ORIGINAL ARTICLE

Analysis, Design, and Optimization of Robust Trajectories in Cislunar Environment for Limited-Capability Spacecraft

Carmine Giordano[1] · Francesco Topputo[1]

Abstract

Nowadays, the space exploration is going in the direction of exploiting small platforms to get high scientific return at significantly lower costs. However, miniaturized spacecraft pose different challenges both from the technological and mission analysis point of view. While the former is in constant evolution due to the manufacturers, the latter is an open point, since it is still based on a traditional approach, not able to cope with the new platforms' peculiarities. In this work, a revised preliminary mission analysis approach, merging the nominal trajectory optimization with a complete navigation assessment, is formulated in a general form and three main blocks composing it are identified. Then, the integrated approach is specialized for a cislunar test case scenario, represented by the transfer trajectory from a low lunar orbit to an halo orbit of the CubeSat LUMIO, and each block is modeled with mathematical means. Eventually, optimal solutions, minimizing the total costs, are sought, showing the benefits of an integrated approach.

Keywords Mission analysis · Trajectory optimization · CubeSat · Robust trajectories

1 Introduction

Since the beginning of the space era, spacecraft have always been equipped with chemical propulsion engines, characterized by a high value of thrust and a good control authority. For traditional spacecraft, nominal trajectories are designed and optimized in order to satisfy only scientific requirements as well as to comply with system constraints. Although, the nominal path will unlikely be followed by the spacecraft in real-life scenarios due to uncertainty in *dynamic model* (e.g.,

✉ Carmine Giordano
   carmine.giordano@polimi.it

   Francesco Topputo
   francesco.topputo@polimi.it

1 Department of Aerospace Science and Technology, Politecnico di Milano, via La Masa, 34, Milan 20156, Italy

🖄 Springer

gravitational parameters or radiation pressure noisy profiles), *navigation* (i.e. imperfect state knowledge or approximations in measurement model), and *command actuation* (i.e., thrust magnitude and pointing angles error) [1], the correction maneuvers needed to compensate deviations are considered to be a minor problem, since changing the trajectory is relatively easy with a single, short burn. Robustness and feasibility assessment of the nominal trajectory against uncertainty are performed a-posteriori through a navigation analysis, with the aim to perform a covariance analysis and compute the achievable state knowledge, and to estimate the correction maneuvers. Thus, the nominal trajectory and the uncertainty assessment are decoupled and their analysis and optimization are done in two separate phases. This approach can lead to sub-optimal solutions. For large spacecraft, this procedure is acceptable since they can produce high thrust levels and they can store relevant propellant quantities; hence, sub-optimal trajectories are not critical.

However, in recent times, the space exploration is going in the direction of exploiting small platforms, such as SmallSat or CubeSat, characterized by: (1) limited-control authority (due to low thrust levels and reduced propellant budget), (2) large uncertainties in the state knowledge (due to novel techniques in navigation [2] or limited access to on-ground facilities), (3) and large errors in command actuation (due to low-maturity components), in order to get scientific and technological return at significantly lower costs [3, 4]. In this kind of probes, the low control authority poses challenges in maneuvering, since a long burn is needed, even for paltry deviations. Therefore, orbit determination and the subsequent correction maneuvers cannot be considered a minor problem and preliminary trajectory design should take them into account, since the classical approach can lead to trajectories requiring an unnecessarily large amount of propellant.

A clear indication of this phenomenon can already be found in some studies about mission having the characteristics summarized above. Lisa PathFinder (LPF) proposed mission extension is a notable example. LPF was a technology demonstrator for the gravitational wave observatory LISA, launched by ESA in 2015. A number of works [5–7] studied the possibility of extending LPF beyond its nominal mission, maneuvering the spacecraft in order to pass through a peculiar point of the solar system, the Earth–Sun Saddle Point, where the net gravitational acceleration is almost null, to collect data for a possible confirmation of the MoND.[1] Although the on-board instruments allowed detecting anomalous MoND gradients nearby the Saddle Point, ESA chose not to go for this option due to high risks, and thus the disposal of LPF was executed in April 2017. At the end of the nominal mission, LPF had a small residual control capacity, estimated into a Δv budget of approximately 1 m/s, that could be provided using cold-gas thrusters with a maximum thrust of 100 μN. Thus, LPF was a very limited control authority spacecraft in a highly unstable environment, and applying velocity changes could be very challenging. Several nominal solutions were found that satisfy the propulsion constraints [8]. However, if the stochastic cost is taken into account, the Δv required to accomplish the mission

---

[1] MoND is a theory, alternative to the theory of dark matter, proposing a modification of the classical Newton's law to account for the observed motion of the galaxies.

increases in a sensible way and the feasibility of the mission extension is no longer guaranteed [5]. For example, the best solution in Paretian sense misses the target within a distance of 10 km and requires a deterministic $\Delta v = 0.657$ m/s. However, the navigation Δv distribution for the given trajectory gives a value of 4.533 m/s for a 95% confidence level in order to counteract errors in model, navigation, and control. This figure is one order of magnitude higher than the deterministic cost, endangering the mission feasibility. A similar behavior can be found also in the LUMIO Phase 0 study [9]. For this mission, a transfer from a Low Lunar Orbit to a halo orbit about Earth–Moon L_2 is foreseen. In this case, the deterministic cost for the transfer amounts to 89.97 m/s, while the 3σ stochastic cost sums up to 97.9 m/s. Hence, the nominal and navigation Δvs have the same order of magnitude. In such cases, a procedure embedding uncertainty in the preliminary mission design can be useful in cutting down the overall mission costs and produce more robust and feasible solutions.

In the last decades, optimal control and optimization theory have been extensively exploited for the nominal design of space trajectories [10, 11]. However, only in the last ten years, some stochastic-optimal approaches, embedding uncertainty in their core, have been developed for diverse problems. In the early 2000, [12] proposed a statistical targeting algorithm, able to incorporate statistical information directly in the trajectory design. While the usual target method solves a deterministic boundary value problem for the nominal trajectory, this algorithm search for a statistically correct trajectory, i.e., a trajectory able to reach the target state in a stochastic sense. However this approach fails whenever the stochastic trajectories envelope cannot be described as a quasi-Gaussian distribution.

The uncertain Lambert's problem has been investigated alike, in several papers by [13–15] by exploiting Taylor differential algebra. An alternative approach, characterizing the stochastic error by means of the first-order variational equations, is presented in [16]. This approach has been extended considering first the explicit partial derivatives of the transfer velocities [17] and later by implementing a derivative free numerical method, exploiting novelties in uncertainty quantification [18]. Uncertain Lambert's problem with differential algebra was also exploited in the gravity assist space pruning algorithm presented in [19].

Similarly, approaches to tackle the rendezvous problem were conceived. A multi-objective optimization method, considering a robust performance index based on final uncertainties, was devised for the linear rendezvous problem [20], taking into account both navigation and control errors. A relation among the performance index, the rendezvous time, and the propellant cost was found for short-duration missions. Nonlinear rendezvous model and the possibility of handling long-duration phases were later addressed [21]. Also the asteroid rendezvous in a stochastic sense was investigated, considering the state uncertainty both of the spacecraft and the target [22] together with the optimization of correction maneuver under the Lambert's problem conditions.

Recently, general procedures of trajectory optimization under uncertainty were developed. A method transcribing the stochastic trajectory optimization into a deterministic problem by means of Polynomial Chaos Expansion and an adaptive pseudospectral collocation method was introduced by [23], while [24]

presented a novel approach, based on Belief Markov Decision Process model and then applied this method to the robust optimization of a flyby trajectory of Europa Clipper mission in a scenario characterized by knowledge, execution and observation errors. Stochastic Differential Dynamic Programming has been investigated to design robust Earth–Mars transfer, considering unscented transform to propagate uncertainties [25]. This methodology has been extended subsequently using a hybrid multiple-shooting technique to overcome the limitation of the Differential Dynamic Programming [26]. More recently, the use of a combination of convex optimization and covariance steering have been introduced to solve different classes of robust continuous control problems in astrodynamics, such as interplanetary transfer [27] and reentry [28]. Finally, Reinforcement Learning has been proven to be effective in solving robust deep-space transfers considering different sources of error [29, 30].

The idea of considering both deterministic and stochastic propellant cost has been employed also in the nominal trajectory design for EQUULEUS [31, 32], a 6U CubeSat developed by the University of Tokyo and JAXA and planned to be inserted in a cis-lunar environment by the Artemis 1 mission by NASA and then brought to an halo orbit about the Earth–Moon L_2 point. However, in this last case, only the transfer cost and the annual station keeping cost are optimized without considering any navigation cost during the transfer phase.

Although uncertainties in the early stages of the trajectory design are considered in recent works to devise robust optimal trajectories, an integrated approach, considering the navigation assessment as part of the trajectory design and optimization, using classical techniques, is still missing. Nevertheless, the paradigm shift proposed in this work can be beneficial in terms of propellant mass consumption. Indeed, it can overtake the natural sub-optimality of the traditional approach by surfing solutions with lower dispersion and better stochastic properties, thus reducing both the navigation costs and the final state scattering with respect to the target. Hence, robust low-cost trajectories in the preliminary mission analysis can increase the scientific return for limited-capability satellite either by giving access to nowadays-impossible mission profiles or by expanding the nominal operative life. On the other hand, even large traditional probes can benefit from an holistic approach, since it can increase the spacecraft performances while reducing the design steps.

This paper is organized as follows. In Sect. 2, the general problem statement of the preliminary mission analysis is presented. Alongside the traditional approach, the revised approach, that embeds deterministic and stochastic considerations in a single step, is introduced. Then, a test case scenario, based on the transfer from a low lunar orbit to an halo orbit, is presented (Sect. 3). The mathematical formulation of the relevant building blocks is shown in Sect. 4. In this section, a novel technique, exploiting the conjugated unscented transformation to solve the stochastic integral associated to the non-intrusive polynomial chaos expansion, is devised in order to efficiently propagate the trajectory under uncertainties. The combination of this technique with kernel estimators to evaluate state statistics and with an ad hoc implementation of the ensemble square root filter to fast determine the spacecraft state knowledge is firstly formulated in the section remainder. In Sect. 2, the revised

approach is adapted to the test case scenario. Finally, results are given and a critical assessment is provided in Sect. 6.

2 Problem Statement

2.1 Sequential Approach

In this work, the approach followed nowadays to compute a nominal trajectory, evaluate its statistical properties and retrieve the navigation costs is labeled as *sequential* or *traditional approach*. Detailed information about this process can be found in several sources [5, 9]. In this case, the whole procedure is subdivided into two *sequential and independent steps* (Fig. 1):

1. *Trajectory Design and Optimization*: nominal trajectory, connecting the initial point to the target, while minimizing the propellant mass, is sought (Fig. 1a). Thus, generally speaking, an optimal control problem is set up, having the aim to determine the state $\mathbf{x}(t)$, the control $\mathbf{u}(t)$ and, possibly, the initial and final times, t_0 and t_f, that minimize the total control effort

$$J = \int_{t_0}^{t_f} \|\mathbf{u}\| \, dt$$

   subject to the ordinary differential equation

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t)$$

   and to the boundary constraints

$$\mathbf{x}(t_0) = \mathbf{x}_0$$
$$\mathbf{x}(t_f) = \mathbf{x}_f$$

   The function $\mathbf{f}$ represents the acceleration vector field associated to the spacecraft dynamics. Some additional terminal and path constraints are normally added, considering the characteristics of the specific orbital problem. Several techniques can be exploited to solve the optimization problem. Classical approaches are subdivided in two classes: direct methods and indirect methods [33], and the choice of the most suitable method is based mainly on the mission profile, the spacecraft characteristics, and the desired accuracy. Usually this step is time- and effort-consuming, due to the large search space. For this reason, a preliminary trade-off and/or pruning can be required in order to relieve the total burden.
2. *Navigation Assessment*: the nominal trajectory feasibility in a real scenario is evaluated by simulating the orbit determination (OD) process and estimating the trajectory correction maneuvers (TCM) along the whole mission. Thus, the Navigation Assessment can be split into two (independent) sub-phases:
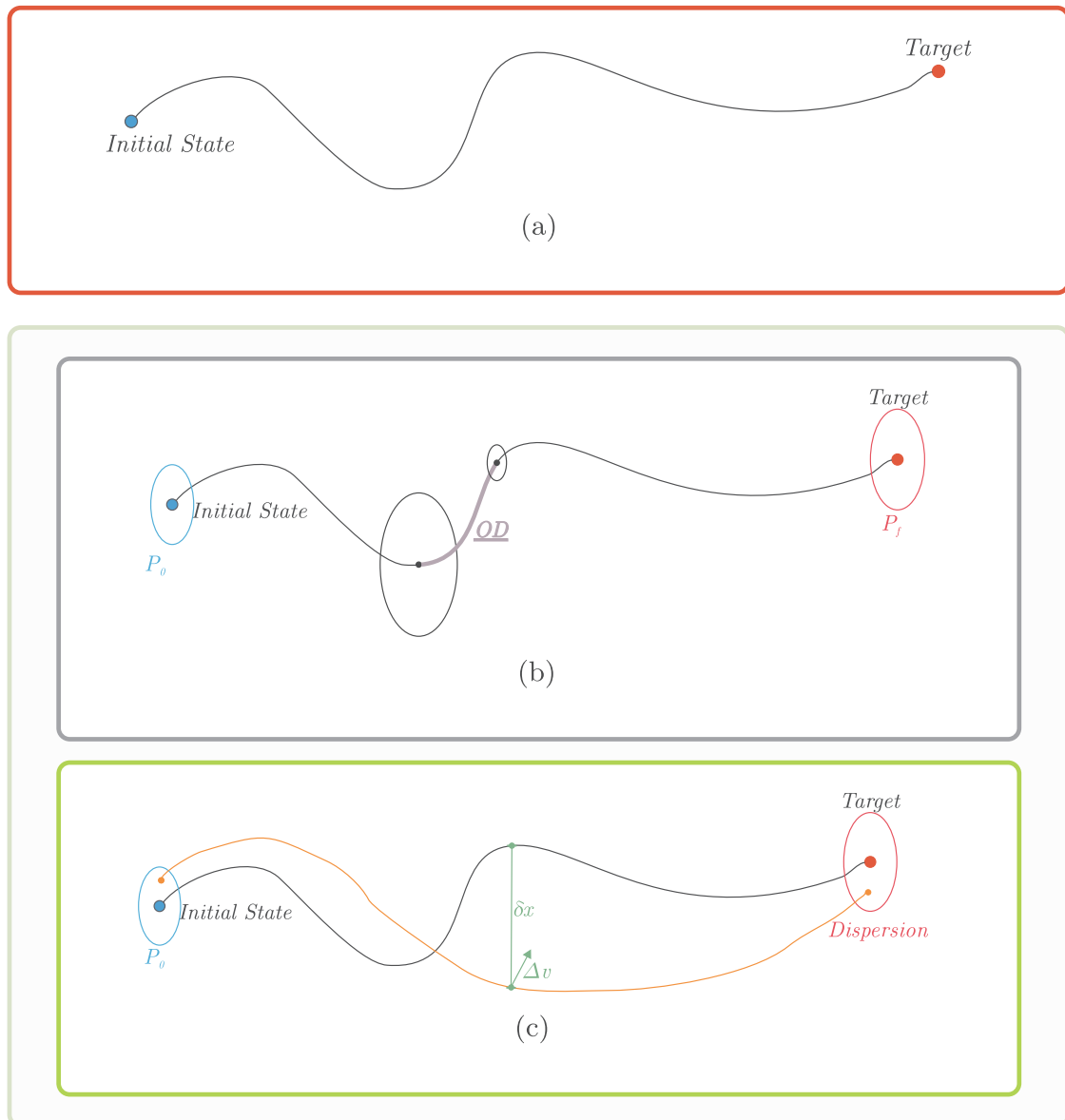
Fig. 1 Traditional approach for preliminary mission analysis: **a** Trajectory design and optimization; **b** Knowledge analysis; **c** Navigation cost evaluation. Nominal trajectory is indicated with a black line, true trajectory as an orange line, OD with a grey thick line. Ellipses represent the instantaneous **b** knowledge **c** or dispersion. Steps **b–c** form the navigation assessment (Color figure online)

i. *Knowledge analysis*: a covariance analysis is performed to estimate the achievable level of accuracy in the spacecraft state *knowledge*, i.e., the deviation of the estimated spacecraft state with respect to the true one (Fig. 1b), along the entire trajectory. The initial knowledge, usually represented as a Gaussian distribution with a given covariance P_0 centered in the initial nominal state $\mathbf{x}_0$, is propagated forward in time. Due to its nature, the knowledge covariance increases in time [12]. Indeed, uncertainties affect the knowledge by enlarging the possible state space. In order to improve the accuracy of the estimated state with respect to the real one, an OD process is implemented: the knowledge covariance is
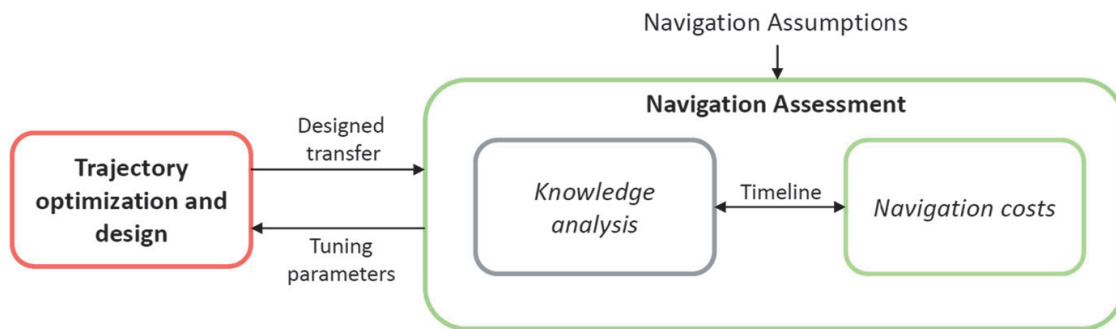
Fig. 2 Traditional approach architecture

reduced by performing multiple indirect measurements of the state in a prescribed time interval, and they are provided to a filtering algorithm, able to reconstruct spacecraft's position and velocity. Thus, the knowledge covariance increases during propagation and it can be reduced only during the OD phase.

ii. *Navigation cost estimation*: a stochastic analysis is performed in order to estimate the navigation cost needed to allow the spacecraft to reach the target (Fig. 1c). First, a guidance cycle is defined. The guidance cycle refers to the epochs at which correction maneuvers are performed. Typically, this can change from one mission to another, or from one phase to another inside the same mission. Usually, a guidance cycle with a correction maneuver once a week is assumed as baseline strategy in order to ease on-ground operations. Indeed, currently navigation maneuvers are computed on-ground and then sent to the spacecraft. For this reason, having a guidance cycle following the working week pattern reduces operational complexity and costs. Then, a guidance law is selected in order to compute the correcting impulse Δv, starting from the deviation from the nominal trajectory δx. At the end, a statistical analysis is performed to give a measure of the needed navigation propellant. Moreover, the trajectory *dispersion*, i.e., the deviation of the true spacecraft state with respect to the nominal one, can be retrieved.

Knowledge analysis and navigation cost estimation are usually performed independently in the preliminary mission analysis. However, in principle, they cannot be considered totally separate: both sub-phases should share a common timeline and a minimum time interval, the *cut-off time*, should be considered between the end of the OD phase and the subsequent correction maneuver. This cut-off time is needed to the flight dynamics team to complete the orbit acquisition process, to compute the correction maneuver and to generate the commands.

Figure 2 shows the general architecture for the traditional approach. This two-step approach can lead to sub-optimal solutions, requiring a gratuitous amount of propellant. This behavior is taken to extremes when small satellites are
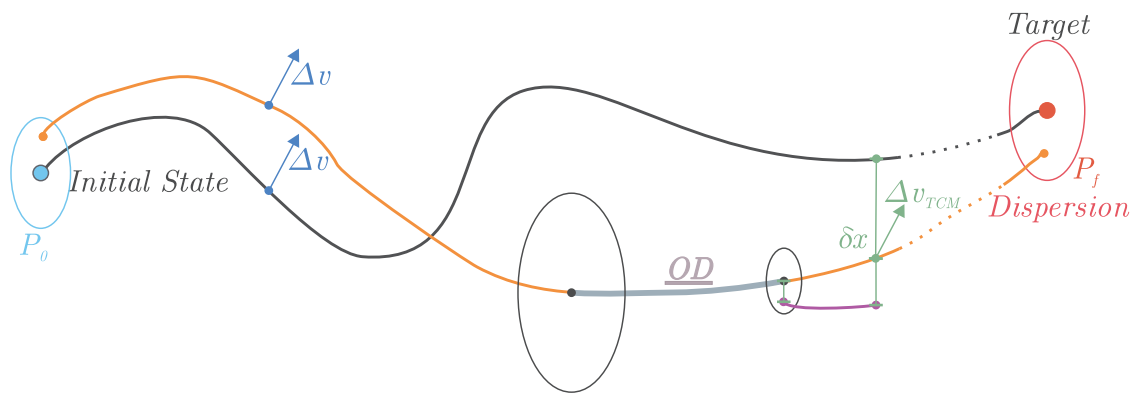
Fig. 3 Revised approach for the preliminary mission analysis. Nominal trajectory is indicated with a black line, a true possible trajectory with an orange line, estimated trajectory with a purple line. The OD process is the gray thick line. Black ellipses represent the instantaneous knowledge; colored ellipses represent the dispersion (Color figure online)

considered, due to maneuver complexity and definite propellant mass. As a matter of fact, some trajectory can be wrongly tagged as infeasible using this approach. An integrated approach is needed to relieve this effect.

