M. Maestrini, M.A. De Luca, P. Di Lizia
Relative Navigation Strategy About Unknown and Uncooperative Targets
Journal of Guidance Control and Dynamics, published online 01/06/2023
doi:10.2514/1.G007337

Relative Navigation Strategy About Unknown and Uncooperative Targets

Michele Maestrini*, Maria Alessandra De Luca† and Pierluigi Di Lizia‡
Polytechnic University of Milan, Via La Masa 34, 20156, Milan (MI), Italy.

In recent years, space debris has become a threat for satellites operating in Low Earth Orbit. Even by applying mitigation guidelines, their number will still increase over the course of the century. As a consequence, active debris removal missions and on-orbit servicing missions have gained momentum at both academic and industrial level. The crucial step in both scenarios is the capability of navigating in the neighborhood of a target resident space object. This problem has been tackled many times in literature with varying level of cooperativeness of the target required. While several techniques are available when the target is cooperative or its shape is known, no approach is mature enough to deal with uncooperative and unknown targets. This paper proposes a hybrid method to tackle this issue called Coarse Model-Based Relative Navigation: CoMBiNa. The main idea of this algorithm is to split the mission in two phases. During the first phase, the algorithm constructs a coarse model of the target. In the second phase, this coarse model is used as a reference for a relative navigation technique, effectively shifting the focus towards state and inertia estimation. In addition, this paper proposes a strategy to leverage the structure of the selected navigation method to detect and reject outliers. To conclude, CoMBiNa is tested on a simulated environment to highlight its benefits and its shortcomings, while also assessing its applicability on a limited resource single board computer.

I. Introduction

Over the past few decades, space debris has become a threat for space operations. The average rate of fragmentation and collisions occurring in orbit yearly causes a continuous increase in the number of debris and in probability of further collisions [1]. The problem of space debris is also strictly related to the ever-increasing number of launched satellites that terminate their operational lifetime. According to the Inter-Agency Space Debris Coordination Committee (IADC)[2] the debris situation in Low Earth Orbit (LEO) may be reaching a catastrophic tipping point, where collisions

*Postdoctoral Research Fellow, Department of Aerospace Science and Technology, michele.maestrini@polimi.it, Corresponding Author.
†Graduate Student, Department of Aerospace Science and Technology, mariaalessandra.deluca@mail.polimi.it.
‡Associate Professor, Department of Aerospace Science and Technology, pierluigi.dilizia@polimi.it, AIAA Member.
Part of this work was presented as Paper No. AAS 2022-2384 at the AIAA 2022 SciTech Forum, January 3 - 7, 2022, San Diego, California, and virtual.

between objects will cause a cascade of further collisions. This phenomenon (also known as Kessler Syndrome [3]) will lead to an exponential increase of the number of space debris even without launching new objects. This catastrophic scenario would prevent the use of any LEO. According to six different models of debris population developed by independent entities (e.g. NASA, ASI, ESA, ISRO, JAXA, and UKSA) the number of objects in critical LEO and GEO regions is expected to grow by $\sim 30\%$ in the next two centuries even with the mitigation of new debris creation [4]. As a consequence, universities, research centers, as well as public agencies have been investing in concepts to tackle this issue on two different levels. The first approach is to perform missions of Active Debris Removal (ADR) aimed at deorbiting debris which are already in orbit. However, the methods to tackle this problem are still under development and are far from the needed level of maturity in terms of both Technology Readiness Level and experience: the first ever mission of this kind is expected to launch in 2025*. Secondly, by mitigating the effect of the ever increasing number of decommissioned objects with the Space Debris Mitigation Compliance Verification Guidelines†. Among the many possible ways to mitigate debris creation, On-Orbit Servicing (OOS) seems to be the most promising one. OOS has been attracting funds from space agencies (e.g. NASA's OSAM-1, ESA's e.Deorbit, and ADRIOS). In addition, OOS is also interesting from a commercial point of view as it is expected to have ~ 4.5 billion USD cumulative revenue by 2028‡. This expected market is owed to the capability of OOS to mitigate the downsides in the design criteria employed in modern day space industry such as the application of high redundancy (which adds to complexity, mass, and cost), the adoption of solutions with high Technology Readiness Level slowing down innovation, and long operational lives (effectively operating the satellite with obsolescent hardware for the majority of its lifetime). OOS could alleviate these issues by providing additional flexibility in design and operations [5]. At the moment, most of the envisioned projects rely on public-private partnerships (e.g. NASA's Brane Craft, DARPA's RSGS, and ESA's RemoveDEBRIS). However, some missions are already undergoing development with fully private funds (e.g. MEV-1§ and ELSA-D [6]). Currently, the main driver for OOS investment is the maintenance of the GEO fleet¶, whose construction and operation constitutes a multi-billion dollar business [7]. Nonetheless, the application to LEO environment constitutes a completely untapped, but promising, market. Indeed, with the imminent advent of mega-constellations, the number of target resident space objects (RSO) for OOS may grow exponentially [8].

In order to perform the high-level tasks required for OOS, specific navigation techniques have to be designed in order to provide the relative state information necessary to guarantee safe operations. In past missions and studies on OOS it has been customary to assume complete knowledge of the shape and parameters of the target RSO. Guidance and navigation techniques for such scenarios were tailored to one specific task, and relied heavily on the available information. Unless

*https://www.esa.int/Safety_Security/Clean_Space/ESA_commissions_world_s_first_space_debris_removal (Accessed on September 27, 2021)

†https://copernicus-masters.com/wp-content/uploads/2017/03/ESSB-HB-U-002-Issue119February20151.pdf (Accessed on September 27, 2021)

‡https://apnews.com/press-release/pr-globenewswire/d875b64e496603d79012f86d4716eeed (Accessed on September 27, 2021)

§https://news.northropgrumman.com/news/releases/intelsat-901-satellite-returns-to-service-using-northrop-grummans-mission-extension-vehicle (Accessed on May 27, 2021)

¶https://www.infiniteorbits.io/missions (Accessed on May 27, 2021)

the target ceased operations due to a malfunction or damage, it was also customary to assume that proximity operations could rely on active radio communications coming from the RSO. However, more complex OOS and ADR scenarios may require different levels of cooperativeness, even including cases of an uncooperative target. For the purpose of this study, an uncooperative target will be defined as an RSO which was not designed for ADR (i.e. lacks specific markers) and does not communicate through radio link information about its state. Moreover, the RSO may also be unknown. In this work an object is considered unknown if no information about its shape and inertia parameters are available a priori. However, it is assumed that its orbital information will be available from orbit determination processes conducted on ground. An RSO may be unknown for many different reasons, which may be as simple as the optical degradation of surfaces due to the prolonged exposure to the space environment. Besides, more drastic changes to the RSO shape can happen due to a change in configuration and damages (e.g. owed to collision and/or explosion of internal tanks). Such changes cannot be foreseen and could compromise the mission outcome of any chaser tailored to act in proximity of a specific object. In fact, there is a substantial difference between performing OOS (or ADR) on a spacecraft which was designed for this scope, and performing it on a generic debris that is already on orbit. This difference has a major impact on the needed technologies, guidance approaches, and navigation instruments. This lack of information requires the inspecting spacecraft to be equipped with a higher level of autonomy, which would allow the flexibility to deal with unexpected changes in the RSO, while also relieving burden on human operators, avoiding human-induced errors, and granting operations even outside visibility windows of the ground stations. The required paradigm shift towards execution of autonomous proximity operations has already been envisioned by NASA as a key enabling technology for the development of next generation space missions [9]. Up until this point, all previously flown missions and experiments either relied on the assumption that the target is cooperative (e.g. ETSVII [10], OE [11], and DART [12]), or they leverage their knowledge of the target's shape to recover its pose (e.g. PRISMA [13], and Seeker-1 [14]). Hence, none of the available algorithms are capable of dealing with an unknown and uncooperative RSO. Therefore, the aim of this work is to fill this gap in literature by developing a tool that is general enough to be applied to any RSO in LEO.

In order to deal with an unknown and tumbling RSO, it is not sufficient to retrieve the relative state and attitude between chaser and target with a navigation filter. It is also necessary to recover inertia properties of this target. Despite being applicable to ADR and OOS scenarios, the solution to this kind of problem for proximity operations constitutes a relevant milestone for a plethora of other missions such as Formation Flying or asteroid exploration.

This work proposes a navigation paradigm designed to achieve state and inertia estimation in the neighborhood of an unknown and uncooperative RSO called Coarse Model-Based Relative Navigation: CoMBiNa. The main idea of CoMBiNa is to split the navigation algorithm into two phases. The preliminary phase is dedicated to the construction of the geometrical shape of the target. While the second phase is the operational one where state and inertia are recovered through a sequential filtering approach. After a brief review of the methods used to perform relative navigation in literature, provided in Section II, this paper gives an outline of the entire CoMBiNa pipeline in Section III. Subsequently,

Section VI presents the numerical results of the two phases of the algorithm. Section VII introduces the outcome of a feasibility study on a Single Board Computer (SBC) with realistic tumbling rates estimated from light curve observations. To conclude, some final remarks are highlighted in Section VIII.

II. Literature Review

The main driver for the selection of a relative navigation algorithm is the expected level of cooperativeness of the target. Several different navigation techniques can deal with known and cooperative targets with increasing level of cooperation required, starting with the basic case of a fully cooperative RSO providing information regarding its state [15]. In more general scenarios, the target is passively cooperating thanks to the presence of a small number of easily identifiable fiducial markers on its surface, which can be combined with the a priori knowledge of the target stored on-board to retrieve pose information [16]. Such algorithms leverage different techniques to estimate the pose (e.g., Perspective-n-Point (PnP) [17], TRIAD [18], QUEST [19], etc.). Their main downside is the detection and matching of markers from measurements: a step which is often taken for granted despite severely impairing convergence if the wrong correspondences are found [20]. The aforementioned knowledge of the shape, size, and location of the markers which is typical of cooperative approaches has been employed extensively in flown missions (e.g. PSX sensor of the ETS-VII mission [10], AVGS sensor of the OE and DART missions [11, 12], VBS sensor of the PRISMA mission [13]). By completely removing any form of cooperation from the RSO, proximity operations can still be conducted with model-based techniques that exploit knowledge of the target model to retrieve the relative pose. Several techniques have been developed, which exploit either image features (e.g., SoftPOSIT [21] and RANSAC [22]), or a database of information stored on-board (e.g., Template Matching [23]). In particular, this latter approach struggles to be compliant with on-board resources due to database storage and its computationally heavy search. This problem has been tackled in recent years by applying Template Matching datasets to Artificial Intelligence approaches through the use of Convolutional Neural Networks [20, 24–26]. Alternatively, pose can be determined by taking the target geometrical characteristics into account during the design process [27]. Finally, among the most general solutions, one can rely on point cloud registration methods [28] with Iterative Closest Point (ICP) being the most common approach in literature[29]. The techniques usually adopted for pose determination in the neighborhood of known but uncooperative targets are closely related to those needed to deal with a passively cooperative target (i.e. has markers). In fact, some of the methods can be used in both scenarios. However, these approaches also share the downsides of passively cooperative techniques. As a consequence, finding correct correspondences between raw data and the on-board model is once again the most crucial aspect. This problem is exacerbated by the absence of fiducial markers specifically designed to be recognized in raw measurements, hence rendering extracted and matched features less reliable and much more numerous. A practical application of such methods has been deployed on the Shuttle's RNS sensor used for the HST servicing [30]. For a thorough overview of relative navigation methods exploiting monocular images of uncooperative

objects, the reader may refer to [31].

To conclude, all the navigation methods presented so far fall short in case the model of the RSO under investigation is not known a priori, which is the case considered in this paper. By removing both assumptions of cooperation and knowledge, the chaser has to operate by computing its trajectory while simultaneously building a model of the RSO. The state of the art approach to tackle this problem is represented by Simultaneous Localization and Mapping (SLAM) [32]. Several examples exploiting particle filters and Extended Kalman Filter are available in literature [33–35]. However most of them assume perfect features matching between on-board map and measures, which is instead one of the key problems of SLAM techniques. Filter updates work best with few accurate matched points rather than with many imprecise measurements. Moreover, the computational cost of these methods owed to features matching and filter steps (i.e. update and propagation) grows with the size of the map, hence requiring a trade-off between computational cost and accuracy of the reconstruction. Graph-SLAM approaches constitute an alternative take to this problem, where online navigation estimates are sacrificed in order to achieve better mapping of the environment and better convergence. Examples of these approaches have been implemented in literature [36, 37], most notably in [38], where the author proposed a smoothing approach in his PhD dissertation.

To overcome limitations of the presented state of the art methods, this work proposes a hybrid approach for relative navigation at an unknown and uncooperative target RSO called Coarse Model Based Relative Navigation (CoMBiNa). The main idea of CoMBiNa is to combine the advantages of SLAM and model-based techniques by splitting the mission in two phases. During the first phase, a SLAM approach can be used on a safe relative orbit [39] to acquire a point cloud and construct a coarse model of the target RSO exploiting state of the art graph-SLAM techniques (e.g., ORBSLAM [36]) to obtain a larger map. In the second phase, the coarse model is used as the base of a model-based relative navigation technique. The proposed approach relies on the extraction of relative pose information from the process of point set registration. To do so, it is necessary to be able to retrieve 3D information (in the form of point clouds) at each measurement time. Any instrument capable of providing 3D point clouds could work in this framework (e.g., RGB-D, LIDAR), however stereo cameras are characterized by lower hardware complexity, cost, and power consumption with respect to LIDARs, they can be used for human-in-the-loop operations (e.g., space situational awareness activities), and they provide denser point clouds. The main disadvantage of stereo cameras is their high sensitivity to illumination conditions, which can be compensated for by increasing the power budget and including light sources in the measurement systems. To obtain the needed visual acquisitions, stereo camera images are synthetically generated in Blender [‖] and processed to retrieve point clouds. At each measurement step, the point cloud extracted from the stereo camera is matched to the coarse on-board model with a point set registration technique that provides the relative pose. The selected registration method is called Bayesian Coherent Point Drift (BCPD)[40], which is a generalization of the ICP algorithm that matches the point clouds in a statistical sense, hence overcoming the aforementioned issue (typical of

[‖] https://www.blender.org/

model based-method) of finding exact features correspondences. Compared with ICP, this algorithm is more robust against noisy data and avoids being trapped in local minima for bad initialization. This method guarantees convergence in a finite amount of iterations thanks to Bayesian Inference and can exploit code accelerations to treat larger point clouds [28]. In this second phase, the focus is shifted from constructing the RSO model to providing relative state predictions and estimating the inertia parameters of the target. To do so, the relative pose generated with BCPD is treated as a measurement in an Unscented Kalman Filter (UKF) [41] which recovers pose and inertia ratios similarly to [38].

III. CoMBiNa: Coarse Model-Based Relative Navigation

This section provides an overview of the proposed method for navigation at an unknown and uncooperative RSO. The CoMBiNa method can only be started once the chaser is close enough to the target and with sufficiently accurate estimate of the relative position. In this work it was assumed that these initial conditions can be achieved with another purposely defined relative navigation approach based, for example, on line-of-sight optical measurements as was demonstrated in the AVANTI experiment [42]. In this study, the chaser was delivered within 50 meters from the target with a meter level relative navigation accuracy. After the target initial conditions for CoMBiNa have been achieved, the proposed algorithm can start. During the first phase of CoMBiNa, a SLAM approach can be used on a relative orbit to acquire a point cloud and construct a coarse model of the target RSO while guaranteeing passive safety. These trajectories are characterized by an out-of-plane elliptical periodic relative motion such that the traced path never crosses the velocity vector of the primary object. In a safe relative trajectory, drift of the two spacecraft, due to relative state estimation errors will not result in collision, so the trajectory is considered passively safe [39].

The main focus of this phase is placed on the model construction and can be performed with state of the art graph-SLAM techniques [36]. Such approaches produce large and accurate maps of the target RSO, however they may not be suitable for online implementation in delicate environments, as these techniques are based on a smoothing approach rather than on a filtering one: neither state estimate nor dynamic parameters are updated between time steps. Without a continuous update of these information it would be impossible to close the GNC loop during nominal operations. The outcome of the first phase provides a coarse and noisy point cloud, which can be processed to filter outliers and produce a model of the RSO. In the second phase, this coarse model is used in a model-based relative navigation technique. This approach exploits 3D point clouds acquired from the processing of stereo images and a point set registration method (i.e. BCPD) to estimate the relative position and attitude of the target. In this second phase, the focus is shifted from constructing the RSO model to providing relative state predictions and estimating the inertia parameters of the target via an UKF. Fig. 1 provides an overview of all the building blocks necessary to carry out this process. The next sections provide better insight in each of the three main building blocks of the algorithm.

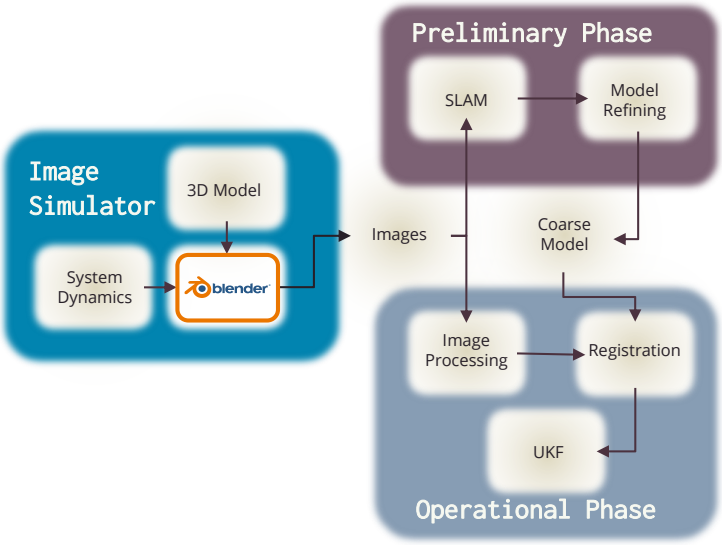

Fig. 1 Pipeline of the entire CoMBiNa approach.

