scheme has been adopted to estimate the relative state parameters of the chaser with respect to the target. Indeed, the UKF allows capturing better the non-linearity of the relative dynamics with respect to the MEKF in case of a freely tumbling target.

For all the test cases described in the next paragraphs, the same LIDAR sensor, whose characteristics are listed in Table 4 have been considered. These data are consistent with the characteristics of this kind of system for spaceborne applications, [64], [74].

Table 4 - Scanning LIDAR operational and noise parameters.

Operational parameters	
Field of view (FOV)	40°x40°
Resolution (δ_{LOS})	1°
Scan Frequency (f_L)	1 Hz
Noise parameters	
Range Uncertainty (σ_ρ)	2.5 cm
LOS uncertainty (σ_{LOS})	0.0007°
Outlier percentage (O%)	5%

4.3.1 Test Case A

The target considered in the first simulation scenario is ENVISAT, an eight tons satellite for Earth observation declared inoperative in 2021. As described in paragraph 3.1, the algorithm requires a target model with which the measured point cloud must be compared for pose determination purposes. Here, a simplified geometric model of the selected target has been built. As shown in Figure 13, its geometry is made of three cuboid-shaped elements which represents the main body, the solar panel and the synthetic aperture radar (SAR) antenna with the related appendixes. Table 5 collects the main geometrical and physical characteristics used to generate the target model and for the LIDAR measurements simulation, as found in the open literature, [75], [76], [77].

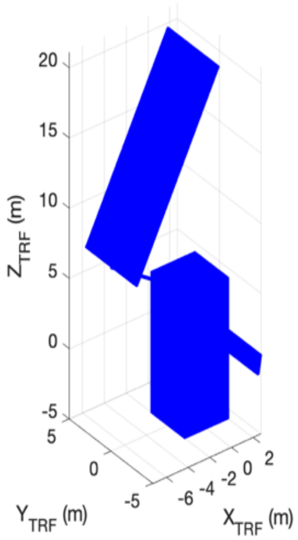


Figure 13 - Point-cloud representation of the simplified geometric model of ENVISAT.

Table 5 - ENVISAT: geometrical and physical characteristics.

Element	**Dimension**	**Reflection coefficient**	**Moment of inertia ($kg \cdot m^2$)**			
Main body	4 m x 4 m x 10 m	0.15	I_{xx}	129112.2	I_{xy}	-344.2
Sola panel	6 m x 15 m	0.97	I_{yy}	124825.7	I_{xz}	271.4
SAR antenna	1.3 m x10 m	0.17	I_{zz}	17023.3	I_{yz}	397.1

The relative trajectory travelled by the chaser has been designed following the approach presented in [78] which ensures passive safety and guarantees favorable relative observation geometries. In Figure 14, the designed safety ellipse for monitoring purpose and the time variation of the target-chaser relative range are depicted: it can be noted that the relative distance varies from 25 to 57 meters. Figure 15 shows the polhode of the target, i.e., the curve traced by the angular velocity vector in the body-fixed frame.

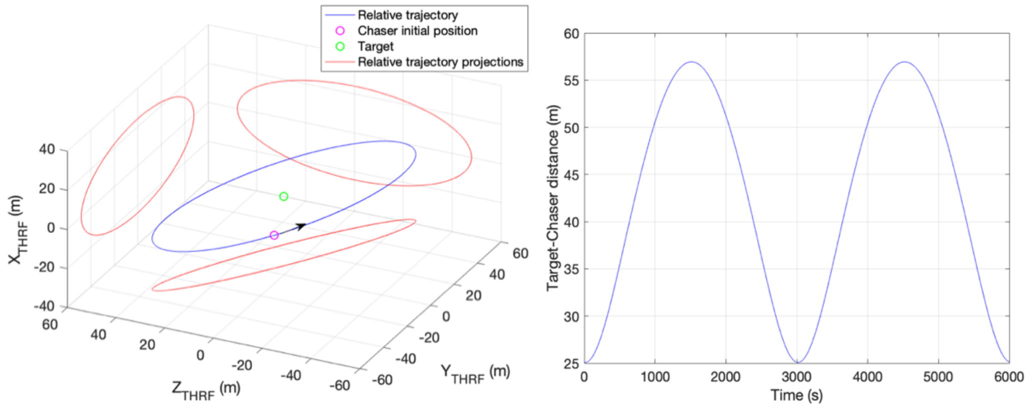


Figure 14 - Relative trajectory around ENVISAT: (left) safety ellipse for monitoring purpose. (right) Time variation of the target-chaser distance.

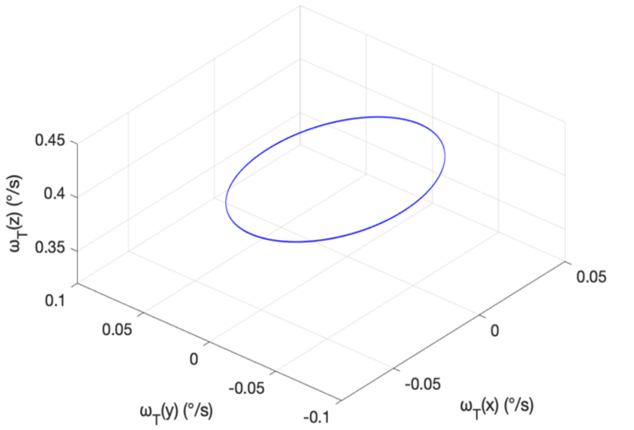


Figure 15 – Polhode of the target. Test Case A.

As described in paragraph 3.1, the first part of the proposed strategy is aimed at collecting relative attitude information useful to estimate the target MIRs relying only on pose determination algorithms. The time variation of both relative position and attitude error metrics averaged over 100 simulations is depicted in Figure 16 , while the corresponding root mean square (RMS) and the maximum error values, evaluated over one relative orbit, are collected in Table 6. These results show that pose determination in a stand-alone configuration is able to achieve a centimeter and sub-degree level of accuracy in the estimate of the relative position and attitude, respectively.

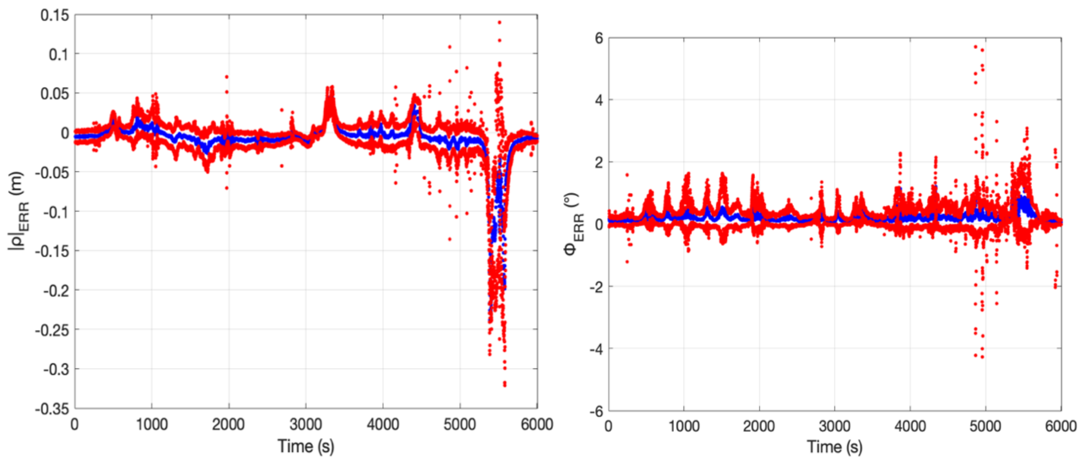


Figure 16 - Mean (blue) and 3σ bounds (red) of the error metrics relevant to the pose tracking phase. (left) relative position, (right) relative attitude. Results are averaged over 100 simulations. Test Case A.

Table 6 - Statistics for the time variation of the error metric functions in the pose tracking phase (averaged over 100 simulations) computed over one relative orbit. Test Case A.

Error metrics	RMS	Maximum value
$\|\rho\|_{ERR}$ (m)	$2.8 \cdot 10^{-2}$	0.3
Φ_{ERR} (°)	0.2	9.4

The time variation of the pose error is characterized by the presence of some peaks related to unfavorable target-chaser observation geometries for pose estimation purposes, which tend to occur while the chaser moves around the target, [62], [79].

Then, the retrieved pose information is exploited to estimate the absolute angular velocity of the target. Specifically, it is estimated with a mean error of $3 \cdot 10^{-4}$ degrees per seconds and a standard deviation of $3.5 \cdot 10^{-2}$ degrees per seconds. Figure 17 shows the time variation of the error metric defined for the absolute angular velocity of the target. It shows that, due to the error in the relative attitude measurements, the absolute angular velocity estimates obtained through a first order derivative scheme are particularly noisy.

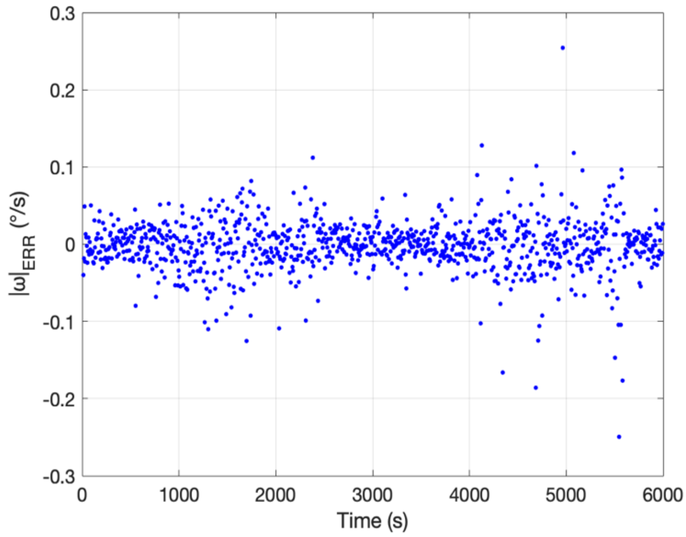


Figure 17 - Time variation of the error metric defined for the absolute angular velocity of the target. Test Case A.

The accuracy in the measurements of the target absolute angular velocity is improved by applying the Savitzky-Golay smoother. This operation is effective in reducing the level of noise. In fact, after smoothing, the mean error evaluated over 100 simulations is 1.2·10-5 degrees per seconds while the standard deviation is $3.9 \cdot 10^{-3}$ degrees per seconds, which means an improvement of one order of magnitude in both values. Figure 18 depicts the mean value and the 3σ bounds of the absolute angular velocity error after smoothing.

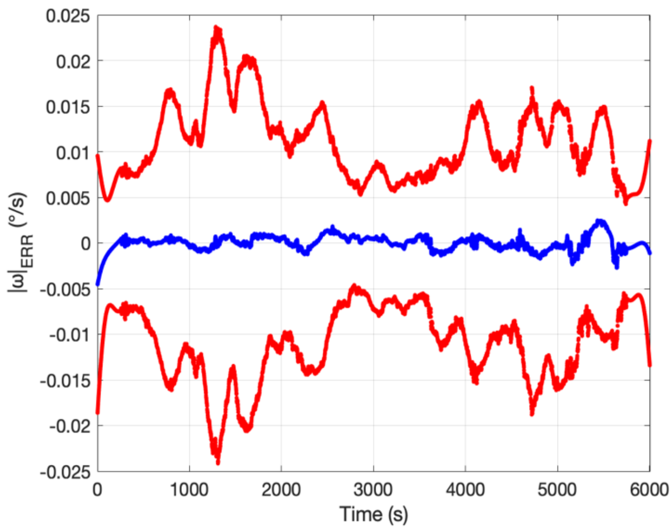


Figure 18 - Time variation of the mean (blue) and 3σ bounds (red) of the error in the target absolute angular velocity measurements after smoothing. Test Case A.

Then, the FFT is applied to the time variation of the estimated target angular velocity, and the resulting estimated polhode period is 1024 seconds (the real polhode period is 1038 seconds). This operation allows discarding absolute angular velocity measurements corresponding to already considered positions on the polhode curve. Figure 19 shows the 10 measurements selected at the end of this process to estimate the inertia properties of the target: they are widely spaced in time, thus ensuring the condition of linearly independence.

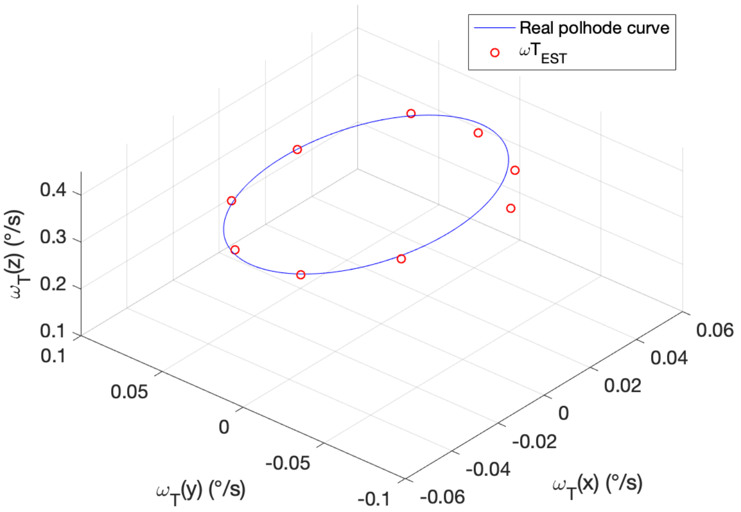


Figure 19 - Measured values of the absolute angular velocity of the target used to estimate the MIRs. Test Case A.

Thus, the system shown in Eq. (59) is solved to obtain an estimate of the MIRs of the target. The achieved results are summarized in Table 7.

Table 7 - Estimation error of the MIRs averaged over 100 simulations. Test Case A.

MIRs	True values	Estimated valued	Percentage error
$\bar{I}_{yy}$	0.9668	0.9069	6.2 %
$\bar{I}_{zz}$	0.1318	0.1230	6.7 %
$\bar{I}_{xy}$	-0.0027	0.0047	275 %
$\bar{I}_{xz}$	0.0168	0.0146	13 %
$\bar{I}_{yz}$	0.0031	0.0006	80 %

The results show that the accuracy in the off-diagonal elements of the MIRs matrix is significantly worse than the one characterizing the diagonal elements. This is justified by the fact the off-diagonal elements are two/three orders of magnitude smaller than the diagonal ones. Consequently, as it will be demonstrated in the following, such a significant error in their estimation does not have a relevant impact on the target-chaser relative state estimation accuracy when the UKF is applied.

Figure 20 shows the level of accuracy achieved by the filtering scheme described in paragraph 3.2, Here the time variation of the error metrics defined for the performance assessment is depicted. Also, the time statistics of the error metrics are averaged over 100 simulations are collected in Table 8.

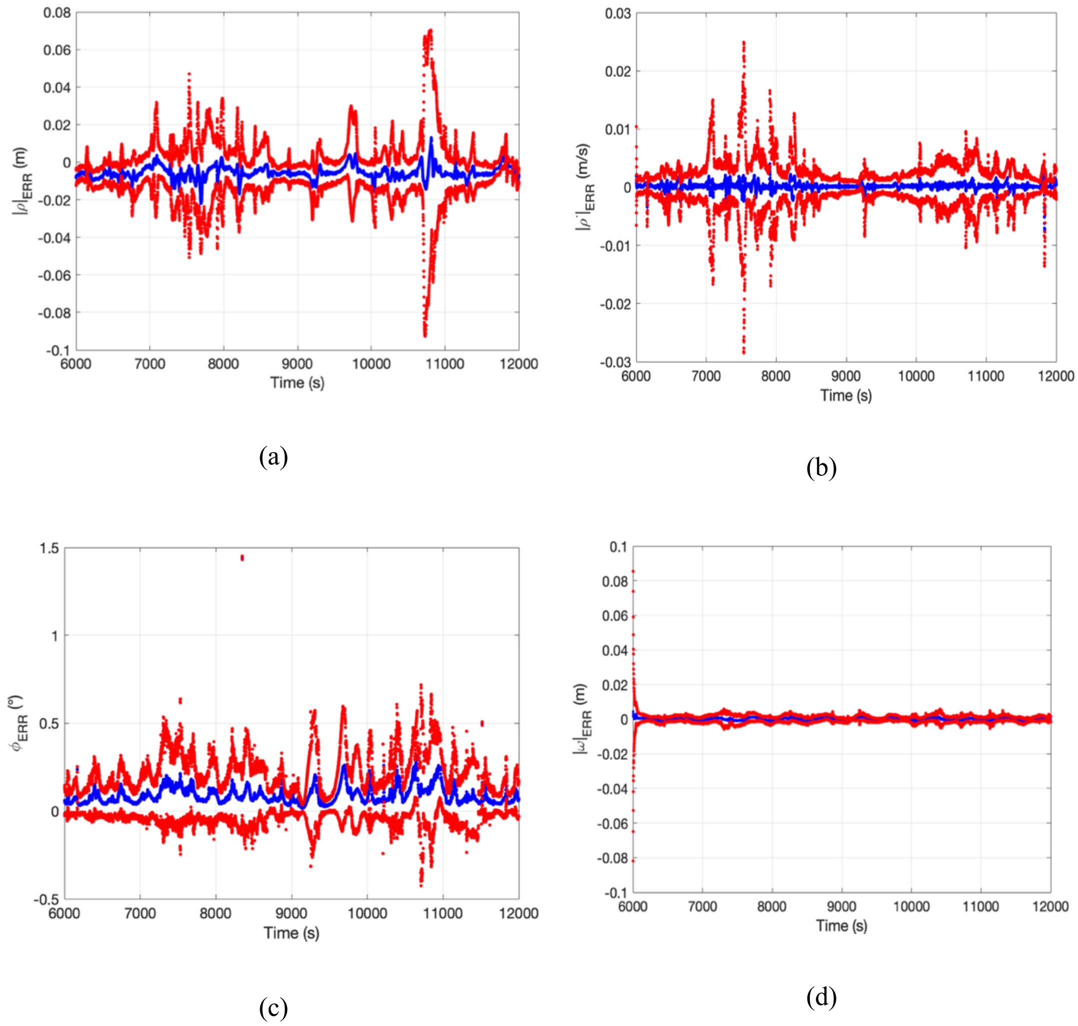


Figure 20 - Time variation of the mean value (blue) and the 3σ-bounds (red) for the error metrics averaged on 100 simulations: (a) relative position, (b) relative velocity, (c) relative attitude, (d) absolute angular velocity of the target. Test Case A.

Table 8 - Time statistics (one relative orbit, i.e., 6000 seconds) of the error metric function describing the performance of the filtering scheme. Results are averaged over 100 simulations. Test Case A.

Error metrics	**RMS**	**Maximum value**
$|\boldsymbol{\rho}|_{\mathbf{ERR}}$ (m)	$0.7 \cdot 10^{-2}$	0.1

$\lvert\dot{\boldsymbol{\rho}}\rvert_{ERR}$ (m/s)	$0.6 \cdot 10^{-3}$	$2.9 \cdot 10^{-2}$
Φ_{ERR} (°)	0.1	1.4
$\lvert\boldsymbol{\omega}\rvert_{ERR}$ (°/s)	$0.7 \cdot 10^{-3}$	$8.1 \cdot 10^{-2}$

It can be noted that the presence of the filtering scheme allows improving the estimation accuracy achieved by the pose determinations run in a stand-alone configuration. Indeed, the presence of the filter coupled with the autonomous failure detection strategy (based on the value of the ICP cost function at convergence, as described in Section 3.2.2) allows limiting the impact of unfavorable target-chaser observation geometry on the accuracy of the estimated relative motion state. By looking at the time variation of the relative attitude error during the entire considered time interval (i.e., two consecutive relative orbits) depicted in Figure 21, it can be noted how, when the filtering scheme starts, the error peaks are drastically reduced, and the global accuracy level improves. The reduction in the error level achieved by the UKF is important in view of the strict control requirements typical of proximity operations. Finally, in order to prove that the larger percentage error on the off-diagonal elements of the MIRs matrix does not have a significant impact on the estimate of the absolute angular velocity of the target, a comparison is done running the simulation using the true and estimated off-diagonal elements MIRs, respectively. The absolute angular velocity errors are compared in Table 9 showing that the use of the estimated off-diagonal MIRs produces a worsening in the estimate of the order of 10^{-5} degrees per seconds. The effect on the accuracy in the target-chaser relative state parameters estimate is negligible.

Table 9 - RMS of $\lvert\boldsymbol{\omega}\rvert_{ERR}$ estimated with the real and estimated off-diagonal elements of the MIRs matrix. Test Case A.

RMS

	True off-diagonal MIRs	Estimated off-diagonal MIRs
$\lvert\boldsymbol{\omega}\rvert_{ERR}$ (°/s)	10^{-4} °/s	$1.3 \cdot 10^{-4}$ °/s

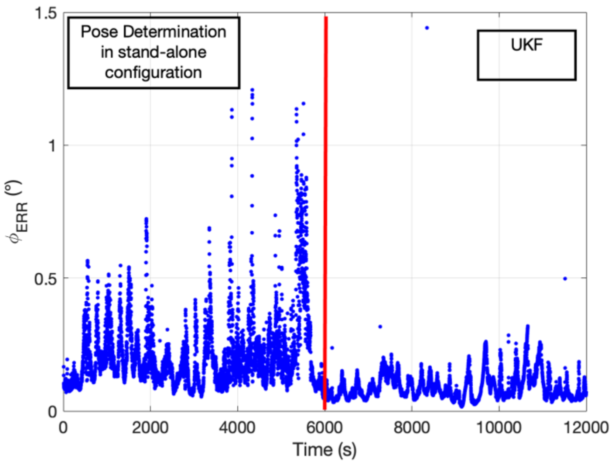


Figure 21 - Time variation of Φ_{ERR} averaged over 100 simulations. Test Case A.

4.3.2 Test Case B

As second test case, a different target geometry has been considered. This geometry represents a typical satellite configuration with a main body to which three appendages, two solar array and a SAR antenna are attached; also, the assigned dimensions of the target model used to evaluate the performance of the relative navigation architecture are comparable to a large amount of the LEO satellites. Figure 22 shows the target used to evaluate the performance of the relative navigation architecture. The target geometry is inspired by a constellation of Earth observation satellites, i.e., COSMO Sky-Med (CSM) [59], [80]. The reflection coefficients of the surface are the same considered in the Test Case A, while the geometrical characteristics and inertia properties are listed in Table 10.

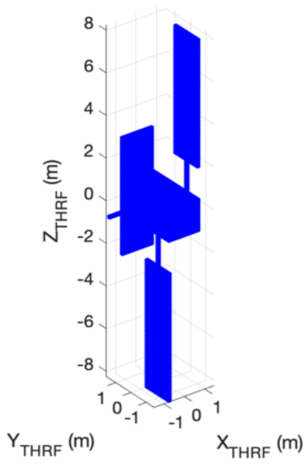


Figure 22 – Point cloud representation of the simplified geometric model for Test Case B.

Table 10 - Test Case B: geometrical and inertia characteristics of the target.

Element	Dimension	Moment of inertia (kg·m²)			
Main body	1.5 m x 3 m x 1.5 m	I_{xx}	21087.34	I_{xy}	10.37
Solar panel	1.4 m x 6.56 m	I_{yy}	31600.95	I_{xz}	18.79
SAR antenna	1.4 m x 5.7 m	I_{zz}	13826.53	I_{yz}	-121.82

As regards the simulated scenario, the trajectory followed by the chaser aroud the target is shown in Figure 23. The minumum and the maximum distances of CRF from the center of mass of the target are 13 and 21 meters, respectively, while the rotational dynamics of the target is described the the polhode depicted in Figure 24. It can be noted that the three components of the absolute angular velocity in TRF are harmonics functions of the time.

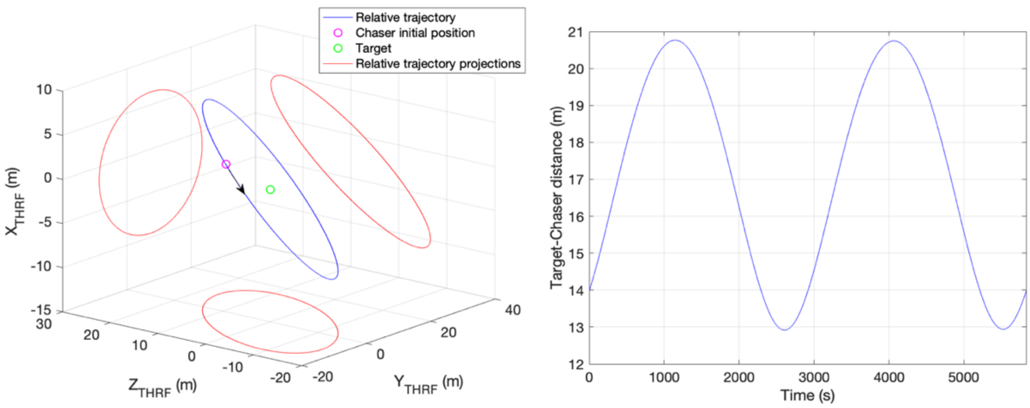


Figure 23 – Relative trajectory around the target. Test Case B. (left) Safety ellipse for monitoring purpose. (right) time variation of the target-chaser distance.