2.2 Integrated Approach

A procedure able to comprehend the whole navigation assessment inside the optimization process has to be designed. This method will be tagged as *integrated* or *revised approach*. Its aims are to (1) evaluate and minimize deterministic and stochastic cost, (2) estimate the knowledge, (3)and compute the dispersion, at the same time. In order to achieve these objectives, the approach depicted in Fig. 3 is devised. The initial nominal state is given together with the associated initial dispersion. For each state belonging to the initial dispersion, an initial knowledge is considered. These three quantities (nominal state, knowledge and dispersion) are propagated forward. At some prescribed times, an OD process is performed in order to estimate the true trajectory and reduce the knowledge covariance. The estimated trajectory is then used to feed a guidance scheme, compute the correction maneuver and reduce the dispersion. At the end, the final nominal state and the final dispersion can be retrieved. For sake of simplicity, considering a Monte Carlo fashion, the revised approach can be summarized as:

For each step of the optimization algorithm:

1. An initial nominal state $\mathbf{x}_0$ (blue dot in Fig. 3) and initial dispersion (blue ellipse) are given;
2. The initial state is propagated up to the final time, in order to generate the nominal trajectory (black line) and compute the deterministic cost (i.e., the impulses on the nominal trajectory that do not depend on the uncertainties, indicated as blue arrows);

3. A number of samples in the initial dispersion $\mathbf{x}_0^i$ (orange dot) are generated, representing the initial state of the possible true trajectories;
4. For each sample:

   (a) The initial state $\mathbf{x}_0^i$ and the associated initial knowledge are propagated forward (orange line) up to the first OD time;
   (b) In a give time span $t \in \left[t_0^{OD}, t_f^{OD} \right]$, the OD is performed (gray thick line) to improve the knowledge (black ellipses);
   (c) An estimated state of the true trajectory (purple dot) is retrieved at the end of the OD and pushed forward in time (purple line), in order to compute the TCM (green arrow) through a guidance law;
   (d) The real trajectory is propagated up to the correction maneuver time t_{TCM}, when the navigation impulse is applied;
   (e) Steps 4a–4d are repeated for each OD and correction maneuver time up to the final time t_f.

5. From the Monte Carlo-like simulation, statistics for the navigation cost can be computed (i.e., the maneuvers needed to control the dispersion) and the final dispersion (red ellipse) can be estimated.
6. The total propellant mass, given by deterministic plus stochastic costs is optimized, while imposing a constraint on the final state.

Even though Fig. 3 shows only a nominal maneuver and only an OD window, it can be easily extended to consider multiple deterministic impulses or to a different concept of operations.

Moreover, this general approach can be modified by means of some simplifying assumptions to reduce the computational burden, if needed by the given mission scenario, e.g., by performing the knowledge analysis only on the nominal trajectory and use its results on the real trajectories. Additionally, it is important to stress a significant difference of this concept with respect to the traditional approach. In fact, the final state is no more deterministic, but it can be more coherently represented in a stochastic way by evaluating the dispersion at the final time. Hence, it is convenient to implement the final constraint as a stochastic constraint, i.e., the final points distribution should be relatively close to the target point.

In conclusion, the general fuel-optimal problem of a spacecraft flying in a perturbed environment under the revised approach can be formalized as:

Problem 1 (*Fuel-Optimal General Problem*) Find the nominal state $\mathbf{x}^*(t)$, the nominal control history $\mathbf{u}^*(t)$ and, possibly, the initial and final times, t_0 and t_f, such that

$$J = \int_{t_0}^{t_f} \|\mathbf{u}^*\| \, dt + Q(\Delta v^s) \tag{1}$$

with $Q(\Delta v^s)$ a measure of the stochastic cost, is minimized, while the state is sub-jected to a simplified Itô stochastic differential equation [34]

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t) + \boldsymbol{\omega}(\mathbf{x}, \mathbf{u}, t) \tag{2}$$

with $\mathbf{f}$ being the deterministic part of the dynamics and $\boldsymbol{\omega}$ the process noise associ-ated to uncertainty in dynamics and in maneuver execution.

Moreover, the state is subjected to initial constraints on dispersion

$$\begin{cases} E\left[\mathbf{x}^*(t_0)\right] = \mathbf{x}_0 \\ E\left[\left(\mathbf{x}^*(t_0) - \mathbf{x}_0\right)\left(\mathbf{x}^*(t_0) - \mathbf{x}_0\right)^T\right] = P_0^d \end{cases} \tag{3}$$

and on knowledge

$$E\left[\left(\mathbf{x}(t_0) - \mathbf{x}_0\right)\left(\mathbf{x}(t_0) - \mathbf{x}_0\right)^T\right] = P_0^k \tag{4}$$

and a final constraint

$$\mathcal{E}\left(\mathbf{x}(t_f), t_f\right) \subseteq \hat{\mathcal{E}}_\delta(t_f) \tag{5}$$

with $\mathcal{E}$ indicating a generalized uncertainty ellipsoid and $\hat{\mathcal{E}}_\delta$ the desired ellipsoid.

The navigation costs are estimated through a guidance law, fed by the OD scheme. It means

$$\Delta v^s = \texttt{GL}\left(\mathbf{x}^*, \hat{\mathbf{x}}, t_{TCM}\right) \tag{6}$$

and

$$\hat{\mathbf{x}}\left(t_f^{OD}\right) = \texttt{OD}\left(\mathbf{x}, \hat{\mathbf{x}}, t_0^{OD}, t_f^{OD}\right) \tag{7}$$

with $\texttt{GL}$ and $\texttt{OD}$ being the guidance law and orbit determination procedures respec-tively, $\hat{\mathbf{x}}$ is the estimated state, $\mathbf{x}$ the real state and $\mathbf{x}^*$ is the nominal state.

Generally speaking, for the integrated approach, three main building blocks can be identified and they are: (1) a procedure to propagate uncertainty and to evaluate the stochastic measures, (2) a *OD scheme*, and (3) a *guidance law* that can vary and should be selected properly, depending on the analyzed scenario.

3 Test Case Scenario

A comprehensive method for robust stochastic mission analysis seems to be unfeasible: deep-space exploration missions have diverse characteristics and mission profiles vary so widely that a single technique will be never able to pro-duce a good solution for each situation. In fact, Problem 1 provides a general
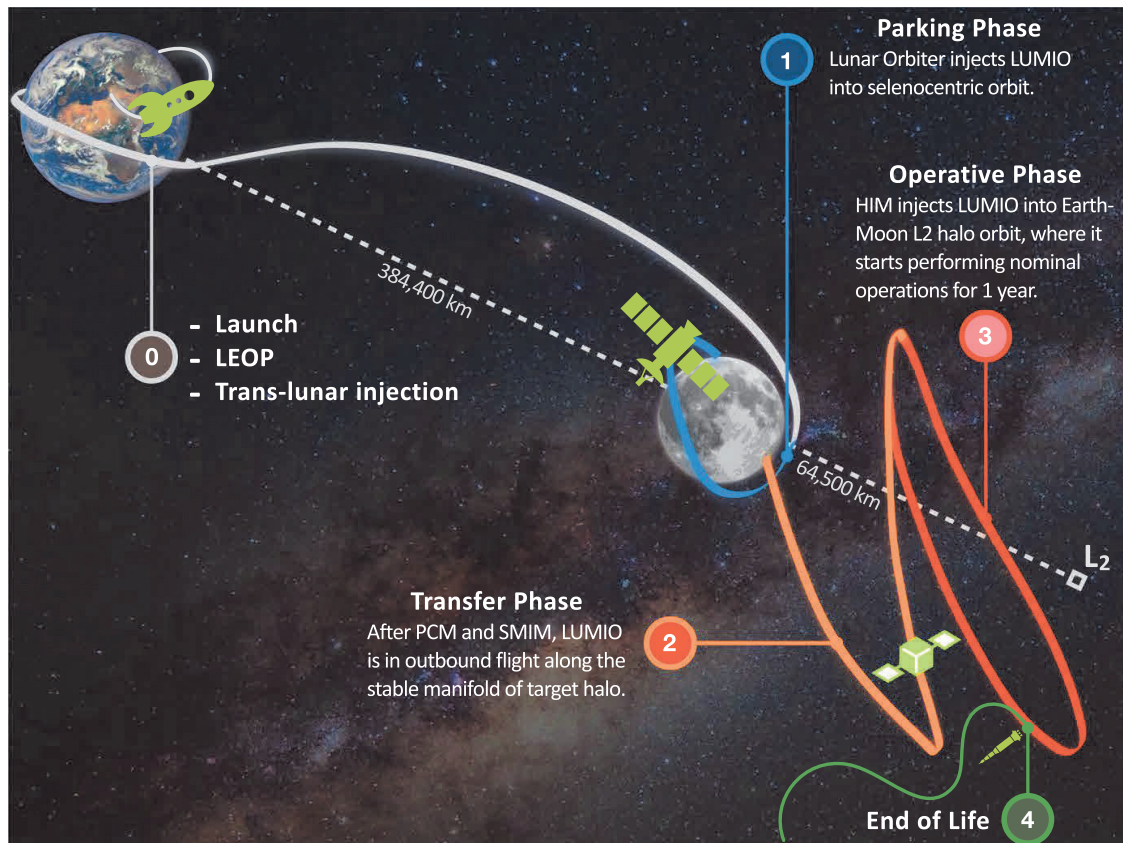
Parking Phase
Lunar Orbiter injects LUMIO into selenocentric orbit.

Operative Phase
HIM injects LUMIO into Earth-Moon L2 halo orbit, where it starts performing nominal operations for 1 year.

- **Launch**
- **LEOP**
- **Trans-lunar injection**

384,400 km

64,500 km

L₂

Transfer Phase
After PCM and SMIM, LUMIO is in outbound flight along the stable manifold of target halo.

End of Life

Fig. 4 LUMIO mission profile (from [9])

framework with its building blocks, that should be adapted to the scenario of interest. In this work, the LUMIO transfer phase, from a low lunar orbit (LLO) to a halo orbit, is considered. As already presented in Sect. 1, this scenario provides a relevant environment to test the revised approach and assess its performances. Moreover, there is a increasing interest in the community about spacecraft flying towards Lagrange point orbits. Numerous space missions will be launched in the time frame 2021-2025, using both traditional spacecraft (e.g., James Webb Space Telescope by NASA [35] or ESA's Euclid [36]) and CubeSats (e.g., EQUULEUS by JAXA [31] or LUMIO [9]). The Lunar Gateway, a small station in lunar environment, will orbit a near-rectilinear halo orbit as well and, of course, several servicing missions are planned to exploit similar transfers [37].

The Lunar Meteoroid Impact Observer (LUMIO) [38] was one of the proposals submitted to the ESA's SysNova Competition *LUnar CubeSats for Exploration (LUCE)*. LUMIO was selected as one of the four concurrent studies run under ESA contract, and it won ex aequo the challenge. In 2020, ESA considered the mission for further implementation. After a successful Preliminary Requirements Review (PRR), LUMIO entered the Phase B in September 2022 (Fig. 4).

LUMIO space segment is composed by a 12U form-factor CubeSat, having the aim to observe, quantify, and characterize meteoroid impacts on the lunar farside by detecting their flashes, to provide global information on the lunar meteoroid environment and contribute to lunar situational awareness. LUMIO will be
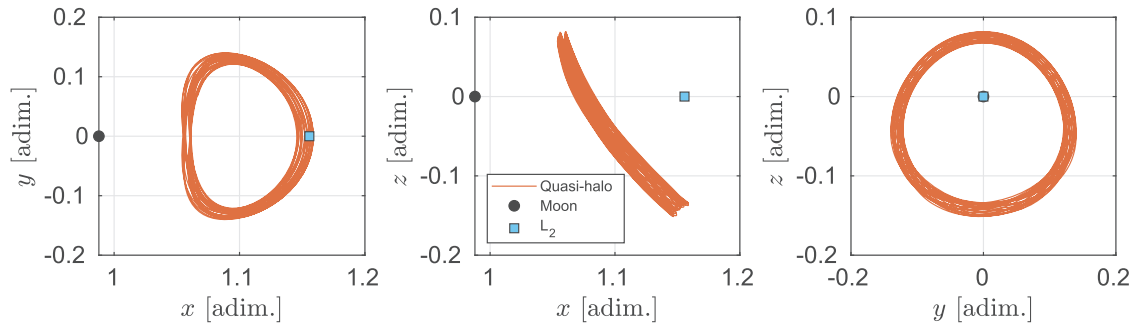
Fig. 5 Projection of the selected operative Earth–Moon L_2 quasi-halo in the Roto-Pulsating Frame for LUMIO
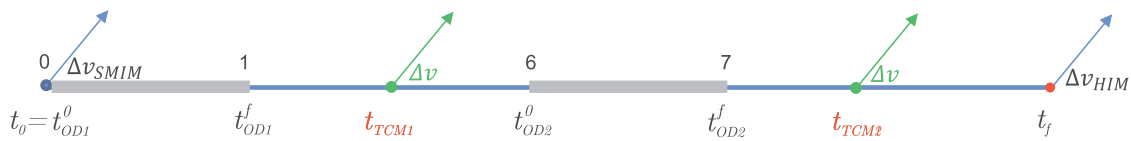

Fig. 6 LUMIO transfer trajectory timeline. The grey bars represents the OD phases, while the green arrows mark the TCMs points. Times, indicated above the timeline, are in days after the SMIM (Color figure online)

released by the carrier on a 600 km×20000 km low lunar orbit with free angular parameters, and must reach the its designated operative orbit, that is a quasi-periodic halo orbit about Earth–Moon L_2 characterized by a Jacobi constant $C_j = 3.09$ [9], depicted in Fig. 5.

The transfer phase, where LUMIO is brought from the low lunar orbit to the operative orbit, will serve as the main topic of this work and will be analyzed in details in the section remainder.

The whole mission profile is summarized in Fig. 4.

During the transfer phase, free transport mechanisms are leveraged to reach the target halo. Specifically, intersection in the configuration space has to be sought between the halo stable manifolds and a selenocentric transition orbit. Since the sought intersection occurs only position-wise, a maneuver is necessary for orbital continuity. This maneuver places the spacecraft on the stable manifold of the target halo and is thus called stable manifold injection maneuver (SMIM) and it will be indicated with Δv_{SMIM}. After the transfer, the halo injection maneuver (HIM), Δv_{HIM}, eventually injects the CubeSat into the final operative orbit. A detailed study of the TCM problem for several LPOs, exploiting simple dynamical systems concepts, has shown that two TCMs provide sufficient degrees of freedom [39]. Thus, two TCMs are scheduled to occur during the transfer along the stable manifold in order to compensate trajectory deviations related to control and dynamics uncertainties. In order to correctly estimate their magnitude an OD phase is foreseen before each TCM, with the first allocated just after the SMIM and the second one scheduled to start after 6 days. Nominally, the first maneuver has to occur at least two days after the SMIM, while the second 8 days after Δv_{SMIM}. The maneuver time is selected in order to give enough time at the ground segment to perform orbit determination, compute correction maneuvers and send

commands to the spacecraft. Indeed, at least one day for the OD and one day cut-off time between the end of the OD phase and the application of the TCM should be considered in order to be compliant with ESOC guidelines. A timeline for the transfer phase is given in Fig. 6. Usually, the nominal trajectory does not have impulses when the correction maneuvers are applied. However, a non-null maneuver can be foreseen at each TCM time in order to broaden the feasible transfer trajectories set.

In conclusion, LUMIO transfer phase, as presented in Fig. 6, can be subdivided into three sub-phases:

1. *OD phase* (between t_{ODi}^0 and t_{ODi}^f): during this phase, a visibility window is identified (see Sect. 4.2), and the OD algorithm is exploited within it. Nominally, OD phases are placed between days 0 and 1 and between days 6 and 7 after t_0;
2. *Cut-off phase* (between t_{ODi}^f and t_{TCMi})): in this phase the Differential Guidance (Sect. 4.3) is exploited to compute the correction maneuver, which is applied at the end of the phase;
3. *Ballistic phase* (between t_{TCM1} and t_{OD2}^0, and between t_{TCM2} and t_f): in this phase, the spacecraft undergoes a ballistic flight.

3.1 Dynamics

The motion of the CubeSat in the transfer phase can be described by using the roto-pulsating restricted n-body problem (RPRnBP) [8], in order to have a high-fidelity dynamics, able to correctly represent the highly non-linear trajectory of LUMIO. The use of an adimensional roto-pulsating frame (RPF) eases the motion description both for the transfer trajectory and for the operative orbit, since they are the generalization of trajectory existing in the restricted 3-body problem.

Thus, a non-uniformly rotating, barycentric, adimensional reference frame ($\hat{\xi}$, $\hat{\eta}$, $\hat{h}$), called *synodic frame*, is defined in order to write the equation of motion. The center of this system is placed at the primaries barycenter (i.e. Earth–Moon barycenter); the $\hat{\xi}$ axis is aligned with the two primaries, with $\hat{h}$ orthogonal to the plane of motion. Distances are normalized accordingly to the instantaneous distance between the primaries. The unit distance can be defined as

$$k(t) = \left\| \mathbf{r}_E(t) - \mathbf{r}_M(t) \right\| \tag{8}$$

where $\mathbf{r}_E$ are $\mathbf{r}_M$ the primaries position in J2000. Therefore, k varies in time according to the mutual position of the two primaries, so creating a pulsating reference system. Moreover, time is adimensionalized such that mean motion about their common barycenter $\omega = \sqrt{\frac{G(m_E + m_M)}{\tilde{a}^3}}$ is set to unity, with (m_E and m_M, the Earth and Moon mass respective and $\tilde{a}$ the mean semi-major axis value. By choosing a constant mean motion, the average primaries revolution period is 2π. In this framework, Earth and the Moon are have fixed position, $[-\mu, 0, 0]^T$ and $[1 - \mu, 0, 0]^T$ respectively, with $\mu = m_M / (m_E + m_M)$ being the mass parameter of the system (Fig. 7).

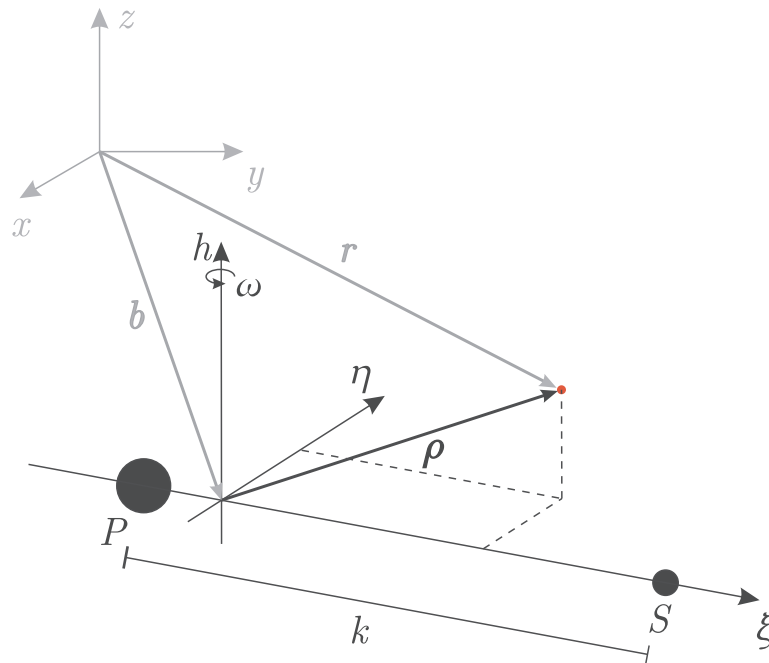The equations of motion for the RPRnBP reads [40]

Fig. 7 Rotating, pulsating, non-inertial reference frame (RPF). The inertial reference frame and quantities making reference to it are drawn in grey. Spacecraft is the red dot (Color figure online)

$$\rho'' + \frac{1}{\omega}\left(\frac{2\dot{k}}{k}I + 2C^T\dot{C}\right)\rho' + \frac{1}{\omega^2}\left(\frac{\ddot{k}}{k}I + 2\frac{\dot{k}}{k}C^T\dot{C} + C^T\ddot{C}\right)\rho + \frac{C^T\ddot{\mathbf{b}}}{k\omega^2} = \nabla\Omega + \mathbf{a}_{SRP}$$
(9)

where ρ is the spacecraft position in the RPF, the primes representing the derivatives with respect to the adimensional time τ, dots indicate the time derivatives and $\nabla\Omega = \partial\Omega/\partial\rho$ is the gradient of the pseudopotential

$$\Omega = \sum_{j\in\mathcal{S}} \frac{\hat{\mu}_j}{\delta_j}\left[1 + \frac{J_{2_j}R_{B_j}^2}{2k^2\delta_j^2}\left(1 - \frac{3\delta_j^T M\delta_j}{\delta_j^2}\right)\right]$$
(10)

where $\mathcal{S}$ is the set containing the primaries and characterized by the adimensional gravitational parameter $\hat{\mu}_j = m_j/(m_E + m_M)$, second harmonics coefficient J_{2_j} related to the non-spherical gravitational distribution and equatorial radius R_{B_j}, $M = C^T I_z C$, while $\delta_j = \rho - \rho_j$, and δ_j is its magnitude. The adimensionalized solar radiation pressure (SRP) acceleration in Eq. (9) can be expressed, using the cannon ball model, as

$$\mathbf{a}_{SRP} = \frac{\gamma_0}{\omega^2 k^3}\frac{\delta_S}{\delta_S^3}$$
(11)

where δ_S is the Sun position and γ_0 is the SRP parameter, defined as

$$\gamma_0 = (1 + c_r)\frac{A}{m}\frac{\Psi_0 d_0^2}{c}$$
(12)

with c_r the spacecraft reflectivity coefficient, A/m its area-to-mass ratio and c the speed of light in the vacuum.