A. Blender Simulated Environment

The main aim of this first block is to obtain the images which need to be supplied to the SLAM algorithm and to the navigation filter. This simulated environment remains the same during the second operational phase. The images are simulated using Blender and an open source 3D model of the target RSO[**]. Despite the existence of rendering softwares specific to the aerospace industry (e.g., ESA's PANGU, and Airbus SurRender), the choice fell on Blender as it provides an open source program with a huge community, has very good modelling capabilities, and can be controlled through Python APIs (e.g., see [43, 44]). Regardless of the RSO model used, Blender is capable of producing its rendering by simulating a camera moving in its proximity. In particular, the rendering engine Cycles is exploited, as it provides a physics-based ray tracer that computes light paths from all the sources in the scene as well as their interactions with the objects. This allows simulating the parallel light rays' interaction coming from the simulated Sun with the target RSO's surface. Fig. 2 shows the working principle of the simulated Blender environment. In particular, relative pose between target and chaser which is output of the dynamics can be used to compute the relative camera pose. Notice that this data generation step from a simulated environment is only necessary in case a test facility is not available.

The first step to retrieve measurements from images consists in extracting features from simulated images and recovering their position in the chaser body frame. In order to treat images as efficiently as possible, only the region of interest (ROI) of the image containing the target is analyzed. For the purpose of this study, a simple metric to extract the region of interest base on a minimum pixel threshold value was used. This metric is sufficient for this test case which has no background, however, it has been thoroughly proven in literature that this task can be robustly and efficiently performed through Artificial Intelligence techniques [20, 24].

[**]https://sketchfab.com/3d-models/envisat (accessed on 27 September 2021)

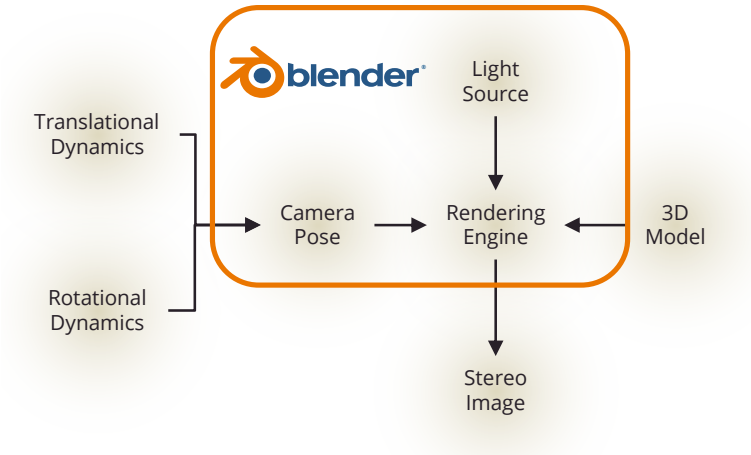

Fig. 2 Flowchart describing the rendering module.

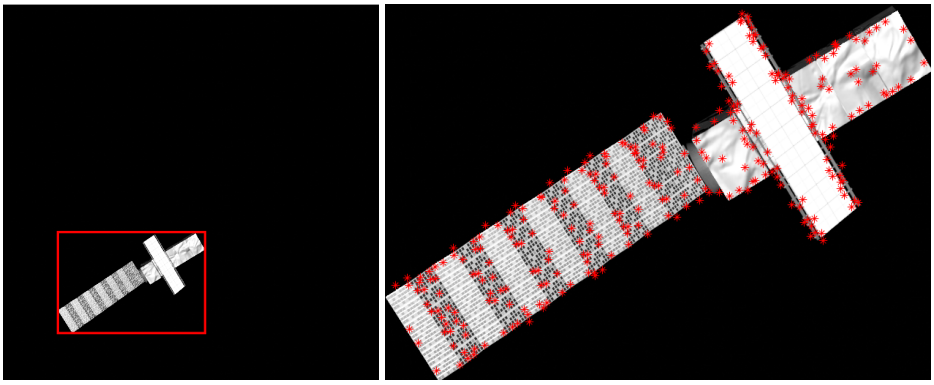

Fig. 3 Images processing step: object extraction as a ROI (left), and extracted SURF features in this ROI (right). Reproduced from [45].

After obtaining a ROI, SURF features are extracted from it [46]. Several other features have been tested before selecting SURF (i.e., BRISK, FAST, Harris, KAZE, MSER, ORB, and SIFT): in the end particular choice of SURF was driven by the need of providing the largest number of uniformly distributed image features, so that the overall shape of the target can be captured. These features should not recognize a few, robust, stable features to use in a PnP like approach, but rather provide a distribution of features capturing the overall shape of the target. Since CoMBiNa has no precise model of the RSO available, any attempt at computing exact matches between image features and coarse model is destined to fail. Matching the model to the 3D point cloud in a statistical sense with a statistical point set registration algorithm could provide the desired outcome. In particular, the more points and the more uniformly distributed on the target they are in a 3D point cloud, the better the registration results will be. This approach is very similar to a normalized cross correlation approach, which has already been used for relative navigation [47]. The main difference is that this algorithm tries to exploit 3D information instead of 2D images. An example of the image processing step is reported in Fig. 3. When the pixel location of the features has been identified, the disparity map can be queried in order to retrieve

their corresponding depth information. The stereo camera model assumed in this work is represented in Fig. 4 and it is characterized by a focal length f, a pixel density d (i.e. number of pixels per unit length of the sensor), and the pixel coordinates of the optical center (i.e. the central pixel in the image) u_0 and v_0. The stereo camera setup is also characterized by a quantity b, which is called the baseline of the stereo camera and represents the shift along one of the camera axis applied to the second sensor of the stereo setup.

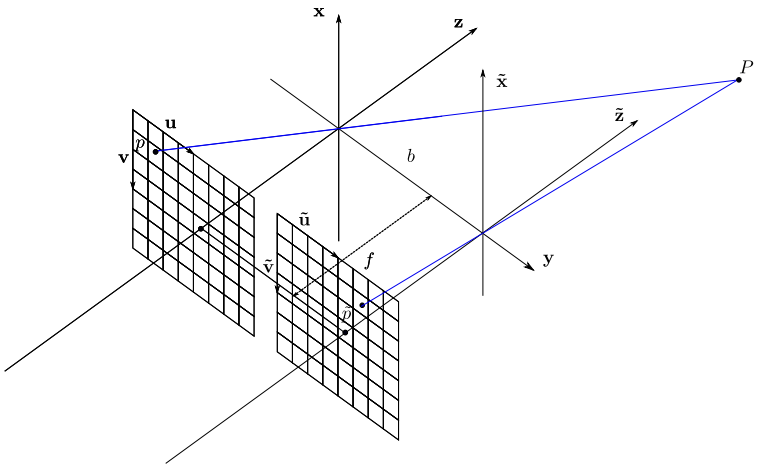

Fig. 4 Parallel axis stereo camera model adopted in this work. Reproduced from [45].

Once the 3D coordinates of the i-th extracted features are known in terms of horizontal pixel coordinate u_i, vertical pixel coordinate v_i, and disparity α_i, the 3D position in the chaser body frame can be retrieved exploiting the measurement inversion equation:

$$\mathbf{p}_i^C = \left[-\frac{b}{\alpha_i}(v_i - v_0)\,,\ \frac{b}{\alpha_i}(u_i - u_0)\,,\ -\frac{f \cdot d \cdot b}{\alpha_i} \right]^\top \tag{1}$$

In this notation, the subscript of the point $\mathbf{p}$ indicates the i-th extracted feature and the superscript C indicates the center of the reference frame (i.e. the chaser in this case). A summary of the camera parameters adopted in this study is reported in Table 1. These values were inspired by [37], although a lower pixel density was adopted.

Table 1 Camera parameters adopted in this study.

Parameter name	Value
Focal length f	34 [mm]
Central pixel coordinates $[u_0, v_0]$	$[0, 0]$ [pix, pix]
Pixel density d	48 [pix/mm]
Baseline b	1 [m]
Sensor size	1024×1024 [pix×pix]

At the end of the image processing step, the 3D point cloud of features in the chaser fixed (i.e. camera fixed) reference frame, centered in the chaser itself, which will be referred to as $\mathcal{P}^C = \{\mathbf{p}_1^C, \ldots, \mathbf{p}_M^C\} \subset \mathbb{R}^{3\times M}$, is available for

registration with the coarse model stored on-board.

B. Preliminary Analysis Phase

During the preliminary phase, the chaser remains on a safe relative orbit in the neighborhood of the target RSO. For the purpose of this study, it is assumed that this orbit is perfectly tracked during this first phase, so that images of the target can be acquired safely and used in a SLAM algorithm. This assumption is acceptable as many different techniques for line-of-sight navigation based on brightest pixel are available [42]. The main difficulty in case of a drift during relative motion would be to keep the target in the camera field of view during coasting phases. To do so, a method similar to [48] based on a succession of control and coast arcs could be executed. By selecting an appropriate relative drift orbit reference [49] showed that it is possible to determine a camera orientation that guarantees that the target stays inside the camera field of view. In the cases investigated it is assumed that the chaser periodically controls its attitude whenever the target exits its field of view by centering the brightest pixel. During the first part of the preliminary phase, a SLAM method is applied to build a large, coarse map of the target RSO. The time to be dedicated to this phase is strongly related to the attitude regime of the target and the illumination conditions. In this work it was assumed that a full model would be needed to carry on with the operational phase. In fact, reference [40] showed that using a partial model for registration may severely impact the accuracy of the matches. It may be possible to carry on operations with a partial model of the target provided that the operational measurements are also collected from the same relative poses, however this scenario was not tackled in this paper in order to see if this approach could work in ideal conditions. In this work, a single relative orbit proved to be sufficient for the full construction of the target model. However, the duration and relative trajectory necessary to this aim may vary. For further details on this topic the readers may refer to the analysis in [50], where a strategy for the selection of best inspection maneuvers with a fixed time horizon was presented. The SLAM technique investigated to build the coarse target model is the open source ORB-SLAM [36], which has already been used for real images acquired in space [37]. This system uses ORB features, which are robust to rotation, scale, and illumination changes. Moreover they are fast to extract and match, which allows for real-time operation [51]. ORB-SLAM still relies on standard image processing routines which can be time consuming. Moreover it is still an approach based on feature matching, therefore it may be hindered by poor illumination conditions and repetitive textures. Nevertheless, this approach demonstrates impressive mapping results, and its ability to operate online on minimal hardware makes it particularly attractive for limited resources systems. Once the SLAM algorithm run concludes, the coarse point cloud needs to be processed in order to remove outliers. The approach exploited in this work for point cloud processing is the probabilistic feature-based space carving method proposed in [52], which relies on a two step method. First, outliers are detected with a k-NNG [53] approach. The main outlier types are owed to points in regions where the local point density is low relative to the mean point density of the map, and points whose mean viewing direction deviates substantially from that of other neighboring points. The second processing step is the probabilistic,

feature-based, space-carving approach [54]. This method uses 3D spatial information provided by the ORB-SLAM algorithm to remove spurious features based on free-space constraints between the point and the camera. To carry out this carving strategy, a Delaunay triangulation is initially produced out of the available mapped 3D points. Afterwards, the built 3D tetrahedra are carved, so that their faces are eliminated if they violate visibility constraints. To summarize the process, each triangular facet of a tetrahedron is taken to be an estimate of an element, and feature points are assumed to be observations of surface elements corrupted by Gaussian noise with a given variance along the visibility direction, centered at the surface of a facet. Given this assumption, this process can be easily explained by looking at Fig. 5. In fact, if the observed feature is located further behind the facet (e.g. p_2), the probability of existence of the facet decreases. While the further ahead of the facet the observed feature is (e.g. p_1), the higher the probability that the facet exists.

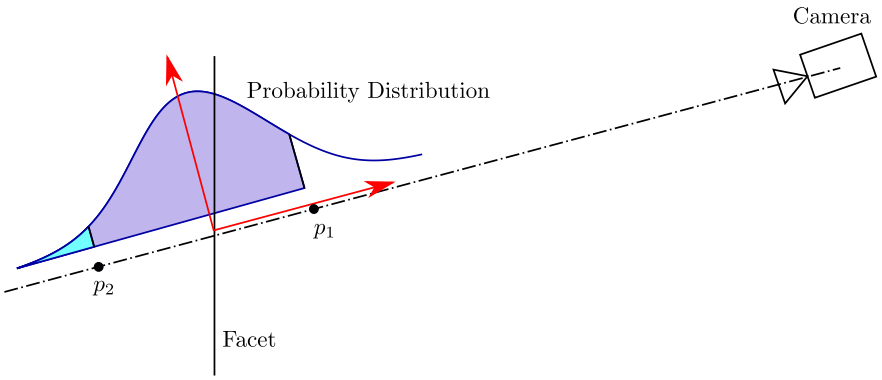

Fig. 5 Probability that a facet exists depends on the relative position between camera, feature (i.e., p_1 and p_2), and facet intersection. Reproduced from [45].

The existence of a single facet is then computed as the combined probability of existence when checked for all viewing directions intersecting the facet provided by the ORB-SLAM algorithm. If the probability of a facet in the triangulation is smaller than a threshold, its vertexes are removed from the point cloud. After performing the probabilistic carving, the coarse model of the target is ready to be exploited in the operational phase. The model obtained with this processing approach is defined in a reference frame which is determined by ORB-SLAM. If the initialization of the map in ORB-SLAM happens at different time instants, this frame can change. However, the operational pipeline requires the constructed model to be defined in the principal inertia axes of the target. To determine the rigid and constant rototranslation necessary to align these frames, the output of ORB-SLAM is matched to a reference frame defined by the principal axes of the constructed model. To do so, Principal Component Analysis is applied to the closest points to the center of the point cloud. This was done to avoid bias induced by points too far from the center of the cloud (i.e., points on appendages). This approach proved effective in this scenario and it led to an error of less than 10 deg between reference frames. However, this method relies on the assumption that the target has a homogeneous density: geometrical symmetry corresponds to material symmetry. In case of a non-homogeneous target, this assumption may

fail and the estimated frame may differ from the principal axes. In such a case, it becomes necessary to estimate also the rigid rotation needed to align the estimated body frame with the principal inertia axes of the target. A possible solution to this problem is to extend the filter state with the parameters of this rigid transformation. Such an approach has been adopted in Sec. V.C for the simplified case of an axisymmetric object which showed satisfactory results.

C. Operational Phase

During the operational phase of the inspection of the unknown and uncooperative RSO, the chaser performs a sequence of operations which is summarized in the pipeline of Fig. 6.

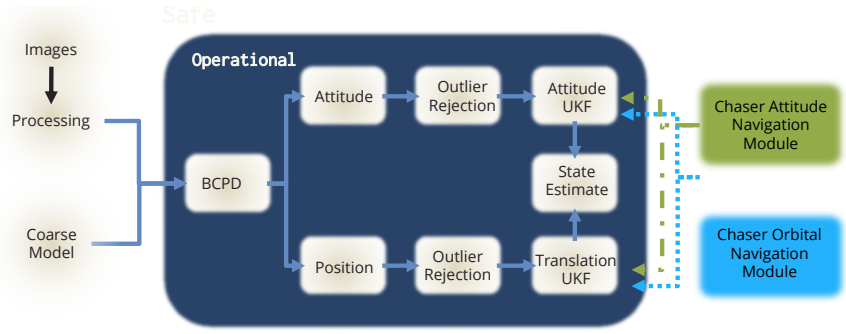

Fig. 6 Flowchart describing the operational pipeline of the CoMBiNa approach.

First, a stereo image is acquired. This process is the same as the preliminary phase and exploits the Blender simulated environment. The stereo acquisition provides an image and a depthmap obtained directly in Blender. This image is then processed to extract a 3d point cloud which is then passed to the nominal operational module. This module is active during the Nominal Operating Mode described in Section IV and it uses a point set registration method (i.e. BCPD) to register the point cloud with the coarse model generated during the preliminary analysis and provides a measurement of relative position and attitude. These measurements are then checked for consistency and treated in two different filters, one estimating the relative translation, the other taking care of attitude and inertia parameters estimation. The choice of separating the two filters was motivated by the study conducted in [45] where, in case of failure of the attitude estimation, the translational filter could still operate and guarantee at least the control of relative position for safety purposes. On the other hand, merging the two filters into a single one may better preserve couplings between attitude and relative state induced by disturbances. Nonetheless, in this work no perturbation was modelled and therefore the separation of filters was retained. In addition, a safety switch is implemented to detect particular failures in the attitude estimation which stem from the presence of a symmetric RSO. In fact, in the case of symmetric targets, BCPD cannot converge to the correct solution as any attitude rotated about the symmetry axis would be acceptable for the point set registration. In [45] a safe operating mode that only recovered the relative state, hence completely disregarding the attitude of the RSO, was presented. This work proposes a different parametrization of the target's attitude (i.e., the estimation of the

direction of the symmetry axis) that leverages the properties of an axisymmetric body. This approach allows to retain as much information as possible in the case of symmetric RSOs and it will be presented into more detail in Section V.
To conclude, it is assumed that the inertial state and attitude of the chaser satellite are perfectly known, which is deemed to be an acceptable hypothesis as dedicated navigation modules could provide the necessary information up to a higher degree of accuracy. The incorporation of inertial navigation errors will induce dynamical modeling errors that severely impact the performances of a closed-loop GNC module. Therefore, the removal of this assumption may even challenge the premises of uncoupled rotation/translation dynamics. This constitutes a strong assumption that was considered acceptable in the framework of this preliminary work where only the navigation module is analyzed and the GNC loop is not closed. The envisioned future works on this problem involve also an analysis addressing the impact of using realistic simulations of navigation modules on the relative navigation filter. The following section is dedicated to a more detailed analysis of the key steps for the relative navigation filters.

