POLITECNICO DI MILANO

Department of Aerospace Science and Technology
Doctoral Program in Aerospace Engineering

POLITECNICO
MILANO 1863

# ANALYSIS, DESIGN, AND OPTIMIZATION OF ROBUST TRAJECTORIES FOR LIMITED-CAPABILITY SMALL SATELLITES

Supervisor
**Prof. Francesco TOPPUTO**

Tutor
**Prof. Michéle LAVAGNA**

The Chair of the Doctoral Program
**Prof. Pierangelo MASARATI**

PhD Candidate
**Carmine GIORDANO**
**Matr. 891833**

ACADEMIC YEAR 2020/2021
XXXIII CYCLE

# Abstract

In this thesis a shift in preliminary mission analysis paradigm is formulated, modeled, and assessed. This concept merges in a single step the nominal trajectory optimization with a complete navigation assessment, able both to perform a high-fidelity orbit determination and compute the navigation costs. The aim is to reduce the overall costs removing the natural sub-optimality given by the traditional two-step process, surfing solutions with a lower dispersion and increased stochastic performances. The revised preliminary mission analysis approach is first formulated in a general form and three blocks composing it are identified. Then it is specialized for three different test cases, representing a benchmark for future space missions, and each block is modeled with mathematical means. Eventually, optimal solutions, minimizing the total costs, are sought. A final assessment shows the superiority of the revised paradigm, since it reduces the propellant mass in all the different mission scenarios.

# Sommario

In questa tesi, è formulato, modellato e valutato un cambio di paradigma per l'analisi di missione preliminare. Seguendo questo concetto, l'ottimizzazione della traiettoria nominale è unito a una valutazione completa della navigazione, capace sia di eseguire un algoritmo ad alta fedeltà per la determinazione orbitale sia di calcolare i costi di navigazione. L'obiettivo è di ridurre i costi complessivi eliminando la naturale sub-ottimalità data dal processo sequenziale tradizionale, navigando soluzioni con una dispersione minore e maggiore prestazioni dal punto di vista stocastico. L'approccio rivisto per l'analisi di missione preliminare è prima formulato in una forma generale e i tre blocchi che lo compongono sono individuati. Poi, questa è specializzata per tre scenari diversi, rappresentanti un punto di riferimento per le missioni spaziali del futuro, e ogni blocco è modellato matematicamente. Infine, sono state cercate soluzioni ottime, che minimizzano i costi totali. La valutazione finale mostra la superiorità del nuovo paradigma, dal momento che la massa di propellente necessario è inferiore in tutti gli scenari di missione.

# TABLE OF CONTENTS

# List of Figures

**Figure** **Page**

# List of Tables

*Preneu seient, companys, parlarem de bellesa*
*[...]*
*Uns juren haver-la vist en fórmules matemàtiques*
*Surant en l'harmonia de les esferes*

CHAPTER 1

# INTRODUCTION

Since the beginning of the space era, satellites have always been equipped with chemical propulsion engines, characterized by a high value of thrust and a good control authority. However, in recent times, the space exploration is going in the direction of exploiting small platforms in order to get scientific and technological return at significantly lower costs, employing low-thrust limited-capability spacecraft, needing reduced propellant mass. In this kind of probes, the low control authority poses challenges in maneuvering. For traditional spacecraft, the correction maneuvers are considered to be a minor problem, since changing the trajectory is relatively easy with a single, short burn; on the other hand, adjusting the orbit can be problematic for low-thrust spacecraft, 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 has to take them into account.

A shift in mission analysis paradigm becomes necessary for future space missions.

Novel trajectory design and optimization techniques, naturally embedding navigation features and correction maneuvers needs to be developed and implemented.

## 1.1 Background

Nowadays, nominal trajectories are designed and optimized in order to satisfy scientific requirements as well as to comply with system constraints. However, the nominal path will unlikely be followed by the spacecraft in real-life scenarios due to uncertainty in *dynamic model* (e.g., 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) [30]. Robustness and feasibility assessment of the nominal trajectory against uncertainty are performed a posteriori through a navigation analysis. The navigation assessment has the aim to perform a covariance analysis, in order to compute the achievable state knowledge, and to estimate the correction maneuvers, required to the spacecraft to reach the target. 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, an increasing number of future space exploration mission is foreseen to exploit miniaturized spacecraft, such as SmallSat or CubeSat [65, 81], 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 [33] or limited access to on-ground facilities), 3) and large errors in command actuation (due to low-maturity components). In this case, the classical approach can lead to sub-optimal trajectories, requiring an unnecessarily large amount of propellant. A clear indication of this phenomenon can be found in some studies about mission having the characteristics summarized above. LISA Pathfinder (LPF) proposed mission extension is one example. LPF was a technology demonstrator for the gravitational wave observatory LISA, launched by ESA in 2015. A number of works [22, 29, 78] 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 Modi-

fied Newtonian Dynamics (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 $\Delta v$ 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 [24]. However, if the stochastic cost is taken into account, the $\Delta v$ required to accomplish the mission increases in a sensible way and the feasibility of the mission extension is no longer guaranteed [22]. Figure 1.1(a) shows the solution space of LPF nominal transfers to the Saddle Point, with the most promising solution indicated: in this case, the target is reached with a missing distance of 10 km and a $\Delta v = 0.657$ m/s. On the contrary, Figure 1.1(b) shows the navigation $\Delta v$ distribution for the given trajectory: a value of 4.533 m/s is required 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. In this case, a comprehensive approach, trying to optimize the deterministic and stochastic $\Delta v$ at the same time, can be beneficial, since it will allow not only to reduce the total costs, but also to dampen the mission risks.

A similar behavior can be found also in the LUMIO Phase 0 study [19]. For this mission, a transfer from a Low Lunar Orbit to a halo orbit about Earth–Moon $L_2$ is foreseen (see Chapter 5 for details on the mission). A summary of the $\Delta v$ budget can be found in Table 1.1. In this case, the deterministic cost of the only transfer amounts to 89.97 m/s, while the $3\sigma$ stochastic cost sums up to 97.9 m/s. Hence, the nominal and navigation $\Delta v$s have the same magnitude and, also in this case, a procedure embedding uncertainty in the preliminary mission design can be useful in cutting down the overall mission costs.

[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.

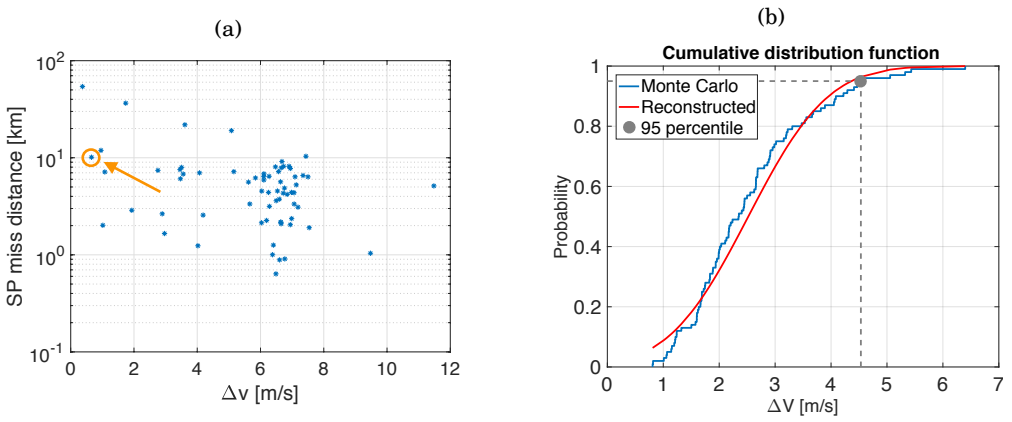



FIGURE 1.1. (a) LISA Pathfinder deterministic $\Delta v$ with respect Saddle Point miss distance. (b) LISA Pathfinder navigation $\Delta v$ cumulative distribution function. Both figures are taken from [22].

TABLE 1.1. Phase 0 LUMIO $\Delta v$ budget (Adapted from [19]).

| **Manuver** | **Deterministic cost [m/s]** | **$3\sigma$ Stochastic cost [m/s] [m/s]** |
|---|---|---|
| SMIM | 89.47 | - |
| TCM1 | - | 73.1 |
| TCM2 | - | 24.8 |
| HIM | 0.5 | - |
| Total | 89.97 | 97.90 |

## 1.2 State of the Art

In the last decades, optimal control and optimization theory have been extensively exploited for the nominal design of space trajectories [52, 66]. 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, Park and Scheeres [63] 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 a paper by Di Lizia et al. [27] by exploiting Taylor differential algebra. An alternative approach, characterizing the stochastic error by means of the first-order variational equations, is presented in [67]. This approach has been extended considering first the explicit partial derivatives of the transfer velocities [91] and later by implementing a derivative free numerical method, exploiting novelties in uncertainty quantification [3]. Uncertain Lambert's problem with differential algebra was also exploited in the gravity assist space pruning algorithm presented in [8].
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 [50], 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 [89]. Also the asteroid rendezvous in a stochastic sense was investigated, considering the state uncertainty both of the spacecraft and the target [10] together with the optimization of correction maneuver under the Lambert's problem conditions.
The uncertainty effect on maneuver execution was investigated in several works by Di Lizia, Armellin et al. [26–28], where optimal control laws, considering perturbations in the nominal states, are obtained, using differential algebra features.
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 Xiong et al. [86], while Greco et al. [42] 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.

The idea of considering both deterministic and stochastic propellant cost has been employed also in the nominal trajectory design for EQUULEUS [62], 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.

## 1.3 Motivations

The design of nominal trajectory and the navigation assessment were quite well investigated in the last years, since they are the bases of the space mission analysis. However, they were studied in two separate phases, leading to possible sub-optimal solutions. This approach has been deemed critical if a miniaturized spacecraft is considered. 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 proposed paradigm shift 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.
To account for this, a novel approach, optimizing the total mission costs while considering trajectory statistical feature, and able to obtain less fuel-expensive and more robust trajectories, has to be investigated.

## 1.4 Research Question

This document has the primary goal to devise and model an integrated brand-new approach, able to consider orbit determination (OD) and trajectory correction maneuvers (TCMs) in preliminary trajectory design. Later, an optimal problem, trying to minimize the overall costs, is set up and solved by means of appropriate optimization schemes. As last point, a comparison between the new technique and the traditional approach has to be carried out. All in all, this work's aim is to answer to the following research question:

> To what extent a holistic approach, embedding stochastic features in the preliminary trajectory optimization, can bring advantages over the traditional methods to compute the nominal spacecraft trajectory?

In order to give a proper answer, some intermediate steps should be covered:

1) Devise a generic integrated approach for the preliminary mission analysis.
2) Select and model its different building blocks.
3) Formulate and solve the optimization problem for this approach.
4) Perform a comparison with the traditional mission analysis approach.

## 1.5 Research outputs

This work introduces some novelties with respect the state of the state, which are used to reply to the research question. It is convenient to summarize these advances in order to easily map them throughout the thesis. Namely they are:

- *an integrated approach*, embedding the navigation assessment in the trajectory optimization (Section 2.1.2) and its applications;
- *a target-driven guidance scheme*, suitable when aiming to targets along the trajectory rather than the nominal trajectory itself (Section 4.2);

- *a brand-new uncertainty quantification technique*, combining polynomial chaos expansion with conjugate unscented transformation, in order to reduce needed samples while improving stability (Section 5.2.1);

Some of these concepts were also exploited for the preliminary mission analysis in LUMIO Phase A study (Chapter 5) and for Hera's Milani CubeSat [31].

## 1.6 Outline of the thesis

In the first part of the thesis, the models and the approach used in the study are appropriately selected, developed and explained. In Chapter 2, the problem of a spacecraft flying in a real-life environment is presented and the model used to describe the guidance and navigation are introduced. Three different scenarios, representative of future deep-space mission, are established. In Chapter 3, methods to propagate uncertainty are discussed and trade offs for each mission scenario are presented. Chapter 4 contains the mathematical foundations for two different guidance schemes, that will be used to estimate the navigation costs under different assumptions and to bring the spacecraft on its nominal path.
In Chapters 5–7, the statement of the optimization problem, the methodology and the results for each of the mission scenario are presented. Real missions, planned in the next future, will be associated to each scenario, in order to ease the optimization problem definition and to assess the aptness and the performances of the revised approach. Eventually the results are presented and the comparison with respect to the traditional approach is performed. Final considerations and possible further developments are given in Chapter 8.

### 1.6.1 Conventions

For clarity's sake, the conventions used in this work will be listed in the remainder. Scalars will be indicated with lower-case letters (e.g., $k$), vectors with bold lower-case letters (e.g., $\mathbf{h}$), and matrices with capital letters (e.g., $A$). Moreover, $(\cdot)^*$ will be used for *nominal* quantities, $(\hat{\cdot})$ for *estimated* quantities, and $(\cdot)$ meant values related to a *real* trajectory. Barred letters (e.g., $\bar{\mathbf{x}}$) will be used for the *mean* value.

CHAPTER 2

# PROBLEM STATEMENT

The preliminary mission analysis problem is introduced in this chapter. A general overview of both the traditional approach and the revised comprehensive strategy are presented, their basic building blocks are identified and a brief description is given. Moreover, three test cases are proposed.

## 2.1 Problem Formulation

### 2.1.1 Sequential Approach

In this thesis, 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 [18, 19, 22]. In this case, the whole procedure is subdivided into two *sequential and independent steps* (Figure 2.1):

1. *Trajectory Design and Optimization*: nominal trajectory, connecting the initial point to the target, while minimizing the propellant mass, is sought (Figure 2.1(a)). 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

$$J = \int_{t_0}^{t_f} \|\mathbf{u}\| \, \mathrm{d}t$$

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 [20], 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 "flyability" in a real scenario is evaluated by simulating the orbit determination process and estimating the trajectory correction maneuvers along the whole mission. Thus, the Navigation Assessment can be split into two (independent) sub-phases:

   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 (Figure 2.1(b)), 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 [63]. 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 orbit determination process is implemented: the knowledge covariance is 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 (Figure 2.1(c)). 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 $\Delta v$, starting from the deviation from the nominal trajectory $\delta x$. 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. In the detailed mission analysis, when higher accuracy is sought, OD can be put in the loop when computing the navigation costs. In this case, the TCM computation will be based on the predicted state deviation, i.e. the deviation of the estimated spacecraft state with respect to the nominal one, and not on the dispersion.

Figure 2.2 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 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. A holistic approach is needed to relieve this effect.

### 2.1.2 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 Figure 2.3 was 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 the 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 Figure 2.3) and initial dispersion (blue ellipse) are given;

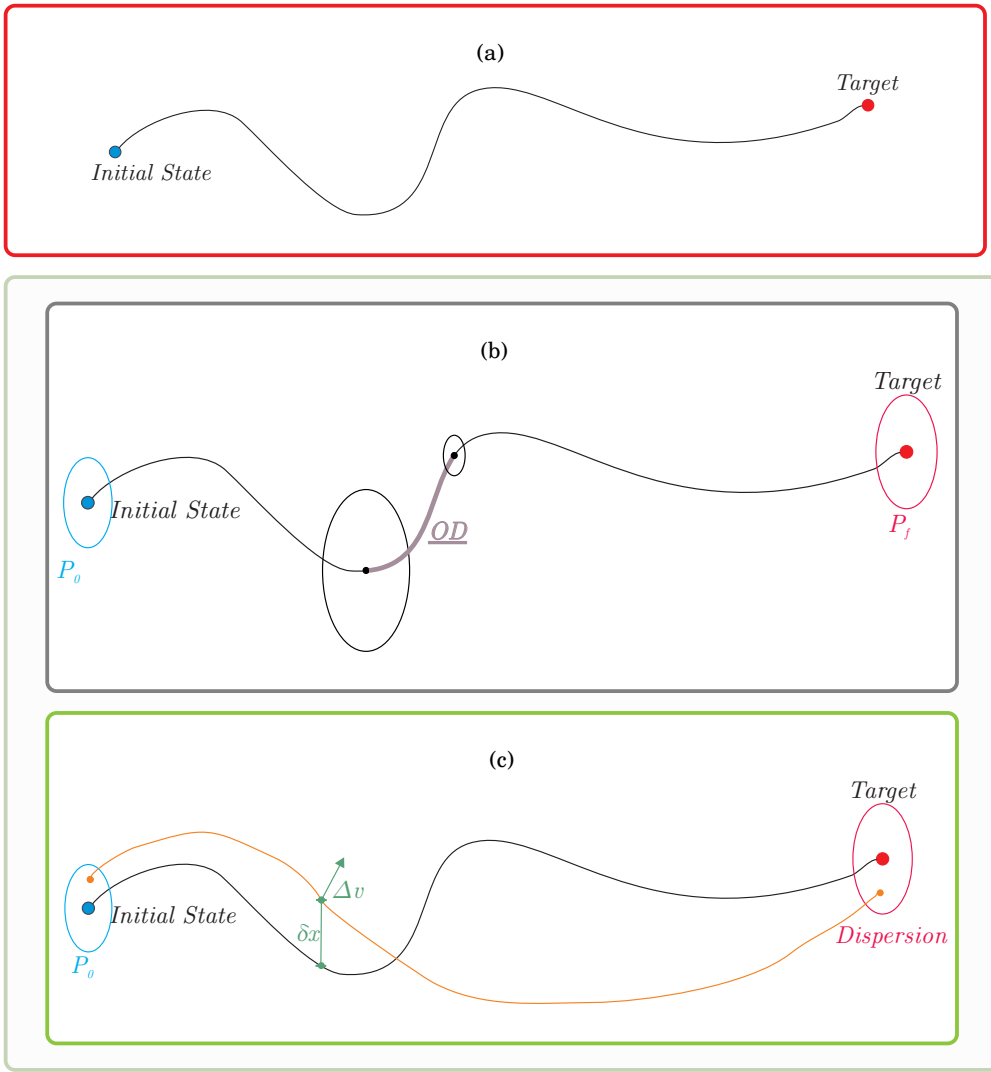



FIGURE 2.1. Traditional approach for preliminary mission analysis: (a) Trajectory design and optimization; (b) Knowledge analysis; (c) Navigation cost evaluation. Nominal trajectory is indicated with a black line, true trajectory as an orange line, OD with a grey thick line. Ellipses represent the instantaneous (b) knowledge (c) or dispersion. Steps (b)-(c) form the navigation assessment.

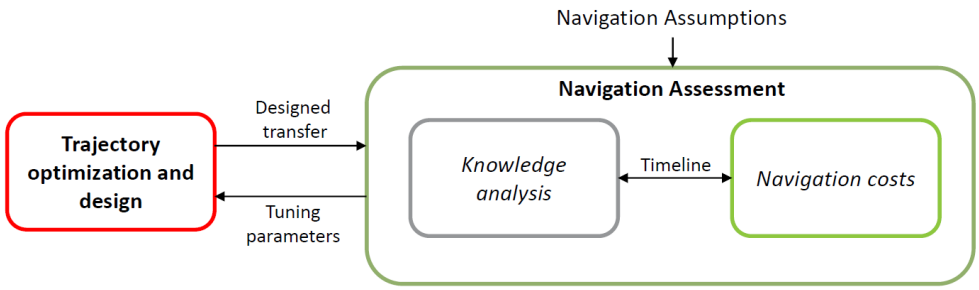



FIGURE 2.2. Traditional approach architecture

2. The initial state is propagated up to the final time, in order to generate the nominal trajectory (black line) and compute the nominal cost;
3. A number of samples in the initial dispersion $\mathbf{x}_0^i$ (orange dot) are generated;
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 orbit determination is performed (gray thick line) to improve the knowledge (black ellipses);
   c) An estimated state (magenta dot) is retrieved at the end of the OD and pushed forward in time, 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 and the final dispersion (red ellipse) can be estimated.
6. The total propellant mass, given by deterministic plus stochastic $\Delta v$ is optimized, while imposing a constraint on the final state.

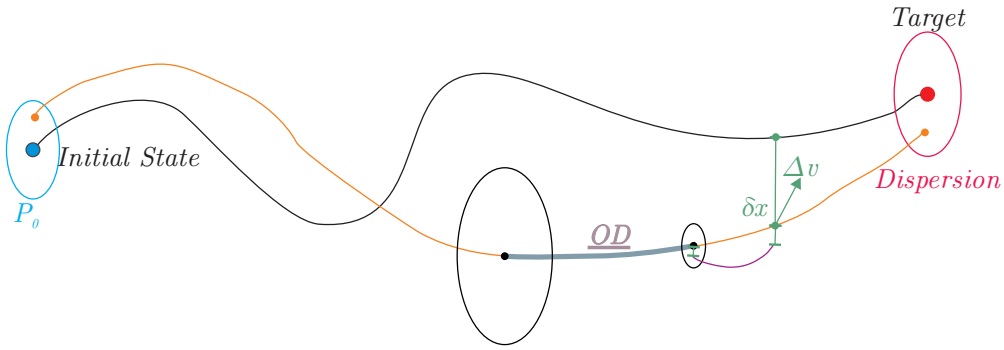



FIGURE 2.3. Revised approach for the preliminary mission analysis. Nominal trajectory is indicated with a black line, a true possible trajectory with an orange line, estimated trajectory with a magenta line. The OD process is the gray thick line. Black ellipses represent the instantaneous knowledge; colored ellipses represent the dispersion.

This general approach can be modified by means of some simplifying assumptions to reduce the computational burden, if needed by the given mission scenario. 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. Moreover, this means that the final point of the nominal trajectory will be unlikely coincident with the target.
Without loss of generality, in this work, the initial state will be always considered distributed as a Gaussian random variable, i.e. $\mathbf{x}_0^i \sim \mathcal{N}(\mathbf{x}_0, P_0^d)$. This assumption is generally valid for a spacecraft trajectory after a first orbit acquisition.
In conclusion, the general fuel-optimal problem of a spacecraft flying in a perturbed environment under the revised approach can be formalized as

**Problem 0** (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

$$J = \int_{t_0}^{t_f} \left\| \mathbf{u}^* \right\| \mathrm{d}t + Q(\Delta v^s) \tag{2.1}$$

with $Q(\Delta v^s)$ a measure of the stochastic cost, is minimized, while the state is subjected to a simplified Itô stochastic differential equation [58]

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t) + \boldsymbol{\omega}(\mathbf{x}, \mathbf{u}, t) \tag{2.2}$$

with $\mathbf{f}$ being the deterministic part of the dynamics and $\boldsymbol{\omega}$ the process noise associated to uncertainty in dynamics and in maneuver execution.
Moreover, the state is subjected to initial constraints

$$\begin{cases} E\left[\mathbf{x}^*(t_0)\right] = \mathbf{x}_0 \\ E\left[(\mathbf{x}^*(t_0) - \mathbf{x}_0)(\mathbf{x}^*(t_0) - \mathbf{x}_0)^T\right] = P_0^d \end{cases} \tag{2.3}$$

and

$$E\left[(\mathbf{x}(t_0) - \mathbf{x}_0)(\mathbf{x}(t_0) - \mathbf{x}_0)^T\right] = P_0^k \tag{2.4}$$

and a final constraint

$$\mathscr{E}\left(\mathbf{x}\left(t_f\right), t_f\right) \subseteq \hat{\mathscr{E}}_\delta\left(t_f\right) \tag{2.5}$$

with $\mathscr{E}$ indicating a generalized uncertainty ellipsoid and $\hat{\mathscr{E}}_\delta$ the desired ellipsoid. The navigation costs are estimated through a guidance law, fed by the orbit determination scheme. It means

$$\Delta v^s = \mathsf{GL}\left(\mathbf{x}^*, \hat{\mathbf{x}}, t_{TCM}\right) \tag{2.6}$$

and

$$\hat{\mathbf{x}}\left(t_f^{OD}\right) = \mathsf{OD}\left(\mathbf{x}, \hat{\mathbf{x}}, t_0^{OD}, t_f^{OD}\right) \tag{2.7}$$

with GL and OD being the Guidance Law and orbit determination procedures respectively, $\hat{\mathbf{x}}$ is the estimated state, $\mathbf{x}$ the real state and $\mathbf{x}^*$ is the nominal state.

## 2.2 Test Case Scenarios

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 produce a good solution for each situation. For this reason, some sample cases are defined and used later in this work to specialize Problem 0, in order to test the revised approach and assess its performances.
Three scenarios are selected as test-bench for the robust preliminary mission analysis, starting from the increasing trends in nowadays space missions:

(1) A spacecraft subject to a strongly nonlinear dynamics, executing impulsive maneuvers (Figure 2.4(a));

(2) A satellite controlled by continuous low-thrust, following a spiral trajectory in a simpler environment (Figure 2.4(b));

(3) A probe performing some fly-bys in a planetary system (Figure 2.4(c)).

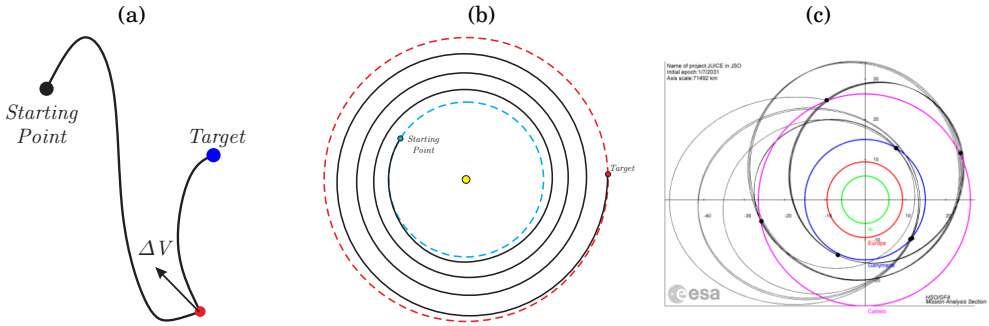



FIGURE 2.4. (a) Scenario 1: Impulsive-engine spacecraft subject to a strongly nonlinear dynamics. (b) Scenario 2: spiral low-thrust trajectory. (c) Scenario 3 (from [17]): Moon tour performing several fly-bys.

Scenario (1) is the prototype of several missions planned in the near future. A clear example is chemical spacecraft flying towards Lagrange points orbits (LPOs), for which there is a increasing interest by space agencies all over the world. Numerous space missions will be launched in the time frame 2021-2025

exploiting such kind of trajectories, using both traditional spacecraft (e.g., James Webb Space Telescope by NASA [37] or ESA's Euclid [49]) and CubeSats (e.g., EQUULEUS by JAXA [62] or LUMIO [19]). The Lunar Gateway, a small station in lunar environment, will orbit a near-rectilinear halo orbit as well and, of course, all its servicing missions are planned to exploit transfer from the Earth [85].
Scenario (2) recalls space missions equipped with electric propulsion. It can be used to describe both LEO-to-GEO orbit raising maneuvers or an interplanetary low-thrust transfer. Examples for this scenario could be the $L_2$-NEO orbit foreseen for the mission M-ARGO, the first interplanetary CubeSat, having the aim to visit a Near-Earth Asteroid in the next 5 years [32] or the mission BepiColombo, launched by ESA in 2018, planned to reach Mercury in 2025 [12]. This kind of transfer is also used by some full-electric geostationary commercial satellites, such as Eutelsat 172B[1], in order to bring the probe to its operative orbit.
Scenario (3) is typical of Moon tours in planetary systems, such as the historical NASA/ESA/ASI Cassini-Huygens mission to Saturn [59] or JUICE by ESA, planned to be launched on 2022 and to reach Jupiter in 2029 [18].

[1] See `https://directory.eoportal.org/web/eoportal/satellite-missions/e/eutelsat-172b`. (Retrieved on December 1, 2020.)

CHAPTER 3

# UNCERTAINTIES PROPAGATION

A necessary step in the holistic approach consists of quantifying the uncertainties as well as their impact on the spacecraft trajectory. This will allow to determine the evolution of both the knowledge and the dispersion and, thus, to estimate the navigation cost. A complete survey on the state-of-the-art techniques used to propagate the uncertainty is needed, focusing particularly on the orbital mechanics field, in order to select an appropriate method.
Uncertainty propagation predicts the state and its stochastic properties at a certain future time, given an initial value and the associated statistical properties, e.g. mean and covariance matrix. Considering a random state vector $\mathbf{x} \in \mathbb{R}^n$, its evolution in time can be described by the Itô stochastic differential equation [58]

$$\mathrm{d}\mathbf{x}(t) = \mathbf{f}(\mathbf{x}, t)\,\mathrm{d}t + G(t)\,\mathrm{d}\boldsymbol{\beta}(t) \tag{3.1}$$

where $\mathbf{f}(\mathbf{x}, t)$ describes the deterministic part of the motion, while the second term of the right-hand side captures the stochastic dynamics, with $\boldsymbol{\beta}(t) \in \mathbb{R}^m$ a stochastic process with zero mean and covariance $Q(t)$ and $G(t) \in \mathbb{R}^{n \times m}$ the diffusion matrix. Given the initial state $\mathbf{x}_0 = \mathbf{x}(t_0)$ and its probability density function (PDF) $p_0 = p(\mathbf{x}_0, t_0)$, the aim of the uncertainty propagation is to determine the

solution of Eq.(3.1) at a given time $t$, i.e., $\mathbf{x}(t) = \boldsymbol{\varphi}(\mathbf{x}_0, t_0; t)$, with $\boldsymbol{\varphi}$ indicating the state flow, together with its statistical characteristics.
This process is usually cumbersome since it requires either to solve the so-called Fokker–Plank equation (FPE), i.e., a partial differential equation describing the evolution of the PDF, or to carry out a computationally expensive Monte Carlo campaign. In order to avoid these drawbacks, approximation methods are needed and alternative techniques were developed in recent times [55].
Uncertainty propagation methods can be grouped in three main categories:

1. Monte Carlo simulation;
2. Linear Methods;
3. Nonlinear Methods.

The latter can be in turn subdivided into three types, according to the approximation choice (Figure 3.1): a) Dynamics-based, i.e., the dynamics is approximated conveniently; b) PDF-based, i.e., the probability density function is estimated through proper techniques; c) Sample-based, i.e., the dynamics is seen as a black-box, and the initial stochastic domain is discretized.

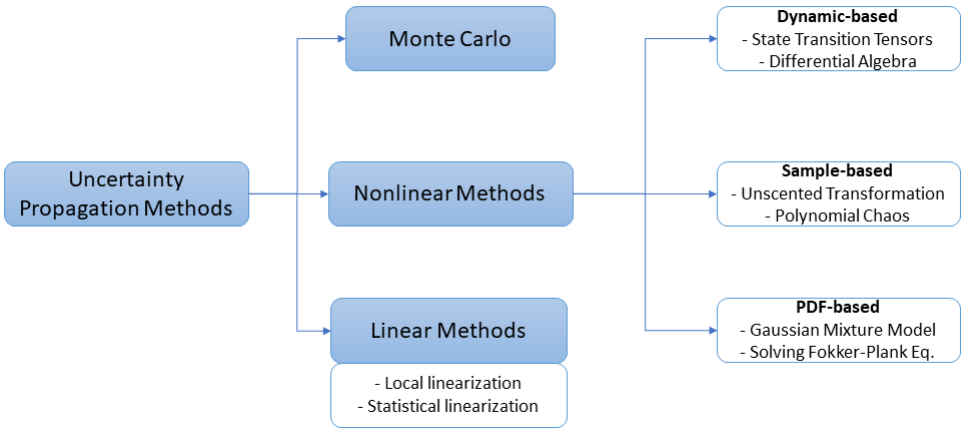



FIGURE 3.1. Overview of uncertainty propagation methods

## 3.1 Methods overview

### 3.1.1 Monte Carlo simulation

Monte Carlo (MC) methods esteem stochastic properties related to the problem of interest through simulations. The stochastic space is properly sampled, taking into account the probability distributions for the diverse stochastic input variables; then, a deterministic computation is done using the input data; at the end, values obtained by simulation are post-processed in order to retrieve stochastic information on the output, such as mean and covariance.
In the uncertainty propagation case, this means, given an initial distribution for the state $p(\mathbf{x}_0, t_0)$, MonteCarlo method defines the mean and the covariance matrix of the propagated state at a given time $t$ as [48]

$$\bar{\mathbf{x}}(t) = \frac{1}{N}\sum_{k=1}^{N} \boldsymbol{\varphi}(t; \mathbf{x}_0^k, t_0) \tag{3.2}$$

$$P(t) = \frac{1}{N-1}\sum_{k=1}^{N} \left[\boldsymbol{\varphi}(t; \mathbf{x}_0^k, t_0) - \bar{\mathbf{x}}(t)\right]\left[\boldsymbol{\varphi}(t; \mathbf{x}_0^k, t_0) - \bar{\mathbf{x}}(t)\right]^T \tag{3.3}$$

where $N$ is the number of random sample, $\boldsymbol{\varphi}(\mathbf{x}_0^k, t_0; t)$ is the propagated $k$-th random point chosen within the initial distribution.
If the problem is well posed, MonteCarlo estimations approach the true values as $N \to \infty$. However, in order to obtain the convergent statistics, the number of realizations tends to increase quite rapidly, MonteCarlo approach is computationally expensive. Several techniques were developed throughout the years in order to reduce the computational burden, such us Latin hypercube sampling (LHS) [45] or GPU-based parallel implementation [9]. Despite its high computational cost, it was used extensively in the past in astrodynamics, since it provides high-precision, nonlinear, non-Gaussian uncertainty propagation with a fast implementation. Due to its reliability and precision, it is also considered a benchmark algorithm to validate alternative uncertainty propagators.

### 3.1.2 Linear Uncertainty Propagation

In order to reduce the computational effort with respect to a MonteCarlo approach and to have fast, but still reliable, results from the uncertainty propagation, tech-

niques based on linearization were developed. These methods are employed while assuming that [55]: 1) the dynamics never drift from the nominal trajectory, so the linearized dynamics is a good approximation for the real orbit; 2) uncertainties are completely described by a Gaussian model, i.e. mean and covariance characterize them completely. Linear methods can be labeled as *local linearization* methods or *statistical linearization* methods.

#### 3.1.2.1 Local Linearization

If the real trajectory, $\mathbf{x}$, remains relatively close to the nominal trajectory $\mathbf{x}^*$ for the time of interest, then it can be expanded in a first-order Taylor's series about the reference trajectory. The dynamics of the state error vector, i.e. the deviation of the real trajectory from the nominal one, can be represented as a system of linear ordinary differential equation (ODE) with time-variable coefficients. As a matter of fact, defining the state deviation vector as

$$\boldsymbol{X}(t) = \mathbf{x}(t) - \mathbf{x}^*(t) \tag{3.4}$$

it follows that

$$\dot{\boldsymbol{X}}(t) = \dot{\mathbf{x}}(t) - \dot{\mathbf{x}}^*(t) \tag{3.5}$$

Since it is possible to represent a general dynamical system as $\dot{\mathbf{x}} = \mathbf{F}(\mathbf{x}, t)$, expanding it in Taylor's series about the reference trajectory and truncating it after the first order terms gives

$$\dot{\mathbf{x}} = \mathbf{F}(\mathbf{x}, t) = \mathbf{F}(\mathbf{x}^*, t) + \left.\frac{\partial \mathbf{F}(\mathbf{x}, t)}{\partial \mathbf{x}(t)}\right|_{\mathbf{x}^*} \left(\mathbf{x}(t) - \mathbf{x}^*(t)\right) + \mathscr{O}\left(\mathbf{x}(t) - \mathbf{x}^*(t)\right) \tag{3.6}$$

Introducing Eq.(3.1) in Eq.(3.6), considering Eq.(3.5), the outcome will be [68]

$$\dot{\boldsymbol{X}}(t) = A(t)\boldsymbol{X}(t) + G(\mathbf{x}, t)\dot{\boldsymbol{\beta}}(t) \tag{3.7}$$

where $A = \left.\frac{\partial \mathbf{F}(\mathbf{x},t)}{\partial \mathbf{x}(t)}\right|_{\mathbf{x}^*}$. An analytical solution can be sought, leading to

$$\boldsymbol{X}(t) = \Phi(t_0, t)\boldsymbol{X}(t_0) + \Gamma(t)\boldsymbol{\beta}(t) \tag{3.8}$$

with $\Phi(t_0, t)$ the state transition matrix (STM) and $\Gamma(t)$ the process transition matrix. From that formulation, exploiting the definition of mean and covariance,

we can characterize the trajectory statistical properties in time as [68]

$$\bar{\mathbf{X}}(t) = \Phi(t_0, t)\bar{\mathbf{X}}(t_0) \tag{3.9}$$

$$P(t) = \Phi(t_0, t)P(t_0)\Phi^T(t_0, t) + G(t)Q(t)G^T(t) \tag{3.10}$$

This technique is very common, since it gives a fast and efficient mean to assess uncertainty propagation under a simple theoretical framework. However, its accuracy is strongly reduced in highly nonlinear environments, long time propagation or large deviation from the nominal solution.

#### 3.1.2.2 Statistical Linearization

The idea beyond the statistical linearization is to approximate the function $\mathbf{f}(\mathbf{x}, t)$ in a linear sense using the expression $\hat{\mathbf{f}}(\mathbf{x}, t) + N\mathbf{x}$, trying to minimize the mean square approximation error, defined as $\mathbf{e}(t) = \mathbf{f}(\mathbf{x}, t) - \hat{\mathbf{f}}(\mathbf{x}, t) - N\mathbf{x}$ in a statistical sense [38]. This results in

$$\hat{\mathbf{f}}(\mathbf{x}, t) = E\left[\mathbf{f}(\mathbf{x}, t)\right] \tag{3.11}$$

$$N = E\left[\mathbf{f}(\mathbf{x}, t)\right]P^{-1} \tag{3.12}$$

where $E[\cdot]$ denotes the expected value.
From this formulation, recalling Eq.(3.1), exploiting the definition of mean and covariance, we can characterize the trajectory statistical properties in time as

$$\dot{\bar{\mathbf{x}}}(t) = \hat{\mathbf{f}}(\mathbf{x}, t) \tag{3.13}$$

$$\dot{P}(t) = N(t)P(t) + P(t)N^T(t) + G(t)Q(t)G^T(t) \tag{3.14}$$

This method is also known as *Covariance Analysis Describing Function Technique* (CADET). The CADET goes past the smoothness of the dynamical function required by the local linearization, since it does not employ Taylor's theorem. However, this method can miss a correct description of highly nonlinear system, because in this case the overall accuracy drops off.

### 3.1.3 Nonlinear uncertainty propagation

In highly nonlinear dynamic systems or problems with expected large deviations with respect the nominal solution, linear propagators fail in representing the uncertainties as function of the time, because assumptions on linear regime are no longer respected [48]. For this reason, in the past two decades several nonlinear methods were developed, giving the possibility to analyze and assess uncertainty propagation in more complex and realistic models.

#### 3.1.3.1 State Transition Tensors

As a natural extension of the local linearization, the nonlinear motion can be described by applying an higher order Taylor series of the solution function in terms of the initial deviation [63]. Mathematically, employing the tensor notation, an expansion up to the $m$-th order can be expressed as

$$\delta\mathbf{x}_i(t) = \sum_{p=1}^{m} \frac{1}{p!} \mathbf{\Phi}_{i,k_1,\ldots,k_p} \delta\mathbf{x}^0_{k_1} \ldots \delta\mathbf{x}^0_{k_p} \tag{3.15}$$

where $\delta\mathbf{x}$ is the deviation of the real trajectory from the nominal one, $\delta\mathbf{x}^0$ is the initial deviation, $\{i,k_p\} \in \{1,\ldots,n\}$ are dummy variables, with $n$ dimension of the state, and $\mathbf{\Phi}_{i,k_1,\ldots,k_p}$ is the state transition tensor (STT), defined as

$$\mathbf{\Phi}_{i,k_1,\ldots,k_p} = \frac{\partial^p \mathbf{x}_i}{\partial\mathbf{x}^0_{k_1} \ldots \partial\mathbf{x}^0_{k_p}} \tag{3.16}$$

computed by numerical integration along the nominal solution.
Using the STT notation, mean and covariance matrix can be computed through their definition, leading to

$$\delta\bar{\mathbf{x}}_i(t) = \sum_{p=1}^{m} \frac{1}{p!} \mathbf{\Phi}_{i,k_1,\ldots,k_p} E\left[\delta\mathbf{x}^0_{k_1} \ldots \delta\mathbf{x}^0_{k_p}\right] \tag{3.17}$$

$$P_{ij}(t) = \left( \sum_{p=1}^{m} \sum_{q=1}^{m} \frac{1}{p!q!} \mathbf{\Phi}_{i,k_1,\ldots,k_p} \mathbf{\Phi}_{j,l_1,\ldots,l_q} E\left[\delta\mathbf{x}^0_{k_1} \ldots \delta\mathbf{x}^0_{k_p} \delta\mathbf{x}^0_{l_1} \ldots \delta\mathbf{x}^0_{l_q}\right] \right) - \delta\bar{\mathbf{x}}_i(t)\delta\bar{\mathbf{x}}_j(t) \tag{3.18}$$

where $\{k_j, l_j\} \in \{1,\ldots,2n\}$.
Differently from the MonteCarlo method, STT technique does not need the generation of random samples, while giving efficient and reliable solutions for mapping

nonlinear uncertainties. However, since high-order differentiation is required, the governing dynamics has to be continuous and differentiable; furthermore, for high-fidelity models, the integration of high-order tensors may be computationally cumbersome, making this method inadequate.

#### 3.1.3.2 Differential Algebra

The STT method requires the computation of increasingly complex partial derivatives. In order to overcome this problem, an accurate nonlinear propagator method based on differential algebra (DA) was developed in the last years[7, 79]. The basis of this method lays in substituting the classical algebra with a new algebra of Taylor polynomials, where any function can be easily expanded in its Taylor series up to an arbitrary order, along with the function evaluation. The DA allows to compute the expansion for the solution flow of a general ODE with respect to the initial conditions. The indipendent uncertain variable $\mathbf{x}$ is defined as a DA variable as $[\mathbf{x}] = \bar{\mathbf{x}} + \delta\mathbf{x}$, then the flow can be expressed as a Taylor expansion with respect the orbit deviation

$$[\boldsymbol{\varphi}] = \mathbf{g}([\mathbf{x}]) = \sum_{p_1+\cdots+p_n \leq m} \mathbf{c}_{p_1,\ldots,p_n} \delta\mathbf{x}_{p_1}^1 \delta\mathbf{x}_{p_n}^n \tag{3.19}$$

where $\mathbf{c}_{p_1,\ldots,p_n}$ are the Taylor coefficients.
The mean and the covariance matrix can be computed as

$$\bar{\mathbf{x}}(t) = E\left\{[\boldsymbol{\varphi}]\right\} = \sum_{p_1+\cdots+p_n \leq m} \mathbf{c}_{p_1,\ldots,p_n} E\left\{\delta\mathbf{x}_{p_1}^1 \ldots \delta\mathbf{x}_{p_n}^n\right\} \tag{3.20}$$

$$P(t) = E\left\{\left([\boldsymbol{\varphi}] - \bar{\mathbf{x}}\right)\left([\boldsymbol{\varphi}] - \bar{\mathbf{x}}\right)^T\right\} = \sum_{\substack{p_1+\cdots+p_n \leq m \\ q_1+\cdots+q_n \leq m}} \mathbf{c}_{p_1,\ldots,p_n} \mathbf{c}_{q_1,\ldots,q_n} E\left\{\delta\mathbf{x}_{p_1+q_1}^1 \ldots \delta\mathbf{x}_{p_n+q_n}^n\right\} \tag{3.21}$$

Starting from the DA approximation, propagation of the uncertainties can be done exploiting MC simulations or high-order Taylor expansions.
The DA allows to expand directly the flow up to an arbitrary order and then using this expansion to non-linearly propagate the uncertainties; this means, that there is no need to integrate the variational equations. However, also in this case, since a Taylor expansion is also performed, the dynamics has to be described by continuous and differentiable equations.