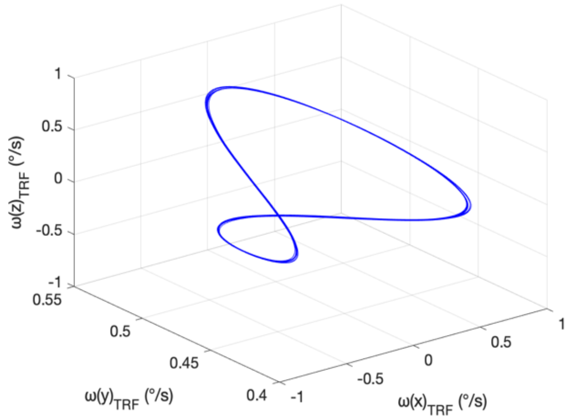


Figure 24 - Polhode of the target. Test Case B.

The results of the tracking phase are summarized in Table 11 and Table 12. In the Test Case B, the pose determination algorithm in a stand-alone configuration guarantees a millimeter level and sub-degree level of accuracy in the estimation of the relative position and attitude, respectively. Then, the accuracy of $2.8 \cdot 10^{-3}$ degrees per seconds is achieved in the target absolute angular velocity measurements by applying the Saitzky-Golay smoother. This level of accuracy allows estimating the main diagonal elemtns of the MIRs matrix with an error lower than 2%, as shown in Table 12.

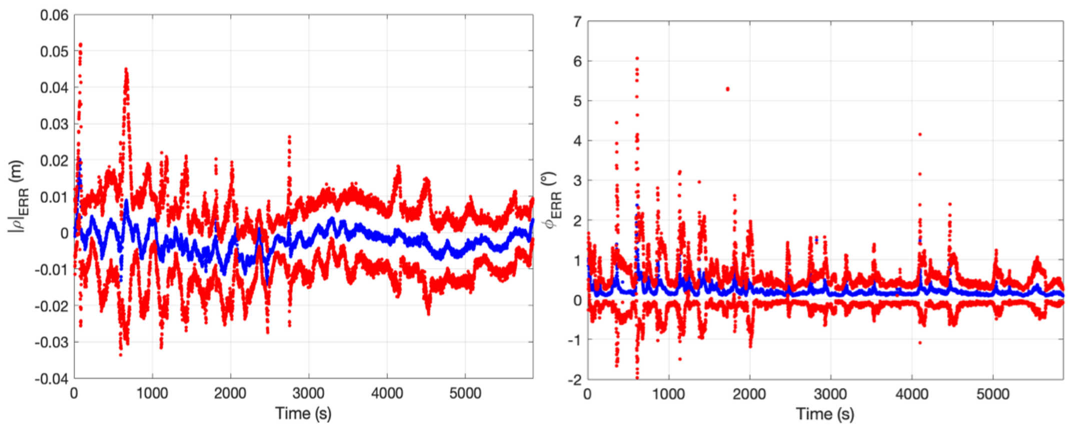


Figure 25 - Time variation of the mean value (blue) and the 3σ-bounds (red) for the error metrics averaged over 100 simulations: (left) relative position, (right) relative attitude. Test Case B.

Table 11 - Time statistics of the error metric functions describing the performance of the pose tracking phase averaged over 100 simulations. Test Case B.

Error metrics	RMS	Maximum value
$\|\rho\|_{ERR}$ (m)	$0.4 \cdot 10^{-2}$	0.06
Φ_{ERR} (°)	0.3	6.1

Table 12 - Estimation error if the diagonal MIRs averaged over 100 simulations. Test Case B.

Error metrics	% Estimation Error
$\bar{I}_{yy}$	0.8 %
$\bar{I}_{zz}$	1.9 %

Also, the time histories of the relative state estimation error achieved in the final step of the relative navigation architecture by UKF are depicted in Figure 26, while their statistics are summarized in Table 13.

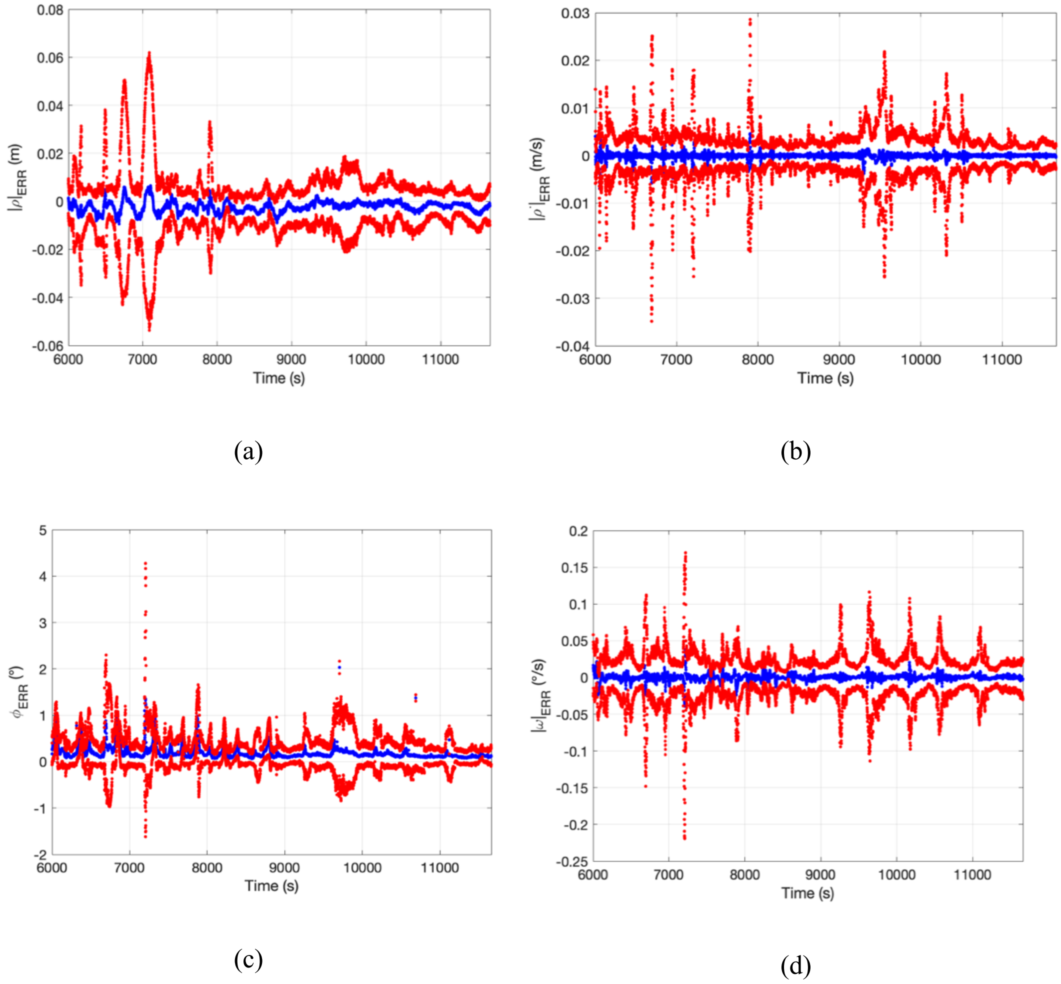


Figure 26 - Time variation of the mean value (blue) and the 3σ-bounds (red) for the error metrics averaged on 100 simulations: (a) relative position, (b) relative velocity, (c) relative attitude, (d) absolute angular velocity of the target. Test Case B.

Table 13 - Time statistics (one relative orbit, i.e., 6000 seconds) of the error metric function describing the performance of the filtering scheme. Results are averaged over 100 simulations. Test Case B.

Error metrics	RMS	Maximum value
$\lvert\boldsymbol{\rho}\rvert_{ERR}$ (m)	$0.3 \cdot 10^{-2}$	$8.8 \cdot 10^{-2}$
$\lvert\dot{\boldsymbol{\rho}}\rvert_{ERR}$ (m/s)	$0.7 \cdot 10^{-4}$	$3.4 \cdot 10^{-2}$
Φ_{ERR} (°)	0.2	5.1
$\lvert\boldsymbol{\omega}\rvert_{ERR}$ (°/s)	$4.6 \cdot 10^{-3}$	$4.1 \cdot 10^{-3}$

4.3.3 Test Case C

In this test-case, an example of abandoned rocket body, i.e., a KOSMOS 3M second stage, has been modelled as a 6 meters length cylinder with a radius of 1.2 meter based on the data found in the open literature [81], as shown in Figure 27. This test case is used to evaluate the performance of the developed architecture when dealing with a completely symmetric target, which may introduce ambiguity when trying to determine the relative pose. As regards the physical properties of the target, a constant reflection coefficient of 0.4 has been assumed, while the inertia matrix is shown in Eq. (91).

$$I = \begin{bmatrix} 4200 & 0 & 0 \\ 0 & 4200 & 0 \\ 0 & 0 & 1238 \end{bmatrix} kg\, m^2 \tag{91}$$

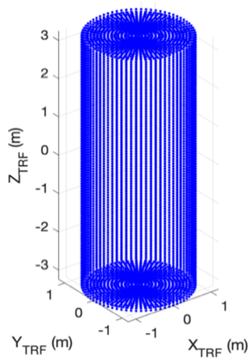


Figure 27 - Point cloud representation of the simplified geometric model for the rocket body.

The designed relative trajectory, shown in Figure 28, allows the chaser to monitor the target from a distance that varies from 10 to 17 meters, while the rocket body is tumbling with an angular velocity of 1 degree per second around the along-track direction[82], as described by the polhode of Figure 29.

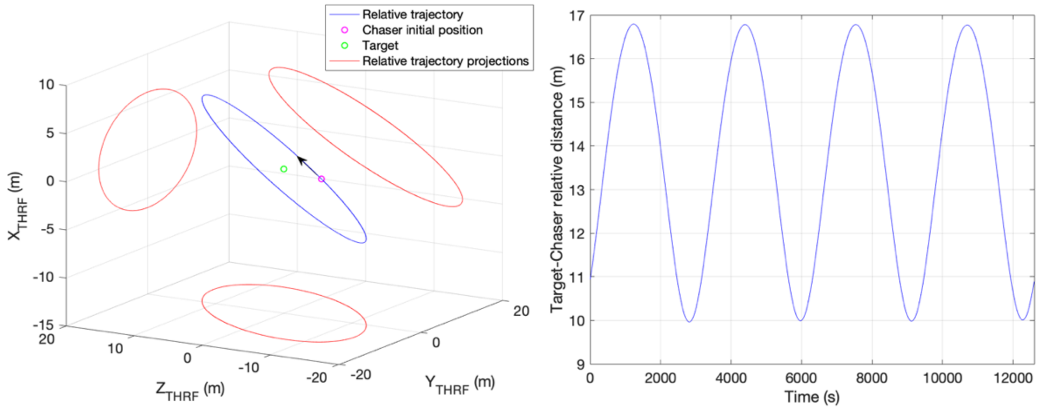


Figure 28 – Relative trajectory around the target. Test Case C. (left) Safety ellipse for monitoring purpose. (right) time variation of the target-chaser distance.

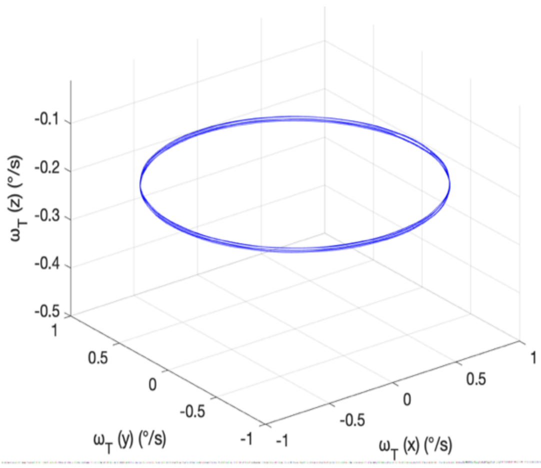


Figure 29 - Polhode of the target. Test Case C.

The performance achieved in the pose tracking phase are shown in Figure 30 and summarized in Table 14. The relative position and attitude are estimated by the model-based pose determination algorithm, despite the presence of some peaks between 1500-2500 seconds and 4000-5000 seconds, with a millimeter and sub-degree level of accuracy.

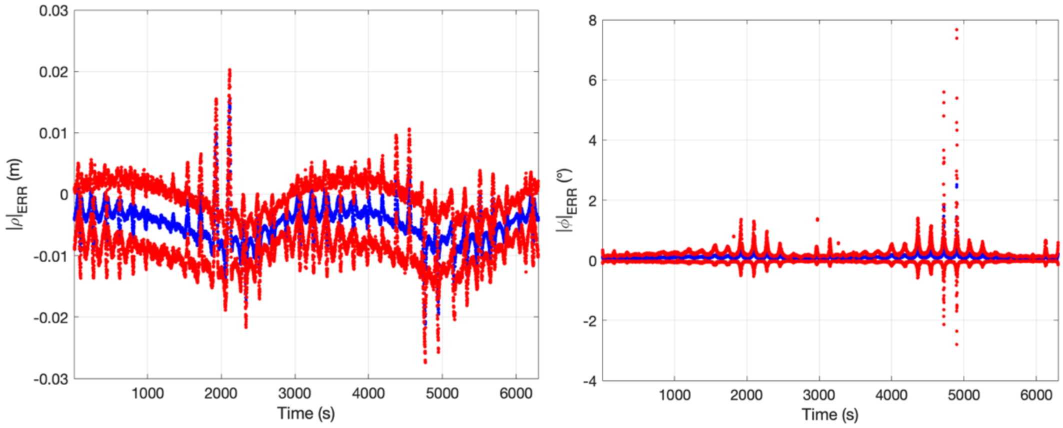
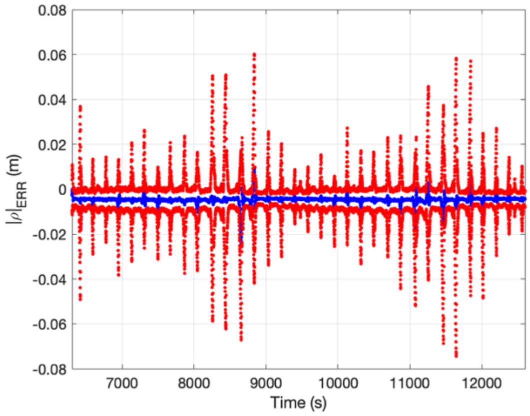


Figure 30 - Time variation of the mean value (blue) and the 3σ-bounds (red) for the error metrics averaged over 100 simulations: (left) relative position, (right) relative attitude. Test Case C.

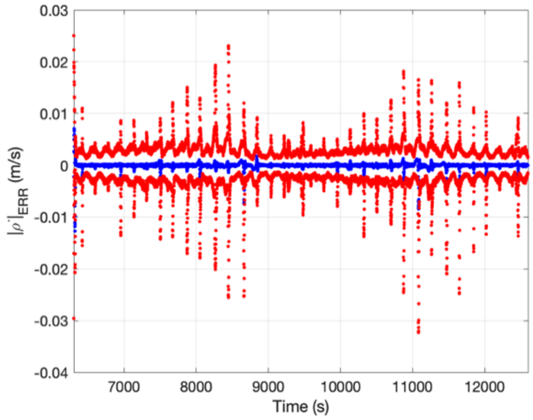
Table 14 - Time statistics of the error metric functions describing the performance of the pose tracking phase averaged over 100 simulations. Test Case C.

Error metrics	RMS	Maximum value
$\|\rho\|_{ERR}$ (m)	$0.6 \cdot 10^{-2}$	$2.7 \cdot 10^{-2}$
Φ_{ERR} (°)	0.1	8.7

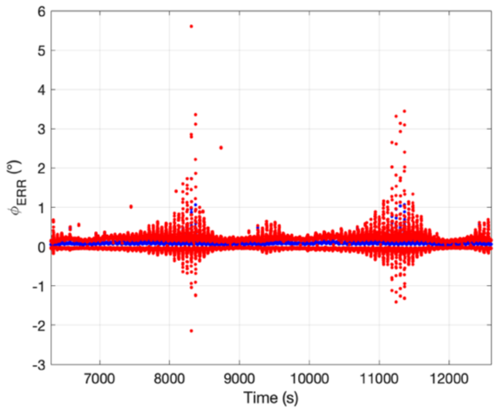
Also, a very accurate estimation of the absolute angular velocity of the target (a mean error of 10^{-4} degrees per seconds is achieved by applying the Savitzky-Golay smoother), allows estimating the diagonal elements of the MIRs matrix with an error of 0.07% and 0.84%, respectively. Finally, the time histories of the error metrics function selected for the performance assessment of the filtering scheme are shown in Figure 31. It is worth noting that the figure referring to the absolute angular velocity error shows a larger initialization error which is quickly reduced by reaching a mean value of 10^{-4} degrees per seconds, as shown in Table 15, where the statistics of the error metrics averaged over 100 simulations are summarized.



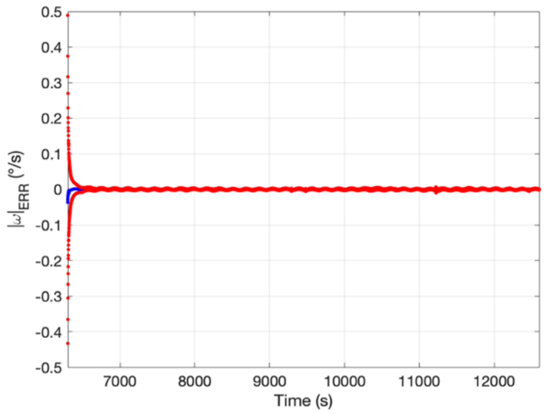
(a)



(b)



(c)



(d)

Figure 31 - Time variation of the mean value (blue) and the 3σ-bounds (red) for the error metrics averaged on 100 simulations: (a) relative position, (b) relative velocity, (c) relative attitude, (d) absolute angular velocity of the target. Test Case C.

Table 15 - Time statistics (one relative orbit, i.e., 6000 seconds) of the error metric function describing the performance of the filtering scheme. Results are averaged over 100 simulations. Test Case C.

Error metrics	**RMS**	**Maximum value**
$\|\boldsymbol{\rho}\|_{ERR}$ (m)	$0.5 \cdot 10^{-2}$	$4.9 \cdot 10^{-2}$
$\|\dot{\boldsymbol{\rho}}\|_{ERR}$ (m/s)	$0.7 \cdot 10^{-3}$	$3.1 \cdot 10^{-2}$
Φ_{ERR} (°)	0.08	4.8
$\|\boldsymbol{\omega}\|_{ERR}$ (°/s)	$1.5 \cdot 10^{-3}$	0.4

4.3.4 Test Case D

In this case, a geostationary spacecraft has been considered as target. Figure 32 shows the target model used in the model-based pose determination algorithm to retrieve pose information from the 3D point-clouds supplied by the LIDAR system. The model is generated considering the physical and geometrical properties of HISPASAT 36W-1 satellite (see Table 16), developed by OHB System AG (Germany) and the European Space Agency for the smallGEO programme [83]–[85].

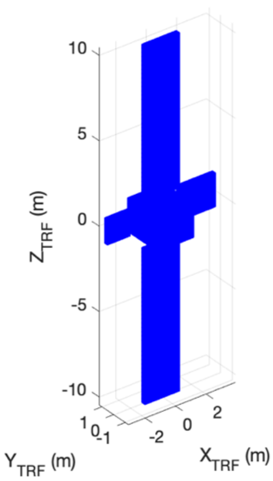
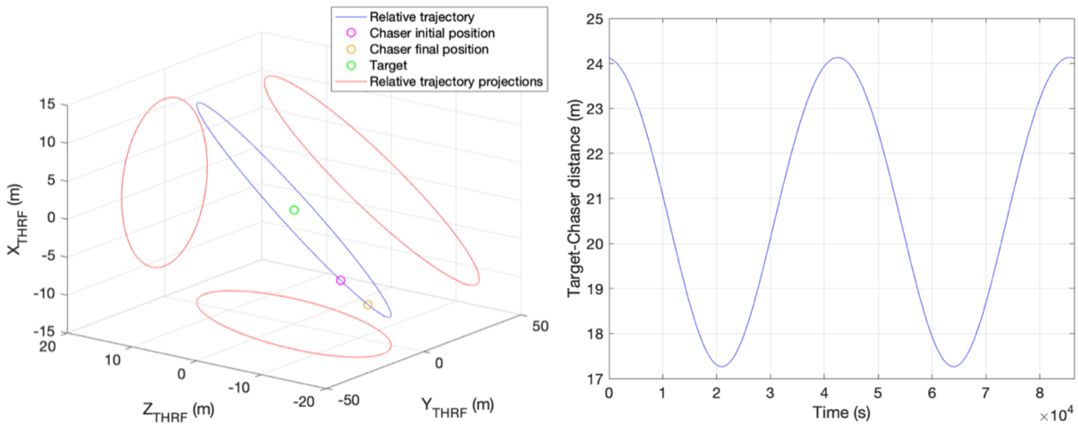


Figure 32 - Point cloud representation of the simplified geometric model for the GEO target.

Table 16 - Test Case C: geometrical and inertia characteristics of the target.

Element	Dimension	Moment of inertia ($kg{\cdot}m^2$)	
Main body	2.5 m x 1.9 m x 1.3 m	I_{xx}	21087.34
Solar panel	0.2 m x 9 m x 2.2 m	I_{yy}	31600.95
Antennas	0.2 m x 1.9 m x 2.3 m 0.15 m x 1.425 m x 1.425 m	I_{zz}	13826.53

Figure 33 shows the designed safety-ellipse for monitoring purpose around the target, but due to the high orbital period of the geostationary satellites, the presented results refer only to a shorter, but relevant time interval. The chaser initial and final position are shown in Figure 33



F
Figure 33 – Relative trajectory around the target. Test Case D. (left) Safety ellipse for monitoring purpose. (right) time variation of the target-chaser distance.

The rotational dynamics of the target is described by the polhode depicted in Figure 34: it tumbles with an angular velocity of 0.6 degrees per seconds and the three vectorial components show a harmonic behavior.

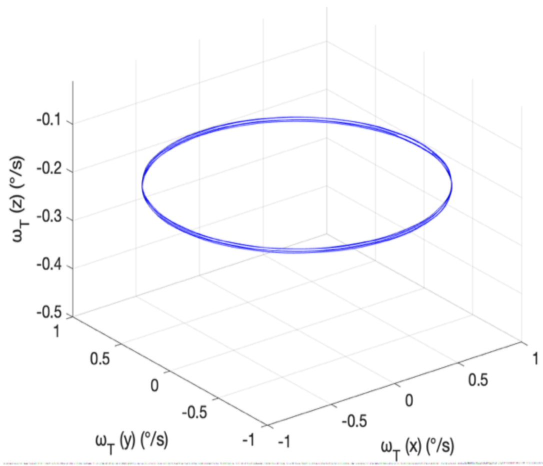


Figure 34 - Polhode of the target. Test Case C.

Figure 35 and Table 17 summarize the performance achieved by the relative navigation architecture in the first step. It can be noted how the initialization error is slightly larger than the one in the previous test cases, but the error quickly converges, and similar level of accuracies is reached. Specifically, the relative position is estimated with a millimeter accuracy, while the relative attitude with a mean error of 0.2°. Also, by applying the smoother to the time-history of the absolute angular velocity of the target measured through the quaternion differentiation technique, the mean error is reduced from 10^{-3} to 10^{-4} degrees per seconds. This level of accuracy allows estimating the diagonal elements of the MIRs matrix with an error of 1% and 0.6%, respectively.