Mixed derivative notation in Eq. (9) acknowledges that ephemeris data are numeric, discrete, and provided for regular dimensional time. Indeed, planets position $\mathbf{r}_j$ in J2000 are retrieved by using SPICE [41, 42] as well as the physical constants. The transformations from the solar barycentric inertial frame of reference (i.e., J2000) and the RPF are

$$\mathbf{r}(t) = \mathbf{b}(t) + k(t)C(t)\boldsymbol{\rho}(t) \tag{13}$$

$$\mathbf{v}(t) = \dot{\mathbf{b}} + \dot{k}C\boldsymbol{\rho} + kC\omega\boldsymbol{\rho}' \tag{14}$$

$$\tau = \omega(t - t_0) \tag{15}$$

with $\mathbf{b}$ the Earth–Moon barycenter position

$$\mathbf{b}(t) = \frac{m_E\mathbf{r}_E + m_M\mathbf{r}_M}{m_E + m_M} \tag{16}$$

and C the cosine angle matrix between J2000 and the RPF

$$C(t) = [\xi, \eta, h] \tag{17}$$

with

$$\xi = \frac{\mathbf{r}_M - \mathbf{r}_E}{k}, \quad h = \frac{(\mathbf{v}_M - \mathbf{v}_E) \times (\mathbf{r}_M - \mathbf{r}_E)}{\left\|(\mathbf{v}_M - \mathbf{v}_E) \times (\mathbf{r}_M - \mathbf{r}_E)\right\|}, \qquad \eta = h \times \xi \tag{18}$$

In Table 1, values for the most useful parameters in the dynamics are presented.

3.1.1 Variational Equations

In order to compute the trajectory correction maneuvers and the derivatives of the spacecraft state, useful in the optimization process, the variational equations are required. Being χ the spacecraft state in the RPF, the state-space representation of Eq. (9) is

$$
\chi' = \mathbf{f}(\chi, \tau) = \begin{bmatrix} \boldsymbol{\rho}' \\ \boldsymbol{v}' \end{bmatrix}
$$
$$
= \begin{bmatrix} \boldsymbol{v} \\ \nabla\Omega + \frac{\gamma_0}{\omega^2 k^3}\frac{\delta_S}{\delta_S^3} - \frac{1}{\omega}\left(\frac{2\dot{k}}{k}I + 2C^T\dot{C}\right)\boldsymbol{v} - \frac{1}{\omega^2}\left(\frac{\ddot{k}}{k}I + 2\frac{\dot{k}}{k}C^T\dot{C} + C^T\ddot{C}\right)\boldsymbol{\rho} - \frac{C^T\ddot{\mathbf{b}}}{k\omega^2} \end{bmatrix} \tag{19}
$$

with $\boldsymbol{v}$ the spacecraft velocity.

The state transition matrix (STM) can be computed by integrating the variational equation

Table 1 Parameters of restricted n-body problem and the spacecraft

Parameter	Symbol	Value
Earth–Moon mass ratio	μ	0.01215058426994
Earth gravitational parameter	μ_E	398 600.435 436 095 9 km^3/s^2
Moon gravitational parameter	μ_M	4902.800 066 163 796 km^3/s^2
Earth mean radius	R_{B_3}	6371.008 366 666 666 km
Moon mean radius	$R_{B_{10}}$	1737.4 km
Earth oblateness coefficient	J_{2_3}	0.001082616
Moon oblateness coefficient	$J_{2_{10}}$	0
SRP parameter	γ_0	2.2106568108 × 10^6 km^3/s^2
Reflectivity coefficient	c_r	0.08
Area-to-mass ratio	A/m	0.02 m^2/kg

$$\dot{\Phi}(\tau_0, \tau) = A(\tau)\Phi(\tau_0, \tau), \qquad \Phi(\tau_0, \tau_0) = I_6 \tag{20}$$

with

$$A(\tau) = \frac{\partial \mathbf{f}}{\partial \chi} = \begin{bmatrix} 0_3 & I_3 \\ \frac{\partial \mathbf{f}_v}{\partial \rho} & \frac{\partial \mathbf{f}_v}{\partial v} \end{bmatrix} \tag{21}$$

being the Jacobian of the dynamics right-hand side, where [8]

$$\begin{aligned}
\frac{\partial \mathbf{f}_v}{\partial \rho} = &-\frac{1}{\omega^2}\left(\frac{\ddot{k}}{k}I_3 + 2\frac{\dot{k}}{k}C^T\dot{C} + C^T\ddot{C}\right) \\
&-\sum_{j\in\mathcal{S}}\hat{\mu}_j\left[\frac{I_3}{\delta_j^3} - \frac{3\delta_j\delta_j^T}{\delta_j^5} + \frac{3J_{2_j}R_{B_j}^2}{2k^2}\left(\frac{I_3 + 2M}{\delta_j^5}\right.\right. \\
&\left.\left. -\frac{5\delta_j\delta_j^T + 10M\delta_j\delta_j^T + 5\delta_j^T M\delta_j + 10\delta_j\delta_j^T M}{\delta_j^7} + 35\delta_j^T M\delta_j\frac{\delta_j\delta_j^T}{\delta_j^9}\right)\right] \\
&+\frac{\gamma_0}{\omega^2 k^3}\left(\frac{I_3}{\delta_S^3} - 3\frac{\delta_S\delta_S^T}{\delta_S^5}\right)
\end{aligned} \tag{22}$$

while

$$\frac{\partial \mathbf{f}_v}{\partial v} = -\frac{2}{\omega}\left(\frac{\dot{k}}{k}I_3 + C^T\dot{C}\right) \tag{23}$$

3.1.2 Coordinates Transformation

The initial LLO is provided using Keplerian elements, i.e., a constant semi-major axis $a = 12037.1$ km and a constant eccentricity $e = 0.65848$ plus a set of free angular parameters $\boldsymbol{\alpha}_0 = \begin{bmatrix} i_0, \Omega_0, \omega_0, \theta_0 \end{bmatrix}$, containing the inclination i_0, the right ascension of the ascending node Ω_0, the argument of the pericenter ω_0 and the true anomaly θ_0. Keplerian parameters are given in Moon-centered Moon-equatorial at date (MCME2000) reference frame. In this frame, the *z-axis* is aligned with the Moon's spin axis on January 01, 2000, the *x-axis* is aligned with the Earth mean equinox (First point of Aries) and *y-axis* completes the right-handed reference frame. Thus an additional transformation is needed to go from the MCME2000 Keplerian elements to the cartesian coordinates in the J2000 reference frame, before being converted in RPF. Indeed, the Keplerian elements are converted into cartesian coordinates $\mathbf{x}_{\text{MCME}}$ [43]

$$\mathbf{r}_{\text{MCME}} = T_1 \begin{bmatrix} \frac{p \cos \theta_0}{1 + e \cos \theta_0} \\ \frac{p \sin \theta_0}{1 + e \cos \theta_0} \\ 0 \end{bmatrix}, \quad \mathbf{v}_{\text{MCME}} = T_1 \begin{bmatrix} -\sqrt{\frac{\mu_M}{p}} \sin \theta_0 \\ \sqrt{\frac{\mu_M}{p}} \left(e + \cos \theta_0 \right) \\ 0 \end{bmatrix} \tag{24}$$

with $p = a\left(1 + e^2\right)$ the semi-latus rectum, where the matrix $T_1 = R_z(\omega_0) R_x(i_0) R_z(\Omega_0)$ is defined through 3-dimensional rotation matrices. Then, the state is rotated in the Moon-Centered J2000

$$\mathbf{x}_{\text{J2000}} = \begin{bmatrix} \mathbf{r}_{\text{J2000}} \\ \mathbf{v}_{\text{J2000}} \end{bmatrix} = T_2 \begin{bmatrix} \mathbf{r}_{\text{MCME}} \\ \mathbf{v}_{\text{MCME}} \end{bmatrix} \tag{25}$$

with

$$T_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(i_M) & -\sin(i_M) \\ 0 & \sin(i_M) & \cos(i_M) \end{bmatrix}$$

where $i_M = 24$ deg is the lunar axial tilt with respect the Earth's equator [44]. Eventually, the position on the LLO is written in the solar barycentric J2000 by translation of the center from the Moon to the Solar System Baricenter, i.e.

$$\mathbf{x}_0 = \mathbf{x}_{\text{J2000}} + \mathbf{x}_M \tag{26}$$

Then, the J2000 initial state $\mathbf{x}_0$ is converted in the RPF one $\boldsymbol{\chi}_0$ by applying Eqs. (13, 14). The SMIM is applied on top of this initial state.

3.2 Uncertainty

In this test case, uncertainties are considered to be related only to the navigation and command errors. Errors generated by uncertainties in the dynamic model

(e.g., solar radiation pressure or residual accelerations) affect the transfer trajectory to a limit extent, due to the short-time propagation, and are dominated by the other errors. Thus, they are not considered in the model.

Navigation errors are taken into account as measurement model deviations in the OD phase and through an imperfect state knowledge at the initial time. The latter leads the initial state to be modeled as a Gaussian random variable with mean as the nominal initial state, i.e.

$$\chi(t_0) \sim \mathcal{N}\left(\chi_0, P_\chi\right) \tag{27}$$

where $P_\chi = \mathrm{diag}\left(\left[\sigma_\rho^2 I_3, \sigma_v^2 I_3\right]\right)$ is the 6-dimensional diagonal covariance matrix, with σ_ρ^2 and σ_v^2, the initial position and velocity covariances, respectively.

Moreover, in order to compensate for differences between physical model and real world, *command actuation errors* in the nominal impulses are considered, while TCMs are assumed free from uncertainties. Since uncertainty in the HIM does not affect the transfer phase and can be compensated with the station keeping algorithm foreseen in the operative orbit, the only significant uncertain maneuver is the SMIM. Thrust magnitude and direction are both modeled as Gaussian variables with a standard deviations $\sigma_{\Delta v}$ in magnitude and σ_δ in pointing angle. The magnitude error is defined as a fraction of the nominal value, i.e. $\sigma_{\Delta v} = u\Delta v_{\mathrm{SMIM}}$, with $u \ll 1$. The covariance matrix computation for the uncertainty on the SMIM requires retrieving SMIM vector in spherical coordinates, thus

$$\Delta v = \sqrt{\Delta v_x^2 + \Delta v_y^2 + \Delta v_z^2} \tag{28a}$$

$$\alpha = \mathrm{atan2}\left(\Delta v_y, \Delta v_x\right) \tag{28b}$$

$$\epsilon = \mathrm{atan2}\left(\Delta v_z, \sqrt{\Delta v_x^2 + \Delta v_y^2}\right) \tag{28c}$$

where Δv is the magnitude, and α and ϵ are the Azimuth and Elevation respectively. Then, the associated spherical covariance, i.e., $P_{\Delta v}^s = \mathrm{diag}\left(\sigma_{\Delta v}^2, \sigma_\delta^2, \sigma_\delta^2\right)$, is transformed in Cartesian coordinates

$$P_{\Delta v} = J P_{\Delta v}^s J^T \tag{29}$$

with J the Jacobian matrix of the cartesian-to-spherical conversion

$$J = \begin{bmatrix} \cos\epsilon\cos\alpha & -\Delta v\cos\epsilon\sin\alpha & -\Delta v\sin\epsilon\cos\alpha \\ \cos\epsilon\sin\alpha & \Delta v\cos\epsilon\cos\alpha & -\Delta v\sin\epsilon\sin\alpha \\ \sin\epsilon & 0 & \Delta v\cos\epsilon \end{bmatrix} \tag{30}$$

The total initial covariance can be computed as a combination of the initial state error, plus the maneuver error

Table 2 Stochastic characteristics of system uncertainty

σ_ρ [km]	σ_v [m/s]	$\sigma_{\Delta v}$ [%]	σ_δ [deg]
1	0.01	1	1

$$P_0 = P_\chi + \begin{bmatrix} 0_3 & 0_3 \\ 0_3 & P_{\Delta v} \end{bmatrix} \tag{31}$$

In doing so, the number of random variables can be reduced from 9, i.e. the 6-dimensional initial state plus the 3 SMIM components, to only 6 stochastic states, reducing the probabilistic space. Characteristics of the random variables are reported in Table 2.

4 Methodology

In order to deal with the revised approach for the LUMIO transfer phase case, a proper methodology should be devised, taking into account its peculiarities. It is of paramount importance to clarify: (i) which is the method used for the uncertainty propagation and how stcohastic variables are estimated, (ii) how the trajectory correction maneuver are computed, (iii) and how the orbit determination is performed. Moreover, the simplifying assumptions are presented as a preliminary for the optimization problem statement.

4.1 Uncertainty Propagation

In order to select an appropriate uncertainty quantification (UQ) scheme, four qualitative criteria are considered: (i) *Accuracy*: it considers how accurate is the method in estimating the propagated uncertainty; (ii) *Feasibility*: it assesses if the hypotheses are compatible with our problem; (iii) *Computational time*: it gives a qualitative measure of computational burden requested by the method; (iv) *Suitability*: it measures the suitability inside an optimization algorithm. The test case scenario is characterized by a limited number of thrust impulses, corresponding to few uncertainties, but the spacecraft is flying in a highly perturbed environment. Indeed, a high nonlinear solution is expected; however, the number of random variables is low. For this reason, a nonlinear uncertainty quantification method is needed [45] and, since the uncertainty vector is small, the curse of dimensionality does have a limited impact. Starting from this assumption, *Polynomial Chaos Expansion*, specifically in a non-intrusive method fashion, seems to feature the best balance of criteria.

Polynomial Chaos Expansion (PCE) is a nonlinear UQ technique, in which he input uncertainties and the solution are approximated using a series expansion based on some orthogonal polynomials, thus the approximated solution can be written as [46]

$$\hat{\mathbf{x}}(t, \xi) = \sum_{\alpha \in \Lambda_{p,d}} \mathbf{c}_\alpha(t) \psi_\alpha(\xi) \tag{32}$$

where $\Lambda_{p,d}$ is a set of the multi-index of size d and order p defined on nonnegative integers, $\xi = [\xi_1, \dots, \xi_d]$ is the set of input random variables, in which each element ξ_i is an independent identically distributed variable. The basis functions $\{\psi_\alpha(\xi)\}$ are multidimensional spectral polynomials, orthonormal with respect to the joint probability measure $\rho(\xi)$ of the vector ξ

$$\int_{\Gamma^d} \psi_\alpha(\xi) \psi_\beta(\xi) \rho(\xi) \mathrm{d}\xi = \delta_{\alpha\beta} \tag{33}$$

with Γ^d representing the d-dimensional hypercube where the random variable ξ are defined and $\delta_{\alpha\beta}$ is the Kronecker delta function. Thus, the basis functions choice depends only on $\rho(\xi)$. For instance, Hermite polynomials are the basis for normal random variables, while Legendre orthogonal polynomials are bases for the uniform distribution [47].

Generation of a PCE means computing the generalized Fourier coefficients $\mathbf{c}_\alpha(t)$ by projection of the exact solution $\mathbf{x}(t, \xi)$ onto each basis function $\psi_\alpha(\xi)$, truncated at the total order p

$$\mathbf{c}_\alpha(t) = E\big[\mathbf{x}(t, \cdot) \psi_\alpha(\cdot)\big] = \int_{\Gamma^d} \mathbf{x}(t, \xi) \psi_\alpha(\xi) \rho(\xi) \mathrm{d}\xi \tag{34}$$

The statistics of $\mathbf{x}(t, \xi)$ can be approximated by those of $\hat{\mathbf{x}}(t, \xi)$ from the coefficients $\mathbf{c}_\alpha(t)$ [48]. In fact, the mean is given by

$$\bar{\mathbf{x}}(t) = E[\mathbf{x}(t, \cdot)] = \mathbf{c}_0(t) \tag{35}$$

because $\psi_0 = 1$ and $E[\psi_\alpha] = 0$ for $\alpha \neq 0$. The covariance can be computed as

$$P(t) = E\Big[\big(\mathbf{x}(t, \cdot) - \bar{\mathbf{x}}(t, \cdot)\big)\big(\mathbf{x}(t, \cdot) - \bar{\mathbf{x}}(t, \cdot)\big)^T\Big] = \sum_{\substack{\alpha \in \Lambda_{p,d} \\ \alpha \neq 0}} \mathbf{c}_\alpha(t) \mathbf{c}_\alpha^T(t) \tag{36}$$

where the orthonormality of the polynomial basis is exploited.

PCE coefficients can be estimated by performing a Galerkin projection of the governing stochastic equations onto the $\{\psi_\alpha(\xi)\}$ subspace (the so-called, *intrusive* method), or solving a least-square regression or pseduospectral collocation (in the so-called, *intrusive* methods) [48].

In the selected scenario pseudospectral collocation will be exploited due to its flexibility and reduced computational burden. Pseudospectral collocation is based on the numerical integration of Eq. (34) by collocating $\mathbf{x}(t, \xi)$ on certain quadrature nodes defined on Γ_d in order to find $\mathbf{c}_\alpha(t)$. This means

$$\mathbf{c}_{\boldsymbol{\alpha}}(t) \simeq \mathcal{Q}\big[\mathbf{x}(t,\cdot)\psi_{\boldsymbol{\alpha}}(\cdot)\big] = \sum_{q=1}^{M} \mathbf{x}(t,\xi_q)\psi_{\boldsymbol{\alpha}}(\xi_q)\omega_q \tag{37}$$

where ξ_q is the set of quadrature nodes, ω_q are the quadrature weights, and $\mathcal{Q}$ is the quadrature scheme. Generally speaking, for a d-dimensional multivariate case, the quadrature formula can be written as

$$\mathcal{Q}[\mathbf{f}] = \sum_{q_1=1}^{m_1} \cdots \sum_{q_d=1}^{m_d} \mathbf{f}(\xi_1^{q_1}, \ldots, \xi_d^{q_d})\omega_{q_1} \ldots \omega_{q_d} \tag{38}$$

A total of $M = \prod_{i=1}^{d} m_i$ points for the function evaluation is needed. If the same number m of points along all the dimensions of ξ_i are taken, i.e., $m_1 = \cdots = m_d = m$, the approximation error for the quadrature formula in Eq. (38) is in the order of $\mathcal{O}(M^{-(2s-1)/d})$ for a $\mathcal{C}^s$ function $\mathbf{f}(\boldsymbol{\xi})$ [48]. Keeping the error constant, the number of function evaluations M grows exponentially fast with respect the dimensionality d of the random inputs. This issue is the so-called *curse of dimensionality*. In order to relieve the computational burden, higher-order interactions can be neglected, so to reduce the number of grid points. Sparse grid, like Smolyak's grid, are designed to serve this purpose.

The number of points can be further reduced using the so-called *conjugate unscented transformation (CUT)* [49]. CUT is the natural extension of unscented transformation, but, instead of employing only sigma-points on the principal axes of the initial distribution function, it propagates sigma-points chosen on some peculiar non-principal axes, giving the possibility to correctly estimate higher order moments of stochastic integrals [50]. Thus, it can be used to efficiently compute the generalized Fourier PCE coefficients exploiting Eq. (34). So, CUT can be seen as just another way to compute the stochastic integral given in Eq. (37). This hybrid technique, using CUT to estimate PCE coefficient, is unimaginatively labeled PCE-CUT. This approach exhibits several advantages over the standard sparse grid interpolation techniques, such as positive quadrature weights and fewer quadrature points.

Conjugate unscented transformation achieve to provide high-order quadrature rules satisfying the so-called momentum constraints equations (MCE). They are a set of equations that can be found by comparing the definition of a stochastic integral [Eq. (34)] with its quadrature approximation [Eq. (37)]. For the LUMIO transfer, quantities of interest are considered to be correctly represented using quadrature points that can completely satisfy the MCE up to the 4th order. In this case, CUT solution can be found by selecting the axes listed in Table 3, with the numerical value for the parameters shown in Table 4 [50].

The use of PCE-CUT4 requires the propagation of 77 samples in order to compute the quantity of interest. The equivalent full grid tensor product would require $3^6 = 729$ samples, while Smolyak's grid needs 85 points. Thus a 10% saving is expected in the computational times. Moreover, the positive quadrature weights improve the numerical stability, giving more accurate and fast results [51].

CUT4 results are computed by considering normalized Gaussian variables. If the random variables are represented by a generic multivariate Gaussian distribution

Table 3 Quadrature points for CUT4

Name	Symbol	Sample axis	Node position	Node weight
Center point	ξ_0	$[0, 0, \ldots, 0]$	$\mathbf{0}_d$	ω_0
Principal axes	σ	$[1, 0, \ldots, 0]$	$r_1 \sigma_i$	ω_1
Conjugated axes	$\mathbf{c}^l$	$[\underbrace{1, 1 \ldots, 1}_{l}, \underbrace{0, 0 \ldots, 0}_{d-l}]$	$r_2 \mathbf{c}_i^l$	ω_2

Table 4 CUT4 parameters solution

Parameter	Value
r_1	2.606009947366509
r_2	1.190556303640186
ω_1	0.021681819437030
ω_2	0.007777146339805
ω_0	0.242080802685967

with mean $\bar{\chi}$ and covariance matrix P, the generic quadrature point ζ_q can be retrieved by exploiting the affine transformation

$$\chi_q = S\xi_q + \bar{\chi} \tag{39}$$

with S being the Cholesky decomposition of P, i.e., $P = S^T S$.

In order to assess the converge accuracy of PCE-CUT4 in the LUMIO transfer phase case, the ratio between the i-th and the first PCE coefficient $\mathbf{c_0}$ is evaluated. Indeed, the fastest is its reduction, the most accurate is the expansion result. Moreover, there is a direct connection between this value and the digit precision [48]. Figure 8 illustrates the convergence accuracy for two representative components of position and velocity, showing that the PCE expansion converges quickly, with 5-precision digit in (adimensionalized) position and 4-precision digit in (adimensionalized) velocity. An alternative assessment can be given by comparing the standard deviation for a Monte Carlo (MC) simulation and the proposed technique. Figure 9 shows that final values are similar both for MC and PCE. However, the PCE-CUT4 converge rate overcomes the MC one, i.e., more Monte Carlo samples are needed to achieve the accuracy given by the PCE.