#### 3.1.3.3 Unscented transformation

The unscentend transformation (UT) is based on fact that it may be easier to approximate the probability distribution rather than the nonlinear transformation. This means that the probability distribution at a given time may be approximated by nonlinearly integrating some points $\mathbf{x}_k(t_0),\ k \in \{0,\dots,P\}$, chosen conveniently from the initial distribution [47]. These peculiar points, called *sigma-points*, are chosen deterministically such that they can be used to reconstruct correctly the initial mean $\bar{\mathbf{x}}(t_0)$ and the initial covariance $P(t_0)$. The nonlinear integration is applied to the initial sigma-points in order to obtain transformed sigma-points, $\mathbf{x}_k(t)$, that can be used to retrieve the mean and the covariance at a given time, as

$$\bar{\mathbf{x}}(t) = \sum_{k=1}^{M} \omega_k \mathbf{x}_k(t) \tag{3.22}$$

$$P(t) = \sum_{k=1}^{M} \omega_k \left[\mathbf{x}_k(t) - \bar{\mathbf{x}}(t)\right]\left[\mathbf{x}_k(t) - \bar{\mathbf{x}}(t)\right]^T \tag{3.23}$$

where $\omega_k$ are the weights associated to the $M$ sigma-points.
With respect to the Monte Carlo method, usually only $2n+1$ samples are required for the unscentend transformation in order to obtain a second order approximation of the first two statistical moments. On the other hand, only the first two moments can be correctly approximated.

#### 3.1.3.4 Polynomial Chaos Expansion

If higher moments or the complete PDF are of interest, Polynomial Chaos Expansion (PCE) methods can be used as uncertainty propagators. The input uncertainties and the solution are approximated using a series expansion based on some orthogonal polynomials, thus the approximated solution can be written as [87]

$$\hat{\mathbf{x}}(t,\boldsymbol{\xi}) = \sum_{\boldsymbol{\alpha}\in\Lambda_{p,d}} \mathbf{c}_{\boldsymbol{\alpha}}(t)\psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}) \tag{3.24}$$

where $\Lambda_{p,d}$ is a set of the multi-index of size $d$ and order $p$ defined on nonnegative integers, $\boldsymbol{\xi} = [\xi_1,\dots,\xi_d]$ is the set of input random variables, in which each element $\xi_i$ is an independent indentically distributed (IID) variable. The basis functions

$\{\psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi})\}$ are multidimensional spectral polynomials, orthonormal with respect to the joint probability measure $\rho(\boldsymbol{\xi})$ of the vector $\boldsymbol{\xi}$

$$\int_{\Gamma^d} \psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi})\psi_{\boldsymbol{\beta}}(\boldsymbol{\xi})\rho(\boldsymbol{\xi})\,\mathrm{d}\boldsymbol{\xi} = \delta_{\boldsymbol{\alpha}\boldsymbol{\beta}} \tag{3.25}$$

with $\Gamma^d$ representing the $d$-dimensional hypercube where the random variable $\boldsymbol{\xi}$ are defined and $\delta_{\boldsymbol{\alpha}\boldsymbol{\beta}}$ is the Kronecker delta function. Thus, the basis functions choice depends only on $\rho(\boldsymbol{\xi})$. For instance [88], Hermite polynomials are the basis for normal random variables, while Legendre orthogonal polynomials are bases for the uniform distribution.

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

$$\mathbf{c}_{\boldsymbol{\alpha}}(t) = E\left[\mathbf{x}(t,\cdot)\psi_{\boldsymbol{\alpha}}(\cdot)\right] = \int_{\Gamma^d} \mathbf{x}(t,\boldsymbol{\xi})\psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi})\rho(\boldsymbol{\xi})\mathrm{d}\boldsymbol{\xi} \tag{3.26}$$

The estimation of the Polynomial Chaos Expansion coefficients falls into two categories: *intrusive* and *non-intrusive*.

*Intrusive* methods [87] solve for $\mathbf{c}_{\boldsymbol{\alpha}}(t)$ by performing a Galerkin projection of the governing stochastic equations onto the $\{\psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi})\}$ subspace. Hence, laborious modifications in the dynamic equations are required. In fact, introducing Eq. (3.24) in Eq. (3.1), neglecting the terms related to the process noise, we get

$$\frac{\sum_{\boldsymbol{\alpha}\in\Lambda_{p,d}} \mathrm{d}\mathbf{c}_{\boldsymbol{\alpha}}(t)\psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi})}{\mathrm{d}t} = \mathbf{f}\left(\sum_{\boldsymbol{\alpha}\in\Lambda_{p,d}} \mathbf{c}_{\boldsymbol{\alpha}}(t)\psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}),\ t\right) \tag{3.27}$$

Now, Eq. (3.27) can be Galerkin-like projected on each orthogonal polynomial in the set $\{\psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi})\}$, obtaining [87]

$$\frac{\mathrm{d}\mathbf{c}_{i,\boldsymbol{\alpha}}(t)}{\mathrm{d}t} = \frac{1}{\langle\psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}),\psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi})\rangle}\left\langle \mathbf{f}\left(\sum_{\boldsymbol{\alpha}\in\Lambda_{p,d}} \mathbf{c}_{i,\boldsymbol{\alpha}}(t)\psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}),\ t\right), \psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi})\right\rangle \tag{3.28}$$

with $i \in \{1,2,\ldots,n\}$, where $n$ is the dimension of $\mathbf{x}(t)$. The symbol $\langle\cdot,\cdot\rangle$ is used to denote the inner product in the subspace spanned by $\{\psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi})\}$ with the probability distribution $\rho(\boldsymbol{\xi})$, and $\mathbf{c}_{i,\boldsymbol{\alpha}}(t)$ are the PCE coefficients associated to the $i$-th component of $\mathbf{x}(t)$. The $n$-dimensional system of stochastic ODEs is equivalently

transformed in a $(L \times n)$-dimensional deterministic ODE system, where

$$L = \left|\Lambda_{p,d}\right| = \frac{(p+d)!}{p!d!} \tag{3.29}$$

All the parameters in Eq. (3.28) are deterministic and so the ODE system can be solved by means of standard solver, like Runge-Kutta methods. Thus, once the distribution for $\mathbf{x}(t_0)$ is given, it can be easily propagated in time. However, intrusive PCE requires much effort in modifying existing orbit propagators, as showed by Eq. (3.28). This work can be cumbersome, or even not possible, for high-fidelity, complex, dynamic models.
If process noise is considered, it can be approximated by means of the Karhunen-Loeve (KL) Expansion [54]. This technique reduces the dimensions in representing a random process and allows to frame it in PCE model.
For what concerns *non-intrusive* methods [46], they rely on already existing models and solvers, employing them as a black-box to compute $\mathbf{c}_{\boldsymbol{\alpha}}(t)$. These estimations can be done by 1) Least-Squares Regression or 2) Pseudospectral Collocation on tensor/sparse grid. Both approaches can be summarized as

1. Generate $M$ realizations, denoted as $\boldsymbol{\xi}_i$, of the random input $\boldsymbol{\xi}$. According to the approach, $\boldsymbol{\xi}_i$ can be drawn randomly from the given distribution $\rho(\boldsymbol{\xi})$ (regression) or determinstically chosen according to some grid points in $\Gamma_d$ (pseudospectral collocation);
2. For each of the $M$ samples, compute the initial sample states $\mathbf{x}(t_0, \boldsymbol{\xi}_i)$;
3. Integrate numerically the initial states up to a desired time $t$, given the dynamics as a black-box, to get $\mathbf{x}(t, \boldsymbol{\xi}_i)$;
4. Solve for the PCE coefficients $\mathbf{c}_{\boldsymbol{\alpha}}(t)$, taking into account $\mathbf{x}(t, \boldsymbol{\xi}_i)$ and the method of choice.

In the following paragraphs, two non-intrusive methods will be analyzed.

**Least-Square Regression** The least-square regression is based on a random sampling of the uncertainties, that are propagated independently to obtain the corresponding propagated samples, used to find $\mathbf{c}_{\boldsymbol{\alpha}}(t)$. The PCE coefficients are

computed such that the sum of squares of differences between $\hat{\mathbf{x}}(t,\boldsymbol{\xi})$ and the propagated samples $\mathbf{x}(t,\boldsymbol{\xi})$, at the sample points $\boldsymbol{\xi}_i$, is minimized [46]

$$\mathbf{c}_{\boldsymbol{\alpha}}(t) \simeq \arg\min_{\{\tilde{\mathbf{c}}_{\boldsymbol{\alpha}}(t)\}} \frac{1}{M}\sum_{i=1}^{M}\left(\mathbf{x}(t,\boldsymbol{\xi}_i) - \sum_{\boldsymbol{\alpha}\in\Lambda_{p,d}} \tilde{\mathbf{c}}_{\boldsymbol{\alpha}}(t)\psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}_i)\right)^2 \tag{3.30}$$

The solution of this problem can be conveniently written as

$$\left(\boldsymbol{\Psi}^T\boldsymbol{\Psi}\right)\mathbf{c}(t) = \boldsymbol{\Psi}^T\mathbf{X}(t) \tag{3.31}$$

where $\mathbf{c}(t) \in \mathbb{R}^{P\times n}$ is the least-squares approximation of the PC coefficients, $\mathbf{X}(t) \in \mathbb{R}^{M\times n}$ is the matrix containing the realizations of the vector $\mathbf{x}(t,\boldsymbol{\xi})$ and $\boldsymbol{\Psi} \in \mathbb{R}^{M\times P}$ is the measurement matrix, defined as

$$\boldsymbol{\Psi}[i,j] = \psi_{\alpha_j}(\boldsymbol{\xi}_i) \qquad \text{with} \quad i = 1,\dots,M, \quad j = 1,\dots,L \tag{3.32}$$

The number of samples $M$ needed is not straightforward and it depends basically on the uncertainties, the polynomial basis, and the required accuracy.

**Pseudospectral Collocation** The pseudospectral collocation approach is based on the numerical integration of Eq.(3.26) by collocating $\mathbf{x}(t,\boldsymbol{\xi})$ on certain quadrature nodes defined on $\Gamma_d$ in order to find $\mathbf{c}_{\boldsymbol{\alpha}}(t)$ [46]. This means

$$\mathbf{c}_{\boldsymbol{\alpha}}(t) \asymp \mathcal{Q}\left[\mathbf{x}(t,\cdot)\psi_{\boldsymbol{\alpha}}(\cdot)\right] = \sum_{q=1}^{M}\mathbf{x}(t,\boldsymbol{\xi}_q)\psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}_q)\omega_q \tag{3.33}$$

where $\mathcal{Q}$ is the quadrature integration, $\{\boldsymbol{\xi}_q\}$ is the set of quadrature nodes and $\{\omega_q\}$ are the quadrature weights. Hence only the evaluation of the solution at prescribed quadrature nodes is required, without the necessity of modifying the already existing orbit propagators.

Several quadrature rules are available. The Guassian quadrature rule is one of the most common methods. In this case, the nodes $\boldsymbol{\xi}_q$ and the corresponding weights $\omega_q$ are selected depending on the orthogonal polynomials associated to the probability density function $\tilde{\rho}(\xi_i)$ related to each random input $\xi_i$. For instance, if the input $\xi_i$ is described as a Gaussian random variable, the nodes $\{\xi_i^{q_i}\}$ and the weights $\{\omega_{q_i}\}$ with $q_i = 1,\dots,m_i$ are the zeros of the Hermite polynomial of order

$m_i$ and its quadrature weights. Under this assumption, the quadrature

$$\mathcal{Q}_{m_i}[\mathbf{f}] = \sum_{q_i=1}^{m_i} \mathbf{f}(\xi_i^{q_i})\omega_{q_i} \tag{3.34}$$

is exact, i.e. $\mathcal{Q}_{m_i}[\mathbf{f}] = \int_\Gamma \mathbf{f}(\xi_i)\tilde{\rho}(\xi_i)\mathrm{d}\xi_i$, if $\mathbf{f}(\xi_i) : \Gamma \to \mathbb{R}$ is a $2m_i - 1$-order polynomial. This approach can be easily extended to the ($d$-dimensional) multivariate case, performing a tensor product of the single univariate cases, leading to

$$\mathcal{Q}[\mathbf{f}] = \left(\mathcal{Q}_{m_1} \otimes \cdots \otimes \mathcal{Q}_{m_d}\right)[\mathbf{f}] = \sum_{q_1=1}^{m_1} \cdots \sum_{q_d=1}^{m_d} \mathbf{f}(\xi_1^{q_1}, \ldots, \xi_d^{q_d})\omega_{q_1} \ldots \omega_{q_d} \tag{3.35}$$

In this way a *quadrature on a full tensor product grid* is performed. A total of $M = \prod_{i=1}^{d} m_i$ points for the function evaluation are needed. If the same number $m$ of points along all the dimensions of $\xi_i$ are taken, i.e. $m_1 = \cdots = m_d = m$, the approximation error for the quadrature formula in Eq. (3.35) is in the order of $\mathcal{O}(M^{-(2s-1)/d})$ for a $\mathcal{C}^s$ function $\mathbf{f}(\boldsymbol{\xi})$. 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*. The full tensor product method is unmanageable beyond certain value of $d$.
In order to reduce the dimensionality of the problem, *quadrature on sparse grid* can be used to approximate the integral in Eq. (3.26). The basic idea of sparse grids is to preserve one-dimensional integration properties in a multidimensional setting, reducing the number of grid points, by neglecting higher-order interactions among dimensions. A classical sparse grid quadrature is the Smolyak's formula, given by

$$\mathcal{Q} = \sum_{m-d+1 \le \|\mathbf{m}\|_1 \le m} (-1)^{m-\|\mathbf{m}\|_1} \binom{d-1}{m-\|\mathbf{m}\|_1} \times \left(\mathcal{Q}_{m_1} \otimes \cdots \otimes \mathcal{Q}_{m_d}\right)[f] \tag{3.36}$$

where $m$ is the level of the sparse grid, $\mathbf{m} = (m_1, \ldots, m_d) \in \mathbb{N}_0^d$ and $\|\mathbf{m}\|_1 = \sum_{i=1}^{d} m_i$. The quadrature rule described in Eq. (3.36) exactly integrates polynomials of total order $\sum_{i=1}^{d} m_i \le 2m - 1$, while the full tensor grid quadrature (Eq.(3.35)), considering an grid of order $m$ along all the directions $\xi_i$, exactly integrates polynomials of maximum order $\max_i m_i \le 2m - 1$. The higher degree in accuracy of the full tensor grid is achieved by employing an overly higher number of points with respect to sparse grid quadrature.

Given the estimates of the PCE coefficients $\mathbf{c}_{\boldsymbol{\alpha}}(t)$, the statistics of $\mathbf{x}(t,\boldsymbol{\xi})$ can be approximated by those of $\hat{\mathbf{x}}(t,\boldsymbol{\xi})$ by a Monte Carlo fashion sampling or directly from the coefficients $\mathbf{c}_{\boldsymbol{\alpha}}(t)$. In this last case, the mean is given by [46]

$$\bar{\mathbf{x}}(t) = E[\mathbf{x}(t,\cdot)] \simeq E[\hat{\mathbf{x}}(t,\cdot)] = \int_{\Gamma^d} \left( \sum_{\boldsymbol{\alpha}\in\Lambda_{p,d}} \mathbf{c}_{\boldsymbol{\alpha}}(t)\psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}) \right) \rho(\boldsymbol{\xi})\,\mathrm{d}\boldsymbol{\xi} = \mathbf{c}_0(t) \tag{3.37}$$

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

$$\begin{aligned} P(t) &= E\left[ (\mathbf{x}(t,\cdot) - \bar{\mathbf{x}}(t,\cdot))(\mathbf{x}(t,\cdot) - \bar{\mathbf{x}}(t,\cdot))^T \right] \\ &= \int_{\Gamma^d} \left( \sum_{\substack{\boldsymbol{\alpha}\in\Lambda_{p,d} \\ \boldsymbol{\alpha}\neq\mathbf{0}}} \mathbf{c}_{\boldsymbol{\alpha}}(t)\psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}) \right) \left( \sum_{\substack{\boldsymbol{\beta}\in\Lambda_{p,d} \\ \boldsymbol{\beta}\neq\mathbf{0}}} \mathbf{c}_{\boldsymbol{\beta}}(t)\psi_{\boldsymbol{\beta}}(\boldsymbol{\xi}) \right)^T \rho(\boldsymbol{\xi})\,\mathrm{d}\boldsymbol{\xi} \\ &= \sum_{\substack{\boldsymbol{\alpha}\in\Lambda_{p,d} \\ \boldsymbol{\alpha}\neq\mathbf{0}}} \mathbf{c}_{\boldsymbol{\alpha}}(t)\mathbf{c}_{\boldsymbol{\alpha}}^T(t) E[\psi_{\boldsymbol{\alpha}}^2] = \sum_{\substack{\boldsymbol{\alpha}\in\Lambda_{p,d} \\ \boldsymbol{\alpha}\neq\mathbf{0}}} \mathbf{c}_{\boldsymbol{\alpha}}(t)\mathbf{c}_{\boldsymbol{\alpha}}^T(t) \end{aligned} \tag{3.38}$$

where the orthonormality of the polynomial basis is exploited. A similar procedure can be used to compute higher-order moments.

The PCE technique gives an efficient method for uncertainty propagation, even for non-Gaussian highly nonlinear models. It enjoys an exponential rate of convergence [87], if certain hypotheses are satisfied and provides accurate estimations on higher order moments and the entire PDF. The main drawback is that the number of PCE terms increases exponentially with the dimensionality of the input uncertainties, which leads to the curse of dimensionality.

#### 3.1.3.5 Gaussian Mixture Model

The idea beyond the Gaussian mixture model (GMM) is to approximate a non-Gaussian PDF by a weighted sum of several Gaussian density functions [75], meaning that

$$\hat{p}(t,\mathbf{x}) = \sum_{i=1}^{N} \omega_i p_g(\mathbf{x};\, \bar{\mathbf{x}}_i, P_i) \tag{3.39}$$

where $N$ is the number of Gaussian kernels, with the $i$-th one having a probability density function $p_g$ with mean $\bar{\mathbf{x}}_i$ and covariance $P_i$, and $\omega_i$ being its weight,

subject to the constraints

$$\sum_{i=1}^{N} \omega_i = 1, \qquad \omega_i \geq 0, \quad i = 1, \ldots, N \tag{3.40}$$

These parameters are not easy to compute and to propagate them in time, hence this can be a serious issue for GMM. A simple GMM algorithm splits the initial distribution in a weighted Gaussian mixture and then propagates it in time, obtaining the final mixture, keeping the weights constant. To locate the parameters related to the GMM, an optimization problem minimizing the distance between $p(t,\mathbf{x})$ and $\hat{p}(t,\mathbf{x})$ is solved. Since each component in $\hat{p}(t,\mathbf{x})$ is Gaussian, this means that it is completely described by only the first two moments and, thus, only the mean and the covariance matrix have to be pushed forward in time. This job can be easily done by all the uncertainty propagators considered so far, such as unscentend transformation or state transition tensor.
After the weights and kernels are determined, the mean and covariance matrix can be computed starting from the Gaussian mixtures as [25]

$$\bar{\mathbf{x}}_m = \sum_{i=1}^{N} \omega_i \bar{\mathbf{x}}_i \tag{3.41}$$

$$P_m = \sum_{i=1}^{N} \omega_i (P_i + \bar{\mathbf{x}}_i \bar{\mathbf{x}}_i^T) - \bar{\mathbf{x}}_m \bar{\mathbf{x}}_m^T \tag{3.42}$$

Thus, Gaussian mixture model decouples a large and cumbersome uncertainty propagation in a series of easier problems. However, GMM may require a large number of Gaussian distributions in order to achieve the desired accuracy, leading so to the curse of dimensionality. Furthermore, in order to compute and propagate the weights of the GMM, an optimization problem has to be solved; thus, the computational time strongly increases.

#### 3.1.3.6 Solving Fokker–Plank Equation

Another way to completely propagate uncertainty is to solve the Fokker–Plank equation. In fact, for a given dynamic system satisfying the Itô stochastic differential equation (Eq. (3.1)), the time evolution of a PDF $p(t,\mathbf{x})$ is described by the

Fokker–Plank Equation [34]

$$\frac{\partial p(t,\mathbf{x})}{\partial t} = -\sum_{i=1}^{n} \frac{\partial}{\partial x^i}\left[p(t,\mathbf{x})\mathbf{f}^i(\mathbf{x},t)\right] + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\partial^2}{\partial x^i \partial x^j}\left\{p(t,\mathbf{x})\left[G(t)Q(t)G^T(t)\right]^{ij}\right\} \tag{3.43}$$

The FPE is a partial differential equation (PDE) that satisfies the propagation of a probability density function in time.
However, solving the Fokker–Plank equation is cumbersome for high dimensional dynamics system and it may be computational intensive [74]. For instance, using traditional discretization methods of the domain, if $m$ nodes are used for each dimension, the total number of unknowns for an $n$-dimensional problem is $n^m$. So, the number of variables increases exponentially and making this approach unbearable for high-dimensional problems.
Techniques to reduce the dimensionality have been studied in the last years. Tensor decomposition was extensively analyzed and assessed [73, 74]. Spatial dimensions and time are separated, in order to obtain a simpler series of one dimensional operations, reducing the computational burden. However, this method is unsuitable if there are non-separable terms in the coordinates [74].

## 3.2 Method selection

An overview of the methods with their pros and cons can be found in Table 3.1. For each scenario described in Section 2.2, the most appropriate method should be selected in order to be used in the revise approach. For this reason, for all the three test cases, all the methods are ranked taking into account the following four qualitative criteria:

- *Accuracy*: it considers how accurate is the method in estimating the propagated uncertainty;
- *Feasibility*: it assesses if the hypotheses are compatible with our problem;
- *Computational time*: it gives a qualitative measure of computational burden requested by the method;
- *Suitability*: it measures the suitability inside an optimization algorithm.

| **Method** | **Comments** | **Advantages** | **Drawbacks** |
|---|---|---|---|
| Local linear. | Based on Taylor's series<br>Linearized equations | Easy to compute<br>Simple | Differentiable dynamics<br>Fail for nonlinear systems |
| CADET | Statistical linearization | Easy to compute<br>No assumption on dynamics | Precise knowledge of PDF<br>Fail for nonlinear systems |
| STTs | Higher order Taylor | Analytical<br>Computational efficient | Class $\mathscr{C}^n$ dynamics<br>Inadequate for high-fidelity |
| DA | New algebra<br>MC or Taylor to propagate | Numerical derivatives<br>Computational efficient | Differentiable dynamics |
| UT | Propagation on given points | Use of existing propagators<br>Computational efficient | Mean and covariance only |
| PCE | Polynomial approximation | Use of existing propagators<br>Exponential convergence<br>Apt with non-Gaussian PDF | Curse of dimensionality |
| GMM | PDF as a sum of Gaussians | Gaussian stats to propagate | Curse of dimensionality<br>Optimization required |
| Solve FPE | Integrates PDF with FPE | True evolution of the PDF | Curse of dimensionality<br>High computational costs |
| Monte Carlo | Integration of random samples<br>Well known | High precision<br>Easy to implement | Computational intensive |

TABLE 3.1. Summary of uncertainty propagators

Results from this analysis can be found in Tables 3.2–3.4, with a single table dedicated to each Scenario. A four-color code is used to rank the different methodologies, with *red* marking a criterion with *unacceptable* performances, while *green* indicates a *completely satisfied* criterion.

### 3.2.1 Results

Scenario (1) 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 and, since the uncertainty vector is small, the curse of dimensionality does have a limited impact. Starting from this assumption, the trade-off shows that the *Polynomial Chaos Expansion*, specifically in a non-intrusive method fashion, seems to feature the best balance of criteria. Local linearization and UT show a good ranking and, for this reason, they are tagged as backup, lower fidelity solutions.
Scenario (2) is characterized by a continuous low-thrust engine and its thrust should be considered a stochastic process both in magnitude and direction. Thus, a method able to handle long transfers with stochastic processes is required. Moreover, spacecraft dynamics will not depart significantly from a controlled two-body problem dynamics. In this case, a *local linearization* method is able to describe accurately the stochastic dynamics, while handling easily the broad random-variable space. Stochastic linearization is taken as backup approach.
In Scenario (3), several leveraging fly-bys are foreseen and thus stochastic dimensions increase exponentially and make the revised approach unbearable. For this reason, only a small leg of the Moon tour, including only few close approaches, is considered. In this case, the random dimensions is still limited. However, several close passages can significantly spread the trajectory. Thus, linear methods are considered inappropriate for this test case. In conclusion, *Polynomial Chaos Expansion* is deemed to be the best feasible option, while unscentend transformation is identified as the backup solution.

**Scenario 1**

| | Accuracy | Feasibility | CPU time | Suitability |
|---|---|---|---|---|
| Local linear. | Very low (Nonlinear environment) | Low (Large deviations from initial orbit) | Fast | High (Easy Jacobian) |
| CADET | Very low (Nonlinear environment) | Low (Large deviations from initial orbit) | Mildly fast (Statistical ODE) | Mildly High (Jacobian may be difficult) |
| STTs | Mildly High | Very low (No possibility to introduce process noise) | Slow (High-fidelity model leads to complexity) | Low (Jacobian may be difficult) |
| DA | Mildly High | Very low (No possibility to introduce process noise) | Fast | Low (Jacobian may be difficult) |
| UT | Low (Only mean and covariance are propagated) | Low (Non-Gaussian final state) | Mildly fast | High (Easy Jacobian) |
| PCE | High | High | Slow (Curse of dimensionality) | High (Easy Jacobian - NI Collocation) |
| GMM | High | High | Slow (Curse of dimensionality) | Very low (Nested optimization) |
| Solve FPE | High | High | Very low (Cumbersome to solve) | Very low (PDE to solve) |
| Monte Carlo | High | High | Very low (Slow convergence) | Very low (No Jacobian) |

TABLE 3.2. Scenario 1 - Ranking matrix for uncertainty methods

**Scenario 2**

| | Accuracy | Feasibility | CPU time | Suitability |
|---|---|---|---|---|
| Local linear. | Low (Nonlinear environment) | High (Stochastic control) | Fast | High (Easy Jacobian) |
| CADET | Low (Nonlinear environment) | High (Stochastic control) | Mildly fast (Statistical ODE) | Mildly High (Jacobian may be difficult) |
| STTs | Mildly High | Very low (No possibility to introduce low thrust) | Slow (High-fidelity model leads to complexity) | Low (Jacobian may be difficult) |
| DA | Mildly High | Very low (No possibility to introduce low thrust) | Fast | Low (Jacobian may be difficult) |
| UT | Low (Only mean and covariance are propagated) | Low (Non-Gaussian final state) | Mildly fast | High (Easy Jacobian) |
| PCE | High | Low (Stochastic control difficult to handle) | Slow (Curse of dimensionality) | High (Easy Jacobian - NI Collocation) |
| GMM | High | High | Slow (Curse of dimensionality) | Very low (Nested optimization) |
| Solve FPE | High | High | Very low (Cumbersome to solve) | Very low (PDE to solve) |
| Monte Carlo | High | High | Very low (Slow convergence) | Very low (No Jacobian) |

TABLE 3.3. Scenario 2 - Ranking matrix for uncertainty methods

**Scenario 3**

| | Accuracy | Feasibility | CPU time | Suitability |
|---|---|---|---|---|
| Local linear. | Very low (Nonlinear environment) | Low (Large deviations from initial orbit) | Fast | High (Easy Jacobian) |
| CADET | Very low (Nonlinear environment) | Low (Large deviations from initial orbit) | Mildly fast (Statistical ODE) | Mildly High (Jacobian may be difficult) |
| STTs | Mildly High | Very low (No possibility to introduce process noise) | Mildly fast | Low (Jacobian may be difficult) |
| DA | Mildly High | Very low (No possibility to introduce process noise) | Fast | Low (Jacobian may be difficult) |
| UT | Low (Only mean and covariance are propagated) | Mildly fast (Non-Gaussian final state) | Mildly fast | High (Easy Jacobian) |
| PCE | High | High | Slow (Curse of dimensionality) | High (Easy Jacobian - NI Collocation) |
| GMM | High | High | Slow (Curse of dimensionality) | Very low (Nested optimization) |
| Solve FPE | High | High | Very low (Cumbersome to solve) | Very low (PDE to solve) |
| Monte Carlo | High | High | Very low (Slow convergence) | Very low (No Jacobian) |

TABLE 3.4. Scenario 3 - Ranking matrix for uncertainty methods

CHAPTER 4

# GUIDANCE METHODS

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 TCMs, 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, based both on the maneuver execution time and on the maneuver computation itself. Indeed, the control thrust can be provided either when a certain quantity (e.g. the state knowledge) exceeds a threshold or at prescribed time intervals, provided by the on-ground flight dynamics team. Moreover, the impulse vector can be computed to control the full state, or just some components, or the unstable directions [44], by an application of generalized Zero-Effort-Miss/Zero-Effort-Velocity feedback guidance algorithm [43], using sliding control [35] or even exploiting state-of-the-art machine learning [36]. 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. Two diverse methods are exploited: the well-known differential guidance and a novel approach, called *target guidance*. Details about these two algorithms are given in the remainder of this chapter.

## 4.1 Differential Guidance

differential guidance (DG) is a commonly used guidance method for deep-space missions [22, 57]. DG aims at canceling the final state deviation using two maneuvers, one at beginning and the other at the end of the considered leg. Since the time interval between navigation maneuvers is relatively short and, thus, the real trajectory does not significantly drift from the nominal one, first-order (linear) approximation can be used to relate the initial and final deviations. Hence

$$\begin{bmatrix} \delta\mathbf{r}_{j+1} \\ \delta\mathbf{v}_{j+1} \end{bmatrix} = \begin{bmatrix} \Phi_{rr} & \Phi_{rv} \\ \Phi_{vr} & \Phi_{vv} \end{bmatrix} \left( \begin{bmatrix} \delta\mathbf{r}_j \\ \delta\mathbf{v}_j \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \Delta\mathbf{v}_j^s \end{bmatrix} \right) + \begin{bmatrix} \mathbf{0} \\ \Delta\mathbf{v}_{j+1}^s \end{bmatrix} \tag{4.1}$$

where $\Phi_{rr}$, $\Phi_{rv}$, $\Phi_{vr}$, and $\Phi_{vv}$ are the 3-by-3 blocks of $\Phi(t_j, t_{j+1})$, i.e., the STM associated to the nominal trajectory from $t_j$ and $t_{j+1}$, and $\Delta\mathbf{v}_j^s$ and $\Delta\mathbf{v}_{j+1}^s$ are the TCMs at the same time instants. Epochs $t_j$ and $t_{j+1}$ are used to indicate two consecutive TCM times, 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

$$\begin{cases} \delta\mathbf{r} = \hat{\mathbf{r}} - \mathbf{r}^* \\ \delta\mathbf{v} = \hat{\mathbf{v}} - \mathbf{v}^* \end{cases} \tag{4.2}$$

Otherwise, the real trajectory is considered, thus

$$\begin{cases} \delta\mathbf{r} = \mathbf{r} - \mathbf{r}^* \\ \delta\mathbf{v} = \mathbf{v} - \mathbf{v}^* \end{cases} \tag{4.3}$$

Usually, the second control maneuver $\Delta\mathbf{v}_{j+1}$ 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. Then, the navigation maneuver $\Delta\mathbf{v}_j$ can be determined such that the final deviation is minimized, i.e.

$$J = \left\|\delta\mathbf{r}_{j+1}\right\|^2 + q\left\|\delta\mathbf{v}_{j+1}\right\|^2 \tag{4.4}$$

where $q$ is a parameter used either to adjust dimensions or the change the guidance algorithm behavior, favoring position deviation at the expense of velocity deviation and vice versa. Applying the variations analysis to Eq. (4.4), considering Eq. (4.1) as a linear constraint, the TCM can be computed as

$$\Delta\mathbf{v}_j^s = -\left(\Phi_{rv}^T\Phi_{rv} + q\Phi_{vv}^T\Phi_{vv}\right)^{-1}\left(\Phi_{rv}^T\Phi_{rr} + q\Phi_{vv}^T\Phi_{vr}\right)\delta\mathbf{r}_j - \delta\mathbf{v}_j \tag{4.5}$$

The control impulse in Eq. (4.5) is applied at each TCM time.

## 4.2 Target Guidance

In some mission scenarios, it can be of paramount importance to meet targets along the trajectory, rather than catching up the nominal state at each correction maneuver epoch. In these cases, algorithms pointing at canceling out deviations at maneuver points can unreasonably increase the navigation costs. This phenomenon can happen in a profile with several back-to-back fly-bys, where a precise close approach altitude is strongly required, or in case of close proximity operations about minor bodies, where scientific objectives can be more stringent than the exact way-point passage. Thus, a guidance law able to fulfill prescribed requirements by giving control impulse at some given times is needed. For this reason, a novel concept, labeled *target guidance (TG)* has been devised.
Considering Figure 4.1, exploiting a linearized approach, the state at the first TCM time $t_1$, can be computed as

$$\mathbf{x}_1 = \mathbf{x}_1^* + \Phi(t_0, t_1)\delta\mathbf{x}_0 \tag{4.6}$$

where indexes correlate to TCM epochs, $\Phi(t_0, t_1)$ is the STM from $t_0$ to $t_1$, and $\delta\mathbf{x}_0$ is the state deviation at the initial time $t_0$, defined conveniently as either Eq. (4.2) or Eq. (4.3). After the correction maneuver, the state reads

$$\mathbf{x}_1^+ = \mathbf{x}_1^* + \underbrace{\Phi(t_0, t_1)\delta\mathbf{x}_0 + \begin{bmatrix}\mathbf{0}\\ \Delta v_1\end{bmatrix}}_{\delta\mathbf{x}_1} \tag{4.7}$$

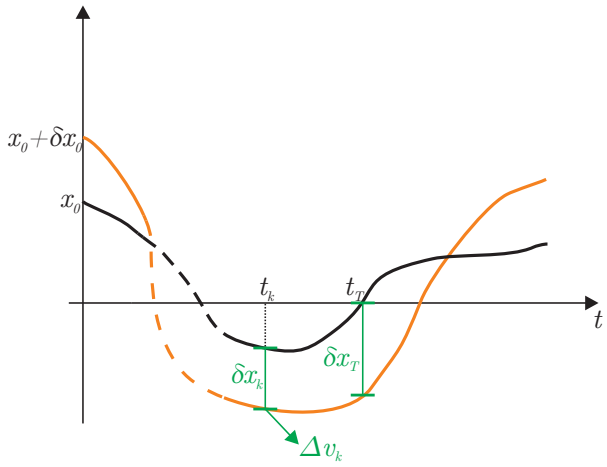



FIGURE 4.1. Target Guidance concept. Black line is the nominal path, while orange line is the real trajectory.

with $\delta\mathbf{x}_1$ is the deviation with respect to the nominal state at the first correction maneuver. If the deviation is pushed forward to the second correction epoch, exploiting the linearization theory, the state is

$$\begin{aligned}\mathbf{x}_2 = \mathbf{x}_2^* + \Phi(t_1,t_2)\delta\mathbf{x}_1 &= \mathbf{x}_2^* + \Phi(t_1,t_2)\left(\Phi(t_0,t_1)\delta\mathbf{x}_0 + \begin{bmatrix}\mathbf{0}\\ \Delta v_1\end{bmatrix}\right)\\ &= \mathbf{x}_2^* + \Phi(t_0,t_2)\delta\mathbf{x}_0 + \Phi(t_1,t_2)\begin{bmatrix}\mathbf{0}\\ \Delta v_1\end{bmatrix}\end{aligned} \tag{4.8}$$

where the composition property for the STM has been exploited. This procedure can be applied up to the time $t_k$, that is the last TCM time before the target (see Figure 4.1), leading to

$$\mathbf{x}_k = \mathbf{x}_k^* + \underbrace{\prod_{i=k}^{1}\Phi_{i-1}^{i}\delta\mathbf{x}_0 + \sum_{i=1}^{k}\prod_{j=k}^{i}\Phi_{j-1}^{j} I_v \Delta v_{j-1}}_{\delta\mathbf{x}_k} \tag{4.9}$$

where $\Phi_{j-1}^{j} = \Phi(t_{j-1},t_j)$ and $I_v$ is a block matrix, able to extract the 3-by-3 bottom-right part of the STM. Thus, the target state (at time $t_T$) can be written as

$$\mathbf{x}_T = \mathbf{x}_T^* + \underbrace{\Phi(t_k,t_T)\delta\mathbf{x}_k}_{\delta\mathbf{x}_T} \tag{4.10}$$

Now, the target function $f$ is defined, i.e. the function that must be zero to fulfill the mission objective. For example, considering as target a desired latitude $\lambda_\delta$ at the close approach, the target function will be $f = \lambda - \lambda_\delta = 0$. Now, expanding $f$ at the first order, it gets

$$f(\mathbf{x}_T) \simeq \cancelto{0}{f(\mathbf{x}_T^*)} + \left.\frac{\mathrm{d}f}{\mathrm{d}\mathbf{x}}\right|_{\mathbf{x}_T^*} \delta\mathbf{x}_T = 0 \tag{4.11}$$

Substituting Eq. (4.9) in Eq. (4.10), and in turn in Eq. (4.11), we obtain

$$\begin{aligned} f(\mathbf{x}_T) &\simeq \left.\frac{\mathrm{d}f}{\mathrm{d}\mathbf{x}}\right|_{\mathbf{x}_T^*} \Phi_k^T \left( \prod_{i=k}^{1} \Phi_{i-1}^i \delta\mathbf{x}_0 + \sum_{i=1}^{k} \prod_{j=k}^{i} \Phi_{j-1}^j I_v \Delta v_{j-1} \right) \\ &= \left.\frac{\mathrm{d}f}{\mathrm{d}\mathbf{x}}\right|_{\mathbf{x}_T^*} \prod_{i=T}^{1} \Phi_{i-1}^i \delta\mathbf{x}_0 + \left.\frac{\mathrm{d}f}{\mathrm{d}\mathbf{x}}\right|_{\mathbf{x}_T^*} \sum_{i=1}^{k} \prod_{j=T}^{i} \Phi_{j-1}^j I_v \Delta v_{j-1} = 0 \end{aligned} \tag{4.12}$$

Considering that $f$ is known and it is at least a $\mathscr{C}^1$ function, Eq. (4.12) can be written in a compact form

$$\tilde{A}_0 \delta\mathbf{x}_0 + A\boldsymbol{\Delta v} = 0 \tag{4.13}$$

where $\tilde{A}_0 = \left.\frac{\partial f}{\partial \mathbf{x}}\right|_{\mathbf{x}_T^*} \prod_{i=T}^{1} \Phi_{i-1}^i$ and $A = \left.\frac{\partial f}{\partial \mathbf{x}}\right|_{\mathbf{x}_T^*} \sum_{i=1}^{k} \prod_{j=T}^{i} \Phi_{j-1}^j I_v$, and

$$\boldsymbol{\Delta v} = \begin{bmatrix} \Delta v_1 \\ \Delta v_2 \\ \vdots \\ \Delta v_M \end{bmatrix} \in \mathbb{R}^{(3M)}$$

The algorithm can be modified to account more than a single target function. In this case, matrices $\tilde{A}_0$ and $A$ should be modified accordingly. For example,

$$A = \frac{\mathrm{d}f_i}{\mathrm{d}\Delta v_j} = \begin{bmatrix} \frac{\mathrm{d}f_1}{\mathrm{d}\Delta v_1} & 0 & \cdots & 0 \\ \frac{\mathrm{d}f_2}{\mathrm{d}\Delta v_1} & \frac{\mathrm{d}f_2}{\mathrm{d}\Delta v_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\mathrm{d}f_N}{\mathrm{d}\Delta v_1} & \frac{\mathrm{d}f_N}{\mathrm{d}\Delta v_2} & \cdots & \frac{\mathrm{d}f_N}{\mathrm{d}\Delta v_M} \end{bmatrix} \in \mathbb{R}^{N\times(3M)} \tag{4.14}$$

where $N$ is the number of target functions, while $M$ the number of control maneuvers. Zeros in Eq. (4.14) are related to the fact that impulses future in time withe respect to the target are not able to influence the target function itself, i.e.

if the $i$-th target function occurs later of the $j$-th TCM, then $\frac{\mathrm{d}f_i}{\mathrm{d}\Delta v_j} = \mathbf{0}$.
Moreover, a *target trigger function* $g$ is introduced, being null whenever the target condition is reached. Thus, it checks then the target specified by function $f$ is happening at the right moment. For example, considering as target a desired latitude $\lambda_\delta$ at the close approach, the target trigger function will be the close approach condition, so $g = (\mathbf{r}\cdot\mathbf{v}) = 0$. Again, considering a first order expansion,

$$g(\mathbf{x}_T) \simeq \cancelto{0}{g\left(\mathbf{x}_T^*\right)} + \left.\frac{\mathrm{d}g}{\mathrm{d}\mathbf{x}}\right|_{\mathbf{x}_T^*} \delta\mathbf{x}_T = 0 \tag{4.15}$$

Repeating mathematical steps of Eqs. (4.12)–(4.13), considering that $g$ is a known $\mathscr{C}^1$-class function, Eq. (4.15) reads

$$\tilde{G}_0 \delta\mathbf{x}_0 + G\boldsymbol{\Delta v} = 0 \tag{4.16}$$

Then, an optimization problem is set in order to minimize the sum of the navigation costs, while satisfying the targeting constraints given by Eqs. (4.13) and (4.16), i.e., find $\Delta v_j,\ j = \{1,\ldots,M\}$ such that

$$J = \sum_{j=1}^{M} \frac{1}{2} \left\|\Delta v_j\right\|^2 = \frac{1}{2}\boldsymbol{\Delta v}^T\boldsymbol{\Delta v} \tag{4.17}$$

is minimized, subjected to

$$\underbrace{\begin{bmatrix} A \\ G \end{bmatrix}}_{\hat{A}} \boldsymbol{\Delta v} = -\underbrace{\begin{bmatrix} \tilde{A}_0 \\ \tilde{G}_0 \end{bmatrix}}_{\hat{A}_0} \delta\mathbf{x}_0 \tag{4.18}$$

Applying the Lagrange theory, a Lagrange function associated to the optimization problem can be written as

$$\mathscr{L} = \frac{1}{2}\boldsymbol{\Delta v}^T\boldsymbol{\Delta v} + \boldsymbol{\lambda}^T\left(\hat{A}\boldsymbol{\Delta v} + \hat{A}_0\delta\mathbf{x}_0\right) \tag{4.19}$$

where $\boldsymbol{\lambda}$ is the Lagrange multipliers vector. In this case, the necessary condition for the optimization problem can be summarized as

$$\begin{cases} \frac{\partial\mathscr{L}}{\partial\boldsymbol{\Delta v}} = \boldsymbol{\Delta v} + \hat{A}^T\boldsymbol{\lambda} = \mathbf{0} \\ \frac{\partial\mathscr{L}}{\partial\boldsymbol{\lambda}} = \hat{A}\boldsymbol{\Delta v} + \hat{A}_0\delta\mathbf{x}_0 = \mathbf{0} \end{cases} \tag{4.20}$$

The two conditions in Eq. (4.20) can be written in a more elegant compact form

$$\begin{bmatrix} I_{3M} & \hat{A}^T \\ \hat{A} & 0_{2N} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Delta v} \\ \boldsymbol{\lambda} \end{bmatrix} = - \begin{bmatrix} \mathbf{0}_{3M} \\ \hat{A}_0 \delta \mathbf{x}_0 \end{bmatrix} \tag{4.21}$$

where $I_{3M}$ is the $3M$-dimensional identity matrix and $0_{2N}$ the (2N × 2N) null matrix. Applying the explicit inverse formula for a 2×2 Hermitian block triangular matrix [53], it is possible to write

$$\begin{aligned} \begin{bmatrix} \boldsymbol{\Delta v} \\ \boldsymbol{\lambda} \end{bmatrix} &= - \begin{bmatrix} I_{3M} & \hat{A}^T \\ \hat{A} & 0_{2N} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{0}_{3M} \\ \hat{A}_0 \delta \mathbf{x}_0 \end{bmatrix} \\ &= - \begin{bmatrix} I_{3M} - \hat{A}^T \left(\hat{A}\hat{A}^T\right)^{-1} \hat{A} & \hat{A}^T \left(\hat{A}\hat{A}^T\right)^{-1} \\ \left(\hat{A}\hat{A}^T\right)^{-1} \hat{A} & -\left(\hat{A}\hat{A}^T\right)^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{0}_{3M} \\ \hat{A}_0 \delta \mathbf{x}_0 \end{bmatrix} \end{aligned} \tag{4.22}$$

Taking the first row of Eq. (4.22), the minimizing impulses can be retrieved as

$$\boldsymbol{\Delta v} = -\hat{A}^T \left(\hat{A}\hat{A}^T\right)^{-1} \hat{A}_0 \delta \mathbf{x}_0 = -\hat{A}^\dagger \hat{A}_0 \delta \mathbf{x}_0 \tag{4.23}$$

where $\hat{A}^\dagger$ is the Moore–Penrose pseudo-inverse matrix of $\hat{A}$ [64].
Note that this system is well-posed only if $3M \geq 2N$; hence, we should have at least 2 trajectory correction maneuvers every 3 targets.
This method can be applied in a receding horizon approach, meaning that, at each OD time, values of $\delta \mathbf{x}_0$, $\hat{A}$, and $\hat{A}_0$ are updated, a new value for $\boldsymbol{\Delta v}$ is computed, but only the first $\Delta v_j$ is applied in practice, while the others are recomputed again in the subsequent legs. Moreover, it is important to notice that the algorithm computes TCMs able to control all target states downstream. This characteristic is of paramount importance whenever several targets close in time are chosen.