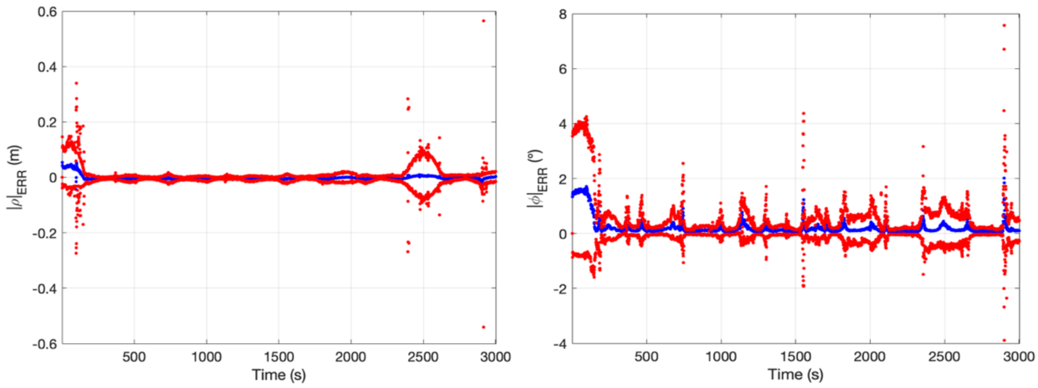
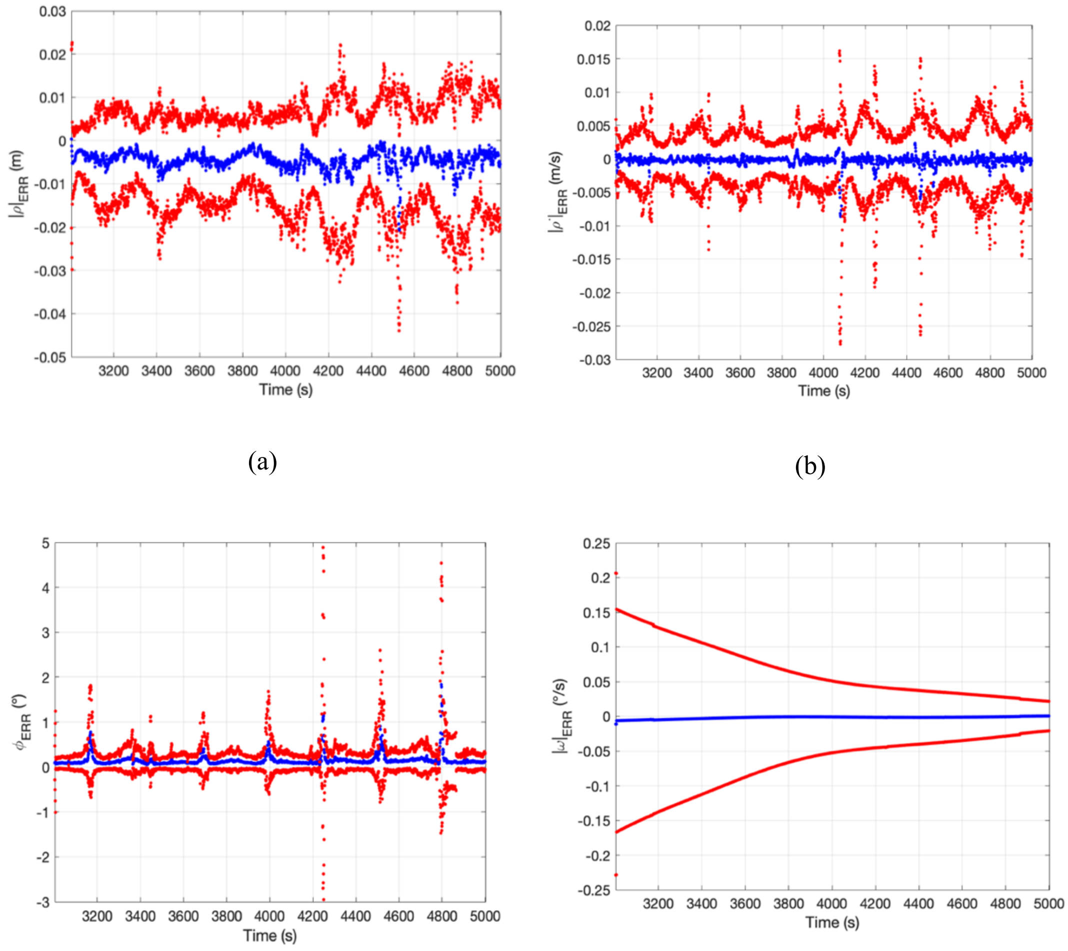


Figure 35 - Time variation of the mean value (blue) and the 3σ-bounds (red) for the error metrics averaged over 100 simulations: (left) relative position, (right) relative attitude. Test Case D.

Table 17 - Time statistics of the error metric functions describing the performance of the pose tracking phase averaged over 100 simulations. Test Case D.

Error metrics	**RMS**	**Maximum value**
$\|\rho\|_{ERR}$ (m)	$0.8 \cdot 10^{-2}$	1.8
Φ_{ERR} (°)	0.4	8.7

The performance achieved by the UKF when dealing with a geostationary satellite are described by Figure 36, where the time-histories of the mean and 3σ bound of the error metrics defined for the performance assessment of the relative navigation filter are shown, and by Table 18, where the related time statistics (averaged over 100 simulations) are collected.



(a) (b)

(c) (d)

Figure 36 - Time variation of the mean value (blue) and the 3σ-bounds (red) for the error metrics averaged on 100 simulations: (a) relative position, (b) relative velocity, (c) relative attitude, (d) absolute angular velocity of the target. Test Case D.

Table 18 - Time of the error metric function describing the performance of the filtering scheme. Results are averaged over 100 simulations. Test Case D.

Error metrics	RMS	Maximum value
$\|\boldsymbol{\rho}\|_{ERR}$ (m)	$0.5 \cdot 10^{-2}$	$7.7 \cdot 10^{-2}$
$\|\dot{\boldsymbol{\rho}}\|_{ERR}$ (m/s)	$0.8 \cdot 10^{-3}$	$2.8 \cdot 10^{-2}$
Φ_{ERR} (°)	0.2	7.8
$\|\boldsymbol{\omega}\|_{ERR}$ (°/s)	$2.5 \cdot 10^{-3}$	0.4

4.4 Relative Navigation for close-approach phase: numerical results

This paragraph describes the simulation scenario and reports the performance verification of the relative navigation architecture for the close approach phase described in paragraph 3.3. Both the UKF and MEKF has been adopted as relative navigation filter. However, once verified that in the close approach phase they provide the same accuracy level, for the sake of brevity, the results corresponding to the MEKF has been reported in the following.

4.4.1 Simulation scenario

The target selected for this scenario is ENVISAT. With respect to the test case presented in paragraph 4.3.1, a more detailed model of the spacecraft, shown in Figure 37, has been adopted. In this case the ray-tracing algorithm of the OPCODE library [86], which exploits the representation of the target geometry with an AA-BB-tree (Axis Aligned Bounding Box-tree) data structure, to generate the ideal point clouds in the *LIDAR measurement* simulator.

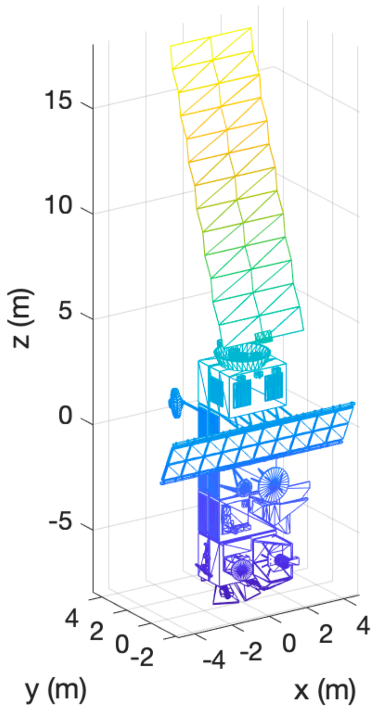


Figure 37 - Detailed ENVISAT geometry.

In the simulated scenario, the chaser approaches the target along the R-bar direction starting from a 10 meters distance between TRF and CRF, as shown in Figure 38.

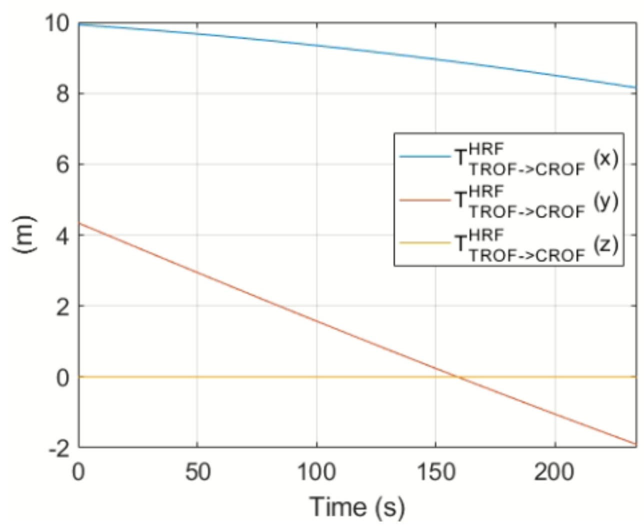


Figure 38 - Time variation of the target-chaser relative distance.

As regards the rotational dynamics, it has been assumed that the target has a tumbling motion around the *z* axis of TRF, which in turn, is aligned with the negative direction of the *x*-axis of HRF. The time variation of the 3-1-3 sequence of Euler angles parametrizing the inertial attitude of TRF and the target absolute angular velocity are shown in Figure 39.

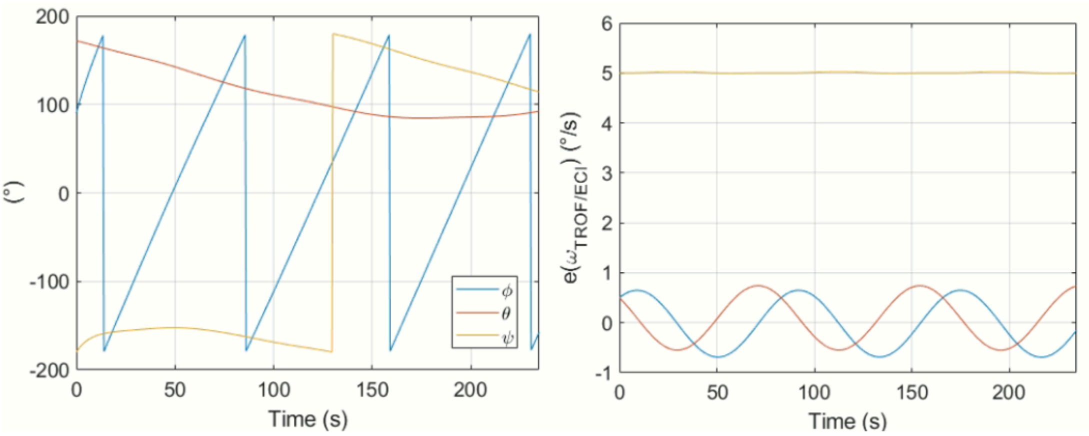


Figure 39 - (left) Time variation of 313 sequence of Euler angles representing the inertial attitude of TRF. (right) Time variation of the target angular velocity.

The chaser is equipped with a 7-DOF robotic arm whose geometry is described by the modified Denavit-Hartenberg parameters, listed in Figure 40. Also, a scheme showing the definition of the adopted convention for the reference frame attached to each link is reported in Table 19. All the joints are purely rotational ones.

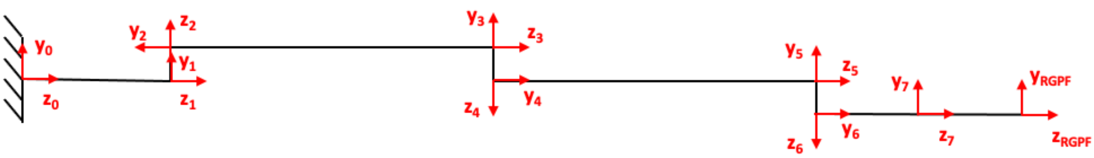


Figure 40 - Geometry of the robotic arm.

Table 19 - Modified DH parameters for the robotic arm.

	a (m)	α (°)	d (m)
Joint 1	0	0	0.300
Joint 2	0	-90	0.160
Joint 3	0	90	1.150
Joint 4	0	90	0.160
Joint 5	0	-90	1.150

Joint 6	0	90	0.160
Joint 7	0	-90	0.200
Joint 8	0	0	0.200

The trajectory of the end-effector with respect to the grasping point (which is placed on the launch adapter ring (LAR)) has been defined starting from the stowed configuration of the robotic arm shown in Figure 41.

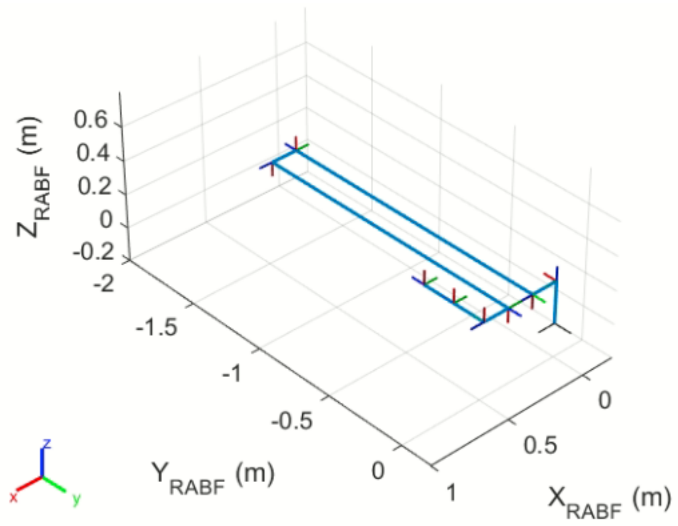


Figure 41 - Stowed configuration of the robotic arm.

The only condition that must be satisfied by the trajectory of the end-effector is that part of the target surface is imaged in the scan window of the TOF camera. The end effector-grasping point relative trajectory, obtained as described in 4.1.1, is described by the time variation of the pose parameters shown in Figure 42.

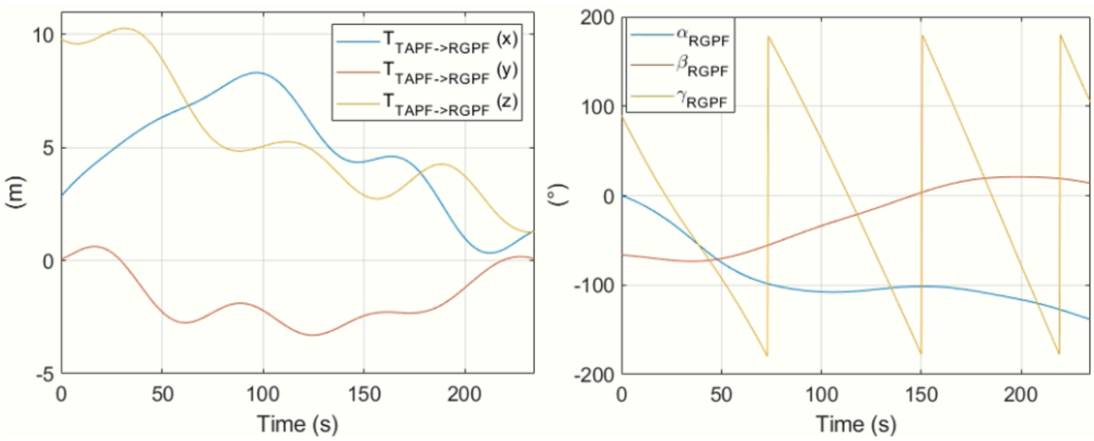


Figure 42 - (left) Time variation of grasping point-end effector relative position. (right) Time variation of Euler angles (3-2-1 sequence) representing the attitude of RGPF with respect to TAPF.

Considering the relative distance involved, both the specifications of the LIDAR mounted on the body of the spacecraft and the TOF camera on the end-effector of the robotic arm, listed in Table 20, have been selected based on coverage constraints:

- The target shall be fully in view in the operative range of both monitoring and close approach phase.
- The LAR shall be fully in view at the start of the close approach phase.

Table 20 - Sensors' operational and noise parameters.

Sensor	*Sensor format*	*FOV*	*IFOV*	*Range uncertainty (1σ)*
Body-mounted Flash LIDAR	128x128	Up to 45°x45°	0.35°x0.35° (514 points/m^2)	2 cm
TOF camera	640x480	57°x43°	0.09°x0.09° (8270 points/m^2)	3 cm

Also, error on the knowledge of the initial sate has been modelled as a Gaussian White Noise whose standard deviation is listed in Table 21. The error on the chaser absolute state parameters have been modelled as described in paragraph 4.1.3, whit a standard deviation as shown Table 22, while a σ equal to 10^{-5} radiant and 10^{-6} radiant per seconds has been set for the joint's rotation and velocity errors, respectively, considering typical performance of incremental encoders.

Table 21 - Uncertainty in the knowledge of the initial state.

	Relative position (m)	Relative velocity (m/s)	Relative attitude (°)	Angular velocity (°/s)
$\boldsymbol{\sigma}$	0.10 (along-boresight direction) 0.033 (cross-boresight direction)	0.0033	1°	0.066

Table 22 - Uncertainty on the knowledge of the inertial state navigation of the chaser.

	Position (m)	Relative velocity (m/s)	Relative attitude (°)	Angular velocity (°/s)
$\boldsymbol{\sigma}$	0.8	0.001	0.0017	0.0117

4.4.2 Numerical results

A 100-run Montecarlo simulation has been performed to verify the robustness of the relative navigation architecture against the uncertainty on the initial conditions. The statistics of the error metrics describing the achieved accuracy in the target-chaser and gripper-grasping point relative estimation task averaged over 100 simulations are summarized in Table 23 and Table 24.

Table 23 - Statistics of the error metric functions for the target-chaser relative state estimation task. Results are averaged over 100 simulations.

Error metrics	Unit	Mean	Standard deviation
$\boldsymbol{e}(\boldsymbol{t}_{TROF \to CROF}^{HRF}(x))$	cm	-0.02	0.08
$\boldsymbol{e}(\boldsymbol{t}_{TROF \to CROF}^{HRF}(y))$	cm	0.01	0.2
$\boldsymbol{e}(\boldsymbol{t}_{TROF \to CROF}^{HRF}(z))$	cm	0.03	0.1
$\boldsymbol{e}(\boldsymbol{v}_{TROF/CROF}^{HRF}(x))$	mm/s	-1.19×10^{-2}	0.3
$\boldsymbol{e}(\boldsymbol{v}_{TROF/CROF}^{HRF}(y))$	mm/s	7.29×10^{-3}	0.5
$\boldsymbol{e}(\boldsymbol{v}_{TROF/CROF}^{HRF}(z))$	mm/s	-1.39×10^{-3}	0.4
$e_{xdirTROF}$	°	0.03	0.01
$e_{ydirTROF}$	°	0.03	0.01

$e_{zdirTROF}$	°	0.02	0.01
$\boldsymbol{e}(\boldsymbol{\omega}_{TROF/ECI}(x))$	°/s	1.42×10^{-4}	1.60×10^{-3}
$\boldsymbol{e}(\boldsymbol{\omega}_{TROF/ECI}(y))$	°/s	-1.91×10^{-4}	1.60×10^{-3}
$\boldsymbol{e}(\boldsymbol{\omega}_{TROF/ECI}(z))$	°/s	-1.21×10^{-4}	5.00×10^{-4}

Table 24 - Statistics of the error metric functions for the gripper-grasping point relative state estimation task. Results are averaged over 100 simulations.

Error metrics	Unit	Mean	Standard deviation
$\boldsymbol{e}(\boldsymbol{t}_{RGPF \to TAPF}(x))$	cm	-0.03	0.2
$\boldsymbol{e}(\boldsymbol{t}_{RGPF \to TAPF}(y))$	cm	-0.04	0.2
$\boldsymbol{e}(\boldsymbol{t}_{RGPF \to TAPF}(z))$	cm	-0.08	0.1
$\boldsymbol{e}(\boldsymbol{v}_{TAPF/RGPF}(x))$	mm/s	11.1	4.0
$\boldsymbol{e}(\boldsymbol{v}_{TAPF/RGPF}(y))$	mm/s	0.64	5.4
$\boldsymbol{e}(\boldsymbol{v}_{TAPF/RGPF}(z))$	mm/s	-11.5	2.4
$e_{xdirTAPF}$	°	0.03	0.01
$e_{ydirTAPF}$	°	0.02	9.4×10^{-3}
$e_{zdirTAPF}$	°	0.03	0.01
$\boldsymbol{e}(\boldsymbol{\omega}_{RGPF/TAPF}(x))$	°/s	-5.09×10^{-4}	2.0×10^{-3}
$\boldsymbol{e}(\boldsymbol{\omega}_{RGPF/TAPF}(y))$	°/s	4.54×10^{-4}	1.7×10^{-3}
$\boldsymbol{e}(\boldsymbol{\omega}_{RGPF/TAPF}(z))$	°/s	-7.18×10^{-4}	2.7×10^{-3}

Statistics of the selected error metrics show that the target chaser relative position and velocity are estimated with a millimeter and sub-millimeter per second level of accuracy, respectively. Instead, the target-chaser relative attitude is estimated with a sub-degree level of accuracy and the target absolute angular velocity with an error level of 10^{-3} degrees per seconds. As regards the gripper-grasping point relative state estimation, the relative position and relative velocity are estimated with millimeter and centimeter per second level of accuracy, respectively; while the error level achieved in the relative attitude and relative angular velocity estimate are 10^{-2} degree and 10^{-3} degree per second. Also, for the sake of completeness, the time variation of the estimation error of all the relative motion parameters for a particular realization of the uncertainty on the knowledge of the initial state is reported in Figure 43 and Figure 44. Specifically, it can be noted that the gripper-grasping point relative navigation filter allows avoiding the worsening in the relative position estimation observed in ff, which correspond to the position and attitude estimation errors of the pose determination algorithm in a stand-alone configuration, thanks to the measurement provided by the forward kinematic model of the robotic arm.

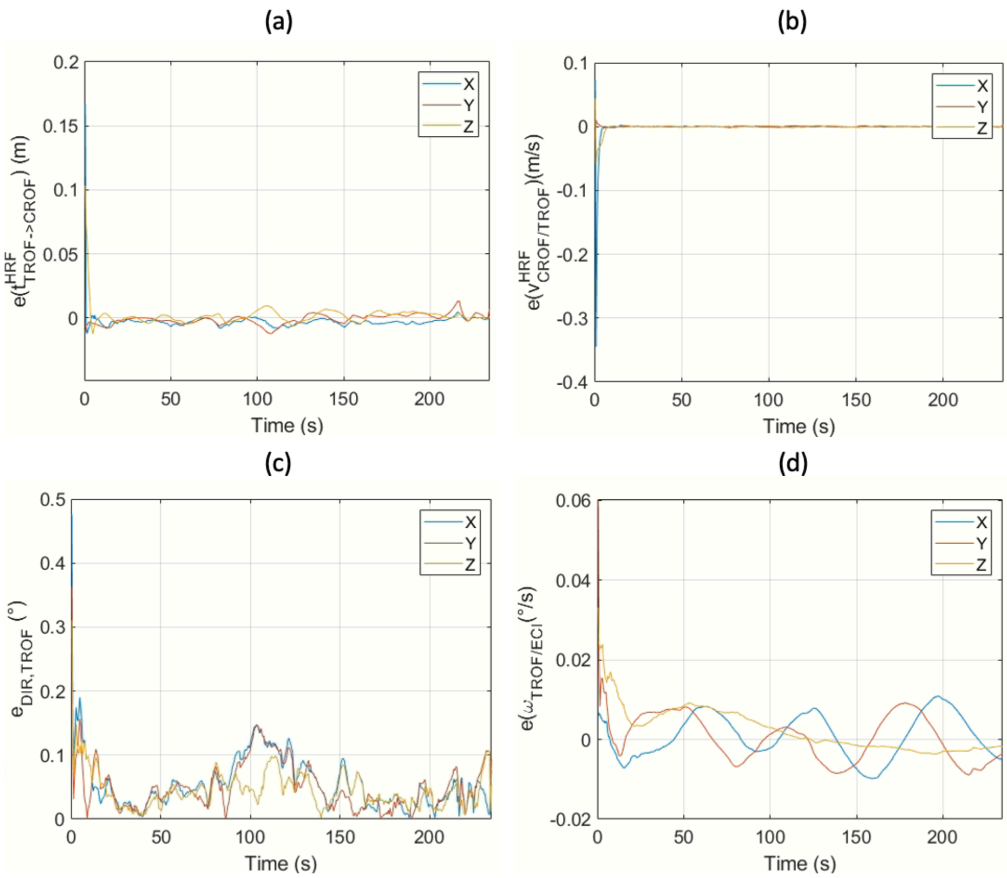


Figure 43 - Time variation of the error metrics defined for the target-chaser relative state estimation task.