4.1.1 Stochastic Variables Estimation

Once the PCE coefficients $\mathbf{c}_{\alpha}$ at a given time τ are retrieved by means of the 4th-order CUT, the stochastic state at a given time can be estimated as [Eq. (32)]

$$\chi(\tau, \xi) = \sum_{\alpha \in \Lambda_{p,d}} \mathbf{c}_{\alpha}(\tau) \psi_{\alpha}(\xi) \tag{40}$$
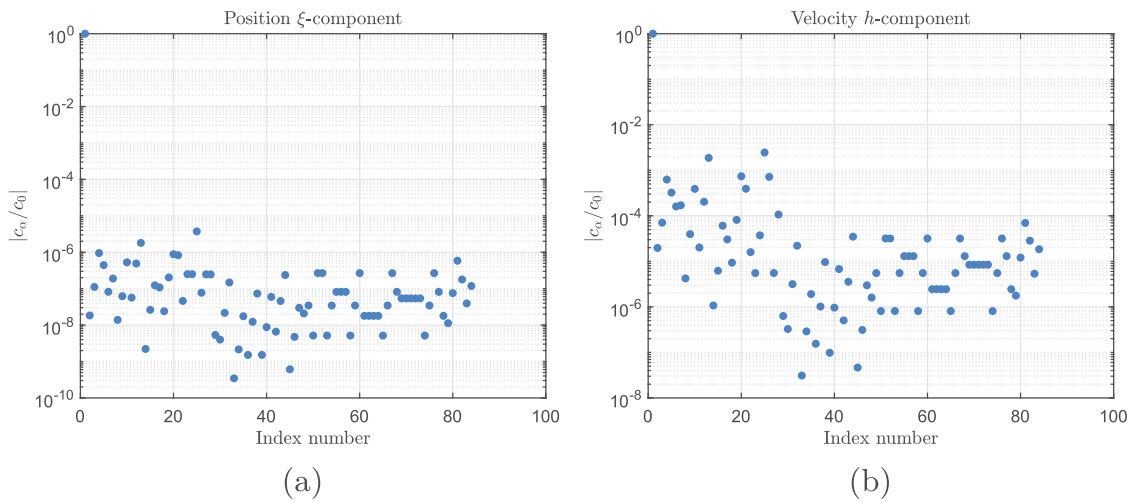
Fig. 8 Normalized PCE coefficients for the PCE-CUT4 for the LUMIO transfer phase final state: **a** ξ component of the position, **b** h component of the velocity
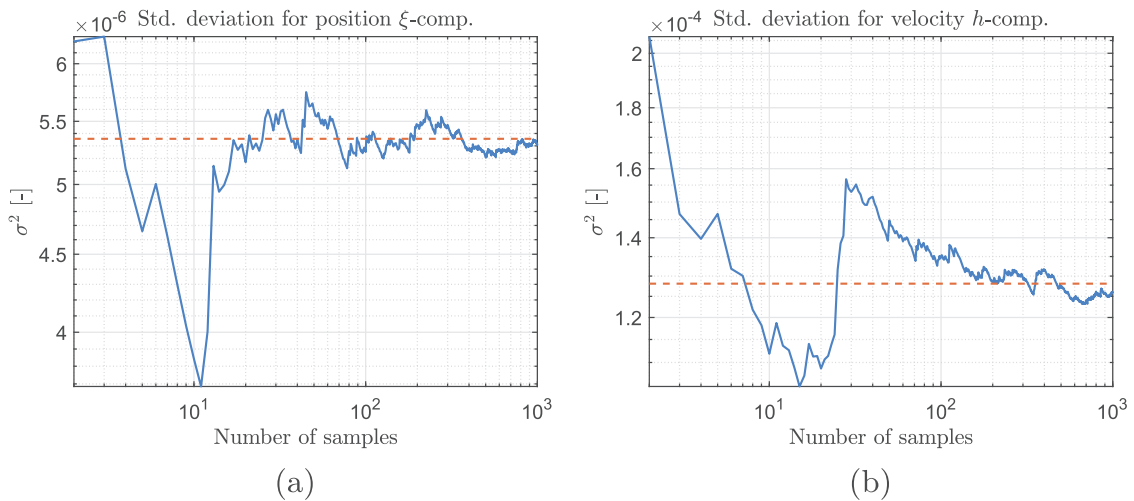

Fig. 9 Standard deviation for the LUMIO transfer phase final state as function of the random sample size of a Monte Carlo simulation: **a** ξ component of the position, **b** h component of the velocity. Dashed line is the PCE-CUT4 solution

This solution is expected to be strongly non-Gaussian. For this reason, the final stochastic state and the functions depending on it cannot be described employing only mean and covariance, but the full probability density function (PDF) has to be estimated and then used to evaluate probabilities. In order to do that, kernel density estimation (KDE) [52] is used. In this technique, the surrogate model is exploited to inexpensively produce a number n of samples of the quantity of interest $q_j = q(\chi(\tau, \xi_j))$, depending on n random variables ξ_j, with $j = \{1, \ldots, n\}$. Then they are used to estimate the PDF as

$$\hat{\rho}(q) = \frac{1}{nh} \sum_{j=1}^{n} K\left(\frac{q - q_j}{h}\right) \tag{41}$$

where h is the bandwidth, and K is the kernel function. The kernel function is selected as the Gaussian PDF, i.e., $K(z) = \frac{1}{\sqrt{2\pi}} \exp\left[-\frac{z^2}{2}\right]$. The cumulative distribution function (CDF) can be computed as

$$\hat{F}(q) = \int_{-\infty}^{q} \hat{\rho}(q)\,dq = \frac{1}{n}\sum_{j=1}^{n} G\left(\frac{q - q_j}{h}\right) \tag{42}$$

where $G(q) = \int_{-\infty}^{q} G(q)\,dq$. In the case of Gaussian kernel, $G(q) = \frac{1}{2}\left[1 + \mathrm{erf}\left(\frac{q}{\sqrt{2}}\right)\right]$
.

The selection of the bandwidth is tricky and different algorithms exist. In this work, the *Silverman's rule of thumb* [53] is considered: the value of h is selected as the bandwidth minimizing the mean integrated squared error for a Gaussian distribution. In this case,

$$h = \left(\frac{4\hat{\sigma}^5}{3n}\right)^{\frac{1}{5}} \tag{43}$$

where $\hat{\sigma}$ is the standard deviation of the n samples. Using KDE is preferred with respect to a simple counting method, since a smooth $\mathcal{C}^\infty$-class CDF is obtained and this is helpful in the optimization procedure.

In order to estimate the population quantiles, a similar technique called kernel quantile estimation (KQE) is employed. The quantile function is the left-continuous inverse of the CDF

$$Q(p) = \inf\{q : F(q) \le p\} \qquad \text{with} \quad 0 \le p \le 1 \tag{44}$$

i.e., the function returning the threshold value of q, such that the probability variable being less than or equal to that value equals the given probability p. Using the KQE, the quantile function can be computed as [54]

$$Q(p, q) = \sum_{j=1}^{n} \frac{1}{nh} K\left[\frac{1}{h}\left(\frac{j}{n} - p\right)\right]\tilde{q}_j \tag{45}$$

where $\tilde{q}_j, j = \{1, \dots n\}$ is the sorted set of q_j and K is the kernel function. The use of this linear KQE formula give the possibility to obtain reliable estimation for the desired quantile value, while having a $\mathcal{C}^\infty$-class function.

4.2 Orbit Determination Process

In order to determine the spacecraft state knowledge along the transfer phase, a covariance analysis is performed and the knowledge is estimated by means of an orbit determination algorithm.

In this scenario, *radiometric tracking* is selected as navigation technique. Thus, the spacecraft state is estimated by means of radiometric data processed by a ground
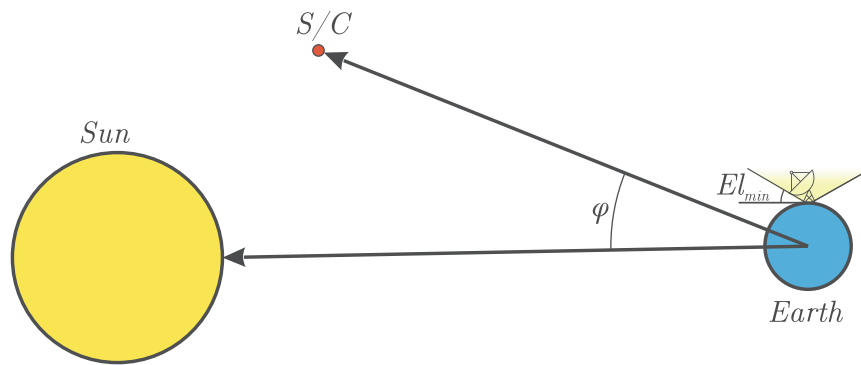
Fig. 10 Tracking problem geometry

station. Radiometric data for range and range-rate are simulated, generating pseudo-measurements as

$$\gamma = \sqrt{\boldsymbol{\gamma}^T \boldsymbol{\gamma}}, \qquad \dot{\gamma} = \frac{\boldsymbol{\gamma}^T \boldsymbol{\eta}}{\gamma} \qquad (46)$$

where γ is the range, $\dot{\gamma}$ is the range rate, $\boldsymbol{\gamma} = \mathbf{r} - \mathbf{r}_{GS}$ is the relative distance between LUMIO and the ground station, while $\boldsymbol{\eta} = \mathbf{v} - \mathbf{v}_{GS}$ is the relative velocity. Pseudo-measurements can be performed only if a link between the spacecraft and the selected ground station can be established. Thus, for each OD phase, a visibility window is identified. Visibility windows are portion of the trajectory inside the OD phases, where some geometric conditions are verified. With reference to Fig. 10, the requirements are:

- The Sun exclusion angle ϕ should be greater than 0.5 deg in order to avoid degradation in the radiometric observable and, in turn, in the trajectory knowledge [55];
- The Spacecraft elevation above the ground El in ground station location should be higher than a minimum value $El_{\min}$ in order to avoid low-quality data related to the atmospheric extinction of the radiometric signal and to cope with the mounting constraints of the ground station sensor.

For the LUMIO case, the Sardinia deep space antenna (SDSA) scientific unit (64 ms) located in San Basilio, Cagliari, is assumed as reception baseline option for the ground communications. Currently, SDSA has X-band reception capability. In the future, reception in the Ka-band and transmission in the X- and Ka-bands will be made available.[2] SDSA performances are summarized in Table 5.

In order to estimate the state, a navigation filter exploiting the pseudo-measurements is needed. The filter embedded in the orbit determination process is an *Ensemble square root filter (EnSRF)* [56, 57]. This method exploits the capability of

---

[2] https://www.asi.it/en/the-agency/the-space-centers/sardina-deep-space-antenna-sdsa/ (Last accessed on January 13, 2022)

Table 5 Assumed characteristics for SDSA ground station

Parameter	Value
Coordinates	39.493 028° N
	9.245 111° E
Altitude	0.2064 km
Range measurements frequency	Once every hour
Doppler measurements frequency	Once every 20 min
Range measurements random error (1σ)	130 m
Range measurements systematic error	130 m
Doppler measurements random error (1σ)	0.2 mm/s
Doppler measurements systematic error	0 mm/s
Minimum ground station elevation $El_{\min}$	15°

PCE to generate inexpensively huge ensembles of samples. Moreover, EnSRF does not require perturbed observations; thus, no sampling error is introduced in Kalman gain matrix, improving the accuracy of the filter.

Inside the visibility window of each OD phase, the time is discretized in evenly spaced intervals following the measurements frequency imposed by the ground station. In these points, the pseudo-measurements are generated based on the current state and then used to feed the EnSRF. Between two consecutive measurements time, the estimated and the real state are propagated forward, together with the associated CUT samples, useful to compute the needed PCE coefficients. Hence,

$$
\begin{aligned}
\chi_{k+1} &= \varphi\big(\chi_k, \tau_k; \tau_{k+1}\big) \\
\hat{\chi}_{k+1} &= \varphi\big(\hat{\chi}_k, \tau_k; \tau_{k+1}\big) \\
\chi_{k+1}^q &= \varphi\big(\chi_k^q, \tau_k; \tau_{k+1}\big), \qquad q \in \{0, 1, \dots, M\}
\end{aligned}
\tag{47}
$$

where subscript k and $k+1$ are referred to the measurement times τ_k and τ_{k+1} respectively, χ is the real trajectory, $\hat{\chi}$ is the estimated trajectory, while χ^q are the CUT samples. A generic perturbed nonlinear measurement model is considered

$$
\mathbf{z} = \mathbf{h}(\chi) + \boldsymbol{\varepsilon}
\tag{48}
$$

with $\mathbf{z}$ the measurement and $\boldsymbol{\varepsilon}$ is the measurement error. A linear measurement operator is therefore defined as $H = \left. \frac{\partial \mathbf{h}(\chi)}{\partial \chi} \right|_{\chi = \hat{\chi}}$.

At each τ_k the EnSRF embedded in the OD, the PCE coefficients for the real and estimated state are retrieved

$$
\mathbf{c}_\alpha^{k,-} = \sum_{q=1}^{M} \chi_{k,-}^q \psi_\alpha^q \omega_q
\tag{49}
$$

$$\hat{\mathbf{c}}_{\alpha}^{k,-} = \sum_{q=1}^{M} \hat{\chi}_{k,-}^{q} \psi_{\alpha}^{q} \omega_{q} \tag{50}$$

with $-$ indicating the variables before the filter correction. The mean and covariance are estimated exploiting the PCE properties [Eqs. (35, 36)]

$$\begin{cases} \bar{\bar{\chi}}_{k,-} = \hat{\mathbf{c}}_{0}^{k,-} \\ \hat{P}_{k}^{-} = \sum_{\substack{\alpha \in \Lambda_{p,d} \\ \alpha \neq \mathbf{0}}} \left(\hat{\mathbf{c}}_{\alpha}^{k,-}\right)\left(\hat{\mathbf{c}}_{\alpha}^{k,-}\right)^{T} \end{cases} \tag{51}$$

together with the real state mean

$$\bar{\chi}_{k,-} = \mathbf{c}_{0}^{k,-} \tag{52}$$

Measurements are obtained for both the forecast (estimated) mean and propagated (real) mean as

$$\mathbf{z}_{k} = \mathbf{h}\left(\bar{\chi}_{k,-}\right), \qquad \hat{\mathbf{z}}_{k} = \mathbf{h}\left(\bar{\bar{\chi}}_{k,-}\right) \tag{53}$$

Then, n realizations of ξ_{i} are generated and associated basis functions are evaluated leading to $\psi_{\alpha}^{i} = \psi_{\alpha}(\xi_{i})$. An ensemble of n forecast state is then computed

$$\hat{\chi}_{k,-}^{i} = \sum_{\alpha \in \Lambda_{p,d}} \hat{\mathbf{c}}_{\alpha}^{k,-} \psi_{\alpha}^{i} \tag{54}$$

The EnSRF Kalman gains are computed

$$S_{k} = H_{k} \hat{P}_{k}^{-} H_{k}^{T} + R \tag{55}$$

$$K_{k} = \hat{P}_{k}^{-} H_{k}^{T} S_{k}^{-1} \tag{56}$$

$$\tilde{K}_{k} = \hat{P}_{k}^{-} H_{k}^{T} \sqrt{S_{k}}^{-T} \left(\sqrt{S_{k}} + \sqrt{R}\right)^{-1} \tag{57}$$

with $R = E\left[\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^{T}\right]$ the measurement error covariance matrix.

In the EnSRF, mean and deviations are updated separately. Thus, firstly the deviation of forecast states with respect to the mean is computed

$$\delta\chi_{k,-}^{i} = \hat{\chi}_{k,-}^{i} - \bar{\chi}_{k,-} \tag{58}$$

and secondly, mean and deviations are updated as

$$\bar{\bar{\chi}}_{k,+} = \bar{\bar{\chi}}_{k,-} + K_{k}\left(\mathbf{z}_{k} - \hat{\mathbf{z}}_{k}\right) \tag{59}$$

$$\delta\chi^i_{k,+} = \delta\chi^i_{k,-} - \tilde{K}_k H \delta\chi^i_{k,-} \tag{60}$$

Eventually, a n-dimensional ensemble of corrected states is computed from the mean and the deviations

$$\hat{\chi}^i_{k,+} = \bar{\bar{\chi}}_{k,+} + \delta\chi^i_{k,+} \tag{61}$$

and then they are used to update the PCE coefficients by using an inexpensive least-square regression [48]

$$\hat{\mathbf{c}}^{k,+}_{\boldsymbol{\alpha}} = \left(\boldsymbol{\Psi}^T\boldsymbol{\Psi}\right)^{-1}\boldsymbol{\Psi}X^+_k \tag{62}$$

with X^+_k is the corrected state matrix, containing $\hat{\chi}^i_{k,+}$, and $\boldsymbol{\Psi}$ is the measurement matrix.

As last point, the new cubature points for the CUT are retrieved, by computing the corrected state statistics

$$\begin{cases} \bar{\bar{\chi}}_{k,+} = \hat{\mathbf{c}}^{k,+}_0 \\ \hat{P}^+_k = \sum_{\substack{\boldsymbol{\alpha}\in\Lambda_{p,d} \\ \boldsymbol{\alpha}\neq\mathbf{0}}} \left(\hat{\mathbf{c}}^{k,+}_{\boldsymbol{\alpha}}\right)\left(\hat{\mathbf{c}}^{k,+}_{\boldsymbol{\alpha}}\right)^T \end{cases} \tag{63}$$

and then projecting it on the generalized conjugated axes space (Eq. (39))

$$\hat{\chi}^q_{k,+} = S\xi_q + \bar{\bar{\chi}}_{k,+} \tag{64}$$

with S being the Cholesky decomposition of $\hat{P}^+_k$, i.e. $\hat{P}^+_k = S^T S$.

4.3 Guidance Law

An algorithm, able to compute tailored impulses in order to let the spacecraft fly the nominal path, is needed. Control maneuvers reduce the dispersion with little propellant effort. In order to estimate the trajectory control maneuvers, a dedicated strategy is implemented. In literature, these techniques are usually subdivided into two main groups: (a) *Closed-loop control*, if control impulses are given to track the reference guidance, or (b) *Closed-loop guidance*, if control impulses are given to update the whole spacecraft trajectory in order to satisfy the mission objectives. Several different guidance and control laws exist, and the choice of the most suitable method is based essentially on the mission profile, spacecraft characteristics and the general scenario. In this work, only *closed-loop control*, i.e. the nominal trajectory tracking is used as control strategy. However, the algorithm computing navigation maneuvers will be always tagged as *guidance algorithm*. Maneuvers are computed at a prescribed time, in order to comply with on-ground segment requirements. The differential guidance (DG) is a commonly used guidance method for deep-space missions [58]. DG aims at canceling the final state deviation using two maneuvers, one

at beginning and the other at the end of the considered leg, even though usually the second control maneuver is not applied, since at the final leg time a new maneuver can be computed and the whole algorithm can be repeated in a receding horizon approach. Since the time interval between navigation maneuvers is relatively short, first-order approximation can be used to relate the initial and final deviations, and the TCM can be computed as [5]

$$\Delta \mathbf{v}_j^s = -\left(\Phi_{rv}^T \Phi_{rv} + q\Phi_{vv}^T \Phi_{vv}\right)^{-1}\left(\Phi_{rv}^T \Phi_{rr} + q\Phi_{vv}^T \Phi_{vr}\right)\delta\mathbf{r}_j - \delta\mathbf{v}_j \qquad (65)$$

where Φ_{rr}, Φ_{rv}, Φ_{vr}, and Φ_{vv} are the 3-by-3 blocks of $\Phi(t_j, t_{j+1})$, i.e., the STM associated to the nominal trajectory between two consecutive TCM times t_j and t_{j+1}, while $\delta\mathbf{r}$ and $\delta\mathbf{v}$ are the deviations with respect to the nominal trajectory. If the OD is considered in the loop, deviations are taken from the estimated trajectory

$$\begin{cases} \delta\mathbf{r} = \hat{\mathbf{r}} - \mathbf{r}^* \\ \delta\mathbf{v} = \hat{\mathbf{v}} - \mathbf{v}^* \end{cases} \qquad (66)$$

The parameter q is used either to adjust dimensions or the change the guidance algorithm behavior, favoring position deviation at the expense of velocity deviation and vice versa. In this work, $q = 0.01$ has been selected after some numerical experiments.

In this work, the control impulse in Eq. (65) is applied at each TCM time.

4.3.1 Simplifying Assumptions

Once the foundational blocks are modeled, combining the timeline (Fig. 6) with the methodology illustrated in this Section, a comprehensive method able to retrace the revised approach and to provide knowledge analysis, final dispersion and an estimation of the navigation costs in a single shot can be devised. However, an algorithm using PCE-CUT on each real orbit to estimate both knowledge and dispersion, and using the OD results to perturb randomly the real state at the TCM location, requires an excessively large amount of computational effort. The number of stochastic variables (being 18, 6 from the uncertain states and 12 from the estimated state errors) leads to a huge number of samples to be propagated to obtain the sought results, and thus to the curse of dimensionality. In order to reduce the computational burden, some simplifying assumptions can be made.