CHAPTER 5

# SCENARIO 1

Scenario (1) describes a spacecraft flying in a strongly perturbed environment, controlling its trajectory using impulsive maneuvers. Since Scenario (1) can be used to represent spacecraft flying on Lagrange points orbit manifolds, it can well describe the trajectory of the CubeSat LUMIO [71].

## 5.1 Introduction

The Lunar Meteoroid Impact Observer (LUMIO) 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. An independent assessment conducted at ESA's Concurrent Design Facility (CDF) has proved the mission feasibility. In 2020, ESA considered the mission for further implementation, funding the Phase A study. LUMIO


FIGURE 5.1. LUMIO logo.

successfully passed the Mission Definition Review (MDR) in July 2020 and the Preliminary Requirements Review (PRR) in February 2021.
LUMIO space segment is composed by a 12U form-factor CubeSat, flying a halo orbit at Earth–Moon $L_2$. The spacecraft is equipped with the LUMIO-Cam, a novel miniaturized optical instrument capable of detecting light flashes in the visible spectrum produced by meteoroid impacts. Indeed, LUMIO shall observe, quantify, and characterize meteoroid impacts on the lunar farside by detecting their flashes, complementing Earth-based observations on the Lunar nearside, to provide global information on the lunar meteoroid environment and contribute to lunar situational awareness. The mission implements a novel orbit design and latest CubeSat technologies to serve as a pioneer in demonstrating how CubeSats can become a viable tool for deep space science and exploration. The whole mission profile is summarized in Figure 5.2. The mission can be subdivided into five sequential phases:

1. *Launch and trajectory from Earth to Moon* is under the responsibility of a mothership, that brings LUMIO on a given Low Lunar Orbit (LLO);

2. In the *Parking phase*, LUMIO is released on the LLO by the mothership, where the system commissioning is started;

3. The *Transfer phase* begins with the Stable Manifold Injection Maneuver (SMIM), bringing LUMIO to the stable manifold, connecting the LLO with the operative orbit, and ends with the Halo Injection Maneuver (HIM);

4. During *Operative phase* LUMIO is placed on its operative orbit, where the science observations are performed in order to fulfill the scientific objectives;

5. The disposal is performed in the *end-of-life phase* in order to avoid risks for future missions.

In this work, the focus will be placed on the transfer phase, that embed clearly the ontological characteristics of Scenario (1).

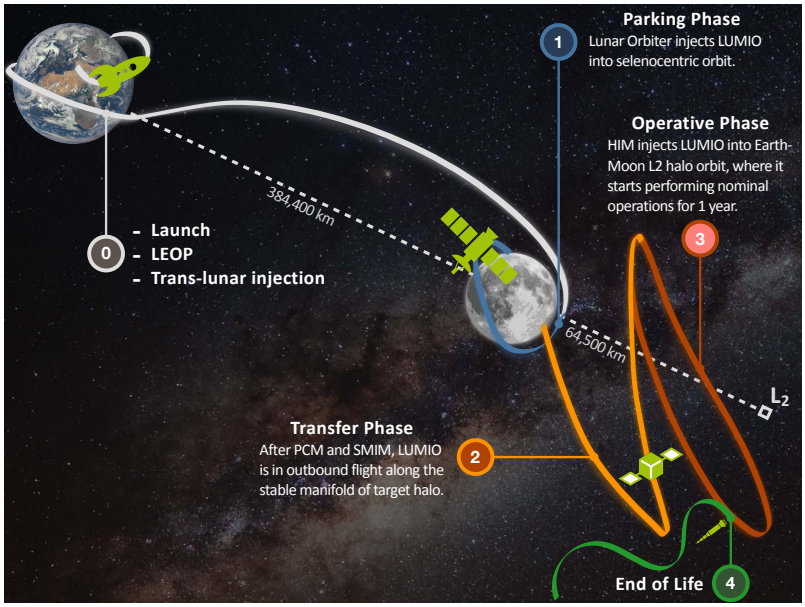



FIGURE 5.2. LUMIO mission profile (from [19]).

### 5.1.1 Parking phase

During the parking phase, LUMIO is released on a Low Lunar Orbit by the carrier. However, the actual release orbit is unknown since it will be driven by the primary payload of the mission, being LUMIO only a secondary payload. During Phase A study, an holistic search has been performed to assess mission robustness against the injection option possibilities by considering variations in the dimensions and in the angular parameters of the release orbit about the Moon. At the end of the analysis, several LLOs are deemed feasible, needing less than 150 m/s for the SMIM impulse. Feasible solutions are characterized by high apocenter and low pericenter altitudes (see Figure 5.3) and the feasible Keplerian parameters set gets wider by lowering the pericenter and increasing the apocenter distance (as shown in Figure 5.4). In order to reduce the search space and relieve the burden given by a large feasible orbits set, a LLO is selected arbitrarily to be used in this work. The 600 km×20 000 km parking orbit, chosen in the CDF study [82], is picked out. Angular parameters are considered free.

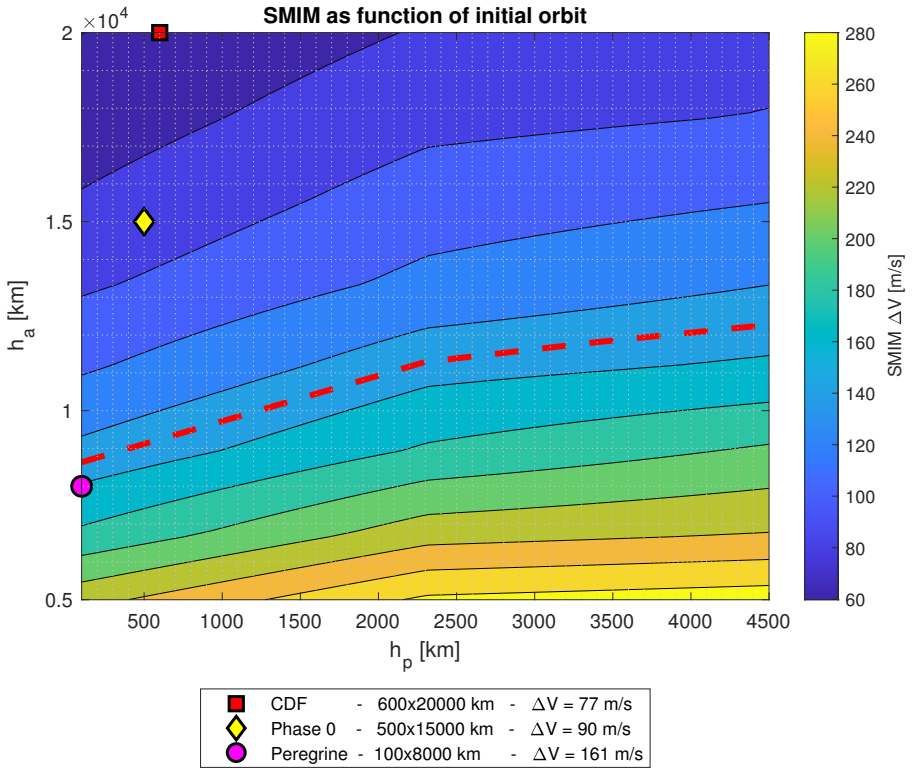



FIGURE 5.3. $\Delta v_{\text{SMIM}}$ as function of parking orbit pericenter and apocenter altitude. Red line marks 150 m/s line. Peregrine is a NASA's CLPS mission, being one of the possible motherships.

### 5.1.2 Operative Orbit

LUMIO operative orbit was selected during Phase 0, thoroughly trading off the scientific requirements and the orbit maintenance costs. The operative orbit was selected from a set of fourteen quasi-halos orbits about Earth–Moon $L_2$ computed in the high-fidelity Roto-Pulsating Resticted $n$-Body Problem (RPRnBP). Details for this process can be found in [19]. The selected halo orbit has been kept fixed in Phase A. This will reduce the space of the feasible parking orbits, allowing the

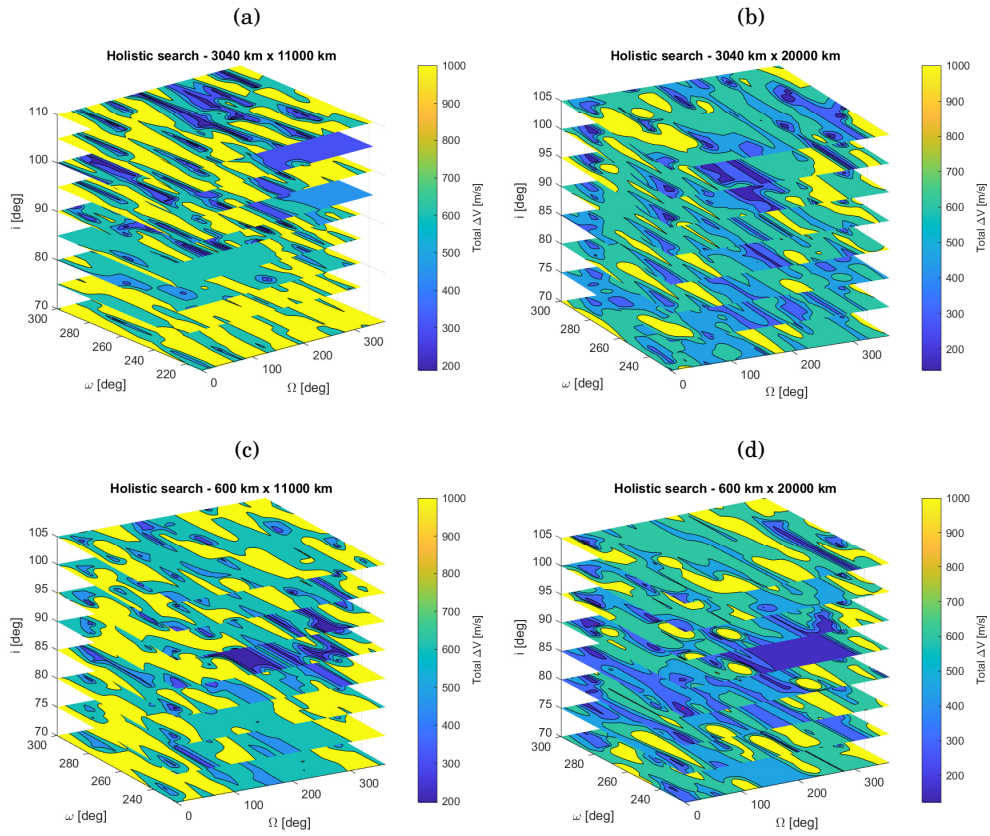



FIGURE 5.4. Holistic search results for four sample LLOs. Color code shows the total $\Delta v$ needed to reach the operative orbit. Deep blue areas are the one with $\Delta v < 150\,\text{m/s}$. Angular parameters are in the MCME2000 reference frame.

stable manifold transfer with the allotted propellant mass, and will constrain the possible injection times. However, this choice eased the payload and system design, reducing the environment indefiniteness.

The designated LUMIO operative orbit is quasi-periodic halo orbit about Earth–Moon $L_2$ characterized by a Jacobi constant $C_j = 3.09$. Table 5.1 shows nondimensional Jacobi energy and the out-of-plane amplitude $A_z$. According to the timeline of the mission, defined during Phase A, the operative phase is expected to start on

21 March, 2024, and to end on 22 March, 2025. The trajectory of the quasi-halo orbit in that time frame is shown in Figure 5.5. In this work, the operative orbit serves as a moving target on a prescribed trajectory.

TABLE 5.1. Parameters of LUMIO operational orbit.

| Name | $C_j$ [-] | $A_z$ [km] |
|---|---|---|
| Halo_Cj3p09 | 3.09 | 28418.41 |

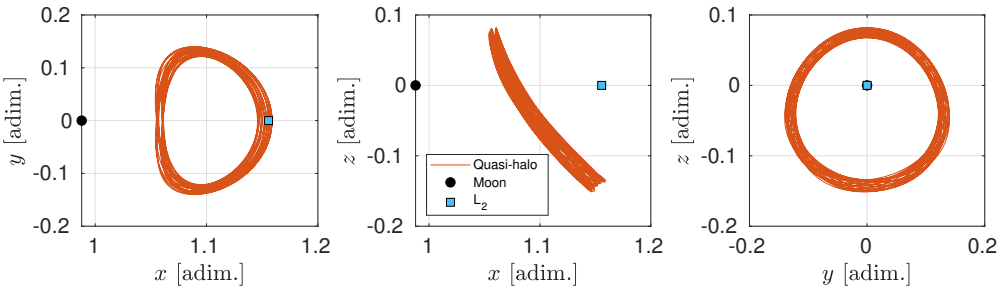



FIGURE 5.5. Projection of the selected operative Earth–Moon $L_2$ quasi-halo in the Roto-Pulsating Frame for LUMIO.

### 5.1.3 Transfer phase

In the transfer phase, LUMIO is brought from a Low Lunar Orbit, where it was released by a mothership, to the operative orbit. 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 in physical space, 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 and it will be indicated with $\Delta v_{\text{SMIM}}$. After the transfer, the Halo Injection Maneuver, $\Delta v_{\text{HIM}}$, eventually injects the CubeSat into the final operative orbit. A detailed study of the TCM problem for several Lagrange points orbits, exploiting simple dynamical systems concepts, has shown that two TCMs

provide sufficient degrees of freedom [41]. 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 orbit determination 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 $\Delta v_{\text{SMIM}}$. 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 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 Figure 5.6. 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 Figure 5.6, can be subdivided into three sub-phases:

1. *OD phase* (between days 0 and 1, or between days 6 and 7 after $t_0$): during this phase, a visibility window is identified (see Section 5.2.3), and the OD algorithm is exploited within it;
2. *Cut-off phase* (between days 1 and $t_{TCM_1}$, or between days 7 and $t_{TCM_2}$ after $t_0$): in this phase the Differential Guidance (Section 4.1) is exploited to compute the correction maneuver, which is applied at the end of the phase;
3. *Ballistic phase* (between days $t_{TCM_1}$ and 6, or between days $t_{TCM_2}$ and $t_f$ after $t_0$): in this phase, the spacecraft undergoes a ballistic flight.

In this work, the transfer trajectory is the main topic and it is analyzed carefully in the section remainder.

#### 5.1.3.1 Dynamics

The motion of the CubeSat in the transfer phase can be described by using the RPRnBP [24], in order to have a high-fidelity dynamics, able to correctly represent

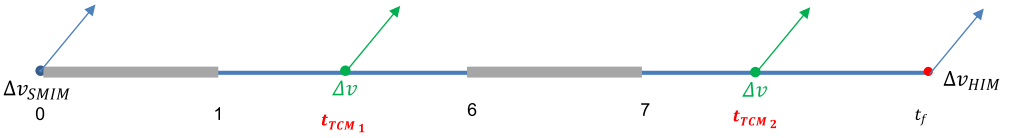



FIGURE 5.6. LUMIO transfer trajectory timeline. The grey bars represents the OD phases, while the green arrows mark the TCMs points. Time in days after the SMIM.

the highly non-linear trajectory of LUMIO. The use of an adimensional Roto-Pulsating Frame (RPF) ease 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 (R3BP).

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

$$k(t) = \|\mathbf{r}_P(t) - \mathbf{r}_S(t)\| \tag{5.1}$$

where $\mathbf{r}_P$ are $\mathbf{r}_S$ 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 $\omega = \sqrt{\frac{G\left(m_{\oplus} + m_{\leftmoon}\right)}{\tilde{a}^3}}$ is set to unity, with ($m_{\oplus}$ and $m_{\leftmoon}$, 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 $2\pi$. In this framework, Earth and the Moon are have fixed position, $[-\mu, 0, 0]^T$ and $[1-\mu, 0, 0]^T$ respectively, with $\mu = m_{\leftmoon}/\left(m_{\oplus} + m_{\leftmoon}\right)$ being the mass parameter of the system (Figure 5.7).

The equations of motion for the RPRnBP reads [23]

$$\boldsymbol{\rho}'' + \frac{1}{\omega}\left(\frac{2\dot{k}}{k} I + 2C^T\dot{C}\right)\boldsymbol{\rho}' + \frac{1}{\omega^2}\left(\frac{\ddot{k}}{k} I + 2\frac{\dot{k}}{k} C^T\dot{C} + C^T\ddot{C}\right)\boldsymbol{\rho} + \frac{C^T\ddot{\mathbf{b}}}{k\omega^2} = \nabla\Omega + \mathbf{a}_{SRP} \tag{5.2}$$

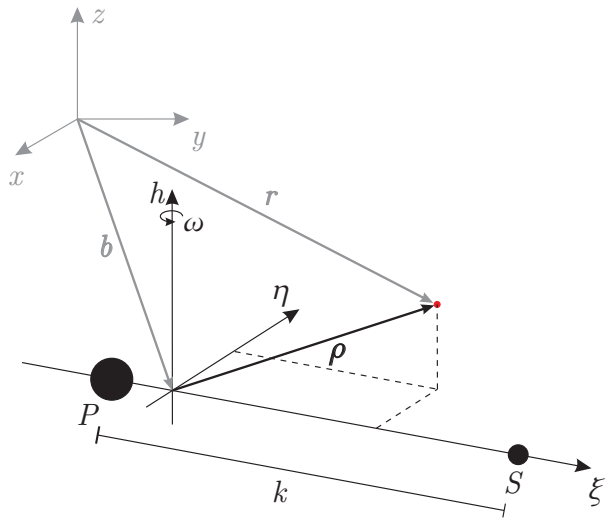



FIGURE 5.7. Rotating, pulsating, non-inertial reference frame (RPF). The inertial reference frame and quantities making reference to it are drawn in grey. Spacecraft is the red dot.

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

$$\Omega = \sum_{j\in\mathscr{S}} \frac{\hat{\mu}_j}{\delta_j}\left[1 + \frac{J_{2_j} R_{B_j}^2}{2k^2\delta_j^2}\left(1 - \frac{3\boldsymbol{\delta}_j^T M \boldsymbol{\delta}_j}{\delta_j^2}\right)\right] \tag{5.3}$$

where $\mathscr{S}$ is the set containing the primaries and characterized by the adimensional gravitational parameter $\hat{\mu}_j = m_j/(m_☉ + m_☽)$, 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 $\boldsymbol{\delta}_j = \boldsymbol{\rho} - \boldsymbol{\rho}_j$, and $\delta_j$ is its magnitude. The adimensionalized Solar radiation pressure (SRP) acceleration in Eq. (5.2) can be expressed, using the cannon ball model, as

$$\mathbf{a}_{SRP} = \frac{\gamma_0}{\omega^2 k^3}\frac{\boldsymbol{\delta}_S}{\delta_S^3} \tag{5.4}$$

where $\boldsymbol{\delta}_S$ is the Sun position and $\gamma_0$ is the SRP parameter, defined as

$$\gamma_0 = (1 + c_r)\frac{A}{m}\frac{\Psi_0 d_0^2}{c} \tag{5.5}$$

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. (5.2) 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 [1, 2] as well as the physical constants. The transformations from the solar barycentric inertial frame of reference (i.e. J2000) and the Roto-Pulsating Frame are

$$\mathbf{r}(t) = \mathbf{b}(t) + k(t)C(t)\boldsymbol{\rho}(t) \tag{5.6}$$

$$\mathbf{v}(t) = \dot{\mathbf{b}} + \dot{k}C\boldsymbol{\rho} + kC\omega\boldsymbol{\rho}' \tag{5.7}$$

$$\tau = \omega(t - t_0) \tag{5.8}$$

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

$$\mathbf{b}(t) = \frac{m_{\earth}\mathbf{r}_{\earth} + m_{\moon}\mathbf{r}_{\moon}}{m_{\earth} + m_{\moon}} \tag{5.9}$$

and $C$ the cosine angle matrix between J2000 and the RPF

$$C(t) = [\xi, \eta, h] \tag{5.10}$$

with

$$\xi = \frac{\mathbf{r}_{\moon} - \mathbf{r}_{\earth}}{k}, \qquad h = \frac{\left(\mathbf{v}_{\moon} - \mathbf{v}_{\earth}\right) \times \left(\mathbf{r}_{\moon} - \mathbf{r}_{\earth}\right)}{\left\| \left(\mathbf{v}_{\moon} - \mathbf{v}_{\earth}\right) \times \left(\mathbf{r}_{\moon} - \mathbf{r}_{\earth}\right) \right\|}, \qquad \eta = h \times \xi \tag{5.11}$$

In Table 5.2, values for the most useful parameters in the dynamics are presented.

#### 5.1.3.2 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 needed for the STM computation are required. Being $\boldsymbol{\chi}$ the spacecraft state in the RPF, the state-space representation of Eq. (5.2) is

$$\boldsymbol{\chi}' = \mathbf{f}(\boldsymbol{\chi}, \tau) = \begin{bmatrix} \boldsymbol{\rho}' \\ \boldsymbol{\nu}' \end{bmatrix} = \begin{bmatrix} \boldsymbol{\nu} \\ \nabla\Omega + \frac{\gamma_0}{\omega^2 k^3}\frac{\boldsymbol{\delta}_S}{\delta_S^3} - \frac{1}{\omega}\left(\frac{2\dot{k}}{k}I + 2C^T\dot{C}\right)\boldsymbol{\nu} - \frac{1}{\omega^2}\left(\frac{\ddot{k}}{k}I + 2\frac{\dot{k}}{k}C^T\dot{C} + C^T\ddot{C}\right)\boldsymbol{\rho} - \frac{C^T\ddot{\mathbf{b}}}{k\omega^2} \end{bmatrix} \tag{5.12}$$

TABLE 5.2. Parameters of Restricted n-Body Problem and the spacecraft.

| **Parameter** | **Symbol** | **Value** |
|---|---|---|
| Earth–Moon mass ratio | $\mu$ | 0.01215058426994 |
| Earth gravitational parameter | $\mu_{\oplus}$ | 398 600.435 436 095 9 $\mathrm{km}^3/\mathrm{s}^2$ |
| Moon gravitational parameter | $\mu_{\leftmoon}$ | 4902.800 066 163 796 $\mathrm{km}^3/\mathrm{s}^2$ |
| 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 | $\gamma_0$ | $2.210\,656\,810\,8 \times 10^6\ \mathrm{km}^3/\mathrm{s}^2$ |
| Reflectivity coefficient | $c_r$ | 0.08 |
| Area-to-mass ratio | $A/m$ | $0.02\ \mathrm{m}^2/\mathrm{kg}$ |

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

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

$$\dot{\Phi}(\tau_0,\tau) = A(\tau)\Phi(\tau_0,\tau), \qquad \Phi(\tau_0,\tau_0) = I_6 \tag{5.13}$$

with

$$A(\tau) = \frac{\partial \mathbf{f}}{\partial \boldsymbol{\chi}} = \begin{bmatrix} 0_3 & I_3 \\ \frac{\partial \mathbf{f}_{\boldsymbol{v}}}{\partial \boldsymbol{\rho}} & \frac{\partial \mathbf{f}_{\boldsymbol{v}}}{\partial \boldsymbol{v}} \end{bmatrix} \tag{5.14}$$

being the Jacobian of the dynamics right-hand side, where [24]

$$\begin{aligned} \frac{\partial \mathbf{f}_{\boldsymbol{v}}}{\partial \boldsymbol{\rho}} = &-\frac{1}{\omega^2}\left(\frac{\ddot{k}}{k} I_3 + 2\frac{\dot{k}}{k} C^T \dot{C} + C^T \ddot{C}\right) - \sum_{j \in \mathscr{S}} \hat{\mu}_j \left[\frac{I_3}{\delta_j^3} - \frac{3\boldsymbol{\delta}_j \boldsymbol{\delta}_j^T}{\delta_j^5}\right. \\ &\left. + \frac{3J_{2_j} R_{B_j}^2}{2k^2}\left(\frac{I_3 + 2M}{\delta_j^5} - \frac{5\boldsymbol{\delta}_j\boldsymbol{\delta}_j^T + 10M\boldsymbol{\delta}_j\boldsymbol{\delta}_j^T + 5\boldsymbol{\delta}_j^T M\boldsymbol{\delta}_j + 10\boldsymbol{\delta}_j\boldsymbol{\delta}_j^T M}{\delta_j^7} + 35\boldsymbol{\delta}_j^T M \boldsymbol{\delta}_j \frac{\boldsymbol{\delta}_j\boldsymbol{\delta}_j^T}{\delta_j^9}\right)\right] \\ &+ \frac{\gamma_0}{\omega^2 k^3}\left(\frac{I_3}{\delta_S^3} - 3\frac{\boldsymbol{\delta}_S\boldsymbol{\delta}_S^T}{\delta_S^5}\right) \end{aligned} \tag{5.15}$$

while

$$\frac{\partial \mathbf{f}_{\boldsymbol{v}}}{\partial \boldsymbol{v}} = -\frac{2}{\omega}\left(\frac{\dot{k}}{k} I_3 + C^T \dot{C}\right) \tag{5.16}$$

#### 5.1.3.3 Initial point

The initial LLO is provided using Keplerian elements, i.e. a constant semi-major axis $a = 12037.1\,\text{km}$ and a constant eccentricity $e = 0.65848$ plus a set of free angular parameters $\boldsymbol{\alpha}_0 = [i_0, \Omega_0, \omega_0, \theta_0]$, containing the inclination $i_0$, the right ascension of the ascending node $\Omega_0$, the argument of the pericenter $\omega_0$ and the true anomaly $\theta_0$. 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 $\mathbf{x}_{\text{MCME}}$ [21]

$$\mathbf{r}_{\text{MCME}} = T_1 \begin{bmatrix} \frac{p\cos\theta_0}{1+e\cos\theta_0} \\ \frac{p\sin\theta_0}{1+e\cos\theta_0} \\ 0 \end{bmatrix}, \qquad \mathbf{v}_{\text{MCME}} = T_1 \begin{bmatrix} -\sqrt{\frac{\mu_{\leftmoon}}{p}}\sin\theta_0 \\ \sqrt{\frac{\mu_{\leftmoon}}{p}}(e+\cos\theta_0) \\ 0 \end{bmatrix} \tag{5.17}$$

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

$$\mathbf{x}_{\text{J2000}} = \begin{bmatrix} \mathbf{r}_{\text{J2000}} \\ \mathbf{v}_{\text{J2000}} \end{bmatrix} = T_2 \begin{bmatrix} \mathbf{r}_{\text{MCME}} \\ \mathbf{v}_{\text{MCME}} \end{bmatrix} \tag{5.18}$$

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 $i_M = 24\,\text{deg}$ is the lunar axial tilt with respect the Earth's equator [56]. 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.

$$\mathbf{x}_0 = \mathbf{x}_{\text{J2000}} + \mathbf{x}_{\leftmoon} \tag{5.19}$$

Then, the J2000 initial state $\mathbf{x}_0$ is converted in the Roto-Pulsating Frame one $\boldsymbol{\chi}_0$ by applying Eqs. (5.6)–(5.7). The SMIM is applied on top of this initial state.

#### 5.1.3.4 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.

$$\boldsymbol{\chi}(t_0) \sim \mathcal{N}\left(\boldsymbol{\chi}_0, P_{\boldsymbol{\chi}}\right) \tag{5.20}$$

where $P_{\boldsymbol{\chi}} = \mathrm{diag}\left(\left[\sigma_{\boldsymbol{\rho}}^2 I_3, \sigma_{\boldsymbol{v}}^2 I_3\right]\right)$ is the 6-dimensional diagonal covariance matrix, with $\sigma_{\boldsymbol{\rho}}^2$ and $\sigma_{\boldsymbol{v}}^2$, 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 trajectory correction maneuvers are assumed free from uncertainties. Since uncertainty in the Halo Injection Maneuver 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 $1\sigma$ error $\sigma_{\Delta v}$ in modulus and $\sigma_\delta$ in pointing angle. The modulus error is defined as a fraction of the nominal modulus, 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

$$\Delta v = \sqrt{\Delta v_x^2 + \Delta v_y^2 + \Delta v_z^2} \tag{5.21a}$$

$$\alpha = \mathrm{atan2}\left(\Delta v_y, \Delta v_x\right) \tag{5.21b}$$

$$\epsilon = \mathrm{atan2}\left(\Delta v_z, \sqrt{\Delta v_x^2 + \Delta v_y^2}\right) \tag{5.21c}$$

where $\Delta v$ is the magnitude, and $\alpha$ and $\epsilon$ are the Azimuth and Elevation respectively. Then, the associated spherical covariance, i.e. $P^s_{\Delta v} = \text{diag}\left(\sigma^2_{\Delta v}, \sigma^2_\delta, \sigma^2_\delta\right)$, is transformed in Cartesian coordinates

$$P_{\Delta v} = J P^s_{\Delta v} J^T \tag{5.22}$$

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

$$J = \begin{bmatrix} \cos\epsilon\cos\alpha & -\Delta v\cos\epsilon\sin\alpha & -\Delta v\sin\epsilon\cos\alpha \\ \cos\epsilon\sin\alpha & \Delta v\cos\epsilon\cos\alpha & -\Delta v\sin\epsilon\sin\alpha \\ \sin\epsilon & 0 & \Delta v\cos\epsilon \end{bmatrix} \tag{5.23}$$

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

$$P_0 = P_{\boldsymbol{\chi}} + \begin{bmatrix} 0_3 & 0_3 \\ 0_3 & P_{\Delta v} \end{bmatrix} \tag{5.24}$$

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 so also the number of samples required in the PCE. Characteristics of the random variables are reported in Table 5.3.

TABLE 5.3. Stochastic characteristics of system uncertainty.

| $\sigma_\rho$ [km] | $\sigma_v$ [m/s] | $\sigma_{\Delta v}$ [%] | $\sigma_\delta$ [deg] |
|---|---|---|---|
| 1 | 0.01 | 1 | 1 |

## 5.2 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, ii) how the stochastic variables are estimated, iii) and how the orbit determination is performed. Moreover, the simplifying assumptions are presented as a preliminary for the optimization problem statement.

### 5.2.1 Uncertainty propagation

Considering the analysis in Section 3.2, the uncertainty quantification (UQ) method for Scenario (1) is based on an Non-Intrusive Polynomial Chaos Expansion algorithm. This technique is exploited both to estimate the trajectory knowledge and dispersion. In order to evaluate the Polynomial Chaos Expansion coefficients in a fast and reliable way, a pseudospectral collocation on a sparse grid is exploited (see Section 3.1.3.4). Instead of a Smolyak's grid, the number of points can be further reduced using the so-called *conjugate unscented transformation (CUT)* [4]. CUT is the natural extension of unscentend 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 [5]. Thus, it can be used to efficiently compute the generalized Fourier PCE coefficients exploiting Eq. (3.26). So, conjugate unscented transformation can be seen as just another way to compute the stochastic integral given in Eq. (3.33). This hybrid technique, using CUT to estimate PCE coefficient, is unimaginatively labeled polynomial chaos expansion–conjugated unscented transformation (PCE-CUT). Even though this approach exhibits several advantages over the standard sparse grid interpolation techniques, such as positive quadrature weights and fewer quadrature points, its implementation in the space sector is a novelty.
Conjugate unscented transformation achieve to provide high-order quadrature rules by satisfy the so-called moment contraints equations (MCEs). They are a set of equations that can be found by comparing the definition of a stochastic integral (Eq. (3.26)) with its quadrature approximation (Eq. (3.33)). By doing so, the MCEs can be built as

$$\sum_{q=1}^{M} \left[ \left(\xi_1^{q_1}\right)^{n_1} \left(\xi_2^{q_2}\right)^{n_2} \dots \left(\xi_d^{q_d}\right)^{n_d} \right] \omega_q = E\left[ (\xi_1)^{n_1} (\xi_2)^{n_2} \dots (\xi_d)^{n_d} \right] \tag{5.25}$$

where $\sum_{i=1}^{d} n_i = m$ is the order of the stochastic moment.
In order to solve the MCEs represented in Eq. (5.25), the quadrature points $\boldsymbol{\xi}_q$ are constrained to lie symmetrically on some conjugate axes. The use of symmetric set of points allows to intrinsically satisfy odd MCEs, since the Gaussian PDF is symmetric. The conjugate axes are peculiar directions in the $d$-dimensional

random variables space, with a common center in the mean, and their definition can be retrieved from Table 5.4. Since they enjoy symmetry, points on the same set are equidistant from the mean and they are scaled with the same variable $r$ and have the same Gauss-Hermite weight $\omega$. These scaled points and weight are substituited in the MCEs (Eq. (5.25)), giving a system of polynomial equations in $r_i$ and $\omega_i$ with $i = \{1,\dots,l\}$, where $l$ being the number of axes chosen. Scenario (1) quantities of interest are considered to be correctly represented using quadrature points that can completely satisfy the MCE up to the forth order. Since the set of symmetric axes automatically satisfies the odd MCEs, only the equations for the even order moments can be considered [5]

$$E\left[\xi_i^2\right] = 1, \qquad E\left[\xi_i^4\right] = 3, \qquad E\left[\xi_i^2\xi_j^2\right] = 1 \tag{5.26}$$

Moreover, the quadrature weights should be normalized, thus an additional equation has to be added, i.e. $\sum_{q=1}^{m} \omega_q = 1$. These polynomial system can be solved by chosing two axes, with four associated variables $r_1$, $r_2$, $\omega_1$ and $\omega_2$. In addition, the central point at the origin can have its weight $\omega_0$. In this case, the problem is undetermined since there are five unknown, but only four equations. In order to both have the minimum number of quadrature points and to avoid negative weights to improve numerical stability, the principal axes in combination with the $c^d$ conjugate axes are selected to satisfy the MCEs [5]. Table 5.5 summarizes the axes selection for a fourth order CUT.

TABLE 5.4. Symmetric conjugated axes.

| **Name** | **Symbol** | **Sample point** | **Number of points** |
|---|---|---|---|
| Principal axes | $\boldsymbol{\sigma}$ | $[1,0,\dots,0]$ | $2d$ |
| m-th conjugated axes | $\mathbf{c}^l$ | $[\underbrace{1,1\dots,1}_{l},\underbrace{0,0\dots,0}_{d-l}]$ | $2^l\binom{d}{l}$ |
| Scaled conjugate axes | $\mathbf{s}^n(h)$ | $[h,1,1,\dots,1]$ | $d2^d$ |

TABLE 5.5. Quadrature points for CUT4.

| Number of points | Position | Weights |
|---|---|---|
| $1 \le i \le 2d$ | $\boldsymbol{\xi}_i = r_1 \boldsymbol{\sigma}_i$ | $\omega_i = \omega_1$ |
| $1 \le i \le 2^d$ | $\boldsymbol{\xi}_{i+2d} = r_2 \mathbf{c}_i^l$ | $\omega_{i+2d} = \omega_2$ |
| 1 | $\boldsymbol{\xi}_0 = \mathbf{0}_d$ | $\omega_0 = \omega_0$ |
| | $M = 2d + 2^d + 1$ | |

Starting from Eq. (5.26), the moment contraints equations system read

$$\begin{cases} 2r_1^2\omega_1 + 2^d r_2^2 \omega_2 = 1 \\ 2r_1^4\omega_1 + 2^d r_2^4 \omega_2 = 3 \\ 2^d r_2^4 \omega_2 = 1 \\ 1 - 2d\omega_1 - 2^d \omega_2 = \omega_0 \end{cases} \tag{5.27}$$

In order to have a well-posed system, a fifth equation, namely one of the sixth order MCE, is added

$$\left(2r_1^6\omega_1 + 2^d r_2^6 \omega_2 - 15\right)^2 = 0 \tag{5.28}$$

Solution of the system given by Eqs. (5.27)–(5.28) for the 6-dimensional space of Scenario (1) is given in Table 5.6. In this case, 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 [72].

CUT4 results are computed by considering normalized Gaussian variables. If the random variables are represented by a generic multivariate Gaussian distribution with mean $\bar{\boldsymbol{\chi}}$ and covariance matrix $P$, the generic quadrature point $\boldsymbol{\zeta}_q$ can be retrieved by exploiting the affine transformation

$$\boldsymbol{\chi}_q = S\boldsymbol{\xi}_q + \bar{\boldsymbol{\chi}} \tag{5.29}$$

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

TABLE 5.6. CUT4 parameters solution.

| Parameter | Value |
|---|---|
| $r_1$ | 2.606009947366509 |
| $r_2$ | 1.190556303640186 |
| $\omega_1$ | 0.021681819437030 |
| $\omega_2$ | 0.007777146339805 |
| $\omega_0$ | 0.242080802685967 |

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 [46]. Figures 5.8 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 simulation and the proposed technique. Figure 5.9 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 for the same Polynomial Chaos Expansion accuracy.

### 5.2.2 Stochastic variables estimation

Once the PCE coefficients $\mathbf{c}_{\boldsymbol{\alpha}}$ at a given time $\tau$ are retrieved by means of the 4th-order conjugate unscented transformation, as explained in Section 5.2.1, the stochastic state at a given time can be estimated as (Eq. (3.24))

$$\boldsymbol{\chi}(\tau, \boldsymbol{\xi}) = \sum_{\boldsymbol{\alpha} \in \Lambda_{p,d}} \mathbf{c}_{\boldsymbol{\alpha}}(\tau) \psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}) \tag{5.30}$$

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 has to be estimated and then used to evaluate probabilities. In order to do that, kernel density estimation (KDE) [15] is used. In this technique, the surrogate model

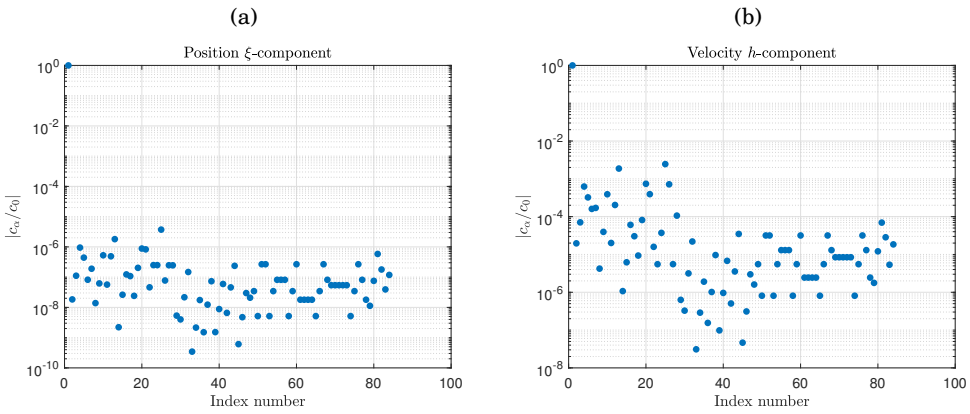



FIGURE 5.8. Normalized PCE coefficients for the PCE-CUT4 for the LUMIO transfer phase final state: (a) $\xi$ component of the position, (b) $h$ component of the velocity.

is exploited to inexpensively produce a number $n$ of samples of the Quantity of Interest $q_j = q\left(\boldsymbol{\chi}(\tau, \boldsymbol{\xi}_j)\right)$, depending on $n$ random variables $\boldsymbol{\xi}_j$, with $j = \{1, \dots, n\}$. Then they are used to estimate the PDF as

$$\hat{\rho}(q) = \frac{1}{nh} \sum_{j=1}^{n} K\left(\frac{q - q_j}{h}\right) \tag{5.31}$$

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

$$\hat{F}(q) = \int_{-\infty}^{q} \hat{\rho}(q)\, \mathrm{d}q = \frac{1}{n} \sum_{j=1}^{n} G\left(\frac{q - q_j}{h}\right) \tag{5.32}$$

where $G(q) = \int_{-\infty}^{q} G(q)\, \mathrm{d}q$. 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* [70] is considered: the value of $h$ is selected as the bandwidth minimizing the mean integrated squared error for a Gaussian distribution. In this case,

$$h = \left(\frac{4\hat{\sigma}^5}{3n}\right)^{\frac{1}{5}} \tag{5.33}$$

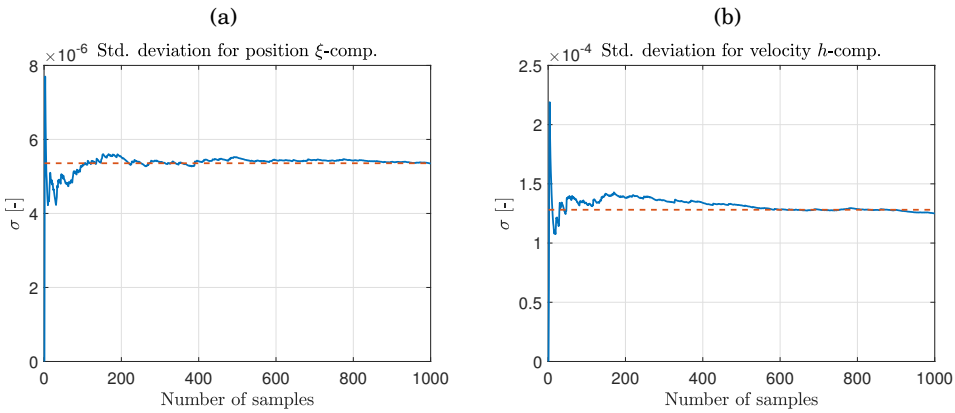



FIGURE 5.9. Standard deviation for the LUMIO transfer phase final state as function of the random sample size of a Monte Carlo simulation: (a) $\xi$ component of the position, (b) $h$ component of the velocity. Dashed line is the PCE-CUT4 solution.