IV. Nominal Operating Mode

The nominal operating mode applies a model-based approach to the estimation process by leveraging the rough model built during the preliminary phase. This coarse model implies that no exact feature matching should be searched for, as the target RSO is unknown a priori. The requirement of having an unknown and uncooperative RSO also implies that no training can be done a priori to help during the navigation process, hence excluding any technique relying on template matching or artificial intelligence. Having no insight on which features could be found on the RSO, the chaser can only try to fit the coarse model built and stored on-board to the point clouds extracted from stereo measurements using a statistical point set registration method: BCPD [40]. This section is dedicated to the analysis of the nominal operating mode. First, an overview of the measurement extraction process is provided, secondly, the filter architecture is presented. To conclude, this section introduces a technique to detect the presence of a symmetric body (e.g., the spent upper stage of a rocket) so that a fall-back option can be enabled.

A. Measurement Extraction from BCPD

The goal of the point set registration step is to find the transformation that better aligns two point sets. The choice of algorithm for this work fell on BCPD, which is a variant of the Coherent Point Drift (CPD) [55] algorithm. As the name of these point set registration methods suggest, both algorithms embed in their formulation a regularization term which penalizes non-coherent moves of neighboring points. However, while CPD relies on the Expectation-Maximization (EM) algorithm to determine point correspondences and parameters of the transformation between source and target sets, BCPD relies on Variational Bayesian Inference (VBI). This main discrepancy between algorithms implies that in standard CPD, the motion coherence is imposed as a penalization term in the minimization step of the EM algorithm, whereas in the BCPD the motion coherence is imposed thanks to the prior distribution of displacement vectors. BCPD

assumes a transformation model between point clouds as a combination of a similarity transformation and a non-rigid transformation applied to the points $\mathbf{y}_k$ as follows:

$$\mathcal{T}(\mathbf{y}_k, \boldsymbol{\rho}, \mathbf{v}_k) = s\mathbf{R}(\mathbf{y}_k + \mathbf{v}_k) + \mathbf{t} \tag{2}$$

here $\boldsymbol{\rho}$ collects all the rigid transformation parameters: $s \in \mathbb{R}$ (a scaling factor), $\mathbf{R} \in \mathbb{R}^{3\times3}$ (a rotation), and $\mathbf{t} \in \mathbb{R}^3$ (a translation). The non-rigid transformation parameters are represented by the coherent displacement vectors $\mathbf{v}_k \in \mathbb{R}^3$. Thanks to the particular selection of priors on these displacement vectors, their lengths are inversely proportional to a regularization factor. Therefore, by selecting a particularly large value of this parameter, it is possible to tailor BCPD to the requirements of CoMBiNa, which needs to retrieve a rigid transformation. The parameters of this transformation are obtained through iterations of VBI, which also guarantees convergence of the algorithm as demonstrated in [40]. BCPD algorithm is applied in this work to retrieve the correction terms necessary for aligning the expected point cloud obtained from the coarse model stored on-board with the measured one. In particular, at each measurement step the stereo camera will retrieve a set of points $\boldsymbol{\mathcal{P}}^C$ expressed in the chaser's reference frame. The output of BCPD will retrieve the rigid transformation (i.e. rotation, translation, and scaling) necessary to align this point set to the source set represented by the coarse model of the target stored on-board $\boldsymbol{\mathcal{M}}^{\mathcal{T}} = \{\mathbf{m}_1^{\mathcal{T}}, \ldots, \mathbf{m}_N^{\mathcal{T}}\} \subset \mathbb{R}^{3\times N}$.
First, the initial guess on the target attitude and position coming from the filter state allows to reproject the 3D measured point cloud from the chaser centered, chaser fixed frame to the estimated target frame $\tilde{\mathcal{T}}$ as:

$$\boldsymbol{\mathcal{P}}^{\tilde{\mathcal{T}}} = \mathbf{C}_{\tilde{\mathcal{T}}C}\boldsymbol{\mathcal{P}}^C - \mathbf{1}_M \otimes \mathbf{C}_{\tilde{\mathcal{T}}\mathcal{L}}\hat{\mathbf{r}} \tag{3}$$

Here, the term $\hat{\mathbf{r}}$ defines the estimated relative position of the target's center of mass with respect to the chaser, expressed in the LVLH frame (i.e. $\mathcal{L}$). To maintain consistent matrix sums, the translation is multiplied with the Kronecker product by a vector of all ones of size $1 \times M$ which is represented as $\mathbf{1}_M$. In this expression, $\mathbf{C}_{\tilde{\mathcal{T}}C} = \mathbf{C}_{\tilde{\mathcal{T}}\mathcal{I}}\mathbf{C}_{\mathcal{I}C}$ and $\mathbf{C}_{\tilde{\mathcal{T}}\mathcal{L}} = \mathbf{C}_{\tilde{\mathcal{T}}\mathcal{I}}\mathbf{C}_{\mathcal{I}\mathcal{L}}$ represent the rotation matrices from the chaser frame and LVLH frame respectively, to the estimated target frame. The chaser absolute orbital state and attitude are assumed to be always available with a much higher precision thanks to dedicated navigation modules as reported in Fig. 6. This assumption is made to focus on the main aim of this paper, which is the relative navigation with respect to an unknown and uncooperative RSO. After retrieving this projected set of points, BCPD is applied to register it with the target set $\boldsymbol{\mathcal{M}}^{\mathcal{T}}$ to determine the parameters of the transformation responsible for registering the measured point cloud with the coarse model.

$$\boldsymbol{\mathcal{P}}^{\mathcal{T}} = s\mathbf{R}\boldsymbol{\mathcal{P}}^{\tilde{\mathcal{T}}} + \mathbf{1}_M \otimes \mathbf{t} \tag{4}$$

This rotation is the one responsible for aligning the estimated target frame $\tilde{\mathcal{T}}$ with the coarse model reference frame $\mathcal{T}$, as such, it can also be expressed as $\mathbf{C}_{\mathcal{T}\tilde{\mathcal{T}}}$. The first information that is extracted from the output of BCPD is a measure of the set of Modified Rodrigues Parameters (MRPs) describing the orientation of the target with respect to the inertial frame. To do so, the matrix describing the orientation of the inertial frame is retrieved as:

$$\mathbf{C}_{\mathcal{T}\mathcal{I}} = \mathbf{C}_{\mathcal{T}\tilde{\mathcal{T}}}\mathbf{C}_{\tilde{\mathcal{T}}\mathcal{I}} \tag{5}$$

Given the direction cosine matrix $\mathbf{C}_{\mathcal{T}\mathcal{I}}$, the measure of the corresponding set of MRP with norm $||\boldsymbol{\sigma}_{\mathcal{T}}|| \leq 1$ can be retrieved thanks to the definition of the skew symmetric matrix:

$$[\widetilde{\boldsymbol{\sigma}_{\mathcal{T}}}] = \frac{\mathbf{C}_{\mathcal{T}\mathcal{I}}^{\top} - \mathbf{C}_{\mathcal{T}\mathcal{I}}}{\sqrt{\mathrm{Tr}(\mathbf{C}_{\mathcal{T}\mathcal{I}}) + 1}(\sqrt{\mathrm{Tr}(\mathbf{C}_{\mathcal{T}\mathcal{I}}) + 1} + 1)} \tag{6}$$

Note that this equation only encounters a singularity at an equivalent principal rotation of 180°. At this configuration, both numerator and denominator become 0, i.e. $\mathbf{C}_{\mathcal{T}\mathcal{I}} = \mathbf{C}_{\mathcal{T}\mathcal{I}}^{\top}$ and $\mathrm{Tr}(\mathbf{C}_{\mathcal{T}\mathcal{I}}) = -1$. In this case, the following matrix equation can be solved [56]

$$\boldsymbol{\sigma}_{\mathcal{T}}\boldsymbol{\sigma}_{\mathcal{T}}^{\top} = \frac{1}{2}(\mathbf{I} + \mathbf{C}_{\mathcal{T}\mathcal{I}}) \tag{7}$$

The equations on the diagonal directly provide the $\boldsymbol{\sigma}_{\mathcal{T}}$ vector, while the extra diagonal components provide the information needed to resolve sign ambiguities. The second information which is extracted from the BCPD output is the measure of the expected target position. Eq. 4 included the translation term $\mathbf{t}$, which describes the correction to be applied to the estimated position of the center of mass of the target in order to realign the point clouds. However, this term is expressed in the $\mathcal{T}$ model reference frame. Therefore, it needs to be rotated into the $\mathcal{L}$ frame for correction exploiting the $\mathbf{R}$ matrix as

$$\mathbf{r} = \hat{\mathbf{r}} - \mathbf{C}_{\mathcal{L}\tilde{\mathcal{T}}}\mathbf{C}_{\mathcal{T}\tilde{\mathcal{T}}}^{\top}\mathbf{t} \tag{8}$$

The scaling factor s has been ignored as it is assumed to always be close to 1. The shape of the output of BCPD can be leveraged to reject incorrect measurement estimates thanks to this scaling factor. In fact, should s violate a set of user defined acceptable bounds about 1 (i.e., $s \in [0.8, 1.2]$ in this work), the measurement would be deforming the body, hence providing an incorrect estimate which should be discarded in the filter.

Throughout this section, the noise inherent to the measurement extraction process was not characterized. For the purpose of this work, no real instrument was used, therefore the value of process noise covariance was assumed as a constant tuning parameter that proved to work for the majority of the analyzed test cases. However, an accurate characterization of these quantities is paramount as they are directly correlated to the relative pose of the two spacecrafts.

With the current strategy for the selection of the measurement noise, the implementation cannot work for more realistic experimental setups. Measurement noise parameters represent key figures in the design of this approach and as such they should be better characterized pre-flight rather than tuned. At the current level of implementation the analysis of these quantities was not further investigated. However, a statistical strategy similar to [20] may be adopted. With this method, a database of many possible relative pose configurations may be generated (either synthetically or with an experimental setup) and processed to extract the measurements. Consequently, the statistical properties of the measurement errors may be studied in order to assess the variation of measurement noise covariance with other key quantities (e.g., relative distance).

B. Filter Architecture

Once the measurements are retrieved, two filters are designed to separately deal with relative position estimation and attitude estimation. Standard navigation filters rely on two main steps which are characterized by prediction of state and covariance, and update of the predicted values with measurements. This dual step procedure is typical for sequential filters. The prediction step for standard schemes usually works by describing the distribution of points by only the mean and covariance. This implicitly assumes that the state distribution remains Gaussian after propagation, which is not valid after a nonlinear propagation. However, nonlinear transformations are fairly common in typical GNC problems. Therefore, the accuracy of standard filters decays in highly-nonlinear systems or if the frequency of measurements is too low. To overcome this limitation, this work exploits UKF [41], which relies on the Unscented Transform (UT) to obtain a fully nonlinear mapping of the first two moments of the probability distribution of the state. Moreover, this algorithm is suitable for any choice of process model, with a rapid implementation that avoids the evaluation of the Jacobians which are instead needed in linearized filters [57]. The propagation method of UKF relies on the propagation of a set of sampled points which are called sigma-points. This set of points is deterministically extracted, and the number of points can be selected a priori. Once every sigma point has undergone the nonlinear propagation, their weighted sums provide the mean and covariance of the new distribution. In this work, two different UKF filters have been implemented in order to deal with attitude and translations separately.

First, the relative translational filter is described. The state of this first filter includes relative position $\hat{\mathbf{r}}$ and relative velocity $\hat{\mathbf{v}}$ expressed in the LVLH frame. The translational process dynamics is represented by the Nonlinear Equations of Relative Motion (NERM) (see [58], Chapter 14):

$$\begin{aligned}
\ddot{x} &= 2n\dot{y} + n^2 x - \frac{\mu(R_t+x)}{[(R_t+x)^2+y^2+z^2]^{\frac{3}{2}}} + \frac{\mu}{R_t^2} \\
\ddot{y} &= -2n\dot{x} + n^2 y - \frac{\mu y}{[(R_t+x)^2+y^2+z^2]^{\frac{3}{2}}} \\
\ddot{z} &= -\frac{\mu z}{[(R_t+x)^2+y^2+z^2]^{\frac{3}{2}}}
\end{aligned} \tag{9}$$

Here, μ represents Earth's standard gravitational parameter, n is the mean orbital motion, and R_t represents the radius of the chaser's orbit. To adopt this model it was assumed that the objects are subject to unperturbed Keplerian acceleration under a constant n which corresponds to a circular orbit. To extend the method to general unperturbed orbits it would be sufficient to increase the set of ODEs to include the variation of mean motion and of R_t. Moreover, several simplified methods are available to deal with general scenarios as described in [59]. For the purpose of this study, this model was considered acceptable. The first step of the filter is to obtain sigma points and weights for the relative state at the current time step k exploiting its mean $\hat{\mathbf{x}}_s(k|k) = [\hat{\mathbf{r}}; \hat{\mathbf{v}}]$ and covariance $\mathbf{P}_s(k|k)$. The set of sigma points is indicated with the symbol $\boldsymbol{\mathcal{X}}$, where the i-th element of this set is indicated as $\boldsymbol{\mathcal{X}}^{[i]}$. To obtain the points and their weights, Eqs. (10) – (11) are applied:

$$\begin{aligned}
\boldsymbol{\mathcal{X}}_s^{[0]}(k|k) &= \hat{\mathbf{x}}_s(k|k) \\
\boldsymbol{\mathcal{X}}_s^{[i]}(k|k) &= \hat{\mathbf{x}}_s(k|k) + \left(\sqrt{(l+\tau)\mathbf{P}_s(k|k)}\right)_i \qquad i = 1, \dots, l \\
\boldsymbol{\mathcal{X}}_s^{[i+l]}(k|k) &= \hat{\mathbf{x}}_s(k|k) - \left(\sqrt{(l+\tau)\mathbf{P}_s(k|k)}\right)_{i+l} \qquad i = 1, \dots, l
\end{aligned} \tag{10}$$

$$\begin{aligned}
\boldsymbol{\Omega}_m^{[0]} &= \frac{\tau}{\tau + l} \\
\boldsymbol{\Omega}_P^{[0]} &= \boldsymbol{\Omega}_m^{[0]} + (1 - \gamma^2 + \beta) \\
\boldsymbol{\Omega}_m^{[i]} &= \boldsymbol{\Omega}_P^{[i]} = \frac{1}{2(l+\tau)} \qquad i = 1, \dots, 2l
\end{aligned} \tag{11}$$

In these equations, l represents the size of the state (i.e. 6) and $\left(\sqrt{\mathbf{P}(\cdot)}\right)_i$ indicates the i-th column vector of the matrix square root. In addition, $\boldsymbol{\Omega}^{[i]}$ is used to indicate the i-th component of the weight vectors $\boldsymbol{\Omega}$, with a distinction between the vectors of weights used for the mean (i.e. $\boldsymbol{\Omega}_m$) or for the covariance (i.e. $\boldsymbol{\Omega}_P$). To conclude, τ, β, and γ represent tuning parameters of the UT and their values are detailed in [41]. While β has an optimal value for Gaussian distributions (i.e. 2), γ and κ, regulate how far from the mean the sigma points are located. Finally, $\tau = \gamma^2(l + \kappa) - l$, is a combination of the others.