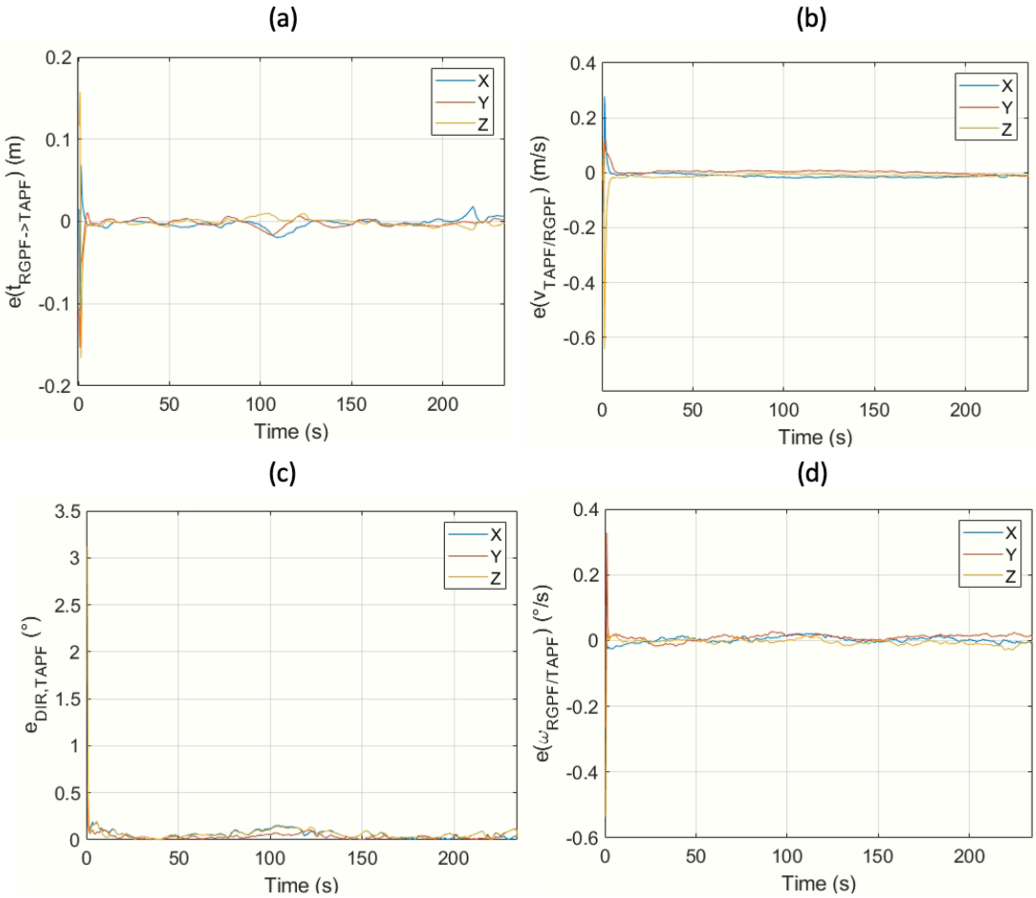


Figure 44 - Time variation of the error metrics defined for the gripper-grasping point relative state estimation task.

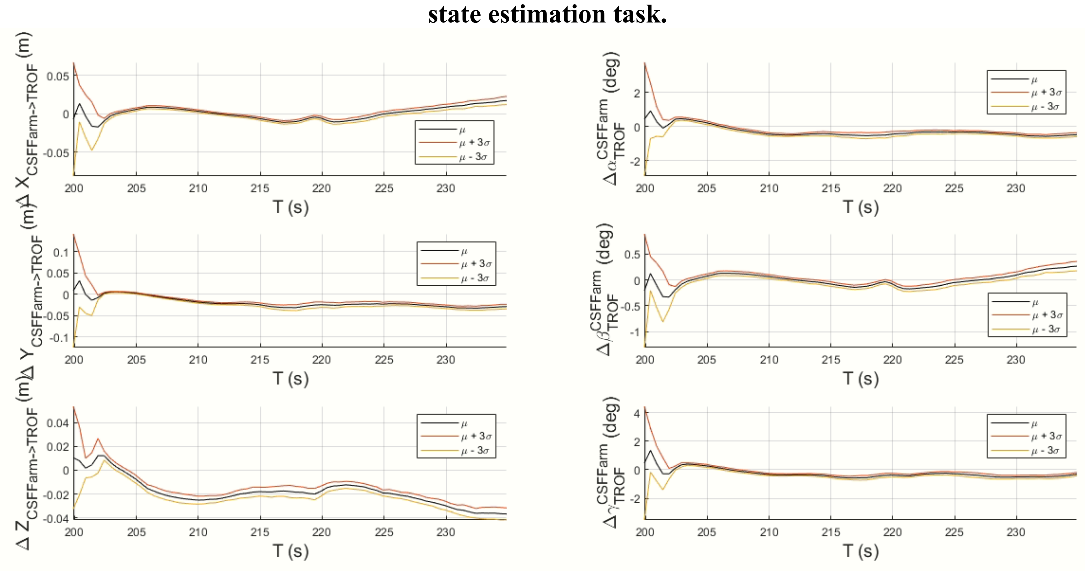


Figure 45 - Time variation of the mean value and the 3σ-bounds for the error metrics averaged on 100 simulations of the pose determination algorithm in stand-alone configuration.

5. Experimental Validation for Inertia Parameters Estimation

This chapter presents the experimental activity conducted at the ADAMUS laboratory of Embry Riddle Aeronautical University (FL, USA) aimed at assessing the performance of techniques for inertia and attitude parameters estimation of an uncooperative but known space target. The adopted experimental set-up along with the experimental results are described in the next sub-chapter, while a list of the adopted reference frame is reported in the following:

- The LIDAR reference frame (LRF) is a sensor-fixed coordinate system with the origin in the sensor optical center, the z_{LRF} axis pointing along the boresight direction and the x_{LRF} and y_{LRF} axis laying on the image plane. It is worth noting that in this case, the LRF can be considered an inertial reference frame, being fixed with respect to the ground.
- The target body fixed reference frame (BRF) is a target fixed coordinate system.
- The Motion Tracking Reference Frame (MTRF) is the one in which the motion tracking system outputs its measurements.

5.1 Experimental set-up

A schematic representation of the experimental set-up is provided in Figure 46.

The spherical air bearing provides the rotational degrees of freedom: it guarantees full 360° rotation around the vertical axis; however, the rotation around the axes parallel to the floor is limited by the presence of structural elements. The air used by the bearing is stored in two 4500 psi paintball tanks and it is stepped down to a 100 psi by pressure regulators. Then, the air flows through the line which connect the tanks to the spherical cup air bearing supporting the spherical segment ball.

A 3D-printed scaled-down satellite mock-up made of Acrilonitrile-Butadiene-Stirene (ABS) is attached to the bearing.

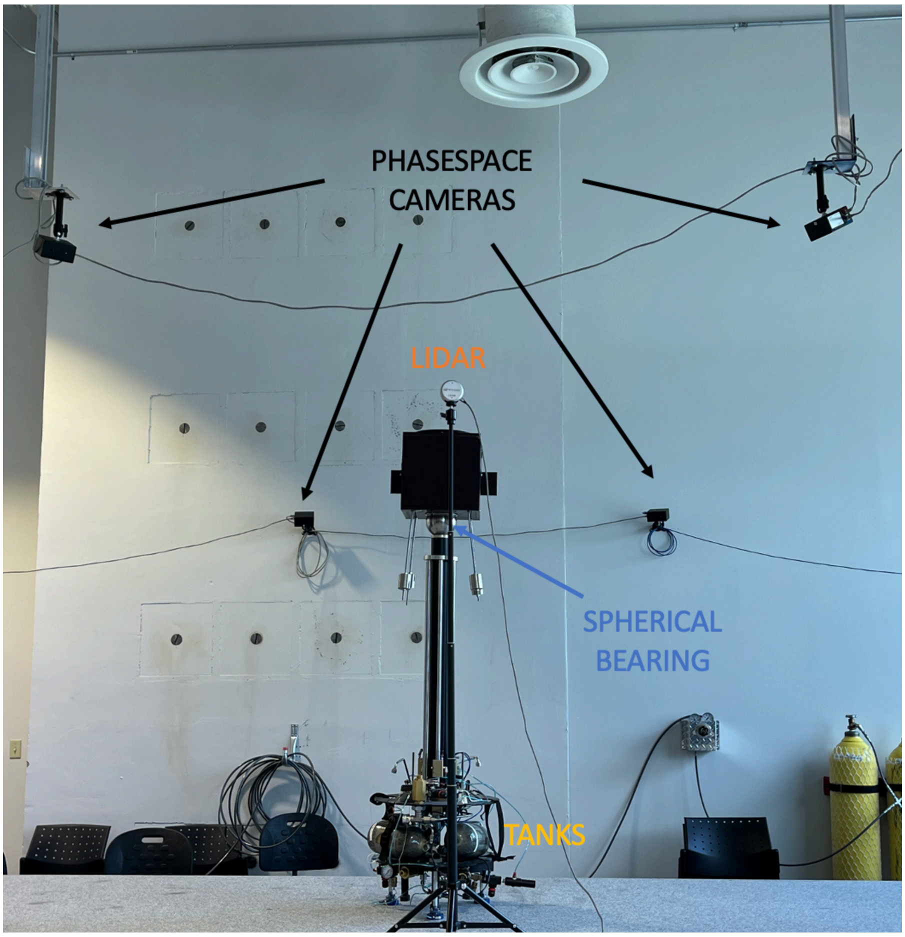


Figure 46 - Experimental set-up.

A counterbalancing system is used to minimize the gravity torque by the misalignment between the center of mass of the system and the center of rotation (which coincides with the center of the spherical bearing). It is a passive system composed of four arms symmetrically extending downwards the 3D printed model to whose ends static weights are placed. The details of both the mock-up and the counterbalancing system are shown in Figure 47.

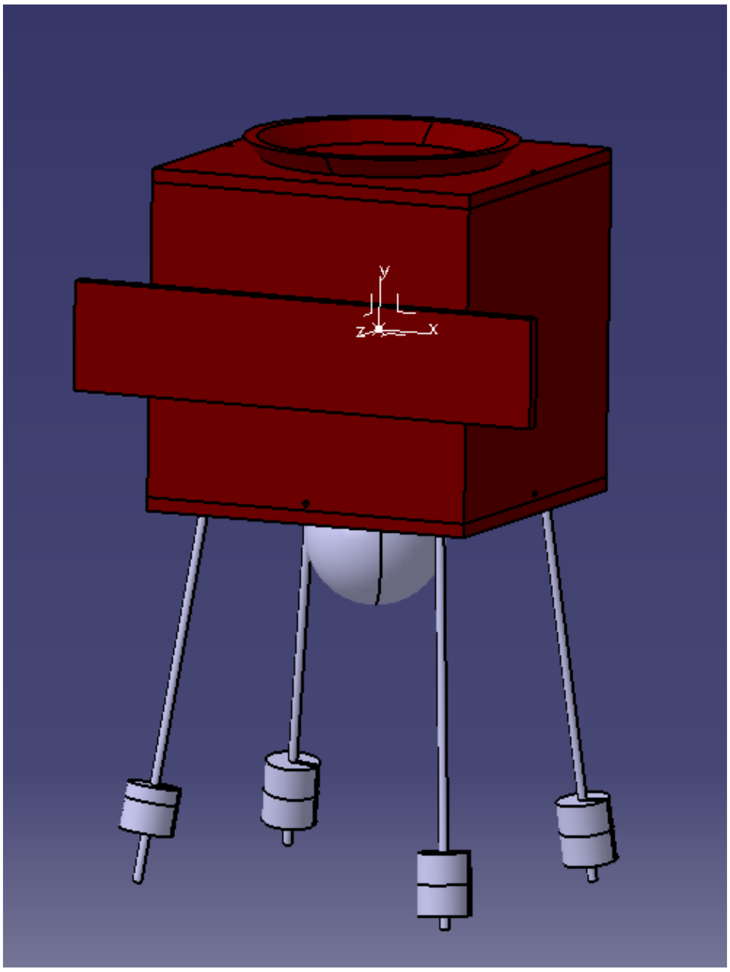


Figure 47 - CAD model of the satellite mock-up, spherical air bearing and counterbalancing system. The reference frame shown is parallel BRF.

The LIDAR selected as navigation sensor is the Intel-RealSense LIDAR camera L515, a low-cost, high-resolution solid state LIDAR depth camera, [87]. An overview of the sensor characteristics is provided in Table 25,

Table 25 - Intel RealSense LIDAR camera L515 characteristics, [87].

Operational parameters		**Noise parameters**	
Field of View	70°x55°	Depth standard deviation	2.5mm @ 1m
Resolution	1024x768		
Frame rate	30 fps		15.5mm @ 9m

The ground truth which allows evaluating the accuracy of the ICP-based pose estimate is provided by the PhaseSpace Impulse System: it is a motion capture system which provides the reference pose parameters of BRF with respect to MTRF. It consists of a string of eight LEDs powered by a small rechargeable battery pack imaged by a set of 12 cameras hung up around the experimental facility. The configuration of the LEDs on the spacecraft model is shown in Figure 48.

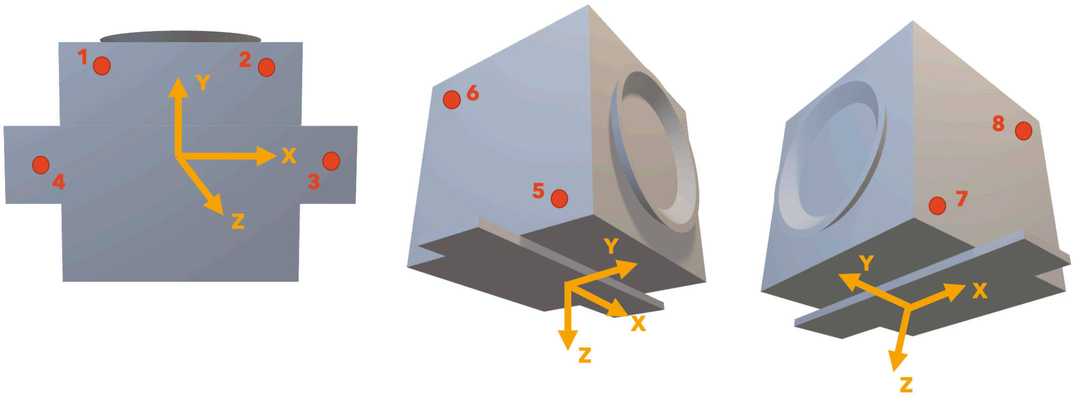


Figure 48 - Position of LEDs on the spacecraft mock-up represented by red dots. The axes shown in the figure are parallel to the ones of BRF.

Each LED blinks according to a unique pattern which make them recognizable and identifiable by the system. The cameras send the acquired data to a server through Ethernet cables which computes the position and attitude of the rigid body equipped with LEDs with a latency of 8 milliseconds and an accuracy of 1-5 millimeters, [88].

5.2 Testing Procedure

Once the system is balanced to a satisfactory level and the spherical bearing is active, an instantaneous torque can be applied to the model, and it will start rotating simulating an almost friction and torque-free motion. While the model is rotating, the LIDAR acquires images of the scene, and the target motion is estimated through the ICP-based pose determination strategy presented in Chapter 3. It is worth noting that the ICP requires an initial estimate of the pose parameters (pose initial guess). However, since this activity does not focus on the pose acquisition task, the pose initial guess is obtained from the PhaseSpace data: to obtain the pose parameters of BRF with respect to MTRF, the position of each LEDs in BRF must be known. They can be determined by combining the knowledge of the geometry of the 3D-printed model and the position of the LEDs tracked by PhaseSpace system.

Also, the extrinsic calibration parameters (i.e., the position vector and the attitude quaternion of LRF with respect to MTRF) must be computed to obtain the pose initial guess of BRF with respect to LRF and to compare the PhaseSpace-based and LIDAR-based pose measurements. The extrinsic calibration procedure is described in the following.

5.2.1 Extrinsic Calibration

Horn's absolute orientation method is applied to obtain the translation and rotation vector of LRF with respect to MTRF, [89]. It requires the knowledge of the position of at least four points in both reference frames. To this aim a string of eight LEDs is placed on a flat panel, as shown in Figure 49. The position of the LEDs in MTRF are straightforward to obtain by turning on the LEDs and tracking them with PhaseSpace.

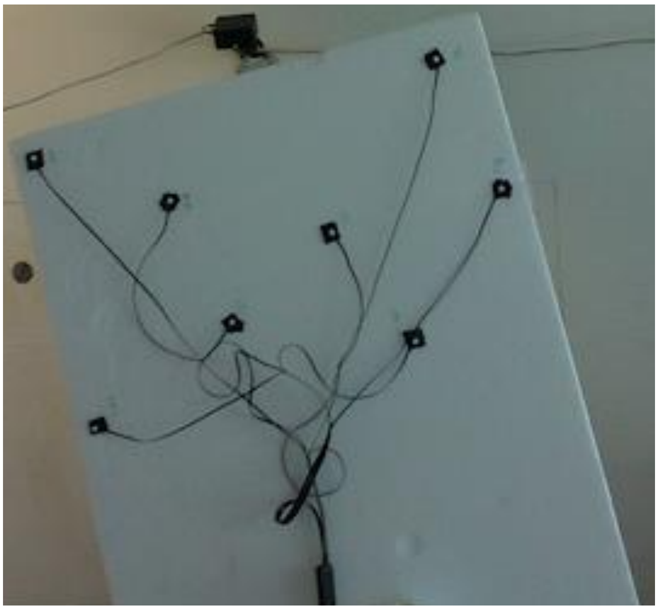
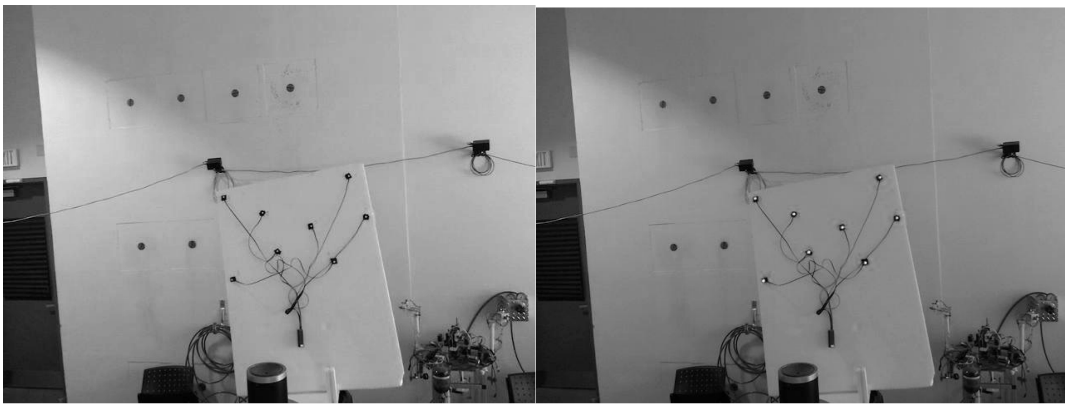
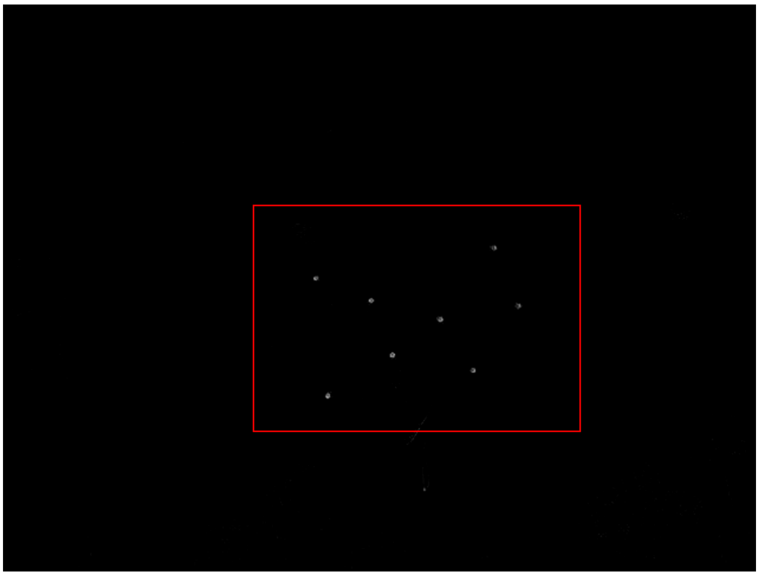


Figure 49 - Calibration object.

On the other hand, to obtain the position of the LEDs in LRF, the RGB sensor of the LIDAR L515 is exploited. Two different images of the calibration object are acquired: one with the LEDs switched on and one with the LEDs switched off. Then, by performing the pixel-wise subtraction of the corresponding grey-scale normalized images in Figure 50 a-b, one containing only the LEDS is obtained (see Figure 50 c).



(a) (b)



(c)

Figure 50 – (a) Grey-scale image with LEDs off (b) Grey-scale image with LEDs on (c) Difference intensity image obtained subtracting the two acquired images.

The image plane coordinated of each LEDs are obtained as the coordinates of the centroid of the blobs of pixels computed by weighting each pixel based on the intensity, as in Eq. (92),

$$U_i = \frac{\sum_{j=1}^{N} u_j I_j}{\sum_{j=1}^{N} I_j}, \quad V_i = \frac{\sum_{j=1}^{N} v_j I_j}{\sum_{j=1}^{N} I_j}, \quad i = 1,\ldots,N \tag{92}$$

where (U_i, V_i) and (u_j, v_j) are the image plane coordinates of the *i-th* blobs of the *j-th* pixel, respectively and I_j its intensity. Then, the normalized image coordinates are obtained as follows:

$$x_{dn,i} = \frac{U_i - p_u}{f_x}, \quad y_{dn,i} = \frac{V_i - p_v}{f_y} \tag{93}$$

where f_x and f_y are the horizontal and vertical focal lengths and (p_u, p_v) are the coordinates of the camera principal point.

Equation (93) represents the distorted normalized coordinates to obtain the undistorted ones, ($x_{n,I}$, $y_{n,I}$), the Brown Conrady distortion model must be applied to remove the radial and tangential distortion, as explained in [90]:

$$
\begin{aligned}
x_{n,i} &= x_{dn,i}\, h + 2k_3 h^2 x_{dn,i} y_{dn,i} + k_4\left(r + 2x_{dn,i}^2 h^2\right) \\
y_{n,i} &= y_{dn,i}\, h + 2k_3 h^2 x_{dn,i} y_{dn,i} + k_4\left(r + 2y_{dn,i}^2 h^2\right)
\end{aligned}
\tag{94}
$$

where

$$h = 1 + k_1 r + k_2 r^2 + k_5 r^3 \tag{95}$$

$$r = \sqrt{\left(x_{dn,i}^2 + y_{dn,i}^2\right)} \tag{96}$$

and k_i is the *i-th* image distortion coefficients.

To improve the accuracy of the extrinsic calibration procedure, multiple acquisitions varying the position and orientation of the calibration object with respect to the camera are considered: this allows solving the absolute orientation problems using 24 points instead of 8.

