Acta Astronautica 234 (2025) 548–559

Contents lists available at ScienceDirect

Acta Astronautica

journal homepage: www.elsevier.com/locate/actaastro

Efficient collision avoidance manoeuvres under multiple polynomial constraints

Zeno Pavanello *, Laura Pirovano, Roberto Armellin

Te Pūnaha Ātea - The Space Institute, The University of Auckland, 20 Symonds Street, Auckland, 1010, New Zealand

ARTICLE INFO

Keywords:
Collision avoidance manoeuvre
Trajectory optimisation
Polynomial programming
Space situational awareness

ABSTRACT

We propose a simple and reliable algorithm for collision avoidance manoeuvre optimisation, capable of computing impulsive, multi-impulsive, and low-thrust manoeuvres. Utilising differential algebra techniques, safety metrics such as probability of collision and miss distance are approximated by a polynomial of arbitrary order as a function of the control. Moreover, operational constraints, namely the return to the nominal trajectory and station-keeping, are employed and treated in the same way. This allows for the transcription of the collision avoidance manoeuvre problem as a polynomial programme with a quadratic (energy-optimal) objective function and high-order polynomial constraints. The polynomial constraints are successively linearised using an iterative approach, wherein the updated linear constraints are calculated from the high-order polynomial maps. This avoids the need for the recomputation of linear maps from a new expansion point. Thus, the original polynomial programme is effectively approximated by a series of quadratic programmes, enabling accurate solutions to be achieved even for highly nonlinear polynomial constraints. The method is showcased in several scenarios involving multiple short-term conjunctions and station keeping requirements. Solutions are found with run times between 0.1 s and 0.5 s, proving that the method is computationally efficient and making it potentially suitable for autonomous applications.

1. Introduction

According to the most recent European Space Agency (ESA) Space Debris report [1], there are now almost 40,000 catalogued resident space objects. This significant increase, driven by spacecraft miniaturisation and the deployment of mega-constellations like Starlink, is transforming space traffic management. A more crowded space environment heightens the frequency of conjunctions, necessitating a corresponding rise in CAMs.

Currently, conjunction analysis and collision avoidance operations are mainly performed by ground-based operators using tools and processes developed over the past two decades. Although these tools support operational activities, the reliance on human intervention for decision-making and manoeuvre design is becoming unsustainable. Consequently, there is an increasing demand for automated conjunction screening, collision avoidance decision-making, and CAM design and execution. This drives the need CAM algorithms suitable for autonomous and potentially onboard computations, which can provide fuel-optimal manoeuvres in a short execution time. This work aims to address this urgent need by proposing a method for CAM optimisation suitable for autonomous applications.

A CAM is executed when operators receive a conjunction data message (CDM) indicating the TCA between a satellite and a secondary object, along with the relevant conjunction information. Typically, the estimated PoC at TCA exceeds a threshold (e.g., 10^{-4}), and the CAM's objective is to reduce this probability while minimising the total Δv [2]. Alternatively, if the uncertainty in the state of the two objects is not a concern, the conjunction can be mitigated by increasing the MD at TCA. Previous research has explored various methodologies for optimising CAMs, often simplifying the problem by assuming small manoeuvres [3]. Alfano [4] introduced a CAM analysis tool capable of conducting parametric studies on single-axis and dual-axis manoeuvres, while the German Aerospace Center and ESA have developed their own CAM optimisation tools with varying degrees of flexibility and optimality [5,6]. Moreover, recent advancements have seen the emergence of multi-objective approaches for CAM design, enabling comprehensive analyses and the identification of Pareto optimal solutions [7]. Although useful for robust and optimal manoeuvre computations, these solutions are not feasible for autonomous applications due to their high computational load. In recent years, substantial research has been devoted to developing efficient CAM algorithms. Bombardelli [8]

* Corresponding author.
E-mail address: zpav176@aucklanduni.ac.nz (Z. Pavanello).

https://doi.org/10.1016/j.actaastro.2025.05.002
Received 22 December 2024; Received in revised form 23 April 2025; Accepted 1 May 2025
Available online 20 May 2025
0094-5765/

and Bombardelli and Hernando-Ayuso [9,10] devised analytical and semi-analytical methods to maximise the MD or minimise PoC using a single impulse or a low-thrust arc, demonstrating promising convergence properties. De Vittori et al. proposed an analytical approach for the energy-optimal problem and a semi-analytical approach for the fuel-optimal problem to handle low-thrust CAMs [11], which are particularly relevant given the increasing popularity of electric thrusters [12]. Direct optimisation methods have also been extensively utilised to compute CAMs. Misra and Dutta [13] cast the problem as a convex programme by linearising the PoC constraint, achieving fast but conservative fuel-optimal solutions. Convex optimisation has also been proposed by one of the authors of his work, who transcribed the problem into a sequential convex programme (SCP) using differential algebra (DA) and lossless convexification and transforming the PoC constraint into a keep-out-zone constraint [14]; the convex programme is then solved iteratively to achieve the original fuel-optimal solution. This method was later extended by the authors of this work to solve long-term encounters [15], multiple consecutive encounters [16], and encounters during low-thrust arcs [17].

Continuing the investigation on direct methods, in this work, we propose a simple and efficient method for CAM design. Building on Ref. [18], where the method is used to solve a polynomial programme (PP) with a single equality constraint on PoC, we expand the methodology to include multiple equality and inequality constraints. In this way, it is possible to optimise manoeuvres to mitigate the risk coming from multiple consecutive conjunctions and include SK constraints on the final target state or the final mean orbital elements. These features are becoming important in real-world operations where conjunctions are ever more frequent, and the disruptions caused by CAMs to mission operations must be quickly mitigated. The CAM design problem is cast as a polynomial programme (PP), utilising the energy-optimal objective function and approximating PoC and any other scalar constraint as high-order Taylor polynomials in the control, which can be represented as a series of impulses or accelerations applied at specified nodes or segments. The PP is solved by leveraging the assumption that high-order terms diminish in a convergent Taylor series. Thus, the problem is initially turned into a quadratic programme (QP) by truncating the constraints in the first order. State-of-the-art solvers, like the active set method, allow for the recovery of the optimal solution of the quadratic programme (QP) in polynomial time. Subsequently, a simple and computationally efficient recursive approach is established to achieve quasi-optimal results. The PP is progressively approximated by a sequence of QPs, which are obtained by linearising the high-order terms of the polynomials using the solution from the previous iteration. In this way, a (locally) optimal solution of the original PP is recovered.

The paper is organised as follows. In Section 2, the dynamics models considered for the CAM applications are introduced. Section 3 sets the PP using automatic Taylor expansions. The recursive algorithm to solve the CAM PP is shown in Section 4. In Section 6, operational results are presented, and the comparison with the interior-point solver is discussed. Lastly, in Section 7, conclusions are drawn.

2. Dynamics and constraints definition

This section introduces the framework of the CAM optimisation problem dynamics, including the conjunction B-plane dynamics. Moreover, the SK requirements are introduced as bound on transformations of the final state of the primary.

2.1. Earth orbit dynamics

Let us assume that the mean state of the objects is given for the time of the first conjunction $t_{CA,1} \in \mathbb{R}$; the manoeuvrable satellite (primary) is back-propagated up to a suitable time to start executing the manoeuvre. Also, it is forward-propagated up to a time $t_f \in \mathbb{R}$ for the inclusion of station keeping constraint. Therefore, the time frame of interest is $t \in \mathbb{R}_{[t_0,t_f]}$. Since the Taylor expansions performed with DA are automatic, the algorithm is compatible with any dynamics model, so any representations of the orbital environment can be employed. In this work, the time frame is relatively short, so orbital perturbations like high-order gravitational harmonics, atmospheric drag, solar radiation pressure, and third-body attraction do not play a significant role. For this reason, we limit our perturbation model to the J_2 term of the gravitational potential of the Earth

$$\begin{aligned}
&\dot{\boldsymbol{r}}(t) = \boldsymbol{v}(t)\\
&\ddot{r}_x(t) = -\frac{\mu}{r(t)^3} x(t)\left(1 + k_{J_2}\left(1 - 5z(t)^2/r(t)^2\right)\right) + u_x(t)\\
&\ddot{r}_y(t) = -\frac{\mu}{r(t)^3} y(t)\left(1 + k_{J_2}\left(1 - 5z(t)^2/r(t)^2\right)\right) + u_y(t)\\
&\ddot{r}_z(t) = -\frac{\mu}{r(t)^3} z(t)\left(1 + k_{J_2}\left(3 - 5z(t)^2/r(t)^2\right)\right) + u_z(t),
\end{aligned} \tag{1}$$

where $\boldsymbol{r}(\cdot) = [r_x(\cdot)\ r_y(\cdot)\ r_z(\cdot)]^\top : \mathbb{R} \to \mathbb{R}^3$ is the position of the spacecraft, $\boldsymbol{v}(\cdot) : \mathbb{R} \to \mathbb{R}^3$ is its velocity, $\mu \in \mathbb{R}_+$ is the gravitational constant of the Earth, $k_{J_2} = \frac{3}{2}\left(\frac{R_E}{r}\right)^2 J_2 \in \mathbb{R}_+$, $R_E \in \mathbb{R}_+$ is the Earth's equatorial radius, $J_2 \in \mathbb{R}_+$ is the Earth's oblateness coefficient, and $\boldsymbol{u}(\cdot) = [u_x(\cdot)\ u_y(\cdot)\ u_z(\cdot)]^\top : \mathbb{R} \to \mathbb{R}^3$ is the acceleration control action. The mass loss due to the manoeuvre is considered negligible because a CAM usually involves a small fuel consumption [9].

2.2. Conjunction in the B-plane

We intend to study the general case of multiple consecutive conjunctions between a primary manoeuvrable spacecraft and $n_{conj} \in \mathbb{N}$ secondary objects. The relative velocity between the two satellites at TCA is sufficiently high for all the encounters to be considered instantaneous events [14]. Each conjunction is fully defined by the ECI states of the two objects involved in terms of mean value and positional covariance matrix. Therefore, the mean state of the primary is called $\boldsymbol{x}_p(t) = [\boldsymbol{r}_p(t);\ \boldsymbol{v}_p(t)]$, while the secondaries have mean state $\boldsymbol{x}_s(t) = [\boldsymbol{r}_s(t);\ \boldsymbol{v}_s(t)]$. Here, $s \in \{1, \ldots, n_{conj}\}$ is the identifier of the secondary and $\boldsymbol{r}_p(\cdot), \boldsymbol{r}_s(\cdot), \boldsymbol{v}_p(\cdot)$ and $\boldsymbol{v}_s(\cdot) : \mathbb{R} \to \mathbb{R}^3$ denote the position and velocity of the centres of mass of the objects. Analogously, the covariances of the two objects are $\boldsymbol{C}_p(\cdot)$ and $\boldsymbol{C}_s(\cdot) : \mathbb{R} \to \mathbb{R}^{6\times 6}$. The relative state is defined as the subtraction of the two absolute states

$$\boldsymbol{x}_{\Delta s}(t) = \boldsymbol{x}_p(t) - \boldsymbol{x}_s(t), \tag{2a}$$

$$\boldsymbol{C}_{\Delta s}(t) = \boldsymbol{C}_p(t) + \boldsymbol{C}_s(t). \tag{2b}$$

Note that $\Delta\boldsymbol{x}_{\Delta s}(t) = [\boldsymbol{r}_{\Delta s}(t)^\top\ \boldsymbol{v}_{\Delta s}(t)^\top]^\top$ and the upper-left 3×3 submatrix of $\boldsymbol{C}_{\Delta s}(t)$ is called $\boldsymbol{P}_{\Delta s}(t)$ (positional combined covariance).