The obtained set of points $\boldsymbol{\mathcal{X}}_s(k|k)$ is propagated through the dynamics of the system by integrating the set of ODEs in Eqs. (9). The transformed set of points $\boldsymbol{\mathcal{X}}_s(k+1|k)$ and corresponding weights are exploited to recompute state $\hat{\mathbf{x}}_s(k+1|k)$ and covariance $\mathbf{P}_s(k+1|k)$ after propagation using Eqs. (12) – (13).

$$\hat{\mathbf{x}}_s(k+1|k) = \sum_{i=0}^{2l} \boldsymbol{\Omega}_m^{[i]} \boldsymbol{\mathcal{X}}_s^{[i]}(k+1|k) \tag{12}$$

$$\begin{aligned}
\mathbf{P}_s(k+1|k) = \sum_{i=0}^{2l} \boldsymbol{\Omega}_P^{[i]} &\left(\boldsymbol{\mathcal{X}}_s^{[i]}(k+1|k) - \hat{\mathbf{x}}_s(k+1|k)\right) \cdot \\
&\cdot \left(\boldsymbol{\mathcal{X}}_s^{[i]}(k+1|k) - \hat{\mathbf{x}}_s(k+1|k)\right)^\top
\end{aligned} \tag{13}$$

Every sigma point is then passed through the measurement equation to provide the set of sigma points for the expected measure $\mathcal{Z}_s(k+1|k)$:

$$\mathcal{Z}_s^{[i]}(k+1|k) = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix} \boldsymbol{\mathcal{X}}_s^{[i]}(k+1|k) \tag{14}$$

The mean expected measurement $\hat{\mathbf{z}}_s(k+1|k)$, measurement covariance $\mathbf{P}_{z,s}(k+1|k)$, and state measurement cross covariance $\mathbf{P}_{sz,s}(k+1|k)$ can then be retrieved using Eqs. (15) – (17) respectively.

$$\hat{\mathbf{z}}_s(k+1|k) = \sum_{i=0}^{2l} \boldsymbol{\Omega}_m^{[i]} \mathcal{Z}_s^{[i]}(k+1|k) \tag{15}$$

$$\begin{aligned} \mathbf{P}_{z,s}(k+1|k) = \sum_{i=0}^{2l} \boldsymbol{\Omega}_P^{[i]} \left(\mathcal{Z}_s^{[i]}(k+1|k) - \hat{\mathbf{z}}_s(k+1|k) \right) \cdot \\ \cdot \left(\mathcal{Z}_s^{[i]}(k+1|k) - \hat{\mathbf{z}}_s(k+1|k) \right)^\top + \mathbf{R}_s(k+1) \end{aligned} \tag{16}$$

$$\begin{aligned} \mathbf{P}_{sz,s}(k+1|k) = \sum_{i=0}^{2l} \boldsymbol{\Omega}_P^{[i]} \left(\boldsymbol{\mathcal{X}}_s^{[i]}(k+1|k) - \hat{\mathbf{x}}_s(k+1|k) \right) \cdot \\ \cdot \left(\mathcal{Z}_s^{[i]}(k+1|k) - \hat{\mathbf{z}}_s(k+1|k) \right)^\top \end{aligned} \tag{17}$$

Before applying the correction step, one last check to reject outlier is performed. If the scale factor s obtained from BCPD deviates too much from unity (i.e., $s \in [0.8, 1.2]$ is assumed acceptable in this work), the measurement is rejected, and the update step does not take place. Additionally, if the measurement obtained from BCPD has less than 99% probability of being correct under its expected mean and covariance, then also in this case the update step is skipped. This particular condition is checked with the squared Mahalanobis distance, which can be computed as:

$$\mathrm{MD}^2 = (\mathbf{r}(k+1) - \hat{\mathbf{z}}_s(k+1|k)) \mathbf{P}_{z,s}^{-1} (\mathbf{r}(k+1) - \hat{\mathbf{z}}_s(k+1|k))^\top \tag{18}$$

To obtain the threshold value, one needs to compute the inverse of the cumulative distribution function of the chi-square distribution as

$$\text{threshold} = F^{-1}(p|\nu) = \{x : F(x|\nu) = p\} \tag{19}$$

where ν represents the degrees of freedom, in this case 3, and

$$p = F(x|\nu) = \int_0^x \frac{t^{(\nu-2)/2} \exp(-t/2)}{2^{\nu/2}\Gamma(\nu/2)} dt \tag{20}$$

In case the measurement is not rejected, the state and covariance are updated with Eqs. (21) – (24).

$$\hat{\mathbf{x}}_s(k+1|k+1) = \hat{\mathbf{x}}_s(k+1|k) + \mathbf{K}(k+1)\mathbf{y}(k+1) \tag{21}$$

$$\mathbf{P}_s(k+1|k+1) = \mathbf{P}_s(k+1|k) - \mathbf{K}(k+1)\mathbf{P}_{z,s}(k+1)\mathbf{K}^\top(k+1) \tag{22}$$

$$\mathbf{y}(k+1) = \mathbf{z}_s(k+1) - \hat{\mathbf{z}}_s(k+1) \tag{23}$$

$$\mathbf{K}(k+1) = \mathbf{P}_{sz,s}(k+1|k)\mathbf{P}_{z,s}^{-1}(k+1) \tag{24}$$

The second filter processes the attitude motion. The attitude state is composed of the MRP describing the target's orientation $\hat{\boldsymbol{\sigma}}_{\tilde{\mathcal{T}}}$, the angular velocity $\hat{\boldsymbol{\omega}}_{\tilde{\mathcal{T}}}$, and it is augmented with the set of two parameters $\hat{\mathbf{k}}$ parametrizing the target's principal inertia as in Eq. (27). As illustrated in [38], in absence of external torques, the three principal inertia components can be estimated up to a scaling factor. By representing the principal matrix of inertia for the target as

$$\mathbf{M}_t = \begin{bmatrix} A & 0 & 0 \\ 0 & B & 0 \\ 0 & 0 & C \end{bmatrix} \tag{25}$$

it becomes possible to define the observable parameters k_1 and k_2:

$$\begin{aligned} k_1 &= \ln \frac{A}{B} \\ k_2 &= \ln \frac{B}{C} \end{aligned} \tag{26}$$

The parametrized principal matrix of inertia then becomes:

$$\mathbf{M}_t = \begin{bmatrix} e^{k_1} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & e^{-k_2} \end{bmatrix} \tag{27}$$

The attitude state at current time step k can therefore be described as the concatenation of the set of MRP $\hat{\boldsymbol{\sigma}}_{\tilde{\mathcal{T}}}$, the angular velocity $\hat{\boldsymbol{\omega}}_{\tilde{\mathcal{T}}}$, and the elements $\hat{\mathbf{k}}$ that are necessary to parametrize the inertia matrix. In this way, a vector of size 8×1 is obtained: $\hat{\mathbf{x}}_a(k|k) = [\hat{\boldsymbol{\sigma}}_{\tilde{\mathcal{T}}}; \hat{\boldsymbol{\omega}}_{\tilde{\mathcal{T}}}; \hat{\mathbf{k}}]$. The filtering procedure described in detail for the translational filter is

repeated in the same way for the attitude filter. Here, particular attention is given to the set of ODEs used for propagation. The first six equations are reported in Eqs. (28) (see [58], Chapter 3) and Eqs. (29) (see [58], Chapter 4), where the parameters $\hat{\mathbf{k}}$ are treated as constants during the propagation (i.e. $\dot{\hat{\mathbf{k}}} = \mathbf{0}$).

$$\dot{\boldsymbol{\sigma}}_{\tilde{\mathcal{T}}} = \frac{1}{4}\left[\left(1-\sigma_{\tilde{\mathcal{T}}}^2\right)\mathbf{I} + 2\left[\tilde{\boldsymbol{\sigma}}_{\tilde{\mathcal{T}}}\right] + 2\boldsymbol{\sigma}_{\tilde{\mathcal{T}}}\boldsymbol{\sigma}_{\tilde{\mathcal{T}}}^{\top}\right]\boldsymbol{\omega}_{\tilde{\mathcal{T}}} \tag{28}$$

$$\mathbf{M}_t\dot{\boldsymbol{\omega}}_{\tilde{\mathcal{T}}} + \left[\tilde{\boldsymbol{\omega}}_{\tilde{\mathcal{T}}}\right]\mathbf{M}_t\boldsymbol{\omega}_{\tilde{\mathcal{T}}} = \mathbf{0} \tag{29}$$

In these equations $\boldsymbol{\sigma}_{\tilde{\mathcal{T}}}$ is the vector of three MRP with norm $\sigma_{\tilde{\mathcal{T}}}$ and $\mathbf{I}$ represents a $3x3$ identity matrix. Additionally, this equation relies on the definition of the skew symmetric matrix $[\tilde{\boldsymbol{\omega}}_{\tilde{\mathcal{T}}}]$. When modelling the attitude dynamics as in Eq. 29 it is assumed that no disturbing torques are present. This supposition is also necessary to model the inertia as a constant matrix as reported in Eq. 27. Once again, Eqs. (10) – (11) are applied to define the sigma points and weights exploiting the knowledge of the estimate of the state at the current time step k (i.e., $\hat{\mathbf{x}}_a(k|k) = [\hat{\boldsymbol{\sigma}}_{\tilde{\mathcal{T}}}; \hat{\boldsymbol{\omega}}_{\tilde{\mathcal{T}}}; \hat{\mathbf{k}}]$), and the current estimate of its covariance $\mathbf{P}_a(k|k)$. The obtained set of points $\boldsymbol{\mathcal{X}}_a(k|k)$ is propagated through the dynamics of the system by integrating the set of ODEs in Eqs. (28) – (29). The transformed set of points $\boldsymbol{\mathcal{X}}_a(k+1|k)$ and corresponding weights are exploited to recompute state $\hat{\mathbf{x}}_a(k+1|k)$ and covariance $\mathbf{P}_a(k+1|k)$ after propagation using Eqs. (12) – (13). Every sigma point is then passed through the measurement equation, which this time will be

$$\mathcal{Z}_a^{[i]}(k+1|k) = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_{3\times5} \\ \mathbf{0}_{5\times3} & \mathbf{0}_5 \end{bmatrix} \boldsymbol{\mathcal{X}}_a^{[i]}(k+1|k) \tag{30}$$

The mean expected measurement $\hat{\mathbf{z}}_a(k+1|k)$, measurement covariance $\mathbf{P}_{z,a}(k+1|k)$, and state measurement cross covariance $\mathbf{P}_{az,a}(k+1|k)$ can be retrieved using Eqs. (15) – (17) respectively. In case the measurement is not rejected by either Mahalanobis Distance criterion or scaling parameter coming from BCPD, the state and covariance are updated with Eqs. (21) – (24). Notice that this procedure is only available when operating in nominal mode. When symmetry is detected, other assumptions will be introduced to try and solve this much more challenging problem as explained in Section V.

Finally, an additional step is taken to ensure that the MRP never encounter a singularity. Every time that $\hat{\boldsymbol{\sigma}}_{\tilde{\mathcal{T}}}\hat{\boldsymbol{\sigma}}_{\tilde{\mathcal{T}}}^{\top} = 1$ after the correction step of the filter, the MRP parametrization is shifted to the shadow set thanks to Eq. 31.

$$\hat{\boldsymbol{\sigma}}_{\tilde{\mathcal{T}}}^{S} = -\frac{\hat{\boldsymbol{\sigma}}_{\tilde{\mathcal{T}}}}{\hat{\sigma}_{\tilde{\mathcal{T}}}^2} \tag{31}$$

This also requires transforming the covariance matrix according to [60] using

$$\mathbf{T} = 2\frac{1}{||\hat{\boldsymbol{\sigma}}_{\tilde{\mathcal{T}}}||^4} - 4\hat{\boldsymbol{\sigma}}_{\tilde{\mathcal{T}}}\hat{\boldsymbol{\sigma}}_{\tilde{\mathcal{T}}}^{\top} - \frac{1}{||\hat{\boldsymbol{\sigma}}_{\tilde{\mathcal{T}}}||^2}\mathbf{I}$$

$$\mathbf{P}_{z,a}^{S}(k+1|k+1) = \begin{bmatrix} \mathbf{T} & \mathbf{0}_{3\times5} \\ \mathbf{0}_{5\times3} & \mathbf{I}_5 \end{bmatrix} \mathbf{P}_{z,a}(k+1|k+1) \begin{bmatrix} \mathbf{T} & \mathbf{0}_{3\times5} \\ \mathbf{0}_{5\times3} & \mathbf{I}_5 \end{bmatrix}^{\top} \tag{32}$$

C. Symmetry Detection

To additionally account for possible symmetry, the symmetry detection algorithm employs a sequence of measurements. Each measurement of attitude $\boldsymbol{\sigma}_{\mathcal{T}}(k)$ has an associated rotation matrix, $\mathbf{C}_{\mathcal{TI}}(k)$. The angular error between two consecutive measurements can therefore be computed as

$$\Phi(k) = \arccos\left(\frac{1}{2(\mathrm{Tr}(\mathbf{C}_{\mathcal{TI}}(k+1)\mathbf{C}_{\mathcal{TI}}^{\top}(k)) - 1)}\right) \tag{33}$$

On the other hand, the principal rotation between two measurements can also be computed by exploiting the knowledge of the angular velocity at the initial timestep $\hat{\boldsymbol{\omega}}_{\tilde{\mathcal{T}}}(k|k)$

$$\hat{\Phi}(k) = ||\hat{\boldsymbol{\omega}}_{\tilde{\mathcal{T}}}(k|k)||\Delta t \tag{34}$$

If the discrepancy between predicted and measured principal angle variation is above a certain threshold for a sequence of measurements, then this may imply the presence of symmetry, and the algorithm should fall back to a fail safe navigation approach which disregards the RSO attitude information. The output of Eq. 34 is expected to vary smoothly between time instants. On the contrary, if a body has symmetry axes with unobservable rotations, the estimate obtained from Eq. 33 could vary widely: as BCPD only matches the geometry of the point clouds, it could indistinctly converge to any configuration that is simply rotated about one of the symmetry axes. The procedure that is followed once this failure has been detected is described in the following section.

V. Symmetric Case Operating Mode

In case of a highly symmetrical body, the matching provided by BCPD could change from one measurement to the following as any rotation about the axes of symmetry would not be detected. Convergence of BCPD would provide attitude results which are unreliable, and, should such behavior be detected, the chaser should switch to a different approach for relative navigation, as the measurement of BCPD may not provide accurate results. Therefore, in the case of a symmetric target, the aim of the navigation module is simply to estimate the orientation of the symmetry axis of the body. To do that, the RSO is modeled as an axisymmetric rigid body and the basic equations of the theory of rigid body motion are exploited; more details are provided in Section V.A.

Therefore, if symmetry is detected by the symmetry detection algorithm presented in IV.C, the measurements extracted by BCPD and, consequently, the state variables estimated by the navigation filters are modified with respect to the asymmetric case, as described in Sections V.B and V.C.

A. Axisymmetric Model

Given the diagonal principal inertia matrix $\mathbf{M}_t$, an axisymmetric rigid body is a structure identified by two equal moments of inertia (namely $A = C \neq B$). In absence of external torques, axisymmetric bodies are characterized by a constant angular momentum vector $\mathbf{\Gamma}_{(0)}$ (where 0 is the body's barycenter) over time and a constant component of the angular velocity about the symmetry axis ω_y. Owing to that, a permanent rotation of the body around the symmetry axis and a regular precession of the symmetry axis itself can be observed when dealing with this type of structures. In a realistic scenario, disturbing torques may arise to produce a variation of the angular momentum vector. However, in this work, no disturbances were modeled, therefore this idealized condition can be considered reasonable. The equations describing the precession of the axis of symmetry $\boldsymbol{J}$ in an inertial frame $\mathcal{I}_\Gamma$ (characterized by one axis aligned with $\mathbf{\Gamma}_{(0)}$) can be easily retrieved starting from Poisson formulas:

$$\dot{\boldsymbol{J}} = \boldsymbol{\omega} \times \boldsymbol{J} \tag{35}$$

where the angular velocity $\boldsymbol{\omega}$ can be written in terms of the angular momentum vector as :

$$\boldsymbol{\omega} = \frac{\mathbf{\Gamma}_{(0)}}{A} - \frac{(B-A)\omega_y \boldsymbol{J}}{A} \tag{36}$$

To obtain Eq. 36, the definition of $\mathbf{\Gamma}_{(0)}$ shall be exploited, which for the case of an axisymmetric body is given by :

$$\mathbf{\Gamma}_{(0)} = A(\omega_x \boldsymbol{I} + \omega_z \boldsymbol{K}) + B\omega_y \boldsymbol{J} \tag{37a}$$

$$= A\boldsymbol{\omega} + (B-A)\omega_y \boldsymbol{J} \tag{37b}$$

from which Eq. 36 can easily be obtained. At this point, by substituting it in Eq. 35, the equation to compute the motion of the symmetry axis becomes :

$$\dot{\boldsymbol{J}} = \frac{\mathbf{\Gamma}_{(0)}}{A} \times \mathbf{J} - \frac{(B-A)\omega_y}{A} \boldsymbol{J} \times \boldsymbol{J} \tag{38a}$$

$$= \frac{\mathbf{\Gamma}_{(0)}}{A} \times \boldsymbol{J} \tag{38b}$$

As introduced before, $\mathbf{\Gamma}_{(0)}$ is considered to be aligned with one axis of the inertial reference frame $\mathcal{I}_\Gamma = \{\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k}\}$; in

particular $\boldsymbol{\Gamma}_{(0)} = \Gamma_{(0)}\boldsymbol{j}$. Eq. 38b can therefore be rewritten as :

$$\dot{\boldsymbol{J}} = \frac{\Gamma_{(0)}}{A}(\boldsymbol{j} \times \boldsymbol{J}) \tag{39}$$

Finally, by expressing the symmetry axis $\boldsymbol{J}$ according to the $\mathcal{I}_\Gamma$ frame coordinates, (i.e. $\boldsymbol{J} = J_x\boldsymbol{i} + J_y\boldsymbol{j} + J_z\boldsymbol{k}$), Eq. 39 becomes :

$$\begin{cases} \dot{J}_x = \dfrac{\Gamma_{(0)}}{A} J_z & \text{(40a)} \\ \dot{J}_y = 0 & \text{(40b)} \\ \dot{J}_z = -\dfrac{\Gamma_{(0)}}{A} J_x & \text{(40c)} \end{cases}$$

which demonstrates that the axis $\boldsymbol{J}$ describes a cone around $\boldsymbol{\Gamma}_{(0)}$ in the inertial frame $\mathcal{I}_\Gamma$.