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 $\mathscr{C}^\infty$-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 cumulative distribution function

$$Q(p) = \inf\{q : F(q) \leq p\} \qquad \text{with} \quad 0 \leq p \leq 1 \tag{5.34}$$

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 [69]

$$Q(p,q) = \sum_{j=1}^{n} \frac{1}{nh} K\left[\frac{1}{h}\left(\frac{j}{n} - p\right)\right] \tilde{q}_j \tag{5.35}$$

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 $\mathscr{C}^\infty$-class function.

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

$$\gamma = \sqrt{\boldsymbol{\gamma}^T \boldsymbol{\gamma}}, \qquad \dot{\gamma} = \frac{\boldsymbol{\gamma}^T \boldsymbol{\eta}}{\gamma} \tag{5.36}$$

where $\gamma$ 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 Figure 5.10, the requirements are:

- the Sun exclusion angle $\phi$ should be greater than 0.5 deg in order to avoid degradation in the radiometric observable and, in turn, in the trajectory knowledge [90];
- 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 meters) 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

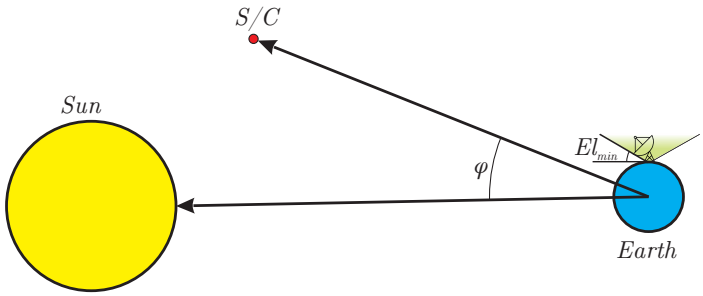



FIGURE 5.10. Tracking problem geometry.

will be made available[1].
SDSA performances are summarized in Table 5.7.

TABLE 5.7. 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 ($1\sigma$) | 130 m |
| Range measurements systematic error | 130 m |
| Doppler measurements random error ($1\sigma$) | 0.2 mm/s |
| Doppler measurements systematic error | 0 mm/s |
| Minimum ground station elevation $El_{\min}$ | 15° |

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)* [51, 84]. This method exploits the capability of PCE to generate inexpensively huge ensembles of samples [87]. 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 discretize in evenly

[1] `https://www.asi.it/en/the-agency/the-space-centers/sardina-deep-space-antenna-sdsa/` (Last accessed on July 10, 2020)

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,

$$\begin{aligned} \boldsymbol{\chi}_{k+1} &= \boldsymbol{\varphi}\left(\boldsymbol{\chi}_k, \tau_k; \tau_{k+1}\right) \\ \hat{\boldsymbol{\chi}}_{k+1} &= \boldsymbol{\varphi}\left(\hat{\boldsymbol{\chi}}_k, \tau_k; \tau_{k+1}\right) \\ \boldsymbol{\chi}^q_{k+1} &= \boldsymbol{\varphi}\left(\boldsymbol{\chi}^q_k, \tau_k; \tau_{k+1}\right), \qquad q \in \{0, 1, \ldots, M\} \end{aligned} \tag{5.37}$$

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

$$\mathbf{z} = \mathbf{h}\left(\boldsymbol{\chi}\right) + \boldsymbol{\varepsilon} \tag{5.38}$$

with $\mathbf{z}$ the measurement and $\boldsymbol{\varepsilon}$ is the measurement error. A linear measurement operator is therefore defined as $H = \left.\frac{\partial \mathbf{h}(\boldsymbol{\chi})}{\partial \boldsymbol{\chi}}\right|_{\boldsymbol{\chi}=\hat{\boldsymbol{\chi}}}$.
At each $\tau_k$ the ensemble square-root filter embedded in the orbit determination, the PCE coefficients for the real and estimated state are retrieved

$$\mathbf{c}^{k,-}_{\boldsymbol{\alpha}} = \sum_{q=1}^{M} \boldsymbol{\chi}^q_{k,-} \psi^q_{\boldsymbol{\alpha}} \omega_q \tag{5.39}$$

$$\hat{\mathbf{c}}^{k,-}_{\boldsymbol{\alpha}} = \sum_{q=1}^{M} \hat{\boldsymbol{\chi}}^q_{k,-} \psi^q_{\boldsymbol{\alpha}} \omega_q \tag{5.40}$$

with $-$ indicating the variables before the filter correction. The mean and covariance are estimated exploiting the PCE properties (Eqs.(3.37)–(3.38))

$$\begin{cases} \bar{\hat{\boldsymbol{\chi}}}_{k,-} = \hat{\mathbf{c}}^{k,-}_0 \\ \hat{P}^-_k = \sum_{\substack{\boldsymbol{\alpha} \in \Lambda_{p,d} \\ \boldsymbol{\alpha} \neq \mathbf{0}}} \left(\hat{\mathbf{c}}^{k,-}_{\boldsymbol{\alpha}}\right)\left(\hat{\mathbf{c}}^{k,-}_{\boldsymbol{\alpha}}\right)^T \end{cases} \tag{5.41}$$

together with the real state mean

$$\bar{\boldsymbol{\chi}}_{k,-} = \mathbf{c}^{k,-}_0 \tag{5.42}$$

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

$$\mathbf{z}_k = \mathbf{h}\left(\bar{\boldsymbol{\chi}}_{k,-}\right), \qquad \hat{\mathbf{z}}_k = \mathbf{h}\left(\bar{\hat{\boldsymbol{\chi}}}_{k,-}\right) \tag{5.43}$$

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

$$\hat{\boldsymbol{\chi}}^i_{k,-} = \sum_{\boldsymbol{\alpha}\in\Lambda_{p,d}} \hat{\mathbf{c}}^{k,-}_{\boldsymbol{\alpha}} \psi^i_{\boldsymbol{\alpha}} \tag{5.44}$$

The EnSRF Kalman gains are computed

$$S_k = H_k \hat{P}^-_k H^T_k + R \tag{5.45}$$

$$K_k = \hat{P}^-_k H^T_k S^{-1}_k \tag{5.46}$$

$$\tilde{K}_k = \hat{P}^-_k H^T_k \sqrt{S_k}^{-T} \left(\sqrt{S_k} + \sqrt{R}\right)^{-1} \tag{5.47}$$

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

$$\delta\boldsymbol{\chi}^i_{k,-} = \hat{\boldsymbol{\chi}}^i_{k,-} - \bar{\boldsymbol{\chi}}_{k,-} \tag{5.48}$$

and secondly, mean and deviations are updated as

$$\bar{\hat{\boldsymbol{\chi}}}_{k,+} = \bar{\hat{\boldsymbol{\chi}}}_{k,-} + K_k(\mathbf{z}_k - \hat{\mathbf{z}}_k) \tag{5.49}$$

$$\delta\boldsymbol{\chi}^i_{k,+} = \delta\boldsymbol{\chi}^i_{k,-} - \tilde{K}_k H \delta\boldsymbol{\chi}^i_{k,-} \tag{5.50}$$

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

$$\hat{\boldsymbol{\chi}}^i_{k,+} = \bar{\hat{\boldsymbol{\chi}}}_{k,+} + \delta\boldsymbol{\chi}^i_{k,+} \tag{5.51}$$

and then they are used to update the PCE coefficients by using an inexpensive least-square regression (Eq.(3.31))

$$\hat{\mathbf{c}}^{k,+}_{\boldsymbol{\alpha}} = \left(\boldsymbol{\Psi}^T\boldsymbol{\Psi}\right)^{-1}\boldsymbol{\Psi} X^+_k \tag{5.52}$$

with $X^+_k$ is the corrected state matrix, containing $\hat{\boldsymbol{\chi}}^i_{k,+}$, and $\boldsymbol{\Psi}$ is the measurement matrix as in Eq. (3.32).

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

$$\begin{cases} \bar{\hat{\boldsymbol{\chi}}}_{k,+} = \hat{\mathbf{c}}_{\mathbf{0}}^{k,+} \\ \hat{P}_k^+ = \sum_{\substack{\boldsymbol{\alpha}\in\Lambda_{p,d} \\ \boldsymbol{\alpha}\neq\mathbf{0}}} \left(\hat{\mathbf{c}}_{\boldsymbol{\alpha}}^{k,+}\right)\left(\hat{\mathbf{c}}_{\boldsymbol{\alpha}}^{k,+}\right)^T \end{cases} \tag{5.53}$$

and then projecting it on the generalized conjugated axes space (Eq. (5.29))

$$\hat{\boldsymbol{\chi}}_{k,+}^q = S\boldsymbol{\xi}_q + \bar{\hat{\boldsymbol{\chi}}}_{k,+} \tag{5.54}$$

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

### 5.2.4 Simplifying assumptions

Once the foundational blocks are modeled, combining the timeline (Figure 5.6) 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 ensemble square-root filter are comparable. Thus, the knowledge evolution, and its quenching in the OD phase, has the same trend on both options. This assumption has been checked by performing a knowledge analysis on a nominal transfer trajectory and a bunch of perturbed ones. Results are presented in Figure 5.11, where only a sample

perturbed trajectory is illustrated for brevity's sake. It shows that there is no significant deviation between them.
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 5.12 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.

## 5.3 Statement of the problem

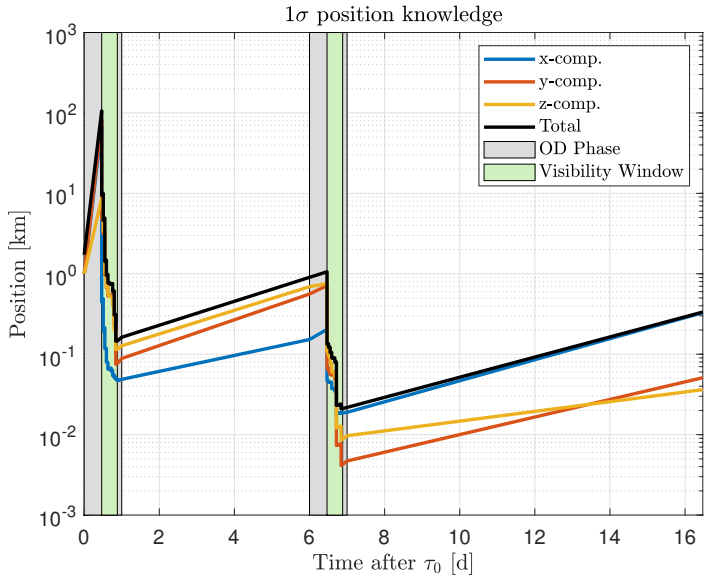
Once the building blocks are established, Problem 0 has to be adapted to cope with Scenario (1) test case, represented by LUMIO transfer phase and the general optimal control problem is converted into a Non-Linear Programming (NLP). Following the trajectory description given in Section 5.1.3, the methodology from Section 5.2 and its timeline, the optimization problem for Scenario (1) under the revised approach can be stated as

**Problem 1** (Scenario (1) Fuel-Optimal Problem)**.** Find the initial and final time, $\tau_0$ and $\tau_f$, the two TCM times, $\tau_{TCM_1}$ and $\tau_{TCM_2}$, the angular parameter vector $\boldsymbol{\alpha}_0$, the SMIM vector $\Delta v_{\text{SMIM}}$, and the nominal trajectory impulse at TCM times, $\Delta v_{\text{TCM}_1}$ and $\Delta v_{\text{TCM}_2}$, such that

$$J = \sum_i \|\Delta v_i\| = \|\Delta v_{\text{SMIM}}\| + \left\|\Delta v_{\text{TCM}_1}\right\| + \left\|\Delta v_{\text{TCM}_2}\right\| + \|\Delta v_{\text{HIM}}\| + \sum_{j=1}^{2} Q(.99, \Delta v_j^s) \tag{5.55}$$

with $Q(0.99, \Delta v_j^s)$ representing the 99-percentile of the stochastic cost computed through Eq. (5.35), is minimized, while the state is subjected to the dynamics

(a)




(b)

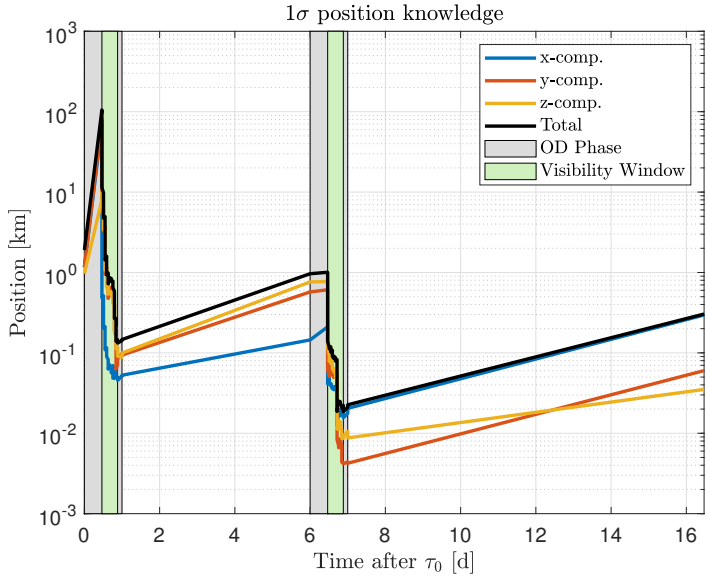



FIGURE 5.11. Comparison for Knowledge Analysis results between (a) nominal and (b) real perturbed trajectory in LUMIO transfer phase.

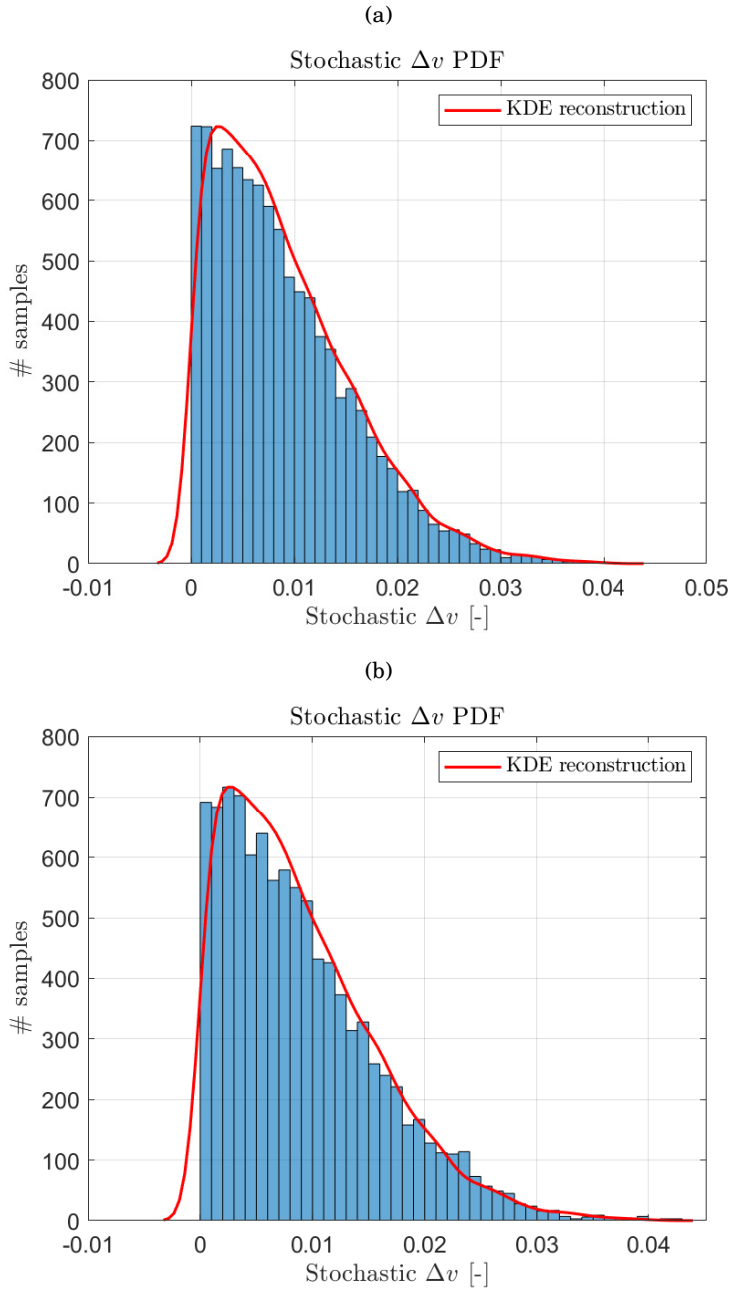



FIGURE 5.12. Comparison of stochastic costs if navigation error is (a) picked randomly, or (b) equal to the error mean in LUMIO case.

illustrated in Eq. (5.12)

$$\boldsymbol{\chi}' = \mathbf{f}(\boldsymbol{\chi}, \tau) = \begin{bmatrix} \boldsymbol{v} \\ \nabla\Omega + \frac{\gamma_0}{\omega^2 k^3}\frac{\boldsymbol{\delta}_S}{\delta_S^3} - \frac{1}{\omega}\left(\frac{2\dot{k}}{k} I + 2C^T\dot{C}\right)\boldsymbol{v} - \frac{1}{\omega^2}\left(\frac{\ddot{k}}{k} I + 2\frac{\dot{k}}{k} C^T\dot{C} + C^T\ddot{C}\right)\boldsymbol{\rho} - \frac{C^T\ddot{\mathbf{b}}}{k\omega^2} \end{bmatrix} \tag{5.56}$$

The HIM is computed as

$$\Delta v_{\text{HIM}} = \boldsymbol{v}^*(\tau_f) - \boldsymbol{v}_\delta(\tau_f) \tag{5.57}$$

with $\boldsymbol{v}^*$ the nominal velocity and $\boldsymbol{\rho}_\delta$ is the target halo velocity.
The state is subjected to initial constraints

$$\begin{cases} E\left[\boldsymbol{\chi}^*(\tau_0)\right] = [\hat{\boldsymbol{\chi}}^*(\tau_0) = \boldsymbol{\chi}_0(\tau_0, \boldsymbol{\alpha}_0) + \Delta v_{\text{SMIM}} \\ E\left[\left(\boldsymbol{\chi}^*(\tau_0) - \boldsymbol{\chi}_0\right)\left(\boldsymbol{\chi}^*(\tau_0) - \boldsymbol{\chi}_0\right)^T\right] = P_0 \end{cases} \tag{5.58}$$

and

$$E\left[\left(\boldsymbol{\chi}(\tau_0) - \boldsymbol{\chi}_0\right)\left(\boldsymbol{\chi}(\tau_0) - \boldsymbol{\chi}_0\right)^T\right] = P_0 \tag{5.59}$$

and a final constraint

$$c = \hat{F}_d(30\,\text{km}) > 0.99 \tag{5.60}$$

with $d = \left\|\boldsymbol{\rho}(\tau_f) - \boldsymbol{\rho}_\delta(\tau_f)\right\|$, being a measure of the distance of the real trajectories from the halo at $\tau_f$, where $\boldsymbol{\rho}_\delta$ is the target halo.
In order to be compliant with on-ground operation requirements, some linear constraints are added

$$2\,\text{d} \leq \left(\tau_{TCM_1} - \tau_0\right) \leq 8\,\text{d} \tag{5.61}$$

$$5.5\,\text{d} \leq \left(\tau_{TCM_2} - \tau_0\right) \leq \tau_f - 3\,\text{d} \tag{5.62}$$

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

$$\Delta v^s = \mathsf{GL}\left(\boldsymbol{\chi}^*, \hat{\boldsymbol{\chi}}, \tau_{TCM}\right) \tag{5.63}$$

and

$$\hat{\boldsymbol{\chi}}\left(\tau_f^{OD}\right) = \mathsf{OD}\left(\boldsymbol{\chi}^*, \hat{\boldsymbol{\chi}}, \tau_0^{OD}, \tau_f^{OD}\right) \tag{5.64}$$

with GL and OD being the Differential Guidance Law (Eq. (4.5)) and orbit determination processes (Section 5.2.3) on the nominal trajectory respectively, $\hat{\boldsymbol{\chi}}$ is the estimated state, $\boldsymbol{\chi}$ the real state and $\boldsymbol{\chi}^*$ is the nominal state.

In this scenario, the quantile value for the $j$-th TCM exploiting the KQE (Eq. (5.35)) is

$$Q\left(0.99, \Delta v_i^s\right) = \sum_{q=1}^{n} \frac{1}{nh} \frac{1}{\sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\frac{1}{h}\left(\frac{i}{n} - 0.99\right)\right)^2\right] \widetilde{\Delta v}_j^{s,i} \tag{5.65}$$

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. (4.5))

$$\Delta v_j^{s,q} = -\left(\Phi_{rv}^T \Phi_{rv} + q\Phi_{vv}^T \Phi_{vv}\right)^{-1} \left(\Phi_{rv}^T \Phi_{rr} + q\Phi_{vv}^T \Phi_{vr}\right) \widetilde{\delta\boldsymbol{\rho}}_j - \widetilde{\delta\boldsymbol{\nu}}_j \tag{5.66}$$

with

$$\widetilde{\delta\boldsymbol{\chi}}_j = \tilde{\boldsymbol{\chi}}_j - \boldsymbol{\chi}^* \tag{5.67}$$

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

$$\tilde{\boldsymbol{\chi}}_j = \varphi\left(\hat{\boldsymbol{\chi}}\left(\tau_f^{OD}\right), \tau_f^{OD}; \tau_j\right) \tag{5.68}$$

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 Section 5.2.4,

$$\hat{\boldsymbol{\chi}}\left(\tau_f^{OD}\right) = \boldsymbol{\chi}\left(\tau_f^{OD}\right) + \left(\bar{\tilde{\boldsymbol{\chi}}}_f^{OD} - \boldsymbol{\chi}\left(\tau_f^{*,OD}\right)\right) \tag{5.69}$$

with the mean final error obtain through Eq. (5.53).
The PCE coefficients can be retrieved as

$$\mathbf{c}_{\boldsymbol{\alpha}} = \sum_{q=1}^{M} \Delta v_j^{s,q} \psi_{\boldsymbol{\alpha}}^q \omega_q \tag{5.70}$$

and then used to obtain the samples

$$\Delta v_{s,i}^j = \sum_{\boldsymbol{\alpha} \in \Lambda_{p,d}} \mathbf{c}_{\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}^i \tag{5.71}$$

where $\psi^i_{\boldsymbol{\alpha}}$ are the basis functions evaluated at a random picked value $\boldsymbol{\xi}_i$. In order to simplify the problem, the bandwidth is considered constant with $h = 0.001$. The value in Eq. (5.60) is obtained in the same way, exploiting the kernel density estimation as in Eq. (5.32) with a constant bandwidth $h = 0.0622$.
In summary, the decision variable vector is defined as $\mathbf{y} = \left[\tau_0, \tau_f, \tau_{TCM_1}, \tau_{TCM_2}, \boldsymbol{\alpha}_0, \Delta v_{\text{SMIM}}, \Delta v_{\text{TCM}_1}, \Delta v_{\text{TCM}_2}\right]^T$ and its bounds are summarized in Table 5.8. No bounds are placed on LLO angular parameters.
The procedure used to estimate the cost function in Eq. (5.55) and the nonlinear constraint in Eq. (5.60) is summarized in Algorithm 1.

TABLE 5.8. Decision vector bounds. $\tau_0$ is the first guess departure date.

| | **Dep. date** | $\tau_f$ [d] | $\tau_{TCM_1}$ [d] | $\tau_{TCM_2}$ [d] | $\Delta v_{\text{TCM}_\text{j}}$ [m/s] |
|---|---|---|---|---|---|
| Upper Bound | $\tau_0 + 7\,\text{d}$ | 28 | 8 | $\tau_f - 3\,\text{d}$ | 10 |
| Lower Bound | $\tau_0 - 7\,\text{d}$ | 15 | 2 | 5.5 | -10 |

### 5.3.1 Optimization

Problem 1 is solved by exploiting a simple shooting technique [13]. This method is selected as the most suitable to solve the Scenario (1) 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, has to be provided. Moreover, in order to simplify some constraints, times are taken as relative times with respect to $\tau_0$.

#### 5.3.1.1 Derivatives

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.

**Algorithm 1:** Integrated approach algorithm for Scenario (1)

**Procedure** INTEGRATED APPROACH
- Define spacecraft and navigation settings; ▷Tables 5.2 and 5.7
- Define the uncertainty properties; ▷Table 5.3
- **Function** INITIALIZATION
  - Evaluate the initial CUT samples $\boldsymbol{\chi}_{k+1}^{q}$ ; ▷See Eq. (5.20)
  - 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
            - Generate the pseudo-measurement; ▷See Eq. (5.36)
            - Apply the EnSRF; ▷See Sec. 5.2.3
            - Get estimated states from the filter;
            - Propagate the estimated state samples to $t_{k+1}$;
          - **end**
          - Propagate the estimated samples up to the final time $t_f^{OD}$;
        - **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. (5.66)
        - Apply the maneuvers to each sample;
      - **case** *OD phase*
        - Propagate the CUT samples up to the final OD time;
        - Estimate the average error; ▷See ORBIT DETERMINATION
        - Evaluate the estimated samples; ▷See Eq. (5.69)
      - **otherwise**
        - Propagate the CUT samples up to the final sub-phase time;
    - **end**
  - **end**
- **Result:** Navigation cost estimate; Final dispersion

**Result:** Cost function (Eq. (5.55)); Dispersion statistic (Eq. (5.60))

**Initial state derivatives.** The initial state $\boldsymbol{\chi}_0$ depends only on the initial time $\tau_0$, 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

$$\frac{\mathrm{d}\boldsymbol{\chi}_0}{\mathrm{d}\tau_0} = \frac{\boldsymbol{\chi}_0(\tau_0+\varepsilon,\boldsymbol{\alpha}_0)-\boldsymbol{\chi}_0(\tau_0-\varepsilon,\boldsymbol{\alpha}_0)}{2\varepsilon} \tag{5.72}$$

$$\frac{\mathrm{d}\boldsymbol{\chi}_0}{\mathrm{d}\boldsymbol{\alpha}_0} = \frac{\boldsymbol{\chi}_0(\tau_0,\boldsymbol{\alpha}_0+\varepsilon)-\boldsymbol{\chi}_0(\tau_0,\boldsymbol{\alpha}_0-\varepsilon)}{2\varepsilon} \tag{5.73}$$

with $\varepsilon = 10^{-6}\max\{1, x\}$, with $x$ a generic variable. Starting from Eq. (5.29), the initial CUT samples derivative can be evaluated as

$$\frac{\mathrm{d}\boldsymbol{\chi}_q(\tau_0)}{\mathrm{d}\mathbf{y}} = \frac{\mathrm{d}S}{\mathrm{d}\mathbf{y}}\boldsymbol{\xi}_q + \frac{\mathrm{d}\bar{\boldsymbol{\chi}}_0}{\mathrm{d}\mathbf{y}} \tag{5.74}$$

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

$$\frac{\mathrm{d}\boldsymbol{\chi}_q(\tau_0)}{\mathrm{d}\tau_0} = \frac{\mathrm{d}\boldsymbol{\chi}_0}{\mathrm{d}\tau_0} \tag{5.75}$$

$$\frac{\mathrm{d}\boldsymbol{\chi}_q(\boldsymbol{\alpha}_0)}{\mathrm{d}\tau_0} = \frac{\mathrm{d}\boldsymbol{\chi}_0}{\mathrm{d}\boldsymbol{\alpha}_0} \tag{5.76}$$

$$\frac{\mathrm{d}\boldsymbol{\chi}_q(\tau_0)}{\mathrm{d}\Delta v_{\mathrm{SMIM}}} = \mathrm{d}S\boldsymbol{\xi}_q + I_v \tag{5.77}$$

with $I_v = \begin{bmatrix} 0_3 & 0_3 \\ 0_3 & I_3 \end{bmatrix}$. Recalling that the initial covariance is function only of $\Delta v_{\mathrm{SMIM}}$ (Eq. (5.24)), in the continuation the symbol d will indicate the derivative with respect to $\Delta v_{\mathrm{SMIM}}$. So

$$\mathrm{d}S = \mathscr{T}\left(S^{-T}\mathrm{d}P_0 S^{-1}\right)S \tag{5.78}$$

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

$$\mathscr{T}_{ij}(A) = \begin{cases} A_{ij} & j > i \\ \frac{1}{2}A_{ij} & i = j \\ 0 & j < i \end{cases} \tag{5.79}$$

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

$$\mathrm{d}P_0 = \frac{\mathrm{d}P_0}{\mathrm{d}\Delta v_{\mathrm{SMIM}}} = \begin{bmatrix} 0_3 & 0_3 \\ 0_3 & \frac{\mathrm{d}P_{\Delta v}}{\mathrm{d}\Delta v_{\mathrm{SMIM}}} \end{bmatrix} \tag{5.80}$$

Starting from Eq. (5.22), it is possible to get

$$\frac{\mathrm{d}P_{\Delta v}}{\mathrm{d}\Delta v_{\mathrm{SMIM}}} = \mathrm{d}J P^s_{\Delta v} J^T + J \mathrm{d}P^s_{\Delta v} J^T + J P^s_{\Delta v} \mathrm{d}J^T \tag{5.81}$$

where

$$\mathrm{d}J = \begin{bmatrix} -\sin\epsilon\cos\alpha\mathrm{d}E - \cos\epsilon\sin\alpha\mathrm{d}A \\ -\sin\epsilon\sin\alpha\mathrm{d}E + \cos\epsilon\cos\alpha\mathrm{d}A \\ \cos\epsilon\mathrm{d}E \end{bmatrix}$$

$$\begin{bmatrix} -\mathrm{d}\Delta v\cos\epsilon\sin\alpha + \Delta v\sin\epsilon\sin\alpha\mathrm{d}E - \Delta v\cos\epsilon\cos\alpha\mathrm{d}A \\ \mathrm{d}\Delta v\cos\epsilon\sin\alpha - \Delta v\sin\epsilon\cos\alpha\mathrm{d}E - \Delta v\cos\epsilon\sin\alpha\mathrm{d}A \\ 0 \end{bmatrix} \tag{5.82}$$

$$\begin{bmatrix} -\mathrm{d}\Delta v\sin\epsilon\cos\alpha - \Delta v\cos\epsilon\cos\alpha\mathrm{d}E + \Delta v\sin\epsilon\sin\alpha\mathrm{d}A \\ -\mathrm{d}\Delta v\sin\epsilon\sin\alpha - \Delta v\cos\epsilon\cos\alpha\mathrm{d}E - \Delta v\sin\epsilon\cos\alpha\mathrm{d}A \\ \mathrm{d}\Delta v\cos\epsilon - \Delta v\sin\epsilon\mathrm{d}E \end{bmatrix}$$

and

$$\mathrm{d}P^s_{\Delta v} = \begin{bmatrix} 2u\Delta v & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{5.83}$$

In the matrix in Eq. (5.82), the missing derivatives can be computed starting from Eqs. (5.21)

$$\mathrm{d}\Delta v = \frac{\left[\Delta v_x, \Delta v_y, \Delta v_z\right]^T}{\Delta v} \tag{5.84a}$$

$$\mathrm{d}E = \left[ -\frac{\Delta v_z}{\Delta v^2}\frac{\Delta v_x}{\sqrt{\Delta v_x^2 + \Delta v_y^2}}, -\frac{\Delta v_z}{\Delta v^2}\frac{\Delta v_y}{\sqrt{\Delta v_x^2 + \Delta v_y^2}}, \frac{\sqrt{\Delta v_x^2 + \Delta v_y^2}}{\Delta v^2} \right]^T \tag{5.84b}$$

$$\mathrm{d}A = \left[ -\frac{\Delta v_y}{\sqrt{\Delta v_x^2 + \Delta v_y^2}}, \frac{\Delta v_x}{\sqrt{\Delta v_x^2 + \Delta v_y^2}}, 0 \right]^T \tag{5.84c}$$

**OD phase derivatives.** Then, the derivatives inside the OD process should be constructed. The derivatives with respect to $\tau_0$, $\tau_{TCM_1}$ and $\Delta v_{\mathrm{TCM}_1}$ are null, while the one in $\tau_{TCM_2}$ and $\Delta v_{\mathrm{TCM}_2}$ 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 $\mathrm{d}\boldsymbol{\chi}_k$, the derivative at the following time can be computed as

$$\mathrm{d}\boldsymbol{\chi}_{k+1} = \Phi(\tau_k, \tau_{k+1})\mathrm{d}\boldsymbol{\chi}_k \tag{5.85}$$

where the subscript $k$ and $k+1$ refers to $\tau_k$ and $\tau_{k+1}$ are two consecutive measurement times and $\Phi$ is the STM. The only exception is done by the derivative with respect to $\tau_0$. In fact, in this case, $\tau_0^{OD}$ and $\tau_f^{OD}$ will vary linearly with $\tau_0$ and thus [24]

$$\begin{cases} \frac{\partial \boldsymbol{\chi}_1}{\partial \tau_0} = \Phi\left(\tau_0^{OD}, \tau_1\right)\left(-\mathbf{f}\left(\boldsymbol{\chi}_0^{OD}, \tau_0^{OD}\right) + \mathrm{d}\boldsymbol{\chi}_0\right) \\ \frac{\partial \boldsymbol{\chi}_{k+1}}{\partial \tau_0} = \Phi(\tau_k, \tau_{k+1})\mathrm{d}\boldsymbol{\chi}_k \\ \frac{\partial \boldsymbol{\chi}_f^{OD}}{\partial \tau_0} = \Phi\left(\tau_M, \tau_f^{OD}\right)\mathrm{d}\boldsymbol{\chi}_M + \mathbf{f}\left(\boldsymbol{\chi}_k^{OD}, \tau_f^{OD}\right) \end{cases} \tag{5.86}$$

with $\tau_M$ the last measurement time.
The estimated state PCE coefficients derivatives are

$$\mathrm{d}\hat{\mathbf{c}}_{\boldsymbol{\alpha}}^{k,-} = \sum_{q=1}^{M} \mathrm{d}\hat{\boldsymbol{\chi}}_{k,-}^{q} \psi_{\boldsymbol{\alpha}}^{q} \omega_q \tag{5.87}$$

and its statistics derivatives

$$\begin{cases} \mathrm{d}\bar{\hat{\boldsymbol{\chi}}}_{k,-} = \mathrm{d}\hat{\mathbf{c}}_0^{k,-} \\ \mathrm{d}\hat{P}_k^- = \sum_{\substack{\boldsymbol{\alpha}\in\Lambda_{p,d} \\ \boldsymbol{\alpha}\neq\mathbf{0}}} \left(\mathrm{d}\hat{\mathbf{c}}_{\boldsymbol{\alpha}}^{k,-}\right)\left(\hat{\mathbf{c}}_{\boldsymbol{\alpha}}^{k,-}\right)^T + \left(\hat{\mathbf{c}}_{\boldsymbol{\alpha}}^{k,-}\right)\left(\mathrm{d}\hat{\mathbf{c}}_{\boldsymbol{\alpha}}^{k,-}\right)^T \end{cases} \tag{5.88}$$

Thay are used to compute the derivatives through the ensemble square-root filter. Kalman gain can be differentiated as

$$\mathrm{d}S_k = \mathrm{d}H_k \hat{P}_k^- H_k^T + H_k \mathrm{d}\hat{P}_k^- H_k^T + H_k \hat{P}_k^- \mathrm{d}H_k^T \tag{5.89}$$

where

$$\mathrm{d}H_k = \begin{bmatrix} \frac{\mathrm{d}\boldsymbol{\rho}_k}{\gamma} - \frac{\boldsymbol{\gamma}}{\gamma^2}\mathrm{d}\gamma & \mathbf{0}_3 \\ \frac{\mathrm{d}\boldsymbol{\nu}_k}{\gamma} - \frac{\boldsymbol{\eta}}{\gamma^2}\mathrm{d}\gamma - \left(\mathrm{d}\boldsymbol{\rho}_k \frac{\dot{\gamma}}{\gamma^2} + \boldsymbol{\gamma}\frac{\mathrm{d}\dot{\gamma}}{\gamma^2} - 2\boldsymbol{\gamma}\frac{\dot{\gamma}\mathrm{d}\gamma}{\gamma^3}\right) & \frac{\mathrm{d}\boldsymbol{\rho}_k}{\gamma} - \frac{\boldsymbol{\gamma}}{\gamma^2}\mathrm{d}\gamma \end{bmatrix} \tag{5.90}$$

with

$$\mathrm{d}\mathbf{z} := \begin{cases} \mathrm{d}\gamma = \frac{\boldsymbol{\gamma}^T \mathrm{d}\boldsymbol{\rho}_k}{\gamma} \\ \mathrm{d}\dot{\gamma} = \left(\frac{\boldsymbol{\eta}}{\gamma} - \boldsymbol{\gamma}\frac{\dot{\gamma}}{\gamma^2}\right)^T \mathrm{d}\boldsymbol{\rho}_k + \frac{\boldsymbol{\gamma}^T \mathrm{d}\mathbf{v}_k}{\gamma} \end{cases} \tag{5.91}$$

Moreover,

$$\mathrm{d}K_k = \mathrm{d}\hat{P}_k^- H_k^T S_k^{-1} + \hat{P}_k^- \mathrm{d}H_k^T S_k^{-1} + \hat{P}_k^- H_k^T \mathrm{d}\left(S_k^{-1}\right) \tag{5.92}$$

where [64]

$$\mathrm{d}\left(S_k^{-1}\right) = -S_k^{-1}\mathrm{d}S_k S_k^{-1} \tag{5.93}$$

The Kalman gain for the deviations can be retrieved by using

$$\begin{aligned} \mathrm{d}\tilde{K}_k &= \mathrm{d}\hat{P}_k^- H_k^T \sqrt{S_k}^{-T}\left(\sqrt{S_k}+\sqrt{R}\right)^{-1} + \hat{P}_k^- \mathrm{d}H_k^T \sqrt{S_k}^{-T}\left(\sqrt{S_k}+\sqrt{R}\right)^{-1} \\ &+ \hat{P}_k^- H_k^T \mathrm{d}\left(\sqrt{S_k}^{-1}\right)^T \left(\sqrt{S_k}+\sqrt{R}\right)^{-1} + \hat{P}_k^- H_k^T \sqrt{S_k}^{-T} \mathrm{d}\left(\left(\sqrt{S_k}+\sqrt{R}\right)^{-1}\right) \end{aligned} \tag{5.94}$$

where

$$\mathrm{d}\left(\sqrt{S_k}^{-1}\right) = -\sqrt{S_k}^{-1}\mathrm{d}\sqrt{S_k}\sqrt{S_k}^{-1} \tag{5.95}$$

and

$$\mathrm{d}\left(\left(\sqrt{S_k}+\sqrt{R}\right)^{-1}\right) = -\left(\sqrt{S_k}+\sqrt{R}\right)^{-1}\mathrm{d}\sqrt{S_k}\left(\sqrt{S_k}+\sqrt{R}\right)^{-1} \tag{5.96}$$

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

$$\mathrm{d}\sqrt{S_k} = \mathrm{mat}\left(\left(\sqrt{S_k}^T \oplus \sqrt{S_k}\right)^{-1} \mathrm{vec}(\mathrm{d}S_k)\right) \tag{5.97}$$

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

$$\mathrm{d}\bar{\tilde{\boldsymbol{\chi}}}_{k,+} = \mathrm{d}\bar{\tilde{\boldsymbol{\chi}}}_{k,-} + \mathrm{d}K_k\left(\mathbf{z}_k - \hat{\mathbf{z}}_k\right) + K_k\left(\mathrm{d}\mathbf{z}_k - \mathrm{d}\hat{\mathbf{z}}_k\right) \tag{5.98}$$

$$\mathrm{d}\delta\boldsymbol{\chi}_{k,+}^i = \mathrm{d}\delta\boldsymbol{\chi}_{k,-}^i - \mathrm{d}\tilde{K}_k H \delta\boldsymbol{\chi}_{k,-}^i + \tilde{K}_k \mathrm{d}H\delta\boldsymbol{\chi}_{k,-}^i + \tilde{K}_k H \delta\mathrm{d}\boldsymbol{\chi}_{k,-}^i \tag{5.99}$$

where

$$\mathrm{d}\delta\boldsymbol{\chi}_{k,-}^i = \mathrm{d}\hat{\boldsymbol{\chi}}_{k,-}^i - \mathrm{d}\bar{\boldsymbol{\chi}}_{k,-} \tag{5.100}$$

and

$$\mathrm{d}\hat{\boldsymbol{\chi}}^i_{k,-} = \sum_{\boldsymbol{\alpha}\in\Lambda_{p,d}} \mathrm{d}\hat{\mathbf{c}}^{k,-}_{\boldsymbol{\alpha}} \psi^i_{\boldsymbol{\alpha}} \tag{5.101}$$

The $n$ realizations of $\boldsymbol{\xi}_i$, in practice, are generated at the beginning of the optimization algorithm and keep fixed for all the optimization method steps. Thus $\psi^i_{\boldsymbol{\alpha}}$ is a constant. Finally, the derivative of the corrected realizations is

$$\mathrm{d}\hat{\boldsymbol{\chi}}^i_{k,+} = \mathrm{d}\bar{\hat{\boldsymbol{\chi}}}_{k,+} + \mathrm{d}\delta\boldsymbol{\chi}^i_{k,+} \tag{5.102}$$

While the updated PCE coefficients have derivative

$$\mathrm{d}\hat{\mathbf{c}}^{k,+}_{\boldsymbol{\alpha}} = \left(\boldsymbol{\Psi}^T\boldsymbol{\Psi}\right)^{-1}\boldsymbol{\Psi}\mathrm{d}X^+_k \tag{5.103}$$

where $\mathrm{d}X^+_k$ is the collection matrix of $\mathrm{d}\hat{\boldsymbol{\chi}}^i_{k,+}$.
As last point, the new cubature points derivative is computed

$$\begin{cases} \mathrm{d}\bar{\hat{\boldsymbol{\chi}}}_{k,+} = \mathrm{d}\hat{\mathbf{c}}^{k,+}_0 \\ \mathrm{d}\hat{P}^+_k = \sum_{\substack{\boldsymbol{\alpha}\in\Lambda_{p,d} \\ \boldsymbol{\alpha}\neq\mathbf{0}}} \left(\mathrm{d}\hat{\mathbf{c}}^{k,+}_{\boldsymbol{\alpha}}\right)\left(\hat{\mathbf{c}}^{k,+}_{\boldsymbol{\alpha}}\right)^T + \left(\hat{\mathbf{c}}^{k,+}_{\boldsymbol{\alpha}}\right)\left(\mathrm{d}\hat{\mathbf{c}}^{k,+}_{\boldsymbol{\alpha}}\right)^T \end{cases} \tag{5.104}$$

and eventually

$$\mathrm{d}\hat{\boldsymbol{\chi}}^q_{k,+} = \mathrm{d}S\boldsymbol{\xi}_q + \mathrm{d}\bar{\hat{\boldsymbol{\chi}}}_{k,+} \tag{5.105}$$

with the derivative of Cholesky decomposition of $\hat{P}^+_k$ computed as in Eq. (5.78)

$$\mathrm{d}S = \mathscr{T}\left(S^{-T}\mathrm{d}\hat{P}^+_k S^{-1}\right)S \tag{5.106}$$

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

$$\mathrm{d}\tilde{\boldsymbol{\chi}}^q\left(\tau^{OD}_f\right) = \mathrm{d}\boldsymbol{\chi}^q\left(\tau^{OD}_f\right) + \left(\mathrm{d}\bar{\hat{\boldsymbol{\chi}}}^{OD}_f - \mathrm{d}\boldsymbol{\chi}^{*,OD}_f\right) \tag{5.107}$$