Given the short-term nature of the encounter, it can be studied in the B-plane reference frame denoted as $\mathcal{B}_s$. As depicted in Fig. 1, $\mathcal{B}_s$ is centred on the secondary object; the η_s axis aligns with the direction of the relative velocity of the primary with respect to the secondary, while the $\xi_s\zeta_s$ plane is perpendicular to the relative velocity axis. Since the nominal TCA for conjunction s, represented by $t^*_{CA,s} \in \mathbb{R}$, is the moment when the MD between the primary the secondary s is minimum, by assumption,

$$\boldsymbol{r}_{\Delta s}(t^*_{CA,s}) \cdot \boldsymbol{v}_{\Delta s}(t^*_{CA,s}) = 0 \tag{3}$$

indicating that the relative position lies within the $\xi_s\zeta_s$ plane. From this point on, the argument of the functions is omitted for brevity: all the quantities introduced in this section are considered as functions of the nominal TCA of the single conjunction. Therefore, for the computation of PoC, we will employ the projection of the relative state and its covariance on the $\xi_s\zeta_s$ plane: $\boldsymbol{r}_{\mathcal{B}_s}(\cdot) : \mathbb{R} \to \mathbb{R}^2$ and $\boldsymbol{P}_{\mathcal{B}_s}(\cdot) : \mathbb{R} \to \mathbb{R}^{2\times 2}$.

$$\boldsymbol{r}_{\mathcal{B}_s} = \boldsymbol{R}_s\ \boldsymbol{r}_{\Delta s}, \tag{4a}$$

$$\boldsymbol{P}_{\mathcal{B}_s} = \boldsymbol{R}_s\ \boldsymbol{P}_{\Delta s}\ \boldsymbol{R}_s^\top. \tag{4b}$$

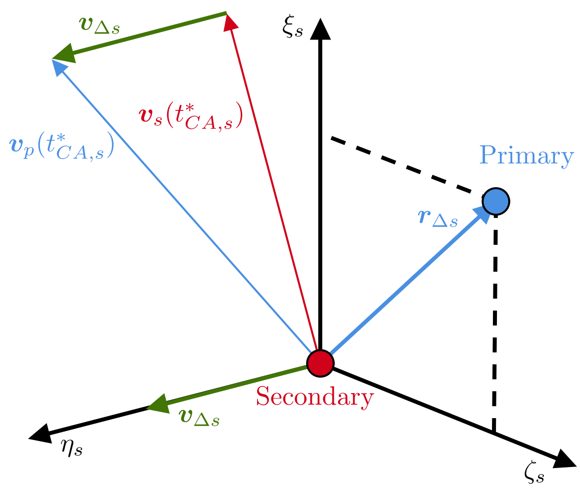

Fig. 1. B-plane construction for conjunction s.
Source: Adapted from [14].

$\boldsymbol{R}_s(\cdot) : \mathbb{R}^3 \times \mathbb{R}^3 \to \mathbb{R}^{2\times3}$ performs the rotation between the reference frame used for the propagation of the states and the $\mathcal{B}_s$

$$\boldsymbol{R}_s = \boldsymbol{R}_s(\boldsymbol{v}_p, \boldsymbol{v}_s) = [\hat{\boldsymbol{u}}_{\xi_s}^\top \ \hat{\boldsymbol{u}}_{\zeta_s}^\top], \tag{5}$$

where $\hat{\boldsymbol{u}}_{\xi_s}$ and $\hat{\boldsymbol{u}}_{\zeta_s}$ are the unit vectors of the ξ_s and η_s axes. The explicit expression of $\boldsymbol{R}_s$ can be found in Ref. [14]. The subsequent discussion assumes that all uncertainty is concentrated around the secondary object, while all mass is concentrated around the primary [19].

A conservative way to compute PoC of the encounter is enveloping each of the bodies in a sphere with a radius equal to the largest dimension. The sum of the radii of the two spheres is the HBR, which defines the combined hard body sphere centred in the primary. The projection of this sphere onto the B-plane is the combined hard body circle $\mathbb{C}_{\text{HBR}}$. To compute PoC of the single conjunction, denoted as $P_{C,s} \in \mathbb{R}_+$, the probability density function of the projection of the relative position onto the B-plane is integrated over the combined hard-body circle, as follows

$$P_{C,s} = \frac{1}{(2\pi)^{3/2}\sqrt{\det\left(\boldsymbol{P}_{\mathcal{B}_s}\right)}} \cdot \iint_{\mathbb{C}_{\text{HBR}}} \mathrm{e}^{-d_{m,s}^2/2}\mathrm{d}A, \tag{6}$$

where $d_{m,s}^2(\cdot) : \mathbb{R} \to \mathbb{R}_+$ is the squared Mahalanobis distance of the relative position in the B-plane and is defined as $d_{m,s}^2 = \boldsymbol{r}_{\mathcal{B}_s}^\top \boldsymbol{P}_{\mathcal{B}_s}^{-1} \boldsymbol{r}_{\mathcal{B}_s}$. Multiple approaches have been proposed to approximate the integral in Eq. (6) [19]. The results presented in Section 6 are obtained using Chan's method [20], which approximates PoC through a convergent series that involves equivalent cross-sections. Nonetheless, our optimisation method is compatible with any PoC model since the automatic Taylor expansion can deal with any function. Therefore, the following discussion is made without assuming any particular PoC model: in general, for a given HBR, PoC is expressed as a function of the relative position, and the combined covariance in the B-plane at TCA

$$P_{C,s} = P_C(\boldsymbol{r}_{\mathcal{B}_s}, \boldsymbol{P}_{\mathcal{B}_s}). \tag{7}$$

By means of an automatic DA map inversion, we include the effect of the TCA shift due to the manoeuvre in the calculation of PoC. For details on the derivation of Eq. (7), including the implicit inclusion of the TCA shift, we refer the reader to Ref. [18].

If multiple consecutive encounters take place in the scenario, PoC of the single conjunction can be computed using Eq. (7), and, under the assumption that the conjunctions are uncorrelated [21], the total probability of collision (TPoC) of the scenario is given by the combination of the probabilities of the individual encounters [16]

$$P_C = 1 - \prod_{s=1}^{n_{conj}} \left(1 - P_{C,s}\right) \tag{8}$$

where P_C is TPoC and $P_{C,s}$ is PoC of the single conjunction.

Alternatively, a common way to mitigate the risk during an encounter is to increase MD over a predefined threshold. This can be imposed on each one of the consecutive conjunctions. The formula for the MD is more effectively expressed in its quadratic form

$$d_{miss,s}^2 = \|\boldsymbol{r}_{\mathcal{B}_s}\|^2 \geq \bar{d}^2, \tag{9}$$

where $d_{miss,s}^2$ and $\bar{d}^2 \in \mathbb{R}_+$ are MD of conjunction s and the arbitrary threshold, respectively.

2.3. Station-keeping requirements

In typical satellite operations, it is pivotal to recover the nominal trajectory of the spacecraft after a manoeuvre has been executed. In fact, during CAMs, the normal operations are suspended until the nominal orbit is achieved back. For this reason, it is very useful to couple the CAM problem with the following SK problem in order to compute a single optimal sequence of impulses to achieve both objectives. In this work, we propose two alternative formulations of the SK requirement: the first one is based on the osculating Keplerian elements of the final state of the satellite, while the second one takes advantage of the mean orbital elements, which are typically used in spacecraft operations.

The former approach consists of defining five constraints on the first five orbital elements at the final time $t_f \in \mathbb{R}$ (the mean anomaly is not controlled). The transformation between final Cartesian and Keplerian states can be found in the literature [22]; here, it is indicated with the vector function $\boldsymbol{f}_{kep}(\cdot) : \mathbb{R}^6 \to \mathbb{R}^6$. The constraints are imposed on each element of the Keplerian coordinates vector

$$\varepsilon_a = a(t_f) - \bar{a}(t_f) = 0, \tag{10a}$$

$$\varepsilon_e = e(t_f) - \bar{e}(t_f) = 0, \tag{10b}$$

$$\varepsilon_i = i(t_f) - \bar{i}(t_f) = 0, \tag{10c}$$

$$\varepsilon_\omega = \omega(t_f) - \bar{\omega}(t_f) = 0, \tag{10d}$$

$$\varepsilon_\Omega = \Omega(t_f) - \bar{\Omega}(t_f) = 0, \tag{10e}$$

where a, e, i, ω, and $\Omega \in \mathbb{R}$ are the semi-major axis, eccentricity, inclination, argument of perigee, and right ascension of the ascending node at the final time; $\varepsilon_w \in \mathbb{R}_+$ and $\bar{w}(t_f)$ for $w = \{a, e, i, \omega, \Omega\}$ are the errors and the nominal values of the Keplerian elements, respectively.

Similarly, the second method imposes that the final mean orbital elements of interest stay within certain bounds. Given the osculating Cartesian states of the primary at the final time t_f, the vector of the mean Keplerian orbital elements $\boldsymbol{p}(t_f) \in \mathbb{R}^6$ can be computed through a nonlinear transformation $\boldsymbol{q}(\cdot) : \mathbb{R}^6 \to \mathbb{R}^6$:

$$\boldsymbol{p}(t_f) = \boldsymbol{q}(\boldsymbol{x}_p(t_f)), \tag{11}$$

The specific expression for $\boldsymbol{q}(\cdot)$ falls out of the scope of this work, but it can be found in [23]. In particular, we are interested in the first two elements of $\boldsymbol{p}(t_f)$, which are the mean semi-major axis $a_m(t_f) \in \mathbb{R}_+$ and the norm of the mean eccentricity vector $e_m(t_f) \in \mathbb{R}_+$

$$a_m(t_f) = q_1(\boldsymbol{x}_p(t_f)) \tag{12a}$$

$$e_m(t_f) = q_2(\boldsymbol{x}_p(t_f)). \tag{12b}$$

A constraint on these two elements is valuable if it imposes that their value cannot deviate more than a given threshold from the nominal. Therefore, the constraint is imposed on the squared value of the deviation

$$\varepsilon_{sma} = (a_m(t_f) - \bar{a}_m(t_f))^2 \leq \Delta a_{max}^2 \tag{13a}$$

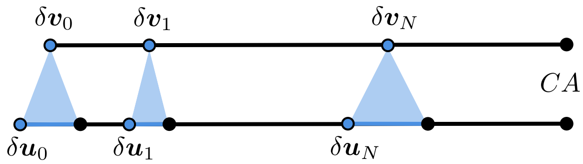

Fig. 2. Impulsive and low-thrust designs. Black nodes represent the start of ballistic segments; blue nodes correspond to impulses or the start of low-thrust segments.

$$\varepsilon_{ecc} = (e_m(t_f) - \bar{e}_m(t_f))^2 \leq \Delta e^2_{max} \tag{13b}$$

where $\bar{a}_m$ and $\bar{e}_m \in \mathbb{R}_+$ are the nominal the value of the mean elements of the unmanoeuvred trajectory, ε_{sma} and $\varepsilon_{ecc} \in \mathbb{R}_+$ are the errors, and Δa_m and $\Delta e_m \in \mathbb{R}_+$ are the maximum allowed deviations.