B. Measurement Extraction from BCPD

As for the asymmetric case, the algorithm adopted for the registration phase is the BCPD. As already seen in IV.A, the algorithm provides, at each time step, the estimated parameters of the transformation that aligns the two point sets, namely the scaling factor s, the translation $\mathbf{t}$ and the rotation matrix $\mathbf{R}$. As seen before, $\mathbf{R}$ can also be expressed as $\mathbf{C}_{\mathcal{T}\tilde{\mathcal{T}}}$ and allows to compute the direction cosine matrix $\mathbf{C}_{\mathcal{T}\mathcal{I}}$ by means of Eq. 5. Unlike the asymmetric case, however, only the second row of $\mathbf{C}_{\mathcal{T}\mathcal{I}}$ can be considered meaningful, since it provides the coordinates of the symmetry axis $\boldsymbol{J}$ in the inertial frame $\mathcal{I}$. The other two rows represent, instead, the orientation of the other two body axes on the equatorial plane (i.e., the plane orthogonal to the symmetry axis); however, since their true orientation is not observable because of the body's geometric symmetry, BCPD converges to a random orientation of the two axes, that does not necessarily correspond to the true one. For this reason, the information given by the first and the third rows of $\mathbf{C}_{\mathcal{T}\mathcal{I}}$ is discarded. Then, as for the asymmetric case, the second information extracted from the BCPD output is the measure of the expected target position, through the same procedure presented in Eq. 8. Indeed, BCPD provides consistent outputs which allows to retrieve the correct translational correction term by compensating possible errors in the definition of the matrix $\mathbf{C}_{\mathcal{T}\tilde{\mathcal{T}}}$.

C. Filter Architecture

Once the measurements are retrieved, still two navigation filters are adopted: the translational filter, that, even in this case, will provide an estimate of the relative position and velocity between the chaser and the target, and the rotational filter that, this time, will estimate just the orientation of the symmetry axis of the target body in the inertial frame $\mathcal{I}_\Gamma$. In particular, as far as the translational filter is concerned, the state vector estimated still includes relative position $\hat{\mathbf{r}}$ and relative velocity $\hat{\mathbf{v}}$ expressed in the LVLH frame; the filtering procedure is exactly the same as described in Section IV.B. The state of the rotational filter, instead, includes the spherical coordinates of the target's symmetry axis $\boldsymbol{J}$ in the inertial

frame $\mathcal{I}_\Gamma$ (indicated as azimuth $\hat{\varphi}$ and elevation $\hat{\delta}$) and the scalar ratio $\widehat{\Gamma}_{(0)}/A$. The set of ODEs used to propagate $\hat{\varphi}$ and $\hat{\delta}$ can be derived from Eq. 40 (where the Cartesian coordinates have to be replaced by their spherical representation), while the parameter $\widehat{\Gamma}_{(0)}/A$ is treated as constant during the propagation (i.e. $\frac{d}{dt}\left(\Gamma_{(0)}/A\right) = 0$) as a property of axisymmetric rigid bodies.

Once again, Eqs. (10) – (11) are applied to define the sigma points and weights exploiting the knowledge of the estimate of the state at the current time step k (i.e., $\hat{\mathbf{x}}_a(k|k) = [\hat{\varphi}; \hat{\delta}; \widehat{\Gamma}_{(0)}/A]$), and the current estimate of its covariance $\mathbf{P}_a(k|k)$. The obtained set of points $\boldsymbol{\mathcal{X}}_a(k|k)$ is propagated through the dynamics of the system by integrating the set of ODEs. Then, the transformed set of points $\boldsymbol{\mathcal{X}}_a(k+1|k)$ and corresponding weights are exploited to recompute state $\hat{\mathbf{x}}_a(k+1|k)$ and covariance $\mathbf{P}_a(k+1|k)$ after propagation using Eqs. (12) – (13). Every sigma point is then passed through the measurement equation, which this time will be

$$\boldsymbol{\mathcal{Z}}_a^{[i]}(k+1|k) = \mathbf{h}\left[\boldsymbol{\sigma}_{\mathcal{I}_\Gamma \mathcal{I}}, \boldsymbol{\mathcal{X}}_a^{[i]}(k+1|k)\right] \tag{41}$$

The measurements extracted from BCPD are expressed in the inertial reference frame $\mathcal{I}$ instead of $\mathcal{I}_\Gamma$; therefore, to solve this inconsistency, the function $\mathbf{h}[\cdot;\cdot]$ symbolizes the fact that the sigma points associated with the variables φ and δ must be transformed into Cartesian coordinates, rotated by exploiting the MRPs $\boldsymbol{\sigma}_{\mathcal{I}_\Gamma \mathcal{I}}$ expressing the rigid rotation between the two inertial frames and transformed back to spherical coordinates.

The mean expected measurement $\hat{\mathbf{z}}_a(k+1|k)$, measurement covariance $\mathbf{P}_{z,a}(k+1|k)$, and state measurement cross covariance $\mathbf{P}_{az,a}(k+1|k)$ can be retrieved using Eqs. (15) – (17) respectively. In case the measurement is not rejected by either Mahalanobis Distance criterion or scaling parameter coming from BCPD, the state and covariance are updated with Eqs. (21) – (24).

Before concluding, two aspects must be emphasized. The first one is that, with the measurements that BCPD is able to provide with its current formulation, all the variables of $\hat{\mathbf{x}}_a$ are observable, and, consequently, correctly estimated. To demonstrate that, a nonlinear observability analysis was performed, which follows the Hermann and Krener's algebraic test for local weak observability based on the computation of the Lie Derivatives [61].

However, it should be noticed that Eq. 41 requires the knowledge of the MRPs $\boldsymbol{\sigma}_{\mathcal{I}_\Gamma \mathcal{I}}$, which, as previously seen, express the rotation between the inertial frame $\mathcal{I}_\Gamma$ and the general Earth-centered inertial frame $\mathcal{I}$. Therefore $\boldsymbol{\sigma}_{\mathcal{I}_\Gamma \mathcal{I}}$ shall be included in the state vector $\hat{\mathbf{x}}_a$ estimated by the rotational filter. This augmentation of the state causes a loss observability (i.e., the observability matrix is not full rank). Nevertheless, as the numerical analysis in the following section shows, the navigation filter is still able to converge. Although, a slower convergence rate is registered, the error still remains bounded within the three-sigma limits; further details can be found in [62] (Chapter 4).

The second aspect that should be noticed is that neither the angular velocity nor the inertia moments of the target are included in the estimated state vectors; the reason behind that is the lack of observability of these variables. Nevertheless,

a vector of two other quantities related to angular velocity and inertia can be computed taking advantage of the estimated variables $\hat{\mathbf{x}}_a$, in particular $\mathbf{x}_e = [e;\ (\omega_y/A)^2]$. The first quantity is the amplitude of the equatorial component of the target's angular velocity, namely $e = \sqrt{(\omega_x^2 + \omega_z^2)}$. It can be demonstrated that the value of e is equal to the norm of the time derivative of the vector $\boldsymbol{J}$, which in turn can be easily retrieved from the knowledge of the estimated axis spherical coordinates $\hat{\varphi}$ and $\hat{\delta}$ (the proof of that can be found in [62], Chapter 3 and 4):

$$e = f_1(\hat{\mathbf{x}}_a) = \frac{\widehat{\Gamma}_{(0)}}{A}\sqrt{1 - \sin^2\hat{\varphi}\cos^2\hat{\delta}} \tag{42}$$

Due to the properties of axisymmetric structures, the equatorial component of the angular velocity e remains constant over time. The second quantity that can be retrieved a posteriori is the square of the ratio between the angular velocity about the symmetry axis $\boldsymbol{J}$ and the moment of inertia A about the other two body axes $\boldsymbol{I}$ and $\boldsymbol{K}$. By looking at the definition of the angular momentum vector (Eq. 37b), one can easily get:

$$\left(\frac{\omega_y}{A}\right)^2 = f_2(\hat{\mathbf{x}}_a) = \left(\frac{\widehat{\Gamma}_{(0)}}{A}\right)^2 - e^2 \tag{43}$$

Even in this case, this quantity remains constant over time. It can be noticed that the unit value of the moment of inertia B has been assumed in Eq. 43; this assumption doesn't have much effect since the moments of inertia are observable only up to a scale factor in absence of external torques as illustrated in [38].

The covariance matrix associated to $\mathbf{x}_e$ can be approximated by exploiting the Jacobian $\mathcal{J}_{\boldsymbol{F}}$ of the mapping, in particular :

$$\mathbf{P}_e = \mathcal{J}_{\boldsymbol{F}}\ \mathbf{P}_a\ \mathcal{J}_{\boldsymbol{F}}^{\top} \tag{44}$$

with :

$$\mathcal{J}_{\boldsymbol{F}} = \left[\frac{\partial f_1}{\partial \hat{\mathbf{x}}_a},\ \frac{\partial f_2}{\partial \hat{\mathbf{x}}_a}\right]^{\top} \tag{45}$$

where f_1 and f_2 are the two scalar functions expressing the dependence of $\mathbf{x}_e$ from the estimated $\hat{\mathbf{x}}_a$ (defined in Eq. 42 – 43).

Therefore, it can be observed that, in the case of a symmetric RSO, the attitude values and inertial properties of the body cannot be estimated independently, but a set of quantities that relate them to each other can be retrieved.

VI. Numerical Simulations

For the purpose of this study, the chaser is considered to be on a circular orbit in LEO with a radius of 800 km, which is located in the most populated band of altitudes in LEO. The mean motion in this case is $n \approx 0.001$ rad/s, whose inverse is used to provide the scaled time unit in all simulations, that is TU = 963.65 s. Meanwhile, distances are scaled

with the radius of the keep out sphere of the RSO, that is the maximum dimension of the RSO LU = 12 m. Given these scaling factors, the initial condition of the nominal relative orbit expressed in the LVLH frame is

$$\mathbf{r}(0) = [2; 0; 0.2]\text{LU}; \ \mathbf{v}(0) = [0; -4; 0]\text{LU/TU}; \tag{46}$$

This particular initial condition produces an inclined football orbit in the Clohessy-Wiltshire dynamics [39], and an approximate one in NERM, provided that the relative separation is small enough with respect to the orbital radius. This relative orbit is represented in Fig. 7. By assuming a periodic relative orbit as a reference it is possible to guarantee that the target will never go too far from the chaser. Moreover, such a choice of inspection trajectory is extremely common in literature [38].

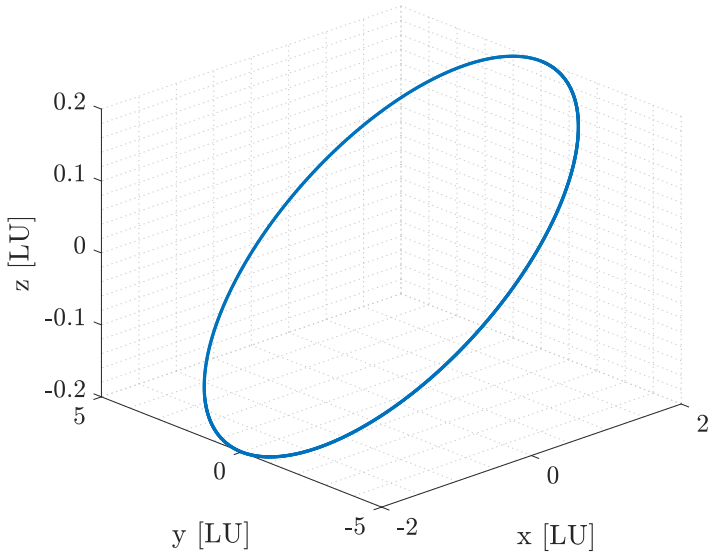

Fig. 7 Representation of the nominal inclined football orbit in the LVLH frame. Reproduced from [45].

The target's attitude is initialized with the following nominal random values, which are used in the simulated environment to produce the images used by the algorithm.

$$\begin{aligned} \boldsymbol{\sigma}_{\mathcal{T}}(0) &= [-0.083; 0.220; -0.500] \\ \boldsymbol{\omega}_{\mathcal{T}}(0) &= [-0.3953; -0.7065; -0.8153] \text{ [rad/TU]} \\ \boldsymbol{k}_{\mathcal{T}} &= [0.03; -0.1] \end{aligned} \tag{47}$$

In the simulation the initial covariances are assumed to be

$$\begin{aligned} \mathbf{P}_{x_s}(0) &= 0.1 \cdot \mathbf{I}_6 \\ \mathbf{P}_{x_a}(0) &= 0.1 \cdot \mathbf{I}_8 \end{aligned} \tag{48}$$

On the other hand, the measurement noise is used as a tuning parameter to help in convergence and the values for the two different filters are set to be $\mathbf{R}_s = \mathbf{R}_a = 0.01 \cdot \mathbf{I}_3$. This assumption was made arbitrarily to aid the convergence of filter iterations. Additional work should be conducted to estimate the actual measurement noise covariance produced by the combination of image processing and BCPD. In fact, this constitutes one of the most interesting aspects for further research on this topic.

As stated in Section III, CoMBiNa will start only after the chaser has been delivered in the desired initial conditions by a separate GNC module for rendezvous. Due to the lack of literature on the relative navigation about unknown targets the values of relative initial separation ($\approx$ 37m on average) and uncertainty (i.e., $1\sigma \approx$ 3m) were selected based on similar cases of relative optical navigation about uncooperative targets. For example, in the framework of an ESA study, the authors of [63, 64] investigated a simulated relative navigation scenario adopting a relative range of $\approx$ 31m, with a 1σ uncertainty of 1m. Additionally, two demonstrator missions were flown with similar relative distances and accuracy. First, the PRISMA mission reached approximately a relative separation of 50m with a 2m position uncertainty [42, 65, 66] during the ARV experiment. Subsequently, the AVANTI experiment reached $\approx$ 50m with a meter level accuracy [42]. All these scenarios, despite different from the application under consideration in this study, motivated the selection of these parameters and were considered a reasonable first guess. Nevertheless, it is worth mentioning that raising the relative separation in the order of $\approx$ 100m may constitute a safer option while flying about an unknown target. However, starting the navigation phase at such a large distance may significantly hinder the estimation of the target's pose. A solution to this problem at higher relative distances is yet to be found and, therefore, constitutes an interesting research outlet for future studies.

First, the preliminary analysis phase is conducted. During this first step, the SLAM technique investigated to build the coarse target model is ORB-SLAM [36]. As ORB-SLAM algorithm is well known and does not represent a main contribution of this work, only a brief summary of the method used for our simulations in this section is included for completeness. The result is reported in Fig. 8, whose raw point cloud is used as an input for the processing step aimed at removing outliers.

The second step in the preliminary shape construction procedure is to filter the point cloud to remove outliers exploiting the probabilistic carving method described in Section III.B. When the filtering procedure is applied, the resulting point cloud can be observed in Fig. 9. As it can be noticed, the overall shape of the point cloud is preserved, while many of the outliers are removed, also reducing the total size of the point cloud. This reduction also causes a decrease in the

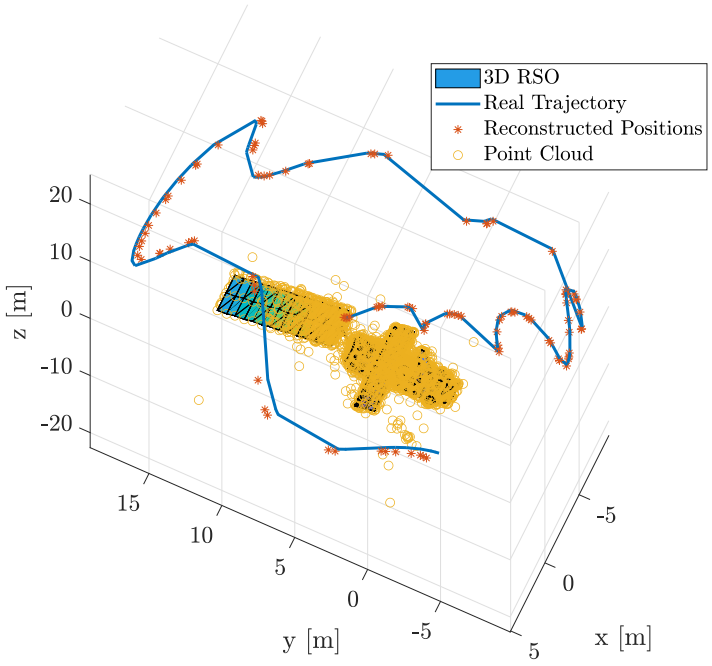

Fig. 8 Trial run of the ORB-SLAM algorithm applied to a simulated trajectory around Envisat. The reference frame is shifted to the target fixed axes for simplicity.

computational cost of using such a map during the operational phase. The initial coarse output of ORB-SLAM includes 4980 points, which are reduced to $\approx$ 3100 after filtering, depending on the tuning parameters of the probabilistic carving method.