Figure 51 shows the reprojection error of the coordinated of each LED $\boldsymbol{P}_i$, computed with the obtained extrinsic calibration parameters ($\boldsymbol{t}_{PRF \to LRF}$, $\boldsymbol{q}_{LRF/PRF}$) as in Eq. (97), while in Table 26 the root mean square (RMS) and maximum of $\boldsymbol{x}_{\boldsymbol{err,rep}}$ are listed.

$$\boldsymbol{x}_{\boldsymbol{err,rep}} = \boldsymbol{P}_i^{MTRF} - \boldsymbol{t}_{MTRF \to LRF} + R_{LRF}^{MTRF} \boldsymbol{P}_i^{LRF} \tag{97}$$

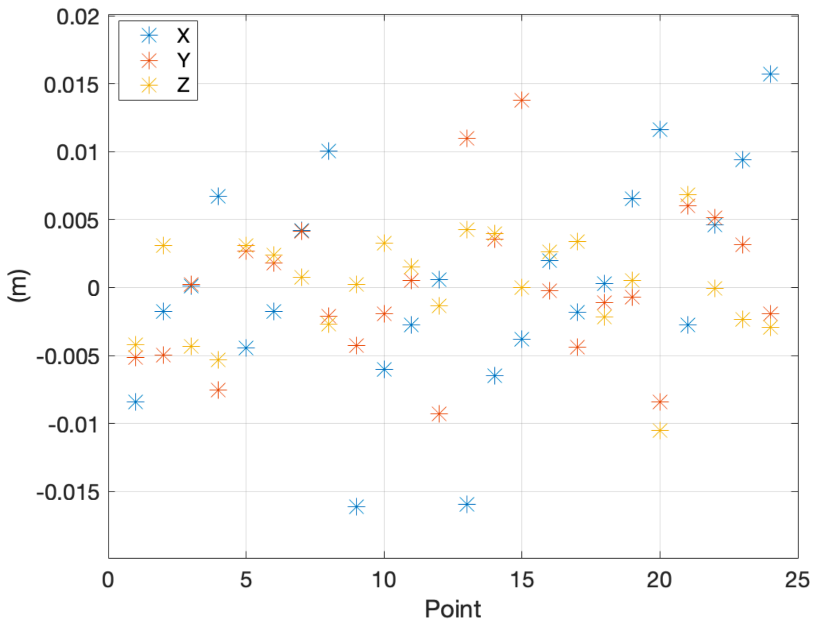


Figure 51 - Reprojection error of the points used for the extrinsic calibration.

Table 26 - Statistics of the reprojection errors.

	$x_{err,rep}(x)$	$x_{err,rep}(y)$	$x_{err,rep}(z)$
RMS (m)	0.0077	0.0056	0.0038
MAX (m)	0.0161	0.0138	0.0105

5.2.2 Residual Gravity Effect

Due to the difficulties of precisely balancing the system by moving the static weights along the balancing bars, it is not possible to obtain a perfectly gravity free-torque motion. However, the offset between the center of gravity (COG) and the center of ration (COR) along with the reference value for the moment of inertia ratios, can be both estimated from the CAD model by defining the density properties of each element of the system. Also, to check possible discrepancies between the real system and the modelled ones, the integral form of the Euler equation in which the gravity toque is added, and the COG-COR offset is included in the unknown vector can be exploited:

$$\Big[\underline{\underline{\Omega}}_{BRF/LRF}^{LRF}(t+\Delta t) - \underline{\underline{\Omega}}_{BRF/LRF}^{LRF}(t) + \\ + \int_t^{t+\Delta t} [\boldsymbol{\omega}_{BRF/LRF}^{LRF} \times]\underline{\underline{\Omega}}_{BRF/LRF}^{LRF} d\tau \,, \int_t^{t+\Delta t} [\boldsymbol{g}\times] d\tau\Big] \left[\bar{\boldsymbol{I}}_v \,;\, \frac{m\boldsymbol{r}}{I_{xx}}\right] = \begin{bmatrix} -\omega_x \\ 0 \\ 0 \end{bmatrix} \tag{98}$$

In Eq. (98) $\boldsymbol{g}$ is the gravity vector; m is the mass of the system and $\boldsymbol{r}$ is the COG-COR offset vector. The system of Eq. (98) is solved using the data acquired by the PhaseSpace system. The results obtained from different acquisitions (denoted as PS-i) along with the values obtained from the CAD model are reported in Table 27.

Table 27 - MIRs and COG-COR offset estimated from the CAD model and PhaseSpace data.

Parameter	CAD	PS-1	PS-2	PS-3	PS-4	PS-5	PS-6
I_{yy}/I_{xx}	0.2440	0.2897	0.2938	0.2413	0.2465	0.2359	0.2741
I_{zz}/I_{xx}	0.9900	1.0040	0.9879	0.9147	0.8928	0.9563	0.9532
I_{xy}/I_{xx}	0.0096	-0.0060	0.0024	0.0060	-0.0031	-0.0072	0.0166
I_{xz}/I_{xx}	-0.0032	-0.0153	-0.0029	-0.0064	-0.0186	0.0033	0.0124
I_{yz}/I_{xx}	-0.0289	-0.0203	-0.0218	-0.0067	-0.0108	-0.0284	-0.0009
mr_x/I_{xx}	0	-0.0014	0.0041	0.0017	0.0034	0.0046	-0.0015
mr_y/I_{xx}	-0.3467	-0.3853	-0.3887	-0.3722	-0.3716	-0.3768	-0.3807
mr_z/I_{xx}	0	0.0103	0.0090	0.0107	0.0088	0.0074	0.0071

The reason for the different results in the PhaseSpace acquisitions lies in the numerical differentiation of the quaternion to estimate the angular velocity. However, as shown in Table 28, the standard deviations of both the principal moment of inertia ratios (I_{yy}/I_{xx} and I_{zz}/I_{xx}) and the dimensionless COR-COG vertical offset are one order of magnitude lower than the estimated values. The high variability of the other parameters between the different acquisitions is due to the fact that, being very low with respect to the principal MIRs and vertical offset, they have a negligible impact on the rotational dynamics of the system which makes their estimate highly sensitive to the measurement noise.

Table 28 – Mean Standard Deviation of MIRs and COG-COR offset.

Parameter	Mean	Standard Deviation
I_{yy}/I_{xx}	0.2680	0.0245
I_{zz}/I_{xx}	0.9570	0.0412

I_{xy}/I_{xx}	0.0026	0.0087
I_{xz}/I_{xx}	-0.0044	0.0106
I_{yz}/I_{xx}	-0.0168	0.0109
mr_x/I_{xx}	0.0016	0.0026
mr_y/I_{xx}	-0.3746	0.0138
mr_z/I_{xx}	0.0076	0.0036

It is worth underlining that the gravity torque term is only needed to counteract the fact that the system is not perfectly balanced. In an ideal case in which the centers of gravity and rotation are coincident, the only perturbation acting in the model is due to the friction between the spherical bearing cup and segment ball which has a negligible effect, [91].

5.3 Test Results

Before introducing and discussing the experimental results, the performance metrics adopted to evaluate the accuracy of both ICP and MIRs estimation algorithm are defined.

The attitude performance metrics of the ICP-based pose determination algorithm are the angular deviations between the reference (i.e., PhaseSpace data) and the estimated direction of BRF in LRF computed as in Eq. (99):

$$\begin{aligned} e_{x,dir} &= \cos^{-1}(\underline{\underline{R}}_{1,EST}, \underline{\underline{R}}_{1,TRUE}) \\ e_{y,dir} &= \cos^{-1}(\underline{\underline{R}}_{2,EST}, \underline{\underline{R}}_{2,TRUE}) \\ e_{z,dir} &= \cos^{-1}(\underline{\underline{R}}_{3,EST}, \underline{\underline{R}}_{3,TRUE}) \end{aligned} \tag{99}$$

where $\underline{\underline{R}}_i$ is the i-th column of the rotation matrix representing the attitude of BRF with respect to LRF. Regarding the moment of inertia ratios, the error metric is defined as follows:

$$I_{ERR} = \frac{|I_{REF} - I_{EST}|}{I_{REF}} \tag{100}$$

where $\boldsymbol{I}_{REF}$ is the vector containing the mean values among the reference solutions obtained by the PhaseSpace-based test and from the CAD model (listed in Table 27). As described in Chapter 3, the first step is to obtain attitude and angular velocity estimates by processing the acquired point clouds. The time variation of the attitude performance metrics for the test case presented in this Chapter is depicted in Figure 52, while the statistics are listed in Table 29.

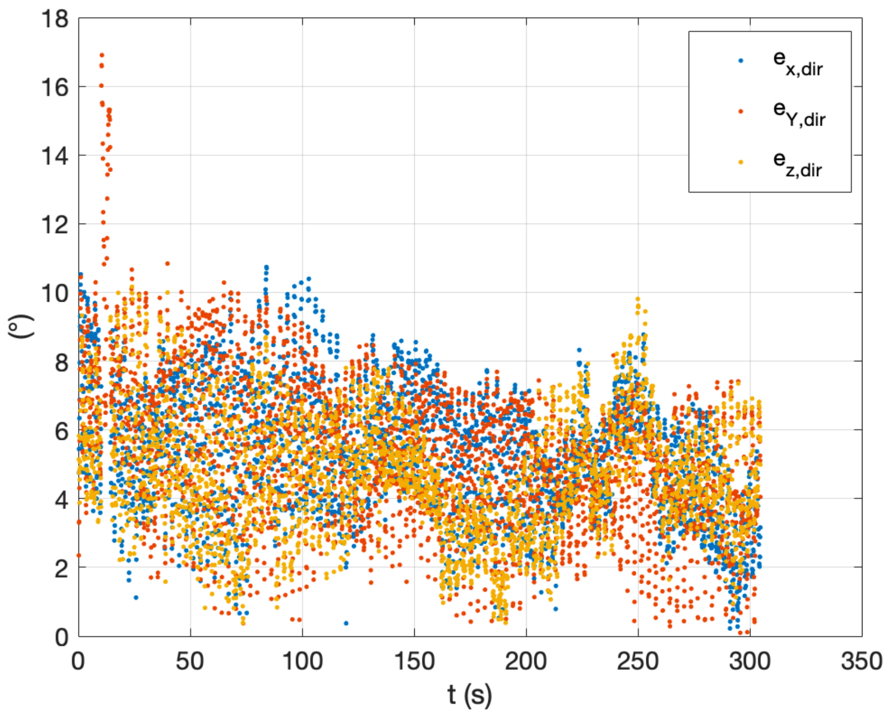


Figure 52 - Time variation of the performance metrics defined for the relative attitude.

Table 29 - Statistics of the performance metrics defined for the relative attitude.

Metrics	**Mean (°)**	**Standard Deviation (°)**
$e_{x,dir}$	5.312	1.897
$e_{y,dir}$	5.299	2.23
$e_{z,dir}$	4.686	1.775

It is worth outlining that the errors of the ICP-based pose determination algorithm include, along with the errors related to the limited accuracy of the sensors, the calibration errors (cm-level error) and the tracking errors of the PhaseSpace system.

These latter errors include two contributions: the errors in the tracking of each LED and the uncertainty due to the imperfect knowledge of their position in BRF.

Once the attitude information is collected, the angular velocity of the system is estimated through Eq. (54). Since the estimate of the angular velocity must be as accurate as possible to obtain a good estimate of the MIRs, the time interval over which the rotation angle $\boldsymbol{\varphi}$ (see Eq. (53)) must be computed is not constant but it is chosen by taking into account the accuracy of the estimated pose provided by the value of the ICP cost function at the last iteration, f_{END}. Thus, the numerical derivative of $\boldsymbol{\varphi}$ is computed only if the value of f_{END} is lower than a user defined threshold.

With the estimated values of attitude and angular velocity, along with the COG-COR offset provided by the CAD model, Eq. (101) is exploited to obtain an estimate of the MIRs whose results are summarized in Table 30. These values are considered as reference value since they fall in the range mean $\pm$ standard deviation obtained from the tests listed in Table 27.

It is worth noting that in this case, with respect to the procedure described in Chapter 3, the hypothesis of the conservation of the angular momentum cannot be applied due to the presence of the gravity torque term.

$$\begin{aligned} &[\underline{\underline{\Omega}}_{TRF/LRF}^{LRF}(t+\Delta t) - \underline{\underline{\Omega}}_{TRF/LRF}^{LRF}(t) + \\ &+ \int_{t}^{t+\Delta t} [\boldsymbol{\omega}_{TRF/LRF}^{LRF} \times] \underline{\underline{\Omega}}_{TRF/LRF}^{LRF} dt] \boldsymbol{I}_T d\tau = \\ &= \begin{bmatrix} -\omega_x \\ 0 \\ 0 \end{bmatrix} - \int_{t}^{t+\Delta t} \frac{m[\boldsymbol{g} \times]\boldsymbol{r}}{I_{xx}} d\tau \end{aligned} \tag{101}$$

Table 30 - Estimation errors of the principal MIRs.

MIRs	Reference values	Estimated values	Performance metrics
I_{yy}/I_{xx}	0.2440	0.2186	10.40%
I_{zz}/I_{xx}	0.9900	1.0751	8.60%

It is important to underline that the results provided in Table 30 are not only influenced by the uncertainty on the angular velocity estimate, but they also depend on the accuracy on the knowledge of the gravity torque (i.e., COG-COM offset) and I_{xx} terms added to Eq. (101) to counteract the effect of an imperfectly balanced system and derived from the CAD model. The performance of the algorithm is expected to improve with a more accurate knowledge of the system parameters and by integrating an active balancing system able to perfectly align the COG with the COR.

6. Aerodynamic de-orbiting control

This chapter presents a trajectory control strategy for the de-orbiting phase of a micro-satellite equipped with a deployable aerobrake. This activity is framed in the design of the Italian space mission MISTRAL (MIcro-SaTellite with Air-Launchable Re-entry capabilities). Developed under the supervision of the *DAC-Campanian Aerospace District*, MISTRAL aims at developing a prototype of a multi-purpose air-launchable 50 kg class micro-platform, able to return limited mass and volume payloads to Earth.

6.1 Literature review

As stated in Chapter 1, deployable aerobrakes for re-entry satellites may offer many advantages in the near future, including the opportunity to recover payloads and samples with reduced risks and costs with respect to conventional systems. A deployable aerobrake can be modulated to control the re-entry trajectory and to correctly guide the capsule towards the selected landing site.

The de-orbit and re-entry control problems have been addressed with several approaches in the scientific literature. Typically, a nominal trajectory is designed, and then a trajectory tracking feedback controller is used to counteract uncertainties and external disturbances; moreover, a gain scheduling of the controller parameters is also adopted to adapt to different flight conditions.

Feedback control laws can be designed as linear-proportional-integrative-derivative (PID) actions, [92], [93]. In [94], a gain scheduling controller is designed on the basis of a Linear Parametric Varying (LPV) model describing the vehicle dynamics. In [95], a double-loop control system with fuzzy gain-scheduling is proposed. Other approaches based on fuzzy logic are presented in [96] and [97], where PID controllers are used to track the drag minimal reference profile. Time-varying linear feedback controllers are proposed in [98], [99]. In [100], an adaptive controller is based on feedback linearization theory, coupled with a sliding mode observer to estimate the drag and its rate of variation. Controllers based on feedback linearization are also presented in [101] and [102]. In [103], the control algorithm, is based on a time-varying Linear Quadratic Regulator (LQR) minimizing the trajectory tracking error. Other controllers based on gain scheduling techniques are presented in [104], [105] and [106]. In [107] a nonlinear PID control is proposed: it guarantees globally asymptotically stable tracking of the reference trajectory [108], or to optimize the control action taking into account the error with respect to the target landing position ([109] and [110]). In [111] and [112], the trajectory task is achieved by using an incremental nonlinear dynamic inversion controller.

When using deployable surfaces, the control signal aims at modifying the ballistic coefficient of the spacecraft. A common idea is to build a dynamic aero-brake with the ability to change its shape, and therefore its area during the flight, [113]-[114], in order to control the trajectory to reach a predefined landing location [115]. The modulation of the ballistic coefficient has been investigated as a viable option in spacecraft formation control and spacecraft de-orbiting and re-entry, so to reduce the risk of reaching populated areas [104], [116], [117]. As regards the use of the aerodynamic force in formation flying applications to control the geometry if a spacecraft constellation, bang-bang control techniques have been applied in open loop configuration [118]. Open-loop control techniques modifying the ballistic coefficient at a certain time instant, have also been exploited to assign the re-entry time and consequently change the longitude and latitude of the re-entry location [117]. Although this approach has been applied for the re-entry of Skylab, it exhibits a low robustness due to the absence of feedbacks: this makes the system not reactive to uncertainties, such as those on the atmospheric density, which is particularly difficult to predict. Performance of a drag-based re-entry control algorithm can be drastically improved by the introduction of a closed-loop control, as proposed in [119], in which a Model Predictive Control (MPC) scheme is implemented to change the ballistic coefficient so to minimize the state error.

In this framework, a feedback control law has been selected for MISTRAL. However, two different approaches have been adopted for the de-orbiting (i.e., from 300 km up to 150 km) and re-entry phase (i.e., 100 km up to 30 km). The idea to adopt two different approaches for the de-orbiting and re-entry phases comes from the consideration that the stability of the control algorithm is one of the main requirements for the first phase, since it lasts longer than the second one. To this end, an LQR based approach with state feedback has been selected, since, among the linear controllers, it results in a control law that is guaranteed to be stable (for the linearized system). Non-linear controls have not been considered due to the associated computational burden. On the other hand, the main objective of the re-entry phase is to land in the correct area. To this purpose, an MPC approach with terminal cost on the prediction time horizon proved to be more effective. This thesis focuses on the de-orbiting phase and the algorithmic details of the LQR-based control along with the results of an extensive simulation campaign are presented in the next sections.

6.2 Control algorithm for the de-orbiting phase

During the de-orbiting phase, MISTRAL shall decay from an initial altitude $H_D = 300$ km to an altitude $H_R = 150$ km following a reference orbital decay trajectory. The control system must be able to keep the spacecraft as close as possible to the nominal trajectory despite all the uncertainties, system constraints (including those deriving from the actuator performance limitations), and the environment perturbations (i.e., atmospheric density variations) by varying the ballistic coefficient with respect to the reference value provided by the guidance.

With respect to previous work [105] where a dead-band approach has been introduced, a fixed control frequency (0.0042 Hz) and a moving average filtering is adopted to avoid high frequency switching of the commanded ballistic coefficient. This also makes the tracking algorithm robust to GPS-noise and unmodelled perturbations (e.g., high order gravity terms).

To describe the deviation of a real spacecraft trajectory with respect to the reference one, a linearized model around the reference path is used. A HRF with the origin in the center of a mass of a fictious spacecraft following the nominal trajectory has been defined. The analytic model used to describe the time evolution of the system is given by Schweighart-Sedwick linearized equations [44], [120]. These equations are similar in form to Hill's equation in [44] but they include the mean effect of J_2, as shown in the following:

$$\begin{cases} \ddot{X}_{\mathrm{HRF}} - 2\dot{Y}_{HRF}nc - (5c^2 - 2)n^2 X_{HRF} = 0 \\ \ddot{Y}_{HRF} + 2\dot{X}_{HRF}nc = 0 \\ \ddot{Z}_{HRF} + 3(c^2 - 2)n^2 Z_{HRF} = 0 \end{cases} \tag{102}$$

In Eq. (102) n is the orbital angular velocity, X_{HRF}, Y_{HRF} and Z_{HRF} are the radial, along-track and cross-track coordinates of the spacecraft in HRF, while c is defined as follows:

$$c = \sqrt{1 + \frac{3J_2 R_e^2}{8|\boldsymbol{R_{ECI}}|^2}(1 + 3\cos 2i)} \tag{103}$$

with R_e being the Earth radius, $|\boldsymbol{R_{ECI}}|$ is the norm of the position vector of the fictious spacecraft and i its orbital inclination.

The control algorithm must provide the time variation of the ballistic coefficient minimizing the deviation of the spacecraft from the reference trajectory. Since the aerodynamic drag has a negligible effect on the out-of-plane motion, only the equations describing the in-plane motion in Eq. (102) are considered. The differential drag $\Delta \boldsymbol{D}$ experienced by the real spacecraft with respect to the reference one is the control input inducing a relative acceleration and it is expressed as follows:

$$\Delta D = -\rho|\boldsymbol{V_{ECI}}|^2 \ \Delta C_b \tag{104}$$

where ρ is the atmospheric density, $|\boldsymbol{V_{ECI}}|$ is the norm of the reference Earth-relative velocity vector, provided by the guidance algorithm, and $\Delta C_b$is the commanded difference between the ballistic coefficient needed to follow the reference trajectory and the nominal one; the variation of the ballistic coefficient ΔC_b represents the parameters on which the control logic can act. It is then converted in the actuation command via a static relationship. The ballistic coefficient is defined as follows:

$$C_b = \frac{C_d A}{2m} \tag{105}$$

where C_d is the drag coefficient, *A* is the cross-sectional area and *m* is the spacecraft mass.

Defined the state vector $\boldsymbol{x^D} = \left[X_{HRF},\ Y_{HRF},\ \dot{X}_{HRF},\ \dot{Y}_{HRF}\right]^T$ and the input vector $u^D = \Delta C_b$, the dynamic model of the system can be written in the matrix state space form as in [105]:

$$\dot{\boldsymbol{x}}_{\boldsymbol{D}} = \begin{bmatrix} \dot{X}_{HRF} \\ \dot{Y}_{HRF} \\ \ddot{X}_{HRF} \\ \ddot{Y}_{HRF} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ (5c^2-2)n^2 & 0 & 0 & 2nc \\ 0 & 0 & -2nc & 0 \end{bmatrix} \begin{bmatrix} X_{HRF} \\ Y_{HRF} \\ \dot{X}_{HRF} \\ \dot{Y}_{HRF} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ -\rho v_g^2 \end{bmatrix} \Delta C_b = \underline{\underline{A}}\boldsymbol{x_D} + \underline{\underline{B}}u_D \tag{106}$$

The adopted control logic is based in a Linear Quadratic Regulator (LQR) approach. Given the linearized model (106), in which the state vector is composed of the radial and along-track distance of the spacecraft from the reference trajectory, optimal control is aimed at nullifying the distance between the actual spacecraft position and the desired one: this is equivalent to minimize the following objective function,

$$\mathcal{J}_D = \int_0^{\infty} \left(\boldsymbol{x}_{\boldsymbol{D}}^{\boldsymbol{T}} \underline{\underline{Q}} \boldsymbol{x}_{\boldsymbol{D}} + u_D^T \underline{\underline{R}} u_D \right) dt \tag{107}$$

where t is the time, $\underline{\underline{Q}}$ and $\underline{\underline{R}}$ are the state and control positive semidefinite penalty matrix weighting the relative importance between two different goals: reaching the desired state as fast as possible and minimizing the control effort. In practical application, they are generally designed as diagonal matrix. The control variable u_D is:

$$u_D = -\underline{K} \boldsymbol{x}_{\boldsymbol{D}} \tag{108}$$

Since the along-track error is greater than the radial one, due to the direction in which the drag force acts on the spacecraft, it can be assumed that the control performance is mostly affected by the along-track error. Thus, the quadratic performance index is modified as follows:

$$\mathcal{J}_D = \int_{\mathbf{0}}^{\infty} (Q_2 \; Y_{HRF}^2 \; + R_1 \, u_D^2) \, dt \tag{109}$$

According to the Bryson principle [121], which is widely applied for the selection of the weighting matrix in LQR-based control law, Q_2 and R_1 can be selected by first normalizing them with respect to the maximum values of the states and control quantities, as in Eq. (110).

$$Q_2 Y_{HRF_{max}}^2 = R_1 \, u_{D_{max}}^2 = 1 \tag{110}$$

The value of Q_2 and R_1 provided by the Bryson principle are used as initial guess. They have been refined to optimize the performance of the system in terms of both tracking error and control effort by means of numerical analysis.

Since Eq. (106) represents a linear time varying model, a practical approach to take into account the time variation of the matrix coefficients, is to update the LQR gains on the basis on the value of the atmospheric density: every time the ratio between the actual atmospheric density and the one used to compute the gains is larger than a given constant (1.2 is set in this case), the linearized matrices are updated and new gains are computed. This ends up in a continuous gain scheduling of the feedback control gains. Obviously, the controller gains matrices can be computed off-line and stored in the on-board computer.

Also, in order to smooth the oscillations of the required command caused by the measurement noise and the high gains, a moving average filter in the discrete time of the control command is applied. The order of the filter is chosen so as to average the control input over a one-hour time window.

