AAS 26-034

AUTONOMOUS VISUAL ACTIVE SLAM FOR CLOSE PROXIMITY OPERATIONS TO SMALL BODIES COMBINING GOAL-ORIENTED GUIDANCE AND GRAPH-BASED NAVIGATION

Pietro Califano[*], Alban Beshaj[*], Felice Piccolo[*], Antonio Rizza[†], Francesco Topputo[‡], and Paolo Panicucci[§]

To date, ground-based mission analysis and achievable scientific return are strongly constrained by navigation uncertainties and communication delays. Autonomous navigation, target characterization, and high-level decision making are therefore essential to enable safe and cost-effective proximity operations in next-generation small-body missions. While Simultaneous Localization and Mapping (SLAM) can reduce reliance on prior target characterization, state estimation alone is insufficient when trajectory generation remains decoupled from perception and mapping objectives. This work proposes a closed-loop Guidance, Navigation, and Control architecture inspired by active SLAM concepts to address these limitations. The navigation subsystem combines vision-based factor-graph SLAM, including orbital dynamics constraints, with high-rate lidar measurements within an asynchronous filter-graph formulation to jointly estimate the spacecraft state and the target map. Guidance and control are achieved through a goal-oriented Model Predictive Control framework that integrates abstract reachability analysis with a dispersion-control strategy. The approach is validated through multiple analysis studies in a simulated close-proximity scenario around asteroid Itokawa, including a detailed assessment of the individual modules. Results demonstrate accurate localization and mapping performance while maintaining safe trajectories that enable observation of predefined target regions with near-global surface coverage. Identified limitations are critically discussed to highlight remaining open challenges.

INTRODUCTION

Over the past two decades, the space sector has undergone remarkable growth in deep-space exploration. A key element of this road map pivots around small Solar System Bodies exploration, such as asteroids and comets. Their growing interests for scientific inspection, exploitation of resources and planetary defense reasons is pushing the need for the development of safe and reliable engineering solutions to better operate in their proximity under navigation, control and dynamic uncertainties. Limitations due to the low signal to noise ratio affecting ground-based observations,[1] is in fact compensated via in-situ exploration with the use of specialized instrumented probes. Several missions successfully performed proximity operations to these bodies such as the Near Earth Asteroid Rendezvous (NEAR) Shoemaker,[2] Dawn,[3] the Origins, Spectral Interpretation, Resource Identification, Security, Regolith Explorer (OSIRIS-REx),[4] Hayabusa,[5] Hayabusa 2,[6] Rosetta,[7] and the Double Asteroid Redirection Test (DART).[8,9] This landscape of exploration milestones share common patterns such as 1) an extended far-range characterization phase prior to safe proximity operations; 2) heavy operational workload; 3) weak coupling between ground-based trajectory design and scientific objectives, driven by the need to prioritize operational constraints; 4) predominantly time-driven onboard automation capabilities and limited mission-level online reasoning.

[*]PhD Student, Department of Aerospace Science and Technology, Politecnico di Milano, Via La Masa 34, 20156, Milan
[†]Postdoctoral Scholar, Aeronautics and Astronautics, Stanford University, 496 Lomita Mall, Stanford, CA 94305
[‡]Full Professor, Department of Aerospace Science and Technology, Politecnico di Milano, Via La Masa 34, 20156, Milan
[§]Assistant Professor, Department of Aerospace Science and Technology, Politecnico di Milano, Via La Masa 34, 20156, Milan

Autonomous navigation, target characterization, and high-level decision making are essential for reducing operational burden and enabling safe, cost-effective proximity operations in next-generation small-body missions. Simultaneous Localization and Mapping (SLAM) addresses these needs by jointly estimating the spacecraft state and the surrounding environment, thereby reducing reliance on extensive prior target characterization.[10] However, improved autonomous state estimation alone is insufficient if trajectory generation remains decoupled from perception and mapping objectives. Current approaches to autonomous trajectory control typically rely on tracking reference trajectories designed on the ground, offering limited or no replanning capability and struggling to accommodate unforeseen events. Such designs are often optimized primarily for fuel consumption or time of flight, neglecting explicit exploration–exploitation trade-offs in the trajectory envelope. This paradigm becomes inadequate when more complex observation objectives are considered, such as detailed reconstruction of the target shape and gravity field. Moreover, effective replanning and target mapping require accurate knowledge of the system dynamics and precise estimation of the spacecraft state, both of which are strongly affected by navigation, control, and dynamical uncertainties. Finally, the often significant light-time delay further limits the achievable responsiveness and scientific objectives during proximity operations within the ground-in-the-loop paradigm. An innovative concept developed in recent years is proposing a paradigm shift towards autonomous goal-oriented approaches.[11–14] The idea is that the probe is provided with the high-level objectives of the mission and these are tracked via a closed-loop guidance Model Predictive Control (MPC) framework.[15] This strategy allows to dynamically update a reference trajectory on board based on mission status, spacecraft dispersion and navigation accuracy. A brief survey of previous works is presented next to outline the current state of the art.

Related works

In recent years, the constraints of space environments have motivated specialized SLAM systems for asteroid proximity operations to support autonomous navigation around unknown targets. While SLAM is mature in terrestrial robotics, spaceborne applications face high dynamical uncertainty, limited onboard resources, and enhanced mapping requirements, precluding direct reuse of terrestrial frameworks. Consequently, early asteroid SLAM research has primarily relied on filtering-based approaches for their computational efficiency and flight heritage.

Reference 16 pioneered the application of Rao-Blackwellised Particle Filters in asteroid proximity, combining monocular vision with range data to construct octree-based occupancy grids for autonomous navigation and landing. To improve the computational efficiency, Kalman filtering is frequently adopted by expanding the estimated state space beyond simple localization, enabling the simultaneous recovery of the target's physical properties and shape (e.g., see Reference 17, 18). Similarly, a distributed architecture is proposed in reference 19 to leverage multi-spacecraft measurements for autonomous characterization of gravity, rotation, and shape without prior knowledge of the target. However, filtering-based formulations inherently approximate or discard long-term correlations between spacecraft poses and the map. In fact, it has been shown that filtering-based approaches relying on recursive marginalization suffer from fundamental limitations, including estimation inconsistency and limited exploitation of loop closures.[20–22] In contrast, full SLAM formulations, which estimate the complete history of navigation states and map parameters, provide a principled framework to overcome these issues. Factor-graph-based approaches, widely adopted in robotics, enable scalable and real-time estimation by exploiting the problem structure and are therefore attractive candidates for autonomous navigation in small-body proximity operations.[23–26] Among the first to introduce factor graphs in space applications, Reference 27 formulated a probabilistic model that jointly estimates the relative dynamics, center of mass, and principal inertia ratios of an unknown spinning space debris using stereo vision. Building on this, Reference 28 applied a similar factor graph approach specifically to small body scenarios, incorporating star tracker data to constrain rotational estimates, although they did not address loop closure. More recent work have included the asteroid dynamics into the factor SLAM formulation[29] introducing a relative dynamics factor into the graph, encoding complex perturbations such as solar radiation pressure to jointly estimate the target's gravitational parameter and spin state alongside the trajectory. Similarly, Reference 30 compared graph-based variants with and without dynamical factors, finding that while including dynamics significantly improves velocity estimation, it comes with an increasing computational cost. These results motivate the integration of dynamics-aware SLAM with higher-level decision-making

and trajectory generation capabilities. Finally, Reference 31 leverages impulsive maneuvers within a similar dynamical factor graph framework to explicitly resolve monocular scale ambiguity, enabling the joint estimation of the spacecraft state, map, and gravitational parameter.

Despite the historical reliance on ground-in-the-loop operations for small-body exploration, the need to autonomously adapt trajectories in response to estimation uncertainty and environmental variability naturally leads to the consideration of onboard guidance and control strategies capable of closed-loop replanning.[14] Current literature addresses this challenge through various guidance and control architectures, generally categorized into feedback control, indirect optimization, direct optimization, and heuristic methods. Feedback control strategies, such as artificial potential functions or differential guidance, offer computational simplicity by transforming the obstacle avoidance problem into a stability analysis or a local correction step. However, these methods often struggle with local minima and cannot easily accommodate complex integral cost functions.[32] Indirect methods, which solve the problem by satisfying the first-order necessary conditions derived from Pontryagin's Minimum Principle, provide theoretical optimality but are hampered by the sensitivity of the associated Two-Point Boundary Value Problem (TPBVP) to initial guesses.[33,34] Direct methods overcome these limitations by transcribing the Optimal Control Problem into a Non-Linear Programming problem. A particularly promising subclass is Sequential Convex Programming (SCP), which relaxes non-convex constraints into a sequence of convex sub-problems.[35,36] This approach has been successfully applied to fuel-optimal transfers, landing, and proximity operations due to its polynomial-time complexity and convergence guarantees.[37,38] Nevertheless, SCP remains a local optimization technique and requires a high-quality initial guess to attain global optimality. In contrast, heuristics methods, such as tree or sampling-based approaches, address the initialization problem by directly exploring either the solution or the configuration space.[39]
In the specific context of asteroid exploration, Reference 12 developed an abstract reachability analysis that maps the reachable control set into an abstract "observation space" to maximize scientific returns, such as surface coverage. While highly flexible, these sampling-based strategies can be computationally expensive and lack deterministic convergence guarantees. Recent frameworks have therefore proposed hybrid architectures, using computationally efficient reachability analysis to rapidly generate a coarse, goal-maximizing reference trajectory, which is subsequently refined by SCP to robustly handle navigation and control uncertainties within a Model Predictive Control (MPC) scheme.[40,41]

The integration between path planning and SLAM is also called Active SLAM in the robotics literature.[42] Active SLAM represents a significant evolution from standard passive frameworks by treating the simultaneous localization and mapping problem as a comprehensive GNC task rather than a pure estimation problem.[43] While passive SLAM processes data from a predefined trajectory, active SLAM couples perception and action, enabling the vehicle to autonomously plan trajectories under probabilistic inference of the external world (e.g., see Reference 44). In space engineering, this problem is exacerbated by dynamical uncertainties and strong requirements in navigation robustness. Moreover, the recent advancements in autonomous spacecraft proximity operations have driven significant research into Active SLAM frameworks that integrate GNC to optimize information acquisition. Addressing the challenges of small body exploration, Reference 45 formulated the mapping problem as a Partially Observable Markov Decision Process (POMDP), developing a method for autonomous orbit selection that explicitly maximizes map quality. Their approach utilizes geometric constraints derived from stereophotoclinometry requirements (i.e., emission and incidence angles) to quantify the utility of candidate orbits, demonstrating that optimized orbital sequences yield superior surface coverage compared to fixed strategies. Building upon this POMDP foundation, Reference 46 applied Reinforcement Learning to handle the high-dimensional belief spaces inherent in autonomous imaging. Their work expanded the action space beyond discrete orbit selection to include continuous thruster maneuvers and introduced realistic operational constraints, such as onboard memory limits and downlink windows, proving that learned policies could outperform human-designed baselines in maximizing map quality.

Alongside learning-based approaches, graph-based frameworks have emerged as an effective tool for Active SLAM in proximity operations. Reference 47 proposed a tightly-coupled 3D Graph SLAM system utilizing flash LIDAR and an evidential grid map to represent environmental uncertainty. Their framework actively switches between exploration (i.e., visiting new areas) and exploitation (i.e., refining localization via loop closures) by evaluating the information gain of potential trajectory samples, and includes a novel module for autonomously identifying safe landing sites. Extending the use of factor graphs, Reference 48 focused on

optimizing the perception action itself by actively controlling camera pointing directions. By computing the information gain directly from the posterior covariance of the factor graph, their method allows the spacecraft to predict the utility of future observations, reportedly reducing localization uncertainty by 50–70% compared to passive sensing strategies.