First of all, the knowledge analysis is performed only on the nominal trajectory and its results are used also on the real orbits stemming from the initial dispersion. This assumption is valid whenever the real trajectory does not deviate too much from the nominal one, relatively to the Spacecraft-to-Ground Station distance. In this case, the pseudo-measurements on the nominal trajectory are similar to the real trajectory ones and the outputs from the EnSRF are comparable.
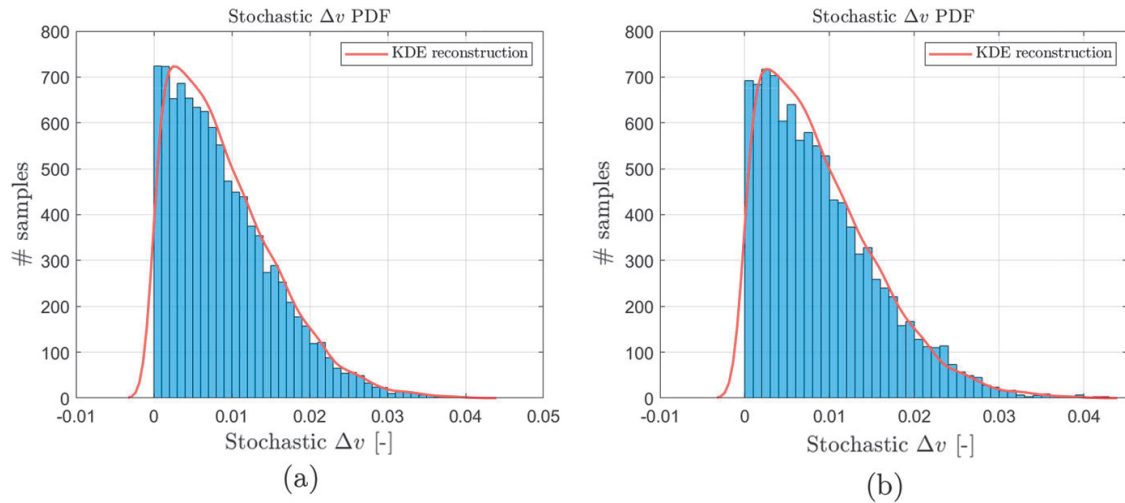
Fig. 11 Comparison of stochastic costs if navigation error is **a** picked randomly, or **b** equal to the error mean in LUMIO case

Secondly, the estimated state error at the end of the OD phase is not picked randomly from the knowledge distribution at the time, but only the average error is considered to assess the estimated state for all the real trajectories. This strong assumption is able to reduce the number of random variables and, thus, lessen the problem dimensionality. Figure 11 shows stochastic costs for LUMIO trajectory both (a) if the navigation error at the orbit determination time is a Gaussian random variable with mean and covariance given by the filter output and (b) if the navigation error is taken as a deterministic value equal only to the mean. The navigation costs and final dispersion have similar distribution in both cases. Hence, for this scenario, this last assumption is valid.

5 Statement of the Problem

Once the building blocks are established, Problem 1 has to be adapted to cope with the test case scenario, represented by LUMIO transfer phase and the general optimal control problem is converted into a non-linear programming problem. The optimization problem for the test case scenario under the revised approach can be stated as

Problem 2 (*Fuel-Optimal Problem*) Find the initial and final time, τ_0 and τ_f, the two TCM times, τ_{TCM_1} and τ_{TCM_2}, the angular parameter vector $\boldsymbol{\alpha}_0$, the SMIM vector Δv_{SMIM}, and the nominal trajectory impulse at TCM times, Δv_{TCM_1} and Δv_{TCM_2}, such that

$$J = \sum_i \left\| \Delta v_i \right\| = \left\| \Delta v_{SMIM} \right\| + \left\| \Delta v_{TCM_1} \right\| + \left\| \Delta v_{TCM_2} \right\| + \left\| \Delta v_{HIM} \right\| + \sum_{j=1}^{2} Q(.99, \Delta v_j^s) \tag{67}$$

with $Q(0.99, \Delta v_j^s)$ representing the 99-percentile of the stochastic cost computed through Eq. (45), is minimized, while the state is subjected to the dynamics illustrated in Eq. (19)

$$
\chi' = \left[\begin{array}{c} \mathbf{v} \\ \nabla\Omega + \frac{\gamma_0}{\omega^2 k^3}\frac{\delta_S}{\delta_S^3} - \frac{1}{\omega}\left(\frac{2\dot{k}}{k}I + 2C^T\dot{C}\right)\mathbf{v} - \frac{1}{\omega^2}\left(\frac{\ddot{k}}{k}I + 2\frac{\dot{k}}{k}C^T\dot{C} + C^T\ddot{C}\right)\rho - \frac{C^T\ddot{\mathbf{b}}}{k\omega^2} \end{array} \right]
\tag{68}
$$

The HIM is computed as

$$
\Delta v_{\text{HIM}} = \mathbf{v}^*\left(\tau_f\right) - \mathbf{v}_\delta\left(\tau_f\right)
\tag{69}
$$

with $\mathbf{v}^*$ the nominal velocity and ρ_δ is the target halo velocity.

The state is subjected to initial constraints

$$
\begin{cases} E\left[\chi^*(\tau_0)\right] = [\hat{\chi}^*(\tau_0) = \chi_0(\tau_0, \alpha_0) + \Delta v_{\text{SMIM}} \\ E\left[\left(\chi^*(\tau_0) - \chi_0\right)\left(\chi^*(\tau_0) - \chi_0\right)^T\right] = P_0 \end{cases}
\tag{70}
$$

and

$$
E\left[\left(\chi(\tau_0) - \chi_0\right)\left(\chi(\tau_0) - \chi_0\right)^T\right] = P_0
\tag{71}
$$

and a final constraint

$$
c = \hat{F}_d(30\,\text{km}) > 0.99
\tag{72}
$$

with $d = \left\|\rho\left(\tau_f\right) - \rho_\delta\left(\tau_f\right)\right\|$, being a measure of the distance of the real trajectories from the halo at τ_f, where ρ_δ is the target halo.

In order to be compliant with on-ground operation requirements, some linear constraints are added

$$
2\,\text{d} \le \left(\tau_{TCM_1} - \tau_0\right) \le 8\,\text{d}
\tag{73}
$$

$$
5.5\,\text{d} \le \left(\tau_{TCM_2} - \tau_0\right) \le \tau_f - 3\,\text{d}
\tag{74}
$$

The navigation costs and the final dispersion are estimated through the comprehensive navigation assessment. It means

$$
\Delta v^s = \text{GL}\left(\chi^*, \hat{\chi}, \tau_{TCM}\right)
\tag{75}
$$

and

$$
\hat{\chi}\left(\tau_f^{OD}\right) = \text{OD}\left(\chi^*, \hat{\chi}, \tau_0^{OD}, \tau_f^{OD}\right)
\tag{76}
$$

with GL and OD being the Differential Guidance Law (Eq. (65)) and orbit determination processes (Sect. 4.2) on the nominal trajectory respectively, $\hat{\chi}$ is the estimated state, χ the real state and χ^* is the nominal state.

The quantile value for the j-th TCM exploiting the KQE (Eq. (45)) is

$$Q\big(0.99, \Delta v_i^s\big) = \sum_{q=1}^{n} \frac{1}{nh} \frac{1}{\sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\frac{1}{h}\left(\frac{i}{n} - 0.99\right)\right)^2\right] \widetilde{\Delta v}_j^{s,i} \tag{77}$$

where $\widetilde{\Delta v}_j^{s,i}$ are the sorted value of $\left\|\Delta v_j^{s,i}\right\|$. The 100, 000 samples $\Delta v_j^{s,i}$ are obtained through an inexpensive Monte Carlo exploiting PCE-CUT technique. The CUT samples $\Delta v_j^{s,q}$ for each TCM are obtained by exploiting the DG algorithm (Eq. (65))

$$\Delta v_j^{s,q} = -\big(\Phi_{rv}^T\Phi_{rv} + q\Phi_{vv}^T\Phi_{vv}\big)^{-1}\big(\Phi_{rv}^T\Phi_{rr} + q\Phi_{vv}^T\Phi_{vr}\big)\widetilde{\delta\rho}_j - \widetilde{\delta v}_j \tag{78}$$

with

$$\widetilde{\delta\chi}_j = \tilde{\chi}_j - \chi^* \tag{79}$$

being the estimated error at k-th TCM time, and where

$$\tilde{\chi}_j = \varphi\Big(\hat{\chi}\big(\tau_f^{OD}\big), \tau_f^{OD}; \tau_j\Big) \tag{80}$$

is the flow of the estimated state, for each CUT sample, propagated from the end of the OD phase up to the next TCM time. As per Sect. 4.3.1,

$$\hat{\chi}\big(\tau_f^{OD}\big) = \chi\big(\tau_f^{OD}\big) + \Big(\bar{\tilde{\chi}}_f^{OD} - \chi\big(\tau_f^{*,OD}\big)\Big) \tag{81}$$

with the mean final error obtain through Eq. (63).

The PCE coefficients can be retrieved as

$$\mathbf{c}_\alpha = \sum_{q=1}^{M} \Delta v_j^{s,q} \psi_\alpha^q \omega_q \tag{82}$$

and then used to obtain the samples

$$\Delta v_{s,i}^j = \sum_{\alpha \in \Lambda_{p,d}} \mathbf{c}_\alpha \psi_\alpha^i \tag{83}$$

where ψ_α^i are the basis functions evaluated at a random picked value ξ_i. In order to simplify the problem, the bandwidth is considered constant with $h = 0.001$.

The value in Eq. (72) is obtained in the same way, exploiting the KDE as in Eq. (42) with a constant bandwidth $h = 0.0622$.

In summary, the decision variable vector is defined as $\mathbf{y} = \big[\tau_0, \tau_f, \tau_{TCM_1}, \tau_{TCM_2}, \alpha_0, \Delta v_{SMIM}, \Delta v_{TCM_1}, \Delta v_{TCM_2}\big]^T$ and its bounds are summarized in Table 6. No bounds are placed on LLO angular parameters.

Table 6 Decision vector bounds

	Dep. date	τ_f [d]	τ_{TCM_1} [d]	τ_{TCM_2} [d]	Δv_{TCM_j} [m/s]
Upper bound	$\tau_0 + 7$ d	28	8	$\tau_f - 3$ d	10
Lower bound	$\tau_0 - 7$ d	15	2	5.5	-10

τ_0 is the first guess departure date

The procedure used to estimate the cost function in Eq. (67) and the nonlinear constraint in Eq. (72) is summarized in Algorithm 1.

Problem 2 is solved by exploiting a simple shooting technique [59]. This method is selected as the most suitable to solve the optimization problem, since (i) the trajectory lasts only few days and nominally no middle correction is enforced, so low numerical noise is expected in the derivatives, (ii) number of variables is strongly reduced, (iii) and Algorithm 1 can be implemented straightforwardly. In order to speed up the NLP solution, the Jacobian, both for the objective function and the nonlinear constraint, can be provided. Details on the Jacobian construction are given in Appendix A.

5.1 First Guess

An educated guess is required to solve the optimization problem, in order to reduce the wide search space represented by all the possible parking low lunar orbits, the stable manifold insertion time, and the arrival state on the halo as well as to cope with the use of a local optimization scheme. In order to compute these first guesses, a generation mechanism exploiting a patching process is devised: (1) the state is *propagated backward* from the operative orbit to the first lunar pericenter; (2) a *grid generation* is used to create all the possible LLO pericenters at a given time; (3) a *patching process* selects the solutions fulfilling some tolerances in time and space at their pericenter; (4) eventually, solutions of the patching process are then used to feed an *optimization algorithm*, able to close the gap between the LLO and the stable manifold, while minimizing the needed propellant. Least expensive trajectory coming from the optimization algorithm are then used as first guess for the optimization problem stated in Problem 2. The algorithm is summarized in Fig 12.

5.1.1 Backward Propagation

In the backward leg, the spacecraft trajectory is propagated backward in time from the halo target orbit for 30 days in the high-fidelity restricted roto-pulsating n-body problem. The epoch and the states of the first lunar pericenter pass are saved to be used for patching purposes. Initial points on the halo are drawn from an equally spaced grid of 6 h, starting form the February 1, 2024 and ending on March 1, 2024. This choice reflects the requirement to start the science operation at latest on March 21, 2024 and to have at least 1 week of commissioning and calibration in the operative orbit. In order to escape the fastest way, a 1 m/sΔv is applied along the minimum stretching direction. Because in the high-fidelity model the operative orbit is

Procedure INTEGRATED APPROACH

  Define spacecraft and navigation settings;   ▷Tables 1 and 5
  Define the uncertainty properties;   ▷Table 2
  Function INITIALIZATION
    Evaluate the initial CUT samples χ^q_{k+1} ;   ▷See Eq. (27)
    Compute the random basis function for KDE-KQE;
  Function KNOWLEDGE ANALYSIS
    Consider the nominal trajectory;
    for $i = 1$ **to** n_P ▷Loop through n_P sub-phases
      switch *sub-phase*
        case *OD phase*
          Function ORBIT DETERMINATION
            Find the visibility windows;
            Retrieve the n_M measurement times t_k;
            for $k = 1$ **to** n_M ▷Loop through n_M meas. times
              Get the pseudo-measurement;   ▷See Eq. (46)
              Apply the EnSRF;   ▷See Sec. 4.2
              Get estimated states from the filter;
              Propagate the estimated state samples to t_{k+1};
            end
            Propagate the estimated samples up to t_f^{OD};
          Result: Average error at each OD final time
        otherwise
          Propagate the states up to the *final sub-phase* time;
      end
    end
  Result: Kwowledge time evolution
  Function NAVIGATION COSTS & DISPERSION
    for $i = 1$ **to** n_P ▷Loop through n_P sub-phases
      switch *sub-phase*
        case *Cut-off phase*
          Propagate the CUT samples up to the TCM time;
          Estimate the correction maneuvers $\Delta v_j^{s,q}$; ▷See Eq. (78)
          Apply the maneuvers to each sample;
        case *OD phase*
          Propagate the CUT samples up to the final OD time;
          Estimate the mean error; ▷See ORBIT DETERMINATION
          Evaluate the estimated samples;   ▷See Eq. (81)
        otherwise
          Propagate the samples up to the *final sub-phase* time;
      end
    end
  Result: Navigation cost estimate; Final dispersion
Result: Cost function (Eq. (67)); Dispersion statistic (Eq. (72))

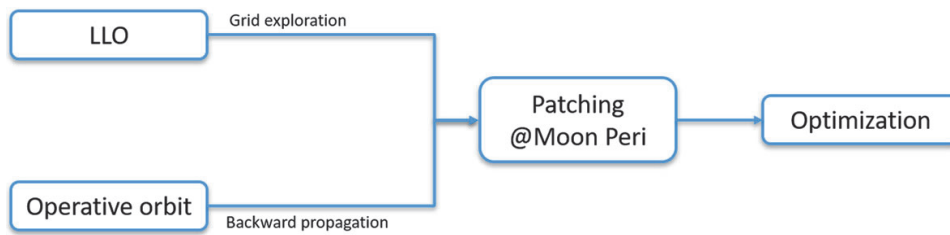**Algorithm 1** Integrated approach algorithm for test cast scenario

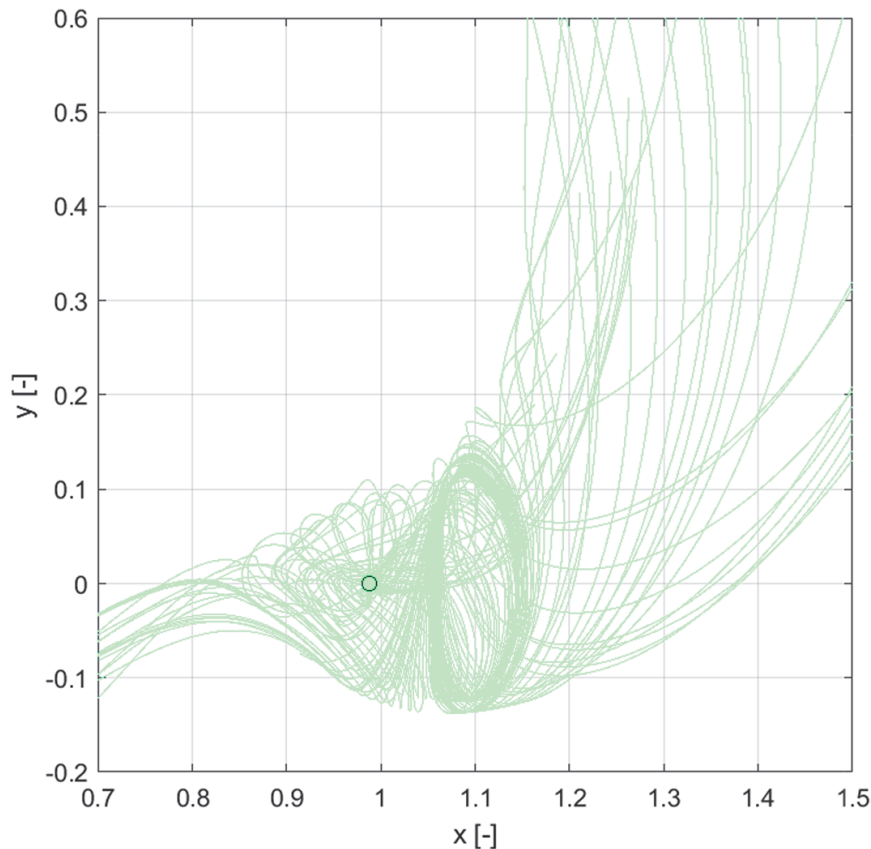**Fig. 12** First guesses generation algorithm

Fig. 13 Sample trajectories propagated backward in time from the target quasi-halo orbit in Earth–Moon synodic frame. Black dot is the Moon. Distances are in Earth–Moon mean distance (Color figure online)

no longer periodic, the minimum stretching direction δ is defined as the direction along which for a given perturbation $\delta\chi_0$ the Euclidian norm of the final perturbation is minimized, i.e.,

$$\delta = \arg\min_{\delta\chi_0} \left\| \delta\chi_f \right\| = \arg\min_{\delta\chi_0} \sqrt{\delta\chi_0^T C \delta\chi_0} \tag{84}$$

where $C = \Phi_{\tau_0,\tau_f}^T \Phi_{\tau_0,\tau_f}$ is the Cauchy–Green tensor, defined by exploiting the STM from τ_0 to τ_f. Starting from Eq. (84), the minimum stretching direction corresponds to the (unit) eigenvector associated to minimum eigenvalues of the Cauchy–Green tensor. The trajectories computed in this way can be seen as an extension of the CRTBP stable manifolds. Figure 13 shows some trajectories propagated backward from the halo orbit.

Table 7 Keplerian elements bounds in MCME2000 for grid search

	i [deg]	Ω [deg]	ω [deg]
Lower bound	0	0	0
Upper bound	180	360	360
Step	20	45	45
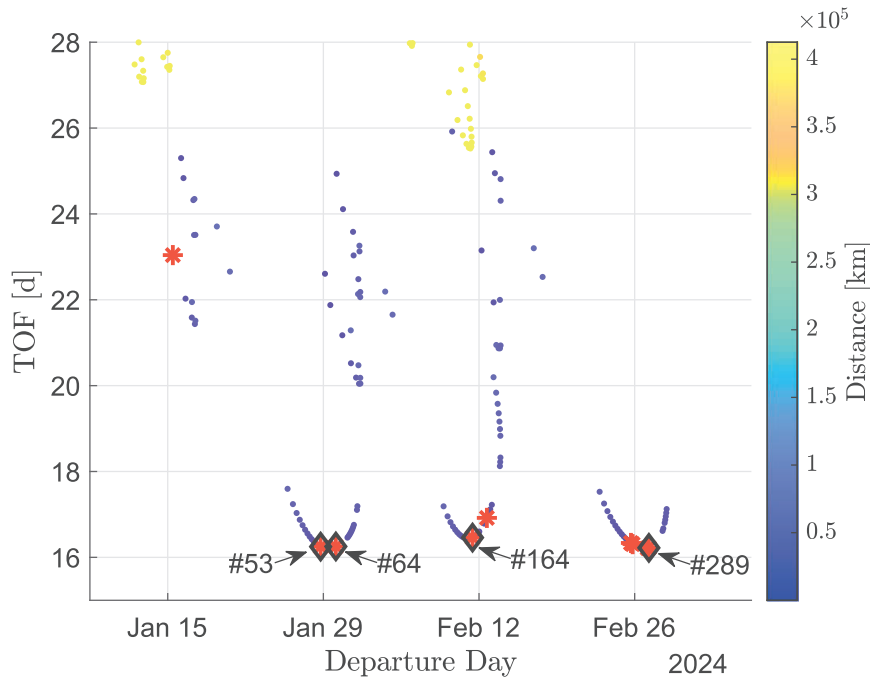
Fig. 14 Pericenter patching process and optimization results. Red asterisks show solutions with $d < 250\,\mathrm{km}$. Black diamonds are optimized solutions with $\Delta v_{\mathrm{SMIM}} < 120\,\mathrm{m/s}$. Numbers refer to solutions from Table 8 (Color figure online)

5.1.2 Grid Generation

At each of the pericenter epochs computed in the backward propagation, a grid of LLO pericenter points is computed, converted to the RPF and then saved. The Keplerian parameters used to build the grid are presented in Table 7.

5.1.3 Patching Process

The patching process patches backward orbits and the LLO grid points at the periselenium. Since by design the pericenter epoch is exactly the same, only the distance between the LLO points and the stable manifold pericenter is used as patching criterion. Figure 14 shows results of the patching process. Starting from the 321 initial guesses, only 9 have a pericenter distance lower than 250 km.

