M. Maestrini, P. Di Lizia
Guidance Strategy for Autonomous Inspection of Unknown Non-Cooperative Resident Space Objects
Journal of Guidance Control and Dynamics, In press - Published online 01/12/2021
doi:10.2514/1.G006126

Guidance Strategy for Autonomous Inspection of Unknown Non-Cooperative Resident Space Objects

Michele Maestrini*
Polytechnic University of Milan, 20156 Milan, Italy

Pierluigi Di Lizia†
Polytechnic University of Milan, 20156 Milan, Italy

As space debris has become a cause of concern for space operations around Earth, active debris removal and satellite servicing missions have gained increasing attention. Within this framework, in specific scenarios, the chaser might be asked to operate autonomously in the vicinity of a non-cooperative, unknown target. This paper presents a sampling-based receding-horizon motion planning algorithm that selects inspection maneuvers while taking many complex constraints into account. The proposed guidance solution is compared with classical approaches and it is shown to take advantage of the characteristics of the natural dynamics of the relative motion to outperform them. In addition, the impact of different input sampling exploration strategies is explored to propose a simple and more robust approach based on subset simulation.

I. Introduction

Over the past few decades, space debris has become a threat for space operations in Low Earth Orbit (LEO). The average rate of four to five break-ups per year occurring in orbit is a hint at the fact that the number of debris is steadily increasing, as well as the probability of in-orbit collisions ‡. Consequently, among the relevant mitigation actions, active debris removal missions have gained increasing attention. Besides, unmanned on-orbit servicing missions are currently being investigated for their commercial attractiveness: expensive satellites in Geostationary Earth Orbit (GEO) and constellations may benefit from the presence of autonomous on-orbit servicing and inspecting. This interest has been largely demonstrated by commercial servicing programs proposed by both private companies like Infinite Orbits or Astroscale with the ELSA-D mission [1], as well as by public agencies (e.g. NASA's *Restore-L* mission [2]),

*PhD Candidate, Department of Aerospace Science and Technology, michele.maestrini@polimi.it.
†Assistant Professor, Department of Aerospace Science and Technology, pierluigi.dilizia@polimi.it, AIAA Member.
Part of this work was presented as Paper No. AAS 20-431 at the 2020 AAS/AIAA Astrodynamics Specialist Conference, August 9 - 12, 2020, South Lake Tahoe, California, (virtual event).
‡http://www.esa.int/Safety_Security/Space_Debris/About_space_debris (Accessed on April 12, 2021)

with the success of MEV-1 mission[§]. In order to perform such feats, inspecting the target resident space object (RSO) is a fundamental, prior step required to proceed with any proximity operation. Moreover, performing the inspection task autonomously has been identified by NASA as a key enabling technology for next generation space missions [3]. In fact, autonomy would grant savings in operational costs for long missions, as well as the reduction of the detrimental impacts of human errors and communication delays. The capability of performing proximity operations enabled missions such as the Apollo program, the servicing flights of the Space Shuttle, and the assembly and supply of the space stations [4]. However, in these missions, the procedures have been mainly performed by the crew, and involved cooperation with the target spacecraft. The first autonomous experiments started with NASDA's ETS-VII mission [5], which represented the first unmanned satellite to perform independently docking maneuvers. Then, the Air Force Research Laboratory developed a sequence of small satellites (i.e. XSS-10, XSS-11, and MiTEx), which demonstrated the possibility to perform autonomous proximity operations using optical navigation as well as to inspect a failed satellite (i.e. DSP-23). However little is disclosed to the public about these missions. Almost at the same time of these demonstrators, NASA's DART [6] was launched with the aim of performing autonomous rendezvous with and a series of maneuvers in close proximity to a communications satellite no longer in use. Despite the mission being a partial failure, the lessons learned helped in the success of the Orbital Express [7] program, which was launched and deployed in 2007 to validate the technical feasibility of robotic, autonomous on-orbit refueling and reconfiguration of satellites. In 2010, the Swedish Space Corporation's mission PRISMA [8] also addressed the possibility of performing approach, rendezvous, and formation flight with a known object (Tango) via autonomous GNC algorithms. Meanwhile, the Canadian Space Agency (CSA) and NASA demonstrated the TriDAR [9] system for performing relative navigation to an uncooperative but known target (i.e. the ISS). More recently, NASA performed single spacecraft remote inspection of the Cygnus spacecraft during the Seeker-1 [10] technology demonstration in September 2019. In addition, another NASA mission named CPOD [11] is scheduled to launch in 2021, and will demonstrate rendezvous, proximity operations and docking using two CubeSats.

Despite the variety of tasks and platforms, all the above mentioned examples relied on some sort of cooperativeness and/or knowledge of the target which is leveraged to simplify proximity operations planning (e.g. fiducial markers or accurate geometrical models). In this context we will define a cooperative target as an object capable of providing sensor data to the chaser spacecraft (actively-cooperative) or presenting markers on its surface (passively-cooperative) which help the chaser to estimate its relative state. Several different techniques were developed for use on GNC problems for proximity operations with known and cooperative targets, with increasing level of generalization. The most simple task involves a target which is fully cooperative, at least partially stabilized, and provides information regarding its state (e.g. formation flight, docking, etc.) [12–14]. Additionally, several studies focused on the determination of the trajectory

[§]https://news.northropgrumman.com/news/releases/intelsat-901-satellite-returns-to-service-using-northrop-grummans-mission-extension-vehicle (Accessed on April 12, 2021) in close proximity to an RSO which is cooperative and providing information on its state, despite being uncontrolled [15–18]. In [19], the target is assumed to be uncontrolled and uncooperative (at least actively), therefore its state is estimated through the knowledge of some fiducial markers on the docking interface. Moreover, some approaches went beyond their method and assumed complete lack of cooperation and tumbling target, while retaining the capability of estimating the relative state given the knowledge of the target model [20].

The idea of removing cooperation and stabilization of the target to replace it with the knowledge of its model has also been exploited for proximity operations to asteroids [21]. However, these approaches fall short in case the model of the RSO under investigation is not known a priori. This can happen for several reasons, for example due to a change in configuration of the solar panels after the mission was over (e.g. Envisat), due to damages caused by collision with other debris and/or explosion of internal tanks, and due to change of optical properties due to prolonged exposure to the space environment. This leads to the need of developing spacecraft guidance algorithms that can take into account stringent safety and observation constraints, which renders classical trajectory design approaches ineffective. The deployment of the experiment SPHERES-VERTIGO [22] by NASA and its research partners in 2013 was the first example of inspection of an unknown and uncooperative target. In this experiment, a set of free-flying satellites aboard the ISS demonstrated the capability of performing simultaneous localization and mapping of an unknown and uncooperative target. Despite focusing on navigation and shape reconstruction without any trajectory planning strategy, this experiment provided a huge inspiration for this work. Indeed, the aim of this paper is to provide a guidance algorithm for the autonomous selection of inspection maneuvers in the neighborhood of an unknown and uncooperative RSO.

Classical guidance approaches for inspection historically exploit linearized dynamics [23] and natural motion trajectories obtained via impulsive maneuvers to plan relative navigation trajectories with various approaches and techniques including manual design [24, 25], waypoint-following strategies [26], and graph searches with constraints [27]. These standard techniques rely on minimizing maneuver cost and guaranteeing passive safety regardless of an inspection metric, as they assume that no information is available on the target's shape and motion. Moreover, they do not embed autonomy and require extensive human intervention for the design of an optimized strategy, which inevitably lead to sub-optimal performances [28]. To improve these strategies, some recent studies propose Motion Planning algorithms [29] as a way to autonomously select optimized maneuvers [30, 31]. These guidance techniques were developed in the field of autonomous driving and they represent robotics' industry standard thanks to their low computational demand, which makes them suitable for implementation on limited-resource systems. They have been investigated in the framework of satellite inspection in recent years by Capolupo et al. [32, 33]. However, these studies relied on the assumption that the target satellite is known, and that its attitude motion can be estimated and predicted with high accuracy thanks to an accurate relative navigation step. The aim of this work is to remove these assumptions, providing a more flexible and general purpose algorithm that embeds the definition of an inspection metric for unknown non-cooperative RSO. A sampling-based receding-horizon algorithm is developed to plan inspection maneuvers while taking many complex constraints into account. The proposed algorithm represents the most general case of satellite motion planning for inspection under relative state uncertainties, as it removes any assumption on the RSO's shape and tumbling motion. Moreover, the resulting guidance strategy can be suited to both on-board autonomous maneuver selection and offline trajectory design. To tackle the problem, the algorithm relies on Sampling Based Model Predictive Control [34] (SBMPC) as it can easily include black box constraints, and it is robust to uncertainties thanks to the periodic re-planning of maneuvers. The guidance strategy is eventually applied to two particular case studies: the first one in the neighborhood of a controlled nadir-pointing target satellite, whereas the second to investigate a tumbling RSO. In both cases, the performance is compared with classical approaches exploiting natural motion trajectories. The results show that the proposed guidance algorithm takes advantage of the characteristics of the natural dynamics of the relative motion to outperform the classical techniques. Additionally, a sampling refinement strategy based on Subset Simulation (SS) [35] is presented, and its performance is compared with the heuristic approach used in [32, 33].

II. Inspection Mission Concept

The envisioned mission scenario requires to inspect a tumbling resident space object located on a circular low Earth orbit. To do so, the chaser should maximize the time spent in favourable position for acquisition of images and other operations. Therefore, better inspection trajectories should try to move closer to the target and to obtain better illumination conditions, while satisfying mission constraints. The trajectory of the complete inspection mission is subdivided into a sequence of optimized trajectory arcs, each consisting of several phases as depicted in Figure 1. Each arc starts with an initial phase during which the on-board computer of the chaser computes the optimal maneuver in terms of impulsive Δv and arc duration. Once the optimal maneuver has been selected, the attitude of the chaser is changed to the required thrust direction. Subsequently, a third phase during which the maneuver is executed with finite control thrust is foreseen. After the maneuver is completed, the attitude of the chaser is changed again such that its observation axis is aimed at the RSO. Finally, the last phase is dedicated to the inspection procedures. This sequence of operations continues autonomously until the allocated time window for inspection runs out. Notice that each maneuver is computed to optimize one single arc over a finite time window, and the guidance module is periodically called to compute a new optimal trajectory until the mission is completed. For planning purposes, it is assumed that an updated estimate of the relative state and attitude together with their covariances is available to the chaser at the beginning of the first phase. This allows the chaser to plan the trajectory with uncertainty up to a maximum time, after which the prediction will be considered unreliable.

