Optimal State Estimation and Control of Space Systems Under Severe Uncertainty

Cristian Greco

Submitted in fulfilment of the requirements
for the degree of Doctor of Philosophy

Aerospace Centre of Excellence
Mechanical and Aerospace Engineering
University of Strathclyde, Glasgow

September 15, 2021

Abstract

This thesis presents novel methods and algorithms for state estimation and optimal control under generalised models of uncertainty. Tracking, scheduling, conjunction assessment, as well as trajectory design and analysis, are typically carried out either considering the nominal scenario only or under assumptions and approximations of the underlying uncertainty to keep the computation tractable. However, neglecting uncertainty or not quantifying it properly may result in lengthy design iterations, mission failures, inaccurate estimation of the satellite state, and poorly assessed risk metrics. To overcome these challenges, this thesis proposes approaches to incorporate proper uncertainty treatment in state estimation, navigation and tracking, and trajectory design. First, epistemic uncertainty is introduced as a generalised model to describe partial probabilistic models, ignorance, scarce or conflicting information, and, overall, a larger umbrella of uncertainty structures. Then, new formulations for state estimation, optimal control, and scheduling under mixed aleatory and epistemic uncertainties are proposed to generalise and robustify their current deterministic or purely aleatory counterparts. Practical solution approaches are developed to numerically solve such problems efficiently. Specifically, a polynomial reinitialisation approach for efficient uncertainty propagation is developed to mitigate the stochastic dimensionality in multi-segment problems. For state estimation and navigation, two robust filtering approaches are presented: a generalisation of the particle filtering to epistemic uncertainty exploiting samples' precomputations; a sequential filtering approach employing a combination of variational inference and importance sampling. For optimal control under uncertainty, direct shooting-like transcriptions with a tunable high-fidelity polynomial representation of the dynamical flow are developed. Uncertainty quantification, orbit determination, and navigation analysis are incorporated in the main optimisation loop to design trajectories that are simultaneously optimal and robust. The methods developed in this thesis are finally applied to a variety of novel test cases, ranging from LEO to deep-space missions, from trajectory design to space traffic management. The epistemic state estimation is employed in the robust estimation of debris' conjunction analyses and incorporated in a robust Bayesian framework capable of autonomous decision-making. An optimisation-based scheduling method is presented to efficiently allocate resources to heterogeneous ground stations and fusing information coming from different sensors, and it is applied to the optimal tracking of a satellite in highly-perturbed very-low Earth orbit, and a low-resource deep-space spacecraft. The optimal control methods are applied to the robust optimisation of an interplanetary low-thrust trajectory to Apophis, and to the robust redesign of a leg of the Europa Clipper tour with an initial infeasibility on the probability of impact with Jupiter's moon.

Acknowledgements

I am deeply grateful to my supervisor, Prof. Massimiliano Vasile, the architect behind the extraordinary research and life experience my PhD has been. He has always provided great expertise, support, and dedication during my path and has helped to shape my future professional career.

A special thanks goes to Dr Stefano Campagnola, my mentor at NASA JPL, for his enthusiasm for my research and the invaluable human and technical suggestions he provided during our countless conversations. My gratitude also goes to Jon Sims, Eric Gustafson, Sonia Hernandez, Frank Laipert, Sumita Nandi, Brian Young, and all the experts in the Mission Design and Navigation section for sharing their expertise and the insightful discussions.

I would like to acknowledge the support of my funding organisation, the European Commission, through the H2020-MSCA-ITN-2016 UTOPIAE Marie Curie Innovative Training Network, grant agreement 722734. Thanks to this fellowship, I had the opportunity to attend valuable training events, enlarge my research horizon through formative secondments, and collaborate with promising fellow researchers, or better, I should say, friends.

To the current and former colleagues and friends in the aerospace department, who made the days in Glasgow sunny: Annalisa, Carlos, Francesco, Gaetano, Giacomo, Gianluca, Giuseppe, Lorenzo G., Lorenzo R., Marilena, Mateusz, Matteo, Romain, Victor.

To my lifetime friends without whom, as I always say, I would be a better engineer but an emptier person: Andrea, Arcangelo, Cosimo, Davide, Francesco, Gabriele, Giuseppe, Lucas, Stefano.

Finally, the most heartfelt mention goes to my family, the fundamental source of confidence and serenity I need to always face new challenges: my strong and loving parents Mimmo and Maria, my beloved grandparents Gaetano and Anna, my caring and funny sisters Noemi and Arianna, my great brothers-in-law Matteo and Claudio, my little nieces Mia and Sara, and last, but definitely not least, my sweet Grace.

Contents

List of Figures

List of Tables

List of Algorithms

List of Acronyms

AI	Artificial Intelligence	8
APR	Approach	176
AzEl	Azimuth and Elevation	139
B&B	Branch & Bound	45
BfM	Budget for Measurements	143
BMDP	Belief Markov Decision Process	85
BOC	Belief Optimal Control	85
CAM	Collision Avoidance Manoeuvre	3
CDM	Conjunction Data Message	8
CU	Clean-Up	175
DCA	Distance of Closest Approach	103
dOCP	deterministic Optimal Control Problem	76
DSN	Deep Space Network	7
ECDF	Empirical Cumulative Distribution Function	110
ESA	European Space Agency	104
FoV	Field of View	135
FPE	Fokker-Planck Equation	25
GA	Genetic Algorithm	7
GPA	Generalised Polynomial Algebra	28

GS	Ground Station	134
HBR	Hard-Body Radius	110
ICS	Intelligent Classification System	121
LEB	Longest Edge Bisection	62
LEO	Low Earth Orbit	3
MEO	Medium Earth Orbit	5
ML	Machine Learning	121
NA	Navigation Analysis	176
NEO	Near-Earth Object	3
NLP	Nonlinear Programming	78
NoM	Number of Measurements	142
OD	Orbit Determination	11
ODE	Ordinary Differential Equation	27
OI	Orbit Index	143
PDE	Partial Differential Equation	25
pdf	probability density function	22
PF	Particle Filter	5
PoC	Probability of Collision	103
PoI	Probability of Impact	176
pSIS	precomputed Sequential Importance Sampling	46
R	Range	139
RMSE	Root Mean Squared Error	107
RPF	Robust Particle Filter	45
RR	Range-Rate	139
rTCA	reference Time of Closest Approach	102

RTN	Radial Transversal Normal	195
SCGA	Structured-Chromosome Genetic Algorithm	138
SGHQF	Sparse Gauss-Hermite Quadrature Filter	136
SMAA	Semi-MAjor Axis	188
SMIA	Semi-MInor Axis	188
SRP	Solar Radiation Pressure	139
STM	Space Traffic Management	8
TCA	Time of Closest Approach	8
TCM	Trajectory Correction Manoeuvre	9
TLE	Two-Line Elements	112
TRG	Targeting	175
UKF	Unscented Kalman Filter	55
UP	Uncertainty Propagation	21
UPF	Unscented Particle Filter	55
UT	Unscented Transform	39

Chapter 1

Introduction

1.1 Motivation and Rationale

Uncertainty is ubiquitous in astrodynamics and space systems. Uncertainty comes in different forms and can represent random phenomena or uncertain quantities, such as imperfect state knowledge, execution errors or missed thrust, system failures, partially known system parameters, and unknown dynamical constants. Such uncertainties affect the majority of disciplines in space, such as state estimation of space objects, trajectory design, guidance navigation and control, operations planning with uncertain information, flight dynamics, navigation analysis, and so on.

Traditional large spacecraft typically rely on multiple redundancies, accurate and established systems, and extensive ground networks to mitigate the potential risks. In the last years, the trend has shifted toward smaller satellites as a low-cost high-efficiency alternative. However, such a rapidly growing class of missions is characterised by limited resources and budgets, therefore entailing new challenges for robust risk mitigation.

In the near-Earth environment, the number of space objects has been swiftly growing due to the enlarged range of terrestrial services that space technology provides. In this context, next to traditional satellites, small satellites have gained significant momentum as they represent a cost-efficient solution for Earth observation, communications, and science [1, 2]. Therefore, access to space has become affordable for smaller organisations like universities, research centres, and companies of any size, given the explosion of low-cost small platforms, which have reached an adequate technology readiness level [3]. Another recent trend is the use of mega-constellations of hundreds to thousands of satellites to achieve global coverage by New Space actors, as SpaceX's Starlink [4] and Planet Labs' Dove [5], or the planned Amazon's Project Kuiper, Telesat LEO, and Boeing's [6, 7]. The consequence of this fast growth is that near-Earth space traffic is becoming more and more congested, causing a sharp increase in the number of conjunction events. Even in the unrealistic "no new launches" scenario, studies suggest that, if no mitigation action is enforced, the current Low Earth Orbit (LEO) environment has already reached an unstable condition [8] where collisions will increase noticeably and cause a domino effect [9]. This planned and unplanned traffic surge will pose substantial pressure on current satellite operations to track satellites accurately, analyse more conjunctions, design optimal Collision Avoidance Manoeuvres (CAMs), and upload commands.

On the other hand, the deep-space scene is still dominated by large traditional spacecraft, e.g. the recent JAXA's Hayabusa 2 [10] and NASA's OSIRIS-REx [11] to visit Near-Earth Objects (NEOs). Only recently, developments in components and launchers are enabling deep-space smallsat missions as a viable alternative as they are capable of low-cost, fast, and efficient deep-space exploration. Among others: NASA's CubeSat pairs MarCO were launched alongside InSight and performed independent flight to Mars and served as communications relay during InSight's entry, descent, and landing [12]; JAXA's PROCYON was a CubeSat flyby mission [13] launched in 2014 but later interrupted due to a malfunction; NASA has recently proposed NEA Scout [14], a 6U CubeSat secondary payload mission, aimed at visiting an asteroid; ESA's M-ARGO CubeSat plans to rendezvous with an asteroid and characterise its mass, shape, and surface features [15]; ESA's LUMIO is a 12U CubeSat aiming at observing and characterising meteoroid impacts on the lunar farside [16]. Other current mission proposals to comets and NEAs have been studied as a compromise between cost, scientific return, planetary defence needs, and completion time [17, 18, 19, 20]. However, such spacecraft have limited orbit control capabilities (e.g., limited ΔV, thrust level, and attitude control accuracy), large uncertainties in the state knowledge (e.g., due to limited ground station access) and in the execution (low Technology Readiness Level components), and limited room for margins and system redundancy (limited size and cost) [21, 22]. These structural and operational limitations cause microsats to be more prone to early failure [23]. Furthermore, despite a constant platform's technological development, small space missions beyond the near-Earth environment are still out of the reach of most small- and medium-size stakeholders.

Most of these challenges, both near-Earth and deep-space, stem from a common cause: the operational segment of space missions has not scaled down—in cost, size, and procedures—at the same pace of technology and platform miniaturisation, nor scaled up fast enough to properly support the growing number of satellites [24]. Employing more and more networks and operators or pushing for ever-increasingly precise systems and sensors are traditional solutions to this modern challenge and may not lead to a definitive fix due to intrinsic limitations. To properly tackle increased space traffic, low-resource platforms, and limited budget missions, there is a compelling need to: i) design for robustness and reliability; ii) enhance ground and onboard autonomy. For the former, these quality metrics need to be achieved by smarter mission designs rather than by mere redundancy. For the latter, a disruptive shift requires the decentralisation of information and algorithmic processing, automatisation of resource allocation and risk assessment, and the development of intelligent decision-making agents. Moving toward unsupervised autonomy, Russell and Norvig state "*A system deployed with an incorrect objective will have negative consequences. Moreover, the more intelligent the system, the more negative the consequences*" [25]. Thus, proper quantification and treatment of uncertainty metrics for objectives and constraints are, even more, a critical requirement to reduce negative consequences in low-resource autonomous systems.

To address these challenges, this thesis will focus on methodological developments that can enable: i) state estimation and tracking under complex uncertainty structures; ii) operational autonomy and efficiency; iii) mission robustness by design.

1.1.1 State estimation

Accurately tracking objects is of fundamental importance for both operational satellites and space debris. To deliver increasingly precise services and solid scientific outcomes, operational satellites require multi-source observations to achieve accurate knowledge of their position and velocity to precisely calibrate the instruments, interpret scientific data, and communicate with ground stations. Non-collaborative objects tracking is needed for collision avoidance and re-entry prediction, events which pose a threat respectively to the around-Earth and the terrestrial environment. While modern operational satellites flying below Medium Earth Orbit (MEO) can exploit global navigation satellite systems to estimate their state accurately, non-operational objects are challenging to track [26]. The major issues are the debris' potential small sizes, data latency, data and association uncertainty, and low reflectivity [27]. As for a low-resource deep-space mission scenario, tracking would be even more critical as the number of ground networks with interplanetary capabilities is limited, the associated efforts more onerous, and the construction of competent amateur stations may be unrealistic.

These challenges highlight the need for robust estimation algorithms to deliver reliable metrics and assess conjunction scenarios accurately. In particular, the robustness of state estimation methods depends on their capability of handling sparse observations, severe data uncertainty, partially known system and dynamical parameters.

However, standard estimation techniques for space object tracking and collision avoidance require the definition of a single precise probability measure, either explicitly or implicitly [28]. Indeed, precise probability distributions are employed to model uncertainty both in the prior knowledge of state and in the received observations. Consequently, the correctness of the posterior distribution depends on the quality of priors and likelihood [29]. Furthermore, most methods for state estimation rely on parametric distributions and simplifying assumptions to keep the analysis computationally tractable [30]. For example, the Kalman Filter is based on the assumption (or approximation for nonlinear variants) of Gaussian distributions describing the state prior, the dynamical process noise, and the observation likelihood. Different methods exist to deal with more complex distributions, such as Particle Filter (PF) [31] or Gaussian mixture models [32], but they still rely on the availability of precise and complete information on the probability distributions associated with prior states and measurements.

These estimation techniques have proven to be optimal or efficient primarily because of the assumptions made in their problem statement rather than their accuracy in describing the true uncertainty structure of certain dynamical systems. Indeed, in many real-world applications, it may be difficult to quantify uncertainty with precise probability distributions since it would require a perfect knowledge of all the factors which concur to the definition of such uncertainty and an abundance of data (e.g. full knowledge of the exact dynamics, full sensor characterisation, complete information on the source and quality of the measurements, etc.). In these cases, a more general uncertainty model is required to allow the integrated treatment of a mix of aleatory (irreducible probabilistic variability) and epistemic (lack of knowledge) uncertainty.

In this context, the H-infinity filter [28, 33] may be a viable alternative as it does not require information on the noise statistics. However, this filter minimises the worst-case estimate error of a linear combination of the state only, whereas the collision probability is a highly nonlinear function of such a state. Furthermore, the symmetric positive-definite weight matrices in the H-infinity cost function need to be precisely set, thus not allowing one to express epistemic uncertainty on their values. Recent works have investigated outer probability measures, and particularly possibility functions, to model partial uncertainty specifications for tracking resident space objects [34], collision assessment [35], and a generalisation of admissible region analysis [36]. Estimation methods to handle generalised uncertainty models have been investigated recently. Specifically, estimation approaches under p-box uncertainty [37], set-valued robust Kalman filters [38, 39, 40], convex polytopes to model epistemic sets [41], coherent lower previsions using closed convex sets of probabilities as the model of imprecision [42] have been investigated. However, most of such methods in literature employ closed convex sets of probability distributions as epistemic sets for prior, measurement, and transition uncertainties, and their developments are more theoretical than practical.

The need for accurate uncertainty treatment in orbit determination also concerns the specific context of navigation analysis, which has great relevance in space mission analysis and operations. Currently, navigation analysis is typically tackled as a Monte Carlo simulation of operations and observations since diverse sources of uncertainty affect the spacecraft trajectory [43, 44]. The resulting slow convergence is often tackled by introducing many approximations and assumptions to reduce the computational burden, e.g. dynamical linearisation and Gaussian uncertainties [45, 46, 47]. However, although the larger admissible number of samples reduces the estimator error, the estimated quantity may differ from the true sought one because of the approximations introduced. In other words, the Monte Carlo simulation may deliver accurate statistics of a possibly inaccurate operational scenario. To overcome these potential inaccuracies, the paradigm of navigation analysis should be updated to incorporate more reliable measures of uncertainty.

1.1.2 Operations

Pushing to the limits the number of Earth-orbiting objects and interplanetary missions, the modern challenge of finding efficient methods to face the growing demand for operational services has to be addressed. As seen, one of the key limiting factors is the maturity of the associated ground segment. While usually large missions have a dedicated but expensive ground network, e.g. the Deep Space Network (DSN) [48], smaller or non-collaborative ones often rely on amateur stations or third-party services with reservation slots and time constraints where still humans are involved in the loop to solve scheduling conflicts [49, 50, 51].

Optimisation-based approaches for operations planning and scheduling of ground stations have been proposed, e.g. using Genetic Algorithm (GA) [52, 53], simulated annealing [54], mixed integer programming [55], as ESA's EPS relies on dynamic programming [56]. Specific approaches for low-cost stations have been investigated [57] and compared [58]. Almost the entirety of methods in literature neglects non-deterministic effects typical of real-world operational scenarios to avoid the computational burden of uncertainty quantification routines. However, communication with and tracking of space objects in highly nonlinear dynamical environments, e.g. LEO, are complex tasks affected by different sources of uncertainty. Usually, the initial state knowledge is defective and its density distribution often unknown, in particular for space debris, the dynamical evolution of the state depends on partially known geometrical and dynamical parameters, and the measurements are generally biased and affected by stochastic effects due to sensor and atmospheric noises. Hence, there is a timely need for methods accounting for these stochastic sources to enhance the reliability of operations, tracking and communication scheduling.

In the context of Space Traffic Management (STM), current operational tools need to be updated to efficiently address modern challenges entailed in the so-called New Space era [59]. One key aspect to enhance is the automation of tasks, both onboard the satellite and in-ground facilities. Indeed, in a situation of an increasing number of conjunctions alerts, multiple processes have to be automated for the sake of sustainability [60]. This is the case of close encounters detection and proper identification of the most appropriate action in the face of a potential collision, e.g. the allocation of avoidance manoeuvres. Artificial Intelligence (AI) has been investigated given its ability to learn from data, approximate the dynamical flow, and support decision agents. Along with the capacity to provide faster results once trained, AI appears as a promising tool for automation in space [61, 62]. However, autonomy entails unsupervised analyses and potentially decision-making, which again stress the importance of proper uncertainty modelling and treatment in STM. These tasks are of major importance when predicting collisions and conjunctions between operational satellites and space debris if a limited number of, possibly, low-quality observations are available or some information is missing. Indeed, most collision avoidance approaches employ Conjunction Data Messages (CDMs) as input to quantify the probability of conjunction. However, a standard CDM presents only an estimation of the covariance of the state at the Time of Closest Approach (TCA) without any information on the uncertainty distribution [63]. Although several studies have focused on covariance realism to quantify and improve the covariance estimation accuracy (size, shape, and orientation) [64], they may produce a larger estimate rather than a more accurate one [65]. A solely larger covariance at TCA, although more conservative, may produce diluted collision probability and therefore an underestimated risk [66]. Given the catastrophic impact each decision has in affecting the future stability of the LEO environment [8], a more robust approach to deal with generalised models of uncertainty has to be contemplated to deliver reliable metrics and informed decisions.

1.1.3 Optimal control

Space trajectories are typically optimised as deterministic optimal control problems to meet the science and flight system constraints in a nominal scenario. However, in real-life applications, perfect compliance to the reference trajectory is impossible to achieve as uncertainty always affects the system; uncertainty can be due to imperfect state knowledge, imperfectly known dynamical parameters, missed thrust events, execution errors, and unmodelled perturbations. Furthermore, even small deviations from the desired initial conditions can translate into significant differences in the terminal states in nonlinear systems. Hence, although optimal, the resulting control law might not be robust against the uncertainty naturally affecting the system.

In the design phase, the robustness and reliability of the reference trajectory are usually evaluated a posteriori through a navigation analysis, and the nominal design adjusted through several iterations. The robustness and reliability evaluation is carried out by assessing the mission outcome when the trajectory is affected by different uncertainty realisations. To improve robustness, common practice is to either add a posteriori empirical margins [67, 68], enforced coasting arcs for Trajectory Correction Manoeuvre (TCM), reduce the thrust level or increasing the flyby altitudes. Hence, the design and optimisation of the nominal trajectory are generally decoupled from the quantification of the uncertainty in its realisation. The iteration and handover between trajectory design and navigation analysis to solve flight dynamics' infeasibilities are generally time-consuming and may lead to sub-optimal trajectories with over-conservative margins.

For smallsats, which, as we have seen, are more prone to early failure due to low-resource platforms, there is a compelling reason to design for robustness and reliability from the start. Besides, one can argue that, while trajectory optimisation under uncertainty is an enabling methodology for small spacecraft, it presents advantages also for larger scale traditional missions. In fact, integrating uncertainty from the first phase of the design process leads to optimal trajectories that achieve a better compromise between performance and robustness than deterministic ones with added empirical margins. It also reduces the number of iterations between trajectory design and navigation analysis with a corresponding saving in cost, time, and complexity.

In the specific case of low-thrust trajectories (i.e. trajectories whose dynamics is controlled via a small thrust action compared to the local gravity field), the limited control authority offered by the propulsion system makes corrections to uncertainty in initial or terminal states challenging. If uncertainty is not properly taken into account during the design of the trajectory and the associated control law, there can be no margin to compensate for possible deviations from the desired states. One common cause of trajectory deviation in low-thrust trajectories is missed-thrust due to sub-systems partial failure or external causes, like experienced by the Dawn [69] and Hayabusa missions [70, 71].

In the past decades, a few authors have addressed the problem of introducing either aleatory or epistemic uncertainty or both in the design of space trajectories. Model predictive control or stochastic closed-loop formulations were used to account for correction terms in the control profile [72, 73]. The use of Taylor algebra was investigated to deal with uncertain boundary conditions around a reference trajectory in optimal control and produce a robust control law [74, 75]. The case of a temporary engine failure was investigated by stochastic programming [76, 77]. Differential dynamic programming was applied to trajectory optimisation with an expected value formulation for Gaussian-modelled uncertainties [78]. Approaches based on evidence theory to model uncertainty were developed to optimise transfers under system and dynamical epistemic uncertainties [79, 80, 81]. A generalisation to multi-objective problems and uncertainty modelled with p-boxes was developed to compute families of control laws and tested on a rendezvous scenario [82]. A robust guidance based on stochastic optimal control using a sequence of convex programming and nonlinear programming was proposed and studied for proximity orbit design around small bodies [83].

The common assumption underneath most of these methods is the existence of the desired reference trajectory. Uncertainty is then producing an undesired deviation from it. This problem statement can be formulated implicitly by working with the mean value of the objective and constraints, or explicitly compensating for the deviations. However, often these techniques can deal only with simple families of precise probability density distributions to represent uncertainty. Besides, these methods do not include a rigorous navigation analysis in the optimisation loop to quantify flight dynamics and science requirements. In particular, the Orbit Determination (OD) process with measurements simulations and uncertainty update is not included in previous works. In addition, some of the previous approaches deal with uncertainties of a single nature while employing a tailored formulation to address a specific application.

1.1.4 Common limitations

Two main mathematical pitfalls are shared amongst the discussed state estimation, operations planning, and optimal control practical methods:

- the common use of low-fidelity approximations, e.g. linearisation, of nonlinear dynamical and observation models. These approaches create simplified models and allow one to obtain closed-form solutions for uncertainty propagation, orbit determination, guidance navigation and control, and navigation analysis. Also, for the conjunction analysis of satellites in Earth orbits, most methods rely on analytical or semi-analytical approaches to speed up the collision probability computation. However, in complex dynamics, these approximations introduce non-negligible errors in the quantified uncertainty metrics, which may result in wrong assessments of the involved risks.

- the use of a precise probabilistic model to describe the uncertainties characterising space applications. Indeed, uncertainties on the initial conditions, sensor noises, and system and dynamical parameters are generally modelled as random variables with precisely known probability density functions. Although this often-implicit assumption may seem legitimate, it actually encodes very precise information on the uncertainty structure of the problem, and it gives rise to the following inconsistency: “how can you be so certain about your uncertainty?” Indeed, precise probability theory is a rather restrictive model as it cannot describe partial probability specifications, scenarios characterised by scarce or conflicting information, poorly characterised systems, and so on.

These (often implicit) models are introduced to keep the uncertainty quantification computationally tractable. However, as discussed in detail in the previous sections for each specific topic, these approximations exacerbate the discrepancy between reality and the restrictive mathematical model, making the analyses' outcomes unreliable or not robust.

1.2 Research Objectives and Contribution

Based on the described considerations, the main research objectives and corresponding sub-objectives behind this thesis are:

- to employ scalable nonlinear uncertainty propagation:
  - mitigate the accumulation of uncertainty in multi-segment problems;
  - tunable surrogate modelling to speed up propagation.
- to develop state estimation and navigation approaches under generalised models of uncertainty:
  - sequential filtering approaches that handle epistemic uncertainty;
  - efficient numerical solutions to estimate robust bounds.
- to investigate efficient strategies for space operations under uncertainty and limited resources:
  - optimal tracking by scheduling observations from heterogeneous sources;
  - autonomous resource allocation and decision-making.
- to introduce proper uncertainty treatment in trajectory design for improving mission robustness, reliability, and autonomy:

  - uncertainty quantification in optimal control under epistemic uncertainty;
  - numerical transcription approaches to find optimal and robust trajectories.

Given the motivations outlined in the previous section, the main scientific contributions of this thesis are:

- Time and state continuous tractable filtering under severe uncertainty:
  - general formulation of state estimation under mixed aleatory and epistemic uncertainty;
  - development and theoretical analysis of robust particle filter approach;
  - theoretical results for Branch & Bound bound computation approach with simplicial domains.
- Preliminary development of the concept of epistemic variational inference:
  - epistemic reinitialisation approach to handle epistemic uncertainty in state estimation;
  - use of importance sampling to reduce bound computation to a linear programming problem.
- Polynomial-time algorithm for optimal control under severe uncertainty for robust trajectory optimisation:
  - general formulation of optimal control problems under mixed aleatory and epistemic uncertainty;
  - direct shooting transcription with polynomial algebra and conservative uncertainty reinitialisation;
  - belief optimal control to formulate and solve control problems as observable Markov decision processes;
  - inclusion of navigation analysis and proper uncertainty quantification for objective and constraints in nominal trajectory design.
- Robust Bayesian framework for closed-loop control and navigation analysis:

  - Bayesian agent for autonomous decision-making for optimal action allocation;
  - inclusion of robust state estimation and CAM design under epistemic uncertainty.

- Multi-source observation scheduling for optimal tracking
  - formulation of ground station scheduling as a variable-size optimisation problem;
  - development of tracking metric with nonlinear uncertainty quantification by sequential filtering;
  - employment of structured-chromosome genetic algorithm to efficiently solve variable-size scheduling problem under budget constraints.

- The application of the proposed methods to novel and challenging problems concerning robust space trajectory design, navigation analysis, reliable conjunction assessment, and scheduling of tracking campaigns.

1.2.1 Publications

Part of the content of this thesis was published in journal articles, book chapters, and conference papers. In the following, the list of publications produced in this research period is reported.

Journal publications

1. **C. Greco** and M. Vasile, "Robust Bayesian Particle Filter for Space Object Tracking Under Severe Uncertainty", *Journal of Guidance, Control, and Dynamics*, under review [84];

2. **C. Greco**, S. Campagnola, and M. Vasile, "Robust space trajectory design using belief optimal control", *Journal of Guidance, Control, and Dynamics*, under review [85];

3. L. Gentile, **C. Greco**, E. Minisci, T. Bartz-Beielstein, and M. Vasile, "Satellite tracking with Constrained Budget via Structured-Chromosome Genetic Algorithms", *Optimization and Engineering*, under review [86];

4. L. Walker, M. Di Carlo, **C. Greco**, M. Vasile, and M. Warden, "A mission concept for the low-cost large-scale exploration and characterisation of Near Earth Objects", *Advances in Space Research*, 2020, https://doi.org/10.1016/j.asr.2020.10.038 [18];

5. **C. Greco**, M. Di Carlo, M. Vasile, and R. Epenoy, "Direct multiple shooting transcription with polynomial algebra for optimal control problems under uncertainty", *Acta Astronautica*, 2020, Vol. 170, pp. 224-234, https://doi.org/10.1016/j.actaastro.2019.12.010 [87].

Book chapters

1. **C. Greco** and M. Vasile, "Fundamentals of Filtering", In *Optimization Under Uncertainty with Applications to Aerospace Engineering*, 2021, pp. 181-222, https://doi.org/10.1007/978-3-030-60166-9_6 [88];

2. A. Riccardi, E. Minisci, K. Akartunali, **C. Greco**, N. Rutledge, A. Kershaw, and A. Hashim, "Introduction to Optimisation", In *Optimization Under Uncertainty with Applications to Aerospace Engineering*, 2021, pp. 223-268, https://doi.org/10.1007/978-3-030-60166-9_7 [89].

Peer-reviewed conference papers and presentations

1. L. Gentile, **C. Greco**, E. Minisci, T. Bartz-Beielstein, and M. Vasile, "Structured-chromosome GA optimisation for satellite tracking," In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, 2019, https://doi.org/10.1145/3319619.3326841 [90];

2. **C. Greco**, L. Gentile, G. Filippi, E. Minisci, M. Vasile, and T. Bartz-Beielstein, "Autonomous generation of observation schedules for tracking satellites with

structured-chromosome GA optimisation", In *2019 IEEE Congress on Evolutionary Computation (CEC)*, 2019, https://doi.org/10.1109/CEC.2019.8790101 [91].

Conference papers and presentations

1. **C. Greco**, L. Sánchez Fernández-Mellado, M. Manzi, and M. Vasile, "A Robust Bayesian Agent for Optimal Collision Avoidance Manoeuvre Planning", *8th European Conference on Space Debris*, 2021 [92];

2. **C. Greco** and M. Vasile, "Closing the loop between mission design and navigation analysis", *International Astronautical Congress, The Cyberspace Edition*, 2020, https://strathprints.strath.ac.uk/74361/ [93];

3. **C. Greco** and M. Vasile, "Robust Particle Filter for Space Navigation under Epistemic Uncertainty", *UQOP2020*, 2020 [94];

4. G. Acciarini, **C. Greco**, and M. Vasile, "On the solution of the Fokker-Planck equation without diffusion for uncertainty propagation in orbital dynamics", *2020 AAS/AIAA Astrodynamics Specialist Conference*, 2020, https://strathprints.strath.ac.uk/73897/ [95];

5. **C. Greco**, S. Campagnola, and M. Vasile, "Robust space trajectory design using belief stochastic optimal control", *AIAA SciTech 2020 Forum*, 2020, https://doi.org/10.2514/6.2020-1471 [96];

6. L. Gentile, **C. Greco**, E. Minisci, T. Bartz-Beielstein, and M. Vasile, "An optimization approach for designing optimal tracking campaigns for low-resources deep-space missions", In *70th International Astronautical Congress*, 2019, https://strathprints.strath.ac.uk/70403/ [97];

7. **C. Greco**, L. Gentile, M. Vasile, E. Minisci, and T. Bartz-Beielstein, "Robust particle filter for space objects tracking under severe uncertainty", *2019 AAS/AIAA Astrodynamics Specialist Conference*, 2019, https://strathprints.strath.ac.uk/70566/ [98];

8. L. Walker, **C. Greco**, M. Di Carlo, A. Wilson, L. Ricciardi, A. Berquand, and M. Vasile, “Nanospacecraft exploration of asteroids by collision and flyby reconnaissance”, *Low-Cost Planetary Missions Conference*, 2019, https://strathprints.strath.ac.uk/68659/ [99];
9. M. Di Carlo, M. Vasile, **C. Greco**, and R. Epenoy, “Robust optimisation of low-thrust interplanetary transfers using evidence theory”, *29th AAS/AIAA Space Flight Mechanics Meeting*, 2019, https://strathprints.strath.ac.uk/67543/ [81];
10. **C. Greco**, M. Di Carlo, M. Vasile, and R. Epenoy, “An intrusive polynomial algebra multiple shooting approach to the solution of optimal control problems”, *69th International Astronautical Congress*, 2018, https://strathprints.strath.ac.uk/65918/ [100];
11. **C. Greco**, M. Di Carlo, L. Walker, and M. Vasile, “Analysis of NEOs reachability with nano-satellites and low-thrust propulsion”, *4S Symposium*, 2018, https://strathprints.strath.ac.uk/64029/ [17].

1.2.2 Thesis structure

This thesis is organised into two parts. Part I concerns the theoretical and methodological developments to address challenges in uncertainty propagation, state estimation, navigation, and control under generalised models of uncertainty. Part II presents advanced applications of such methods to conjunctions analysis, space trajectory design, navigation analysis, and optimal scheduling.

In the first part, Chapter 2 first introduces some common definitions of dynamical and uncertainty models which will be used throughout the thesis, then it describes the common nonlinear uncertainty propagation approaches employed for the state estimation, tracking, and optimal control methods developed in later chapters of the thesis. Specifically, first tunable polynomial approximations of the dynamical flow constructed via both intrusive algebra or stochastic collocation are discussed to speed up the propagation approach. In this section, a reinitialisation approach is developed to avoid the accumulation of uncertainty in different time segments and mitigate the curse of dimensionality. Then, numerical approaches for computing expectations and moments of the propagated distribution are discussed.

Chapter 3 concerns developments in the field of state estimation and navigation under mixed aleatory and epistemic uncertainty. The filtering problem is formulated in terms of expectations to estimate given noisy indirect observations of the state of a dynamical system. Thus, the epistemic filter output is a robust interval whose bounds are the expectation's extrema resulting from epistemic uncertainty. Two solution approaches are developed: a generalisation of the particle filter to solve the optimisation routine to compute the robust bounds efficiently; a sequential filtering approach based on a combination of variational inference and importance sampling. These approaches aim at removing the need for large two-level Monte Carlo sampling while maintaining the dynamical and observation nonlinearities.

Chapter 4 deals with the development of novel problem formulations and solution approaches for optimal control problems under mixed aleatory and epistemic uncertainty. Two methods are presented: one tailored for open-loop trajectory optimisation, which employs generalised intrusive polynomial algebra to describe the dynamical flow; one designed for open- and closed-loop control design, which employs a belief optimal control formulation and non-intrusive stochastic collocation as a surrogate of the dynamics. The latter generalises the former by incorporating a full navigation analysis in robust trajectory optimisation. These approaches allow one to design trajectories that are both optimal and robust against uncertainties by closing the loop between trajectory design and navigation analysis.

In the second part, Chapter 5 concerns the application of the state estimation approach developed as well as the use of a robust Bayesian framework for autonomous decision-making for STM. The common test case involves the robust estimation of the collision probability in a space debris conjunction scenario. In this application, the crucial effect of accounting for epistemic uncertainty is demonstrated, and the impact of introducing new measurements on the risk metric is analysed. Successively, the robust Bayesian agent is employed to automatically allocate optimal actions, e.g. CAMs, based on the robust filter metrics for the different conjunctions studied.

Chapter 6 presents a new scheduling algorithm for optimal tracking of satellites from ground to reduce the knowledge uncertainty while taking into account operational budgets. The fitness metric is the satellite final uncertainty, and therefore all the pipeline composed of stations' visibility, observation simulation, uncertainty propagation, and orbit determination is implemented. Hence, the scheduling problem is formulated as a variable-size optimisation and a tailored solver employed. Then, the optimisation-based approach is applied to the optimal tracking of both a LEO satellite and a low-resource deep-space spacecraft. For the former, different ground station network configurations and budget constraints are analysed. In the latter, TCMs and their execution errors are included as well after each observation arc in the overall uncertainty quantification.

Chapters 7 and 8 show the application of the optimal control methods under uncertainty developed. Specifically, the first chapter discusses the robust optimisation of a low-thrust rendezvous trajectory to the near-Earth asteroid 99942 Apophis subject to uncertain hyperbolic excess velocity at departure. The second one presents the robust re-design of one flyby of Europa Clipper's tour to simultaneously optimise the total cost of the executed manoeuvres and to respect a constraint on the probability of impact with Europa during the close encounter. Furthermore, on Europa Clipper, the epistemic variational inference is applied for a preliminary navigation analysis to consider epistemic uncertainty also on the likelihood distributions.

Finally, Chapter 9 closes the thesis with a discussion on the main developments and results obtained and a discussion of potential future research directions.

Part I

Theory and Algorithms

Chapter 2

Uncertainty Propagation and Expectation Estimation

The content of this chapter was published in:
C. Greco and M. Vasile, “Fundamentals of Filtering”, In *Optimization Under Uncertainty with Applications to Aerospace Engineering*, 2021 [88]
C. Greco, M. Di Carlo, M. Vasile, and R. Epenoy, “Direct multiple shooting transcription with polynomial algebra for optimal control problems under uncertainty”, *Acta Astronautica*, 2020 [87].

This chapter introduces the uncertainty models and propagation techniques for a state- and time-continuous dynamical system, which will be employed throughout the thesis. In particular, it first introduces the epistemic model, which is used to describe severe uncertainty stemming from the lack of knowledge of system and dynamic parameters. Then, it contextualises the full Uncertainty Propagation (UP) problem for probability distributions in dynamical systems and Markov processes and discusses why numerical methods and approximations are needed for practical UP. Hence, the two-step approach employed in this thesis for state estimation, tracking, and optimal control is presented. First, a tunable polynomial expansion of the dynamical flow is constructed utilising either intrusive algebra or stochastic collocation. For this expansion, a reinitialisation approach is developed to reduce the dimensionality in multi-segment or multi-phase problems. In the end, Monte Carlo and Gauss-Hermite approaches are discussed for computing expectations and moments of the propagated distribution. The first class is employed for generic densities, the second one for the special case of normal ones as Gaussian distributions are extensively employed for uncertainty modelling in estimation, tracking, and control problems in space. These practical methods, as well as the intrusive algebra and stochastic collocation, can be found in literature, so only a brief description is provided to improve the thesis readability, whereas more space is dedicated to the novel reinitialisation contribution.

The chapter is structured as follows. The uncertainty model employed in this thesis is presented in Section 2.1. The general problem statement of nonlinear uncertainty propagation in dynamical systems and the challenges of its practical computation are discussed in Section 2.2. Then, the two-step approach for efficient UP is described. The tunable polynomial representation of the flow and the reinitialisation strategy are discussed in Section 2.3. Then, numerical techniques for computing expectations or moments of the propagated distribution are described in Section 2.4.

2.1 Epistemic Uncertainty

For a generic random variable $\boldsymbol{\Xi} : \boldsymbol{\Omega}_{\boldsymbol{\Xi}} \to \mathbb{R}^{n_\xi}$, from its sample set $\boldsymbol{\Omega}_{\boldsymbol{\Xi}}$ to its measurement space $\mathbb{R}^{n_\xi}$ (n_ξ real coordinate space), we will write parametric epistemic uncertainty as

$$\boldsymbol{\Xi} \sim p(\boldsymbol{\xi}; \boldsymbol{\lambda}_\xi) \in \mathcal{P}_{\xi;\boldsymbol{\lambda}_\xi} \ , \tag{2.1}$$

to indicate that its probability density function (pdf) belongs to the family $\mathcal{P}_{\xi;\boldsymbol{\lambda}_\xi}$ parameterised in the epistemic parameter $\boldsymbol{\lambda}_\xi$ as

$$\mathcal{P}_{\xi;\boldsymbol{\lambda}_\xi} = \left\{ p(\boldsymbol{\xi}; \boldsymbol{\lambda}_\xi) \,|\, \boldsymbol{\lambda}_\xi \in \boldsymbol{\Omega}_{\boldsymbol{\lambda}_\xi} \right\} \ , \tag{2.2}$$

where $\boldsymbol{\Omega}_{\boldsymbol{\lambda}_\xi}$, a compact subset of $\mathbb{R}^{n_\lambda}$, is the epistemic parameter domain [101]. When information is sufficient to identify a single distribution, uncertainty is purely aleatory and $\boldsymbol{\Omega}_{\boldsymbol{\lambda}_\xi}$ degenerate to a singleton. Therefore, Eq. (2.2) is as a generalisation of the pure aleatory model.

For ease of notation, in some cases we will use uppercase special characters $\mathcal{Z}_{\boldsymbol{\lambda}_\xi} = p(\boldsymbol{\xi}; \boldsymbol{\lambda}_\xi)$ to indicate the pdf of a random variable $\boldsymbol{\Xi}$. The explicit dependence on the epistemic parameter will be dropped when not essential.

Given a functional of interest

$$\Psi : \mathcal{P}_{\xi;\boldsymbol{\lambda}_\xi} \to \mathbb{R} \ , \tag{2.3}$$

which is a mapping from the pdf space to the real line, its inferior and superior values over the set $\mathcal{P}_{\xi,\boldsymbol{\lambda}_\xi}$ are named lower bound $\underline{\Psi}$ and upper bound $\overline{\Psi}$, respectively. For the epistemic specification as in Eq. (2.2), the lower and upper bounds are defined as

$$\underline{\Psi} = \inf_{\boldsymbol{\lambda}_\xi \in \boldsymbol{\Omega}_{\boldsymbol{\lambda}_\xi}} \Psi \left[p(\boldsymbol{\xi}; \boldsymbol{\lambda}_\xi) \right] \tag{2.4a}$$

$$\overline{\Psi} = \sup_{\boldsymbol{\lambda}_\xi \in \boldsymbol{\Omega}_{\boldsymbol{\lambda}_\xi}} \Psi \left[p(\boldsymbol{\xi}; \boldsymbol{\lambda}_\xi) \right] \ . \tag{2.4b}$$

For the special case of aleatory uncertainty, we have $\underline{\Psi} = \overline{\Psi}$ because $\boldsymbol{\Omega}_{\lambda_\xi}$ is a singleton.

In the context of optimal control under uncertainty and filtering, we will often employ an integral functional in the form of expectation of a generic function $\psi : \mathbb{R}^{n_\xi} \to \mathbb{R}$ as

$$\Psi \left[p(\boldsymbol{\xi}; \boldsymbol{\lambda}_\xi) \right] = \mathbb{E}_{p_{\boldsymbol{\lambda}_\xi}} \left[\psi(\boldsymbol{\xi}) \right] = \int_{\boldsymbol{\Omega}_{\boldsymbol{\Xi}}} \psi(\boldsymbol{\xi})\, p(\boldsymbol{\xi}; \boldsymbol{\lambda}_\xi)\, d\boldsymbol{\xi} \ . \tag{2.5}$$

In the following, we show how this functional formulation encloses common alternatives:

- *Expected Value*: by definition, the expectation operator of a generic function of a random variable returns the expected value of that function, which is Eq. (2.5) directly.

- *Probability*: the probability of an event is the expectation of the indicator function of such event. Thus, to compute the probability of a generic event $\boldsymbol{\xi} \in A$, the auxiliary function ψ is defined as

$$\psi(\boldsymbol{\xi}) = \mathbb{I}_A(\boldsymbol{\xi}) \ , \tag{2.6}$$

where

$$\mathbb{I}_A(\boldsymbol{\xi}) = \begin{cases} 1 & \text{if } \boldsymbol{\xi} \in A \\ 0 & \text{if } \boldsymbol{\xi} \notin A \ . \end{cases} \tag{2.7}$$

Hence, chance constraints can be formulated using the indicator function and the functional formulation.

Another relevant example is the percentile of a random variable, often used as an objective or constraint. Let ψ^* be the $(1-\alpha)$-percentile of $\psi(\boldsymbol{\xi})$ such that

$$Pr(\psi(\boldsymbol{\xi}) < \psi^*) = 1 - \alpha \ . \tag{2.8}$$

We can compute the probability on the left as an expectation by using the indicator function as

$$\psi(\boldsymbol{\xi}) = \mathbb{I}_{\psi<\psi^*}(\boldsymbol{\xi}) \ , \tag{2.9}$$

where

$$\mathbb{I}_{\psi<\psi^*}(\boldsymbol{\xi}) = \begin{cases} 1 & \text{if } \psi(\boldsymbol{\xi}) < \psi^* \\ 0 & \text{if } \psi(\boldsymbol{\xi}) \geq \psi^* \ . \end{cases} \tag{2.10}$$

- *Higher-order moments*: by definition, the n^{th} moment of a pdf is the expectation of its n^{th} power. For example, the covariance of the random variable is computed by setting the auxiliary function ψ to

$$\psi(\boldsymbol{\xi}) = \big(\boldsymbol{\xi} - \mathbb{E}[\boldsymbol{\xi}]\big)\big(\boldsymbol{\xi} - \mathbb{E}[\boldsymbol{\xi}]\big)^T \ . \tag{2.11}$$

2.2 Nonlinear uncertainty propagation

Uncertainty Propagation (UP) deals with the computation of the probability density function (pdf) $p(\mathbf{z})$ of an output variable $\mathbf{z}$ given a generic function $H : \mathbb{R}^{n_\xi} \to \mathbb{R}^{n_z}$ and the probability density function of the input variable $p(\boldsymbol{\xi})$. This task is of fundamental importance for all the topics treated in this thesis. Indeed, state estimation, tracking, and optimal control under uncertainty all require the ability to compute how input variables' errors map to state uncertainty, how the state distribution evolves in time, how state uncertainty maps to observation uncertainty, and so on. Thus, in the remainder of this chapter, $p(\boldsymbol{\xi})$ can represent either the input pdf or the prior distribution for sequential filtering and control.

The explicit expression of $p(\mathbf{z})$ is possible only in a restricted number of cases. For example, if H is a diffeomorphism with $n_\xi = n_z$, then the output distribution can be expressed by means of the transformation's Jacobian [102]

$$p(\mathbf{z}) = \left| \frac{\partial H}{\partial \boldsymbol{\xi}} \right|^{-1} p(\boldsymbol{\xi}) \ . \tag{2.12}$$

As another well-known example, Kolmogorov derived equations for the exact evolution of the state density function $p(\mathbf{x})$ in the context of stochastic differential equations governed by a Wiener process. The Kolmogorov forward equation, or mostly known as Fokker-Planck Equation (FPE), is a Partial Differential Equation (PDE) that is given by [103, 104]:

$$\frac{\partial p(\mathbf{x})}{\partial t} = -\sum_{i=1}^{n_x} \frac{\partial}{\partial x_i} \big[f_i(t, \mathbf{x}) p(\mathbf{x}) \big] + \sum_{i=1}^{n_x} \sum_{j=1}^{n_x} \frac{\partial^2}{\partial x_i \partial x_j} \big[D_{ij}(t, \mathbf{x}) p(\mathbf{x}) \big] \ , \tag{2.13}$$

where f_i is i^{th} equation of the dynamical model, whereas D_{ij} are the elements of the diffusion tensor. It goes without saying that Eq. (2.13) holds under existence and continuity assumptions on the involved partial derivatives. This equation has been generalised to include more general stochastic perturbations other than the Gaussian white noise [105]. This equation has a closed-form solution in a limited number of simplified cases only [106]. UP through the FPE requires the solution of a partial differential equation, which is a laborious and time-consuming task.

In the context of Markov processes, the transition probability

$$p(\mathbf{x}_t | \mathbf{x}_0) \tag{2.14}$$

is often used as probabilistic model of the dynamical flow. This conditional distribution expresses the likelihood of the random system state to be $\mathbf{X}_t = \mathbf{x}_t$ given that the initial state was $\mathbf{X}_0 = \mathbf{x}_0$. Indeed, in contrast to deterministic flows, the state $\mathbf{x}_t$ is not uniquely defined for a given initial condition $\mathbf{x}_0$ because of uncertainty on the static parameters and possible process noise. Hence, in the Markov model, the propagated density can be computed by the Chapman-Kolmogorov equation [30]:

$$p(\mathbf{x}_t) = \int p(\mathbf{x}_t, \mathbf{x}_0)\, d\mathbf{x}_0 = \int p(\mathbf{x}_t|\mathbf{x}_0)\, p(\mathbf{x}_0)\, d\mathbf{x}_0 \ , \tag{2.15}$$

where the first equality follows from the definition of marginal densities, the second equality stems from the definition of the joint probability with respect to conditional one.

In general, the main difficulty stems from the transformation being nonlinear and non-diffeomorphic. Even when the input distribution is set to be a well-behaved parametric distribution, e.g. a normal distribution, the output pdf is not conjugate as it would have higher-order moments due to the transformations (2.12), (2.13), or (2.15). Indeed, although the deterministic dynamical flow is a diffeomorphism for any fixed time and static parameter, its nonlinearity yields a non-Gaussian evolution of the state uncertainty [107].

As we have seen, the full propagated distribution is generally expensive to compute, and its complete knowledge provides greater information than needed for most practical applications. Therefore, state estimation and control algorithms primarily focus on the computation of the expectation of the generic function ψ as in Eq. (2.5). In the UP context, the expectation can be written with respect to the input distribution

$$\mathbb{E}_z\big[\psi(\mathbf{z})\big] = \int \psi(\mathbf{z})p(\mathbf{z})d\mathbf{z} = \int \psi(H(\boldsymbol{\xi}))p(\boldsymbol{\xi})d\boldsymbol{\xi} = \mathbb{E}_\xi\big[\psi(H(\boldsymbol{\xi}))\big] \ . \tag{2.16}$$

by exploiting the law of the unconscious statistician theorem [108]. This powerful property allows one to compute generic expectations without computing the full propagated distribution first.

Under epistemic uncertainty, such expectation is interval-valued and the extrema need to be computed through optimisation, as in Eq. (2.4). Hence, efficient numerical tools are even more critical for high-fidelity uncertainty propagation under epistemic uncertainty, either to approximate the full propagated distribution or some expectation with respect to it.

2.3 Polynomial dynamical flow

Let us study a dynamical system $\mathbf{x} \in \mathbb{R}^{n_x}$ evolving according to a nonlinear Ordinary Differential Equation (ODE)

$$\dot{\mathbf{x}} = f(t, \mathbf{x}, \mathbf{d}, \mathbf{u}), \tag{2.17}$$

where $t \in \mathbb{R}$ is the independent variable (usually time), $\mathbf{d} \in \mathbb{R}^{n_d}$ are static parameters, and $\mathbf{u} \in \mathbb{R}^{n_u}$ control variables.

Under uncertainty, let the true initial state $\mathbf{x}_0$ and the true value of the parameters $\mathbf{d}$ be unknown realisations of the random variables $\mathbf{X}_0 : \boldsymbol{\Omega}_{\mathbf{X}_0} \to \mathbb{R}^{n_x}$ and $\mathbf{D} : \boldsymbol{\Omega}_{\mathbf{D}} \to \mathbb{R}^{n_d}$ respectively, where $\boldsymbol{\Omega}_{(\cdot)}$ denotes the sample space, with pdf $p(\mathbf{x}_0)$ and $p(\mathbf{d})$. These random variables together with Eq. (2.17) induce the state at a later time to be a random variable itself $\mathbf{X}_t$. Explicitly, if for all $t \geq 0$ the map $F_0^t : \mathbb{R}^{n_x} \to \mathbb{R}^{n_x}$ is the point-wise solution to (2.17)

$$F_0^t(\mathbf{x}_0, \mathbf{d}) := \mathbf{x}_0 + \int_0^t f(\tau, \mathbf{x}, \mathbf{d}, \mathbf{u}) \, d\tau \ , \tag{2.18}$$

then $\mathbf{X}_t$ is a random variable governed by the push-forward measure $p(\mathbf{x}_t) := p(\mathbf{x}_0, \mathbf{d}) \circ F_0^{t^{-1}}$.

Let us collect the uncertain variables in a single joint random variable $\boldsymbol{\Xi}$ with realisations $\boldsymbol{\xi} = [\mathbf{x}_0, \mathbf{d}]$. Consider now the sample space $\boldsymbol{\Omega}_{\boldsymbol{\Xi}}$ of all the possible values of the uncertain variables, and a polynomial of degree q in n_ξ variables $P_{\mathbf{x}}(t, \boldsymbol{\xi}) \in T_{q,n_\xi}$, where T_{q,n_ξ} is a polynomial space for any fixed t. The interest lays in the time evolution of the set $\boldsymbol{\Omega}_{\boldsymbol{\Xi}}$, so that the evolved set $\mathcal{F}_0^t(\boldsymbol{\xi})$ can be calculated at any point in time. In other words, first the set (assumed compact and finite) is propagated in time with a polynomial model. Once the set is propagated, the relevant statistical quantities can be computed by sampling the polynomial representation according to a specific input distribution $p(\boldsymbol{\xi})$. As long as the dynamics is integrable for all values in the set, no additional assumption on $\boldsymbol{\xi}$ is required to propagate the set.

Let the set $\mathcal{F}_0^t(\boldsymbol{\xi})$ be formally defined as follows. For a given realisation of the uncertain vector $\boldsymbol{\xi}$ and a control law $\mathbf{u}$ the propagated state $\mathbf{x}(t)$ at time t is:

$$\mathbf{x}(t, \boldsymbol{\xi}) = \mathbf{x}_0(\boldsymbol{\xi}) + \int_0^t f(\tau, \mathbf{x}_0(\boldsymbol{\xi}), \mathbf{u}, \mathbf{d}(\boldsymbol{\xi}))d\tau \ , \tag{2.19}$$

and the set $\mathcal{F}_0^t(\boldsymbol{\xi})$ is:

$$\mathcal{F}_0^t(\boldsymbol{\xi}) = \{\mathbf{x}(t, \boldsymbol{\xi}) \,|\, \boldsymbol{\xi} \in \boldsymbol{\Omega}_{\boldsymbol{\Xi}}\} \ . \tag{2.20}$$

Here the assumption is that the dynamics f is integrable for all values of $\boldsymbol{\xi} \in \boldsymbol{\Omega}_{\boldsymbol{\Xi}}$ and $\tau \in [t_0, t]$. Note that if $\mathbf{x}$ was not a continuous function of $\boldsymbol{\xi}$, a set $\hat{\boldsymbol{\Omega}}_{\boldsymbol{\Xi}}$ of a continuous function $\hat{\mathbf{x}}$ of $\boldsymbol{\xi}$ that contains all the realisation of $\mathbf{x}$ induced by $\boldsymbol{\xi}$ could still be considered.

The set $\boldsymbol{\Omega}_{\mathbf{X}_0}$ is now approximated with the polynomial $P_{\mathbf{x}}(t_0, \boldsymbol{\xi}) = P_{\mathbf{x}}(t_0, \mathbf{x}_0(\boldsymbol{\xi}))$ and this approximation propagated forward in time to obtain:

$$\mathcal{F}_0^t(\boldsymbol{\xi}) \approx \widetilde{F}_0^t(\boldsymbol{\xi}) = P_{\mathbf{x}}(t, \boldsymbol{\xi}) = \sum_{i=1}^{N_q} \alpha_i(t) P_i(\boldsymbol{\xi}) \ , \tag{2.21}$$

where $N_q = \binom{n_\xi + q}{q}$ are the number of terms retained in the truncated representation, P_i is a multivariate polynomial basis and $\alpha_i(t)$ the associated coefficient.

Once $\widetilde{F}_0^t(\boldsymbol{\xi})$ is available, one can compute the expectation (2.16) of a generic function of the propagated uncertain state as

$$\mathbb{E}\big[\psi(\mathbf{x})\big] = \mathbb{E}\big[\psi\big(\widetilde{F}_0^t(\boldsymbol{\xi})\big)\big] = \int_{\boldsymbol{\Omega}_{\boldsymbol{\Xi}}} \psi\big(\widetilde{F}_0^t(\boldsymbol{\xi})\big) p(\boldsymbol{\xi}) d\boldsymbol{\xi} \ . \tag{2.22}$$

2.3.1 Generalised intrusive polynomial algebra

The polynomial approximation $\widetilde{F}_0^t(\boldsymbol{\xi})$ can be computed with Generalised Polynomial Algebra (GPA) first introduced in [109] and then further developed in [110, 111]. This approach generalises the well-known Taylor algebra [112] which has been extensively applied in orbital mechanics [113, 114, 115], trajectory control under uncertainty [116, 74, 75], nonlinear filtering [117], conjunction analysis [118, 119], and other areas. This generalisation to other polynomial bases allows one to exploit different bases' properties which may be beneficial for specific applications. For example, Chebyshev polynomials have better uniform convergence over the expansion region even in the presence of a finite number of discontinuities but are characterised by a higher computational complexity [120, 121, 111]. This property is particularly valuable under epistemic uncertainty. Indeed, the dynamical approximation should be accurate on the envelope of all the pdfs' supports within the epistemic set. Besides, in general, there will not be a single central point around which the representation could be constructed.

Hence, GPA starts by expressing the generic polynomial $P_{\mathbf{x}}(t_0, \boldsymbol{\xi})$ into monomial bases so that $P_{\mathbf{x}}(t_0, \boldsymbol{\xi}) \in T'_{q,n_\xi}$, where T'_{q,n_ξ} is the functional space of the polynomial in monomial bases. A set of algebraic operations between monomials is then introduced defining an algebra in T'_{q,n_ξ}. Denoting with V and W the multivariate polynomial expansions of two functions v and w, the algebraic operation $\oplus = \{+, -, \cdot, /\}$ between real-valued functions has its correspondent $\otimes$ in the monomials space:

$$v(\boldsymbol{\xi}) \oplus w(\boldsymbol{\xi}) \sim V(\boldsymbol{\xi}) \otimes W(\boldsymbol{\xi}) \in T'_{q,n_\xi} \ . \tag{2.23}$$

The result of the addition (or equivalently subtraction) of two elements of T'_{q,n_ξ} is still an element of the same functional space (closed under addition). To keep the order of the polynomials constant, the multiplication result is truncated to the order q, and so is the result of any composition. Composition rules are defined to handle division and other elementary functions such as trigonometric functions, exponents, logarithms, etc. These polynomial operations are implemented using the C++ overloading operator within the Strathclyde Mechanical and Aerospace Research Toolbox for Uncertainty Quantification (SMART-UQ) [110].

Given this set of operations, any integrator for the propagation of ordinary differential equations can be directly applied to the propagation of a set of states through the dynamical system Eq. (2.17) once the set is expressed as an element of T'_{q,n_ξ} (for more details and examples refer to [111]).

During propagation, the uncertain model parameters are handled similarly to the initial conditions. The parameter sample domain $\boldsymbol{\Omega}_{\mathbf{D}}$, induced by the random variable

Ξ, is bounded by a constant multivariate polynomial $P_{\mathbf{d}} \in T_{q,n_\xi}$, which is composed in $P_{\mathbf{x}}(t, \boldsymbol{\xi})$ through the dynamical operations.

Intrusive polynomial algebra has shown high representation accuracy even with rather low-order polynomials at competitive time complexities, and it has demonstrated good scalability properties for higher-dimensional problems [111]. This methodology has been successfully applied to multiple uncertainty propagation scenarios in aerospace applications [109, 122, 110, 111].

2.3.2 Non-intrusive polynomial representation

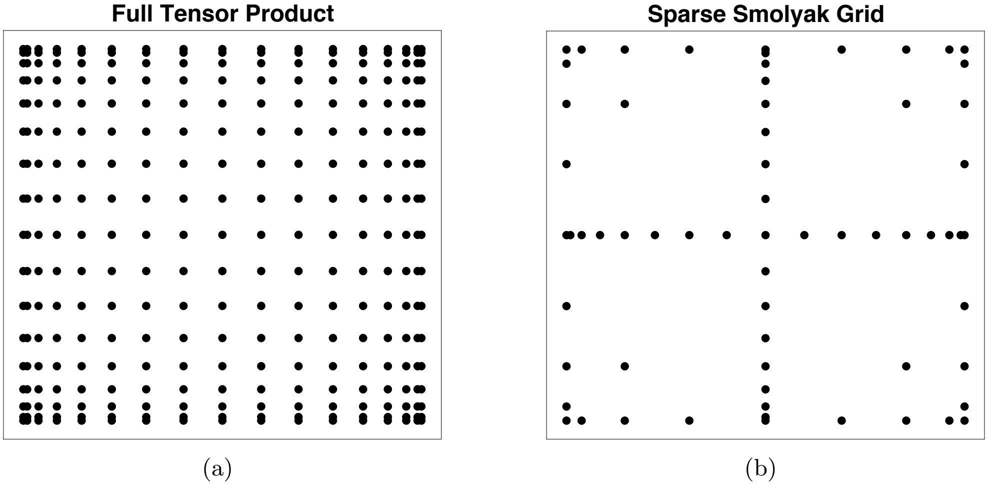
Intrusive algebra requires a very specific interface with the dynamical models. When the dynamics is available as a black-box model only, e.g. part of a numerical tool as GMAT [123] or jTOP [21], intrusive approaches cannot be employed, and non-intrusive alternatives should be preferred.

In this thesis, for constructing the non-intrusive polynomial surrogate, the coefficients α_i in Eq. (2.21) are computed using stochastic collocation [124]: a number of samples $\mathbf{x}(t, \boldsymbol{\xi}^{(j)})$ is first evaluated over a structured grid of collocation points $\boldsymbol{\xi}^{(j)}$; then the polynomial approximation $\widetilde{F}_0^t$ is built to fit the set of $\mathbf{x}(t, \boldsymbol{\xi}^{(j)})$ vectors. Because the stochastic and control dimensions can be rather large, the Smolyak polynomial space-variant is used to limit the growth of collocation points with increasing stochastic dimensions. Figure 2.1 shows an example of a grid coming from a full tensor product (a) and a sparse Smolyak grid (b) both constructed with a level two grid, that is 17 nodes per dimension in 2D.

The grid level is a tunable parameter that determines the number of collocation points in each dimension. Samples are unique across different grid levels. Therefore, one can sequentially increment the grid level while reusing previously computed samples until reaching the requested accuracy.

There are several advantages of this sparse polynomial mapping by stochastic collocation over other methods for uncertainty propagation in trajectory optimisation applications:

- being a non-intrusive method, the dynamical model is called as a black-box func-



(a) (b)

Figure 2.1: Comparison of two-dimensional grids constructed by (a) full tensor product and (b) sparse Smolyak rule of one-dimensional grids.

tion, thus pre-existing libraries can be easily interfaced with such transcription without alteration;

- the accuracy of the polynomial approximation can be improved by increasing the degree q of the polynomial space; the growth of the number of collocation points with the degree q is limited by the sparse Smolyak variant;
- Chebyshev polynomials can be employed for their global convergence properties over a compact set [111] and previous applications in a number of aerospace cases [87, 122, 109].

2.3.3 Dynamic Reinitialisation

The uncertainty propagation could be performed with a single numerical integration from initial to time of interest to get:

$$\widetilde{F}_0^t(\boldsymbol{\xi}) = P_{\mathbf{x}}(t, \boldsymbol{\xi}) \ . \tag{2.24}$$

The advantage of this approach is that the numerical uncertainty propagation has to be performed only once, and then the expectation can be computed by means of an inexpensive polynomial evaluation. However, using a single polynomial representation exacerbates the curse of dimensionality typical of uncertainty propagation, that is, a critical increase in computation complexity [125] or decrease in the representation accuracy as the number of uncertain variables grows [125]. In GPA for example, the number of terms in the polynomial expansion grows with $(n_\xi + q)!/(n_\xi!q!)$ and consequently the algorithmic complexity of its computations. Thus, this approach can only handle a limited number of uncertain model parameters. When different (and independent) uncertain parameters affect the system evolution at different instants of time (e.g. multi-segment multi-phase trajectories, discretised control with disturbances, etc.), the size of the polynomial would quickly become intractable.

Consider now the uncertain model parameter vector to be defined as follows:

$$\mathbf{D} = [\mathbf{D}_0, \mathbf{D}_1, \ldots, \mathbf{D}_k, \ldots] \ ,$$

such that the realisations of the uncertain model parameters $\mathbf{D}_k$ manifest only over the time interval $(t_k, t_{k+1}]$. For instance, the errors due to manoeuvres executed in a given temporal interval influence the belief state until the end time of that interval. Observation errors are another example of uncertainty sources that affect the belief state update at a given time only. The effect of these errors, beyond the end of the interval where they manifest, is carried by the belief state at the start of the next interval. In this case, the polynomial UP can be applied to each sub-segment independently of the uncertain model parameters affecting the other segments. Consequently the algebra dimension in the k^{th} segment reduces to $n_{\xi_k} = n_x + n_{d_k}$, namely the number of the uncertain state variables $\mathbf{X}_k$ at the beginning of the segment and the number of uncertain model parameters $\mathbf{D}_k$ affecting the system for $t \in (t_k, t_{k+1}]$ only. With this scheme, the polynomial $P_{\mathbf{x}_k}$ representing the uncertain states at t_k is propagated to $P_{\mathbf{x}_{k+1}}$ at t_{k+1} only under the effect of uncertain parameters $\mathbf{D}_k$. In this decomposition, uncertainties that affect the dynamics over multiple or all arcs are allowed, and they need to be replicated across all the segments they affect, e.g. uncertain dynamical model parameters. Nonetheless, this decoupling enables one to favourably exploit the Markov property typical of these dynamical systems.

However, the uncertainty decoupling and sequential approach pose a key challenge: how to match the polynomial representation at the boundaries of each time segment. Indeed, the main difficulty of the proposed discretisation arises from the necessity to impose continuity conditions between two hyper-dimensional uncertainty sets at the boundary of two adjacent segments.

If one calls $F_{k+1}^{(p)}$ the set of states at time t_{k+1} and $F_{k+1}^{(g)}$ the superset such that $F_{k+1}^{(p)} \subseteq F_{k+1}^{(g)}$, the corresponding polynomial approximations become $P_{\mathbf{x}_{k+1}}^{(p)}$ and $P_{\mathbf{x}_{k+1}}^{(g)}$, where $P_{\mathbf{x}_{k+1}}^{(p)}$ is a function of $\mathbf{X}_k$ and $\mathbf{D}_k$, and $P_{\mathbf{x}_{k+1}}^{(g)}$ is the so-called re-initialised set of uncertain states $\mathbf{X}_{k+1}$ at time t_{k+1}. Note that the stochastic vector $\mathbf{D}_{k+1}$ has realisation only over the open interval $(t_{k+1}, t_{k+2}]$.

Hence, the propagation phase is carried out as described in Algorithm 1, and is represented in the cartoon in Fig. 2.2 for a two-dimensional example. In Fig. 2.2 the initial uncertainty set expanded in polynomial basis $P_{\mathbf{x}_0}$ at t_0 (depicted in grey) is propagated through the dynamical system (intrusively or non-intrusively) to $P_{\mathbf{x}_1}^{(p)}$ at the end of the first segment t_1 (depicted in blue). In order to obtain an uncertainty representation function of the variables of the next segment only, the uncertain region $P_{\mathbf{x}_1}^{(p)}$ is reinitialised as a hyper-box $P_{\mathbf{x}_1}^{(g)}$ (again, in grey in Fig. 2.2) which conservatively bounds the propagated set. The propagation and reinitialisation steps are repeated for each segment.