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

**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. (4.5)

$$D = -\left(\Phi^T_{rv}\Phi_{rv} + q\Phi^T_{vv}\Phi_{vv}\right)^{-1}\left(\Phi^T_{rv}\Phi_{rr} + q\Phi^T_{vv}\Phi_{vr}\right) \tag{5.108}$$

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

$$\mathrm{d}\boldsymbol{\chi}_{TCM}^{q} = \Phi\left(\tau_f^{OD}, \tau_{TCM}\right)\mathrm{d}\boldsymbol{\chi}^{q}\left(\tau_f^{OD}\right) + \begin{bmatrix} \mathbf{0}_3 \\ \mathrm{d}\Delta v^{s,q} \end{bmatrix} \tag{5.109}$$

where d represents the generic partial derivative, and

$$\mathrm{d}\Delta v^{s,q} = \mathrm{d}D\delta\boldsymbol{\rho}_{TCM} + D\mathrm{d}\delta\boldsymbol{\rho}_{TCM} - \mathrm{d}\delta\boldsymbol{\nu}_{TCM} \tag{5.110}$$

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

$$\begin{aligned} \begin{bmatrix} \mathrm{d}\delta\boldsymbol{\rho}_{TCM} \\ \mathrm{d}\delta\boldsymbol{\nu}_{TCM} \end{bmatrix} &= \begin{bmatrix} \mathrm{d}\tilde{\boldsymbol{\rho}}_{TCM} - \mathrm{d}\boldsymbol{\rho}_{TCM}^{*} \\ \mathrm{d}\tilde{\boldsymbol{\nu}}_{TCM} - \mathrm{d}\boldsymbol{\nu}_{TCM}^{*} \end{bmatrix} \\ = \tilde{\Phi}\left(\tau_f^{OD}, \tau_{TCM}\right)\mathrm{d}\tilde{\boldsymbol{\chi}}^{q}\left(\tau_f^{OD}\right) &- \Phi^{*}\left(\tau_f^{OD}, \tau_{TCM}\right)\mathrm{d}\boldsymbol{\chi}^{*}\left(\tau_f^{OD}\right) \end{aligned} \tag{5.111}$$

where $\Phi^*$ and $\tilde{\Phi}$ are the TCM of the nominal and the estimated trajectory respectively, while $\mathrm{d}\tilde{\boldsymbol{\chi}}\left(\tau_f^{OD}\right)$ is computed in Eq. (5.107).
The derivative of the Guidance Matrix is

$$\mathrm{d}D = \mathrm{d}\left(Q^{-1}\right)\left(\Phi_{rv}^{T}\Phi_{rr} + q\Phi_{vv}^{T}\Phi_{vr}\right) + Q^{-1}\left(\mathrm{d}\Phi_{rv}^{T}\Phi_{rr} + \Phi_{rv}^{T}\mathrm{d}\Phi_{rr} + q\left(\mathrm{d}\Phi_{vv}^{T}\Phi_{vr} + \Phi_{vv}^{T}\mathrm{d}\Phi_{vr}\right)\right) \tag{5.112}$$

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

$$\mathrm{d}Q = -\left(\mathrm{d}\Phi_{rv}^{T}\Phi_{rv} + \Phi_{rv}^{T}\mathrm{d}\Phi_{rv} + q\left(\mathrm{d}\Phi_{vv}^{T}\Phi_{vv} + \mathrm{d}\Phi_{vv}^{T}\Phi_{vv}\right)\right) \tag{5.113}$$

which is used to compute

$$\mathrm{d}\left(Q^{-1}\right) = -Q^{-1}\mathrm{d}QQ^{-1} \tag{5.114}$$

The derivative of the state transition matrix 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

$$\boldsymbol{\Phi} = \frac{1}{\varepsilon}\left[\frac{1}{12}\Phi\left(\boldsymbol{\chi} - 2\varepsilon\right) - \frac{2}{3}\Phi\left(\boldsymbol{\chi} - \varepsilon\right) + \frac{2}{3}\Phi\left(\boldsymbol{\chi} + \varepsilon\right) - \frac{1}{12}\Phi\left(\boldsymbol{\chi} + 2\varepsilon\right)\right] \tag{5.115}$$

where $\varepsilon = \max\{\epsilon_m, 10^{-6}\boldsymbol{\chi}_0\}$, with $\epsilon_m$ is the machine epsilon, and $\boldsymbol{\chi} = \boldsymbol{\chi}^{*}\left(\tau_f^{OD}\right)$. 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

$$\mathrm{d}\Phi = \mathbf{\Phi}\mathrm{d}\boldsymbol{\chi}^*\left(\tau_f^{OD}\right) \tag{5.116}$$

The derivative with respect to $\tau_0$ can be computed by following the same steps in Eqs. (5.110)–(5.116). However, since all the times times are linearly dependent from $\tau_0$, the derivative of the estimated and nominal state (in Eq. (5.111)) are

$$\frac{\partial \tilde{\boldsymbol{\chi}}_{TCM}^q}{\partial \tau_0} = -\tilde{\Phi}\left(\tau_f^{OD}, \tau_{TCM}\right)\left(\mathbf{f}\left(\tilde{\boldsymbol{\chi}}^q, \tau_f^{OD}\right) - \frac{\partial \tilde{\boldsymbol{\chi}}^q\left(\tau_f^{OD}\right)}{\partial \tau_0}\right) + \mathbf{f}\left(\tilde{\boldsymbol{\chi}}^q, \tau_{TCM}\right) \tag{5.117}$$

and

$$\frac{\partial \boldsymbol{\chi}_{TCM}^*}{\partial \tau_0} = -\Phi^*\left(\tau_f^{OD}, \tau_{TCM}\right)\left(\mathbf{f}\left(\boldsymbol{\chi}^*, \tau_f^{OD}\right) - \frac{\partial \boldsymbol{\chi}^*\left(\tau_f^{OD}\right)}{\partial \tau_0}\right) + \mathbf{f}\left(\boldsymbol{\chi}^*, \tau_{TCM}\right) \tag{5.118}$$

The state transition matrix derivative inside the DG formula is in turn

$$\frac{\partial \Phi}{\partial \tau_0} = -\Phi A\left(\tau_{TCM_j}\right) + A\left(\tau_{TCM_{j+1}}\right)\Phi - \mathbf{\Phi}\left(\mathbf{f}\left(\boldsymbol{\chi}^*, \tau_{TCM_j}\right) - \frac{\partial \boldsymbol{\chi}^*\left(\tau_f^{OD}\right)}{\partial \tau_0}\right) \tag{5.119}$$

where $A$ is the dynamics Jacobian as in Eq. (5.14). 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. (5.110)–(5.116) again. However,

$$\frac{\partial \tilde{\boldsymbol{\chi}}_{TCM}^q}{\partial \tau_{TCM}} = \mathbf{f}\left(\tilde{\boldsymbol{\chi}}^q, \tau_{TCM}\right) \tag{5.120}$$

and

$$\frac{\partial \boldsymbol{\chi}_{TCM}^*}{\partial \tau_{TCM}} = \mathbf{f}\left(\boldsymbol{\chi}^*, \tau_{TCM}\right) \tag{5.121}$$

while the STM derivative is

$$\frac{\partial \Phi}{\partial \tau_{TCM}} = -\Phi A\left(\tau_{TCM}\right) \tag{5.122}$$

The derivative of the state at the end of the cut-off phase with respect to the next TCM time (i.e. $\tau_{TCM_2}$ and $\tau_f$ for the first and second phase respectively) can be retrieved from Eqs. (5.110)–(5.116), considering that

$$\frac{\partial \tilde{\boldsymbol{\chi}}^q_{TCM}}{\partial \tau_{TCM_{j+1}}} = \mathbf{0}_6 \tag{5.123}$$

$$\frac{\partial \boldsymbol{\chi}^*_{TCM}}{\partial \tau_{TCM_{j+1}}} = \mathbf{0}_6 \tag{5.124}$$

and

$$\frac{\partial \Phi}{\partial \tau_{TCM_{j+1}}} = A\left(\tau_{TCM_{j+1}}\right)\Phi \tag{5.125}$$

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

**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

$$\mathrm{d}\boldsymbol{\chi}^q\left(\tau_f^{BP}\right) = \Phi\left(\tau_{TCM}, \tau_f^{BP}\right)\mathrm{d}\boldsymbol{\chi}^q\left(\tau_{TCM}\right) \tag{5.126}$$

with $\tau_f^{BP}$ being the final time of the ballistic state. Eq. (5.126) is valid for every $\mathbf{y}$ element, but $\tau_0$

$$\frac{\partial \boldsymbol{\chi}^q\left(\tau_f^{BP}\right)}{\partial \tau_0} = -\Phi\left(\tau_{TCM}, \tau_f^{BP}\right)\left(\frac{\partial \boldsymbol{\chi}^q\left(\tau_{TCM}\right)}{\partial \tau_0} - \mathbf{f}\left(\tilde{\boldsymbol{\chi}}^q, \tau_{TCM}\right)\right) \tag{5.127}$$

and $\tau_f^{BP}$ (i.e. $\tau_0^{OD}$ for the first ballistic phase and $\tau_f$ for the second one)

$$\frac{\partial \boldsymbol{\chi}^q\left(\tau_f^{BP}\right)}{\partial \tau_f^{BP}} = \Phi\left(\tau_{TCM}, \tau_f^{BP}\right)\frac{\partial \boldsymbol{\chi}^q\left(\tau_{TCM}\right)}{\partial \tau_f^{BP}} + \mathbf{f}\left(\tilde{\boldsymbol{\chi}}^q, \tau_f^{BP}\right) \tag{5.128}$$

**Cost function gradient.** The cost function gradient is defined as

$$\mathbf{g}(\mathbf{y}) = \nabla J = \frac{\partial}{\partial \mathbf{y}}\left(\|\Delta v_{\mathrm{SMIM}}\| + \left\|\Delta v_{\mathrm{TCM}_1}\right\| + \left\|\Delta v_{\mathrm{TCM}_2}\right\| + \|\Delta v_{\mathrm{HIM}}\| + \sum_{j=1}^{2} Q\left(0.99, \Delta v_j^s\right)\right) \tag{5.129}$$

Ana 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 [64]

$$\frac{\partial \|\Delta v_{\mathrm{SMIM}}\|}{\partial \Delta v_{\mathrm{SMIM}}} = \frac{\Delta v_{\mathrm{SMIM}}}{\|\Delta v_{\mathrm{SMIM}}\|} \tag{5.130}$$

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

$$\frac{\partial \left\|\Delta v_{\mathrm{TCM}_j}\right\|}{\partial \Delta v_{\mathrm{TCM}_j}} = \frac{\Delta v_{\mathrm{TCM}_j}}{\left\|\Delta v_{\mathrm{TCM}_j}\right\|} \tag{5.131}$$

Moreover, starting from Eq. (5.57), it can be found that

$$\frac{\partial \|\Delta v_{\mathrm{HIM}}\|}{\partial \tau_f} = \frac{\Delta v_{\mathrm{HIM}}}{\|\Delta v_{\mathrm{HIM}}\|}\left(\boldsymbol{\nu}'\left(\boldsymbol{\chi}^*, \tau_f\right) - \frac{\mathrm{d}\boldsymbol{\nu}_\delta}{\mathrm{d}\tau_f}\right) \tag{5.132}$$

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

$$\mathrm{d}\|\Delta v_{\mathrm{HIM}}\| = \frac{\Delta v_{\mathrm{HIM}}}{\|\Delta v_{\mathrm{HIM}}\|}\mathrm{d}\boldsymbol{\nu}\left(\tau_f\right) \tag{5.133}$$

where the value of $\mathrm{d}\boldsymbol{\nu}\left(\tau_f\right)$ can be retrieved from Eq. (5.128).
Finally, the generic derivative for the stochastic cost should be estimated. Starting from Eq. (5.65), the generic derivative can be computed as

$$\mathrm{d}Q = \sum_{q=1}^{n} \frac{1}{nh}\frac{1}{\sqrt{2\pi}}\exp\left[-\frac{1}{2}\left(\frac{1}{h}\left(\frac{i}{n} - 0.99\right)\right)^2\right]\mathrm{d}\widetilde{\Delta v}_j^{s,i} \tag{5.134}$$

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

$$\mathrm{d}\widetilde{\Delta v}_j^{s,i} = S\mathrm{d}\left\|\Delta v_j^{s,i}\right\| \tag{5.135}$$

where $S$ is the sorting matrix and

$$\mathrm{d}\left\|\Delta v_j^{s,i}\right\| = \frac{\Delta v_j^{s,i}}{\left\|\Delta v_j^{s,i}\right\|} \tag{5.136}$$

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

$$\mathrm{d}\Delta v_j^{s,i} = \sum_{\boldsymbol{\alpha}\in\Lambda_{p,d}} \mathrm{d}\mathbf{c}_{\boldsymbol{\alpha}}\psi_{\boldsymbol{\alpha}}^i \tag{5.137}$$

with

$$\mathrm{d}\mathbf{c}_{\boldsymbol{\alpha}} = \sum_{q=1}^{M} \mathrm{d}\Delta v_j^{s,q} \psi_{\boldsymbol{\alpha}}^{q} \omega_q \tag{5.138}$$

The derivative of the trajectory correction maneuvers CUT samples is evaluated through Eq. (5.110).

**Inequality constraint Jacobian.** The inequality constraint in Eq. (5.60) should be reformulated as

$$c = 0.99 - \hat{F}(d - 30\,\mathrm{km}) < 0 \tag{5.139}$$

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

$$G_c(\mathbf{y}) = \frac{\partial c}{\partial \mathbf{y}} = -\frac{\partial \hat{F}_d(30\,\mathrm{km})}{\partial \mathbf{y}} \tag{5.140}$$

Starting from the definition of cumulative distribution function given by the KDE

$$\hat{F}_d(30\,\mathrm{km}) = \frac{1}{n} \sum_{i=1}^{n} \left[ \frac{1}{2} \left( 1 + \mathrm{erf}\left( \frac{d_\delta - d_i}{h} \right) \right) \right] \tag{5.141}$$

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

$$\mathrm{d}\hat{F}_d(d_\delta) = \frac{1}{nh} \sum_{i=1}^{n} \frac{1}{\sqrt{2\pi}} \exp\left( -\frac{1}{2} \left( \frac{d_\delta - d_i}{h} \right)^2 \right) \mathrm{d}(d_i) \tag{5.142}$$

The derivative of the distance at the final point is

$$\mathrm{d}(d_i) = \frac{\boldsymbol{\rho}_i(\tau_f) - \boldsymbol{\rho}_\delta(\tau_f)}{\left\| \boldsymbol{\rho}_i(\tau_f) - \boldsymbol{\rho}_\delta(\tau_f) \right\|} \left( \mathrm{d}\boldsymbol{\rho}_i(\tau_f) - \mathrm{d}\boldsymbol{\rho}_\delta(\tau_f) \right) \tag{5.143}$$

The derivative of the halo distance is always null, except

$$\frac{\partial \boldsymbol{\rho}_\delta(\tau_f)}{\partial \tau_f} = \boldsymbol{\nu}_\delta(\tau_f) \tag{5.144}$$

while

$$\mathrm{d}\boldsymbol{\rho}_i(\tau_f) = \sum_{\boldsymbol{\alpha} \in \Lambda_{p,d}} \mathrm{d}\mathbf{c}_{\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}^{i} \tag{5.145}$$

with

$$\mathrm{d}\mathbf{c}_{\boldsymbol{\alpha}} = \sum_{q=1}^{M} \mathrm{d}\boldsymbol{\rho}(\tau_f) \psi_{\boldsymbol{\alpha}}^{q} \omega_q \tag{5.146}$$

where $\mathrm{d}\boldsymbol{\rho}(\tau_f)$ is evaluated through Eq. (5.126).

## 5.4 Results

In this Section, the results for the Scenario (1) are presented. After an introduction on the first guesses generation, the solutions are presented and comparison between them and traditional approach solutions is shown.

### 5.4.1 First guess

An educated guess is required to solve the optimization problem for the Scenario (1), 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 1. The algorithm is summarized in Figure 5.13.

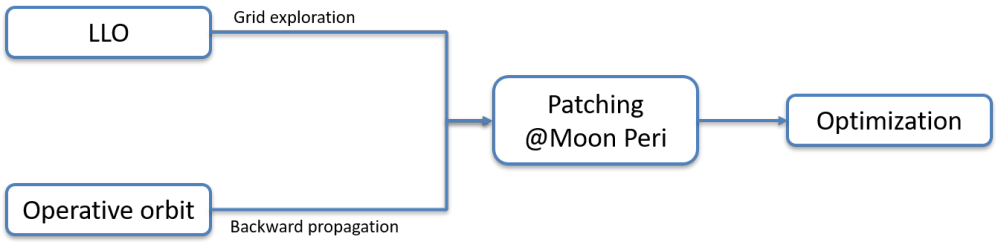



FIGURE 5.13. First guesses generation algorithm.

#### 5.4.1.1 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 Roto-Pulsating Resticted $n$-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 hours, 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/s $\Delta v$ is applied along the minimum stretching direction. Because in the high-fidelity model the operative orbit is no longer periodic, the minimum stretching direction $\boldsymbol{\delta}$ is defined as the direction along which for a given perturbation $\delta\boldsymbol{\chi}_0$ the Euclidian norm of the final perturbation is minimized, i.e.,

$$\boldsymbol{\delta} = \arg\min_{\delta\boldsymbol{\chi}_0} \left\| \delta\boldsymbol{\chi}_f \right\| = \arg\min_{\delta\boldsymbol{\chi}_0} \sqrt{\delta\boldsymbol{\chi}_0^T C \delta\boldsymbol{\chi}_0} \tag{5.147}$$

where $C = \Phi_{\tau_0,\tau_f}^T \Phi_{\tau_0,\tau_f}$ is the Cauchy–Green tensor, defined by exploiting the STM from $\tau_0$ to $\tau_f$. Starting from Eq. (5.147), 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 Circular Restricted 3-Body Problem (CR3BP) stable manifolds. Figure 5.14 shows some trajectories propagated backward from the halo orbit.

#### 5.4.1.2 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 5.9.

#### 5.4.1.3 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

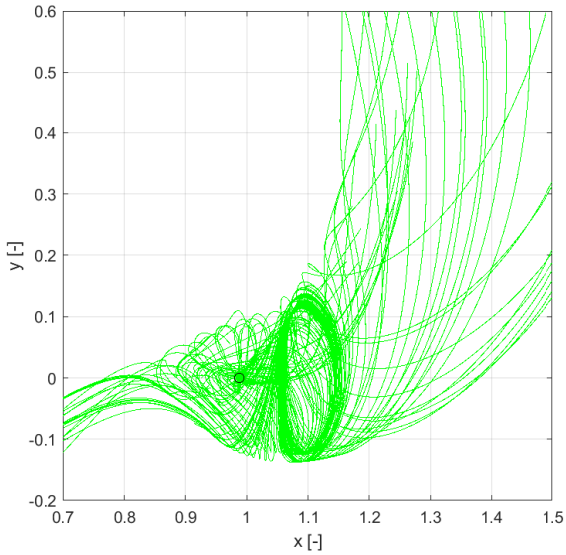



FIGURE 5.14. Sample trajectories propagated backward in time from the target quasi-halo orbit in Earth–Moon synodic frame. Black dot is the Moon. Distances are in Earth–Moon mean distance.

TABLE 5.9. Keplerian elements bounds in MCME2000 for grid search.

| | $i$ [deg] | $\Omega$ [deg] | $\omega$ [deg] |
|---|---|---|---|
| **Lower bound** | 0 | 0 | 0 |
| **Upper bound** | 180 | 360 | 360 |
| **Step** | 20 | 45 | 45 |

patching criterion. Figure 5.15 shows results of the patching process. Starting from the 321 initial guesses, only 9 have a pericenter distance lower than 250 km.

#### 5.4.1.4 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, $\tau_0$ and $\tau_f$, the LLO Keplerian parameters $\boldsymbol{\alpha}_0$ and the $\Delta v_{\text{HIM}}$ such that

$$J = \frac{1}{2}\|\Delta v_{\text{SMIM}}\|^2 + \frac{1}{2}\|\Delta v_{\text{HIM}}\|^2 \tag{5.148}$$

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

$$\boldsymbol{\rho}_{SM}(\tau_0) - \boldsymbol{\rho}_0(\tau_0, \boldsymbol{\alpha}_0) = 0 \tag{5.149}$$

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

$$\Delta v_{\text{SMIM}} = \boldsymbol{\nu}_{SM}(\tau_0) - \boldsymbol{\nu}_0(\tau_0, \boldsymbol{\alpha}_0) \tag{5.150}$$

with the state on the LLO computed through Eq. (5.19).
Figure 5.15 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 5.10. Althogh the time of flight is similar for the four optimal solutions, the required propellant amount varied widely, going from only 67.33 m/s to 83.11 m/s, being one third higher than the minimum.

TABLE 5.10. First guess optimal solutions whit $\Delta v \leq 120$ m/s.

| # | **Departure date** | $ToF$ [d] | $\Delta v_{TOT}$ [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 |

### 5.4.2 Solutions

The trajectories listed in Table 5.10 are used as first guesses for Problem 1. 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 minutes.

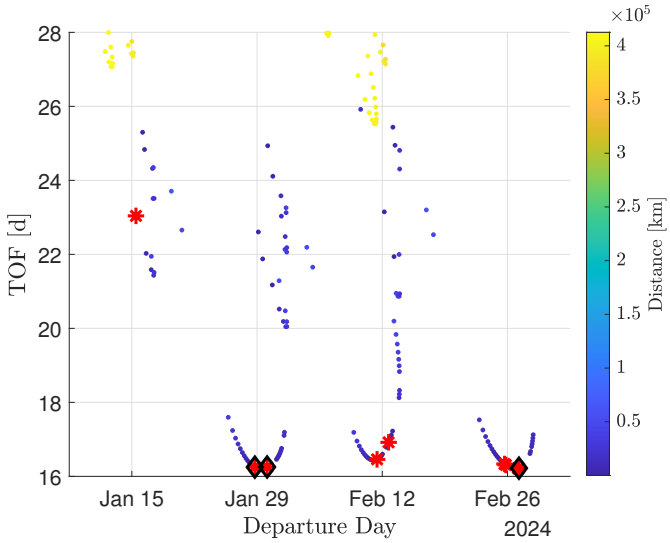



FIGURE 5.15. Pericenter patching process and optimization results. Red asterisks show solutions with $d < 250\,\text{km}$. Black diamonds are optimized solutions with $\Delta v_{\text{SMIM}} < 120\,\text{m/s}$.

Since CUT samples can be propagated forward independently, the runtime can be easily reduced by exploiting parallelization on a multi-core workstation. However, the RAM can be a showstopper, since up to 4 GB can be required.

Results are summarized in Table 5.11. 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 (Figure 5.16) shows a similar trend for both Solution #53 and #64, the velocity dispersion (Figure 5.17) 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 $\Delta v$ 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 the integrated approach give different results, leading to a wrong choice of the nominal orbit if the stochastic optimization is not performed.

TABLE 5.11. Integrated approach optimal solutions. Subscript $D$ stay for deterministic, while $S$ is stochastic. The asterisk is used for the value after the optimization.

| # | $\Delta v_D$ [m/s] | $\Delta v^0_{D+S}$ [m/s] | $\Delta v^*_{D+S}$ [m/s] |
|---|---|---|---|
| 53 | 74.00 | 102.60 | 100.14 |
| 64 | 67.33 | 108.8 | 108.75 |
| 164 | 83.11 | 111.88 | 111.75 |
| 289 | 68.65 | 125.47 | 101.91 |

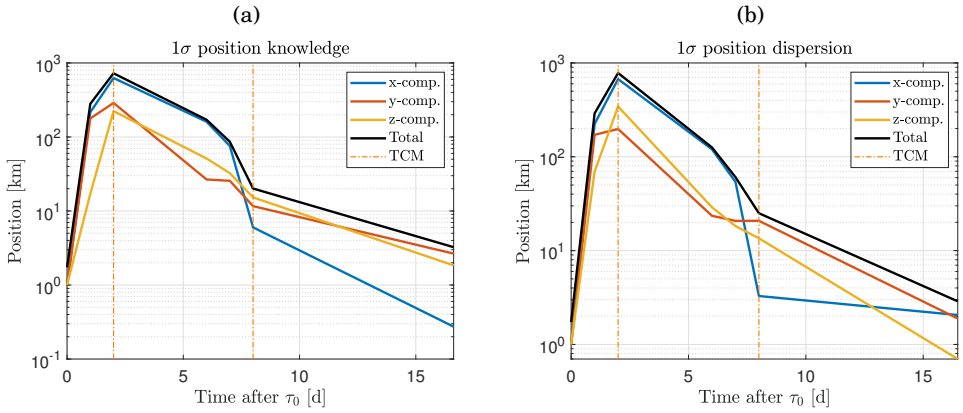



FIGURE 5.16. Position $1\sigma$ dispersion evolution for the optimized cases: (a) Trajectory #53, (b) Trajectory #64.

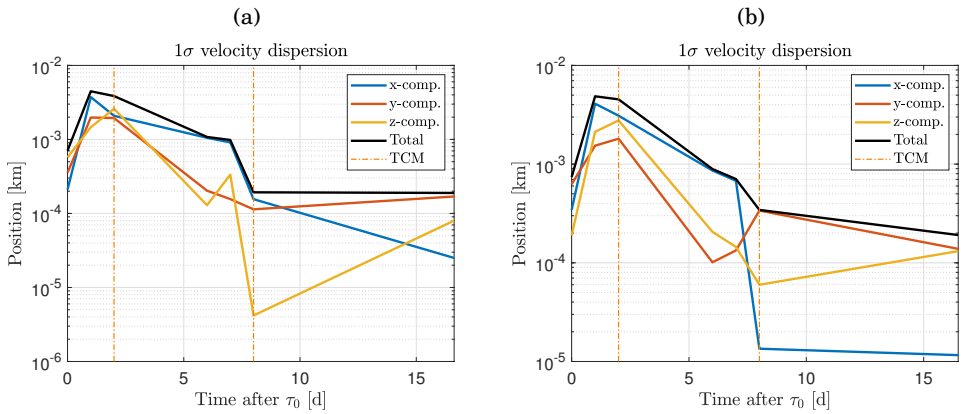



FIGURE 5.17. Velocity $1\sigma$ dispersion evolution for the optimized cases: (a) Trajectory #53, (b) Trajectory #64.

The decision variables of Problem 1 for Solution #53 are listed in Table 5.12, while the maneuvers magnitude is provided in Table 5.16. Variables for the other three solutions are showed in Tables 5.13–5.15. It is important to notice that the change in the decision variables between the first guesses and the optimization solutions is limited. This is the result of a good first guess choice. Only solution #289 has a greater variables variation and this is reflected in a greater reduction in the total optimized $\Delta v$ (see Table 5.11). Figure 5.18 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 Figure 5.19.

TABLE 5.12. Solution #53 decision vector.

| | **Departure date** | $\tau_f$ [d] | $\tau_{TCM_1}$ [d] | $\tau_{TCM_2}$ [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] | $\Omega_0$ [deg] | $\omega_0$ [d] | $\theta_0$ [d] |
| First guess | 98.15 | 34.67 | 63.96 | 319.75 |
| Optimized | 99.79 | 38.29 | 64.20 | 321.75 |

TABLE 5.13. Solution #64 decision vector.

| | **Departure date** | $\tau_f$ [d] | $\tau_{TCM_1}$ [d] | $\tau_{TCM_2}$ [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] | $\Omega_0$ [deg] | $\omega_0$ [d] | $\theta_0$ [d] |
| First guess | 130.06 | 225.08 | 60.15 | 22.58 |
| Optimized | 130.05 | 225.17 | 60.14 | 22.72 |

TABLE 5.14. Solution #164 decision vector.

| | **Departure date** | $\tau_f$ [d] | $\tau_{TCM_1}$ [d] | $\tau_{TCM_2}$ [d] |
|---|---|---|---|---|
| First guess | 11 FEB 2024 16:47 | 16.71 | 2.00 | 8.00 |
| Optimized | 11 FEB 2024 16:47 | 16.71 | 2.00 | 8.03 |
| | $i_0$ [deg] | $\Omega_0$ [deg] | $\omega_0$ [d] | $\theta_0$ [d] |
| First guess | 97.68 | 221.30 | 70.56 | 33.47 |
| Optimized | 98.70 | 228.87 | 63.63 | 31.02 |

TABLE 5.15. Solution #289 decision vector.

| | **Departure date** | $\tau_f$ [d] | $\tau_{TCM_1}$ [d] | $\tau_{TCM_2}$ [d] |
|---|---|---|---|---|
| First guess | 27 FEB 2024 03:04 | 15.92 | 2.00 | 8.00 |
| Optimized | 27 FEB 2024 02:22 | 20.55 | 2.00 | 8.00 |
| | $i_0$ [deg] | $\Omega_0$ [deg] | $\omega_0$ [d] | $\theta_0$ [d] |
| First guess | 261.41 | 348.8 | 316.11 | 19.15 |
| Optimized | 263.08 | 348.27 | 313.97 | 18.59 |

TABLE 5.16. Solution #53 maneuvers magnitude.

| $\Delta v_{\text{SMIM}}$ [m/s] | $\Delta v_{\text{HIM}}$ [m/s] | $Q(.99, \Delta v_1^s)$ | $Q(.99, \Delta v_2^s)$ | **Total [m/s]** |
|---|---|---|---|---|
| 68.73 | 3.50 | 25.24 | 2.66 | 100.14 |

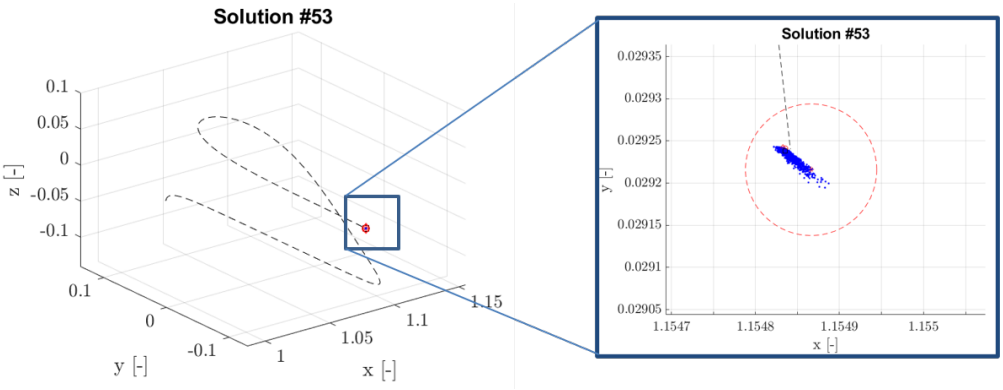



FIGURE 5.18. Transfer phase trajectory for Solution #53. In the close-up, the final dispersion is showed. The red circle represents the final constraint 30 km sphere.

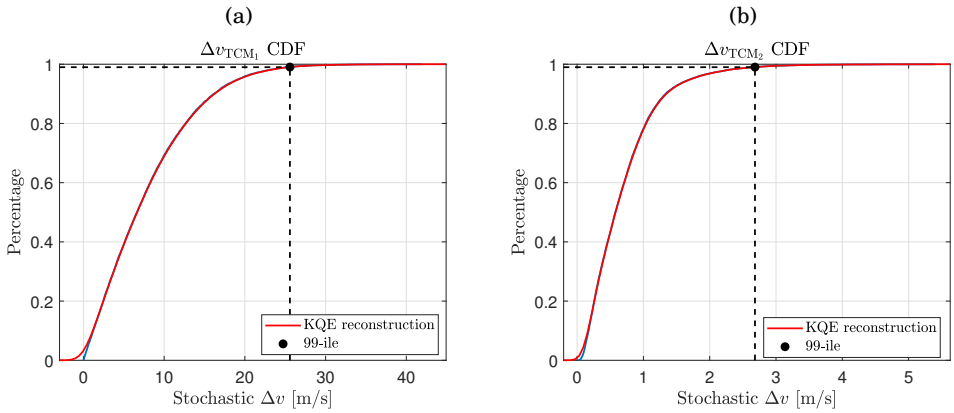



FIGURE 5.19. TCMs cumulative distribution function in Solution #53.

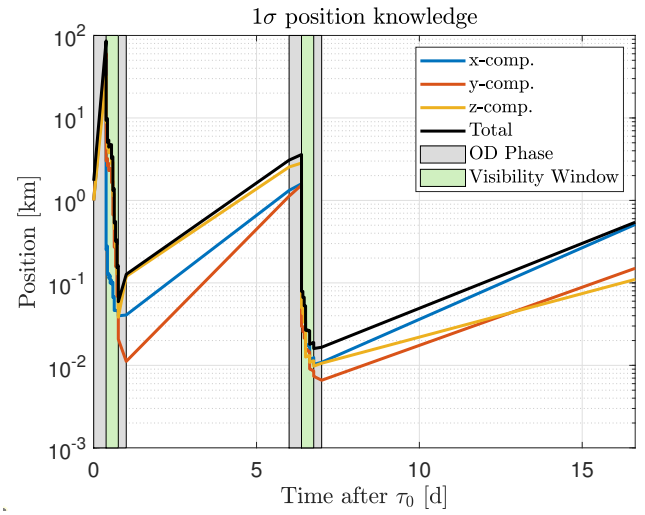



FIGURE 5.20. Position Knowledge Analysis for Solution #53.

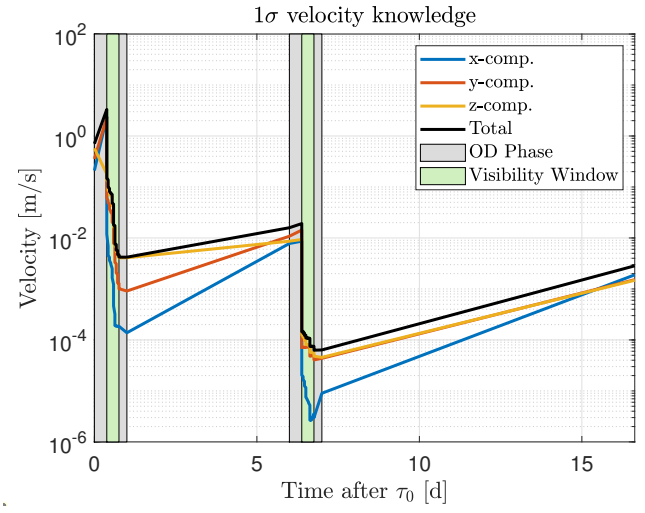



FIGURE 5.21. Position Knowledge Analysis for Solution #53 of LUMIO.

CHAPTER 6

# SCENARIO 2

Scenario (2) depicts a spacecraft, performing low-thrust transfer, in a quasi-Keplerian motion. Scenario (2) can be used to describe a low-thrust deep space trajectory. An example for this kind of dynamics can be the transfer phase of the CubeSat M-ARGO [81].

## 6.1 Introduction

The Miniaturised Asteroid Remote Geophysical Observer (M-ARGO) is planned to be the first standalone ESA deep-space CubeSat to rendezvous a near-Earth asteroid (NEA) [83]. The M-ARGO concept was developed by ESA's Concurrent Design Facility. In 2019, ESA funded the Phase A study, that was conducted by GomSpace Luxemburg in collaboration with Politecnico di Milano. M-ARGO successfully passed the (MDR) in November 2019 and the PRR in July 2020.


FIGURE 6.1. M-ARGO logo.

The M-ARGO project foresaw a 12U CubeSat, planned to piggyback on the launch of another large spacecraft going towards the Sun–Earth Lagrange point $L_2$. After insertion into a parking orbit at $L_2$, M-ARGO will depart from there performing a deep-space cruise towards a NEA target using low-thrust electric propulsion, demonstrating the capability of CubeSats to independently explore deep-space objects. Behind the scientific objectives, M-ARGO will demonstrate in-orbit a number of miniaturized key technologies such as [81]: i) a miniaturized X-band transponder and reflectarray high gain antenna for communication with Earth at distances of up to 1.5 AU; ii) a miniaturized solar drive array mechanism for maximizing solar power generation from two deployable steerable wings; iii) miniaturized electric propulsion for orbital maneuvers. In conclusion, M-ARGO mission objectives can be summarized in 1) demonstrate the capability of CubeSats to independently explore deep-space objects; 2) rendezvous with a near-Earth asteroid and characterize its physical properties; 3) push forward miniaturized European key technologies; 4) test autonomous guidance, navigation, and control techniques during transfer to target object. M-ARGO is departing window from the Sun–Earth $L_2$ point is set to be between January 01, 2023 and December 31, 2024. The maximum transfer time to the selected target shall be less than 3 years. The spacecraft will have an initial wet mass $m_0 = 22.6\,\text{kg}$ with the available propellant amounting to $m_{p,\max} = 2.8\,\text{kg}$. The ballistic coefficient, useful for the computation of the SRP, is $A/m = 0.013\,27\,\text{m}^2/\text{kg}$ with a reflectivity coefficient equal to $c_r = 0.3$. These values are given in Table 6.1.

TABLE 6.1. M-ARGO mission time-frame and spacecraft data.

| **Departure window** | **Transfer** | $m_0$ | $m_{p,\max}$ | $A/m$ | $c_r$ |
|---|---|---|---|---|---|
| 2023 – 2024 | ≤ 3 years | 22.6 kg | 2.8 kg | $0.013\,27\,\text{m}^2/\text{kg}$ | 1.3 |

### 6.1.1 NEO target screening

The number of M-ARGO possible targets among near-Earth objects (NEOs) is huge and it keeps increasing. For this reason, it is required to identify a subset of asteroids that are reachable by M-ARGO CubeSat and can serve as appropriate

targets. The solution of this pruning will be used as first guess for the revised approach and to make a final comparison. Figure 6.2 shows the approach followed to produce the reachable asteroids shortlist, which is the input for the final target selection. This procedure is made of the different steps listed below.

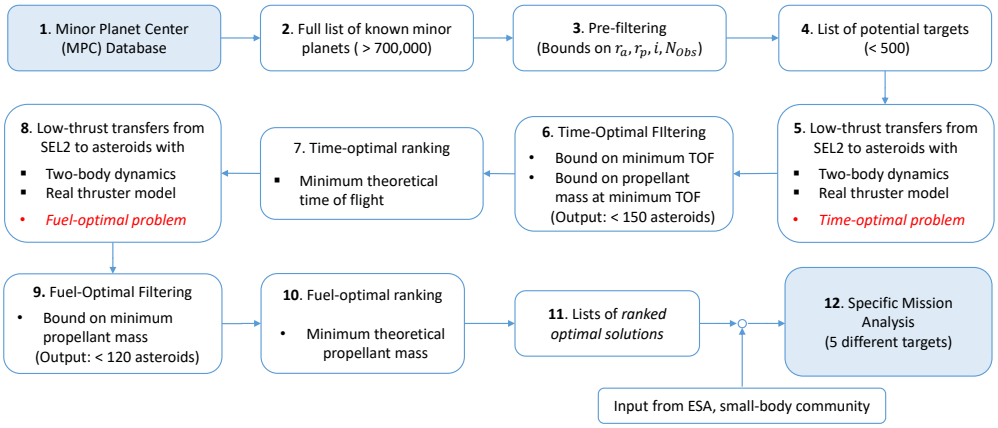



FIGURE 6.2. Methodology of the NEO target screening.

1-2 **Database retrieval.** The Minor Planet Center (MPC) Database[1] is considered as the source of information for the minor planets in the solar system. It comprehends the designation and the orbit computation of all the discovered minor planets and it is updated daily. More than 1,000,000 objects are accounted for as of January 2021. However, the target selection was performed during M-ARGO Phase A and thus in this work only the 700,000 objects are accounted.

3-4 **Pre-Filtering.** The full list of asteroids is pre-filtered using ranges of orbital parameters. Educated guesses on these parameters have been inferred from [60]. These involve capping the aphelion, bottoming the perihelion, and bounding the inclination as well as the number of observations. This filtering reduces the full list of asteroids (~ 700,000) to a preliminary list of approximally 500 potential targets.

[1]See `https://minorplanetcenter.net/` (Last access on December 21, 2020)

5-6 **Time-optimal transfers.** A massive search is conducted to compute time-optimal transfers to each of the asteroids in the preliminary list. The optimization considers the two-body problem with the realistic thruster model, departure from Sun–Earth $L_2$, and departure window as specified in Table 6.1. The aim of this step is to determine the *minimum theoretical transfer time* to each asteroid and for each departure epoch. The objects whose minimum transfer time is greater than the upper bound in Table 6.1 are filtered-out.

7 **Time-optimal ranking.** The filtered time-optimal solutions are ordered to produce a time-optimal ranking. The number of targets is then reduced to $\sim 170$ objects.

8-9 **Fuel-optimal transfers.** The objects resulting feasible after the time-optimal analysis are processed under the perspective of a fuel-optimal optimization. This analysis finds the minimum propellant mass for each combination of departure epoch and transfer time. The objects whose minimum required propellant mass is greater then the bound in Table 6.1 are filtered-out.

10 **Fuel-optimal ranking.** The fuel-optimal solutions as output of step 9 are ordered to produce a fuel-optimal ranking made of approximately 150 reachable objects.

11 **Lists of ranked optimal solutions.** The ranked lists of fuel-optimal solutions has been examined in collaboration with GomSpace and ESA in order to obtain 5 shortlisted targets.

A detailed analysis of this procedure is given in [77].

#### 6.1.1.1 Minor planets database filtering

Consistently with the preliminary work in [60], the Minor Planet Center database is filtered by applying some bounds to the NEO orbital parameters. Upper and lower bounds are set to the aphelium and perihelium respectively, in order to prune out targets either unlikely reachable by M-ARGO or posing communication

and propusion challenges due to their distance from the Earth and the Sun. Moreover, an upper bound on the inclination has been inserted in order to cope with the CubeSat limited propulsive capabilities. Finally, a lower bound of 10 observations ($N_{\text{obs}}$) is enforced to assure accuracy in the orbital elements of the asteroids. Table 6.2 summarizes the filtering parameters. At the end, 456 asteroids passes the filter scheme.

TABLE 6.2. NEO database filtering parameters.

| **Parameter** | **LB** | **UB** | **Rationale** |
|---|---|---|---|
| $r_a$ | – | 1.25 AU | NEA targets |
| $r_p$ | 0.75 AU | – | NEA targets |
| $i$ | 0 deg | 10 deg | CubeSat capability |
| $N_{\text{obs}}$ | 10 | – | Orbital Elements accuracy |

#### 6.1.1.2 Fuel- and Time-Optimal transfers

The NEOs surviving the filtering process underwent a further analysis in order to determine the transfer feasibility within the bound given in Table 6.1.
First, a time-optimal analysis is performed, in order to compute the minimum time needed to reach each target. Performing a time-optimal search in a two-year departure window for different objects requires solving approximately $3.3 \times 10^5$ optimization problems with a one-day time discretization. An even higher number of fuel-optimal problem should be solved for the time-compliant asteroids, since a two-dimensional grid in departure date and time of flight is required. Therefore, an agile solver is needed to scan the solution space in the allotted time. The indirect solver LT2.0 (Low-Thrust Trajectory Optimiser 2.0) internally developed at Politecnico di Milano [76] has been adapted for this purpose. Details on the methodology can be found in [77]. Results of this analysis can be found in Figures 6.3–6.4. Solutions either exceeding a time of flight of 900 d or employing more than 4 kg of propellant are filtered out. These criteria are used to prune out solutions that unlikely will be able to cope with requirements in Table 6.1. In conclusion, 172 asteroids are able to pass the cut-off threshold and will be used as input for the subsequent fuel-optimal analysis.