6.3 Numerical simulation environment and results

A 3-degree of freedom orbital propagator based on Eq. (111), has been developed in MATLAB/Simulink to simulate the orbital flight dynamics of the MISTRAL capsule. It includes the main LEO perturbations, i.e., the atmospheric drag and the gravitational harmonics up to the 4th order.

$$\dot{\boldsymbol{R}}_{\boldsymbol{ECI}} = \boldsymbol{V}_{\boldsymbol{ECI}}$$

$$\dot{\boldsymbol{V}}_{\boldsymbol{ECI}} = -\mu \cdot \frac{\boldsymbol{R}_{\boldsymbol{ECI}}}{|\boldsymbol{R}_{\boldsymbol{ECI}}|^3} + \boldsymbol{a}_{\boldsymbol{J}_2} + \boldsymbol{a}_{\boldsymbol{J}_3} + \boldsymbol{a}_{\boldsymbol{J}_4} - \frac{1}{2 \cdot M_c} \cdot \rho \cdot |\boldsymbol{V}|^2 \cdot C_D \cdot S \cdot \hat{\boldsymbol{V}} \tag{111}$$

In Eq. (111):

- $\boldsymbol{R}_{\boldsymbol{ECI}} = [X_{ECI}, Y_{ECI}, Z_{ECI}\ ]^T$ is the position vector of the capsule in the ECI frame
- $\boldsymbol{V}_{\boldsymbol{ECI}} = \left[V_{X,ECI}, V_{Y,ECI}, V_{Z,ECI}\ \right]^T$ is the velocity vector of the capsule in the ECI frame

- $\boldsymbol{V} = \boldsymbol{V_{ECI}} - \boldsymbol{\Omega_E} \times \boldsymbol{R_{ECI}}$ is the Earth-relative speed of the capsule,
- $\boldsymbol{\Omega_E}$ is the rotation speed of the Earth,
- $\rho = \rho(H)$ is the atmospheric density,
- $C_D = C_D(H)$ is the drag coefficient,
- S is the surface of the aerobrake,
- M_c is the mass of the capsule,
- μ is the gravitation universal constant,
- $H = |\boldsymbol{R_{ECI}}| - R_e$ is the geometric altitude,
- R_e is the approximate radius of the Earth,
- $\boldsymbol{a_{J_2}}$, $\boldsymbol{a_{J_3}}$, and $\boldsymbol{a_{J_4}}$ are the accelerations related to zonal harmonics J_2, J_3 e J_4,
- $|\cdot|$ indicates the Euclidean norm.

The model assumes that the spacecraft is a single point of mass, no rotational dynamics are considered, and the aerodynamic side force is assumed negligible. It also considers a standard atmospheric model USSA76 with a perturbed density model reported in Eq. (112),

$$k_{err} = k_0 + \sum_{i=1}^{3} k_i \sin\left(\frac{2\pi}{Ti} t - \varphi_i\right) \tag{112}$$

where T_i values are set to 26 days, 1 day and 5400 seconds, respectively, k_i values are set to 0.25, 0.1 and 0.1, while φ_i and k_0 are randomly selected from a uniform distribution between 0 and 2π and 0.77 and 1.3, respectively.

Clearly, the atmospheric density is, among all the variables at play, the most difficult to predict. The use of this perturbation model guarantees a realistic uncertainty in the knowledge of the atmospheric density with respect to the one used to generate the reference trajectory. This allows to test the control robustness against the high variability of this parameter.

Also, a simplified actuation system model is included in the simulation environment taking into account quantization errors of the umbrella like device which is reflected in a quantization level of the aero-brake surface. The actuator command is subject to a maximum/minimum amplitude, as shown in Figure 53, a maximum rate of variation of 3 mm/s, duty cycle limitations of 10% and position error uniformly distributed between -0.07 and 0.07 mm. Figure 53 shows the non-linear relation between the aero-brake surface and the aero-brake linear command.

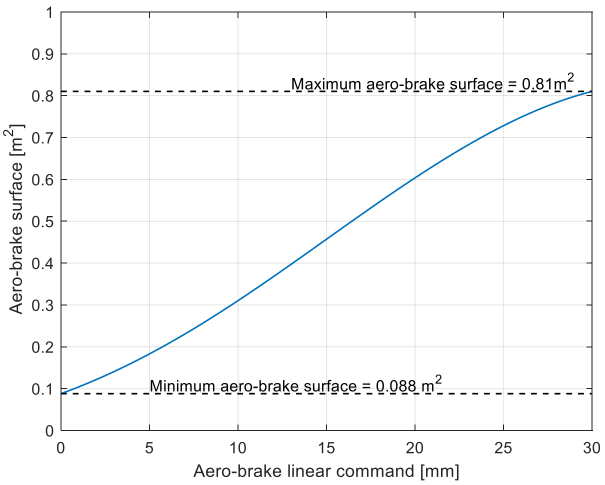


Figure 53 - Aero-brake linear command vs aero-brake exposed area.

Also, an uncertainty on the navigation data based on the performance of OEM719 Multi-Frequency GNSS Receiver has been simulated. Specifically, the navigation error has been considered as a sum of two contributions:

- A Gaussian white noise with standard deviation of $1.5/\sqrt{2}$ meters for both horizontal and vertical error and a standard deviation of $0.03/\sqrt{3}$ meters per seconds for speed error. Those values have been defined on the basis of Cubesat kit GPSRM-1 GPS Receiver which uses a OEM719 Multi-Frequency GNSS Receiver, [122].

- A sinusoidally varying bias errors with a frequency equal to the average orbital angular velocity and a millimeter level amplitude for position error and 10^{-2} millimeter per second amplitude for velocity error.

The proposed numerical simulation results are obtained starting from the main mission scenario orbit which is characterized by the parameters reported in Table 31. The nominal trajectory for the de-orbit phase is generated as described in [104] by considering a piecewise constant aero-brake surface divided as follows:

- $S^{Ref} = 0.15\ m^2$ for altitude in the range $300k\ m \leq H < 285\ km$.
- $S^{Ref} = 0.5\ m^2$ for altitude in the range $285\ km \leq H < 160\ km$.
- $S^{Ref} = 0.15\ m^2$ for altitude in the range $160\ km \leq H < 150\ km$.

Table 31 - MISTRAL mission scenario.

Parameter	Value
Altitude	300 km
Semi-major axis	6678 km
Eccentricity	0
Right Ascension of Ascending Node	32.57°
Inclination	51.6°
Argument of perigee	0°
True anomaly	356.60°
Satellite Epoch	25/11/2020 10:00:00 (UTC)
Target coordinates	[-29.52°, 133.35°]

Finally, to test the validity of the control approach, a large family of simulations is generated considering uncertainties on the initial conditions, vehicle parameters randomly selected from uniform distributions which have the following characteristics:

- Maximum uncertainty of ±20% on the drag coefficient.
- Maximum uncertainty of ±10% on the mass of the spacecraft.
- Maximum uncertainty on the actuator position ±0.07 mm.

- Maximum deviation of ±1 km from the nominal initial position.
- Maximum deviation of 0.00013 from the nominal orbital eccentricity (which leads to a velocity error of about 1 meter per second.

The control time cycle was set to 240 seconds (i.e., about 0.0042 Hz) and the performance of the control system is evaluated by considering the position and velocity error of the spacecraft with respect to the reference trajectory at the end of the de-orbiting phase.

Three different simulation cases and statistical analysis are reported hereinafter.

The uncertainties considered for each simulation are listed in Table 32, whereas the results are reported in Table 33: the purpose of comparison between the presented simulation results is to give an idea of how the different uncertainties and their combination affects the performance of the system.

Table 32 - De-orbiting phase: simulation uncertainties.

Uncertainties			
	Simulation	Simulation #2	Simulation #3
Uncertainty on actuator position (1σ)	0.0055	0.0055 mm	0.0055 mm
Uncertainty on navigation data (sensor noise)	No	Yes	Yes
Uncertainty on drag coefficient	0%	0%	3.21 %
Uncertainty on the mass of the spacecraft	0%	0%	1.1 %
Uncertainty on the atmospheric density	Yes	No	Yes
Uncertainty on the initial state	No	Along track error 1 km, Velocity error 1 m/s	Along track error 885.5 m, Velocity error 0.4 m/s

Table 33 - De-orbiting phase: position and velocity errors at H_R = 150 km.

Errors @ H_R			
	Simulation #1	Simulation #2	Simulation #3
ECI Position Error $E_{\lvert\boldsymbol{R}_{ECI}\rvert}^{@H_R}$	$0.24\ km$	$0.012\ km$	$0.074\ km$
ECI Speed Error $E_{\lvert\boldsymbol{V}_{ECI}\rvert}^{@H_R}$	$0.24\ m/s$	$0.030\ m/s$	$0.060\ m/s$
Distance from target point on the Earth surface	$0.32\ km$	$0.017\ km$	$0.11\ km$

In Simulation #1, the effect of the uncertainty in the knowledge of the atmospheric density is firstly analyzed. The time variation of the commanded aerobrake is depicted in Figure 54, compared with the nominal command calculated by the Guidance algorithm. Figure 55 and Figure 56 show the time variation of the position and velocity errors, respectively: the uncertainty in the knowledge of the atmospheric density causes a position error in the order of 100 meters, and a speed error in the order of 1 meter per second at the final height (i.e., 150 km); the commanded value if the aero-brake surface is biased with respect to the reference values provided by the guidance with moderate oscillations due to the difference between the known value of the atmospheric density (provided by the model USSA76) and the actual one (perturbed by model of Eq. (112)): this results in a steady-state term in both tracking position and velocity errors.

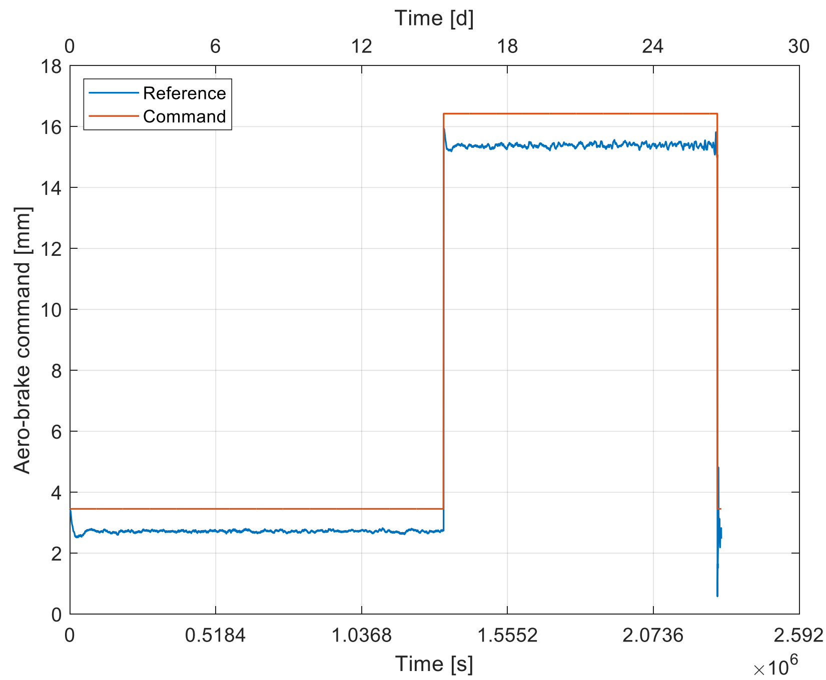


Figure 54 - De-orbiting phase, Simulation #1: Aero-brake control command (blue) and aero-brake command of the reference trajectory (red).

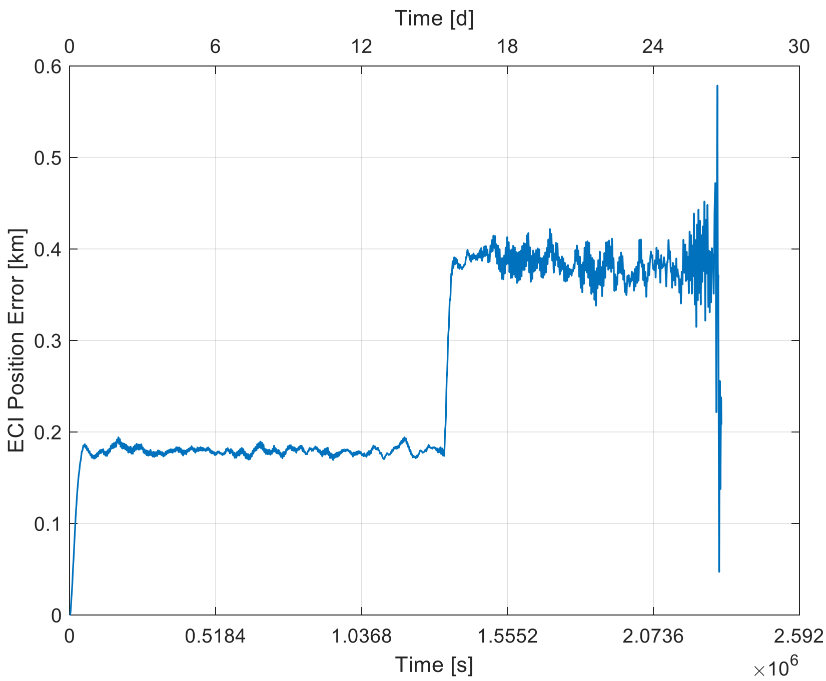


Figure 55- De-orbiting phase, Simulation #1: ECI position error.

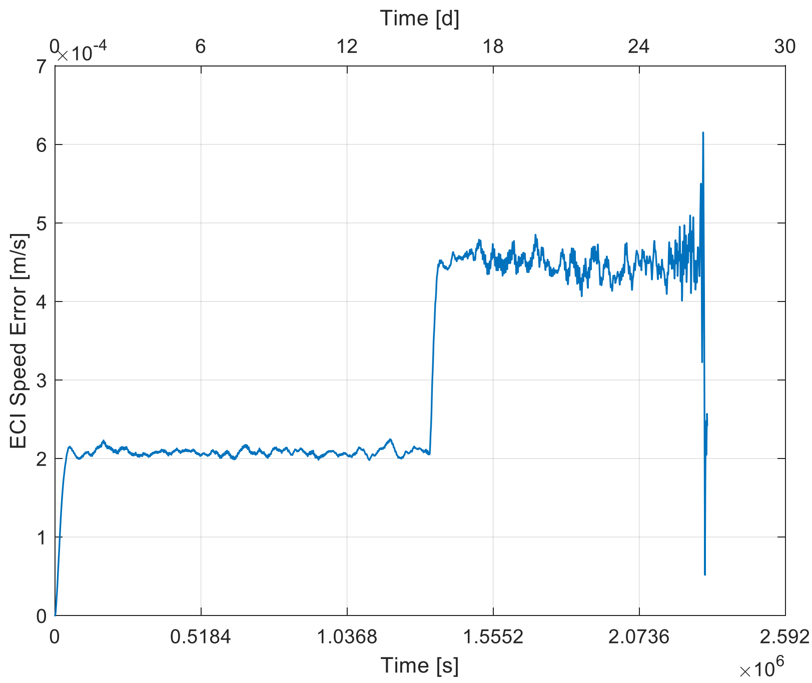


Figure 56 - De-orbiting phase, Simulation #1: ECI speed error.

As a second test case (called Simulation #2), the effect of initial state displacements with respect to the nominal one is analyzed. Assuming an initial along-track error of 1 kilometer, and a velocity error of 1 meter per second, Figure 57 shows the time variation of the commanded value of the aero-brake surface over the de-orbiting trajectory. Also, a highlight of the first 6000 seconds shows the oscillation of the aero-brake command over one orbital period. Figure 58 and Figure 59 show the time variation of the position and velocity errors. It can be noted that, although the perturbation on the initial state of the orbit causes higher oscillations of the aero-brake command with respect to Simulation #1, the final position and speed errors are comparable to the one of the previous simulation, proving that the control system is able to nullify the initial tracking error.

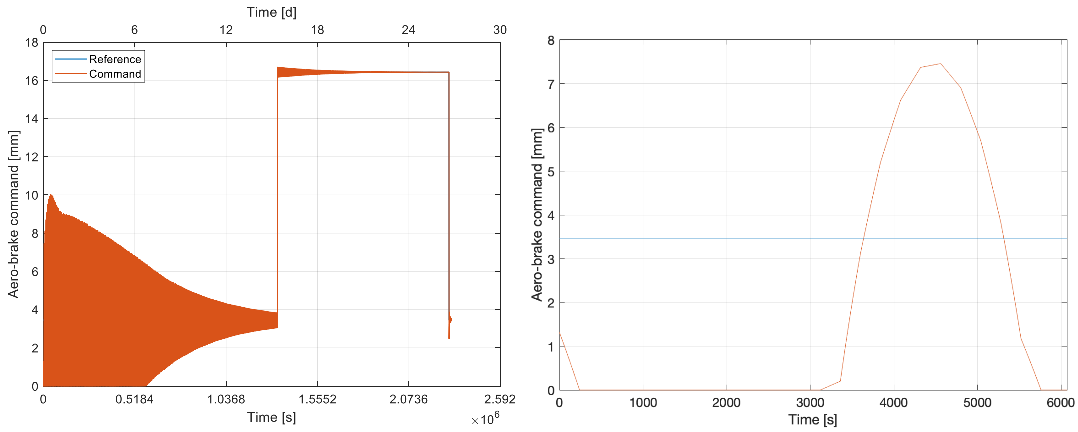


Figure 57 - De-orbiting phase, Simulation #2: (left) Aero-brake control command (blue) and aero-brake command of the reference trajectory (red). (right) Zoom of the first 6000 seconds.

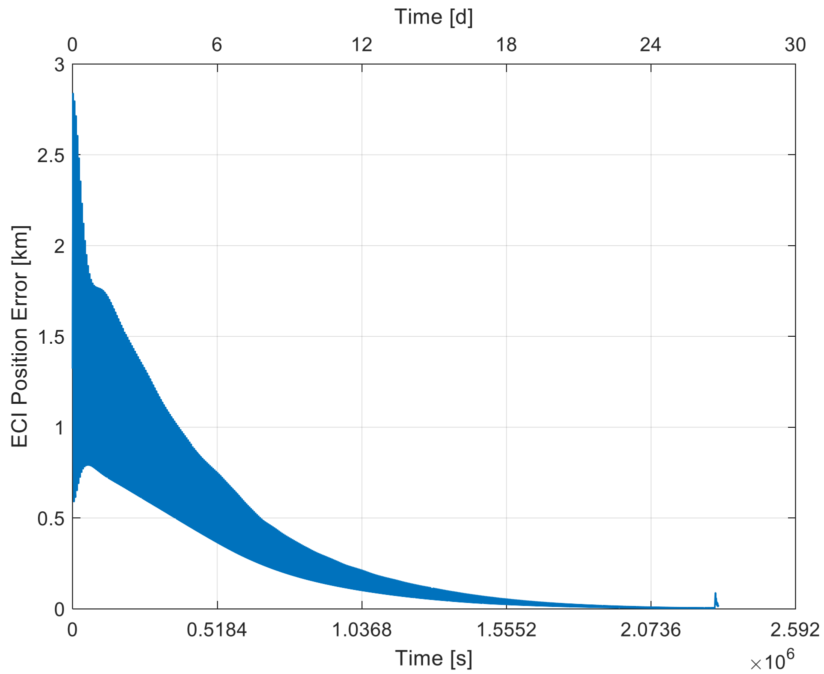


Figure 58 - De-orbiting phase, Simulation #2: ECI position error.

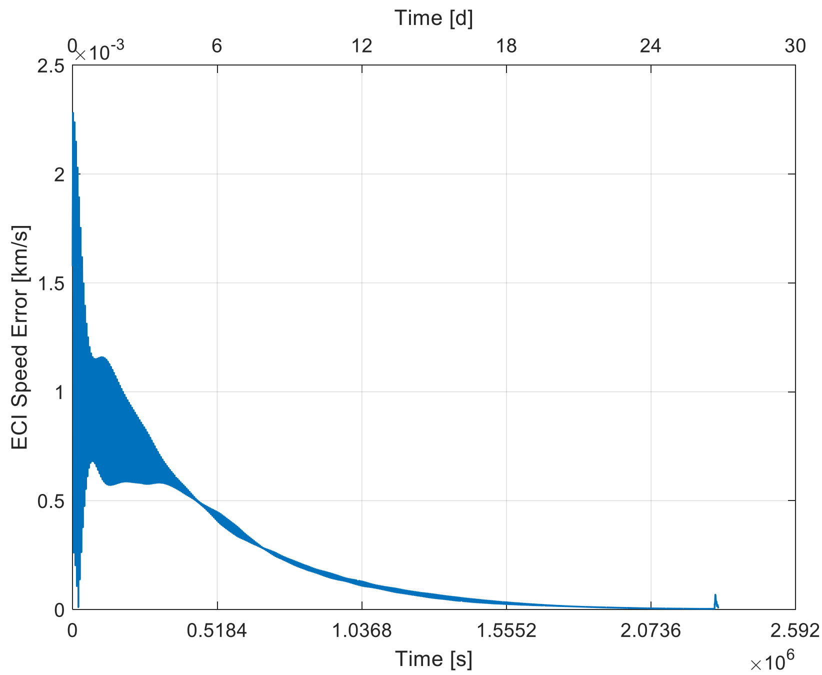


Figure 59 - De-orbiting phase, Simulation #2, ECI speed error.

In Simulation #3, the effect of the uncertainties on drag coefficient and satellite's mass are also taken into account (the values of the uncertainties are listed in Table 32). Figure 60, Figure 61 and Figure 62 depict the time behavior of the commanded value of the aero-brake and of the position and velocity errors.

The first part of the de-orbiting phase is characterized by a higher control effort as a consequence of the large initial errors and of the low value of the atmospheric density, which reduces the command effectiveness. Once the error has been nullified, the spacecraft starts following the reference trajectory and small corrections are required. The final position error is in the order of 200 meters, and the speed error is in the order of 0.3 meters per seconds.

As shown in Figure 60, the de-orbiting control command has low amplitude oscillations, and the spacecraft follows the reference trajectory with a position error at 150 km in the order of hundreds of meters.

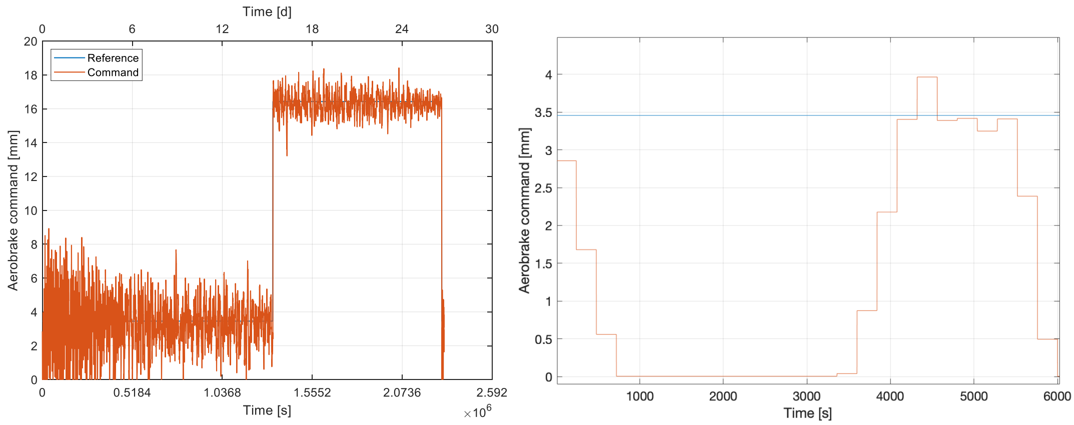


Figure 60 - De-orbiting phase, Simulation #3: (left) Aero-brake control command (blue) and aero-brake command of the reference trajectory (red). (right) Zoom of the first 6000 seconds.

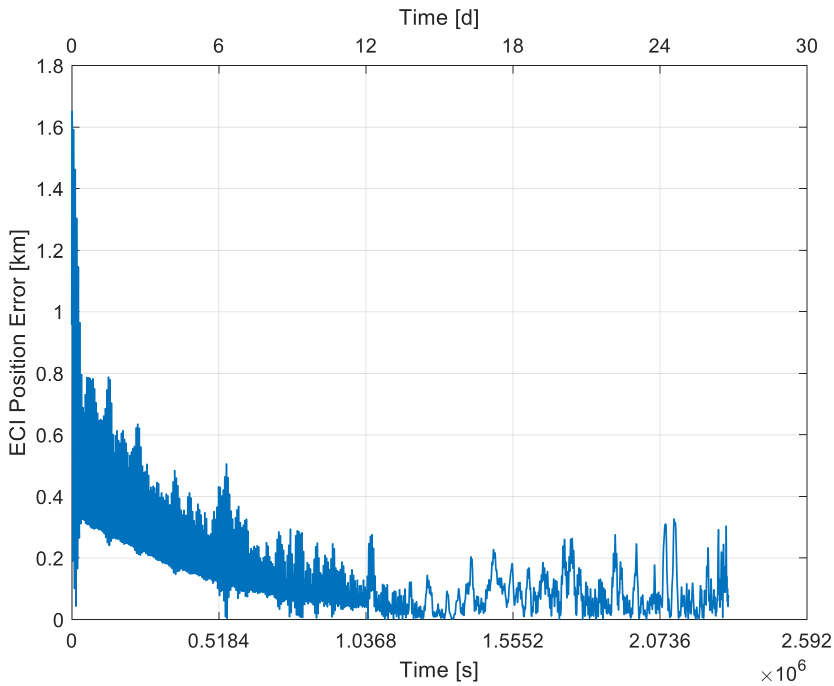


Figure 61 – De-orbiting phase, Simulation #3: ECI position error.