Algorithm 1 Polynomial Propagation Phase of Reinitialisation Approach

```
Initialise $k = 0$, $P_{\mathbf{x}_k}^{(g)} = P_{\mathbf{x}_0}$
for Discretisation intervals do
    Propagate uncertainty region from $t_k$ to $t_{k+1}$ through intrusive
    polynomial algebra
    $P_{\mathbf{x}_k}^{(g)} \rightarrow P_{\mathbf{x}_{k+1}}^{(p)}$
    Compute propagated lower and upper ranges of $P_{\mathbf{x}_{k+1}}^{(p)}$
    $\mathbf{XL}_{k+1}$, $\mathbf{XU}_{k+1}$
    Reinitialise uncertainty region as hyper-box with computed range
    $\mathbf{XL}_{k+1}$, $\mathbf{XU}_{k+1} \rightarrow P_{\mathbf{x}_{k+1}}^{(g)}$
    Update quantities for loop iteration
    $k = k + 1$
end for
```

By outer bounding the propagated region, the key advantage is that all the possible

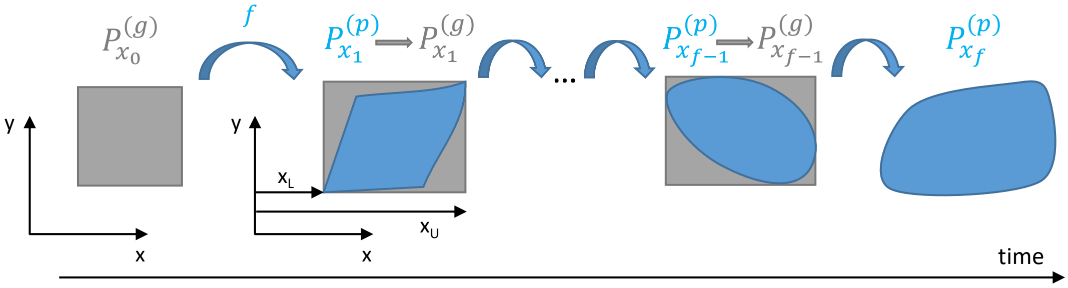


Figure 2.2: Graphical sketch of intrusive polynomial propagation approach. The blue regions depict the propagated polynomials, whereas the grey boxes represent the reinitalisation hyper-boxes.

state realisations are included, thus granting the continuity of any pointwise trajectory realisation. This simple approach comes at the expense of propagating larger regions than strictly needed.

It is worth noting that the state uncertainty region can be propagated at any intermediate time of interest $t \in (t_k, t_{k+1}]$ without the need for further discretisation and reinitialisation. Trivially, the coefficients of an intermediate polynomial can be saved during the propagation phase. Therefore, the result of this propagation approach is a chain of polynomial surrogates describing the uncertain state $\mathbf{X}_t$ as a composite function of the initial conditions $\mathbf{X}_0$ and the uncertain parameters $\mathbf{D}_{0:k} = [\mathbf{D}_0, \ldots, \mathbf{D}_k]$ which affected the system evolution up to time $t_k < t \leq t_{k+1}$. However, at this step, the hyper-box reinitialisation caused the uncertain state polynomial representation to be an over-estimation of the true uncertainty space.

An estimation of the correct size of the actual propagated set F_t is recovered by sampling the polynomial representation. The approach to propagate samples and recover the actual state uncertainty region is described in Algorithm 2, and graphically represented in Fig. 2.3.

The approach starts by drawing N state samples $\mathbf{x}_0^{(j)}$ from the initial uncertainty state space and N parameter samples $\mathbf{d}_0^{(j)}$ from the first segment uncertainty parameter space. Each of the N pairs $(\mathbf{x}_0^{(j)}, \mathbf{d}_0^{(j)})$ corresponds to a deterministic realisation of the trajectory (black dots and lines in Fig. 2.3), which is propagated using the polynomial mapping $P^{(p)}_{\mathbf{x}_1}$ computed as in Algorithm 1. The propagated state $\mathbf{x}_1^{(j)}$ is scaled within

Algorithm 2 Samples Propagation Phase of Reinitialisation Approach

```
Sample $N$-times the initial uncertainty space
    $\mathbf{x}_0^{(j)} \in \mathbf{\Omega}_{\mathbf{X}_0}$
Initialise $k = 0$
for Discretisation intervals do
    Sample $N$-times the $k-$uncertain parameter space:
    $\mathbf{d}_k^{(j)} \in \mathbf{\Omega}_{\mathbf{d}_k}$
    Propagate each uncertain sample from $t_k$ to $t_{k+1}$ with polynomial mapping
    $\mathbf{x}_{k+1}^{(j)} = P_{\mathbf{x}_{k+1}}^{(p)}(\mathbf{x}_k^{(j)}, \mathbf{d}_k^{(j)})$
    Scale propagated samples within the polynomial input domain using
    ranges $\mathbf{XL}_{k+1}$ and $\mathbf{XU}_{k+1}$
    Update quantities for loop iteration
    $k = k + 1$
end for
```

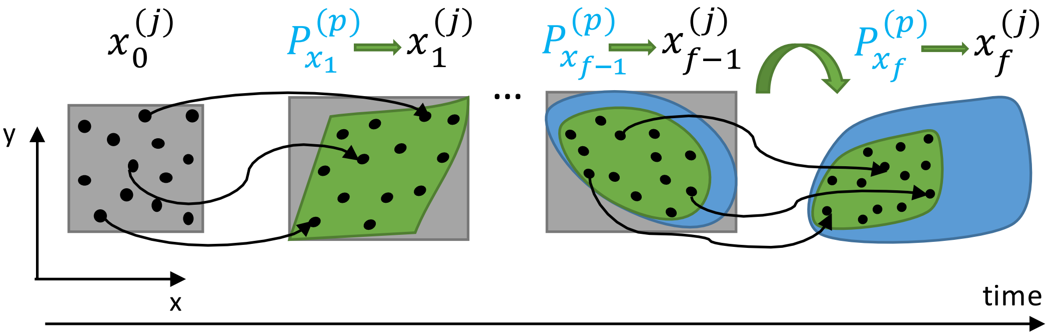


Figure 2.3: Graphical sketch of the recovery approach by sampling. The grey boxes represent the reinitialisation hyper-boxes, the blue regions depict the propagated polynomials, while the green areas symbolise the true uncertainty regions reconstructed by the black samples.

the polynomial input domain, and new parameter samples $\mathbf{d}_1^{(j)}$ are drawn for the successive segment to form N new pairs $(\mathbf{x}_1^{(j)}, \mathbf{d}_1^{(j)})$. The propagation and sampling steps are iterated for each segment. Thanks to this approach, the true uncertainty region (green in Fig. 2.3) is recovered from the conservative polynomial expansion (grey in Fig. 2.3), and represented by a cloud of discrete samples. These samples can be used to numerically compute moments or expectations with respect to the input pdfs as shown in the next section.

2.4 Expectation estimators

In this section, practical sample-based techniques will be presented to compute the expectation $\mathbb{E}\left[\psi\big(H(\boldsymbol{\xi})\big)\right]$. Once these methods are available, the propagated distribution can be approximated in a number of ways. One alternative is to approximate the continuous propagated pdf using a discrete distribution using the drawn samples as atoms (see Section 2.4.1). Another alternative is to approximate the propagated pdf by its first few moments. Indeed, the moments are defined as expectations for specific auxiliary function ψ. Thus, once a method is available to compute generic expectations efficiently, the moments of the propagated pdf can be derived, and the distribution can be approximated by retaining only a few moments (see Section 2.4.2). These two approaches are the ones used in the methods for state estimation, optimal control, and tracking throughout this thesis.

2.4.1 Monte Carlo methods

A large family of techniques to approximate expectations with respect to the propagated density is Monte Carlo methods. Among them, direct Monte Carlo aims at solving the expectation in Eq. (2.16) by directly taking samples from the known input distribution. Then, the expectation is approximated as

$$\mathbb{E}\big[\psi(H(\boldsymbol{\xi}))\big] = \int \psi(H(\boldsymbol{\xi}))p(\boldsymbol{\xi})d\boldsymbol{\xi} \approx \frac{1}{N}\sum_{j} \psi\big(H(\boldsymbol{\xi}^{(j)})\big) \ , \tag{2.25}$$

with $\boldsymbol{\xi}^{(j)} \sim p(\boldsymbol{\xi})$.

Therefore, this method does not require any linearity or Gaussian assumption on the model. Furthermore, unlike deterministic methods, the number of samples required for the mean to converge is theoretically independent of the problem's dimensionality [126].

In this vanilla Monte Carlo, it is evident to infer how crucial it is to properly select the random samples in accordance with the original probability distribution. This is numerically straightforward when $p(\boldsymbol{\xi})$ belongs to any simple distribution family.

Thus, this method is optimal when there is a direct, simple map between the input distribution and the quantity of interest. However, in sequential state estimation or optimal control applications, when the propagated distribution becomes the prior for the next time segment, such a direct map may not be tractable.

Hence, another special class of Monte Carlo methods is importance sampling, which draws samples from an approximated density $\pi(\boldsymbol{\xi})$, simpler to sample from, instead of the original $p(\boldsymbol{\xi})$

$$\mathbb{E}_p\left[\psi(H(\boldsymbol{\xi}))\right] = \int \psi(H(\boldsymbol{\xi}))p(\boldsymbol{\xi})d\boldsymbol{\xi} = \int \psi(H(\boldsymbol{\xi}))\frac{p(\boldsymbol{\xi})}{\pi(\boldsymbol{\xi})}\pi(\boldsymbol{\xi})d\boldsymbol{\xi} = \mathbb{E}_\pi\left[\psi(H(\boldsymbol{\xi}))\frac{p(\boldsymbol{\xi})}{\pi(\boldsymbol{\xi})}\right] . \tag{2.26}$$

The requirement on the importance density is that its support should be greater or equal to the one of $p(\boldsymbol{\xi})$ [127]. Thus, the expectation can be computed again as a weighted sum by weighting each sample with a measure of the deviation between the original and sampled distribution

$$\mathbb{E}\left[\psi(H(\boldsymbol{\xi}))\right] \approx \sum_j w^{(j)}\psi\big(H(\boldsymbol{\xi}^{(j)})\big) , \tag{2.27}$$

where

$$w^{(j)} = \frac{1}{N}\frac{p(\boldsymbol{\xi}^{(j)})}{\pi(\boldsymbol{\xi}^{(j)})} \tag{2.28}$$

and $\boldsymbol{\xi}^{(j)} \sim \pi(\boldsymbol{\xi})$.

Intuitively, the weights correct the bias associated to the samples selected from a non-ideal distribution. Clearly, the closer the importance distribution is to the original one, the smaller the required bias correction is, i.e $w^{(j)} \approx 1/N$. The same weights computed for the expected value can be used for the covariance, or higher-order moment approximation. Liu [127] suggests to use the normalised weights:

$$\hat{w}^{(j)} = \frac{w^{(j)}}{\sum_j w^{(j)}} . \tag{2.29}$$

The resulting estimate, although biased, often results in a smaller mean squared error. The same weight choice is found in Sarkka [30] when deriving the importance sampling form for conditional probabilities.

Indeed, this framework applies also to the density functions in the sequential filtering algorithms, where conditional distributions substitute $p(\boldsymbol{\xi})$ and $\pi(\boldsymbol{\xi})$. The specific form importance sampling assumes for sequential filtering formulations will be discussed in the relevant sections of Chapter 3.

Importance sampling should be preferred when the probability distribution is complex to sample but rather easy to evaluate, or when the expectation has to be evaluated for a set of possible density functions $p \in \mathcal{P}$, as in the case of epistemic uncertainty (2.2). Indeed, in the latter case, the most expensive evaluations $\psi\big(H(\boldsymbol{\xi}^{(j)})\big)$ (which typically require numerical propagations) should be performed only once per sample. Then, the expectation corresponding to different distributions is computed by using different weights $w^{(j)}$, which are inexpensive evaluations of the specific density function.

Formally, in Monte Carlo methods, the propagated pdf is approximated as a discrete distribution as [30]

$$p(\mathbf{z}) = \sum_{j=1}^{N} w^{(j)}\, \delta\big(\mathbf{z} - \mathbf{z}^{(j)}\big) \ , \tag{2.30}$$

where δ is the Dirac function.

2.4.2 Gaussian cubature

Powerful methods exist to solve integrals in the expectation form when the density function is Gaussian

$$\mathbb{E}\big[\psi\big(H(\boldsymbol{\xi})\big)\big] = \int \psi\big(H(\boldsymbol{\xi})\big)\mathcal{N}\big(\boldsymbol{\xi}; \boldsymbol{\mu}_\xi, \boldsymbol{\Sigma}_\xi\big) d\boldsymbol{\xi} \ , \tag{2.31}$$

where $\boldsymbol{\mu}_\xi$ and $\boldsymbol{\Sigma}_\xi$ are the mean and covariance of the normal distribution, respectively.

The strategy is again to approximate the integral as a weighted sum

$$\mathbb{E}\big[\psi\big(H(\boldsymbol{\xi})\big)\big] \approx \sum_{j=1}^{N} w^{(j)} \psi\big(H(\boldsymbol{\xi}^{(j)})\big) \ , \tag{2.32}$$

but, now, the weights and samples are not sampled from the pdf but constructed deterministically. The weights, positive or negative, should sum to one to have an unbiased estimate.

The Unscented Transform (UT) builds the sigma points by fitting a discrete distribution to the input density. The most common variant relies on the use of $2n_\xi + 1$ sigma points [128]. In this case, the UT can approximate a Gaussian distribution up to the third-order, while errors appear as a result of fourth-order cross-kurtoses terms. In the derivation by Wan and Van Der Merwe, the points and weights are selected around the mean value and scaled according to the covariance [129].

On the other hand, in Gauss-Hermite cubature, the weights and samples are chosen to integrate polynomial integrands exactly. Thus, $\boldsymbol{\xi}^{(j)}$ are chosen to be the roots of multivariate Hermite polynomials of order q, which exactly integrate polynomial integrands up to order $2q - 1$ [130]. The multivariate Gauss-Hermite grid is usually constructed by Cartesian products of univariate ones, therefore the number of samples increases exponentially with the uncertainty dimension. Sparse Gauss-Hermite rules are available to achieve only a polynomial increase [124, 131] and fight the curse of dimensionality encountered in high-dimensional problems. Again, the grid level is a tunable parameter connected to the polynomials' order which determines the number of collocation points in each dimension, and samples are different for different levels. Therefore, Gauss-Hermite cubature enables adaptive approximations of the estimator as the grid level can be sequentially increased until a convergence metric is met.

Once the set of deterministic samples has been selected by either technique, the first two moments of the propagated distribution $p(\mathbf{z})$ are reconstructed as [132, 130]:

$$\boldsymbol{\mu}_z \approx \sum_{j=1}^{N} w^{(j)} \, H(\boldsymbol{\xi}^{(j)}) \tag{2.33}$$

$$\boldsymbol{\Sigma}_z \approx \sum_{j=1}^{N} w^{(j)} \left(H(\boldsymbol{\xi}^{(j)}) - \boldsymbol{\mu}_z\right)\left(H(\boldsymbol{\xi}^{(j)}) - \boldsymbol{\mu}_z\right)^T . \tag{2.34}$$

Higher-order moments can be computed as well by plugging the relevant ψ in Eq. (2.32).

The main advantage of such techniques over statistical or Taylor linearisation, e.g. as for the extended Kalman Filter, is that they do not require differentiability or derivative information of the nonlinear mappings ψ and H.

2.5 Chapter summary

This chapter introduced the methods for efficient UP in dynamical systems that are used in this thesis. A two-step approach was employed: first, a polynomial representation of the dynamics was constructed; then, the propagated distribution or some summary statistics was computed by exploiting the polynomial representation.

Two approaches were presented for constructing the polynomial, one intrusive based on generalised polynomial algebra, the other non-intrusive using stochastic collocation. For both, a reinitialisation strategy was presented to keep the uncertain dimensionality contained in multi-segment or multi-phase problems.

Finally, practical approaches for computing the propagated distribution or its moments were presented, differentiating between Monte Carlo methods for generic distributions and quadrature approaches tailored for Gaussian densities, which will be extensively employed in the applications' part.

Chapter 3

Robust State Estimation

The content of this chapter was published in:
C. Greco, and M. Vasile, "Robust Bayesian Particle Filter for Space Object Tracking Under Severe Uncertainty", *Journal of Guidance, Control, and Dynamics*, under review [84];
C. Greco, and M. Vasile, "Closing the loop between mission design and navigation analysis", *International Astronautical Congress, The Cyberspace Edition*, 2020 [93].

This chapter presents two novel methods for state estimation of dynamical systems under an imprecise specification of the probability distributions modelling the problem uncertainty [101]. The proposed state estimation formulation encompasses both mixed aleatory and epistemic models for the uncertain components. The prior, likelihood, and transition distributions are set-valued rather than precisely specified. Specifically, set-valued hyperparameters are employed to parameterise the epistemic sets, making this model more general than the closed convex sets found in most methods in literature. Bayes inference is used to compute a posterior distribution leading to an infinite number of posteriors compatible with this set-valued specification. This epistemic filtering scheme can also be used to study how the filter output is affected by changes in input parameters of predefined families of distributions.

The chapter is structured as follows. Section 3.1 introduces the robust filtering formulation under epistemic uncertainty. Section 3.2 presents the robust particle filtering approach to solve the filtering problem under epistemic uncertainty. Section 3.3

describes the second approach, which employs variational inference and epistemic uncertainty reinitialisation. Finally, Section 3.4 concludes the chapter with a short recap of the introduced methods.

3.1 Filtering formulation

The state estimation problem addressed in this thesis is a generalisation of the mixed continuous-discrete one [133]: the system state evolves according to a time-continuous differential equation, whereas indirect noisy observations are collected at discrete instants of time.

3.1.1 Precise filtering

Again, let $\mathbf{x} \in \mathbb{R}^x$ be the state of a dynamical system evolving according to a nonlinear differential equation (2.17). If the system initial condition $\mathbf{x}(t_0) = \mathbf{x}_0$ and the dynamical model parameters $\mathbf{d}$ were known perfectly, there would be no need for measurements, as the equations of motion could be (usually numerically) integrated to obtain the system evolution in time. However, in real-life scenarios, uncertainty is always involved in such systems, and measurements are needed to refine the state knowledge at a later time. Thus, let us consider uncertainty on the initial conditions, model parameters, and on the state at a generic time, as introduced in Section 2.3.

Hence, the system is observed through indirect noisy measurements. The measurements $\mathbf{y}_k \in \mathbb{R}^y$, taken at time instances t_k for $k = 1, \ldots, M$ with $t_k < t_{k+1}$, is a known realisation of a random variable $\mathbf{Y}_k$ governed by a stochastic measurement model

$$\mathbf{y}_k = h(t_k, \mathbf{x}_k, \boldsymbol{\varepsilon}_k), \tag{3.1}$$

where $\boldsymbol{\varepsilon}_k$ is a stochastic term modelling sensor noise.

Hence, the standard continuous-discrete state-space is formulated in a probabilistic fashion [30] to explicitly describe the uncertain nature of the system in terms of probability density functions p as

$$\begin{cases} \mathbf{X}_0 \sim p(\mathbf{x}_0) & \text{(3.2a)} \\ \mathbf{X}_k \sim p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}) & \text{(3.2b)} \\ \mathbf{Y}_k \sim p(\mathbf{y}_k \,|\, \mathbf{x}_k) \qquad \text{for} \quad k = 1, \dots, M \ , & \text{(3.2c)} \end{cases}$$

where $p(\mathbf{x}_0)$ is the initial density function, $p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1})$ is the transition density function and $p(\mathbf{y}_k \,|\, \mathbf{x}_k)$ is the observation likelihood. The transition probability describes the system dynamics as a Markov chain as resulting from the uncertainty on the model parameters. The likelihood function models the probabilistic state-observation spaces relationship governed by sensor noise.

Given a precise specification of the densities in Equation (3.2), the complete solution of the filtering problem of general state estimation is the posterior distribution of the state conditional to the previously received observations

$$p(\mathbf{x}_k \,|\, \mathbf{y}_{1:k}) \, . \tag{3.3}$$

Generally, this posterior distribution is arduous to compute, and its complete knowledge is not required, as already discussed in Chapter 2. Therefore, state estimation and filtering algorithms primarily focus on the computation of the expectation of a generic function $\psi : \mathbb{R}^x \to \mathbb{R}$ of the uncertain state given the received measurements as

$$\mathbb{E}_p\big[\psi(\mathbf{x}_k) \,|\, \mathbf{y}_{1:k}\big] = \int \psi(\mathbf{x}_k) \, p(\mathbf{x}_k \,|\, \mathbf{y}_{1:k}) \, d\mathbf{x}_k \ , \tag{3.4}$$

with respect to the posterior distribution.

3.1.2 Imprecise filtering

In the case of epistemic uncertainty, the probability distributions cannot be assumed to be known precisely, but they are only specified within an imprecise set $\mathcal{P}$. Within these sets of distributions, no judgement is made about their relative likeliness. This definition allows one to model more faithfully uncertain scenarios in which information is too scarce to specify a single distribution. In this thesis, we focus on parametric distributions with epistemic uncertainty on their parameter $\boldsymbol{\lambda} \in \mathbb{R}^{\lambda}$ as described in Section 2.1. The use of families of distributions captures the epistemic component in the knowledge of the exact probability governing the realisation of an event. In the filtering context, epistemic uncertainty can be associated with the poor characterisation of physical, e.g. celestial body's physical characteristics, or system parameters, e.g. sensor noise or engine performance. When information is sufficient to identify a single distribution, uncertainty is purely aleatory and $\boldsymbol{\Omega}_{\boldsymbol{\lambda}}$ degenerate to a singleton. Therefore, this model is a generalisation of the pure aleatoric model.

In this generalised scenario, the probabilistic continuous-discrete estimation problem can be written to account for the new epistemic component as

$$\begin{cases} \mathbf{X}_0 \sim p(\mathbf{x}_0; \boldsymbol{\lambda}_0) \qquad \in \mathcal{P}_{x_0} & \text{(3.5a)} \\ \mathbf{X}_k \sim p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}; \boldsymbol{\lambda}_x) \in \mathcal{P}_{x_k|x_{k-1}} & \text{(3.5b)} \\ \mathbf{Y}_k \sim p(\mathbf{y}_k \,|\, \mathbf{x}_k; \boldsymbol{\lambda}_y) \quad \in \mathcal{P}_{y_k|x_k} \quad \text{for } k = 1, \dots, M, & \text{(3.5c)} \end{cases}$$

where $\boldsymbol{\lambda}_0 \in \boldsymbol{\Omega}_{\boldsymbol{\lambda}_0}$, $\boldsymbol{\lambda}_x \in \boldsymbol{\Omega}_{\boldsymbol{\lambda}_x}$ and $\boldsymbol{\lambda}_y \in \boldsymbol{\Omega}_{\boldsymbol{\lambda}_y}$ are the epistemic parameters for the initial, transition and likelihood distributions, respectively. Hereafter, $\boldsymbol{\lambda} = [\boldsymbol{\lambda}_0, \boldsymbol{\lambda}_x, \boldsymbol{\lambda}_y]$ is used to indicate the collection of the three epistemic parameters, $\boldsymbol{\Omega}_{\boldsymbol{\lambda}}$ the collection of their respective sets, and $p_{\boldsymbol{\lambda}} = \left\{p(\mathbf{x}_0; \boldsymbol{\lambda}_0), p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}; \boldsymbol{\lambda}_x), p(\mathbf{y}_k \,|\, \mathbf{x}_k; \boldsymbol{\lambda}_y)\right\}$ the collection of the corresponding densities.

Such uncertainty specification induces an infinite number of posterior distributions

$$p(\mathbf{x}_k \,|\, \mathbf{y}_{1:k}; \boldsymbol{\lambda}) \tag{3.6}$$

parameterised in $\boldsymbol{\lambda}$, which represent the complete solution of the robust estimation process. In the epistemic setting, the expectation of interest has a parametric dependency on $\boldsymbol{\lambda}$

$$\mathbb{E}_{p_{\boldsymbol{\lambda}}}\left[\psi(\mathbf{x}_k) \,|\, \mathbf{y}_{1:k}\right] = \int \psi(\mathbf{x}_k)\, p(\mathbf{x}_k \,|\, \mathbf{y}_{1:k}; \boldsymbol{\lambda})\, d\mathbf{x}_k \ . \tag{3.7}$$

Being the set of posterior distributions infinite in general, there are infinite values of such expectation for varying $\boldsymbol{\lambda}$.

Hence, the outcome of the presented robust estimation process is an interval whose bounds are the lower and upper values of the expectation of the quantity of interest conditional on the observations. The lower and upper expectations are defined as

$$\underline{\mathbb{E}}\big[\psi(\mathbf{x}_k)\,|\,\mathbf{y}_{1:k}\big] = \min_{\boldsymbol{\lambda}\in\boldsymbol{\Omega}_{\boldsymbol{\lambda}}} \mathbb{E}_{p_{\boldsymbol{\lambda}}}\big[\psi(\mathbf{x}_k)\,|\,\mathbf{y}_{1:k}\big] \tag{3.8a}$$

$$\overline{\mathbb{E}}\big[\psi(\mathbf{x}_k)\,|\,\mathbf{y}_{1:k}\big] = \max_{\boldsymbol{\lambda}\in\boldsymbol{\Omega}_{\boldsymbol{\lambda}}} \mathbb{E}_{p_{\boldsymbol{\lambda}}}\big[\psi(\mathbf{x}_k)\,|\,\mathbf{y}_{1:k}\big], \tag{3.8b}$$

that is the minimum and maximum of the expectation computed with respect to the posterior $p(\mathbf{x}_k\,|\,\mathbf{y}_{1:k};\boldsymbol{\lambda})$. If the imprecise sets were precise, i.e. $\boldsymbol{\Omega}_{\boldsymbol{\lambda}}$ was a singleton, Equations (3.8a) and (3.8b) would coincide. We will refer to the distributions with epistemic parameters corresponding to the argument of the minimum and maximum in Eq. (3.8) as lower and upper distributions, respectively.

A direct optimisation could compute these bounds over the parameterised imprecise sets where each objective function evaluation requires an independent precise filtering run for each candidate epistemic parameter $\boldsymbol{\lambda}$ to evaluate. In this case, however, the optimisation would be inefficient because the expectation $\mathbb{E}_{p_{\boldsymbol{\lambda}}}$ would be used as a black-box cost function. Therefore, the computations performed during the filter call run would be discarded each time. Furthermore, if bounds on the expectation should be updated as new observations arrive, the optimisation would need to be rerun completely.

3.2 Robust particle filter

Hence, in this section, we propose the Robust Particle Filter (RPF) to efficiently compute bounds on the expectation of a given quantity of interest by exploiting a precomputation of the particles. Given the sequential nature of this filter, when a new observation is available, the bounds can be updated by still exploiting the precomputations performed at the previous step.

The overall RPF algorithm to compute robust bounds under epistemic uncertainty is thus schematised in Algorithm 3. Line 1 is the precomputation step performed using a standard particle filter. Line 2 is the bound computation routine performed by a Branch & Bound (B&B) optimisation of the expectation estimator $\hat{\theta}$. These two fundamental steps will be discussed in greater detail in the next two sections.

Algorithm 3 Algorithmic scheme for the Robust Particle Filter.

```
Given:
    · the estimation problem (3.5)
    · the proposal distributions $\pi(\mathbf{x}_0)$ and $\pi(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}, \mathbf{y}_{1:k})$
Precomputation step by standard Particle Filter
Save collection of particles $\boldsymbol{\mathcal{X}}_{0:M}$, densities $\pi_{0:M}^{(i)}$, and function evaluations $\psi^{(i)}$.
Bound computation by Branch & Bound
    $\underline{\hat{\theta}}(\boldsymbol{\mathcal{X}}_{0:M}) = \min_{\boldsymbol{\lambda} \in \Omega_{\boldsymbol{\lambda}}} \hat{\theta}(\boldsymbol{\mathcal{X}}_{0:M}, \boldsymbol{\lambda})$
    $\overline{\hat{\theta}}(\boldsymbol{\mathcal{X}}_{0:M}) = \max_{\boldsymbol{\lambda} \in \Omega_{\boldsymbol{\lambda}}} \hat{\theta}(\boldsymbol{\mathcal{X}}_{0:M}, \boldsymbol{\lambda})$
```

3.2.1 Expectation estimator

If no specific assumption, or parameterisation, is imposed on the distributions $p_{\boldsymbol{\lambda}}$, the expectation has no closed-form solution, and numerical techniques are required. In this section, we employ a precomputed Sequential Importance Sampling (pSIS) to construct an estimator $\hat{\theta}(\mathbf{x}_k, \boldsymbol{\lambda})$ for the expectation $\mathbb{E}_{p_{\boldsymbol{\lambda}}}\big[\psi(\mathbf{x}_k) \,|\, \mathbf{y}_{1:k}\big]$.

Precomputed sequential importance sampling

One of the main difficulty in computing the sought expectation is that in most practical applications it is impractical to draw samples directly from the posterior distribution $p(\mathbf{x}_k \,|\, \mathbf{y}_{1:k}; \boldsymbol{\lambda})$, although it is possible to evaluate its density. To overcome this challenge, the integral in Eq. (3.7) is written by using a proposal distribution as

$$\begin{aligned} \mathbb{E}_{p_{\boldsymbol{\lambda}}}\big[\psi(\mathbf{x}_k) \,|\, \mathbf{y}_{1:k}\big] &= \int \psi(\mathbf{x}_k) \, \frac{p(\mathbf{x}_k \,|\, \mathbf{y}_{1:k}; \boldsymbol{\lambda})}{\pi(\mathbf{x}_k \,|\, \mathbf{y}_{1:k})} \, \pi(\mathbf{x}_k \,|\, \mathbf{y}_{1:k}) \, \mathrm{d}\mathbf{x}_k \\ &= \int \psi(\mathbf{x}_k) \, w(\mathbf{x}_k) \, \pi(\mathbf{x}_k \,|\, \mathbf{y}_{1:k}) \, \mathrm{d}\mathbf{x}_k, \end{aligned} \tag{3.9}$$

under the condition that π, the so-called *proposal* distribution, has a larger support than $p(\mathbf{x}_k \,|\, \mathbf{y}_{1:k}; \boldsymbol{\lambda})$. The function $w(\mathbf{x}_k)$ is the ratio between the target distribution and the proposal one functioning as a weight measuring their deviations. The proposal distribution should be chosen such that it is simple to draw samples from it.

Furthermore, to exploit the sequential nature of the tracking problem (3.5) the posterior in Eq. (3.6) can be decomposed as

$$\begin{aligned} p(\mathbf{x}_k \,|\, \mathbf{y}_{1:k}; \boldsymbol{\lambda}) &\propto p(\mathbf{y}_k \,|\, \mathbf{x}_k, \mathbf{y}_{1:k-1}; \boldsymbol{\lambda})\; p(\mathbf{x}_k \,|\, \mathbf{y}_{1:k-1}; \boldsymbol{\lambda}) \\ &= p(\mathbf{y}_k \,|\, \mathbf{x}_k; \boldsymbol{\lambda})\; p(\mathbf{x}_k \,|\, \mathbf{x}_{0:k-1}, \mathbf{y}_{1:k-1}; \boldsymbol{\lambda})\; p(\mathbf{x}_{0:k-1} \,|\, \mathbf{y}_{1:k-1}; \boldsymbol{\lambda}) \\ &= p(\mathbf{y}_k \,|\, \mathbf{x}_k; \boldsymbol{\lambda}_y)\; p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}; \boldsymbol{\lambda}_x)\; p(\mathbf{x}_{0:k-1} \,|\, \mathbf{y}_{1:k-1}; \boldsymbol{\lambda}), \end{aligned} \tag{3.10}$$

with initial condition $p(\mathbf{x}_{0:0} \,|\, \mathbf{y}_{1:0}; \boldsymbol{\lambda}) = p(\mathbf{x}_0; \boldsymbol{\lambda}_0)$. The first relationship stems from unnormalised Bayes' inference, where the normalisation constant $p(\mathbf{y}_k \,|\, \mathbf{y}_{1:k-1})$ is not computed as too expensive to evaluate. Neglecting the normalisation constant requires a self-normalisation step (see Eq. (3.13b)) that causes the final estimator to be only asymptotically unbiased [134]. The second one results from the conditional independence of the observation $\mathbf{y}_k$ given the state $\mathbf{x}_k$, and from writing the joint distribution $p(\mathbf{x}_k \,|\, \mathbf{y}_{1:k-1}; \boldsymbol{\lambda})$ in terms of the conditional one. The last relation comes from the Markov property of the dynamical system, that is, conditional independence on the previous history of $\mathbf{x}_k$ once $\mathbf{x}_{k-1}$ is given.

Also, the proposal distribution should be chosen as

$$\pi(\mathbf{x}_k \,|\, \mathbf{y}_{1:k}) = \pi(\mathbf{x}_k \,|\, \mathbf{x}_{0:k-1}, \mathbf{y}_{1:k})\; \pi(\mathbf{x}_{0:k-1} \,|\, \mathbf{y}_{1:k-1}), \tag{3.11}$$

with initial condition $\pi(\mathbf{x}_{0:0} \,|\, \mathbf{y}_{1:0}) = \pi(\mathbf{x}_0)$. If the initial distribution could be inexpensively sampled, the optimal choice is to set $\pi(\mathbf{x}_0) = p(\mathbf{x}_0)$. With a proposal in this form, the samples can be drawn sequentially, that is, when the samples $\boldsymbol{\mathcal{X}}_k = \{\mathbf{x}_k^{(1)}, \dots, \mathbf{x}_k^{(N)}\}$ at time t_k need to be drawn, the samples until t_{k-1} need not to be drawn again. The new collection of samples is formed as $\boldsymbol{\mathcal{X}}_{0:k} = \{\boldsymbol{\mathcal{X}}_{0:k-1}, \boldsymbol{\mathcal{X}}_k\}$.

Hence, the estimator is

$$\hat{\theta}(\boldsymbol{\mathcal{X}}_{0:k}; \boldsymbol{\lambda}) = \sum_{i=1}^{N} \hat{w}_k^{(i)}(\boldsymbol{\lambda})\, \psi(\mathbf{x}_k^{(i)}), \tag{3.12}$$

where samples $\mathbf{x}_k^{(i)}$ are drawn from $\pi(\mathbf{x}_k \,|\, \mathbf{x}_{0:k-1}, \mathbf{y}_{1:k})$ and $\mathbf{x}_k = \{\mathbf{x}_{0:k-1}, \mathbf{x}_k\}$ thanks to the sequential choice of the proposal (3.11), and $w_k^{(i)}(\boldsymbol{\lambda})$ are the *importance* weights, computed in a sequential fashion as

$$w_k^{(i)}(\boldsymbol{\lambda}) = \frac{p(\mathbf{y}_k \,|\, \mathbf{x}_k^{(i)}; \boldsymbol{\lambda}_y)\; p(\mathbf{x}_k^{(i)} \,|\, \mathbf{x}_{k-1}^{(i)}; \boldsymbol{\lambda}_x)}{\pi(\mathbf{x}_k^{(i)} \,|\, \mathbf{x}_{0:k-1}^{(i)}, \mathbf{y}_{1:k})}\; \hat{w}_{k-1}^{(i)}(\boldsymbol{\lambda}) \tag{3.13a}$$

$$\hat{w}_k^{(i)}(\boldsymbol{\lambda}) = \frac{w_k^{(i)}(\boldsymbol{\lambda})}{\sum_{j=1}^{N} w_k^{(j)}(\boldsymbol{\lambda})}, \tag{3.13b}$$

with $\hat{w}_0^{(i)}(\boldsymbol{\lambda}_0) = p(\mathbf{x}_0^{(i)}; \boldsymbol{\lambda}_0)/\pi(\mathbf{x}_0^{(i)})/N$. The weight update rule in Eq. (3.13a) does not include the posterior normalisation constant, as it has been neglected in Eq. (3.10), therefore Eq. (3.13b) is the weight self-normalisation which ensure the estimator consistency, that is, it converges in probability to the true value with increasing sample size [135].

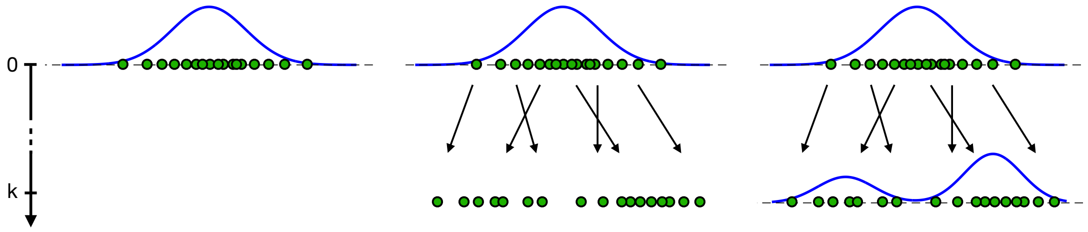
The proposal distribution may not be representative for all the distributions in the posterior epistemic set. Therefore, the estimate may be significantly biased when using the same samples for different epistemic parameters $\boldsymbol{\lambda}$. We use an a posteriori check on the figure of the effective sample size to ensure that the estimator yields a satisfactory accuracy in the computation of the expectation. This will be discussed in detail in a later section concerning the proposal selection.

Such estimator brings an advantageous decoupling between the weights, dependent on the epistemic parameter, and the particle sampling, independent from the epistemic parameter. Therefore, once a set of particles has been drawn from a fixed proposal and propagated, the optimisations in Eq. (3.8) can be carried out by optimising the importance weights $w_k^{(i)}(\boldsymbol{\lambda})$. Hence, all the quantities which are independent of the epistemic parameter are precomputed before the optimisation. Indeed, once the proposal distributions have been chosen, the particles sampling, the proposal density evaluations and the function ψ evaluations can be computed offline. The precomputations are outlined in Algorithm 4. A graphical representation is sketched in Fig. 3.1. First, the initial samples are drawn from the initial proposal distribution and the initial densities computed by evaluating the proposal (Line 1 of Algorithm 4 and Fig. 3.1(a)). Then, for each of the M observations, the particles are drawn from the transition proposal and the densities evaluated for such samples (Line 2 and Fig. 3.1(b)). The particles rep-

Algorithm 4 Precomputation for sequential importance sampling estimator.

```
   Given:
      · the estimation problem (3.5)
      · the proposal distributions $\pi(\mathbf{x}_0)$ and $\pi(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}, \mathbf{y}_{1:k})$
1: Draw $N$ particles from the initial distribution $\pi(\mathbf{x}_0)$ and evaluate density
   $\mathbf{x}_0^{(i)} \sim \pi(\mathbf{x}_0)\ ,\ \pi_0^{(i)} = \pi(\mathbf{x}_0^{(i)})$
   for $k = 1 : M$ do
2:    Draw N particles from the proposal transition and evaluate density
      $\mathbf{x}_k^{(i)} \sim \pi(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}^{(i)}, \mathbf{y}_{1:k})\ ,\ \pi_k^{(i)} = \pi(\mathbf{x}_k^{(i)} \,|\, \mathbf{x}_{k-1}^{(i)}, \mathbf{y}_{1:k})$
   end for
3: Evaluate function $\psi$ for each particle path realisation
   $\psi^{(i)} = \psi(\mathbf{x}_{0:M}^{(i)})$
   Save collection of particles $\boldsymbol{\mathcal{X}}_{0:M}$, densities $\pi_{0:M}^{(i)}$, and function evaluations $\psi^{(i)}$.
```

resent a discrete approximation of the propagated distribution (Fig. 3.1(c)). Finally, once the whole collection of particles $\boldsymbol{\mathcal{X}}_{0:M}$ has been generated, the function of interest ψ is evaluated, and the resulting values saved (Line 3).



(a) (b) (c)

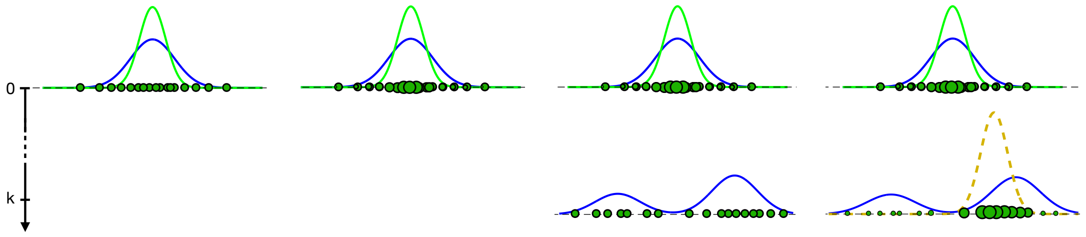
Figure 3.1: Representation of precomputation step of sequential importance sampling estimator: a) initial samples, b) propagated samples, c) propagated distribution approximation.

Once this precomputation step has been performed, the pSIS estimator is evaluated for an epistemic parameter $\boldsymbol{\lambda}$ as described in Algorithm 5. Its graphical representation is shown in Fig. 3.2. The pSIS is evaluated by computing the new weights $\hat{w}_k^{(i)}(\boldsymbol{\lambda})$ for a given $\boldsymbol{\lambda}$. By starting from the precomputed initial samples (Fig. 3.2(a)), the initial distribution with epistemic parameter $\boldsymbol{\lambda}_0$ is targeted by adjusting the weights (Line 1 of Algorithm 5 and Fig. 3.2(b)) as the ratio between the distribution (in green) and the

Algorithm 5 Algorithmic scheme for precomputed sequential importance sampling estimator.

```
   Given:
       · the estimation problem (3.5)
       · the precomputed quantities $\boldsymbol{\mathcal{X}}_{0:M}$, $\pi_{0:M}^{(i)}$ and $\psi^{(i)}$
       · epistemic parameter $\boldsymbol{\lambda} = [\boldsymbol{\lambda}_0, \boldsymbol{\lambda}_x, \boldsymbol{\lambda}_y]$ to evaluate
1: Set the weights of the $N$ precomputed initial samples $\boldsymbol{\mathcal{X}}_0$ and normalise them
   $w_0^{(i)} = p(\mathbf{x}_0^{(i)}; \boldsymbol{\lambda}_0)/\pi_0^{(i)}/N$
   for $k = 1 : M$ do
2:     Update the weights of the precomputed samples $\boldsymbol{\mathcal{X}}_k$ and self-normalise them
       $w_k^{(i)} = \hat{w}_{k-1}^{(i)}\, p(\mathbf{y}_k \,|\, \mathbf{x}_k^{(i)}; \boldsymbol{\lambda}_y)\; p(\mathbf{x}_k^{(i)} \,|\, \mathbf{x}_{k-1}^{(i)}; \boldsymbol{\lambda}_x)/\pi_k^{(i)}$ ,   $\hat{w}_k^{(i)} = w_k^{(i)} / \sum_{j=1}^{N} w_k^{(j)}$
   end for
3: Evaluate the estimator with the updated weights $\hat{w}_k^{(i)}$
   $\hat{\theta}(\boldsymbol{\mathcal{X}}_{0:M}) = \sum_{i=1}^{N} \hat{w}_M^{(i)}\, \psi^{(i)}$
```

initial proposal (in blue) normalised by the number of particles, such that Eq. (3.12) is coherent also for $k = 0$. For each observation instance, the precomputed samples are used to save new propagations (Fig. 3.2(c)). Their weights are updated (Line 2 and Fig. 3.2(d)) according to Eq. (3.13) with the new transition and likelihood (represented in dashed dark yellow line). Finally, the estimator is computed as a weighted sum of the updated weights and the precomputed function evaluations (Line 3).



(a) (b) (c) (d)

Figure 3.2: Representation of precomputed sequential importance sampling estimator: a) initial samples, b) adjusted weights, c) precomputed samples at next time step, d) updated weights after conditioning with the likelihood (dashed curve).

Estimator derivatives

The derivatives of the pSIS estimator can be computed analytically. The derivative knowledge is precious in the bound computation (3.8) as it enables the use of efficient gradient descent methods to improve the exploitation stage in the optimisation. Furthermore, because the precomputed SIS works on fixed samples $\boldsymbol{\mathcal{X}}_{0:k}$, there would be no noise due to sampling in the derivative information.

Let us assume that we can compute the derivatives of the density functions in Equation (3.5), that is, we can evaluate

$$\nabla_{\boldsymbol{\lambda}_0}\, p(\mathbf{x}_0; \boldsymbol{\lambda}_0) \tag{3.14a}$$

$$\nabla_{\boldsymbol{\lambda}_x}\, p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}; \boldsymbol{\lambda}_x) \tag{3.14b}$$

$$\nabla_{\boldsymbol{\lambda}_y}\, p(\mathbf{y}_k \,|\, \mathbf{x}_k; \boldsymbol{\lambda}_y). \tag{3.14c}$$

The quantity to compute is the gradient of the estimator with respect to the epistemic parameters

$$\nabla_{\boldsymbol{\lambda}}\, \hat{\theta} = \begin{bmatrix} \nabla_{\boldsymbol{\lambda}_0}\, \hat{\theta} \\ \nabla_{\boldsymbol{\lambda}_x}\, \hat{\theta} \\ \nabla_{\boldsymbol{\lambda}_y}\, \hat{\theta} \end{bmatrix}. \tag{3.15}$$

By the linearity of the derivative operator and the chain rule, the gradient can be computed as

$$\nabla_{\boldsymbol{\lambda}}\, \hat{\theta} = \sum_{i=1}^{N} \psi^{(i)}\, \nabla_{\boldsymbol{\lambda}} \hat{w}_M^{(i)}, \tag{3.16}$$

since the precomputed function evaluations $\psi^{(i)} = \psi(\mathbf{x}_k^{(i)})$ are independent from $\boldsymbol{\lambda}$.

Hence, the estimator gradient is obtained by computing the derivative of the weights. By Eq. (3.13) and chain rule again, the weight derivative $\nabla_{\boldsymbol{\lambda}}\, \hat{w}_k^{(i)}$ can be computed by the previous weight derivative $\nabla_{\boldsymbol{\lambda}}\, \hat{w}_{k-1}^{(i)}$ as

$$\begin{aligned}\nabla_{\boldsymbol{\lambda}}\, w_k^{(i)} &= \begin{bmatrix} 0 \\ p(\mathbf{y}_k \,|\, \mathbf{x}_k^{(i)}; \boldsymbol{\lambda}_y)\ \nabla_{\boldsymbol{\lambda}_x} p(\mathbf{x}_k^{(i)} \,|\, \mathbf{x}_{k-1}^{(i)}; \boldsymbol{\lambda}_x) \\ \nabla_{\boldsymbol{\lambda}_y} p(\mathbf{y}_k \,|\, \mathbf{x}_k^{(i)}; \boldsymbol{\lambda}_y)\ p(\mathbf{x}_k^{(i)} \,|\, \mathbf{x}_{k-1}^{(i)}; \boldsymbol{\lambda}_x) \end{bmatrix} \frac{\hat{w}_{k-1}^{(i)}}{\pi_k^{(i)}} + \\ &\qquad \frac{p(\mathbf{y}_k \,|\, \mathbf{x}_k^{(i)}; \boldsymbol{\lambda}_y)\ p(\mathbf{x}_k^{(i)} \,|\, \mathbf{x}_{k-1}^{(i)}; \boldsymbol{\lambda}_x)}{\pi_k^{(i)}}\ \nabla_{\boldsymbol{\lambda}}\, \hat{w}_{k-1}^{(i)} \qquad (3.17a)\\ \nabla_{\boldsymbol{\lambda}}\, \hat{w}_k^{(i)} &= \sum_{j=1}^{N} \frac{\partial w_k^{(i)}}{\partial w_k^{(j)}}\ \nabla_{\boldsymbol{\lambda}}\, w_k^{(j)}, \qquad (3.17b)\end{aligned}$$

with

$$\frac{\partial w_k^{(i)}}{\partial w_k^{(j)}} = -\frac{w_k^{(i)}}{\left(\sum_{k=1}^{N} w_k^{(k)}\right)^2} + \frac{\delta_{ij}}{\sum_{k=1}^{N} w_k^{(k)}} \tag{3.18}$$

where $\delta_{ij} = 1$ if $i = j$, $\delta_{ij} = 0$ otherwise. The initial conditions for the weights derivative computation are

$$\nabla_{\boldsymbol{\lambda}}\, w_0^{(i)} = \begin{bmatrix} \nabla_{\boldsymbol{\lambda}_0} p(\mathbf{x}_0^{(i)}; \boldsymbol{\lambda}_0)/\pi_0^{(i)} \\ 0 \\ 0 \end{bmatrix} \tag{3.19}$$

because the initial distribution does not depend on the epistemic parameters $\boldsymbol{\lambda}_x$ and $\boldsymbol{\lambda}_y$.

Algorithmic complexity

In this section, the algorithmic complexity of evaluating the estimator (3.12) is assessed. Specifically, the complexities of the precomputation step and the pSIS estimator are analysed separately to pinpoint the main saving of precomputation.

Let c_{x_0} be the complexity of drawing one particle from $\pi(\mathbf{x}_0)$, c_x the complexity to sample the proposal $\pi\big(\mathbf{x}_k \,|\, \mathbf{x}_{0:k-1}, \mathbf{y}_{1:k}\big)$, c_π the complexity of evaluating each term of the proposal (3.11) (assumed equal complexity), c_p the complexity of evaluating each of the distributions in problem (3.5) (assumed equal complexity), and c_ψ the complexity of evaluating the function ψ.

The complexity of the precomputation step can be assessed by looking at the complexity of each step in Algorithm 4:

1. $N\,(c_{x_0} + c_\pi)$ as N particles are drawn from and evaluated with the initial proposal;
2. $N\,(c_x + c_\pi)$ since N particles are drawn from and evaluated with the transition proposal;
3. Nc_ψ because there are N evaluations of the function ψ.

Step 2) is performed M times because of the for-loop over the number of observation instances. Therefore, the computational complexity of the precomputation step of the pSIS estimator is

$$C_{pre}\big(M,N\big) = N(c_{x_0} + c_\pi + c_\psi) + M\,N\,(c_x + c_\pi) = \mathcal{O}\big(N + MN\big), \tag{3.20}$$

which is linear both with the number of observations M and the number of particles N employed and displays an interaction between them.

Once the precomputations have been performed, the computational complexity of the pSIS estimator is assessed by studying each step in Algorithm 5:

1. $N\,(c_p + 2)$ because the importance weights are computed by evaluating the initial distribution and performing two divisions per weight;
2. $N\,(c_p + 3) + 2N - 1$ since the unnormalised weights are computed by evaluating two densities and performing three products or divisions, whereas the normalisation requires $N - 1$ sums and N divisions;
3. $2N - 1$ because the estimator is computed by N multiplications and $N - 1$ sums.

Step 2) is performed M times. Thus, the computational complexity of the pSIS is

$$C_{pSIS}\big(M,N\big) = N(c_p + 4) - 1 + M\,[N(2c_p + 5) - 1] = \mathcal{O}\big(N + M + MN\big), \tag{3.21}$$

which again is linear both with the number of observations M and in the number of particles N employed and showing the interaction between the two of them. In space tracking, the main saving stems from removing the term MNc_x, that is, sampling from the proposal distribution, which implies sparing MN new numerical integrations of the equations of motion every time the estimator is evaluated in the optimisation routine.

As for the estimator derivatives, let $c_{\partial p}$ be the computational complexity of evaluating one of the derivative in Eq. (3.14) (assumed equal complexity), and $n_\lambda = n_{\lambda_0} + n_{\lambda_x} + n_{\lambda_y}$ the total number of epistemic parameters. The computational complexity of evaluating the derivative of the pSIS estimator is

$$C_{\partial psis}(M, N) = \mathcal{O}(MNn_\lambda c_{\partial p} + MN^2) = \mathcal{O}(MN + MN^2), \tag{3.22}$$

where the quadratic term comes from Eq. (3.17b), and the quantities already computed in Algorithm 5 have not been recomputed, e.g. evaluating the densities and the computation of the weights. The terms $c_{\partial p}$ and n_λ have been retained in the first big O notation because their value influence significantly the algorithm complexity.

Thus, evaluating analytically the estimator derivatives requires an algorithmic complexity of $\mathcal{O}(MN^2 + MNn_\lambda c_{\partial p} + MNc_\pi)$. If the same quantities were to be computed by finite differences, the computational complexity would be $\mathcal{O}(MNn_\lambda c_\pi)$, that is evaluating the estimator n_λ times (or a small multiple of n_λ). Because of the quadratic term and because typically $c_{\partial p} > c_\pi$, the finite differences method has lower complexity in general. On the other hand, finite differences provide only an approximation of the actual derivative value and therefore are affected by truncation errors. Hence, the choice between an exact but slower derivative computation and a faster but less accurate finite difference approximation should be resolved for each specific test case. Factors influencing this trade-off are c_π, n_λ, $c_{\partial p}$ and the assessed accuracy of the finite difference approximation.

Proposal selection

As discussed, the self-normalised pSIS estimator is asymptotically unbiased, but the bias could be significant for small sample sizes [134]. Even when enough particles are employed, the sequential estimator often suffers from the degeneracy problem, that is, when most of the particles have zero weights [30], which causes the estimate to rely on a small effective sample. The main cause of degeneracy is the choice of a non-optimal (in terms of bias) proposal distribution, which is one that differs significantly from the target posterior. This problem is exacerbated in the epistemic scenario because there may not be a single proposal distribution which keeps the bias low for all the posteriors resulting from the epistemic set $\boldsymbol{\lambda} \in \Omega_{\boldsymbol{\lambda}}$.

In this work, the effective sample size is computed as $n_{\text{eff}} = 1/\sum_{i=1}^{N} {w_k^{(i)}}^2$. The condition of effective size below a minimum threshold is often used to trigger resampling in particle filtering [126]. However, resampling is not possible in the constructed precomputed estimator as the particles are drawn before the optimisation. Thus, one would need to check the effective sample size at the end of the estimation process and loop back to update the proposal, as done in iterated importance sampling [134].

An alternative to ensure a satisfactory sample size is to use a strategy analogous to the one implemented in recent Unscented Particle Filter (UPF) [136]. The main idea is to construct an approximated posterior by introducing information on the expected measurement through an Unscented Kalman Filter (here in Line 2 of Algorithm 4). Starting from a set of particles generated with a given proposal, one would apply a Unscented Kalman Filter (UKF) to each particle and do conditioning with the likelihood of the expected measurements. The main computational advantage of a UKF-based proposal in this method is that the sigma points can be propagated inexpensively through the surrogate polynomial expansion approximating the dynamical flow. The posterior of this process is a new proposal that includes information about the last observation resulting in more particles distributed in regions of higher expected likelihood, a crucial step to ensure a satisfactory effective sample size even without resampling.

In the case in which measurements are affected by epistemic uncertainty, this approach presents two challenges: 1) approximate the likelihood with a normal distribution such that the UKF can be applied; 2) construct a posterior which takes into account that the statistical moments of the likelihood are set-valued. We address these two challenges with the following solutions. First, we approximate the generic likelihood with a normal

$$p(\mathbf{y}_k|\mathbf{x}_k; \boldsymbol{\lambda}_y^{(j)}) \approx \mathcal{N}\left(\mathbf{y}_k|\mathbf{x}_k; \boldsymbol{\mu}_y^{(j)}, \boldsymbol{\Sigma}_y^{(j)}\right) \tag{3.23}$$

where $\boldsymbol{\mu}_y^{(j)} = \boldsymbol{\mu}_y^{(j)}(\boldsymbol{\lambda}_y^{(j)})$ and $\boldsymbol{\Sigma}_y^{(j)} = \boldsymbol{\Sigma}_y^{(j)}(\boldsymbol{\lambda}_y^{(j)})$ are the parameterised mean and covariance. For distributions with undefined or infinite mean or covariance, e.g. Lévy or

Cauchy distributions, the location and scale parameters could be used as substitutes for the mean and covariance. The second challenge is addressed by constructing the proposal as a convex combination of UKF posteriors, each resulting from a different likelihood in the epistemic set

$$\pi(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}^{(i)}, \mathbf{y}_{1:k}) = \sum_{j}^{N_\pi} b_j \, \mathcal{N}\left(\mathbf{x}_k; \boldsymbol{\mu}_k^{(i,j)}, \boldsymbol{\Sigma}_k^{(i,j)}\right), \tag{3.24}$$

where $\sum_j b_j = 1$.

The overall approach is described in Algorithm 6. Right arrows represent input to output relationships for the corresponding line in the algorithm. The initial proposal is constructed as a convex combination of multiple distributions in the initial epistemic set (Line 1). N particles are drawn from the proposal (Line 2), and for each of them, the

Algorithm 6 Construction of proposal distributions.

```
Given the estimation problem (3.5)
1: Set initial proposal as convex combination of distributions in epistemic set
   $\pi(\mathbf{x}_0) = \sum_j^{N_\pi} p(\mathbf{x}_0; \boldsymbol{\lambda}_0^{(j)})/N_\pi$ with $\boldsymbol{\lambda}_0^{(j)} \in \Omega_{\lambda_0}$
2: Draw $N$ particles from the initial proposal $\pi(\mathbf{x}_0)$
   $\mathbf{x}_0^{(i)} \sim \pi(\mathbf{x}_0)$
3: For each particle, initialise UKF mean as particle value and covariance as sample covariance
   $\boldsymbol{\mu}_0^{(i)} = \mathbf{x}_0^{(i)}$, $\boldsymbol{\Sigma}_0^{(i)} = \sum_j (\mathbf{x}_0^{(j)} - \boldsymbol{\mu}_0^{(i)})(\mathbf{x}_0^{(j)} - \boldsymbol{\mu}_0^{(i)})^T/(N-1)$
   for $k = 1 : M$ do
4:    Propagate UKF mean and covariance with UT
      $\boldsymbol{\mu}_{k-1}^{(i)}, \boldsymbol{\Sigma}_{k-1}^{(i)} \rightarrow \boldsymbol{\mu}_k^{(i)}, \boldsymbol{\Sigma}_k^{(i)}$
      for $j = 1 : N_\pi$ do
5:       Select an epistemic parameter $\boldsymbol{\lambda}_y^{(j)} \in \Omega_{\lambda_y}$ and approximate the likelihood
         $p(\mathbf{y}_k|\mathbf{x}_k; \boldsymbol{\lambda}_y^{(j)}) \rightarrow \mathcal{N}\left(\mathbf{y}_k|\mathbf{x}_k; \boldsymbol{\mu}_y^{(j)}, \boldsymbol{\Sigma}_y^{(j)}\right)$
6:       Update UKF mean and covariance with Bayes' update
         $\boldsymbol{\mu}_k^{(i)}, \boldsymbol{\Sigma}_k^{(i)} \rightarrow \boldsymbol{\mu}_k^{(i,j)}, \boldsymbol{\Sigma}_k^{(i,j)}$
      end for
7:    Construct proposal with UKF posteriors
      $\pi(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}^{(i)}, \mathbf{y}_{1:k}) = \sum_j^{N_\pi} b_j \, \mathcal{N}\left(\mathbf{x}_k; \boldsymbol{\mu}_k^{(i,j)}, \boldsymbol{\Sigma}_k^{(i,j)}\right)$
   end for
```

mean and covariance of the parallel UKFs are initialised (Line 3). For each observation instance, the UKF distributions are propagated forward in time with the UT [137] (Line 4). Then, N_π likelihoods are selected within their epistemic set according to a low-discrepancy sequence, and approximated with normal distributions (Line 5) according to the rule discussed above (see Eq. (3.23)). For each normal likelihood, the UKF Bayes' update is performed to compute the posterior mean and covariance (Line 6). From here, the proposal is constructed as the collection of UKF posteriors (Line 7). The weights b_j, which sum to one, could express the belief on the relative likelihood of each epistemic parameter.

Such procedure requires application-specific information as working input, particularly the received observations' likelihood and the epistemic domain specification. Nonetheless, this information is part of the epistemic filtering specification and therefore this proposal can be constructed for any specification of the likelihood set and application to solve.

With this approach, the proposal is constructed to cover the support of each likelihood in the epistemic set while concentrating more particles in high-likelihood areas. Furthermore, the UKF-based proposal concentrates more particles in high-likelihood areas because it explicitly incorporates information on the observations, thus reducing the risk of sample degeneracy. This behaviour is particularly valuable in space applications characterised by multi-dimensional spaces, highly peaked likelihoods (accurate observations), and little overlap between the predicted distribution and the likelihood.

Polynomial propagator

Propagating the particles in time is the most computationally intensive step in space applications as it requires solving the initial value problem in Eq. (2.17) by numerical integration as many times as the number of particles employed. In this filter, a polynomial surrogate of the dynamical flow is employed to speed up the uncertainty propagation step and enable the use of a large number of particles to improve the estimate accuracy.

The polynomial expansion discussed in Section 2.3 is adapted here to approximate the dynamical flow $\widetilde{F}_k^{k+1}$. The particles are then propagated from t_k to t_{k+1} through inexpensive polynomial evaluations.

3.2.2 Bound estimator

The robust estimation method computes the bounds in Eq. (3.8) by taking advantage of the efficient estimator constructed above. In the following, the routine to estimate the lower bound only will be discussed, as the same approach holds for the upper bound with appropriate modifications.

The lower bound estimator and the epistemic parameter achieving it are denoted as

$$\underline{\hat{\theta}}(\boldsymbol{\mathcal{X}}_{0:M}) = \min_{\boldsymbol{\lambda}\in\Omega_{\boldsymbol{\lambda}}} \hat{\theta}(\boldsymbol{\mathcal{X}}_{0:M}, \boldsymbol{\lambda}) \tag{3.25a}$$

$$\underline{\boldsymbol{\lambda}}(\boldsymbol{\mathcal{X}}_{0:M}) = \arg\min_{\boldsymbol{\lambda}\in\Omega_{\boldsymbol{\lambda}}} \hat{\theta}(\boldsymbol{\mathcal{X}}_{0:M}, \boldsymbol{\lambda}), \tag{3.25b}$$

that is, the minimum of the expectation estimator and the argument of the minimum given the set of samples $\boldsymbol{\mathcal{X}}_{0:M}$. By employing the precomputation step, and Algorithm 5 to evaluate the estimator for a candidate $\boldsymbol{\lambda}$, the estimator is written in the form of $\hat{\theta}\big(\boldsymbol{\mathcal{X}}_{0:M}, \boldsymbol{\lambda}\big) = \sum_{i=1}^{N} \hat{w}_M^{(i)}(\boldsymbol{\lambda})\, \psi^{(i)}$, with precomputed $\psi^{(i)}$, so that the optimisation process operates on the importance weights only $w_M^{(i)}(\boldsymbol{\lambda})$.

Eq. (3.25) entails two challenges: first, the optimisation problem is generally multimodal since the objective function results from numerous operations between nonlinear functions as in Eq. (3.13); second, even if the global extremum is found, $\underline{\hat{\theta}}$ could still deviate from the true sought bound $\underline{\mathbb{E}}$. The first challenge is addressed with a numerical global optimisation scheme with ensured convergence, presented in detail in Section 3.2.3. For the latter, indicators and confidence intervals to improve and quantify how well the estimator approximates the true bound are here discussed.

In a previous work [98] we analysed the estimator landscape (collision probability in that instance) as a function of the epistemic parameters and their correlation. The results showed a strong nonlinearity and multi-modality of the objective function. Thus, a local optimiser cannot handle the bound computation as it may converge to a local optimum only, and therefore it may yield an inner approximation leading to a loss of robustness. Besides, if an outer approximation is used the risk is to obtain a vacuous model where bounds are too wide to be useful. The B&B scheme presented in this section assures asymptotic convergence to the exact bounds and provides an outer approximation if stopped before convergence to the global optimum.

The approximation of the bound value itself depends on two additional factors: the precision of the pSIS estimator (3.12) in approximating the expectation for a given epistemic parameter; the accuracy of the bound estimator (3.25) in approximating the true bound.

The former requires the estimator $\hat{\theta}(\boldsymbol{\mathcal{X}}_{0:M}, \underline{\boldsymbol{\lambda}})$ to suitably approximate the expectation $\mathbb{E}_{p_{\underline{\boldsymbol{\lambda}}}}\big[\psi(\mathbf{x}_k) \,|\, \mathbf{y}_{1:k}\big]$. A good approximation is ensured by the check on the effective sample size.

The latter requires the estimator $\hat{\underline{\theta}}(\boldsymbol{\mathcal{X}}_{0:M})$ to suitably approximate the true bound $\underline{\mathbb{E}}\big[\psi(\mathbf{x}_k) \,|\, \mathbf{y}_{1:k}\big]$. [134] proved that the estimator (3.25) is a coherent lower expectation if $\hat{\theta}(\boldsymbol{\mathcal{X}}_{0:M}, \boldsymbol{\lambda})$ is a coherent expectation, which is true for the constructed estimator thanks to the self-normalisation of the weights. Furthermore, confidence bounds can be constructed for the bound estimator [134]. Let $\boldsymbol{\mathcal{X}}^1_{0:M}, \dots, \boldsymbol{\mathcal{X}}^{2n}_{0:M}$ be $2n$ sets of samples. Define the quantities

$$\begin{aligned} \hat{\underline{\theta}}^j &= \hat{\theta}\big(\boldsymbol{\mathcal{X}}^j_{0:M}, \underline{\boldsymbol{\lambda}}(\boldsymbol{\mathcal{X}}^j_{0:M})\big) \;\text{ for }\; j = 1, \dots, n \\ \hat{\underline{\theta}}^j &= \hat{\theta}\big(\boldsymbol{\mathcal{X}}^j_{0:M}, \underline{\boldsymbol{\lambda}}(\boldsymbol{\mathcal{X}}^{j-n}_{0:M})\big) \;\text{ for }\; j = n+1, \dots, 2n, \end{aligned} \tag{3.26}$$

with $\underline{\boldsymbol{\lambda}}(\boldsymbol{\mathcal{X}}_{0:M})$ defined in Eq. (3.25b). For the first half, these are the classical bound estimators, whereas for the other half, these require evaluating $\hat{\theta}$ for a set of samples $\boldsymbol{\mathcal{X}}^{j-n}_{0:M}$ with the epistemic parameter minimizing the estimator for another set of samples $\boldsymbol{\mathcal{X}}^j_{0:M}$. Let $\mu^{1H}_{\boldsymbol{\mathcal{X}}}$ and $\sigma^{1H}_{\boldsymbol{\mathcal{X}}}$ be the mean and standard deviation of the first half of $\hat{\underline{\theta}}^j$, i.e. for $j = 1, \dots, n$, and $\mu^{LH}_{\boldsymbol{\mathcal{X}}}$ and $\sigma^{LH}_{\boldsymbol{\mathcal{X}}}$ the ones of the last half of $\hat{\underline{\theta}}^j$, that is for $j = n+1, \dots, 2n$. Then, under the assumption that enough particles have been used to keep the bias small and bounded for all $\boldsymbol{\lambda}$, the confidence interval for the confidence level $1 - \alpha$ is

$$\left[\mu^{1H}_{\boldsymbol{\mathcal{X}}} - t_{\alpha,n-1}\frac{\sigma^{1H}_{\boldsymbol{\mathcal{X}}}}{\sqrt{N}}, \mu^{LH}_{\boldsymbol{\mathcal{X}}} + t_{\alpha,n-1}\frac{\sigma^{LH}_{\boldsymbol{\mathcal{X}}}}{\sqrt{N}}\right], \tag{3.27}$$

where $t_{\alpha,n-1}$ is the $1-\alpha$ two-sided critical value of the t-distribution with $n-1$ degrees of freedom.

3.2.3 Global search

The optimisation problem (3.25) is generally nonlinear, nonconvex and multi-modal because of the nonlinear dependencies of the distributions from the epistemic parameters and the nonlinear operations in the weights update step. Thus, a global optimisation routine is employed to explore expensively and robustly the epistemic domain to find the extremum of $\hat{\theta}$.

The global search for solving the optimisation problem in Eq. (3.25) is tackled with a B&B approach using simplexes as subdomains and a Lipschitz-based lower bound estimation. The advantage of using B&B is that asymptotic convergence to the global optimum is granted and that an estimation of the distance from it is known at each iteration. The advantage of using simplexes is that the number of branched sub-simplexes is decoupled from the problem dimension rather than growing exponentially as with a simple lattice grid. The Lipschitz constant can be estimated using the estimator analytical derivatives. Indeed, under the assumption that the densities derivatives exist and are bounded for $\boldsymbol{\lambda} \in \Omega_{\boldsymbol{\lambda}}$, the precomputed SIS estimator is continuous and differentiable with bounded derivatives in $\boldsymbol{\lambda}$ (from Eqs. (3.16) to (3.18)).