Table 8 First guess optimal solutions whit $\Delta v \leq 120$ m/s

#	Departure date	*ToF* [d]	Δv_{TOT} [m/s]
53	28 JAN 2024	16.62	74.00
64	30 JAN 2024	15.56	67.33
164	11 FEB 2024	16.71	83.11
289	27 FEB 2024	15.92	68.65

5.1.4 Optimization

Most promising solutions coming from the patching process (i.e. with pericenter distance $d < 250$ km) are used to feed an optimization algorithm, whose aim is to find the intersection between the LLO and the operative orbit stable manifold, while minimizing the propellant mass. Thus the objective is to determine the initial and final times, τ_0 and τ_f, the LLO Keplerian parameters $\boldsymbol{\alpha}_0$ and the Δv_{HIM} such that

$$J = \frac{1}{2}\left\|\Delta v_{\mathrm{SMIM}}\right\|^2 + \frac{1}{2}\left\|\Delta v_{\mathrm{HIM}}\right\|^2 \tag{85}$$

is minimized. Moreover, the state is subjected to the constraint

$$\boldsymbol{\rho}_{SM}(\tau_0) - \boldsymbol{\rho}_0(\tau_0, \boldsymbol{\alpha}_0) = 0 \tag{86}$$

where $\boldsymbol{\rho}_{SM}$ is position of the stable manifold at τ_0, i.e. the first three components of backward-integrated flow $\boldsymbol{\varphi}(\boldsymbol{\chi}_f, \tau_f; \tau_0)$, having as initial state $\boldsymbol{\chi}_f = \left(\boldsymbol{\chi}_H(\tau_f) - [\mathbf{0}_3\ \Delta v_{\mathrm{HIM}}]^T\right)$, that is the halo position at τ_f perturbed by the HIM. The stable manifold injection maneuver is computed as

$$\Delta v_{\mathrm{SMIM}} = \boldsymbol{v}_{SM}(\tau_0) - \boldsymbol{v}_0(\tau_0, \boldsymbol{\alpha}_0) \tag{87}$$

with the state on the LLO computed through Eq. (26).

Figure 14 shows the results after the optimization. From the 9 solutions that survived the patching filter, only 4 solutions have a $\Delta v < 120$ m/s, which is compatible with LUMIO requirements. The characteristic of these trajectories are listed in Table 8. Althogh the time of flight is similar for the four optimal solutions, the required propellant amount varied widely, going from only 67.33 to 83.11 m/s, being one third higher than the minimum.

6 Results

The trajectories listed in Table 8 are used as first guesses for Problem 2. The NLP is solved for each of them. The average computational time for the optimization algorithm on a quad-core Intel i7 2.80 GHz processor is about 20 min. Since CUT samples can be propagated forward independently, the runtime can be easily reduced by exploiting parallelization on a multi-core workstation.

Table 9 Integrated approach optimal solutions

#	Δv_D [m/s]	Δv_{D+S}^0 [m/s]	Δv_{D+S}^* [m/s]
53	74.00	102.60	100.14
64	67.33	108.80	108.75
164	83.11	111.88	111.75
289	68.65	125.47	101.91

Subscript D stay for deterministic, while S is stochastic. The superscript 0 indicates the cost function evaluated at the first guess, while the asterisk is used for the value after the optimization

Results are summarized in Table 9. Surprisingly, the solution having the best deterministic value (i.e., #64) is not the one having the best performances when stochastic costs are considered, neither in the non-optimized or in the optimized case, and this can lead to an unnecessary waste of propellant mass. However, this choice will be sub-optimal when the stochastic costs are considered. Indeed, Solution #53 needs less propellant in the first guess comprehensive approach, allowing to save about 6% of fuel. This figure increases to 8% in the *optimized* solution under the integrated approach.

This feature is mainly related to have to possibility to fly a lower dispersion trajectory. Indeed, although the position dispersion (Fig. 15) shows a similar trend for both Solution #53 and #64, the velocity dispersion (Fig. 16) is lower when Solution #53 is considered and this helps the trajectory to have smaller navigation costs. Moreover, a final lower dispersion is beneficial since it allows to satisfy the final constraint with less effort.

The solution #53 has the minimum overall Δv both considering the first guess and the optimized trajectory. Thus, solution #53 would have been selected as the best-performing solution even in the sequential approach. However, solution #289 results show that a great improvement (about the 25%) can be obtained under the stochastic optimization. This feature indicates that, considering a different time-frame or a different operational orbit, it could be possible that the sequential and
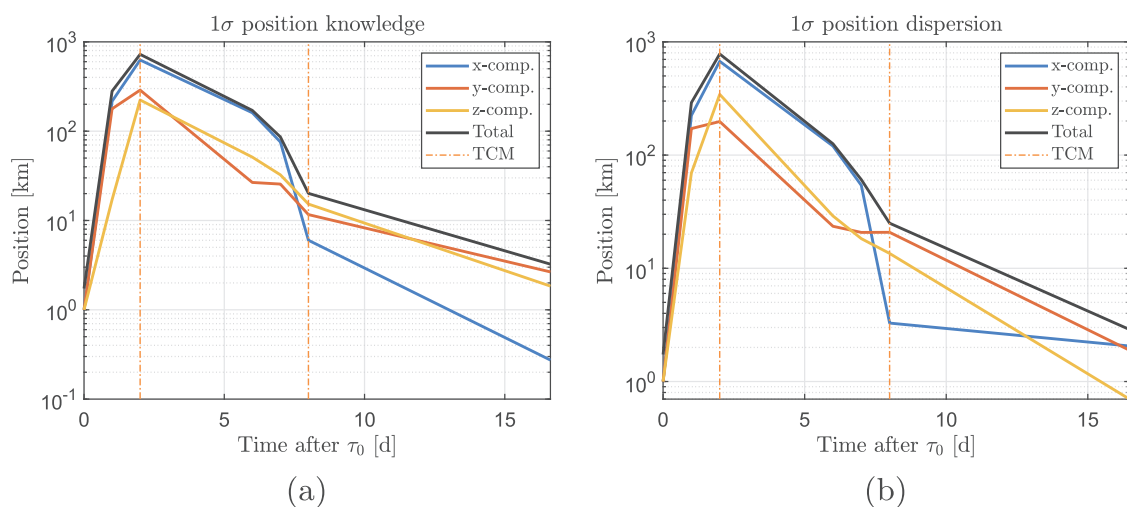
(a)   (b)

Fig. 15 Position 1σ dispersion evolution for the optimized cases: **a** Trajectory #53, **b** Trajectory #64
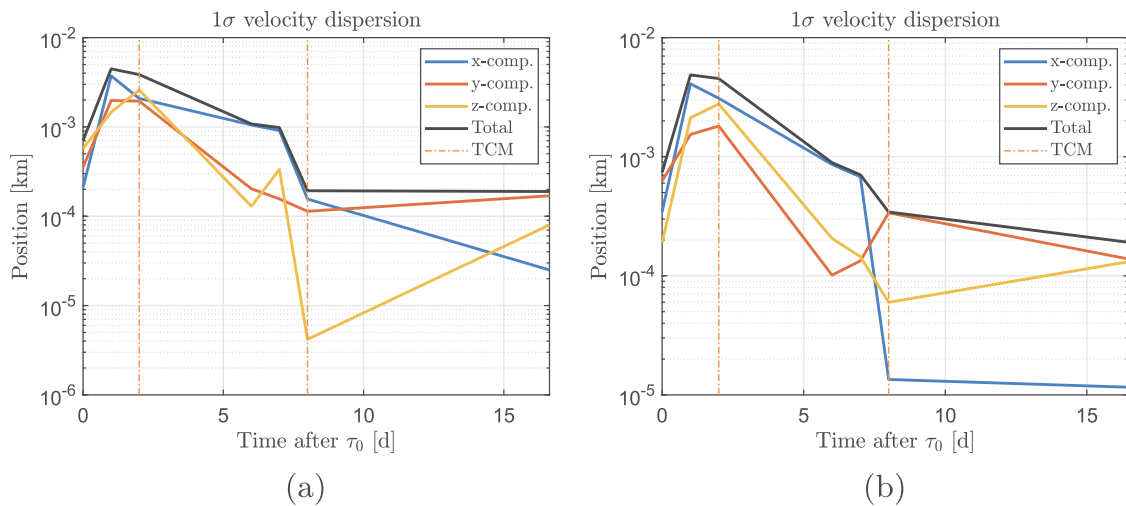
Fig. 16 Velocity 1σ dispersion evolution for the optimized cases: **a** Trajectory #53, **b** Trajectory #64

the integrated approach give different results, leading to a wrong choice of the nominal orbit if the stochastic optimization is not performed.

The decision variables of Problem 2 for Solution #53 are listed in Table 10, while the maneuvers magnitude is provided in Table 11. It can be noted that some propellant is saved also if only the deterministic costs are considered. This behavior relates with the different implementation of final constraint. In fact, in Problem 2 it is enforced with a stochastic formulation. This change modifies the structure of the optimization problem, allowing also a different deterministic solution. Figure 17 shows its trajectory from the LLO to the halo target orbit, with a close-up on the final dispersion. The final dispersion is fully enclosed in the desired 30 km sphere. An analysis of the cumulative distribution function for the two TCMs is given in Fig. 18.

Table 10 Solution #53 decision vector

	Departure date	τ_f [d]	τ_{TCM_1} [d]	τ_{TCM_2} [d]
First guess	28 JAN 2024 21:25	16.62	2.00	8.00
Optimized	28 JAN 2024 21:40	16.63	2.05	8.10
	i_0 [deg]	Ω_0 [deg]	ω_0 [d]	θ_0 [d]
First guess	98.15	34.67	63.96	319.75
Optimized	99.79	38.29	64.20	321.75

Table 11 Solution #53 maneuvers magnitude

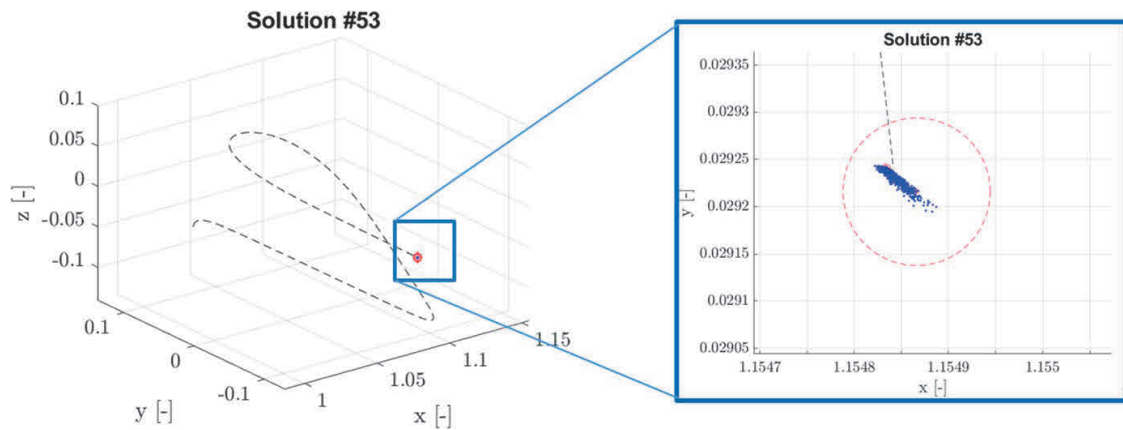|	Δv_{SMIM} [m/s]	Δv_{HIM} [m/s]	$Q(.99, \Delta v_1^s)$	$Q(.99, \Delta v_2^s)$	Total [m/s]
First guess	70.12	3.88	26.12	2.48	102.60
Optimized	68.73	3.50	25.24	2.66	100.14

Fig. 17 Transfer phase trajectory for Solution #53. In the close-up, the final dispersion is showed. The red circle represents the final constraint 30 km sphere (Color figure online)
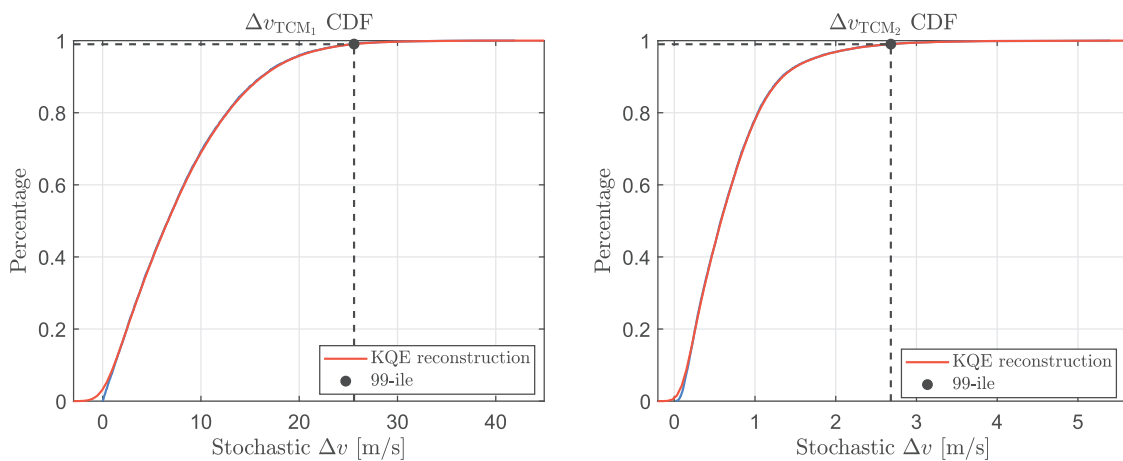

Fig. 18 TCMs cumulative distribution function in Solution #53

For the sake of completeness, the decision variables for Solution #64 are listed in Table 12, while the associated maneuvers are given in Table 13.

7 Conclusions

In this work, an integrated approach for preliminary mission analysis is devised. This technique has the aim to reduce the propellant mass needed to fly a trajectory by embedding in the trajectory design and optimization the navigation assessment and the associated stochastic costs directly in the preliminary mission analysis. This method can be fundamental in future space mission exploiting limited-capability small spacecraft, where high navigation costs may jeopardize the mission feasibility. In order to assess the performances against the traditional technique, the revised approach has been applied in a test case scenario, representing the transfer from a low lunar orbit to the operational halo orbit of the CubeSat LUMIO. For this

Table 12 Solution #64 decision vector

	Departure date	τ_f [d]	τ_{TCM_1} [d]	τ_{TCM_2} [d]
First guess	30 JAN 2024 19:25	15.56	2.00	8.00
Optimized	30 JAN 2024 19:28	15.47	2.03	8.05
	i_0 [deg]	Ω_0 [deg]	ω_0 [d]	θ_0 [d]
First guess	130.06	225.08	60.15	22.58
Optimized	130.05	225.17	60.14	22.72

Table 13 Solution #64 maneuvers magnitude

	Δv_{SMIM} [m/s]	Δv_{HIM} [m/s]	$Q(.99, \Delta v_1^s)$	$Q(.99, \Delta v_2^s)$	Total [m/s]
First guess	65.18	2.15	27.26	14.21	108.80
Optimized	65.17	2.13	27.23	14.22	108.75

scenario, a new technique, using conjugate unscented transformation to compute the polynomial chaos expansion coefficients, labeled PCE-CUT, is devised and used to propagate the uncertainties and estimate both the dispersion and the stochastic costs, while the knowledge analysis is performed by a combination of this technique with an ensemble square-root filter. This method is inserted in the optimization scheme. Four trajectories, coming from a grid search algorithm, are used as educated initial guesses. After the optimization is found that the solution with the best deterministic value is not the one with the minimum overall cost and the 8% of the propellant mass is saved by the integrated approach optimal solution (Fig. 19).
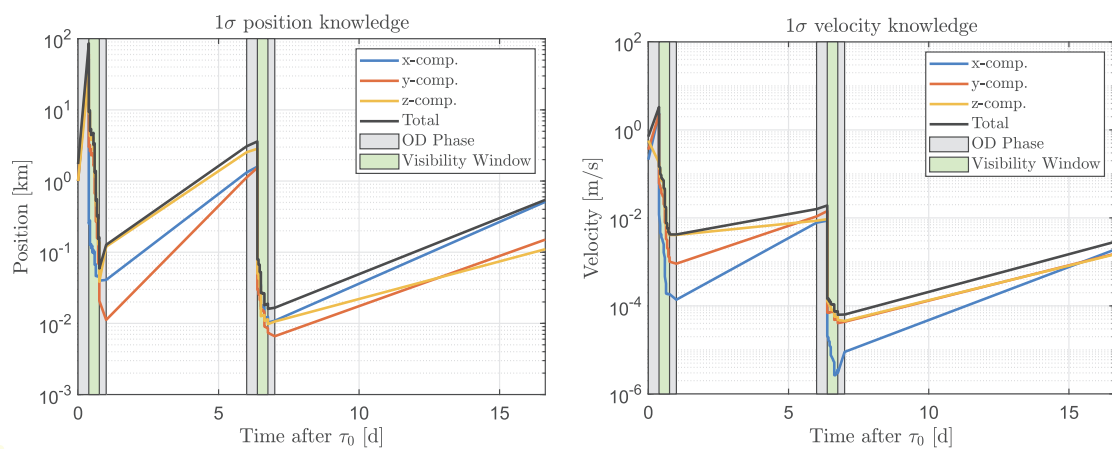

Fig. 19 Knowledge analysis for Solution #53

Appendix A: Derivatives for the Optimization Problem

The decision variable vector is defined as $\mathbf{y} = \left[\tau_0, \tau_f, \tau_{TCM_1}, \tau_{TCM_2}, \boldsymbol{\alpha}_0, \Delta v_{\text{SMIM}}, \Delta v_{\text{TCM}_1}, \Delta v_{\text{TCM}_2}\right]^T$. The chain rule is intensely exploited to simplify both the notation and the computational times. For this reason, some building blocks for the derivatives are firstly introduced.

A.1 Initial State Derivatives

The initial state $\boldsymbol{\chi}_0$ depends only on the initial time τ_0, and $\boldsymbol{\alpha}_0$. However, the numerical computation of these derivatives is cumbersome, since transformation through the SPICE routines is required. For this reason, finite difference is used. Thus

$$\frac{\mathrm{d}\boldsymbol{\chi}_0}{\mathrm{d}\tau_0} = \frac{\boldsymbol{\chi}_0(\tau_0 + \varepsilon, \boldsymbol{\alpha}_0) - \boldsymbol{\chi}_0(\tau_0 - \varepsilon, \boldsymbol{\alpha}_0)}{2\varepsilon} \tag{A1}$$

$$\frac{\mathrm{d}\boldsymbol{\chi}_0}{\mathrm{d}\boldsymbol{\alpha}_0} = \frac{\boldsymbol{\chi}_0(\tau_0, \boldsymbol{\alpha}_0 + \varepsilon) - \boldsymbol{\chi}_0(\tau_0, \boldsymbol{\alpha}_0 - \varepsilon)}{2\varepsilon} \tag{A2}$$

with $\varepsilon = 10^{-6} \max\{1, x\}$, with x a generic variable. Starting from Eq. (39), the initial CUT samples derivative can be evaluated as

$$\frac{\mathrm{d}\boldsymbol{\chi}_q(\tau_0)}{\mathrm{d}\mathbf{y}} = \frac{\mathrm{d}S}{\mathrm{d}\mathbf{y}}\xi_q + \frac{\mathrm{d}\bar{\boldsymbol{\chi}}_0}{\mathrm{d}\mathbf{y}} \tag{A3}$$

with S the Cholesky decomposition of P_0. This derivative has the following non-null components

$$\frac{\mathrm{d}\boldsymbol{\chi}_q(\tau_0)}{\mathrm{d}\tau_0} = \frac{\mathrm{d}\boldsymbol{\chi}_0}{\mathrm{d}\tau_0} \tag{A4}$$

$$\frac{\mathrm{d}\boldsymbol{\chi}_q(\boldsymbol{\alpha}_0)}{\mathrm{d}\tau_0} = \frac{\mathrm{d}\boldsymbol{\chi}_0}{\mathrm{d}\boldsymbol{\alpha}_0} \tag{A5}$$

$$\frac{\mathrm{d}\boldsymbol{\chi}_q(\tau_0)}{\mathrm{d}\Delta v_{\text{SMIM}}} = \mathrm{d}S\xi_q + I_v \tag{A6}$$

with $I_v = \begin{bmatrix} 0_3 & 0_3 \\ 0_3 & I_3 \end{bmatrix}$. Recalling that the initial covariance is function only of Δv_{SMIM} (Eq. (31)), in the remainder of the Section the symbol d will indicate the derivative with respect to Δv_{SMIM}. So

$$dS = \mathcal{T}\left(S^{-T}dP_0 S^{-1}\right)S \tag{A7}$$

with $\mathcal{T}$ being a function that takes the triangular upper part of a matrix and halves the diagonal, i.e.,

$$\mathcal{T}_{ij}(A) = \begin{cases} A_{ij} & j > i \\ \frac{1}{2}A_{ij} & i = j \\ 0 & j < i \end{cases} \tag{A8}$$

A proof of the Cholesky decomposition derivative can be found in Appendix C. Then,

$$dP_0 = \frac{dP_0}{d\Delta v_{\text{SMIM}}} = \begin{bmatrix} 0_3 & 0_3 \\ 0_3 & \frac{dP_{\Delta v}}{d\Delta v_{\text{SMIM}}} \end{bmatrix} \tag{A9}$$

Starting from Eq. (29), it is possible to get

$$\frac{dP_{\Delta v}}{d\Delta v_{\text{SMIM}}} = dJP^s_{\Delta v}J^T + JdP^s_{\Delta v}J^T + JP^s_{\Delta v}dJ^T \tag{A10}$$

where

$$dJ = \begin{bmatrix} -\sin\epsilon\cos\alpha dE - \cos\epsilon\sin\alpha dA \\ -\sin\epsilon\sin\alpha dE + \cos\epsilon\cos\alpha dA \\ \cos\epsilon dE \end{bmatrix}$$

$$\begin{matrix} -d\Delta v\cos\epsilon\sin\alpha + \Delta v\sin\epsilon\sin\alpha dE - \Delta v\cos\epsilon\cos\alpha dA \\ d\Delta v\cos\epsilon\sin\alpha - \Delta v\sin\epsilon\cos\alpha dE - \Delta v\cos\epsilon\sin\alpha dA \\ 0 \end{matrix} \tag{A11}$$

$$\begin{matrix} -d\Delta v\sin\epsilon\cos\alpha - \Delta v\cos\epsilon\cos\alpha dE + \Delta v\sin\epsilon\sin\alpha dA \\ -d\Delta v\sin\epsilon\sin\alpha - \Delta v\cos\epsilon\cos\alpha dE - \Delta v\sin\epsilon\cos\alpha dA \\ d\Delta v\cos\epsilon - \Delta v\sin\epsilon dE \end{matrix} \Bigg]$$

and

$$dP^s_{\Delta v} = \begin{bmatrix} 2u\Delta v & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{A12}$$

In the matrix in Eq. (A11), the missing derivatives can be computed starting from Eqs. (28)

$$d\Delta v = \frac{\left[\Delta v_x,\ \Delta v_y,\ \Delta v_z\right]^T}{\Delta v} \tag{A13a}$$

$$dE = \left[-\frac{\Delta v_z}{\Delta v^2} \frac{\Delta v_x}{\sqrt{\Delta v_x^2 + \Delta v_y^2}}, \; -\frac{\Delta v_z}{\Delta v^2} \frac{\Delta v_y}{\sqrt{\Delta v_x^2 + \Delta v_y^2}}, \; \frac{\sqrt{\Delta v_x^2 + \Delta v_y^2}}{\Delta v^2} \right]^T \quad \text{(A13b)}$$

$$dA = \left[-\frac{\Delta v_y}{\sqrt{\Delta v_x^2 + \Delta v_y^2}}, \; \frac{\Delta v_x}{\sqrt{\Delta v_x^2 + \Delta v_y^2}}, \; 0 \right]^T \quad \text{(A13c)}$$

A.2 OD Phase Derivatives

Then, the derivatives inside the OD process should be constructed. The derivatives with respect to τ_0, τ_{TCM_1} and Δv_{TCM_1} are null, while the one in τ_{TCM_2} and Δv_{TCM_2} are different from zero only for the second OD phase. The derivative procedure is invariant to the considered variable, thus a simplified notation with the symbol d, indicating the differentiation with respect to one of the $\mathbf{y}$ elements, is used. Given the state derivative at a given measurement time $d\chi_k$, the derivative at the following time can be computed as

$$d\chi_{k+1} = \Phi(\tau_k, \tau_{k+1}) d\chi_k \quad \text{(A14)}$$

where the subscript k and $k + 1$ refers to τ_k and τ_{k+1} are two consecutive measurement times and Φ is the STM. The only exception is done by the derivative with respect to τ_0. In fact, in this case, τ_0^{OD} and τ_f^{OD} will vary linearly with τ_0 and thus [8]

$$\begin{cases} \frac{\partial \chi_1}{\partial \tau_0} = \Phi(\tau_0^{OD}, \tau_1)\left(-\mathbf{f}(\chi_0^{OD}, \tau_0^{OD}) + d\chi_0\right) \\ \frac{\partial \chi_{k+1}}{\partial \tau_0} = \Phi(\tau_k, \tau_{k+1}) d\chi_k \\ \frac{\partial \chi_f^{OD}}{\partial \tau_0} = \Phi(\tau_M, \tau_f^{OD}) d\chi_M + \mathbf{f}(\chi_k^{OD}, \tau_f^{OD}) \end{cases} \quad \text{(A15)}$$

with τ_M the last measurement time.