3. CAM optimisation problem

In this section, the CAM optimisation Problem is set first as an optimal control problem (OCP) and afterwards as a PP. The problem formulation is kept as generic as possible, with the inclusion of multiple operational constraints.

3.1. Optimal control problem

Let us define a fuel-optimal OCP. The thrusting opportunities are fixed at predefined times before TCA. These are collected into the set $\mathbb{T}_{man} = \{t_0, \ldots, t_{N_{man}-1}\}$, where $N_{man} \in \mathbb{N}$ is the number of manoeuvres. The algorithm proposed in this work is intended to compute manoeuvres either using a low-thrust dynamics model or impulsive approximation. In the former case, the optimisation variable is the continuous $\boldsymbol{u}(t)$. In the latter, $\boldsymbol{u}(t) = \boldsymbol{0}_3$ at every time, and the control is given by the discrete impulses $\Delta \boldsymbol{v}_i$ for $i \in \{0, \ldots, N_{man}-1\}$. In Fig. 2, the equivalent impulsive and low-thrust schemes are shown: the blue nodes depict the times where the control variables are placed, while the black ones indicate ballistic propagation points. To include both possibilities, in the following discussion, the control variable is denominated $\boldsymbol{\phi} \in \mathbb{R}^3$.

To generalise the discussion, including any operational constraint, let us consider a set of $n_{constr} \in \mathbb{N}$ scalar constraints in the control history $\boldsymbol{\phi}$. In particular, we have $n_{eq} \in \mathbb{N}$ equality and $n_{in} \in \mathbb{N}$ inequality constraints. Also, the control perturbation history is collected into the stacked vector $\delta\boldsymbol{\Phi} = [\delta\boldsymbol{\phi}_0^\top \ \ldots \ \delta\boldsymbol{\phi}_{N_{man}-1}^\top]^\top \in \mathbb{R}^M$, where $M = 3N_{man}$.

The fuel-optimal OCP can be generally stated as

$$\min_{\boldsymbol{\Phi}} \quad J = \begin{cases} \sum_{i=0}^{N_{man}-1} \left(\int_{t_i}^{t_i+\Delta t_i} \|\boldsymbol{u}(\tau)\| \mathrm{d}\tau \right) \\ \sum_{i=0}^{N_{man}-1} \|\Delta \boldsymbol{v}_i\| \end{cases} \tag{14a}$$

$$\text{s.t.} \quad \text{Eq. (1)} \tag{14b}$$

$$g_v(\boldsymbol{\Phi}) \leq \bar{g}_v, \qquad v \in \{1, \ldots, n_{in}\} \tag{14c}$$

$$h_c(\boldsymbol{\Phi}) = \bar{h}_c, \qquad c \in \{1, \ldots, n_{eq}\} \tag{14d}$$

$$\boldsymbol{r}_p(t_0) = \boldsymbol{r}_{p,0}, \tag{14e}$$

where $\Delta t_i \in \mathbb{R}_+$ is the duration of the low-thrust window i; $g_v(\cdot)$ and $h_c(\cdot) : \mathbb{R}^M \to \mathbb{R}$ are the scalar functions that define the equality and inequality constraints, respectively; $\bar{g}_v$ and $\bar{h}_c \in \mathbb{R}$ are the constraint thresholds. Eq. (14a) imposes the minimisation of the fuel-optimal objective function $J(\cdot) : \mathbb{R}^M \to \mathbb{R}_+$: the first case is used for low-thrust dynamics, the second one for the multi-impulsive; Eq. (14b) is the dynamics constraint; Eqs. (14c) and (14d) include all the inequality and equality nonlinear constraints; Eq. (14e) defines the initial position of the satellite, which cannot be altered.

3.2. Polynomial programme

Problem Eq. (14) is discretised by using a Runge–Kutta 7–8 integration scheme. This method is preferred over more classic ones like Runge–Kutta 4–5 because it requires fewer steps for the integration of the dynamics; this makes it more efficient in the computation of high-order DA coefficients. Each constraint is defined on one or more of the discretised time nodes: for example, the PoC constraint for the single conjunction is defined at node $t_{CA,s} \in \mathbb{R}_{[t_0,t_f]}$ and the SK constraints are defined at t_f. Therefore, we appropriately discretise the time dominion to account for thrusting opportunities and nodes needed for the constraints. The complete discretised time grid is called $\mathbb{T} = \{t_0, t_{CA,1}, \ldots, t_N\}$, where $t_N = t_f$ and $N \in \mathbb{N}$. $\mathbb{T}_{man} \subset \mathbb{T}$. The initial state $\boldsymbol{x}_0$ is yielded by back-propagating the ballistic motion of the primary starting from TCA. The nominal state at subsequent thrusting opportunities can be obtained via successive forward propagations. Therefore, the discretised dynamics equations at any discretisation node t_i are generally written

$$\boldsymbol{x}_i = \boldsymbol{f}_i(\boldsymbol{x}_{i-1}, \boldsymbol{u}_{i-1}) \qquad i \in \{1, \ldots, N\}, \tag{15}$$

where $\boldsymbol{f}_i(\cdot) : \mathbb{R}^6 \times \mathbb{R}^3 \times \mathbb{R}^{m_p} \to \mathbb{R}^6$ is the discretised dynamics function. The discretisation assumes a first-order hold for the input action; therefore, the control between two consecutive firing opportunities is constant.

By means of DA, we introduce perturbations on the control variable at each thrusting opportunity ($\boldsymbol{\phi}_i + \delta\boldsymbol{\phi}_i$) in a forward-propagation scheme. Considering a ballistic reference trajectory (no initial control), in the multi-impulsive case, the velocity perturbation is added to the state at every node

$$\boldsymbol{x}_i = \mathcal{T}^n_{\boldsymbol{x}_i}(\boldsymbol{x}_{i-1}) + [\boldsymbol{0}_3 \ \delta\boldsymbol{v}_i]^\top i \in \{1, \ldots, N_{man}\} \tag{16a}$$

$$\boldsymbol{x}_{p,0} = \boldsymbol{x}_{p,0} + [\boldsymbol{0}_3 \ \delta\boldsymbol{v}_0]^\top, \tag{16b}$$

where $\mathcal{T}^n_{\boldsymbol{x}_i}(\cdot) : \mathbb{R}^6 \to \mathbb{R}^6$ is the nth-order Taylor series approximation of $\boldsymbol{x}_i$. In the low-thrust case, the acceleration is defined at every node using DA perturbations, and it directly contributes to the dynamics[1]

$$\boldsymbol{x}_i = \mathcal{T}^n_{\boldsymbol{x}_i}(\boldsymbol{x}_{i-1}, \boldsymbol{u}_{i-1}) \quad i \in \{1, \ldots, N\}, \tag{17}$$

where $\mathcal{T}^n_{\boldsymbol{x}_i}(\cdot) : \mathbb{R}^6 \times \mathbb{R}^3 \to \mathbb{R}^6$. A detailed explanation of the use of DA can be found in [24]. As this is a polynomial expansion, its accuracy is inversely proportional to the entity of the perturbation and directly proportional to the polynomial order.

Assuming that each of the variables constrained in Eqs. (14c) and (14d) (e.g., PoC from Eq. (7), or the mean final semi-major axis from Eq. (12a)) are dependent on the control history, by using the chain rule, one can build a multivariate polynomial representation of each g_v and h_c as dependent on the control perturbations stacked vector

$$\begin{aligned} g_v(\boldsymbol{\Phi}) &= \mathcal{T}^n_{g_v}(\delta\boldsymbol{\Phi}), \qquad v \in \{1, \ldots, n_{in}\} \\ h_c(\boldsymbol{\Phi}) &= \mathcal{T}^n_{h_c}(\delta\boldsymbol{\Phi}), \qquad c \in \{1, \ldots, n_{eq}\} \end{aligned} \tag{18}$$

where $\mathcal{T}^n_{h_c}(\cdot) : \mathbb{R}^M \to \mathbb{R}$. We define the vector of the control history as

$$\boldsymbol{\Phi} = \left[\boldsymbol{\phi}_0^\top \quad \ldots \quad \boldsymbol{\phi}_N^\top\right]^\top \in \mathbb{R}^M. \tag{19}$$

A PP is constructed in the following form

$$\min_{\boldsymbol{\Phi}} \quad \boldsymbol{\Phi}^\top\boldsymbol{\Phi} \tag{20a}$$

$$\text{s.t.} \quad \mathcal{T}^n_{g_v}(\boldsymbol{\Phi}) \leq \bar{g}_v, \qquad v \in \{1, \ldots, n_{in}\} \tag{20b}$$

[1] As illustrated in Fig. 2, the last node of each low-thrust arc is idle to set where the arc stops. This implies that a low-thrust design is more computationally demanding than an impulsive one since at least two nodes are needed to define the low-thrust arc, while only one is necessary for the impulse.

$$\mathcal{T}^n_{h_c}(\boldsymbol{\Phi}) = \bar{h}_c, \qquad c \in \{1, \ldots, n_{eq}\} \tag{20c}$$

where Eq. (20a) is the energy-optimal objective function and Eqs. (20b) and (20c) are the scalar polynomial constraints. Problem (20) can be solved using a global optimisation tool[2] or MATLAB's fmincon.[3]

4. Sequential polynomial recursive method

We employ a recursive approach to find solutions to PPs of increasing polynomial orders. The polynomials of the constraints are initially truncated in the 1st order, turning the PP into a QP to obtain an approximate solution; this solution is used as a first guess to solve the problem with a polynomial constraint truncated in the 2nd order. Therefore, the 2nd- order PP is approximated as a series of QPs, where the quadratic constraints are linearised using the solution from the previous iteration; when convergence is reached, the recovered solution is fed to the 3rd-order truncated PP, yielding a new sequence of QPs that approximate the cubic constraints, and this process is repeated up to the nth order.