The B&B algorithm is presented in Appendix A.1. In this section, the bounding strategy is discussed in detail along with a lemma on the bounding step needed for the convergence proof. Then, the branching strategy is presented together with a lemma showing that this approach eventually generates sub-domains of arbitrary small size, and providing an upper bound on the number of branching steps. Finally, the theorem proving the algorithm convergence is stated.

Preliminary definitions

Let $\mathbb{S} \subset \mathbb{R}^n$ be a n-simplex with vertexes $[\boldsymbol{\lambda}_0, \dots, \boldsymbol{\lambda}_n]$. Let L, with $0 < L < \infty$, be the Lipschitz constant of $\hat{\theta}$ over $\Omega_{\boldsymbol{\lambda}}$ such that

$$\left|\hat{\theta}(\boldsymbol{\chi}_k, \boldsymbol{\lambda}_1) - \hat{\theta}(\boldsymbol{\chi}_k, \boldsymbol{\lambda}_2)\right| \leq L\|\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_2\| \quad \forall \boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2 \in \Omega_{\boldsymbol{\lambda}}. \tag{3.28}$$

Let the diameter of a n-simplex $\mathbb{S}$ be the maximum distance between any two points belonging to the simplex, which is the maximum distance between two vertexes

$$\sigma\left(S\right) = \max_{i,j \in [0:n]} \|\boldsymbol{\lambda}_i - \boldsymbol{\lambda}_j\|,$$

and let $\boldsymbol{\lambda}_{i^*}$ and $\boldsymbol{\lambda}_{j^*}$ be the most distant vertices such that $\sigma\left(S\right) = \left\|\boldsymbol{\lambda}_{i^*} - \boldsymbol{\lambda}_{j^*}\right\|$. For each branching step $k = 1, \dots, K$, let $\mathcal{L}_k$ be the list of disjoint simplexes covering the feasible domain, that is $\bigcup_j \mathbb{S}_k^{(j)} = \Omega$ for $\mathbb{S}_k^{(j)} \in \mathcal{L}_k$ and $\mathbb{S}_k^{(i)} \bigcap \mathbb{S}_k^{(j)} = \varnothing \ \forall \mathbb{S}_k^{(i)}, \mathbb{S}_k^{(j)} \in \mathcal{L}_k$. Finally, let $lb : \mathbb{S} \to \mathbb{R}$ be a function bounding $\hat{\theta}$ from below in the simplex $\mathbb{S}$

$$lb(\boldsymbol{\lambda}) \leq \hat{\theta}(\boldsymbol{\lambda}) \ \ \forall \boldsymbol{\lambda} \in \mathbb{S}$$

and denote $lb(\mathbb{S})$ its minimum

$$lb(\mathbb{S}) = \min_{\boldsymbol{\lambda} \in \mathbb{S}} lb(\boldsymbol{\lambda}).$$

Thus, $lb(\mathbb{S})$ is the lower bound for the minimum of $\hat{\theta}$ in the simplex. Similarly, denote $ub(\mathbb{S})$ the upper bound for the minimum of $\hat{\theta}$ in the simplex.

Bounding

Because $\hat{\theta}$ is Lipschitz continuous, the lower bounding function $lb : \mathbb{S} \to \mathbb{R}$ can be defined as

$$lb(\boldsymbol{\lambda}) = \max_j \left[\hat{\theta}(\boldsymbol{\lambda}_j) - L\|\boldsymbol{\lambda} - \boldsymbol{\lambda}_j\|\right], \tag{3.29}$$

where $\boldsymbol{\lambda}_j$ are the vertexes of the simplex. The lower bound value $lb(\mathbb{S})$ is therefore the minimum of $lb(\boldsymbol{\lambda})$ over the simplex $\mathbb{S}$

$$lb(\mathbb{S}) = \min_{\boldsymbol{\lambda}\in\mathbb{S}} \max_{j} \left[\hat{\theta}(\boldsymbol{\lambda}_j) - L\|\boldsymbol{\lambda} - \boldsymbol{\lambda}_j\|\right]. \tag{3.30}$$

For the upper bound of the minimum, the minimum of the fitness values at the simplex vertexes is chosen as trivial bound

$$ub(\mathbb{S}) = \min_{j} \hat{\theta}(\boldsymbol{\lambda}_j). \tag{3.31}$$

Hence, we state a lemma to show that these values actually bound the minimum over the simplex. The proof is presented in Appendix A.2.

Lemma 1. *The bounds $lb(\mathbb{S})$ and $ub(\mathbb{S})$ in Eqs.* (3.30) *and* (3.31) *satisfy*

$$lb(\mathbb{S}) \leq \min_{\boldsymbol{\lambda}\in\mathbb{S}} \hat{\theta}(\boldsymbol{\lambda}) \leq ub(\mathbb{S}). \tag{3.32}$$

Another property required to prove convergence of Algorithm 7 is that as the simplex diameter goes to zero, the bounds difference should uniformly converge to zero. This is proven in the following lemma for the B&B presented here. Again, the proof is discussed in Appendix A.2.

Lemma 2. *The bounds lb and ub satisfy*

$$\forall\varepsilon \in \mathbb{R}^+ \ \exists\delta \in \mathbb{R}^+ \ s.t. \ \forall\mathbb{S} \subset \mathbb{R}^n, \ \sigma(\mathbb{S}) \leq \delta \implies ub(\mathbb{S}) - lb(\mathbb{S}) \leq \varepsilon. \tag{3.33}$$

An analytical procedure has been developed to compute the lower bound $lb(\mathbb{S})$ for the Lipschitz bounding function in Eq. (3.30), and it is presented in Appendix A.3.

Branching

To prove B&B convergence, we need to show that the branching scheme eventually produces a simplex of arbitrary size δ. We adopt Longest Edge Bisection (LEB) as branching rule. The simplex $\mathbb{S}$ is split into two simplexes by bisection along its longest edge, that is, the one connecting the vertices $\boldsymbol{\lambda}_{i^*}$ and $\boldsymbol{\lambda}_{j^*}$, such that

$$\boldsymbol{\lambda}_{ij}^* = \frac{\boldsymbol{\lambda}_i^* + \boldsymbol{\lambda}_j^*}{2} \tag{3.34}$$

is the new vertex. The two offspringing simplexes $S^{(1)}$ and $S^{(2)}$ share the same vertices of $\mathbb{S}$ except that one has $\boldsymbol{\lambda}_{ij}^*$ in place of $\boldsymbol{\lambda}_i^*$, whereas the other has $\boldsymbol{\lambda}_{ij}^*$ in place of $\boldsymbol{\lambda}_j^*$. [138] showed that LEB reduces the diameter of a n-simplex of at least $\left(\sqrt{3}/2\right)^{\lfloor k/n \rfloor}$ after k splittings. On the other hand, as trivial lower bound, LEB reduces the diameter of a n-simplex no more than half the size of the immediate parent after 1 split.

For each branching step $k = 1, \ldots, K$, choose a simplex $S \in \mathcal{L}_{k-1}$ and split it into 2 disjoint simplexes $S^{(1)}$ and $S^{(2)}$ by LEB. Define the next list of simplexes as $\mathcal{L}_k = (\mathcal{L}_{k-1} \setminus S) \cup S^{(1)} \cup S^{(2)}$.

We now state that this branching rule eventually creates a simplex of arbitrarily small diameter and define an upper bound on the number of iterations. The proof is reported in Appendix A.2.

Lemma 3. *Let $\mathcal{L}_0$ be the initial list of N_0 n-simplexes and*

$$\sigma_0 = \max_{\mathbb{S}_0^{(i)} \in \mathcal{L}_0} \sigma\left(\mathbb{S}_0^{(i)}\right)$$

the largest diameter among them. For any $\delta \in \mathbb{R}^+$, there exist $K \in \mathbb{N}$ such that the LEB rule yields

$$\min_{\mathbb{S}_k^{(j)} \in \mathcal{L}_K} \sigma\left(\mathbb{S}_k^{(j)}\right) \leq \delta.$$

Specifically, the requested diameter is realised after at most K splitting of the initial simplexes with

$$K = N_0\, n \left\lceil \log_{\frac{2}{\sqrt{3}}} \frac{\sigma_0}{\delta} \right\rceil . \tag{3.35}$$

Convergence

The algorithm terminates when the difference between the upper and lower bounds on the most promising simplex $\mathbb{S}^*$, that is the one with the lowest bound for the minimum

$lb(\mathbb{S}^*) \leq lb(\mathbb{S}_k^{(j)})$ for any j, is below a given threshold. Let

$$\begin{aligned} L_k &= \min_{\mathbb{S}_k^{(j)} \in \mathcal{L}_k} lb\big(\mathbb{S}_k^{(j)}\big) \\ U_k &= \min_{\mathbb{S}_k^{(j)} \in \mathcal{L}_k} ub\big(\mathbb{S}_k^{(j)}\big), \end{aligned}$$

then with the lemmas presented above, we can now state the theorem proving the convergence of Algorithm 7. The proof is detailed in Appendix A.2.

Theorem 1. *Algorithm 7 converges in a finite number of steps to within a set threshold* $\varepsilon \in \mathbb{R}^+$ *of the global minimum* $\underline{\hat{\theta}}$. *Specifically*

$$\forall \varepsilon \in \mathbb{R}^+ \; \exists K_\varepsilon \text{ such that } U_{K_\varepsilon} - L_{K_\varepsilon} \leq \varepsilon, \; L_{K_\varepsilon} \leq \underline{\hat{\theta}} \leq U_{K_\varepsilon}.$$

This ensures the global minimum to be within the interval $\underline{\hat{\theta}} \in [lb(\mathbb{S}^*_{K_\varepsilon}), ub(\mathbb{S}^*_{K_\varepsilon})]$ at convergence after K_ε steps.

Filter complexity with epistemic dimension

In this section, we quantify the computational complexity of the filter and its dependency on the number of dimensions of the epistemic space. This dependency gives a measure of the scalability of the filter with the number of epistemic parameters. We start by noting that the number of particles in the pSIS estimator depends on the dimension of the state space and on the shape of the distributions of the random variables $\mathbf{X}_k$, $\mathbf{D}$, and $\mathbf{Y}$ but not directly on the dimension of the epistemic space. Similarly, the polynomial surrogate depends on the number of aleatory variable, but not on the number of epistemic ones. Thus the only part of the filter whose complexity is directly affected by the number of epistemic parameters is the global optimisation algorithm.

Let n_λ be the number of epistemic variables. The initial number N_0 of simplexes in $\mathcal{L}_0$ covering the n_λ-dimensional epistemic space depends on the rule employed to partition Ω_λ. We currently use a standard algorithm for triangulation of a hyper-box into $N_0 = n_\lambda!$ simplexes [139]. Nonetheless, a more convenient decomposition that yields a lower number of initial simplexes could be employed. Among them, an initial circumscribing simplex with a fitness penalisation for points outside the domain yield $N_0 = 1$ regardless of n_λ [140, 141], a minimum-cardinality triangulation can be achieved by means of linear programming [142], or covering with intersecting simplexes could be employed in place of triangulation [143]. Therefore, in our complexity analysis we will retain $N_0 = N_0(n_\lambda)$ without making such dependence explicit.

In the first iteration, at most $N_0(n_\lambda + 1)$ evaluations of the pSIS estimator are required. For each LEB branching step, only one estimator evaluation is needed at the new common vertex $\boldsymbol{\lambda}^*_{ij}$. By using the maximum number of branching steps K from Eq. (3.35), the upper bound on the filter complexity is

$$C_{\mathrm{RPF}}(n_\lambda) = 2\left[N_0(n_\lambda + 1) + K\right] C_{pSIS} = 2\left[N_0 + N_0\, n_\lambda\left(1 + \left\lceil \log_{\frac{2}{\sqrt{3}}} \frac{\sigma_0}{\delta} \right\rceil\right)\right] C_{pSIS}, \tag{3.36}$$

where C_{pSIS} was defined in Eq. (3.21). Therefore, the complexity upper bound is linear with the number of epistemic variables if N_0 is independent of n_λ, e.g. [140], otherwise it scales as $n_\lambda N_0(n_\lambda)$. By pruning the simplexes, as the B&B proceeds with the search, the actual filter complexity is significantly reduced from this conservative estimate.

Lipschitz constant estimation

The Lipschitz constant is estimated as the maximum of the estimator gradient evaluated on the vertexes of the simplexes. Specifically, at a generic iteration k, the domain is partitioned in disjoint simplexes $\mathbb{S}^{(i)}_k$ each with vertexes $[\boldsymbol{\lambda}^{(i)}_0, \dots, \boldsymbol{\lambda}^{(i)}_n]$. The Lipschitz constant is set to

$$L = \max_{i,j} \left\| \nabla_{\boldsymbol{\lambda}}\, \hat{\theta}(\boldsymbol{\mathcal{X}}_{0:k}, \boldsymbol{\lambda}^{(i)}_j) \right\|. \tag{3.37}$$

This is an adaptive, although lower, approximation of the true Lipschitz constant. Therefore, it could lead to over-pruning potentially optimal simplexes. However, lower approximations are often employed in literature and have proven to be efficient on a large number of test cases [144, 145, 146]. Furthermore, if a larger Lipschitz constant is found, each pruned simplex is re-assessed and recovered if it was wrongly excluded.

3.3 Epistemic variational inference

This section presents a second approach designed specifically to handle epistemic uncertainty in Navigation Analysis (NA) for space applications. In particular, the approach is again mathematically framed as a sequential filtering problem. A typical trait of sequential filtering is the alternation of UP arcs, through the dynamical model, and OD instances, when a new measurement is received and the state distribution is updated.

3.3.1 Variational inference

For a generic observation time t_k, let us recall that $p(\mathbf{x}_k|\mathbf{y}_{1:k-1})$ is the prior distribution, $p(\mathbf{y}_k|\mathbf{x}_k)$ the likelihood function of the received observation, and $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ the sought posterior. By Bayesian inference, the target distribution (3.3) is expressed as [30]

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) = \frac{p(\mathbf{y}_k|\mathbf{x}_k)\, p(\mathbf{x}_k|\mathbf{y}_{1:k-1})}{p(\mathbf{y}_k|\mathbf{y}_{1:k-1})} \ , \tag{3.38}$$

where

$$p(\mathbf{y}_k|\mathbf{y}_{1:k-1}) = \int p(\mathbf{y}_k|\mathbf{x}_k)\, p(\mathbf{x}_k|\mathbf{y}_{1:k-1})\, d\mathbf{x}_k.$$

The Bayes update process is represented in Figure 3.3, where the prior is represented in blue, the likelihood in yellow and the posterior in green. The denominator is the most complex term to compute as it requires solving a multidimensional integral. Hence, the

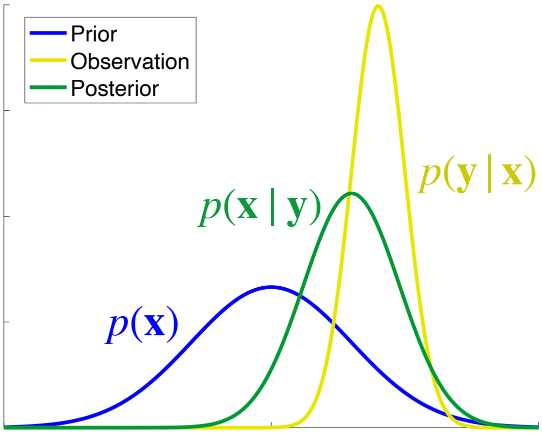


Figure 3.3: Representation of Bayes' inference.

goal is to approximate $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ with a parametric variational distribution $q(\mathbf{x}_k)$. The parameter of the variational distribution is not reported explicitly for ease of notation until the development of epistemic variational inference.

Variational distribution optimisation

We use KL divergence as a measure of dissimilarity between target p and variational distribution q

$$D_{KL}\left[p\|q\right] = \mathbb{E}_q\left[\ln\frac{q(\mathbf{x}_k)}{p(\mathbf{x}_k|\mathbf{y}_{1:k})}\right] = \int \ln\frac{q(\mathbf{x}_k)}{p(\mathbf{x}_k|\mathbf{y}_{1:k})}\, q(\mathbf{x}_k)\, d\mathbf{x}_k \ , \tag{3.39}$$

where $\mathbb{E}_q$ indicates the expectation with respect to q. The goal is, therefore, to find the variational distribution minimising the divergence

$$q^*(\mathbf{x}_k) = \arg\min_{q\in\mathcal{Q}} D_{KL}\left[p\|q\right], \tag{3.40}$$

where $\mathcal{Q}$ is the set of admissible variational distributions. To simplify the optimisation, the divergence can be written by exploiting logarithmic properties as

$$D_{KL}\left[p\|q\right] = \mathbb{E}_q\left[\ln q(\mathbf{x}_k) - \ln p(\mathbf{x}_k|\mathbf{y}_{1:k})\right] \ . \tag{3.41}$$

Substituting Equation (3.38) and using the expectation linearity, one gets

$$\begin{aligned} D_{KL}\left[p\|q\right] &= \mathbb{E}_q\left[\ln\frac{q(\mathbf{x}_k)}{p(\mathbf{y}_k|\mathbf{x}_k)\, p(\mathbf{x}_k|\mathbf{y}_{1:k-1})}\right] + \mathbb{E}_q\left[\ln p(\mathbf{y}_k|\mathbf{y}_{1:k-1})\right] \\ &= \mathbb{E}_q\left[\ln\frac{q(\mathbf{x}_k)}{p(\mathbf{y}_k|\mathbf{x}_k)\, p(\mathbf{x}_k|\mathbf{y}_{1:k-1})}\right] + \ln p(\mathbf{y}_k|\mathbf{y}_{1:k-1}) \ , \end{aligned} \tag{3.42}$$

where $\mathbb{E}_q\left[\ln p(\mathbf{y}_k|\mathbf{y}_{1:k-1})\right] = \ln p(\mathbf{y}_k|\mathbf{y}_{1:k-1})$ because $\ln p(\mathbf{y}_k|\mathbf{y}_{1:k-1})$ is independent from $\mathbf{x}$. Therefore, $\ln p(\mathbf{y})$ is a constant and can be neglected in the optimisation, which can now be written as

$$\min_{q\in\mathcal{Q}} \mathbb{E}_q\left[\ln\frac{q(\mathbf{x}_k)}{p(\mathbf{y}_k|\mathbf{x}_k)\, p(\mathbf{x}_k|\mathbf{y}_{1:k-1})}\right] \ . \tag{3.43}$$

As seen before, this expectation has no closed-form solution and needs to be computed numerically. As the expectation is a n-dimensional integral, employing an efficient numerical method is crucial in solving the minimisation problem efficiently. In this development, we employ importance sampling to the expectation computation, that is, a Monte Carlo method sampling a proposal distribution $\pi(\mathbf{x}_k)$ which has a larger support than all the distributions in $\mathcal{Q}$. Hence, we can rewrite the generic expectation $\mathbb{E}_q\left[\psi(\mathbf{x}_k)\right]$ as

$$\mathbb{E}_q\left[\psi(\mathbf{x}_k)\right] = \int \psi(\mathbf{x}_k)\, q(\mathbf{x}_k)\, d\mathbf{x}_k = \int \psi(\mathbf{x}_k)\, \frac{q(\mathbf{x}_k)}{\pi(\mathbf{x}_k)}\, \pi(\mathbf{x}_k)\, d\mathbf{x}_k = \mathbb{E}_\pi\left[\psi(\mathbf{x}_k)\, \frac{q(\mathbf{x}_k)}{\pi(\mathbf{x}_k)}\right] . \tag{3.44}$$

Thus, a Monte Carlo method taking N samples from the proposal $\mathbf{x}_k^{(i)} \sim \pi(\mathbf{x}_k)$ approximates the posterior as

$$\mathbb{E}_q\left[\psi(\mathbf{x}_k)\right] \approx \frac{1}{N} \sum_{i=1}^{N} \frac{q(\mathbf{x}_k^{(i)})}{\pi(\mathbf{x}_k^{(i)})}\, \psi(\mathbf{x}_k^{(i)}) . \tag{3.45}$$

Therefore, we can run the optimisation over the variational distributions using a single proposal and fixed samples. Furthermore, many of the quantities are now independent on q and therefore can be precomputed. The discretised optimisation problem to be solved now is

$$\min_{q\in\mathcal{Q}} \sum_{i=1}^{N} \frac{q(\mathbf{x}_k^{(i)})}{\pi(\mathbf{x}_k^{(i)})} \ln \frac{q(\mathbf{x}_k^{(i)})}{p(\mathbf{y}_k|\mathbf{x}_k^{(i)})\, p(\mathbf{x}_k^{(i)}|\mathbf{y}_{1:k-1})} . \tag{3.46}$$

Epistemic variational inference

In the epistemic setting, both the prior distribution, coming from the propagation, and the likelihood function can be affected by epistemic uncertainty as

$$\begin{gathered} p(\mathbf{x}_k|\mathbf{y}_{1:k-1};\boldsymbol{\lambda}_{x_k}) \in \mathcal{P}_{x_k} \\ p(\mathbf{y}_k|\mathbf{x}_k;\boldsymbol{\lambda}_{y_k}) \in \mathcal{P}_{y_k} \end{gathered} . \tag{3.47}$$

The posterior distribution resulting from the variational inference (3.40) is therefore set-valued as well

$$q(\mathbf{x}_k;\boldsymbol{\lambda}_k) \in \mathcal{P}_{x_k|y_k} \tag{3.48}$$

with

$$\mathcal{P}_{x_k|y_k} = \left\{ q^*(\mathbf{x}_k;\boldsymbol{\lambda}_k) \;\middle|\; \underset{q\in\mathcal{Q}}{\arg\min}\, D_{KL}\big[\, p(\mathbf{x}_k|\mathbf{y}_{1:k};\boldsymbol{\lambda}_k) \| \, q(\mathbf{x}_k) \,\big] \right\} , \tag{3.49}$$

where $\boldsymbol{\lambda}_k = [\boldsymbol{\lambda}_{x_k}, \boldsymbol{\lambda}_{y_k}]$ is the shorthand for the collection of the different epistemic parameters involved. The disadvantage of this representation is that the epistemic uncertainty accumulates when new observations are received, leading to an increase in the uncertainty space dimensionality.

Hence, we aim at constructing a posterior set representation as

$$q(\mathbf{x}_k;\boldsymbol{\lambda}_{q_k}) \in \mathcal{P}_{q_k} \;, \tag{3.50}$$

by introducing a new epistemic parameter $\boldsymbol{\lambda}_{q_k}$. In this way, the epistemic posterior set is independent on $\boldsymbol{\lambda}_{x_k}$ and $\boldsymbol{\lambda}_{y_k}$ once the inference step is completed. To achieve this goal, the variational distribution employed is a mixture one as

$$q(\mathbf{x}_k;\boldsymbol{\lambda}_{q_k}) = \sum_{j=1}^{M} \lambda_{q_k}^{(j)} q^{(j)}(\mathbf{x}_k) \tag{3.51}$$

subject to

$$\begin{gathered} \sum_{j=1}^{M} \lambda_{q_k}^{(j)} = 1 \\ \int q^{(j)}(\mathbf{x}_k) d\mathbf{x}_k = 1 \;\; \forall j \in [1,\dots,M] \;. \end{gathered} \tag{3.52}$$

The conditions (3.52) ensures that the distribution (3.53) is a valid density function integrating to one. The epistemic parameters are the mixture weights

$$\boldsymbol{\lambda}_{q_k} = [\lambda_{q_k}^{(1)},\dots,\lambda_{q_k}^{(M)}] \;. \tag{3.53}$$

Thus, by plugging Eq. (3.51) in Eq. (3.40), the variational optimisation now becomes

$$\begin{aligned} \min_{\boldsymbol{\lambda}_{q_k}} & \sum_{i=1}^{N} \frac{\sum_{j=1}^{M} \lambda_{q_k}^{(j)} q^{(j)}(\mathbf{x}_k^{(i)})}{\pi(\mathbf{x}_k^{(i)})} \ln \frac{\sum_{j=1}^{M} \lambda_{q_k}^{(j)} q^{(j)}(\mathbf{x}_k^{(i)})}{p(\mathbf{y}_k|\mathbf{x}_k^{(i)})\, p(\mathbf{x}_k^{(i)}|\mathbf{y}_{1:k-1})} \\ \text{s.t.} & \sum_{j=1}^{M} \lambda_{q_k}^{(j)} = 1 \qquad . \end{aligned} \tag{3.54}$$

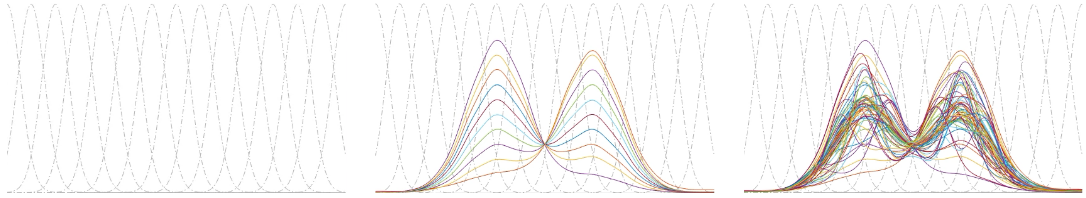
Hence, the procedure to perform the variational inference under epistemic uncertainty is structured as follows:

1. Take S different pairs of prior and likelihood epistemic parameters $\boldsymbol{\lambda}_{x_k} \in \Omega_{\boldsymbol{\lambda}_{x_k}}$ and $\boldsymbol{\lambda}_{y_k} \in \Omega_{\boldsymbol{\lambda}_{y_k}}$;
2. For each s-th pair of epistemic parameters, with $s = 1, \dots, S$, solve the optimisation (3.54) to get the optimal coefficients $\boldsymbol{\lambda}_{q_k}^{s*}$;
3. Construct the posterior epistemic set as

$$\mathcal{P}_{x_k|y_k} = \left\{ \sum_{j=1}^{M} \lambda_{q_k}^{(j)} q^{(j)}(\mathbf{x}_k) \;\middle|\; \lambda_{q_k}^{(j)} \in \left[\min_s \lambda_{q_k}^{s*(j)}, \max_s \lambda_{q_k}^{s*(j)} \right], \sum_{j=1}^{M} \lambda_{q_k}^{(j)} = 1 \right\} . \quad (3.55)$$

Therefore, the posterior epistemic set is constructed by taking an outer approximation on the coefficients found by performing variational inference on S epistemic instances of the prior and likelihood.

Such process is illustrated in Figure 3.4. Specifically, Figure 3.4(a) represents the kernels $q^{(j)}(\mathbf{x}_k)$ of the variational distribution. Figure 3.4(b) depicts the posteriors with coefficients $\boldsymbol{\lambda}_{q_k}^{s*}$ resulting from the S variational inferences performed. Once the epistemic posterior is formed as in Equation (3.55), Figure 3.4(c) represents multiple instances of such posterior obtained by sampling the epistemic coefficients within the componentwise bounds $\left[\min_s \lambda_{q_k}^{s*(j)}, \max_s \lambda_{q_k}^{s*(j)}\right]$ under the conditions that they sum to one.



(a) Kernels of mixture to represent posterior.

(b) S variational inference optimisations given epistemic instances of prior and posterior.

(c) Instances of the epistemic posterior distribution (3.55).

Figure 3.4: Representation of the epistemic variational inference approach.

Such epistemic inference enables the re-initialisation of the state uncertainty after each observation update. The posterior imprecise set is indeed constructed with epistemic parameters $\boldsymbol{\lambda}_{q_k}$ which do not depend on the previous history. Therefore, this procedure avoids the aforementioned accumulation of uncertainty.

The optimisation in Equation (3.54) is performed using a local solver because analytical derivatives of the objective are available and easy to compute, and the constraints are linear. It may be the case that the global optimum is not found. However, the goal is to select a variational distribution that suitably approximates the target posterior, although not the best possible fit. Therefore, local optimality is considered acceptable because of the advantages in computational time. The derivative of the objective with respect to the j-th free variable is

$$\sum_{i=1}^{N} \frac{q^{(j)}(\mathbf{x}_k^{(i)})}{\pi(\mathbf{x}_k^{(i)})} \left(\ln \frac{\sum_{j=1}^{M} \lambda_{q_k}^{(j)} q^{(j)}(\mathbf{x}_k^{(i)})}{p(\mathbf{y}_k|\mathbf{x}_k^{(i)})\, p(\mathbf{x}_k^{(i)}|\mathbf{y}_{1:k-1})} + 1 \right) . \tag{3.56}$$

Hence, the analytical objective's gradient can be easily computed.

3.3.2 Mixture uncertainty propagation

In the epistemic inference developed, the variational distribution has been imposed to be a mixture of kernels. In this section, we will focus on UP for a mixture of normal distributions, although different kernels can be easily adapted.

The usual two-step polynomial UP is exploited. The polynomial mapping $\widetilde{F}_{k-1}^{k}$ is constructed non-intrusively as described in Section 2.3.2. The Gaussian mixture propagation through the sparse Gauss-Hermite cubature discussed in Section 2.4.2 is employed. In the next section, a short adaptation to the mixture scenario is presented.

Gaussian mixture propagation

The state probability distribution at time t_{k-1} is a Gaussian mixture one with interval-valued weights (see Eq. (3.55))

$$p(\mathbf{x}_{k-1}|\mathbf{y}_{0:k-1}) = \sum_{j=1}^{M} \lambda_{k-1}^{(j)} \mathcal{N}(\mathbf{x}_{k-1}; \boldsymbol{\mu}_{k-1}^{(j)}, \boldsymbol{\Sigma}_{k-1}^{(j)}) , \tag{3.57}$$

where $\boldsymbol{\mu}^{(j)}$ is the j-kernel mean and $\boldsymbol{\Sigma}^{(j)}$ its covariance. The goal is to compute the propagated distribution at time t_k.

In general, an uncertainty distribution propagated through a nonlinear dynamics does not keep its Gaussianity. However, when using a Gaussian mixture, each kernel acts on a more localised portion of the domain. Therefore, the dynamical nonlinearities experienced in each component are smaller than they would have been by using a single larger Gaussian distribution.

Hence, the propagated distribution is still assumed as a Gaussian mixture in the form of

$$p(\mathbf{x}_k|\mathbf{y}_{0:k-1}) = \sum_{j=1}^{M} \lambda_{k-1}^{(j)} \mathcal{N}(\mathbf{x}_k; \boldsymbol{\mu}_k^{(j)}, \boldsymbol{\Sigma}_k^{(j)}) \ , \tag{3.58}$$

where the means and covariances are computed by means of Gauss-Hermite cubature (see Section 2.4.2) as

$$\begin{aligned} \boldsymbol{\mu}_k^{(j)} &= \sum_{i=1}^{N} w^{(j,i)} \, \widetilde{F}_{k-1}^{k}(\boldsymbol{\xi}_{k-1}^{(j,i)}) \\ \boldsymbol{\Sigma}_k^{(j)} &= \sum_{i=1}^{N} w^{(j,i)} \left(\widetilde{F}_{k-1}^{k}(\boldsymbol{\xi}_{k-1}^{(j,i)}) - \boldsymbol{\mu}_k^{(j)}\right) \left(\widetilde{F}_{k-1}^{k}(\boldsymbol{\xi}_{k-1}^{(j,i)}) - \boldsymbol{\mu}_k^{(j)}\right)^T \ , \end{aligned} \tag{3.59}$$

where, again, $\boldsymbol{\xi}_k^{(j,i)} = [\mathbf{x}_k^{(j,i)}, \mathbf{d}_k^{(j,i)}]^T$ are the roots of the multivariate Hermite polynomial and $w^{(j,i)}$ the corresponding quadrature weights for the j-th belief component [124].

Lower and upper expectations

In navigation analysis often the interest is to compute the expectation of a quantity ψ connected to the uncertain state $\mathbb{E}_p\left[\psi(\mathbf{x}_k)\right]$. When the state distribution is epistemic, the value of such expectation depends on the epistemic parameter as in Eq. (3.7). Hence, we can compute tight bounds on such expectation such that $\mathbb{E}_{p_{\lambda_k}} \in \left[\underline{\mathbb{E}}_{p_{\lambda_k}}, \overline{\mathbb{E}}_{p_{\lambda_k}}\right]$. By plugging the variational form (3.55) in the expectation definition, we obtain

$$\begin{aligned}\mathbb{E}_{p_{\lambda_k}}[\psi(\mathbf{x}_k)] &= \int \psi(\mathbf{x}_k) \sum_{j=1}^{M} \lambda_{q_k}^{(j)} q^{(j)}(\mathbf{x}_k) d\mathbf{x}_k \\ &= \sum_{j=1}^{M} \lambda_{q_k}^{(j)} \int \psi(\mathbf{x}_k) q^{(j)}(\mathbf{x}_k) d\mathbf{x}_k \\ &= \sum_{j=1}^{M} \lambda_{q_k}^{(j)} \mathbb{E}_{q^{(j)}}[\psi(\mathbf{x}_k)] \ .\end{aligned} \tag{3.60}$$

Therefore, once the kernels expectations $\mathbb{E}_{q^{(j)}}[\psi(\mathbf{x}_k)]$ are computed, e.g. using Gauss-Hermite quadrature rules, the lower bound can be easily computed as the solution of a linear programming problem

$$\begin{aligned}\underline{\mathbb{E}}_{p_{\lambda_k}} = \min_{\lambda_k} \ & \sum_{j=1}^{M} \lambda_{q_k}^{(j)} \mathbb{E}_{q^{(j)}}[\psi(\mathbf{x}_k)] \\ \text{s.t.} \ & \sum_{j=1}^{M} \lambda_{q_k}^{(j)} = 1 \ .\end{aligned} \tag{3.61}$$

The same holds for the upper bound, where a maximisation is performed in place of a minimisation. Plenty of efficient numerical routines exist to solve linear programming problems, e.g. MATLAB's *linprog*. In addition, because the linear programming problem is a subclass of convex optimisation, the found optimum is also granted to be globally optimum. Therefore, the chosen variational parameterisation makes it extremely efficient and robust to compute tight lower and upper expectations.

3.4 Chapter summary

This chapter proposed a formulation to the state estimation problem under mixed aleatory and epistemic uncertainty. The filtering problem was formulated in terms of expectations to estimate given noisy indirect observations of the state of a dynamical system. Mixed aleatoric and epistemic uncertainty can affect the prior, likelihood, and model parameters' distributions. The epistemic filter output is a robust interval whose bounds are the expectation's extrema resulting from epistemic uncertainty. Two solution approaches have been developed.

One of the main contributions was the development of the robust particle filter in Section 3.2. Estimators for the expectation resulting from different epistemic parameters and its robust bounds were proposed, their computational complexity derived, and their accuracy assessed. By exploiting precomputation performed with a standard particle filter, the RPF approach was designed to efficiently solve the epistemic state estimation problem by inexpensively tuning the particles' importance weights. A simplicial B&B optimiser exploiting the estimator's Lipschitz continuity was developed to ensure convergence to the true bounds. Theoretical results proving the optimiser convergence with the employed bounding functions were derived. This approach will be tested in Chapter 5.

The second contribution was specifically tailored for sequential filtering under epistemic uncertainty as a fundamental step towards an integrated approach for trajectory design under generalised uncertainty models. Specifically, the main contribution of Section 3.3 was the development of a combination of variational inference and importance sampling to solve the Bayes' update step in the presence of epistemic priors and likelihoods. Methods for performing both UP and OD under epistemic mixture distributions were introduced and discussed. The OD step was solved by using the developed epistemic variational inference approach, which requires the solution of several local optimisations to find the posterior distributions. The UP was realised by Gauss-Hermite quadrature rules sped up by sparse polynomial mappings to approximate the dynamical propagation. Such epistemic steps remove the need for large two-level Monte Carlo sampling over the initial dispersion, observation errors and model uncertainties. Another advantage of the developed approach is the formulation of lower and upper expectations as easy-to-solve linear programming problems. This method will be applied in Chapter 8.

Chapter 4

Optimal Control under Uncertainty

The content of this chapter was published in:
C. Greco, M. Di Carlo, M. Vasile, and R. Epenoy, “Direct multiple shooting transcription with polynomial algebra for optimal control problems under uncertainty”, *Acta Astronautica*, 2020 [87];
C. Greco, S. Campagnola, and M. Vasile, “Robust space trajectory design using belief optimal control”, *Journal of Guidance, Control, and Dynamics*, under review [85].

This chapter presents two developments for the solution of optimal control problems under uncertainty. Optimal control problems under uncertainty are recast in a general framework suitable to model a wide class of stochastic problems, including the case of non-parametric distributions and epistemic uncertainty. Moreover, the premise of a reference trajectory is abandoned in favour of an extended uncertainty set representation. Each sample within the uncertainty set is a fully admissible pointwise trajectory with associated density.

The first approach implements a generalised multiple shooting transcription using polynomial algebra. The second one extends the multiple shooting transcription method to incorporate a full navigation analysis in the robust trajectory optimisation problem. This generalisation enables the direct coupling of trajectory optimisation and navigation analysis by incorporating the quantification of uncertainty metrics within the optimisation cycle.

The remainder of the chapter is structured as follows: Section 4.1 shows the transcription approach with GPA; Section 4.2 presents the shooting generalisation to include navigation analysis in the design cycle; Section 4.3 summarises the concepts introduced in this chapter.

4.1 Direct transcription with polynomial algebra

In the general Bolza formulation, the deterministic Optimal Control Problem (dOCP) reads as

$$\min_{\mathbf{u}\in U} \; J = \Phi(t_f, \mathbf{x}_f) + \int_{t_0}^{t_f} L(t, \mathbf{x}, \mathbf{u})\, dt \tag{4.1a}$$

$$\text{s.t.} \quad \dot{\mathbf{x}} = f(t, \mathbf{x}, \mathbf{d}, \mathbf{u}) \tag{4.1b}$$

$$\mathbf{g}\left(t, \mathbf{x}, \mathbf{d}, \mathbf{u}\right) \in \boldsymbol{G} \tag{4.1c}$$

$$\mathbf{b}\left(t_0, \mathbf{x}_0, t_f, \mathbf{x}_f\right) \in \boldsymbol{B} \;, \tag{4.1d}$$

where again $t \in \mathbb{R}$ denotes the independent variable, $\mathbf{x} \in \mathbb{R}^{n_x}$ the state vector, $\mathbf{u}(t) \in \mathbb{R}^{n_u}$ the control, and $\mathbf{d} \in \mathbb{R}^{n_d}$ the static model parameters. The aim of the deterministic optimal control problem is to find an optimal control law $\mathbf{u}^*(t) \in U \subset \mathbb{R}^{n_u}$ such that objective function J in Equation (4.1a) is minimised while fulfilling, respectively, the dynamical nonlinear equations of motion in Equation (4.1b), the path constraints in Equation (4.1c), and the boundary conditions in Equation (4.1d). Path and boundary constraints are expressed with a set inclusion condition that represents both equality and inequality constraints. This formulation is suitable for optimising a deterministic trajectory governed by a single control profile and deterministic dynamics.

When uncertainties come into play, a set of admissible trajectories is associated with a single control profile. Hence, Problem (4.1) needs to be extended to allow the treatment of a general optimal control problem for a dynamical system under uncertainty. Two different formulations will be presented in the following sections.

4.1.1 Optimal control under uncertainty

The parametric uncertainty described in Section 2.1 will be considered in this section. The two uncertain variables are collected in the joint variable $\boldsymbol{\Xi} = [\mathbf{X}_0, \mathbf{D}]$, with realisation $\boldsymbol{\xi} = [\mathbf{x}_0, \mathbf{d}]$ and joint pdf $p(\boldsymbol{\xi}) = p(\mathbf{x}_0)\, p(\mathbf{d})$ because the initial conditions and static parameters are assumed independent. The control variables will be treated as a completely deterministic input, since possible execution errors can be modelled in the dynamics as uncertain variables incorporated in $\mathbf{D}$.

As seen, the dynamical nature of the system induces the state at time t to be a random variable $\mathbf{X}_t$ as well. In the general nonlinear case, directly computing the evolution of the density distribution over time $p(\mathbf{x}_t)$ is an ambitious and, when possible, laborious task. Therefore, this problem is often tackled with sampling techniques. Indeed, the solution of the dynamical equations can be directly used as a map from the state and parameter sample space at a given time to the state sample space at another time. Then, the distribution at the time of interest may be reconstructed according to the sample responses, usually by fitting a parametric distribution.

In the context of optimal control under uncertainty, the definition of objective and constraint functions is critical because their formulation greatly affects the result and its interpretation. Common choices in stochastic programming are to formulate objective and constraints either in expected value or in probability [147]. In this section, we adopt the unified formulation in expectation described in Section 2.1, that makes use of the auxiliary function ψ to customise the constraints and objective function under uncertainty.

We can now use the formulation in expectation to cast a general optimal control problem under uncertainty in the following form:

$$\min_{\mathbf{u}\in U} \mathbb{E}\big[\psi_J(t, \mathbf{x}, \mathbf{u}, \mathbf{d})\big] \tag{4.2a}$$

$$\text{s.t.} \quad \dot{\mathbf{x}} = f(t, \mathbf{x}, \mathbf{u}, \mathbf{d}) \tag{4.2b}$$

$$\mathbb{E}\big[\psi_{\mathbf{g}}(t, \mathbf{x}, \mathbf{u}, \mathbf{d})\big] \in \boldsymbol{\Psi}_{\boldsymbol{G}} \tag{4.2c}$$

$$\mathbb{E}\big[\psi_{\mathbf{b}}(t_0, \mathbf{x}_0, t_f, \mathbf{x}_f)\big] \in \boldsymbol{\Psi}_{\boldsymbol{B}} \ , \tag{4.2d}$$

where ψ_J, $\psi_{\mathbf{g}}$ and $\psi_{\mathbf{b}}$ are the problem-dependent auxiliary functions for the objective, path and boundary constraints respectively, and $\boldsymbol{\Psi_G}$ and $\boldsymbol{\Psi_B}$ are the feasible sets for such constraints.

Generally, problem (4.2) does not have a closed-form solution. Thus a numerical solution scheme is required. The next section will introduce a novel numerical method for the solution of optimal control problems under uncertainty in the form of problem (4.2).

4.1.2 Direct shooting with generalised polynomial algebra

A well-established method for the solution of optimal control problems is known as direct multiple shooting [148]. Direct multiple shooting starts by discretising the time domain into time segments $[t_k, t_{k+1}]$. The k^{th} segment is defined by the initial state $\mathbf{x}_k$ and a parameterisation of the control profile $\mathbf{u}_k(t) = \mathcal{U}_k(t, \boldsymbol{\beta}_k)$ as function of some free parameters $\boldsymbol{\beta}_k$. A Nonlinear Programming (NLP) solver is then used to identify the optimal value of the states $\mathbf{x}_k$ and parameters $\boldsymbol{\beta}_k$ for all the segments. Every time the NLP solver assigns a value to $\mathbf{x}_k$ and $\boldsymbol{\beta}_k$, $\mathbf{x}_k$ is propagated from t_k to t_{k+1}. The NLP solver enforces continuity constraints on the states at the boundary of two adjacent segments to ensure the continuity of the final solution and the satisfaction of the boundary conditions. Once all the segments are propagated, the values of the objective function and constraints are computed.

When uncertainties are introduced, the pointwise state $\mathbf{x}_k$ becomes a set, and thus a propagation method that can handle sets of values is required. In Section 2.3, we proposed the use of GPA to model and propagate the state variables as functions of the uncertain variables [110, 111]. Hence, here GPA is incorporated in a direct multiple shooting scheme to solve Problem (4.2). The resulting approach is named *IPANeMA* (Intrusive Polynomial Algebra aNd Multiple shooting Approach).

The goal of the direct transcription method proposed in this section is to evaluate the expectations of the objective function and constraints in problem (4.2) resulting from a given control $\mathbf{u}(t)$. The standard direct multiple shooting scheme is modified so that GPA is used to propagate the control law and the effect of uncertainty on each time segment $[t_k, t_{k+1}]$.

The reinitialisation approach presented in Section 2.3.3 is employed with GPA to construct a composite polynomial mapping in the form

$$\widetilde{F}_0^t(\boldsymbol{\xi}) = P_{\mathbf{x}}(t, \boldsymbol{\xi}) = P^{(p)}_{\mathbf{x}_t}(\dots(P^{(p)}_{\mathbf{x}_2}(P^{(p)}_{\mathbf{x}_1}(\mathbf{x}_0, \mathbf{d}_0), \mathbf{d}_1), \dots), \mathbf{d}_k) \ , \tag{4.3}$$

that is a sequential evaluation of a cascade of polynomial surrogates that maps the uncertain initial conditions and model parameters to the state vector at any time $t \in (t_k, t_{k+1}]$.

The composite polynomial formula is the equation of the direct multiple shooting based on GPA. Eq. (4.3) intrinsically guarantees continuity at the boundaries of each time interval thanks for the outer reinitialisation strategy, and maintain the dimensionality of the uncertainty space on each and any time interval equal to $n_{\xi_k} = n_x + n_{d_k}$. Note that using an outer set approximation removes the need for explicit defect constraints and free state variables at the start of each time interval, reducing the dimensionality of the associated constrained optimisation problem. The only free variables to be optimised are the control parameters in each sub-segment.

Objective and constraints computation

With the composite polynomial mapping in Eq. (4.3) the set of uncertain states can be recovered at any time $t \in [t_0, t_f]$ via the simple evaluation of a polynomial through sampling. These samples can be exploited to compute the general expectation form as in Eq. (2.25). The expected value of objective and constraints can be computed either by direct Monte Carlo or by importance sampling (see Section 2.4.1). Hence, the sampling in steps 1 and 4 of Algorithm 2 can be performed either directly from the corresponding density function or using an importance distribution. Then, the expectations associated with the objective and constraints can be computed by plugging the propagated samples in the relevant formula.

Clearly, other techniques could be applied to the computation of the expectation, but Monte Carlo methods have been selected because: they are not restricted to specific families of probability distributions; their convergence rate to the true expected value is independent of the uncertainty space dimensionality [149], making it appropriate for realistic, practical applications.

Depending on the uncertain distribution and the degree of nonlinearity of the equations of motion, the number of samples required to estimate the expectation accurately may be rather high. Furthermore, the transcription is called numerous times during the optimisation process. Therefore, the employment of an inexpensive polynomial mapping in place of direct numerical propagation plays a crucial role in the practical application of *IPANeMA* to realistic scenarios.

From the approach presented, it follows that the proposed shooting generalisation to the uncertain scenario is a natural extension of the classical deterministic transcription. A single propagation is replaced by multiple trajectory realisations weighted according to the uncertainty probability distribution to compute the objective and constraints values.

Representation of the control set

In order to avoid a new polynomial propagation each time a new control law $\mathbf{u}(t)$ is generated by the NLP solver, we expanded also the controls with GPA. Specifically, the set of control parameters $\mathbf{\Omega}_{\boldsymbol{\beta}_k}$ can be approximated by a time-static multivariate polynomial $P_{\boldsymbol{\beta}_k} \in T'_{q,n_{\xi_k}}$. The number of uncertain variables $n_{\xi_k} = n_x + n_{d_k} + n_{\boldsymbol{\beta}_k}$ is increased accordingly. Then, the set of control profiles in each time interval descends from the parameter-control relationship $P_{\mathbf{u}_k}(t) = \mathcal{U}_k(t, P_{\boldsymbol{\beta}_k})$, where the composition rules defined in the GPA apply. Once the expansion is computed, both the control profile and the associated set of propagated states can be computed with a single polynomial evaluation for each $\boldsymbol{\beta}_k \in \mathbf{\Omega}_{\boldsymbol{\beta}_k}$. With this procedure, only one uncertainty polynomial propagation is needed, and it can be precomputed before the optimisation cycle. Consequently, updating the set of propagated states for a new vector of the control parameters comes at a very low computational cost.

On the other hand, increasing the input dimensionality causes an higher computational complexity for the intrusive propagation as well as an higher polynomial degree to achieve a given representation accuracy. These two aspects make the trade-off between the computational cost of constructing the polynomial map and the representation accuracy to compute reliable metrics even more critical. Nonetheless, including the control variables in the polynomial map is always advantageous from a computational standpoint as the optimisation is then carried out over a sequence of polynomials. The representation accuracy needs to be assessed after the intrusive propagation in each test case and, if not satisfactory, the degree of the polynomials needs to be increased. In Chapter 7, the polynomial accuracy is verified against numerical propagations for the robust trajectory analysed.

Smoothing of the indicator function

When a gradient-based method is employed to solve the NLP problem, one practical difficulty arises from the computation of objective and constraints in probability. Although probability constraints (and objectives) are an intuitive and general way to impose conditions on random variables, they require evaluating the indicator function, which is discontinuous in nature. This discontinuity introduces a significant difficulty when computing the derivatives. In the general case, one cannot guarantee the differentiability of the expectation with respect to the controls. Even more so when the expectation is computed with a sample-based numerical technique.

To overcome this difficulty, *IPANeMA* implements the option to substitute the indicator function with a smoother approximation obtained by convolution [147]. Convolution is applied by first redefining the indicator function with the following membership condition: $\eta_A : \mathbb{R}^{n_x} \to \mathbb{R}$ such that:

$$\begin{cases} |\eta_A(\mathbf{x})| \leq 1 \quad \text{if } \mathbf{x} \in A \\ |\eta_A(\mathbf{x})| > 1 \quad \text{if } \mathbf{x} \notin A \; . \end{cases} \tag{4.4}$$

Hence, the indicator function can be equivalently expressed as

$$\mathbb{I}_A(\mathbf{x}) = \mathbb{I}_{[-1,+1]}(\eta_A(\mathbf{x})) \; . \tag{4.5}$$

Now, if we introduce the smoothing function h, for each state realisation $\mathbf{x}$, the convolution of the indicator function with h results in:

$$\begin{aligned}\mathbb{I}^{(r)}_{[-1,+1]}\big(\eta_A(\mathbf{x})\big) &= \int_{-\infty}^{+\infty} \mathbb{I}_{[-1,+1]}(y)\frac{1}{r}h\left(\frac{\eta_A(\mathbf{x})-y}{r}\right)dy \\ &= \int_{-1}^{+1} \mathbb{I}_{[-1,+1]}(y)\frac{1}{r}h\left(\frac{\eta_A(\mathbf{x})-y}{r}\right)dy \ ,\end{aligned} \tag{4.6}$$

with $r > 0$ a small positive scaling parameter. The integration interval is restricted to the interval $[-1, +1]$ because of the membership function η_A. The smoothing function h is chosen so that $h : \mathbb{R} \to \mathbb{R}$, h is non-negative, symmetric, with a unique maximum in 0, and integral equal to 1 over the interval $[-1, 1]$. These properties imply $\lim_{r\to 0} h(\cdot/r)/r = \delta$, with δ the Dirac delta. Hence, for $r \to 0$ the convolution result tends to the original indicator function [147].

Finally, the sampling grid used to compute the smoothed indicator function is kept constant within one major iteration of the NLP solver. This is particularly important when derivatives are approximated with finite differences to avoid introducing noise in the computation of the gradients of objective and constraints.

Numerical solution of the NLP problem

The NLP problem resulting from the transcription approach implemented in *IPANeMA* is dense and low-dimensional, as no intermediate state vectors and continuity constraints are present, as explained above, and the control parameters $\boldsymbol{\beta}_k$ for each time segment are the only free variables. In the current C++ implementation, this NLP problem is solved using WORHP [150].

4.2 Belief optimal control

In this section, the dOCP (4.1) is employed again as starting formulation. In the high-thrust case, the controls $\mathbf{u}(t)$ are to be intended in the form:

$$\mathbf{u}(t) = \sum_i \mathbf{u}_i \delta(t - t_i) \tag{4.7}$$

with δ the Dirac delta function and $\mathbf{u}_i \in \mathbb{R}^u$.

When boundary conditions and system parameters are affected by uncertainty, problem (4.1) needs to be again recast in a form that allows one to derive a control law that is optimal with respect to a metric that accounts for the effect of uncertainty. In the remainder of this section, we propose a belief formulation of optimal control problems that directly work with the probability distributions of the uncertain quantities and incorporate system-level and navigation uncertainties in the derivation of optimal control laws.

Preliminary definitions

In this section, we define several elements required to introduce the belief formulation of optimal control problems.

- *Parameter uncertainty.* Parameter uncertainty refers again to uncertainty in initial conditions and system parameters. We will now specialise Eq. (2.2) for such uncertainties and write

$$\mathbf{X}_0 \sim p(\mathbf{x}_0; \boldsymbol{\lambda}_{x_0}) \in \mathcal{P}_{X_0;\boldsymbol{\lambda}_{x_0}} \tag{4.8}$$

$$\mathbf{D} \sim p(\mathbf{d}; \boldsymbol{\lambda}_d) \quad \in \mathcal{P}_{\mathcal{D};\boldsymbol{\lambda}_d} \tag{4.9}$$

  to indicate that the stochastic variables $\mathbf{X}_0$ and $\mathbf{D}$ have probability distributions $p(\mathbf{x}_0; \boldsymbol{\lambda}_{x_0})$ and $p(\mathbf{d}; \boldsymbol{\lambda}_d)$ and that these pdfs belong to the two families $\mathcal{P}_{X_0;\boldsymbol{\lambda}_{x_0}}$ and $\mathcal{P}_{\mathcal{D};\boldsymbol{\lambda}_d}$ parametrised in $\boldsymbol{\lambda}_{x_0}$ and $\boldsymbol{\lambda}_d$, respectively. In the remainder of the section, we will present the methodology starting from the treatment of epistemic uncertainty, under the assumption of a known family of distributions, and then derive the solution for aleatory uncertainty as a special case for $\boldsymbol{\lambda}$ precisely known.

- *Execution errors.* The general form of the *executed* control $\mathbf{U}_e(\mathbf{u}, \mathbf{X}) : \mathbb{R}^{n_u} \times \mathbb{R}^{n_x} \rightarrow \mathbb{R}^{n_u}$ considered is a random variable defined as

$$\mathbf{U}_e(\mathbf{u}, \mathbf{X}) = \sum_i [\overline{\mathbf{u}}_i + \delta\mathbf{u}_i(\mathbf{X}) + \boldsymbol{\Theta}_i(\mathbf{u}_i + \delta\mathbf{u}_i)]\delta(t - t_i) \ , \tag{4.10}$$

  where $\overline{\mathbf{u}}_i$ and $\delta\mathbf{u}_i$ are the *commanded* open- and closed-loop control terms respectively, and $\boldsymbol{\Theta}_i$ models the control errors. The sum of the commanded components will be denoted as $\mathbf{u}_i = \overline{\mathbf{u}}_i + \delta\mathbf{u}_i$. In this form, we assumed that the control uncertainty affects only the components of the manoeuvre and not the time of the execution and is a function of the commanded control. The meaning and form of the feedback component and errors will be better defined in the remainder. As discussed in Section 2.1, for ease of notation, the uppercase special character $\mathcal{X}$ will be used to indicate the pdf of the state $\mathbf{X}$, and similarly for other random variables.

- *Observation uncertainty.* Observations are employed to reduce the knowledge uncertainty associated with the system state. The observation model is a nonlinear function of the state and environment noise realisations $\boldsymbol{\varepsilon}_k$:

$$\mathbf{y}_k = h(\mathbf{x}_k, \boldsymbol{\varepsilon}_k) \ . \tag{4.11}$$

Hence, a generic observation $\mathbf{y}_k \in \mathbb{R}^{n_y}$ is a realisation of the random variable $\mathbf{Y}_k : \Omega_{\mathcal{E}_k} \to \mathbb{R}^{n_y}$ induced by the sensor noise $\mathcal{E}_k$. The random variable observation is described by the conditional likelihood

$$\mathbf{Y}_k \sim p(\mathbf{y}_k | \, \mathbf{x}_k; \boldsymbol{\lambda}_y) \in \mathcal{P}_{\mathcal{Y};\boldsymbol{\lambda}_y} \ . \tag{4.12}$$

This relation expresses that, given the state, the observation likelihood $\mathcal{Y}_k = p(\mathbf{y}_k | \, \mathbf{x}_k; \boldsymbol{\lambda}_y)$ is completely determined by the sensor noise $\mathcal{E}_k$ [151]. At each observation k, the state distribution is updated given the last received observation according to an inference rule $\mathcal{T}_I$ which returns the posterior distribution, conditional on all the observations received until time t_k, as

$$\mathcal{X}_k = \mathcal{T}_I\left(\mathcal{X}_k^-, \mathcal{Y}_k\right) = p(\mathbf{x}_k | \mathbf{y}_{1:k}) \ , \tag{4.13}$$

given the predicted state uncertainty $\mathcal{X}_k^- = p(\mathbf{x}_k | \mathbf{y}_{1:k-1})$ and the last observation $\mathbf{y}_k$. The superscript $(\cdot)^-$ indicates the predicted distribution right before the new observation $\mathbf{y}_k$ is received.

- *Dynamical evolution.* $\mathbf{X}_0$ and $\mathbf{D}$, together with the dynamical equation (4.1b), induce the state at a later time to be a random variable through the push-forward measure resulting from the pointwise dynamical flow. As already seen, the distribution of the random variable $\mathbf{X}$ evolves according to a PDE, written in the general notation

  $$\partial_t \mathcal{X} - \mathcal{F}_x(t, \mathcal{X}, \mathcal{D}, \mathcal{U}_e) = 0 \ , \tag{4.14}$$

  where the term $\mathcal{F}_x$ includes the partial derivatives with respect to the state variables, and it depends on the pointwise dynamics (4.1b). Eq.(4.14) is the Fokker-Planck (or Forward Kolmogorov) equation [152] (see Section 2.2). Epistemic uncertainty on either of the sources of uncertainty induces, through the dynamical evolution, the state distribution at a later time to be set-valued itself as

  $$\mathbf{X} \sim \mathcal{X} = p(\mathbf{x}; \boldsymbol{\lambda}) \in \mathcal{P}_{\mathcal{X};\boldsymbol{\lambda}} \ , \tag{4.15}$$

  where $\boldsymbol{\lambda} = [\boldsymbol{\lambda}_{x_0}, \boldsymbol{\lambda}_d, \boldsymbol{\lambda}_y]$ is the collection of all the epistemic parameters.

4.2.1 Belief formulation

Given the elements defined in the previous section, the optimal control problem under uncertainty is well described as a Partially Observable Markov Decision Process [153], i.e. the state is observed only through indirect measurements. This model can be re-framed as a Belief Markov Decision Process (BMDP) [154], which employs an advantageous belief state representation. That is, the state of the model is not a specific realisation $\mathbf{x}$, but rather the state is the density function $\mathcal{X}$. The probability distribution of the dynamical system state is henceforth called *belief* state. An advantage of such a formulation is that the belief state can be computed at any time, even if the specific state realisation is not observable. We write the Belief Optimal Control (BOC) as

$$\min_{\bar{\mathbf{u}}} \sup_{\boldsymbol{\lambda}} \left\{ C_f\big(t_f, \mathcal{X}_f\big) + \int_{t_0}^{t_f} C\big(t, \mathcal{X}, \mathcal{U}_e\big)\, dt \right\} \tag{4.16a}$$

$$\text{s.t.} \quad \begin{cases} \partial_t \mathcal{X} - \mathcal{F}_x(t, \mathcal{X}, \mathcal{D}, \mathcal{U}_e) = 0 & \text{between observation times} \\ \mathcal{X}_k = \mathcal{T}_I\big(\mathcal{X}_k^-, \mathcal{Y}_k\big) & \text{at each observation } k \end{cases} \tag{4.16b}$$

$$\underline{G}\big(t, \mathcal{X}, \mathcal{D}, \mathcal{U}_e\big) \in \boldsymbol{\Psi}_{\underline{G}}\ , \qquad \overline{G}\big(t, \mathcal{X}, \mathcal{D}, \mathcal{U}_e\big) \in \boldsymbol{\Psi}_{\overline{G}} \tag{4.16c}$$

$$\underline{B}\big(t_0, \mathcal{X}_0, t_f, \mathcal{X}_f\big) \in \boldsymbol{\Psi}_{\underline{B}}\ , \quad \overline{B}\big(t_0, \mathcal{X}_0, t_f, \mathcal{X}_f\big) \in \boldsymbol{\Psi}_{\overline{B}} \tag{4.16d}$$

$$\mathcal{X}_0 \in \mathcal{P}_{\mathcal{X}_0;\boldsymbol{\lambda}_{x_0}},\ \ \mathcal{D} \in \mathcal{P}_{\mathcal{D};\boldsymbol{\lambda}_d},\ \ \mathcal{Y}_k \in \mathcal{P}_{\mathcal{Y}_k;\boldsymbol{\lambda}_y} \tag{4.16e}$$

where, C_f and C are the functionals (2.3) associated respectively to Φ and L, $\underline{G}$, $\overline{G}$, $\underline{B}$, $\overline{B}$, are the lower and upper limits defined in (2.4a) and (2.4b) on the functionals (2.3) associated to the path and boundary constraints $\mathbf{g}$ and $\mathbf{b}$. The target sets $\boldsymbol{\Psi}_{\underline{G}}$, $\boldsymbol{\Psi}_{\overline{G}}$, $\boldsymbol{\Psi}_{\underline{B}}$ and $\boldsymbol{\Psi}_{\overline{B}}$ for the expectations on $\mathbf{g}$ and $\mathbf{b}$ are predefined quantities. The objective in (4.16a) is the upper bound of the objective function realisations under uncertainty. Eqs. (4.16b) describe how the belief state evolves in time according to a PDE and updates with an inference rule each time an observation is acquired. They are the dual of the equations of motion (4.1b) in the dOCP. Finally, Eq.(4.16e) describes the uncertainty structure of the problem, namely expressing the initial condition for the belief state, the uncertain parameter distribution and the observation likelihood. The BOC aims at optimising the open-loop component to find the nominal trajectory which minimises the objective function and satisfies the constraints under uncertainty.

In general, the BOC problem (4.16) has no closed-form solution, just like the deterministic dOCP (4.1). Thus, in the following section, we propose a direct transcription method that allows incorporating navigation analysis in the optimisation of the control law under the type of epistemic and observation uncertainty defined in this section.

4.2.2 Transcription method

Solving problem (4.16) requires propagating the belief state from the initial conditions through the dynamics and updating it with observations. This section presents an efficient multiple shooting-like direct transcription method for the solution of problem (4.16), which employs a direct propagation of the belief state. This work generalises the direct transcription method developed in [87] to the case in which orbit determination arcs intermingle control arcs.

Sequential belief transcription

Following the same idea of general multiple shooting schemes, we start by partitioning the independent variable domain into segments. In this case we partition the time domain in the following time segments:

$$[t_k, t_{k+1}] \quad \text{for } k = 0, 1, \ldots, F-1 \ . \tag{4.17}$$

On each segment we define a vector of commanded control parameters $\mathbf{u}_k$ in the following form:

$$\mathbf{u}_k(t_k, \mathbf{X}_k) = \overline{\mathbf{u}}_k(t_k) + \delta\mathbf{u}_k\big(\mathbf{X}_k\big) \tag{4.18}$$

where the first term is an unknown parameter to be optimised, whereas the second one is a correction to $\overline{\mathbf{u}}_k(t_k)$ coming from a pre-defined function of the state at time t_k. Operationally speaking, the term $\overline{\mathbf{u}}_k(t_k)$ has to be understood as an open-loop control while $\delta\mathbf{u}_k\big(\mathbf{X}_k\big)$ as a feedback control policy. In addition, on each segment we define a vector of uncertain parameters $\mathbf{D}_k$ such that:

$$\mathbf{D} = [\mathbf{D}_0, \mathbf{D}_1 \ldots, \mathbf{D}_k, \ldots] \ .$$

Again, the assumption is that each $\mathbf{D}_k$ affects the dynamics only over the time interval $t \in (t_k, t_{k+1}]$. Uncertain model parameters which affect the dynamics over multiple time segments are replicated across all the segments they affect. From (4.10) the executed controls $\mathbf{U}_{e_k}$ at time t_k is:

$$\mathbf{U}_{e_k} = \mathbf{u}_k(t_k, \mathbf{X}_k) + \boldsymbol{\Theta}_k(\mathbf{u}_k(t_k, \mathbf{X}_k), \boldsymbol{\lambda}_k) \ , \tag{4.19}$$

which combines the commanded controls with execution errors.

For each time segment, the belief state $\mathcal{X}_k$ is first propagated from t_k to t_{k+1} under the effect of the dynamics (4.1b) and the possible control actions $\mathbf{u}_k$, if they are present on that segment. We indicate this propagation with the symbol $\mathcal{T}_P$ that represents the mapping of the probability distribution $p(\mathbf{x}_k \,|\, \mathbf{y}_{1:k})$ from t_k to t_{k+1}:

$$\mathcal{X}_{k+1}^{-} = \mathcal{T}_P\big(\mathcal{X}_k, \mathcal{D}_k, \mathcal{U}_{e_k}\big) = p(\mathbf{x}_{k+1} \,|\, \mathbf{y}_{1:k}) \ . \tag{4.20}$$

We now consider the case of observation instances at the end of some sub-intervals $[t_k, t_{k+1}]$. Thus, suppose now that a new measurement $\mathbf{y}_{k+1}$ is available. By using Bayes' inference rule one can calculate the posterior distribution:

$$\mathcal{X}_{k+1} = \mathcal{T}_I\big(\mathcal{X}_{k+1}^{-}, \mathcal{Y}_{k+1}\big) = p(\mathbf{x}_{k+1} \,|\, \mathbf{y}_{1:k+1}) = \frac{p(\mathbf{y}_{k+1}|\mathbf{x}_{k+1})\ p(\mathbf{x}_{k+1} \,|\, \mathbf{y}_{1:k})}{\int p(\mathbf{y}_{k+1}|\mathbf{x}_{k+1})\ p(\mathbf{x}_{k+1} \,|\, \mathbf{y}_{1:k})\, d\mathbf{x}_{k+1}} \ . \tag{4.21}$$

Again, the dependencies on the deterministic parameters have not been written explicitly. We assume that the measurement $\mathbf{y}_{k+1}$ is conditionally independent of the observation and control history, given the state at time t_{k+1}, thus those terms disappear from the observation likelihood $p(\mathbf{y}_{k+1}|\mathbf{x}_{k+1})$.

Hence, the calculation of the belief state at time t_{k+1} can be written, in compact form, as the composition of the state propagation map $\mathcal{T}_P$ and the Bayes inference map $\mathcal{T}_I$:

$$\mathcal{X}_{k+1} = \mathcal{T}\big(\mathcal{X}_k, \mathcal{D}_k, \mathcal{U}_{e_k}, \mathcal{Y}_{k+1}\big) = \mathcal{T}_I \circ \mathcal{T}_P \ . \tag{4.22}$$

Eq. (4.22) models the most general case of a segment where both control actions at t_k and measurements at t_{k+1} are present. In the following, arcs can have both control and observations, only control actions, only observations or neither of the two (pure propagation). Depending on the specific case, we will apply either the propagation map alone, the inference map or the composition of the two.

One interesting feature of the belief formulation is that, while the system pointwise state would dynamically evolve through a one-to-many relationship in a standard Markov Decision Process, the belief state of the Belief Markov Decision Process evolves through the prediction and update steps according to a one-to-one relationship.