A. Planning Phase

The Planning Phase is the first one to take place during the inspection and it is indicated in Figure 1 as Computation. The proposed inspection guidance relies on a Sampling Based Model Predictive Optimization (SBMPO) algorithm,

Fig. 1 Breakdown of a typical optimization arc into its five basic phases.

inspired by the work of Dunlap et al. [34] and Surovik et al.[36] . During the first phase, the algorithm is applied to compute the optimal maneuver to be executed during the current arc. The idea behind this approach is to build a map that associates a score c to each point of the 4D sampled command space $\mathcal{S}$. The latter is defined as the set of points $s = [\mathbf{\Delta v}, t_{arc}] \in \mathcal{S}$ of 3D impulsive maneuvers and arc duration times for which:

$$\mathcal{S} \subset \mathcal{R}^4 : \left\{ ||\mathbf{\Delta v}|| \in [\Delta v_{min}, \Delta v_{max}] \quad \cup \quad \{\mathbf{0}\}, \quad t_{arc} \in \left[t_{min}^{arc}, t_{max}^{arc}\right] \right\} \tag{1}$$

The maximum and minimum Δv have been selected according to [32] with a value of 90 mm/s and 3 mm/s respectively. Whereas the maximum and minimum time for the arc are selected to be 1 h and 3 h respectively. The score c associated with each command will depend on the propellant used for the maneuver, the exploration made throughout the resulting trajectory $\Gamma(t)$, and the geometry of the trajectory as explained in the following subsection.

For each inspection arc, the algorithm starts its iterations by sampling n_0 commands uniformly from $\mathcal{S}$. The commands are then propagated using the Hill-Clohessy-Wiltshire (HCW) [23] equations in order to evaluate the trajectory in a timely manner, and the resulting trajectories are scored. Then, additional n_s commands are sampled, refining the sampling in the most promising regions of the 4D map. This work proposes a comparison between two different refinement strategies in order to find and explore the most interesting regions of the command space. The first one employs a heuristic described in depth in [32, 33], which uses the observation score, the observation score gradient, and the current level of exploration of $\mathcal{S}$ to bias the sampling. On the other hand, the second approach relies on Subset Simulation (SS) [35], which is a statistical approach to explore a space given its assumed statistical properties. This second approach is presented thoroughly in subsection II.E. In total n_{mesh} refining steps are taken for each inspection leg. Then, the command associated with the highest score c_{opt} is selected. Notice that the chaser only employs simplified CW equations of motion in order to speed up the evaluation of the samples. However, in order to assess the robustness of the trajectory selection to variations with respect to the ideal planning, the maneuver is executed in a higher accuracy simulator. Therefore, this simulated environment exploits Nonlinear Equations of Relative Motion [37]. In addition, the assumption of ideal impulsive maneuvers is removed in the simulated environment, hence it is mandatory to adopt a technique to convert the impulsive maneuver to continuous control. The aim of this conversion technique should be to achieve the same relative trajectory that would be provided by the impulsive maneuver at the end of the maneuver execution phase. This mapping is performed using an analytical approach which exploits polynomial control laws [38]. This particular conversion guarantees also that, by selecting an appropriate maneuver execution time, the maximum thrust available is never violated. Moreover, this approach relies on completely analytical and iteration free techniques, which constitute an appealing solution for usage on limited resources systems. The conversion from impulsive maneuver to continuous thrust becomes then necessary to execute the maneuver in a simulator using. The process of maneuver optimization and execution is repeated until the inspection mission time window is over. An overview of the overall maneuver planning algorithm is given in Algorithm 1.

Algorithm 1 Optimal Maneuver Planning

$S \leftarrow \emptyset$
$C \leftarrow \emptyset$
for $i = 1 : n_{\text{mesh}}$ **do**
  if $C \neq \emptyset$; **then**
    $S_i \leftarrow$ Draw n_0 initial samples from $\mathcal{S}$;
  else
    $S_i \leftarrow$ Refine the sampling with n_s samples from $\mathcal{S}$ using a sampling refinement strategy;
  end if
  $C_i \leftarrow \emptyset$;
  for $k = 1 : |S_i|$ **do**
    $s_k \leftarrow S_i(k)$;
    $\Gamma_k \leftarrow$ Propagate trajectory from initial conditions using s_k;
    if s_k or Γ_k violate the constraints **then**
      $c_k \leftarrow 0$;
    else
      $c_k \leftarrow$ Score trajectory based on Γ_k and s_k;
    end if
    $C_i(k) \leftarrow c_k$
  end for
  $C \leftarrow C \cup C_i$;
  $S \leftarrow S \cup S_i$;
end for
$c_{opt} \leftarrow \max C$;
$s_{opt} \leftarrow$ retrieve value of s corresponding to c_{opt};

B. Score Function

One of the major advantages of SBMPO is that it can handle any kind of scoring function that a specific planning problem may request. The only requirement that must be fulfilled by the scoring function is to return a score equal to zero for non-acceptable trajectories (i.e., trajectories that violate any user-specified constraint or condition). For the inspection problem considered in this work, the score is composed of three terms. The first two terms are directly taken from [33] and they are related to the cost and geometry of the trajectory as specified in Equation 2 and 3 respectively.

$$C_{\Delta v} = \frac{\omega_v}{\omega_v + \Delta v} \tag{2}$$

$$C_\gamma = \omega_\gamma \int_{t_0}^{t^{arc}} f(t) \frac{\pi - \gamma(t)}{\pi} dt \tag{3}$$

In particular, ω_γ and ω_v are weighting coefficients, and $\gamma(t)$ is the angle between the chaser and the Sun direction. In addition, $f(t)$ is a weighting function which encourages trajectories closer to the RSO and it is defined as

$$f(t) = \begin{cases} r/r_{min} & \text{if} \quad r < r_{min} \\ 1 & \text{if} \quad r_{min} \leq r \leq r_{max} \\ (r_{max}/r)^3 & \text{if} \quad r > r_{max} \end{cases} \tag{4}$$

Here, r is the norm of the vector $\mathbf{r}$, relative distance between chaser and target expressed in the LVLH reference frame. In this particular application, the minimum and maximum radius for this function have been selected to be 50m and 300 m respectively.

The following paragraph introduces some useful quantities and definitions which serve the purpose of better identifying the terms necessary for the description of the third component of the score. At the beginning of the planning phase, it is assumed that a navigation update is available which provides an estimate of the relative state and the relative covariances. Future works on the topic should be aimed at investigating the effect of varying state uncertainties retrieved from the navigation module. Indeed, at the moment, the case of relative navigation at unknown and uncooperative RSOs has never been investigated, therefore there is no clear indication on the accuracy that may be retrieved. Moreover, it is assumed that a measurement of 3D landmark positions will be available thanks to the equipped stereo camera which is aligned with $-x$ axis of the target body frame. These landmarks represent visual features of interest which may be extracted from image acquisition either via classical image processing, or through purposely trained Convolutional Neural Networks [39]. In particular, these features are independent from the ones that may be used for navigation purposes, and only influence the selection of the best actions that the chaser should take. The stereo camera will provide a measurement vector $\mathbf{y}_i$ composed of horizontal and vertical pixel coordinates (i.e. u_i and v_i) and disparity d_i obtained from stereo-matching. Thanks to the invertible nature of such measurement, the measure of the i^{th} landmark can be inverted and expressed in the target fixed reference frame as in Equation 5, which is a function of the measure $\mathbf{y}_i$ and an augmented state defined as $\mathbf{x} = [\mathbf{r}, \mathbf{q_{TI}}, \mathbf{q_{CI}}, \mathbf{k}]$. Here, $\mathbf{q_{TI}}$ and $\mathbf{q_{CI}}$ are the quaternions expressing the orientation of the target and chaser fixed body frames with respect to the inertial frame respectively, whereas $\mathbf{k} = [k_1, k_2]$ is the vector that parametrizes the inertia matrix of the target as in [40] . By expressing a vector with a left superscript for its reference frame and with a left subscript representing the origin of the frame, it is possible to express the inverse measurement as

$$\mathbf{h}(\mathbf{x}, \mathbf{y}_i) = {}^{T}_{T}\mathbf{L}_i = (\mathbf{q_{TI}} \times \mathbf{q_{CI}^{-1}}) \otimes {}^{C}_{C}\mathbf{L}_i(\mathbf{y}_i) + (\mathbf{q_{TI}} \times \mathbf{q_{IL}}) \otimes {}^{L}_{T}\mathbf{r} \tag{5}$$

Here, $\otimes$ is the operator that rotates a vector by a quaternion, $\times$ is quaternion multiplication. ${}^{C}_{C}\mathbf{L}_i$ can be obtained from Equation 6 by making use of a baseline b, focal foc and horizontal and vertical pixel densities p_x and p_y respectively, which are among the geometrical parameters of the camera.

$$ {}^{C}_{C}\mathbf{L}_i(\mathbf{y}_i) = \begin{bmatrix} foc\ p_x \frac{b}{d_i}, & b \frac{p_x}{p_y} \frac{v_i}{d_i}, & b \frac{u_i}{d_i} \end{bmatrix}^T \tag{6} $$

Together with the landmarks estimated position, it is also possible to project their position covariance as:

$$ \mathbf{P}_{L,i} = \mathbf{H}_i \mathbf{P} \mathbf{H}_i^T + \mathbf{G}_i \mathbf{R} \mathbf{G}_i^T \tag{7} $$

where $\mathbf{P}$ is the covariance of the augmented state $\mathbf{x}$, $\mathbf{R}$ is the covariance of the measurement noise, whereas the matrices $\mathbf{H}$ and $\mathbf{G}$ can be computed as the jacobians of the inverse measurement equations with respect to the augmented state and the measurement as in Equation 8 and Equation 9 respectively.