4.1. Mathematical background

The formulation of the recursive method is based on the use of tensor notation for Taylor expansions. Given a function $f(\cdot) : \mathbb{R}^m \to \mathbb{R}$, its Taylor expansion can be computed up to an arbitrary order n. The polynomial approximating $f(\cdot)$, as computed on the expansion point $\boldsymbol{y}^* \in \mathbb{R}^m$ can be written as

$$\mathcal{T}^n_{f(y)}(\boldsymbol{y}) = f(\boldsymbol{y}^*) + \sum_{k=1}^{n} \frac{1}{k!} \frac{\partial^k f}{\partial \boldsymbol{y}^k}(\boldsymbol{y}^*)(\boldsymbol{y} - \boldsymbol{y}^*)^k. \tag{21}$$

Eq. (21) is equivalent to writing the polynomial using tensors of increasing orders [25]. The gradient of $g(\cdot)$ is a 1st-order tensor, i.e., a vector; its Hessian is a 2nd-order tensor, i.e., a matrix, and so on. Since the derivative operation is commutative, these tensors have the useful property of super-symmetry, i.e., they are invariant to permutations of the indices. The set of all super-symmetric tensors of dimension m and order k is denominated $\mathbb{S}^{m^k} \subset \mathbb{R}^{m^k}$. It is possible to define the kth-order tensor using k different indices: the vector $\boldsymbol{a} \in \mathbb{S}^m$ is indexed as a_i ($i \in \{1, \ldots, m\}$), the matrix $\boldsymbol{A} \in \mathbb{S}^{m^2}$ is indexed as $A_{i_1 i_2}$ ($i_1, i_2 \in \{1, \ldots, m\}$), the third order tensor $\boldsymbol{\mathcal{A}} \in \mathbb{S}^{m^3}$ is indexed $\mathcal{A}_{i_1 i_2 i_3}$ ($i_1, i_2, i_3 \in \{1, \ldots, m\}$), and so on. With this tool, we can now express the kth-order derivatives of $g(\cdot)$ using kth-order tensors. In the following discussion, the order of the tensor will be indicated as a bracketed superscript, $\boldsymbol{\mathcal{A}}^{(k)}$. Given a tensor $\boldsymbol{\mathcal{A}}^{(k)}$ and a vector $\boldsymbol{v} \in \mathbb{R}^m$, their multi-linear form $\boldsymbol{\mathcal{A}}^{(k)}\boldsymbol{v}^k \in \mathbb{R}$ is defined as:

$$\boldsymbol{\mathcal{A}}^{(k)}\boldsymbol{v}^k = \sum_{i_1, i_2, \ldots, i_k = 1}^{m} \mathcal{A}_{i_1 i_2 \ldots i_k} v_{i_1} v_{i_2} \ldots v_{i_k}, \tag{22}$$

where the notation $\boldsymbol{v}^k$ refers to the repetition of $\boldsymbol{v}$ for k times. The summation is only performed over indices that are unique to the right-hand side of the equation. With this knowledge, we can now re-write Eq. (21) using multi-linear forms

$$\mathcal{T}^n_{f(y)}(\boldsymbol{y}) = f(\boldsymbol{y}^*) + \sum_{k=1}^{n} \boldsymbol{\mathcal{F}}^{(k)}(\boldsymbol{y} - \boldsymbol{y}^*)^k, \tag{23}$$

where $\boldsymbol{\mathcal{F}}^{(k)} \in \mathbb{S}^{m^k}$ is the kth-order tensor that collects the kth-order derivatives with respect to $\boldsymbol{y}$.

If the multiplication by the first tensor mode is eliminated, we obtain a row-vector output,[4] indicated as $\boldsymbol{\mathcal{A}}^{(k)}\boldsymbol{v}^{k-1} \in \mathbb{R}^m$. Its definition is as follows

$$(\boldsymbol{\mathcal{A}}^{(k)}\boldsymbol{v}^{k-1})_j = \sum_{i_2, \ldots, i_k = 1}^{m} \mathcal{A}_{j i_2 i_3 \ldots i_k} v_{i_2} v_{i_3} \ldots v_{i_k}, \tag{24}$$

which is valid for each element of the row $j \in \{1, \ldots, m\}$.

4.2. Recursive method

The polynomials $\mathcal{T}^n_{h_c}(\boldsymbol{\Phi})$ from Eq. (20b) are reformulated using the multi-linear forms introduced in Section 4.1 for $c \in \{1, \ldots, n_{eq}\}$

$$\mathcal{T}^n_{h_c}(\boldsymbol{\Phi}) = h_c^0 + \sum_{k=1}^{n} \boldsymbol{\mathcal{F}}_c^{(k)} \boldsymbol{\Phi}^k, \tag{25a}$$

$$\boldsymbol{\mathcal{F}}_c^{(k)} = \frac{1}{k!} \frac{\partial^k h_c}{\partial \boldsymbol{\Phi}^k} \tag{25b}$$

$$h_c^0 = \mathcal{T}^0_{h_c}(\boldsymbol{\Phi}) \tag{25c}$$

where $h_c^0 \in \mathbb{R}$ is the value of h_c in the unperturbed trajectory, i.e., the 0th-order of the Taylor polynomial; $\boldsymbol{\mathcal{F}}_c^{(k)} \in \mathbb{S}^{M^k}$ - for $k \in \{1, \ldots, n\}$ - are symmetric kth-order tensors that represent the contributions of all the kth-order partial derivatives of h_c with respect to $\boldsymbol{\Phi}$. The number of dimensions of the tensor $\boldsymbol{\mathcal{F}}_c^{(k)}$ is equal to the order of the associated polynomial term, e.g., $\boldsymbol{\mathcal{F}}_c^{(2)} \in \mathbb{R}^{M^2}$, $\boldsymbol{\mathcal{F}}_c^{(3)} \in \mathbb{R}^{M^3}$, and so on. Let us also define the constraint constant part $\rho_c \in \mathbb{R}$ as

$$\rho_c := \bar{h}_c - h_c^0 \quad c \in \{1, \ldots, n_{eq}\} \tag{26}$$

The same rearranging in Eq. (25) and Eq. (26) can be performed for the inequality constraints in Eq. (20c), yielding $\mathcal{T}^n_{g_v}(\boldsymbol{\Phi})$, $\boldsymbol{\mathcal{F}}_v^{(k)}$, g_v^0, and ρ_v for $v \in \{1, \ldots, n_{in}\}$. Therefore, Problem (20) is equivalent to

$$\min_{\boldsymbol{\Phi}} \quad \boldsymbol{\Phi}^\top \boldsymbol{\Phi} \tag{27a}$$

$$\text{s.t.} \quad \sum_{k=1}^{n} \boldsymbol{\mathcal{F}}_v^{(k)} \boldsymbol{\Phi}^k \leq \rho_v, \qquad v \in \{1, \ldots, n_{in}\} \tag{27b}$$

$$\sum_{k=1}^{n} \boldsymbol{\mathcal{F}}_c^{(k)} \boldsymbol{\Phi}^k = \rho_c, \qquad c \in \{1, \ldots, n_{eq}\}. \tag{27c}$$

The idea of the solution method is to successively linearise the high-order polynomials that constitute the constraints. So, Problem (27) is initially linearised by truncating the polynomials at the first order

$$\min_{\boldsymbol{\Phi}} \quad \boldsymbol{\Phi}^\top \boldsymbol{\Phi} \tag{28a}$$

$$\text{s.t.} \quad \boldsymbol{\mathcal{F}}_v^{(1)} \boldsymbol{\Phi} \leq \rho_v, \qquad v \in \{1, \ldots, n_{in}\} \tag{28b}$$

$$\boldsymbol{\mathcal{F}}_c^{(1)} \boldsymbol{\Phi} = \rho_c, \qquad c \in \{1, \ldots, n_{eq}\}. \tag{28c}$$

This problem can be solved using any quadratic programme method, like the interior-point, or the active set methods, in polynomial time. The solution of Problem (28) is called $[\boldsymbol{\Phi}]_1 \in \mathbb{R}^M$.

Now, the method is generalisable to higher orders using the tensor notation. Let us assume that a greedy solution has been found for the PP with constraints truncated at the $(j-1)$th order. Problem (27) approximated with constraints that include polynomials of order up to j

$$\min_{\boldsymbol{\Phi}} \quad \boldsymbol{\Phi}^\top \boldsymbol{\Phi} \tag{29a}$$

$$\text{s.t.} \quad \sum_{k=1}^{j} \boldsymbol{\mathcal{F}}_v^{(k)} \boldsymbol{\Phi}^k \leq \rho_v, \qquad v \in \{1, \ldots, n_{in}\} \tag{29b}$$

$$\sum_{k=1}^{j} \boldsymbol{\mathcal{F}}_c^{(k)} \boldsymbol{\Phi}^k = \rho_c, \qquad c \in \{1, \ldots, n_{eq}\}. \tag{29c}$$

An iterative process is used to refine the solution. In the first iteration, the linearisation point is the output of the previous order $\tilde{\boldsymbol{\Phi}} = [\boldsymbol{\Phi}]_{j-1}^{b_{end}}$.

[2] https://yalmip.github.io/tutorial/globaloptimisation/.

[3] https://au.mathworks.com/help/optim/ug/fmincon.html.

[4] In the literature it is often assumed to be a column-vector, but a row-vector is preferred for the purpose of this work since it will be used to define a pseudo-gradient.

To express the linearised polynomials, we make use of the multi-linear mode with the exclusion of the first tensor mode defined in Eq. (24)

$$\sum_{k=1}^{j}\left(\mathcal{F}_{v}^{(k)}\tilde{\boldsymbol{\Phi}}^{k-1}\right)\boldsymbol{\Phi} \le \rho_v \quad v \in \{1,\ \dots,\ n_{in}\} \tag{30a}$$

$$\sum_{k=1}^{j}\left(\mathcal{F}_{c}^{(k)}\tilde{\boldsymbol{\Phi}}^{k-1}\right)\boldsymbol{\Phi} = \rho_c \quad c \in \{1,\ \dots,\ n_{eq}\}. \tag{30b}$$

(30) is a linear function because $\tilde{\boldsymbol{\Phi}}$ is a known value. Therefore, it is possible to define the jth-order *pseudo-gradient* of each constraint which includes the contribution of the high-order terms

$$\boldsymbol{\gamma}_c^{(j)} := \mathcal{F}_{c}^{(1)} + \sum_{k=2}^{j}\mathcal{F}_{c}^{(k)}\tilde{\boldsymbol{\Phi}}^{k-1} \tag{31a}$$

$$\boldsymbol{\gamma}_v^{(j)} := \mathcal{F}_{v}^{(1)} + \sum_{k=2}^{j}\mathcal{F}_{v}^{(k)}\tilde{\boldsymbol{\Phi}}^{k-1} \tag{31b}$$

which is valid for $v \in \{1,\ \dots,\ n_{in}\}$ and $c \in \{1,\ \dots,\ n_{eq}\}$; each $\boldsymbol{\gamma}_c^{(j)}$ is a row vector, as is each $\boldsymbol{\gamma}_v^{(j)}$. Now, one can write the QP with linearised constraints in the same form as Problem (28)

$$\min_{\boldsymbol{\Phi}} \quad \boldsymbol{\Phi}^\top\boldsymbol{\Phi} \tag{32a}$$

$$\text{s.t.} \quad \boldsymbol{\gamma}_v\boldsymbol{\Phi} \le \rho_v, \qquad v \in \{1,\ \dots,\ n_{in}\} \tag{32b}$$

$$\boldsymbol{\gamma}_c\boldsymbol{\Phi} = \rho_c, \qquad c \in \{1,\ \dots,\ n_{eq}\}. \tag{32c}$$

The solution of Problem (32) can be obtained with the same method as Problem (28), and it is called is $[\boldsymbol{\Phi}]_j^b \in \mathbb{R}^M$. This becomes the new linearisation point for the high-order polynomials

$$\tilde{\boldsymbol{\Phi}} = [\boldsymbol{\Phi}]_j^b. \tag{33}$$

From here onwards, the linearisations are repeated, yielding new QPs in the form of Problem (32) until a certain tolerance $e_{tol} \in \mathbb{R}_+$ is respected on the convergence at the order j. Convergence is checked by evaluating the norm of the difference between the solution of iteration b and the previous one, $b-1$

$$\|[\boldsymbol{\Phi}]_j^b - [\boldsymbol{\Phi}]_j^{b-1}\|_2 \le e_{tol}. \tag{34}$$

If Eq. (34) is not satisfied, the procedure is repeated from Eq. (31) to Eq. (33). When convergence within the iterations is reached, the truncation order is increased up to the original order of the PP n.

The power of the method mainly relies on the fact that, although being a sequential linearisation approach, the polynomial maps are computed only once. So, there is no need for multiple computationally expensive numerical propagations. This distinguishes it from other classic sequential methods, like SCP, in which the reference point is used to progressively update the expansion point for the linear maps.