After the preliminary construction has been carried out, the operational phase can start. The duration of the preliminary phase can be fixed to an orbital period (as in this case) or with a longer time horizon. The duration required to retrieve a full mapping of the target is influenced by the RSO's tumbling frequency, illumination conditions, and relative orbit. A method to optimally select the inspection orbit about an unknown and uncooperative target with a fixed time horizon was proposed in [50]. Should a partial model be used, the proposed approach may still work provided that the measurements obtained during the operational phase are taken from the same relative poses that were used to build the model. However, the results of BCPD are usually degraded if partial models are used, which is already the case since the stereo measurements cannot capture features in shadow or occluded by the target itself. Hence, for the purpose of this study, the analysis was limited to cases with a full model construction. The translational filter produces the convergence figures reported in Fig. 10, while the outcome of the attitude filter is represented in Fig. 11. As it can be observed, the inertia estimation is slower to converge with respect to the other states. This is owed to the fact that the speed of convergence of this parameter is strongly related to the tumbling speed of the target as demonstrated in [33]. To demonstrate the capability to detect symmetry with the approach described in Section IV.C, a simulation with equal

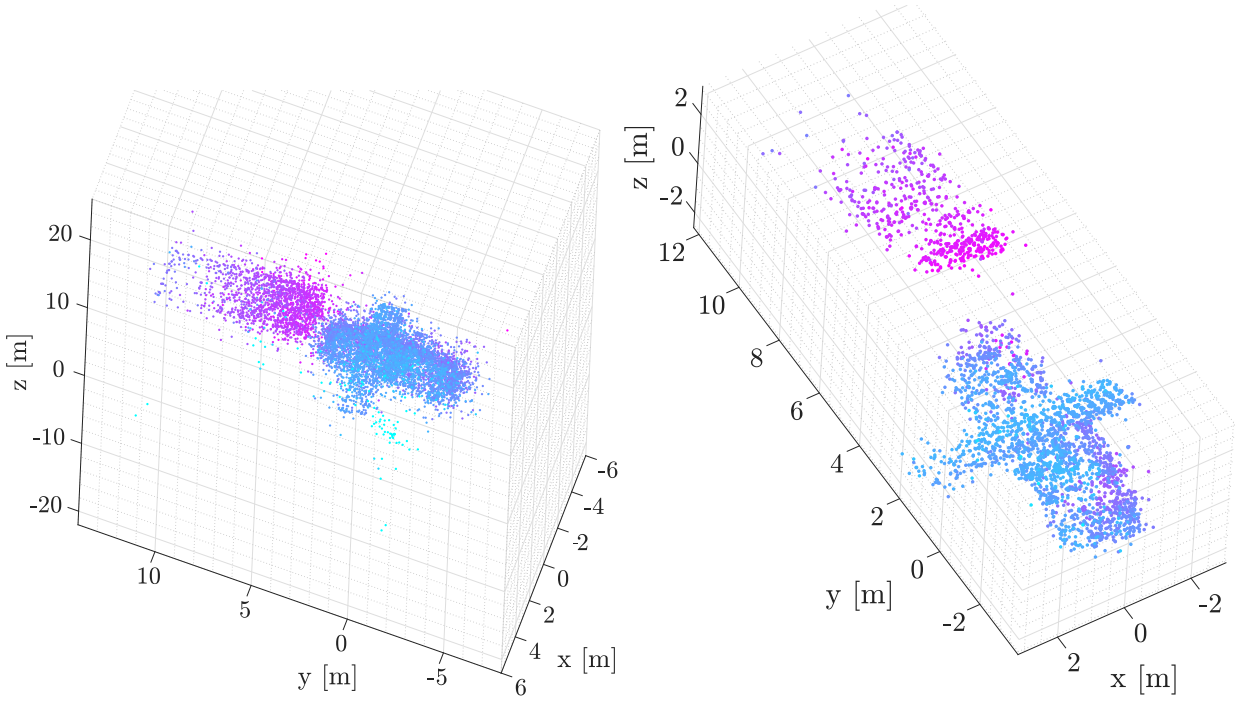

Fig. 9 Point cloud raw output of ORB-SLAM (left) and its filtered counterpart (right). Reproduced from [45].

parameters but using a symmetric target is conducted. In this case the built target coarse point cloud is reported in Fig. 12.

As expected, the attitude estimate leveraging BCPD with a full attitude state estimation fails as illustrated in Fig. 13.

By looking at the outlier rejection criteria described in Section IV.C, when applied to these two test cases it can be seen in Fig. 14 that the average difference between principal angle variation estimates, as well as their standard deviation, is one order of magnitude larger in the case of symmetric targets. This figure also includes the points which have been marked as outliers by the scaling of BCPD (black circles) and those marked as outliers by the Mahalanobis distance criterion (red circles). In the case of symmetric targets the BCPD run detects lots of outliers which are not used inside the attitude filter, hence explaining the plateau reached at about 1 TU. It is clear now how this criterion can be used to detect symmetry in the target and switch to a different attitude parametrization presented in Section V.

By adopting this approach specifically designed for symmetric targets, the convergence graphs obtained from the translational and rotational filters are shown in Fig. 15 and Fig. 16, respectively. As regards the latter figure, the dashed line refers to the case in which the MRPs expressing the rotation between the two inertial frames $\boldsymbol{\sigma}_{\mathcal{I}_\Gamma \mathcal{I}}$ are assumed as a known input to the filter and, as a result, are not included in the estimated state vector $\hat{\mathbf{x}}_a$. When this assumption is removed, the convergence graphs produced are the ones represented through the solid line. As expected, a slower convergence rate can be observed, as well as a higher error and three-sigma limit; the reason for this can be attributed to the unobservable degree of freedom of this formulation. Finally, in Fig. 17 the trends over time of the error and the

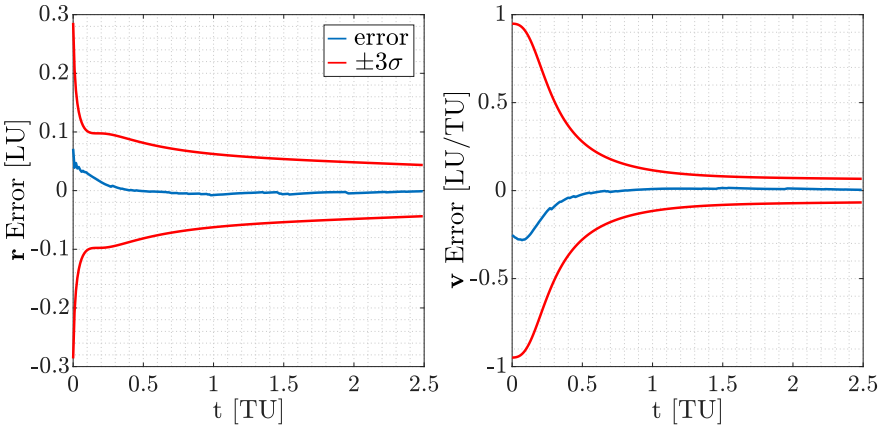

Fig. 10 Convergence of the translational filter in the asymmetric case for position (left) and velocity (right). Reproduced from [45].

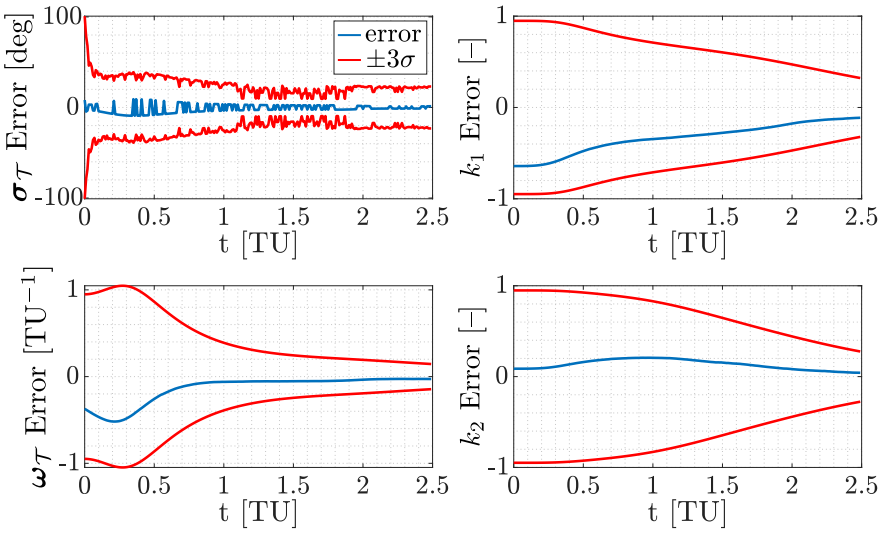

Fig. 11 Convergence of the rotational filter in the asymmetric case. Attitude (top left), angular velocity (bottom left), and inertia parameters (right). Reproduced from [45].

three-sigma limit of the equatorial component of the angular velocity e and the ratio ω_y^2/A^2 are displayed.

VII. Expected Capabilities and Limitations

To conclude, this section is dedicated to the analysis of the capabilities and limitations of the application of CoMBiNa in real world scenarios. First, this section describes the behavior of the algorithm with a processor in the loop analysis. This analysis provides an estimate of the expected computational time per filter iteration obtained with a limited resources system such as a Raspberry Pi 4 Single Board Computer (SBC). The thus obtained upper bound measurement frequency is then used to perform an analysis aimed at estimating the capabilities of CoMBiNa when applied to a real set of tumbling objects in LEO, whose estimates have been obtained from lightcurve processing by the the MMT-9 sky monitoring system [67]. In particular, tumbling rates are provided regardless of symmetry of objects, which in principle

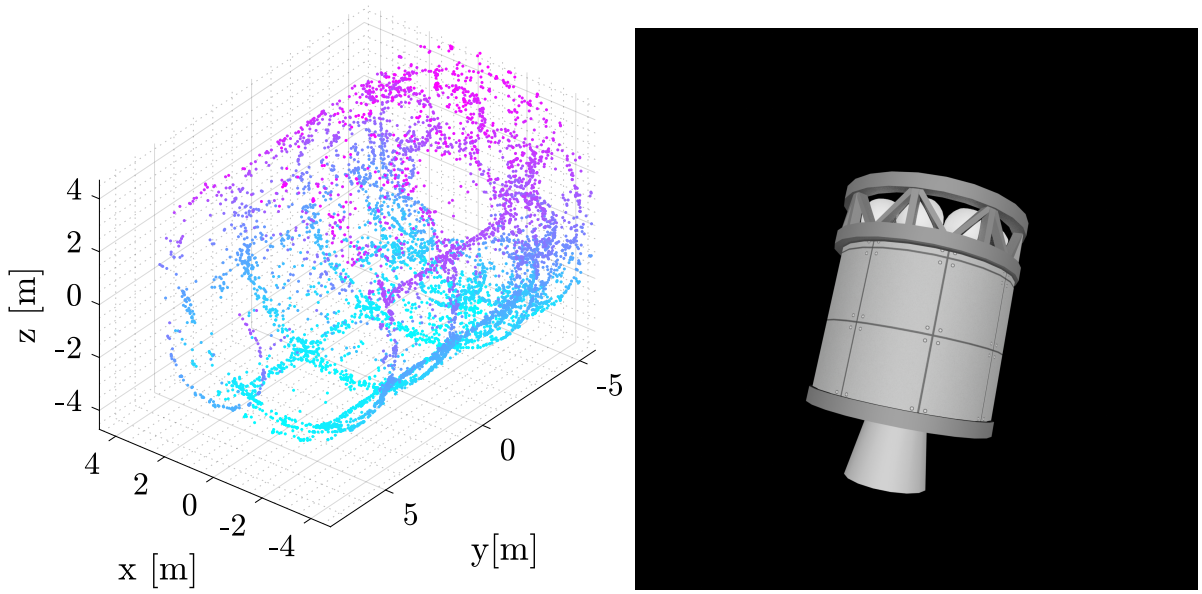

Fig. 12 Coarse point cloud of the symmetric target (left) and corresponding ground truth model (right).

may produce lightcurves with higher frequency than the actual tumbling rate. Nonetheless, this information cannot be inferred a priori, therefore the tumbling rates should provide a conservative estimate of the achievable performance.

Initially, the population of the dataset is reduced to only account for LEO RSOs. This set of objects constitute roughly 83% of the dataset, which account for 543 elements. The overall breakdown of the dataset is given in Fig. 18. This figure (right) also shows the distribution of LEO RSOs for different tumbling regimes. The red curve representing the cumulative distribution of the population clearly shows that the RSOs are concentrated at a tumbling rate < 1.5 Hz with $\approx 95\%$ of the elements in the dataset falling in this region.

The first step in determining the limitations of the CoMBiNa pipeline is to determine the maximum and minimum rates at which this algorithm can run. In order to do so, a Monte Carlo (MC) analysis is performed. In particular, 1000 different iterations of the navigation filter are sampled from simulated trajectories. The filter prediction, update, and measurement registration with BCPD are executed on a Raspberry Pi 4 platform, equipped with 4GB of RAM, 64GB MicroSD acting both as storage and OS drive. This SBC runs on a quad-core ARM Cortex-A72 processor at up to 1.5GHz, which enables both parallelization of the code as well as parallel execution of other tasks. This latter feature, enables the optimization of the compilation of the BCPD code for faster execution, allowing a significant code speedup. The use of this SBC also allowed to perform a sensitivity analysis to observe the degradation of performance for lower clock speeds. In particular, it was possible to directly analyze the range of frequencies of $600 - 1500$ MHz, while an extrapolation was adopted to predict the expected performances down to 100 MHz. It is important to mention that, as expected, the computational time of each filter iteration, composed of BCPD, update, and prediction, was mostly occupied with the BCPD step, which accounts for $\sim 99\%$ of the computational time. The results of this analysis are

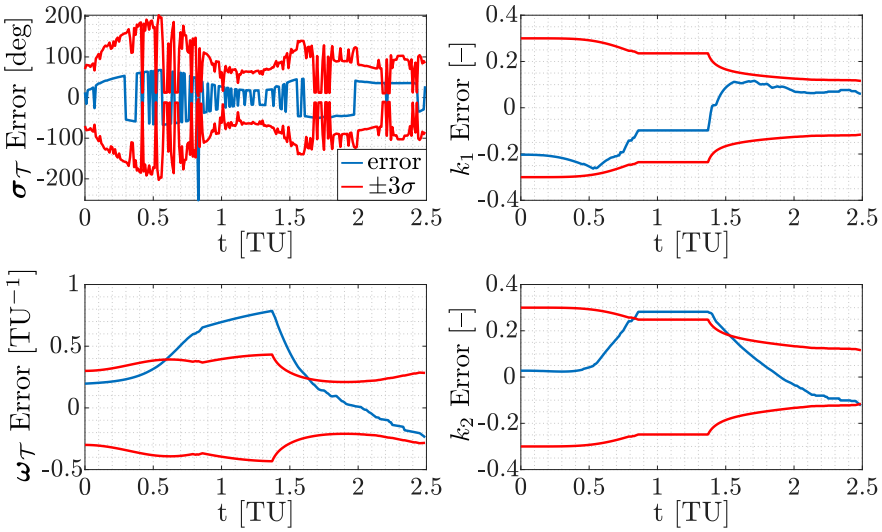

Fig. 13 Convergence of the rotational filter with full attitude characterization in the symmetric case. Attitude (top left), angular velocity (bottom left), and inertia parameters (right)

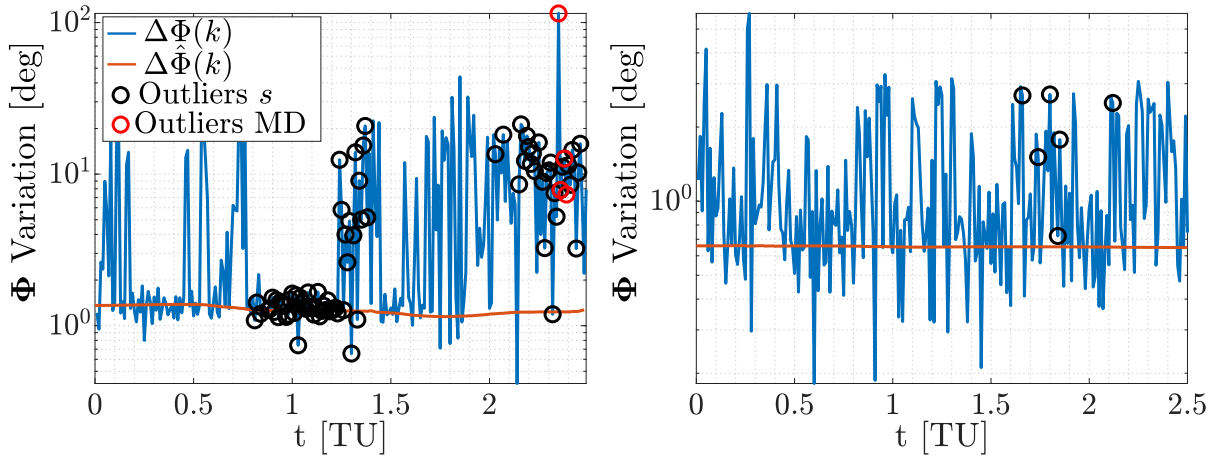

Fig. 14 Variation of principal angle between consecutive timesteps with two different estimation techniques described in Section IV.C for symmetric (left) and asymmetric (right) targets.

reported in Fig. 19, where each line represents a different percentile of the iterations converging in less than the time reported on the ordinate axis. The color bar on the right hand side of the image is used to discern these percentiles. The figure clearly shows that 1% of the iterations are responsible for the greater computational time, since they provide a larger increase of computational time than the gaps obtained between lower percentiles. In particular, in the best case scenario of a 1.5 GHz clock speed, a measurement frequency of 0.34 Hz would be enough to guarantee convergence of 99% of the iterations. On the other hand, if it is necessary to guarantee convergence in 100% of the iterations, this sampling frequency needs to be lowered to 0.30 Hz (i.e. a measurement every 3.3 seconds). By adding the computational time for other on board applications, it is reasonable to assume a maximum measurement frequency of 0.25 Hz, that is a measurement every 4 seconds.