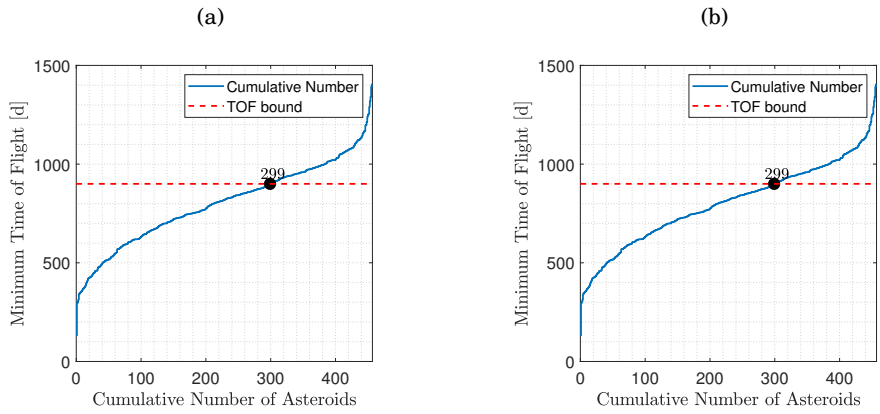



FIGURE 6.3. Cumulative number of asteroids for (a) increasing $\tau_{\min}$ and (b) associated $m_p(\tau_{\min})$. The filtering bounds are the dashed lines, while the number indicates the asteroids below the threshold.

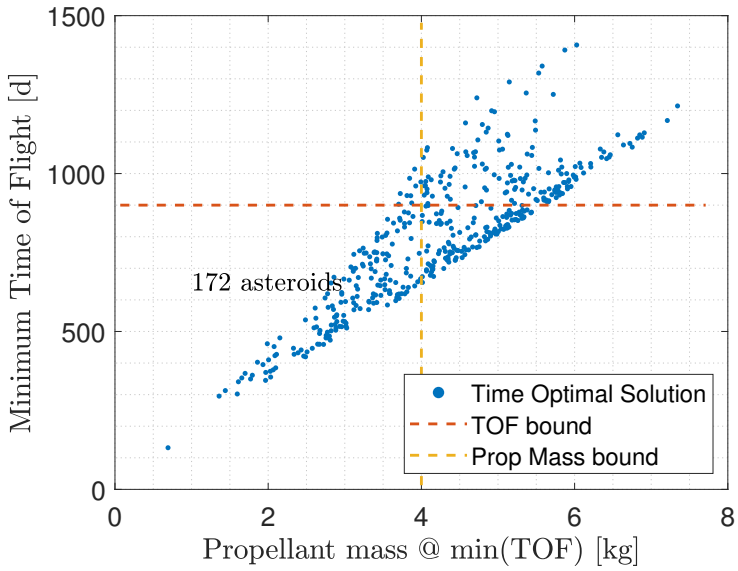



FIGURE 6.4. Time of Flight for the time-optimal solutions against the associated propellant mass. The filtering bounds are the dashed lines, while the number indicates the asteroids below both thresholds.

A fuel-optimal problem is implemented and used to compute the minimum propellant mass needed to reach the target asteroids. It is important to notice that the fuel-optimal process widens the search space since both the departure epoch $\tau_0$ and the time of flight $tof$ are let to vary. Results from this two-dimensional grid are used to construct a porkchop plot for each target. For the 172 asteroid processed $m_{p_{\min}}$ is retrieved, as well as the corresponding value of $t_0$ and $tof$. Eventually, a barrier is set up in order to remove solutions with a $m_{p_{\min}}$ not compliant with the maximum allocated propellant mass (as per Table 6.1). Results for this analysis are illustrated in Figure 6.5. It is shown that only 148 asteroids can be considered feasible targets, having a time of flight lower than 3 years and required propellant mass lower than 2.8 kg.

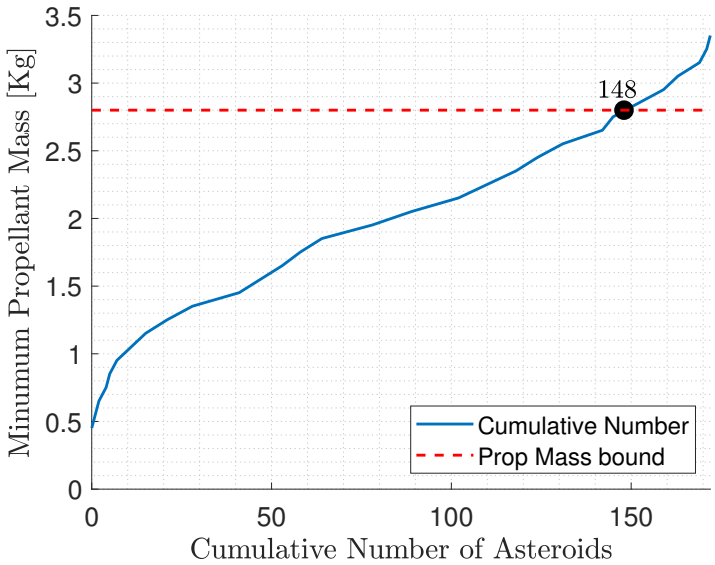



FIGURE 6.5. Cumulative number of asteroids for increasing global minimum propellant mass. The available propellant mass ($m_p = 2.8$ kg) is indicated by the dashed line, while the number shows the amount of asteroids below the threshold.

#### 6.1.1.3 Target shortlisting

The aim of the Phase A target screening performed during the mission analysis was to extract a shortlist of 5 baseline asteroids out of the more of 700,000 objects in the Minor Planet Center database. Once the reachable targets are obtained after the fuel-optimal problem, a pruning is done to reduce the list of potential mission targets. This has been done by analyzing from a statistical point of view the the reachable targets and their transfers and by analyzing the pork chop plots obtained as result of the fuel-optimal problem, in order to remove asteroids requiring a long time of flight, propellant mass close to the maximum value possible or having a limited departure window in the full mission time frame. After this analysis, 41 asteroids are deemed as realistic candidates for the mission future implementation. Results of the NEO target screening, in terms of number of possible targets, is presented in Table 6.3.

Table 6.3: NEO target screening process and results.

| **Step** | **Target screening step** | **No. of objects** |
|---|---|---|
| #1 | Asteroids in the Minor Planet Center database | 700,000+ |
| #2 | Potential targets after pre-filtering | 456 |
| #3 | Possible targets after minimum-time optimisation | 172 |
| #4 | Reachable targets after minimum-fuel optimisation | 148 |
| #5 | Downselected targets after statistical analysis | 41 |
| #6 | Shortlisted targets after close-up analysis | 5 |

Finally, once the list of the 41 desirable, downselected targets has been defined, a deliberate choice was made. The final selection was made by considering

- Light-Curve Database: the presence of the asteroid in the Light-Curve Database is associated to better knowledge of the physical and orbital parameter of the target and thus it is a nice-to-have information;
- Asteroid spin-rate: it is a relevant data from the scientific point of view;
- Future observability: the possibility to observe the target in the future is connected to the reduction of orbital uncertainty and thus a better knowledge of the target state;

- Promising targets: the targets needing low propellant mass for a large time span are preferred.

Following these criteria, five targets are eventually selected [32]:

1. **2014 YD**: Known high spin rate; favorable mission opportunity;
2. **2010 UE51**: #1 on time-optimal and fuel-optimal solution list;
3. **2011 MD**: Present in light curve database, favorable mission opportunity;
4. **2000 SG344**: Observable in the near future, inclined, low uncertainty;
5. **2012 UV136**: Known spin rate, largest target size

Keplerian parameters for the shortlisted asteroids are given in Table 6.4).

Table 6.4: Orbital elements for the selected 5 asteroids (ecliptic J2000).

| Name | a [AU] | e [-] | i [deg] | $\omega$ [deg] | $\Omega$ [deg] |
|---|---|---|---|---|---|
| 2000 SG344 | 0.9775 | 0.0669 | 0.1121 | 275.3026 | 191.9599 |
| 2010 UE51 | 1.0552 | 0.0597 | 0.6239 | 47.2479 | 32.2993 |
| 2011 MD | 1.0562 | 0.0371 | 2.4455 | 5.9818 | 271.5986 |
| 2012 UV136 | 1.0073 | 0.1392 | 2.2134 | 288.6071 | 209.9001 |
| 2014 YD | 1.0721 | 0.0866 | 1.7357 | 34.1161 | 117.6401 |

### 6.1.2 High-fidelity transfers

The NEO target screening was conducted by exploiting a two-body dynamics. This assumption reduces the computational workload, but it can give trajectories representing inaccurately the spacecraft motion. For this reason, a step toward a more complex dynamical model is required in order to better assess trajectories for the five short-listed targets. Thus, solutions in more complex dynamics are sought with the purpose of designing high-fidelity asteroid targeting problems. The detailed mission analysis has the aim to 1) produce refined porkchop plots, 2) select nominal trajectories, and for each of them: i. compute control profiles, ii. evaluate distances from the Earth and Sun, iii. check phase angle plots. In

order to do that, transfers in a full-ephemeris model are designed and optimized. Porkchop-like plots are produced by solving several fuel-optimal problems.

#### 6.1.2.1 Methodology

The $L_2$-Asteroid transfers are made by only few spirals about the Sun. Thus, tackling this problem should not require an excessive amount of parameters to represent the time-variation of state and controls, if a direct problem is exploited. Moreover, the equations of motion right-hand side becomes computational intensive by introducing the gravitational pulling from other Solar System bodies and the Solar radiation pressure perturbation. Therefore, a direct method using collocation is judged to have the best balance between computational time and accuracy. The tool DIREct collocation Tool for Trajectory Optimization (DIRETTO) internally developed at Politecnico di Milano [76] is exploited for this task.

**Dynamics.** The M-ARGO transfer problem can be modeled as a Two-Point Boundary Value Problem (TPBVP), with the initial point placed at Sun-Earth $L_2$ at Departure Date, while the arrival point is the Asteroid state at final time. A full-ephemeris model, considering also the SRP, is used, and spherical coordinates are employed. This choice improves convergences of the solution for this kind of trajectories. Thus, the states are defined as $\mathbf{x} = [r, \theta, \phi, v_r, v_\theta, v_\phi, m]$, with $\phi$ and $\theta$ the Azimuth and Elevation angles in J2000 respectively. The control vector is $\mathbf{u} = [T, \alpha, \beta]$, with $\alpha$ taken from $\hat{\theta}$-vector in the orbit plane, while $\beta$ is the out-of-plane thrust angle. The dynamics can be written as

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t) = \begin{bmatrix} v_r \\ \frac{v_\theta}{r\cos\phi} \\ \frac{v_\phi}{r} \\ P\left(\mathbf{f}_G + \mathbf{f}_{SRP}\right) + S \begin{bmatrix} v_r \\ v_\theta \\ v_\phi \end{bmatrix} + \mathbf{f}_T \\ -\frac{T}{I_{sp} g_0} \end{bmatrix} \tag{6.1}$$

The gravitational force is

$$\mathbf{f}_G = -\frac{\mu}{r^3}\mathbf{r} - \sum_{i\in\mathscr{P}} \mu_i \left( \frac{\mathbf{r}_i}{r_i^3} - \frac{\mathbf{r}_i - \mathbf{r}}{\|\mathbf{r}_i - \mathbf{r}\|^3} \right) \tag{6.2}$$

where $\mathbf{r}$ is the position vector of the spacecraft in J2000 and $r$ is its magnitude, $\mathbf{r}_i$ is the $i$-th planet position vector and $r_i$ its magnitude, and $\mu_i$ is the planetary gravitational constant. The symbol $\mathscr{P}$ indicates a set containing all the planet of the Solar System. Planet states are retrieved by means of SPICE kernels [1, 2].
The Solar radiation pressure is defined as

$$\mathbf{f}_{SRP} = \frac{QA}{m}\frac{\mathbf{r}}{r^3} \tag{6.3}$$

where $Q$ is the solar pressure constant.
The thrust vector is written as

$$\mathbf{f}_T = \frac{T}{m}\begin{bmatrix} \sin\alpha\cos\beta \\ \cos\alpha\cos\beta \\ \sin\beta \end{bmatrix} \tag{6.4}$$

while the matrices $P$ and $S$ are

$$P = \begin{bmatrix} \cos\phi\cos\theta & \cos\phi\sin\theta & \sin\phi \\ -\sin\theta & \cos\theta & 0 \\ -\sin\phi\cos\theta & -\sin\phi\sin\theta & \cos\phi \end{bmatrix}, \qquad S = \begin{bmatrix} 0 & \dot\theta\cos\phi & \dot\phi \\ -\dot\theta\cos\phi & 0 & \dot\theta\sin\phi \\ -\dot\phi & -\dot\theta\sin\phi & 0 \end{bmatrix}$$

**Thruster model.** A realistic thruster model is implemented in the equation of motion in order to simulate accurately the available thrust and instantaneous specific impulse. This feature is important in case of limited-capability low-thrust spacecraft since it helps in precisely define the CubeSat maneuver capability. This model differs from the one in [60], where more simplified assumptions were made on thrust and specific impulse.
The thruster model assumes that both the maximum thrust, $T_{\max}$, and the specific impulse, $I_{sp}$, depend on the instantaneous input power, $P_{in}$, which in turn is a function of the Sun distance, $r$. It means that

$$T_{\max} = T_{\max}(P_{in}), \qquad I_{sp} = I_{sp}(P_{in}), \qquad P_{in} = P_{in}(r) \tag{6.5}$$

In order to correctly represent functions in Eq. (6.5), fourth-order polynomials are employed. They are able to correctly represent the behavior of the M-ARGO electric power system and miniaturized ion thruster, while having smooth, non-singular derivatives. Thus

$$T_{\max}(P_{in}) = a_0 + a_1 P_{in} + a_2 P_{in}^2 + a_3 P_{in}^3 + a_4 P_{in}^4 \tag{6.6}$$

$$I_{sp}(P_{in}) = b_0 + b_1 P_{in} + b_2 P_{in}^2 + b_3 P_{in}^3 + b_4 P_{in}^4 \tag{6.7}$$

$$P_{in}(r) = c_0 + c_1 r + c_2 r^2 + c_3 r^3 + c_4 r^4 \tag{6.8}$$

Moreover, the thruster input power is characterized by upper and lower boundaries, $P_{in,\min}$ and $P_{in,\max}$, respectively, due to technological constraints, related to the managing of thermal and power subsystems. For this reason, a saturation logic for the input power should be considered and it is

$$\begin{aligned}
\text{if } P_{in}(r) > P_{in,\max} \quad &\text{then} \quad P_{in} = P_{in,\max},\ T_{\max} = T_{\max}(P_{in}),\ I_{sp} = I_{sp}(P_{in}) \\
\text{if } P_{in} \in [P_{in,\min}, P_{in,\max}] \quad &\text{then} \quad P_{in} = P_{in}(r),\ T_{\max} = T_{\max}(P_{in}),\ I_{sp} = I_{sp}(P_{in}) \\
\text{if } P_{in}(r) < P_{in,\min} \quad &\text{then} \quad P_{in} = P_{in}(r),\ T_{\max} = 0,\ I_{sp} \text{ not defined}
\end{aligned}$$

Preliminary design values for Eqs. (6.6)–(6.8) have been provided by GomSpace to the consortium in a private communication within the Phase A. The coefficients and the input power limits used in the mission analysis are given in Table 6.5. A graphical representation of the thruster model is reported in Figure 6.6. Parameters at 1 au can be retrieved from those data:

- Input power of 105.4 W;
- Maximum thrust of 1.89 mN and thrust-to-mass ratio of $8.36 \times 10^{-5}$ m/s$^2$;
- Specific impulse of 3022.59 s.

**Collocation method.** In the collocation scheme employed by DIRETTO, the time is discretized evenly in $N$ nodes $t_d = t_0 < t_1 < \cdots < t_N = t_f$ and states are

TABLE 6.5. Thruster model parameters ($r$ in Eq. (6.8) is in AU).

| **Version** | $T_{\max}$ coeff. (mN) | $I_{sp}$ coeff. (s) | $P_{in}$ coeff. (W) |
|---|---|---|---|
| | $a_0 = -0.7253$ | $b_0 = 2652$ | $c_0 = 840.11$ |
| | $a_1 = 0.02481$ | $b_1 = -18.123$ | $c_1 = -1754.3$ |
| v 1.0 (07/07/2019) | $a_1 = 0.02481$ | $b_1 = -18.123$ | $c_1 = -1754.3$ |
| | $a_1 = 0.02481$ | $b_1 = -18.123$ | $c_1 = -1754.3$ |
| $P_{in,\min} = 20\,\mathrm{W}$ | $a_2 = 0$ | $b_2 = 0.3887$ | $c_2 = 1625.01$ |
| $P_{in,\max} = 120\,\mathrm{W}$ | $a_3 = 0$ | $b_3 = -0.00174$ | $c_3 = -739.87$ |
| | $a_4 = 0$ | $b_4 = 0$ | $c_4 = 134.45$ |

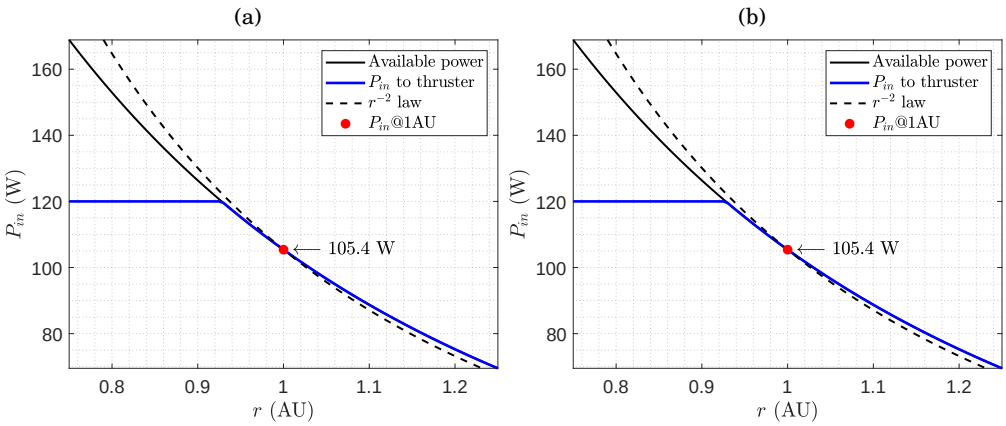



FIGURE 6.6. Graphical representation of the M-ARGO thruster model.

enforced over this grid, with the control approximated linearly. So, for each time $t_k$, the following quantities can be defined

$$\mathbf{x}(t_k) = \mathbf{x}_k, \qquad \mathbf{u}(t_k) = \mathbf{u}_k, \qquad \forall\ k = 0,\ldots,N \tag{6.9}$$

while in each segment

$$t_k \le t \le t_{k+1}, \qquad \mathbf{u}(t) = \mathbf{u}_k + \frac{t - t_k}{t_{k+1} - t_k}(\mathbf{u}_{k+1} - \mathbf{u}_k) \tag{6.10}$$

For each segment a TPBVP is solved by enforcing continuity of the solutions at both ends. The defect constraints $\zeta_k$ are constructed by using an Hermite-Simpson

quadrature scheme

$$\zeta_k = \mathbf{x}_{k+1} - \frac{h}{6}(\mathbf{f}_k + 4\mathbf{f}_c + \mathbf{f}_{k+1}) \tag{6.11}$$

with

$$\mathbf{f}_k = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k, t_k), \qquad \mathbf{f}_{k+1} = \mathbf{f}(\mathbf{x}_{k+1}, \mathbf{u}_{k+1}, t_{k+1}), \qquad \mathbf{f}_c = \mathbf{f}(\mathbf{x}_c, \mathbf{u}_c, t_c) \tag{6.12}$$

and the central points are defined as

$$t_c = t_k + \frac{h}{2}, \qquad \mathbf{x}_c = \frac{1}{2}(\mathbf{x}_k + \mathbf{x}_{k+1}) + \frac{h}{8}(\mathbf{f}_k - \mathbf{f}_{k+1}), \qquad \mathbf{u}_c = \frac{1}{2}(\mathbf{u}_k + \mathbf{u}_{k+1}) \tag{6.13}$$

with the time step being $h = t_{k+1} - t_k$, constant by construction.
The Departure Date is kept fixed, while the final time can vary.
Thus, the fuel-optimal control problem is translated in a NLP Problem, i.e. finding the variable vector

$$\mathbf{y} = \left[\mathbf{x}_1, \ \ldots, \ \mathbf{x}_N, \ \mathbf{u}_1, \ \ldots, \ \mathbf{u}_N, \ t_f\right] \tag{6.14}$$

minimizing the cost function

$$J(\mathbf{y}) = -m(t_f) \tag{6.15}$$

i.e. maximizing the mass at the final time, subjected to the equality constraints:

- Defect vectors must be null to ensure continuity between adjoint segments:

$$\zeta_k = 0, \qquad \forall \ k \in \{1, N\} \tag{6.16}$$

- Boundary conditions at the initial and final time are enforced:

$$\mathbf{r}(t_0) - \mathbf{r}_{L_2}(t_0) = 0, \qquad \mathbf{v}(t_0) - \mathbf{v}_{L_2}(t_0) = 0, \qquad m(t_0) = m_0 \tag{6.17}$$

$$\mathbf{r}(t_f) - \mathbf{r}_{Ast}(t_f) = 0, \qquad \mathbf{v}(t_f) - \mathbf{v}_{Ast}(t_f) = 0 \tag{6.18}$$

  States for both $L_2$ and Asteroid are retrieved using SPICE kernels.

and to the inequality constraints:

- Thrust cannot overcome its maximum value at the instant distance for the Sun:

$$T - T_{\max}(P_{\max}) \leq 0 \tag{6.19}$$

- Transfer time is less than 2.5 years, being consistent with the Table 6.1:

$$t_f - t_0 \leq 900\text{days} \tag{6.20}$$

- Final mass must be positive:

$$-m(t_f) \leq 0 \tag{6.21}$$

A cap on the propellant mass is not imposed, since the aim is to obtain all the possible transfers, even the infeasible one, in order to create a complete porkchop. Eqs. (6.16)–(6.18) and Eqs. (6.19)–(6.21) can be grouped respectively in an equality constraints vector $\mathbf{c}_{eq}$ and an inequality constraints vector $\mathbf{c}$.
The problem of rendezvous with an Asteroid, departing from Sun-Earth $L_2$ exploiting a variable-low-thrust spacecraft can be formally stated as a constrained minimization, i.e.

$$\min_{\mathbf{y}} J(\mathbf{y}) \qquad \text{s.t.} \quad \begin{cases} \mathbf{c}_{eq} = 0 \\ \mathbf{c} \leq 0 \end{cases} \tag{6.22}$$

The minimization is solved in a NLP sense. The software IpOpt [80], in the implementation by OPTI Toolbox[2], is exploited to solve the optimization problem, by means of an internal point method.

#### 6.1.2.2 Results

For the 5 selected targets, the optimization problem in the last paragraph has been solved a number of times and results are used to design porkchop plots. Departure Date is discretized in 15 evenly-spaced points from the interval in Table 6.1, while the guessed Time of Flight goes from 500 d to 900 d with a step of 100 days. Actually, since the final time can vary, so it does the Time of Flight. Results from this analysis are shown in Figure 6.7. They are used as starting point for the revised approach analysis.

[2]See `https://inverseproblem.co.nz/OPTI/index.php/Main/HomePage` (Last retrieved on December 28, 2020.)

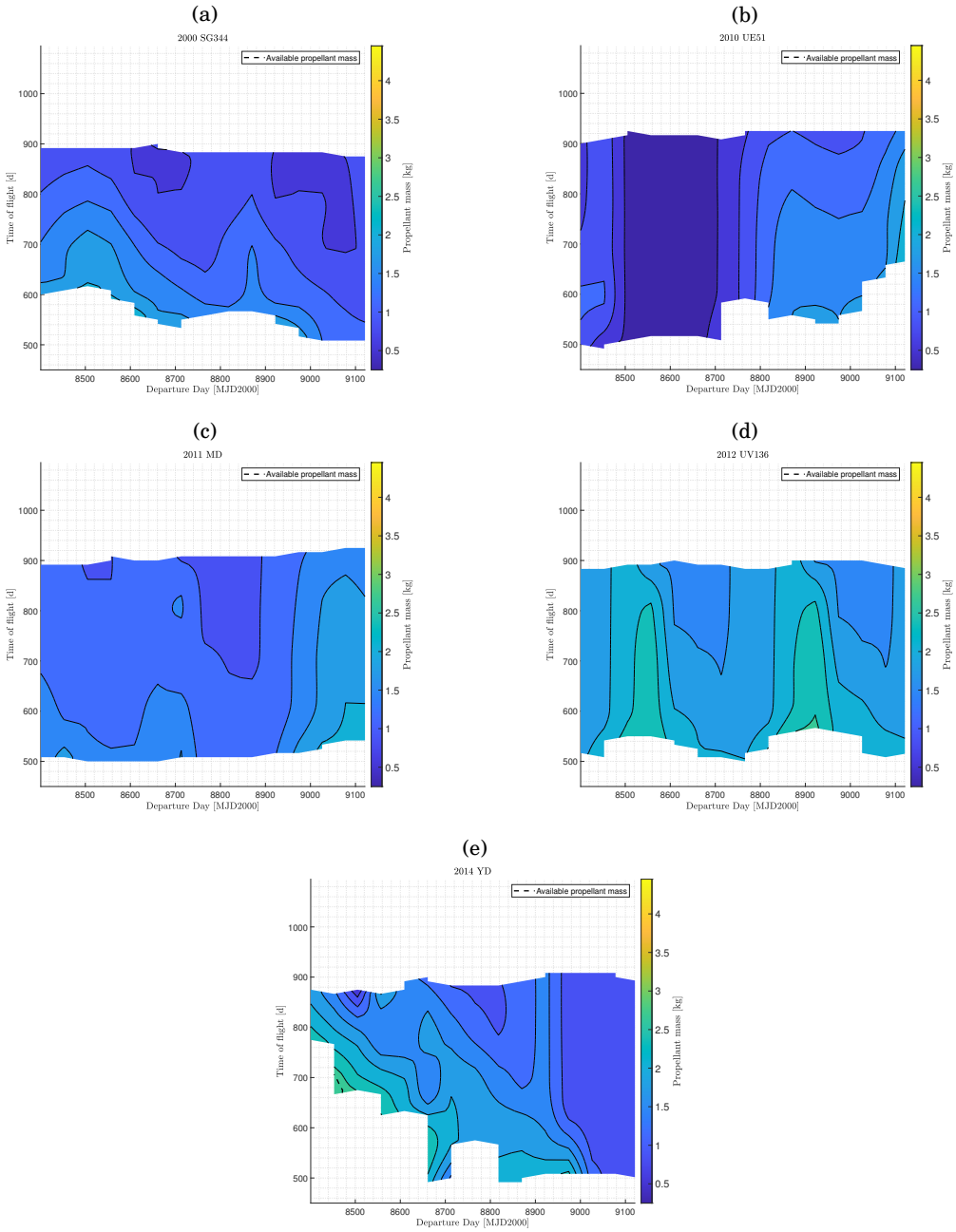



FIGURE 6.7. Porkchop-like plots for shortlisted target in the full-ephemeris model: (a) 2000 SG344, (b) 2010 UE51, (c) 2011 MD, (d) 2012 UV135, (e) 2014 YD. The available propellant mass ($m_p = 2.8$ kg) is indicated with a black dashed line. The color code is the propellant mass used.

## 6.2 Methodology

Some building blocks are stated to apply the revised approach to the Scenario (2). A navigation phase should be foreseen during the M-ARGO transfer trajectory in order to perform orbit determination processes, compute trajectory correction maneuvers to compensate for the state deviations and send proper commands to the spacecraft. Thus, a full day is devoted to flight dynamics tasks for each week during interplanetary cruise. This assumption allows to have regular and repetitive operations. Hence, M-ARGO trajectory from $L_2$ to the target asteroid can be subdivided into two legs, repeating cyclically, as shown in Figure 6.8:

1. *Thrusting phase* (from day 0 to day 6 of each week): the spacecraft surfs its trajectory with no or minimal interaction with the ground;
2. *OD phase* (from day 6 to day 7 of each week): the orbit determination algorithm is performed in order to track the spacecraft and to compute the subsequent correction maneuver.

In a real life scenario, the spacecraft will shut down the thruster during the OD phase, since a large amount of power is required to communicate with the Earth and angles are locked in order to point the antenna toward the ground. However, in this test case, operation compliance is not considered in order to simplify the dynamics and the optimization.

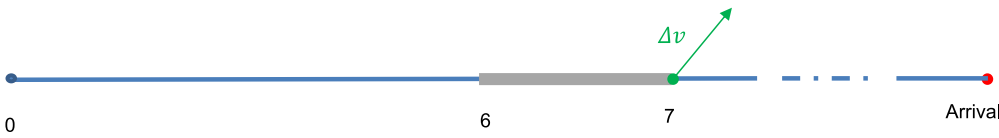



FIGURE 6.8. M-ARGO transfer trajectory timeline. The grey bars represents the OD phases, while the green arrows mark the TCMs points. Time in days after the departure time from $L_2$.

*Uncertainty quantification* follows results from the analysis in Chapter 3. Knowledge and dispersion analyses are performed exploiting two different techniques

in order to cope with the characteristics of the M-ARGO case, having a continuous thrust, that should be modeled as a stochastic process. For this reason:

- *Knowledge analysis* is performed employing local linearization (Section 3.1.2) in a fashion similar to [68];
- *Navigation cost estimation* is achieved by using a Monte Carlo simulation (Section 3.1.1).

### 6.2.1 Uncertainty

To compensate for differences between physical model and real world, a number of uncertainty are taken into account in the revised approach methodology.
The error in the state knowledge at the initial time make the initial state to be modeled as a Gaussian random variable, i.e.

$$\mathbf{x}(t_0) = \mathcal{N}(\mathbf{x}_0, P_0) \tag{6.23}$$

where $P_0 = \text{diag}\left(\left[\sigma_r^2 I_3, \sigma_v^2 I_3\right]\right)$ is the initial covariance matrix for the state. No uncertainty is considered for the spacecraft mass. A standard deviation $\sigma$ of 1000 km in position and 1 m/s in velocity are used for the uncertain initial state. Moreover, dynamics and control error are taken into account in the revised approach in order to accurate model a realistic environment, leading to a higher fidelity evaluation of the designed trajectories and a more conservative state estimation process. Dynamics and control error are modeled as Gauss–Markov process, also know as exponentially correlated random variables (ECRVs). A Gauss–Markov process obeys Langevin differential equation [68], i.e.

$$\dot{\xi}(t) = -\beta\xi(t) + \omega(t) \tag{6.24}$$

where $\xi$ is the Gauss–Markov process, $\omega$ is a white noise with a given variance, and $\beta = 1/\tau$ is the inverse of the correlation time. The Gauss–Markov processes and their characteristics, accounted in Scenario (2), are listed in Table 6.6. Gauss–Markov processes are injected in the covariance dynamics as in [68]. It means

$$P_k = \tilde{\Phi}(t_{k-1}, t_k) P_{k-1} \tilde{\Phi}^T(t_{k-1}, t_k) + \int_{t_{k-1}}^{t_k} \tilde{\Phi}(t_{k-1}, \tau) B(\tau) Q(\tau) B^T(\tau) \tilde{\Phi}^T(t_{k-1}, \tau)\, \mathrm{d}\tau \tag{6.25}$$

with $Q(\tau) = \sigma^2$ and the other quantities conveniently defined in Section 6.2.3. The integral is computed using the trapezoidal rule.

TABLE 6.6. M-ARGO Gauss–Markov processes.

| **Uncertain process** | $\sigma$ | $\tau$ |
|---|---|---|
| Solar radiation pressure | 10% in magnitude | 1 d |
| Thrust | 1% in magnitude | 1 d |
| | 1° in pointing angles | 1 d |
| Residual acceleration | $1 \times 10^{-11}$ km/s$^2$ | 1 d |

### 6.2.2 Orbit Determination

The spacecraft state knowledge during the interplanetary transfer is improved by an orbit determination algorithm, exploiting radiometric measures coming from on-ground facilities. Radiometric tracking is modeled as per Section 5.2.3. Pseudo-measurements for range and range-rate are simulated using Eq. (5.36) whenever a visibility window exists within the timespan dedicated to OD. Pseudo-measurements are then used to feed an extended Kalman filter (EKF) [68] in order to estimate the spacecraft state.
The ground station available for M-ARGO is the 35-meter DSA-1 antenna by ESTRACK, located in New Norcia, Australia. Its performances are in Table 6.7.

### 6.2.3 Dispersion and navigation costs

The stochastic $\Delta v$ is estimated by exploiting a linearized trajectory and the nominal STM. The deviation of the real state from the nominal one is

$$\delta\mathbf{x}(t) = \mathbf{x}(t) - \mathbf{x}^*(t) \tag{6.26}$$

while the *process noise* vector, i.e. the vector of the Gauss–Markov processes indicated in Table 6.6, is defined as $\boldsymbol{\omega}(t) \in \mathbb{R}^{3d}$, where $d$ is the number of stochastic processes. The extended state, including both the state and the process noise is

$$\boldsymbol{\chi}(t) = \begin{bmatrix} \delta\mathbf{x}(t) \\ \boldsymbol{\omega}(t) \end{bmatrix} \tag{6.27}$$

TABLE 6.7. Assumed characteristics for New Norcia ground station.

| **Parameter** | **Value** |
|---|---|
| Coordinates | 31.048 225° S |
| | 116.1915° E |
| Altitude | 0.2523 km |
| Range measurements frequency | twice per pass |
| Doppler measurements frequency | once every 10 min |
| Range measurements random error ($1\sigma$) | 200 m |
| Range measurements systematic error | 200 m |
| Doppler measurements random error ($1\sigma$) | 0.3 mm/s |
| Doppler measurements systematic error | 0 mm/s |
| Station coordinates in equator plane ($1\sigma$) | 30 cm |
| Station coordinates out of equator plane ($1\sigma$) | 1 m |
| Minimum ground station elevation $El_{\min}$ | 15° |

An associated state transition matrix can be stated as

$$\tilde{\Phi}(t,t_0)=\frac{\partial \boldsymbol{\chi}(t)}{\partial \boldsymbol{\chi}(t_0)}=\begin{bmatrix}\Phi & M\\ 0_{3d\times n} & W\end{bmatrix} \tag{6.28}$$

where $n$ is the number of the states. Matrices $M$ and $W$ represent the derivative of the process noise with respect to the state deviation and itself, respectively. Recalling that process noises satisfy Eq. (6.24), for a given process [68]

$$M_i=\frac{\partial \mathbf{x}(t)}{\partial \boldsymbol{\omega}_i(t_0)}=\begin{bmatrix}\left[\frac{1}{\beta}\delta t+\frac{1}{\beta^2}\left(e^{-\beta\delta t}-1\right)\right]I_3\\ \frac{1}{\beta}\left(1-e^{-\beta\delta t}\right)I_3\\ \mathbf{0}_3\end{bmatrix} \tag{6.29}$$

where $M_i$ represents the block of matrix $M$ associated to the $i$-th 3-dimensional process noise $\boldsymbol{\omega}_i$ and $\delta t=t-t_0$, while

$$W_i=\frac{\partial \boldsymbol{\omega}_i(t)}{\partial \boldsymbol{\omega}_i(t_0)}=\left[e^{-\beta\delta t}I_3\right] \tag{6.30}$$

The extended state can be propagated forward by using the extended STM, thus

$$\boldsymbol{\chi}_k=\tilde{\Phi}(t_{k-1},t_k)\boldsymbol{\chi}_{k-1}+\Gamma(t_{k-1},t_k)\boldsymbol{\omega}_{k-1} \tag{6.31}$$

where the subscript $k-1$ and $k$ refers to two consecutive TCM times, $t_{k-1}$ and $t_k$, and the process noise $\boldsymbol{\omega}_{k-1}$ is modeled as a random sequence, that is a random piecewise-constant function. Moreover,

$$\Gamma(t_{k-1}, t_k) = \int_{t_{k-1}}^{t_k} \tilde{\Phi}(t_{k-1}, \tau) B(\tau) \mathrm{d}\tau \tag{6.32}$$

is the process noise transition matrix and the $B = [0_n, I_{3d}]^T$.
As consequence the correction maneuver can be computed by exploiting the differential guidance (Eq. (4.5))

$$\Delta \mathbf{v}_k^s = -\left(\Phi_{rv}^T \Phi_{rv} + q\Phi_{vv}^T \Phi_{vv}\right)^{-1} \left(\Phi_{rv}^T \Phi_{rr} + q\Phi_{vv}^T \Phi_{vr}\right) \widetilde{\delta \mathbf{r}}_k - \widetilde{\delta \mathbf{v}}_k \tag{6.33}$$

with

$$\widetilde{\delta \mathbf{x}}_k = \delta \mathbf{x}_k + \mathcal{N}\left(\bar{\mathbf{X}}_k, P_k\right) \tag{6.34}$$

with $\mathbf{X}_k$ and $P_k$ output of the EKF filter.
For each Monte Carlo sample, all the $\Delta v_k^s$ are converted in needed propellant mass by the Tsiolkovsky equation

$$m_p^s(t_k) = m_k \left(1 - e^{-\frac{\Delta v^s}{g_0 I_{sp}^c}}\right) \tag{6.35}$$

Even if the spacecraft mass should be reduced by the stochastic correction, this behavior is not considered, since its effects are limited. Moreover, TCMs are performed using the same thruster, but in a cold gas fashion. This feature is a characteristic of the M-ARGO propulsive system. In this case, $I_{sp}^c = 40\,\mathrm{s}$.
Once the stochastic propellant mass is computed for each Monte Carlo sample, the kernel quantile estimation (Section 5.2.2) is used for the quantile evaluation. Indeed, although Monte Carlo is used to compute the navigation costs, and, thus a simple samples counting is possible, the use of KQE gives smooth, continuous derivatives and, for this reason, it is preferred.
The final dispersion can be evaluated by exploiting Monte Carlo final states. Indeed, it is possible to retrieve mean $\bar{\mathbf{X}}$ and covariance $P$ (i.e. a Gaussian distribution) for the error with respect to the nominal state. These value can be used to build an uncertainty ellipsoid [68], useful to infer information on the state

uncertainty at a given time. Since $P$ is a positive-definite real matrix, it is always possible to diagonalize it by applying the spectral theorem. Thus

$$D = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} = U^T P U \tag{6.36}$$

with $\lambda_i$ are the eigenvalues and $U = [\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3]$ is an ortho-normal matrix, containing the eigenvectors as columns. The principal axes of $P$ are

$$\hat{\mathbf{r}}' = \begin{bmatrix} \hat{x}' \\ \hat{y}' \\ \hat{z}' \end{bmatrix} = U^T \hat{\mathbf{r}} \tag{6.37}$$

where $\hat{\mathbf{x}}$ are the J2000 axes. Eventually, the probability ellipsoid is defined as

$$\frac{\hat{x}'^2}{\lambda_1} + \frac{\hat{y}'^2}{\lambda_2} + \frac{\hat{z}'^2}{\lambda_3} = \sigma^2 \tag{6.38}$$

where $\sigma$ is the confidence level desired expressed in standard deviation levels. Thus, the uncertainty ellipsoid will be defined as the ellipsoid having as semi-major axes $\sqrt{\lambda_i}$, oriented as the direction of the eigenspace principal axes associated to the covariance matrix.
At the final time, it is desirable that the trajectory ellipsoid is contained at a certain confidence level inside the asteroid uncertainty ellipsoid, i.e.

$$\mathscr{E}\left(\mathbf{x}\left(t_f\right), t_f\right) \subseteq \mathscr{E}_{Ast}\left(t_f\right) \tag{6.39}$$

This condition can be checked by performing an eigenvalue analysis on the ellipsoid representing matrices [6].
Generally speaking, a 3D ellipsoid can be represented using a matrix formulation

$$ZTCT^T Z^T = 0 \tag{6.40}$$

with $Z = [x, y, z, 1]^T$, where

$$C = \begin{bmatrix} P^{-1} & \mathbf{0}_3 \\ \mathbf{0}_3^T & -1 \end{bmatrix} \tag{6.41}$$

is a matrix containing dimensions and axis direction of the ellipsoid, and

$$T = \begin{bmatrix} I & \mathbf{0}_3 \\ -\mathbf{x}_f & 1 \end{bmatrix} \tag{6.42}$$

is the translation matrix, used to bring the center of the ellipsoid in $\mathbf{x}_f$ (i.e., $\mathbf{x}^*(t_f) + \bar{\mathbf{X}}$ for the spacecraft and $\mathbf{x}_{Ast}(t_f)$ for the target). Defining $A$ the matrix $TCT^T$ for the spacecraft and $B$ the one of the asteroid, it is possible to show that the condition in Eq. (6.39) is mathematically equivalent to [6]

$$\begin{cases} \mathrm{Im}\left(\mathrm{eig}\left(A^{-1}B\right)\right) = 0 \\ \mathrm{eig}\left(A^{-1}B\right) \geq 1 \end{cases} \tag{6.43}$$

Thus, the eigenvalues of matrix $A^{-1}B$ should be real and grater than the unity.

## 6.3 Statement of the problem

Once the methodology has been defined, Problem 0 can be adapted to cope with the Scenario (2) using M-ARGO as test case. Following the collocation method explained in Section 6.1.2, the general optimal control problem can be translated in a NLP. Thus, the optimization problem for the Scenario (2) under the revised approach can be stated as

**Problem 2** (Scenario (2) fuel-optimal problem)**.** Find the variable vector

$$\mathbf{y} = \left[\mathbf{x}_1^*,\ \ldots,\ \mathbf{x}_N^*,\ \mathbf{u}_1,\ \ldots,\ \mathbf{u}_N,\ t_f\right] \tag{6.44}$$

minimizing the cost function

$$J(\mathbf{y}) = -\left(m(t_f) - Q\left(0.99, m_s\right)\right) \tag{6.45}$$

i.e. maximizing the sum of the mass at the final time and a measure of the propellant mass needed for the stochastic correction (Eq. (6.35)) computed through the KQE, subjected to the equality constraints:

- Defect vectors for the nominal state must be null to ensure continuity between adjoint segments:

$$\zeta_k = 0, \qquad \forall\ k \in \{1, N\} \tag{6.46}$$

- Boundary condition at the initial time is enforced:

$$\begin{cases} E\left[\mathbf{x}^*(t_0)\right] = \mathbf{x}_0 = \mathbf{x}_{L_2}(t_0) \\ E\left[(\mathbf{x}^*(t_0) - \mathbf{x}_0)(\mathbf{x}^*(t_0) - \mathbf{x}_0)^T\right] = P_0 \\ m(t_0) = m_0 \end{cases} \tag{6.47}$$

- Stochastic boundary condition on the final state must be satisfied:

$$\begin{cases} \mathrm{Im}\left(\mathrm{eig}\left(A^{-1}B\right)\right) = 0 \\ \mathrm{eig}\left(A^{-1}B\right) \geq 1 \end{cases} \tag{6.48}$$

  with $A$ and $B$ computed as per Section 6.2.3

and to the inequality constraints:

- Thrust cannot overcome its allowable maximum value:

$$T - T_{max}(P_{max}) \leq 0 \tag{6.49}$$

- Transfer time is less than 2.5 years, being consistent with the Table 6.1:

$$t_f - t_0 \leq 900\text{days} \tag{6.50}$$

- Final mass must be positive:

$$-m(t_f) \leq 0 \tag{6.51}$$

States for both $L_2$ and Asteroid are retrieved using SPICE kernels, while target covariance at the final time is retrieved by exploiting a telnet connection with NASA Horizon system [40]. Asteroid covariance matrix is retrieved at some specific times and then interpolated by means of a cubic splines using not-a-knot end conditions, in order to have a $\mathscr{C}^2$-class function.
The procedure used to estimate relevant quantities for the optimization problem are collected in Algorithm 2.