Algorithm 1 Pipeline for recursive polynomial CAM design.

Get parameters for the scenario.
Define maximum polynomial expansion order n.
Define the constraints
Define the control dynamics: low-thrust or multi-impulsive.
Discretise the time domain into $\boldsymbol{t} = \{t_0,\ \dots,\ t_f\}$.
if low-thrust **then**
  $\boldsymbol{\phi}_i \leftarrow \boldsymbol{u}_i$, for $i \in \{0,\ \dots,\ N_{man}-1\}$
else
  $\boldsymbol{\phi}_i \leftarrow \boldsymbol{\Delta v}_i$, for $i \in \{0,\ \dots,\ N_{man}-1\}$
end if
$\boldsymbol{\Phi} \leftarrow [\boldsymbol{\phi}_0^\top,\ \dots,\ \boldsymbol{\phi}_{N_{man}-1}^\top]^\top$
$\boldsymbol{x}_0 \leftarrow \boldsymbol{f}(\boldsymbol{x}_{CA,1}, t_{CA,1}, t_0)$
for $i = 1 : N$ **do**
  $\boldsymbol{x}_i \leftarrow \boldsymbol{f}(\boldsymbol{x}_{i-1}, \boldsymbol{\phi}_{i-1}, t_{i-1}, t_i)$
end for
for $v = 1 : n_{in}$ **do**
  $\mathcal{T}_{g_v}^n(\boldsymbol{\Phi}), \mathcal{T}_{h_c}^n(\boldsymbol{\Phi}) \leftarrow$ Eq. (25a)
  $g_v^0, h_c^0 \leftarrow$ Eq. (25c)
  $\rho_v, \rho_c \leftarrow$ Eq. (26)
  for $k = 1 : n$ **do**
    $\mathcal{F}_v^{(k)}, \mathcal{F}_c^{(k)} \leftarrow$ Eq. (25b)
  end for
end for
$[\boldsymbol{\Phi}]_1 \leftarrow$ Solution of Problem (28)
$\tilde{\boldsymbol{\Phi}} \leftarrow [\boldsymbol{\Phi}]_1$
for $j \in \{2,\ \dots,\ n\}$ **do**
  $b \leftarrow 0$
  $e \leftarrow 1$
  while $e \le e_{tol}$ && $b \le b_{max}$ **do**
    $\boldsymbol{\gamma}_c^{(j)}, \boldsymbol{\gamma}_v^{(j)} \leftarrow$ Eq. (31)
    $[\boldsymbol{\Phi}]_j^b \leftarrow$ Solution of Problem (32)
    $\tilde{\boldsymbol{\Phi}} \leftarrow [\boldsymbol{\Phi}]_j^b$
    $e \leftarrow \|\tilde{\boldsymbol{\Phi}} - [\boldsymbol{\Phi}]_j^{b-1}\|_2$
    $b \leftarrow b+1$
  end while
end for
$\tilde{\boldsymbol{\phi}}_i \leftarrow \tilde{\boldsymbol{\Phi}}_{opt}$
Validate the constraints values from forward-propagating the trajectory from t_0 to t_f using $\tilde{\boldsymbol{\phi}}_i$.

5. Applications

The method described in Section 4 is applicable to any fuel-optimal problem like Problem Eq. (14), where the single scalar constraints can be approximated using a convergent Taylor series, leading to Problem (20). This work's main focus is on CAMs, but nothing prohibits applying the method to other problems. In the following, the method will be applied to multiple CAMs and to scenarios where the CAM is combined with SK requirements.

5.1. Single encounter

When just the CAM is of interest, Problem (20) simplifies to the single constraint case

$$\min_{\boldsymbol{\Phi}} \quad \boldsymbol{\Phi}^\top\boldsymbol{\Phi} \tag{35a}$$

$$\text{s.t.} \quad \mathcal{T}_{P_C}^n(\boldsymbol{\Phi}) = \bar{P}_C. \tag{35b}$$

In this case, there is no need to use a state-of-the-art solver to find the solution to the sequential QPs: it is sufficient to impose a control that is parallel to the (pseudo-)gradient of PoC

$$[\boldsymbol{\Phi}]_j^b = \frac{\rho}{\|\boldsymbol{\gamma}^{(j)}\|}\hat{\boldsymbol{\gamma}}^{(j)} \tag{36}$$

This case was extensively analysed in [18], considering impulsive, multi-impulsive, and low-thrust dynamics and different thrusting opportunities. The constraint is set as an equality leveraging the conclusions from multiple studies [10,11,14], which show that for a single fast encounter, the relative position at TCA ends up on the border of the iso-probability ellipse.

5.2. Multiple encounters

When multiple consecutive encounters are considered, two options are available to design the CAM. Either we control TPoC as defined in Eq. (8), which allows us to keep the formulation from Problem (35), or we consider a separate PoC for the different encounters, which results in the following problem

$$\min_{\boldsymbol{\Phi}} \quad \boldsymbol{\Phi}^\top\boldsymbol{\Phi} \tag{37a}$$

$$\text{s.t.} \quad \mathcal{T}^{n}_{P_{C,s}}(\boldsymbol{\Phi}) \leq \bar{P}_C, \quad s \in \{1, \ldots, n_{conj}\} \tag{37b}$$

As shown in [16], the first approach is theoretically more rigorous since the overall combined risk is taken into account. Nonetheless, PoC from Eq. (8) encompasses a much higher degree of nonlinearity than that of a single encounter. So, the method can become inaccurate even at very high-orders. This problem is solved by using the formulation in Problem (37), where each conjunction is treated as a separate constraint, and the combined effect of the multiple conjunctions on TPoC is not taken into account. In this case, it is necessary to set inequality constraints because the manoeuvre performed to avoid one conjunction is likely to have a mitigating effect on the others. So, it is not necessarily true that in multiple conjunctions scenarios, the PoC threshold of the single conjunctions is respected with the equality sign.

Lastly, let us consider mitigating the risk using MD, which is achieved by expanding Eq. (9)

$$\min_{\boldsymbol{\Phi}} \quad \boldsymbol{\Phi}^\top \boldsymbol{\Phi} \tag{38a}$$

$$\text{s.t.} \quad \mathcal{T}^{n}_{-d^2_{miss,s}}(\boldsymbol{\Phi}) \leq -\bar{d}_s^2, \quad s \in \{1, \ldots, n_{conj}\}, \tag{38b}$$

where the constraints are imposed on the negative value of the squared MD for consistency with the other formulations.

5.3. CAM with station-keeping on osculating orbital elements

Another useful application of the method consists of coupling the CAM with the return to the nominal orbit constraints. Utilising the automatic Taylor expansion, Eq. (10) allows us to define the CAM PP with the five additional SK constraints

$$\min_{\boldsymbol{\Phi}} \quad \boldsymbol{\Phi}^\top \boldsymbol{\Phi} \tag{39a}$$

$$\text{s.t.} \quad \mathcal{T}^{n}_{P_{C,s}}(\boldsymbol{\Phi}) \leq \bar{P}_C, \qquad s \in \{1, \ldots, n_{conj}\} \tag{39b}$$

$$\mathcal{T}^{n}_{\varepsilon_a}(\boldsymbol{\Phi}) = 0, \tag{39c}$$

$$\mathcal{T}^{n}_{\varepsilon_e}(\boldsymbol{\Phi}) = 0, \tag{39d}$$

$$\mathcal{T}^{n}_{\varepsilon_i}(\boldsymbol{\Phi}) = 0, \tag{39e}$$

$$\mathcal{T}^{n}_{\varepsilon_\omega}(\boldsymbol{\Phi}) = 0, \tag{39f}$$

$$\mathcal{T}^{n}_{\varepsilon_\Omega}(\boldsymbol{\Phi}) = 0. \tag{39g}$$

If needed, Eq. (39b) can be substituted by Eq. (38b).

5.4. CAM with station-keeping on mean orbital elements

Another approach consists of limiting the deviation from the nominal mean semi-major axis and eccentricity

$$\min_{\boldsymbol{\Phi}} \quad \boldsymbol{\Phi}^\top \boldsymbol{\Phi} \tag{40a}$$

$$\text{s.t.} \quad \mathcal{T}^{n}_{P_{C,s}}(\boldsymbol{\Phi}) \leq \bar{P}_C, \quad s \in \{1, \ldots, n_{conj}\} \tag{40b}$$

$$\mathcal{T}^{n}_{\varepsilon_{sma}}(\boldsymbol{\Phi}) \leq \Delta a^2_{max}, \tag{40c}$$

$$\mathcal{T}^{n}_{\varepsilon_{ecc}}(\boldsymbol{\Phi}) \leq \Delta e^2_{max}, \tag{40d}$$

where the SK constraints Eqs. (40c) and (40d) are derived from the polynomial expansion of Eq. (13) with respect to the control history. Again, if needed, Eq. (40b) can be replaced by Eq. (38b).

6. Results

In this section, some exemplary applications of the method are showcased. In the first two applications, a campaign of simulations where a single conjunction in LEO is coupled with a SK constraint is considered. In the third one, a scenario in which multiple consecutive conjunctions take place is addressed. The fourth one consists of a scenario where multiple conjunctions are followed by the SK. In Table 1, the parameters used in the optimisation are shown: Although the MD threshold might be too low for some common operational pipelines, its value leads to the generation of manoeuvres of similar magnitude to PoC simulations. All the simulations are run on an 11th Gen Intel(R) Core(TM) i7-11700 @2.5 GHz, 2496 Mhz, 8 Cores, 16 Logical Processors, using MATLAB R2024a and C++.

Table 1
Simulation parameters.

$\bar{P}_C$ [–]	$\bar{d}$ [km]	e_{tol} [mm/s]	b_{max} [–]
10^{-6}	0.5	0.1	200

6.1. CAM with station-keeping on osculating orbital elements

The ESA Collision Avoidance Challenge database, utilised in Refs. [14,18], is used to assess the validity of the method when the CAM is coupled with SK. A total of 2170 cases are studied. The manoeuvre is bi-impulsive: since the method is limited by an *a priori* selection of the impulse times particular care must be devoted to this aspect. In a single-encounter scenario, it is common practice to fire at odd multiples of half-orbits before the conjunction, granting the minimum fuel expenditure [9]. In this study, the first impulse is placed 0.5 orbits before TCA to ensure comparability with earlier work [18]. A second impulse is enforced 0.5 orbits after conjunction to satisfy the SK condition, as the spacecraft is then approximately co-located with the first impulse point. While this timing is consistent with previous literature and enables a clear demonstration of the method's performance, we acknowledge that it is not representative of standard operational practice. In reality, operators often seek longer lead times—ideally several orbital periods before TCA—to allow for robust re-tracking and re-planning in case of manoeuvre execution uncertainties or tracking errors. However, delays are frequently tolerated in the hope that accurate orbit determination might passively mitigate the risk [26]. The proposed framework remains general and can accommodate alternative manoeuvre timings based on specific operational constraints or policies.