Contributions

Building on prior efforts, this paper presents a closed-loop Guidance, Navigation, and Control (GNC) architecture inspired by Active SLAM techniques for close-proximity operations around small bodies under navigation, control, and dynamical uncertainties. The methodology is evaluated in a simulated proximity-operations scenario inspired by asteroid Itokawa, assuming no prior knowledge of the asteroid shape or control profile and accounting for navigation, control, and dynamical uncertainties. Nevertheless, prior knowledge is assumed for the mean radius, rotational state, gravitational parameter, and spherical harmonic coefficients up to the fourth degree. In addition, spacecraft attitude pointing is assumed to be ideal and generated from the ground-truth simulated trajectory. Future work will focus on relaxing these assumptions by including the corresponding parameters in the estimation problem and closing the loop with the Attitude Determination and Control System (ADCS). The key contributions can be thus summarized in the following three points:

- An autonomous closed-loop GNC architecture is proposed, combining a SLAM-based navigation solution with reachability-based reference trajectory generation. Navigation relies on an asynchronous filter-graph formulation for joint spacecraft state and asteroid map estimation, while guidance exploits abstract reachability analysis within a goal-oriented MPC framework to target predefined asteroid-fixed observation regions and to enable autonomous trajectory generation and replanning in uncertain environments.
- A dispersion-mitigation logic based on navigation state estimates is introduced to trigger trajectory replanning whenever the estimated state deviates excessively from the reference, demonstrating the capability to maintain safe proximity orbits while adapting online to uncertainties.
- Closed-loop performance is assessed through a high-fidelity single-run simulation using synthetic imagery, followed by multiple reduced-fidelity Monte Carlo campaigns to characterize the algorithms in a statistically meaningful manner. Navigation accuracy, guidance performance, and critical system behaviors are analyzed and discussed.

The paper is organized as follows. Section Background provides the necessary background concepts. Section Algorithms and integrated GNC system outlines the proposed architecture and briefly touches on the details of each component. Finally, section Analysis and results reports in-depth analysis and results from simulations of the close-proximity scenario.

BACKGROUND

Conventions and reference frames

The following reference frames are required in the problem definition.

- The inertial frame $\mathcal{I}$, defined as the J2000 reference frame centred in the asteroid centre of mass.
- The frame $\mathcal{B}$, a non-inertial reference frame with the z-axis aligned with the rotational axis of the asteroid, x-axis aligned with its prime meridian, and the y-axis completing the right-hand triad. The origin is assumed coincident with the inertial frame $\mathcal{I}$.
- The camera frame $\mathcal{C}$ is defined as centred in the camera origin, with +Z axis oriented as the camera boresight whereas the +X axis and the +Y axis are aligned with the image horizontal and vertical directions. This is also assumed to coincide with a spacecraft fixed frame, which in practice corresponds to perfect knowledge of the camera configuration.

For what concerns the notation, rotation matrices from $\mathcal{F}_1$ to $\mathcal{F}_2$ are indicated as $[\mathcal{F}_2\mathcal{F}_1]$, while vectors are represented with lower case bold text, e.g., $\mathbf{r}, \mathbf{v}$. Where necessary, the reference frame vectors are expressed is indicated on the top right corner, e.g. $\mathbf{r}^{\mathcal{F}_1}$. Estimated quantities are referenced to by using the hat $\hat{}$ symbol.

Navigation

Factor graphs are bipartite graphical models that represent the factorization of a global function into local factors over subsets of variables. This structure explicitly encodes conditional independence relations, which is well suited for estimation problems. In probabilistic inference, each factor typically models the likelihood of a measurement given the relevant state variables. When Gaussian noise is assumed, factors correspond to squared Mahalanobis norms of residuals. As a result, MAP estimation over a factor graph is equivalent to solving a nonlinear least-squares problem.[22,23] Classical optimization methods such as Gauss–Newton can be used to compute the solution. Variable elimination in the graph provides insight into efficient ordering and sparsity exploitation, which can be exploited to enable scalable inference even for large problems. Incremental algorithms, such as iSAM2 (Incremental Smoothing And Mapping), update the solution as new measurements arrive or constraints can be added, e.g. from motion models. By exploiting the Bayes Tree data structure, relinearization and updates are restricted to affected subsets of variables, significantly reducing computational cost.[25] The local nature of navigation measurements makes iSAM2 particularly effective, as it preserves sparsity in the information matrix while supporting incremental updates required for online estimation. Moreover, factor graphs naturally accommodate heterogeneous state variables, such as poses, velocities, and landmarks, as well as multi-sensor measurements, without the need for explicit latency-management mechanisms typically required in recursive filtering.

Visual SLAM algorithms typically estimates pose and velocity states. In terms of factor graphs, these are linked through reprojection factors in Eq. (1), dynamics in Eq. (2) and fully constrained by prior state knowledge as Eq. (3):

$$f_{reproj}(\mathbf{p}_k, \mathbf{l}_j) \;\propto\; \exp\Big(-\tfrac{1}{2}\left\|\hat{\mathbf{u}}_{k,j} - \boldsymbol{\pi}(\mathbf{p}_k, \mathbf{l}_j)\right\|^2_{\Sigma_u^{-1}}\Big)\,, \tag{1}$$

$$f_{dyn}([\mathbf{r},\mathbf{v}]_k, [\mathbf{r},\mathbf{v}]_{k-1}) \;\propto\; \exp\left(-\frac{1}{2}\left\|\begin{bmatrix}[\mathcal{IB}]_k\,\mathbf{r}_k^{\mathcal{B}}\\ [\mathcal{IB}]_k\,\mathbf{v}_k^{\mathcal{B}}\end{bmatrix} - \boldsymbol{\phi}\big([\mathcal{IB}]_{k-1}\mathbf{r}_{k-1}^{\mathcal{B}}, [\mathcal{IB}]_{k-1}\mathbf{v}_{k-1}^{\mathcal{B}}\big)\right\|^2_{Q_{dyn}^{-1}}\right) \tag{2}$$

$$f_{prior}(\mathbf{p}_k) \;\propto\; \exp\left(-\tfrac{1}{2}\left\|\log\big(\hat{\mathbf{p}}_k^{-1}\mathbf{p}_k\big)^{\vee}\right\|^2_{\Sigma_p^{-1}}\right), \tag{3}$$

where $\boldsymbol{\pi}(\mathbf{p}_k, \mathbf{l}_j)$ is the projective transformation of the 3D landmark $\mathbf{l}_j$ onto the image plane of pose $\mathbf{p}_k$, assuming a known camera calibration matrix, and $\mathbf{u}_{k,j}$ is the resulting keypoint in image coordinates. $\Sigma_{(\cdot)}$ is the covariance weighting the residual error. Function $\phi(\mathbf{r}, \mathbf{v})$ is the flow function computed through dynamics integration, between two time instants $k-1$ and k. The factor graph framework also allows the inclusion of states that are not constrained by dynamics, such as pose-only variable nodes, in a unified and seamless manner.

Guidance

The guidance objective considered in this work is to determine trajectories that simultaneously satisfy scientific requirements and typical engineering constraints characterizing CPOs around small bodies. To this end, the guidance problem is formulated as a goal-oriented trajectory design problem, a well-established modeling approach introduced in Reference 49. Within this framework, scientifically relevant parameters are used to define Regions Of Interest (ROIs) around the small body, within which the spacecraft must fly to fulfill mission requirements and maximize scientific return.

In this work, observations are performed using a passive sensor. Consequently, ROIs are defined in terms of the emission angle ϵ, the incidence angle i, and the range ρ, which determine the quality of this type of observation and are relevant for mapping campaigns.[45] Given a specific feature on the small body surface $\mathbf{r}_f$, the ROI can be defined as

$$\mathcal{R}_f = \left\{ \mathbf{r} \in \mathbb{R}^3 \;\middle|\; \rho_f \in [\rho_f^-,\, \rho_f^+],\; \epsilon_f \in [\epsilon_f^-,\, \epsilon_f^+],\; i_f \in [i_f^-,\, i_f^+] \right\} \tag{4}$$

where $\mathbf{r}$ is the generic observer position, $\mathbf{n}_f$ is the outward surface normal of the feature; $\rho_{\min}$, $i_{\min}$, $\epsilon_{\min}$ and $\rho_{\max}$, $i_{\max}$, $\epsilon_{\max}$ represent respectively the upper and lower bounds of the parameters to consider the observation of the feature as a valid one. The parameters are expressed as follows

$$\rho_f(\mathbf{r}) = \|\mathbf{r} - \mathbf{r}_f\|, \tag{5a}$$

$$\epsilon_f(\mathbf{r}) = \arccos\left(\frac{(\mathbf{n}_f)^\mathsf{T}(\mathbf{r} - \mathbf{r}_f)}{\|\mathbf{r} - \mathbf{r}_f\|}\right), \tag{5b}$$

$$i_f(t, \mathbf{r}) = \arccos\left(\frac{(\mathbf{n}_f)^\mathsf{T}(\mathbf{r}_s(t) - \mathbf{r}_f)}{\|\mathbf{r}_s(t) - \mathbf{r}_f\|}\right). \tag{5c}$$

where $\mathbf{r}_s$ is the position of the Sun, the only variable with time dependence, as all vector quantities in the previous equations are expressed in the frame $\mathcal{B}$.

Dynamical environment

The orbital dynamics around the asteroid under impulsive guidance is expressed in the frame $\mathcal{I}$ as follows:

$$\dot{\mathbf{x}} = \mathbf{f}_f(\mathbf{x}, t) + \mathrm{B}\sum_{i=1}^{N_m} \delta\big(t - t_i\big)\, \boldsymbol{\Delta}\mathbf{v}_\mathbf{i}. \tag{6}$$

where $\mathbf{x}$ is the spacecraft state, composed of the position $\mathbf{r}$ and velocity $\mathbf{v}$. $\mathbf{f}_f(\mathbf{x}, t)$ is the free dynamics contribution, $\boldsymbol{\Delta}\mathbf{v}_\mathbf{i}$ is the i-th control impulse executed at time t_i. $\delta(\cdot)$ is the Dirac function and B represents the control matrix, which is constant under the chosen parametrization. The free dynamics $\mathbf{f}_f(\mathbf{x}, t)$ is given by

$$\mathbf{f}_f(\mathbf{x}, t) = \mathbf{f}_g(\mathbf{r}, t) + \underbrace{C_r \frac{A}{m} \frac{P_0 d_0^2}{c} \frac{\mathbf{r} - \mathbf{r}_s}{\|\mathbf{r} - \mathbf{r}_s\|^3}}_{\text{SRP}}. \tag{7}$$

where the Solar Radiation Pressure (SRP) contribution is modeled through the cannonball model and defined by the reflectivity coefficient C_r, the spacecraft effective area A, the spacecraft mass m, the solar flux at 1 Astronomical Unit (AU) P_0, the Sun-Earth distance d_0, and the position of the Sun $\mathbf{r_s}$. $\mathbf{f}_g(\mathbf{r}, t)$ represents the gravitational force, expressed as

$$\mathbf{f}_g(\mathbf{r}, t) = [\mathcal{I}\mathcal{B}]\, \nabla U \tag{8}$$

where $[\mathcal{I}\mathcal{B}]$ is the rotation matrix from the frame $\mathcal{B}$ to the frame $\mathcal{I}$, U represents the gravitational potential, approximated with a spherical expansion truncated to order N[32]

$$U(r, \phi, \lambda) = \frac{\mu}{r} + \frac{\mu}{r}\Big[\sum_{n=2}^{N}\sum_{m=0}^{n}\left(\frac{R}{r}\right)^n \bar{P}_{n,m}(\sin\phi)\big(\bar{C}_{n,m}\cos(m\lambda) + \bar{S}_{n,m}\sin(m\lambda)\big)\Big]. \tag{9}$$

where (r, ϕ, λ) are the spherical coordinates of the spacecraft position in the frame $\mathcal{B}$, μ is the gravitational parameter of the asteroid, $\bar{P}_{n,m}[x]$ is the associated normalized Legendre function of order n and degree m, and $\bar{C}_{n,m}$ and $\bar{S}_{n,m}$ are the normalized potential coefficients.