**Algorithm 2:** Integrated approach algorithm for Scenario (2)

**Procedure** INTEGRATED APPROACH
- Define spacecraft and navigation settings, and uncertainty properties;
- **Function** INITIALIZATION
  - Compute the random variables for the Monte Carlo simulation;
- **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
            - Retrieve observation data;
            - Generate the pseudo-measurement;
            - Apply the extended Kalman filter;
            - Get mean and covariance from the filter;
            - Propagate mean and coviariance to $t_{k+1}$;
          - **end**
          - Propagate mean and coviariance up to the final time $t_f^{OD}$;
        - **Result:** $\bar{X}_k$ and $P_k$ at each OD final time
      - **otherwise**
        - Propagate statistics up to the *final sub-phase* time;
    - **end**
  - **end**
- **Result:** Knowledge time evolution
- **Function** NAVIGATION COSTS & FINAL ELLIPSOID
  - **for** $\forall$ *Monte Carlo sample*
    - **for** $i = 1$ **to** $n_P$ ▷Loop through $n_P$ sub-phases
      - **switch** *sub-phase*
        - **case** *Thrusting phase*
          - Propagate the deviation up to the TCM time;
          - Estimate the correction maneuvers; ▷See Eq. (6.33)
          - Compute the needed propellant mass; ▷See Eq. (6.35)
          - Apply the maneuvers to each sample;
        - **case** *OD phase*
          - Propagate the the deviation up to the final OD time;
          - Evaluate the estimated deviaton; ▷See Eq. (6.34)
      - **end**
    - **end**
  - **end**
  - Compute the final state ellipsoid;
- **Result:** Navigation cost estimate; Final state ellipsoid

**Result:** Cost function (Eq. (6.45)); Dispersion statistic (Eq. (6.48))

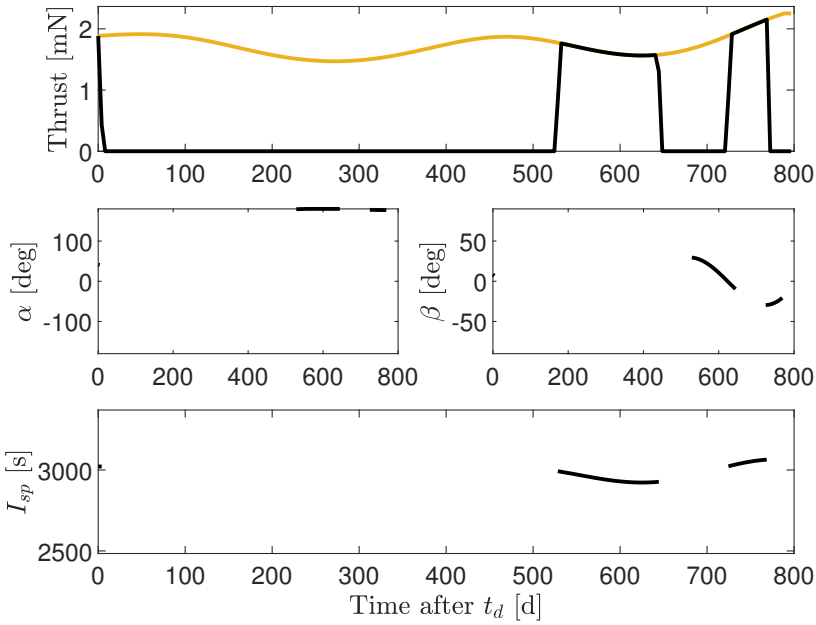
## 6.4 Results

In order to evaluate the revised approach performances in the Scenario (2), a benchmark trajectory should be selected. Asteroid 2000 SG344 is chosen as nominal target due to the repeatability of its porkchop pattern and the needed low propellant. By looking to its porkchop in Figure 6.7(a), a baseline solution, to be used as Problem 2 first guess, is picked, following these criteria:

1. Sensitivity of $m_p$ both in horizontal (Dep. Date) and vertical (ToF) direction;
2. Far from the boundaries in order to surely have feasible solutions;
3. Baseline solution is chosen between 01 Jan 2023 and 31 Dec 2023, in order to have a backup in the year 2024;
4. Among solutions with the same $m_p$, a lower Time of Flight is preferred.

The trajectory selection is listed in Table 6.8.
Optimization solution is reported in Table 6.9 together with results associated to the first guess. Exploiting a very good educated guess, the optimization algorithm requires only 5 minutes on a quad-core Intel i7 2.80 GHz processor. However, a bad initial guess can lead to long computational times and difficult converge.
The optimization under the revised approach leads to savings in propellant mass amounting to 3%. However, if only the stochastic component is considered, the needed propellant mass in the optimized case is only a half with respect to the first guess one. Allegedly, this result is obtained by flying a trajectory with lower dispersion. Figure 6.9 shows the control for the first guess and the optimized solution. In the optimized case, the spacecraft thrusts during the last powered arc in a region with a lower maximum thrust. Since the dispersion depends on the thrust magnitude, a lower thrust level means lower dispersion and, thus, TCMs. Figure 6.10 shows some important geometrical quantities for the revised approach optimal solution, while the knowledge analysis is shown in Figure 6.11. Final state uncertainty ellipsoids are depicted in Figure 6.12. The asteroid position uncertainty, taken from the HORIZON system, is an order of magnitude higher than the $3\sigma$ trajectory ellipsoid, deeming feasible the transfer to a wide extent.

(a)




(b)

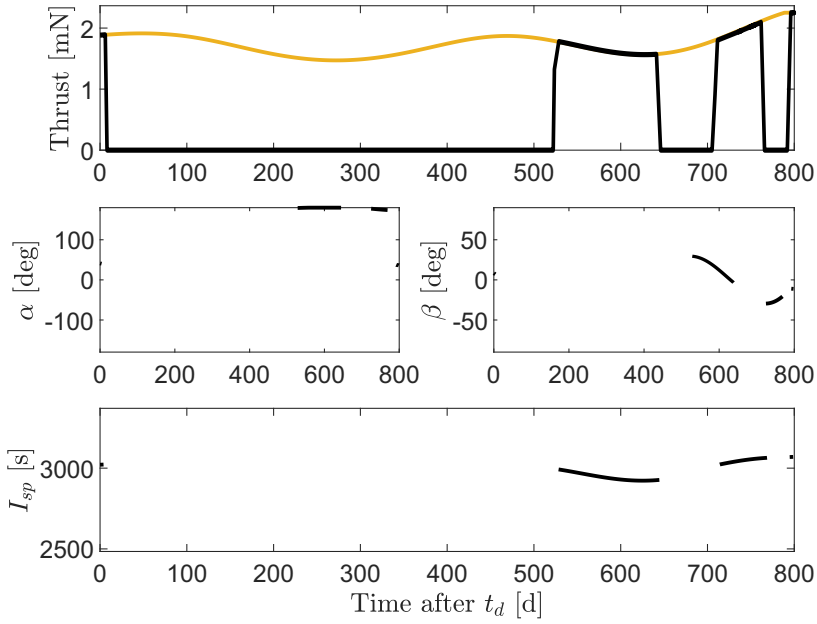



FIGURE 6.9. Control profile for M-ARGO baseline solution to 2000 SG344: (a) First guess; (b) Optimized.

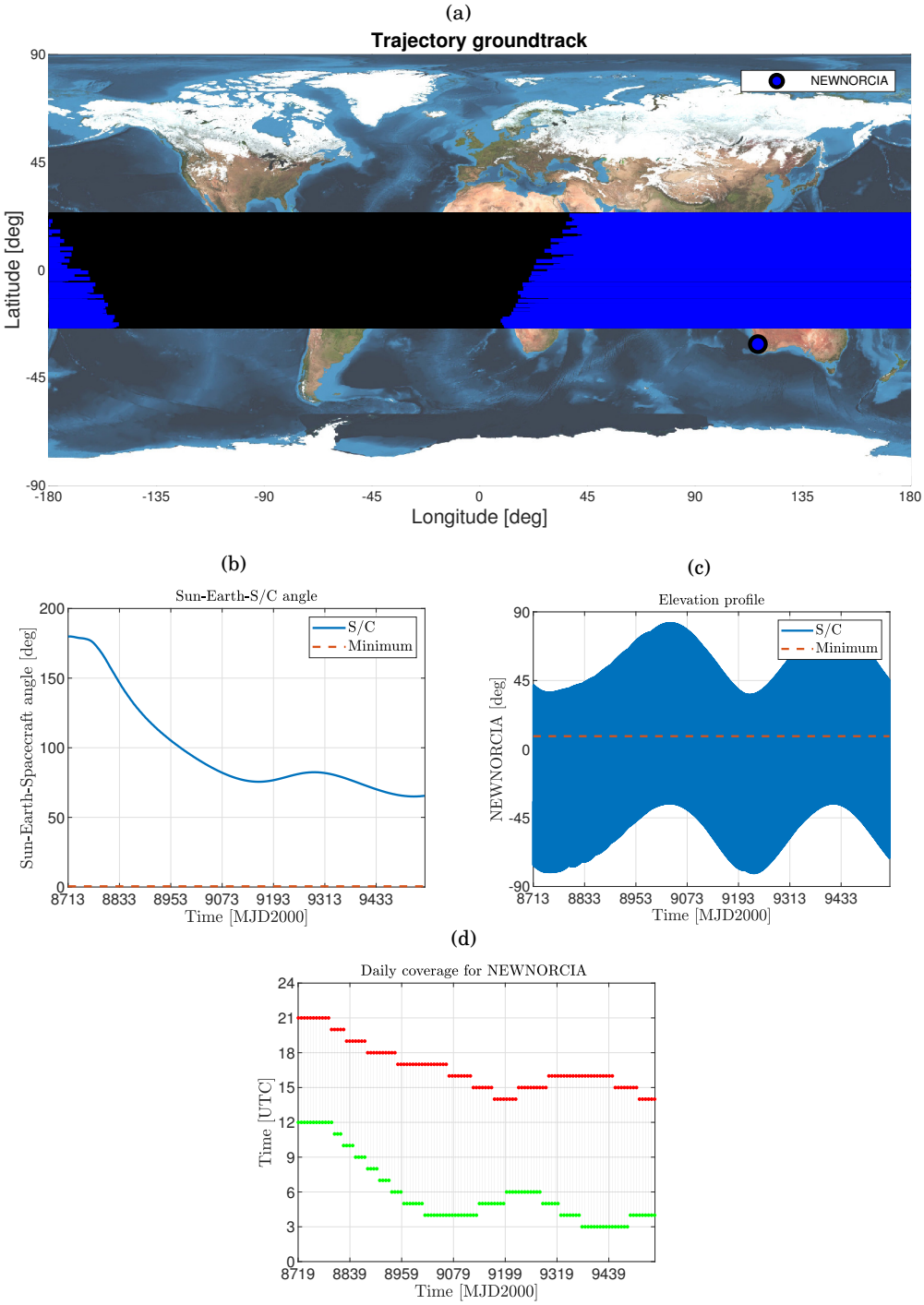
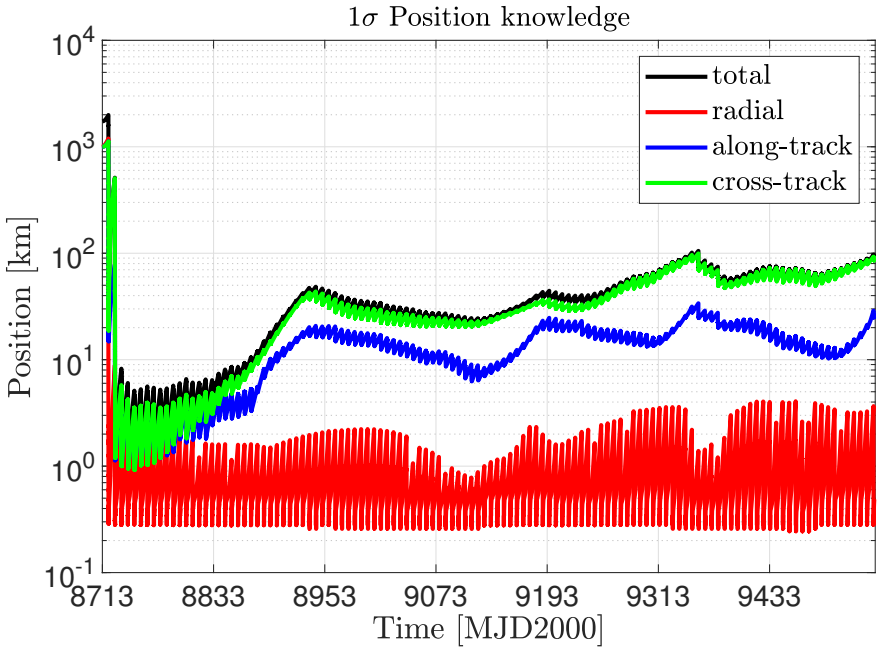



FIGURE 6.10. Geometry assessment for 2000 SG344 Baseline Optimized Trajectory: (a) Trajectory groundtrack (visibility in blue); (b) Exclusion angle; (c) Elevation profile; (d) Daily Coverage from New Norcia.

(a)




(b)

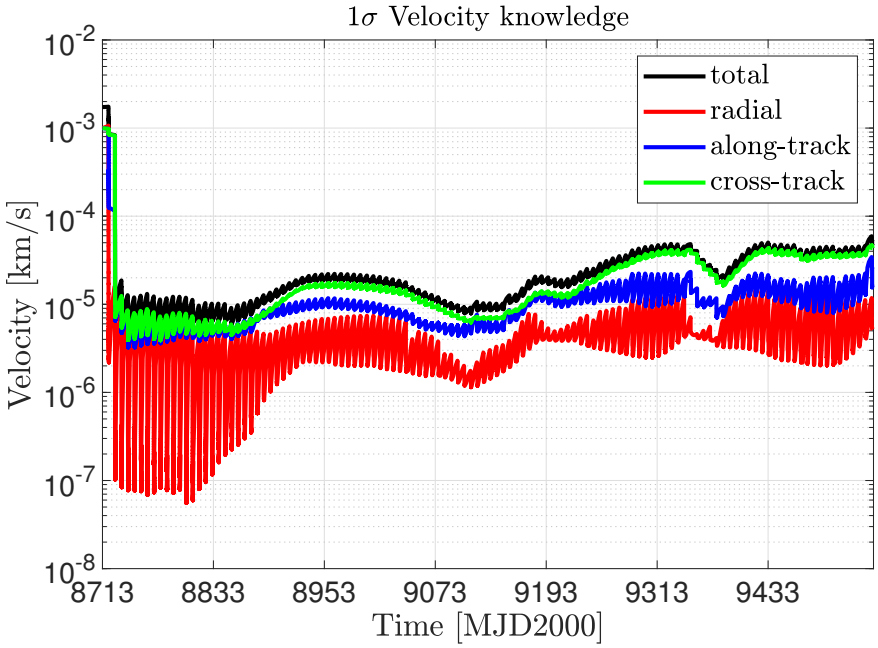



FIGURE 6.11. Knowledge analysis for 2000 SG344 Baseline Optimized Trajectory: (a) Position; (b) Velocity.

TABLE 6.8. Characteristics of baseline solution for M-ARGO.

| **Name** | **Departure Date** | **TOF [d]** | $m_p$ [kg] |
|---|---|---|---|
| 2000 SG344 | 09 NOV 2023 | 796 | 0.857 |

TABLE 6.9. Revised approach solution for M-ARGO case.

| | $ToF$ [d] | $m_p$ (det.) [kg] | $m_p$ (stoc.) [kg] | $m_p$ **(Total)** [kg] |
|---|---|---|---|---|
| **First Guess** | 796 | 0.857 | 0.067 | 0.924 |
| **Optimized** | 798.12 | 0.858 | 0.038 | 0.896 |

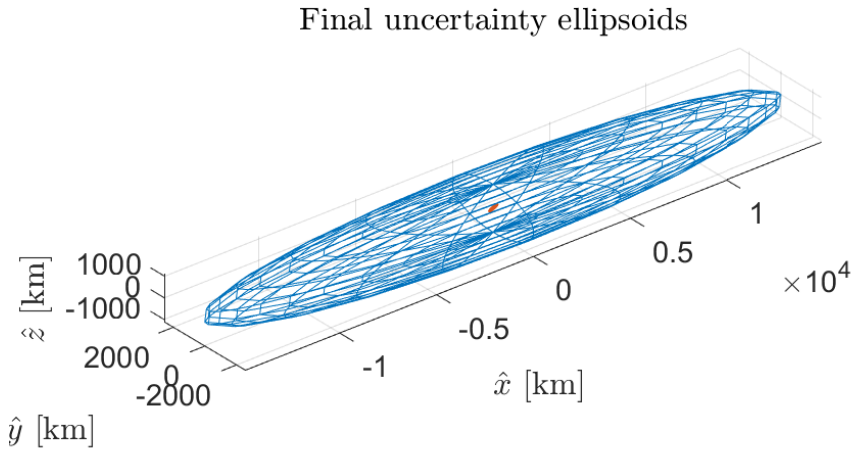



FIGURE 6.12. Final time uncertainty ellipsoids for the optimized M-ARGO case. Asteroid ellipsoid is shown in blue, while the spacecraft $3\sigma$ ellipsoid is depicted in red.

CHAPTER 7

# SCENARIO 3

Scenario (3) displays a spacecraft performing several fly-bys in a planetary system, maneuvering impulsively. This mission profile is representative of ESA mission JUICE [16], planning to perform a tour of the Jovian system. The work presented in this Section was performed during a visiting period in ESOC, Darmstadt, partially funded by the European program Erasmus+[1]. The whole investigation has been conducted by using the Generic Orbit Determination and Optimisation Tool (GODOT)[2], a powerful and flexible software under development by Mission Analysis and Flight Dynamics Sections at ESOC. GODOT is planned to be used for mission analysis at ESA as well as for operations of space probes throughout the solar system.

## 7.1 Introduction

The Jupiter Icy Moon Explorer (JUICE) is an interplanetary mission, developed by ESA, having the aim to study three of the Galilean Jupiter moons, namely Ganymede, Callisto and Europa. The spacecraft is planned to be launched in

[1] `https://ec.europa.eu/programmes/erasmus-plus/` (Retrieved on December 18, 2020)
[2] `https://orbit_software.io.esa.int/godot/` (Retrieved on January 30, 2021)

2022 and it will reach Jupiter in 2029. A tour of the Jupiter moons is foreseen, having the twofold aim to change the spacecraft trajectory by reducing its energy and increasing the inclination by exploiting several lunar fly-bys, and to perform scientific observation during the close approaches. The tour ends in 2032 when the spacecraft is inserted on a elliptic orbit about Ganymede. After some months the spacecraft enters a circular orbit in order to study the moon at higher details. The mission aims to perform a thorough investigation of the Jovian system, with a special focus to Ganymede, in order to characterize the body and evaluate its potential to support life. Surveys on Europa and Callisto give a complete picture of the Galilean moons, providing a comparison among them. Two main objectives can be identified [18], each one can be split into three sub-objectives:

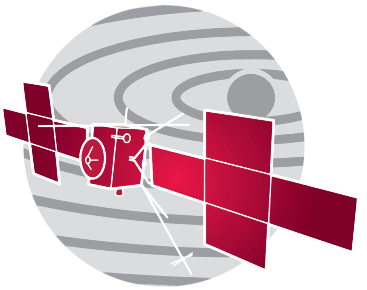


FIGURE 7.1. JUICE insignia.

1. Investigate Jupiter, its complex system and interactions between elements:
    a) Characterize the Jovian atmosphere;
    b) Characterize the Jovian magnetosphere;
    c) Characterize the Jovian environment, its satellites and ring system.
2. Study the three Jupiter icy moons, Ganymede, Callisto and Europa, to understand the habitability of icy worlds:
    a) Examine in detail Ganymede to assess body properties and to probe its capacity to support life;
    b) Explore Europa and its geological active zones;
    c) Perform sounding of Callisto as a remnant of the early Jovian system.

JUICE mission can be conveniently split into two phases:

1. *Interplanetary Transfer Phase (ITP)*: in this phase, the spacecraft perform a deep-space cruise from the Earth to Jupiter. A number of gravity assists exploiting Earth, Venus and Mars, is exploited in order to increase the

trajectory mechanical energy. A special case of Earth fly-by, tagged as Lunar Earth Gravity Assist (LEGA), can be used to increase the launch mass, having higher navigation risks. However, in the nominal trajectory, this feature is not included. Several options for the ITP are possible to accomplish JUICE mission and some of them are listed in [18]. In this work, only the nominal transfer is considered. In this case, the launch date is set to be June 1, 2022 with an arrival in the Jupiter system on October 07, 2029, after an EVEME-GA sequence. A summary of the nominal interplanetary phase can be found in Table 7.1 and is depicted in Figure 7.2.

2. *Nominal Science Phase (NSP)* starts when the spacecraft enters the Jovian system. In order to be captured by the giant planet, a Jupiter orbit insertion (JOI) is performed at the incoming hyperbola pericenter. A first gravity assist on Ganymede (1G1) is foreseen before the JOI with the aim to reduce its magnitude. At the first apojove a Pericenter Raising Maneuver (PRM) is applied in order to target the arrival velocity at the second Ganymede gravity assist and to stabilize the perijove and compensate for the Sun perturbations. Then, the first sub-phase of the moon tour, used for *energy reduction*, is started. In this subleg, four Ganymede flybys are used to reduce the dimension of the Joviocentric orbit and to bring the spacecraft on the equatorial plane. Then the *Europa Science* sub-phase begins. The aim is to perform two fly-bys of Europa in order to accomplish scientific observations of the surface features. Thus, specific latitudes are targeted during Europa gravity assists. Then, the *inclination is raised* by exploiting a series of Callisto fly-bys. This leg brings JUICE on a orbit having an inclination of 29 deg with respect to Jupiter equator, having the possibility to investigate Jupiter magnetosphere. Then the spacecraft is led back to the equatorial plane, when the *transfer to Ganymede* is started. By performing the so called standard Callisto–Ganymede ladder and a low energy endgame, the spacecraft changes its orbit parameter in order to reduce the needed propellant to close the orbit about Ganymede. At the end of this phase, JUICE is weakly captured by the moon and a Ganymede Orbit Insertion (GOI) is performed and the spacecraft is placed on a $200 \times 10\,000$ km elliptic

orbit about Ganymede, followed by a 500 km circular orbit to perform global mapping. The science phase ends on June 26, 2033, three years and a half after entering the Jupiter system, exploiting 29 fly-bys and requiring a $\Delta v = 1606$ m/s. A detailed summary can be found in Table 7.2.

An Extended Science Phase (ESP) is foreseen after the NSP. Additional 100 days about Ganymede are planned ending with an uncontrolled impact on its surface.

TABLE 7.1. Summary of the nominal Interplanetary Transfer Phase.

| **Event** | **Epoch** | $\Delta v$ [m/s] |
|---|---|---|
| Launch | 01 JUN 2022 | ($v_\infty = 3.05$ km/s) |
| DSM1 | 09 SEP 2022 | 44.37 |
| DSM2 | 04 MAR 2023 | 59.32 |
| First Earth swing-by | 31 MAY 2023 | - |
| Venus swing-by | 23 OCT 2023 | - |
| Second Earth swing-by | 02 SEP 2024 | - |
| Mars swing-by | 11 FEB 2025 | - |
| Third Earth swing-by | 26 NOV 2026 | - |
| Arrival | 07 OCT 2029 | ($v_\infty = 5.49$ km/s) |
| **Total** | 7.4 y | 103.69 |

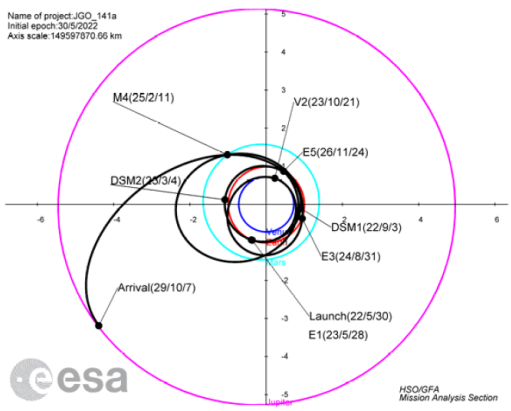



FIGURE 7.2. JUICE nominal interplanetary transfer phase (from [18]).

Table 7.2: Summary of the Nominal Science Phase. In fly-bys labels nXm, n is the total progressive number, X the initial of the Moon name, m the progressive number of moon X. (Adapted from [18])

| Phase | Label | Date | I/O | C/A [km] | $v_\infty$ [km/s] | C/A Long. [deg] | C/A Lat. [deg] | Post Fly-by Period [d] | n:m | $R_j$ | Inc [°] |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Energy red. | 1G1 | 06/10/2029 | I | 400 | 7.7 | 239 | 3 | 213.2 | 30:1 | 11.5 | 4 |
| | 2G2 | 09/05/2030 | I | 400 | 6.5 | 241 | -23 | 57.2 | 8:1 | 12 | 2 |
| | 3G3 | 05/07/2030 | I | 1720 | 6.5 | 249 | -45 | 36.8 | 5:1$^+$ | 11.4 | 0 |
| | 4G4 | 13/08/2030 | O | 791 | 6.5 | 279 | -4 | 21.5 | 3:1 | 10.5 | 0 |
| Europa sc. | 5G5 | 03/09/2030 | O | 1282 | 6.5 | 273 | -3 | 14.8 | / | 9.5 | 1 |
| | 6E1 | 17/09/2030 | I | 403 | 3.7 | 188 | -47 | 14.2 | 4:1 | 9.4 | 2 |
| | 7E2 | 01/10/2030 | I | 403 | 3.7 | 179 | 47 | 14.3 | / | 9.4 | 0 |
| | 8C1 | 13/10/2030 | I | 412 | 5 | 108 | 0 | 19.7 | / | 12.4 | 0 |
| Jupiter High Latitudes | 9G6 | 03/11/2030 | I | 2544 | 4.9 | 269 | -79 | 18 | / | 12.3 | 3 |
| | 10C2 | 13/12/2030 | O | 200 | 4.8 | 222 | 71 | 16.7 | 1:1 | 12.1 | 11 |
| | 11C3 | 30/12/2030 | O | 200 | 4.8 | 182 | 67 | 16.7 | 1:1 | 13.4 | 19 |
| | 12C4 | 15/01/2031 | O | 200 | 4.8 | 182 | 53 | 16.7 | 1:1 | 15.3 | 25 |
| | 13C5 | 01/02/2031 | O | 200 | 4.8 | 216 | 25 | 13.9 | 5:6 | 14.7 | 29 |
| | 14C6 | 25/04/2031 | O | 200 | 4.8 | 35 | -25 | 16.7 | 1:1 | 15.3 | 25 |
| | 15C7 | 12/05/2031 | O | 200 | 4.8 | 360 | -54 | 16.7 | 1:1 | 13.4 | 18 |
| | 16C8 | 29/05/2031 | O | 1020 | 4.8 | 359 | -65 | 16.7 | 1:1 | 12.3 | 13 |
| | 17C9 | 14/06/2031 | O | 1547 | 4.8 | 356 | -75 | 16.7 | 1:1 | 11.7 | 7 |

Table 7.2: Summary of the Nominal Science Phase. In fly-bys labels nXm, n is the total progressive number, X the initial of the Moon name, m the progressive number of moon X. (Adapted from [18])

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | 18C10 | 01/07/2031 | O | 200 | 4.8 | 71 | -49 | 2106 | / | 13.8 | 1 |
| Transfer to Ganymede | 19G7 | 20/07/2031 | I | 10920 | 3.9 | 239 | 9 | 17.9 | 5:1 | 13.6 | 0 |
| | 20G8 | 24/08/2031 | I | 9838 | 3.9 | 244 | 9 | 14.8 | $2{:}1^{+}$ | 13.2 | 0 |
| | 21G9 | 10/09/2031 | O | 3169 | 3.7 | 286 | -8 | 11.1 | / | 12.3 | 0 |
| | 22C11 | 27/08/2031 | I | 200 | 2.2 | 116 | 0 | 17.6 | $3{:}3^{+}$ | 19.7 | 0 |
| | 23C12 | 25/11/2031 | O | 3414 | 2.2 | 256 | -3 | 13.3 | / | 15.4 | 1 |
| | 24G10 | 13/01/2032 | I | 24364 | / | 191 | -9 | 12.5 | 7:4 | 15.3 | 0 |
| | 25G11 | 03/03/2032 | I | 97807 | / | 244 | -2 | 12.0 | 5:3 | 15.2 | 1 |
| | 26G12 | 08/04/2032 | I | 10840 | / | 185 | -24 | 11.4 | 8:5 | 15.0 | 1 |
| | 27G13 | 05/06/2032 | I | 41755 | / | 212 | -1 | 10.8 | 3:2 | 15.6 | 1 |
| | 28G14 | 26/06/2032 | I | 63356 | / | 227 | 4 | 10 | 7:5 | 15.6 | 1 |
| | 29G15 | 16/08/2032 | I | 32234 | / | 187 | 22 | 9.5 | 4:3 | 15.6 | 1 |
| Ganymede sc. | GOI | 17/09/2032 | / | 313 | / | 318 | 12 | / | / | / | / |
| | Man1 | 13/02/2033 | / | 680 | / | 14 | -40 | / | / | / | / |
| | Man2 | 15/02/2033 | / | 570 | / | 277 | -20 | / | / | / | / |
| | End | 26/06/2033 | / | 483 | / | 52 | 35 | / | / | / | / |

### 7.1.1 Europa Phase

In this work, the focus is placed on the Europa scientific phase (highlighted in blue in Table 7.2), with the beginning placed two days before 5G5 and the end two days after Callisto first fly-by. Between the first and the second Europa swing-by, a maneuver, tagged as EU1, is performed in order to prepare the spacecraft for the second Europa encounter. This leg is summarized in Table 7.3.

TABLE 7.3. Europa scientific phase summary. Only relevant quantities for this work are reported.

| **Event** | **Epoch** | **$\Delta v$ [m/s]** | **C/A altitude [km]** | **C/A lat. [deg]** |
|---|---|---|---|---|
| *Start* | 01 SEP 2030 | - | - | - |
| 5G5 | 03 SEP 2030 | - | 1282 | - |
| 6E1 | 17 SEP 2030 | - | 403 | -47 |
| EU1 | 19 SEP 2030 | 19.452 | - | - |
| 7E2 | 01 OCT 2030 | - | 403 | - |
| 8C1 | 13 OCT 2030 | - | 412 | - |
| *End* | 15 OCT 2030 | - | - | - |
| **Total** | 45 d | 19.452 | | |

During the Europa Phase, three sub-phases repeating in loop can be identified, similar to one in the other Scenarios:

1. *OD phase*, in which the spacecraft performs flight dynamics tasks in order to determine its position and schedule the subsequent correction maneuver;
2. *Cut-off phase*, the target guidance is exploited to compute the TCM, which is given at the end of the phase;
3. *Ballistic phase*, a simple uncontrolled flight is followed by the probe.

The orbit determination is performed following a weekly regular scheme. However, in order to minimize the possible trajectory errors related to incorrect fly-bys, an OD session is placed three days before each close approach, and another OD session is scheduled to happen 5 days after each fly-bys in order to correct as soon as possible the state deviations related to imprecise swing-bys. A cut-off

time lasting 2 days is enforced between the OD and the subsequent TCM. All these hypotheses are reported in Table 7.4. From these navigation assumptions, a timeline can be inferred and it is reported in Figure 7.3.

TABLE 7.4. JUICE navigation assumptions during Europa phase.

| Time before C/A | Time after C/A | Min. cut-off time | Duty-cycle |
|---|---|---|---|
| 3 days | 5 days | 2 days | 7 days |

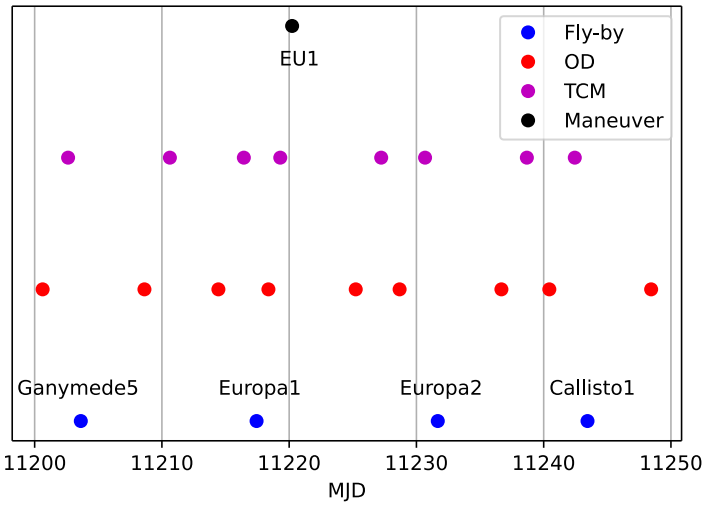



FIGURE 7.3. JUICE Europa phase timeline in the non-optimized case, when nominal navigation assumptions for the times are considered. Dots indicate when a specific event is triggered. The final OD phase is not considered in practice.

#### 7.1.1.1 Dynamics

A high-fidelity model is needed to properly describe the motion of the spacecraft orbiting Jupiter and performing several fly-bys about its moons. A widespread model used to describe the dynamics of a probe executing a series of swing-bys is the *patched conics* approach: the whole trajectory is cut in different legs, each one subjected to the gravitational force of a single body, patched properly. Although

this approach is simple to implement and gives fast results, since it has closed solutions for each leg, trajectories generated by exploiting it are not accurate and reliable for the application in this work. For this reason, a high-fidelity dynamical model comprehensive of all the possible force sources, affecting significantly the spacecraft motion, is used. In Table 7.5, parameters used to build the right-hand side of the equation of motion are presented.

TABLE 7.5. Parameters of the dynamical model used in JUICE case.

| **Body** | **Parameter** | **Value** |
|---|---|---|
| Jupiter | $\mu$ | $126\,686\,534.921\,800\,8\,\mathrm{km}^3/\mathrm{s}^2$ |
| | $J_2$ | 0.014735 |
| Europa | $\mu$ | $3202.738\,774\,922\,892\,\mathrm{km}^3/\mathrm{s}^2$ |
| Ganymede | $\mu$ | $9887.834\,453\,334\,144\,\mathrm{km}^3/\mathrm{s}^2$ |
| Callisto | $\mu$ | $7179.289\,361\,397\,27\,\mathrm{km}^3/\mathrm{s}^2$ |
| Io | $\mu$ | $5959.916\,033\,410\,404\,\mathrm{km}^3/\mathrm{s}^2$ |

#### 7.1.1.2 Uncertainty

In Scenario (3), some uncertainty related both to navigation and command errors are considered.

*Navigation errors* are taken into account as state deviations at the end of 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.

$$\mathbf{x}(t_0) \sim \mathcal{N}(\mathbf{x}_0, P_0) \tag{7.1}$$

where $P_0$ is the 6-dimensional covariance matrix related to the initial time.

In this test case, orbit determination is not simulated by performing pseudo-measurements and exploiting a navigation filter, but a simpler mechanism is preferred, in order to reduce computational effort. For this reason, *mechanization error* is employed as surrogate model for the OD algorithm. Since the spacecraft state relative to the on-ground Earth facilities can be considered almost constant during the short Europa phase time-span, the assumption of an OD error having

a prescribed statistics can be considered valid. However, in order to increase the accuracy of the model, a smaller error is considered in the radial direction from the spacecraft to the Earth, while higher errors are considered in the track and cross-track directions.
Uncertainty associated to deterministic maneuver is put also in the loop in order to increase the model reliability. Mechanization error is taken into account both for modulus and direction. Eq. (5.22) is used to transform the covariance matrix from spherical to Cartesian coordinates.
In conclusion, 57 (i.e. 6 related to the initial state, 3 to EU1, and 48 ($6 \times 8$) to the OD) uncertainties affect Europa phase for JUICE. Table 7.6 contains all the characteristics related to Scenario (3) uncertainties.

TABLE 7.6. Stochastic characteristics of JUICE uncertainty.

| $\sigma_{r_0}$ [km] | $\sigma_{v_0}$ [m/s] | $\sigma_{\Delta v}$ [%] | $\sigma_{\delta}$ [deg] |
|---|---|---|---|
| 1 | 0.01 | 1 | 1 |
| $\sigma_r$ [m] | $\sigma_{t,c}$ [m] | $\sigma_{v_r}$ [mm/s] | $\sigma_{v_{t,c}}$ [mm/s] |
| 6.108 | 66.1 | 0.121 | 0.606 |

## 7.2 Methodology

The basic blocks, needed to apply the revised approach to Scenario (3), are devised. A great attention is posed on how to estimate the uncertainty propagation, and how to compute the stochastic maneuvers and the final dispersion.

### 7.2.1 Uncertainty propagation

Following the trade-off performed in Section 3.2, in the Scenario (3), uncertainties are propagated by exploiting Polynomial Chaos Expansion. However, two different approaches are used following the presence or absence of OD errors. In the first case, the number of uncertainties is limited to 9 and a *quadrature on a sparse grid* is exploited; in the second case, *least-square regression* is implemented (Eq. (3.31)) in order to mitigate the curse of dimensionality.

If navigation is not considered in the loop, the technique labeled as PCE-CUT is exploited, as per Section 5.2.1. However, the several gravity assists undergone by JUICE are expected to heavily spread and twist the final state stochastic distribution. For this reason, a higher-order accurate cubature method is selected. Sixth-order conjugate unscented transformation (CUT6) is considered to have the best balance between computational effort required and accuracy and it is able to properly describe the stochastic state associated to Scenario (3). The even order MCEs, up to the sixth-order, that should be satisfied are [5]

$$\begin{aligned} &E\left[\xi_i^2\right]=1, && E\left[\xi_i^4\right]=3, && E\left[\xi_i^2\xi_j^2\right]=1, \\ &E\left[\xi_i^6\right]=15, && E\left[\xi_i^4\xi_j^2\right]=3, && E\left[\xi_i^2\xi_j^2\xi_k^2\right]=1 \end{aligned} \tag{7.2}$$

The conjugated axes (Table 5.4) are chosen such that the number of points is minimized and the quadrature weights are positive. Table 7.7 summarizes the axes selection. Combining the axes definition with Eq. (7.2), the MCEs read [4]

$$\begin{cases} 2r_1^2\omega_1+2^d r_2^2\omega_2+4(d-1)(d-2)r_3^2\omega_3=1 \\ 2r_1^4\omega_1+2^d r_2^4\omega_2+4(d-1)(d-2)r_3^4\omega_3=3 \\ 2^d r_2^4\omega_2+8(d-2)r_3^4\omega_3=1 \\ 2r_1^6\omega_1+2^d r_2^6\omega_2+4(d-1)(d-2)r_3^6\omega_3=15 \\ 2^d r_2^6\omega_2+8(d-2)r_3^6\omega_3=3 \\ 2^d r_2^6\omega_2+8r_3^6\omega_3=1 \\ 1-2d\omega_1-2^d\omega_2-\frac{4d(d-1)(d-2)}{3}\omega_3=\omega_0 \end{cases} \tag{7.3}$$

System in Eq. (7.3) can be numerically solved and, for a 9-dimensional scenario, has the solutions listed in Table 7.8. Under these assumptions, PCE-CUT6 requires the propagation of 1203 samples, while using the full grid tensor product would require 262144 samples, while sparse quadrature on a Smolyak's grid will used 1177 points. Even if Smolyak's method needs a slightly less initial points to be propagated, the high number of negative weights adversely impacts the numerical stability, giving less accurate results. Thus, a penalty of only 26 nodes is considered to have a negligible effect on the computational times, but it helps in preventing quadrature value to go berserk.

TABLE 7.7. Quadrature points for CUT6.

| Number of points | Position | Weights |
|---|---|---|
| $1 \le i \le 2d$ | $\boldsymbol{\xi}_i = r_1 \boldsymbol{\sigma}_i$ | $\omega_i = \omega_1$ |
| $1 \le i \le 2^d$ | $\boldsymbol{\xi}_{i+2d} = r_2 \mathbf{c}_i^l$ | $\omega_{i+2d} = \omega_2$ |
| $1 \le i \le 4d(d-1)(d-2)/3$ | $\boldsymbol{\xi}_{i+2d+2^d} = r_3 \mathbf{c}_i^3$ | $\omega_{i+2d+2^d} = \omega_3$ |
| 1 | $\boldsymbol{\xi}_0 = \mathbf{0}_d$ | $\omega_0 = \omega_0$ |
| $M = 2d + 2^d + 4d(d-1)(d-2) + 1$ | | |

TABLE 7.8. CUT6 parameters solution for $d = 9$.

| Parameter | Value |
|---|---|
| $r_1$ | 2.343907321979811 |
| $r_2$ | 1.023262223057007 |
| $r_3$ | 2.534286449570529 |
| $\omega_1$ | 0.015076391080550 |
| $\omega_2$ | 0.001134271796355 |
| $\omega_3$ | 0.000157273136994 |
| $\omega_0$ | 0.042190252761366 |

If the mechanization error affecting the estimated state at the OD times is considered, the number of uncertainties needs an amount of samples to be propagated that is unbearable even for a modern powerful computer. For this reason, the use of a least-squares approach to evaluate the PCE coefficients is preferred (Eq. (3.31)). In order to reduce the computational effort to a manageable level, Polynomial Chaos Expansion is limited to polynomial bases up to order 3 (i.e. $p = 3$). Thus, the total number of coefficients (computed through Eq. (3.29)) is $L = 24804$. As suggested by [46], a number of samples $M = L$ is chosen.