$$ \mathbf{H}_i = \frac{\partial \mathbf{h}(\mathbf{x}, \mathbf{y}_i)}{\partial \mathbf{x}} \tag{8} $$

$$ \mathbf{G}_i = \frac{\partial \mathbf{h}(\mathbf{x}, \mathbf{y}_i)}{\partial \mathbf{y}_i} \tag{9} $$

Finally, the third component of the score (which is scaled by a coefficient ω_{exp} similarly to the first two terms of the score) can then be defined by Equation 10 which takes advantage of the described inverse measurements and covariances to plan exploratory maneuvers under uncertainties by exploiting the statistical information provided. The meaning of the terms which appear in Equation 10 are clarified later in this section.

$$ C_{exp} = \omega_{exp} \sum_{j=0}^{n_{visible}} \frac{-1}{\mathrm{Tr}(\mathbf{P}_j)} \log \left[g \left(KOZ \frac{\mathbf{r}_j}{r_j} \right) \right] \tag{10} $$

The score of exploration is computed as the sum over all the $n_{visible}$ time instants of a trajectory $\Gamma(t)$ where the target is sufficiently close (i.e. < 300 m) and not in eclipse. The function $g(\mathbf{x})$ represents a 3D probability density function (PDF), which is updated at the beginning of each planning phase. In fact, once each measured landmark and corresponding covariance are obtained as in Equation 5 and 7, they are added to a memorized set of initialized landmarks. Then, these N landmarks are used in a Gaussian mixture to obtain the density function $g(\mathbf{p})$ as in Equation 11

$$ g(\mathbf{p}) = \frac{1}{N} \sum_{i=1}^{N} \frac{\exp\left((\mathbf{p} - \mathbf{L}_i)^T \mathbf{P}_{L,i}^{-1} (\mathbf{p} - \mathbf{L}_i) \right)}{\sqrt{(2\pi)^3 \det \mathbf{P}_{L,i}}} \tag{11} $$

The interesting property about the covariance in Equation 7 is that, being the points fixed in the target body frame, this matrix will stay constant throughout the planning arc. Therefore, also the function g will only need to be computed once per every maneuver, allowing for a fast evaluation of all the trajectories. Moreover, g is defined so that it is always smaller than 1, hence taking $-\log(g(\mathbf{p}))$ as in Equation 10 will give a higher value when further from higher density regions, encouraging trajectories exploring regions with lower landmark density. Since the distance from the target is already scored in Equation 3, the function g is evaluated on the maximum estimated target size, which coincides with a spherical Keep Out Zone $KOZ = 15$ m, along the line of sight with the chaser at time instant t_j of the trajectory $\mathbf{r}_j/r_j$. The PDF is also weighted by the trace of the matrix $\mathbf{P}_j$, which represents the covariance of the extended state (already presented in Equation 7) at time t_j. The effect of using $1/\mathrm{Tr}(\mathbf{P})$ in Equation 10 is that trajectories featuring higher uncertainties will be given a lower weight. Hence, the exploration score C_{exp} will be higher for those trajectories that explore more and are less uncertain.

C. Constraints

The algorithm takes into account several constraints. First of all, the score of every trajectory $\Gamma(t)$ going further than a distance $R_{max} = 1200$ m is set to zero to avoid orbiting too far from the target. Secondly, the maximum slew rate of the chaser is constrained during phases 2-5 in Figure 1. It is assumed that the chaser is always capable of tracking the desired thrusting direction and observation direction (i.e. during phases 3 and 5), whereas during the slewing maneuvers (i.e. phases 2 and 4) a rest-to-rest slewing maneuver is assumed to be executed between initial and target attitudes. There are many other possible ways of performing these slewing maneuvers. However the rest-to-rest approach has been selected in this work. The choice of this particular family of maneuvers is intended to provide the maximum slew rate in a timely manner, which is paramount to having a fast evaluation of this constraint for a large set of sampled trajectories. Finally, the safety of the trajectory is considered as in [32, 33]. In particular, for an arbitrarily long time $t_{safe} = 8$h, which is much larger than t_{max}^{arc} in order to accommodate enough time to respond to failures or anomalies, it is checked that the distance distance d between the 3σ relative position uncertainty ellipsoid and the target is never lower than the KOZ radius (i.e. $d(t) \geq KOZ$). Unfortunately, the analytical expression for d is not available, hence it is substituted by its minimum bound $d_{min}(t) = \min[d_{cyl}(t), d_{sph}(t)]$.

$$d_{cyl}(t) = \frac{1}{a(t)}||\mathbf{r} \wedge \mathbf{a}|| - 3b(t) \tag{12}$$

$$d_{sph}(t) = r(t) - 3a(t) \tag{13}$$

Here, $a(t)$ and $b(t)$ are the major and median eigenvalues of the relative position covariance. Equation 12 computes the minimum bound as the distance from the infinite cylinder of radius 3 times the median eigenvalue of the position covariance matrix. Said cylinder is aligned with the maximum eigenvector of the position covariance matrix $\mathbf{a}$, whereas Equation 13 computes the bound as the distance from the sphere with radius 3 times the maximum eigenvalue of the position covariance matrix. A graphical illustration of such constraint is reported in Figure 2.

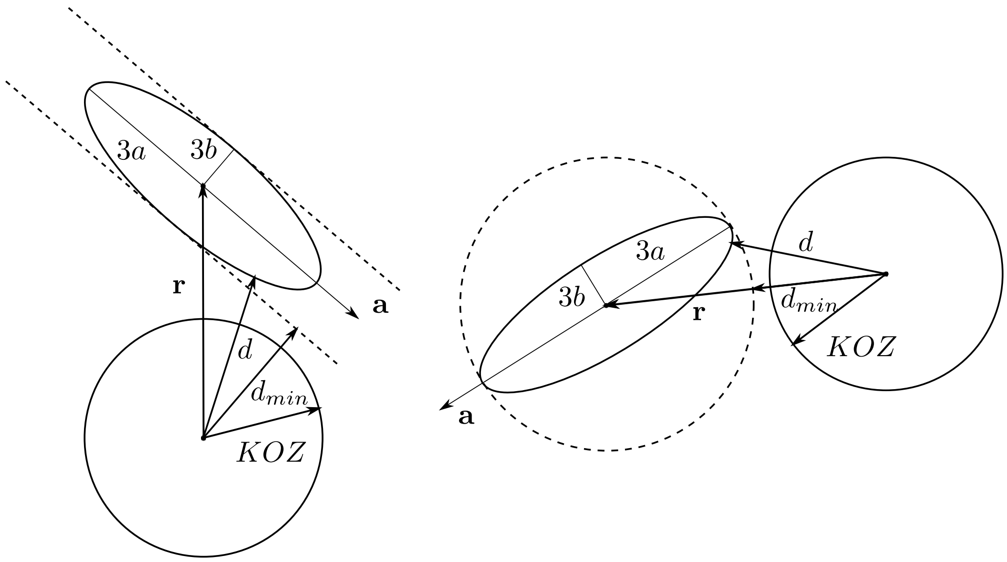

Fig. 2 The 3σ relative position uncertainty ellipsoid is encompassed either by an infinitely long cylinder (left) or by a sphere (right).

For improved efficiency, the covariance is propagated from time t_1 to time t_2 using the HCW state transition matrix $\mathbf{\Phi}(t)$ as in Equation 14 [23] .

$$\mathbf{P}_x(t_2) = \mathbf{\Phi}(t_2 - t_1)\mathbf{P}_x(t_1)\mathbf{\Phi}(t_2 - t_1)^T \tag{14}$$

The covariance is also corrected to account for impulsive maneuvers with the Gates model [41]. The values of σ_p covariance of pointing error and σ_v covariance of maneuver amplitude error are taken from [32, 33] .

$$\mathbf{P}_x(t_m^+) = \mathbf{P}_x(t_m^-) + \begin{bmatrix} \mathbf{0}_{3x3} & \mathbf{0}_{3x3} \\ \mathbf{0}_{3x3} & \sigma_v^2 \mathbf{\Delta v \Delta v}^T + \sigma_p^2 [\mathbf{\Delta v}\times][\mathbf{\Delta v}\times] \end{bmatrix} \tag{15}$$

Here, t_m^+ and t_m^- represent the time instants immediately before and after the impulsive maneuver is applied. Additionally, the symbol $[\mathbf{\Delta v}\times]$ is the matrix outcome of the application of the Levi-Civita operator to the vector $\mathbf{\Delta v}$.

D. Propagation of the Augmented State Covariance

As noted in the first part of this work, computing score and constraints requires the propagation of the covariance for the relative state, target attitude and chaser attitude. In particular, being the dynamics uncoupled, it is possible to obtain a single covariance matrix by stacking three different contributes in a block diagonal matrix. The first contribution is given by the relative position and velocity $\mathbf{P}_x$. Such matrix is propagated with the HCW state transition matrix as in the previous subsection. This grants a fast analytical propagation of the covariance.