Being the uncertain parameters partitioned as $\mathbf{D} = \big[\mathbf{D}_0, \ldots, \mathbf{D}_k, \ldots, \mathbf{D}_{F-1}\big]$, the main advantage of the shooting scheme is that it decouples the uncertainty in the different time segments $[t_k, t_{k+1}]$. When the belief is propagated from t_k to t_{k+1}, the stochastic dimensionality is $n_{\xi_k} = n_x + n_{d_k}$, where n_x and n_{d_k} are respectively the state and uncertain parameters $\mathbf{D}_k$ dimensionality, instead of being the total $n_\xi = n_x + n_d$ as it would be with a vanilla Monte Carlo approach. This efficient decoupling avoids the accumulation of uncertainty and the growth of the belief stochastic dimension in time, thus helping to contain the curse of dimensionality, typical of uncertainty quantification problems. This feature is computationally crucial for an uncertainty quantification method called within an optimisation loop numerous times.

The propagation map $\mathcal{T}_P$

Propagating the belief state in time is the most computationally intensive step of the transcription method as it would require solving a PDE which has no closed-form solution in the general nonlinear case.

The BOC formulation, however, permits the propagation of the exact dynamics and, in the general case, of a family of probability distributions. Thus we split the mapping $\mathcal{T}_P$ in typical two steps adopted in this thesis: first we create a non-intrusive polynomial representation $\widetilde{F}_k^{k+1}$ of all possible states at time t_{k+1} given the states at time t_k, the uncertain parameters and the control variables as described in Section 2.3; then we sample the distribution at time t_k to represent the distribution at time t_{k+1}. This second step requires only multiple evaluations of the polynomial at time t_{k+1}. Let us shortly recall how the polynomial mapping is represented and adapt it to the case under consideration.

To assess the surrogate accuracy, one option is to construct a polynomial representation by using a collocation grid of level $\bar{l}$. Then, the accuracy can be quantified as the root mean square error of the polynomial mapping versus the numerical one on a subset of samples of grid level $\bar{l} + 1$. If such error is below a user-set threshold, the polynomial representation is accepted, otherwise the polynomial is re-constructed with a collocation grid of level $\bar{l} + 1$ and the process iterated until the requested accuracy is met.

Once the polynomial $\widetilde{F}_k^{k+1}$ is available, the output of the propagation map $\mathcal{T}_P$ can be approximated as

$$\mathcal{X}_{k+1}^- \approx p\big(\widetilde{F}_k^{k+1}(\boldsymbol{\xi}_k) \,|\, \mathbf{y}_{1:k}\big) \ . \tag{4.23}$$

Navigation analysis and the inference map $\mathcal{T}_I$

Treating observations in offline trajectory optimisation is a subtle task. Indeed, a measurement is a random function of the true state of the system and sensor noise, which is, however, unknown at the time of offline planning. Therefore, simulating a specific observation value $\mathbf{y}_k$, and its corresponding likelihood, rather than another value, is an arbitrary choice corresponding to an arbitrary reduction in the state uncertainty (through the *update* equations). This problem is depicted in Figure 4.1, where a single observation reduces the uncertainty dispersion. Specifically, Figure 4.1(a) is the uncertainty propagation without any observation. Figure 4.1(b) depicts the generation of a single observation. Figure 4.1(c) shows the uncertainty update of the belief $\mathcal{X}_k$ given the new observation. Figure 4.1(d) is the new belief propagation $\mathcal{X}_{k+1}^-$ after the new observation has been processed. While reducing the uncertainty is the goal of orbit determination in the operational life of a spacecraft, that is, when actual measurements are available, when performing robust trajectory optimisation, this simulation approach leads to discarding entirely feasible regions of uncertainty.

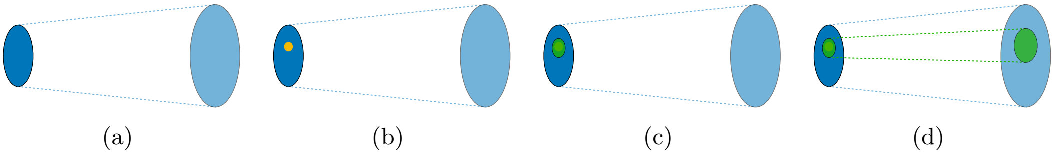

(a) (b) (c) (d)

Figure 4.1: Effect of single measurement simulation on the uncertainty propagation process.

Hence, the inference map $\mathcal{T}_I$ comes from the simulation of the expected observations along the trajectory considering observation errors. Indeed, for a given observation at time t_k, the measurement $\mathbf{y}_k$ is conditional to the true state $\mathbf{x}_k$ of the spacecraft, which is unknown. Thus one needs to simulate a set of possible measurements over the space of possible predicted states [43, 44].

Considering M measurement realisations $\mathbf{y}_k^{(j)}$, with $j = 1, \dots, M$, each with an associated probability density, an inference step is required for each of them. This results in a set of possible posterior beliefs

$$\mathcal{X}_k^{(j)} = p\left(\mathbf{x}_k \,|\, \mathbf{y}_{1:k}^{(j)}\right) \ , \tag{4.24}$$

each resulting from a different observation realisation. Such a scenario is depicted in Figure 4.2, where multiple belief states are generated as a consequence of the inference with multiple sample measurements. Specifically, Figure 4.2(a) depicts the uncertainty propagation without any observation. Figure 4.2(b) indicates the generation of multiple observation samples by Monte Carlo. Figure 4.2(c) shows the multiple uncertainty updates of the belief $\mathcal{X}_k$ given the Monte Carlo observation samples. Figure 4.2(d) represents the belief components propagation after the new observations have been processed. Hence, we first apply the inference map (4.21) M times, one for each $\mathbf{y}_k^{(j)}$ at time t_k, then we use map (4.23) to propagate each posterior to obtain M distributions at time t_{k+1}.

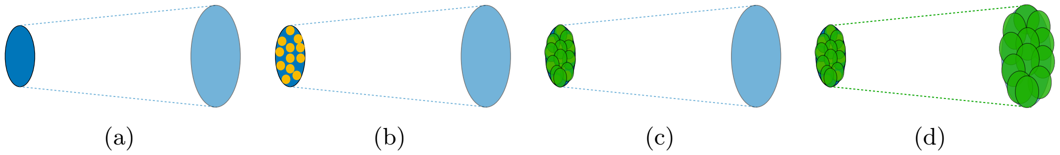


Figure 4.2: Effect of Monte Carlo measurement simulation on the uncertainty propagation process.

If this process was iterated for each segment, one would have an exponential growth of the number of expected trajectories and associated control laws. Indeed, if we were to simulate M new observations for each belief component at time t_{k+1}, we would end up with M^2 belief components, which would grow to M^3 at t_{k+2}, and so on. On the contrary, once the M distributions at time t_{k+1} are available we build the following convex combination:

$$\begin{aligned} \mathcal{X}_{k+1}^{-} &= \sum_{j=1}^{M} b^{(j)}\, \mathcal{X}_{k+1}^{-(j)} \\ \sum_{j=1}^{M} b^{(j)} &= 1 \ , \end{aligned} \tag{4.25}$$

where $b^{(j)}$ is the *belief degree*, a weight quantifying the relative degree of likeliness of each $\mathcal{X}_{k+1}^{(j)}$. The value of each $b^{(j)}$ depends on the credibility of each measurement $\mathbf{y}_k^{(j)}$. Then, we use this belief state as prior at time t_{k+1} to be updated with the M simulated likelihoods. With this approach, the number of belief components is kept constant to M regardless of the number of shooting segments.

The convex combination in Eq. (4.25) can be derived from the concept of Jeffrey conditionalisation [155]. This conditionalisation rule describes how the probability of the occurrence of an event A depends on the realisation of events $B^{(j)}$ with confidence belief degrees $b^{(j)}$:

$$\Pr\big(A \,|\, B^{(1)} \equiv b^{(1)}, \dots, B^{(M)} \equiv b^{(M)}\big) = \sum_{j=1}^{M} b^{(j)} \Pr\big(A \,|\, B^{(j)}\big) \ , \tag{4.26}$$

when the condition events form a partition. This conditionalisation generalises the traditional conditional probability measure, which is now a special case when event $B^{(j)}$ has been observed with certainty, i.e. when its belief degree is $b^{(j)} = 1$, and therefore the belief degree of its conjugate is zero. Given this probability measure, we can interpret the belief in Eq. (4.25) as an overall inference step by Jeffrey conditionalisation over sampled observations $\mathbf{y}_k^{(j)}$ with belief degrees $b^{(j)}$:

$$\mathcal{X}_k = p\big(\mathbf{x}_k \,|\, \mathbf{y}_k^{(1)} \equiv b^{(1)}, \dots, \mathbf{y}_k^{(M)} \equiv b^{(M)}\big) = \sum_{j=1}^{M} b^{(j)}\, p\big(\mathbf{x}_k | \mathbf{y}_k^{(j)}\big) \tag{4.27}$$

The problem is that, in general, the sampled observations do not form a partition of the observation space, which is indeed continuous. However, we can approximate any continuous distribution as a probability mass function by using samples drawn from the original one and the Dirac delta function as [30]

$$p(\boldsymbol{\xi}) \approx \sum_{j=1}^{M} b^{(j)} \delta\big(\boldsymbol{\xi} - \boldsymbol{\xi}^{(j)}\big) \ . \tag{4.28}$$

If such approximation is introduced for the observation likelihood, then the Jeffrey conditionalisation provides a key formal interpretation of the employed inference step with different sampled measurements for the navigation analysis approach considered.

Solution of the transcribed problem

Once the belief state is propagated at each stage k with the approximated propagation map $\mathcal{T}_P$ and the inference map $\mathcal{T}_I$ is applied when observations are available, Problem (4.16) is transcribed into the following nested NLP problem:

$$\begin{aligned}
&\min_{\bar{\mathbf{u}}} \sup_{\boldsymbol{\lambda}} \left\{ C_f\big(t_f, \mathcal{X}_f\big) + \sum_k \sum_s w_s C\big(t_s, \mathcal{X}_k, \mathcal{U}_{e_k}\big) \right\} \\
&\quad \text{s.t.} \begin{cases} \mathcal{X}_{k+1}^{-} = \mathcal{T}_P\big(\mathcal{X}_k, \mathcal{D}_k, \mathcal{U}_{e_k}\big) & \text{between observation times} \\ \mathcal{X}_k = \mathcal{T}_I\big(\mathcal{X}_k^{-}, \mathcal{Y}_k\big) & \text{at observation instances} \end{cases} \\
&\qquad\quad \underline{G}_k\big(t_k, \mathcal{X}_k, \mathcal{D}_k, \mathcal{U}_{e_k}\big) \in \boldsymbol{\Psi}_{\underline{G}_k}\,, \quad \overline{G}_k\big(t_k, \mathcal{X}_k, \mathcal{D}_k, \mathcal{U}_{e_k}\big) \in \boldsymbol{\Psi}_{\overline{G}_k} \\
&\qquad\quad \underline{B}\big(t_0, \mathcal{X}_0, t_f, \mathcal{X}_f\big) \in \boldsymbol{\Psi}_{\underline{B}}\,, \qquad \overline{B}\big(t_0, \mathcal{X}_0, t_f, \mathcal{X}_f\big) \in \boldsymbol{\Psi}_{\overline{B}} \\
&\qquad\quad \mathcal{X}_0 \in \mathcal{P}_{\mathcal{X}_0}, \ \ \mathcal{D} \in \mathcal{P}_{\mathcal{D}}, \ \ \mathcal{Y}_k \in \mathcal{P}_{\mathcal{Y}_k;\boldsymbol{\lambda}_y}
\end{aligned} \tag{4.29}$$

where w_s are quadrature weights used to discretise the objective integral in time. The outer optimisation is carried out on the transcribed open-loop control components to find the most robust and reliable nominal trajectory. The quantities C_f,C, G_k, B are integrals over the space of the uncertain parameters as in Eq. (2.3). In the general case, these integrals need to be computed numerically, either by sampling or with a numerical quadrature formula, as

$$\mathbb{E}_{p_{\boldsymbol{\lambda}_z}}[\psi(\mathbf{z})] = \int_{\boldsymbol{\Omega}_{\mathbf{Z}}} \psi(\mathbf{z})\, p(\mathbf{z}; \boldsymbol{\lambda}_z)\, d\mathbf{z} \approx \sum_j w_j(\boldsymbol{\lambda})\, \psi(\mathbf{z}^j) \ . \tag{4.30}$$

For given control vector and epistemic parameter to evaluate, the transcription complexity depends primarily on the number of aleatory variables and observation arcs. The number of grid samples to construct the expansion by sparse collocation scales polynomially with the number of aleatory dimensions, whereas the complexity of evaluating the surrogate for samples' propagation is secondary. Updating the belief state requires only simple evaluations of the likelihood function and low-dimensional matrix operations, as discussed in the next sections. Hence, the main computational and accuracy advantages of the transcription come from the shooting discretisation. This transcription enables an efficient decoupling of the uncertainties and, therefore, lower aleatory dimensions in each segment. Thus, adding new control or observation instances does not yield an exponential increase in computational complexity.

The two-level NLP problem is solved by nested local optimisations. In the general case of epistemic uncertainty, for each integral (4.30) we need to solve an optimisation problem over the space of the parameters $\boldsymbol{\lambda}$. To alleviate the computational burden of these optimisations, the polynomial mapping is constructed only once per iteration of the minimisation over the space of the controls, and the optimisations over $\boldsymbol{\lambda}$ are performed exploiting the inexpensive surrogate model. This is made possible by the fact that the polynomial representation covers the entire space of admissible controls. This approach causes the input set $\Omega_{\boldsymbol{\xi}_k}$ to be larger than in the purely aleatory case. Nonetheless, the collocation grid level can be tuned to ensure the required surrogate accuracy under epistemic uncertainty as well. Thus, propagation of uncertainty in the epistemic setting may imply a higher computational cost due to additional grid levels, whereas the representation accuracy is preserved.

The local optimisations rely on finite differences to compute the first- and second-order derivative information. When sampling is employed to compute Equation (4.30), the sampling grid is kept constant within one major NLP iteration to avoid introducing noise in the computation of the objective's gradient and constraints' Jacobian.

In the following, we will present two approaches to the computation of the integrals in the two cases in which the distribution functions are explicitly available or not. In the former case, we limit our attention to the treatment of Gaussian distributions.

The Gaussian case In the Gaussian case the distribution of the uncertain parameters and initial conditions is normal, and so is the likelihood of the measurements:

$$\mathcal{X}_0 = \mathcal{N}\left(\mathbf{x}_0;\, \boldsymbol{\mu}_{X_0},\, \boldsymbol{\Sigma}_{X_0}\right) \tag{4.31a}$$

$$\mathcal{D} = \mathcal{N}\left(\mathbf{d}\,;\, \boldsymbol{\mu}_{D},\, \boldsymbol{\Sigma}_{D}\right) \tag{4.31b}$$

$$\mathcal{Y}_k = \mathcal{N}\left(\mathbf{y}_k \,|\, \mathbf{x}_k;\, \boldsymbol{\mu}_{Y_k},\, \boldsymbol{\Sigma}_{Y_k}\right), \tag{4.31c}$$

where $\boldsymbol{\mu}_{(\cdot)}$ indicates the mean and $\boldsymbol{\Sigma}_{(\cdot)}$ the covariance of the normal distributions. The posterior distribution (4.22) at each stage k is generally not Gaussian. However, under certain conditions, one can approximate the actual posterior with only the first two statistical moments [130] as it is commonly done in Kalman-type of sequential filters. Thus we can approximate the prior at stage k with:

$$\mathcal{X}_k^- = \sum_{j=1}^{M} b^{(j)} \mathcal{N}\left(\mathbf{x}_k\,;\, \boldsymbol{\mu}^{(j)}_{X_k^-}, \boldsymbol{\Sigma}^{(j)}_{X_k^-}\right) \tag{4.32}$$

with each mean and covariance computed by Gauss-Hermite quadrature (see Section 2.4.2). The M observations $\mathbf{y}_k^{(j)}$ are sampled from the prior (4.32). The posterior

$$\mathcal{X}_k = \sum_{j=1}^{M} b^{(j)}\, \mathcal{N}\left(\mathbf{x}_k;\, \boldsymbol{\mu}^{(j)}_{X_k},\, \boldsymbol{\Sigma}^{(j)}_{X_k}\right) \tag{4.33}$$

is computed by updating the mean and the covariance matrix according to the observations received:

$$\begin{aligned} \boldsymbol{\mu}^{(j)}_{X_k} &= \boldsymbol{\mu}^{(j)}_{X_k^-} + \mathbf{K}_k(\mathbf{y}_k^{(j)} - \boldsymbol{\mu}_{Y_k}) \\ \boldsymbol{\Sigma}^{(j)}_{X_k} &= \boldsymbol{\Sigma}^{(j)}_{X_k^-} - \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^T \ . \end{aligned} \tag{4.34}$$

where the observation mean $\boldsymbol{\mu}_{Y_k}$ and Kalman gain $\mathbf{K}_k$ are defined as

$$\begin{aligned}
\boldsymbol{\mu}_{Y_k} &= \sum_{j=1}^{M} b^{(j)} \sum_{i=1}^{N} w^{(j,i)}\, h\big(\mathbf{x}_k^{(j,i)}, \boldsymbol{\varepsilon}_k^{(j,i)}\big) \\
\mathbf{S}_{Y_k} &= \sum_{j=1}^{M} b^{(j)} \sum_{i=1}^{N} w^{(j,i)} \big(h\big(\mathbf{x}_k^{(j,i)}, \boldsymbol{\varepsilon}_k^{(j,i)}\big) - \boldsymbol{\mu}_{Y_k}\big)\, \big(h\big(\mathbf{x}_k^{(j,i)}, \boldsymbol{\varepsilon}_k^{(j,i)}\big) - \boldsymbol{\mu}_{Y_k}\big)^T \\
\mathbf{C}_{XY_k} &= \sum_{j=1}^{M} b^{(j)} \sum_{i=1}^{N} w^{(j,i)} \big(\, \mathbf{x}_k^{(j,i)} - \boldsymbol{\mu}_{X_k}^{-} \,\big)\, \big(\, h\big(\mathbf{x}_k^{(j,i)}, \boldsymbol{\varepsilon}_k^{(j,i)}\big) - \boldsymbol{\mu}_{Y_k} \,\big)^T \\
\mathbf{K}_k &= \mathbf{C}_{XY_k}\, \mathbf{S}_{Y_k}^{-1} \ ,
\end{aligned} \tag{4.35}$$

with $\boldsymbol{\varepsilon}_k^{(j,i)}$ are realisations of the sensor noise, $\mathbf{S}_{Y_k}$ is the observation covariance, whereas $\mathbf{C}_{XY_k}$ is the cross covariance between state and observation.

From the posterior, one can then compute the expectation of any function ψ as

$$\mathbb{E}\big[\psi(\mathbf{x}_k) \,|\, \mathbf{y}_{1:k}\big] \approx \sum_{j=1}^{M} b^{(j)} \sum_{i=1}^{N} w^{(j,i)} \psi\big(\mathbf{x}_k^{(j,i)}\big) \ . \tag{4.36}$$

For ease of notation, the dependency on the control and parameter uncertainty has not been reported, but the same form applies when ψ depends on them by taking samples from their distributions.

The advantage of this approach over similar sample-based ones is that the estimation fidelity can be made as accurate as desired simply using more quadrature points, that is a higher-order Hermite polynomial. A high accuracy computation of the prior moments in Eq. (4.32) is made computationally possible by the polynomial mapping $\widetilde{F}_{k-1}^{k}$, which can be employed to propagate a great number of samples inexpensively.

General distribution case When the posterior distribution at stage k cannot be easily expressed with an explicit function of the state variables, the prediction and update steps of the belief transition function, in Eq. (4.22), can be computed with the particle filter described in Section 3.2. Specifically in this setting, the posterior density is approximated as a discrete one by using samples $\mathbf{x}_k^{(i)}$ drawn from the proposal

$\pi\left(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}, \mathbf{y}_{1:k}\right)$, as

$$\mathcal{X}_k^- = \sum_{j=1}^{M} b^{(j)} \sum_{i=1}^{N} w_k^{(i,j)}\, \delta\big(\mathbf{x}_k - \mathbf{x}_k^{(i,j)}\big)\ , \tag{4.37}$$

where $\delta(\cdot)$ is again the Dirac function, $w_k^{(i,j)}$ and $\mathbf{x}_k^{(i,j)}$ are the i-th weight and sample to approximate the j-th belief component. To construct the posterior approximation, the inference step is then carried out by updating the weights of the predicted samples as

$$w_k^{(i,j)} = w_{k-1}^{(i,j)}\, \frac{p\big(\,\mathbf{y}_k^{(j)} \,|\, \mathbf{x}_k^{(i,j)}\big)\ p\big(\mathbf{x}_k^{(i,j)} \,|\, \mathbf{x}_{k-1}^{(i,j)}\big)}{\pi\left(\mathbf{x}_k^{(i,j)} \,|\, \mathbf{x}_{k-1}^{(i,j)}, \mathbf{y}_{1:k}^{(j)}\right)}\ , \tag{4.38}$$

to account for the received observation $\mathbf{y}_k^{(j)}$ and the known dynamical evolution. Once again, the conditional dependencies on the deterministic parameters have not been written down explicitly for ease of notation. Nonetheless, the proposal, transition and likelihood distributions could depend on them.

From this posterior, one can then compute the expectation as

$$\mathbb{E}\big[\psi\big(\mathbf{X}_k\big) \,|\, \mathbf{y}_{1:k}\big] \approx \sum_{j=1}^{M} b^{(j)} \sum_{i=1}^{N} w_k^{(i,j)} \psi\big(\mathbf{x}_k^{(i,j)}\big)\ , \tag{4.39}$$

which is the same as Eq. (4.36)) but with weights and samples computed differently.

The accuracy of the discrete approximation and the particle filter performance greatly depend on the number of particles used, which is further critical for high-dimensional nonlinear problems like navigation analysis for space trajectory. Therefore, the polynomial mapping $\widetilde{F}_{k-1}^{k}$ results crucial in this scheme as it allows one to employ a larger number of samples, which can be propagated through an inexpensive polynomial evaluation.

This particle filter-based approach to the practical computation of the shooting transcription should be employed in the most general case of non-Gaussian uncertainty, nonlinear dynamical system and observation model. This concludes the transcription method for the BOC as formulated in Eq. (4.16).

4.3 Chapter summary

This chapter introduced two transcription methods for the solution of optimal control problems under uncertainties of different nature.

The proposed formulations can accommodate different forms of the quantities of interest and do not require any underlying assumption on the model used to quantify uncertainty. In fact, it was shown that it could also incorporate epistemic uncertainties and imprecision.

Specifically, Section 4.1 introduced *IPANeMA*, a numerical approach which integrates a multiple shooting numerical scheme with a GPA expansion to represent and propagate uncertainty regions. The reinitialisation and recovery approach, introduced in Chapter 2, was employed to limit the dimensionality of the uncertain space across multiple time segments, which intrinsically satisfies the continuity conditions between two adjacent segments and removes the need for additional constraints. A specific convolution approach was implemented to avoid discontinuities in the evaluation of objective and constraints and to enable the use of gradient-based methods. *IPANeMA* is capable of handling uncertainties both in the initial state and in the model parameters. Furthermore, it is suitable to work with a large variety of uncertainty models, hence it is not restricted to purely Gaussian, uniform or other basic probability distribution families.

Section 4.2 proposed a further development in optimal control problems under uncertainty for the design of space trajectories. The main contribution of this section was the development of the BOC formulation and the generalisation of a shooting-like stochastic transcription to the case of orbit determination arcs and statistical controllers. It was shown that the proposed belief-based formulation could accommodate both aleatory and epistemic and allows the treatment of families of probability measures since no assumptions are made on the nature of the probability distributions describing the uncertainties or on the transition function to propagate them. This formulation further enables the optimisation of the *open-loop* thrust profile, taking into account the effect of *closed-loop* components in continuous state space applications with sparse observations, thus coupling the trajectory optimisation process with the navigation analysis. Indeed, a belief-based model is particularly suited to model the inference step necessary for the state knowledge update when an orbit determination campaign is carried out. The proposed transcription was generalised to the presence of orbit determination by adapting the shooting-like discretisation through propagation and inference maps. A sparse polynomial mapping was employed to propagate the belief state through the nonlinear dynamics. The developed scheme was shown to transcribe the continuous problem under uncertainty in a discrete form to be optimised with a local NLP solver. The optimised solution resulting from this approach is highly informative as it determines the nominal control profile given a predefined control policy for possible deviations due to uncertainty, hence directly providing multiple control laws and the associated empirical margins for correction manoeuvres.

These approaches developed in this chapter will be applied in Chapters 7 and 8.

Part II

Applications to Space Systems Estimation and Control

Chapter 5

Robust Collision Analysis and Avoidance

The content of this chapter was published in:
C. Greco, and M. Vasile, "Robust Bayesian Particle Filter for Space Object Tracking Under Severe Uncertainty", *Journal of Guidance, Control, and Dynamics*, under review [84];
C. Greco, L. Sánchez Fernández-Mellado, M. Manzi, and M. Vasile, "A Robust Bayesian Agent for Optimal Collision Avoidance Manoeuvre Planning", *8th European Conference on Space Debris*, 2021 [92].

This chapter presents the applications of the robust particle filter for state estimation developed in Section 3.2 and introduces an autonomous decision-making Bayesian agent. In detail, Section 5.1 shows the application of the RPF to space debris conjunction analyses. Then Section 5.2 presents an application of the robust Bayesian framework to an autonomous action allocation and CAM planning scenario.

5.1 Robust Particle Filter for collision analysis

This section presents the test case to assess the performance of the robust filter. First, it introduces the general potential collision case and the uncertainty models for the initial and measurement uncertainty. Then, a numerical assessment of the pSIS wall time as a function of the number of particles and observations is carried out to verify the theoretical numerical complexity derived. Finally, three collision/non-collision scenarios are considered for testing the filter in different operational conditions.

5.1.1 General scenario definition

The scenario considered consists of a piece of debris on a potential collision orbit with a known operational satellite. SOCRATES (Satellite Orbital Conjunction Reports Assessing Threatening Encounters in Space)[156], an online service that provides twice-daily reports on the most likely collision events based on NORAD two-line elements (TLEs) and the SGP4 propagator, was employed to select two space objects with low distance at their close approach. The orbital elements of these two objects are reported in Table 5.1. The first object is an operational satellite, SENTINEL 2B (NORAD ID 42063), whose ephemerides are assumed to be known perfectly (or at least very well) as it carries a GPS receiver. The second object is a piece of debris (NORAD ID 30141) resulted from the anti-satellite test carried out on FENGYUN 1C DEB in 2007 [157]. Thus, the state of the debris is the latent variable $\mathbf{x}_k$ to filter. The reference Time of Closest Approach (rTCA) is the 13-Jan-2021 at 13:24:25 UTC. We will define the final time of our propagation as $t_F = rTCA$, with $t_F > t_M$ the time of the last observation before the possible collision.

Table 5.1: Spacecraft orbital elements at reference epoch from NORAD TLEs.

NORAD ID	EPOCH [UTC]	a [km]	e [-]	i [deg]	Ω [deg]	ω [deg]	M [deg]
42063	08-Jan-2021 01:17:15	7167.14	1.1e-4	98.57	85.33	81.09	279.04
30141	07-Jan-2021 16:24:07	7180.78	2.5e-3	99.08	183.01	252.25	107.59

Note that in general the uncertainty in both objects should be taken into account. However, the scope of this example is to test the ability of the filter to generate robust bounds on the probability of collision rather than the assessment of the ability of the filter to handle high dimensional uncertainty spaces. For this reason we deliberately selected one of the two objects to be well-tracked, and assumed that the main source of epistemic uncertainty was the piece of debris. The application of our RPF to a scenario in which both objects are uncertain can be found in Section 5.2, where the operational satellite performs a CAM characterised by execution errors.

In this research, the motion of the body is described in Cartesian coordinates in an Earth-centered inertial reference frame. The dynamical model in Eq. (2.17) includes the following components [158]: the Earth gravitational force derived from the EGM96 geopotential model up to degree and order 4; the atmospheric drag, according to Jacchia-Gill model; the third-body disturbances due to the Moon and Sun gravitational attraction; the solar radiation pressure with a conical shadow model for Earth's eclipses.

The quantity to compute and bound is the Probability of Collision (PoC). A collision is defined when the minimum distance between the two objects is smaller than a given threshold δ_{DCA}. Hence, the collision indicator is expressed as:

$$\mathbb{I}_C(\mathbf{x}_F) = \begin{cases} 1 & DCA(\mathbf{x}_F) \leq \delta_{DCA} \\ 0 & DCA(\mathbf{x}_F) > \delta_{DCA}\,, \end{cases} \tag{5.1}$$

where DCA is the function extracting the Distance of Closest Approach (DCA), that is the minimum of the relative position norm between the debris state realisation and the known operational satellite.

To detect the correct DCA for each sample, the relative distance at rTCA cannot be not used directly because different state realisations have different TCAs. Because the relative velocity is approximately 11 km/s, it is paramount to identify the correct TCA for each sample to ensure an accurate DCA computation. Hence, the polynomial map is used to propagate all the particles to rTCA, and then the hyperbolic assumption is employed to compute the specific TCA for each sample. Therefore, the minimum DCA for a realisation $\mathbf{x}_F$ is obtained by computing the pericenter distance of the relative hyperbolic trajectory between the two objects [159].

From here, the probability of collision is evaluated by computing the expectation of the indicator function

$$\mathrm{PoC}(\boldsymbol{\lambda}) = \int \mathbb{I}_C(\mathbf{x}_F)\, p(\mathbf{x}_F|\mathbf{y}_{1:M};\boldsymbol{\lambda})\, d\mathbf{x}_F\,. \tag{5.2}$$

Therefore, the goal is to compute robust bounds on the PoC as

$$\underline{\mathrm{PoC}} = \min_{\boldsymbol{\lambda} \in \Omega_{\boldsymbol{\lambda}}} \mathrm{PoC}(\boldsymbol{\lambda}) \tag{5.3a}$$

$$\overline{\mathrm{PoC}} = \max_{\boldsymbol{\lambda} \in \Omega_{\boldsymbol{\lambda}}} \mathrm{PoC}(\boldsymbol{\lambda}) \,, \tag{5.3b}$$

with a specific interest for $\overline{\mathrm{PoC}}$ which represents the epistemic worst-case scenario.

Initial state uncertainty

The initial distribution is constructed by imposing an uncertainty measure on the debris' TLE in Table 5.1. The value of such uncertainty is defined by following the guidelines of the European Space Agency (ESA) [160]. For inclinations larger than 60 deg, the 1σ uncertainty in the radial, transversal and normal components of position and velocity are reported in Table 5.2. A Gaussian parametric family is employed to describe the initial state uncertainty.

Table 5.2: 1σ position (r) and velocity (v) uncertainty of TLEs for orbits with $e < 0.1$, $i > 60\,\mathrm{deg}$, perigee altitude $\leq 800km$, in radial (U), transversal (V), and normal components (W).

$1\sigma_{r_U}$ [m]	$1\sigma_{r_V}$ [m]	$1\sigma_{r_W}$ [m]	$1\sigma_{v_U}$ [mm/s]	$1\sigma_{v_V}$ [mm/s]	$1\sigma_{v_W}$ [mm/s]
104	556	139	559	110	148

The covariance matrix $\boldsymbol{\Sigma}_{\mathbf{x}_0}$ in inertial coordinates is computed using the Jacobian of the transformation from radial, transversal and normal to Cartesian coordinates. From here, the importance initial distribution is defined as a normal distribution

$$\pi(\mathbf{x}_0) = \mathcal{N}\Big(\mathbf{x}_0; \mu_{\mathbf{x}_0}, \boldsymbol{\Sigma}_{\mathbf{x}_0}\Big) \tag{5.4}$$

where $\mu_{\mathbf{x}_0}$ is the Cartesian state retrieved from the TLE. Epistemic uncertainty is introduced in the probability distribution associated with the initial conditions by using two epistemic parameters. The initial epistemic set is defined as:

$$\begin{aligned}\mathcal{P}_{x_0} = \{\, p(\mathbf{x}_0;\boldsymbol{\lambda}_0) :\ p(\mathbf{x}_0) = \mathcal{N}\Big(\mathbf{x}_0; \mu_{\mathbf{x}_0}, \widetilde{\boldsymbol{\Sigma}}_{\mathbf{x}_0}; \boldsymbol{\lambda}_0\Big),& \\ \widetilde{\boldsymbol{\Sigma}}_{\mathbf{x}_0} = \mathrm{diag}(\, \lambda_{\mathbf{x}_{0-1}}\, \boldsymbol{\Sigma}_{\mathbf{x}_0}(1:3,1:3),\, \lambda_{\mathbf{x}_{0-2}}\, \boldsymbol{\Sigma}_{\mathbf{x}_0}(4:6,4:6)),& \\ \lambda_{\mathbf{x}_{0-1}} \in [1/5^2, 5^2],\ \lambda_{\mathbf{x}_{0-2}} \in [1/5^2, 5^2]\,\}\,,&\end{aligned} \tag{5.5}$$

where $\boldsymbol{\lambda}_0 = [\lambda_{0-1}, \lambda_{0-2}]$ is the epistemic parameter on the initial distribution, the symbols $\boldsymbol{\Sigma}_{\mathbf{x}_0}(1:3,1:3)$ and $\boldsymbol{\Sigma}_{\mathbf{x}_0}(4:6,4:6)$ indicate respectively the position block and the velocity block of the covariance matrix $\boldsymbol{\Sigma}_{\mathbf{x}_0}$, and the operator diag indicates a block-diagonal matrix. Therefore, the set $\mathcal{P}_{x_0}$ is parameterised using two epistemic multipliers $\lambda_{\mathbf{x}_{0-1}}$ and $\lambda_{\mathbf{x}_{0-2}}$ which scale the covariance matrix, reducing the initial uncertainty for multipliers < 1, or increasing the initial uncertainty for multipliers > 1. The multiplier range $[1/5^2, 5^2]$ means that the standard deviations of the initial state may be reduced or increased by a factor 5 of their reference value computed from Table 5.2.

Note that the definition and parameterisation of the family of distributions are very much dependent on the nature of the epistemic uncertainty that one is considering. In this illustrative example, we maintain the parameterisation of a Gaussian type of family of distributions. More general parameterisations are also possible, see [161] for an example of the use of Bernstein polynomials to represent generic families on bounded support. Although the inputs' distribution is parametric, the state distribution in time will have a generic form thanks to the particle-based method, capturing the dynamical and observational nonlinearities. The range and distribution of the uncertain parameters $\lambda_{x_{0-1}}$ and $\lambda_{x_{0-2}}$ can be derived from a statistical analysis of a time series of observations or CDMs. In this example, we take the more general case in which the parameters are defined within intervals, but their distribution is unspecified.

Observation model and errors

The simulated scenario involves indirect measurements of the state between the initial time and the TCA which are employed to improve the state knowledge. The measured quantities are the debris azimuth and elevation with respect to the equatorial plane [162], such that the ideal measurement model is expressed as

$$h(t_k, \mathbf{x}_k, \boldsymbol{\varepsilon}_k) = h(t_k, \mathbf{x}_k) + \boldsymbol{\varepsilon}_k = \begin{bmatrix} h_{az}(t_k, \mathbf{x}_k) \\ h_{el}(t_k, \mathbf{x}_k) \end{bmatrix} + \boldsymbol{\varepsilon}_k = \begin{bmatrix} \arctan \frac{\mathbf{x}_k(2)}{\mathbf{x}_k(1)} \\ \arcsin \frac{\mathbf{x}_k(3)}{\|\mathbf{x}_k(1{:}3)\|} \end{bmatrix} + \boldsymbol{\varepsilon}_k\,, \quad (5.6)$$

where $\mathbf{x}_k(i)$ indicates the i-th element of the vector $\mathbf{x}_k$, i.e. the Cartesian position elements for $i = 1, \ldots, 3$, $h(t_k, \mathbf{x}_k)$ indicates the ideal model, and the noise term is assumed additive. The noise is modelled as zero-mean normal with diagonal covariance $\boldsymbol{\Sigma}_{\mathbf{y}_k}$ generated using the standard deviations in Table 5.3. The simulated measurements are generated using the debris reference trajectory, i.e. with initial conditions $\mu_{\mathbf{x}_0}$, which is one of the collision trajectories of the space debris. Observation errors are drawn from the distribution of $\boldsymbol{\varepsilon}_k$. Observations are taken every eight hours between the initial epoch and the rTCA. The resulting noisy observations are indicated as $\bar{\mathbf{y}}_k$.

Table 5.3: 1σ azimuth (az) and elevation (el) uncertainty for noisy measurements of debris.

$1\sigma_{az}$ [arcsec]	$1\sigma_{el}$ [arcsec]
10	10

In the epistemic scenario, we assume that the observation covariance $\boldsymbol{\Sigma}_{\mathbf{y}_k}$ is not known precisely due to poor sensor characterisation. The likelihood epistemic set $\mathcal{P}_{y_k|x_k}$ is therefore parameterised as

$$\begin{aligned} \mathcal{P}_{y_k|x_k} = \Big\{ p(\bar{\mathbf{y}}_k|\mathbf{x}_k) : \; & p(\bar{\mathbf{y}}_k|\mathbf{x}_k) = \mathcal{N}\Big(h(t_k, \mathbf{x}_k); \bar{\mathbf{y}}_k, \widetilde{\boldsymbol{\Sigma}}_{\mathbf{y}_k}\Big), \\ & \widetilde{\boldsymbol{\Sigma}}_{\mathbf{y}_k} = \mathrm{diag}(\, \lambda_{\mathbf{y}_{\Sigma-1}}\, \boldsymbol{\Sigma}_{\mathbf{y}_k}(1,1),\, \lambda_{\mathbf{y}_{\Sigma-2}}\, \boldsymbol{\Sigma}_{\mathbf{y}_k}(2,2)), \\ & \lambda_{\mathbf{y}_{\Sigma-1}} \in [1/5^2, 5^2],\, \lambda_{\mathbf{y}_{\Sigma-2}} \in [1/5^2, 5^2] \Big\}, \end{aligned} \quad (5.7)$$

where $\boldsymbol{\Sigma}_{\mathbf{y}_k}(1,1)$ and $\boldsymbol{\Sigma}_{\mathbf{y}_k}(2,2)$ indicate respectively the azimuth and elevation variance values of the reference covariance matrix $\boldsymbol{\Sigma}_{\mathbf{y}_k}$ as resulting from the standard deviations in Table 5.3. The unknown epistemic parameters $\lambda_{\mathbf{y}_{\Sigma-1}}$ and $\lambda_{\mathbf{y}_{\Sigma-2}}$ are constant for all the acquired measurements $k = 1, \ldots, M$ as they are intended to describe the lack of knowledge about the accuracy of the sensor and not the accuracy of a specific observation. The resulting ranges of standard deviation for the azimuth and elevation measurements encompass diverse figures found in literature [163, 164, 165].

5.1.2 Wall time analysis

In this section, the wall time T_{pSIS} of a pSIS call is used as a metric to numerically check the algorithm complexity in Eq. (3.21) for varying numbers of observations and particles. The upper value $N = 50000$ was selected experimentally as the number of particles after which the pSIS estimator value changed only marginally (less than 1% from the previous N). A two dimensional grid is constructed by using $M \in \{10 + 5i : i = 1, \dots, 8\}$ observations and $N = \{5000j : j = 1, \dots, 10\}$ particles. The filter is run for each pair, and the average wall time for 100 pSIS calls is stored. The process is repeated 30 times, i.e. until their average varied only marginally for most of the pairs (less than 1% from the previous iteration), and the average time is stored. This simulation was performed on Matlab R2020b on a macOS Big Sur 3.5GHz Dual-Core i7. The same system will be employed for the successive simulations.

Hence, linear regressions on the average wall time are run with three different candidate models:

a) *linear* $T_{pSIS} \sim \beta_0 + \beta_1 M + \beta_2 N$;

b) *linear with interaction* $T_{pSIS} \sim \beta_0 + \beta_1 M + \beta_2 N + \beta_3 MN$;

c) *quadratic* $T_{pSIS} \sim \beta_0 + \beta_1 M + \beta_2 N + \beta_3 MN + \beta_4 M^2 + \beta_5 N^2$.

The results of the linear regression for each of this models is shown in Tables 5.4 to 5.6. Each table reports: the Root Mean Squared Error (RMSE) on the residuals; R^2, that is the ratio between the regression's sum of squared residuals and the total sum of squares, thus it measures the variance proportion explained by the model; p-value, that corresponds to an F-test on the regression model. Besides, it shows the estimate of the coefficients, the standard error of their estimate, the t-statistic, and the corresponding p-value. In addition, Table 5.7 shows the change in R^2 when the interaction and the quadratic terms are included in the model as well as the p-value of the F-test on these ΔR^2.

Table 5.4: Linear regression of T_{pSIS} with linear model.

Model	RMSE	R^2	p-value
linear	0.0578	0.928	$< .001$
Predictors	β_0	β_1	β_2
Estimate	-2.84e-01	9.15e-03	1.21e-05
SE	2.30e-02	5.64e-04	4.50e-07
t-stat	-1.23e+01	1.62e+01	2.69e+01
p-value	$< .001$	$< .001$	$< .001$

Table 5.4 shows that the linear model is significant given the observed wall times as well as the estimate of the coefficients. The R^2 value shows that the linear model explains 92.8% of the observed variance.

Table 5.5: Linear regression of T_{pSIS} with linear model with interaction.

Model	RMSE	R^2	p-value	
linear w/ int	0.0360	0.972	$< .001$	
Predictors	β_0	β_1	β_2	β_3
Estimate	-4.18e-02	1.71e-03	3.31e-06	2.71e-07
SE	2.62e-02	7.59e-04	8.43e-07	2.45e-08
t-stat	-1.60e+00	2.25e+00	3.93e+00	1.11e+01
p-value	1.14e-01	2.76e-02	$< .001.$	$< .001$

Table 5.5 reports the results when the interaction term MN is included in the regression model. The R^2 value increased to 97.2% of the observed variance showing that the inclusion of the interaction actually explains the observed wall time variance better. Also, the ΔR^2 is significant as it can be seen in Table 5.7. Also, the RMSE in this model decreased by approximately 37% with respect to the linear model.

Table 5.6: Linear regression of T_{pSIS} with quadratic model.

Model	RMSE	R^2	p-value			
quadratic	0.0357	0.973	$< .001$			
Predictors	β_0	β_1	β_2	β_3	β_4	β_5
Estimate	-9.50e-02	5.70e-03	2.95e-06	2.71e-07	-6.14e-05	6.56e-12
SE	4.31e-02	2.39e-03	1.47e-06	2.43e-08	3.49e-05	2.20e-11
t-stat	-2.20e+00	2.39e+00	2.01e+00	1.11e+01	-1.76e+00	2.98e-01
p-value	3.07e-02	1.96e-02	4.83e-02	$< .001$	8.24e-02	7.66e-01

Table 5.6 presents the regression results for the quadratic model. The p-values for the coefficients of the quadratic terms, that is, β_4 and β_5, are larger than 0.05, and thus they are not statistically significant. Furthermore, when compared to the linear model with interaction, the R^2 increased only marginally, and the RMSE decreased by less than 1%. The ΔR^2 increase is not significant as reported in Table 5.7.

Table 5.7: Progressive change in R^2 and change significance.

Model	R^2	ΔR^2	p-value ΔR^2
linear	0.928	0.928	$< .001$
linear w/ int	0.972	0.045	$< .001$
quadratic	0.973	0.001	.21

Thus, this analysis provides empirical evidence that the numerical complexity of a pSIS call is indeed linear with interaction on the number of observations and particles as derived in Eq. (3.21).

5.1.3 Results

Four main instances of the conjunction scenario are considered in this section. In the first, no observations are simulated and we compute the collision probability resulting from uncertainty propagation only. In the last three, we simulate observations up until 24 hours before the rTCA. In the first two instances, the true unknown trajectory of the FENGYUN 1C DEB debris eventually results in a collision with SENTINEL 2B. In the second, after the robust PoC analysis 24 hours before rTCA, the operator decides to take further measurements before deciding whether to manoeuvre or not. In the third one, the debris' true unknown trajectory does not result in a collision although it passes close to SENTINEL. Similarly, extra measurements are employed to decide whether to implement a manoeuvre. The last instance is again a collision trajectory, but instead of taking new measurements, they are simulated to see how they would impact the PoC. Different subcases are tested for this last instance to test the filter scalability further. The filter settings and performance for each instance are reported in Table 5.8.

Uncertainty propagation

For instance A, an initial uncertainty propagation using $N = 50000$ particles, the maximum N used in the wall time analysis, and without any measurements, that is $M = 0$, is performed to compute the probability bounds resulting from uncertainty on the initial conditions. The PoC is computed using a Hard-Body Radius (HBR) corresponding to the threshold $\delta_{DCA} = 50$m, which is conservative for the combined satellites' dimensions. Fig. 5.1 represents 3σ ellipses in b-plane coordinates. The ellipses are reconstructed from the propagated samples and weights. The different ellipses correspond to different values of the epistemic parameters, which impact the weights. Besides, each sample is characterised by a different incoming asymptote, which correspond to different B-plane orientation [166]. Hence, Fig. 5.1 and successive ones represent all the ellipses in a unique plane, which is the average of all the realisations-specific b-planes for a given conjunction instance. The ellipses corresponding to the epistemic parameters $\underline{\boldsymbol{\lambda}}$ and $\overline{\boldsymbol{\lambda}}$, respectively yielding the lower $\underline{\text{PoC}}$ and upper $\overline{\text{PoC}}$, are plotted as well with the same line style.

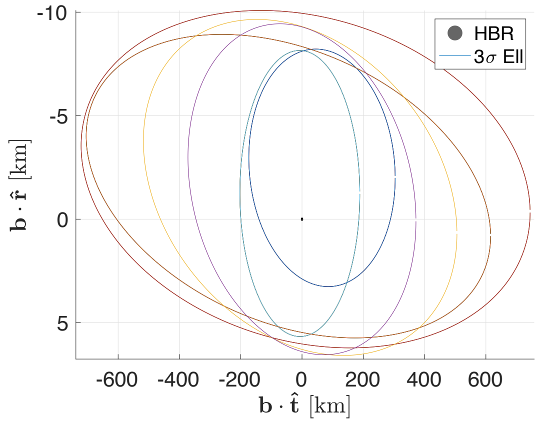


Figure 5.1: Debris' HBR and b-plane 3σ ellipses for instance A.

One could also plot the Empirical Cumulative Distribution Function (ECDF) of the DCA from the propagated samples and weights as displayed in Fig. 5.2. The distribution corresponding to the lower probability, henceforth labelled as lower distribution, is displayed in green, whereas the one corresponding to the upper probability, labelled upper distribution, is represented in red. The dashed grey distributions are the ones

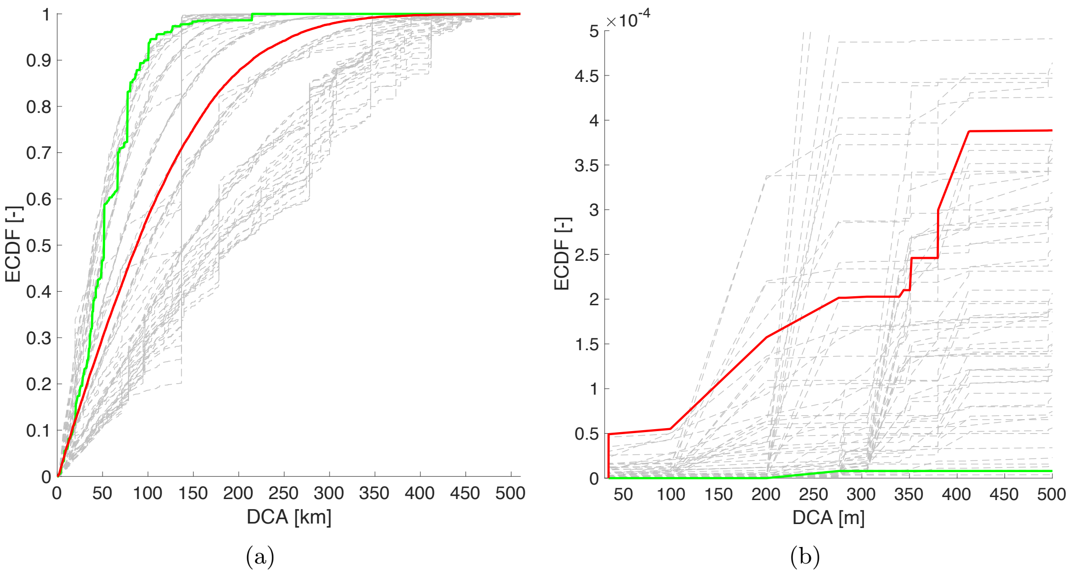


Figure 5.2: DCA's ECDFs for instance A. Fig. 5.2(b) is a zoom of Fig. 5.2(a) for DCAs up to 500 meters.

resulting from different epistemic parameters within the imprecise sets. Fig. 5.2(a) displays the full distributions. Fig. 5.2(b) is a zoom of Fig. 5.2(a) for DCA up to 500m. For the threshold distance $\delta_{DCA} = 50\text{m}$, the collision probability is within the robust interval

$$\mathrm{PoC}_{50\mathrm{m}} \in \left[0, 4.97 \cdot 10^{-5}\right].$$

The filter performance for this instance A is reported in Table 5.8. The propagation step, including surrogate construction and particle propagations, requires approximately half of the total computational time. The time dedicated to construct the proposal is zero as no observations are employed in this instance.

It is important to underline that the lower and upper distributions do not necessarily result in lower and upper bounds for the collision probability at different distance thresholds $\delta_{DCA} \neq 50\text{m}$. Specifically, the lower and upper distributions are not lower and upper envelopes for the ECDFs resulting from the imprecise set, but they are distributions belonging to the set itself bounding the expectation of a specific quantity of interest. From Fig. 5.2 we can notice that indeed the lower and upper distributions do not yield to the probability lower and upper bounds for DCA thresholds larger than

$\approx 100\,\mathrm{m}$. Even more, from Fig. 5.2(a) we can notice that the green line is above the red one for DCA thresholds larger than $\approx 20\,\mathrm{km}$.

The ellipses and ECDFs are reconstructed from uncertainty propagation of TLE-derived uncertainties. ESA's guidelines for pure Two-Line Elements (TLE) screening set a threshold of PoC $> 10^{-4}$ for $\delta_{DCA} = 300\,\mathrm{m}$ to trigger further analyses. For such distance, by visual enquiry of Fig. 5.2(b), the upper probability is $\overline{\mathrm{PoC}}_{300\mathrm{m}} > 5 \cdot 10^{-4}$. Thus, the upper probability value flags up a potential collision that needs to be further investigated. On the other hand, the lower probability is $\underline{\mathrm{PoC}}_{300\mathrm{m}} < 1 \cdot 10^{-5}$, which would indicate a near-safe scenario. The difference between these two scenarios indicates why it is critical to model and handle the epistemic uncertainty in collision risk assessment. Specifically, the width of the probability interval indicates the degree of ignorance of the probability estimate. Thus, it should be used to support the decision-maker and decide whether to acquire new measurements or to manoeuvre. In the next sections, we will use the distance threshold of $\delta_{DCA} = 50\mathrm{m}$ as it is tailored for the satellites under assessment.

Collision scenario

For instance B, a tracking campaign with $M = 12$ noisy observations is considered between the satellites' epoch dates and 24 h before the rTCA. At this cut-off time, the RPF is run to estimate the PoC bounds. The ellipses resulting from said simulation are displayed in Fig. 5.3. The collision probability now belongs to the robust interval

$$\mathrm{PoC}_{50\mathrm{m}} \in \left[0, 2.15 \cdot 10^{-4}\right] .$$

The effective sample size at the last propagation step is greater than 75%, and therefore it is deemed suitable for the estimator approximation given the high number of particles employed. The upper PoC increased by almost an order of magnitude when the new tracking observations are taken into account. Still, for some combination of the initial and likelihood epistemic parameters, the PoC could be zero.

It is important to note how epistemic uncertainty on the initial distribution and

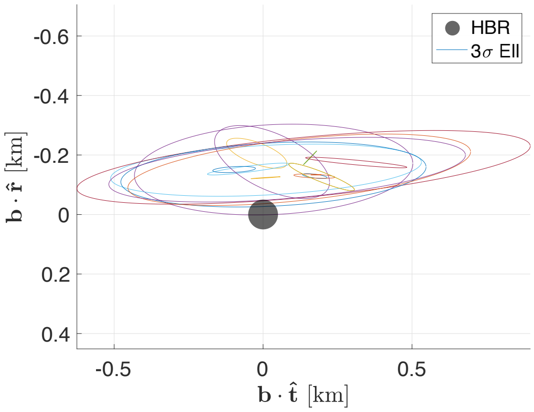


Figure 5.3: Debris' HBR and b-plane 3σ ellipses for instance B with observations until 24 h before rTCA.

observation likelihood result in B-plane ellipses, which vary in size, orientation, and central point. Thus, most algorithms that tackle the dilution of collision probability issue by just tuning the covariance at TCA would fail to capture the actual large variety of distributions resulting from epistemic uncertainty.

We now investigate how these bounds change if new observations are taken into account. In particular, two new sets of azimuth and elevation measurements are employed until 9 h before the rTCA. This second cut-off time is chosen to leave enough time for further observations' processing, decision making and, in case, communications with the satellite to manoeuvre.

The new DCA distributions are shown in Fig. 5.4. The ellipses are more concentrated near the HBR of the operational satellite. Thus, the collision probability interval is now

$$\mathrm{PoC}_{50\mathrm{m}} \in \left[0, 9.69 \cdot 10^{-3}\right] .$$

The upper probability is almost 1%, which flags up a critical collision risk. Again, the lower bound is 0 although the unknown trajectory is actually a collision one, and the last observation is relatively close to the rTCA mainly because of probability dilution resulting from specific combinations of the initial condition and likelihood covariance matrices. This difference further highlights the importance of considering epistemic

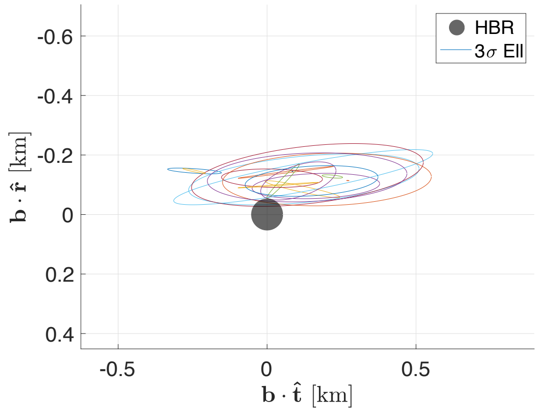


Figure 5.4: Debris' HBR and b-plane 3σ ellipses for instance B with observations until 9 h before rTCA.

uncertainty in collision risk assessment.

The filter performance for this instance B is reported in Table 5.8. The propagation requires the most computational time whereas the optimisation converges in less than 1000 iterations. When compared to instance A, the use of observations causes a higher computational time for: i) the surrogate construction as a new polynomial expansion needs to be constructed for each observation time; ii) the bound computation, despite the lower number of B&B iterations, as the pSIS estimator complexity increases with the number of measurements. The proposal requires a smaller share of time as it employs the surrogate to propagate the sigma points and it requires only few efficient matrix operations.

No-collision scenario

For instance C, the true unknown initial velocity is slightly perturbed by approximately 1m/s to cause a reference miss distance of 1 km. Again a tracking campaign with $M = 12$ is considered up to 24 h before the rTCA. The resulting ellipses are shown in Fig. 5.5. The collision probability now belongs to

$$\mathrm{PoC}_{50\mathrm{m}} \in \left[0, 2.82 \cdot 10^{-4}\right] .$$

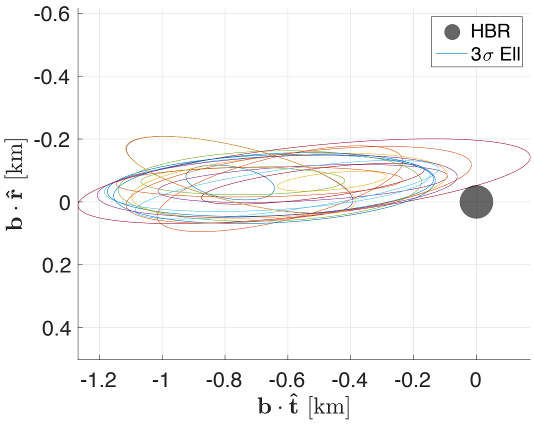


Figure 5.5: Debris' HBR and b-plane 3σ ellipses for instance C with observations until 24 h before rTCA.

Although the true unknown trajectory does not result in a collision, we see how the upper PoC computed 24h before rTCA is similar to the one in the collision scenario. This happens especially for two reasons: the measurements are affected by noise; the dynamics is highly nonlinear and has a fast time scale. The former causes larger uncertainties in the state estimates. The latter causes even small deviations at the cut-off time to translate into significant differences at rTCA. Indeed, the satellites revolve more than 14 times in the 24h time span.

Again, we investigate how the bounds change with the two new sets of measurements until 9 h before rTCA. The new ellipses are plotted in Fig. 5.6, and the probability interval is

$$\mathrm{PoC}_{50\mathrm{m}} \in [0, 0] \; .$$

The width of the probability interval is zero, meaning that the estimated probability is 0 no matter the epistemic parameter. Hence, although the $\overline{\mathrm{PoC}}$ at the 24 h cut-off was larger than the one in the collision scenario, the PoC interval at the 9 h cut-off actually reflects the no-collision scenario.

Again, the performance for this instance C is displayed in Table 5.8. In this case, the bound computation routine requires more iterations to estimate the robust bounds, whereas the propagation and proposal steps take approximately the same computational time.

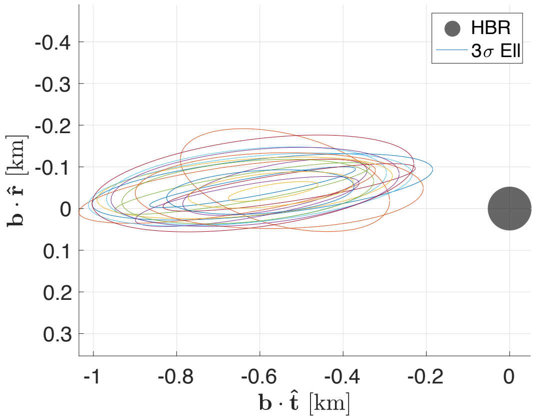


Figure 5.6: Debris' HBR and b-plane 3σ ellipses for instance C with observations until 9 h before rTCA.

Measurements simulation scenario

The last instance D is again the collision one in Section 5.1.3, but the measurements after the 24h cut-off are simulated rather than acquired. The measurement simulation is useful for the operator to understand and quantify the impact of future observations on the probability figures. If the PoC estimates do not change significantly with successive measurements, it would be optimal to manoeuvre, if necessary, as soon as possible without waiting for further measurements. If the probability interval changes significantly and, specifically, becomes wider, successive observations can help to discriminate whether a manoeuvre is truly needed or not.

The likelihood mean in Eq. (5.7) was set to be the received measurement itself. Simulated measurements are characterised by further epistemic uncertainty on the likelihood mean because no observation has been received yet. Thus, for this scenario, the likelihood epistemic set for the simulated future measurements $k = M+1, M+2$ is parameterised as

$$
\begin{aligned}
\mathcal{P}_{y_k|x_k} = \Bigg\{ p(\mathbf{y}_k|\mathbf{x}_k): \quad & p(\mathbf{y}_k|\mathbf{x}_k) = \mathcal{N}\Big(h(t_k, \mathbf{x}_k); \mathbf{y}_k, \widetilde{\boldsymbol{\Sigma}}_{\mathbf{y}_k}\Big), \\
& \mathbf{y}_k = \begin{bmatrix} h_{az}(t_k, \bar{\mathbf{x}}_k) + \lambda_{\mathbf{y}_{\mu_k-1}} \sqrt{\boldsymbol{\Sigma}_{\mathbf{y}_k}(1,1)} \\ h_{el}(t_k, \bar{\mathbf{x}}_k) + \lambda_{\mathbf{y}_{\mu_k-2}} \sqrt{\boldsymbol{\Sigma}_{\mathbf{y}_k}(2,2)} \end{bmatrix} \\
& \widetilde{\boldsymbol{\Sigma}}_{\mathbf{y}_k} = \mathrm{diag}(\,\lambda_{\mathbf{y}_{\Sigma-3}}\, \boldsymbol{\Sigma}_{\mathbf{y}_k}(1,1),\ \lambda_{\mathbf{y}_{\Sigma-4}}\, \boldsymbol{\Sigma}_{\mathbf{y}_k}(2,2)), \\
& \lambda_{\mathbf{y}_{\mu_k-1}} \in [-5, 5],\ \lambda_{\mathbf{y}_{\mu_k-2}} \in [-5, 5] \\
& \lambda_{\mathbf{y}_{\Sigma-3}} \in [1/5^2, 5^2],\ \lambda_{\mathbf{y}_{\Sigma-4}} \in [1/5^2, 5^2] \Bigg\},
\end{aligned}
\tag{5.8}
$$

where $\lambda_{\mathbf{y}_{\mu_k-1}}$ and $\lambda_{\mathbf{y}_{\mu_k-2}}$ are two new epistemic parameters modelling the uncertainty on each observation value. These epistemic parameters regulate deviations from a reference measurement value $h(t_k, \bar{\mathbf{x}}_k)$ taken along the reference trajectory. The epistemic parameters $\lambda_{\mathbf{y}_{\Sigma-3}}$ and $\lambda_{\mathbf{y}_{\Sigma-4}}$ are the covariance multipliers which are constant for each new simulated measurement as before.

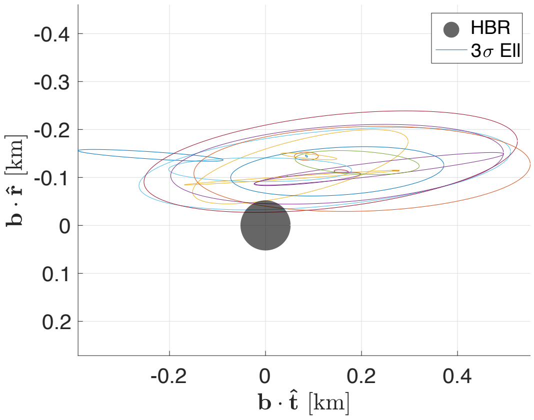
For this instance, six subcases are run to investigate the filter scalability. The first three (D1, D2 and D3) employ a set of simulated measurements until 9 h before collision, while the last three (D4, D5 and D6) simulate an extra set of observations until 6 h before rTCA. Then, they differ for the B&B convergence thresholds ε as reported in Table 5.8.

The results of the RPF runs on instances D3 and D6 are shown in Fig. 5.7. Specifically, in Fig. 5.7(a) the measurements are simulated at the same time instance as in instance B. The probability interval for D3 is

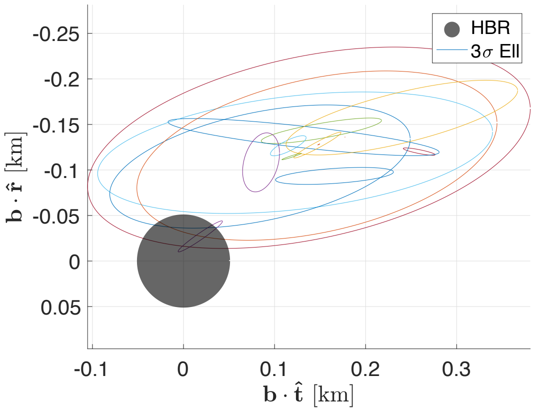
$$
\mathrm{PoC}_{50\mathrm{m}} \in \left[0, 2.94 \cdot 10^{-2}\right].
$$

By comparing this interval with the corresponding one in Fig. 5.4, we can see that the value of the upper probability is larger, up to almost 3%. Likely, this is due to the noise on the measurement in the first instance, whereas here the observation corresponding to the collision trajectory is included in the epistemic set.

Fig. 5.7(b) shows the ellipses for D6 when the measurements are simulated until 6 h before rTCA, that is less than 4 revolutions before the possible impact. The probability



(a)



(b)

Figure 5.7: Debris' HBR and b-plane 3σ ellipses for instance (a) D3 and (b) D6.

interval becomes

$$\mathrm{PoC}_{50\mathrm{m}} \in \left[1.20 \cdot 10^{-8}, 9.94 \cdot 10^{-1}\right] \,.$$

The upper probability shows that for specific measurement values within the imprecise set, the PoC estimate would be very close to 100%, therefore predicting the collision. On the other hand, the lower probability bound indicates that specific measurements can still implicate an extremely low collision risk. The width of such interval indicates a great degree of ignorance. In such a case, new better measurements should be acquired to narrow the PoC interval and to allow a better informed decision.

The filter performances for these instances are shown in Table 5.8 as well. The precomputation times are in line with instances B and C. The time to compute the bounds sharply increases primarily with the number of epistemic dimensions, mainly due to the initial triangulation complexity, and then with a more demanding convergence threshold. Nonetheless, the time increase is not factorial as the filter upper bound complexity thanks to early pruning.

Table 5.8: Filter performance for different conjunction instances.

Instance	n_λ	ε	L	Time				pSIS calls
				Surrogate [s]	Proposal [s]	Optimisation [s]	Total [min]	
A	2	10^{-6}	$6.03{\cdot}10^{-7}$	76.38	.00	80.02	2.60	1023
B	4	10^{-6}	$2.63{\cdot}10^{-6}$	150.62	26.97	117.91	4.93	903
C	4	10^{-6}	$1.67{\cdot}10^{-6}$	150.04	32.42	280.39	7.71	2774
D1	8	10^{-4}	$7.83{\cdot}10^{-5}$	154.27	33.17	784.82	16.20	4140
D2		10^{-5}				1265.25	23.76	6227
D3		10^{-6}				1466.72	27.57	7376
D4	10	10^{-4}	$2.57{\cdot}10^{-4}$	159.58	34.44	1555.66	29.16	9752
D5		10^{-5}				1736.89	32.18	11636
D6		10^{-6}				1922.37	35.27	13043

5.2 Bayesian framework for CAM

This section presents the initial design of a robust Bayesian framework for autonomous decision-making in conjunction scenarios. The goal is to reduce the workload on operators and improve autonomy on the ground segment as the number of space objects is rising sharply.

The proposed framework aims to automatically process observations, analyse conjunction geometry, compute reliable risk metrics, and decide the optimal action to take. Hence, this system aims to close the loop between the several topics developed in this thesis, that is, tracking, state estimation, uncertainty propagation, and optimal control for improving STM. Thus, this section will specifically discuss this agent's high-level architecture, whereas most of the single modules were developed in the relevant chapters.

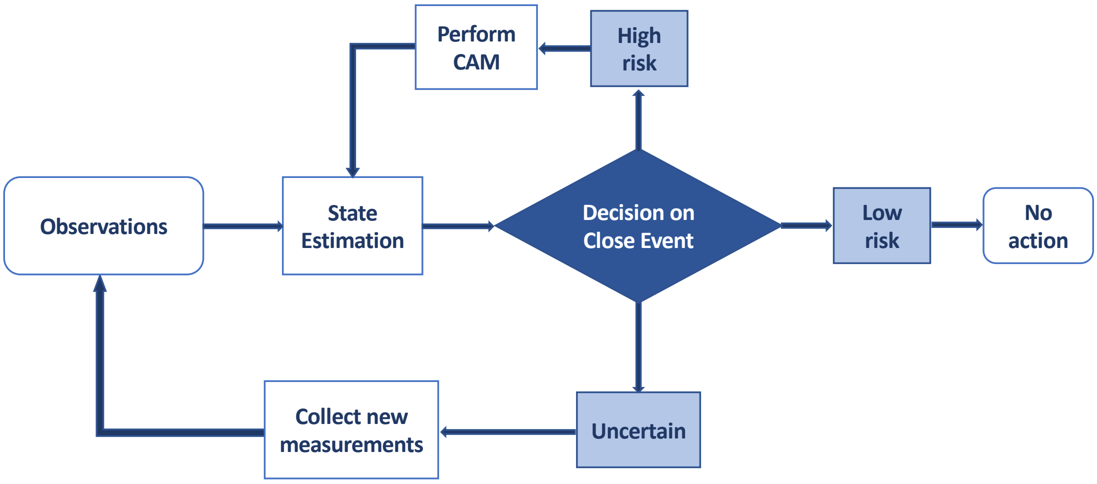


Figure 5.8: Diagram of robust Bayesian pipeline for optimal collision analysis and avoidance planning.