Formally, we solve Problem (39) with an original 2nd-order expansion of the constraints. In Figs. 3–6, the summary of the obtained solutions is given using statistical distributions. The total Δv in most scenarios is below 50 mm/s, but it can reach values of up to 1 m/s in extreme cases. The first impulse used to perform the CAM is counterbalanced by the next one, which is used to reinsert the satellite into the nominal trajectory. The constraint violation is evaluated in Fig. 4 for PoC and in Fig. 5 for the return conditions. The latter is the difference between the target and the reached orbital element, the former is computed as

$$\varepsilon_{P_C} = \left| \frac{P_C}{\bar{P}_C} - 1 \right|.$$

PoC is typically targeted very precisely, with a maximum absolute error of 4×10^{-9}. The return errors are also acceptable even at an operation level, being always below 2 cm and 0.15 mm/s. The computation time, shown in the left part of Fig. 6, highlights the fact that the method is also computationally efficient because its 95th percentile is always below 0.15 s. For comparison, we used MATLAB's fmincon to directly solve the second-order form of Problem (39): on average, the recursive method is 30% faster. Moreover, the spread of the distribution of computation times is much lower than that of fmincon.

6.2. CAM with station keeping on mean orbital elements

In this section, the same test cases from the previous section are studied using the same bi-impulsive manoeuvre scheme, with the first Δv happening 0.5 orbits before TCA, and the second 0.5 orbits after

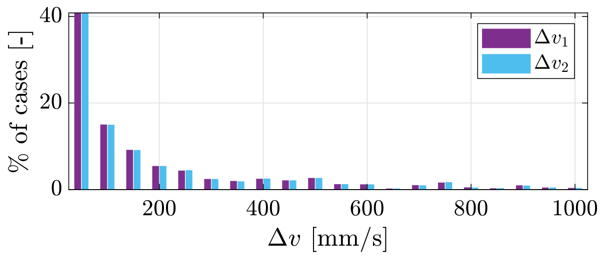

Fig. 3. Distribution of the magnitude of the bi-impulsive Δv for Problem (39).

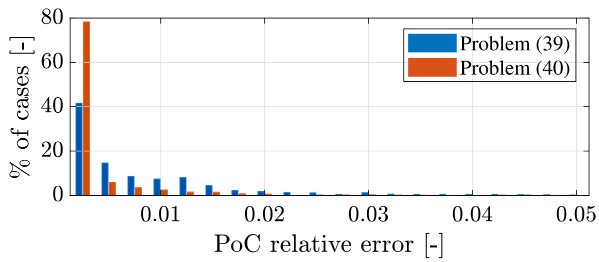

Fig. 4. Distribution of the PoC relative error.

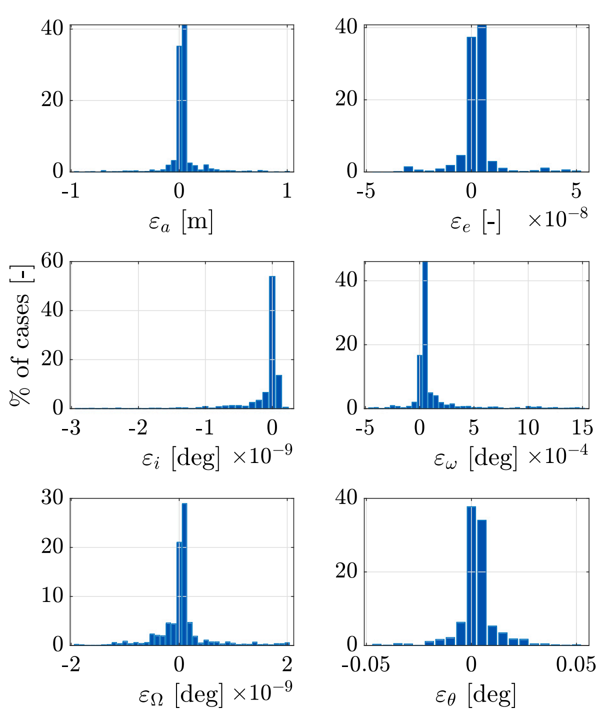

Fig. 5. Distribution of the SK error for Problem (39).

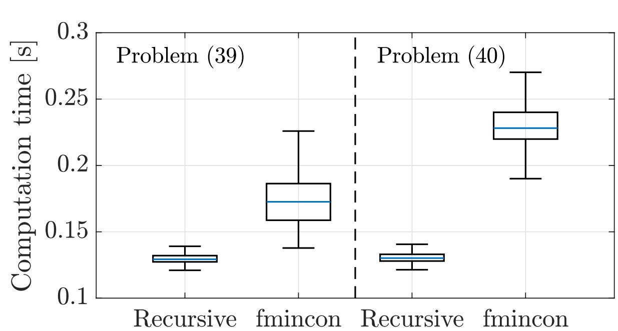

Fig. 6. Box plot of the computation times limited to the 95*th* percentiles.

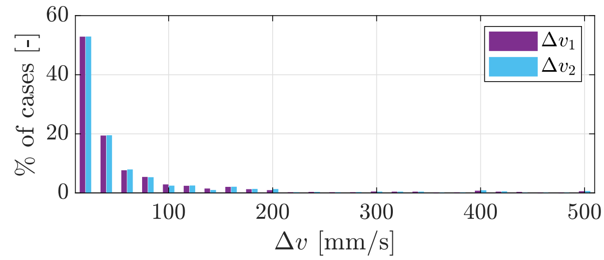

Fig. 7. Distribution of the bi-impulsive Δv for Problem (40).

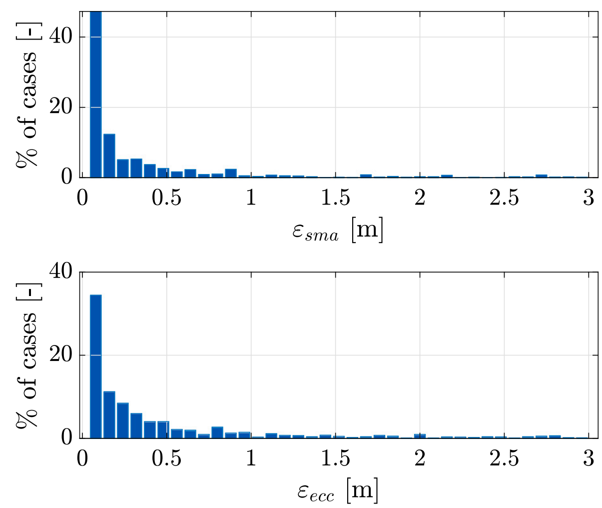

Fig. 8. Distribution of the SK error for Problem (40).

TCA. After 1 orbits from TCA, the spacecraft must respect the requirements on mean eccentricity and mean semi-major axis, as expressed in Problem (40), with $\Delta a_{max} = 3$ m and $\Delta e_{max} = 3$ m. Clearly, the requirement of controlling the two mean orbital elements is less stringent than controlling the whole final state, so the total computed Δv is typically lower: in Fig. 7, one can see its distribution. Similar considerations to those drawn in the previous section apply to the PoC relative error, which is typically slightly lower, and the SK constraint violations, which are always null in this case. They are shown in Figs. 4 and 8, respectively. The computation time, whose distribution is shown in the right-hand side of Fig. 6, is comparable to the case of Problem (39). As in the previous case, the method outperforms fmincon's interior-point method, used to solve Problem (40) directly: on average, the recursive method is 40% faster. Considerations on the convergence of the method are drawn in Section 6.5.

Taking as reference case study conjunction #1 from the database, we perform a preliminary analysis to identify the optimal manoeuvring point to respect the MD (with a threshold of 2 km) and the return constraints. This analysis is to showcase the potential of the method not only for autonomous manoeuvre generation, but also as a conjunction analysis tool. To simplify the analysis, the return time is always set to one orbit, and the second impulse takes place 0.5 orbits after conjunction. Clearly, looking at Fig. 9, earlier CAMs allow for the most fuel savings, and, as expected, the best firing windows are at 0.5, 1.5, 2.5, 3.5, 4.5, and 5.5 orbital periods before conjunction; the worst firing opportunities, instead, are around the 3/4 of the orbital period, as they necessitate significant out-of-plane components. Fig. 10 shows that the MD constraint at TCA is consistently respected with an acceptable accuracy, reaching a maximum relative error of 0.1% for a manoeuvre 1 orbit before conjunction. The SK requirement, however, is only met with reasonable accuracy when the first manoeuvre occurs at half-orbit intervals. This behaviour aligns with classical trajectory optimisation

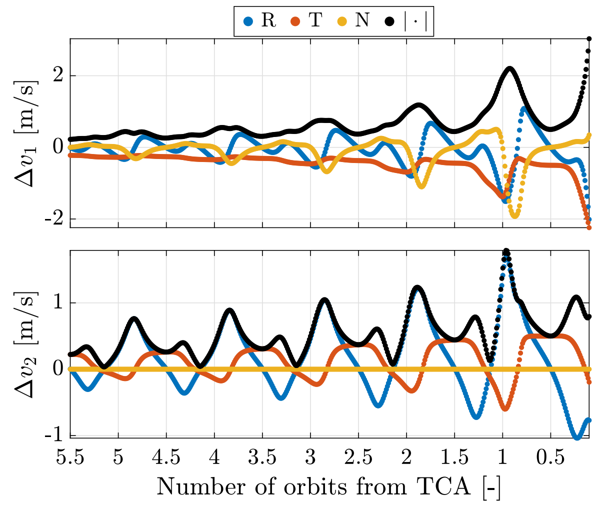

Fig. 9. Δv before (top) and after (bottom) TCA.

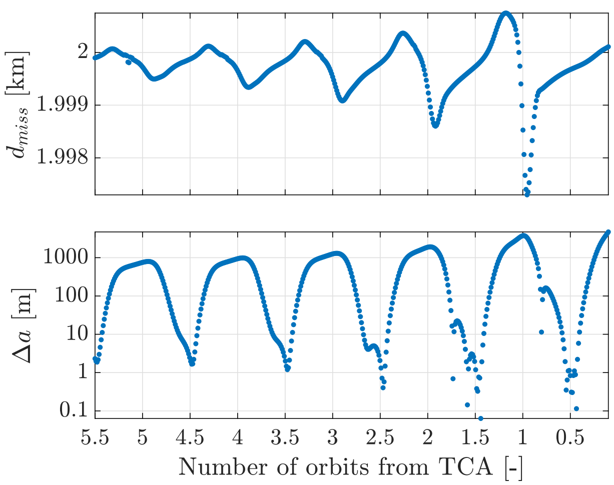

Fig. 10. MD at TCA and mean semi-major axis error at the return time.

principles, as a bi-impulsive manoeuvre requires the two impulses to be applied at the same point in the orbit to return to the original state. If the impulses are not spaced exactly one orbital period apart, a third correction manoeuvre is typically needed to more accurately shape the spacecraft's final orbit. Therefore, from this analysis it appears reasonable to assume that the first Δv has to happen at any half-orbit before conjunction – the sooner the more optimal – and the second one can occur any half-orbit after TCA.

6.3. Multiple encounters

Let us consider a primary satellite in low Earth orbit (LEO), which undergoes either two or three consecutive encounters with different secondary objects. The ECI position and velocity of the primary satellite at the first conjunction are

$$\boldsymbol{r}_{p,CA_1} = [6776.606 \quad -866.862 \quad -1150.364]^{\top}\ \text{km},$$
$$\boldsymbol{v}_{p,CA_1} = [1.577044 \quad 4.465112 \quad 5.925404]^{\top}\ \text{km/s}.$$

In Table 2, the ECI coordinates of the secondaries at the corresponding conjunctions are reported, along with the elapsed time from the first TCA. The state of the primary at each conjunction is obtained using Keplerian propagation. The combined HBR of each conjunction is assumed for simplicity equal to 6 m. The combined positional covariances in RTN components of the primary for each conjunction are reported in Table 3.