Control errors are simulated stochastically using a classical magnitude–direction error model in a reference frame built such that +Z corresponds to the thrusting direction, in which both error components are independently sampled from Gaussian distributions.[14] Therefore, in the thruster frame:

$$\Delta\tilde{\mathbf{V}} = (1 + \delta V_{err})||\Delta\bar{\mathbf{V}}|| \begin{bmatrix} \cos(\tilde{\alpha}) \\ \sin(\tilde{\alpha})\cos(\theta) \\ \sin(\tilde{\alpha})\sin(\theta) \end{bmatrix} \tag{10}$$

where δV_{err} is the magnitude error, and $\Delta\bar{\mathbf{V}}$ is the nominal control action as computed by the guidance algorithms, rotated in the thruster frame. The term $\tilde{\alpha}$ corresponds to the half-cone error angle from +Z axis in the thruster frame. The so-computed control action is then rotated back to the inertial frame.

ALGORITHMS AND INTEGRATED GNC SYSTEM

This section outlines the navigation and guidance system architecture and information flow. High-level details of the algorithms are also given. The reader is referred to references for in-depth descriptions of the different components.[30,39,50,51]

GNC system architecture

At the current stage of development, the proposed GNC system implements a loosely coupled architecture in which navigation and guidance are clearly separated but operate in closed loop. Figure 1 illustrates the high-level architecture, highlighting the key relations between the different system components and algorithms:

- **External inputs/ouputs**: from sensors and attitude determination to Navigation; from Guidance to propulsion for manoeuvres.
- **Navigation internal data flow**: from image processing to backend initializer and optimizers; from graph-based smoother to filter; and feedback from filter to backend initializer.
- **Navigation to dispersion evaluation module to Guidance**: provides best state estimate and the interpolated reference state at the evaluation timestamp for replanning trigger check;
- **Navigation to Guidance**: provides state estimates for trajectory generation.
- **Guidance to Navigation**: provides nominal value of control actions and the new reference trajectory.

Note that some interfaces and components are omitted to improve readability. Each relation is described in more detail in the dedicated sectionsNavigation system algorithms and Guidance algorithms.

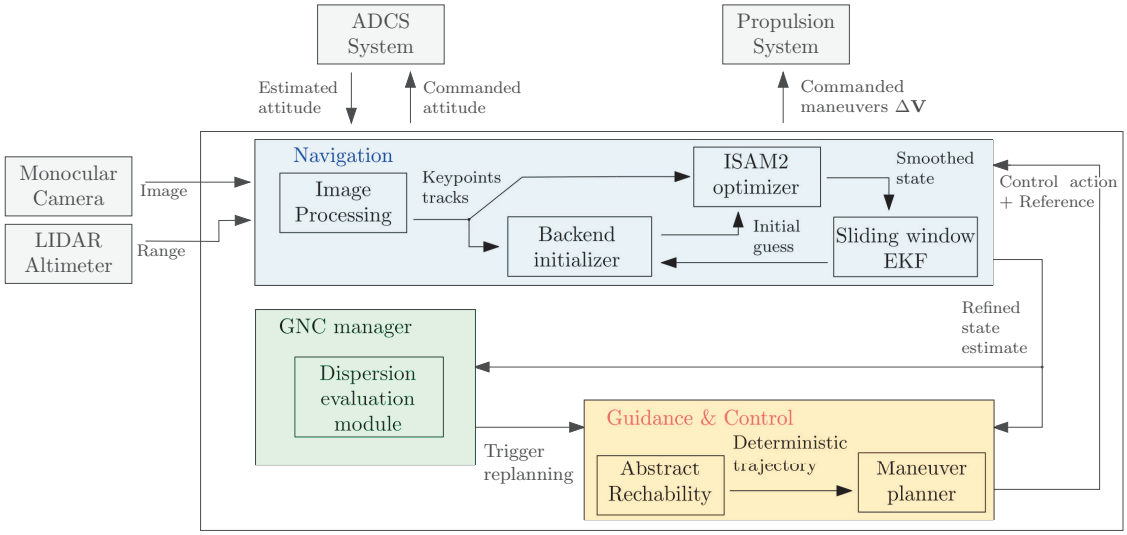


Figure 1: High level architecture of the proposed GNC solution.

The guidance module uses a set of pre-defined asteroid-fixed observation goals, as outlined in the Background section and relies exclusively on navigation outputs to perform trajectory planning, manoeuvre computation, and dispersion mitigation. In turn, the navigation estimates the spacecraft state taking advantage of the replanning capability to map the entirety of target asteroid, while remaining on a safe and stable trajectory.

The replanning trigger logic is the most relevant feedback the navigation provides to the guidance algorithms. Therefore, its robustness to errors is of paramount importance to prevent incorrect decisions and unnecessary control actions. The reference trajectory generated by the guidance algorithms during the latest run $[\bar{\mathbf{r}}(t); \bar{\mathbf{v}}(t)]$ is defined on a coarser timegrid with respect to the inertial navigation solution $[\hat{\mathbf{r}}(t_{eval}); \hat{\mathbf{v}}(t_{eval})]$.

Therefore, a simple spline interpolation is first performed to retrieve the state at the same time instant t_{eval}. Despite being approximate, this solution was found to be sufficiently reliable for the typically slow dynamics of asteroids' environment considering a discretization of the reference trajectory in the order of hundreds of seconds.

Due to the probabilistic nature of the navigation solution, the Mahalanobis distance is a natural choice, as it accounts for state estimation uncertainty while operating on normalized quantities, thereby simplifying threshold selection. The metric is thus evaluated as

$$\Delta x(t_{eval}) = (\bar{\mathbf{x}} - \hat{\mathbf{x}})^T \, \Sigma_{\mathbf{xx}}^{-1} \, (\bar{\mathbf{x}} - \hat{\mathbf{x}}) \tag{11}$$

where $\mathbf{x}^T = [\mathbf{r}^T, \mathbf{v}^T]$ collects both position and velocity. This joint evaluation is adopted because evaluating the deviation of the orbital state as a whole, rather than treating position and velocity separately, inherently accounts for their statistical correlation. At the same time, the metric is sensitive not only to position errors, which are more directly affected by measurement errors, but also to inconsistencies in the dynamical evolution. In fact, these results from modeling uncertainties and simplifying assumptions (e.g., for computational reasons), which manifest primarily through velocity deviations.

To further improve robustness, the dispersion evaluation is not performed at isolated time instants, but over a sequence of consecutive navigation steps, here set to five. A replanning request is therefore issued only when the Mahalanobis distance between the estimated state and the interpolated reference trajectory exceeds a prescribed threshold for the selected number of consecutive steps. The threshold is selected equal to $\Delta \mathbf{x}_{thr} = 10.5$. This choice is motivated by the fact that the state deviation defined from the estimated navigation solution is, by construction, expected to lie within the 3σ confidence region associated with the estimator covariance. However, even under the ideal assumption of a reference trajectory perfectly matching the true trajectory, stochastic estimation errors may cause the Mahalanobis distance to intermittently exceed this bound. To prevent spurious replanning triggers, the threshold is therefore empirically increased by a safety margin of 3.5, yielding the value adopted in this work for the integrated analysis in section Referencessec:results.

Navigation system algorithms

The proposed navigation system combines a graph-based SLAM estimator with a recursive estimator to simultaneously support target-relative and inertial navigation. This dual structure exploits the complementary strengths of batch-like smoothing and recursive filtering.[30] Visual measurements are generated by a dedicated frontend module.[50,51] Finally, the backend-initializer serves as interface component between frontend and backend.

Salient image features are extracted only at selected frontend keyframes and tracked across intermediate frames using a pyramidal KLT tracker.[52] The frontend keyframes determines keyframe insertion into the factor graph, whereas the opposite is not enforced. This ensures that newly extracted features are always anchored to a backend keyframe and maximizes their information. To mitigate the influence of outliers in the tracked features, a MSAC-based outlier rejection strategy using epipolar-geometry consistency checks is embedded in the frontend. This operates between the most recent keyframe and the newest acquired frame.

To improve feature extraction capabilities using classical computer vision algorithms, an additional preprocessing steps remove features near the target outline and terminator and promote spatial coverage in the image plane.[53] The frontend maintains complete feature-track histories and passes them to the backend at each update. Tracks contribute to SLAM only when they link at least two keyframes. Because no feature descriptors are used for tracking, lost tracks are not re-associated, and refinement relies on backend optimization. SURF features are used in the selected scenario, as they provide a good trade-off between computational cost and robustness to non-sharp image appearance.

Lidar range measurements to the surface are considered as an additional source of information. The sensor is assumed to provide one measurement at each filter update, occurring every 5 s. In addition, spacecraft attitude measurements with respect to the inertial frame are assumed to be available at all time instants and are corrupted by multiplicative white-noise errors to model attitude estimation uncertainty.

The graph-based estimator operates in the target-fixed frame and estimates spacecraft pose and velocity, namely the states $\{[\mathcal{BC}]_k, \hat{\mathbf{r}}_k^{\mathcal{B}}, \hat{\mathbf{v}}_k^{\mathcal{B}}\}$ where $k = 1, ..., N_t$, together with a sparse 3D landmark map $\{\hat{\mathbf{l}}_i^{\mathcal{B}}\}$ $i =$

$1, ..., N_l$, treating landmarks as static. N_t and N_l is the total number of time instants corresponding to keyframes and the number of landmarks, respectively. Based on ISAM2 optimizer, the estimator incrementally builds and optimizes a factor graph containing keyframe states and landmark variables using projection, dynamics, pose and attitude prior factors as described in section Background.

Keyframes are inserted based on a combination of frontend-driven criteria, such as the number of active tracks, and backend time-based triggers. Therefore, this estimator can operate at the imaging acquisition rate at most. New landmarks are initialized by multi-view triangulation, namely the Direct Linear Transform, using a bounded window of frames covering at least the number of frames between two keyframes.[53] These landmarks estimates serve only as initial guesses and are subsequently refined through nonlinear optimization. To improve robustness during early operation, the first initialization is performed through a bootstrap mode consisting in a limited size bundle-adjustment step over the initial window before running incremental smoothing. In this work, the limit is set to 8 frames, after which the smoother switches to ISAM2. Finally, dynamical constraints are included between consecutive keyframes to enforce consistency and enable velocity estimation. Although the states are estimated in the target-fixed frame, the dynamics is evaluated in the inertial frame using the target attitude at the keyframe time instants.[30] The covariance associated to the error is obtained through trapezoidal integration rule of the process noise term across intermediate timesteps between the two states.