5.2.1 Framework architecture

Fig. 5.8 shows the pipeline of the Bayesian framework. Given a pair of satellites in a potential collision, the workflow starts from sparse, noisy measurements of the satellites' state collected from external sensors. The state estimation module powered by the RPF developed in Section 3.2 is run sequentially to estimate the collision geometry as new measurements are available. The RPF enables the computation of the posterior state distributions and the expectation on the collision probability accounting, at all steps, for aleatory and epistemic uncertainty. The Bayesian agent performs risk assessment using an AI-based classification system that considers the proximity of the encounter as well as all the uncertainty affecting the state. The system can provide feedback on the most suitable action to be taken. For events presenting low-risk, the suggestion would be to wait for new measurements or take no further actions. For events possessing a higher risk, the need of executing a CAM would be indicated and the optimal manoeuvre computed by a CAM optimiser. The CAM engine works in close collaboration with the estimation module. Indeed, the manoeuvre optimisation model employs the full nonlinear UP to re-estimate the collision probability under the initial uncertainty and newly introduced CAM execution errors. The goal is to find a CAM which ensures a safe encounter with collision probability below a safety threshold.

After manoeuvring, the new state is estimated and the risk metric updated as post-manoeuvre measurements are available. Finally, on events affected by a high degree of uncertainty, mainly epistemic due to lack of knowledge, the suggested action would be to acquire more observations and refine the state estimation before re-assessing the collision risk.

To accomplish these tasks, the Bayesian framework is composed of different modules that relate to each other: the robust state estimation; the Intelligent Classification System (ICS); the CAM optimiser. As said, the first module is based on the RPF previously introduced. In the following, the last two modules are introduced in more detail.

Intelligent classification system

This section describes the *Decision on Close Event* module of Fig. 5.8. This module of Fig. 5.8 employs a Machine Learning (ML) based ICS to automatically allocate the most appropriate action to be taken by an operator under the event of a close encounter. This ICS was first introduced in [167]. This criterion classifies a close encounter event based on: the time to the TCA; two epistemic variables, that are the lower expectation and the degree of confidence (the difference between the upper and lower expectation) of the collision probability being bigger than a given threshold. Oppositely to the common probabilistic approaches, where the decision is made exclusively based on the value of the computed PoC to raised binary alert flags (i.e. collision/no-collision), this method works with the lower and upper probabilities to provide a more complete information. Thus, the outcome of the classification criterion is not only able to indicate if an event is high risk or low risk, but also if it is affected by a high degree of uncertainty.

The output of the criterion is a 5-class classification aiming to address the uncertainty associated with the events properly and, therefore, provide a more meaningful decision to the operators. This five classes can be grouped on: high risk events, classes 1 and 2, which require a CAM (*High risk* module on Fig. 5.8); low risk events, classes 4 and 5, where no further action is needed (*Low risk* and *No action* modules on Fig. 5.8); event highly affected by uncertainty, class 3, where more observations should be acquired before making any decision on the manoeuvring (*Uncertain* and *Collect new measurements* modules on Fig. 5.8). The details of the ML-based classifier are discussed in [167, 92].

Collision avoidance manoeuvre optimisation

This section illustrates the *Perform CAM* module of Fig. 5.8. Specifically, it introduces the Collision Avoidance Manoeuvre model to compute the optimal CAM in those cases that require a correction in the orbits for avoiding a potential collision. This allows us to close the loop between state estimation, decision-making and CAM performance.

The optimal CAM is formulated as a BOC (see (4.29))

$$\begin{aligned} &\min_{\Delta \mathbf{v}} \ \Delta v \\ &\text{s.t.} \ \ \overline{\text{PoC}} < \varepsilon \ , \end{aligned} \tag{5.9}$$

that is a bi-level optimisation problem where again

$$\overline{\text{PoC}} = \max_{\boldsymbol{\lambda} \in \Omega_{\boldsymbol{\lambda}}} \text{PoC} \ , \tag{5.10}$$

and Δv is the magnitude of the nominal $\Delta \mathbf{v}$. Therefore, the goal is to find the optimal impulsive manoeuvre that minimises the nominal propellant expenditure while realising a PoC below the safe threshold ε even in the epistemic worst-case scenario. The safe PoC threshold is set to $\varepsilon = 10^{-6}$ according to European Space Agency's (ESA) guidelines [168].

When computing the PoC, it is important to account for possible deviations due to execution errors on the CAM. Indeed, the executed CAM will differ from the commanded one due to pointing errors, thrust inaccuracies, time delays and so on. The same model (4.10) for execution errors introduced in Section 4.2 will be employed in the robust Bayesian framework.

In this scenario, both the objects' states are modelled as random variables. Indeed, although the operational satellite may be very well tracked, CAM execution errors introduce uncertainty on its state knowledge as well. The PoC is computed as expectation of the indicator function (see Section 2.1) of a condition on the DCA between the two space objects. By labelling the spacecraft states $\mathbf{x}_{s/c_1}$ and $\mathbf{x}_{s/c_2}$, the collision indicator is

$$I_C = \begin{cases} 1 & \text{if } DCA(\mathbf{x}_{s/c_1}, \mathbf{x}_{s/c_2}) \leq \delta_C \\ 0 & \text{if } DCA(\mathbf{x}_{s/c_1}, \mathbf{x}_{s/c_2}) > \delta_C \ , \end{cases} \tag{5.11}$$

where δ_C is a collision threshold depending on the satellites' geometrical characteristics. The DCA for two objects in a close encounter is computed as the pericentre distance of their relative hyperbolic trajectory [159].

5.2.2 Case Study

This section discusses the applications to test the robust Bayesian agent for autonomous decision-making and CAM design. The same conjunction scenario, dynamical and observation models, initial and likelihood epistemic uncertainties introduced in Section 5.1.1 hold for this test case. The difference is that now the optimal action to take will be taken autonomously by the ICS. Furthermore, in a high-risk scenario, the robust Bayesian framework will design a robust CAM to actively avoid the collision encountered before.

The CAM and estimation engines of the Bayesian agent require the definition of execution errors. The definition of such random variable $\mathbf{\Theta}$ (see Eq. (4.10)) follows the Gates' model [169, 170]. A sketch of this model is displayed in Figure 5.9. In this model, the disturbance is defined in terms of modulus and angular variations on the commanded $\Delta\mathbf{v}$. In a frame centred in the spacecraft with the z-axis aligned with

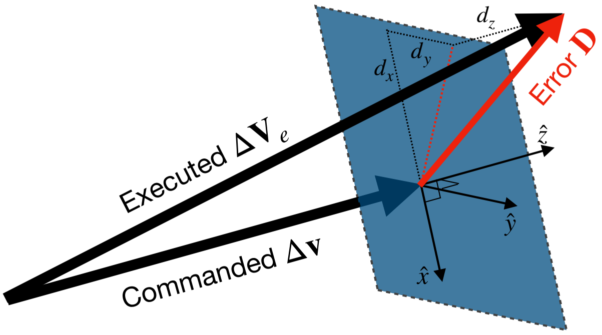


Figure 5.9: Execution error by Gates' model.

$\Delta\mathbf{v}$, the y-axis perpendicular to both the $\Delta\mathbf{v}$ and the ecliptic normal, and the x-axis completing the right-handed frame, the execution error components are taken from zero-mean normal distributions

$$\begin{aligned}
\Theta_x &\sim \mathcal{N}(0, \sigma_{pf}^2 + \sigma_{pp}^2 \Delta v) \\
\Theta_y &\sim \mathcal{N}(0, \sigma_{pf}^2 + \sigma_{pp}^2 \Delta v) \\
\Theta_z &\sim \mathcal{N}(0, \sigma_{mf}^2 + \sigma_{mp}^2 \Delta v) \ ,
\end{aligned} \tag{5.12}$$

where Δv is the magnitude of $\Delta\mathbf{v}$, σ_{mf} and σ_{mp} are respectively the standard deviations of the fixed and proportional magnitude component, while σ_{pf} and σ_{pp} concern the pointing components. In this model, the pointing error is decomposed into two components in the plane normal to the commanded $\Delta\mathbf{v}$, which is equivalent to decomposing it in terms of magnitude and angular components for suitable distributions [170]. These three components are then rotated into the inertial reference frame in which $\Delta\mathbf{v}$ is defined to yield the vector $\boldsymbol{\Theta}$. The parameters for the Gates' execution error are set as in Table 5.9.

Table 5.9: Gates' parameters for execution errors on CAM test case.

Component	Value	
σ_{pf}	3.0	[mm/s]
σ_{pp}	7.0	[mrad]
σ_{mf}	5.0	[mm/s]
σ_{mp}	.33%	[-]

The values of the thresholds used by the ICS for risk assessment are shown on Table 5.10. They include the two-time thresholds (T_1, T_2) to discern among short-term, mid-term and long term encounter; the threshold on the probability of collision (P_{C_0}) at which evaluate the lower expectation and the Degree of Confidence; the value of the lower expectation at which the trust on the value is lost $(\underline{\mathbb{E}}_0)$; and the value of the Degree of Confidence above which the event is considered to be highly affected by uncertainty (Δ).

Table 5.10: ICS thresholds values.

Threshold	Value	
T_1	2.0	[days]
T_2	4.0	[days]
P_{C_0}	10^{-6}	[-]
$\mathbb{E}_0$	0.5	[-]
Δ	0.3	[-]

5.2.3 Results

The first two operational scenarios introduced in Section 5.1.3 are considered here, that is, one resulting in a collision and one near-miss conjunction.

Collision scenario

In this case, the true unknown trajectory of the FENGYUN 1C DEB debris eventually results in a collision with SENTINEL 2B. We assume observations are available up to 48 hours before rTCA.

The result of the robust Bayesian estimation at this point is displayed in Fig. 5.10. In the plot, several 3σ ellipses, which correspond to different values of the epistemic

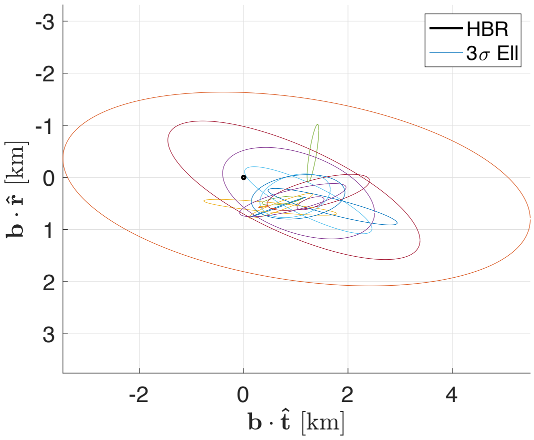


Figure 5.10: B-plane 3σ ellipses for collision scenario with observations up to 48h before TCA.

parameters, are displayed. The collision probability is robustly estimated as

$$\mathrm{PoC} \in [0, 2 \cdot 10^{-4}].$$

Hence, the ICS is run to analyse the conjunction. The classification returns a class 3 scenario, a highly uncertain conjunction requiring further measurements to take an informed action. Therefore, the agent decides to wait for new measurements until the next checkpoint, 24 hours before rTCA.

At this point, after new observations are available, the robust Bayesian estimation is rerun and the results displayed in Fig. 5.11. The collision probability is now

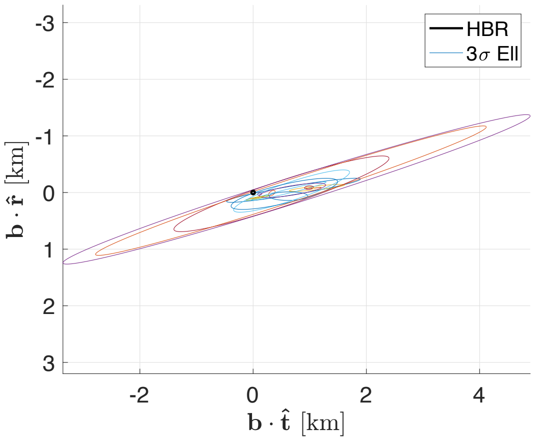


Figure 5.11: B-plane 3σ ellipses for collision scenario with observations up to 24h before TCA.

$$\mathrm{PoC} \in [0, 1.8 \cdot 10^{-3}].$$

The ICS return a class 1 conjunction, which is a high-risk event that requires a CAM.

Thus, the CAM module is run to design a robust and optimal manoeuvre which realise a $\overline{\mathrm{PoC}}$ smaller than 10^{-6} while taking into account state and CAM execution errors. The CAM on SENTINEL 2B is designed 10 revolutions before the rTCA.

After the manoeuvre execution, the robust estimation is rerun to check that the upper collision probability indeed satisfies the safety threshold in the presence of state and execution errors. The resulting graph is shown in Fig. 5.12, where it can be seen that there is no intersection between the 3σ ellipses and the HBR.

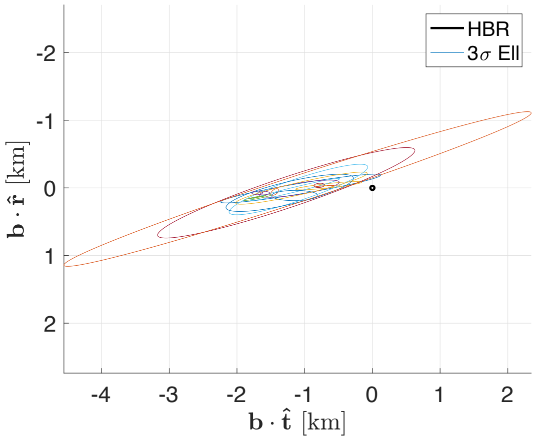


Figure 5.12: B-plane 3σ ellipses for collision scenario after CAM.

The upper probability bound post CAM is $\overline{\text{PoC}} < 10^{-7}$. The evolution of the PoC bounds at the several checkpoints considered is shown in Figure 5.13.

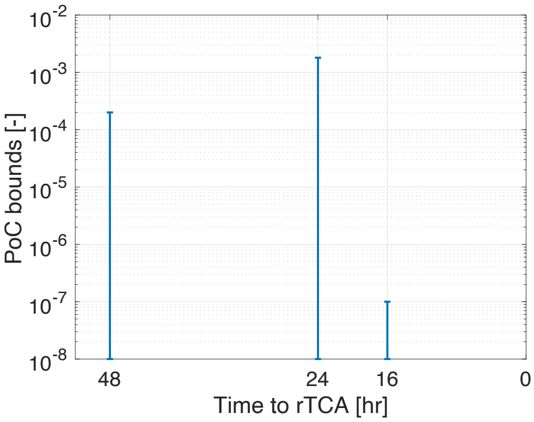


Figure 5.13: Evolution of PoC bounds for collision scenario.

No-collision scenario

In this scenario, the true unknown trajectory of the FENGYUN 1C DEB debris misses SENTINEL 2B.Again, we start assuming observations are available up to 48 hours before rTCA.

The result of the robust Bayesian estimation 48 hours before rTCA is depicted in Fig. 5.14. The robust probability interval is

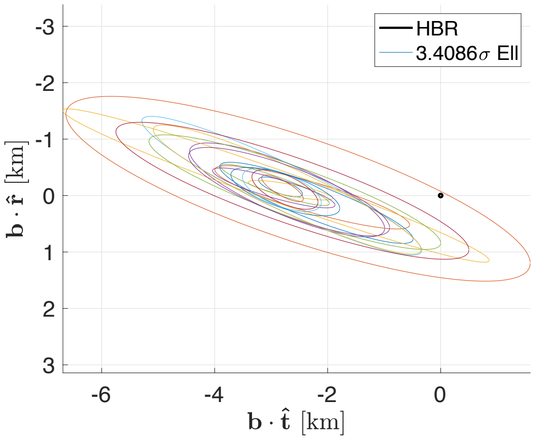


Figure 5.14: B-plane 3σ ellipses for no-collision scenario with observations up to 48h before TCA.

$$\text{PoC} \in [0, 1.8 \cdot 10^{-4}].$$

Thus, the conjunction analysis is similar to the collision one at 48 hours before rTCA. The ICS is run, and it returns a class 3 conjunction, again labelling a highly uncertain scenario and the need for further measurements to implement an informed action.

Further measurements are acquired until the 24 hours checkpoint, where the Bayesian estimation is updated. The resulting conjunction geometry is displayed in Fig. 5.15. Here, it can be seen how the debris 3σ ellipses are rather distant from the HBR of the

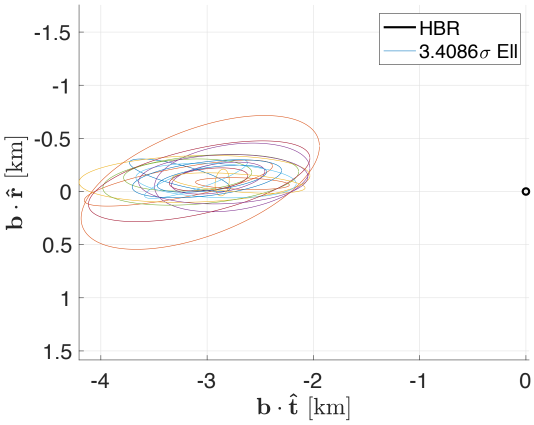


Figure 5.15: B-plane 3σ ellipses for no-collision scenario with observations up to 24h before TCA.

operational satellite and indeed centred on the true (unknown) trajectory. The ICS returns a class 5 conjunction labelling a safe conjunction. Indeed, the corresponding upper probability is estimated to be $\overline{\mathrm{PoC}} < 10^{-10}$. The intelligent agent, therefore, requires no further action.

For the no-collision scenario, the PoC bounds estimated at the two different checkpoints are displayed in Figure 5.16.

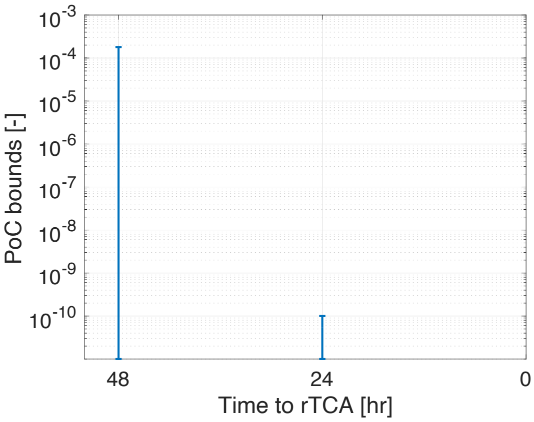


Figure 5.16: Evolution of PoC bounds for no-collision scenario.

5.3 Chapter summary

This chapter presented the applications of the methods developed in Chapter 3.

On the robust PoC estimation between SENTINEL 2B and FENGYUN 1C debris, the RPF was shown to efficiently compute robust probability bounds under both aleatory and epistemic uncertainty in the prior and likelihood distributions. The combined use of a UKF-based proposal and the polynomial propagator ensured a large and satisfactory effective sample size for the estimators' computation on the challenging test cases analysed. The filter proved to be a useful tool for the decision-maker in the different operational conditions considered. The output of the RPF is not the robust interval only. Indeed, it was shown that the full ECDFs of the quantity of interest could be reconstructed, thus providing informative information for different threshold values. Besides, for the last scenario in particular, the measurements at later times were simulated rather than acquired, and the RPF provided a valuable analysis to assess the potential impact of future observations on the PoC bounds. This process could assist the operator in deciding whether to manoeuvre immediately or acquire new measurements in highly uncertain scenarios.

With the numerical experiments we confirmed the correctness of the theoretical prediction of the performance of the filter and demonstrated how the filter can compute tight bounds enclosing all realisations of the PoC for a given threshold. We also demonstrated that in some cases epistemic uncertainty can lead to fairly wide bounds, which suggests that a poor consideration for epistemic uncertainty would give operators a false confidence in the value of the PoC.

The conjunction instances analysed in this example illustrated how the filter could handle epistemic uncertainty. However, more realistic and general scenarios would need to include epistemic uncertainty on both objects and on the prior mean.

Successively, the RPF was integrated within a robust Bayesian framework for optimal CAM allocation. This framework is constituted by three main modules that relate to each other. Other than the RPF for state estimation, the modules constituting the Bayesian pipeline are: an intelligent classification that automatically indicates the most convenient action to be taken when a close encounter is detected; the optimal CAM engine on the high-risk events to steer the thrusting spacecraft on a safe trajectory. Aleatory and epistemic uncertainties are considered on the state knowledge and manoeuvre errors, which are included to model inevitable errors in the CAM execution. The manoeuvre is designed such that the post-CAM upper collision probability is below a safety threshold set to 10^{-6}.

Such a system was applied to the same SENTINEL-FENGYUN conjunction analyses. In both the collision and no-collision cases, the initial uncertain state has been propagated and updated with measurements using the robust Bayesian estimator and the Robust Particle Filter before performing a risk assessment. Then, the Intelligent Classification System has been used for indicating the most suitable actions. Through both scenarios, the three possibilities have been shown: obtaining more measurement to reduce the uncertainty of the event, executing a CAM for avoiding a potential collision, and taking no further actions in case of a safe encounter. In all cases, after executing the appropriate action, the state has been re-estimated to re-evaluate the encounter risk. On the collision scenario, it is shown the CAM optimiser provides an optimal and robust manoeuvre accounting for the uncertainty affecting the state and including the uncertainty introduced by the manoeuvre itself.

Chapter 6

Optimal Scheduling

The content of this chapter was published in:
C. Greco, L. Gentile, G. Filippi, E. Minisci, M. Vasile, and T. Bartz-Beielstein, "Autonomous generation of observation schedules for tracking satellites with structured-chromosome GA optimisation", In *2019 IEEE Congress on Evolutionary Computation (CEC)*, 2019 [91];
L. Gentile, C. Greco, E. Minisci, T. Bartz-Beielstein, and M. Vasile, "Satellite tracking with Constrained Budget via Structured-Chromosome Genetic Algorithms", *Optimization and Engineering*, under review [86];
L. Gentile, C. Greco, E. Minisci, T. Bartz-Beielstein, and M. Vasile, "An optimization approach for designing optimal tracking campaigns for low-resources deep-space missions", In *70th International Astronautical Congress*, 2019 [97].

This chapter presents advancements in efficient and autonomous scheduling of tracking campaigns. The goal is to contribute toward an augmented autonomy, efficiency, and safety in the ground segment by introducing proper uncertainty quantification in observation and communication scheduling. To this end, an optimisation-based approach is developed to generate multi-source optimal tracking campaigns to improve the satellite state knowledge given noisy indirect measurements of its state.

The chapter is structured as follows. Section 6.1 describes the scheduling algorithm by first introducing the fitness metric using a sequential filtering scheme and then showing the tailored optimisation approach for the variable-size formulation used. Then, this approach is applied to the optimal tracking of a LEO satellite in Section 6.2, and a deep-space scenario in Section 6.3.

6.1 Multi-source tracking

In general, the task of tracking space objects in a dynamical environment is highly complex because it is affected by different sources of uncertainty, e.g. initial state dispersion, partially known dynamic parameters, noisy and sparse measurements. Hence, the proposed method aims to find the optimal observation schedule for minimising the uncertainty associated with the spacecraft state while considering limited budget resources. From another standpoint, the presented approach can also be employed for reducing the resources needed to meet an accuracy requirement on the state level of uncertainty.

This problem requires the coupling of two major disciplines in aerospace engineering. The outer loop is an optimisation routine for generating candidate observation schedules. The inner model involves an orbit determination routine for evaluating sensible objective and constraint functions associated with a specific schedule.

The free variables considered are: 1) the number of times a specific ground station is used; 2) the selection of adequate observation windows; 3) the number of observations to acquire. Consequently, without simplifying assumptions, the number of design variables changes among different solutions, and the observation scheduling falls under the area of variable-size mixed-discrete global optimisation.

6.1.1 Tracking model

This section shows the problem formulation and the fitness metric for optimal scheduling of tracking campaigns from heterogeneous sensors.

Scheduling formulation

The optimal scheduling is formulated as a sensor control problem [171], that is a decision-making process to find the best sensor actions $\mathbf{u}_k$, associated to the measurement $\mathbf{y}_k(\mathbf{u}_k)$, that optimise a fitness metric $\mathcal{J}$. Thus, the sensor action influences the observation directly and, as a consequence, the state knowledge. Uncertainty over the initial conditions, dynamical parameters, and observations values is accounted for in the scheduling model. Thus, the tracking model is again framed in a filtering probabilistic fashion as in Eq. (3.2). The posterior $p(\mathbf{x}_f|\mathbf{y}_{1:M})$ is again computed sequentially by iterating uncertainty propagation arcs and orbit determination updates.

In this thesis, offline scheduling is of interest, that is, the whole tracking schedule is optimised before the first satellite passage. Thus, the scheduling problem is formulated as

$$\begin{aligned} \mathbf{u}^*_{1:M} = \underset{\mathbf{u}_{1:M}\in\mathcal{U}_{1:M}}{\arg\min} \quad & \mathcal{J}\big(\mathbf{u}_{1:M}, p(\mathbf{x}_f|\mathbf{y}_{1:M})\big) \\ \text{s.t.} \quad & \mathcal{G}_k\big(\mathbf{u}_k, p(\mathbf{x}_k|\mathbf{y}_{1:k})\big) \in \mathbf{\Phi}_{\mathcal{G}_k} \ \text{ for } \ k = 1, \ldots, M \ , \end{aligned} \tag{6.1}$$

where $\mathcal{U}_{1:M}$ is the set of admissible sensor controls, $\mathcal{J}$ and $\mathcal{G}_k$ are functionals that return deterministic statistics from the posterior density and the actions, respectively, for the objective and constraints, and $\mathbf{\Phi}_{\mathcal{G}_k}$ is the set of admissible constraints.

Given the sensor control formulation, the specific problem is scheduling optimal tracking campaigns to reduce the satellite knowledge uncertainty using observations from a heterogeneous ground network.

Ground stations

The specific observation model depends on the nature of the tracking station employed, its geographical location, and the sensors available. Thus, each Ground Station (GS) is specified by the following characteristics:

- geodetic coordinates $(lat, lon, alt)_j$, that is latitude, longitude and altitude over the Earth reference ellipsoid;
- specific observation model h_j depending on the physical hardware available, e.g. range, range-rate, azimuth and elevation measurements;
- reference sensor accuracy at 90 deg elevation over the station's local horizon; at different elevation angles the covariance elements worsen as $\sim 1/\sin\left(El(\mathbf{x})\right)$ [162];

- the operational cost of the ground station $Cost_j$, depending on the tracking window.

The latter is used to impose budget constraints on the optimal schedules and avoid the trivial solution of using as many accurate measurements as possible. Such list of GS' characteristic is rather preliminary but it serves the scope to test the optimisation-based scheduling in this preliminary analysis. A more faithful modelling of physical characteristics and operational requirements is left for future developments of the scheduling method, e.g. slewing speed and reservation conflicts.

Given the network specification, the satellite could fall within the station Field of View (FoV) never, once, or multiple times, depending on the specific orbit and absolute time interval considered. Hence, a first deterministic visibility analysis is performed by numerically propagating the satellite reference trajectory and checking the condition $El(\mathbf{x}) > 0$ deg. From this, the total number of passages $\overline{p}_j$ in the GS-j FoV, and the contact window times $[T_{in}, T_{out}]_{p_j}$ for each passage $p_j = 0, \ldots, \overline{p}_j$ when the object is visible, are computed.

An optimal combination of ground stations, observation types, high-elevation passages, and close measurements helps to improve the estimation accuracy. A graphical 2D simplified depiction of a possible solution candidate is reported in Fig. 6.1, where only three ground stations are employed using a different number of observations.

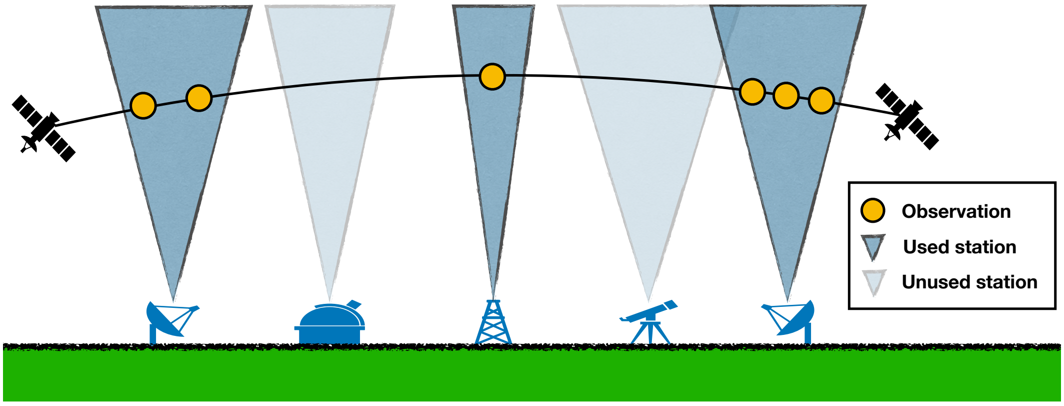


Figure 6.1: Sketch of space object passage over ground station network.

Fitness metric

The goal of tracking is to improve the state knowledge as much as possible by using measurements. Different metrics were examined to quantify the estimation accuracy given a candidate tracking campaign [162]. Among others, a common metric is the RMSE of the measurement residuals. However, small residuals do not necessarily imply an accurate state estimate as there may be unobservable state components. Another approach to quantify the estimation accuracy requires comparing estimates coming from different filtering schemes. However, this metric would be rather expensive to call numerous times within the scheduling search. Finally, the covariance is a metric that describes how spread the pdf is around the state. Furthermore, it is directly computed in Kalman-based filters or easily retrieved from samples in particle filters. As a downside, the covariance results typically optimistic and highly dependent on the selected (if any) process noise [162]. However, only a relative measure of the estimation accuracy between different candidate schedules is of interest in the optimisation. Thus, the process noise can be kept constant in each fitness evaluation. Therefore, the covariance was chosen as the accuracy metric of the estimation process for the scope of this analysis. As the optimisation works on a scalar quantity, the trace of the covariance is employed as fitness to quantify the confidence in the state elements

$$\mathcal{J} = \mathrm{Tr}\left\{\mathbf{\Sigma}_{\mathbf{x}_f|\mathbf{y}_{1:M}}\right\} . \tag{6.2}$$

Since at the time of scheduling the actual observations are not available yet, a covariance analysis [45, 47, 46] is employed to compute the final covariance by using the expected measurements' accuracy. Covariance analysis is mostly used for the navigation analysis of space missions to design suitable tracking strategies [46, 172, 173]. In this approach, OD is performed around the reference trajectory and the measurements are employed just to model the reduction of the second moment of the state distribution. Practically, this is achieved by simulating observations along the reference trajectory with zero noise.

A Sparse Gauss-Hermite Quadrature Filter (SGHQF) [174] is employed as a filtering algorithm suitable for normal distributions and non-linear dynamical and observational models. This nonlinear filter employs sparse grid quadrature rules to compute the mean and covariance prediction and update. The square root of the covariance matrix is employed in place of the covariance itself to improve the numerical stability of the method [162]. The covariance needed for Eq. (6.2) is then computed by multiplying the covariance square root by its transpose.

Budget constraint

In general, the more accurate measurements employed, the better the estimation accuracy would be. However, operating a tracking station has an associated monetary and human cost. Thus, each observation has an associated cost to quantify the budget corresponding to a specific schedule. In this model, the cost $Cost_j$ will depend on the sensors' accuracy, station architecture, ground operation costs, number of measurements, etc. The total cost $Cost$ is the sum of the individual costs, and the budget constraint is formulated as

$$Cost(\mathbf{u}_{1:M}) \leq \overline{Cost} \ , \tag{6.3}$$

where $\overline{Cost}$ is the maximum budget available.

6.1.2 Optimisation approach

The scheduling problem has a hierarchical nature. Indeed, by looking for example at Fig. 6.1, the number of measurements from a given station is a decision variable that is activated only once the station is actively employed in the tracking schedule. Thus, the number of variables changes for different candidate schedules depending on the sensor controls $\mathbf{u}_{1:M}$. The specific hierarchical structures for the applications addressed in this chapter will be discussed in Sections 6.2 and 6.3.

Thus, the search space has a varying size which entails additional challenges, e.g. the generation of the initial population and an increased complexity of the optimisation algorithm. GAs proved to be efficient for variable size global optimisation when employing appropriate encoding [175], and were employed for space trajectory design [176, 177, 178]. Specifically, hidden-genes GA was employed for the interplanetary trajectory optimisation [178]. In this variant, the variable size problem is translated into a fixed size one by introducing boolean activation genes, each associated with an original gene, to indicate whether a gene should be considered in the objective and constraint computation. As a downside, the number of free variables doubles while the operators work on the inactive variables as well.

Hence, in this work, a hierarchical multi-level chromosome structure is adopted as proposed by [176, 177]. In this scheme, known as Structured-Chromosome Genetic Algorithm (SCGA), the problem structure is encoded into the genes by employing vicinity and hierarchy relationships, thus enabling a meaningful exchange of information between chromosomes. SCGA bases its strategies on a revised version of standard GA operators [179]. In particular, SCGA encodes a candidate as a structure rather than a string. Then, the operators, nowadays established in stochastic fixed-length mixed-discrete optimisation, have been redefined to manipulate candidates characterised by different length and structure. In particular, the design of experiment, crossover, mutation operators, and repairing strategies have been redefined to work on hierarchical chromosomes. To make the search of the optimum more reliable, the proposed SCGA also uses other strategies such as the population restart mechanism and the partial local optimisation. This implementation of SCGA is available as open-source `SCGA R`-package at [180] and was developed by Lorenzo Gentile as part of his PhD thesis and research works, e.g. [90]. Since global optimisation is not a central theme of this dissertation, a detailed description of SCGA is not reported, but it can be found in [91, 86].

6.2 LEO space debris

In this section, first the details of the setup used in the LEO experiments will be described in Section 6.2.1, then the results are presented in Section 6.2.2.

6.2.1 Experimental setup

The dynamical system is parameterised in Cartesian coordinates, and several forces are included to model the motion in LEO faithfully [158]: Earth's gravitational force using the EGM96 geopotential model up to degree and order 10; atmospheric drag with Jacchia-Gill model; third-body disturbances due to the Moon and Sun gravitational pull; Solar Radiation Pressure (SRP) with a conical shadow model. The satellite epoch date is 2018 October 29 12:00 UTC, with reference Keplerian elements

$$\begin{aligned}\boldsymbol{\mu}_{kep_0} &= (a[\text{km}], e[\text{-}], i[\text{rad}], \Omega[\text{rad}], \omega[\text{rad}], \theta[\text{rad}]) \\ &= (6608.17, 1.61 \cdot 10^{-3}, 1.685, 5.662, 1.199, 1.589) \ .\end{aligned}$$

After conversion to Cartesian coordinates $\boldsymbol{\mu}_0$, this estimate is the mean of the Gaussian initial state distribution $p(\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_0; \boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$ with covariance set as

$$\begin{aligned}\boldsymbol{\Sigma}_0 = \text{diag}(&10^{-4}\ \text{km}^2, 10^{-4}\ \text{km}^2, 10^{-4}\ \text{km}^2, \\ &10^{-8}\ \text{km}^2/\text{s}^2, 10^{-8}\ \text{km}^2/\text{s}^2, 10^{-8}\ \text{km}^2/\text{s}^2) \ .\end{aligned}$$

The drag cross-section is set to 15.0 m^2. The mean solar flux considered is 106.4 in solar flux units, with a mean SRP cross-section of 1.625 m^2 and a SRP coefficient of 1.3. These system and orbital parameters are set to resemble a GOCE-like satellite. The tracking window lasts 8 hours, and it ends 2018 October 29 20:00 UTC.

Ground Station network

In this work, a hypothetical network composed of $N_{GS} = 9$ GSs is modelled. Three different GS configurations have been implemented in terms of sensor types, accuracy, and cost (see Table 6.1) to analyse the impact of GS' characteristics on the final schedules and the behaviour of the optimisation strategies. Each GS can measure one or more of the following scalar observables: the station-satellite Range (R), Range-Rate (RR) and Azimuth and Elevation (AzEl). The first two configurations (Conf-1 and Conf-2) have been randomly generated to construct a network of variegated GSs. The third (Conf-3) implements a simplified configuration in which the GSs have the same features and cost. This configuration serves as a baseline and helps to simplify the results' analysis by isolating the effects of the station-satellite geometry. The station coordinates have been exported in data kernels for NASA's SPICE toolkit [181] to retrieve their inertial

Table 6.1: Tracking network configurations reporting each station's geodetic latitude [deg], longitude [deg], 1σ accuracy in measurements for Range (R) [km], Range-Rate (RR) [km/s] and Azimuth and Elevation (AzEl) [rad] and cost per measurement.

GS	Latitude	Longitude	Feature	Conf-1	Conf-2	Conf-3
GS-1	35.43	243.11	R	NO	NO	5.00E-01
			RR	NO	7.76E-05	NO
			AzEl	3.64E-03	6.52E-03	NO
			Cost	1.04E-01	1.57E-01	1.25E-01
GS-2	-35.4	148.98	R	NO	1.36E-01	5.00E-01
			RR	NO	NO	NO
			AzEl	2.56E-03	NO	NO
			Cost	4.36E-02	7.04E-02	1.25E-01
GS-3	40.43	355.75	R	NO	2.24E-01	5.00E-01
			RR	4.39E-05	NO	NO
			AzEl	2.90E-03	NO	NO
			Cost	1.04E-01	8.36E-02	1.25E-01
GS-4	40.53	17.43	R	NO	7.14E-01	5.00E-01
			RR	1.13E-05	1.00E-04	NO
			AzEl	NO	2.44E-03	NO
			Cost	6.56E-02	3.00E-01	1.25E-01
GS-5	41.89	12.48	R	NO	NO	5.00E-01
			RR	2.66E-05	8.14E-05	NO
			AzEl	9.04E-03	NO	NO
			Cost	1.37E-01	8.59E-02	1.25E-01
GS-6	40.65	16.7	R	NO	9.74E-01	5.00E-01
			RR	NO	6.97E-05	NO
			AzEl	2.00E-03	7.75E-03	NO
			Cost	3.94E-02	3.56E-01	1.25E-01
GS-7	67.86	20.96	R	3.72E-01	9.10E-01	5.00E-01
			RR	NO	1.92E-05	NO
			AzEl	3.56E-03	NO	NO
			Cost	2.08E-01	2.25E-01	1.25E-01
GS-8	-72.01	2.53	R	4.78E-01	7.53E-01	5.00E-01
			RR	NO	2.53E-05	NO
			AzEl	2.94E-04	NO	NO
			Cost	8.73E-02	2.06E-01	1.25E-01
GS-9	78.23	15.4	R	1.09E-01	5.18E-01	5.00E-01
			RR	6.28E-05	1.26E-05	NO
			AzEl	8.47E-03	NO	NO
			Cost	1.93E-01	1.62E-01	1.25E-01

state at different epochs while accounting for Earth's rotation.

In the test case, an 8-hour time span is considered to track a satellite in LEO. As a result of the short orbital period, the object falls in the stations' FoVs multiple times. The geometry of the satellite passes over the GSs considered is visualised in the sky plots in Fig. 6.2, which displays the azimuth and elevation angles over the GS' local horizon. The total number of potential tracking passages over all the stations is equal to 21. The maximum elevation angle ≈ 71 deg is achieved over the tracking station GS-9, which in addition sees the satellite in multiple passes. Good elevation angles ≥ 50 deg are also realised above stations GS-3, GS-4, GS-8, whereas the worst ones correspond to station GS-2 and GS-5.

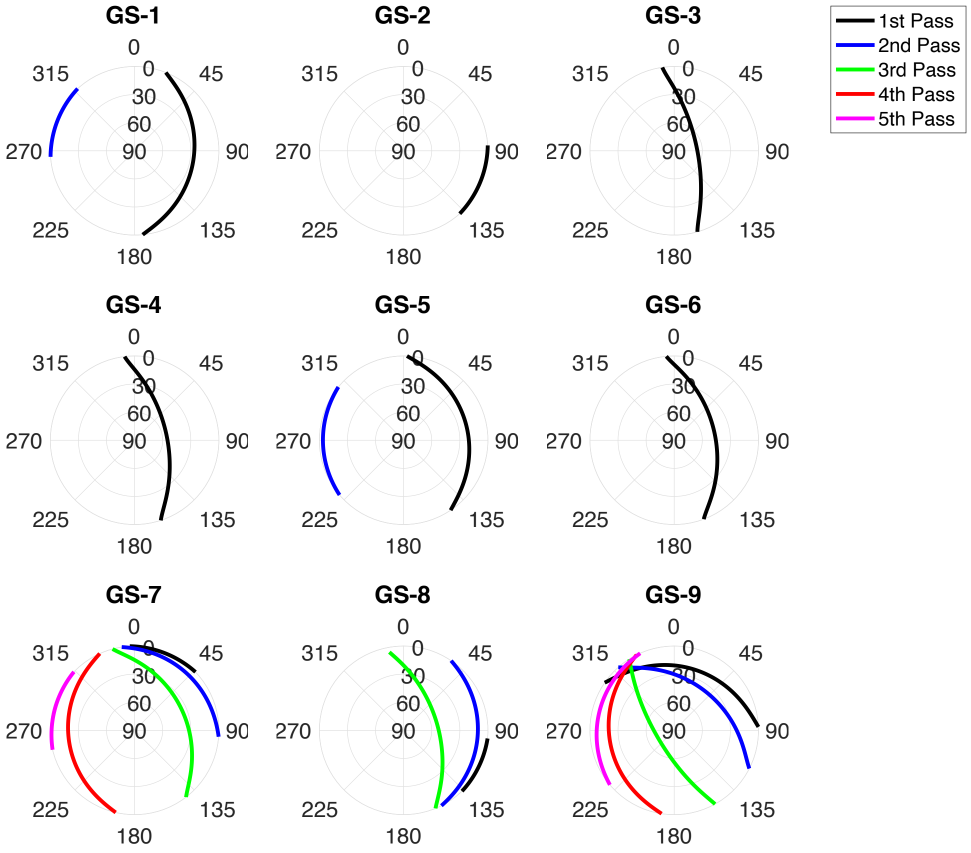


Figure 6.2: Sky plots of satellite passes over the ground stations, with concentric circles indicating different elevation levels, while the angular quantity represents the azimuth measured eastwards from the local north. Different colours indicate different satellite passes over the same station.

Sensor controls

It is expected that the optimal tracking schedule would have accurate measurements at high elevation angles to improve the satellite visibility. Besides, the final accuracy depends also greatly on the observation timing, that is, when the observations are acquired along the trajectory. All these features make the problem highly multi-factorial.

The actual measurement and its estimated accuracy depend on the time at which it is taken and the station-satellite relative geometry. The sensor action can influence these factors and therefore helps to acquire a measurement with a greater information content. In this example, the action should first describe whether a measurement is taken from GS-j or not, then should set the number of tracking arcs, the specific orbital passages, and the times within each passage. The control variables encoded in the sensor action are reported in Table 6.2. Thus, a boolean variable sets if the ground station GS-j is used. If it is employed, an integer variable describes in how many orbital passages measurements are acquired (up to $\overline{p}_j$), and a corresponding number of discrete variables is used to pinpoint the specific passage. Each orbital passage can be used only once. Thus, each of these discrete variables should have a unique value to have a feasible schedule. Finally, for each orbit passage, an integer variable describes the Number of Measurements (NoM) to acquire, and continuous variables set the specific measurement times. Hence, the scheduling problem looks for the optimal combination of tracking stations, orbital passages, and observation instances to optimise the fitness metric and improve the system observability.

Table 6.2: Free variables and types encoded within sensor action $\mathbf{u}_k$.

Description	Variable Type	Values
Use ground station j	Discrete	ON/OFF
Number of passages to use	Integer	$[0, \overline{p}_j] \in \mathbb{N}$
Indexes of passages to use	Discrete	$[1, \overline{p}_j] \in \mathbb{N}$
Number of observations per passage	Integer	$[0, \mathrm{NoM}_{max}] \in \mathbb{N}$
Times of observations per passage	Continuous	$t \in [T_{in}, T_{out}]_j^p$

Tracking encoding

Once the optimiser has set a collection of sensor controls, a hierarchical structure with three levels has been employed for the chromosome encoding, as depicted in Fig. 6.3. The top class *Ground Station* indicates the number of different satellite passes in which a specific ground station will be employed, and therefore N_{GS} genes of this class will

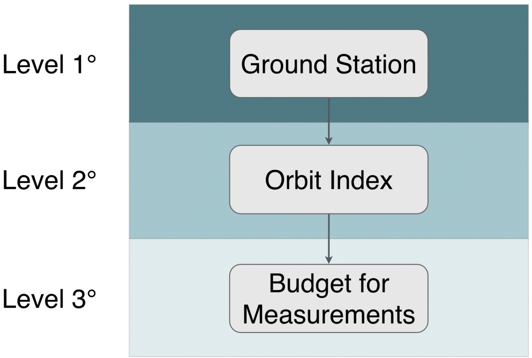


Figure 6.3: Chromosomes-hierarchy in the SCGA's formulation.

be present in all the solutions. However, because different GSs have a different number of orbital passages (see Fig. 6.2), the upper bound $\overline{p}_j$ depends on the specific GS. The intermediate class Orbit Index (OI) represents the indexes of the orbital passages in which observations are acquired. The bottom gene class Budget for Measurements (BfM) indicates the percentage of the total available budget for each specific orbital passage of a given GS. The actual NoM_j to be acquired is expressed by the equation

$$\mathrm{NoM}_j = \left\lfloor \frac{\mathrm{BfM}_j}{Cost_j} \right\rfloor \tag{6.4}$$

where BfM_j is the allocated BfM to GS-j.

As an example, two candidate schedules are shown in Fig. 6.4 by depicting the multi-level structure for GS-1, GS-2 and GS-9 only. The solution in Fig. 6.4(a) allocates

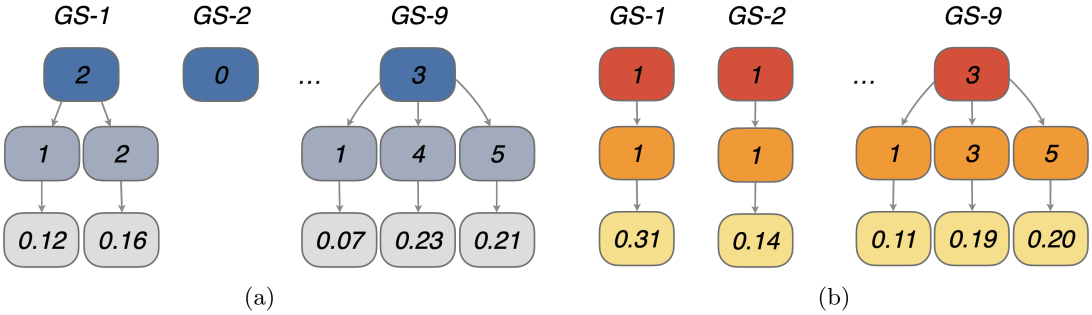


Figure 6.4: Example of two candidate observation schedules.

observations as: GS-1 on 2 different orbital passages OI-1 and OI-2 with BfM of 0.12

and 0.16 respectively; GS-2 has no observations; GS-9 on 3 different orbital passages OI-1, OI-4, and OI-5 with the reported BfM. The schedule in Fig. 6.4(b) has observations for: GS-1 on 1 orbital passage, OI-1; GS-2 on 1 orbital passage, OI-1; GS-9 on 3 orbital passage, OI-1, OI-3, and OI-5. The two candidates have differences on each structure level. Indeed, the second candidate uses GS-2 while the first does not, the second schedule requires observations on the OI-3 of GS-9 whereas the first employs the orbital passage OI-4, and the BfM are all different.

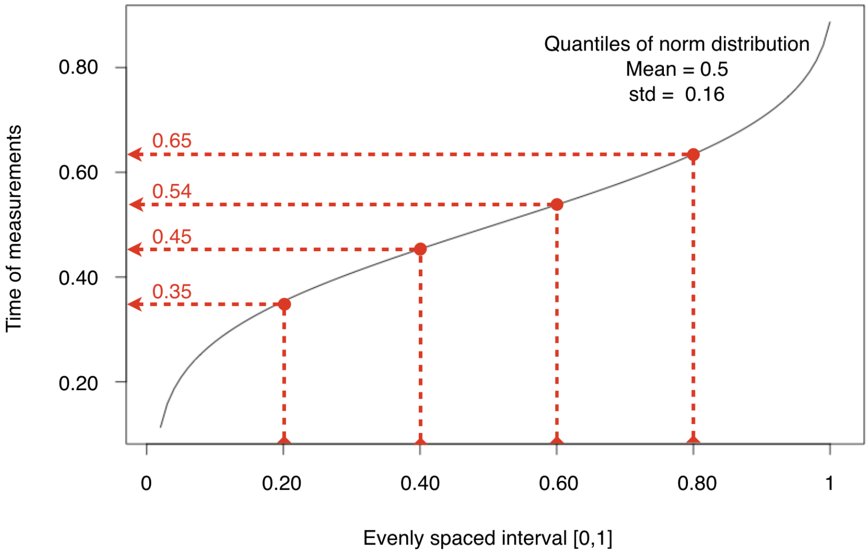


Figure 6.5: Cumulative distribution function employed to compute measurement times within the FoV.

The accuracy of the estimation process strongly depends on the exact measurement times. The choice of measurement times within an orbital passage in the station FoV comes from the compromise of two conflicting factors: acquiring the most accurate measures by concentrating them around the maximum elevation point; acquiring the most independent observations possible spreading them in the whole observation window. Hence, in this work, given the NoM_j, the measurement times are computed as the values of the normal cumulative distribution with mean $= 0.5$ and standard deviation $= 0.16$ at the points that evenly divide the interval [0,1] NoM_j times. The times are then linearly scaled within the corresponding time interval $[T_{in}, T_{out}]_j^p$ (see Table 6.2). A graphical representation is given in Figure 6.5. This method concentrates the measurements around the maximum elevation point (scaled time 0.5) while adding observations at the interval extrema as NoM increases.

Schedule cost

The cost of operations has been assumed to change linearly with the measurement accuracy between the bounds defined in Table 6.3. Thus, if a station GS-j takes a specific measurement $M = \{\text{R,RR,AzEl}\}$ with accuracy Acc_j^M (at 90 deg), then the cost of a single observation is

Table 6.3: Minimum and maximum 1σ accuracy and cost for each measurement type $M = \{\text{R,RR,AzEl}\}$.

Parameter		Symbol	R [km]	RR [km/s]	AzEl [rad]
Accuracy	Min	Acc_{min}^M	10^{-4}	10^{-6}	10^{-4}
	Max	Acc_{max}^M	1	10^{-4}	10^{-2}
Cost	Min	$Cost_{min}^M$	$5 \cdot 10^{-2}$	$25 \cdot 10^{-2}$	$25 \cdot 10^{-2}$
	Max	$Cost_{max}^M$	$2 \cdot 10^{-1}$	10^{-1}	10^{-1}

$$Cost_j^M = Cost_{min}^M + (Acc_j^M - Acc_{min}^M) \cdot \frac{(Cost_{max}^M - Cost_{min}^M)}{(Acc_{max}^M - Acc_{min}^M)} \ . \tag{6.5}$$

This unit cost is then summed over the types of observations and the number of measurements acquired by GS-j to compute its operational cost for an orbital passage

$$Cost_j = \text{NoM}_j \sum_M Cost_j^M \ . \tag{6.6}$$

Hence, the total tracking cost required for the budget constraint (see Eq. (6.3)) is given by

$$Cost(\mathbf{u}_{1:M}) = \sum_j Cost_j \ . \tag{6.7}$$

The threshold value $\overline{Cost}$ is varied for different runs of the optimisation to analyse the impact of limited versus large budgets in the estimation accuracy. The maximum budget is varied in the range from 1.5 to 9 with a step of 1.5.

Search algorithms

The free variables for SCGA are summarised in Table 6.4. SCGA is tested using three different configurations: SCGA with population restart mechanism, SCGA with local search and plain SCGA (see [86]). The restart and local search strategies are activated only after 50 consecutive generations do not improve the best-found fitness above 1%.

The proposed optimiser is tested against a standard GA [182] and the hidden-genes GA [178]. The former was not designed to tackle variable-sized problems, thus a special encoding had to be designed to adapt it. In this case, the adopted strategy is to activate the top and intermediate levels (see Fig. 6.3) for each GS and passage, and the free variables are the BfM for every passage only. Thus, the decision variables are 21 real variables, one for each time the satellite falls in the FoV of at least one station. As said, the hidden-genes employs activation genes to control the activation of a specific GS and OI while keeping the typical GA architecture and use of operators. Thus, the free variables are 42 in total, 21 real-valued as before and 21 categorical representing activation genes. The control variables with the corresponding bounds for the comparison algorithms are reported in Table 6.4.

Table 6.4: Decision variables for different optimisation algorithms in observation scheduling problem.

Algorithm	Gene	Type	Size
SCGA	*Ground Station* OI *Budget*	Integer Discrete Real	9 variable variable
Hidden-genes GA	*Budget* *Activation*	Real Discrete	21 21
GA	*Budget*	Real	21

For all the strategies, the number of candidates in the population is set to 30 and the maximum objective evaluations to $13,500$. Additional optimisations of hidden-genes GA and SCGA with population restart are run with maximum objective evaluations set to $27,000$ (respectively labelled GA-Hidden-DoubleEval and SCGA-Restart-DoubleEval in the results) to check the convergence of the algorithms, only for the instances with maximum budget equal to 3 and 9. To have statistical significance, 50 independent runs initialised with different seeds are performed for each combination of GSs network configuration (see Table 6.1) and a maximum allocated budget.

6.2.2 Results

The results for all the GS network configurations, budget allocations, and optimisation algorithms are displayed in Figs. 6.6 and 6.7. The covariance traces of the 50 runs' best solutions are reported using box and violin plots, respectively, with a linear and logarithmic scale. As expected, the optimal tracking schedules of Conf-1 and Conf-2 deliver a significantly higher estimation accuracy than the Conf-3 schedules because the latter uses only coarse range measurements. The boxes' arrangement in Fig. 6.7 helps to show how increasing the allocated budget results in more accurate estimations. For tighter budgets, SCGA manages to deliver tracking schedules whose estimation accuracy is several orders of magnitude better than the one of plain GA and hidden-genes GA. Thus, severe budget constraints undermine classical genetic algorithms' ability to find optimal solutions for this scheduling problem. On the other hand, for less constrained scenarios, the estimation accuracy's difference resulting from different methods is not statistically significant.

The convergence history of the algorithms is displayed in Fig. 6.8. SCGA shows a significantly better convergence velocity especially in the low-budget configuration in Fig. 6.8(a). Besides, in the less constrained scenario reported in Fig. 6.8(b), plain and hidden-genes GAs are almost anytime outperformed by SCGA reinforcing the hypothesis that traditional fixed-size optimisation strategies actually struggle to face variable-size problems even when reformulated.

A visualisation of the 4 best solutions for each algorithm in the considered configurations is shown in Fig. 6.9. These plots show the measurements allocation to each GS and the fitness value. For each GS, the number of rectangles indicates the number of orbital passages tracked, and the number in a rectangle specifies the OI. The colours discriminate the algorithm employed as usual, whereas the hues indicate the NoM allocated for each pass with brighter shades indicating more measurements. Furthermore,

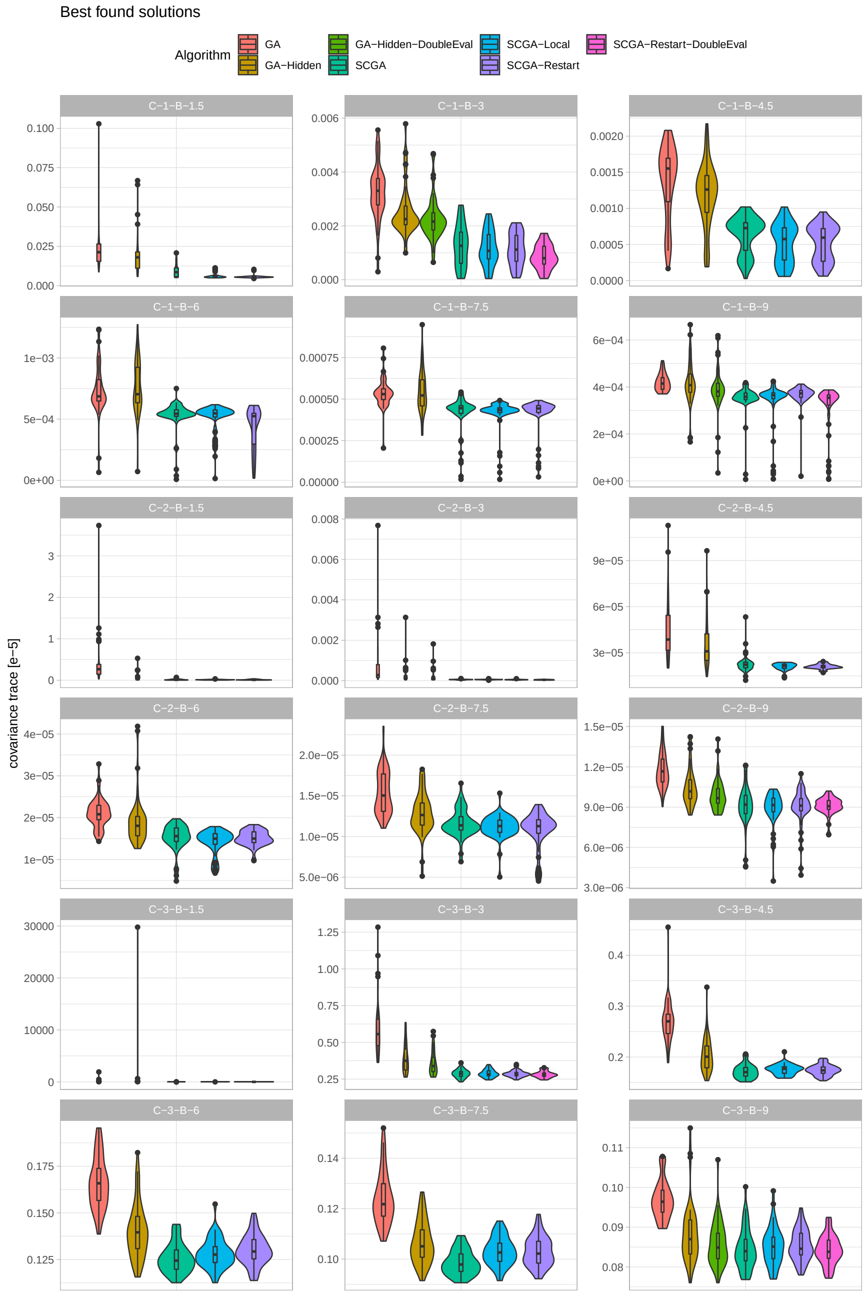


Figure 6.6: Box and violin plots showing the accuracy of the best schedules for each algorithm in the 50 independent runs. The labels indicate the combination of GS configuration (C: Conf) and maximum budget (B: Budget).

given the floor function in the definition of NoM_j (see Eq. (6.4)), some resources are discarded when the budget allocated to a GS is not exactly a multiple of its cost of

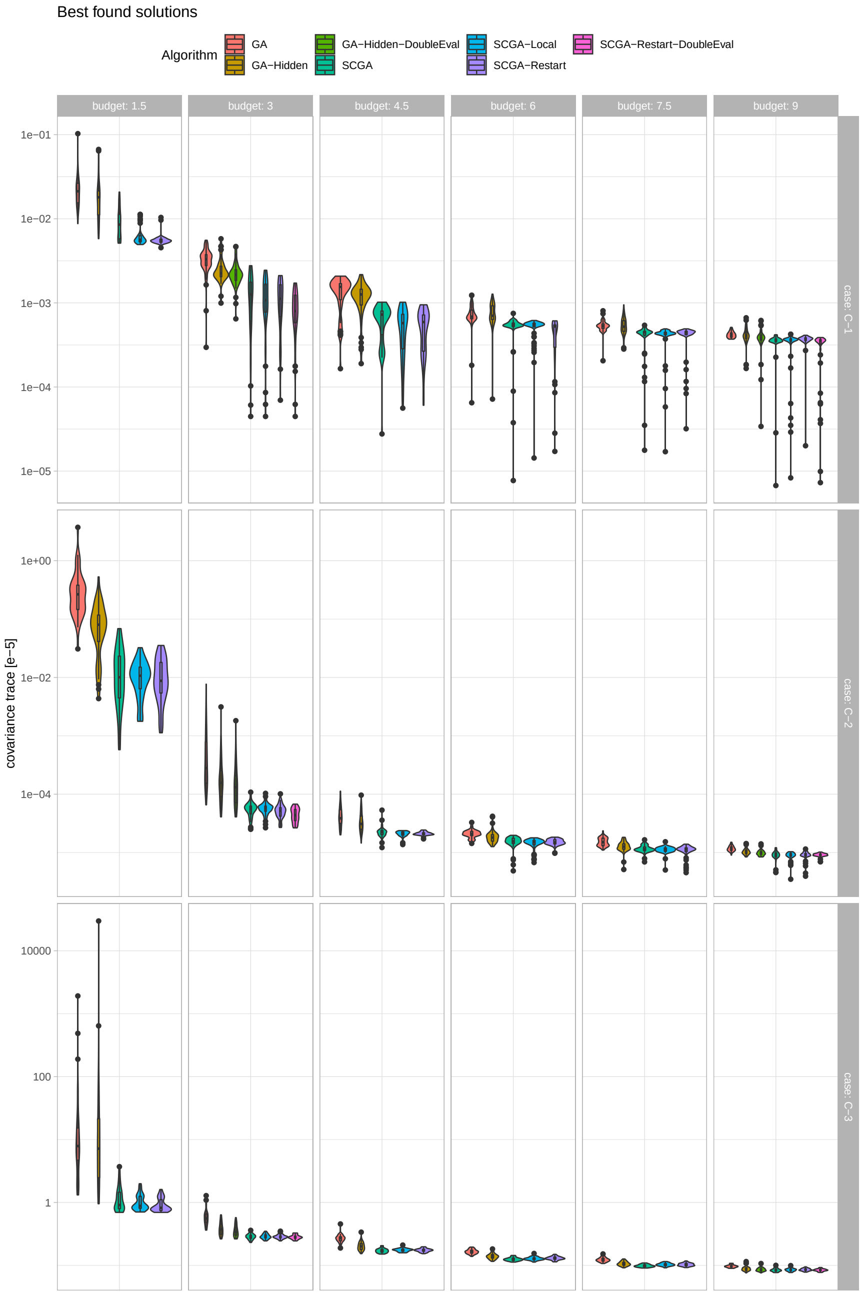
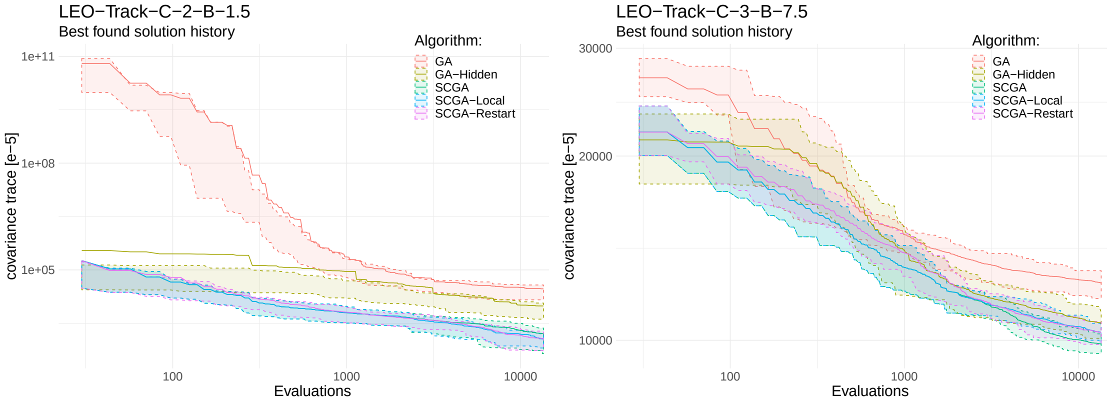


Figure 6.7: Box and violin plots showing the accuracy of the best schedules for each algorithm in the 50 independent runs using logarithmic y-axis. The labels indicate the combination of GS configuration (C: Conf) and maximum budget (B: Budget).

employment. Thus, an efficiency metric defined as the percentage of unused budget

$$\text{Eff} = \frac{\sum_j Cost_j}{Cost} \cdot 100 \tag{6.8}$$



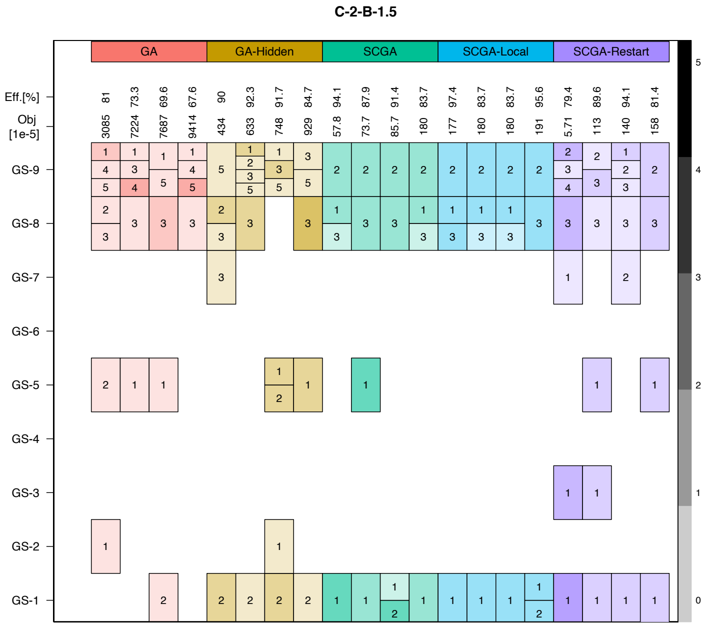
(a) $\overline{Cost}$=1.5 , Conf-2.

(b) $\overline{Cost}$=7.5 , Conf-3.