Secondly, the chaser attitude is assumed to be controlled in order to guarantee constraint and pointing requirements, hence the covariance matrix is held constant at $\overline{\mathbf{P}}_{\mathbf{CI}}$. Therefore, the only covariance to be propagated is the one related to the target attitude motion $\mathbf{P_{TI}}$. Given an initial estimate of the state and covariance of the target tumbling in the extended state $\mathbf{s_{TI}} = [\boldsymbol{\omega}_{\mathbf{TI}}, \mathbf{q_{TI}}, \mathbf{k}]$, the dynamics governing the evolution of the extended state is described by the system of ordinary differential equations in Equation 16 [37]:

$$\mathbf{s_{TI}} = \begin{cases} \dot{\boldsymbol{\omega}}_{\mathbf{TI}} \\ \dot{\mathbf{q}}_{\mathbf{TI}} \\ \dot{\mathbf{k}} \end{cases} = \begin{cases} \mathbf{J}(\mathbf{k})^{-1}(\boldsymbol{\omega}_{\mathbf{TI}} \wedge \mathbf{J}(\mathbf{k})\boldsymbol{\omega}_{\mathbf{TI}}) \\ \frac{1}{2}\mathbf{B}(\boldsymbol{\omega}_{\mathbf{TI}})\mathbf{q_{TI}} \\ \mathbf{0}_{2\times 1} \end{cases} = \mathbf{l}(\mathbf{s_{TI}}) \tag{16}$$

Notice that the free tumbling motion is assumed for the target. Furthermore, the expression for the matrix $\mathbf{J}(\mathbf{k})$ is given in [40] as Equation 17

$$\mathbf{J}(\mathbf{k}) = \begin{bmatrix} \exp(k_1) & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \exp(-k_2) \end{bmatrix} \tag{17}$$

whereas $\mathbf{B}(\boldsymbol{\omega}_{\mathbf{TI}})$ is formulated as:

$$\mathbf{B}(\boldsymbol{\omega}_{\mathbf{TI}}) = \begin{bmatrix} 0 & -\boldsymbol{\omega}_{\mathbf{TI}}(1) & -\boldsymbol{\omega}_{\mathbf{TI}}(2) & -\boldsymbol{\omega}_{\mathbf{TI}}(3) \\ \boldsymbol{\omega}_{\mathbf{TI}}(1) & 0 & \boldsymbol{\omega}_{\mathbf{TI}}(3) & -\boldsymbol{\omega}_{\mathbf{TI}}(2) \\ \boldsymbol{\omega}_{\mathbf{TI}}(2) & -\boldsymbol{\omega}_{\mathbf{TI}}(3) & 0 & \boldsymbol{\omega}_{\mathbf{TI}}(1) \\ \boldsymbol{\omega}_{\mathbf{TI}}(3) & \boldsymbol{\omega}_{\mathbf{TI}}(2) & -\boldsymbol{\omega}_{\mathbf{TI}}(1) & 0 \end{bmatrix} \tag{18}$$

Since the Jacobian of $\mathbf{l}$ is defined as $\mathbf{F} = \frac{\partial \mathbf{l}(\mathbf{s_{TI}})}{\partial \mathbf{s_{TI}}}$, the covariance can be propagated as:

$$\dot{\mathbf{P}}_{\mathbf{TI}} = \mathbf{F}\mathbf{P_{TI}} + \mathbf{P_{TI}}\mathbf{F}^T + \mathbf{Q_{TI}} \tag{19}$$

The state and covariance for the target attitude can therefore be propagated from the initial time instant only once per each arc until t_{max}^{arc}, and afterwards their values at any t of the inspection arc can be obtained via interpolation.

E. Input Space Exploration using Subset Simulation

Once the score function has been obtained as described in the previous subsections, each sample command can be assigned a score using the sum of the three score components. To explore the command space, an heuristic has been implemented in [32, 33]. However, this work introduces a novel approach for input sampling refinement which relies on Subset Simulation [35]. This technique is an adaptive stochastic simulation method traditionally used to efficiently compute small failure probabilities, but it was also used in different research areas in reliability [42]. The idea at the basis of the method is to compute the probability as a product of larger conditional probabilities which describe a sequence of intermediate regions. The standard method is initialized using a standard Monte Carlo simulation to generate a first set of samples at the first conditional probability level. Once the first failure region is determined and the first conditional probability is computed, a Monte Carlo Markov Chain (MCMC) algorithm [43] is used to generate samples conditional to this first region. The process continues iteratively until a stopping criterion is met, by using the latest level of samples to locate the successive region, from which other samples are generated with MCMC. This paper exploits the idea that SS application can be used to solve optimization problems by considering a maximization event as a rare event. In [44], the author suggests the possibility of converting a global optimization problem into a rare event simulation problem, where the feasible potentially optimal solutions are analogously the rare event samples close to failure in the reliability problem. To view an optimization problem in the context of a reliability problem, consider the input variables s as random variables so that the objective function C given by the sum of the three components Equation 2,10 and 3 is now random and has its own PDF, and cumulative distribution function (CDF). From probability theory, every CDF is monotone, non-decreasing and right continuous. Let C_{opt} be the global maximum of C: by definition the CDF value at this optimum is unity. Consider the reliability problem of finding the "failure probability" defined as:

$$P_F = P(F) = P(C(s) \geq C_{opt}) \tag{20}$$

where the failure event is $F = \{C(s) \leq C_{opt}\}$. Clearly, this is zero because C_{opt} is the global maximum. In the context of an optimization problem, the failure probability is of little concern, instead the focus is placed on the point or region where the objective function attains the largest value(s). The rare failure region in the reliability problem corresponds to the region where the objective function attains its global maximum. Along the same spirit of SS for reliability analysis, $P(C(s) \geq C_{opt})$ can be expressed as a product of a sequence of conditional probabilities. During SS, a series of intermediate threshold values $\{C_i : i = 1, 2...\}$ are generated that correspond to boundaries consistent with the specified conditional probabilities $\{p_i : i = 1, 2...\}$.

$$p_i = P(C(s) \geq C_i) = P(F_i) \tag{21}$$

$$p_i = P(C(s) \geq C_i | C(s) \geq C_{i-1}) = P(F_i | F_{i-1}) \tag{22}$$

where F_i is the intermediate event determined by the objective function value C_i. By the definition of conditional probability, we have:

$$P(F) = \prod_i p_i \tag{23}$$

We can generate samples using the modified Metropolis-Hasting algorithm [35] that progressively towards the maximum, while the rare event region is gradually explored. For an optimization problem with at least one global point, one can expect that $C_i \to C_{opt}$ as $P(F) \to 0$. In this particular formulation, the level probability p_k becomes a parameter that regulates the convergence of the optimization process. If a small value is used, the algorithm would have a low probability of reaching a global optimum. The level probabilities must be high enough to permit the locally developed Markov Chain samples to move out of a local optimum in favor of finding a global optimum, especially in early simulation levels. However, high level probabilities would increase the number of simulation levels. Here, we adopted a decreasing strategy as described in [44] to handle this difficulty. The advantage of using SS instead of the heuristic algorithm is that SS requires far less tuning and still provides exploration of the input space to a good extent. In addition, it has been validated in a number of different fields [45, 46] and provides a plug-and-play approach. A pseudo-code reporting the crucial passages of the sample refinement code exploiting SS in this work is reported in 2 and a numerical comparison of the two sampling techniques is reported in subsection III.D. The introduction of an iterative procedure, such as the proposed SS based input space exploration algorithm, on a system which should work in real time may encounter run time issues. Indeed, should the algorithm struggle to reach convergence, or should it take more iterations than expected, the entire operations would be delayed. However, as it will become clear later in subsection III.D, it is possible to overcome these limitations while using SS by formulating it so that the number of function evaluations is limited. This in turn could cause convergence to suboptimal solutions. Nonetheless, by sacrificing optimality, one can retrieve a command in a timely manner. This procedure requires a careful trade off, which is the subject of the preliminary analysis of the input sampling strategy.

III. Numerical Simulations

The algorithm is tested in two scenarios. For both study cases the selected target altitude is of 700km, whereas the initial condition of the chaser is that of an inclined football orbit (IFO) provided in the LVLH reference frame, with the x axis aligned along the radial of the target location, the z axis aligned with the orbital momentum vector and the y axis completing the right handed reference:

$$\mathbf{r_0} = [100; 0; 0] \text{ m} \qquad \mathbf{v_0} = [0; -0.107; 0.0431] \text{ m/s} \tag{24}$$

In the first scenario, the target has a nadir pointing fixed attitude, whereas in the second one the target is tumbling with a random initial angular velocity of 0.1 deg/s. Moreover, it is assumed that the chaser only has a single low-thrust engine providing a throttable control thrust of maximum 0.05N. To enforce the maximum slew rate of the chaser, we selected a value of 5.7 deg/s, which allowed to obtain many feasible input samples without being too penalizing. Finally, the chaser is be equipped with a parallel baseline stereo-vision camera with a focal lenght of 50 mm, a baseline of 1 m and

Algorithm 2 Sampling Refinement using Subset Simulation

Select the probability distribution $f_i(s_i)$ for each component of the state s_i. In this study these probabilities are modeled as truncated Gaussians with mean μ_i on the mean value of their interval and $3\sigma_i$ equal to the half-interval amplitude as reported in Equation 1.
$t \leftarrow max(\sigma_i)/(ub_i - lb_i)$: stopping criterion is based on the maximum standard deviation of the PDF for each variable divided by the interval amplitude given by upper bound (ub_i) minus lower bound (lb_i), which allows to eliminate the effects of different scaling of input variables.
Generate N samples by direct Monte Carlo according to the initial f_i.
Compute the score function $C(s)$ for each sample of the $k = 0$ level and sort them.
$k \leftarrow 0$, Initialization of probability level 0.
while t >threshold **do**
    Select the C_k threshold as the Np_kth element of the sorted array of scores. Due to this choice, there are Np_k samples whose score is larger than C_k and hence corresponding samples belong to the event F_k.
    These samples provide the "seeds" samples for next simulation level. The modified Metropolis-Hasting algorithm [35] is employed for generating conditional samples. In the $k+1$th simulation level, starting from each of samples in F_k a Markov chain of length $1/p_k$ can be generated with the same conditioning.
    Compute the score function $C(s)$ for each sample of the new level and sort them.
    Update the artificial pdf of each variable according to the mean and standard deviation of the samples at level $k+1$.
    Update t.
    Update $p_k + 1$.
    $k \leftarrow k + 1$
end while

a standard 35 mm sensor size. The total inspection allocated time is 48h during which the chaser periodically calls the GNC module to compute the best trajectory. The strategy is compared with a stationary IFO [24]. For the sake of a fair comparison with the proposed strategy, it is assumed that in the IFO approach, the chaser uses a computation time window to perform station keeping operations of the same amplitude as the one used for the SBMPO algorithm. Additionally, the chaser will have to perform some slewing maneuvers to reach the correct attitude for target pointing. The RSO is modeled with a generic spacecraft 3D model as reported in Figure 3, and the measured landmarks are considered to be the vertices of the triangulation.