In parallel and asynchronously, a recursive estimator designed as an Extended Kalman Filter (EKF) estimates the inertially referenced orbital state and provides high-rate estimates of spacecraft position and velocity, contrarily to the smoother. The sliding window of past states in the EKF is retained to allow delayed updates when required. In addition to the orbital states, the filter state vector includes dynamical and measurement biases, modelled as First Order Gauss Markov processes or as consider state.[54,55] The full state vector is therefore written as

$$\mathbf{x}_{\,\mathrm{filter}} = \left[\mathbf{r}^{\mathcal{I}};\, \mathbf{v}^{\mathcal{I}};\, \Delta C_{\mathrm{SRP}};\, \mathbf{a}_{\mathrm{res}};\, b_{\mathrm{lidar}};\, \log_{10}(\mu)\right]. \tag{12}$$

where the log10 is used to ensure positivity of the gravitational parameter. This is treated as consider state together with the scalar bias of the SRP coefficient defined as $\hat{C}_{SRP} = \bar{C}_{SRP} + \Delta C_{SRP}$, which aims to capture discrepancy in the SRP magnitude. The residual-acceleration term acts as an additive perturbation in the equations of motion and is estimated. Finally, the lidar bias b_{lidar} enters the lidar measurement prediction model as $\hat{d}_{\mathrm{corr}} = \hat{d} + b_{\mathrm{lidar}}$, where $\hat{d}$ is the predicted measurement through a ray-sphere intersection model using the mean radius of Itokawa.

Lastly, it is noted that the dynamical model adopted in both estimators includes only central gravity and cannonball solar radiation pressure as deterministic accelerations. Moreover, the smoother does not incorporate stochastic bias estimation within the factor graph.

The interaction between the two estimators is intentionally loose to mitigate the effect of biases in the smoother estimates. The recursive estimator does not process raw visual measurements and does not estimate landmarks. Instead, it is updated at keyframes by treating the smoothed spacecraft position produced by the graph-based estimator as a measurement expressed in the inertial frame, through the classical Kalman filtering equations. The covariance of the measurement is retrieved as marginal covariance from the smoother and inflated by a manually tuned factor. Furthermore, underweighting is used to prevent excessively fast convergence of the filter.[56] Importantly, prior knowledge of the target asteroid rotational state is assumed at this stage. Future work will investigate improved formulations to enhance statistical consistency, mitigate potential error feedback loops, and enable estimation of the asteroid attitude.

Conversely, the filter forward state prediction is used only to support the backend initializer. Depending on the configuration, the filter position may be used as an initial guess for the smoother. However, preference is given to initializing new variable nodes through the dynamics factor model itself, leaving triangulation as the sole source of feedback from the filter. In fact, no additional information from the filter enters the smoother.

Guidance algorithms

In order to generate a trajectory, the general concepts of the framework introduced in Section Background must be translated into a form usable by an optimization algorithm. Starting from the scientific requirements,

it is possible to build an objective using the following continuous function[39]

$$\omega_f(\mathbf{y_f}) = \mathcal{D}_f\big(y_{f,1}\big) \cdot \mathcal{E}_f\big(y_{f,2}\big) \cdot \mathcal{I}_f\big(y_{f,3}\big). \tag{13}$$

where

$$\mathbf{y_f}(\mathbf{r}, t) = \begin{bmatrix} y_{f,1} \\ y_{f,2} \\ y_{f,3} \end{bmatrix} = \begin{bmatrix} \rho_f(\mathbf{r}) \\ \epsilon_f(\mathbf{r}) \\ i_f(t, \mathbf{r}) \end{bmatrix}. \tag{14}$$

The functions $\mathcal{D}_f$, $\mathcal{E}_f$, and $\mathcal{I}_f$ are sigmoid functions:

$$\begin{aligned} \mathcal{D}_f(y_{f,1}) &= \frac{1}{1 + e^{-K_1\left(y_{f,1} - \left(\frac{\rho_f^-}{\bar{R}}\right)^2\right)}} \cdot \frac{1}{1 + e^{K_1\left(y_{f,1} - \left(\frac{\rho_f^+}{\bar{R}}\right)^2\right)}}, \\ \mathcal{E}_f(y_{f,2}) &= \frac{1}{1 + e^{-K_2\left(y_{f,2} - \cos\epsilon_f^+\right)}}, \\ \mathcal{I}_f(y_{f,3}) &= \frac{1}{1 + e^{-K_3\left(y_{f,3} - \cos i_f^+\right)}}. \end{aligned} \tag{15}$$

where K_1, K_2, K_3 define the steepness of the sigmoid functions, $\bar{R}$ is used to normalize the range. With the function in Eq. (13), it is possible to define the objective function

$$T_f = \int_{t_0}^{T_h} \omega_f(\mathbf{y_f})\, d\tau \tag{16}$$

where $[t_0,\ T_h]$ is the considered control horizon. T_f is the time spent satisfying the conditions defining the generic $\mathcal{R}_f$. In addition to the scientific requirements, engineering constraints must be enforced to ensure safe operational conditions and compliance with the available propulsion capabilities. In this work, safety is imposed by constraining the spacecraft state to lie within the set

$$\mathcal{X} = \{\mathbf{r} \in \mathbb{R}^3 \mid\ R_{imp} \leq\ \|\mathbf{r}\| \leq R_{esc}\} \tag{17}$$

where R_{imp} and R_{esc} are the minimum and maximum ranges within which the spacecraft state must evolve. This operational constraint can also be enforced in a soft form through the definition of the following penalty function:

$$V_P(\mathbf{r}) = -\left(\frac{K_{imp,1}}{1 + e^{K_{imp,2}(\|\mathbf{r}\| - R_{\text{imp}})}} + \frac{K_{esc,1}}{1 + e^{-K_{esc,2}(\|\mathbf{r}\| - R_{\text{esc}})}}\right) \tag{18}$$

where similarly to Eq. (15), the parameters $K_{imp,1}$, $K_{imp,2}$, $K_{esc,1}$ and $K_{esc,2}$ determine the shape of the penalty sigmoids.

The final engineering requirement considered in this work is compliance with the maximum impulse Δv_{max} that the system can provide. Accordingly, the control impulses must belong to the set

$$\mathcal{U} = \{\mathbf{\Delta v} \in \mathbb{R}^3 \mid\ \|\mathbf{\Delta v}\| \leq \Delta v_{max}\}. \tag{19}$$

Using the objective and constraints formulations provided above, the goal-oriented guidance problem is expressed as follows:

$$\underset{t_{i+1},\, \mathbf{\Delta v}_i}{\text{maximize}} \quad \sum_{f=1}^{N_f} T_f\ +\ \int_{t_i}^{t_{i+1}} V_p(\mathbf{x})\, d\tau \tag{20a}$$

$$\text{subject to} \quad \dot{\mathbf{x}} = \mathbf{f}_f(\mathbf{x}, t) + \mathrm{B} \sum_{i=1}^{N_m} \delta\big(t - t_i\big)\, \mathbf{\Delta v_i}., \qquad \forall\, t \in [t_i, t_{i+1}], \tag{20b}$$

$$\mathbf{x}(t_i) = \mathbf{x}_i, \tag{20c}$$

$$\mathbf{\Delta v}_i \in \mathcal{U}, \tag{20d}$$

$$t_{i+1} \in \mathcal{T}_i. \tag{20e}$$

where T_f is computed over the time horizon $[t_i,\ t_{i+1}]$, and $\mathbf{x}_i$ is the initial state provided by the navigation module. $\mathcal{T}_j$ is a time interval defined by the minimum maneuver separation time ΔT_m and the control horizon T_h. At each trajectory update step, the next maneuver time t_{i+1} is required to belong to the set $\mathcal{T}_j$ and is determined according to the evolution of $T_{f,tot} = \sum_{f=1}^{N_f} T_f$ and V_c following the heuristic introduced in Reference 39. The guidance problem in Eq. (20) is solved through abstract reachability analysis. The scheme is summarized in Algorithm 1. Starting from an initial control domain sampling, the objective values of the reachable set are evaluated and used to perform a refinement step that enhances the optimality of the final solution. Both the initial and refinement samplings exploit a Fibonacci distribution[57] to have an approximately uniform coverage of the control space.

Algorithm 1: Sampling-based scheme

Input: initial samples n_0, max iterations n_{it}, best samples n_{best}, refinement samples n_{ref}
Output: $\boldsymbol{\Delta}\mathbf{v}^*$ maximizing $T_{f,\mathrm{tot}}$
(1) Initial sampling: $U^{(0)} \leftarrow \textsc{InitialSampling}(n_0)$;
for $k = 0, \ldots, n_{\mathrm{it}}$ **do**
  (2) Forward propagate: **for all** $\boldsymbol{\Delta}\mathbf{v} \in U^{(k)}$ $\mathrm{traj}(\boldsymbol{\Delta}\mathbf{v}) \leftarrow \textsc{ForwardPropagate}(\boldsymbol{\Delta}\mathbf{v})$;
  (3) Evaluate objective: **for all** $\boldsymbol{\Delta}\mathbf{v} \in U^{(k)}$ $T_{f,\mathrm{tot}}(\boldsymbol{\Delta}\mathbf{v}) \leftarrow \textsc{Evaluate}(\mathrm{traj}(\boldsymbol{\Delta}\mathbf{v}))$;
  (4) Select most promising: $S^{(k)} \leftarrow \textsc{SelectBest}(U^{(k)},\ T_{f,\mathrm{tot}},\ n_{\mathrm{best}})$;
  if $k = n_{\mathrm{it}}$ **then**
    break
  (5) Refine mesh: $U^{(k+1)} \leftarrow \textsc{RefineMesh}(S^{(k)},\ n_{\mathrm{ref}})$;

ANALYSIS AND RESULTS

Scenario definition

The numerical evaluation of the proposed methodology is performed in a simulated proximity-operations scenario around asteroid Itokawa. The main parameters defining the dynamical environment and spacecraft properties are summarized in Table 1, while the algorithm configuration parameters are reported in Table 2. Estimator tuning parameters are omitted for brevity, as they are strongly scenario-dependent. The asteroid gravity model used for both the environment and the guidance dynamical models is based on Spherical harmonics computed from the Hayabusa's shape model[58] through the methodology in Reference 59. From the same shape model, a set of 14 features is extracted, and the associated ROIs are constructed using the parameter intervals listed in Table 2. The features are selected randomly on the unit sphere in order to ensure a uniform mapping. Fig. 2 shows the initial conditions used for all analyses described in the following sections, which are obtained through a uniform sampling along a sun synchronous terminator orbit (SSTO) with a semi-major axis of $1.4\ \mathrm{km}$, propagated over one orbital period. The initial ephemeris time and state of the SSTO are reported in Table 1. Attitude pointing is generated assuming ideal nadir pointing with respect to the true trajectory. This simplifying assumption is adopted to decouple attitude pointing from the navigation problem, thereby preventing potential failures due to their mutual coupling and allowing the analysis to focus on translational motion only. The joint treatment of attitude pointing and navigation is left as future work.

Analysis sets and simulation tools

To comprehensively characterize the performance of both the overall system and its individual modules, several analysis sets with different fidelity levels are defined. This is enabled by the flexibility of the implemented simulation framework, which supports multiple degrees of modeling fidelity. In this work, three fidelity levels are considered.

1. **Fully integrated system with rendering in the loop.** Blender is employed as the rendering engine, ensuring geometric consistency of the generated images. Radiometric consistency is neglected, as it is

Table 1: Scenario simulation parameters.