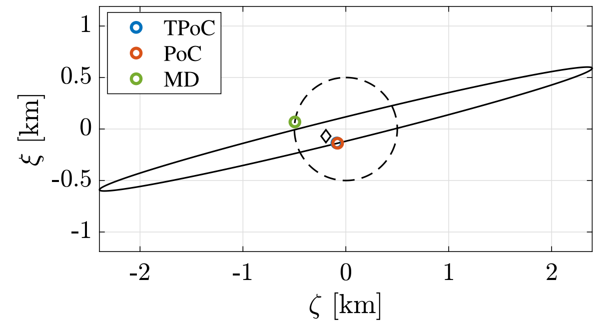

Fig. 11. B-plane of the first conjunction.

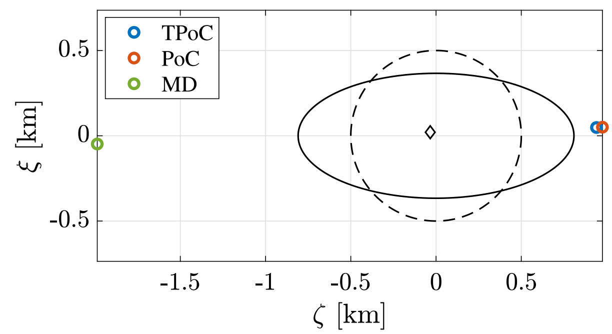

Fig. 12. B-plane of the last conjunction.

The problem can be solved using the formulation in Problem (35), Problem (37) or Problem (38). In the first case, Eq. (8) is used, and the conjunctions are all included in the TPoC constraint. In the last case, the threshold value is set to $\bar{d} = 0.5$ km. Here, a 4th-order expansion is used to build the PPs. First, let us consider only the first and the last conjunctions, with a single impulse opportunity 0.5 orbits before TCA.

The manoeuvre computed using each of the considered collision metrics is reported in Table 4. PoC and TPoC yield very similar results, but MD gives a higher Δv because the Δv is proportional to the miss distance threshold. The B-planes of the two conjunctions are shown in Figs. 11 and 12, where the iso-probability ellipse corresponding to the PoC threshold of 10^{-6} is shown as a continuous black line, while the dashed line is the MD threshold. As expected, since they produce the same manoeuvre, the first two methods reach almost the same points in both B-planes, while the manoeuvre computed using MD leads to the opposite direction. In all three cases, the only active constraint is the one of the first conjunction since in the second B-plane, all the methods yield a trajectory that passes far from the keep-out-zone limit.

6.4. Multiple encounters with station-keeping on mean orbital elements

Lastly, let us consider the most complex problem for our application, which involves three multiple consecutive conjunctions and the SK constraint with mean orbital elements. The scenario with three conjunctions described in the previous section is studied, with three continuous thrust opportunities spanning 0.2 orbits and centred at 0.5 orbits before the first TCA and 0.5 and 1.5 orbits after it. The return condition must be achieved 3 orbits after the first TCA. We study the problem with the formulation in Problem (40) using either PoC or MD. The dynamics are low-thrust, and the expansion order is 4. In this case, the tolerance to exit the linearisation iterations is 0.01 mm/s^2.

In Figs. 13 and 14, the acceleration profile output by the two optimisations is shown. The former only uses two of the available thrust

Table 2
ECI state, HBR and conjunction time for the consecutive encounters.

Conj	$\boldsymbol{x}_s$ [km]/[km/s]						TCA [s]
1	[6776.528	−866.842	−1150.571	−1.208085	−5.314340	4.604883]$^{\mathrm{T}}$	0.0
2	[6890.038	−435.868	−578.300	−3.48320	5.177411	−5.541674]$^{\mathrm{T}}$	11 573.30
3	[−2817.863	3808.910	5054.577	6.590243	0.106133	−3.753949]$^{\mathrm{T}}$	15 590.47

Table 3
Combined covariance components in RTN coordinates of the secondary; The unit of measure is [km^2].

Conj	P_{rr}	P_{tt}	P_{nn}	P_{rt}	P_{tn}	P_{nr}
1	1×10^{-3}	0.8	4×10^{-5}	0	0	0
2	2×10^{-3}	0.6	4×10^{-4}	0	0	0
3	0.01	0.05	5×10^{-4}	0	0	0

Table 4
Δv of the multiple conjunction scenario with different collision metrics in [mm/s].

Metric	Δv_R	Δv_T	Δv_N	$\|\|\Delta v\|\|$
TPoC	2.3	19.0	0.0	19.1
PoC	1.2	18.4	0.1	18.4
MD	−15.1	−38.4	0.0	41.4

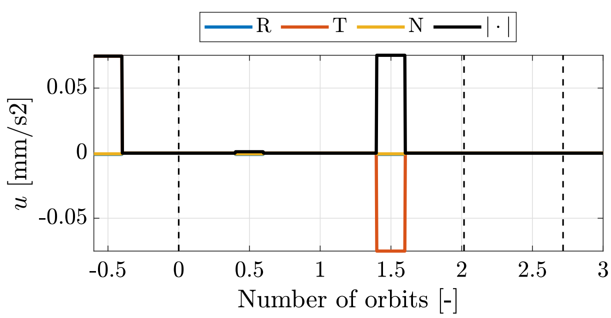

Fig. 13. Acceleration profile for the PoC simulation with SK constraints.

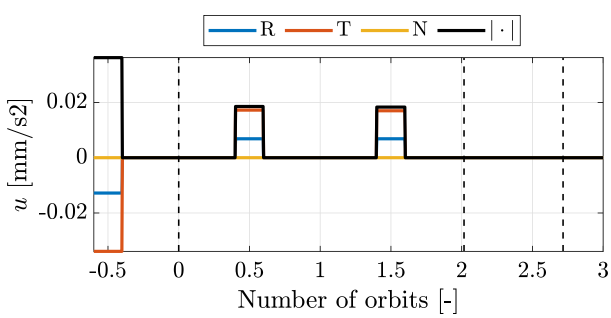

Fig. 14. Acceleration profile for the MD simulation with SK constraints.

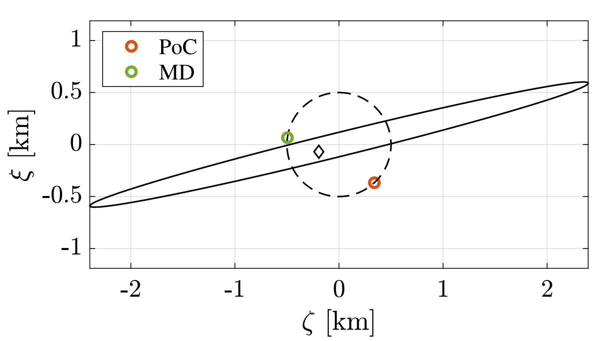

Fig. 15. B-plane of the first conjunction with SK constraints.

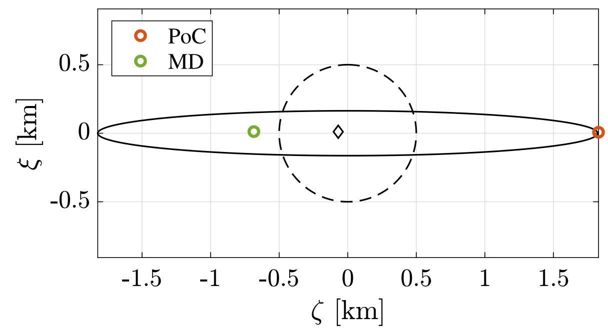

Fig. 16. B-plane of the second conjunction with SK constraints.

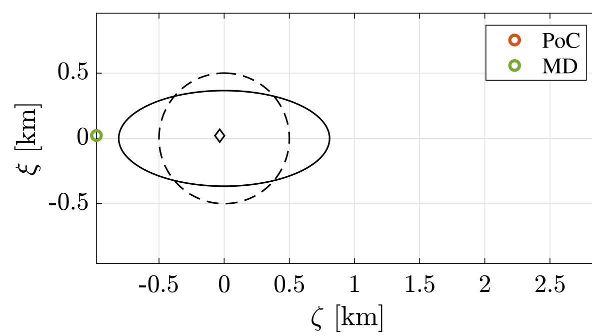

Fig. 17. B-plane of the last conjunction with SK constraints.

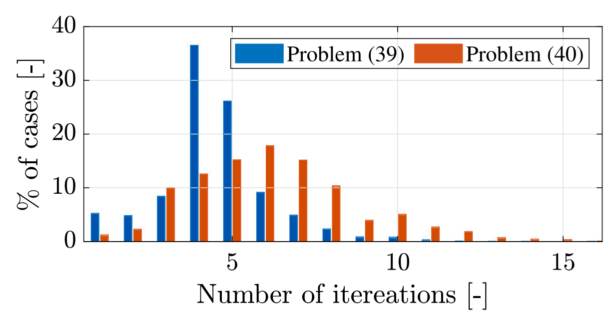

Fig. 18. Number of iterations distribution for the simulations in Sections 6.1 and 6.2.

opportunities, while the latter uses all of them. These are not bang–bang profiles, but, assuming a maximum throttle is given, they can be regularised by adjusting the thrust durations. Likewise, assuming a minimum operational throttle is given, the windows for which the control magnitude falls below the threshold – e.g. the second window of Fig. 13 – can be filtered out. Interestingly, in this case, the total thrust required by the PoC simulation is higher than the MD. Indeed, as can be seen in Fig. 15 and Fig. 16, the constraint of the first conjunction is inactive and the second conjunction is active, with the optimised point lying on the semi-major axis of the ellipse, in which the gradient of PoC is lowest. This means that the overall optimal manoeuvre causes a thrust in the direction of minimum change for PoC of the second conjunction. The constraints of the third conjunction, shown in Fig. 17, instead, are inactive, as in the simulation without SK. The SK constraints are respected with an accuracy of 11 mm for

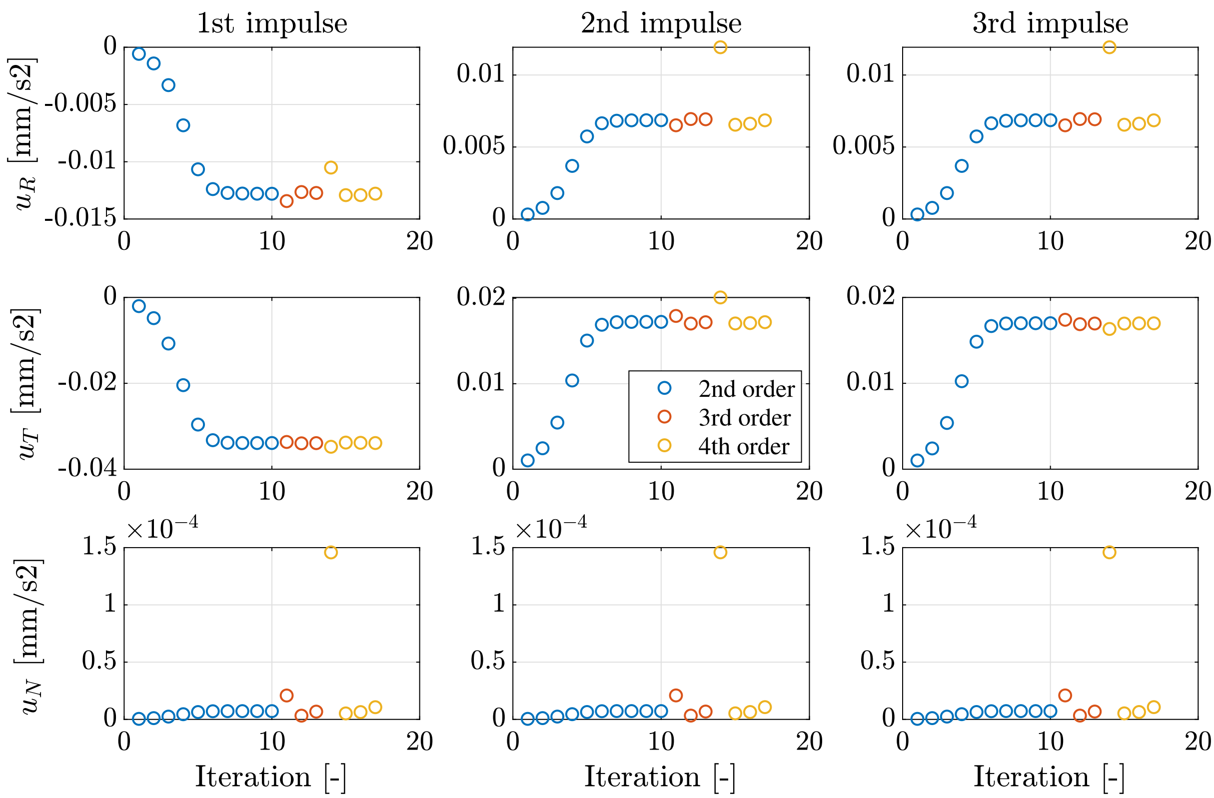

Fig. 19. Evolution of each component of the control acceleration across the iterations.