A. Nadir Pointing Target

The first test case analyzes the behavior of the chaser in the neighborhood of a nadir pointing target. Figure 4 shows the stationary IFO, whereas the trajectory obtained with the proposed guidance strategy for 48h is reported in Figure 5. From the elliptical trajectories followed by the SBMPO algorithm it is evident that the proposed guidance takes advantage of the characteristics of the natural dynamics to outperform the classic approach. Indeed, every periodical orbit will not drift away from the target and hence will spend more time close to a useful region for inspection. However, the SBMPO algorithm switches between multiple periodic and quasi-periodic orbits, hence allowing it to gain more different points of view, which increase the information gain. Furthermore, in Figure 6 the exploration score is reported. Said score is normalized per each maneuver on its maximum value found over the first sampling of n_0 inputs. Therefore, seeing a score which is equal to or greater than 1 means that the exploration is enhanced by the trajectory selection. In

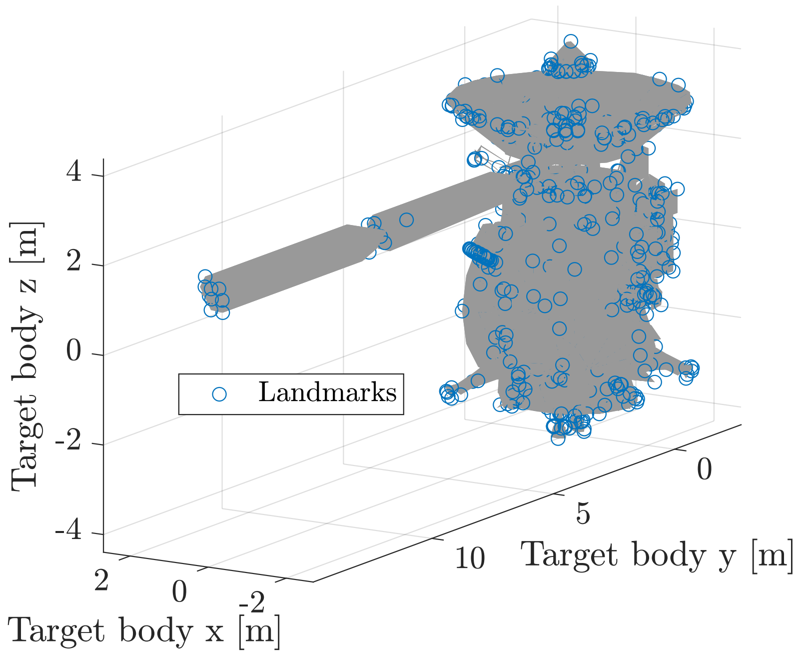

Fig. 3 Model used as a reference target, the vertices of the 3D model represent the landmarks used by the algorithm.

Figure 6, the SBMPO algorithm concludes the inspection with 20 maneuvers, always selecting an exploratory one (i.e. score ≥ 1). On the other hand, the chaser on the IFO cannot decide to follow more promising trajectories, therefore the exploratory score shows greater degradation. To conclude, Figure 7 shows the final reconstructed landmark density function evaluated on a sphere with KOZ radius. In particular, this representation is not scaled by the trace of the covariance as in Equation 10, therefore a region in the plot with higher value is less explored and hence more appealing. By observing this density, it can be noticed how the landmarks are reconstructed according to a better coverage thanks to the varying inspection strategy with respect to the stationary IFO, achieving a more uniform distribution of landmarks on the surface of the target and, as a consequence, a lower and uniform value. On the contrary, the points collected during IFO execution are always concentrated in the same regions, therefore the score function remain high in those that are not explored.

B. Tumbling target

Once the trivial case of a nadir pointing target is analyzed, the analysis is extended to a randomly tumbling target. The exploration score obtained by each maneuver is reported for each strategy in Figure 8. Once again, the SBMPO algorithm proposed consistently recovers exploratory maneuvers, whereas the IFO sees its performance further reduced due to the random tumbling motion of the target. To conclude, the representation of the reconstructed 3D PDF associated

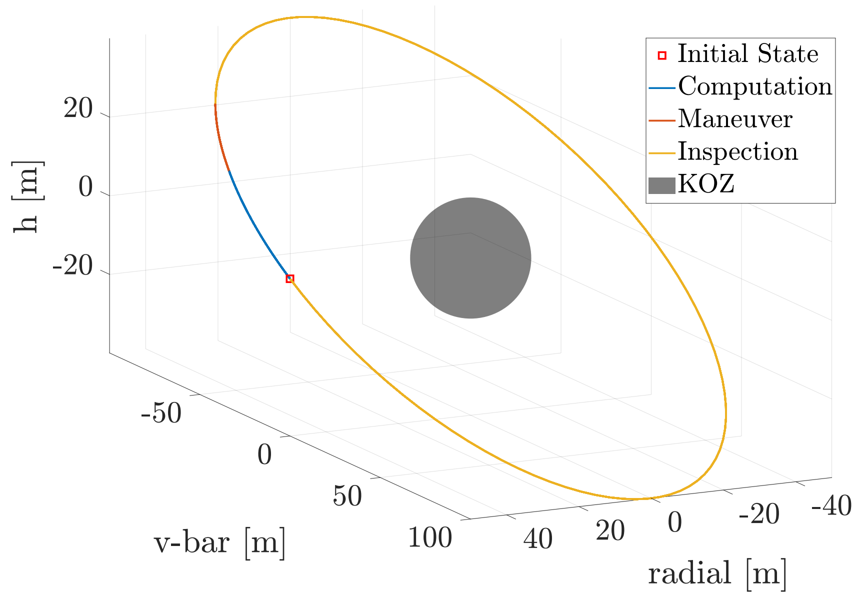

Fig. 4 Whole mission trajectory of the stationary football orbit.

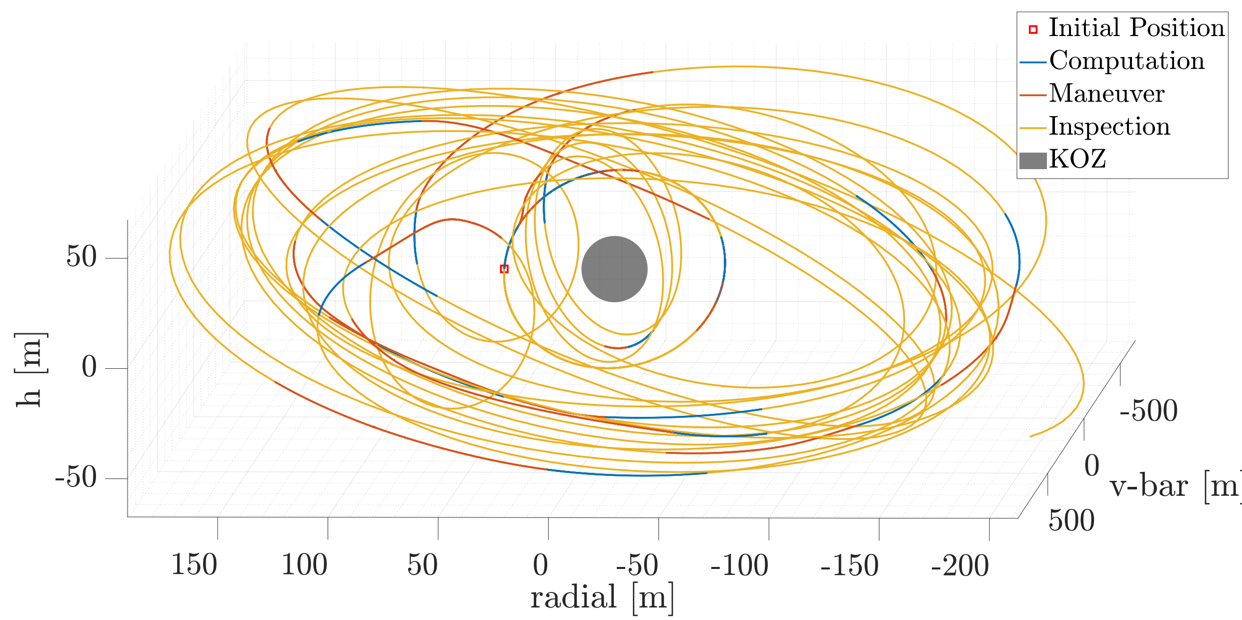

Fig. 5 Whole mission trajectory following the proposed guidance strategy.

with target surface features is reported in Figure 9. As it can be observed, the distribution of surface features in the case of the proposed strategy is much more uniform with respect to the standard football orbit. Moreover, the tumbling motion of the target actually helped spreading the landmarks distribution also in the IFO case, improving the previous results.

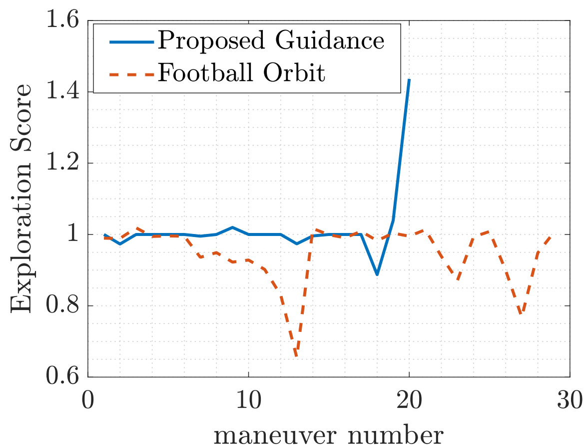

Fig. 6 Exploration score C_{exp} obtained for each maneuver by the two strategies in the first study case.