	Parameter	Value	Uncertainty (knowledge, 3σ)
Spacecraft	m	12.5 kg	5%
	C_r	1.29	15%
	A_{SRP}	0.5329 m^2	30%
Control action	σ_{mag}	1%	[-]
	σ_{dir}	0.5°	[-]
Asteroid target	Gravity model	Spherical harmonics	[-]
	μ	2.36×10^{-9} $\mathrm{km}^3/\mathrm{s}^2$	7.5%
	Order and degree	16×16	[-]
	T	12.1324 h	[-]
Trajectory	Initial Ephemeris Time	89667200 s	[-]
	Initial position	$[-0.238,\ 0.377,\ -1.19]^{\mathrm{T}}$ km	[-]
	Initial velocity	$[2.87,\ -2.83,\ -1.47]^{\mathrm{T}} \times 10^{-5}$ km/s	[-]
	Position prior knowledge	150 m (3σ)	[-]
	Velocity prior knowledge	4 mm/s (3σ)	[-]
Measurements	Attitude knowledge	35 arcsec (1σ)	[-]
	Lidar range measurement	Raytracing + 5 meters white gaussian noise (1σ)	[-]

Table 2: Algorithms tuning parameters.

	Parameter	Setting
Guidance	Gravity type	Spherical harmonics
	Order and degree	4×4
	$(K_1,\ K_2,\ K_3,\ \bar{R})$	$(10,\ 100,\ 100,\ 1\ \mathrm{km})$
	$\Delta v_{max},\ \Delta T_m,\ T_h$	$(0.1\ \mathrm{m/s},\ 15\ \mathrm{h},\ 20\ \mathrm{h})$
	$(R_{imp},\ K_{imp,1},\ K_{imp,2})$	$(0.7\ \mathrm{km},\ 1,\ 100)$
	$(R_{esc},\ K_{esc,1},\ K_{esc,2})$	$(3\ \mathrm{km},\ 1,\ 5)$
	$(n_0,\ n_{it},\ n_{best},\ n_{ref})$	$(349,\ 3,\ 6,\ 18)$
	$(\rho_f^-,\ \rho_f^+)$	$(0.9\ \mathrm{km},\ 1.3\ \mathrm{km})$
	$(\epsilon_f^-,\ \epsilon_f^+)$	$(0^\circ,\ 15^\circ)$
	$(i_f^-,\ i_f^+)$	$(0^\circ,\ 90^\circ)$
Navigation	Camera matrix K	$\begin{bmatrix} 5891.54 & 0 & 1024 \\ 0 & 5889.13 & 768 \\ 0 & 0 & 1 \end{bmatrix}$
	Lidar acquisition rate	0.2 Hz
	Camera imaging interval	45 s
	Max keyframes step	6 frames
	Min/Max features count	90/300
	Max track length	150 frames

not relevant to the scope of this study. The Blender model is based on sample scenarios provided by the open-source tool described in Reference 60.

2. **Integrated system with simulated image processing.** The emulation model introduced in Reference 30 is used to generate synthetic keypoint tracks, eliminating the need for online rendering during simulation. This significantly reduces computational cost, enabling extensive Monte Carlo analyses.

Keypoints are generated using ray-tracing algorithms, including shadow-ray illumination checks, ensuring consistency with both scene geometry and lighting conditions. All navigation and guidance algorithms, except for feature extraction and tracking, are executed as in the highest-fidelity configuration. The tracking input noise is set to $\sigma_{\text{track}} = 1.5\,\text{pix}$.

3. **Surrogate navigation model for guidance analysis.** The navigation module is replaced by a stochastic error model that generates synthetic state estimates by perturbing the true simulation trajectory with white Gaussian noise. Perturbations are sampled assuming a diagonal covariance matrix, tuned to represent attainable navigation performance. Specifically, a knowledge error of $3\sigma_r = 50\,\text{m}$ for each position component and $3\sigma_v = 3\,\text{mm/s}$ for each velocity component is assumed. The guidance algorithm is executed with the same configuration, while no navigation algorithm is run.

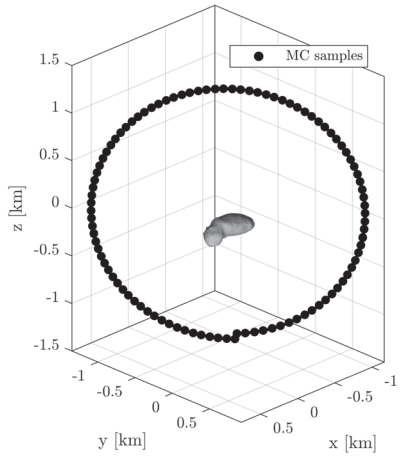


Figure 2: Initial spacecraft positions considered in the MC simulations - Inertial frame.

Each fidelity level is used to support a specific analysis set:

1. **Single-run simulation with synthetic images.** A single simulation run is performed using the full rendering-in-the-loop configuration. Although this does not allow for statistically robust conclusions, it demonstrates the correct operation of the complete system when processing actual rendered images rather than simulated measurements. Results for this case are presented in the Fully integrated system single run results Section.

2. **Monte Carlo analysis of the integrated system (100 samples).** Image processing algorithms are replaced by the emulation model to avoid the computational cost of rendering a large number of images at high acquisition rates. This analysis provides a statistical characterization of the combined navigation and guidance system. Results are presented in Navigation MC analysis using emulated image processing Section.

3. **Monte Carlo analyses of the guidance subsystem (100 samples).** The navigation solution is replaced by the surrogate error model to isolate the performance of the guidance scheme and reduce the computational cost of the simulations. The simulations enable a comparative assessment of the guidance module performances under different replanning triggers. The results of this analysis are presented in Guidance MC analysis using navigation dummy Section.

Fully integrated system single run results

Fig. 3 illustrates the results of a single-run simulation of the fully integrated GNC architecture, showing the simulated inertial trajectory and the executed maneuver and observation events in both the inertial and

body-fixed frames, respectively on the left and on the right side. The total duration of the simulation is set to 2.30 days, with the first guidance execution scheduled after 3.5 hours since the start. This was chosen to ensure a stable navigation solution before the first manoeuvre.

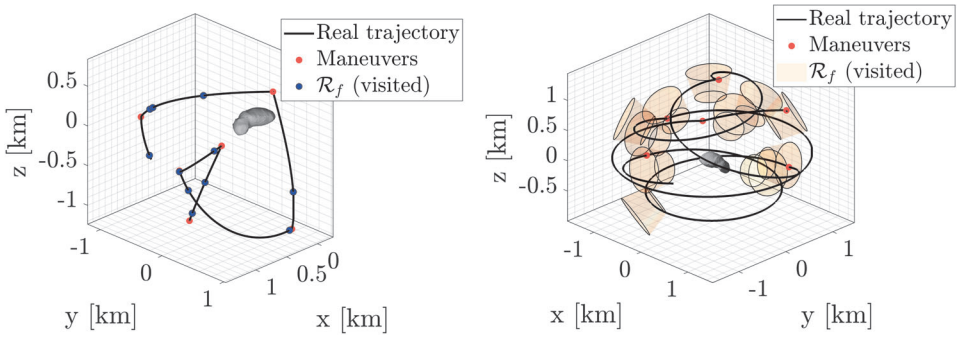
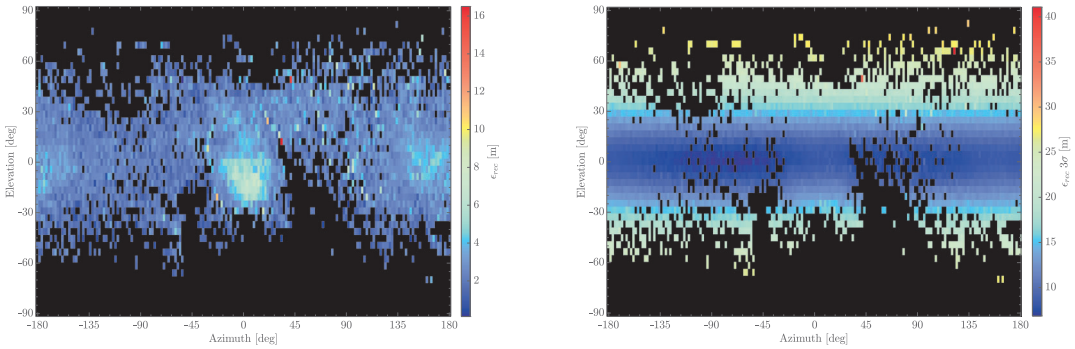


Figure 3: Fully integrated system single-run guidance results in inertial and target-fixed frames.

From the navigation standpoint, the single run simulation is mainly analyzed in terms of final estimated landmarks quality. Figures 4a and 4b represent binned 2D maps in spherical coordinates (azimuth and elevation), respectively for the position norm error and the estimator position norm variance computed through linear covariance transformation. Each cell of the grid corresponds to 2 degrees of azimuth and elevation of the target. Landmarks estimation errors are computed with respect to their true position, obtained via ray-tracing of the initially extracted keypoint and assigned to a bin depending on the true spherical coordinates. Finally, the bin value mapped onto the colormap is determined by averaging all the values assigned to it.



(a) Position norm estimation error.

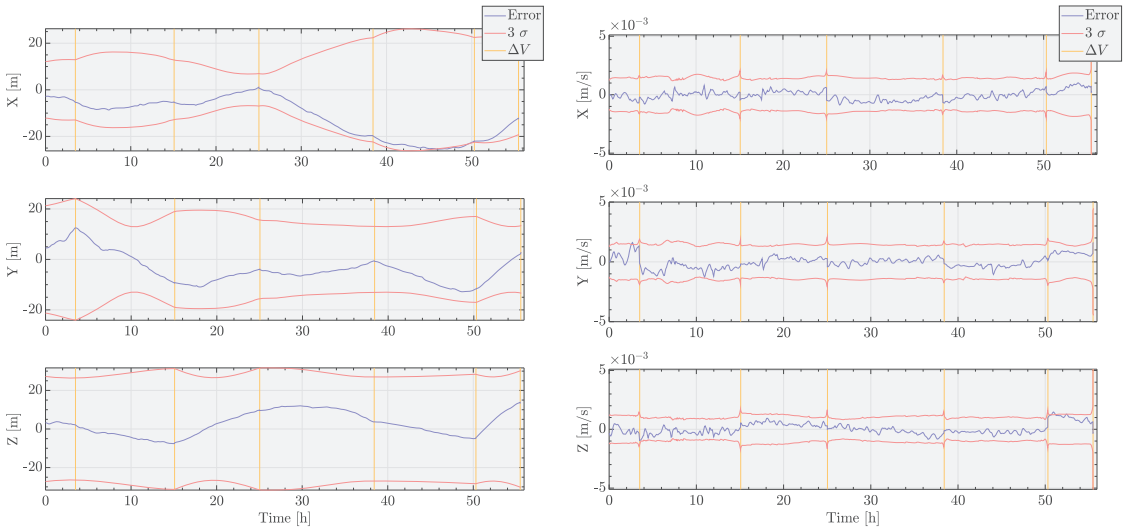
(b) Position norm estimator 3σ std. deviation.

Figure 4: Binned 2D map of landmarks estimates and covariance (2° resolution).