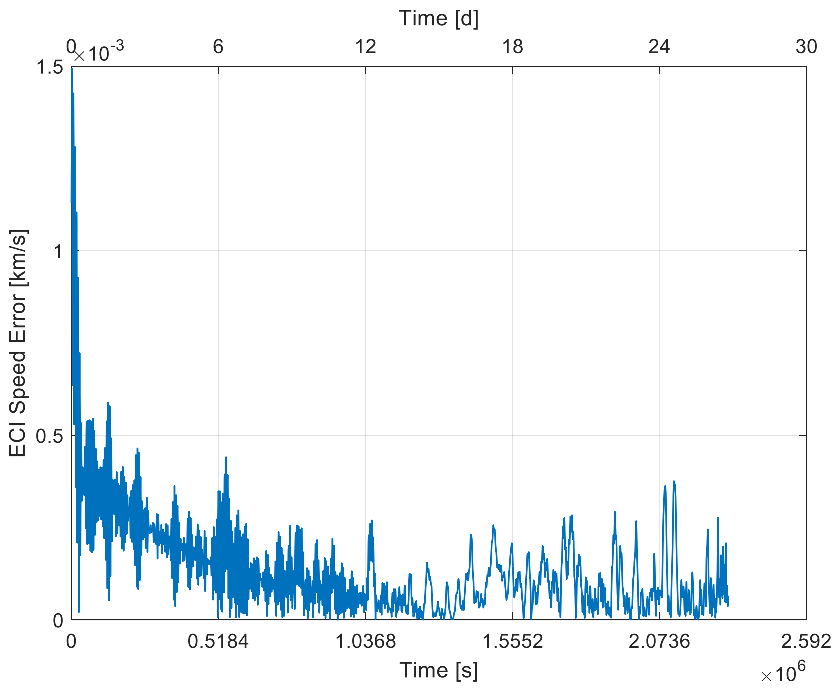


Figure 62 - De-orbiting phase, Simulation #3: ECI speed error.

Finally, a statistical analysis has been conducted in the results of 100 simulations with different combination of uncertainties, whose characteristics are described at the beginning of this paragraph. Table 34 summarizes the results of this simulation campaign. It turns out that the proposed technique can guarantee a mean ECI position error of about 400 meters with a standard deviation of about 600 meters, and a mean ECI velocity error of about 0.5 meter per seconds with a standard deviation of about 0.7 meter per seconds. Finally, the distance from the target point on the Earth surface is reports (mean value of 530 meters with a standard deviation of 860 meters).

Table 34 - De-orbiting phase: Statics of position and velocity errors at HR = 150 km evaluated over 100 simulations.

Statistical Analysis: Errors @ H_R		
	Mean	Standard Deviation
ECI Position Error $E^{@H_R}_{\|\boldsymbol{R}_{ECI}\|}$	0.4 km	0.64 km
ECI Speed Error $E^{\backslash @H_R}_{\|\boldsymbol{V}_{ECI}\|}$	0.49 m/s	0.73 m/s

Distance from target point on the Earth surface	$0.53\ km$	$0.86\ km$

7. Conclusions

This thesis presented the development and performance assessment of original autonomous GNC functions for operations in close proximity of uncooperative space targets and re-entry applications.

<u>Relative Navigation architecture for spacecraft close proximity operations</u>

In the frame of ADR and IOS operations, a LIDAR based relative navigation architecture able to deal with non-cooperative and partially unknown target has been proposed for both monitoring and final approach phase. The architecture has a multi-steps configuration which allows to separate the inertia properties estimation task from the relative navigation task, thus avoiding the influence of a time-varying uncertainty on the inertial properties on the estimate of the relative motion parameters when they are included in the state vector of the filter.

Also, a loosely coupled configuration has been preferred to a tightly coupled one for two main reasons. On one hand, a loosely coupled approach allows realizing a modular architecture in which different algorithmic approaches can be plugged-in to adapt it to the operational scenario without significantly affecting its structure. For instance, one can change the relative navigation sensor with a monocular camera without modifying the process or measurement model of the navigation filters, as long as the algorithms aims at processing the raw data of the camera provides an estimate of the pose parameters. On the other hand, the loosely coupled configuration allows avoiding dealing with the recursive appearance/disappearance of the target features from the FOV of the sensor which requires the development of robust feature detection algorithm.

The robustness of the overall architecture is provided by an autonomous failure detection strategy in which the measured poses are fed to the Kalman filter only if the associated error metric function of the ICP is lower than a selected threshold.

The performance assessment has been carried out by means of numerical simulations considering targets with different characteristics in terms of size, shape and orbital/rotational dynamics. For all the considered test cases, the diagonal elements of the MIRs matrix were estimated with a maximum error of 6% and a best error of 0.07%. Results also showed that the accuracy level in the estimation of the off-diagonal elements of the MIRs matrix does not have a noticeable impact on the target-chaser relative state estimation performance.

Exploiting the estimated moments of inertia ratios, the Kalman Filter, initialized with highly accurate pose estimates provided by the previous step, allows obtaining a millimeter level of accuracy in the relative position, a millimeter per second level of accuracy in the relative velocity, while the relative attitude and the target absolute angular velocity are estimated with 10^{-1} degrees and 10^{-4} degrees per seconds level of accuracy, respectively. It is also worth noting how the filtering scheme is effective in improving the performance in the estimation of the relative state parameters with respect to the pose determination algorithms in standalone configuration. This is also due to the capability of obtaining highly accurate estimate of the inertia ratios to be used within the filter dynamics model, thanks to the multi-step approach.

The attitude and moment of inertia estimation algorithm has been validated within an experimental set-up which, by means of a spherical air-bearing, simulates the tumbling motion of an uncooperative space target represented by a scaled down satellite mock-up, whose motion is tracked with a solid-state LIDAR. The presented test case shows that the ICP allows estimating the attitude with a mean error of 5° and a standard deviation of 2°, while the principal moment of inertia can be estimated with an error lower than 20%. However, it is worth pointing out that this accuracy is influenced by all the uncertainties of the set-up which includes the residual gravity torque effect due to a non-perfect alignment of the center of mass and the center of rotation of the system, the extrinsic calibration error and the intrinsic accuracy of the PhaseSpace Impulse system which represents the benchmark. Also, it is important to underline that the pose has been tracked with a low-cost solid-state LIDAR whose performance are not comparable with the recent technological solution for space applications.

With regards to the final approach phase, in which the chaser has to capture the target by means a robotic arm, a trade-off analysis on the relative navigation architecture has been carried out to evaluate advantages and disadvantages of the state-of-the-art solutions. The analysis shows that an additional EO sensor (a TOF camera in this case) shall be foreseen on the end-effector of the robotic arm in order to have direct measurements of the pose of the end-effector with respect to the selected grasping point on the target which is independent from the ego-motion uncertainty of the robotic arm. These measurements, along with the pose of the end-effector with respect to its base (the chaser), are processed by a navigation filter, separate from the one which estimate the target-chaser motion parameters. This architecture has been tested within a numerical simulation environment in which the final approach phase toward ENVISAT of a chaser equipped with a 7-DOF robotic arm. The numerical results shows that the architecture is able to reach the same level of accuracy of the monitoring phase for the target-chaser motion estimation parameters despite the additional uncertainties considered in this test case (e.g., the uncertainty in the knowledge of the absolute navigation state of the chaser). Also. relative position and relative velocity of the gripper with respect to the grasping point are estimated with millimeter and centimeter per second level of accuracy, respectively; while the error level achieved in the relative attitude and relative angular velocity estimate are 10^{-2} degree and 10^{-3} degree per second.

The future of this activity regards the introduction of a shape reconstruction step which allows to validate the target model on board the chaser to check for possible hit or collision that could have modified the target geometry. From the point of view of the performance assessment, the real time implementation of the relative navigation function can be verified with Software in the Loop Tests and Hardware in the Loop Tests (HIL). The HIL Tests will require an experimental set-up which is able to simulate the full 6-DOF relative motion between two spacecraft.

Aerodynamic de-orbit control

Regarding the trajectory tracking control of a de-orbiting micro-satellite by means of aerodynamic drag, an LQR-based control system has been developed and tested in a numerical simulation environment in the framework of the activities related to the MISTRAL mission. Specifically, the control system must determine the aperture of an umbrella-like actuator, which has the twofold function of a thermal shield and aero-braking device, that minimizes the deviation of the spacecraft from a reference trajectory.

The robustness of the control algorithm has been assessed in presence of different sources of uncertainties, including those related to the atmosphere, to the knowledge of the spacecraft parameters, the limitation of its actuation systems (e.g., saturation and duty cycle) and the perturbation on the initial state. The simulation campaign shows that the spacecraft reaches the atmospheric interface at 150 kilometers with a hundred of kilometers level position error and meters per second in speed error. Specifically, it can be noted that the initial state perturbation is the one that most stresses the actuator in terms of oscillation amplitude: it causes the system to reach the saturation value multiple times. Even if this is not ideal from a mechanical point of view, the numerical simulations shows that the final accuracy reached at the end of the de-orbiting phase is not affected by this phenomenon. To this aim, future works will include the development of a trajectory generation algorithm which takes into account the actuation system limitations as well as an improved control algorithm which optimize the control effort to be exerted the umbrella-like actuator by considering the actuator's physical constraints.

References

[1] H. Klinkrad, *Space Debris*. 2010. doi: 10.1002/9780470686652.eae325.

[2] R. A. Williamson, "Assuring the sustainability of space activities," *Space Policy*, vol. 28, no. 3, pp. 154–160, 2012, doi: https://doi.org/10.1016/j.spacepol.2012.06.010.

[3] D. Kessler, N. Johnson, J.-C. Liou, and M. Matney, "The Kessler Syndrome: Implications to Future Space operations," *Advances in the Astronautical Sciences*, vol. 137, Jan. 2010.

[4] J. Drmola and T. Hubik, "Kessler Syndrome: System Dynamics Model," *Space Policy*, vol. 44–45, pp. 29–39, 2018, doi: https://doi.org/10.1016/j.spacepol.2018.03.003.

[5] H. Klinkrad *et al.*, "The ESA Space Debris Mitigation Handbook 2002," *Advances in Space Research*, vol. 34, no. 5, pp. 1251–1259, 2004, doi: https://doi.org/10.1016/j.asr.2003.01.018.

[6] J.-C. Liou, N. L. Johnson, and N. M. Hill, "Controlling the growth of future LEO debris populations with active debris removal," *Acta Astronaut*, vol. 66, no. 5, pp. 648–653, 2010, doi: https://doi.org/10.1016/j.actaastro.2009.08.005.

[7] J.-C. Liou, "An active debris removal parametric study for LEO environment remediation," *Advances in Space Research*, vol. 47, no. 11, pp. 1865–1876, 2011, doi: https://doi.org/10.1016/j.asr.2011.02.003.

[8] J.-C. Liou and N. L. Johnson, "A sensitivity study of the effectiveness of active debris removal in LEO," *Acta Astronaut*, vol. 64, no. 2, pp. 236–243, 2009, doi: https://doi.org/10.1016/j.actaastro.2008.07.009.

[9] C. Bonnal, J.-M. Ruault, and M.-C. Desjean, "Active debris removal: Recent progress and current trends," *Acta Astronaut*, vol. 85, pp. 51–60, 2013, doi: https://doi.org/10.1016/j.actaastro.2012.11.009.

[10] R. Opromolla, G. Fasano, G. Rufino, and M. Grassi, "A review of cooperative and uncooperative spacecraft pose determination techniques for close-proximity operations," *Progress in Aerospace Sciences*, vol. 93, pp. 53–72, 2017, doi: https://doi.org/10.1016/j.paerosci.2017.07.001.

[11] V. Pesce, M. Lavagna, and R. Bevilacqua, "Stereovision-based pose and inertia estimation of unknown and uncooperative space objects," *Advances in Space Research*, vol. 59, no. 1, pp. 236–251, 2017, doi: 10.1016/j.asr.2016.10.002.

[12] R. Opromolla and A. Nocerino, "Uncooperative spacecraft relative navigation with LIDAR-based Unscented Kalman Filter," *Submitted to IEEE, under review*, 2019.

[13] M. Nimelman, J. Tripp, G. Bailak, and J. Bolger, "Spaceborne scanning lidar system (SSLS)," in *Proc.SPIE*, May 2005, vol. 5798. doi: 10.1117/12.604203.

[14] M. Horsley, S. Nikolaev, and A. Pertica, "Small Satellite Rendezvous Using Differential Lift and Drag," *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 2, pp. 445–453, Jan. 2013, doi: 10.2514/1.57327.

[15] R. Bevilacqua and M. Romano, "Rendezvous Maneuvers of Multiple Spacecraft Using Differential Drag Under J2 Perturbation," *Journal of Guidance, Control, and Dynamics*, vol. 31, no. 6, pp. 1595–1607, Nov. 2008, doi: 10.2514/1.36362.

[16] N. X. Vinh, J. R. Johannesen, K. D. Mease, and J. M. Hanson, "Explicit guidance of drag-modulated aeroassisted transfer between elliptical orbits,"

Journal of Guidance, Control, and Dynamics, vol. 9, no. 3, pp. 274–280, May 1986, doi: 10.2514/3.20103.

[17] C. L. Leonard, W. M. Hollister, and E. V Bergmann, "Orbital formationkeeping with differential drag," *Journal of Guidance, Control, and Dynamics*, vol. 12, no. 1, pp. 108–113, Jan. 1989, doi: 10.2514/3.20374.

[18] R. RAVINDRAN and P. C. HUGHES, "Optimal Aerodynamic Attitude Stabilization of Near-Earth Satellites," *J Spacecr Rockets*, vol. 9, no. 7, pp. 499–506, Jul. 1972, doi: 10.2514/3.61730.

[19] D. Akin, "The Parashield Entry Vehicle Concept: Basic Theory and Flight Test Development," 1990.

[20] M. Wiegand and H. Konigsmann, "A small re-entry capsule-brem-sat 2," 1996.

[21] E. Venkatapathy *et al.*, "Adaptive Deployable Entry and Placement Technology (ADEPT): A Feasibility Study for Human Missions to Mars," in *21st AIAA Aerodynamic Decelerator Systems Technology Conference and Seminar*, American Institute of Aeronautics and Astronautics, 2011. doi: doi:10.2514/6.2011-2608.

[22] E. Bassano *et al.*, "IRENE--Italian re-entry nacelle for microgravity experiments," in *62nd International Astronautical Congress, Cape Town (South Africa)*, 2011.

[23] A. Fedele and S. Mungiguerra, "Aerodynamics and flight mechanics activities for a suborbital flight test of a deployable heat shield capsule," *Acta Astronaut*, vol. 151, pp. 324–333, 2018, doi: https://doi.org/10.1016/j.actaastro.2018.05.044.

[24] D. Wilde, S. Walther, K. Pitchadze, S. Alexsaschkin, D. Vennemann, and L. Marraffa, “Flight test and ISS application of the Inflatable Reentry and Descent Technology (IRDT),” *Acta Astronaut*, vol. 51, no. 1, pp. 83–88, 2002, doi: https://doi.org/10.1016/S0094-5765(02)00078-4.

[25] S. Hughes *et al.*, “Inflatable Re-entry Vehicle Experiment (IRVE) Design Overview,” in *18th AIAA Aerodynamic Decelerator Systems Technology Conference and Seminar*, American Institute of Aeronautics and Astronautics, 2005. doi: doi:10.2514/6.2005-1636.

[26] V. Carandente, R. Savino, V. D’Oriano, and R. Fortezza, “A Study on Earth Re-entry Capsules with Deployable Aerobrakes for Recoverable Microgravity Experiments,” *Microgravity Sci Technol*, vol. 27, no. 3, pp. 181–191, 2015, doi: 10.1007/s12217-015-9430-4.

[27] S. Sharma and S. D’Amico, “Neural Network-Based Pose Estimation for Noncooperative Spacecraft Rendezvous,” *IEEE Trans Aerosp Electron Syst*, p. 1, 2020, doi: 10.1109/TAES.2020.2999148.

[28] S. Segal, A. Carmi, and P. Gurfil, “Stereovision-Based Estimation of Relative Dynamics Between Noncooperative Satellites: Theory and Experiments,” *Control Systems Technology, IEEE Transactions on*, vol. 22, pp. 568–584, Mar. 2014, doi: 10.1109/TCST.2013.2255288.

[29] J. O. Woods and J. A. Christian, “Lidar-based relative navigation with respect to non-cooperative objects,” *Acta Astronaut*, vol. 126, pp. 298–311, 2016, doi: https://doi.org/10.1016/j.actaastro.2016.05.007.

[30] O. Kechagias-Stamatis, N. Aouf, and M. A. Richardson, “High-speed multi-dimensional relative navigation for uncooperative space objects,” *Acta*

Astronaut, vol. 160, pp. 388–400, 2019, doi: https://doi.org/10.1016/j.actaastro.2019.04.050.

[31] H. Gómez Martínez, G. Giorgi, and B. Eissfeller, "Pose estimation and tracking of non-cooperative rocket bodies using Time-of-Flight cameras," *Acta Astronaut*, vol. 139, pp. 165–175, 2017, doi: https://doi.org/10.1016/j.actaastro.2017.07.002.

[32] T. Tzschichholz, T. Boge, and K. Schilling, "Relative pose estimation of satellites using PMD-/CCD-sensor data fusion," *Acta Astronaut*, vol. 109, pp. 25–33, 2015, doi: https://doi.org/10.1016/j.actaastro.2014.12.010.

[33] R. Volpe, M. Sabatini, and G. B. Palmerini, "Pose and Shape Reconstruction of a Noncooperative Spacecraft Using Camera and Range Measurements," *International Journal of Aerospace Engineering*, vol. 2017, 2017, doi: 10.1155/2017/4535316.

[34] J. L. Forshaw *et al.*, "RemoveDEBRIS: An in-orbit active debris removal demonstration mission," *Acta Astronaut*, vol. 127, pp. 448–463, 2016, doi: https://doi.org/10.1016/j.actaastro.2016.06.018.

[35] G. S. Aglietti *et al.*, "The active space debris removal mission RemoveDebris. Part 2: In orbit operations," *Acta Astronaut*, vol. 168, pp. 310–322, 2020, doi: https://doi.org/10.1016/j.actaastro.2019.09.001.

[36] G. Welch, G. Bishop, and others, "An introduction to the Kalman filter," 1995.

[37] M. I. Ribeiro, "Kalman and extended kalman filters: Concept, derivation and properties," *Institute for Systems and Robotics*, vol. 43, p. 46, 2004.

[38] E. A. Wan and R. Van Der Merwe, "The unscented Kalman filter for nonlinear estimation," in *Proceedings of the IEEE 2000 Adaptive Systems for Signal*

Processing, Communications, and Control Symposium (Cat. No.00EX373), 2000, pp. 153–158. doi: 10.1109/ASSPCC.2000.882463.

[39] R. E. Mortensen, “Maximum-likelihood recursive nonlinear filtering,” *J Optim Theory Appl*, vol. 2, no. 6, pp. 386–394, 1968, doi: 10.1007/BF00925744.

[40] D. Simon, *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley \& Sons, 2006.

[41] M. S. Grewal and A. P. Andrews, “Applications of Kalman Filtering in Aerospace 1960 to the Present [Historical Perspectives],” *IEEE Control Systems Magazine*, vol. 30, no. 3, pp. 69–78, 2010, doi: 10.1109/MCS.2010.936465.

[42] X. Wang, Z. Wang, and Y. Zhang, “Stereovision-based relative states and inertia parameter estimation of noncooperative spacecraft,” *Proc Inst Mech Eng G J Aerosp Eng*, vol. 0, no. 109, pp. 1–14, 2018, doi: 10.1177/0954410018782021.

[43] X. Hou, C. Ma, Z. Wang, and J. Yuan, “Adaptive pose and inertial parameters estimation of free-floating tumbling space objects using dual vector quaternions,” *Advances in Mechanical Engineering*, vol. 9, no. 10, pp. 1–17, 2017, doi: 10.1177/1687814017714210.

[44] P. Gurfil and K. v. Kholshevnikov, “Manifolds and metrics in the relative spacecraft motion problem,” *Journal of Guidance, Control, and Dynamics*, vol. 29, no. 4, pp. 1004–1010, 2006, doi: 10.2514/1.15531.

[45] K. Yamanaka and F. Ankersen, “New State Transition Matrix for Relative Motion on an Arbitrary Elliptical Orbit,” *Journal of Guidance, Control, and Dynamics*, vol. 25, no. 1, pp. 60–66, Jan. 2002, doi: 10.2514/2.4875.

[46] W. H. CLOHESSY and R. S. WILTSHIRE, “Terminal Guidance System for Satellite Rendezvous,” *Journal of the Aerospace Sciences*, vol. 27, no. 9, pp. 653–658, Sep. 1960, doi: 10.2514/8.8704.

[47] Q. Meng, J. Liang, and O. Ma, *Estimate of All the Inertial Parameters of a Free-Floating Object in Orbit*. 2018. doi: 10.2514/6.2018-1606.

[48] Q. Meng, C. Zhao, H. Ji, and J. Liang, “Identify the full inertial parameters of a non-cooperative target with eddy current detumbling,” *Advances in Space Research*, 2020, doi: https://doi.org/10.1016/j.asr.2020.05.044.

[49] O.-O. Christidi-Loumpasefski and E. Papadopoulos, “Parameter identification of a space object in the pre-capture phase,” in *Proc. Int. Symp. Artif. Intell. Robot. Autom. Space*, 2018.

[50] T. P. Setterfield, D. W. Miller, A. Saenz-Otero, E. Frazzoli, and J. J. Leonard, “Inertial Properties Estimation of a Passive On-orbit Object Using Polhode Analysis,” *Journal of Guidance, Control, and Dynamics*, vol. 41, no. 10, pp. 2214–2231, Jun. 2018, doi: 10.2514/1.G003394.

[51] D. Sheinfeld and S. Rock, “Rigid body inertia estimation with applications to the capture of a tumbling satellite,” in *Proceedings of 19th AAS/AIAA spaceflight mechanics meeting*, 2009, vol. 134, pp. 343–356.

[52] H. Benninghoff and T. Boge, *Rendezvous Involving a Non-Cooperative, Tumbling Target - Estimation of Moments of Inertia and Center of Mass of an Unknown Target*. 2015.

[53] H. Yu, X. Zhang, L. Liu, S. Wang, and S. Song, “Relative dynamics estimation of non-cooperative spacecraft with unknown orbit elements and inertial tensor,”

Chinese Journal of Aeronautics, vol. 29, no. 2, pp. 479–491, 2016, doi: https://doi.org/10.1016/j.cja.2016.01.013.

[54] B. Tweddle, "Computer vision-based localization and mapping of an unknown, uncooperative and spinning target for spacecraft proximity operations," Mar. 2014.

[55] B.-Z. Zhou, G.-P. Cai, Y.-M. Liu, and P. Liu, "Motion prediction of a non-cooperative space target," *Advances in Space Research*, vol. 61, no. 1, pp. 207–222, 2018, doi: https://doi.org/10.1016/j.asr.2017.10.028.

[56] D. Ivanov, M. Ovchinnikov, and M. Sakovich, "Relative Pose and Inertia Determination of Unknown Satellite Using Monocular Vision," *International Journal of Aerospace Engineering*, Sep. 2018, doi: 10.1155/2018/9731512.

[57] F. Aghili and K. Parsa, "Motion and Parameter Estimation of Space Objects Using Laser-Vision Data," *Journal of Guidance Control and Dynamics - J GUID CONTROL DYNAM*, vol. 32, pp. 538–550, Mar. 2009, doi: 10.2514/1.37129.

[58] M. Lichter and S. Dubowsky, *State, Shape, and Parameter Estimation of Space Objects from Range Images.* 2004. doi: 10.1109/ROBOT.2004.1307513.

[59] Y. Li, Y. Wang, and Y. Xie, "Using consecutive point clouds for pose and motion estimation of tumbling non-cooperative target," *Advances in Space Research*, vol. 63, no. 5, pp. 1576–1587, 2019, doi: https://doi.org/10.1016/j.asr.2018.11.024.

[60] P. Colmenarejo *et al.*, "Methods and outcomes of the COMRADE project-Design of robust Combined control for robotic spacecraft and manipulator in

servicing missions: comparison between between Hinf and nonlinear Lyapunov-based approaches," 2018.