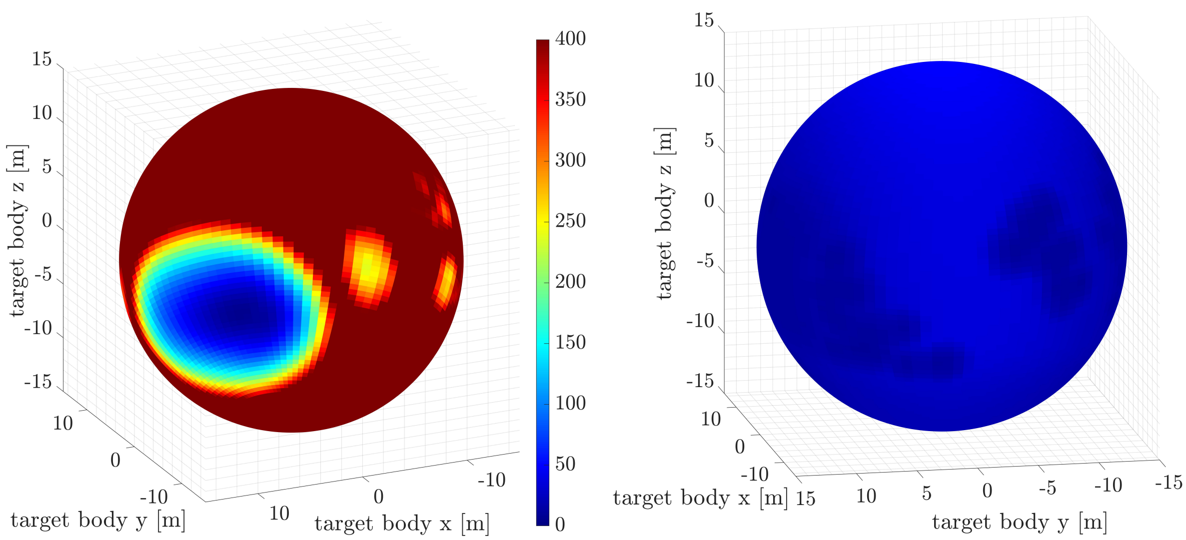

Fig. 7 Reconstructed score density function for the nadir pointing target by the IFO (left) and by the proposed guidance (right).

C. Exploration Efficiency

This section is devoted to analyze the performance of the test cases in terms of capability to exploit an observation time window. First of all some useful quantities for this analysis are defined, which use data from the tumbling satellite test cases, but whose results extend also to the nadir pointing study case. In particular, three parameters are defined to assess the performance of the proposed guidance strategy. First, the *Operations Time*, which is defined as the time dedicated to maneuver computation and attitude acquisition in the case of the proposed guidance strategy. Similarly to the previous subsection, for the sake of a fair comparison with the IFO case, we allocated an equal amount of time as *Operations time*, during which the IFO to performs station keeping operations. This assumption may seem punitive

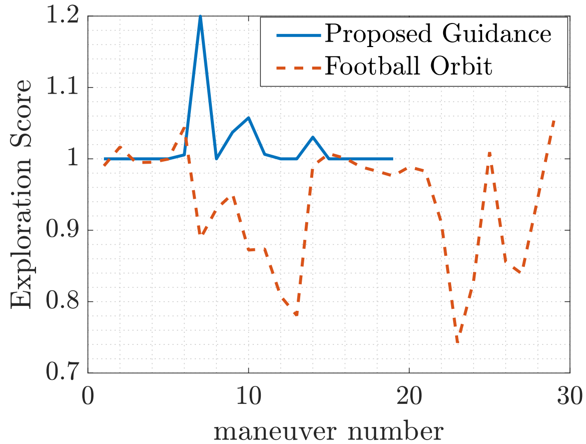

Fig. 8 Exploration score C_{exp} obtained for each maneuver by the two strategies in the second study case.

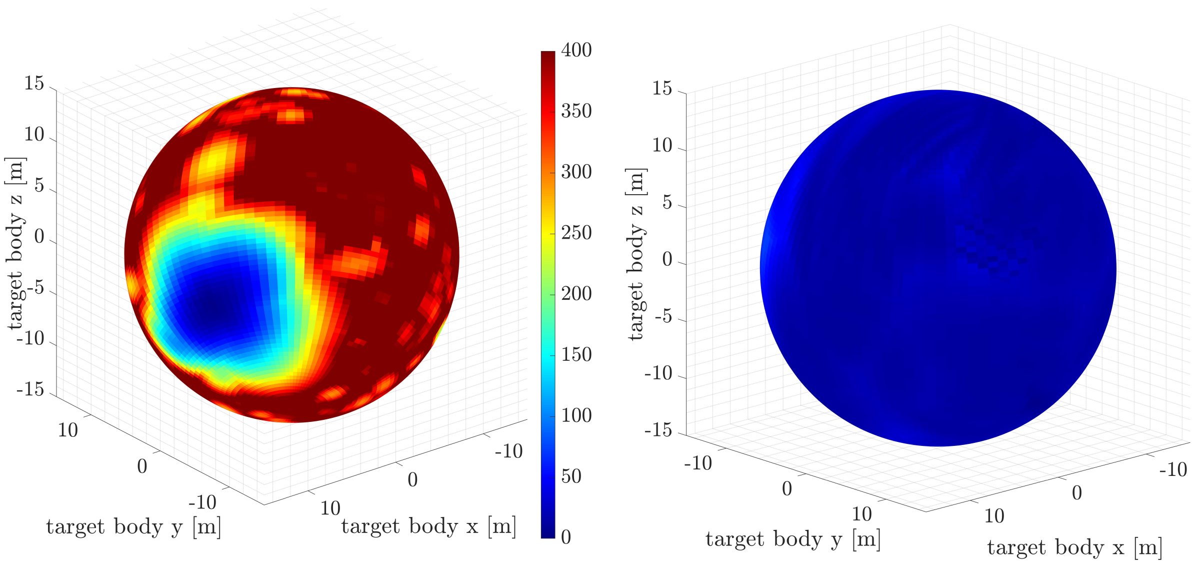

Fig. 9 Reconstructed score density function for the tumbling target by the IFO (left) and by the proposed guidance (right).

for the IFO, but it is indeed justifiable thanks to the fact that a very small deviation from the ideal initial state of this trajectory causes drifts from the nominal path. Hence, this strategy requires more effort to preserve the desired orbital shape and more frequent checks on the relative state. Secondly, a *Proximity Operations Time* is introduced, which is the time allocated to inspection in addition to the time spent during operations that could still be considered of scientific interest (i.e. the execution of relative maneuvers). Additionally, the pure *Inspection Time* is defined as the time spent during inspection for each maneuver arc. To conclude, the *Total Used Time* index is introduced, which sums *Proximity Operations Time* and *Operations Time* hence constituting the entire time of an inspection leg. These time measures accumulate during an inspection mission, thus their cumulative value gives information on how well the allocated time window has been exploited. The results of this analysis are reported in Table 1.

Table 1 Comparison of efficiency of exploration for both IFO ans SBMPO given as % of the time window

	IFO Performance [% of time window]	Proposed Guidance Performance [% of time window]
Total Used Time	97.8	100.0
Proximity Operations Time	85.7	91.6
Inspection Time	85.7	87.9

The results show that the proposed guidance algorithm is much more effective in exploiting the allocated time. In fact, it spends more than 91% of the time window performing proximity operations. Even when comparing just the inspection time, the proposed guidance still retains a margin of more than 2% of the time window. Furthemore, it can be observed that the capability of adapting inspection phases duration for the proposed algorithm allowed to exploit completely the inspection time window, whereas the IFO falls short of approximately 2% of the available time. Notice that this discrepancy could, in principle, still be smoothed by using a time window which is an exact multiple of the orbital period of the target. However, this preliminary analysis shows that in fact the IFO lacks such flexibility, and still requires human in the loop for improvement.

D. Sampling Refinement Tradeoff

In this work, we also proposed an alternative approach to the heuristic input sampling approach implemented in [33]. The proposed method, described in subsection II.E, relies on a SS simulation paradigm to explore the input space and retrieve the best maneuver. In order to assess the impact of the sampling strategy, a comparison between the two implemented strategies is hereby provided. To determine the SS algorithm operational parameters and to have an adequate comparison with the heuristic sampling refinement, a subsequent approach of 4 test cases has been applied, where each test builds on the results of the previous. First of all, the two test cases necessary for the determination of the SS parameters are reported. Subsequently, a third test case is addressed to compare the repeatability of the two strategies. Finally, in the fourth test case, the strategies are compared in terms of obtained score during a simulated trajectory.

1. Test Case 1

As illustrated in subsection II.E, the SS algorithm employs a varying number of iterations to reach convergence, also influencing the number of total samples drawn. Therefore, in this preliminary analysis we estimated how many levels would be needed on average to reach convergence. To do so, 10 different initial conditions for the chaser are selected from a set of states tracked during a simulation of the full guidance algorithm. For each of these initial conditions, the initial sample size has been varied in a given range:

$$n_0 = \{50, 100, 200, 300, 400, 500, 1000, 1500, 2000\} \tag{25}$$

Afterwards, the SS sampling refinement has been run to convergence in each of the combinations. The outcome of this simulation provided an indication of the average number of iterations required to converge for a given n_0 in Equation 25. In particular the results showed that 10 iterations of the SS algorithm would lead to a good trade off between convergence and computational cost.

2. Test Case 2

The second analysis is aimed at determining the best number of samples for the SS algorithm. In particular, one initial state for the chaser is selected, whereas the size of the initial sample is varied as in Equation 25. The number of levels is fixed at 10 as per the previous test case results. To obtain a fair comparison, the heuristic refinement algorithm is run from the same initial condition with the same initial sampling but with as many n_{mesh} steps as necessary to obtain the same number of function evaluations as given by the SS algorithm. This analysis allowed to formulate a comparison with the heuristic algorithm both in terms of score and computational time as illustrated in Figure 10.