### 7.2.2 Dispersion and navigation costs

The estimation for the stochastic $\Delta v$ is based on the target guidance. In fact, in this scenario there is no interest in exactly following the nominal trajectory. However, it is of paramount importance to precisely target the *close approach altitude* in order to have the post-fly-by state as close as possible to the its

nominal value. Moreover, some requirement on the *close approach latitude* should be respect during the first Europa gravity assist. The scientific objective is to observe in detail some surface features, namely the Thera Macula and the Thrace Macula, to evaluate the presence of biosignatures. This objective can be achieved only having the incoming hyperbola pericenter at a specific location with respect to Europa soil. Additionally, the final state is considered as a target, in order to avoid large deviations at the final time. However, this target can be removed if the whole NSP is considered, instead of a partial leg.
Following these consideration, the target functions $f$ and the target trigger functions $g$ can be stated as

$$\text{5G5 C/A altitude:} \quad \begin{cases} f(\mathbf{x}(t_{5G5})) = \|\mathbf{r} - \mathbf{r}_G\| = 1282\,\text{km} \\ g(\mathbf{x}(t_{5G5})) = (\mathbf{r} - \mathbf{r}_G) \cdot (\mathbf{v} - \mathbf{v}_G) = 0 \end{cases} \tag{7.4}$$

$$\text{6E1 C/A altitude:} \quad \begin{cases} f(\mathbf{x}(t_{6E1})) = \|\mathbf{r} - \mathbf{r}_E\| = 403\,\text{km} \\ g(\mathbf{x}(t_{6E1})) = (\mathbf{r} - \mathbf{r}_E) \cdot (\mathbf{v} - \mathbf{v}_E) = 0 \end{cases} \tag{7.5}$$

$$\text{6E1 C/A latitude:} \quad \begin{cases} f(\mathbf{x}(t_{6E1})) = \mathscr{L}(\mathbf{r}) = -47\,\text{deg} \\ g(\mathbf{x}(t_{6E1})) = (\mathbf{r} - \mathbf{r}_E) \cdot (\mathbf{v} - \mathbf{v}_E) = 0 \end{cases} \tag{7.6}$$

$$\text{7E2 C/A altitude:} \quad \begin{cases} f(\mathbf{x}(t_{7E2})) = \|\mathbf{r} - \mathbf{r}_E\| = 403\,\text{km} \\ g(\mathbf{x}(t_{7E2})) = (\mathbf{r} - \mathbf{r}_E) \cdot (\mathbf{v} - \mathbf{v}_E) = 0 \end{cases} \tag{7.7}$$

$$\text{8C1 C/A altitude:} \quad \begin{cases} f(\mathbf{x}(t_{8C1})) = \|\mathbf{r} - \mathbf{r}_C\| = 412\,\text{km} \\ g(\mathbf{x}(t_{8C1})) = (\mathbf{r} - \mathbf{r}_C) \cdot (\mathbf{v} - \mathbf{v}_C) = 0 \end{cases} \tag{7.8}$$

$$\text{State at final time:} \quad \begin{cases} f\left(\mathbf{x}\left(t_f\right)\right) = \mathbf{x} = \mathbf{x}_f \\ g\left(\mathbf{x}\left(t_f\right)\right) = t = t_f \end{cases} \tag{7.9}$$

where the subscripts $G$, $E$, and $C$ refer to Ganymede, Europa, and Callisto, respectively, while $\mathscr{L}$ is a map, embedded in GODOT, computing the projected latitude on Europa given the position of the spacecraft.
Once the functions associated to the desired targets are built, TG matrices $\hat{A}$ and $\hat{A}_0$ can be evaluated. The *automatic differentiation* routines presented in

GODOT are exploited to compute them inexpensively. Using their definition, the correction maneuver can be evaluated at each TCM time as

$$\mathbf{\Delta v}_k^s = -\hat{A}_k^{\dagger}\hat{A}_{0,k}\widetilde{\delta\mathbf{x}}_k \tag{7.10}$$

with the subscript $k$ indicating the quantities computed with respect to the $k$-th TCM time, while

$$\widetilde{\delta\mathbf{x}}_k = \Phi\left(t_k^{OD}, t_k\right)\left(\delta\mathbf{x}_k^{OD} + \mathcal{N}(0, P_k)\right) \tag{7.11}$$

with $\delta\mathbf{x}_k^{OD}$ being the state at the $k$-th OD time, and $P_k$ the mechanization error covariance matrix, constructed by following values in Table 7.6.
It is important to recall that only the first $\Delta v_k^s$ computed in Eq. (7.10) is applied in reality; thus, only the first impulse computed in the TG algorithm is saved to be used for statistical analysis. Procedure described in Section 5.3 is exploited to compute the stochastic costs through KQE. The quantile value is estimated as

$$Q\left(p, \Delta v_k^s\right) = \sum_{q=1}^{n} \frac{1}{nh}\frac{1}{\sqrt{2\pi}}\exp\left[-\frac{1}{2}\left(\frac{1}{h}\left(\frac{i}{n} - p\right)\right)^2\right]\widetilde{\Delta v}_k^{s,i} \tag{7.12}$$

where $\widetilde{\Delta v}_k^{s,i}$ are the sorted value of $\left\|\Delta v_k^{s,i}\right\|$. A number of 100000 $\Delta v_k^{s,i}$ samples are obtained through an inexpensive Monte Carlo simulation associated to PCE. Once the samples $\Delta v_k^{s,q}$ for each TCM are obtained, the PCE coefficients can be retrieved as

$$\mathbf{c}_{\boldsymbol{\alpha}} = \sum_{q=1}^{M} \Delta v_j^{s,q}\psi_{\boldsymbol{\alpha}}^q\omega_q \tag{7.13}$$

in case of perfect knowledge or as

$$\mathbf{c}_{\boldsymbol{\alpha}} = \left(\mathbf{\Psi}^T\mathbf{\Psi}\right)^{-1}\mathbf{\Psi}^T\Delta v_k^{s,q} \tag{7.14}$$

in case of OD-in-the-loop, and then used to obtain the samples

$$\Delta v_{s,i}^k = \sum_{\boldsymbol{\alpha}\in\Lambda_{p,d}} \mathbf{c}_{\boldsymbol{\alpha}}\psi_{\boldsymbol{\alpha}}^i \tag{7.15}$$

where $\psi_{\boldsymbol{\alpha}}^i$ are the basis functions evaluated at a random picked value $\boldsymbol{\xi}_i$.
On the other hand, the generalized Fourier coefficients related to the final state can be retrieved in the same way and used to feed a KDE algorithm (Eq. (5.32)) in order to find final state statistics. Differently from Scenario (1), in this case, bandwidth $h$ is left free to vary.

## 7.3 Statement of the problem

Once the basics are modeled, Problem 0 has to be adapted to cope with Scenario (3) test case, represented by Europa scientific phase of the spacecraft JUICE. The general optimal control problem is converted into a NLP. The optimization problem for Scenario (3) under the revised approach can be stated as

**Problem 3** (Scenario (3) Fuel-Optimal Problem)**.** Find the eight TCM times, $t_{TCM_k}$, the EU1 maneuver time $t_m$, and its nominal impulse $\Delta v_{EU1}$, such that

$$J = \|\Delta v_{EU1}\| + \sum_{k=1}^{8} Q(.95, \Delta v_k^s) \tag{7.16}$$

with $Q(0.95, \Delta v_k^s)$ representing the 95-percentile of the stochastic cost computed through Eq. (5.35), is minimized. The state is subjected to initial constraints

$$\begin{cases} E\left[\mathbf{x}^*(t_0)\right] = \mathbf{x}_0 \\ E\left[(\mathbf{x}^*(t_0) - \mathbf{x}_0)(\mathbf{x}^*(t_0) - \mathbf{x}_0)^T\right] = P_0 \end{cases} \tag{7.17}$$

and a final constraint

$$c = \hat{F}_d(10\,\text{km}) > 0.99 \tag{7.18}$$

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

$$t_{TCM_k} - t_{OD_k} > 2\,\text{d} \qquad \forall k \in \{1,8\} \tag{7.19}$$

The navigation costs are estimated through Eq. (7.15), while the OD is simulated using mechanization error as per Table 7.6.

In summary, the decision vector is $\mathbf{y} = \left[t_{TCM_k}, t_m, \Delta v_{EU1}\right]$ with $k \in \{1,8\}$ are the eight TCM times. Its bounds are listed in Table 7.9.
The process exploited to compute the cost function and constraints in Problem 3 is summarized in Algorithm 3.

TABLE 7.9. Decision vector bounds. Asterisk labels nominal values.

| | $t_{TCM_k}$ | $t_m$ | $\Delta v_{EU1}$ [m/s] |
|---|---|---|---|
| Upper Bound | $t_{OD_{k+1}} - 3\,\mathrm{d}$ | $t_m^* + 3\,\mathrm{d}$ | 20 |
| Lower Bound | $t_{OD_k} + 2\,\mathrm{d}$ | $t_m^* + 3\,\mathrm{d}$ | 5 |

**Algorithm 3:** Integrated approach algorithm for Scenario (3)

**Procedure** INTEGRATED APPROACH
- Define spacecraft;
- Define uncertainty properties and navigation settings; ▷Table 7.6
- **Function** INITIALIZATION
  - Evaluate the initial samples (CUT6 or MC);
  - Compute the random basis function for KDE-KQE;
  - Initialize the mechanization error;
  - Build the target guidance matrices;
- **Function** NAVIGATION COSTS & DISPERSION
  - **for** $i = 1$ **to** $n_P$ ▷Loop through $n_P$ sub-phases
    - **switch** *sub-phase*
      - **case** *OD phase*
        - Propagate the samples up to the final OD time;
        - Draw the mechanization error;
        - Evaluate the estimated samples; ▷See Eq. (7.11)
      - **case** *Cut-off phase*
        - Propagate the samples up to the TCM time;
        - Estimate the correction maneuvers $\mathbf{\Delta v}_k^s$ ; ▷See Eq. (7.15)
        - Apply the maneuver to each sample;
      - **otherwise**
        - Propagate the samples up to the final sub-phase time;
    - **end**
  - **end**
- **Result:** Navigation cost estimate; Target error; Final dispersion

**Result:** Cost function (Eq. (7.16)); Dispersion statistic (Eq. (7.18))

## 7.4 Results

Problem 3 is applied to Europa phase of the JUICE trajectory.

Optimization is performed using SNOPT [39], exploiting the Python interface provided by Pygmo [14] to pass input and obtain outputs. The whole optimization process takes up to 2 hours on a 20 cores workstation, due to the difficult dynamics and the number of samples to be propagated. Moreover, the RAM required to

handle it can easily reach 5 GB.
Results for Scenario (3) are provided in Table 7.10, while the new timeline is depicted in Figure 7.4. The TCM time is increased up to 2.2 days. Furthermore, if OD error is considered, the revised approach optimization allows to reduce the needed $\Delta v$ of about a 5%. This result is mainly due to lower stochastic costs. Navigation cost distribution is represented in Figure 7.5. In the optimized case, a reduction of about the 50% is attained, allowing to save about 1 m/s.
The performances of the target guidance can be retrieved from Table 7.6. The mean error of the first Europa gravity assist altitude is some tens of meters, while on the final position the error is less than 4 km, proving the great precision of the target guidance in this kind of problems. In order to make a comparison, the stochastic costs CDF and 6E1 gravity assist altitude are presented in Figure 7.7. TG performs better in both the cases, requiring 5 times less propellant and giving 4.5 times smaller errors in final position.

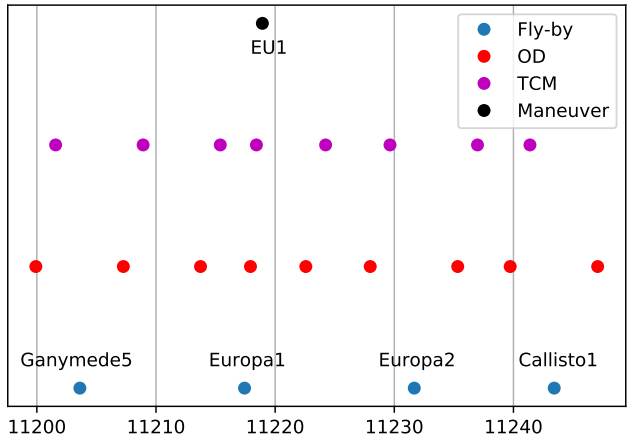



FIGURE 7.4. JUICE Europa phase optimized timeline. The TCM times are moved with respect to Figure 7.3.

TABLE 7.10. Revised approach solution for JUICE case.

| | $\Delta v$ (det.) [m/s] | $\Delta v$ (stoc.) [m/s] | $\Delta v$ **(Total)** [m/s] |
|---|---|---|---|
| **First Guess** | 19.46 | 1.95 | 21.41 |
| **Optimized** | 19.46 | 0.90 | 20.36 |

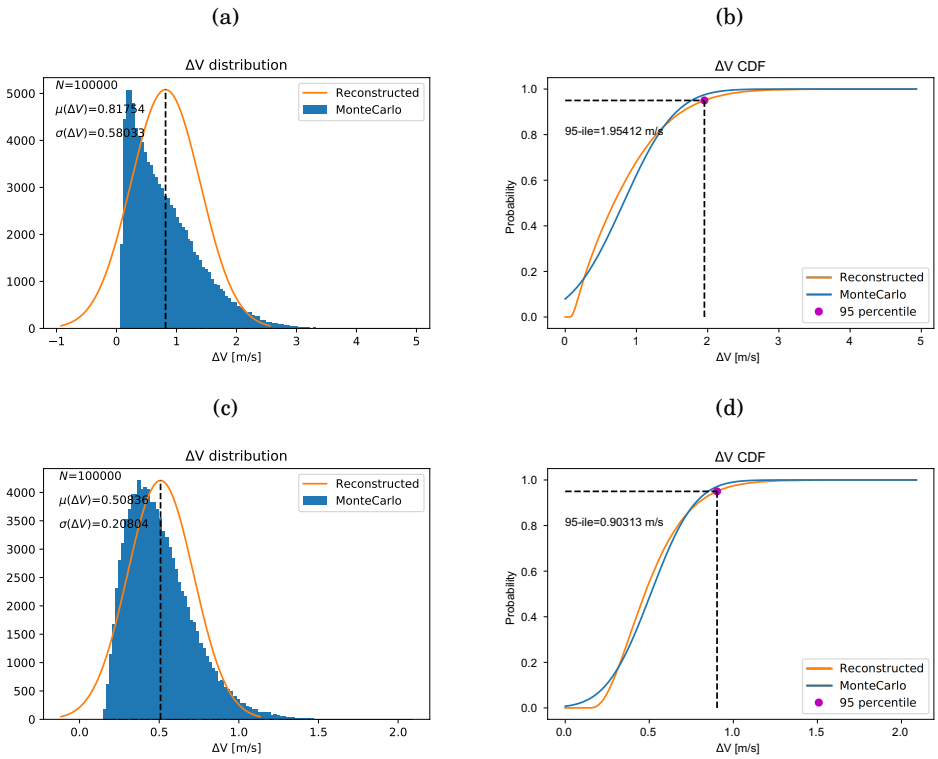



FIGURE 7.5. Navigation cost for the revised approach in the JUICE Europa phase for (a) non-optimal and (b) optimal cases.

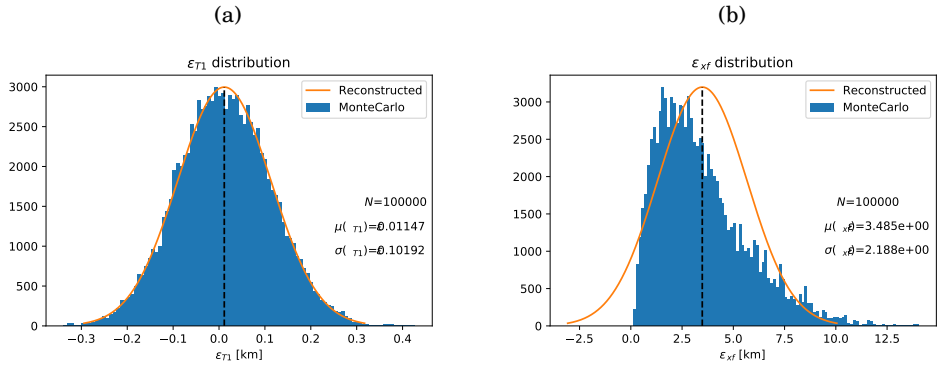



FIGURE 7.6. Targets error in JUICE Europa phase for (a) the first fly-by at Europa altitude and (b) final position.

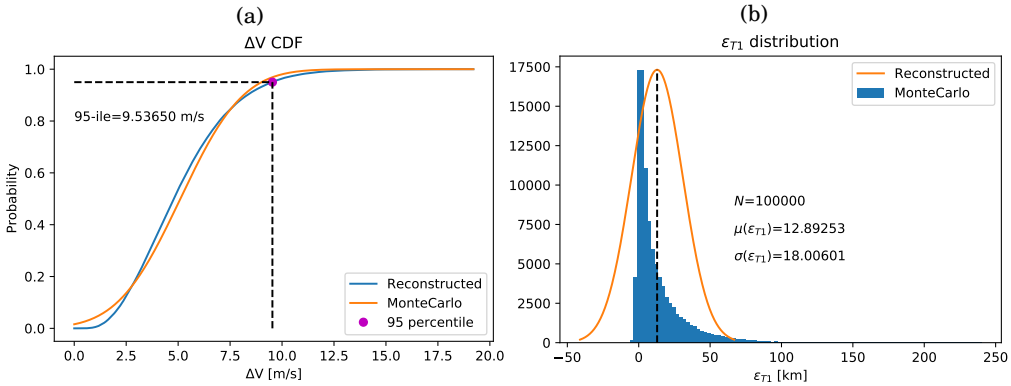



FIGURE 7.7. Differential Guidance performances in JUICE case: (a) stochastic costs CDF and (b) first fly-by at Europa altitude error.

CHAPTER 8

# CONCLUSIONS

In conclusion, a summary of the results attained in this work is presented. Furthermore, possible future developments are addressed and suggestions for the future work are given in the final section.

## 8.1 Summary of results

In this work, an *integrated approach* for preliminary mission analysis is devised and its performances against a traditional sequential approach are discussed. 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.
Then, this method has been applied in three different scenarios, representing trending topics in nowadays space exploration missions. Namely, they are

(1) A spacecraft flying in a strongly nonlinear environment, controlled by impulsive maneuvers, represented by the CubeSat LUMIO (Chapter 5);

(2) A low-thrust satellite, following a spiral trajectory, as per the CubeSat M-ARGO (Chapter 6);

(3) A probe performing several fly-bys in a planetary system, similar to the mission profile of ESA's JUICE (Chapter 7).

For Scenario (1), a new technique, using conjugate unscented transformation to compute the Polynomial Chaos Expansion coefficients, labeled polynomial chaos expansion–conjugated unscented transformation, 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 an 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.
Scenario (2) employs a linearized approach and the navigation costs are evaluated through a Monte Carlo simulation. The continuous thrust is injected as a Gauss–Markov process. The M-ARGO baseline trajectory to asteroid 2000 SG344 is selected as test case and values coming from the Phase A high-fidelity optimization are used as first guess. The optimization under the integrated approach shows that a 3% of propellant mass can be saved.
In the Scenario (3), PCE is used for uncertainty quantification. A novel closed-loop control algorithm, called target guidance, is implemented, having the aim to target some engineering or scientific way-points along the trajectory while minimizing the navigation costs. In this case, the revised approach allows to reduce the total $\Delta v$ of about a 5%, and to halve the navigation cost.

The integrated approach, embedding stochastic features in the trajectory optimization, is able to reduce the overall propellant mass in all the mission scenarios proposed in this work, while satisfying the constraints.

This method will be fundamental in future space mission exploiting limited-capability small spacecraft, where high navigation costs may jeopardize the mission feasibility. Also large missions can benefit from this approach by reducing propellant mass and by shortening the loop needed to design flyable trajectories.

## 8.2 Future work

This thesis is just a starting point for the *revised approach*. However, more steps can be performed in the future to improve some aspects with low fidelity in the current work or that were only partially touched by this thesis. For example, in Scenario (1), Gauss-Markov processes should be added to take into account uncertainty continuous forces, due to Solar radiation pressure and residual non-modeled accelerations. In Scenario (2), the whole porkchop can be obtained and larger consideration should be done on the whole search space. Moreover, in the JUICE case, representing Scenario (3), the orbit determination can be included and the whole scientific phase can be considered to obtain a higher-fidelity result. Finally, the revised approach can be applied in other mission scenarios and it should be tested, in order to better assess its performances, when diverse test cases are taken into account.

APPENDIX A

# NOTABLE DERIVATIVES

In this appendix, a proof for some important derivatives is given. Their result is useful for the optimization problems found in this work. They are presented here for the sake of clarity.

## A.1 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

$$A = S^T S \tag{A.1}$$

Taking an infinitesimal perturbation to this expression, we got

$$\mathrm{d}A = \mathrm{d}S^T S + S^T \mathrm{d}S \tag{A.2}$$

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

$$S^{-T}\mathrm{d}A S^{-1} = S^{-T}\mathrm{d}S^T + \mathrm{d}S S^{-1} \tag{A.3}$$

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. (A.3). Building $\mathcal{T}$ in order to be a function that takes the triangular upper part of a matrix and halves the diagonal, i.e.

$$\mathcal{T}_{ij}(A) = \begin{cases} A_{ij} & j > i \\ \frac{1}{2}A_{ij} & i = j \\ 0 & j < i \end{cases} \tag{A.4}$$

and applying to Eq. (A.3), we got

$$\mathcal{T}\left(S^{-T}\mathrm{d}AS^{-1}\right) = \mathrm{d}SS^{-1} \tag{A.5}$$

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

$$\mathrm{d}S = \mathcal{T}\left(S^{-T}\mathrm{d}AS^{-1}\right)S \tag{A.6}$$

□

## A.2 Matrix square-root derivative

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

$$A = \sqrt{A}\sqrt{A} \tag{A.7}$$

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

$$\mathrm{d}A = \mathrm{d}\sqrt{A}\sqrt{A} + \sqrt{A}\mathrm{d}\sqrt{A} \tag{A.8}$$

Eq. (A.8) is a special case of the Sylvester equation. For the Sylvester equation in the form

$$PX + XQ = R \tag{A.9}$$

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 $\otimes$ [11]

$$\left(I_m \otimes P + Q^T \otimes I_n\right) \mathrm{vec}\, X = \mathrm{vec}\, R \tag{A.10}$$

In Eq. (A.8), $P = Q = \sqrt{A}$ and $C = \mathrm{d}A$, thus its solution is

$$\mathrm{vec}\left(\mathrm{d}\sqrt{A}\right) = \left(\sqrt{A}^T \oplus \sqrt{A}\right)^{-1} \mathrm{vec}(\mathrm{d}A) \tag{A.11}$$

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

# BIBLIOGRAPHY


[1] C. ACTON, N. BACHMAN, B. SEMENOV, AND E. WRIGHT, *A look towards the future in the handling of space science mission geometry*, Planetary and Space Science, 150 (2018), pp. 9–12, DOI: 10.1016/j.pss.2017.02.013.

[2] C. H. ACTON JR, *Ancillary data services of NASA's Navigation and Ancillary Information Facility*, Planetary and Space Science, 44 (1996), pp. 65–70, DOI: 10.1016/0032-0633(95)00107-7.

[3] N. ADURTHI AND M. MAJJI, *Uncertain Lambert Problem: A Probabilistic Approach*, The Journal of the Astronautical Sciences, 67 (2020), pp. 361–386, DOI: 10.1007/s40295-019-00205-z.

[4] N. ADURTHI AND P. SINGLA, *Conjugate Unscented Transformation-based Approach for Accurate Conjunction Analysis*, Journal of Guidance, Control, and Dynamics, 38 (2015), pp. 1642–1658, DOI: 10.2514/1.G001027.

[5] N. ADURTHI, P. SINGLA, AND T. SINGH, *Conjugate Unscented Transformation: Applications to Estimation and Control*, Journal of Dynamic Systems, Measurement, and Control, 140 (2018), DOI: 10.1115/1.4037783.

[6] S. ALFANO AND M. L. GREER, *Determining If Two Solid Ellipsoids Intersect*, Journal of Guidance, Control, and Dynamics, 26 (2003), pp. 106–110, DOI: 10.2514/2.5020.

[7] R. ARMELLIN, P. DI LIZIA, F. BERNELLI-ZAZZERA, AND M. BERZ, *Asteroid close encounters characterization using differential algebra: the case of Apophis*, Celestial Mechanics and Dynamical Astronomy, 107 (2010), pp. 451–470, DOI: 10.1007/s10569-010-9283-5.

[8] R. ARMELLIN, P. DI LIZIA, F. TOPPUTO, M. LAVAGNA, F. BERNELLI-ZAZZERA, AND M. BERZ, *Gravity Assist Space Pruning based on Differential Algebra*, Celestial mechanics and dynamical astronomy, 106 (2010), pp. 1–24, DOI: 10.1007/s10569-009-9235-0.

[9] N. ARORA, V. VITTALDEV, AND R. P. RUSSELL, *Parallel Computation of Trajectories using Graphics Processing Units and Interpolated Gravity Models*, Journal of Guidance, Control, and Dynamics, 38 (2015), pp. 1345–1355, DOI: 10.2514/1.G000571.

[10] M. BALDUCCI AND B. A. JONES, *Asteroid Rendezvous Maneuver Design Considering Uncertainty*, in Advances in the Astronautical Sciences, Univelt, ed., vol. 168, 2019, pp. 2951–2967.

[11] R. H. BARTELS AND G. W. STEWART, *Solution of the matrix equation AX+XB=C [F4]*, Communications of the ACM, 15 (1972), pp. 820–826, DOI: 10.1145/361573.361582.

[12] J. BENKHOFF, J. VAN CASTEREN, H. HAYAKAWA, M. FUJIMOTO, H. LAAKSO, M. NOVARA, P. FERRI, H. R. MIDDLETON, AND R. ZIETHE, *BepiColombo–Comprehensive exploration of Mercury: Mission overview and science goals*, Planetary and Space Science, 58 (2010), pp. 2–20, DOI: 10.1016/j.pss.2009.09.020.

[13] J. T. BETTS, *Survey of Numerical Methods for Trajectory Optimization*, Journal of guidance, control, and dynamics, 21 (1998), pp. 193–207, DOI: 10.2514/2.4231.

[14] F. BISCANI AND D. IZZO, *A parallel global multiobjective framework for optimization: pagmo*, Journal of Open Source Software, 5 (2020), p. 2338, DOI: 10.21105/joss.02338.

[15] Z. I. BOTEV, J. F. GROTOWSKI, D. P. KROESE, ET AL., *Kernel density estimation via diffusion*, The Annals of Statistics, 38 (2010), pp. 2916–2957, DOI: 10.1214/10-AOS799.

[16] A. BOUTONNET AND J. SCHOENMAEKERS, *Mission Analysis for the JUICE Mission*, in Advances in the Astronautical Sciences, vol. 143, 2012, pp. 1561–1578.

[17] A. BOUTONNET AND G. VARGA, *JUICE Equinox Jupiter Tour: the Challenge of Long Eclipses*, in Advances in the Astronautical Sciences, vol. 168, 2019, pp. 3139–3149.

[18] A. BOUTONNET, G. VARGA, A. ROCCHI, W. MARTENS, AND R. MACKENZIE, *JUICE - Jupiter Icy moons Explorer Consolidated Report on Mission Analysis (CReMA)*, 2018.

[19] A. M. CIPRIANO, D. A. DEI TOS, AND F. TOPPUTO, *Orbit design for LUMIO: The lunar meteoroid impacts observer*, Frontiers in Astronomy and Space Sciences, 5 (2018), p. 29, DOI: 10.3389/fspas.2018.00029.

[20] B. A. CONWAY, *A Survey of Methods Available for the Numerical Optimization of Continuous Dynamic Systems*, Journal of Optimization Theory and Applications, 152 (2012), pp. 271–306, DOI: 10.1007/s10957-011-9918-z.

[21] H. D. CURTIS, *Orbital Mechanics for Engineering Students*, Butterworth-Heinemann, 2013.

[22] D. A. DEI TOS, M. RASOTTO, F. RENK, AND F. TOPPUTO, *Lisa Pathfinder mission extension: A feasibility analysis*, Advances in Space Research, 63 (2019), pp. 3863–3883, DOI: 10.1016/j.asr.2019.02.035.

[23] D. A. DEI TOS AND F. TOPPUTO, *Trajectory refinement of three-body orbits in the real solar system model*, Advances in Space Research, 59 (2017), pp. 2117–2132, DOI: 10.1016/j.asr.2017.01.039.

[24] ——, *High-Fidelity Trajectory Optimization with Application to Saddle-Point Transfers*, Journal of Guidance, Control, and Dynamics, 42 (2019), pp. 1343–1352, DOI: 10.2514/1.G003838.

[25] K. J. DEMARS, *Nonlinear orbit uncertainty prediction and rectification for space situational awareness*, PhD thesis, University of Texas, 2010.

[26] P. Di Lizia, R. Armellin, F. Bernelli-Zazzera, and M. Berz, *High order optimal control of space trajectories with uncertain boundary conditions*, Acta Astronautica, 93 (2014), pp. 217–229, DOI: 10.1016/j.actaastro.2013.07.007.

[27] P. Di Lizia, R. Armellin, A. Ercoli-Finzi, and M. Berz, *High-order robust guidance of interplanetary trajectories based on differential algebra*, Journal of Aerospace Engineering, Sciences and Applications, 1 (2008), pp. 43–57.

[28] P. Di Lizia, R. Armellin, A. Morselli, and F. Bernelli-Zazzera, *High order optimal feedback control of space trajectories with bounded control*, Acta Astronautica, 94 (2014), pp. 383–394, DOI: 10.1016/j.actaastro.2013.02.011.

[29] E. Fabacher, S. Kemble, C. Trenkel, and N. Dunbar, *Multiple Sun–Earth saddle point flybys for LISA Pathfinder*, Advances in Space Research, 52 (2013), pp. 105–116, DOI: 10.1016/j.asr.2013.02.005.

[30] W. Fehse, *Automated rendezvous and docking of spacecraft*, vol. 16, Cambridge University Press, 2003.

[31] F. Ferrari, V. Franzese, M. Pugliatti, C. Giordano, and F. Topputo, *Preliminary mission profile of Hera's Milani CubeSat*, Advances in Space Research, 67 (2021), pp. 2010–2029, DOI: 10.1016/j.asr.2020.12.034.

[32] V. Franzese, C. Giordano, Y. Wang, F. Topputo, H. Goldberg, A. Gonzalez, R. Walker, et al., *Target Selection for M-ARGO Interplanetary CubeSat*, in 71st International Astronautical Congress (IAC 2020), 2020, pp. 1–15.

[33] V. Franzese and F. Topputo, *Optimal Beacons Selection for Deep-Space Optical Navigation*, The Journal of the Astronautical Sciences, 67 (2020), pp. 1775–1792, DOI: 10.1007/s40295-020-00242-z.

[34] A. FULLER, *Analysis of nonlinear stochastic systems by means of the Fokker–Planck Equation*, International Journal of Control, 9 (1969), pp. 603–655, DOI: 10.1080/00207176908905786.

[35] R. FURFARO, *Hovering in Asteroid Dynamical Environments Using Higher-Order Sliding Control*, Journal of Guidance, Control, and Dynamics, 38 (2015), pp. 263–279, DOI: 10.2514/1.G000631.

[36] R. FURFARO, A. SCORSOGLIO, R. LINARES, AND M. MASSARI, *Adaptive generalized ZEM-ZEV feedback guidance for planetary landing via a deep reinforcement learning approach*, Acta Astronautica, (2020), DOI: j.actaastro.2020.02.051.

[37] J. P. GARDNER, J. C. MATHER, M. CLAMPIN, R. DOYON, M. A. GREENHOUSE, H. B. HAMMEL, J. B. HUTCHINGS, P. JAKOBSEN, S. J. LILLY, K. S. LONG, ET AL., *The James Webb Space Telescope*, Space Science Reviews, 123 (2006), pp. 485–606, DOI: 10.1007/s11214-006-8315-7.

[38] A. GELB AND R. WARREN, *Direct Statistical Analysis of Nonlinear Systems: CADET*, AIAA Journal, 11 (1973), pp. 689–694, DOI: 10.2514/3.50507.

[39] P. E. GILL, W. MURRAY, AND M. SAUNDERS, *SNOPT: An SQP algorithm for large-scale constrained optimization*, SIAM review, 47 (2005), pp. 99–131, DOI: 10.1137/S0036144504446096.

[40] J. GIORGINI AND D. YEOMANS, *On-line system provides accurate ephemeris and related data*, tech. rep., NASA, 1999.

[41] G. GÓMEZ, M. MARCOTE, AND J. J. MASDEMONT, *Trajectory correction manoeuvres in the transfer to libration point orbits*, Acta Astronautica, 56 (2005), pp. 652–669, DOI: 10.1016/j.actaastro.2004.11.005.

[42] C. GRECO, S. CAMPAGNOLA, AND M. L. VASILE, *Robust Space Trajectory Design using Belief Stochastic Optimal Control*, in AIAA Scitech 2020 Forum, 2020, p. 1471, DOI: 10.2514/6.2020-1471.

[43] Y. GUO, M. HAWKINS, AND B. WIE, *Applications of Generalized Zero-Effort-Miss/Zero-Effort-Velocity Feedback Guidance Algorithm*, Journal of Guidance, Control, and Dynamics, 36 (2013), pp. 810–820, DOI: 10.2514/1.58099.

[44] K. C. HOWELL AND T. M. KEETER, *Station-Keeping Strategies for Libration Point Orbits: Target Point and Floquet Mode Approaches*, in Spaceflight Mechanics Meeting, 1995, pp. 1377–1396.

[45] R. L. IMAN, J. C. HELTON, AND J. E. CAMPBELL, *An Approach to Sensitivity Analysis of Computer Models: Part I–Introduction, Input Variable Selection and Preliminary Variable Assessment*, Journal of Quality Technology, 13 (1981), pp. 174–183, DOI: 10.1080/00224065.1981.11978748.

[46] B. A. JONES, A. DOOSTAN, AND G. H. BORN, *Nonlinear propagation of orbit uncertainty using Non-Intrusive Polynomial Chaos*, Journal of Guidance, Control, and Dynamics, 36 (2013), pp. 430–444, DOI: 10.2514/1.57599.

[47] S. J. JULIER, J. K. UHLMANN, AND H. F. DURRANT-WHYTE, *A new approach for filtering nonlinear systems*, in Proceedings of 1995 American Control Conference-ACC'95, vol. 3, IEEE, 1995, pp. 1628–1632.

[48] J. L. JUNKINS, M. R. AKELLA, AND K. T. ALFRINED, *Non-Gaussian Error Propagation in Orbital Mechanics*, in Advances in the Astronautical Sciences, 1996, pp. 283–298.

[49] R. LAUREIJS, P. GONDOIN, L. DUVET, G. S. CRIADO, J. HOAR, J. AMIAUX, J.-L. AUGUÈRES, R. COLE, M. CROPPER, A. EALET, 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, vol. 8442, International Society for Optics and Photonics, 2012, p. 84420T, DOI: 10.1117/12.926496.

[50] H.-Y. LI, Y.-Z. LUO, G.-J. TANG, ET AL., *Optimal multi-objective linearized impulsive rendezvous under uncertainty*, Acta Astronautica, 66 (2010), pp. 439–445, DOI: 10.1016/j.actaastro.2009.06.019.

[51] J. LI AND D. XIU, *A generalized polynomial chaos based ensemble Kalman filter with high accuracy*, Journal of computational physics, 228 (2009), pp. 5454–5469, DOI: 10.1016/j.jcp.2009.04.029.

[52] J. M. LONGUSKI, J. J. GUZMÁN, AND J. E. PRUSSING, *Optimal Control with Aerospace Applications*, Springer, 2014.

[53] T.-T. LU AND S.-H. SHIOU, *Inverses of 2× 2 block matrices*, Computers & Mathematics with Applications, 43 (2002), pp. 119–129, DOI: 10.1016/S0898-1221(01)00278-4.

[54] D. LUCOR, C.-H. SU, AND G. E. KARNIADAKIS, *Generalized Polynomial Chaos and random oscillators*, International Journal for Numerical Methods in Engineering, 60 (2004), pp. 571–596, DOI: 10.1002/nme.976.

[55] Y.-Z. LUO AND Z. YANG, *A review of uncertainty propagation in orbital mechanics*, Progress in Aerospace Sciences, 89 (2017), pp. 23–39, DOI: 10.1016/j.paerosci.2016.12.002.

[56] M. W. MAKEMSON, *Determination of selenographic positions*, The Moon, 2 (1971), pp. 293–308, DOI: 10.1007/BF00561882.

[57] F. L. MARKLEY AND J. L. CRASSIDIS, *Fundamentals of Spacecraft Attitude Determination and Control*, vol. 33, Springer, 2014.

[58] P. S. MAYBECK, *Stochastic Models, Estimation, and Control*, vol. 3, Academic Press, 1982.

[59] M. MELTZER, *The Cassini-Huygens visit to Saturn: An historic mission to the ringed planet*, Springer, 2015.

[60] A. MERETA AND D. IZZO, *Target selection for a small low-thrust mission to near-earth asteroids*, Astrodynamics, 2 (2018), pp. 249–263, DOI: 10.1007/s42064-018-0024-y.

[61] I. MURRAY, *Differentiation of the Cholesky decomposition*, arXiv preprint, arXiv:1602.07527, (2016), https://arxiv.org/abs/1602.07527 (Last accessed on December 18, 2020).

[62] K. OGURI, K. OSHIMA, S. CAMPAGNOLA, K. KAKIHARA, N. OZAKI, N. BARESI, Y. KAWAKATSU, AND R. FUNASE, *EQUULEUS Trajectory Design*, The Journal of the Astronautical Sciences, 67 (2020), pp. 950–976, DOI: 10.1007/s40295-019-00206-y.

[63] R. S. PARK AND D. J. SCHEERES, *Nonlinear mapping of Gaussian statistics: Theory and applications to spacecraft trajectory design*, Journal of Guidance, Control, and Dynamics, 29 (2006), pp. 1367–1375, DOI: 10.2514/1.20177.

[64] K. B. PETERSEN AND M. S. PEDERSEN, *The Matrix Cookbook*, Technical University of Denmark, 2012, https://www.math.uwaterloo.ca/~hwolkowi/matrixcookbook.pdf (Last accessed on December 18, 2020).

[65] A. POGHOSYAN AND A. GOLKAR, *CubeSat evolution: Analyzing CubeSat capabilities for conducting science missions*, Progress in Aerospace Sciences, 88 (2017), pp. 59–83, DOI: 10.1016/j.paerosci.2016.11.002.

[66] I. M. ROSS, *A Historical Introduction to the Convector Mapping Principle*, in Proceedings of Astrodynamics Specialists Conference, Citeseer, 2005.

[67] P. W. SCHUMACHER JR, C. SABOL, C. C. HIGGINSON, AND K. T. ALFRIEND, *Uncertain Lambert Problem*, Journal of Guidance, Control, and Dynamics, 38 (2015), pp. 1573–1584, DOI: 10.2514/1.G001019.

[68] B. SCHUTZ, B. TAPLEY, AND G. H. BORN, *Statistical orbit determination*, Elsevier, 2004.

[69] S. J. SHEATHER AND J. S. MARRON, *Kernel Quantile Estimators*, Journal of the American Statistical Association, 85 (1990), pp. 410–416, DOI: 10.2307/2289777.

[70] B. W. SILVERMAN, *Density Estimation for Statistics and Data Analysis*, vol. 26, CRC Press, 1986.

[71] S. SPERETTA, A. CERVONE, P. SUNDARAMOORTHY, R. NOOMEN, S. MESTRY, A. CIPRIANO, F. TOPPUTO, J. BIGGS, P. DI LIZIA, M. MASSARI, ET AL., *LUMIO: An Autonomous CubeSat for Lunar Exploration*,

in Space Operations: Inspiring Humankind's Future, Springer, 2019, pp. 103–134, DOI: 10.1007/978-3-030-11536-4_6.

[72] A. H. STROUD, *Approximate Calculation of Multiple Integrals*, Prentice-Hall, 1971.

[73] Y. SUN AND M. KUMAR, *Numerical solution of high dimensional stationary Fokker–Planck Equations via tensor decomposition and Chebyshev spectral differentiation*, Computers & Mathematics with Applications, 67 (2014), pp. 1960–1977, DOI: 10.1016/j.camwa.2014.04.017.

[74] ——, *Uncertainty propagation in orbital mechanics via tensor decomposition*, Celestial Mechanics and Dynamical Astronomy, 124 (2016), pp. 269–294, DOI: 10.1007/s10569-015-9662-z.

[75] G. TEREJANU, P. SINGLA, T. SINGH, AND P. D. SCOTT, *Uncertainty Propagation for Nonlinear Dynamic Systems using Gaussian Mixture Models*, Journal of Guidance, Control, and Dynamics, 31 (2008), pp. 1623–1633, DOI: 10.2514/1.36247.

[76] F. TOPPUTO, D. A. DEI TOS, K. MANI, ET AL., *Trajectory design in high-fidelity models*, in 7th International Conference on Astrodynamics Tools and Techniques (ICATT), Oberpfaffenhofen, Germany, 2018, pp. 1–9.

[77] F. TOPPUTO, Y. WANG, C. GIORDANO, V. FRANZESE, H. GOLDBERG, F. PEREZ-LISSI, AND R. WALKER, *Envelop of reachable asteroids by M-ARGO CubeSat*, Advances in Space Research, (2021).
In Press.

[78] C. TRENKEL, S. KEMBLE, N. BEVIS, AND J. MAGUEIJO, *Testing Modified Newtonian Dynamics with LISA Pathfinder*, Advances in Space Research, 50 (2012), pp. 1570–1580, DOI: 10.1016/j.asr.2012.07.024.

[79] M. VALLI, R. ARMELLIN, P. DI LIZIA, AND M. LAVAGNA, *Nonlinear Mapping of Uncertainties in Celestial Mechanics*, Journal of Guidance, Control, and Dynamics, 36 (2013), pp. 48–63, DOI: 10.2514/1.58068.

[80] A. WÄCHTER AND L. T. BIEGLER, *On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming*, Mathematical programming, 106 (2006), pp. 25–57, DOI: 10.1007/s10107-004-0559-y.

[81] R. WALKER, D. BINNS, C. BRAMANTI, M. CASASCO, P. CONCARI, D. IZZO, D. FEILI, P. FERNANDEZ, J. G. FERNANDEZ, P. HAGER, ET AL., *Deep-space CubeSats: thinking inside the box*, Astronomy & Geophysics, 59 (2018), pp. 24–30, DOI: 10.1093/astrogeo/aty232.

[82] R. WALKER, A. CROPP, P. CONCARI, D. DE WILDE, G. CIFANI, T. SZEWCZYK, W. MARTENS, L. BUCCI, I. CARNELLI, ET AL., *LUMIO CDF study final report*, CDF Report, CDF-R-36 (February 2018).

[83] R. WALKER, D. KOSCHNY, C. BRAMANTI, ET AL., *Miniaturised Asteroid Remote Geophysical Observer (M-ARGO): A stand-alone deep space CubeSat system for low-cost science and exploration missions*, in 6th Interplanetary CubeSat Workshop, Cambridge, UK, 2017, pp. 1–20.

[84] J. S. WHITAKER AND T. M. HAMILL, *Ensemble Data Assimilation without Perturbed Observations*, Monthly Weather Review, 130 (2002), pp. 1913–1924, DOI: 10.1175/1520-0493(2002)130<1913:EDAWPO>2.0.CO;2.

[85] R. J. WHITLEY, D. C. DAVIS, L. M. BURKE, B. P. MCCARTHY, R. J. POWER, M. L. MCGUIRE, AND K. C. HOWELL, *Earth–Moon near rectilinear halo and butterfly orbits for lunar surface exploration*, in AAS/AIAA Astrodynamics Specialists Conference, Snowbird, Utah, 2018.

[86] F. XIONG, Y. XIONG, AND B. XUE, *Trajectory Optimization under Uncertainty based on Polynomial Chaos Expansion*, in AIAA Guidance, Navigation, and Control Conference, 2015, p. 1761, DOI: 10.2514/6.2015-1761.

[87] D. XIU, *Numerical Methods for Stochastic Computations: A Spectral Method Approach*, Princeton University Press, 2010.

[88] D. XIU AND G. E. KARNIADAKIS, *The Wiener–Askey Polynomial Chaos for Stochastic Differential Equations*, SIAM Journal on Scientific Computing, 24 (2002), pp. 619–644, DOI: 10.1137/S1064827501387826.

[89] Z. YANG, Y.-Z. LUO, AND J. ZHANG, *Robust Planning of Nonlinear Rendezvous with Uncertainty*, Journal of Guidance, Control, and Dynamics, 40 (2017), pp. 1954–1967, DOI: 10.2514/1.G002319.

[90] D. G. YÁRNOZ, R. JEHN, AND M. CROON, *Interplanetary navigation along the low-thrust trajectory of BepiColombo*, Acta Astronautica, 59 (2006), pp. 284–293, DOI: 10.1016/j.actaastro.2006.02.028.

[91] G. ZHANG, D. ZHOU, D. MORTARI, AND M. R. AKELLA, *Covariance analysis of Lambert's problem via Lagrange's transfer-time formulation*, Aerospace Science and Technology, 77 (2018), pp. 765–773, DOI: 10.1016/j.ast.2018.03.039.