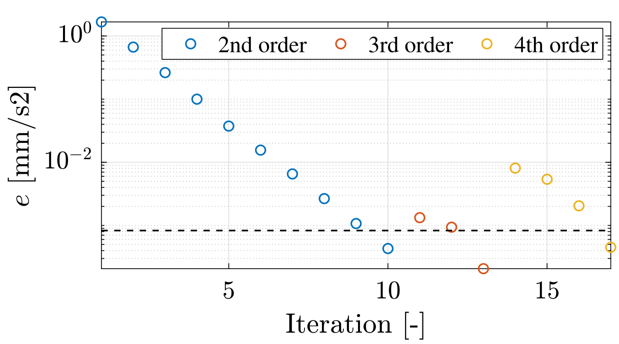

Fig. 20. Convergence profile of the simulation with MD, three thrust arcs and SK.

the semi-major axis and 3.9 m for the eccentricity. This simulation is particularly interesting because it shows that dealing with multiple conjunctions is not trivial, and it requires particular care in defining the manoeuvres. Indeed, while in the single short-term encounter, the optimal manoeuvre is oftentimes a pure tangential one performed at multiples of half-orbits before TCA, that is clearly not the case when multiple conjunctions are spread over a short time span. So, optimising the thrust is not a trivial challenge in these situations (see Fig. 15).

6.5. Convergence analysis

In this section, a sketch of the statistics of convergence of the algorithm is provided. This is not a proof of the assured convergence of the method, but it is an indication of its ability to solve the problems presented in this work. In Fig. 18, the convergence speed of the method applied to the problems from Sections 6.1 and 6.2 is shown: The formulation with osculating Keplerian elements is typically slightly faster in reaching convergence, but in general the number of iterations is very small (always below 15) for both formulations. The remaining simulations, from Sections 6.3 and 6.4, require between 14 and 33 iterations.

To exemplify the convergence process, in Figs. 19 and 20, the evolution of each component of the control acceleration and the convergence error are shown for the simulation with MD of Section 6.4. Most of the iterations are used to gradually shift the control components from the first-order solution to the second-order one, for which convergence is achieved around the 11th iteration. The third and fourth-order PPs yield very similar solutions to the second-order. So, in this case, a second-order expansion would be sufficient to obtain an accurate and effective manoeuvre.

7. Conclusions

A recursive method for the design of collision avoidance manoeuvres (CAMs) was presented based on the polynomial expansion of the nonlinear constraints. Differential algebra (DA) is used to automatically build polynomial representations of the required metrics as functions of the control. This allows us to build an energy-optimal polynomial programme (PP), which is iteratively approximated into a series of quadratic programmes that can be solved with state-of-the-art solvers in polynomial time. The method is generally applicable to a variety of scenarios involving multiple short-term encounters and operational SK constraints. Moreover, it can deal with an arbitrary number of manoeuvring windows, and it can account both for low-thrust and impulsive dynamics.

A number of different constraint combinations are addressed in the numerical simulations, including multiple short-term encounters, different collision metrics, and SK requirements. The numerical results show that the method can recover an accurate solution to the original PP in short computation times (<0.5 s) for the considered applications.

CRediT authorship contribution statement

Zeno Pavanello: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Laura Pirovano:** Writing – review & editing, Supervision. **Roberto Armellin:** Writing – review & editing, Supervision, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] ESOC (European Space Operations Centre), ESA's annual space environment report, Tech. rep., ESA Space Debris Office, 2025, ESA TR GEN-DBLOG-00288-OPS-SD.
[2] Z. Pavanello, L. De Maria, A. De Vittori, M. Maestrini, P. Di Lizia, R. Armellin, CAMmary: A review of spacecraft collision avoidance manoeuvre design methods, 2025, pp. 1–40, http://dx.doi.org/10.48550/arXiv.2503.22555, ArXiv.
[3] R.P. Patera, G.E. Peterson, Space vehicle maneuver method to lower collision risk to an acceptable level, J. Guid. Control Dyn. 26 (2) (2003) 233–237, http://dx.doi.org/10.2514/2.5063.
[4] S. Alfano, Collision Avoidance Maneuver Planning Tools, in: 15th AAS/AIAA Astrodynamics Specialist Conference, Lake Tahoe, California, 2005.
[5] S. Aida, Conjunction risk assessment and avoidance maneuver planning tools, 2016.
[6] I. Grande-Olalla, N. Sanchez-Ortiz, J.A. Pulido, K. Merz, Collision risk assesment and avoidance manoeuvres. New tool coram for esa, 2013, pp. 22–25.
[7] A. Morselli, R. Armellin, P. Di Lizia, F. Bernelli Zazzera, F. Bernelli-Zazzera, Collision avoidance maneuver design based on multi-objective optimization, in: AIAA/AAS Astrodynamics Specialist Conference 2014, vol. 152, San Diego, California, 2014, pp. 1819–1838.
[8] C. Bombardelli, Analytical formulation of impulsive collision avoidance dynamics, Celest. Mech. Dyn. Astron. 118 (2) (2014) 99–114, http://dx.doi.org/10.1007/s10569-013-9526-3.
[9] J. Hernando-Ayuso, C. Bombardelli, Low-thrust collision avoidance in circular orbits, J. Guid. Control Dyn. 44 (5) (2021) 983–995, http://dx.doi.org/10.2514/1.G005547.
[10] C. Bombardelli, J. Hernando-Ayuso, Optimal impulsive collision avoidance in low earth orbit, J. Guid. Control Dyn. 38 (2015) 217–225, http://dx.doi.org/10.2514/1.G000742.
[11] A. De Vittori, M.F. Palermo, P. Di Lizia, R. Armellin, Low-thrust collision avoidance maneuver optimization, J. Guid. Control Dyn. 45 (10) (2022) 1815–1829, http://dx.doi.org/10.2514/1.G006630.
[12] S. Mazouffre, Electric propulsion for satellites and spacecraft: Established technologies and novel approaches, Plasma Sources Sci. Technol. 25 (3) (2016) http://dx.doi.org/10.1088/0963-0252/25/3/033002.
[13] S. Dutta, A.K. Misra, Convex optimization of collision avoidance maneuvers in the presence of uncertainty, Acta Astronaut. 197 (2022) 257–268, http://dx.doi.org/10.1016/j.actaastro.2022.05.038.
[14] R. Armellin, Collision avoidance maneuver optimization with a multiple-impulse convex formulation, Acta Astronaut. 186 (2021) 347–362, http://dx.doi.org/10.1016/j.actaastro.2021.05.046, arXiv:2101.07403.
[15] Z. Pavanello, L. Pirovano, R. Armellin, Long-Term Encounters Collision Avoidance Maneuver Optimization with a Convex Formulation, in: 33rd AAS/AIAA Space Flight Mechanics Meeting, Austin, TX, 2023, pp. 1–20.
[16] Z. Pavanello, L. Pirovano, R. Armellin, A. De Vittori, P. Di Lizia, A convex optimization method for multiple encounters collision avoidance maneuvers, in: AIAA SciTech Forum, (January) Orlando, Florida, 2024, pp. 1–17, http://dx.doi.org/10.2514/6.2024-0845.
[17] Z. Pavanello, L. Pirovano, R. Armellin, A. De Vittori, P. Di Lizia, Collision avoidance maneuver optimization during low-thrust propelled trajectories, Astrodyn. (2025) 1–25, http://dx.doi.org/10.1007/s42064-024-0227-3.
[18] Z. Pavanello, L. Pirovano, R. Armellin, Recursive Polynomial Method for Fast Collision Avoidance Maneuver Design, IEEE Trans. Aerosp. Electron. Syst. 61 (2) (2025) 2914–2925, http://dx.doi.org/10.1109/TAES.2024.3480041.
[19] J.S. Li, Z. Yang, Y.Z. Luo, A review of space-object collision probability computation methods, Astrodyn. 6 (2) (2022) 95–120, http://dx.doi.org/10.1007/s42064-021-0125-x.
[20] K. Chan, International Space Station Collision Probability, Aerosp. Corp. Chantilly, VA, USA 6 (2) (2009) 307–314.
[21] R.C. Frigm, M.D. Hejduk, L.C.. Johnson, D. Plakalovic, Total probability of collision as a metric for finite conjunction assessment and collision risk management, in: Advanced Maui Optical and Space Surveillance Technologies (AMOS) Conference, NASA, Maui, Hawaii, 2015.
[22] H.D. Curtis, Orbital Mechanics for Engineering Students, third ed., Butterworth-Heinemann, Kundli, Haryana, India, 2014.
[23] S. D'Amico, O. Montenbruck, Proximity operations of formation-flying spacecraft using an eccentricity/inclination vector separation, J. Guid. Control Dyn. 29 (3) (2006) 554–563, http://dx.doi.org/10.2514/1.15114.
[24] R. Armellin, P. Di Lizia, F. Bernelli-Zazzera, M. Berz, Asteroid close encounters characterization using differential algebra: The case of Apophis, Celest. Mech. Dyn. Astron. 107 (4) (2010) 451–470, http://dx.doi.org/10.1007/s10569-010-9283-5.
[25] E.L. Jenson, D.J. Scheeres, Bounding nonlinear stretching about spacecraft trajectories using tensor eigenpairs, Acta Astronaut. 214 (October 2023) (2024) 159–166, http://dx.doi.org/10.1016/j.actaastro.2023.10.013.
[26] A. Zollo, Z. Pavanello, R. Armellin, J.F. San Juan Díaz, B. Schlepp, R. Kahle, Validation of a Fuel-Efficient Collision Avoidance Manoeuvre Optimiser for the GRACE-FO Mission, in: 75th International Astronautical Congress: IAC Proceedings, (October) IAF, Milan, Italy, 2024, pp. 1–14.