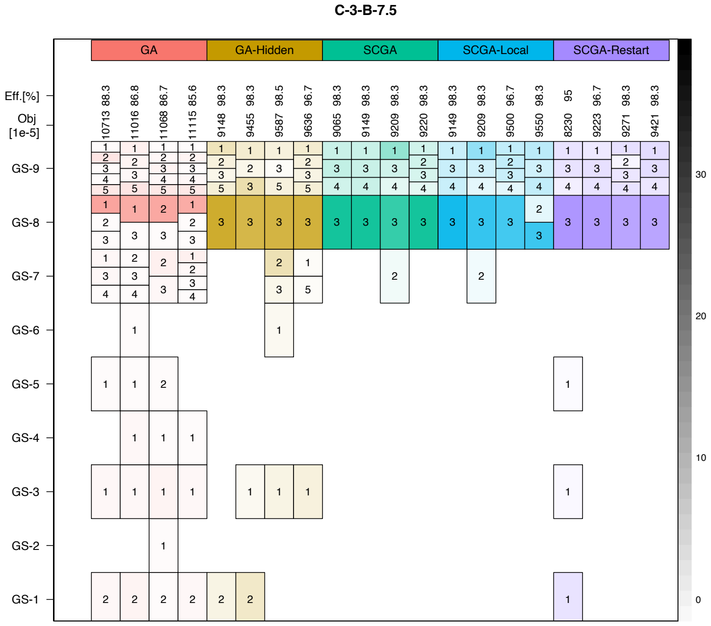
Figure 6.8: Anytime performance plot for two GS and budget configurations.

is reported in Fig. 6.9 as well. Looking at Fig. 6.9(a), the tracking schedules found by the SCGA variants are very similar while there are some important differences with the ones optimised by fixed-size GAs. As an example, SCGA always uses the first high-elevation passage over GS-1 (see Fig. 6.2), which helps to improve the estimation accuracy early in the tracking window. GS-8 and GS-9 are always employed having very high elevation angles, GS-5 is employed in several solutions as well due to its accurate RR measurement capability, which is highly informative, at a relatively low cost (see Table 6.1). The high-elevation R measurement from GS-3 is utilised only by SCGA with restart mechanism. In the second configuration, depicted in Fig. 6.9(b), the optimal tracking schedules by hidden-genes and SCGA are similar, whereas the results from GA are still significantly different. One difficulty seems to stem from the inability of standard GA to totally discard a station, possibly due to the chromosome encoding. Indeed, GS 1 to 6 are employed with a very low number of observations. Furthermore, by looking at the allocation efficiency metric, standard GA does not manage to allocate the whole budget efficiently, and therefore it discards tracking resources. Again, all the best schedules employ GS-8 and GS-9 primarily because of their high elevation passages. Multiple passages over GS-9 are tracked to acquire measurements during different revolutions rather than only the second passage like in Conf-2.

Fig. 6.10 shows the distribution of the observation over the whole tracking window



(a) $\overline{Cost}$=1.5 , Conf-2.



(b) $\overline{Cost}$=7.5 , Conf-3.

Figure 6.9: Visualisation of the four best solutions found by all the algorithms on two tracking configurations.

for all the 50 final solutions. The rectangles composing the histogram are also colour-coded for the ground station employed to take the measurement. For a given GS

Figure 6.10: Distribution of the observations' times over the tracking window in all the final solutions.

configuration and budget, all the SCGA share the same qualitative distribution. By increasing the budget for a fixed GS configuration, the number of employed observations tends to increase sharply. This implies that often a suitable schedule is found already at lower budgets. For Conf-2 and Conf-3, the observations are well spread in time and tend to focus at the end of the tracking campaign when the budget increases. For Conf-1, the observations are taken mainly at the middle of the observation interval. Besides, larger budgets tend to concentrate on specific most-informative GSs and passages, that is GS-4 for Conf-1, and GS-8 and GS-9 for Conf-2 and Conf-3. This difference is due to the significantly higher cost of operating GS-4 in the second and third GS configurations (see Table 6.1).

Fig. 6.11 shows the observation distribution for the best-found schedule only. The y-axis now reports the final estimation accuracy while the colours and marks discriminate the algorithms. For lower budgets, most of the measurements are acquired toward the end of the tracking campaign such that the reduced uncertainty has less time to

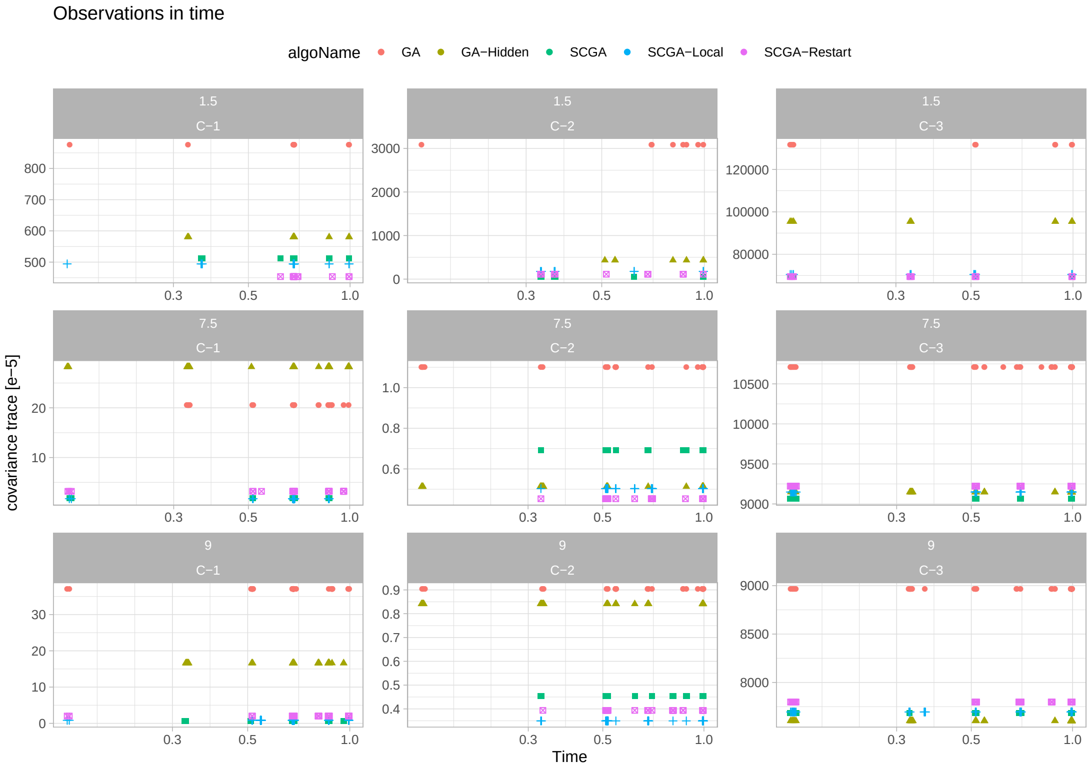


Figure 6.11: Distribution of the observations over the tracking window in the best found solutions by all the optimisers.

grow due to dynamical propagation. As the budget increases, earlier observations are employed as well to improve the accuracy more gradually during the tracking window.

In all the configurations, the variable size optimisation demonstrated to be more reliable and better performing than fixed-size GAs. This performance difference was even more evident for tighter budgets which significantly shrink the search space.

6.3 Deep-space tracking

In this section, the setup of a low-budget deep-space mission test case is described in Section 6.3.1. The observation schedules obtained by SCGA are then presented in Section 6.3.2.

6.3.1 Experimental setup

A proper definition of the observation schedules of spacecraft cruising in the deep-space is of utmost importance for the success of the missions. However, the methodologies used nowadays often need a priori parameter specification that limits the search for optimal observation schedules. One of these is the number of observation campaigns to be performed. Based on previous experience, common practice is to fix this parameter at the early stage of the schedule design and keep free the other parameters defining the observation campaign. Indeed, in cases where the operator already gained a deep knowledge about the mission, this approach may be successful and lead to optimal low-budget observation schedules. In others, this can represent a severe limitation and compromise the quality of the state estimation of the spacecraft and the success of the mission. In light of these considerations, the approach used in this section aims at increasing the degrees of freedom, keeping the number of observation campaigns as a free variable of the observation schedules design process.

Problem scenario

The goal is to track a low-thrust spacecraft departing from the Earth in its propellant-optimal trajectory to rendezvous the near-Earth asteroid 99942 Apophis (2004 MN_4). The ephemerides of Earth and Apophis at different epochs are computed with NASA's SPICE through high-fidelity prediction routines [181].

An interplanetary trajectory to 99942 Apophis is considered as an application to test the tracking scheduling. The nanoSat departs from an Earth escape trajectory on the 22nd of October 2026 and reaches Apophis on the 10th of April 2028 for a total time of flight of 537 days. Only the Sun's central gravitational force and the high-thrust impulses are considered in the dynamical model as the primary interest is on the OD schedules optimisation. The reference orbit for the interplanetary transfer is shown in Fig. 6.12.

The uncertainty considered on this transfer is due to

- knowledge error for the initial conditions modelled as a multivariate Gaussian

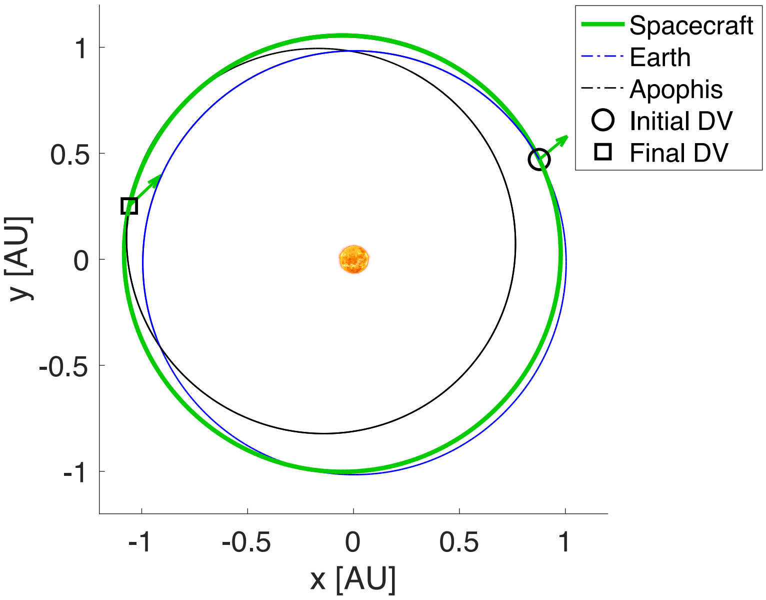


Figure 6.12: Reference trajectory of spacecraft from Earth to Apophis.

distribution;

- execution errors for TCMs modelled with the Gates' model [169] (already introduced in Section 5.2.2); the TCMs are executed two days after the end of each tracking arc;
- noisy observations $\mathbf{y}_k$ with likelihood distribution, modelled as normal, accounting for sensor and external noises.

Table 6.5 reports the parameters for the first two uncertainty models: the standard deviation for the initial distribution and the Gates' parameters. The observation accuracy is reported in Table 6.6 where the GSs parameters are detailed.

Ground station network

In this example, three GSs constitute the network for tracking the spacecraft during its interplanetary transfer. Each of them can measure the station-spacecraft relative R, RR and AzEl, but the accuracy and operating cost vary for the three of them. The accuracies and costs have been set such that GS-1, GS-2, and GS-3 provide respectively low-, medium- and high-fidelity observations as detailed in Table 6.6.

Table 6.5: 1σ standard deviations for position and velocity in RTN components of initial distribution, and execution error parameters for Gates' model.

Uncertainty	Component	Value	
Initial Dispersion	Position (RTN)	[166, 166, 166]	[m]
	Velocity (RTN)	[70, 10, 15]	[mm/s]
Execution Error	Fixed Pointing	3.0	[mm/s]
	Prop. Pointing	7.0	[mrad]
	Fixed Magnitude	5.0	[mm/s]
	Prop. Magnitude	3.3e-3	[-]

Table 6.6: Ground stations' accuracy and operating costs.

Ground Station	Measure	1σ std.	Cost
GS-1	Range	1	0.05
	Range rate	10	0.025
	Azimuth and Elevation	50	0.025
GS-2	Range	0.1	0.1
	Range rate	4	0.05
	Azimuth and Elevation	20	0.05
GS-3	Range	0.02	0.2
	Range rate	1	0.1
	Azimuth and Elevation	2	0.1

As the adopted schedule is a trade-off between final accuracy and operating cost, several budget constraints have been tested, ranging from 0.2 to 15.

SCGA encoding

A full tracking campaign is composed of multiple OD arcs distributed at different times during the trajectory. In this example, a single OD arc is defined by four quantities: the time of the last observation; the ground station to employ; the type of measurements; the NoM in that arc. Thus, schedules with a different number of OD arcs are encoded by a different number of design variables making the use of variable size optimisation advantageous.

The adopted formulation aims at reducing the number of free variables generally considered by applying the concept of hierarchy. In this case, the hierarchy is consti-

Table 6.7: Free variables types and possible values for scheduling of deep-space tracking schedules.

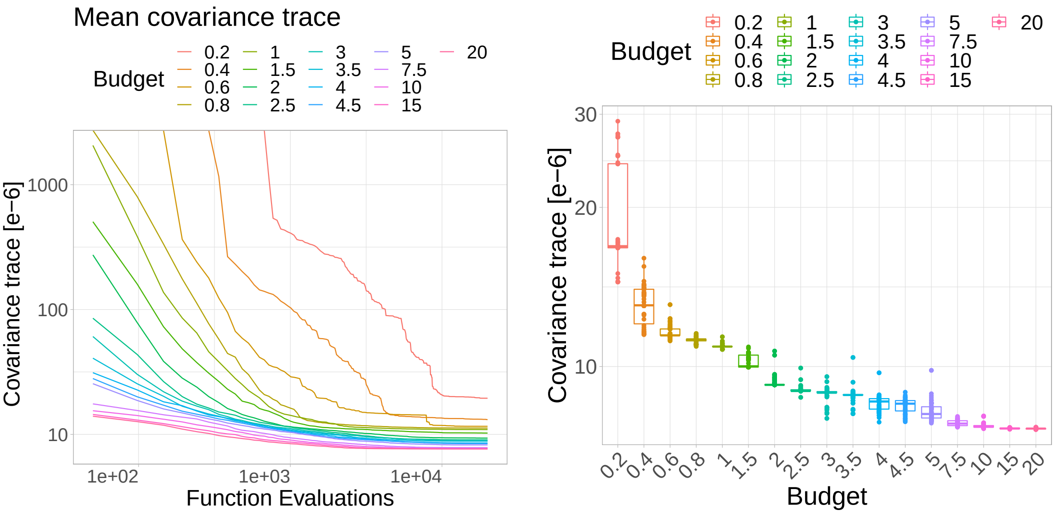
Top level			
Gene	Variable type	Lower Bound	Upper Bound
Number of OD arcs	Integer	1	10
Bottom level			
Gene	Variable type	Lower Bound	Upper Bound
Ending time	Real	0	1
Number of measurements	Integer	1	20
Gene	Variable	Possibilities	
Measurement type	Discrete	$[R, RR, AE, [R, RR], [R, AE], [RR, AE], [R, RR, AE]]$	
Ground Station	Discrete	[GS-1,GS-2,GS-3]	

tuted by two levels. The top class is the number of OD arcs for a specific schedule. The bottom level encodes all the other genes that define a single OD arc as described above. The variable types, lower and upper bounds, or discrete possibilities for each of these genes are reported in Table 6.7. The time has been scaled in $[0, 1]$ as the departure and arrival dates are known. The population size is set to 50, and for each budget level 58 instances with different random seed have been analysed to obtain statistically significant outcomes.

6.3.2 Results

The convergence history during the optimisation for the best-found schedule is depicted in Fig. 6.13. In Fig. 6.13(a) the history of the mean of the performance of the best-found solution during the optimization is depicted for each configuration investigated. As expected, this picture shows that better estimation accuracy is achieved for higher budgets. This result is even more evident from Fig. 6.13(b) which reports box-plots of the covariance trace as a function of the allocated budget. The width of the boxes at low budgets indicates the difficulty in finding optimal tracking schedules under severely limited resources across multiple simulations.

Fig. 6.14 shows histograms counting the values of the genes (see Table 6.7) for the optimal tracking schedule in the 58 independent runs for varying budgets. Specifically, Fig. 6.14(a) counts the number of OD arcs for the optimal schedule. For most of the

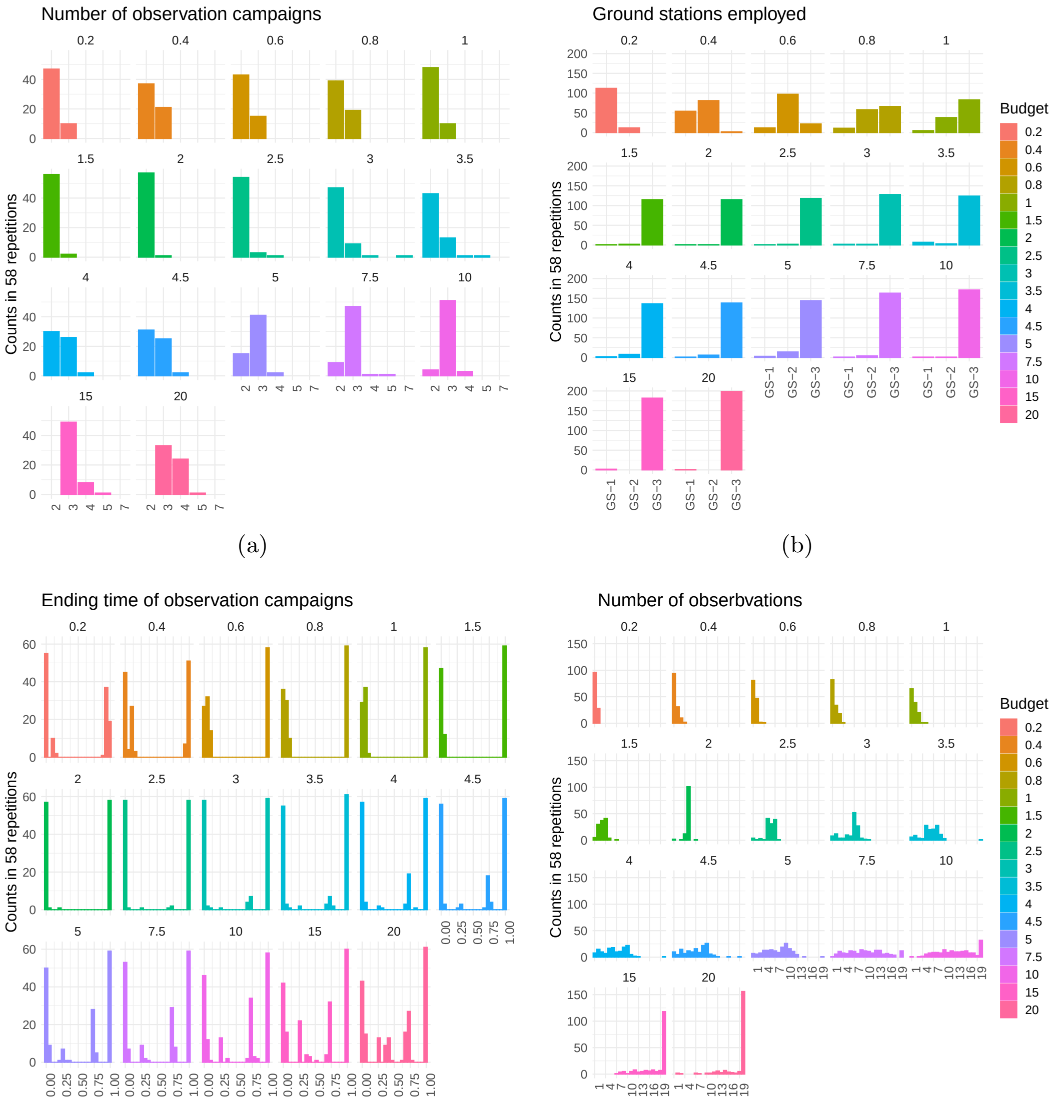


(a)

(b)

Figure 6.13: (a) Convergence history of the mean of the best found schedule and (b) box-plots of optimal fitness value as function of budget over 58 independent simulations.

budgets, only one or two OD arcs are employed, whereas three or more arcs appear for higher budgets. Increasing the budget, the optimal number of OD arcs is 3 for the majority of the simulations. Fig. 6.14(b) reports the GS usage for the optimal schedules. The medium- and high-fidelity stations, that is, GS-2 and GS-3, dominate the schedules for budgets over 1. In contrast, schedules with tighter resources tend to employ the low-fidelity GS-1 more. Fig. 6.14(c) displays the time distribution of the OD arcs. The results suggest that tracking at the beginning and toward the end of the trajectory yields a more accurate estimation. Indeed, from a navigation analysis take, early measurements are beneficial because: they reduce the initial dispersion uncertainty, therefore decreasing the dispersion growth with dynamical propagation; they enable early, thus more efficient, impulsive TCMs. Late measurements are crucial in improving the state knowledge at arrival and adjusting the final braking manoeuvre to rendezvous the asteroid. Allocating a larger budget and, as a consequence, increasing the number of OD arcs, the scheduling method found that it is beneficial to first track toward the 75% of the mission, and secondly toward the 25%. Indeed, intermediate observation windows help to reduce the uncertainty growth due to pure uncertainty propagation and enable TCMs with large time margins before the final rendezvous.



(a) (b)

(c) (d)

Figure 6.14: Analysis of the free variables considered in consideration of the budget imposed. The histograms show the occurrences of the values assumed over the 58 solutions obtained for each configuration.

Finally, Fig. 6.14(d) counts the number of measurements per OD arcs in the 58 optimal schedules. For low budgets, very few measurements are employed, whereas for high budgets, almost all the OD arcs are characterised by the maximum number of measurements. The NoM increases almost monotonically as more budget is allocated. Indeed, the transition from tight to large budgets is rather smooth, with more observations distributed over a larger range of NoMs.

6.4 Chapter summary

An observation scheduling method was developed and tested on different tracking scenarios.

First, the optimisation-based scheduling approach to automatically generate optimal tracking schedules was introduced to enhance the knowledge of the satellites' state evolving under the effect of strong dynamical perturbations. This work contributes to the crucial need for methodologies to efficiently allocate ground resources to the constantly increasing number of objects, operational and non, in space. All the ground pipeline consisting of stations' modelling, observation simulation, uncertainty propagation, and orbit determination was designed. A structured-chromosome GA was proposed to tackle the variable-size optimisation problem efficiently. The algorithm was implemented to work with any dynamic, measurement model, and station network such that test cases with different constraints and characteristics can be tackled.

Then, a test case on a LEO satellite was analysed. In particular, the optimisations were repeated using three GS network configurations and using six levels of severity for the maximum available budget limitation. The characteristics of the GS network available were varied by changing the measurement accuracy of each GS and its architecture together with its cost of utilisation. Three settings of SCGA were implemented and tested: SCGA with population restart mechanism, SCGA with local search and plain SCGA. For comparison, standard and hidden-genes GAs were tested on the same problems.

A second application was analysed for limited-resources allocation strategies in deep-space objects tracking with stations characterised by different cost and accuracy. In this test case, TCMs and their execution errors are included after each observation arc. In this case, the effect of new measurements improves the knowledge at the observation instance but introduces errors on the velocity, which tends to worsen the future prediction accuracy.

The results of both test cases indicated that the presented methodology could successfully enhance resource allocation strategies in space object tracking problems, in particular under severe budget limitations. The results show the clear benefits granted by employing a variable-size algorithm regardless of the problem instance. The advantages of using SCGA, in comparison to standard or hidden-genes GA, were even more evident for severe budget limitations, that is, when only limited portions of the search space yield feasible schedules.

Chapter 7

Low-thrust Trajectory Design

The content of this chapter was published in:
C. Greco, M. Di Carlo, M. Vasile, and R. Epenoy, "Direct multiple shooting transcription with polynomial algebra for optimal control problems under uncertainty", *Acta Astronautica*, 2020 [87].

In this chapter, *IPANeMA*, developed in Section 4.1, is applied to the robust optimisation of a low-thrust space trajectory to the asteroid Apophis. This test case aims at assessing the suitability and performance of the approach implemented in *IPANeMA* when applied to the preliminary design of robust space trajectories.

7.1 Mission scenario

The goal is to compute the propellant-optimal rendezvous to the near-Earth asteroid 99942 Apophis (2004 MN_4) with a low-thrust spacecraft departing from the Earth. The ephemerides of Earth and Apophis at different epochs are computed with NASA's SPICE through high-fidelity prediction routines [181]. The initial date of the interplanetary leg is 22/10/2026 for a total time of flight of 628 days. The engine has a maximum thrust of $T_{max} = 53$ mN, a specific impulse of 3080 s, and an initial mass of $m_0 = 644.3$ kg.

The three-dimensional motion of the spacecraft is described with nonsingular equinoctial elements [183]:

$$\begin{aligned}
&a \\
&P_1 = e \sin(\Omega + \omega) \\
&P_2 = e \cos(\Omega + \omega) \\
&Q_1 = \tan(i/2) \sin(\Omega) \\
&Q_2 = \tan(i/2) \cos(\Omega) \\
&L = \Omega + \omega + \theta \qquad ,
\end{aligned} \tag{7.1}$$

where $[a, e, i, \Omega, \omega, \theta]$ are the standard Keplerian elements.

The governing equations of motion are the Gauss' planetary equations in a radial-transverse-normal (RTN) reference frame. The fast angular variable L, i.e. the true longitude, will be used as independent variable to replace the time. Under the assumption that the magnitude of the low-thrust control is significantly smaller than the local gravitational force, the resulting system of equations is [184]:

$$\begin{aligned}
\frac{da}{dL} &= \frac{2a^3B^2}{\mu}\left[\frac{P_2 \sin L - P_1 \cos L}{\Phi^2(L)} f_R + \frac{1}{\Phi(L)} f_T\right] \\
\frac{dP_1}{dL} &= \frac{B^4a^2}{\mu}\left[-\frac{\cos L}{\Phi^2(L)} f_R + \left(\frac{P_1 + \sin L}{\Phi^3(L)} + \frac{\sin L}{\Phi^2(L)}\right) f_T - P_2 \frac{Q_1 \cos L - Q_2 \sin L}{\Phi^3(L)} f_N\right] \\
\frac{dP_2}{dL} &= \frac{B^4a^2}{\mu}\left[+\frac{\sin L}{\Phi^2(L)} f_R + \left(\frac{P_2 + \cos L}{\Phi^3(L)} + \frac{\cos L}{\Phi^2(L)}\right) f_T + +P_1 \frac{Q_1 \cos L - Q_2 \sin L}{\Phi^3(L)} f_N\right] \\
\frac{dQ_1}{dL} &= \frac{B^4a^2}{2\mu}\left(1 + Q_1^2 + Q_2^2\right) \frac{\sin L}{\Phi^3(L)} f_N \\
\frac{dQ_2}{dL} &= \frac{B^4a^2}{2\mu}\left(1 + Q_1^2 + Q_2^2\right) \frac{\cos L}{\Phi^3(L)} f_N \qquad ,
\end{aligned} \tag{7.2}$$

where $B = \sqrt{1 - P_1^2 - P_2^2}$ and $\Phi(L) = 1 + P_1 \sin L + P_2 \cos L$. The thrust vector is defined by its acceleration magnitude, azimuth angle and elevation angle in the RTN reference frame:

$$\begin{bmatrix} f_R \\ f_T \\ f_N \end{bmatrix} = \begin{bmatrix} \epsilon \sin\alpha \cos\beta \\ \epsilon \cos\alpha \cos\beta \\ \epsilon \sin\beta \end{bmatrix} \quad . \tag{7.3}$$

First, we calculate a reference solution with no uncertainty that departs from Earth with the hyperbolic excess velocity reported in Table 7.1, where the vector is defined in terms of magnitude, azimuth and elevation in the Earth-centered inertial reference frame. In this deterministic case, the constraint is imposed on the final state to exactly match the equinoctial elements of Apophis at the final time, and the objective function, to be minimised, is the low-thrust ΔV. The reference trajectory was generated with FABLE (Fast Analytical Boundary-value Low-thrust Estimator) [185], transcribing the optimal control problem into a sequence of coast and constant thrust arcs with variable length. The size of each thrust and cost arc is then optimised to minimise the objective function and satisfy the boundary conditions. In FABLE, the dynamics in Eq. (7.2) is analytically propagated using a first-order expansion in the perturbing control acceleration [184]. The value of the objective function for the reference trajectory is $\Delta V = 2.0318$ km/s.

Table 7.1: Deterministic hyperbolic excess velocity at departure from Earth.

Component	Value
v_∞^{det}	3.34 [km/s]
α_∞^{det}	38.39 [deg]
β_∞^{det}	-28.68 [deg]

7.2 Uncertainty formulation

In the following, we consider uncertainties stemming from the interplanetary orbit injection velocity vector. Uncertainty is modelled with the density functions defined in Table 7.2. Note that a support of 4-σ from the peak value is considered.

Table 7.2: Density functions modelling uncertainties on the hyperbolic excess velocity at departure from Earth. The deterministic excess velocity magnitude in Table 7.1 is the mode of the reversed Gaussian tail density distribution modelling the uncertainty on v_∞^{unc}, whereas the angular components are the mean of the Gaussian density distributions describing the uncertainty on α_∞^{unc} and β_∞^{unc}.

Component	Density Function	Param.	Std. deviation	Support
v_∞^{unc}	Reversed Gaussian Tail	3.34 [km/s]	25.0 [m/s]	$[3.24, 3.44]$ [km/s]
α_∞^{unc}	Gaussian	38.39 [deg]	0.250 [deg]	$[37.39, 39.39]$ [deg]
β_∞^{unc}	Gaussian	-28.68 [deg]	0.125 [deg]	$[-29.18, -28.18]$ [deg]

The uncertain vector, defined as $\boldsymbol{\xi} = [v_\infty^{unc}, \alpha_\infty^{unc}, \beta_\infty^{unc},]^T$, induces uncertainty in the initial conditions. The resulting distributions of the equinoctial elements at the initial time are displayed in Fig. 7.1. The colour code indicates the probability density associated with each sample. In the same figures, the black dot is the expected value, and the dashed line represents the projected 1σ ellipsoid corresponding to the covariance matrix reconstructed from the samples. It is possible to see that, by construction, the distribution is asymmetric, and the mean significantly deviates from the mode, located in the dark red coloured area. Hence, the first two moments are not fully representative of the real distribution. In such a case, the method implemented in *IPANeMA* is further justified as it does not require any hypothesis on the shape of the distribution or the linearity of the dynamics.

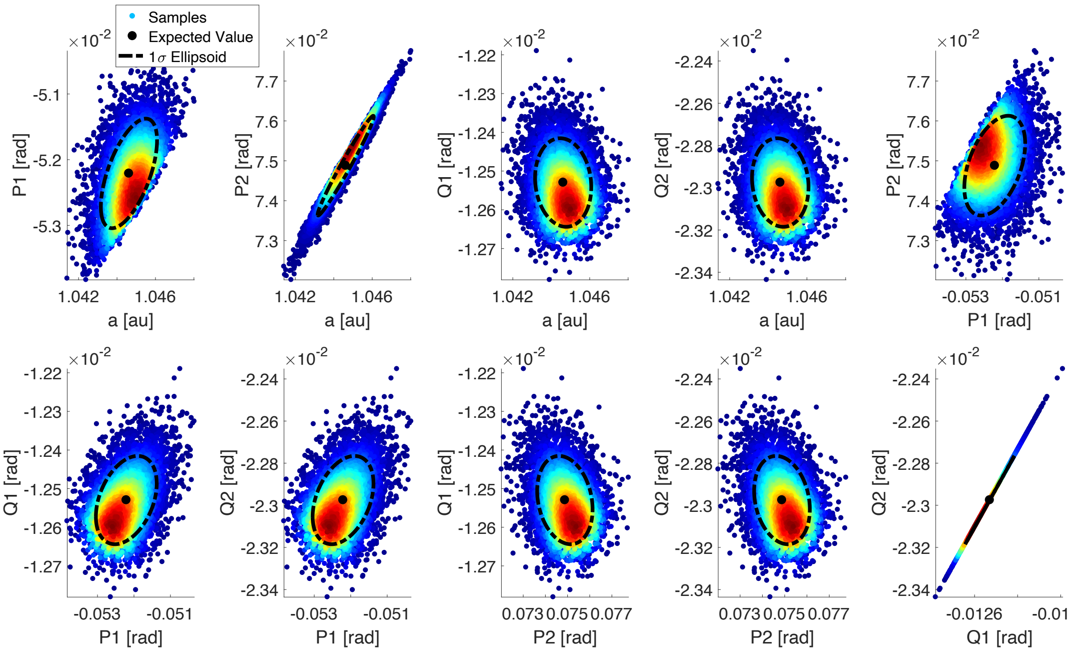


Figure 7.1: Initial projections of the distribution of equinoctial elements induced by the uncertainty in the hyperbolic excess velocity at departure. The black dot is the expected value and the dashed line is the boundary of the 1σ ellipsoid associated to the sample distribution. The color code indicates the probability density associated to each sample.

This optimal control problem under uncertainty is formulated by substituting the final boundary condition with a probability constraint. Specifically, the probability that the final position and velocity have a difference, Δr and Δv, with respect to the target position and velocity, smaller than or equal to the two thresholds Δr_{thres} and

Δv_{thres}. The values of the two thresholds are defined in Table 7.3.

Table 7.3: Position and velocity threshold for probability constraint.

Deviation	Value
Δr_{thres}	$3.0e5$ [km]
Δv_{thres}	100.0 [m/s]

For a given state realisation $\mathbf{x}$, the probability constraint is then computed via the auxiliary positive continuous function:

$$\eta_T(\mathbf{X}=\mathbf{x}) = \begin{cases} \leq 1 & \text{if} \;\; \Delta r \leq \Delta r_{thres} \wedge \Delta v \leq \Delta v_{thres} \\ > 1 & \text{if} \;\; \Delta r > \Delta r_{thres} \vee \Delta v > \Delta v_{thres} \;, \end{cases} \tag{7.4}$$

and by setting the probability threshold $1-\rho$ to 95%. A quadratic smoothing function is used for the convolution operator [147]:

$$h(\eta_T(\mathbf{x})) = 3(1-\eta_T(\mathbf{x})^2)\mathbb{I}_{[-1,+1]}/4 \;. \tag{7.5}$$

Uncertainty is propagated with 4-degree Chebyshev polynomials, given their proven superior global convergence properties [109, 122]. Degree 4 has been selected as a suitable trade-off between representation accuracy (see Section 7.4) and numerical complexity for the intrusive propagation. The fourth-order Runge-Kutta numerical scheme is employed to integrate the dynamical equations of motion. For the transcription with multiple shooting 6 equally sized time intervals, constant controls on each interval and 1000 samples for the calculation of the sample-based expectation in Eq. (2.25) are employed.

7.3 Results

The optimised robust control profile components are shown in Fig. 7.2 and compared to the control profile of the deterministic solution. While the thrust magnitude is essentially unaltered, the thrust angles changed significantly to steer the set of final states within the required target region. In fact, the robust solution achieves a probability

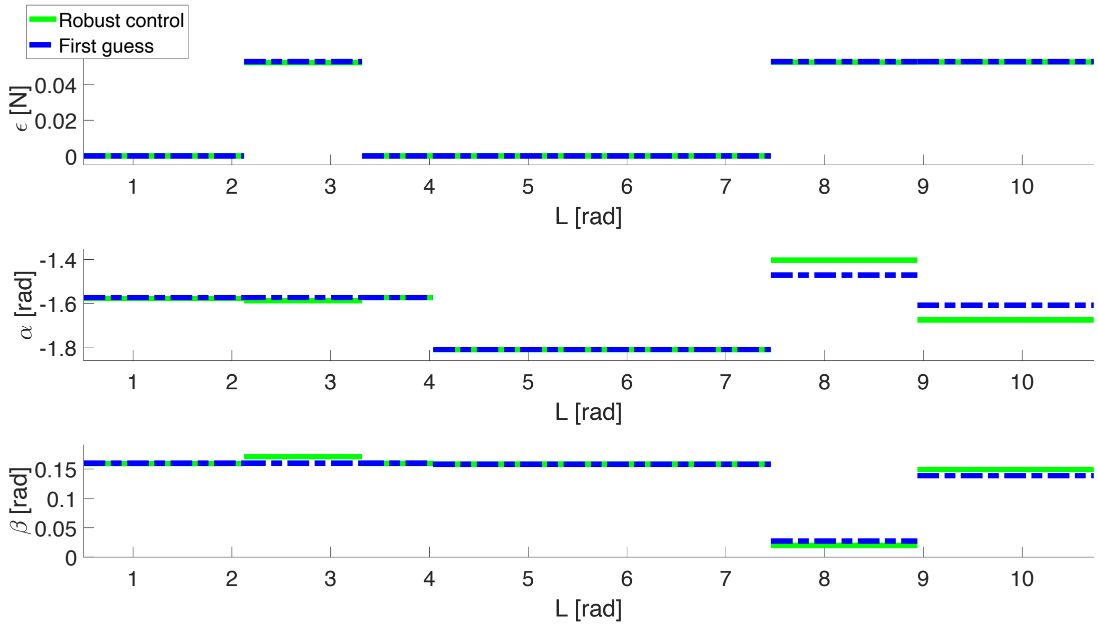


Figure 7.2: Optimised robust control profile and deterministic control components.

of 95.60% that the final state is within the target region. On the contrary, when the robust control profile is replaced with the control profile of the deterministic reference trajectory, the probability of meeting the final target region drops to 69.80%.

Fig. 7.3 shows the probability to have a deviation in the final state that is lower than a given threshold (in position and velocity). The blue dash-dot line is the cumulative probability associated with the control profile of the deterministic reference solution, while the green solid line represents the cumulative probability of the optimal robust solution. The 0.95-quantile is highlighted for both the distributions to show how the robust solution satisfies the probabilistic threshold with a significantly lower constraint deviation. Furthermore, the figure shows that robust distribution always has a higher cumulative probability value for any threshold. This demonstrates that the robust solution probabilistically outperforms the deterministic control for every possible deviation threshold value.

The three-dimensional interplanetary trajectory, corresponding to the peak values of the injection uncertain conditions in Table 7.2, is shown in Fig. 7.4, decomposed in its ecliptic in-plane and out-of-plane projections.

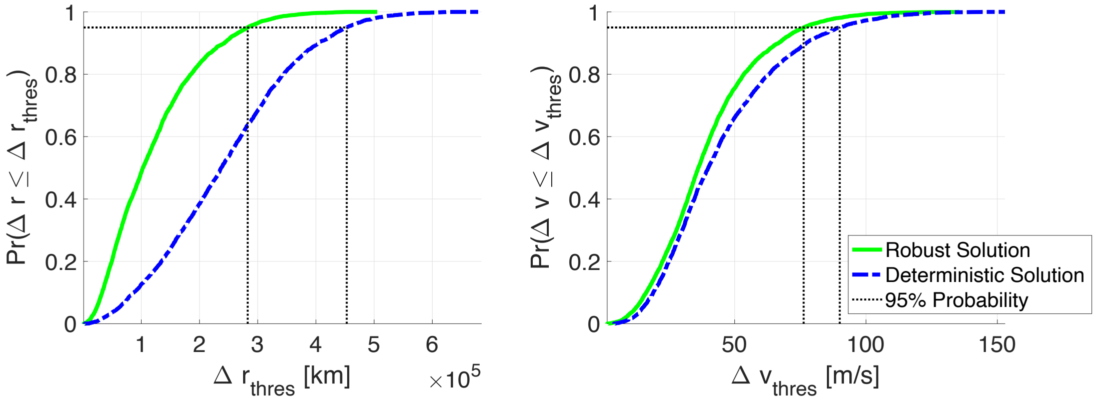


Figure 7.3: Cumulative distribution functions of final position and velocity deviations for robust and deterministic reference solution.

7.4 Validation and verification

In this section, we verify the accuracy of the two key approximations employed in the optimisation routine, that is, the propagation with GPA and the probability approximation by convolution on a limited number of samples.

To this end, 10^5 samples drawn from the initial distribution are propagated forward in time with the robust control profile, either using the polynomial approximation coming from GPA or by direct numerical integration of the dynamics with a fourth-order Runge-Kutta integrator. The resulting root-mean-square (RMS) of the difference between the GPA approximation and the numerical integration of the equinoctial elements, at the final time, is reported in Table 7.4. These RMS deviations result in an error of 10^{-4} relative to the threshold values as set in Table 7.3. Therefore, these results confirm that GPA produces a fully satisfactory propagation approximation as required for computing accurately the probability constraint.

Table 7.4: Root-mean-square error per state component between 10^5 samples propagated by numerical integration and by intrusive polynomial algebra.

a [AU]	P_1 [deg]	P_2 [deg]	Q_1 [deg]	Q_2 [deg]
$1.17 \cdot 10^{-6}$	$3.80 \cdot 10^{-6}$	$1.88 \cdot 10^{-5}$	$4.37 \cdot 10^{-6}$	$8.88 \cdot 10^{-6}$

Then, the numerically propagated samples are used to compute the probability of hitting the final target region without the convolution approximation. The result is

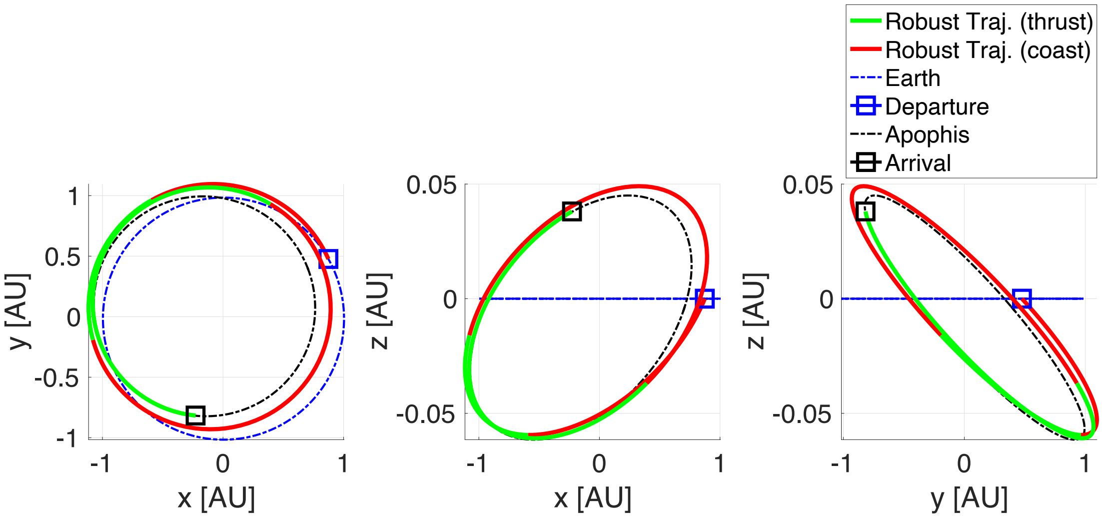


Figure 7.4: Three-dimensional robust interplanetary trajectory of spacecraft departing from Earth and arriving at Apophis with initial conditions resulting from the peak values of the uncertainties in Table 7.2. The green and red lines represent the spacecraft trajectory during thrust and coast arcs, respectively.

95.97%, slightly higher than the value 95.60% that was calculated in the optimisation loop. Table 7.5 reports the probability resulting from different combinations of control profiles, propagation schemes, number of samples, and with or without the convolution operator.

Table 7.5: Probability of matching the final target region for different control laws and approximation schemes.

Solution	Propagation	Samples	Convolution	Probability
Deterministic	Polynomial	10^3	Yes	69.80%
Robust	Polynomial	10^3	Yes	95.60%
Robust	Polynomial	10^5	No	96.00%
Robust	Numerical	10^5	No	95.97%

The results on lines two and four of Table 7.5 confirm that the use of GPA and the convolution function provide an excellent approximation of the probability of meeting the final target region.

A time comparison is then performed between the average time employed to propagate a sample (the average over the 10^5 samples) with a fourth-order Runge-Kutta numerical integrator and the recursive polynomial mapping with GPA. A time step of 2 hours was used in both propagations. The results, averaged out of ten tests on an OptiPlex7050 desktop with Intel(R) Core i7-7700 @ 3.60GHz with 8GB of RAM running Ubuntu 18.04.1 LTS, are reported in Table 7.6. The table shows that the polynomial mapping is almost one order of magnitude less computationally expensive, per propagated sample, than a full numerical integration. Since this time has to be multiplied by the number of samples and the number of times sampling is required in the optimisation loop, the employment of intrusive polynomial algebra results crucial to the practical solution of realistic scenarios.

Table 7.6: Average time comparison for propagation of a sample with fourth order Runge-Kutta numerical integrator and with polynomial surrogate mapping.

Propagation	Average Time $[ms]$
Polynomial	1.5
Numerical	13.4

7.5 Epistemic uncertainty

The method employed in this section can also accommodate the case in which uncertainty is epistemic or probabilities are imprecise. When uncertainty is epistemic, there is a lack of knowledge on which distribution p has to be used, and we can only know that p belongs to a set of possible probability density functions: $p \in \mathcal{P}$. On the other hand, when imprecision affects the probabilities, one can argue that the probability of a realisation depends on a family of distributions rather than on a single one. In both cases, in general, the probability distribution p is not exactly known, and one can only know the boundaries of the set $\mathcal{P}$ of possible density functions. This section considers the less general case in which p belongs to a known family of probability distribution with parameters affected by epistemic uncertainty. Although this is not the most general case of imprecision and epistemic uncertainty, it serves the scope of showing how *IPANeMA* can handle distributions that are not completely defined a priori.

For an efficient computation of the expectation, the *importance sampling* scheme in Eq. (2.27) should be preferred, in this case, in contrast to the *direct Monte Carlo* sampling as used in Section 7.2.

Since the key aspect of handling imprecision and epistemic uncertainty is the computation of the expectation of quantities of interest, in the following, we use the robust control optimised previously computed in Section 7.3, and we calculate the different cumulative density functions which result from the set $\mathcal{P}$.

For each uncertain parameter, we consider a different family of density functions: a family of reversed Gaussian tail for the norm of the escape velocity and two different families of Gaussian for azimuth and elevation angles (see Table 7.7). Epistemic uncertainty affects the knowledge of the standard deviation of these distributions, which is now defined as an interval. By construction, the precise uncertainty in Table 7.2 falls within the imprecise set.

Table 7.7: Parameterisation of imprecise set of density functions $\mathcal{P}$ modelling uncertainties on the hyperbolic excess velocity at departure from Earth for the same supports defined in Table 7.2.

Component	Density Function	Param.	Std. deviation
v_∞^{unc}	Reversed Gaussian Tail	3.34 [km/s]	$[20.0, 33.3]$ [m/s]
α_∞^{unc}	Gaussian	38.39 [deg]	$[0.2, 0.33]$ [deg]
β_∞^{unc}	Gaussian	-28.68 [deg]	$[0.1, 0.166]$ [deg]

A three-dimensional uniform density function over the defined support is used as *proposal distribution*

$$\pi(\boldsymbol{\xi}) = \mathcal{U}_{[3.24,3.44]}(v_\infty^{unc}) \cdot \mathcal{U}_{[37.39,39.39]}(\alpha_\infty^{unc}) \cdot \mathcal{U}_{[-29.18,-28.18]}(\beta_\infty^{unc}) \tag{7.6}$$

to draw samples. The samples are propagated only once, and then the expectations corresponding to different admissible densities $p \in \mathcal{P}$ are computed with different weights as in Eq. (2.27). The empirical cumulative distribution functions for a number of distributions p are shown in Fig. 7.5 (dash-dot line with different markers). For comparison, also the CDF corresponding to the robust solution in Fig. 7.3 is added to the figure (green solid line). Given the construction of $\mathcal{P}$, the robust solution falls within the set of empirical CDFs.

Note that using a set of distributions rather than a single one does not affect the transcription and propagation schemes. However, the NLP solver needs to evaluate only one expectation function. Hence, one should replace the expectation functions

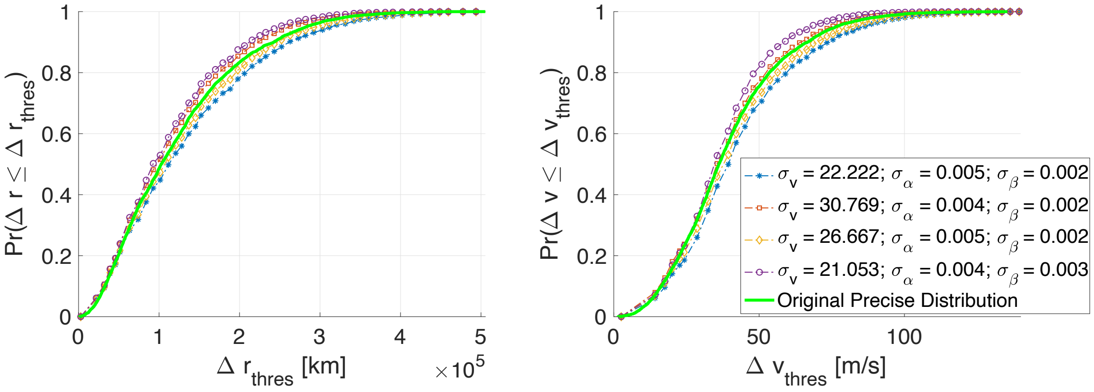


Figure 7.5: Empirical cumulative distribution functions of final position and velocity deviations for different uncertainty density functions out of the imprecise set $\mathcal{P}$ and for the original precise distribution defined in Table 7.2.

in problem (4.2) with either the lower or upper expectations as in Eq. (2.4). For the constraint in probability defined in Table 7.3, the idea is to evaluate the lowest probability of satisfying the constraint, given the set $\mathcal{P}$, which corresponds to the worst-case constraint violation. Therefore, the constraint would be imposed on the lower expectation:

$$\underline{\mathbb{E}}\left[\mathbb{I}_T(\mathbf{X}_f)\right] = \inf_{p \in \mathcal{P}} Pr(\mathbf{X}_f \in T) \in [1-\rho, 1] \tag{7.7}$$

The optimisation under epistemic uncertainty is not directly solved in this section but it is left as the main advancement of the next chapter.

7.6 Chapter summary

In this chapter, *IPANeMA* was applied to the robust optimisation of a low-thrust rendezvous trajectory to the near-Earth asteroid 99942 Apophis subject to uncertain hyperbolic excess velocity at departure. The resulting robust control law achieved over 95% of probability to reach the final target set. Furthermore, it was shown that this robust solution significantly outperforms the reference, deterministic, one.

It was also shown that the method for the propagation of uncertainty and the calculation of the expectation is fast and accurate and has low computational complexity.

In particular, it was shown that the polynomial mapping is very accurate when compared to a direct numerically integration of the dynamics, and the convolution operator produces an accurate estimation of the correct probability.

Chapter 8

Robust flyby design and analysis

The content of this chapter was published in:
C. Greco, S. Campagnola, and M. Vasile, “Robust space trajectory design using belief optimal control”, *Journal of Guidance, Control, and Dynamics*, under review [85].
C. Greco, and M. Vasile, “Closing the loop between mission design and navigation analysis”, *International Astronautical Congress, The Cyberspace Edition*, 2020 [93].

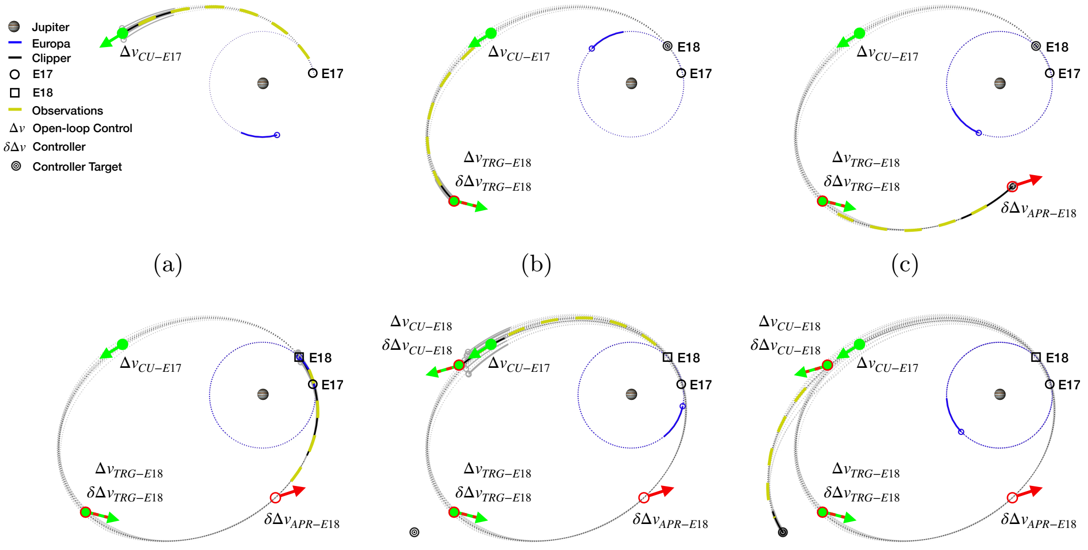
In this chapter, first BOC, developed in Section 4.2, is applied to the robust optimisation of one leg of the Europa Clipper tour [186] in Section 8.1. Successively, the epistemic variational inference presented in Section 3.3 is employed for reliable navigation analysis on the same flyby scenario in Section 8.2.

8.1 Robust trajectory design

8.1.1 Mission scenario

The original full tour comes from a deterministic design. However, from a navigation analysis, the leg of interest, labelled E17-E18, yields a dangerously high probability of impact with Europa at flyby E18. Such deterministic trajectory is used as initial guess for the robust optimisation. Therefore, the goal of this test case is to re-optimise the E17-E18 open-loop trajectory with BOC so that the flight dynamics requirements are satisfied and the remainder of the tour remains feasible. The problem is to minimise the sum of deterministic and statistical manoeuvres, while respecting the constraints on the desired B-plane flyby parameter **b**, the hyperbolic TCA [187], the expected value of the final position, and a constraint on the collision probability with Europa. We will study both the case of purely aleatory uncertainty and mixed aleatory and epistemic uncertainty.

The scenario addressed in this section is depicted in Figure 8.1. Each subplot in



(a) (b) (c)

(d) (e) (f)

Figure 8.1: Schematic representation of navigation analysis setup for part of Europa Clipper leg belief optimisation test case.

Figure 8.1 represents a phase of the analysed trajectory, specifically:

(a) The $E17-CU\text{-}E17$ phase (Fig. 8.1(a)) starts from the initial belief $\mathcal{X}_0$, and goes till the first control point, denoted as Clean-Up (CU).

(b) The $CU\text{-}E17 - TRG\text{-}E18$ phase (Fig. 8.1(b)) starts from the CU, where the belief state is given by the ensemble $\sum_j b_j p_j$, executes the open-loop manoeuvre $\Delta\overline{\mathbf{v}}_{CU\text{-}E17}$ and propagates till the next control point near the trajectory apocenter, denoted as Targeting (TRG).

(c) The $TRG\text{-}E18 - APR\text{-}E18$ phase (Fig. 8.1(c)) starts from TRG with the belief state given by the ensemble $\sum_j b_j p_j$, performs manoeuvre $\Delta\overline{\mathbf{v}}_{TRG\text{-}E18}$ +

$\delta\Delta\mathbf{v}_{TRG\text{-}E18}$ and propagates till the pre-flyby control point, denoted as Approach (APR).

(d) The $APR\text{-}E18 - E18$ phase (Fig. 8.1(d)) starts from APR with the belief state given by the ensemble $\sum_j b_j p_j$, performs the feedback manoeuvre $\delta\Delta\mathbf{v}_{APR\text{-}E18}$ and extends till flyby $E18$.

(e) The $E18 - CU\text{-}E18$ phase (Fig. 8.1(e)) starts from $E18$ with the belief state given by the ensemble $\sum_j b_j p_j$, extends till the successive clean-up point $CU\text{-}E18$.

(f) The $CU\text{-}E18 - T_F$ phase (Fig. 8.1(f)) starts from T_F with the belief state given by the ensemble $\sum_j b_j p_j$, performs manoeuvre $\Delta\overline{\mathbf{v}}_{APR\text{-}E18} + \delta\Delta\mathbf{v}_{CU\text{-}E18}$ and extends till the leg final time.

During each phase, OD campaigns are carried out, represented in yellow, with an 8 hours ON 8 hours OFF schedule (dashed line) to improve the knowledge of the trajectory. The 8 hours ON 8 hours OFF is the access schedule of the DSN for Europa Clipper, with range and range-rate measurements generally employed, whereas Delta-DOR is used only when specifically needed. Therefore, this latter measurement type will not be included in the observation model. The OD stops at a cut-off time before the subsequent manoeuvre, here set to one day, to model the time needed by the operators to compute the updated trajectory. The Navigation Analysis (NA) approach solving the nonlinear Bayes' step developed in Section 4.2.2 is employed here to simulate observation samples and update the belief state.

The Probability of Impact (PoI) with Europa is computed by propagating the belief state after the targeting manoeuvre $\Delta\overline{\mathbf{v}}_{TRG\text{-}E18} + \delta\Delta\mathbf{v}_{TRG\text{-}E18}$ to the nominal flyby time without applying the successive approach manoeuvre $\delta\Delta\mathbf{v}_{APR\text{-}E18}$ or performing additional OD. This mapped uncertainty projected onto the B-plane is computed by applying map $\mathcal{T}_P$ only. A constraint on the PoI is then enforced to ensure environmental protection to Europa even in the event of spacecraft loss after the main manoeuvre.

The whole trajectory lasts for 21 days, from E17 to the second apocenter passage. The trajectory in Fig. 8.1 is defined in an inertial reference frame, mean equinox and ecliptic of J2000 (ECLIPJ2000) [188], centred in Jupiter. This frame has the $\hat{\mathbf{x}}$ axis pointing to the mean vernal equinox at January 1, 2000, the $\hat{\mathbf{z}}$ axis normal to the ecliptic plane, and $\hat{\mathbf{y}}$ completing the right-handed frame. A high-fidelity full-ephemeris dynamics governs the spacecraft motion $\mathbf{x}(t)$, taking into account the gravitational field of Jupiter (central and J_2 effects), of its moons Europa (central and J_2), Io, Ganymede and Callisto, and of the Sun. The dynamics is integrated with the propagation module of the trajectory optimisation tool jTOP [21]. This black-box module has been used to propagate collocation points to train the non-intrusive surrogate model described in Section 4.2.2. The outer and inner optimisation loops are solved with two nested instances of the local solver MATLAB *fmincon*.

In this section, we will talk about delivered and mapped uncertainty. By delivered uncertainty, we intend the output of pure dynamical propagation of the belief state in time without further OD. By mapped uncertainty, we intend the output of a generic transformation of the belief state, e.g. conversion to B-plane parameters.

8.1.2 Problem statement

For this application, the interest is in finding the optimal *open-loop* control $\Delta\overline{\mathbf{v}}$ which yields the most robust and reliable trajectory under uncertainty in navigation analysis and manoeuvre execution errors. Hence, the executed control $\mathbf{U}_e$ acting on the belief components is written as

$$\Delta\mathbf{V}_e(t_k, \mathbf{X}_k) = \Delta\overline{\mathbf{v}}(t_k) + \delta\Delta\mathbf{v}(\mathbf{X}_k^{(j)}) + \boldsymbol{\Theta}\big(\Delta\overline{\mathbf{v}}(t_k) + \delta\Delta\mathbf{v}(\mathbf{X}_k^{(j)}), \boldsymbol{\lambda}_k\big) \ , \tag{8.1}$$

where the open-loop nominal impulse $\overline{\mathbf{u}} = \Delta\overline{\mathbf{v}}(t_k)$ is to be optimised and the disturbance $\boldsymbol{\Theta}$ depends nonlinearly on the commanded control $\Delta\mathbf{v} = \Delta\overline{\mathbf{v}}(t_k) + \delta\Delta\mathbf{v}(\mathbf{X}_k^{(j)})$. The definition of $\boldsymbol{\Theta}$ follows the Gates' model, as already described in Section 5.2.2.

The components of the Gates' parameters vector $\boldsymbol{\lambda}_k = [\sigma_{mf}, \sigma_{mp}, \sigma_{pf}, \sigma_{pp}]^T$ have crisp values for the pure aleatory case, as defined in Table 8.1, whereas they are interval-valued for the epistemic case, as in Table 8.2.

The closed-loop control component $\delta\Delta\mathbf{v}(\mathbf{X}_k^{(j)})$ is an analytical linear function from time t_k to a generic target state at time $t_{\circledcirc}$, which therefore needs not to coincide with the end of that segment t_{k+1}. Let $\mathbf{x} = [\mathbf{r}, \mathbf{v}]$ be the decomposition of the spacecraft inertial state in position and velocity. The linear guidance law to target the position $\mathbf{r}_{\circledcirc}$ is computed as follows. First, the dynamics is linearised around the nominal trajectory in $[t_k, t_{\circledcirc}]$, that is, the one computed using only the optimisable open-loop control $\Delta\overline{\mathbf{v}}$, to obtain a linear mapping

$$\begin{bmatrix} \delta\mathbf{r}_{\circledcirc} \\ \delta\mathbf{v}_{\circledcirc} \end{bmatrix} = \Phi_k^{\circledcirc} \begin{bmatrix} \delta\mathbf{r}_k \\ \delta\mathbf{v}_k + \delta\Delta\mathbf{v}_k \end{bmatrix} = \begin{bmatrix} \Phi^{\circledcirc}_{k\,1,1} & \Phi^{\circledcirc}_{k\,1,2} \\ \Phi^{\circledcirc}_{k\,2,1} & \Phi^{\circledcirc}_{k\,2,2} \end{bmatrix} \begin{bmatrix} \delta\mathbf{r}_k \\ \delta\mathbf{v}_k + \delta\Delta\mathbf{v}_k \end{bmatrix} , \tag{8.2}$$

where the Cartesian state deviation $\delta\mathbf{x}_k$ has been decomposed in position $\delta\mathbf{r}_k$ and velocity $\delta\mathbf{v}_k$ deviations, and $\Phi_k^{\circledcirc}$ has been accordingly partitioned in 3 × 3 blocks. The linear guidance is obtained by imposing

$$\delta\mathbf{r}_{\circledcirc} = \mathbf{0}$$

which leads to

$$\delta\Delta\mathbf{v}_k = -\Phi^{\circledcirc\,-1}_{k\,1,2} \begin{bmatrix} \Phi^{\circledcirc}_{k\,1,1} & \Phi^{\circledcirc}_{k\,1,2} \end{bmatrix} \begin{bmatrix} \delta\mathbf{r}_k \\ \delta\mathbf{v}_k \end{bmatrix} . \tag{8.3}$$

This guidance law is employed for the controller $\delta\Delta\mathbf{v}_{CU\text{-}E18}$ to target the final nominal position.

The targeting and approach manoeuvres, respectively $\delta\Delta\mathbf{v}_{TRG\text{-}E18}$ and $\delta\Delta\mathbf{v}_{APR\text{-}E18}$, target the B-plane parameters instead [187]. Let $\mathbf{B}_{\circledcirc} = [\mathbf{b}_{\circledcirc}, \mathrm{TCA}_{\circledcirc}]$ be the B-plane parameters targeted, with $\mathbf{b}$-vector expressed in two components ($b_t = \mathbf{b} \cdot \hat{\mathbf{t}}$ and $b_r = \mathbf{b} \cdot \hat{\mathbf{r}}$) and TCA being the hyperbolic time of closest approach, $\mathcal{B}$ the corresponding coordinate transformation from the inertial state $\mathbf{B} = \mathcal{B}(\mathbf{x})$, and $\mathbf{J}_{\mathcal{B}}$ its Jacobian (see Appendix B). Using the linearisation as above, the deviations in B-plane parameters from the nominal ones are written as

$$\delta\mathbf{B}_{\circledcirc} = \mathbf{J}_{\mathcal{B}}\,\Phi_k^{\circledcirc} \begin{bmatrix} \delta\mathbf{r}_k \\ \delta\mathbf{v}_k + \delta\Delta\mathbf{v}_k \end{bmatrix} = \begin{bmatrix} \partial\mathbf{B}/\partial\mathbf{r}_{\circledcirc} & \partial\mathbf{B}/\partial\mathbf{v}_{\circledcirc} \end{bmatrix} \begin{bmatrix} \Phi^{\circledcirc}_{k\,1,1} & \Phi^{\circledcirc}_{k\,1,2} \\ \Phi^{\circledcirc}_{k\,2,1} & \Phi^{\circledcirc}_{k\,2,2} \end{bmatrix} \begin{bmatrix} \delta\mathbf{r}_k \\ \delta\mathbf{v}_k + \delta\Delta\mathbf{v}_k \end{bmatrix} . \tag{8.4}$$

By imposing

$$\delta \mathbf{B}_{\circledcirc} = \mathbf{0}$$

and by matrix manipulation, the B-plane targeting is obtained as a linear guidance law as

$$\delta \Delta \mathbf{v}_k = -\left(\frac{\partial \mathbf{B}}{\partial \mathbf{r}_{\circledcirc}} \Phi^{\circledcirc}_{k\,1,2} + \frac{\partial \mathbf{B}}{\partial \mathbf{v}_{\circledcirc}} \Phi^{\circledcirc}_{k\,2,2} \right)^{-1} \mathbf{J}_{\mathcal{B}}\, \Phi^{\circledcirc}_{k} \begin{bmatrix} \delta \mathbf{r}_k \\ \delta \mathbf{v}_k \end{bmatrix} . \tag{8.5}$$

This guidance law is employed for the controllers $\delta\Delta\mathbf{v}_{TRG\text{-}E18}$ and $\delta\Delta\mathbf{v}_{APR\text{-}E18}$ to target the nominal B-plane parameters. Hence, the belief component-dependent controller $\delta\Delta\mathbf{v}(\mathbf{X}_k^{(j)})$ employed in this test case is computed according to either Eq. (8.3) or (8.5) depending on the targeted parameters. The quantities $\delta\mathbf{r}_k$ and $\delta\mathbf{v}_k$ are the deviations of the belief component expected value $\mathbb{E}\big[\mathbf{X}_k^{(j)}\big]$ with respect to the nominal trajectory at time t_k.

As for the orbit determination campaigns, the measured quantity is the range and range rate of the spacecraft with respect to Earth (see Sections 3.2.1-3.2.2 in [162]). The likelihood function is modelled as Gaussian, and the associated covariance characterising the observation accuracy is assumed diagonal (see Table 8.1).