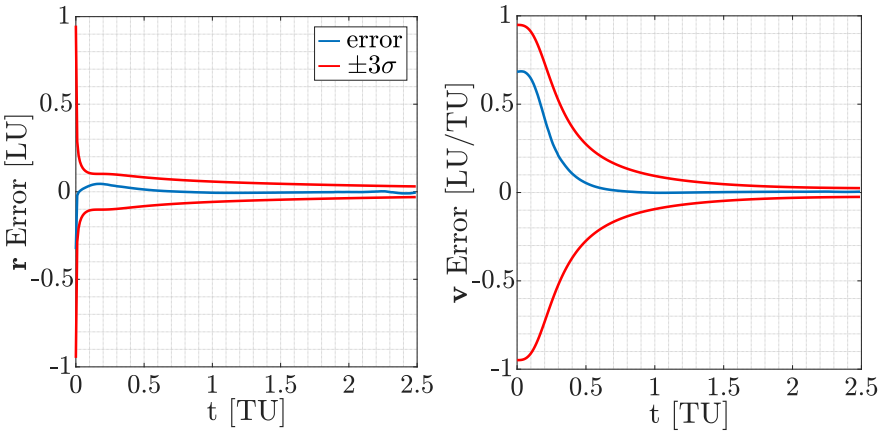

Fig. 15 Convergence of the translational filter in the symmetric case with the adapted axisymmetric rigid body state presented in Section V.

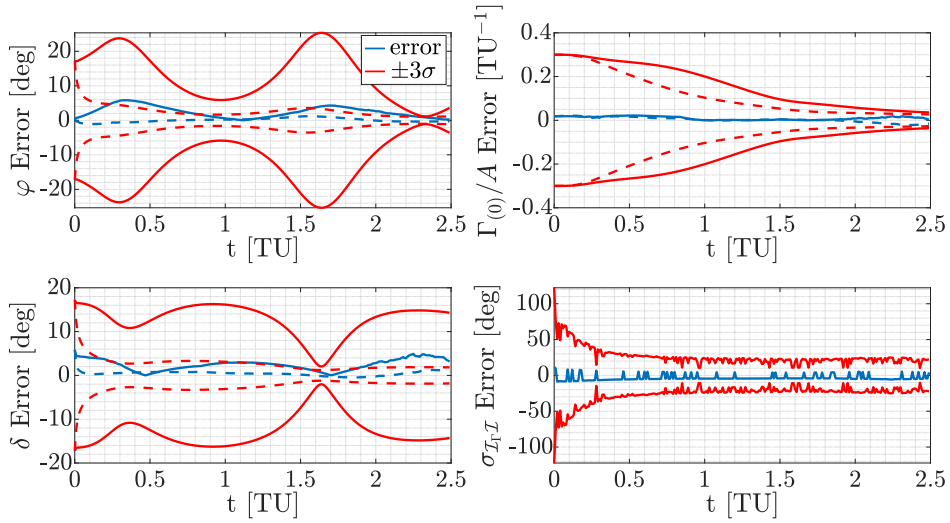

Fig. 16 Convergence of the rotational filter in the symmetric case using the adapted axisymmetric rigid body state introduced in Section V.

Once an estimate of the maximum acquisition rate is available, the capability of a chaser employing the CoMBiNa method to cover the population of LEO RSOs can be estimated. The analysis is performed through simulated measurements to provide an estimate of the maximum achievable performance using CoMBiNa algorithm. As observed in the previous subsection, the maximum frequency of measurements is roughly 0.33 Hz, which is assumed as the rate for these simulations in order to retrieve the maximum achievable performance. A range of 10 different levels of tumbling frequency of the target RSO are taken into account for this analysis. For each of these ten regimes, the analysis takes into account a propagation of 100 different initial conditions taken randomly by varying $\mathbf{k}_{\mathcal{T}}$, $\boldsymbol{\sigma}_{\mathcal{T}}$, and the direction of $\boldsymbol{\omega}_{\mathcal{T}}$ (whose module is given by the assumed tumbling frequency amplitude). Each initial condition is propagated up to a final time horizon of half a chaser's orbit to reduce the computational time and effort. The convergence is assessed in terms of final Mahalanobis Distance. In particular, if the errors of the last 10 timesteps have a probability of at least

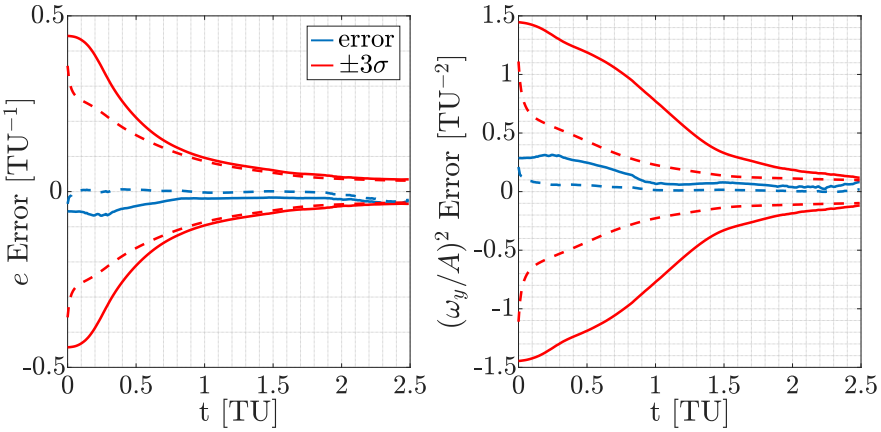

Fig. 17 Estimation of additional attitude parameters with the post-processing of the symmetric attitude filter.

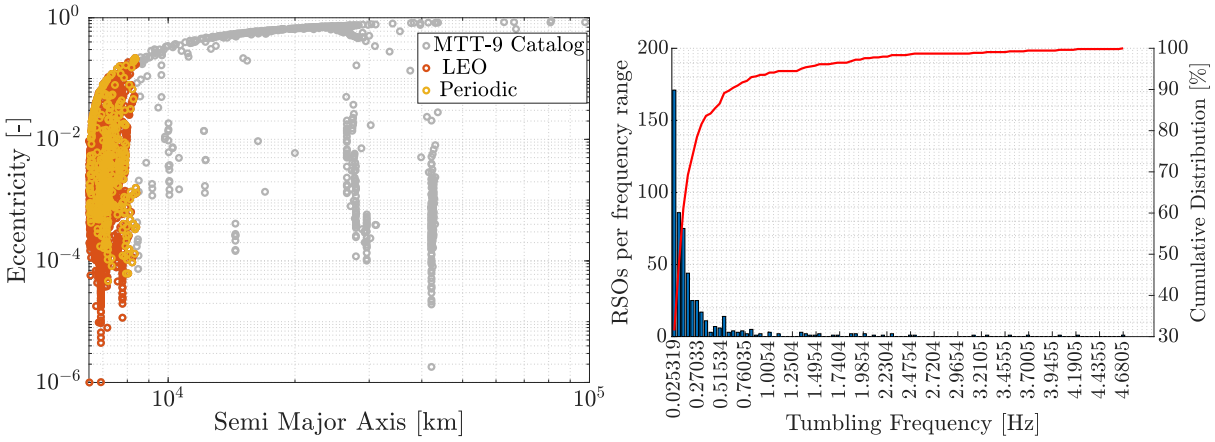

Fig. 18 MTT-9 dataset characterization. Periodic objects (left) and tumbling rates distribution (right).

99% of happening under the current assumption of Gaussian distribution, then the simulation is considered converged. The results of this analysis are summarized in Fig. 20.

This figure shows that for each of the initial tumbling rates of the RSO below 1.05 Hz, the CoMBiNa navigation filter works well and is capable of converging in ~ 90% of the iterations, as shown by the bar charts. After this frequency, there is a sudden drop in convergence for the algorithm, which continues to decrease as the RSO's tumbling frequency approaches 1.5 Hz. As observed from the figure, the higher concentration of RSOs tumbling frequencies is located below this transition area. The red numbers reported above the bars represent the percentage of the RSO dataset which have a frequency lower than the one under consideration. In particular, the algorithm proved to be efficient enough to cover up to ~ 93% of the dataset. In other words, even if there are RSOs with higher tumbling rates measured in the dataset, those account for less than ~ 7% of the population, hence making the use of CoMBiNa appealing.

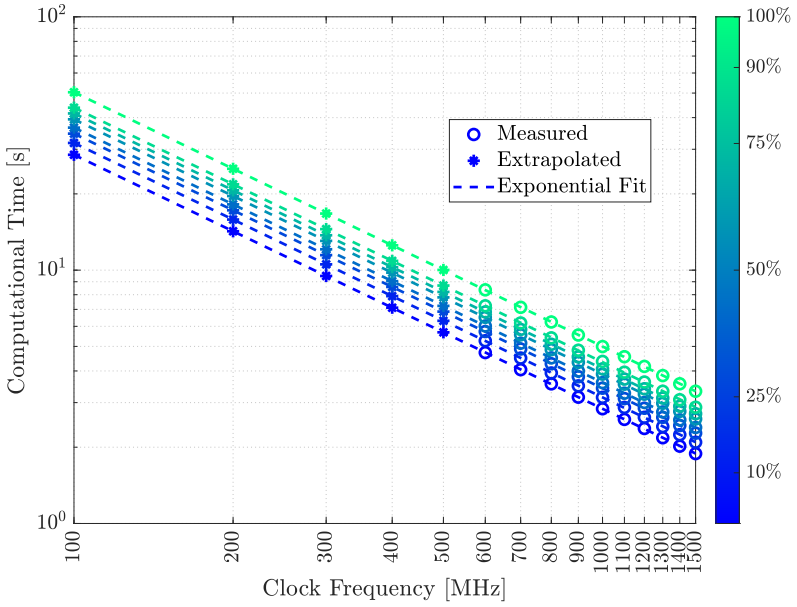

Fig. 19 Computational time of the CoMBiNa navigation filter on 1000 iterations repeated at different clock speeds.

VIII. Conclusions

This paper presents a pipeline for dealing with the navigation problem at an uncooperative and unknown target RSO. The aim of this study is not only to provide an estimate of the state of the target, but also an estimation of its inertia matrix up to a scaling factor. The proposed solution to the problem is formulated in the CoMBiNa algorithm. CoMBiNa combines the benefits of large maps obtained by state of the art graph-SLAM approaches with the prediction capability of sequential filters. In particular, the proposed algorithm relies on validated concepts for model-based relative navigation, extending their applicability to a challenging scenario in which the model has to be built online. In order to do so, the mission is split in two phases. During the first preliminary phase, a 3D point cloud model of the target is built. Subsequently, this coarse model is used in a model-based relative navigation approach exploiting statistical point set registration methods. The algorithm also leverages the structure of the point set registration algorithm to automatically detect and reject outliers. The key idea behind the algorithm is that it does not rely on robust feature matching with a known target model, which is one of the most challenging steps in standard model-based relative navigation. On the contrary, this work presents a solution to this issue by providing a statistical matching of the measured and stored models, so that even in the case of uncertain features, the relative state and position can still be obtained. The main flaw of this approach is that it cannot tackle objects with geometrically unobservable rotations (i.e., symmetric targets). In these cases, BCPD cannot predict attitude measurements. Nonetheless, this paper proposed a method to automatically detect this behavior based on two concurring estimates of the principal rotation angle between consecutive time frames. If this behavior is detected, an alternative approach is proposed that relies on the assumption of axisymmetric rigid body

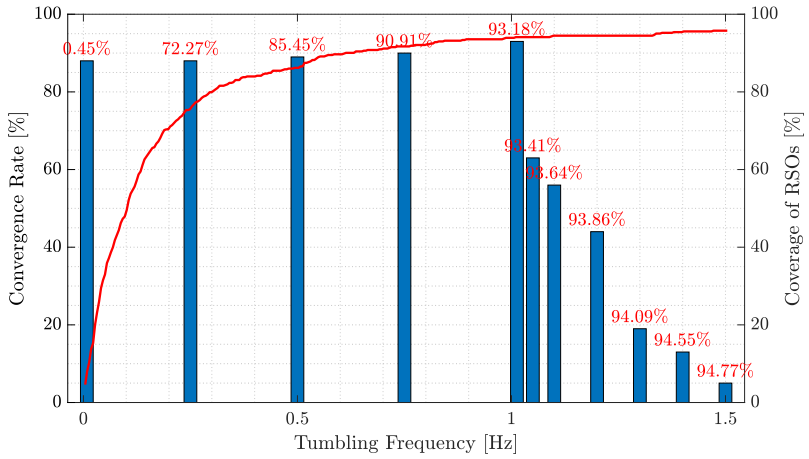

Fig. 20 Coverage of the RSOs in LEO retrieved from the MTT-9 dataset. Numbers in red represent the percentage of this dataset covered by the corresponding frequency.

to retrieve at least the direction of the symmetry axis. The algorithm relies on the assumptions of unperturbed attitude and orbital dynamics with perfect knowledge of the chaser's state orbiting on a circular trajectory with small relative separation from the RSO. Under these simplifying assumptions, the proposed algorithm provided promising results in simulated test cases, and it was also demonstrated on a SBC with limited resources while considering tumbling rates representative of those observed in the LEO population of debris.

This work relies on several simplifying assumptions that make the application of CoMBiNa to real mission scenarios far from the needed level of maturity. Nevertheless, this paper aims at providing a much needed first step to tackle the problem of navigation about unknown and uncooperative RSOs. In its current version CoMBiNa is also not free from limitations. First of all, the fact that the image simulator is not integrated in the loop, prevented the MC analysis that would be needed to extensively validate the approach. Indeed, at the moment, trajectories and measurements need to be simulated beforehand. This limitation also excluded the possibility of applying a chaser closed loop control of the relative trajectory and attitude, so that better tracking of the target could be achieved. This application would also challenge the hypothesis that orbital and attitude state can be decoupled for such close proximity operations, where translation-attitude coupling may arise. More research is needed in this direction and in fact this still remains one of the most interesting open areas of investigation. Secondly, the algorithm makes important assumptions on the capability of taking images and processing them with stereo vision algorithms (i.e., ideal illumination and camera model), whose feasibility should be additionally studied on an experimental testbed. By adopting realistic hardware and sensor performances it would be possible to better capture the sensitivity to key design parameters. For example the baseline of the camera is known to impact the accuracy of the depth estimation. Using a real instrument would surely affect the current performances obtained in the ideal setup adopted for this study. In addition, the performance of the measurement system consisting of image processing and point cloud registration has not been properly evaluated.

At the current level of implementation, some constant values were assumed and used as tuning parameters for the measurement noise covariance. However, a statistical analysis should be carried out to properly characterize these quantities. The reason for this shortcoming is once again to be imputed to the lack of integration of the image simulation and processing pipeline. Carrying out this analysis at the current level of implementation would require a time and storage intensive process. Finally, further analysis can be made to understand whether the proposed axisymmetric formulation of CoMBiNa can be extended to a more general case, since in its current formulation it can only deal with ideal and unperturbed rigid body motion. Current research efforts are aimed at applying this paradigm to the case where the target, while geometrically symmetric, has a density such that its third moment of inertia is slightly different from the first (and thus does not fall within the definition of axisymmetry), by simply augmenting the attitude state vector.

References

[1] ESA Space Debris Office, "ESA'S Annual Space Environment Report," Tech. Rep. GEN-DB-LOG-00288-OPS-SD, European Space Agency, 2022. Iss. 6.

[2] Committee on the Peaceful Uses of Outer Space, "Report of the Scientific and Technical Subcommittee on its fiftieth session, held in Vienna from 11 to 22 February 2013," Tech. Rep. A/AC.105/1038, United Nations, Office for Outer Space Affairs, 2013. STSC 50th session.

[3] Kessler, D., Johnson, N., Liou, J. C., and Matney, M., "The Kessler Syndrome: Implications to Future Space operations," *Advances in the Astronautical Sciences*, Vol. 137, 2010, pp. 47–62.

[4] Liou, J. C., Anilkumar, A. K., Bastida, B., Hanada, T., Krag, H., Lewis, H., Raj, M., Rao, M., Rossi, A., and Sharma, R., "Stability of the future LEO environment–an IADC comparison study," *Proceedings of the 6th European Conference on Space Debris, 22-25 April, 2013, ESOC, Darmstadt, Germany*, Vol. 6, edited by L. Ouwehand, ESA Space Debris Office, 2013, pp. 1–7. URL `https://conference.sdo.esoc.esa.int/proceedings/sdc6/paper/199`.

[5] Saleh, J. H., Lamassoure, E., and Hastings, D. E., "Space Systems Flexibility Provided by On-Orbit Servicing: Part 1," *Journal of Spacecraft and Rockets*, Vol. 39, No. 4, 2002, pp. 551–560. https://doi.org/10.2514/2.3844.

[6] Forshaw, J., Lopez, R., Okamoto, A., Blackerby, C., and Okada, N., "The ELSA-d End-of-life Debris Removal Mission: Mission Design, In-flight Safety, and Preparations for Launch," *Proceedings of the Advanced Maui Optical and Space Surveillance Technologies Conference, 17-20 September, 2019, Maui, Hawaii*, edited by S. Ryan, AMOS, 2019, pp. 44 – 50.

[7] Akin, D., and Sullivan, B., "A survey of serviceable spacecraft failures," *AIAA Space 2001 Conference and Exposition, 28-30 August, 2001, Albuquerque, New Mexico*, AIAA, 2001, pp. 1–9. https://doi.org/10.2514/6.2001-4540, Paper No. AIAA 2001-4540.

[8] Larbi, B., Grzesik, M. K., Radtke, B., Trentlage, C. J., and Stoll, E., "Active debris removal for mega constellations: Cubesat

possible," *Proceedings of the 9th International Workshop on Satellite Constellations and Formation Flying, IWSCFF2017, 19-21 June, 2017, Boulder, Colorado*, IAF, 2017, pp. 1–19. Paper No. IWSCFF 17-20.