[61] X. Du, Y. He, L. Chen, and S. Gao, "Pose estimation of large non-cooperative spacecraft based on extended PnP model," in *2016 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2016, pp. 413–418. doi: 10.1109/ROBIO.2016.7866357.

[62] R. Opromolla, G. Fasano, G. Rufino, and M. Grassi, "Pose Estimation for Spacecraft Relative Navigation Using Model-Based Algorithms," *IEEE Trans Aerosp Electron Syst*, vol. Vol. 53, pp. 431–447, Jan. 2017, doi: 10.1109/TAES.2017.2650785.

[63] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 2, no. 1, pp. 37–52, 1987, doi: https://doi.org/10.1016/0169-7439(87)80084-9.

[64] S. Ruel, T. Luu, M. Anctil, and S. Gagnon, *Target Localization from 3D data for On-Orbit Autonomous Rendezvous & Docking*. 2008. doi: 10.1109/AERO.2008.4526516.

[65] R. Opromolla, G. Fasano, G. Rufino, and M. Grassi, "Uncooperative pose estimation with a LIDAR-based system," *Acta Astronaut*, vol. 110, pp. 287–297, 2015, doi: https://doi.org/10.1016/j.actaastro.2014.11.003.

[66] T. Boge, S. Tuttas, and U. Stilla, "Fast Initial Pose Estimation of Spacecraft from LiDAR Point Cloud Data," May 2015.

[67] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Trans Pattern Anal Mach Intell*, vol. 14, no. 2, pp. 239–256, 1992, doi: 10.1109/34.121791.

[68] W. H. Press and S. A. Teukolsky, “Savitzky-Golay Smoothing Filters,” *Computers in Physics*, vol. 4, no. 6, pp. 669–672, Nov. 1990, doi: 10.1063/1.4822961.

[69] F. L. Markley, “Attitude representations for Kalman Filtering,” *Advances in the Astronautical Sciences*, vol. 109 I, no. March 2003, pp. 133–151, 2002, doi: 10.2514/2.5048.

[70] J. M. Galante, J. Van Eepoel, C. D’Souza, and B. Patrick, “Fast Kalman filtering for relative spacecraft position and attitude estimation for the raven ISS hosted payload,” *Advances in the Astronautical Sciences*, vol. 157, pp. 179–196, 2016.

[71] M. Kaplan, “Modern spacecraft dynamics and control,” 2020, Accessed: Jan. 11, 2023. [Online]. Available: https://books.google.it/books?hl=it&lr=&id=aF0EEAAAQBAJ&oi=fnd&pg=PA1&dq=Modern+spacecraft+dynamics+and+control&ots=qSj5YWEBab&sig=P-4kT7bwzR3ndpcQJdIsHndIH-8

[72] R. Opromolla, G. Rufino, G. Fasano, and M. Grassi, “LIDAR-based autonomous pose determination for a large space debris,” in *Proceedings of the International Astronautical Congress, IAC*, Oct. 2014, vol. 2, pp. 1512–1520.

[73] G. Euliss, A. Christiansen, and R. Athale, “Analysis of laser-ranging technology for sense and avoid operation of unmanned aircraft systems: The tradeoff between resolution and power,” *Proc. SPIE*, vol. 6962, Apr. 2008, doi: 10.1117/12.777562.

[74] J. Christian and S. Cryan, *A Survey of LIDAR Technology and its Use in Spacecraft Relative Navigation*. 2013. doi: 10.2514/6.2013-4641.

[75] Y.-L. Desnos, C. Buck, J. Guijarro, J.-L. Suchail, R. Torres, and E. Attema, "ASAR - Envisat's advanced synthetic aperture radar," *ESA Bulletin*, vol. 102, pp. 91–100, Jan. 2000.

[76] B.F. Laguane, "Determination of the attitude state of space debris objects based on Satellite Laser Ranging".

[77] E. A. Spatial, "Envisat-1," 1998.

[78] R. Opromolla, G. Fasano, G. Rufino, and M. Grassi, "Design of relative trajectories for in orbit proximity operations," *Acta Astronaut*, vol. 145, pp. 342–356, 2018, doi: https://doi.org/10.1016/j.actaastro.2018.01.062.

[79] A. Rhodes, J. Christian, and T. Evans, "A Concise Guide to Feature Histograms with Applications to LIDAR-Based Spacecraft Relative Navigation," *J Astronaut Sci*, vol. 64, Jan. 2017, doi: 10.1007/s40295-016-0108-y.

[80] J. Guo *et al.*, "Real-time measurement and estimation of the 3D geometry and motion parameters for spatially unknown moving targets," *Aerosp Sci Technol*, vol. 97, p. 105619, 2020, doi: https://doi.org/10.1016/j.ast.2019.105619.

[81] "Cosmos-1, 3, 3M and 3MU | SL-8 | C-1." https://www.russianspaceweb.com/cosmos3.html (accessed Jan. 12, 2023).

[82] T. Peters and D. Escorial Olmos, "COBRA contactless detumbling," *CEAS Space Journal*, vol. 8, Apr. 2016, doi: 10.1007/s12567-016-0116-6.

[83] "ESA - Small Geostationary Satellite (SGEO)." https://www.esa.int/Applications/Telecommunications_Integrated_Applications/Small_Geostationary_Satellite_SGEO (accessed Jan. 12, 2023).

[84] “Earth Observation Missions - eoPortal.” https://www.eoportal.org/ (accessed Jan. 12, 2023).

[85] “Hispasat 36W-1.” https://www.hispasat.com/en/satellite-fleet/hispasat-36w-1 (accessed Jan. 12, 2023).

[86] “Ray casting for deformable triangular 3D meshes - File Exchange - MATLAB Central.” https://www.mathworks.com/matlabcentral/fileexchange/41504-ray-casting-for-deformable-triangular-3d-meshes (accessed Jan. 12, 2023).

[87] “https://www.intelrealsense.com/lidar-camera-l515/.”

[88] K. Saulnier, D. Pérez, R. C. Huang, D. Gallardo, G. Tilton, and R. Bevilacqua, “A six-degree-of-freedom hardware-in-the-loop simulator for small spacecraft,” *Acta Astronaut*, vol. 105, no. 2, pp. 444–462, 2014, doi: https://doi.org/10.1016/j.actaastro.2014.10.027.

[89] B. K. P. Horn, “Closed-form solution of absolute orientation using unit quaternions,” *Journal of the Optical Society of America A*, vol. 4, no. 4, pp. 629–642, 1987, doi: 10.1364/JOSAA.4.000629.

[90] “https://dev.intelrealsense.com/docs/projection-texture-mapping-and-occlusion-with-intel-realsense-depth-cameras#6-appendix-1---distortion-models.”

[91] J. L. Schwartz, M. A. Peck, and C. D. Hall, “Historical Review of Air-Bearing Spacecraft Simulators,” *Journal of Guidance, Control, and Dynamics*, vol. 26, no. 4, pp. 513–522, Jul. 2003, doi: 10.2514/2.5085.

[92] G. A. Dukeman and M. W. Gallaher, “Guidance and Control Concepts for the X-33 Technology Demonstrator,” 1998.

[93] B. B. Chirayath and G. R. Bindu, “Longitudinal guidance and control of reentry vehicle in the approach and landing phase,” in *2014 International Conference on Power Signals Control and Computations (EPSCICON)*, 2014, pp. 1–5. doi: 10.1109/EPSCICON.2014.6887480.

[94] A. Ohara, Y. Yamaguchi, and T. Morito, “LPV modeling and gain scheduled control of re-entry vehicle in approach and landing phase,” in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, doi: 10.2514/6.2001-4038.

[95] A. Fujimori, F. Terui, and P. N. Nikiforuk, “Flight Control Design of an Unmanned Space Vehicle Using Gain Scheduling,” *Journal of Guidance, Control, and Dynamics*, vol. 28, no. 1, pp. 96–105, Jan. 2005, doi: 10.2514/1.5144.

[96] C. O. Min, D. W. Lee, K. R. Cho, S. J. Jo, J. S. Yang, and W. B. Lee, “Control of approach and landing phase for reentry vehicle using fuzzy logic,” *Aerosp Sci Technol*, vol. 15, no. 4, pp. 269–282, Jun. 2011, doi: 10.1016/J.AST.2010.07.006.

[97] F. Jouhaud, “Fuzzy logic control — application to the guidance of a re-entry space capsule,” *Acta Astronaut*, vol. 46, no. 8, pp. 507–517, Apr. 2000, doi: 10.1016/S0094-5765(00)00017-5.

[98] J. C. Harpold and D. E. Gavert, “Space Shuttle entry guidance performance results,” *https://doi.org/10.2514/3.8523*, vol. 6, no. 6, pp. 442–447, May 2012, doi: 10.2514/3.8523.

[99] W. Grimm, J. G. van der Meulen, and A. J. Roenneke, “Update Scheme for Drag Reference Profiles in an Entry Guidance,” *https://doi.org/10.2514/2.5123*, vol. 26, no. 5, pp. 695–701, May 2012, doi: 10.2514/2.5123.

[100] S. E. Talole, J. Benito, and K. D. Mease, “Sliding mode observer for drag tracking in entry guidance,” *Collection of Technical Papers - AIAA Guidance, Navigation, and Control Conference 2007*, vol. 5, pp. 5122–5137, 2007, doi: 10.2514/6.2007-6851.

[101] A. J. Roenneke, “Adaptive on-board guidance for entry vehicles,” *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2001, doi: 10.2514/6.2001-4048.

[102] K. D. Mease, D. T. Chen, P. Teufel, and H. Schönenberger, “Reduced-Order Entry Trajectory Planning for Acceleration Guidance,” *https://doi.org/10.2514/2.4906*, vol. 25, no. 2, pp. 257–266, May 2012, doi: 10.2514/2.4906.

[103] A. Cavallo and F. Ferrara, “Atmospheric re-entry control for low lift/drag vehicles,” *https://doi.org/10.2514/3.21578*, vol. 19, no. 1, pp. 47–53, May 2012, doi: 10.2514/3.21578.

[104] A. Fedele, S. Omar, S. Cantoni, R. Savino, and R. Bevilacqua, “Precise re-entry and landing of propellantless spacecraft,” *Advances in Space Research*, vol. 68, no. 11, pp. 4336–4358, Dec. 2021, doi: 10.1016/J.ASR.2021.09.029.

[105] S. Omar and R. Bevilacqua, “Guidance, navigation, and control solutions for spacecraft re-entry point targeting using aerodynamic drag,” *Acta Astronaut*, vol. 155, pp. 389–405, Feb. 2019, doi: 10.1016/J.ACTAASTRO.2018.10.016.

[106] A. Fedele, S. Carannante, M. Grassi, and R. Savino, “Aerodynamic Control System for a Deployable Re-entry Capsule,” *Acta Astronaut*, vol. 181, pp. 707–716, Apr. 2021, doi: 10.1016/J.ACTAASTRO.2020.05.049.

[107] P. Lu, “Entry Guidance and Trajectory Control for Reusable Launch Vehicle,” *https://doi.org/10.2514/2.4008*, vol. 20, no. 1, pp. 143–149, May 2012, doi: 10.2514/2.4008.

[108] J. Benito and K. D. Mease, “Nonlinear predictive controller for drag tracking in entry guidance,” *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, 2008, doi: 10.2514/6.2008-7350.

[109] L. Blasi, E. D’Amato, M. Mattei, and I. Notaro, “Re-entry trajectory tracking control of a micro-satellite with a deployable front structure,” *7th International Conference on Control, Decision and Information Technologies, CoDIT 2020*, pp. 697–702, Jun. 2020, doi: 10.1109/CODIT49905.2020.9263953.

[110] J. Virgili, P. C. E. Roberts, and N. C. Hara, “Atmospheric Interface Reentry Point Targeting Using Aerodynamic Drag Control,” *https://doi.org/10.2514/1.G000884*, vol. 38, no. 3, pp. 403–413, Feb. 2015, doi: 10.2514/1.G000884.

[111] R. Gifty, U. P. Rajeev, V. R. Lalithambika, and M. v. Dhekane, “Entry guidance with smooth drag planning and non-linear tracking,” *Control Eng Pract*, vol. 63, pp. 24–33, Jun. 2017, doi: 10.1016/J.CONENGPRAC.2017.03.014.

[112] R. R. da Costa, Q. P. Chu, and J. A. Mulder, “Reentry Flight Controller Design Using Nonlinear Dynamic Inversion,” *https://doi.org/10.2514/2.3916*, vol. 40, no. 1, pp. 64–71, May 2012, doi: 10.2514/2.3916.

[113] D. Guglielmo *et al.*, “Drag deorbit device: A new standard reentry actuator for cubeSats,” *J Spacecr Rockets*, vol. 56, no. 1, pp. 129–145, Sep. 2019, doi: 10.2514/1.A34218/ASSET/IMAGES/LARGE/FIGURE26.JPEG.

[114] S. R. Omar and R. Bevilacqua, “Hardware and GNC solutions for controlled spacecraft re-entry using aerodynamic drag,” *Acta Astronaut*, vol. 159, pp. 49–64, Jun. 2019, doi: 10.1016/J.ACTAASTRO.2019.03.051.

[115] S. Dutta, A. L. Bowes, A. D. Cianciolo, C. E. Glass, and R. W. Powell, “Guidance scheme for modulation of drag devices to enable return from low earth orbit,” *AIAA Atmospheric Flight Mechanics Conference, 2017*, 2017, doi: 10.2514/6.2017-0467.

[116] B. S. Kumar, A. Ng, K. Yoshihara, and A. de Ruiter, “Differential drag as a means of spacecraft formation control,” *IEEE Aerospace Conference Proceedings*, 2007, doi: 10.1109/AERO.2007.352790.

[117] R. P. Patera, “Drag modulation as a means of mitigating casualty risk for random reentry,” *Collection of Technical Papers - AIAA Atmospheric Flight Mechanics Conference*, vol. 2, pp. 1169–1178, 2005, doi: 10.2514/6.2005-6228.

[118] S. R. Omar and J. M. Wersinger, “Satellite formation control using differential drag,” *53rd AIAA Aerospace Sciences Meeting*, 2015, doi: 10.2514/6.2015-0002.

[119] J. Virgili, P. C. E. Roberts, and N. C. Hara, “Atmospheric Interface Reentry Point Targeting Using Aerodynamic Drag Control,” *https://doi.org/10.2514/1.G000884*, vol. 38, no. 3, pp. 403–413, Feb. 2015, doi: 10.2514/1.G000884.

[120] S. A. Schweighart and R. J. Sedwick, “Development and analysis of a high fidelity linearized J2 model for satellite formation flying,” *AIAA Space 2001 Conference and Exposition*, 2001, doi: 10.2514/6.2001-4744.

[121] A. E. Bryson, *Control of spacecraft and aircraft*, vol. 41. Princeton university press Princeton, 1993.

[122] "OEM719 | NovAtel." https://novatel.com/products/receivers/gnss-gps-receiver-boards/oem719 (accessed Jan. 23, 2023).

List of publications

Journal papers

- R. Opromolla, A. Nocerino, "Uncooperative spacecraft relative navigation with LIDAR-based unscented Kalman filter." *IEEE Access*, vol. 7, pp. 180012-180026, 2019. DOI: 10.1109/ACCESS.2019.2959438.
- A. Nocerino, R. Opromolla, G. Fasano, M. Grassi, “LIDAR-based multi-step approach for relative state and inertia parameters determination of an uncooperative target.” *Acta Astronautica*, vol. 181, pp. 662-678, 2021. DOI: 10.1016/j.actaastro.2021.02.019.
- R. Opromolla, C. Vela, A. Nocerino, C. Lombardi, “Monocular-Based Pose Estimation Based on Fiducial Markers for Space Robotic Capture Operations in GEO.” *Remote Sensing.* vol. 14, 4483, 2022. DOI: 10.3390/rs14184483.
- E. D’Amato, I. Notaro, G. Panico, L. Blasi, M. Mattei, A. Nocerino, “Trajectory Planning and Tracking for a Re-Entry Capsule with a Deployable Aero-Brake.” *Aerospace*, vol. 9, 841, 2022. DOI: 10.3390/aerospace9120841.
- A. Nocerino, I. Notaro, G. Morani, M. Poderico, E. D'Amato, L. Blasi, A. Fedele, R. Fortezza, M. Grassi, M. Mattei, “Trajectory control algorithms for the de-orbiting and Re-entry of the MISTRAL satellite.” Acta Astronautica, vol. 203, pp. 392-406, 2023. DOI: 10.1016/j.actaastro.2022.12.010.
- F. Basana, Z. Pavanello, F. Branz, A Francesconi, G Borelli, D. Invernizzi, M. Massari, M. Lovera, A. Nocerino, R. Opromolla, I. Huertas Garcia and P Simplicio, “GNC and robotic arm control for spacecraft close-proximity operations”, Multibody Sistem Dynamics, under review.

- A. Nocerino, R. Opromolla, G. Fasano, M. Grassi, "LIDAR parameters selection and chaser attitude control for relative navigation with non-cooperative space target." IEEE Trnasactions on Aerospace and Electronic Systems, under review.
- G. Napolano, C. Vela, A. Nocerino, R. Opromolla, M. Grassi, "A multi-sensor optical relative navigation system for small satellite system", Acta Astronautica., accepted for publication.
- A. Nocerino, K. E. Larsen, R. Bevilacqua, E. Jerome, "Stereoscopic-Based Mass Properties Estimation for Warhead Fragments", submitted as Technical Note to AIAA journal, under review.
- A. Nocerino, R. Opromolla, G. Fasano, M. Grassi, P. Balaguer, S. Jhon, H. Cho, R. Bevilacqua, "Experimental validation of inertia parameters and attitude estimation of uncooperative space target using solid state LIDAR", Acta Astronautica, in-press.

Conference papers

- A. Nocerino, R. Opromolla, M. Grassi, G. Fasano, G. Rufino, "Lidar-Based Relative Navigation In Close-Proximity Of Uncooperative Space Target", in Proceedings of the AIDAA XXV International Conference, Rome, Italy. (2019). (speaker).
- A. Nocerino, R. Opromolla, M. Grassi, "Robust relative navigation scheme for the autonomous flight in close proximity of an uncooperative space target", in Proceedings of 2nd IAA Conference on Space Situational Awareness, Washington D.C., USA. (2020). (speaker).

- A. Nocerino, R. Opromolla, G. Fasano, M. Grassi, “Analysis of LIDAR-based relative navigation performance during close-range rendezvous toward an uncooperative space target”, in Proceedings of 2020 IEEE International Workshop on Metrology for Aerospace, (2020). (speaker).
- A. Nocerino, N. Cimmino, R. Opromolla, G. Fasano, M. Grassi, “Analysis of manoeuvring strategies and relative navigation for multi-step rendezvous toward uncooperative targets”, in Proceedings of 71st International Astronautical Congress, (2020). (speaker).
- A. Nocerino, R. Opromolla, G. Fasano, M. Grassi “LIDAR pointing and parameters control for close proximity operations with uncooperative target”, in Proceedings of 2021 IEEE International Workshop on Metrology for Aerospace, (2021). (speaker).
- L. Falanga, G. Caso, A. Ciccarelli, F. Cuomo, S. De Masi, F. Della Vecchia, D. Di Maio, S. Gala, S. Piatti, A. Saveriano, M. Scamardella, D. Scilla, R. Tarone, A. Renga, G. Isoletta, A. Nocerino. “A Microsatellite mission to a long-period comet in a heliocentric inner orbit”, in Proceedings of the AIDAA XXVI International Conference, (2021)
- V. Porrino, A. Argirò, M. Chiariello, A. Contaldi, A. Cristiano, L. D’Albenzio, B. De Leonardis, C. Di Mauro, F. J. Fernandez, M. Gargiulo, R. Guida, C. Ibello, G. Montano, L. Zennamo, G. Isoletta, A. Nocerino, A. Renga. “SCORE: a scientific mission for the exploration of a long-period comet with micro/nano satellite”, in Proceedings of the AIDAA XXVI International Conference. (2021)
- R. Opromolla, F. Branz, A. Francesconi, A. Cenedese, R. Antonello, P. Iob, Z. Pavanello, D. Vertuani, M. Grassi, G. Fasano, A. Nocerino, C. Lombardi, C.

Vela, M. Massari, C. Colombo, M. Lovera, D. Invernizzi, P. Ghignoni, L. M. Ticozzi, G. Borelli, I. Huertas, P. Simplicio. "Chaser-robotic arm combined control and optical relative navigation for space target capture", in Proceedings of the AIDAA 2021 XXVI International Conference, Rome, Italy.

- Z. Pavanello, F. Branz, A. Francesconi, A. Cenedese, R. Antonello, F. Basana, P. Iob, D. Vertuani, M. Massari, C. Colombo, M. Lovera, D. Invernizzi, P. Ghignoni, L. Ticozzi, G. Borelli, R. Opromolla, M. Grassi, G. Fasano, A. Nocerino, C. Lombardi, C. Vela, I. Huertas, P. Simplicio. "Combined control and navigation approach to the robotic capture of space vehicles", in Proceedings of 72st International Astronautical Congress, (2021).
- C. Vela, G. Napolano, A. Nocerino, R. Opromolla, M. Grassi, M. Manzo, s. Amoruso, G. Di Donfrancesco, A Relative Navigation Module Based on Visual and Ranging Measurements for CubeSat Applications. In *IEEE 9th International Workshop on Metrology for AeroSpace (MetroAeroSpace)* (pp. 305-310). (2022).
- A. Nocerino, R. Opromolla, G. Fasano, M. Grassi, H. Cho, R. Bevilacqua. "Experimental validation of inertia parameters and attitude estimation of uncooperative space target using solid state LIDAR", in Proceedings of 73rd International Astronautical Congress, (2022). (speaker).
- A. Nocerino, S. Piatti, F. Saggiomo, M. Grassi, G. Fasano, R. Opromolla, C. Schmitt. Numerical And Experimental Validation of Lidar- Based Template Matching Algorithms For Non- Cooperative Spacecraft Pose Initialization, in Proceedings of 33rd Space Flight Mechanics Meeting, Austin, Texas. (2023). (speaker).

- J. Jordan, D. Zuehlke, D. Posada, A. Nocerino, P. Fontdegloria, J. Spencer, A. Malik, R. Bevilacqua, T. Henderson. Estimation of Uncooperative Satellite Inertia Parameters for Space Debris Removal Using PSO, accepted at 2023 IEEE Aerospace Conference, Big-Sky, Montana. (2023).

Acknowledgments

Now I would like to save some space to thank the people who contributed to make these last four years a special adventure!

First and foremost, thanks to my family, Mum and Dad who have always supported my choices, and Andrea who taught me that, despite all the difficulties, you always must follow your dreams.

There are no words to say how grateful I am to my supervisors, Prof. Roberto Opromolla, Prof. Giancarmine Fasano and Prof. Michele Grassi. Working together has been an amazing experience which made me grow as a researcher... and as a person. It would have not been possible without the friendly and comfortable environment you have created through the years.

Roberto, thanks for being an amazing mentor, for your unwavering support and GUIDANCE on every step of the way. Your passion about research has been an inspiration for me. Thanks to you I learnt a lot ... except for grasping your quotes.

Giancarmine, thanks for all your precious advice and for providing a reliable NAVIGATION aid along the way. Thanks for sharing with me your enthusiasm for astrodynamics… and for rock n' roll. On the second one we still have a lot of work to do!

Michele, thanks for giving me the opportunity to start this journey and for teaching me the art of self-CONTROL.

A special thanks to Prof. Maria Daniela Graziano and Prof. Alfredo Renga for our insightful discussions during lunchbreaks.

Dani, thanks for passing down to me your way of thinking out of the box. During these years we found out that we are of the same crazy minds.

Alfredo, thanks for giving me the chance of having my first teaching experience. Your t(h)rust and support have been priceless.

I cannot but thanks my colleagues who have become more than friends. Giorgio, Marco, Flavia, Federica, Giuseppe, Claudio (and his twins), Enrico, Paolo, Nicola, Tobia, and all the other guys of the GNC lab team. I was very lucky to be part of such an incredible team. You are very special people. Sharing laughs (and burden) has been the main mean of survival throughout these years. I'll cherish the time we spent together in and out of the lab.

I would also like to thank Prof. Riccardo Bevilacqua for giving me the opportunity to join his team at Embry-Riddle Aeronautical University for a semester and all the ADAMUS lab team for the support they provide to my research activities.

A final thanks to all the people I met along the way. Everyone one of you have concurred in shaping my future!