While a two-day navigation session is insufficient to map the entire target, as indicated by bins marked in black (no landmarks assigned), 11 out of 14 regions of interest were successfully visited. Most of the surface was mapped between -60° and 60° latitude, with map estimation errors well below 10 m on average. The polar regions were either poorly observed or affected by adverse illumination conditions, as evidenced by the sparse distribution of landmarks. Localized error peaks are observed in some bins, likely caused by slight outliers in the landmarks associated with those regions. By comparing the geometry of the target-fixed trajectory in Fig. 3 with the estimator variance map, a clear correlation emerges: regions with lower variance and errors well contained within the 3σ bounds correspond to areas with better observability, characterized by longer feature tracks, a higher number of measurements, and a larger set of initialized landmarks. Conversely, the polar regions exhibit larger covariance values. Nevertheless, in all cases, the estimator consistently provides a reliable representation of the underlying uncertainty, including in correspondence with the observed estimation error peaks.

The final smoothed ISAM2 solutions for position and velocity, shown in Figs. 5a and 5b, represent the trajectory counterpart of the landmark error maps. In a SLAM formulation, trajectory and map errors are intrinsically coupled: inaccuracies in the estimated states propagate to the reconstructed map and, conversely, map uncertainties affect the trajectory estimates.

The solutions are expressed in the inertial frame by transforming through the true asteroid attitude, while the inherent 6-DOF ambiguity of the SLAM problem is removed by fixing the first pose through a prior factor. As a result, both the estimated map and trajectory can be directly compared against the simulated reference trajectory. The localization solution exhibits consistent trends with respect to accuracy and covariance of the map estimates in both position and velocity. Maneuvers events are indicated by yellow vertical lines and clearly show that the smoothing process effectively mitigates the adverse effects of control errors, although they remain slightly noticeable.



(a) Position estimation errors.

(b) Velocity estimation errors.

Figure 5: Estimation errors of the smoothed solution at the final time instant, expressed in the Inertial frame.

Guidance MC analysis using navigation dummy

In order to assess the impact of different replanning triggers on guidance performance, a comparative analysis is conducted. The analysis focuses on the evolution of trajectory dispersion, the satisfaction of scientific requirements, and fuel consumption. Six MC simulations are performed using the set of 100 initial conditions reported in Scenario definition Section. The simulations differ only in the replanning threshold used to trigger a guidance update. Specifically, the dispersion-based replanning strategy is evaluated for five threshold levels $\Delta\mathbf{x}_{thr}$, expressed as percentage of the 3σ knowledge, namely 200%, 300%, 400%, 500%, and 600%. In addition, the replanning trigger reported in Guidance algorithms Section is included as a reference case and will be referred to as "heuristic". The mission duration is set to 48 h for all the simulations.

From Fig. 6, which represents the dispersion evolution in terms of ensemble mean μ and 3σ upper bound $u_{3\sigma}$, it can be noted that lower dispersions are associated with smaller replanning thresholds, given the higher frequency of trajectory updates. The higher guidance frequency can also be observed from the more irregular trends of $u_{3\sigma}$, which also present sharp decreases at multiples of 20 h, as the heuristic dictates the maximum time horizon of a planned trajectory. The reduced dispersions achieved with lower replanning thresholds translate into improved scientific return, as shown in Fig. 7, where the ensemble mean Cumulative Scientific Time $\overline{CST}$ and the number of ROIs satisfied $\bar{N}^{sat}_{ROI}$ are represented. Table 3 reports these metrics at the final time t_f, revealing a 12.8% increase in $\overline{CST}$ for the 200% threshold with respect to the heuristic-based replanning. On the other hand, the improvement in $\bar{N}^{sat}_{ROI}$ is reduced (6.5%), with generally contained vari-

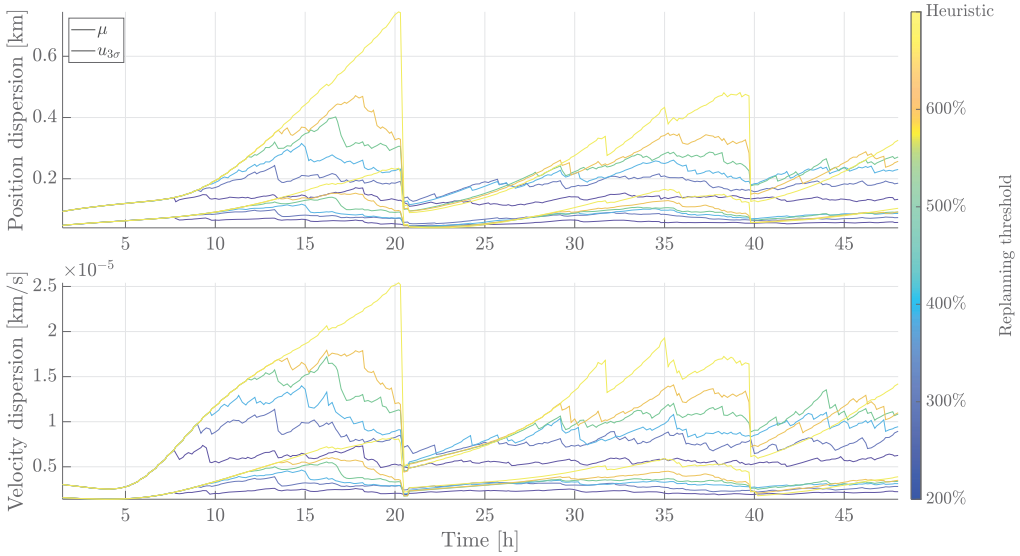


Figure 6: Evolution of position and velocity dispersion for different replanning triggers.

ations between all the triggers. This reduced gain is due to the fact that the CST is the objective explicitly maximized by the guidance scheme.

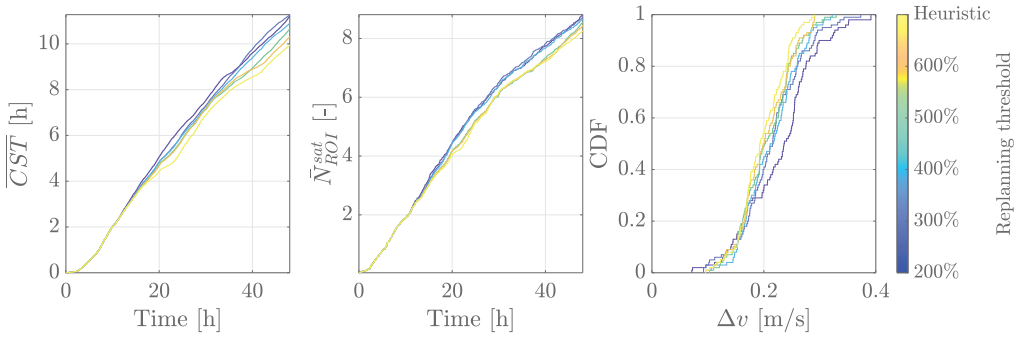


Figure 7: Evolution of goals metrics and cumulative distribution function of the total Δv for different replanning triggers.

As anticipated, these improvements are strictly related to the reduced dispersion levels observed over the mission time. However, the increase in the mission objective is significantly smaller than the 51.6% reduction observed in the temporal mean of $u_{3\sigma}$ of the dispersion in the 200% threshold compared to the heuristic, as reported in Table 3. Consistently, this increased control effort is reflected in the propellant consumption: the stochastic Δv_{99}, estimated as the 99-th percentile of the cumulative distribution function of the Δv distributions shown in Fig. 7, is 34.5% higher for the 200% threshold.

The limited commensurability between the increased control effort and the resulting gains in mission objectives could be related to the fact that, within the goal-oriented framework, trajectory dispersion is less relevant than in other operational scenarios. In this context, where the scientific return plays a central role, a replanning strategy based on the difference between the nominal and the predicted scientific return could be particularly effective. However, this approach would require increased computational resources allocated to the GNC manager, as it would rely on trajectory propagation to evaluate the predicted scientific return.

Table 3: Guidance performance metrics for different replanning triggers.

Threshold	$\bar{u}_{3\sigma}$ [km]	$\bar{\mu}$ [km]	$\overline{CST}(t_f)$ [h]	$\overline{N}^{sat}_{ROI}(t_f)$ [-]	Δv_{99} [m/s]
200%	0.133	0.059	11.22	8.81	0.39
300%	0.175	0.073	11.28	8.72	0.36
400%	0.200	0.079	10.92	8.66	0.32
500%	0.221	0.083	10.68	8.56	0.32
600%	0.237	0.089	10.32	8.45	0.29
Heuristic	0.275	0.103	9.95	8.27	0.29

Navigation MC analysis using emulated image processing

Navigation performance, with emphasis on localization accuracy and consistency, is characterized through 100 Monte Carlo simulations using emulated keypoint tracks. These correspond to the 100 samples shown in Fig. 2, each initialized with different errors in the initial orbital state and dynamical parameters. The errors are sampled from Gaussian distributions with standard deviations reported in Table 1. Attitude estimation errors are neglected in this analysis, as attitude pointing is assumed ideal and derived from ephemerides. The first guidance execution is scheduled 3.5 hours after the start of the simulation.

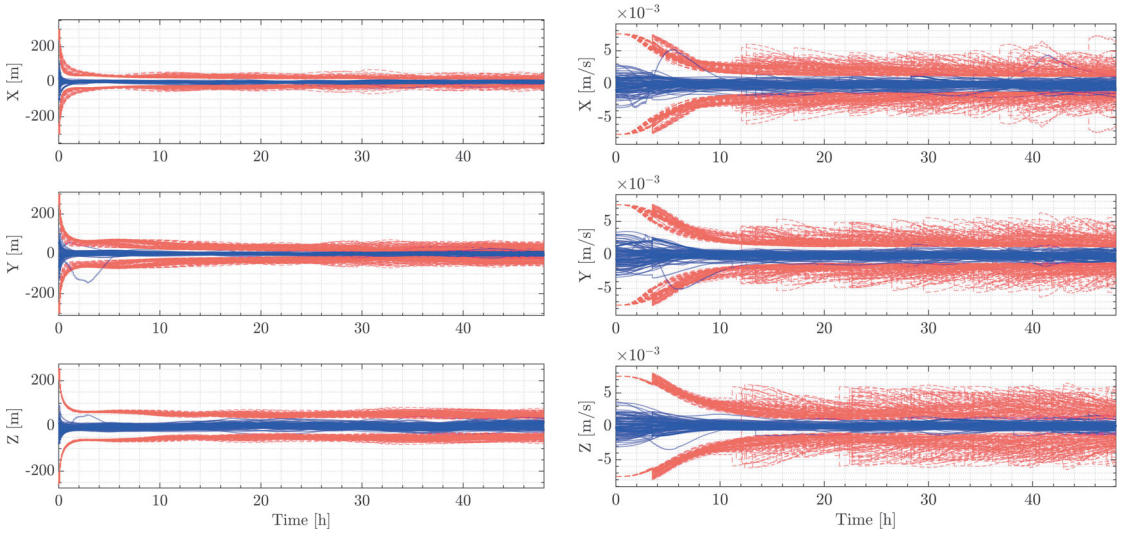
The proposed navigation system provides three navigation solutions with different update rates and accuracy levels. First, the most frequent online estimates are produced by the filter in the inertial frame. These estimates fuse high-rate measurements, namely lidar range measurements in this scenario, with position information provided by the smoother, while accounting for relevant dynamical uncertainties through bias and consider states. Second, an *online incremental* smoother solution is available at keyframes in the target-fixed frame, providing estimates of both position and velocity. Third, a *smoothed incremental* solution can be computed on demand by re-evaluating the Maximum A Posteriori estimate of the complete orbital state history and landmark map. The term *incremental* here refers to the fact that the solution is obtained from ISAM2 and not by solving the entire estimation problem using a batch optimizer.