The estimated state PCE coefficients derivatives are

$$d\hat{\mathbf{c}}_\alpha^{k,-} = \sum_{q=1}^{M} d\hat{\chi}_{k,-}^q \psi_\alpha^q \omega_q \quad \text{(A16)}$$

and its statistics derivatives

$$\begin{cases} \mathrm{d}\bar{\hat{\chi}}_{k,-} = \mathrm{d}\hat{\mathbf{c}}_0^{k,-} \\ \mathrm{d}\hat{P}_k^- = \displaystyle\sum_{\substack{\alpha\in\Lambda_{p,d} \\ \alpha\neq\mathbf{0}}} \left(\mathrm{d}\hat{\mathbf{c}}_\alpha^{k,-}\right)\left(\hat{\mathbf{c}}_\alpha^{k,-}\right)^T + \left(\hat{\mathbf{c}}_\alpha^{k,-}\right)\left(\mathrm{d}\hat{\mathbf{c}}_\alpha^{k,-}\right)^T \end{cases} \tag{A17}$$

They are used to compute the derivatives through the EnSRF. Kalman gain can be differentiated as

$$\mathrm{d}S_k = \mathrm{d}H_k \hat{P}_k^- H_k^T + H_k \mathrm{d}\hat{P}_k^- H_k^T + H_k \hat{P}_k^- \mathrm{d}H_k^T \tag{A18}$$

where

$$\mathrm{d}H_k = \begin{bmatrix} \dfrac{\mathrm{d}\boldsymbol{\rho}_k}{\gamma} - \dfrac{\boldsymbol{\gamma}}{\gamma^2}\mathrm{d}\gamma & \mathbf{0}_3 \\[2mm] \dfrac{\mathrm{d}\boldsymbol{\nu}_k}{\gamma} - \dfrac{\boldsymbol{\eta}}{\gamma^2}\mathrm{d}\gamma - \left(\mathrm{d}\boldsymbol{\rho}_k\dfrac{\dot{\gamma}}{\gamma^2} + \boldsymbol{\gamma}\dfrac{\mathrm{d}\dot{\gamma}}{\gamma^2} - 2\boldsymbol{\gamma}\dfrac{\dot{\gamma}\mathrm{d}\gamma}{\gamma^3}\right) & \dfrac{\mathrm{d}\boldsymbol{\rho}_k}{\gamma} - \dfrac{\boldsymbol{\gamma}}{\gamma^2}\mathrm{d}\gamma \end{bmatrix} \tag{A19}$$

with

$$\mathrm{d}\mathbf{z} := \begin{cases} \mathrm{d}\gamma = \dfrac{\boldsymbol{\gamma}^T\mathrm{d}\boldsymbol{\rho}_k}{\gamma} \\[3mm] \mathrm{d}\dot{\gamma} = \left(\dfrac{\boldsymbol{\eta}}{\gamma} - \boldsymbol{\gamma}\dfrac{\dot{\gamma}}{\gamma^2}\right)^T\mathrm{d}\boldsymbol{\rho}_k + \dfrac{\boldsymbol{\gamma}^T\mathrm{d}\boldsymbol{\nu}_k}{\gamma} \end{cases} \tag{A20}$$

Moreover,

$$\mathrm{d}K_k = \mathrm{d}\hat{P}_k^- H_k^T S_k^{-1} + \hat{P}_k^- \mathrm{d}H_k^T S_k^{-1} + \hat{P}_k^- H_k^T \mathrm{d}\left(S_k^{-1}\right) \tag{A21}$$

where [60]

$$\mathrm{d}\left(S_k^{-1}\right) = -S_k^{-1}\mathrm{d}S_k S_k^{-1} \tag{A22}$$

The Kalman gain for the deviations can be retrieved by using

$$\mathrm{d}\tilde{K}_k = \mathrm{d}\hat{P}_k^- H_k^T \sqrt{S_k}^{-T}\left(\sqrt{S_k} + \sqrt{R}\right)^{-1} + \hat{P}_k^- \mathrm{d}H_k^T \sqrt{S_k}^{-T}\left(\sqrt{S_k} + \sqrt{R}\right)^{-1}$$
$$+ \hat{P}_k^- H_k^T \mathrm{d}\left(\sqrt{S_k}^{-1}\right)^T\left(\sqrt{S_k} + \sqrt{R}\right)^{-1} + \hat{P}_k^- H_k^T \sqrt{S_k}^{-T}\mathrm{d}\left(\left(\sqrt{S_k} + \sqrt{R}\right)^{-1}\right) \tag{A23}$$

where

$$\mathrm{d}\left(\sqrt{S_k}^{-1}\right) = -\sqrt{S_k}^{-1}\mathrm{d}\sqrt{S_k}\sqrt{S_k}^{-1} \tag{A24}$$

and

$$\mathrm{d}\left(\left(\sqrt{S_k} + \sqrt{R}\right)^{-1}\right) = -\left(\sqrt{S_k} + \sqrt{R}\right)^{-1}\mathrm{d}\sqrt{S_k}\left(\sqrt{S_k} + \sqrt{R}\right)^{-1} \tag{A25}$$

The matrix square-root derivative can be computed by following the proof contained in Appendix C

$$d\sqrt{S_k} = \text{mat}\left(\left(\sqrt{S_k}^T \oplus \sqrt{S_k}\right)^{-1} \text{vec}\left(dS_k\right)\right) \tag{A26}$$

where mat is the matricization function (similar to the MATLAB® `reshape`) and vec the vectorization. The symbol $\oplus$ indicates the Kronecker sum.

The mean and deviation correction derivatives can be computed as

$$d\bar{\hat{\chi}}_{k,+} = d\bar{\hat{\chi}}_{k,-} + dK_k\left(\mathbf{z}_k - \hat{\mathbf{z}}_k\right) + K_k\left(d\mathbf{z}_k - d\hat{\mathbf{z}}_k\right) \tag{A27}$$

$$d\delta\chi_{k,+}^i = d\delta\chi_{k,-}^i - d\tilde{K}_k H\delta\chi_{k,-}^i + \tilde{K}_k dH\delta\chi_{k,-}^i + \tilde{K}_k H\delta d\chi_{k,-}^i \tag{A28}$$

where

$$d\delta\chi_{k,-}^i = d\hat{\chi}_{k,-}^i - d\bar{\chi}_{k,-} \tag{A29}$$

and

$$d\hat{\chi}_{k,-}^i = \sum_{\alpha \in \Lambda_{p,d}} d\hat{\mathbf{c}}_\alpha^{k,-} \psi_\alpha^i \tag{A30}$$

The n realizations of ξ_i, in practice, are generated at the beginning of the optimization algorithm and keep fixed for all the optimization method steps. Thus ψ_α^i is a constant. Finally, the derivative of the corrected realizations is

$$d\hat{\chi}_{k,+}^i = d\bar{\hat{\chi}}_{k,+} + d\delta\chi_{k,+}^i \tag{A31}$$

While the updated PCE coefficients have derivative

$$d\hat{\mathbf{c}}_\alpha^{k,+} = \left(\mathbf{\Psi}^T\mathbf{\Psi}\right)^{-1}\mathbf{\Psi}dX_k^+ \tag{A32}$$

where dX_k^+ is the collection matrix of $d\hat{\chi}_{k,+}^i$.

As last point, the new cubature points derivative is computed

$$\begin{cases} d\bar{\hat{\chi}}_{k,+} = d\hat{\mathbf{c}}_0^{k,+} \\ d\hat{P}_k^+ = \sum_{\substack{\alpha \in \Lambda_{p,d} \\ \alpha \neq \mathbf{0}}} \left(d\hat{\mathbf{c}}_\alpha^{k,+}\right)\left(\hat{\mathbf{c}}_\alpha^{k,+}\right)^T + \left(\hat{\mathbf{c}}_\alpha^{k,+}\right)\left(d\hat{\mathbf{c}}_\alpha^{k,+}\right)^T \end{cases} \tag{A33}$$

and eventually

$$d\hat{\chi}_{k,+}^q = dS\xi_q + d\bar{\hat{\chi}}_{k,+} \tag{A34}$$

with the derivative of Cholesky decomposition of $\hat{P}_k^+$ computed as in Eq. (A7)

$$dS = \mathcal{T}\left(S^{-T}d\hat{P}_k^+ S^{-1}\right)S \tag{A35}$$

The last step is to compute the derivative of the CUT samples for stochastic cost estimation, i.e.

$$d\tilde{\chi}^q\left(\tau_f^{OD}\right) = d\chi^q\left(\tau_f^{OD}\right) + \left(d\bar{\tilde{\chi}}_f^{OD} - d\chi_f^{*,OD}\right) \tag{A36}$$

where $\chi_f^{*,OD}$ is the nominal trajectory derivative at the end of the OD phase.

A.3 Cut-Off Phase Derivatives

The next step is to evaluate derivatives for the cut-off phase. In order to simplify the notation, a guidance matrix is defined starting from Eq. (65)

$$D = -\left(\Phi_{rv}^T\Phi_{rv} + q\Phi_{vv}^T\Phi_{vv}\right)^{-1}\left(\Phi_{rv}^T\Phi_{rr} + q\Phi_{vv}^T\Phi_{vr}\right) \tag{A37}$$

Then, for both the phases, the derivative of each sample at the final cut-off time (i.e., the TCM time) with respect to Δv_{SMIM} and $\boldsymbol{\alpha}_0$, can be computed as

$$d\chi_{TCM}^q = \Phi\left(\tau_f^{OD}, \tau_{TCM}\right)d\chi^q\left(\tau_f^{OD}\right) + \begin{bmatrix} \mathbf{0}_3 \\ d\Delta v^{s,q} \end{bmatrix} \tag{A38}$$

where d represents the generic partial derivative, and

$$d\Delta v^{s,q} = dD\delta\boldsymbol{\rho}_{TCM} + Dd\delta\boldsymbol{\rho}_{TCM} - d\delta\boldsymbol{v}_{TCM} \tag{A39}$$

The derivative of the state deviation at the maneuver time can be retrieved by

$$\begin{aligned}
\begin{bmatrix} d\delta\boldsymbol{\rho}_{TCM} \\ d\delta\boldsymbol{v}_{TCM} \end{bmatrix} &= \begin{bmatrix} d\tilde{\boldsymbol{\rho}}_{TCM} - d\boldsymbol{\rho}_{TCM}^* \\ d\tilde{\boldsymbol{v}}_{TCM} - d\boldsymbol{v}_{TCM}^* \end{bmatrix} \\
&= \tilde{\Phi}\left(\tau_f^{OD}, \tau_{TCM}\right)d\tilde{\chi}^q\left(\tau_f^{OD}\right) - \Phi^*\left(\tau_f^{OD}, \tau_{TCM}\right)d\chi^*\left(\tau_f^{OD}\right)
\end{aligned} \tag{A40}$$

where Φ^* and $\tilde{\Phi}$ are the TCM of the nominal and the estimated trajectory respectively, while $d\tilde{\chi}\left(\tau_f^{OD}\right)$ is computed in Eq. (A36).

The derivative of the Guidance Matrix is

$$\begin{aligned}
dD = &d\left(Q^{-1}\right)\left(\Phi_{rv}^T\Phi_{rr} + q\Phi_{vv}^T\Phi_{vr}\right) + Q^{-1} \\
&\left(d\Phi_{rv}^T\Phi_{rr} + \Phi_{rv}^T d\Phi_{rr} + q\left(d\Phi_{vv}^T\Phi_{vr} + \Phi_{vv}^T d\Phi_{vr}\right)\right)
\end{aligned} \tag{A41}$$

where $Q = -\left(\Phi_{rv}^T\Phi_{rv} + q\Phi_{vv}^T\Phi_{vv}\right)$. Its derivative can be evaluated as

$$dQ = -\left(d\Phi_{rv}^T\Phi_{rv} + \Phi_{rv}^T d\Phi_{rv} + q\left(d\Phi_{vv}^T\Phi_{vv} + d\Phi_{vv}^T\Phi_{vv}\right)\right) \tag{A42}$$

which is used to compute

$$d\left(Q^{-1}\right) = -Q^{-1}dQQ^{-1} \tag{A43}$$

The derivative of the STM needs the evaluation of the second-order state-transition tensor $\boldsymbol{\Phi}$. Its computation is cumbersome, since it requires to integrate 258 (i.e., $6 + 6^2 + 6^3$) ODEs. For this reason, a finite difference approach is preferred. Thus

$$\boldsymbol{\Phi} = \frac{1}{\varepsilon}\left[\frac{1}{12}\Phi(\chi - 2\varepsilon) - \frac{2}{3}\Phi(\chi - \varepsilon) + \frac{2}{3}\Phi(\chi + \varepsilon) - \frac{1}{12}\Phi(\chi + 2\varepsilon)\right] \tag{A44}$$

where $\varepsilon = \max\{\epsilon_m, 10^{-6}\chi_0\}$, with ϵ_m is the machine epsilon, and $\chi = \chi^*\left(\tau_f^{OD}\right)$. A fourth-order accuracy central finite difference scheme is exploited, because it showed the best balance between precision and computational time.

Finally, the STM derivative expression can be retrieved as

$$d\boldsymbol{\Phi} = \boldsymbol{\Phi}d\chi^*\left(\tau_f^{OD}\right) \tag{A45}$$

The derivative with respect to τ_0 can be computed by following the same steps in Eqs. (A39)–(A45). However, since all the times times are linearly dependent from τ_0, the derivative of the estimated and nominal state (in Eq. (A40)) are

$$\frac{\partial \tilde{\chi}_{TCM}^q}{\partial \tau_0} = -\tilde{\Phi}\left(\tau_f^{OD}, \tau_{TCM}\right)\left(\mathbf{f}\left(\tilde{\chi}^q, \tau_f^{OD}\right) - \frac{\partial \tilde{\chi}^q\left(\tau_f^{OD}\right)}{\partial \tau_0}\right) + \mathbf{f}\left(\tilde{\chi}^q, \tau_{TCM}\right) \tag{A46}$$

and

$$\frac{\partial \chi_{TCM}^*}{\partial \tau_0} = -\Phi^*\left(\tau_f^{OD}, \tau_{TCM}\right)\left(\mathbf{f}\left(\chi^*, \tau_f^{OD}\right) - \frac{\partial \chi^*\left(\tau_f^{OD}\right)}{\partial \tau_0}\right) + \mathbf{f}\left(\chi^*, \tau_{TCM}\right) \tag{A47}$$

The STM derivative inside the differential guidance formula is in turn

$$\frac{\partial \boldsymbol{\Phi}}{\partial \tau_0} = -\boldsymbol{\Phi}A\left(\tau_{TCM_j}\right) + A\left(\tau_{TCM_{j+1}}\right)\boldsymbol{\Phi} - \boldsymbol{\Phi}\left(\mathbf{f}\left(\chi^*, \tau_{TCM_j}\right) - \frac{\partial \chi^*\left(\tau_f^{OD}\right)}{\partial \tau_0}\right) \tag{A48}$$

where A is the dynamics Jacobian as in Eq. (21). The subscript j and $j + 1$ indicates two consecutive maneuver times.

The derivatives with respect to the maneuver time can be computed by following Eqs. (A39)–(A45) again. However,

$$\frac{\partial \tilde{\chi}_{TCM}^q}{\partial \tau_{TCM}} = \mathbf{f}\left(\tilde{\chi}^q, \tau_{TCM}\right) \tag{A49}$$

and

$$\frac{\partial \chi^*_{TCM}}{\partial \tau_{TCM}} = \mathbf{f}\left(\chi^*, \tau_{TCM}\right) \tag{A50}$$

while the STM derivative is

$$\frac{\partial \Phi}{\partial \tau_{TCM}} = -\Phi A\left(\tau_{TCM}\right) \tag{A51}$$

The derivative of the state at the end of the cut-off phase with respect to the next TCM time (i.e., τ_{TCM_2} and τ_f for the first and second phase respectively) can be retrieved from Eqs. (A39–A45), considering that

$$\frac{\partial \tilde{\chi}^q_{TCM}}{\partial \tau_{TCM_{j+1}}} = \mathbf{0}_6 \tag{A52}$$

$$\frac{\partial \chi^*_{TCM}}{\partial \tau_{TCM_{j+1}}} = \mathbf{0}_6 \tag{A53}$$

and

$$\frac{\partial \Phi}{\partial \tau_{TCM_{j+1}}} = A\left(\tau_{TCM_{j+1}}\right)\Phi \tag{A54}$$

with the subscript $j+1$ indicating the next maneuver instant.

A.4 Ballistic Phase Derivatives

Eventually, the derivatives of the ballistic phase final state with respect to $\mathbf{y}$ have to be retrieved. The derivative respect to one of the elements in $\mathbf{y}$ can be computed as

$$\mathrm{d}\chi^q\left(\tau_f^{BP}\right) = \Phi\left(\tau_{TCM}, \tau_f^{BP}\right)\mathrm{d}\chi^q\left(\tau_{TCM}\right) \tag{A55}$$

with τ_f^{BP} being the final time of the ballistic state. Equation (A55) is valid for every $\mathbf{y}$ element, but τ_0

$$\frac{\partial \chi^q\left(\tau_f^{BP}\right)}{\partial \tau_0} = -\Phi\left(\tau_{TCM}, \tau_f^{BP}\right)\left(\frac{\partial \chi^q\left(\tau_{TCM}\right)}{\partial \tau_0} - \mathbf{f}\left(\tilde{\chi}^q, \tau_{TCM}\right)\right) \tag{A56}$$

and τ_f^{BP} (i.e. τ_0^{OD} for the first ballistic phase and τ_f for the second one)

$$\frac{\partial \chi^q\left(\tau_f^{BP}\right)}{\partial \tau_f^{BP}} = \Phi\left(\tau_{TCM}, \tau_f^{BP}\right)\frac{\partial \chi^q\left(\tau_{TCM}\right)}{\partial \tau_f^{BP}} + \mathbf{f}\left(\tilde{\chi}^q, \tau_f^{BP}\right) \tag{A57}$$

A.5 Cost Function Gradient

The cost function gradient is defined as

$$\nabla J = \frac{\partial}{\partial \mathbf{y}}\left(\left\|\Delta v_{\text{SMIM}}\right\| + \left\|\Delta v_{\text{TCM}_1}\right\| + \left\|\Delta v_{\text{TCM}_2}\right\| + \left\|\Delta v_{\text{HIM}}\right\| + \sum_{j=1}^{2} Q\left(0.99, \Delta v_j^s\right) \right) \quad (A58)$$

An analysis of this definition component by component eases the computations.