[9] Starek, J. A., Açıkmeşe, B., Nesnas, I. A., and Pavone, M., *Spacecraft Autonomy Challenges for Next-Generation Space Missions*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2016, Lecture Notes on Control and Information Sciences, Vol. 460, Chap. 1, pp. 1–48. https://doi.org/10.1007/978-3-662-47694-9_1.

[10] Yoshida, K., Hashizume, K., Nenchev, D., Inaba, N., and Oda, M., "Control of a space manipulator for autonomous target capture - ETS-VII flight experiments and analysis," *AIAA Guidance, Navigation, and Control Conference and Exhibit, 14-17 August, 2000, Denver, Colorado*, AIAA, 2000, pp. 1–10. https://doi.org/10.2514/6.2000-4376, Paper No. AIAA-2000-4376.

[11] Friend, R. B., "Orbital Express program summary and mission overview," *Sensors and Systems for Space Applications II, SPIE Defense and Security Symposium, 16-20 March, 2008, Orlando, Florida*, Vol. 6958, edited by R. T. Howard and P. Motaghedi, International Society for Optics and Photonics, SPIE, 2008, pp. 11–21. https://doi.org/10.1117/12.783792.

[12] Rumford, T. E., "Demonstration of autonomous rendezvous technology (DART) project summary," *Space Systems Technology and Operations, Aerosense 2003, 21-25 April, 2003, Orlando, Florida*, Vol. 5088, edited by P. Tchoryk and J. Shoemaker, International Society for Optics and Photonics, SPIE, 2003, pp. 10–19. https://doi.org/10.1117/12.498811.

[13] D'Amico, S., Ardaens, J. S., and Larsson, R., "Spaceborne Autonomous Formation-Flying Experiment on the PRISMA Mission," *Journal of Guidance, Control, and Dynamics*, Vol. 35, No. 3, 2012, pp. 834–850. https://doi.org/10.2514/1.55638.

[14] Pedrotty, S., Sullivan, J., Gambone, E., and Kirven, T., "Seeker Free-Flying inspector GNC System Overview," *Advances in the Astronautical Sciences*, Vol. 169, 2019, pp. 1081–1094.

[15] Ventura, J., Ciarcià, M., Romano, M., and Ulrich, W. H., "Fast and Near-Optimal Guidance for Docking to Uncontrolled Spacecraft," *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 12, 2017, pp. 3138–3154. https://doi.org/10.2514/1.G001843.

[16] Christian, J. A., Robinson, S. B., D'Souza, C. N., and Ruiz, J. P., "Cooperative Relative Navigation of Spacecraft Using Flash Light Detection and Ranging Sensors," *Journal of Guidance, Control, and Dynamics*, Vol. 37, No. 2, 2014, pp. 452–465. https://doi.org/10.2514/1.61234.

[17] Wu, Y., and Hu, Z., "PnP Problem Revisited," *Journal of Mathematical Imaging and Vision*, Vol. 24, No. 1, 2006, pp. 131–141. https://doi.org/10.1007/s10851-005-3617-z.

[18] Tanygin, S., and Shuster, M., "The many TRIAD algorithms," *Advances in the Astronautical Sciences*, Vol. 122, 2007, pp. 81–99.

[19] Markley, F. L., and Mortari, D., "Quaternion Attitude Estimation Using Vector Observations," *The Journal of the Astronautical Sciences*, Vol. 48, No. 2, 2000, pp. 359–380. https://doi.org/10.1007/BF03546284.

[20] Piazza, M., Maestrini, M., and Di Lizia, P., "Monocular Relative Pose Estimation Pipeline for Uncooperative Resident Space Objects," *Journal of Aerospace Information Systems*, Vol. 19, No. 9, 2022, pp. 613–632. https://doi.org/10.2514/1.I011064.

[21] David, P., DeMenthon, D., Duraiswami, R., and Samet, H., "SoftPOSIT: Simultaneous Pose and Correspondence Determination," *International Journal of Computer Vision*, Vol. 59, No. 3, 2004, pp. 259–284. https://doi.org/10.1023/B:VISI.0000025800.10423.1f.

[22] Fischler, M. A., and Bolles, R. C., "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography," *Communications of the ACM*, Vol. 24, No. 6, 1981, pp. 381—395. https://doi.org/10.1145/358669.358692.

[23] Comellini, A., Mavé, F., Dubanchet, V., Casu, D., Zenou, E., and Espinosa, C., "Robust Navigation Solution for Vision-Based Autonomous Rendezvous," *Proceedings of the 2021 IEEE Aerospace Conference (50100), 6-13 March, 2021, Big Sky, Montana*, IEEE, 2021, pp. 1076–1089. https://doi.org/10.1109/AERO50100.2021.9438241.

[24] Sharma, S., and D'Amico, S., "Neural Network-Based Pose Estimation for Noncooperative Spacecraft Rendezvous," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 56, No. 6, 2020, pp. 4638–4658. https://doi.org/10.1109/TAES.2020.2999148.

[25] Pasqualetto Cassinis, L., Menicucci, A., Gill, E., Ahrns, I., and Sanchez-Gestido, M., "On-ground validation of a CNN-based monocular pose estimation system for uncooperative spacecraft: Bridging domain shift in rendezvous scenarios," *Acta Astronautica*, Vol. 196, 2022, pp. 123–138. https://doi.org/https://doi.org/10.1016/j.actaastro.2022.04.002.

[26] Pasqualetto Cassinis, L., Fonod, R., Gill, E., Ahrns, I., and Gil-Fernández, J., "Evaluation of tightly- and loosely-coupled approaches in CNN-based pose estimation systems for uncooperative spacecraft," *Acta Astronautica*, Vol. 182, 2021, pp. 189–202. https://doi.org/https://doi.org/10.1016/j.actaastro.2021.01.035.

[27] Liu, C., and Hu, W., "Relative pose estimation for cylinder-shaped spacecrafts using single image," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 50, No. 4, 2014, pp. 3036–3056. https://doi.org/10.1109/TAES.2014.120757.

[28] Salvi, J., Matabosch, C., Fofi, D., and Forest, J., "A review of recent range image registration methods with accuracy evaluation," *Image and Vision Computing*, Vol. 25, No. 5, 2007, pp. 578–596. https://doi.org/10.1016/j.imavis.2006.05.012.

[29] Opromolla, R., Fasano, G., Rufino, G., and Grassi, M., "Pose Estimation for Spacecraft Relative Navigation Using Model-Based Algorithms," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 53, No. 1, 2017, pp. 431–447. https://doi.org/10.1109/TAES.2017.2650785.

[30] Naasz, B. J., Burns, R. D., Queen, S. Z., Van Eepoel, J., Hannah, J., and Skelton, E., "The HST SM4 Relative Navigation Sensor System: Overview and Preliminary Testing Results from the Flight Robotics Lab," *The Journal of the Astronautical Sciences*, Vol. 57, No. 1, 2009, pp. 457–483. https://doi.org/10.1007/BF03321512.

[31] Pasqualetto Cassinis, L., Fonod, R., and Gill, E., "Review of the robustness and applicability of monocular pose estimation systems for relative navigation with an uncooperative spacecraft," *Progress in Aerospace Sciences*, Vol. 110, 2019, p. 100548. https://doi.org/https://doi.org/10.1016/j.paerosci.2019.05.008.

[32] Durrant-Whyte, H., and Bailey, T., "Simultaneous localization and mapping: part I," *IEEE Robotics Automation Magazine*, Vol. 13, No. 2, 2006, pp. 99–110. https://doi.org/10.1109/MRA.2006.1638022.

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

[34] Capuano, V., Kim, K., Harvard, A., and Chung, S. J., "Monocular-based pose determination of uncooperative space objects," *Acta Astronautica*, Vol. 166, 2020, pp. 493–506. https://doi.org/10.1016/j.actaastro.2019.09.027.

[35] Lichter, M., and Dubowsky, S., "State, shape, and parameter estimation of space objects from range images," *IEEE International Conference on Robotics and Automation, 26 April - 1 May, 2004, New Orleans, Louisiana*, Vol. 3, IEEE, 2004, pp. 2974–2979. https://doi.org/10.1109/ROBOT.2004.1307513.

[36] Mur-Artal, R., and Tardós, J. D., "ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras," *IEEE Transactions on Robotics*, Vol. 33, No. 5, 2017, pp. 1255–1262. https://doi.org/10.1109/TRO.2017.2705103.

[37] Dor, M., and Tsiotras, P., "ORB-SLAM Applied to Spacecraft Non-Cooperative Rendezvous," *2018 Space Flight Mechanics Meeting*, American Institute of Aeronautics and Astronautics, Kissimmee, Florida, 2018, pp. 1–17. https://doi.org/10.2514/6.2018-1963.

[38] Tweddle, B. E., "Computer vision-based localization and mapping of an unknown, uncooperative and spinning target for spacecraft proximity operations," Ph.D. thesis, Massachussetts Institute of Technology, 12 2013. URL https://dspace.mit.edu/handle/1721.1/85693.

[39] Gaylor, D. E., and Barbee, B. W., "Algorithms for safe spacecraft proximity operations," *Advances in the Astronautical Sciences*, Vol. 127, 2007, pp. 133–152.

[40] Hirose, O., "A Bayesian Formulation of Coherent Point Drift," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 43, No. 7, 2021, pp. 2269–2286. https://doi.org/10.1109/TPAMI.2020.2971687.

[41] Wan, E. A., and Van Der Merwe, R., "The unscented Kalman filter for nonlinear estimation," *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373), 1-4 October, 2000, Lake Louise, Alberta Canada*, IEEE, 2000, pp. 153–158. https://doi.org/10.1109/ASSPCC.2000.882463.

[42] Gaias, G., and Ardaens, J.-S., "Flight Demonstration of Autonomous Noncooperative Rendezvous in Low Earth Orbit," *Journal of Guidance, Control, and Dynamics*, Vol. 41, No. 6, 2018, pp. 1337–1354. https://doi.org/10.2514/1.G003239.

[43] Pugliatti, M., Maestrini, M., Di Lizia, P., and Topputo, F., "Onboard Small-Body semantic segmentation based on morphological features with U-Net," *31st AAS/AIAA Space Flight Mechanics Meeting, February 1–4, 2021, virtual event*, AAS, 2021, pp. 1 – 20. URL https://tinyurl.com/37w5vrak, Paper No. AAS 21-378.

[44] Scorsoglio, A., D'Ambrosio, A., Ghilardi, L., Gaudet, B., Curti, F., and Furfaro, R., "Image-Based Deep Reinforcement Meta-Learning for Autonomous Lunar Landing," *Journal of Spacecraft and Rockets*, Vol. 59, No. 1, 2022, pp. 153–165. https://doi.org/10.2514/1.A35072.

[45] Maestrini, M., and Di Lizia, P., "COMBINA: Relative Navigation for Unknown Uncooperative Resident Space Object," *AIAA Scitech Forum, 3-7 January 2022, San Diego, CA*, AIAA, 2022, pp. 1–19. https://doi.org/10.2514/6.2022-2384, Paper No. AIAA 2022-2384.

[46] Bay, H., Tuytelaars, T., and Van Gool, L., "SURF: Speeded Up Robust Features," *Proceedings of the 9th European Conference on Computer Vision, 7-13 May, 2006, Graz, Austria*, edited by A. Leonardis, H. Bischof, and A. Pinz, Springer, Berlin, Heidelberg, 2006, pp. 404–417. https://doi.org/10.1007/11744023_32.

[47] Pugliatti, M., and Topputo, F., "Navigation about irregular bodies through segmentation maps," *31st AAS/AIAA Space Flight Mechanics Meeting, February 1–4, 2021, virtual event*, AAS, 2021, pp. 1–19. URL https://tinyurl.com/5y9jm87a, Paper No. AAS 21-383.

[48] Sanchez, J. C., Louembet, C., Gavilan, F., and Vazquez, R., "Event-Based Impulsive Control for Spacecraft Rendezvous Hovering Phases," *Journal of Guidance, Control, and Dynamics*, Vol. 44, No. 10, 2021, pp. 1794–1810. https://doi.org/10.2514/1.G005507.

[49] Horri, N. M., Kristiansen, K. U., Palmer, P., and Roberts, M., "Relative attitude dynamics and control for a satellite inspection mission," *Acta Astronautica*, Vol. 71, 2012, pp. 109–118. https://doi.org/10.1016/j.actaastro.2011.07.029.

[50] Maestrini, M., and Di Lizia, P., "Guidance Strategy for Autonomous Inspection of Unknown Non-Cooperative Resident Space Objects," *Journal of Guidance, Control, and Dynamics*, Vol. 45, No. 6, 2022, pp. 1126–1136. https://doi.org/10.2514/1.G006126.

[51] Piazza, E., Romanoni, A., and Matteucci, M., "Real-Time CPU-Based Large-Scale Three-Dimensional Mesh Reconstruction," *IEEE Robotics and Automation Letters*, Vol. 3, No. 3, 2018, pp. 1584–1591. https://doi.org/10.1109/LRA.2018.2800104.

[52] Driver, T., Dor, M., Skinner, K. A., and Tsiotras, P., "Shape Carving in Space: A Visual-Slam Approach to 3D Shape Reconstruction of a Small Celestial Body," *Advances in the Astronautical Sciences*, Vol. 175, 2020, pp. 1619–1638.

[53] Eppstein, D., Paterson, M. S., and Yao, F. F., "On Nearest-Neighbor Graphs," *Discrete & Computational Geometry*, Vol. 17, No. 3, 1997, pp. 263–282. https://doi.org/10.1007/PL00009293.

[54] Pan, Q., Reitmayr, G., and Drummond, T., "ProFORMA: Probabilistic Feature-based On-line Rapid Model Acquisition," *Proceedings of the British Machine Vision Conference, 7-10 September, 2009, London, United Kingdom*, edited by A. Cavallaro, S. Prince, and D. Alexander, BMVA Press, 2009, pp. 112.1–112.11. https://doi.org/10.5244/C.23.112.

[55] Myronenko, A., and Song, X., "Point Set Registration: Coherent Point Drift," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 32, No. 12, 2010, pp. 2262–2275. https://doi.org/10.1109/TPAMI.2010.46.

[56] Hughes, P. C., *Spacecraft Attitude Dynamics*, 1st ed., Dover, Mineola, NY, 2004, Chap. 2, pp. 6–38.

[57] Pesce, V., Haydar, M. F., Lavagna, M., and Lovera, M., "Comparison of filtering techniques for relative attitude estimation of uncooperative space objects," *Aerospace Science and Technology*, Vol. 84, 2019, pp. 318–328. https://doi.org/10.1016/j.ast.2018.10.031.

[58] Schaub, H., and Junkins, J. L., *Analytical Mechanics of Space Systems*, 2nd ed., AIAA Education Series, Reston, VA, 2009. https://doi.org/10.2514/4.867231.

[59] Sullivan, J., Grimberg, S., and D'Amico, S., "Comprehensive Survey and Assessment of Spacecraft Relative Motion Dynamics Models," *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 8, 2017, pp. 1837–1859. https://doi.org/10.2514/1.G002309.

[60] Karlgaard, C. D., and Schaub, H., "Nonsingular Attitude Filtering Using Modified Rodrigues Parameters," *The Journal of the Astronautical Sciences*, Vol. 57, No. 4, 2009, pp. 777–791. https://doi.org/10.1007/BF03321529.

[61] Hermann, R., and Krener, A., "Nonlinear controllability and observability," *IEEE Transactions on Automatic Control*, Vol. 22, No. 5, 1977, pp. 728–740. https://doi.org/10.1109/TAC.1977.1101601.

[62] De Luca, M. A., "Relative Navigation at Unknown Uncooperative Symmetric Objects," Master's thesis, Politecnico di Milano, 10 2022.

[63] Cavenago, F., Di Lizia, P., Massari, M., and Wittig, A., "On-board spacecraft relative pose estimation with high-order extended Kalman filter," *Acta Astronautica*, Vol. 158, 2019, pp. 55–67. https://doi.org/https://doi.org/10.1016/j.actaastro.2018.11.020.

[64] Servadio, S., Cavenago, F., Di Lizia, P., and Massari, M., "Nonlinear Prediction in Marker-Based Spacecraft Pose Estimation with Polynomial Transition Maps," *Journal of Spacecraft and Rockets*, Vol. 59, No. 2, 2022, pp. 511–523. https://doi.org/10.2514/1.A35068.

[65] Bodin, P., Noteborn, R., Larsson Nordström, R., Karlsson, T., D'Amico, S., Ardaens, J.-S., Delpech, M., and Berges, J.-C., "Prisma Formation Flying Demonstrator: Overview and Conclusions from the Nominal Mission," American Astronautical Society, Santa Diego, CA, 2012, pp. 441–460.

[66] Noteborn, R., Bodin, P., Larsson, R., and Chasset, C., "Flight results from the PRISMA optical line of sight based autonomous rendezvous experiment," *4th international conference on spacecraft formation flying missions & technologies, St-Hubert, Quebec*, 2011, pp. 18–20. QC 20150506.

[67] Beskin, G. M., Karpov, S. V., Biryukov, A. V., Bondar, S. F., Ivanov, E. A., Katkova, E. V., Orekhova, N. V., Perkov, A. V., and Sasyuk, V. V., "Wide-field optical monitoring with Mini-MegaTORTORA (MMT-9) multichannel high temporal resolution telescope," *Astrophysical Bulletin*, Vol. 72, No. 1, 2017, pp. 81–92. https://doi.org/10.1134/S1990341317030105.