The key difference between the second and third solutions lies in the treatment of past states: the *online incremental* solution reports errors associated with the most recent estimates only, whereas the *smoothed incremental* solution incorporates retroactive updates of past states as additional measurements are processed. Position and velocity errors for the three solutions are shown in Figs. 8a, 9a and 10a and Figs. 8b, 9b and 10b, respectively. Filter position errors are reported in the camera frame to highlight the contribution of lidar range measurements, while smoother solutions are expressed in the inertial frame, as they enter the filter as measurements and better reveal time intervals affected by systematic errors.

Overall, all navigation solutions provide accurate and consistent state estimates, with the fully smoothed solution achieving the highest accuracy despite not explicitly exploiting lidar range measurements. Notably, the effects of maneuver execution are clearly visible in the online solutions, both in the estimated errors and associated covariances, reflecting the impact of control and knowledge uncertainties. These effects, however, are almost entirely eliminated in the smoothed solution, highlighting the effectiveness of smoothing in mitigating control-knowledge errors, which are typically among the dominant contributors in ground-based mission analysis. Furthermore, this performance is achieved without explicitly estimating their mean error value, which is instead modeled as zero mean Gaussian random variables, and relies solely on dynamical constraints and keypoint-based visual measurements.

As shown in Figs. 8 and 9, all Monte Carlo samples eventually converge to low error levels, largely below 50 m and well contained within the associated 3σ covariance bounds. However, in at least two cases, the online incremental smoother exhibits difficulties during initialization and maneuver execution, resulting in noticeably large error peaks near the beginning of the simulation and around 28 hours. In both instances, these deviations are subsequently absorbed and falls back within the estimator covariance. While the fully smoothed solutions reported in Fig. 10 no longer display such transient peaks, the same two samples still exhibit mild outlying behavior, with covariance estimates that slightly underestimate the actual error.

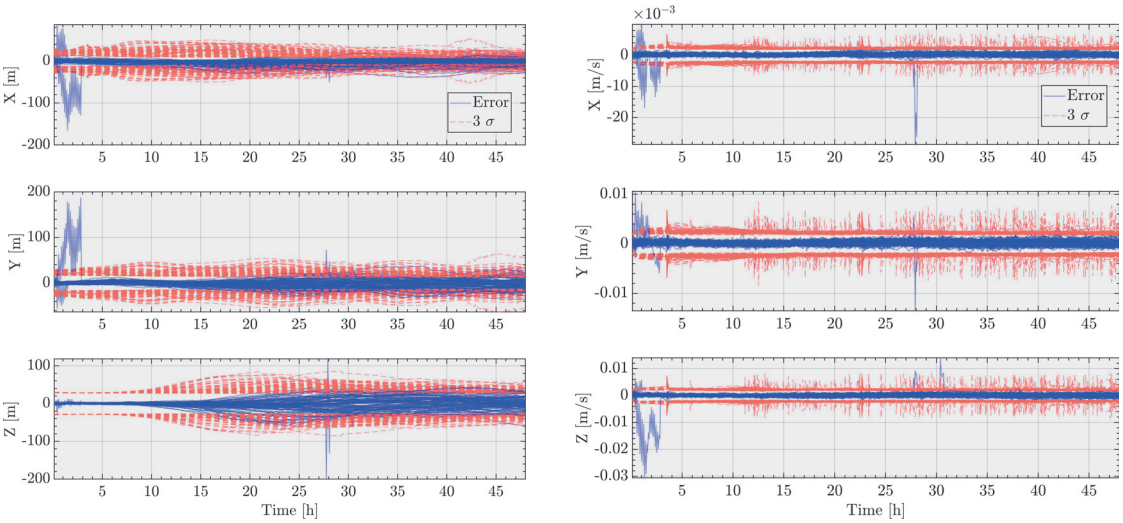
With the exception of the initial failure at the beginning of the time window, the filter generally acts as a mitigation layer, limiting the propagation of these errors to the guidance algorithms. This mitigation, how-



(a) Position estimation errors - Camera frame.

(b) Velocity estimation errors - Inertial frame.

Figure 8: Filter estimation errors and 3σ estimator knowledge bounds.



(a) Position estimation errors - Inertial frame.

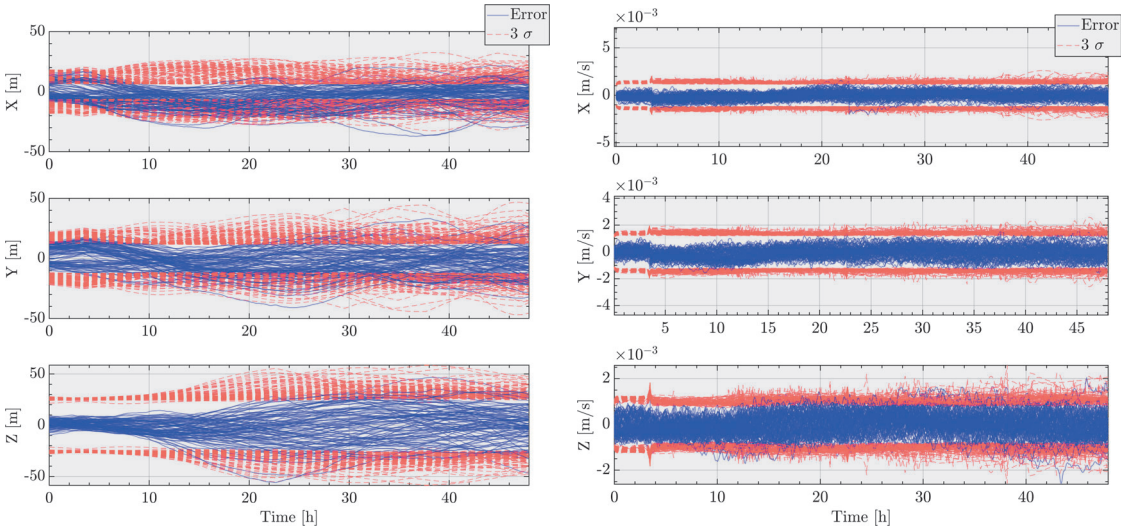
(b) Velocity estimation errors - Inertial frame.

Figure 9: ISAM2 "online" estimation errors and 3σ estimator knowledge bounds.

ever, is not uniformly effective across all position components. Since the online incremental smoother output is effectively treated as a measurement by the filter, extended intervals in which iSAM2 provides significantly biased estimates directly translate into biased filter states, which cannot properly be accounted for. Improving this interface to prevent the emergence of detrimental feedback loops is therefore identified as an important direction for future work, particularly from a navigation safety perspective.

Finally, Fig. 8 shows that the filter delivers highly accurate position and velocity estimates, characterized by low covariance values, around 60 meters and 3 mm/s 3σ on average, while effectively handling the additional noise introduced by maneuver execution. After maneuvers, the filter typically reconverges within 5–6 hours. The inclusion of lidar range measurements provides benefits in the position component along camera

boresight, such that accuracy comparable with the across-boresight components is achieved. This contrasts with the typical behavior of vision-only navigation, in which uncertainty along the boresight direction is always significantly larger than in the transverse directions.



(a) Position estimation errors - Inertial frame.

(b) Velocity estimation errors - Inertial frame.

Figure 10: ISAM2 "smoothed" estimation errors and 3σ estimator knowledge bounds.

As a final remark for future work, the importance of robust initialization and an improved lidar prediction model is emphasized. The authors observed that specific combinations of illumination conditions and target attitude are more likely to induce large errors at startup or even lead to failure of the incremental estimator, primarily due to the limited number of available keyframes at initialization.

Secondarily, owing to the morphology of Itokawa, lidar residuals exhibit strong biases when the spacecraft flies over the elongated lobe at low latitudes. This behavior is a direct consequence of adopting a spherical asteroid shape model with a fixed mean radius, which is largely approximate for highly elongated bodies. Addressing these limitations, and improving robustness in both initialization and lidar modeling, will be the focus of future extensions of this work.

CONCLUSION

This work presented a novel closed-loop GNC architecture designed to enable autonomous active SLAM during close-proximity operations around small bodies. By integrating a graph-based visual SLAM navigation system with a goal-oriented MPC guidance framework, the proposed solution addresses the coupling between state estimation and trajectory generation often neglected in state-of-the-art research. The navigation subsystem, employing an asynchronous filter-graph formulation, demonstrated the capability to jointly estimate the spacecraft trajectory and a sparse landmark map with high accuracy, achieving positioning errors below 10 meters and effective map reconstruction in a high-fidelity simulation around asteroid Itokawa.

Numerical analyses validated the effectiveness of the guidance strategy, which utilizes abstract reachability analysis and a dispersion-based replanning trigger to autonomously adapt to navigation uncertainties. Results indicated that tighter replanning thresholds significantly reduce trajectory dispersion by over 50% and improve cumulative scientific return by approximately 12.8% compared to heuristic baselines, albeit at the cost of increased propellant consumption. While the current implementation relies on simplifying assumptions regarding attitude control and prior target knowledge, the findings confirm that coupling perception-aware navigation with flexible, goal-driven guidance is a viable path toward fully autonomous asteroid exploration.

ACKNOWLEDGMENT

This work is partially sponsored by the COSMICA project funded by the Italian Ministry of University and Research (MUR) through Decree No. R.0000398.24-10-2022.h.10:43. P.C. and A.B. are sponsored by the European Union under the Next Generation EU program, Mission 4 Component 1 (CUP D43C23002190008).

REFERENCES

[1] D. J. Scheeres, *Orbital motion in strongly perturbed environments: applications to asteroid, comet and planetary satellite orbiters*. Chichester, UK: Springer, 2016, pp. 18-33.

[2] D. W. Dunham, J. V. McAdams, and R. W. Farquhar, "NEAR mission design," *Johns Hopkins APL technical digest*, Vol. 23-1, 2002, pp. 18–33.

[3] M. D. Rayman, "Lessons from the Dawn mission to Ceres and Vesta," *Acta Astronautica*, Vol. 176, 2020, pp. 233-237.

[4] O. Barnouin, M. Daly, E. Palmer, C. Johnson, R. Gaskell, M. Al Asad, E. Bierhaus, K. Craft, C. Ernst, R. Espiritu, *et al.*, "Digital terrain mapping by the OSIRIS-REx mission," *Planetary and Space Science*, Vol. 180, 2020, pp. 104764.

[5] M. Yoshikawa, J. Kawaguchi, A. Fujiwara, and A. Tsuchiyama, "The Hayabusa mission," *Sample Return Missions*, pp. 123–146, Radarweg 29, PO Box 211, 1000 AE Amsterdam, Netherlands: Elsevier, 2021.

[6] S.-i. Watanabe, Y. Tsuda, M. Yoshikawa, S. Tanaka, T. Saiki, and S. Nakazawa, "Hayabusa2 mission overview," *Space Science Reviews*, Vol. 208, No. 1, 2017, pp. 3–16.

[7] K.-H. Glassmeier, H. Boehnhardt, D. Koschny, E. Kührt, and I. Richter, "The Rosetta mission: Flying towards the origin of the solar system," *Space Science Reviews*, Vol. 128, No. 1, 2007, pp. 1–21.

[8] A. F. Cheng, A. S. Rivkin, P. Michel, J. Atchison, O. Barnouin, L. Benner, N. L. Chabot, C. Ernst, E. G. Fahnestock, M. Kueppers, *et al.*, "AIDA DART asteroid deflection test: Planetary defense and science objectives," *Planetary and Space Science*, Vol. 157, 2018, pp. 104–115.