The only non-null derivative of the SMIM norm with respect to vector $\mathbf{y}$ is [60]

$$\frac{\partial\left\|\Delta v_{\text{SMIM}}\right\|}{\partial \Delta v_{\text{SMIM}}} = \frac{\Delta v_{\text{SMIM}}}{\left\|\Delta v_{\text{SMIM}}\right\|} \quad (A59)$$

In the same way, the derivative of the TCM impulses can be retrieved generically as

$$\frac{\partial\left\|\Delta v_{\text{TCM}_j}\right\|}{\partial \Delta v_{\text{TCM}_j}} = \frac{\Delta v_{\text{TCM}_j}}{\left\|\Delta v_{\text{TCM}_j}\right\|} \quad (A60)$$

Moreover, starting from Eq. (69), it can be found that

$$\frac{\partial\left\|\Delta v_{\text{HIM}}\right\|}{\partial \tau_f} = \frac{\Delta v_{\text{HIM}}}{\left\|\Delta v_{\text{HIM}}\right\|}\left(v'\left(\chi^*, \tau_f\right) - \frac{\mathrm{d}v_\delta}{\mathrm{d}\tau_f} \right) \quad (A61)$$

with the halo velocity derivative computed numerically, while for the other variables, the derivative is

$$\mathrm{d}\left\|\Delta v_{\text{HIM}}\right\| = \frac{\Delta v_{\text{HIM}}}{\left\|\Delta v_{\text{HIM}}\right\|}\mathrm{d}v\left(\tau_f\right) \quad (A62)$$

where the value of $\mathrm{d}v\left(\tau_f\right)$ can be retrieved from Eq. (A57).

Finally, the generic derivative for the stochastic cost should be estimated. Starting from Eq. (77), the generic derivative can be computed as

$$\mathrm{d}Q = \sum_{q=1}^{n} \frac{1}{nh}\frac{1}{\sqrt{2\pi}}\exp\left[-\frac{1}{2}\left(\frac{1}{h}\left(\frac{i}{n} - 0.99\right)\right)^2\right]\mathrm{d}\widetilde{\Delta v}_j^{s,i} \quad (A63)$$

Assuming that the sorting algorithm is invariant with respect to the derivative, we have that

$$\mathrm{d}\widetilde{\Delta v}_j^{s,i} = S\mathrm{d}\left\|\Delta v_j^{s,i}\right\| \quad (A64)$$

where S is the sorting matrix and

$$\mathrm{d}\left\|\Delta v_j^{s,i}\right\| = \frac{\Delta v_j^{s,i}}{\left\|\Delta v_j^{s,i}\right\|} \quad (A65)$$

Recalling the definition of the Monte Carlo samples produced through PCE-CUT, the derivative can be estimated

$$d\Delta v_j^{s,i} = \sum_{\alpha \in \Lambda_{p,d}} dc_\alpha \psi_\alpha^i \tag{A66}$$

with

$$dc_\alpha = \sum_{q=1}^{M} d\Delta v_j^{s,q} \psi_\alpha^q \omega_q \tag{A67}$$

The derivative of the trajectory correction maneuvers CUT samples is evaluated through Eq. (A39).

A.6 Inequality Constraint Jacobian

The inequality constraint in Eq. (72) should be reformulated as

$$c = 0.99 - \hat{F}(d - 30\,\text{km}) < 0 \tag{A68}$$

in order to have the definition usually used in optimization problems. Its Jacobian can be defined as

$$G_c(\mathbf{y}) = \frac{\partial c}{\partial \mathbf{y}} = -\frac{\partial \hat{F}_d(30\,\text{km})}{\partial \mathbf{y}} \tag{A69}$$

Starting from the definition of CDF given by the KDE

$$\hat{F}_d(30\,\text{km}) = \frac{1}{n} \sum_{i=1}^{n} \left[\frac{1}{2} \left(1 + \text{erf} \left(\frac{d_\delta - d_i}{h} \right) \right) \right] \tag{A70}$$

where $d_\delta = 30\,\text{km}$. Its derivative can be evaluated as

$$d\hat{F}_d(d_\delta) = \frac{1}{nh} \sum_{i=1}^{n} \frac{1}{\sqrt{2\pi}} \exp \left(-\frac{1}{2} \left(\frac{d_\delta - d_i}{h} \right)^2 \right) d(d_i) \tag{A71}$$

The derivative of the distance at the final point is

$$d(d_i) = \frac{\boldsymbol{\rho}_i(\tau_f) - \boldsymbol{\rho}_\delta(\tau_f)}{\left\| \boldsymbol{\rho}_i(\tau_f) - \boldsymbol{\rho}_\delta(\tau_f) \right\|} \left(d\boldsymbol{\rho}_i(\tau_f) - d\boldsymbol{\rho}_\delta(\tau_f) \right) \tag{A72}$$

The derivative of the halo distance is always null, except

$$\frac{\partial \boldsymbol{\rho}_\delta(\tau_f)}{\partial \tau_f} = \mathbf{v}_\delta(\tau_f) \tag{A73}$$

while

$$d\boldsymbol{\rho}_i(\tau_f) = \sum_{\boldsymbol{\alpha} \in \Lambda_{p,d}} dc_{\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}^i \tag{A74}$$

with

$$d\mathbf{c}_{\boldsymbol{\alpha}} = \sum_{q=1}^{M} d\boldsymbol{\rho}(\tau_f) \psi_{\boldsymbol{\alpha}}^q \omega_q \tag{A75}$$

where $d\boldsymbol{\rho}(\tau_f)$ is evaluated through Eq. (A55).

Appendix B Cholesky Decomposition Derivative

Starting from the approach followed in [61], a proof for the derivative of the Cholesky decomposition is sought.

Considering a symmetric positive-definite matrix A, the Cholesky decomposition is a factorization such that

$$A = S^T S \tag{B76}$$

Taking an infinitesimal perturbation to this expression, we got

$$dA = dS^T S + S^T dS \tag{B77}$$

Now, left-multiplying for S^{-T} and right-multiplying for S^{-1}, it is found that

$$S^{-T} dA S^{-1} = S^{-T} dS^T + dS S^{-1} \tag{B78}$$

The right-hand side of this equation is made by two terms, one the transpose of the other. Moreover, the first term is upper triangular, while the second one is lower triangular and they share the same diagonal. For this reason, a function $\mathcal{T}$, appropriately defined, can be used to simplify the Eq. (B78). Building $\mathcal{T}$ in order to be a function that takes the triangular upper part of a matrix and halves the diagonal, i.e.

$$\mathcal{T}_{ij}(A) = \begin{cases} A_{ij} & j > i \\ \frac{1}{2} A_{ij} & i = j \\ 0 & j < i \end{cases} \tag{B79}$$

and applying to Eq. (B78), we got

$$\mathcal{T}(S^{-T} dA S^{-1}) = dS S^{-1} \tag{B80}$$

Finally, multiplying by S, the Cholesky decomposition derivative is found as

$$dS = \mathcal{T}(S^{-T} dA S^{-1}) S \tag{B81}$$

$\square$

Appendix C: Matrix Square-Root Derivative

The aim of this section is to compute the derivative of the square root of a matrix, defined as

$$A = \sqrt{A}\sqrt{A} \qquad (C82)$$

By differentiating both the sides, it is possible to find that

$$dA = d\sqrt{A}\sqrt{A} + \sqrt{A}d\sqrt{A} \qquad (C83)$$

Eq. (C83) is a special case of the Sylvester equation. For the Sylvester equation in the form

$$PX + XQ = R \qquad (C84)$$

with P a $(n \times n)$ matrix and Q a m-dimensional square matrix, an analytic solution exists and it can be written as function of the vectorization function vec and the Kronecker product $\otimes$ [62]

$$\left(I_m \otimes P + Q^T \otimes I_n\right) \text{vec}\, X = \text{vec}\, R \qquad (C85)$$

In Eq. (C83), $P = Q = \sqrt{A}$ and $C = dA$, thus its solution is

$$\text{vec}\left(d\sqrt{A}\right) = \left(\sqrt{A}^T \oplus \sqrt{A}\right)^{-1} \text{vec}\,(dA) \qquad (C86)$$

where the properties of the Kronecker sum $\oplus$ are exploited. $\qquad \square$

Data availability Data generated during the current study are available from the corresponding author on request.

Declarations

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

1. Fehse, W.: Automated Rendezvous and Docking of Spacecraft. Cambridge Aerospace Series, vol. 16. Cambridge University Press, Cambridge (2003). https://doi.org/10.1017/CBO9780511543388
2. Franzese, V., Topputo, F.: Optimal beacons selection for deep-space optical navigation. J. Astronaut. Sci. **67**, 1775–1792 (2020). https://doi.org/10.1007/s40295-020-00242-z
3. Poghosyan, A., Golkar, A.: CubeSat evolution: analyzing CubeSat capabilities for conducting science missions. Prog. Aerosp. Sci. **88**, 59–83 (2017). https://doi.org/10.1016/j.paerosci.2016.11.002
4. Walker, R., Binns, D., Bramanti, C., et al.: Deep-space CubeSats: thinking inside the box. Astron. Geophys. **59**(5), 24–30 (2018). https://doi.org/10.1093/astrogeo/aty232
5. Dei Tos, D.A., Rasotto, M., Renk, F., et al.: LISA Pathfinder mission extension: a feasibility analysis. Adv. Space Res. **63**(12), 3863–3883 (2019). https://doi.org/10.1016/j.asr.2019.02.035
6. Trenkel, C., Kemble, S., Bevis, N., et al.: Testing Modified Newtonian Dynamics with LISA Pathfinder. Adv. Space Res. **50**(11), 1570–1580 (2012). https://doi.org/10.1016/j.asr.2012.07.024
7. Fabacher, E., Kemble, S., Trenkel, C., et al.: Multiple Sun–Earth saddle point flybys for LISA Pathfinder. Adv. Space Res. **52**(1), 105–116 (2013). https://doi.org/10.1016/j.asr.2013.02.005
8. Dei Tos, D.A., Topputo, F.: High-fidelity trajectory optimization with application to saddle-point transfers. J. Guid. Control Dyn. **42**(6), 1343–1352 (2019). https://doi.org/10.2514/1.G003838
9. Cipriano, A.M., Dei Tos, D.A., Topputo, F.: Orbit design for LUMIO: the lunar meteoroid impacts observer. Front. Astron. Space Sci. **5**, 29 (2018). https://doi.org/10.3389/fspas.2018.00029
10. Longuski, J.M., Guzmán, J.J., Prussing, J.E.: Optimal Control with Aerospace Applications. Space Technology Library, vol. 32. Springer, New York (2014). https://doi.org/10.1007/978-1-4614-8945-0
11. Ross, I.M.: A Historical Introduction to the Convector Mapping Principle. In: Proceedings of Astrodynamics Specialists Conference, AAS 05-332 (2005)
12. Park, R.S., Scheeres, D.J.: Nonlinear mapping of Gaussian statistics: theory and applications to spacecraft trajectory design. J. Guid. Control Dyn. **29**(6), 1367–1375 (2006). https://doi.org/10.2514/1.20177
13. Di Lizia, P., Armellin, R., Ercoli-Finzi, A., et al.: High-order robust guidance of interplanetary trajectories based on differential algebra. J. Aerosp. Eng. Sci. Appl. **1**(1), 43–57 (2008)
14. Di Lizia, P., Armellin, R., Bernelli-Zazzera, F., et al.: High order optimal control of space trajectories with uncertain boundary conditions. Acta Astronaut. **93**, 217–229 (2014). https://doi.org/10.1016/j.actaastro.2013.07.007
15. Di Lizia, P., Armellin, R., Morselli, A., et al.: High order optimal feedback control of space trajectories with bounded control. Acta Astronaut. **94**(1), 383–394 (2014). https://doi.org/10.1016/j.actaastro.2013.02.011
16. Schumacher, P.W., Jr., Sabol, C., Higginson, C.C., et al.: Uncertain Lambert Problem. J. Guid. Control Dyn. **38**(9), 1573–1584 (2015). https://doi.org/10.2514/1.G001019
17. Zhang, G., Zhou, D., Mortari, D., et al.: Covariance analysis of Lambert's problem via Lagrange's transfer-time formulation. Aerosp. Sci. Technol. **77**, 765–773 (2018). https://doi.org/10.1016/j.ast.2018.03.039
18. Adurthi, N., Majji, M.: Uncertain Lambert problem: a probabilistic approach. J. Astronaut. Sci. **67**, 361–386 (2020). https://doi.org/10.1007/s40295-019-00205-z
19. Armellin, R., Di Lizia, P., Topputo, F., et al.: Gravity assist space pruning based on differential algebra. Celest. Mech. Dyn. Astron. **106**(1), 1–24 (2010). https://doi.org/10.1007/s10569-009-9235-0
20. Hy, Li., Luo, Y.Z., Tang, G.J., et al.: Optimal multi-objective linearized impulsive rendezvous under uncertainty. Acta Astronaut. **66**(3–4), 439–445 (2010). https://doi.org/10.1016/j.actaastro.2009.06.019
21. Yang, Z., Yz, Luo, Zhang, J.: Robust planning of nonlinear rendezvous with uncertainty. J. Guid. Control Dyn. **40**(8), 1954–1967 (2017). https://doi.org/10.2514/1.G002319

22. Balducci, M., Jones, B.A.: Asteroid rendezvous maneuver design considering uncertainty. In: Spaceflight Mechanics 2019. Advances in the Astronautical Sciences, vol. 168, pp 2951–2967 (2019)

23. Xiong, F., Xiong, Y., Xue, B.: Trajectory Optimization under Uncertainty based on Polynomial Chaos Expansion. In: AIAA Guidance, Navigation, and Control Conference, p 1761, (2015) https://doi.org/10.2514/6.2015-1761

24. Greco, C., Campagnola, S., Vasile, M.: Robust space trajectory design using belief optimal control. J. Guid. Control Dyn. **45**(6), 1060–1077 (2022). https://doi.org/10.2514/1.G005704

25. Ozaki, N., Campagnola, S., Funase, R.: Tube stochastic optimal control for nonlinear constrained trajectory optimization problems. J. Guid. Control Dyn. **43**(4), 645–655 (2020). https://doi.org/10.2514/1.G004363

26. Marmo, N., Zavoli, A., Ozaki, N., et al.: A hybrid multiple-shooting approach for covariance control of interplanetary missions with navigation errors. In: 33rd AAS/AIAA Space Flight Mechanics Meeting, Austin, TX (2023)

27. Benedikter, B., Zavoli, A., Wang, Z., et al.: Convex approach to covariance control with application to stochastic low-thrust trajectory optimization. J. Guid. Control, Dyn. **45**(11), 2061–2075 (2022). https://doi.org/10.2514/1.G006806

28. Ridderhof, J., Tsiotras, P., Johnson, B.J.: Stochastic entry guidance. J. Guid. Control Dyn. **45**(2), 320–334 (2022). https://doi.org/10.2514/1.G005964

29. Zavoli, A., Federici, L.: Reinforcement learning for robust trajectory design of interplanetary missions. J. Guid. Control Dyn. **44**(8), 1440–1453 (2021). https://doi.org/10.2514/1.G005794

30. LaFarge, N.B., Miller, D., Howell, K.C., et al.: Autonomous closed-loop guidance using reinforcement learning in a low-thrust, multi-body dynamical environment. Acta Astronaut. **186**, 1–23 (2021). https://doi.org/10.1016/j.actaastro.2021.05.014

31. Oguri, K., Oshima, K., Campagnola, S., et al.: EQUULEUS Trajectory Design. J. Astronaut. Sci. **67**, 950–976 (2020). https://doi.org/10.1007/s40295-019-00206-y

32. Dei Tos, D.A., Chikazawa, T., Kawabata, Y., et al.: EQUULEUS launch window analysis and mission design. In: 31st AAS/AIAA Space Flight Mechanics Meeting, Virtual (2021)

33. Conway, B.A.: A survey of methods available for the numerical optimization of continuous dynamic systems. J. Optim. Theory Appl. **152**(2), 271–306 (2012). https://doi.org/10.1007/s10957-011-9918-z

34. Maybeck, P.S.: Stochastic Models, Estimation, and Control, vol. 3. Academic Press, New York, London (1982)

35. Gardner, J.P., Mather, J.C., Clampin, M., et al.: The James Webb Space Telescope. Space Sci. Rev. **123**(4), 485–606 (2006). https://doi.org/10.1007/s11214-006-8315-7

36. Laureijs, R., Gondoin, P., Duvet, L., et al.: Euclid: ESA's mission to map the geometry of the dark universe. In: Space Telescopes and Instrumentation 2012: Optical, Infrared, and Millimeter Wave, International Society for Optics and Photonics, p 84420T, (2012). https://doi.org/10.1117/12.926496

37. Whitley, R.J., Davis, D.C., Burke, L.M., et al.: Earth–Moon near rectilinear halo and butterfly orbits for lunar surface exploration. In: AAS/AIAA Astrodynamics Specialists Conference, Snowbird, Utah (2018)

38. Cervone, A., Topputo, F., Speretta, S., et al.: LUMIO: a CubeSat for observing and characterizing micro-meteoroid impacts on the Lunar far side. Acta Astronaut. **195**, 309–317 (2022). https://doi.org/10.1016/j.actaastro.2022.03.032

39. Gómez, G., Marcote, M., Masdemont, J.J.: Trajectory correction manoeuvres in the transfer to libration point orbits. Acta Astronaut. **56**(7), 652–669 (2005). https://doi.org/10.1016/j.actaastro.2004.11.005

40. Dei Tos, D.A., Topputo, F.: Trajectory refinement of three-body orbits in the real solar system model. Adv. Space Res. **59**(8), 2117–2132 (2017). https://doi.org/10.1016/j.asr.2017.01.039

41. Acton, C.H., Jr.: Ancillary data services of NASA's Navigation and Ancillary Information Facility. Planet. Space Sci. **44**(1), 65–70 (1996). https://doi.org/10.1016/0032-0633(95)00107-7

42. Acton, C., Bachman, N., Semenov, B., et al.: A look towards the future in the handling of space science mission geometry. Planet. Space Sci. **150**, 9–12 (2018). https://doi.org/10.1016/j.pss.2017.02.013

43. Curtis, H.D.: Orbital Mechanics for Engineering Students. Butterworth-Heinemann, Oxford (2013)

44. Makemson, M.W.: Determination of selenographic positions. Moon **2**(3), 293–308 (1971). https://doi.org/10.1007/BF00561882

45. Yz, Luo, Yang, Z.: A review of uncertainty propagation in orbital mechanics. Prog. Aerosp. Sci. **89**, 23–39 (2017). https://doi.org/10.1016/j.paerosci.2016.12.002

46. Xiu, D.: Numerical methods for stochastic computations: a spectral method approach. Princeton University Press, Princeton (2010)

47. Xiu, D., Karniadakis, G.E.: The Wiener-Askey polynomial chaos for stochastic differential equations. SIAM J. Sci. Comput. **24**(2), 619–644 (2002). https://doi.org/10.1137/S1064827501387826

48. Jones, B.A., Doostan, A., Born, G.H.: Nonlinear propagation of orbit uncertainty using non-intrusive polynomial chaos. J. Guid. Control Dyn. **36**(2), 430–444 (2013). https://doi.org/10.2514/1.57599

49. Adurthi, N., Singla, P.: Conjugate unscented transformation-based approach for accurate conjunction analysis. J. Guid. Control Dyn. **38**(9), 1642–1658 (2015). https://doi.org/10.2514/1.G001027

50. Adurthi, N., Singla, P., Singh, T.: Conjugate unscented transformation: applications to estimation and control. J. Dyn. Sys. Meas. Control. **140**(3), 030907 (2018). https://doi.org/10.1115/1.4037783

51. Stroud, A.H.: Approximate calculation of multiple integrals. Prentice-Hall, Hoboken (1971)

52. Botev, Z.I., Grotowski, J.F., Kroese, D.P., et al.: Kernel density estimation via diffusion. Ann. Stat. **38**(5), 2916–2957 (2010). https://doi.org/10.1214/10-AOS799

53. Silverman, B.W.: Density Estimation for Statistics and Data Analysis. Monographs on Statistics and Applied Probability, vol. 26. CRC Press, Boca Raton (1986)

54. Sheather, S.J., Marron, J.S.: Kernel quantile estimators. J. Am. Stat. Assoc. **85**(410), 410–416 (1990). https://doi.org/10.2307/2289777

55. Yárnoz, D.G., Jehn, R., Croon, M.: Interplanetary navigation along the low-thrust trajectory of BepiColombo. Acta Astronaut. **59**(1–5), 284–293 (2006). https://doi.org/10.1016/j.actaastro.2006.02.028

56. Whitaker, J.S., Hamill, T.M.: Ensemble data assimilation without perturbed observations. Mon. Weather Rev. **130**(7), 1913–1924 (2002). https://doi.org/10.1175/1520-0493(2002)130<1913:EDAWPO>2.0.CO;2

57. Li, J., Xiu, D.: A generalized polynomial chaos based ensemble Kalman filter with high accuracy. J. Comput. Phys. **228**(15), 5454–5469 (2009). https://doi.org/10.1016/j.jcp.2009.04.029

58. Markley, F.L., Crassidis, J.L.: Fundamentals of Spacecraft Attitude Determination and Control. Space Technology Library, vol. 33. Springer, New York (2014). https://doi.org/10.1007/978-1-4939-0802-8

59. Betts, J.T.: Survey of numerical methods for trajectory optimization. J. Guid. Control Dyn. **21**(2), 193–207 (1998). https://doi.org/10.2514/2.4231

60. Petersen, K.B., Pedersen, M.S.: The Matrix Cookbook. Technical University of Denmark, (2012). https://www.math.uwaterloo.ca/~hwolkowi/matrixcookbook.pdf

61. Murray, I.: Differentiation of the Cholesky decomposition. arXiv preprint, arXiv:1602.07527 (2016)

62. Bartels, R.H., Stewart, G.W.: Solution of the matrix equation AX+XB=C [F4]. Commun. ACM **15**(9), 820–826 (1972). https://doi.org/10.1145/361573.361582