Once the execution and navigation errors are defined we can explicitly write the BOC formulation as follows:

$$\min_{\Delta \overline{\mathbf{v}}_i} \quad \Delta v99 \tag{8.6a}$$

$$\text{s.t.} \quad \mathcal{X}_k = \mathcal{T}\big(\, \mathcal{X}_{k-1},\, \mathcal{D}_{k-1},\, \Delta\mathcal{V}_{e_{k-1}},\, \mathcal{E}_k\big) \tag{8.6b}$$

$$\underline{P}\,(\|\Delta\mathbf{V}_{e_{tot}}\| < \Delta v99) = 0.99 \tag{8.6c}$$

$$\overline{\mathrm{PoI}} < \varepsilon \tag{8.6d}$$

$$\underline{\mathbb{E}}\big[\mathbf{B}_{E18}\big],\, \overline{\mathbb{E}}\big[\mathbf{B}_{E18}\big] \in \mathbf{\Phi}_{\mathbf{B}_{E18}} \tag{8.6e}$$

$$\underline{\mathbb{E}}\big[\mathbf{R}_{F}\big],\, \overline{\mathbb{E}}\big[\mathbf{R}_{F}\big] \in \mathbf{\Phi}_{\mathbf{R}_{F}} \tag{8.6f}$$

$$\mathcal{X}_0 \in \mathcal{P}_{x_0}\ ,\ \vartheta \in \mathcal{P}_{\mathbf{\Theta}}\ ,\ \mathcal{Y}_k = \mathcal{N}_{\mathbf{Y}_k|\mathbf{X}_k}\big(\boldsymbol{\mu}_{y_k}, \mathbf{\Sigma}_{y_k}\big)\ , \tag{8.6g}$$

where k denotes the time discretisation with multiple arcs which can contain both control and observations, only control, only observations or neither of the two according to the phases described in Figs. 8.1. The objective to be minimised is the threshold value $\Delta v99$ on the quantile of the magnitude of the total $\|\Delta \mathbf{V}_{e_{tot}}\|$. Eq. (8.6c) expresses the quantile with a constraint on the lower probability P that the total cost is below the threshold $\Delta v99$ (see (2.8) for the expression of P). The total magnitude $\|\Delta \mathbf{V}_{e_{tot}}\|$ including execution errors is defined as:

$$\|\Delta \mathbf{V}_{e_{tot}}\| = \sum_i \|\Delta \mathbf{V}_{e_i}(t_k, \mathbf{X}_k)\| + \|\Delta \mathbf{V}_F\| \ ,$$

which is a random variable encompassing all manoeuvres (indexed by $i \in \{CU\text{-}E17, TRG\text{-}E18, APR\text{-}E18, CU\text{-}E18\}$) plus the final velocity mismatch magnitude $\|\Delta \mathbf{V}_F\| = \|\mathbf{V}_F - \overline{\mathbf{v}}_F\|$ between the final state velocity and the original final velocity. Numerically, this constraint is computed with Equation (4.36) using $\phi = \mathbb{I}(\|\Delta \mathbf{v}_{e_{tot}}\| < \Delta v99)$ by taking samples from the executed control distribution $\Delta \mathcal{V}_e$ and the final velocity mismatch $\Delta \mathcal{V}_F$. In particular, for the latter, samples $\mathbf{x}_F^{(i)}$ are first generated from the final belief state $\mathcal{X}_F$, then the velocity vector of each sample $\mathbf{v}_F^{(i)}$ is subtracted by the target final velocity $\overline{\mathbf{v}}_F$, and finally, the norm of this difference is used as a sample of the velocity mismatch magnitude. Constraint (8.6f) imposing the expected value of the final position vector, together with the mismatch $\|\Delta \mathbf{V}_F\|$ in the objective function, ensures that the expectation of the $E17 - E18$ trajectory connects with the remainder of the original tour.

Eq. (8.6b) is the belief transition function from segment k to segment $k+1$. The transition function incorporates the effect of observations and manoeuvres when present within a segment.

Eq. (8.6d) is the upper bound on the probability of impact $\overline{\text{PoI}}$ after targeting, and it is written as the upper bound on the probability of the minimum distance from Europa, at flyby E18, to be smaller or equal to the radius of Europa written as

$$\text{PoI} = \int \mathbb{I}(\phi(\mathbf{x}_{E18}) \leq R_{EUR})\, p(\mathbf{x}_{E18}; \boldsymbol{\lambda})\, d\mathbf{x}_{E18} \ , \tag{8.7}$$

where $r_{E18} = \phi(\mathbf{x}_{E18})$ is the function mapping the Cartesian state at $E18$ to the pericenter distance of the hyperbolic trajectory with respect to Europa. This probability is constrained to be less than $\varepsilon = 0.1\%$. Numerically, this constraint is computed using Equation (4.36) with $\phi = \mathbb{I}(r_{E18} \leq R_{EUR})$.

Eq. (8.6e) imposes the expected values of the B-plane flyby conditions to be within a target set $\mathbf{\Phi}_{\mathbf{B}_{E18}}$. In the first test case with purely aleatory uncertainty, the set is a singleton composed of the target B-parameter $\mathbf{\Phi}_{\mathbf{B}_{E18}} = \{\widehat{\mathbf{B}}\}$. In the second case with epistemic uncertainty, the target set is defined as an hyperbox around the target value $\widehat{\mathbf{B}} = [\widehat{b}_t, \hat{b}_r, \widehat{\mathrm{TCA}}]$ as

$$\mathbf{\Phi}_{\mathbf{B}_{E18}} = [\hat{b}_t - \delta b_t, \hat{b}_t + \delta b_t] \times [\hat{b}_r - \delta b_r, \hat{b}_r + \delta b_r] \times [\widehat{\mathrm{TCA}} - \delta\,\mathrm{TCA}, \widehat{\mathrm{TCA}} + \delta\,\mathrm{TCA}] \quad (8.8)$$

where the tolerances are set to $\delta b_t = 0.5$ [km], $\delta b_r = 0.5$ [km] and $\delta\,\mathrm{TCA} = 0.05$ [s].

Similarly, Eq. (8.6f) requires the lower and upper expectations of the final position to be within the set $\mathbf{\Phi}_{\mathbf{R}_F}$. In the aleatory case, the set is composed of a single element $\mathbf{\Phi}_{\mathbf{R}_F} = \{\hat{\mathbf{r}}_F\}$. In the epistemic case, the set is defined as a hyper-box around the precise target position as

$$\mathbf{\Phi}_{\mathbf{R}_F} = [\hat{\mathbf{r}}_{F_x} - \delta\mathbf{r}_{F_x}, \hat{\mathbf{r}}_{F_x} + \delta\mathbf{r}_{F_x}] \times [\hat{\mathbf{r}}_{F_y} - \delta\mathbf{r}_{F_y}, \hat{\mathbf{r}}_{F_y} + \delta\mathbf{r}_{F_y}] \times [\hat{\mathbf{r}}_{F_z} - \delta\mathbf{r}_{F_z}, \hat{\mathbf{r}}_{F_z} + \delta\mathbf{r}_{F_z}] \quad (8.9)$$

for the $x-$, $y-$ and $z-$components of the position vector and where each tolerance is $\delta\mathbf{r}_{F_{(\cdot)}} = 10$ km because variations in the next apocenter position are less critical and can be compensated with successive manoeuvres. In the epistemic case, these constraints imply that the open-loop optimum under uncertainty needs to satisfy, in expectation, the flyby and terminal conditions within a set for all distributions in the imprecise set $\mathcal{P}_\lambda$.

Finally, Eqs.(8.6g) define the initial belief condition, the uncertain parameters distribution and the observation likelihood.

For comparison, the dOCP for the first guess generation optimises only the nominal Δv while respecting the reference initial conditions, flyby B-plane parameters $\widehat{\mathbf{B}}$, and final boundary conditions $\hat{\mathbf{r}}_F$. No observations are employed in the deterministic first guess generation.

In the pure aleatory case, the epistemic sets are composed of a single distribution each, that is $\mathcal{P}_{x_0} = \left\{\mathcal{N}\left(\boldsymbol{\mu}_{x_0}, \boldsymbol{\Sigma}_{x_0}\right)\right\}$ and $\mathcal{P}_{\boldsymbol{\Theta}} = \{p(\boldsymbol{\theta}; \boldsymbol{\lambda}_{\boldsymbol{\Theta}})\}$ is described by a single Gates' model distribution as described above. The parameters for these uncertainty models are reported in Table 8.1. These values come from previous navigation analysis studies for Europa Clipper [189, 44]. The table reports respectively the square root of the diagonal values (standard deviations) for the initial dispersion covariance $\boldsymbol{\Sigma}_{x_0}$, with typical values reconstructed from OD campaigns post-flyby, the Gates' model parameters $\boldsymbol{\lambda}_{\boldsymbol{\Theta}}$ for the execution errors, and the standard deviation values for the accuracy of each observation type $\boldsymbol{\Sigma}_{y_k}$. The initial mean $\boldsymbol{\mu}_{x_0}$ is the spacecraft initial state coming from the deterministic tour design used as the initial guess.

In the epistemic scenario, we consider the following uncertainty components. The initial dispersion considered in the aleatory scenario is the reconstructed uncertainty from simulated OD arcs post-flyby. Therefore, the values in Table 8.1 are estimated during the navigation analysis in the mission design phase, whereas the actual dispersion to consider during operations may vary from these values. This further uncertainty is modelled as epistemic, and the imprecise initial set is parameterised as

$$\begin{aligned} \mathcal{P}_{x_0} = \{\, p(\mathbf{x}_0) :\ & p(\mathbf{x}_0) = \mathcal{N}\Big(\mathbf{x}_0; \boldsymbol{\mu}_{x_0}, \widetilde{\boldsymbol{\Sigma}}_{x_0}\Big), \\ & \widetilde{\boldsymbol{\Sigma}}_{x_0} = \text{blkdiag}(\, \lambda_{x_{0-1}}\, \boldsymbol{\Sigma}_{x_0}(1{:}3, 1{:}3),\ \lambda_{x_{0-2}}\, \boldsymbol{\Sigma}_{x_0}(4{:}6, 4{:}6)), \\ & \lambda_{x_{0-1}} \in [0.5, 2.0],\ \lambda_{x_{0-2}} \in [0.5, 2.0] \,\}\ , \end{aligned} \tag{8.10}$$

where $\boldsymbol{\Sigma}_{x_0}(1{:}3, 1{:}3)$ and $\boldsymbol{\Sigma}_{x_0}(4{:}6, 4{:}6)$ indicate respectively the position block and the velocity block, the operator blkdiag indicates a block-diagonal matrix, $\lambda_{x_{0-1}}$ and $\lambda_{x_{0-2}}$ are two multipliers scaling the precise covariance matrix $\boldsymbol{\Sigma}_{x_0}$ defined from the standard deviations in Table 8.1. Being the multipliers defined within $[0.5, 2.0]$, they encompass distributions with covariance from half up to double the magnitude of the pure aleatory one. That is, the epistemic multipliers can give more or less confidence to the initial state knowledge. For the execution errors, the epistemic set $\mathcal{P}_{\boldsymbol{\Theta}} = \left\{p(\boldsymbol{\theta}; \widetilde{\boldsymbol{\lambda}}_{\boldsymbol{\Theta}})\right\}$ is constructed by allowing interval-valued parameters $\widetilde{\boldsymbol{\lambda}}_{\boldsymbol{\Theta}}$ in the Gates' model. Indeed, these parameters are estimated by testing the engine in nonoperational conditions and then updated multiple times during the spacecraft operational life with possible substantial changes, as it happened, for example, during the Cassini mission [190]. Specifically, the intervals for the model parameters considered in the epistemic analysis are reported in Table 8.2, which include the precise values employed in the pure aleatory scenario. For missions like Europa Clipper, the observations from Earth are generally performed from the DSN. In such advanced facilities, the instruments and operating conditions are well known and precisely controlled. Therefore, the likelihood distribution describing the observation noise is well characterised. Hence, the zero-mean observation errors are assumed to remain purely aleatory also in the case of epistemic uncertainty on initial conditions and manoeuvre execution. Nonetheless, the overall OD remains an epistemic process because the priors, resulting from the initial conditions and execution errors, are epistemic.

Other model uncertainty sources are not included in this preliminary test case, e.g. celestial bodies' ephemerides uncertainty, although they may be relevant for a complete navigation analysis.

Table 8.1: Parameters of aleatory uncertainty models considered in Europa's moon flyby belief optimisation.

Uncertainty	Component	Value	
Initial Dispersion	1σ Position (RTN)	[3.7, 5.3, 9.3]	[m]
	1σ Velocity (RTN)	[2.3, 3.4, 5.9]	[mm/s]
Execution Error	Fixed Pointing σ_{pf}	3.33	[mm/s]
	Proportional Pointing σ_{pp}	6.67	[mrad]
	Fixed Magnitude σ_{mf}	4.67	[mm/s]
	Proportional Magnitude σ_{mp}	0.33%	[-]
Observation Accuracy	1σ Range	3.0	[m]
	1σ Range-rate	0.1	[mm/s]

8.1.3 Results

First, we analyse the case of purely aleatory uncertainty. We start from the deterministic optimal control solution reported in Table 8.3, which was computed by optimising the nominal trajectory with the open-loop control only. This solution meets the flyby and final position constraints but violates the required probability of impact with the

Table 8.2: Interval-valued epistemic parameters for initial covariance multipliers and Gates' parameters in Europa's moon flyby belief optimisation.

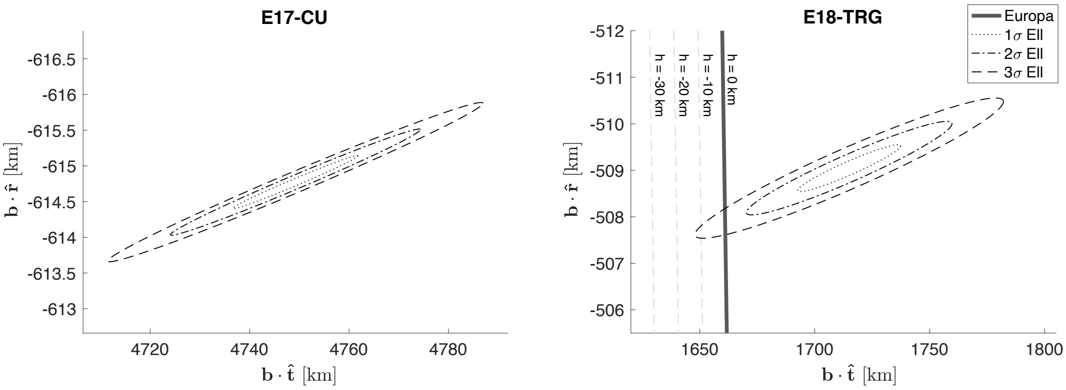
Uncertainty	Component	Value	
Initial Dispersion	Position multiplier $\lambda_{x_{0-1}}$	$[0.5, 2.0]$	[-]
	Velocity multiplier $\lambda_{x_{0-2}}$	$[0.5, 2.0]$	[-]
Execution Error	Fixed Pointing $\widetilde{\sigma}_{pf}$	$[1.67, 4.00]$	[mm/s]
	Proportional Pointing $\widetilde{\sigma}_{pp}$	$[3.33, 8.00]$	[mrad]
	Fixed Magnitude $\widetilde{\sigma}_{mf}$	$[2.33, 6.60]$	[mm/s]
	Proportional Magnitude $\widetilde{\sigma}_{mp}$	$[0.17, 0.40]\%$	[-]

moon when a navigation analysis is performed. In fact, the probability of impact is $\text{PoI} = 0.75\% \nless 0.1\%$. A visualization of the probability of impact for this solution is displayed in Fig. 8.2, where the state uncertainty at different times along the trajectory is propagated onto the B-plane of the Europa flyby E18, without applying successive manoeuvres or performing any new orbit determination campaign. Different samples drawn from the belief state distribution have different velocity vectors resulting in different incoming asymptotes, which, in turn, result in different B-plane orientations. In this application, the mapping of the uncertainty onto the B-plane coordinates is realised by letting the B-plane frame vary for each sample as suggested in [166] for collision scenarios. Hence, Fig. 8.2 employs a single plane for representing all the B-parameters resulting from different state realisations, although their corresponding B-planes are different. Thus, the confidence ellipses are reconstructed from the mapped $\mathbf{b}-$vector

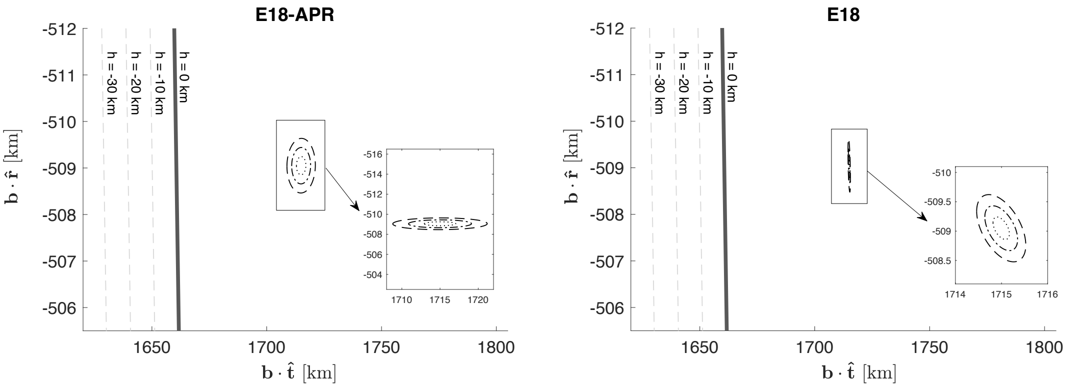
Table 8.3: Free variables, open-loop magnitude $\Delta\overline{v}$, total $\Delta v99$, their ratio and PoI for first guess under aleatoric uncertainty.

Solution	Aleatory First Guess
$\Delta\mathbf{v}_{CU\text{-}E17}$ [m/s]	$[\ 0.0\ ,\ \ 0.0\ ,\ \ 0.0\ ]$
$\Delta\mathbf{v}_{TRG\text{-}E18}$ [m/s]	$[-1.30, +2.86, +3.23]$
$\Delta\mathbf{v}_{CU\text{-}E18}$ [m/s]	$[\ 0.0\ ,\ \ 0.0\ ,\ \ 0.0\ ]$
$\Delta\overline{v}$ [m/s]	4.51
$\Delta v99$ [m/s]	8.05
$\Delta v99/\Delta\overline{v}$ [-]	1.79
PoI [-]	0.75%

samples, each of which has a different reference frame.



(a) *CU-E*17 post-flyby manoeuvre (b) *TRG-E*18 apocenter manoeuvre



(c) *APR-E*18 approach manoeuvre (d) *E*18 closest approach

Figure 8.2: Confidence ellipses in B-vector components mapped from different instances of the first guess trajectory without successive manoeuvres and observations under aleatory uncertainty.

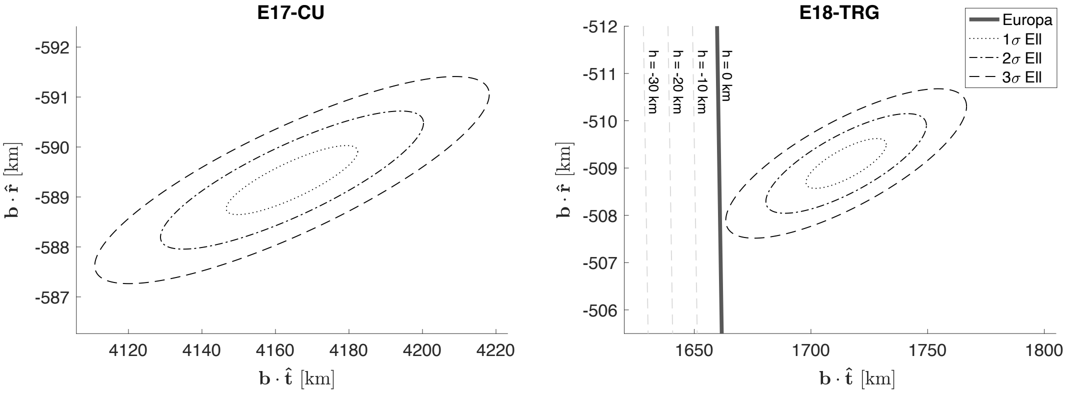
Fig. 8.2(a) shows the ellipses that would result on the B-plane if the spacecraft was not controlled or observed anymore after the E17 clean-up. Fig. 8.2(b) shows the ellipse of uncertainty on the B-plane propagated from the TRG (see Fig. 8.1(b)). The figure shows that without corrections coming from a navigation analysis, the uncertainty on this deterministic trajectory has an intersection with the surface of the Moon (represented by the thick black line at $h = 0$ km), which corresponds to an undesirably high PoI. Fig. 8.2(c) displays the B-plane uncertainty propagated without new OD after the APR (see also Fig. 8.3(c)), which is notably smaller than the post-targeting one because of the new OD arcs and the controller $\delta\Delta\mathbf{v}_{APR\text{-}E18}$. Fig. 8.2(d) finally shows the uncertainty on the B-plane as reconstructed using the OD arcs after the APR. The estimated overall $\Delta v99$ is approximately 79% larger than the open-loop one (see Table 8.3), a cost increase in line with previous navigation analysis for the Europa Clipper trajectory [43].

We now apply BOC, starting from the deterministic solution, to find a trajectory that minimises the $\Delta v99$ while respecting the $\overline{\text{PoI}}$ constraint, and satisfying the constraints on the flyby conditions and terminal position. The resulting solution fulfils the constraint on the probability of impact. A summary of its characteristics is reported in Table 8.4. Looking at the open-loop $\Delta\overline{\mathbf{v}}$ allocation, one can infer that the feasibility on the PoI constraint was realised by trading-off part of the targeting manoeuvre with the clean-up ones. The controller at the apocenter can adjust the execution errors coming from an increased $\Delta\overline{\mathbf{v}}_{CU\text{-}E17}$, while the smaller $\Delta\overline{\mathbf{v}}_{TRG\text{-}E18}$ and associated execution errors result in a smaller uncertainty at the flyby B-plane (see Fig. 8.3(b)). After the flyby, a $\Delta\overline{\mathbf{v}}_{CU\text{-}E18}$ manoeuvre is needed to meet the target conditions at the final time.

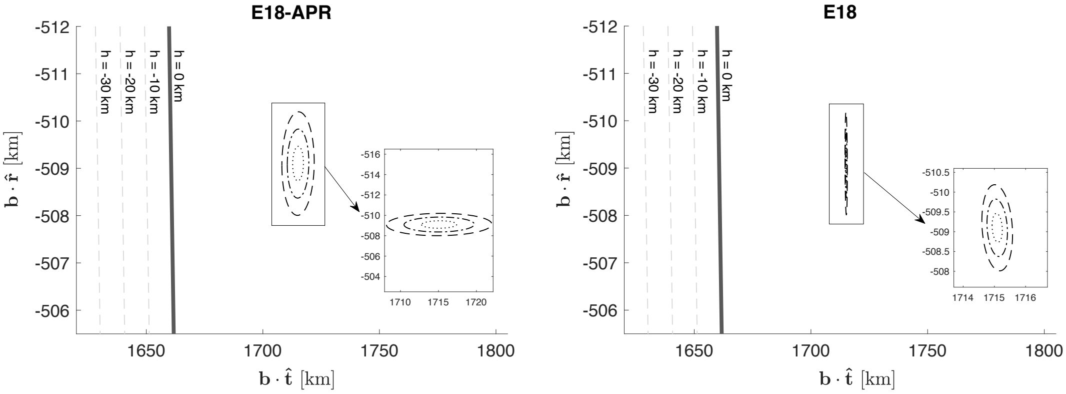
Table 8.4: Free variables, open-loop magnitude $\Delta\overline{v}$, total $\Delta v99$, their ratio and PoI for the robust solution under aleatoric uncertainty.

Solution	Aleatory Robust
$\Delta\overline{\mathbf{v}}_{CU\text{-}E17}$ [m/s]	$[+0.12, +0.18, +0.09]$
$\Delta\overline{\mathbf{v}}_{TRG\text{-}E18}$ [m/s]	$[-2.09, +2.01, +1.99]$
$\Delta\overline{\mathbf{v}}_{CU\text{-}E18}$ [m/s]	$[-1.21, +0.56, -1.79]$
$\Delta\overline{v}$ [m/s]	5.98
$\Delta v99$ [m/s]	9.91
$\Delta v99/\Delta\overline{v}$ [-]	1.66
PoI [-]	0.09%

Overall the optimal trajectory has a higher open-loop magnitude $\Delta\overline{v}$ than the first guess, but the infeasibility is restored. We observe that the percentage increase of the $\Delta v99$, with respect to the open-loop $\Delta\overline{v}$, is now lower, i.e. 66%, indicating that the BOC trajectory can compensate for the possible uncertainty realisations more efficiently. Fig. 8.3 shows the B-plane uncertainties of the robust trajectory. Fig. 8.3(b), representing the uncertainty mapped after TRG, shows that the 3-σ ellipse does not cross the surface



(a) *CU*-*E*17 post-flyby manoeuvre (b) *TRG*-*E*18 apocenter manoeuvre



(c) *APR*-*E*18 approach manoeuvre (d) *E*18 closest approach

Figure 8.3: Confidence ellipses in B-vector components mapped from different instances of the robust trajectory without successive manoeuvres and observations under aleatory uncertainty.

of Europa anymore. By comparing it with the corresponding plot in Figure 8.2(b), the robust B-plane ellipse has a smaller semi-major axis, which mainly contributes to the PoI, whereas it has a larger semi-minor axis, which has a limited contribution to the PoI constraint. The robust ellipse is also rotated counterclockwise, which leads to an even lower impact probability.

To quickly verify that the increase in total $\Delta v99$ between the robust solution and the first guess is due to the initial infeasibility, the BOC problem has been solved after removing the PoI constraint. Table 8.5 shows the unconstrained optimal solution, which displays a lower overall $\Delta v99$ compared to that of the deterministic solution, although the open-loop magnitude $\Delta \overline{v}$ is slightly higher. This result further confirms that the robust optimum differs from the deterministic one and that the statistical performance is indeed improved.

Table 8.5: Free variables, open-loop magnitude $\Delta\overline{v}$, total $\Delta v99$, their ratio and PoI for verification solution without imposing the PoI constraint.

Solution	Aleatory Robust w/o PoI
$\Delta\overline{\mathbf{v}}_{CU\text{-}E17}$ [m/s] $\Delta\overline{\mathbf{v}}_{TRG\text{-}E18}$ [m/s] $\Delta\overline{\mathbf{v}}_{CU\text{-}E18}$ [m/s]	$[\ 0.0\ , -0.01, -0.01]$ $[-1.28, +2.89, +3.25]$ $[-0.04, -0.02, +0.03]$
$\Delta\overline{v}$ [m/s]	4.60
$\Delta v99$ [m/s]	8.02
$\Delta v99/\Delta\overline{v}$ [-]	1.74
PoI [-]	0.76%

Fig. 8.4 provides an insightful visualisation of how the manoeuvres and observation arcs affect the flyby uncertainties. In particular, the figure shows, on the y-axis, the Semi-MAjor Axis (SMAA) and Semi-MInor Axis (SMIA) of the B-plane 1-σ confidence ellipse, and of the uncertainty on the hyperbolic TCA for different times along the trajectory (x-axis), if no other action is taken after that time. The observation instances are represented with vertical dashed black lines, while manoeuvres are indicated by black solid lines.

The value of the mapped uncertainty is around 10 km and 500 m in SMAA and SMIA, and 10 seconds in TCA, as resulting purely from the initial dispersion as in Table 8.1. The mapped uncertainty exhibits then a jump at E17, the time of the clean-up manoeuvre, because of the executions errors. Successively, the orbit determination arcs reduce the B-plane ellipsoid by more than one order of magnitude in SMAA and TCA, whereas the reduction in SMIA is more contained. The main targeting manoeuvre $\Delta\overline{\mathbf{v}}_{TRG\text{-}E18}$ and its high execution errors cause a major spike in the delivered uncertainty. The values of SMAA and SMIA at this event are critical for the robust optimisation process, as this B-plane mapped uncertainty is the one employed for the probability of impact computation. Successive OD arcs help to reduce the SMAA significantly and TCA mapped dispersion until another, more contained, jump at the approach manoeuvre

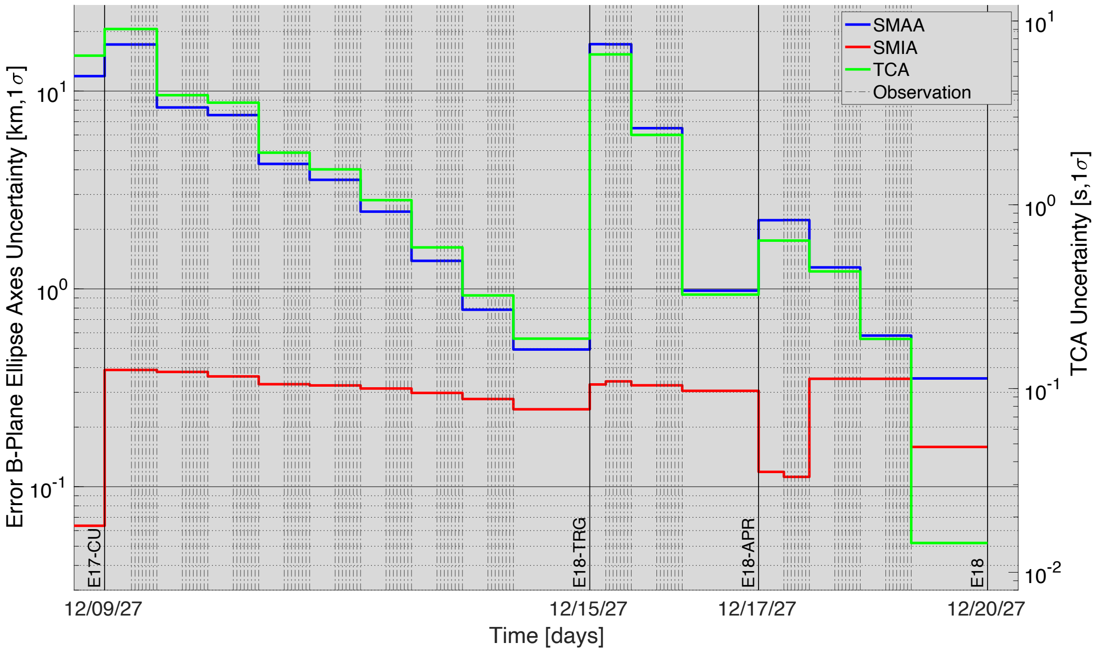


Figure 8.4: Uncertainty mapped from different times during the robust trajectory to the reference flyby time and transformed in B-plane coordinates.

before E18. Finally, the measurements arcs before E18 reduce the mapped uncertainty even further, to have an expected 1-σ uncertainty at flyby of a few hundred meters in SMAA and SMIA, and a few tenths of a second for the TCA.

Starting from the same deterministic solution, we now introduce epistemic uncertainties in the distributions and solve the full problem (8.6). In this case, for the deterministic solution reported in Table 8.6 we have $\overline{\text{PoI}} = 2.29\% \not< 0.1\%$, which violates the impact constraint even more severely than in the aleatory case. The total ΔV_{tot} that provides the required percentile $\Delta v99$ is significantly higher than the one in the purely aleatory case, mainly due to the larger execution errors.

The corresponding B-plane uncertainty ellipses mapped and delivered from the manoeuvre instances along the trajectory are visualised in Fig. 8.5. For each subfigure, multiple 3-σ ellipsoids are represented by taking samples of the epistemic parameters within their intervals (see Table 8.2) and running a full navigation analysis for each epistemic value. By comparing Fig 8.2 with Fig. 8.5, one can see that in every subplot the ellipses change in size and, at times, rotate due to the epistemic uncertainty. For

Table 8.6: Free variables, open-loop magnitude $\Delta\overline{v}$, total $\overline{\Delta v99}$, their ratio and upper PoI for first guess under epistemic uncertainty.

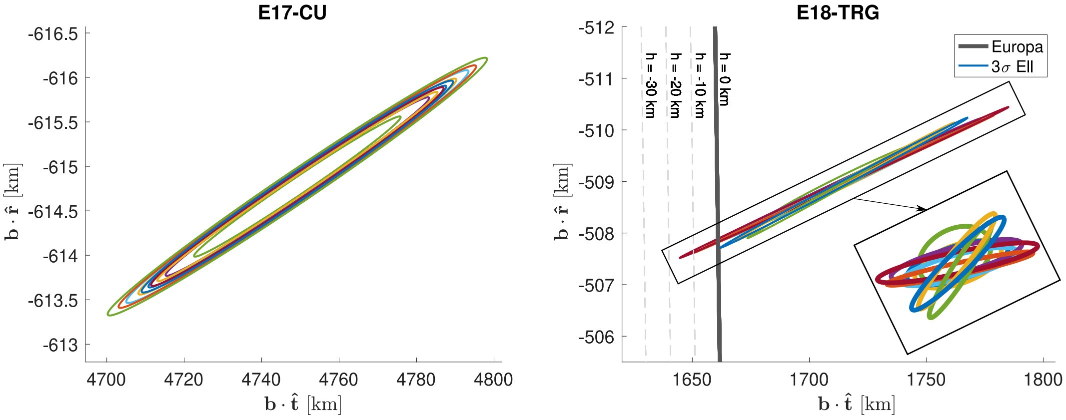
Solution	Epistemic First Guess
$\Delta\overline{\mathbf{v}}_{CU\text{-}E17}$ [m/s]	$[\ 0.0\ ,\ 0.0\ ,\ 0.0\ ]$
$\Delta\overline{\mathbf{v}}_{TRG\text{-}E18}$ [m/s]	$[-1.30, +2.86, +3.23]$
$\Delta\overline{\mathbf{v}}_{CU\text{-}E18}$ [m/s]	$[\ 0.0\ ,\ 0.0\ ,\ 0.0\ ]$
$\Delta\overline{v}$ [m/s]	4.51
$\overline{\Delta v99}$ [m/s]	8.65
$\overline{\Delta v99}/\Delta\overline{v}$ [-]	1.92
$\overline{\text{PoI}}$ [-]	2.29%

the chance constraint on the $\overline{\text{PoI}}$ after targeting, the inner optimisation routine looks for the epistemic sample, which yields the ellipse with the largest intersection with the equivalent Europa surface in Fig. 8.5(b).

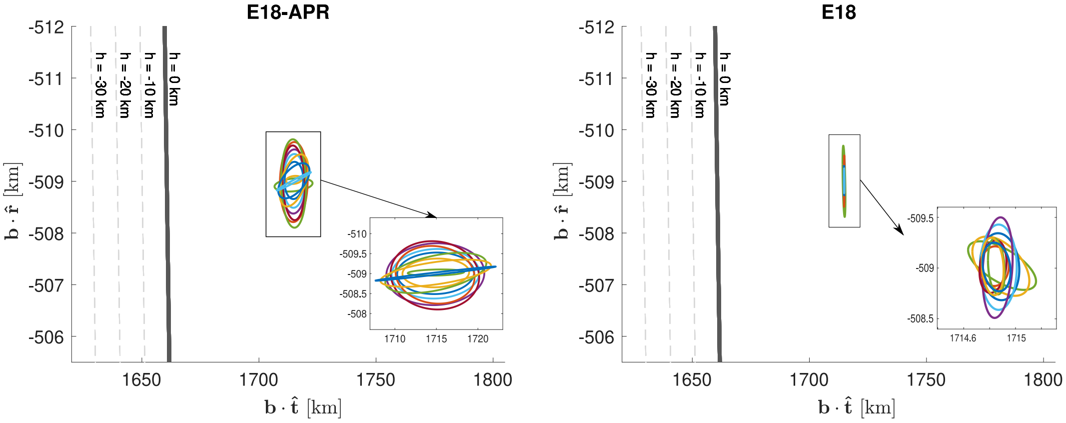
The main features of the BOC solution to problem (8.6) are reported in Table 8.7. Again, the values of the expected value constraints are not reported as they are met up to the required threshold, that is, the new trajectory respects the required flyby conditions and final position in expected value. The $\overline{\text{PoI}}$ constraint is satisfied by trading part of the targeting manoeuvre with the E17 and E18 clean-up manoeuvres. The latter is now the largest manoeuvre employed to steer the spacecraft back to the desired final conditions. The value of $\Delta v99$ is significantly higher than in the aleatory

Table 8.7: Free variables, open-loop magnitude $\Delta\overline{v}$, total $\overline{\Delta v99}$, their ratio and upper PoI for the robust solution under epistemic uncertainty.

Solution	Epistemic Robust
$\Delta\overline{\mathbf{v}}_{CU\text{-}E17}$ [m/s]	$[-0.35, +1.12, +0.16]$
$\Delta\overline{\mathbf{v}}_{TRG\text{-}E18}$ [m/s]	$[-1.73, +1.57, +1.30]$
$\Delta\mathbf{v}_{CU\text{-}E18}$ [m/s]	$[+1.43, -2.13, -3.24]$
$\Delta\overline{v}$ [m/s]	7.99
$\overline{\Delta v99}$ [m/s]	14.76
$\overline{\Delta v99}/\Delta\overline{v}$ [-]	1.84
$\overline{\text{PoI}}$ [-]	0.04%



(a) *CU-E*17 post-flyby manoeuvre (b) *TRG-E*18 apocenter manoeuvre



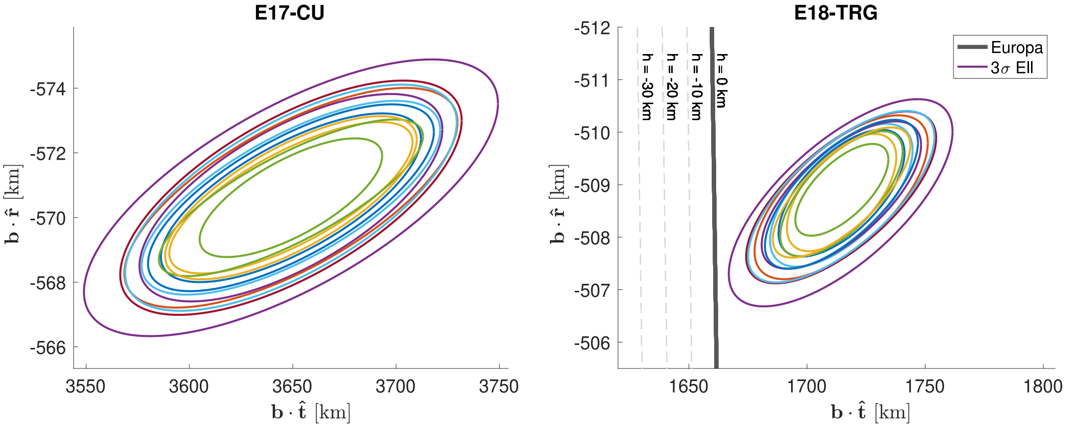
(c) *APR-E*18 approach manoeuvre (d) *E*18 closest approach

Figure 8.5: Confidence ellipses in B-vector components mapped from different instances of the first guess trajectory without successive manoeuvres and observations under epistemic uncertainty.

case in Table 8.4 due to:

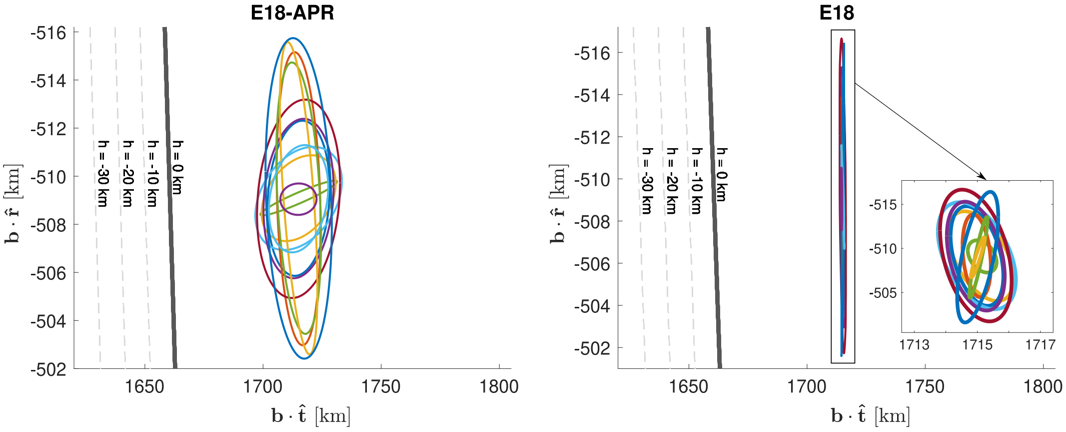
- the more severe uncertainty coming from the unknown probability distribution;
- the larger execution errors causing bigger deviations and, therefore, higher statistical manoeuvres.

A visualisation of the 3-σ uncertainty ellipses on the B-plane for the robust solution can be found in Fig. 8.6, where again each ellipsoid results from a different value of the epistemic parameters. By comparing this plot with Fig. 8.5, Fig. 8.6(a) displays an evident change in the B-plane parameters after E17-CU, as the larger clean-up manoeu-



(a) *CU-E*17 post-flyby manoeuvre

(b) *TRG-E*18 apocenter manoeuvre



(c) *APR-E*18 approach manoeuvre

(d) *E*18 closest approach

Figure 8.6: Confidence ellipses in B-vector components mapped from different instances of the robust trajectory without successive manoeuvres and observations under epistemic uncertainty.

vre steers the delivered uncertainty closer to the target flyby conditions. Fig. 8.6(b), representing the mapped uncertainty after targeting, shows that there is no intersection between the largest 3-σ ellipse and the equivalent surface of Europa, confirming that the PoI constraint is also met in the epistemic case. Similarly to the precise case, the ellipses have smaller semi-major axes, larger semi-minor axes and are rotated counterclockwise to reduce the $\overline{\text{PoI}}$ while keeping the mean on the desired flyby conditions. Figure 8.6(c) reveals an increase in the delivered uncertainty after approach due to the larger statistical manoeuvres. Finally, Fig. 8.6(d) shows how the range and range-rate observations reduce the reconstructed uncertainty at flyby, mainly in the $\mathbf{b} \cdot \hat{\mathbf{t}}$ component, whereas the $\mathbf{b} \cdot \hat{\mathbf{r}}$ one is more difficult to observe from range and range-rate Earth-based measurements.

8.2 Epistemic navigation analysis

In this section, the developed epistemic filtering is applied for a preliminary NA of one leg of the Europa Clipper tour [186], already discussed in Section 8.1, to quantify the PoI during a close encounter with Jupiter's moon Europa under epistemic observation's likelihoods as well.

8.2.1 Problem definition

The simplified scenario studied is represented in Figure 8.7. The trajectory spans again two successive flybys with an apocenter impulsive manoeuvre to target the second flyby (E18) identified by the black square. The spacecraft trajectory and its deviations are depicted with dotted black lines, Europa's orbit with a dotted blue line, the observations with a dashed yellow line and the $\Delta\mathbf{v}$ manoeuvre with a green arrow. Hence, the spacecraft trajectory starts from a close flyby, and the initial uncertainty is propagated until the apocentre while performing OD. The OD campaign is carried out again with

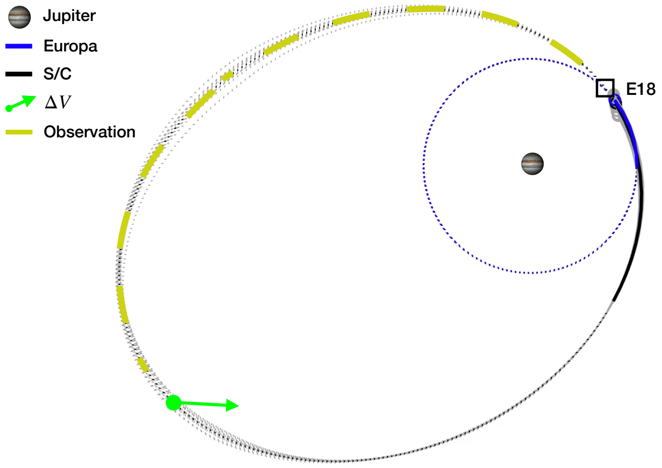


Figure 8.7: Sketch of navigation analysis scenario for Europa Clipper leg.

an 8 hours ON 8 hours OFF schedule to faithfully model the part-time availability of the tracking stations availability. The PoI is then computed by propagating the uncertainty after the $\Delta\mathbf{v}$, and its associated execution errors, to the nominal flyby time without performing additional OD. It is worth reminding that the PoI is computed in such a way to ensure spacecraft safety and environmental protection to Europa even in the event of a communication loss after the main manoeuvre. Also, the delivered PoI after the apocentre manoeuvre is usually the more critical measure because: this $\Delta\mathbf{v}$ brings the spacecraft close to the moon; it introduces the largest execution errors; the state uncertainty can grow severely during the long propagation time.

The trajectory considered here lasts for 14 days. The motion is described in a Europa-centred inertial reference frame, the ECLIPJ2000. The spacecraft dynamics is a high-fidelity full-ephemeris one, including the gravitational fields of Jupiter (central and J_2), of its moons Europa (central and J_2), Io, Ganymede and Callisto, and of the Sun. Again, the dynamics is numerically propagated with the library jTOP [21].

The goal is to quantify tight bounds on the PoI at flyby $[\underline{\mathrm{PoI}}, \overline{\mathrm{PoI}}]$. The PoI is computed as in Eq. (8.8).

8.2.2 Uncertainty model

The initial distribution $p(\mathbf{x}_0; \boldsymbol{\lambda}_0)$ is written in the Gaussian mixture form as

$$p(\mathbf{x}_0; \boldsymbol{\lambda}_0) = \sum_{j=1}^{M} \lambda_0^{(j)} \mathcal{N}(\mathbf{x}_0; \boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0^{(j)}) \tag{8.11}$$

with interval-valued weights

$$\lambda_0^{(j)} = [0, 1] \quad \forall j = 1, \dots, M \ . \tag{8.12}$$

The mean is the same for all the components, and it is the nominal initial state. The covariances are defined as

$$\boldsymbol{\Sigma}_0^{(j)} = \mathrm{blkdiag}\Big(\lambda_{0-1}^{(j)} \, \boldsymbol{\Sigma}_0(1{:}3, 1{:}3), \lambda_{0-2}^{(j)} \, \boldsymbol{\Sigma}_0(4{:}6, 4{:}6) \Big) \tag{8.13}$$

where $\lambda_{0-1}^{(j)}, \lambda_{0-2}^{(j)} \in [0.5, 2.0]$ are two multipliers scaling the reference covariance $\boldsymbol{\Sigma}_0$, $\boldsymbol{\Sigma}_0(1{:}3, 1{:}3)$ and $\boldsymbol{\Sigma}_0(4{:}6, 4{:}6)$ indicate respectively the position block and the velocity block and the operator blkdiag indicates a block-diagonal matrix. Being the multipliers defined within $[0.5, 2.0]$, they encompass kernels with covariance from half up to double the extent of the reference one. The reference covariance is defined from the one in Radial Transversal Normal (RTN) components

$$\boldsymbol{\Sigma}_{RTN} = \mathrm{diag}\Big([3.7^2, 5.3^2, 9.3^2, 2.3e\text{-}3^2, 3.4e\text{-}3^2, 5.9e\text{-}3^2]\Big) \tag{8.14}$$

where diag indicates a diagonal matrix and the unit of the first three elements is [m^2] while for the last three is [m^2/s^2]. Then, the reference covariance in inertial rectangular coordinates is computed as

$$\boldsymbol{\Sigma}_0 = \mathcal{J}_{RTN}^{Car} \boldsymbol{\Sigma}_{RTN} \mathcal{J}_{RTN}^{Car^T} \tag{8.15}$$

where $\mathcal{J}_{RTN}^{Car}$ is the Jacobian of the transformation from RTN to Cartesian coordinates.

The execution errors of the impulsive manoeuvre are modelled with Gates' model [169, 170], which decompose the additive error in magnitude (along the commanded $\Delta\mathbf{v}$ direction) and pointing components (perpendicular to the commanded $\Delta\mathbf{v}$ direction) as depicted in Figure 5.9. The parameters of the Gates' model are reported in Table 8.8. This model has both fixed parameters, that is, an error appears whenever a manoeuvre is performed, and proportional parameters, that is, the error is larger for larger manoeuvres.

Table 8.8: Parameters Gates' model for execution errors.

Fixed Pointing σ_{pf}	3.33	[mm/s]
Proportional Pointing σ_{pp}	6.67	[mrad]
Fixed Magnitude σ_{mf}	4.67	[mm/s]
Proportional Magnitude σ_{mp}	0.33%	[-]

As for the OD, the tracking stations measure the spacecraft range and range rate with respect to Earth [162]. The likelihood function is modelled as Gaussian mixture with epistemic coefficients as

$$p(\mathbf{y}_k|\mathbf{x}_k;\boldsymbol{\lambda}_y) = \sum_{j=1}^{M} \lambda_y^{(j)} \mathcal{N}(\mathbf{y}_k^{(j)}; h(\mathbf{x}_k), \boldsymbol{\Sigma}_y) \tag{8.16}$$

where h is the range and range-rate observation model, and the weights are interval-valued as

$$\lambda_y^{(j)} = [0,1] \quad \forall j = 1, \ldots, M \ . \tag{8.17}$$

This time the covariance, quantifying the measurement accuracy, is fixed and set to

$$\boldsymbol{\Sigma}_y = \mathrm{diag}\Big([3.0^2, 0.1^2]\Big) \ , \tag{8.18}$$

where the unit of the range is [m^2] and for the range-rate is [m^2/s^2]. The exact value of the received measurement is unknown at the time of mission design, and therefore it is considered as an epistemic parameter $\mathbf{y}_k^{(j)}$.

The number of mixture components has been set to $M = 100$ for the initial distribution, the likelihood function and the variational distribution.

8.2.3 Results

The developed epistemic navigation analysis has been run on the described test case to robustly quantify the PoI range resulting from epistemic uncertainty. Figure 8.8 displays the resulting 3-σ uncertainty in B-plane coordinates at flyby E18. In detail,

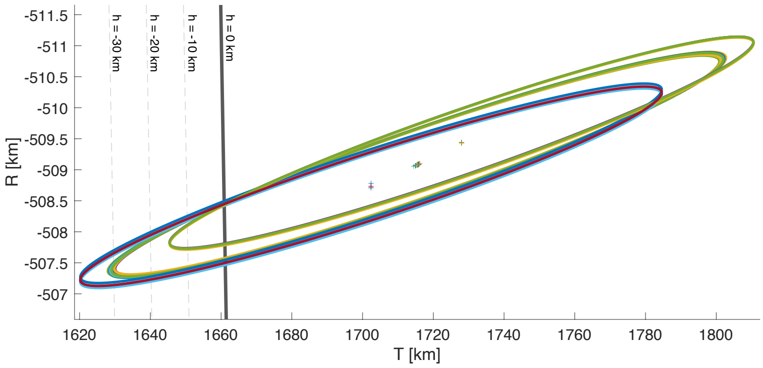


Figure 8.8: Spacecraft 3-σ epistemic ellipses in B-plane coordinates.

the several coloured ellipses result from different instances of the epistemic coefficients in the mixture representation. The equivalent Europa's surface is represented by the bold black line at $h = 0$ km, while the dashed grey ones represent Europa's subsurface for different depths. It can be seen that all the ellipses have a significant nonzero intersection with Europa's surface, but different instances of the epistemic parameters have a relevant influence on both the extension and the displacement of the uncertainty region.

The bounds on the PoI computed by Equation (3.61) using $\psi_{r_{E18}}$ are

$$\mathrm{PoI} \in [0.6, 6.5]\,\% \; . \tag{8.19}$$

The width of this interval indicates that epistemic uncertainty has a large effect on the value of the collision probability. In particular, epistemic uncertainty on the initial covariance and the observation realisation has a large impact on the delivered uncertainty.

8.3 Chapter summary

On the re-design of the Europa Clipper flyby tour, the BOC formulation was shown to simultaneously optimise the total cost of the executed manoeuvres and to satisfy all constraints under both aleatory and epistemic uncertainty. Both aleatory and epistemic uncertainties were considered in the initial conditions, execution errors and observation noises. The BOC was able to solve for the initial infeasibility in PoI of the deterministic solution and find a robust trajectory that simultaneously satisfies the statistical constraints and minimises the $\Delta v99$. As a verification, it was shown that the BOC solution outperforms the deterministic optimal initial guess when no PoI constraint is considered.

Successively, the developed NA with variational inference was applied to the robust quantification of collision probability bounds for Europa Clipper during one of its flybys. This test case encompassed epistemic uncertainty again on the initial dispersion and, in addition to the previous study, in the observation likelihood, thus resulting in epistemic priors and posteriors along all the trajectory. OD arcs were performed before the main manoeuvre, which introduced executions errors as well, and then the PoI was quantified on the delivered uncertainty at the flyby. The range of PoI resulting from epistemic uncertainty was rather large, showing the importance of modelling and processing this systematic component as well in mission design. Indeed, in general, while the lower value $\underline{\mathrm{PoI}}$ may be considered safe for some applications, the upper value $\overline{\mathrm{PoI}}$ may not. A purely aleatoric approach, in which only precise distributions can be specified, would only be able to return a single value within the interval $[\underline{\mathrm{PoI}}, \overline{\mathrm{PoI}}]$, therefore providing only limited information to the mission designer. The upper probability displayed a significant increase with respect to the quantification in the BOC case, thus showing the impact of accounting for observation likelihood's epistemic uncertainty on the probability metric.

Chapter 9

Conclusion

The main objective of this thesis was to develop new methods for state estimation, optimal control, and operations under mixed aleatory and epistemic uncertainty. The ultimate interest for space applications was to improve the space and ground segment robustness, reliability, and autonomy by introducing proper uncertainty treatment in mission design and analysis as well as space traffic management. The multiple objectives defined in the introduction of the thesis have been met and addressed in the different chapters.

Part I of the thesis was allocated to theoretical and methodological developments to meet the research objectives and tackle the entailed challenges. Each chapter was dedicated to one major topic: uncertainty propagation, state estimation, and optimal control.

Chapter 2 presented the epistemic uncertainty model employed and the two-step approach for uncertainty propagation employed throughout this thesis. This approach consists of constructing a polynomial approximation of the dynamical flow and then employing the polynomial mapping for efficiently computing expectations. For the first, two approaches were proposed, one intrusive using generalised polynomial algebra and one non-intrusive using stochastic collocation, and the applicability and advantages of both methods were discussed. In this context, a reinitialisation approach was proposed to mitigate the curse of dimensionality in multi-segment and multi-phase problems. This approach consists of outer bounding the propagated polynomial with a reinitialised one to ensure continuity of pointwise trajectories, which are recovered by evaluating a sequence of polynomials. The reduced dimensionality comes at the expenses of propagating larger regions than strictly required, and therefore a potential decline in accuracy. The reinitialisation approach proved crucial in constructing direct transcriptions and polynomial-time algorithms for optimal control under uncertainty. Successively, practical approaches for computing expectations or moments of the propagated distribution were discussed: Monte Carlo methods for generic distributions and Gauss-Hermite cubature rules for normal densities. Among them, importance sampling was extensively applied in all the developments concerning epistemic uncertainty for its ability to draw samples from a single proposal distribution. These methods enabled efficient nonlinear uncertainty propagation with tunable accuracy depending on the polynomial degrees.

Chapter 3 presented developments in the field of state estimation and navigation under generalised models of uncertainty. First, the formulation of the state estimation problem under epistemic uncertainty was presented. Such a problem aims to compute robust bounds on the expectation of a generic quantity of interest, e.g. state variable, collision probability, etc. Then, two solution approaches were developed. The robust particle filter is a sequential Monte Carlo method that exploits particles' precomputation and performs an optimisation over the estimator weights to compute the sought bounds efficiently. The expectation and bound estimators were presented and their theoretical properties analysed. Complexity analyses for the precomputation, estimator and its derivative evaluations were performed. It was shown that the estimator complexity is linear with the number of particles and observations with interaction. The numerical analysis performed in the corresponding application chapter confirmed the theoretical result. A B&B approach with simplicial domains was developed to ensure asymptotic convergence to the global bound. Theoretical proofs for the optimisation convergence and an analytical method to find the minimum of the bounding function were derived. The second approach is a sequential filtering method that exploits a combination of variational inference and importance sampling to reinitialise the epistemic representation of the state at each observation instance. This specific reinitialisation procedure makes this filtering method sequential and helps avoid the accumulation of uncertainty sources at different instances, e.g. new observations. This epistemic variational inference was specialised for Gaussian mixture distributions, with the weights being the epistemic parameters. In this context, the bound computation routine was shown to reduce to a simple linear programming problem that can be efficiently solved. Both the methods were developed with the goal of maintaining the dynamical and observation nonlinearities and without enforcing restricting approximations or assumptions. Besides, such an epistemic approach to filtering allows one to include broader models of uncertainty and, therefore, to characterise more faithfully the uncertainty and information structure available during different operational scenarios.

Chapter 4 dealt with optimal control problems under severe uncertainty to incorporate proper uncertainty quantification in trajectory and manoeuvre design. A formulation was developed that generalises optimal control to mixed aleatory and epistemic models of uncertainty. For this, a direct shooting transcription employing generalised intrusive polynomial algebra was developed. The low-thrust control was embedded in the polynomial representation to avoid a new polynomial propagation for every control profile to evaluate. The composite polynomial surrogate constructed using the reinitialisation approach was employed to ensure lower computational complexity and a higher approximation accuracy for the same polynomial degree. Successively, this transcription was expanded to include orbit determination and closed-loop control profiles. In this setting, a belief optimal control formulation was proposed to model the uncertain problem in terms of probability densities directly rather than state realisations. This formulation directly derives from belief Markov decision processes which make partially observable processed become observable by changing the state notation. Thus, BMDP is an elegant model to incorporate noisy measurements in the optimal control under uncertainty. Hence, propagation and inference maps were developed to describe the belief state evolution respectively in time and at an observation instance. The former employs again a polynomial representation to speed up the propagation of samples, whereas the latter uses the Bayes' update rule. The polynomial surrogates employed in both approaches ensure a high-fidelity and tunable representation of the dynamical flow, which can be made more accurate by increasing the polynomial degree. These approaches allow one to handle both aleatory and epistemic uncertainty. Therefore they are suited for robust manoeuvre design in highly uncertain scenarios, e.g. collision avoidance in LEO with a non-operational debris or preliminary trajectory design when most system parameters are only partially specified. Besides, by including flight dynamics and navigation metrics, these approaches potentially yield a reduced number of iterations between the trajectory optimisation and flight dynamics teams during the mission design process.

Then, Part II of the thesis focused on the application of the developed approaches to STM conjunction assessment as well as space mission design and analysis. These test cases were characterised by nonlinear dynamical and observation models, complex optimisation and navigation challenges, and generalised uncertainty models on the system state and sensor outputs.

Chapter 5 focused on applications of the novel robust filtering and navigation approaches developed. The first test case dealt with the robust estimation of PoC between SENTINEL 2B and a FENGYUN 1C debris under mixed aleatory and epistemic uncertainty on the initial conditions and observation likelihoods. Different operational scenarios were considered. The RPF was shown to efficiently and accurately quantify the probability bounds in all cases and to provide insightful information to be used for decision making. The evolution of the probability bounds as new observations are acquired was discussed in the different scenarios. The effective sample size of the estimates was large mainly thanks to the employed UKF-based proposal and the polynomial propagator. The large differences between the lower and upper distributions provided evidence that further highlights how sensitive the collision probability metric is to input distributions specification and, therefore, the importance of considering epistemic uncertainty. Then, the same scenarios were analysed using a robust Bayesian framework to allocate actions given risk metrics and the conjunction geometry autonomously. In addition to before, avoidance manoeuvres were designed and executed in high-risk scenarios. The CAM model was designed as a BOC problem to include execution errors on the manoeuvre. The RPF was incorporated within the Bayesian framework as state estimation engine to compute the uncontrolled collision metric and within the CAM optimisation to design manoeuvres that yield risk figures below a safety threshold. The filter proved to be a useful tool for the intelligent agent in deciding which action to take. In all cases, after executing the appropriate action, the state was re-estimated by the RPF to re-evaluate the collision probability. On the collision scenario, the CAM module was able to design a robust manoeuvre to reduce the upper PoC below the safety threshold. This test case proved the crucial role of modelling epistemic uncertainty in highly uncertain scenarios as the risk metrics were shown to be greatly sensitive to the input distributions. The developed methods proved effective in estimating reliable metrics both in collision and non-collision scenarios, as well as designing robust manoeuvres to steer the satellite in a safe orbit with the minimum propellant expenditure.

Chapter 6 presented a scheduling method for optimal tracking under state and observation uncertainties. Specifically, it showed the development of an optimisation-based approach to automatically generate optimal observation schedules from heterogeneous sensors under budget constraints. The optimal scheduling was framed as a variable-size problem, where the fitness function required a full nonlinear quantification of the satellite state uncertainty given noisy observations. A structured-chromosome genetic algorithm was specifically designed to handle variable-size problems and automatically construct optimal observation campaigns. For the test cases, uncertainty was included in both the initial dispersion and observation likelihoods. The first application focused on the observation campaigns generation for a satellite in a very-low Earth orbit characterised by a highly perturbed dynamical environment. The developed approach was tested on different ground networks configurations and different levels of budget constraints. The second case dealt with the robust observation and TCMs allocation for a low-budget deep-space satellite. Execution errors are included after each observation arc. Thus, this test case was characterised by a critical trade-off between the instantaneous state knowledge improvement provided by a measurement arc and the introduced velocity errors, which tended to worsen future predictions. The results of both test cases have shown the ability of the optimisation-based method to optimally generate observation campaigns by trading off accuracy with budget allocation and execution errors. The presented analyses validate the design choice of using a variable-size structured genetic algorithm when compared to the standard and hidden-genes GAs, in particular for scenarios with tighter constraints. These applications have proven that the developed approach is a solid tool for automatic resources allocation in space tracking applications under severe budget limitations.

In Chapter 7, the direct transcription with generalised intrusive polynomial algebra was applied to the robust optimisation of a low-thrust rendezvous to the near-Earth asteroid 99942 Apophis. Uncertainty was considered on the initial conditions as resulting from errors on the hyperbolic excess velocity at departure. The robust control approach managed to find a trajectory achieving the required probability to reach the final target set. Furthermore, the robust solution was shown to outperform the reference deterministic one on this uncertainty metric significantly. The performance of the intrusive polynomial-based method was assessed, and it resulted to be a fast and accurate approach for nonlinear UP. Epistemic uncertainty in the hyperbolic excess velocity was also considered for a sensitivity analysis on the found solution.

Finally, Chapter 8 then presented the application of the belief transcription to the robust re-design of an infeasible flyby of Europa Clipper. Both aleatory and epistemic uncertainties were considered in the initial conditions, execution errors, and observation noises directly within the trajectory design cycle. Constraints were imposed both in expectation and in probability (chance constraint). The robust formulation and solution approach were able to find a trajectory that simultaneously optimises the statistical $\Delta v99$ of the executed manoeuvres and satisfies all the constraints. Indeed, the belief optimal control was able to solve the initial infeasibility in PoI with Europa during the close encounter, as well as to respect the nominal flyby conditions and final position in expectation, such that the rest of the tour is left unchanged. As a verification, it was shown that the robust approach finds a solution that outperforms the deterministic one on the robust measure $\Delta v99$ when no PoI constraint is considered. The discrepancy between the upper and precise PoI highlights the sensitivity of the estimation process to the specification of the input distributions. Therefore, it proves the importance of modelling epistemic uncertainty in robust trajectory design. This example proved the usefulness of BOC in practical trajectory design as it solved for the PoI infeasibility automatically, which otherwise would have taken several iterations between the trajectory optimisation and flight dynamics teams to fix. In this chapter, a preliminary application of the epistemic variational inference on the same test case was also proposed for the robust quantification of PoI. In addition to the uncertainty model used before, epistemic uncertainty was considered on the observation likelihood as well. The wide range of PoI highlighted how crucial epistemic uncertainty modelling and processing are to quantify the navigation analysis' risk metrics accurately. Indeed, different actions might be taken when considering either the lower or the upper impact probability. The increased upper impact probability with respect to the BOC metric was due mainly to further epistemic uncertainty on the observation likelihood. Therefore, this extended analysis proved how sensitive the impact probability is to the specification of observation errors and, therefore, the usefulness of epistemic variational inference.

Although the research objectives of this thesis were met, future developments would help improve the maturity, applicability, and efficiency of the methods presented, as well as overcome some limitations currently affecting them.

As for the robust particle filter, the main limitation comes from the computational complexity for increasing epistemic dimensionality, primarily due to the initial domain decomposition which currently grows factorially. Hence, different aspects would deserve further investigations: a more scalable initial domain decomposition strategy to reduce the computational complexity with the number of epistemic dimension; the use of a resampling strategy to enhance the representation accuracy of the estimator if the effective sample size degrades; an analysis of alternative methods to construct the proposal; the use of alternative optimisation approaches in place of the branch and bound method. Last but not least future test cases will consider more complex collision models, e.g., accounting for both objects uncertainties, their shape, and orientation to provide more accurate risk metrics.

Future work on the epistemic variational inference should focus on both the steps characterising the filter. For the UP, an epistemic update routine for the mixture weights during propagation should be investigated as the current implementation keeps them fixed. For the update step, one limitation of the present algorithm is the use of local Gaussian kernels with fixed mean which may not provide a good global coverage of the posterior set. Thus, future developments should revolve around the employment of different kernels, which provide better global coverage of the domain, such as Bernstein polynomials and nonlinear epistemic parameters. From here, theoretical work should focus on developing enhanced methods and proofs for the epistemic reinitialisation with nonlinear parameters that conservatively approximates the infinite set of possible posteriors resulting from Bayes' inference. Finally, a complexity analysis should be performed to compare the numerical performance of epistemic variational inference versus existing methods.

For the optimisation-based stochastic scheduling, the current implementation is characterised by a number of limitations which prevent its applicability in real-world scheduling. Among them, the approach can now schedule the observation campaign for a single object only, optimise for a single objective, that is the final estimate accuracy, and it relies on a simplified model of the GS' requirements and characteristics. Thus, next developments will concern a faithful modelling of the GSs' physical and operational characteristics as well as the method applicability to the scheduling of multiple objects concurrently while automatically addressing reservation conflicts. Furthermore, future work should focus on its formal generalisation as a multi-objective problem. Indeed, an efficient implementation of multiple budget levels to generate Pareto fronts for a tracking window, comparing different levels of accuracy and allocated budget, could help the operator decide the most suitable schedule to select. Besides, a dynamic approach could be developed where the tracking optimisation decides a subset of observations ahead in time, updates the state estimate with the actual observations received, and re-optimises future tracking schedules. This would improve the tracking optimality in online applications and partially remove the need for covariance analysis.

The robust Bayesian framework could be further expanded by working both on the individual modules and on their interactions. Besides, the main limiting factor to its applicability to current STM procedures is that the estimation process currently relies on raw measurements, which however are not publicly available in general. Thus, future developments would need to incorporate CDM observations directly and understand how to define robust manoeuvres in the case of uncertainty in the correctness of the CDMs. Besides, another development could be to extend the CAM module to a multi-manoeuvre optimisation such that a sequence of avoidance manoeuvres ensures safety not only for the imminent conjunction but also for possible subsequent ones.

Finally, an interesting direction for the control methods would be to incorporate epistemic variational inference as sequential filtering engine. Indeed, the current implementation of the observation update step does not enable the use of sequential epistemic reinitialisation. Using an epistemically sequential filtering approach would enable easy treatment of epistemic uncertainty on the observation likelihood and a simplified optimisation over the epistemic variables as the inner loop would reduce to a linear programming problem. This improvement would also enhance the computational complexity of BOC with an increasing number of epistemic dimension. Besides, as the BOC framework already enables concurrent optimisation of open- and closed-loop control profiles, it could be further developed to work efficiently in online scenarios and tested on GNC applications.