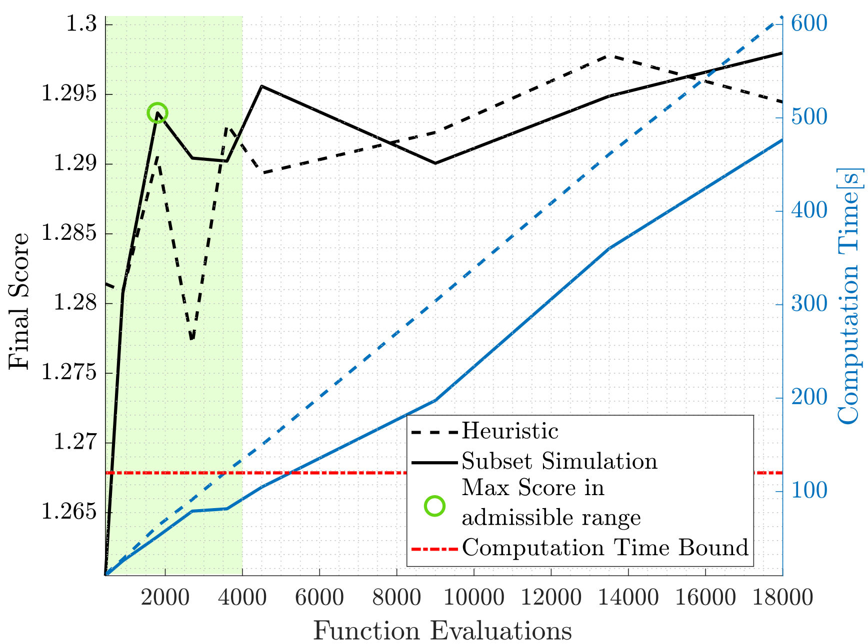

Fig. 10 Determination of the best number of score function evaluation for the SS algorithm.

As evidenced by Figure 10, the computational time of the heuristic algorithm is always higher than the SS, mainly due to the simplicity of implementation of the SS algorithm. Secondly, it can be observed that the score obtained by the two algorithms is always comparable. Due to the selection of a computation time window of approximately $120s$, the only accessible region of this graph is the one highlighted in green. In this region, the best trade-off between number of function evaluations and achieved final score is given when the initial sample size is set to 200.

3. Test Case 3

In this third test, the repeatability of the two algorithms is compared by sampling 10 different initial states of the chaser from the trajectory tracked during a trial run of the guidance algorithm. By keeping the parameters of the SS as set from the results of the two previous test cases, both algorithms are run 100 times for each initial condition, and an average performance is retrieved.

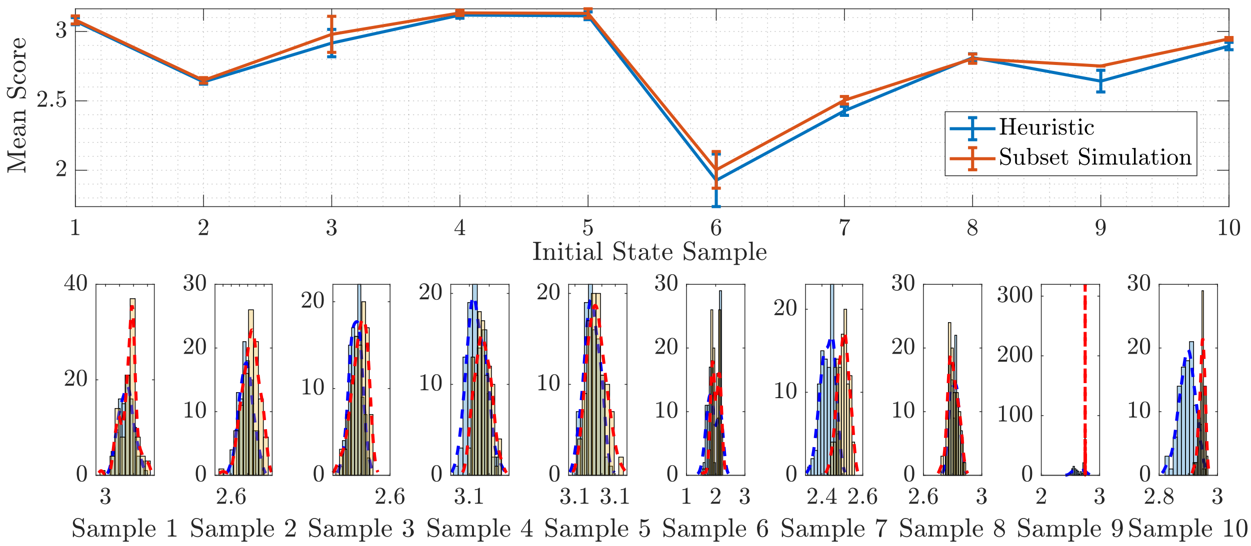

Fig. 11 Maneuver score distributions across the 10 different initial conditions given by SS and heuristic approach.

The mean and standard deviation of the scores retrieved from both algorithms for a given initial condition are summarized in Figure 11. As it can be observed, their values are always comparable and the SS often retrieves better scores as observed from the mean of the score distribution, which is reported on the second row of the figure. It is interesting to notice that for the initial condition #9 the SS focuses on a very narrow region of the maneuver space, which is indeed providing always the best values in all trial runs. In this particular case in fact, the best maneuver is the null maneuver, which is correctly identified each time by the SS. On the contrary the heuristic sampling still draws "useless" samples, spreading the distribution.

4. Test Case 4

To conclude, this last test case analyzes the performance difference obtained by the two different sampling strategies during a run of the entire guidance algorithm lasting 100 maneuvers. For each maneuver computation, the SS algorithm always achieves a higher level of average score as illustrated in Figure 12. Furthermore, in 63% of the initial conditions the best maneuver is obtained by the SS algorithm. In addition, the test showed that the first level sampling of the SS simulation included the best maneuver only 11% of times. This means that further sampling refinement is effective in producing a score increase in 89% of the cases, producing an average score improvement of 0.2281. On the contrary,

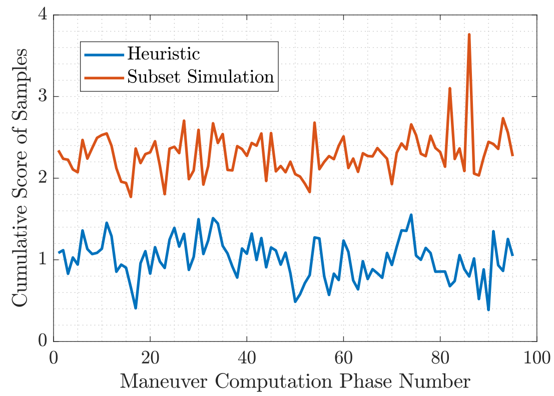

Fig. 12 Average score obtained by the samples of SS and heuristic algorithm for each maneuver computation case.

the heuristic approach found the best sample in the first level 66% of the times; consequently, in 34% of the maneuver computations, all samples drawn in subsequent levels are useless. Indeed, the average score improvement for this approach is only 0.0776.

The outcomes of the test cases 3 and 4 suggest that a further tuning of the sampling heuristic refinement would be required, and that there is in fact still a margin for improvement. This goes to prove that even if the outcome of this approach is acceptable, it requires more in depth knowledge of the problem and tuning with respect to a simple yet robust algorithm such as SS, which can be used as plug-and-play after a trivial preliminary analysis.

E. Conclusion

In this work we proposed an autonomous guidance algorithm which relies on state-of-the-art Sampling Based Motion Planning techniques. This guidance approach allows to optimize information gain while inspecting an unknown and non-cooperative RSO while abiding to many constraints. The proposed approach relies on an heuristic input sampling strategy, for which we proposed an easier, more robust alternative based on Subset Simulation. In this work, we also provided an example application of this guidance technique for the inspection of an unknown and non-cooperative RSO in two different configurations (i.e. nadir-pointing and tumbling). The results obtained by the newly proposed guidance strategy are also compared to a classical natural relative motion trajectory. The results show that the proposed guidance algorithm outperforms the standard inspection baseline in terms of effectiveness of the inspection, allowing for a flexible exploitation of the inspection time window. Overall, the strategy shows greater performance stability across the two case studies analyzed. In fact, the part of the score related to the exploration shows less dips and oscillations with respect to the one obtained by the fixed inspection strategy. It can also be noticed that the effect of the tumbling motion degrades the exploration score of the stationary football orbit, whereas the proposed algorithm is capable of rejecting this disturbance. In addition, a noticeable difference can be observed in the reconstructed landmark density function used to compute the score. Indeed, the surface landmark density is always more homogeneous for the proposed guidance algorithm, even in the case of a tumbling target, where the tumbling motion improves the performance of the standard strategy. This work also proposes a more robust and simple approach for sample refinement based on SS which allowed to rival the state-of-the-art heuristic approach. Indeed, the proposed SS approach outperforms the heuristic sampling thanks to its easiness of use, which allowed to retrieve comparable performance with little to no parameter tuning.

References

[1] 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.

[2] Reed, B. B., Smith, R. C., Naasz, B. J., Pellegrino, J. F., and Bacon, C. E., "The Restore-L Servicing Mission," *AIAA SPACE 2016, 13-16 September, 2016, Long Beach, California*, AIAA, 2016, pp. 1–8. https://doi.org/10.2514/6.2016-5478, Paper No. AIAA 2016-5478.

[3] 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.

[4] Zimpfer, D., Kachmar, P., and Tuohy, S., "Autonomous Rendezvous, Capture and In-Space Assembly: Past, Present and Future," *1st Space Exploration Conference: Continuing the Voyage of Discovery, 30 January - 1 February, 2005, Orlando, Florida*, AIAA, 2005, pp. 1–12. https://doi.org/10.2514/6.2005-2523, Paper No. AIAA 2005-2523.

[5] 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.

[6] 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.

[7] 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.

[8] 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.

[9] Ruel, S., Luu, T., and Berube, A., "Space shuttle testing of the TriDAR 3D rendezvous and docking sensor," *Journal of Field Robotics*, Vol. 29, No. 4, 2012, pp. 535–553. https://doi.org/10.1002/rob.20420.

[10] 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.