[9] A. F. Cheng, H. F. Agrusa, B. W. Barbee, A. J. Meyer, T. L. Farnham, S. D. Raducan, D. C. Richardson, E. Dotto, A. Zinzi, V. Della Corte, *et al.*, "Momentum transfer from the DART mission kinetic impact on asteroid Dimorphos," *Nature*, Vol. 616, No. 7957, 2023, pp. 457–460.

[10] P. Panicucci, *Autonomous vision-based navigation and shape reconstruction of an unknown asteroid during approach phase*. PhD thesis, Toulouse, ISAE, 2021.

[11] F. d. Novaes Kucinskis and M. G. V. Ferreira, "Taking the ECSS autonomy concepts one step further," *SpaceOps 2010 Conference "Delivering on the Dream" Hosted by NASA Mars*, 2010, pp. 25–30.

[12] D. A. Surovik, *Autonomous mission design in extreme orbit environments*. PhD thesis, University of Colorado at Boulder, 2016.

[13] A. Rizza, C. Giordano, and F. Topputo, "A goal-oriented guidance approach for binary asteroids exploration," *Astrodynamics*, Vol. 9, No. 2, 2025, pp. 289–302.

[14] A. Rizza, S. D'Amico, and F. Topputo, "Goal-Oriented Trajectory Refinement for Asteroid Mapping Using Sequential Convex Programming," *Journal of Guidance, Control, and Dynamics*, Vol. 0, No. 0, 0, pp. 1–15.

[15] A. Rizza, C. Giordano, and F. Topputo, "A goal-oriented guidance approach for binary asteroids exploration," *Astrodynamics*, Vol. 9, No. 2, 2025, pp. 289–302.

[16] C. Cocaud and T. Kubota, "Autonomous navigation near asteroids based on visual SLAM," *Proceedings of the 23rd International Symposium on Space Flight Dynamics, Pasadena, California*, 2012.

[17] M. W. Givens and J. W. McMahon, "Square-root extended information filter for visual-inertial odometry for planetary landing," *Journal of Guidance, Control, and Dynamics*, Vol. 46, No. 2, 2023, pp. 231–245.

[18] S. Takahashi and D. J. Scheeres, "Autonomous Exploration of a Small Near-Earth Asteroid," *Journal of Guidance, Control, and Dynamics*, Vol. 44, Apr. 2021, pp. 701–718.

[19] K. Dennison, N. Stacey, and S. D'Amico, "Autonomous asteroid characterization through nanosatellite swarming," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 59, No. 4, 2023, pp. 4604–4624.

[20] S. Julier and J. Uhlmann, "A counter example to the theory of simultaneous localization and map building," *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No.01CH37164)*, Vol. 4, Seoul, South Korea, IEEE, 2001, pp. 4238–4243.

[21] H. Strasdat, J. M. M. Montiel, and A. J. Davison, "Real-time monocular SLAM: Why filter?," *2010 IEEE International Conference on Robotics and Automation*, Anchorage, AK, IEEE, May 2010, pp. 2657–2664.

[22] F. Dellaert and M. Kaess, "Square Root SAM: Simultaneous Localization and Mapping via Square Root Information Smoothing," *The International Journal of Robotics Research*, Vol. 25, Dec. 2006, pp. 1181–1203.

[23] F. Dellaert and M. Kaess, "Factor Graphs for Robot Perception," *Foundations and Trends in Robotics*, Vol. 6, No. 1-2, 2017, pp. 1–139.

[24] R. Kummerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "G^2o: A general framework for graph optimization," *2011 IEEE International Conference on Robotics and Automation*, Shanghai, China, IEEE, May 2011, pp. 3607–3613.

[25] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. J. Leonard, and F. Dellaert, "iSAM2: Incremental smoothing and mapping using the Bayes tree," *The International Journal of Robotics Research*, Vol. 31, Feb. 2012, pp. 216–235.

[26] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age," *IEEE Transactions on Robotics*, Vol. 32, Dec. 2016, pp. 1309–1332.

[27] B. E. Tweddle, A. Saenz-Otero, J. J. Leonard, and D. W. Miller, "Factor graph modeling of rigid-body dynamics for localization, mapping, and parameter estimation of a spinning object in space," *Journal of Field Robotics*, Vol. 32, No. 6, 2015, pp. 897–933.

[28] N. Takeishi and T. Yairi, "Visual monocular localization, mapping, and motion estimation of a rotating small celestial body," *Journal of Robotics and Mechatronics*, Vol. 29, No. 5, 2017, pp. 856–863.
[29] M. Dor, T. Driver, K. Getzandanner, and P. Tsiotras, "AstroSLAM: Autonomous monocular navigation in the vicinity of a celestial small body—Theory and experiments," *The International Journal of Robotics Research*, Vol. 43, No. 11, 2024, pp. 1770–1808.
[30] P. Califano, F. Piccolo, P. Panicucci, F. Topputo, *et al.*, "Enhanced Full Visual Slam for Close Proximity Exploration of Asteroids Combining Incremental Smoothing with Filtering Techniques," *35th AAS/AIAA Space Flight Mechanics Meeting*, 2025, pp. 1–22.
[31] C. Matarrese, P. Califano, F. Piccolo, and P. Panicucci, "Maneuver-Based Scale Estimation in Visual SLAM for Autonomous Asteroid Exploration," *48th AAS Guidance, Navigation and Control Conference*, 2026, pp. 1–20.
[32] D. A. Vallado, *Fundamentals of astrodynamics and applications*. Hawthorne, CA: Microcosm Press Year, 2001, pp. 538-550.
[33] F. Spada, M. Sagliano, and F. Topputo, "Direct–indirect hybrid strategy for optimal powered descent and landing," *Journal of Spacecraft and Rockets*, Vol. 60, No. 6, 2023, pp. 1787–1804.
[34] A. Mannocchi, C. Giordano, and F. Topputo, "Control-Constrained Indirect Optimization for Low-Thrust Trajectories with Duty Cycles," *Journal of Guidance, Control, and Dynamics*, 2026, pp. 1–19.
[35] S. P. Boyd and L. Vandenberghe, *Convex optimization*. The Edinburgh Building, Cambridge, CB2 8RU, UK: Cambridge University Press, 2004, pp. 397.
[36] Y. Mao, M. Szmuk, and B. Açıkmeşe, "Successive convexification of non-convex optimal control problems and its convergence properties," *2016 IEEE 55th Conference on Decision and Control (CDC)*, IEEE, 2016, pp. 3636–3641.
[37] Y. Mao, D. Dueri, M. Szmuk, and B. Açıkmeşe, "Successive convexification of non-convex optimal control problems with state constraints," *Ifac-PapersOnline*, Vol. 50, No. 1, 2017, pp. 4063–4069.
[38] X. Liu and P. Lu, "Solving nonconvex optimal control problems by convex optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 37, No. 3, 2014, pp. 750–765.
[39] A. Rizza, *Goal-oriented guidance under uncertainties for small bodies exploration*. PhD thesis, Politecnico di Milano in Milan, 2024.
[40] A. Rizza, D. Simone, and T. Francesco, "Goal-oriented guidance under uncertainties for small bodies proximity operations," *35th AAS/AIAA Space Flight Mechanics Meeting*, 2025.
[41] A. Beshaj, A. Rizza, C. Giordano, F. Topputo, *et al.*, "Robust guidance for surface characterization of minor bodies," *76th International Astronautical Congress (IAC 2025)*, 2025, pp. 1–17.
[42] M. F. Ahmed, K. Masood, V. Fremont, and I. Fantoni, "Active slam: A review on last decade," *Sensors*, Vol. 23, No. 19, 2023, p. 8097.
[43] H. J. S. Feder, J. J. Leonard, and C. M. Smith, "Adaptive mobile robot navigation and mapping," *The International Journal of Robotics Research*, Vol. 18, No. 7, 1999, pp. 650–668.
[44] L. Carlone, J. Du, M. Kaouk Ng, B. Bona, and M. Indri, "Active SLAM and exploration with particle filters using Kullback-Leibler divergence," *Journal of Intelligent & Robotic Systems*, Vol. 75, No. 2, 2014, pp. 291–311.
[45] V. Pesce, A.-a. Agha-mohammadi, and M. Lavagna, "Autonomous navigation & mapping of small bodies," *2018 IEEE aerospace conference*, IEEE, 2018, pp. 1–10.
[46] D. M. Chan and A.-a. Agha-mohammadi, "Autonomous imaging and mapping of small bodies using deep reinforcement learning," *2019 IEEE aerospace conference*, IEEE, 2019, pp. 1–12.
[47] D. Nakath, J. Clemens, and C. Rachuy, "Active asteroid-slam: Active graph slam with landing site discovery in a deep space proximity operations scenario," *Journal of Intelligent & Robotic Systems*, Vol. 99, No. 2, 2020, pp. 303–333.
[48] L. Ticozzi and P. Tsiotras, "Factor Graph-Based Active SLAM for Spacecraft Proximity Operations," *arXiv preprint arXiv:2501.10950*, 2025.
[49] D. A. Surovik and D. J. Scheeres, "Adaptive Reachability Analysis to Achieve Mission Objectives in Strongly Non-Keplerian Systems," *Journal of Guidance, Control, and Dynamics*, Vol. 38, Mar. 2015, pp. 468–477.
[50] F. Piccolo, C. Balossi, P. Panicucci, M. Pugliatti, F. Topputo, and F. Capolupo, "Resource-Constrained Vision-Based Relative Navigation About Small Bodies," *46th AAS Guidance, Navigation and Control Conference*, 2024, pp. 1–18.
[51] F. Piccolo, M. Pugliatti, J. W. McMahon, and F. Topputo, "Autonomous Vision-Based Navigation at Small Bodies Combining Centroiding and Visual Odometry," *Journal of Spacecraft and Rockets*, 2025, pp. 1–19.
[52] S. Baker and I. Matthews, "Lucas-Kanade 20 Years On: A Unifying Framework," *International Journal of Computer Vision*, Vol. 56, Feb. 2004, pp. 221–255.
[53] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision, Second Edition*. Cambridge University Press, 2004.
[54] D. Woodbury and J. Junkins, "On the Consider Kalman Filter," *AIAA Guidance, Navigation, and Control Conference*, Toronto, Ontario, Canada, American Institute of Aeronautics and Astronautics, Aug. 2010.
[55] J. R. Carpenter and C. N. D'Souza, "Navigation Filter Best Practices," 2018.
[56] R. Zanetti and R. H. Bishop, "Kalman filters with uncompensated biases," *Journal of Guidance, Control, and Dynamics*, Vol. 35, No. 1, 2012, pp. 327–335.
[57] Á. González, "Measurement of areas on a sphere using Fibonacci and latitude–longitude lattices," *Mathematical geosciences*, Vol. 42, No. 1, 2010, pp. 49–64.
[58] R. Gaskell, J. Saito, M. Ishiguro, T. Kubota, T. Hashimoto, N. Hirata, S. Abe, O. Barnouin-Jha, and D. Scheeres, "Gaskell itokawa shape model v1. 0," *NASA Planetary Data System*, Vol. 92, 2008, pp. HAY–A.
[59] R. A. Werner, "Spherical harmonic coefficients for the potential of a constant-density polyhedron," *Computers & Geosciences*, Vol. 23, No. 10, 1997, pp. 1071–1077.
[60] M. Pugliatti, C. Buonagura, and F. Topputo, "Corto: The celestial object rendering tool at dart lab," *Sensors*, Vol. 23, No. 23, 2023, p. 9595.