Appendix A

B&B Algorithm and Proofs

A.1 Algorithm

The algorithmic flow of the global optimisation is presented in Algorithm 7. The

Algorithm 7 Algorithmic scheme for B&B over simplexes.

```
Let:
    · $\varepsilon \in \mathbb{R}^+$ a termination threshold
    · $\mathcal{L}_0$ initial list of simplexes covering $\Omega$
    · $L_0 = \min_{\mathbb{S}_0^{(i)} \in \mathcal{L}_0} lb(\mathbb{S}_0^{(i)})$ initial lower bound of $\underline{\hat{\theta}}$
    · $U_0 = \min_{\mathbb{S}_0^{(i)} \in \mathcal{L}_0} ub(\mathbb{S}_0^{(i)})$ initial upper bound of $\underline{\hat{\theta}}$
    · $k = 0$ the iterator
while $U_k - L_k > \varepsilon$
    Branch most promising simplex $lb\left(\mathbb{S}_k^*\right) = L_k$
    $\mathbb{S}_k^* \to {\mathbb{S}_{k+1}^*}^1, {\mathbb{S}_{k+1}^*}^2$
    Construct new list of simplexes
    $\mathcal{L}_{k+1} = (\mathcal{L}_k \setminus \mathbb{S}_k^*) \cup {\mathbb{S}_{k+1}^*}^1 \cup {\mathbb{S}_{k+1}^*}^2$
    Compute new lower and upper bounds of $\underline{\hat{\theta}}$
    $L_{k+1} = \min_{\mathbb{S}_{k+1}^{(j)} \in \mathcal{L}_{k+1}} lb(\mathbb{S}_{k+1}^{(j)})$
    $U_{k+1} = \min_{\mathbb{S}_{k+1}^{(j)} \in \mathcal{L}_{k+1}} ub(\mathbb{S}_{k+1}^{(j)})$
    Discard non-optimal simplexes
    $\mathcal{L}_{k+1} = \mathcal{L}_{k+1} \setminus \mathbb{S}_{k+1}^{(j)}$ if $lb(\mathbb{S}_{k+1}^{(j)}) > U_{k+1}$
    Update iterator
    $k = k + 1$
end while
```

algorithm is a classical B&B starting with an initial list of simplexes $\mathcal{L}_0$ covering the search space Ω. On each simplex $\mathbb{S}_0^{(i)}$, the function minimum is bounded from below $lb(\mathbb{S}_0^{(i)})$ and from above $ub(\mathbb{S}_0^{(i)})$. The minimum of both is kept in the variables L_0 and U_0 respectively, such that $\underline{\hat{\theta}}$ is bounded between them $L_0 \leq \underline{\hat{\theta}} \leq U_0$. Then, starting from $k = 0$, the B&B iterates until the upper and lower bound are closer than the set threshold ε. The most promising simplex is branched into two sub-simplexes in each iteration according to the branching rule employed. The two new simplexes are added to the list $\mathcal{L}_{k+1}$ and the old one removed. The bounding procedure is repeated on the new simplexes, and the values L_{k+1} and U_{k+1} updated. Because $L_{k+1} \leq \underline{\hat{\theta}} \leq U_{k+1}$, the simplexes whose lower bound $lb(\mathbb{S}_{k+1}^{(j)})$ is larger than U_{k+1} can be automatically deemed as non-optimal and therefore discarded. Finally, the iterator is updated, and the loop repeated until convergence.

A.2 B&B proofs

The proof of Lemma 1 is as follows.

Proof. As for the lower bound, from Eq. (3.28) we have $n + 1$ inequalities

$$\hat{\theta}(\boldsymbol{\lambda}) \geq \hat{\theta}(\boldsymbol{\lambda}_j) - L\|\boldsymbol{\lambda} - \boldsymbol{\lambda}_j\| \ , \tag{A.1}$$

for every $\boldsymbol{\lambda}_j$. Therefore, $\hat{\theta}(\boldsymbol{\lambda})$ must be larger than the largest r.h.s.

$$\hat{\theta}(\boldsymbol{\lambda}) \geq \max_j \hat{\theta}(\boldsymbol{\lambda}_j) - L\|\boldsymbol{\lambda} - \boldsymbol{\lambda}_j\| \ . \tag{A.2}$$

This r.h.s. is lb so Eq. (A.2) holds for every point in the simplex. In the argument of the minimum of $\hat{\theta}$ over the simplex, call it $\underline{\boldsymbol{\lambda}}_{\mathbb{S}}$ (which may be non-unique), we have that

$$\underline{\hat{\theta}}_{\mathbb{S}} = \hat{\theta}(\underline{\boldsymbol{\lambda}}_{\mathbb{S}}) \geq lb(\underline{\boldsymbol{\lambda}}_{\mathbb{S}}) \tag{A.3}$$

and by Eq. (3.30) we have that

$$lb(\underline{\boldsymbol{\lambda}}_{\mathbb{S}}) \geq lb^*_{\mathbb{S}} \tag{A.4}$$

resulting in the lower bound inequality in Eq. (3.32)

$$\underline{\hat{\theta}}_{\mathbb{S}} \geq lb^*_{\mathbb{S}} \ . \tag{A.5}$$

For the upper bound of the minimum, we have the trivial relationship

$$\hat{\theta}(\boldsymbol{\lambda}_0) = ub(\mathbb{S}) \geq \underline{\hat{\theta}}_{\mathbb{S}} \tag{A.6}$$

because $\boldsymbol{\lambda}_0$ is in the simplex, proving the upper bound inequality.

$\square$

The proof of Lemma 2 is presented below.

Proof. Let

$$\boldsymbol{\lambda}^*_{lb} = \arg\min_{\boldsymbol{\lambda}\in\mathbb{S}} lb(\boldsymbol{\lambda})$$

be the argument of the minimum of lb over the simplex. Equation $\sigma(\mathbb{S}) \leq \delta$ implies that

$$\|\boldsymbol{\lambda}^*_{lb} - \boldsymbol{\lambda}\| \leq \delta \ .$$

Hence, let us consider the bound difference

$$\begin{aligned} ub(\mathbb{S}) - lb(\mathbb{S}) &= \left|\hat{\theta}(\boldsymbol{\lambda}_0) - lb(\boldsymbol{\lambda}^*_{lb})\right| \\ &= \hat{\theta}(\boldsymbol{\lambda}_0) - \max_j \left[\hat{\theta}(\boldsymbol{\lambda}_j) - L\|\boldsymbol{\lambda}^*_{lb} - \boldsymbol{\lambda}_j\|\right] \ . \end{aligned}$$

For each $\boldsymbol{\lambda}_j$, we have

$$\begin{aligned} |\hat{\theta}(\boldsymbol{\lambda}_0) - \hat{\theta}(\boldsymbol{\lambda}_j) + L\|\boldsymbol{\lambda}^*_{lb} - \boldsymbol{\lambda}_j\|| &\leq |\hat{\theta}(\boldsymbol{\lambda}_0) - \hat{\theta}(\boldsymbol{\lambda}_j)| + L\|\boldsymbol{\lambda}^*_{lb} - \boldsymbol{\lambda}_j\| \\ &\leq L\|\boldsymbol{\lambda}_0 - \boldsymbol{\lambda}_j\| + L\|\boldsymbol{\lambda}^*_{lb} - \boldsymbol{\lambda}_j\| \\ &\leq L\delta + L\delta \ , \end{aligned}$$

where the first inequality comes from the triangle inequality, the second one from the Lipschitz condition and the last one from $\sigma(\mathbb{S}) \leq \delta$. Therefore, also for the $\boldsymbol{\lambda}_j$ argument of the maximum in the lb expression, we have that

$$ub(\mathbb{S}) - lb(\mathbb{S}) = \left|\hat{\theta}(\boldsymbol{\lambda}_0) - lb(\boldsymbol{\lambda}_{lb}^*)\right| \leq 2L\delta \ .$$

Hence, there exists δ such that

$$ub(\mathbb{S}) - lb(\mathbb{S}) \leq 2L\delta \leq \varepsilon \ ,$$

and it can be chosen as

$$0 < \delta \leq \varepsilon/2L \ .$$

$\square$

The proof of Lemma 3 is constructed as follows.

Proof. The diameter of a n-simplex $\mathbb{S}_0^{(i)} \in \mathcal{L}_0$ is reduced at least by a factor $\sqrt{3}/2$ every n splits [138] such that

$$\min_{\mathbb{S}_k^{(j)} \in \mathcal{L}_k} \sigma\left(\mathbb{S}_k^{(j)}\right) \leq \left(\frac{\sqrt{3}}{2}\right)^{\left\lfloor \frac{k}{n} \right\rfloor} \sigma\left(\mathbb{S}_0^{(i)}\right)$$

holds. For increasing k the right hand side goes to zero because $\left(\sqrt{3}/2\right)^{\left\lfloor \frac{k}{n} \right\rfloor}$ is a monotonically decreasing sequence. Therefore the minimum diameter of the simplexes generated by $\mathbb{S}_0^{(i)}$ converges to within the tolerance δ in a finite number of steps. By repeating this argument for each $\mathbb{S}_0^{(i)}$ of the N_0 simplexes in $\mathcal{L}_0$, we get the first part of the lemma.

By focusing again on a single $\mathbb{S}_0^{(i)} \in \mathcal{L}_0$, we can find the number of splittings $K^{(i)}$ to achieve a minimum diameter smaller or equal to a given threshold δ. Using again the bound in [138], we can write

$$\min_{\mathbb{S}_k^{(j)} \in \mathcal{L}_k} \sigma\left(\mathbb{S}_k^{(j)}\right) \leq \left(\frac{\sqrt{3}}{2}\right)^{\left\lfloor \frac{k}{n} \right\rfloor} \sigma\left(\mathbb{S}_0^{(i)}\right) \leq \delta \ .$$

By rearranging the terms and taking the logarithm, one obtains

$$\left\lfloor \frac{k}{n} \right\rfloor \geq \log_{\frac{2}{\sqrt{3}}} \frac{\sigma\left(\mathbb{S}_0^{(i)}\right)}{\delta} \ ,$$

which is equivalent to

$$k \geq n \left\lceil \log_{\frac{2}{\sqrt{3}}} \frac{\sigma\left(\mathbb{S}_0^{(i)}\right)}{\delta} \right\rceil \ .$$

For being conservative, one can employ σ_0 in place of $\sigma\left(\mathbb{S}_0^{(i)}\right)$, that is using the largest initial diameter for every initial simplex $\mathbb{S}_0^{(i)}$ and set an upper bound on the number of simplex splits as

$$K^{(i)} = n \left\lceil \log_{\frac{2}{\sqrt{3}}} \frac{\sigma_0}{\delta} \right\rceil$$

By repeating this argument for each of the N_0 initial simplexes, we obtain a conservative upper bound on the number of LEB splits to obtain the required diameter threshold δ as

$$K = N_0 \, n \left\lceil \log_{\frac{2}{\sqrt{3}}} \frac{\sigma_0}{\delta} \right\rceil \ ,$$

thus concluding the proof. □

The proof of the convergence Theorem 1 is outlined below.

Proof. Let $\mathbb{S}_K^* = \arg\min_{\mathbb{S}_K \in \mathcal{L}_K} \sigma(\mathbb{S}_K)$ be the (possibly non-unique) simplex with smallest diameter at the k-th iteration of the B&B. Let $\mathbb{S}_{K_\varepsilon}^* \in \mathcal{L}_{K_\varepsilon}$ be the parent simplex which was split at the K_ε-th iteration, with $K_\varepsilon < k$, to obtain $\mathbb{S}_k^*$. From Lemma 2, there exists $\delta \in \mathbb{R}^+$ such that

$$\forall \mathbb{S} \subset \mathbb{R}^n, \, \sigma(\mathbb{S}) \leq 2\,\delta \implies ub(\mathbb{S}) - lb(\mathbb{S}) \leq \varepsilon \ .$$

Hence, select $K \in \mathbb{N}$ sufficiently large such that

$$\sigma(\mathbb{S}_K^*) \leq \delta$$

which is granted to happen in a finite number of steps by Lemma 3. Then, because of the selected LEB branching rule, we have a lower bound on the diameter change as

$$\sigma(\mathbb{S}^*_{K_\varepsilon}) \leq 2\,\sigma(\mathbb{S}^*_K) \leq 2\,\delta \ .$$

As shown before, this implies that

$$ub(\mathbb{S}^*_{K_\varepsilon}) - lb(\mathbb{S}^*_{K_\varepsilon}) \leq \varepsilon \ .$$

Since $\mathbb{S}^*_{K_\varepsilon}$ was split, from passage 1 of Algorithm 7, we have that $L_{K_\varepsilon} = lb(\mathbb{S}^*_{K_\varepsilon})$, and by definition $U_{K_\varepsilon} \leq ub(\mathbb{S}^*_{K_\varepsilon})$. Thus, we conclude that

$$U_{K_\varepsilon} - L_{K_\varepsilon} \leq ub(\mathbb{S}^*_{K_\varepsilon}) - lb(\mathbb{S}^*_{K_\varepsilon}) \leq \varepsilon$$

as stated in the first part of the theorem.

Finally, we need to show that $L_{K_\varepsilon} \leq \underline{\hat{\theta}} \leq U_{K_\varepsilon}$. First, we need to show that the simplex containing the optimum has not been discarded in any iteration up to K_ε. Let the argument of the minimum $\underline{\boldsymbol{\lambda}}^* = \arg\min_{\boldsymbol{\lambda}\in\Omega} \hat{\theta}(\boldsymbol{\lambda})$ be attained in the simplex $\mathbb{S}^*_k \in \mathcal{L}_k$ at a generic iteration $k \in \mathbb{N}$, that is the optimal simplex has not been discarded. By Lemma 1, in this simplex we have

$$lb(\mathbb{S}^*_k) \leq \underline{\hat{\theta}} \leq ub(\mathbb{S}^*_k) \ .$$

For all the simplexes in the list $\mathbb{S}_k \in \mathcal{L}_k$, we have by the definition of ub and U_k

$$U_k \leq ub(\mathbb{S}_k) \ .$$

Hence, we have that the simplex containing $\underline{\boldsymbol{\lambda}}^*$ cannot be discarded because

$$lb(\mathbb{S}^*_k) \not> U_k \ .$$

Finally, because $\underline{\boldsymbol{\lambda}}^* \in \mathbb{S}_k^*$, $\underline{\hat{\theta}} \geq L_{K_\varepsilon}$ and $\underline{\hat{\theta}} \leq U_{K_\varepsilon}$ it follows that

$$\begin{aligned} U_{K_\varepsilon} - \underline{\hat{\theta}} \leq U_{K_\varepsilon} - L_{K_\varepsilon} \leq \varepsilon \\ \underline{\hat{\theta}} - L_{K_\varepsilon} \leq U_{K_\varepsilon} - L_{K_\varepsilon} \leq \varepsilon \ . \end{aligned}$$

These two inequalities directly imply

$$L_{K_\varepsilon} \leq \underline{\hat{\theta}} \leq U_{K_\varepsilon} \ ,$$

thus concluding the proof.

□

A.3 Lower bound computation

The function

$$\hat{\theta}(\boldsymbol{\lambda}_j) - L\|\boldsymbol{\lambda} - \boldsymbol{\lambda}_j\|$$

can be seen as the boundary of a hyper-cone C_j with apex in $[\boldsymbol{\lambda}_j, \hat{\theta}(\boldsymbol{\lambda}_j)]$ and half-aperture $\tan^{-1}(1/L)$. The lower bounding function $lb(\boldsymbol{\lambda})$ in a simplex $\mathbb{S}$ (see Eq. (3.29)) results from the maximum of the cones generated from all the vertexes $[\boldsymbol{\lambda}_0, \ldots, \boldsymbol{\lambda}_n]$. By construction, the minimum $lb(\mathbb{S})$ is either attained at the common intersection of these $n+1$ hyper-cones, call it $\boldsymbol{\lambda}_\cap$ if this intersection is inside the simplex, or at the boundary of the simplex, call it $\boldsymbol{\lambda}_\partial$ if the intersection is outside. In the latter case, the common intersection has a lower function value than the point on the boundary, that is $lb(\boldsymbol{\lambda}_\cap) < lb(\boldsymbol{\lambda}_\partial)$, because $lb(\boldsymbol{\lambda}_\cap)$ is the minimum of the convex hull $\mathbb{CH} = conv\{\boldsymbol{\lambda}_0, \ldots, \boldsymbol{\lambda}_n, \boldsymbol{\lambda}_\cap\}$ and $\mathbb{S} \subset \mathbb{CH}$ [191].

The intersection point $\boldsymbol{\lambda}_\cap$ can be found by reformulating Eq. (3.30) by introducing a slack variable $s \in \mathbb{R}$

$$\min_{s} \quad s \tag{A.7a}$$

$$\text{s.t.} \quad \|\boldsymbol{\lambda} - \boldsymbol{\lambda}_j\| \geq (\hat{\theta}(\boldsymbol{\lambda}_j) - s)/L \ , \tag{A.7b}$$

where Eq. (A.7b) holds for $j = 0, \ldots, n$. Problem (A.7) tells us that we need to find for the minimum s such that there exists a feasible point outside the n-dimensional hyper-spheres centered in $\boldsymbol{\lambda}_j$ and with radius $(\hat{\theta}(\boldsymbol{\lambda}_j) - s)/L$. This value is achieved at the common intersection of the boundary S_j of such hyper-spheres for the minimum feasible s.

For a given s, the intersection $\boldsymbol{\lambda}_{\cap_{ij}}$ between two hyper-spheres S_i and S_j lays on an hyperplane perpendicular to the line connecting $\boldsymbol{\lambda}_i$ and $\boldsymbol{\lambda}_j$. Hence, the intersection belongs to the plane parameterised as

$$\boldsymbol{\lambda}_{\cap_{ij}} \in \frac{(\boldsymbol{\lambda}_j - \boldsymbol{\lambda}_i)^T}{\|\boldsymbol{\lambda}_j - \boldsymbol{\lambda}_i\|}\, \boldsymbol{\lambda} - \lambda_{\cap_{0j}} = 0 \tag{A.8}$$

where $\lambda_{\cap_{0j}} \in \mathbb{R}$. This holds between each pair of vertexes of the simplex. To discard redundant pairs, we can write everything with respect to $\boldsymbol{\lambda}_i = \boldsymbol{\lambda}_0$ for $j = 1, \ldots, n$. To find the value of $\lambda_{\cap_{0j}}$, let us suppose for simplicity that $\boldsymbol{\lambda}_0$ is in the origin and that $\boldsymbol{\lambda}_j = [d_{0j}, 0, \ldots, 0]$, with $d_{0j} = \|\boldsymbol{\lambda}_j - \boldsymbol{\lambda}_0\|$. The intersection $\boldsymbol{\lambda}_{\cap_{0j}}$ between S_0 and S_j comes from the solution of the system

$$\begin{aligned} \lambda_0^2 + \lambda_1^2 + \cdots + \lambda_{n-1}^2 &= (\hat{\theta}(\boldsymbol{\lambda}_0) - s)^2/L^2 \\ (\lambda_0 - d_{0j})^2 + \lambda_1^2 + \cdots + \lambda_{n-1}^2 &= (\hat{\theta}(\boldsymbol{\lambda}_1) - s)^2/L^2 \ , \end{aligned}$$

where λ_0, λ_1 and so on are the components of the $\boldsymbol{\lambda}$ vector. By subtracting the equations and re-arranging the terms, we obtain

$$\lambda_{\cap_{0j}} := \lambda_0 = m_{0j}\, s + c_{0j} = \frac{\hat{\theta}(\boldsymbol{\lambda}_j) - \hat{\theta}(\boldsymbol{\lambda}_0)}{L^2 d_{0j}}\, s + \left(\frac{\hat{\theta}(\boldsymbol{\lambda}_0)^2 - \hat{\theta}(\boldsymbol{\lambda}_j)^2}{2L^2 d_{0j}} + \frac{d_{0j}}{2} \right) \ , \tag{A.9}$$

that is, the intersection lies on a hyperplane with coordinate $\lambda_0 = \lambda_{\cap_{0j}}$. This holds true for generic $\boldsymbol{\lambda}_0$ and $\boldsymbol{\lambda}_j$ because we can interpret $\lambda_{\cap_{0j}}$ as the distance of the hyperplane with normal $\boldsymbol{\lambda}_j - \boldsymbol{\lambda}_0$ from the vertex $\boldsymbol{\lambda}_0$. Hence, we can write the system of linear equations $\boldsymbol{\lambda}_{\cap_{ij}}$ needs to satisfy as

$$\begin{bmatrix} \frac{(\boldsymbol{\lambda}_1-\boldsymbol{\lambda}_i)^T}{\|\boldsymbol{\lambda}_1-\boldsymbol{\lambda}_i\|} \\ \frac{(\boldsymbol{\lambda}_2-\boldsymbol{\lambda}_i)^T}{\|\boldsymbol{\lambda}_2-\boldsymbol{\lambda}_i\|} \\ \vdots \\ \frac{(\boldsymbol{\lambda}_n-\boldsymbol{\lambda}_i)^T}{\|\boldsymbol{\lambda}_n-\boldsymbol{\lambda}_i\|} \end{bmatrix} \boldsymbol{\lambda} = \begin{bmatrix} m_{01} \\ m_{02} \\ \vdots \\ m_{0n} \end{bmatrix} s + \begin{bmatrix} c_{01} \\ c_{02} \\ \vdots \\ c_{0n} \end{bmatrix} \tag{A.10}$$

which is a classical linear equation in the form $\mathbf{dV}\boldsymbol{\lambda} = \mathbf{m}s + \mathbf{c}$. The hyperplane intersection is linear in s. Indeed, it is worth underlining that the hyperplanes' intersection generally differs from the spheres' intersection because Eq. (A.10) is a set of necessary but not sufficient conditions for $\boldsymbol{\lambda}_{\cap_{ij}}$, and they coincide only for a specific value of s. Hence, to find a unique value, we need to add an additional constraint in s. This can be solved by requiring that the hyperplanes' intersection lays on the hyper-sphere centred in $\boldsymbol{\lambda}_0$ as

$$\begin{cases} \quad \mathbf{dV}\boldsymbol{\lambda} & = \mathbf{m}s + \mathbf{c} \\ \|\boldsymbol{\lambda} - \boldsymbol{\lambda}_0\|^2 & = (\hat{\theta}(\boldsymbol{\lambda}_0) - s)^2/L^2 \ . \end{cases} \tag{A.11}$$

This system has a closed-form solution. Again, without loss of generality, let us suppose that $\boldsymbol{\lambda}_0$ coincides with the origin. Then, let $\boldsymbol{\lambda} = \widetilde{\mathbf{m}}s + \widetilde{\mathbf{c}}$ with $\widetilde{\mathbf{m}} = \mathbf{dV}^{-1}\mathbf{m}$ and $\widetilde{\mathbf{c}} = \mathbf{dV}^{-1}\mathbf{c}$. Plug $\boldsymbol{\lambda}$ into the quadratic equation to get

$$\sum_{j=1}^{n} (\widetilde{m}_{0j}s + \widetilde{c}_{0j})^2 = (\hat{\theta}(\boldsymbol{\lambda}_0) - s)^2/L^2 \ , \tag{A.12}$$

that is a simple second-order equation in s that can be solved analytically. The smallest root $s_\cap$ out of the two should be selected because the largest one corresponds to the upper half of the double cone. The spheres' intersection can then be computed as $\boldsymbol{\lambda}_\cap = \widetilde{\mathbf{m}}\, s_\cap + \widetilde{\mathbf{c}}$. By construction, the sought lower bound is

$$lb(\mathbb{S}) = lb(\boldsymbol{\lambda}_\cap) = s_\cap \ . \tag{A.13}$$

Appendix B

B-plane coordinate tranformation

The B-plane is the plane passing through the target body centre of mass and perpendicular to the hyperbola incoming asymptote [187, 192, 159]. It is defined by three unit vectors: $\hat{\mathbf{s}}$ is the unit vector parallel to the relative incoming asymptote with positive direction aligned to the incoming asymptotic velocity; $\hat{\mathbf{t}}$ is the cross product between $\hat{\mathbf{s}}$ and the normal to the ecliptic, that is $\hat{\mathbf{z}}$ vector in the inertial frame; $\hat{\mathbf{r}}$ is the cross product between $\hat{\mathbf{s}}$ and $\hat{\mathbf{t}}$.

The transformation from inertial to B-plane coordinates is

$$\mathbf{B} = [\,\mathbf{b},\, \mathrm{TCA}] = \mathcal{B}(\mathbf{x}(t)) \ ,$$

where the *aiming point* $\mathbf{b} = (\mathbf{b} \cdot \hat{\mathbf{t}})\,\hat{\mathbf{t}} + (\mathbf{b} \cdot \hat{\mathbf{r}})\,\hat{\mathbf{r}}$ is the target-centred vector to the intersection point between the incoming asymptote and the B-plane (it would be the closest approach point if the target body were massless), whereas the hyperbolic TCA is the time interval that the spacecraft needs to travel from $\mathbf{x}(t)$ to the flyby closest approach along the relative hyperbolic orbit.

The transformation $\mathcal{B}$ therefore returns the two components, $\mathbf{b} \cdot \hat{\mathbf{t}}$ and $\mathbf{b} \cdot \hat{\mathbf{r}}$, and the hyperbolic TCA for a given inertial state $\mathbf{x}$ in the proximity of the flyby. First, the inertial state with respect to the central body is converted to the inertial state with respect to the target body to flyby by a simple translation. Hence, let $\mathbf{r}$ and $\mathbf{v}$ be the relative position and velocity of the spacecraft. Then, following [192], the unit vectors are first derived by computing the unit normal vector

$$\hat{\mathbf{h}} = \mathbf{r} \times \mathbf{v} \,/\, \|\mathbf{r} \times \mathbf{v}\| \ ,$$

the eccentricity vector

$$\mathbf{e} = \Big(\frac{v^2}{\mu} - \frac{1}{r}\Big)\mathbf{r} - \frac{(\mathbf{r} \cdot \mathbf{v})}{\mu}\mathbf{v} \ ,$$

and the asymptote angle

$$\beta = \cos^{-1}(1/e) \ .$$

Hence, $\hat{\mathbf{s}}$ can be written as

$$\hat{\mathbf{s}} = \cos\beta \frac{\mathbf{e}}{\|\mathbf{e}\|} + \sin\beta \frac{\hat{\mathbf{h}} \times \mathbf{e}}{\|\hat{\mathbf{h}} \times \mathbf{e}\|} \ ,$$

and consequently

$$\hat{\mathbf{t}} = \hat{\mathbf{s}} \times \hat{\mathbf{z}}$$
$$\hat{\mathbf{r}} = \hat{\mathbf{s}} \times \hat{\mathbf{t}} \ .$$

Hence, from the hyperbolic semi-major axis

$$a = -\frac{\mu}{2}\Big/\Big(\frac{v^2}{2} - \frac{\mu}{2}\Big) \ ,$$

the $\mathbf{b}$ vector magnitude is computed as

$$\|\mathbf{b}\| = \frac{\|\mathbf{h}\|}{v_\infty} = -a\sqrt{e^2 - 1} \ .$$

Begin the aiming point within the B-plane (perpendicular to $\hat{\mathbf{s}}$) and within the relative orbital motion (perpendicular to $\hat{\mathbf{h}}$), the $\mathbf{b}$ vector can be finally written as

$$\mathbf{b} = \|\mathbf{b}\| \left(\hat{\mathbf{s}} \times \hat{\mathbf{h}}\right) \ ,$$

and its projections onto $\hat{\mathbf{t}}$ and $\hat{\mathbf{r}}$ can be found by a scalar product. The hyperbolic TCA is computed by the relation between hyperbolic anomaly H and time (see Equation (8.23-1) of [159])

$$\mathrm{TCA} = \frac{e \sinh H - H}{\sqrt{\mu/a^3}} \ ,$$

where H is derived by its definition

$$r = a(1 - e \cosh H) \ .$$

As the B-plane change of coordinates involves only algebric passages, the Jacobian of this transformation with respect to the inertial state

$$J_{\mathcal{B}} = \frac{\partial \mathbf{B}}{\partial \mathbf{x}}$$

can computed analytically. Specifically, the derivative of the equations above with respect to the parameters appearing on the right-hand side are derived by symbolic differentiation, and the Jacobian is constructed by chain rule.

Bibliography

[1] H. Helvajian and S. Janson, *Small Satellites: Past, Present, and Future.* Aerospace Press El Segundo, CA, 2009.

[2] J. R. Kopacz, R. Herschitz, and J. Roney, "Small satellites an overview and assessment," *Acta Astronautica*, vol. 170, pp. 93–105, 2020.

[3] A. Toorian, K. Diaz, and S. Lee, "The cubesat approach to space access," in *2008 IEEE Aerospace Conference*, 2008, pp. 1–14.

[4] J. C. McDowell, "The low earth orbit satellite population and impacts of the SpaceX starlink constellation," *The Astrophysical Journal*, vol. 892, no. 2, apr 2020.

[5] M. Safyan, *Planet's Dove Satellite Constellation.* Cham: Springer International Publishing, 2020, pp. 1–17.

[6] A. Rossi, E. Alessi, G. Valsecchi, H. Lewis, J. Radtke, C. Bombardelli, and B. B. Virgili, "A quantitative evaluation of the environmental impact of the mega constellations," in *Proceedings of the 7th European Conference on Space Debris*, T. Flohrer and F. Schmitz, Eds., vol. 7. The European Space Agency (ESA), 2017.

[7] H. Lewis, J. Radtke, A. Rossi, J. Beck, M. Oswald, P. Anderson, B. B. Virgili, and H. Krag, "Sensitivity of the space debris environment to large constellations and small satellites," *Journal of the British Interplanetary Society*, vol. 70, no. 2-4, pp. 105–117, September 2017.

[8] J.-C. Liou and N. Johnson, "Instability of the present leo satellite populations," *Advances in Space Research*, vol. 41, no. 7, pp. 1046–1053, 2008.

[9] D. J. Kessler, N. L. Johnson, J. Liou, and M. Matney, "The kessler syndrome: implications to future space operations," *Advances in the Astronautical Sciences*, vol. 137, no. 8, p. 2010, 2010.

[10] Y. Tsuda, T. Saiki, F. Terui, S. Nakazawa, M. Yoshikawa, S.-i. Watanabe, and H. P. Team, "Hayabusa2 mission status: Landing, roving and cratering on asteroid ryugu," *Acta Astronautica*, vol. 171, pp. 42–54, 2020.

[11] D. R. Wibben, A. Levine, S. Rieger, J. V. McAdams, P. G. Antreasian, J. M. Leonard, M. C. Moreau, and D. S. Lauretta, "Osiris-rex frozen orbit design and flight experience," in *AAS/AIAA Astrodynamics Specialist Conference, 2019*. Univelt Inc., 2020, pp. 2959–2971.

[12] J. Schoolcraft, A. Klesh, and T. Werne, *MarCO: Interplanetary Mission Development on a CubeSat Scale.* Cham: Springer International Publishing, 2017, pp. 221–231.

[13] R. Funase, H. Koizumi, S. Nakasuka, Y. Kawakatsu, Y. Fukushima, A. Tomiki, Y. Kobayashi, J. Nakatsuka, M. Mita, D. Kobayashi *et al.*, "50kg-class deep space exploration technology demonstration micro-spacecraft procyon," in *28th Annual AIAA/USU Conference on Small Satellites*, 2014, pp. 4435–4435.

[14] T. R. Lockett, J. Castillo-Rogez, L. Johnson, J. Matus, J. Lightholder, A. Marinan, and A. Few, "Near-earth asteroid scout flight mission," *IEEE Aerospace and Electronic Systems Magazine*, vol. 35, no. 3, pp. 20–29, 2020.

[15] R. Walker, K. Detlef, and B. Chiara, "Miniaturised asteroid remote geophysical observer (m-argo): a stand-alone deep space cubesat system for low-cost science and exploration missions," May 2017, presentation at 6th Interplanetary CubeSat Workshop, Cambridge, UK.

[16] S. Speretta, A. Cervone, P. Sundaramoorthy, R. Noomen, S. Mestry, A. Cipriano, F. Topputo, J. Biggs, P. Di Lizia, M. Massari, K. V. Mani, D. A. Dei Tos, S. Ceccherini, V. Franzese, A. Ivanov, D. Labate, L. Tommasi, A. Jochemsen, J. Gailis, R. Furfaro, V. Reddy, J. Vennekens, and R. Walker, *LUMIO: An Autonomous CubeSat for Lunar Exploration.* Cham: Springer International Publishing, 2019, pp. 103–134.

[17] C. Greco, M. Di Carlo, L. Walker, and M. Vasile, “Analysis of NEOs Reachability with Nano-Satellites and Low-Thrust Propulsion,” in *4S Symposium*, Sorrento, Italy, 2018.

[18] L. Walker, M. Di Carlo, C. Greco, M. Vasile, and M. Warden, “A mission concept for the low-cost large-scale exploration and characterisation of near earth objects,” *Advances in Space Research*, 2020.

[19] C. Snodgrass and G. H. Jones, “The european space agency’s comet interceptor lies in wait,” *Nature Communications*, vol. 10, no. 1, pp. 1–4, 2019.

[20] P. Machuca, J. Sánchez, J. Masdemont, and G. Gómez, “High-fidelity trajectory design to flyby near-earth asteroids using cubesats,” *Acta Astronautica*, vol. 167, pp. 146 – 163, 2020.

[21] S. Campagnola, N. Ozaki, Y. Sugimoto, C. H. Yam, H. Chen, Y. Kawabata, S. Ogura, B. Sarli, Y. Kawakatsu, R. Funase, and S. Nakasuka, “Low-thrust trajectory design and operations of procyon, the first deep-space micro-spacecraft,” in *24th International Symposium on Space Flight Dynamics*, Munich, Germany, 2015.

[22] S. Campagnola, J. Hernando-Ayuso, K. Kakihara, Y. Kawabata, T. Chikazawa, R. Funase, N. Ozaki, N. Baresi, T. Hashimoto, Y. Kawakatsu, T. Ikenaga, K. Oguri, and K. Oshima, “Mission analysis for the EM-1 CubeSats EQUULEUS and OMOTENASHI,” *IEEE Aerospace and Electronic Systems Magazine*, vol. 34, no. 4, 2019.

[23] J. Guo, L. Monas, and E. Gill, “Statistical analysis and modelling of small satellite reliability,” *Acta Astronautica*, vol. 98, pp. 97–110, 2014.

[24] B. K. Malphrus, A. Freeman, R. Staehle, A. T. Klesh, and R. Walker, “Interplanetary cubesat missions,” in *Cubesat Handbook*, C. Cappelletti, S. Battistini, and B. K. Malphrus, Eds. Academic Press, 2021, pp. 85–121.

[25] S. Russell and P. Norvig, *Artificial intelligence: a modern approach*, 4th ed. Pearson, 2020.

[26] D. Mehrholz, L. Leushacke, W. Flury, R. Jehn, H. Klinkrad, and M. Landgraf, “Detecting, tracking and imaging space debris,” *ESA Bulletin(0376-4265)*, no. 109, pp. 128–134, 2002.

[27] J. Xi, Y. Xiang, O. K. Ersoy, M. Cong, X. Wei, and J. Gu, “Space debris detection using feature learning of candidate regions in optical image sequences,” *IEEE Access*, vol. 8, pp. 150 864–150 877, 2020.

[28] D. Simon, *Optimal state estimation: Kalman, H infinity, and nonlinear approaches.* John Wiley & Sons, 2006.

[29] R. Weiss, “An approach to Bayesian sensitivity analysis,” *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 4, pp. 739–750, 1996.

[30] S. Sarkka, *Bayesian filtering and smoothing*, 1st ed. Cambridge University Press, New York, 2013.

[31] N. J. Gordon, D. J. Salmond, and A. F. Smith, “Novel approach to nonlinear/non-Gaussian Bayesian state estimation,” *IEE proceedings F (radar and signal processing)*, vol. 140 (2), pp. 107–113, 1993.

[32] A. G. Wills, J. Hendriks, C. Renton, and B. Ninness, “A Bayesian filtering algorithm for Gaussian mixture models,” *arXiv preprint arXiv:1705.05495*, 2017.

[33] W. Li and Y. Jia, “H-infinity filtering for a class of nonlinear discrete-time systems based on unscented transform,” *Signal Processing*, vol. 90, no. 12, pp. 3301–3307, 2010.

[34] E. Delande, J. Houssineau, and M. Jah, “Physics and human-based information fusion for improved resident space object tracking,” *Advances in Space Research*, vol. 62, no. 7, pp. 1800–1812, 2018.

[35] E. Delande, M. Jah, and B. Jones, “A new representation of uncertainty for collision assessment,” in *AAS/AIAA Spaceflight Mechanics 2019*, 2019, http://www.univelt.com/book=7406.

[36] H. Cai, I. Hussein, and M. Jah, “Possibilistic admissible region using outer probability measure theory,” *Acta Astronautica*, vol. 177, pp. 246–257, 2020.

[37] W.-S. Wang and M. Orshansky, “Robust estimation of parametric yield under limited descriptions of uncertainty,” in *Proceedings of the 2006 IEEE/ACM international conference on Computer-aided design*, 2006, pp. 884–890.

[38] C. Masreliez and R. Martin, “Robust Bayesian estimation for the linear model and robustifying the Kalman filter,” *IEEE transactions on Automatic Control*, vol. 22, no. 3, pp. 361–371, 1977.

[39] D. R. Morrell and W. C. Stirling, “Set-values filtering and smoothing,” *IEEE transactions on systems, man, and cybernetics*, vol. 21, no. 1, pp. 184–193, 1991.

[40] P. Smets and B. Ristic, “Kalman filter and joint tracking and classification in the tbm framework,” in *Proceedings of the Seventh International Conference on Information Fusion*, vol. 1. Citeseer, 2004, pp. 46–53.

[41] B. Noack, V. Klumpp, D. Brunn, and U. D. Hanebeck, “Nonlinear Bayesian estimation with convex sets of probability densities,” in *2008 11th International Conference on Information Fusion.* IEEE, 2008, pp. 1–8.

[42] A. Benavoli, M. Zaffalon, and E. Miranda, “Robust filtering through coherent lower previsions,” *IEEE Transactions on Automatic Control*, vol. 56, no. 7, pp. 1567–1581, 2010.

[43] P. N. Valerino, B. Buffington, K. Criddle, Y. Hahn, R. Ionasescu, J. A. Kangas, T. Martin-Mur, R. B. Roncoli, and J. A. Sims, “Preliminary maneuver analy-

sis for the europa clipper multiple-flyby mission," in *AIAA/AAS Astrodynamics Specialist Conference*, 2014.

[44] S. Nandi, J. Kangas, P. N. Valerino, B. Buffington, R. Ionasescu, and D. Boone, "Initial navigation analysis for the europa multiple flyby mission concept," in *26th AAS/AIAA Space Flight Mechanics Meeting*, 2016.

[45] A. Gelb, *Applied optimal estimation.* MIT press, 1974, chap. 7.

[46] N. B. Stastny and D. K. Geller, "Autonomous optical navigation at jupiter: A linear covariance analysis," *Journal of Spacecraft and Rockets*, vol. 45, no. 2, pp. 290–298, 2008.

[47] D. K. Geller, "Linear covariance techniques for orbital rendezvous analysis and autonomous onboard mission planning," *Journal of Guidance, Control, and Dynamics*, vol. 29, no. 6, pp. 1404–1414, 2006.

[48] B. Clement and M. Johnston, "The deep space network scheduling problem," in *Seventeenth Annual Conference on Innovative Applications of Artificial Intelligence*, 01 2005, pp. 1514–1520.

[49] A. Kleinschrodt, A. Freimann, S. Christall, M. Lankl, and K. Schilling, "Advances in modulation and communication protocols for small satellite ground stations," in *Proceedings of the 68th International Astronautical Congress*, 2017, pp. 2–s2.

[50] K. M. Riesing, "Portable optical ground stations for satellite communication," Ph.D. dissertation, Massachusetts Institute of Technology, 2018.

[51] F. Z. Ali, S. N. M. Rahim, and M. H. Jusoh, "Amateur satellite ground station: Troubleshooting and lesson learned," *Journal of Physics: Conference Series*, vol. 1768, no. 1, p. 012013, jan 2021.

[52] P. Soma, S. Venkateswarlu, S. Santhalakshmi, T. Bagchi, and S. Kumar, "Multisatellite scheduling using genetic algorithms," in *Space OPS 2004 Conference*, 2004, p. 515.

[53] J. Sun and F. Xhafa, “A genetic algorithm for ground station scheduling,” in *2011 International Conference on Complex, Intelligent, and Software Intensive Systems.* IEEE, 2011, pp. 138–145.

[54] F. Xhafa, X. Herrero, A. Barolli, and M. Takizawa, “A simulated annealing algorithm for ground station scheduling problem,” in *2013 16th International Conference on Network-Based Information Systems.* IEEE, 2013, pp. 24–30.

[55] S. Spangelo, J. Cutler, K. Gilson, and A. Cohn, “Optimization-based scheduling for the single-satellite, multi-ground station communication problem,” *Computers & Operations Research*, vol. 57, pp. 1–16, 2015.

[56] S. Damiani, H. Dreihahn, J. Noll, M. Nizette, and G. P. Calzolari, “A planning and scheduling system to allocate esa ground station network services,” in *The Int’l Conference on Automated Planning and Scheduling.* Citeseer, 2007.

[57] M. Schmidt, M. Rybysc, and K. Schilling, “A scheduling system for small ground station networks,” in *SpaceOps 2008 Conference*, 2008.

[58] A. Kleinschrodt, N. Reed, and K. Schilling, “A comparison of scheduling algorithms for low cost ground station networks,” in *67st international astronautical congress. Guadalajara, Mexico*, 2016, pp. 1–15.

[59] T. J. Muelhaupt, M. E. Sorge, J. Morin, and R. S. Wilson, “Space traffic management in the new space era,” *Journal of Space Safety Engineering*, vol. 6, no. 2, pp. 80–87, 2019.

[60] S. Nag, D. Murakami, M. Lifson, and P. Kopardekar, “System autonomy for space traffic management,” in *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*, 2018, pp. 1–10.

[61] T. Flohrer, H. Krag, K. Merz, and S. Lemmens, “Cream-esa’s proposal for collision risk estimation and automated mitigation,” in *Proceedings of the Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS)*, 2019.

[62] A. Mashiku, C. Frueh, N. Memarsadeghi, E. Gizzi, M. Zielinski, and A. Burton, "Predicting satellite close approaches using statistical parameters in the context of artificial intelligence," in *AAS/AIAA Astrodynamics Specialist Conference*, 2019.

[63] The Consultative Committee for Space Data Systems, "Recommendation for space data system standards: Conjunction Data Message," CCSDS 133.0-B-1. Blue Book, Tech. Rep., 2013.

[64] A. B. Poore, J. M. Aristoff, J. T. Horwood, R. Armellin, W. T. Cerven, Y. Cheng, C. M. Cox, R. S. Erwin, J. H. Frisbee, M. D. Hejduk *et al.*, "Covariance and uncertainty realism in space surveillance and tracking," Numerica Corporation Fort Collins United States, Tech. Rep., 2016.

[65] J. R. Carpenter, "Covariance realism is not enough," in *AAS/AIAA Astrodynamics Specialist Conference*, 2019.

[66] M. S. Balch, R. Martin, and S. Ferson, "Satellite conjunction analysis and the false confidence theorem," *Proceedings of the Royal Society A*, vol. 475, no. 2227, p. 20180565, 2019.

[67] M. D. Rayman and S. N. Williams, "Design of the first interplanetary solar electric propulsion mission," *Journal of Spacecraft and Rockets*, vol. 3, pp. 589–595, 2002.

[68] F. E. Laipert and J. M. Longuski, "Automated Missed-Thrust Propellant Margin Analysis for Low-Thrust Trajectories," *Journal of Spacecraft and Rockets*, vol. 52, pp. 1135–1143, 2015.

[69] D. Grebow, G. J. Whiffen, D. Han, and B. Kennedy, "Dawn safing approach to Ceres re-design," in *AIAA/AAS Astrodynamics Specialist Conference*, 2016.

[70] H. Kuninaka, K. Nishiyama, Y. Shimizu, I. Funaki, H. Koizumi, S. Hosoda, and D. Nakata, "Hayabusa asteroid explorer powered by ion engines on the way to earth," in *Proceedings of the 31st International Electric Propulsion Conference*, 2009.

[71] H. Kuninaka and J. I. Kawaguchi, “Lessons learned from round trip of Hayabusa asteroid explorer in deep space,” in *2011 Aerospace Conference.* IEEE, 2011, pp. 1–8.

[72] A. Richards and J. How, “Robust stable model predictive control with constraint tightening,” in *American Control Conference 2006 IEEE*, 2006.

[73] E. D. Gustafson, “Stochastic optimal control of spacecraft,” Ph.D. dissertation, The University of Michigan, 2010.

[74] P. Di Lizia, R. Armellin, F. Bernelli-Zazzera, and M. Berz, “High order optimal control of space trajectories with uncertain boundary conditions,” *Acta Astronautica*, vol. 93, pp. 217–229, 2014.

[75] P. Di Lizia, R. Armellin, A. Morselli, and F. Bernelli-Zazzera, “High order optimal feedback control of space trajectories with bounded control,” *Acta Astronautica*, vol. 94, no. 1, pp. 383–394, 2014.

[76] J. T. Olympio and C. H. Yam, “Deterministic method for space trajectory design with mission margin constraints,” in *61st International Astronautical Congress*, Prague, Czech Republic, 2010.

[77] J. T. Olympio, “Designing robust low-thrust interplanetary trajectories subject to one temporary engine failure,” in *20th AAS/AIAA Space Flight Meeting*, San Diego, CA, US, 2010.

[78] N. Ozaki, S. Campagnola, R. Funase, and C. H. Yam, “Stochastic differential dynamic programming with unscented transform for low-thrust trajectory design,” *Journal of Guidance, Control, and Dynamics*, vol. 41, pp. 377–387, 2018.

[79] M. Vasile, “Robustness optimisation of aerocapture trajectory design using a hybrid co-evolutionary approach,” in *18th International Symposium on Space Flight Dynamics*, Munich, Germany, 2004, pp. 1–6.

[80] F. Zuiani, M. Vasile, and A. Gibbings, “Evidence-based robust design of deflection

actions for near earth objects," *Celestial Mechanics and Dynamical Astronomy*, vol. 114, pp. 107–136, 2012.

[81] M. Di Carlo, M. Vasile, C. Greco, and R. Epenoy, "Robust optimisation of low-thrust interplanetary transfers using evidence theory," in *29th AAS/AIAA Space Flight Mechanics Meeting*, Ka'anapali, Hawaii, US, 2019, pp. 339–358, URL strathprints:67543.

[82] S. Graça Marto, M. Vasile, and R. Epenoy, "Multi-objective robust trajectory optimisation under epistemic uncertainty and imprecision," in *Proceedings of the 70th International Astronautical Congress, IAC19*, Washington, DC, US, 2019, URL strathprints:70454.

[83] K. Oguri and J. W. McMahon, "Robust spacecraft guidance around small bodies under uncertainty: Stochastic optimal control approach," *Journal of Guidance, Control, and Dynamics*, pp. 1–19, 2021.

[84] C. Greco and M. Vasile, "Robust Bayesian particle filter for space object tracking under severe uncertainty," *Journal of Guidance, Control, and Dynamics*, Manuscript submitted for publication.

[85] C. Greco, S. Campagnola, and M. Vasile, "Robust space trajectory design using belief optimal control," *Journal of Guidance, Control, and Dynamics*, Manuscript submitted for publication.

[86] L. Gentile, C. Greco, E. Minisci, T. Bartz-Beielstein, and M. Vasile, "Satellite tracking with constrained budget via structured-chromosome genetic algorithms," *Optimization and Engineering*, Manuscript submitted for publication.

[87] C. Greco, M. Di Carlo, M. Vasile, and R. Epenoy, "Direct multiple shooting transcription with polynomial algebra for optimal control problems under uncertainty," *Acta Astronautica*, vol. 170, pp. 224–234, 2020.

[88] C. Greco and M. Vasile, "Fundamentals of filtering," in *Optimization Under Un-*

certainty with Applications to Aerospace Engineering, M. Vasile, Ed. Cham: Springer International Publishing, 2021, pp. 181–222.

[89] A. Riccardi, E. Minisci, K. Akartunali, C. Greco, N. Rutledge, A. Kershaw, and A. Hashim, "Introduction to optimisation," in *Optimization Under Uncertainty with Applications to Aerospace Engineering*, M. Vasile, Ed. Cham: Springer International Publishing, 2021, pp. 223–268.

[90] L. Gentile, C. Greco, E. Minisci, T. Bartz-Beielstein, and M. Vasile, "Structured-chromosome ga optimisation for satellite tracking," in *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, ser. GECCO '19. New York, NY, USA: ACM, 2019, pp. 1955–1963.

[91] C. Greco, L. Gentile, G. Filippi, E. Minisci, M. Vasile, and T. Bartz-Beielstein, "Autonomous generation of observation schedules for tracking satellites with structured-chromosome ga optimisation," in *2019 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2019, pp. 497–505.

[92] C. Greco, L. Sánchez Fernández-Mellado, M. Manzi, and M. Vasile, "A robust Bayesian agent for optimal collision avoidance manoeuvre planning," in *8th European Conference on Space Debris*, 2021.

[93] C. Greco and M. Vasile, "Closing the loop between mission design and navigation analysis," in *International Astronautical Congress, The Cyberspace Edition*, 2020.

[94] C. Greco and M. Vasile, "Robust particle filter for space navigation under epistemic uncertainty," in *UQOP2020*, 2020.

[95] G. Acciarini, C. Greco, and M. Vasile, "On the solution of the fokker-planck equation without diffusion for uncertainty propagation in orbital dynamics," in *2020 AAS/AIAA Astrodynamics Specialist Conference*, 2020.

[96] 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.

[97] L. Gentile, C. Greco, E. Minisci, T. Bartz-Beielstein, and M. Vasile, "An optimization approach for designing optimal tracking campaigns for low-resources deep-space missions," in *Proceedings of the 70th International Astronautical Congress, IAC19*, Washington, DC, US, 2019.

[98] C. Greco, L. Gentile, M. Vasile, E. Minisci, and T. Bartz-Beielstein, "Robust particle filter for space objects tracking under severe uncertainty," in *2019 AAS/AIAA Astrodynamics Specialist Conference*, Portland, ME, US, 2019, URL strathprints:70566.

[99] L. Walker, C. Greco, M. Di Carlo, A. Wilson, L. Ricciardi, A. Berquand, and M. Vasile, "Nanospacecraft exploration of asteroids by collision and flyby reconnaissance," in *Low-Cost Planetary Missions Conference*, 2019.

[100] C. Greco, M. Di Carlo, M. Vasile, and R. Epenoy, "An intrusive polynomial algebra multiple shooting approach to the solution of optimal control problems," in *Proceedings of the 69th International Astronautical Congress, IAC18*, Bremen, Germany, 2018, URL strathprints:65918.

[101] T. Augustin, F. P. A. Coolen, G. De Cooman, and M. Troffaes, *Introduction to Imprecise Probabilities.* John Wiley & Sons, 2014.

[102] E. G. Tabak and E. Vanden-Eijnden, "Density estimation by dual ascent of the log-likelihood," *Communications in Mathematical Sciences*, vol. 8, no. 1, pp. 217–233, 2010.

[103] H. Risken, *The Fokker-Planck Equation: Methods of Solution and Applications*, 2nd ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 1996.

[104] S. Challa and Y. Bar-Shalom, "Nonlinear filter design using fokker-planck-kolmogorov probability density evolutions," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 36, no. 1, pp. 309–315, 2000.

[105] T. P. Sapsis and G. A. Athanassoulis, "New partial differential equations governing the joint, response–excitation, probability distributions of nonlinear systems,

under general stochastic excitation," *Probabilistic Engineering Mechanics*, vol. 23, no. 2, pp. 289–306, 2008.

[106] A. T. Bharucha-Reid, *Elements of the Theory of Markov Processes and their Applications.* Courier Corporation, 1997.

[107] C. Yanez, M. Gupta, V. Morand, and J. Dolado, "On the gaussianity validity time for orbital uncertainty propagation," in *ESA NEO and Debris Detection Conference, Darmstadt*, 2019.

[108] M. DeGroot and M. Schervish, *Probability and Statistics*, 4th ed. Addison-Wesley, 2012.

[109] A. Riccardi, C. Tardioli, and M. Vasile, "An intrusive approach to uncertainty propagation in orbital mechanics based on Tchebycheff polynomial algebra," *Advances in Astronautical Sciences*, pp. 707–722, 2015. [Online]. Available: https://strathprints.strath.ac.uk/60560/

[110] C. A. Ortega, A. Riccardi, M. Vasile, and C. Tardioli, "SMART-UQ: Uncertainty Quantification Toolbox for Generalised Intrusive and non Intrusive Polynomial Algebra," in *6th International Conference on Astrodynamics Tools and Techniques*, 2016. [Online]. Available: https://strathprints.strath.ac.uk/58920/

[111] M. Vasile, C. A. Ortega, and A. Riccardi, "Set propagation in dynamical systems with generalised polynomial algebra and its computational complexity," *Communications in Nonlinear Science and Numerical Simulation*, vol. 75, pp. 22–49, 2019.

[112] M. Berz, "From taylor series to taylor models," in *AIP Conference Proceedings CONF-961208*, vol. 405, no. 1. American Institute of Physics, 1997, pp. 1–23.

[113] P. Di Lizia, R. Armellin, and M. Lavagna, "Application of high order expansions of two-point boundary value problems to astrodynamics," *Celestial Mechanics and Dynamical Astronomy*, vol. 102, no. 4, pp. 355–375, 2008.

[114] 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*, vol. 107, no. 4, pp. 451–470, 2010.

[115] M. Valli, R. Armellin, P. Di Lizia, and M. R. Lavagna, “Nonlinear mapping of uncertainties in celestial mechanics,” *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 1, pp. 48–63, 2013.

[116] P. Di Lizia, R. Armellin, E. Finzi, and M. Berz, “High-order robust guidance of interplanetary trajectories based on differential algebra,” *Journal of Aerospace Engineering, Sciences and Applications*, vol. 1, pp. 43–57, 01 2008.

[117] M. Valli, R. Armellin, P. Di Lizia, and M. R. Lavagna, “Nonlinear filtering methods for spacecraft navigation based on differential algebra,” *Acta Astronautica*, vol. 94, no. 1, pp. 363–374, 2014.

[118] A. Morselli, R. Armellin, P. Di Lizia, and F. Bernelli Zazzera, “A high order method for orbital conjunctions analysis: Sensitivity to initial uncertainties,” *Advances in Space Research*, vol. 53, no. 3, pp. 490–508, 2014.

[119] A. Morselli, R. Armellin, P. Di Lizia, and F. Bernelli Zazzera, “A high order method for orbital conjunctions analysis: Monte carlo collision probability computation,” *Advances in Space Research*, vol. 55, no. 1, pp. 311–333, 2015.

[120] N. Brisebarre and M. Joldeş, “Chebyshev interpolation polynomial-based tools for rigorous computing,” in *Proceedings of the 2010 International Symposium on Symbolic and Algebraic Computation*, ser. ISSAC ’10. New York, NY, USA: Association for Computing Machinery, 2010, p. 147–154.

[121] L. N. Trefethen, *Approximation Theory and Approximation Practice, Extended Edition.* SIAM, 2019.

[122] C. A. Ortega, R. Serra, A. Riccardi, and M. Vasile, “De-orbiting and re-entry analysis with generalised intrusive polynomial expansions,” in *67th International Astronautical Congress*, 2016, URL strathprints:60596.

[123] S. P. Hughes, “General Mission Analysis Tool (GMAT),” in *International Conference on Astrodynamics Tools and Techniques (ICATT) Darmstadt*, 2016.

[124] J. Bäck, F. Nobile, L. Tamellini, and R. Tempone, “Stochastic spectral Galerkin and collocation methods for PDEs with random coefficients: a numerical comparison,” in *Spectral and High Order Methods for Partial Differential Equations*, ser. Lecture Notes in Computational Science and Engineering, J. Hesthaven and E. Ronquist, Eds. Springer, 2011, vol. 76, pp. 43–62.

[125] M. Bessa, R. Bostanabad, Z. Liu, A. Hu, D. W. Apley, C. Brinson, W. Chen, and W. Liu, “A framework for data-driven analysis of materials under uncertainty: Countering the curse of dimensionality,” *Computer Methods in Applied Mechanics and Engineering*, vol. 320, pp. 633–667, 2017.

[126] J. S. Liu and R. Chen, “Blind deconvolution via sequential imputations,” *Journal of the american statistical association*, vol. 90, no. 430, pp. 567–576, 1995.

[127] J. S. Liu, *Monte Carlo Strategies in Scientific Computing.* New York, NY: Springer New York, 2004.

[128] S. Julier and J. K. Uhlmann, “A general method for approximating nonlinear transformations of probability distributions,” University of Oxford, Tech. Rep., 1996.

[129] E. Wan and R. Van Der Merwe, “The unscented kalman filter for nonlinear estimation,” in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*, 2000, pp. 153–158.

[130] K. Ito and K. Xiong, “Gaussian filters for nonlinear filtering problems,” *IEEE Transactions on Automatic Control*, vol. 45, no. 5, pp. 910–927, 2000.

[131] B. Jia, M. Xin, and Y. Cheng, “Sparse gauss-hermite quadrature filter with application to spacecraft attitude estimation,” *Journal of Guidance, Control, and Dynamics*, vol. 34, no. 2, pp. 367–379, 2011.

[132] S. Julier and J. Uhlmann, “Unscented filtering and nonlinear estimation,” *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, 2004.

[133] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*, 1st ed., ser. Mathematics in Science and Engineering. Bellman R. Academic Press, New York, 1970, vol. 64.

[134] M. C. Troffaes, “Imprecise monte carlo simulation and iterative importance sampling for the estimation of lower previsions,” *International Journal of Approximate Reasoning*, vol. 101, pp. 31–48, 2018.

[135] A. Doucet, N. de Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*, A. Doucet, N. de Freitas, and N. Gordon, Eds. New York, NY: Springer New York, 2001.

[136] R. Van Der Merwe, A. Doucet, N. De Freitas, and E. A. Wan, “The unscented particle filter,” in *Advances in neural information processing systems*, 2001, pp. 584–590.

[137] S. Julier, J. Uhlmann, and H. Durrant-Whyte, “A new method for the nonlinear transformation of means and covariances in filters and estimators,” *IEEE Transactions on Automatic Control*, vol. 45, no. 3, pp. 477–482, 2000.

[138] B. Kearfott, “A proof of convergence and an error bound for the method of bisection in $\mathbb{R}^n$,” *Mathematics of Computation*, vol. 32, no. 144, pp. 1147–1153, 1978.

[139] J. Žilinskas, “Branch and bound with simplicial partitions for global optimization,” *Mathematical Modelling and Analysis*, vol. 13, no. 1, pp. 145–159, 2008.

[140] J. Clausen and A. Žilinskas, “Subdivision, sampling, and initialization strategies for simplical branch and bound in global optimization,” *Computers & Mathematics with Applications*, vol. 44, no. 7, pp. 943–955, 2002.

[141] K. Witczyński, “n-cubes inscribed in simplices,” *Journal of Geometry*, vol. 81, no. 1-2, pp. 192–198, 2004.

[142] R. B. Hughes and M. R. Anderson, “Simplexity of the cube,” *Discrete Mathematics*, vol. 158, no. 1, pp. 99–150, 1996.

[143] A. Bliss and F. E. Su, “Lower bounds for simplicial covers and triangulations of cubes,” *Discrete & Computational Geometry*, vol. 33, no. 4, pp. 669–686, 2005.

[144] C. Malherbe and N. Vayatis, “Global optimization of lipschitz functions,” in *Proceedings of the 34th International Conference on Machine Learning-Volume 70.* JMLR. org, 2017, pp. 2314–2323.

[145] A. Gimbutas and A. Žilinskas, “On global optimization using an estimate of lipschitz constant and simplicial partition,” in *AIP Conference Proceedings*, vol. 1776 (1). AIP Publishing LLC, 2016, pp. 060 012–1–4.

[146] Y. D. Sergeyev, R. G. Strongin, and D. Lera, *Introduction to global optimization exploiting space-filling curves.* Springer Science & Business Media, 2013.

[147] L. Andrieu, G. Cohen, and F. Vázquez-Abad, “Stochastic programming with probability constraints,” *arXiv preprint arXiv:0708.0281*, 2007.

[148] J. T. Betts, *Practical methods for optimal control and estimation using nonlinear programming.* Siam, 2010, vol. 19.

[149] A. Doucet, N. De Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*, 1st ed. New York: Springer, 2001.

[150] C. Büskens and D. Wassel, “The ESA NLP-Solver WORHP,” in *Modeling and Optimization in Space Engineering*, G. Fasano and J. Pintér, Eds. New York, NY: Springer, 2012, vol. 73, pp. 85–110.

[151] A. Bagchi, *Optimal control of stochastic systems.* Prentice Hall International (UK) Limited, 1993.

[152] T. D. Frank, *Nonlinear Fokker-Planck equations: fundamentals and applications.* Springer Science & Business Media, 2005.

[153] V. Krishnamurthy, *Partially observed Markov decision processes.* Cambridge University Press, 2016.

[154] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and acting in partially observable stochastic domains," *Artificial intelligence*, vol. 101, no. 1-2, pp. 99–134, 1998.

[155] R. C. Jeffrey, *The Logic of Decision*, 2nd ed. University of Chicago Press, 1983.

[156] "Socrates - satellite orbital conjunction reports assessing threatening encounters in space," 2019, accessed: 10-Jul-2019. [Online]. Available: http://celestrak.com/SOCRATES/

[157] N. L. Johnson, E. Stansbery, J.-C. Liou, M. Horstman, C. Stokely, and D. Whitlock, "The characteristics and consequences of the break-up of the fengyun-1c spacecraft," *Acta Astronautica*, vol. 63, no. 1, pp. 128–135, 2008. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0094576507003281

[158] O. Montenbruck and E. Gill, *Satellite orbits: models, methods and applications*, 1st ed. Springer Science & Business Media, 2000.

[159] K. F. Wakker, *Fundamentals of astrodynamics.* TU Delft Library, 2015.

[160] H. Klinkrad, J. Alarcon, and N. Sanchez, "Collision avoidance for operational esa satellites," in *4th European Conference on Space Debris*, vol. 587, 2005, p. 509.

[161] M. Vasile and C. Tardioli, "On the use of positive polynomials for the estimation of upper and lower expectations in orbital dynamics," in *Stardust Final Conference.* Springer Cham, 2018.

[162] B. Schutz, B. Tapley, and G. H. Born, *Statistical orbit determination.* Elsevier, 2004.

[163] J. Sang and C. Smith, "An analysis of observations from eos space debris tracking system," in *11th Australian Space Science Conference, Canberra, Australia.* Citeseer, 2011, pp. 179–189.

[164] J. C. Bennett, J. Sang, C. Smith, and K. Zhang, “An analysis of very short-arc orbit determination for low-earth objects using sparse optical and laser tracking data,” *Advances in Space Research*, vol. 55, no. 2, pp. 617–629, 2015.

[165] B. Li, J. Sang, and Z. Zhang, “A real-time orbit determination method for smooth transition from optical tracking to laser ranging of debris,” *Sensors*, vol. 16, no. 7, p. 962, 2016.

[166] D. Farnocchia, S. Eggl, P. W. Chodas, J. D. Giorgini, and S. R. Chesley, “Planetary encounter analysis on the b-plane: a comprehensive formulation,” *Celestial Mechanics and Dynamical Astronomy*, vol. 131, no. 8, p. 36, 2019.

[167] L. Sánchez Fernández-Mellado and M. Vasile, “On the use of machine learning and evidence theory to improve collision risk management,” *Acta Astronautica*, 2020.

[168] V. Braun, T. Flohrer, H. Krag, K. Merz, S. Lemmens, B. B. Virgili, and Q. Funke, “Operational support to collision avoidance activities by esa’s space debris office,” *CEAS Space Journal*, vol. 8, no. 3, pp. 177–189, 2016.

[169] C. R. Gates, “A simplified model of midcourse maneuver execution errors,” Jet Propulsion Laboratory, California Institute of Technology, Tech. Rep., 1963.

[170] T. D. Goodson, “Execution-error modeling and analysis of the grail spacecraft pair,” in *23rd AAS/AIAA Spaceflight Mechanics Meeting, Kauai, Hawaii*, 2013.

[171] A. O. Hero, D. Castañón, D. Cochran, and K. Kastella, *Foundations and applications of sensor management.* Springer Science & Business Media, 2007.

[172] R. Ionasescu, “Orbit determination covariance analysis for the Cassini solstice mission,” in *AIAA/AAS Astrodynamics Specialist Conference*, 2010.

[173] R. Ionasescu, T. Martin-Mury, P. Valerinoy, K. Criddley, B. Buffington, and T. McElrath, “Orbit determination covariance analysis for the Europa Clipper mission,” in *AIAA Space 2014*, 2014.

[174] B. Jia, M. Xin, and Y. Cheng, “Sparse gauss–hermite quadrature filter with application to spacecraft attitude estimation,” *Journal of Guidance, Control, and Dynamics*, vol. 34, pp. 367–379, 03 2011.

[175] J. Pelamatti, L. Brevault, M. Balesdent, E.-G. Talbi, and Y. Guerin, “How to deal with mixed-variable optimization problems: An overview of algorithms and formulations,” in *World Congress of Structural and Multidisciplinary Optimisation.* Springer, 2017, pp. 64–82.

[176] H. M. Nyew, O. Abdelkhalik, and N. Onder, “Autonomous interplanetary trajectory planning using structured-chromosome evolutionary algorithms,” in *AIAA/AAS Astrodynamics Specialist Conference*, 2012, p. 4522.

[177] H. M. Nyew, O. Abdelkhalik, and N. Onder, “Structured-chromosome evolutionary algorithms for variable-size autonomous interplanetary trajectory planning optimization,” *Journal of Aerospace Information Systems*, vol. 12, no. 3, pp. 314–328, 2015.

[178] S. A. Darani and O. Abdelkhalik, “Space trajectory optimization using hidden genes genetic algorithms,” *Journal of Spacecraft and Rockets*, vol. 55, no. 3, pp. 764–774, 2017.

[179] M. Mitchell, *An introduction to genetic algorithms.* MIT press, 1998.

[180] L. Gentile, “LorenzoGentile/SCGA: SCGA for satellite tracking,” Jul. 2020. [Online]. Available: https://doi.org/10.5281/zenodo.3968260

[181] C. Acton, N. Bachman, B. Semenov, and E. Wright, “A look toward the future in the handling of space science mission geometry,” *Planetary and Space Science*, vol. 150, pp. 9–12, 2018.

[182] L. Scrucca *et al.*, “Ga: a package for genetic algorithms in r,” *Journal of Statistical Software*, vol. 53, no. 4, pp. 1–37, 2013.

[183] R. Broucke and P. J. Cefola, “On the equinoctial orbit elements,” *Celestial Mechanics*, vol. 5, no. 3, pp. 303–310, 1972.

[184] F. Zuiani and M. Vasile, “Extended analytical formulas for the perturbed Keplerian motion under a constant control acceleration,” *Celestial Mechanics and Dynamical Astronomy*, vol. 121, pp. 275–300, 2015.

[185] M. Di Carlo, J. M. Romero Martin, and M. Vasile, “CAMELOT: Computational-Analytical Multi-fidElity Low-thrust Optimisation Toolbox,” *CEAS Space Journal*, vol. 10, pp. 25–36, 2018.

[186] T. Lam, B. Buffington, and S. Campagnola, “A robust mission tour for nasa’s planned europa clipper mission,” in *2018 Space Flight Mechanics Meeting*, 2018, p. 0202.

[187] A. E. Lynam and J. M. Longuski, “Preliminary analysis for the navigation of multiple-satellite-aided capture sequences at jupiter,” *Acta Astronautica*, vol. 79, pp. 33–43, 2012.

[188] 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*, vol. 150, pp. 9–12, 2018.

[189] D. R. Boone, S. Nandi, J. Kangas, and B. Young, “Orbit determination sensitivity analysis for the europa multiple flyby mission concept,” in *AIAA/AAS Astrodynamics Specialist Conference*, 2016.

[190] S. V. Wagner, “Cassini maneuver performance assessment and execution-error modeling through 2015,” in *26th AAS/AIAA Space Flight Mechanics Meeting*, Napa, CA, US, February 2016.

[191] L. G. Casado, I. García, B. Tóth, and E. M. Hendrix, “On determining the cover of a simplex by spheres centered at its vertices,” *Journal of Global Optimization*, vol. 50, no. 4, pp. 645–655, 2011.

[192] D.-H. Cho, Y. Chung, and H. Bang, “Trajectory correction maneuver design using an improved b-plane targeting method,” *Acta Astronautica*, vol. 72, pp. 47–61, 2012.