[11] Bowen, J., Tsuda, A., Abel, J., and Villa, M., "CubeSat Proximity Operations Demonstration (CPOD) mission update," *2015 IEEE Aerospace Conference, 7-14 March, 2015, Big Sky, Montana*, IEEE, 2015, pp. 2328–2335. https://doi.org/10.1109/AERO.2015.7119124.

[12] Lu, P., and Liu, X., "Autonomous Trajectory Planning for Rendezvous and Proximity Operations by Conic Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 36, No. 2, 2013, pp. 375–389. https://doi.org/10.2514/1.58436.

[13] Kobilarov, M., and Pellegrino, S., "Trajectory Planning for CubeSat Short-Time-Scale Proximity Operations," *Journal of Guidance, Control, and Dynamics*, Vol. 37, No. 2, 2014, pp. 566 – 579. https://doi.org/10.2514/1.60289.

[14] Leomanni, M., Rogers, E., and Gabriel, S. B., "Explicit Model Predictive Control Approach for Low-Thrust Spacecraft Proximity Operations," *Journal of Guidance, Control, and Dynamics*, Vol. 37, No. 6, 2014, pp. 1780 – 1790. https://doi.org/10.2514/1.G000477.

[15] Breger, L., and How, J. P., "Safe Trajectories for Autonomous Rendezvous of Spacecraft," *Journal of Guidance, Control, and Dynamics*, Vol. 31, No. 5, 2008, pp. 1478 – 1489. https://doi.org/10.2514/1.29590.

[16] Boyarko, G., Yakimenko, O., and Romano, M., "Optimal Rendezvous Trajectories of a Controlled Spacecraft and a Tumbling Object," *Journal of Guidance, Control, and Dynamics*, Vol. 34, No. 4, 2011, pp. 1239 – 1252. https://doi.org/10.2514/1.47645.

[17] Ventura, J., Ciarcià, M., Romano, M., and Walter, U., "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.

[18] Dong, H., Hu, Q., Liu, Y., and Akella, M. R., "Adaptive Pose Tracking Control for Spacecraft Proximity Operations Under Motion Constraints," *Journal of Guidance, Control, and Dynamics*, Vol. 42, No. 10, 2019, pp. 2258 – 2271. https://doi.org/10.2514/1.G004231.

[19] Woffinden, D. C., and Geller, D. K., "Relative Angles-Only Navigation and Pose Estimation For Autonomous Orbital Rendezvous," *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 5, 2007, pp. 1455 – 1469. https://doi.org/10.2514/1.28216.

[20] Corpino, S., Mauro, S., Pastorelli, S., Stesina, F., Biondi, G., Franchi, L., and Mohtar, T., "Control of a Noncooperative Approach Maneuver Based on Debris Dynamics Feedback," *Journal of Guidance, Control, and Dynamics*, Vol. 41, No. 2, 2018, pp. 431 – 448. https://doi.org/10.2514/1.G002685.

[21] Razgus, B., Mooij, E., and Choukroun, D., "Relative Navigation in Asteroid Missions Using Dual Quaternion Filtering," *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 9, 2017, pp. 2151 – 2166. https://doi.org/10.2514/1.G002805.

[22] Fourie, D., Tweddle, B. E., Ulrich, S., and Saenz-Otero, A., "Flight Results of Vision-Based Navigation for Autonomous Spacecraft Inspection of Unknown Objects," *Journal of Spacecraft and Rockets*, Vol. 51, No. 6, 2014, pp. 2016 – 2026. https://doi.org/10.2514/1.A32813.

[23] Clohessy, W. H., and Wiltshire, R. S., “Terminal Guidance System for Satellite Redezvous,” *Journal of the Aerospace Sciences*, Vol. 27, 1960, pp. 653 – 658. https://doi.org/10.2514/8.8704.

[24] Woffinden, D. C., “On-orbit satellite inspection : navigation and delta-v analysis,” Master’s thesis, Massachussetts Institute of Technology, 09 2005. URL https://dspace.mit.edu/handle/1721.1/28862.

[25] 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.

[26] Williams, T., “Orbital Inspection Vehicle Trajectories Based on Line-of-Sight Maneuvers,” *Acta Astronautica*, Vol. 50, 2002, pp. 49 – 53. https://doi.org/10.1016/S0094-5765(01)00140-0.

[27] Frey, G. R., Petersen, C. D., Leve, F. A., Kolmanovsky, I. V., and Girard, A. R., “Constrained Spacecraft Relative Motion Planning Exploiting Periodic Natural Motion Trajectories and Invariance,” *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 12, 2017, pp. 3100 – 3115. https://doi.org/10.2514/1.G002914.

[28] Woffinden, D. C., and Geller, D. K., “Navigating the Road to Autonomous Orbital Rendezvous,” *Journal of Spacecraft and Rockets*, Vol. 44, No. 4, 2007, pp. 898 – 909. https://doi.org/10.2514/1.30734.

[29] Latombe, J. C., *Robot Motion Planning*, Springer US, Boston, MA, 1991, The Springer International Series in Engineering and Computer Science, Vol. 124, Chap. 1, pp. 1–57. https://doi.org/10.1007/978-1-4615-4022-9_1.

[30] Starek, J. A., Schmerling, E., Maher, G. D., Barbee, B. W., and Pavone, M., “Real-time, propellant-optimized spacecraft motion planning under Clohessy-Wiltshire-Hill dynamics,” *Proceedings of the 2016 IEEE Aerospace Conference, 5-12 March, 2016, Big Sky, Montana*, IEEE, 2016, pp. 2132–2147. https://doi.org/10.1109/AERO.2016.7500704.

[31] Francis, G., Collins, E., Chuy, O., and Sharma, A., “Sampling-Based Trajectory Generation for Autonomous Spacecraft Rendezvous and Docking,” *AIAA Guidance, Navigation, and Control Conference, August 19-22, 2013, Boston, Massachussetts*, AIAA, 2013, pp. 1–11. https://doi.org/10.2514/6.2013-4549, Paper No. AIAA 2013-4549.

[32] Capolupo, F., and Mast, S., “Sampling Based Receding Horizon Guidance for the Safe Inspection of a Tumbling Spacecraft,” *Advances in the Astronautical Sciences*, Vol. 164, 2018, pp. 1149–1160.

[33] Capolupo, F., and Labourdette, P., “Receding-Horizon Trajectory Planning Algorithm for Passively Safe On-Orbit Inspection Missions,” *Journal of Guidance, Control and Dynamics*, Vol. 42, No. 5, 2019, pp. 1023–1032. https://doi.org/10.2514/1.G003736.

[34] Dunlap, D. D., Collins, E. G., and Caldwell, C. V., “Sampling based model predictive control with application to autonomous vehicle guidance,” *Proceedings of the 2008 Florida Conference on Recent Advances in Robotics, FCRAR 2008, 8-9 May, 2008, Melbourne, Florida*, Florida Institute of Technology, 2008, pp. 1–6.

[35] Au, S. K., and Beck, J. L., “Estimation of small failure probabilities in high dimensions by subset simulation,” *Probabilistic Engineering Mechanics*, Vol. 16, No. 4, 2001, pp. 263–277. https://doi.org/10.1016/S0266-8920(01)00019-4.

[36] Surovik, D. A., and Scheeres, D. J., "Autonomous Maneuver Planning at Small Bodies via Mission Objective Reachability Analysis," *AIAA/AAS Astrodynamics Specialist Conference, 4-7 August, 2014, San Diego, California*, AIAA, 2014, pp. 1–12. https://doi.org/10.2514/6.2014-4147, Paper No. AIAA 2014-4147.

[37] Schaub, H., and Junkins, J. L., *Analytical Mechanics of Space Systems*, 2^{nd} ed., AIAA Education Series, Reston, VA, 2009, Chap. 14, pp. 676–681. https://doi.org/10.2514/4.867231.

[38] Maestrini, M., Di Lizia, P., and Topputo, F., "Analytical Impulsive-to-Continuous Thrust Conversion in Linearized Relative Dynamics," *Journal of Guidance, Control, and Dynamics*, Vol. 44, No. 4, 2021, pp. 862–871. https://doi.org/10.2514/1.G005520.

[39] Xu, B., Wang, S., and Zhao, L., "Solar Panel Recognition of Non-cooperative Spacecraft Based on Deep Learnin," *2019 3rd International Conference on Robotics and Automation Sciences (ICRAS)*, 2019, pp. 206–210. https://doi.org/10.1109/ICRAS.2019.8808993.

[40] 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.

[41] Gates, C., "A Simplified Model of Midcourse Maneuver Execution Errors," Tech. Rep. 32-504, Jet Propulsion Laboratory, 1963. URL https://books.google.it/books?id=YIdTQwAACAAJ.

[42] Zio, E., and Pedroni, N., "Estimation of the functional failure probability of a thermal–hydraulic passive system by Subset Simulation," *Nuclear Engineering and Design*, Vol. 239, No. 3, 2009, pp. 580–599. https://doi.org/10.1016/j.nucengdes.2008.11.005.

[43] Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E., "Equation of State Calculations by Fast Computing Machines," *The Journal of Chemical Physics*, Vol. 21, No. 6, 1953, pp. 1087–1092. https://doi.org/10.1063/1.1699114.

[44] Li, H. S., "Subset simulation for unconstrained global optimization," *Applied Mathematical Modelling*, Vol. 35, No. 10, 2011, pp. 5108–5120. https://doi.org/10.1016/j.apm.2011.04.023.

[45] Morselli, A., Armellin, R., Di Lizia, P., and Bernelli Zazzera, F., "A high order method for orbital conjunctions analysis: Monte Carlo collision probability computation," *Advances in Space Research*, Vol. 55, No. 1, 2015, pp. 311–333. https://doi.org/10.1016/j.asr.2014.09.003.

[46] Li, H. S., and Ma, Y. Z., "Discrete Optimum Design for Truss Structures by Subset Simulation Algorithm," *Journal of Aerospace Engineering*, Vol. 28, No. 4, 2015, pp. 0401409–1–040140911–15. https://doi.org/10.1061/(ASCE)AS.1943-5525.0000411.