COLLISION AVOIDANCE MANEUVERS OPTIMIZATION WITH A MULTI-IMPULSE CONVEX FORMULATION

A PREPRINT

Roberto Armellin *
Te Pūnaha Ātea – Auckland Space Institute,
The University of Auckland
20 Symonds Street, 1010 Auckland, New Zealand

January 21, 2021

ABSTRACT

A method to compute optimal collision avoidance maneuvers for short-term encounters is presented. The maneuver is modeled as a multi-impulse one to handle impulsive cases or to approximate finite burn arcs associated either with short alert times or the use of low-thrust propulsion. The maneuver design is formulated as a sequence of convex optimization problems solved in polynomial time by state-of-the-art primal-dual interior-point algorithms. The proposed approach calculates optimal solutions without assumptions on thrust arc structure and thrust direction, with execution time compatible with onboard use.

1 Introduction

The number of catalogued Resident Space Object (RSO) is growing due to miniaturization of spacecraft (e.g., Cube-Sats) and the launch of mega-constellations (e.g., Starlink [1]). In parallel, the number of tracked space debris is expected to increase due to the improvement in tracking systems (e.g., the space fence system [2]), resulting in a larger number of conjunctions to be processed and more Collision Avoidance Maneuver (CAM)s designed and executed. Conjunction analysis and collision avoidance are currently performed by agencies and operators on the ground using several tools and processes that were developed over the last twenty years [3]. These tools support operators’ activities; however, the decision process still requires human intervention. This approach will not be practical in the future when conjunctions screening, collision avoidance decision process, and CAM design and execution need to be automated. Additionally, performing as many of these tasks onboard represents an enabling technology for safer and cheaper space missions. Furthermore, more accurate conjunction services (i.e., limiting unnecessary maneuvers) and optimal CAMs will be required to reduce the propellant budget allocated for conjunction management. Within this context, this work aims to propose a method for CAM optimization that can be suitable for the use onboard of spacecraft with either high- or low-thrust capabilities.

A CAM is performed when, at the time of closest approach, a threshold on the miss distance, or the collision probability, is exceeded [4]. In [5] a method to optimize an impulsive CAM based on the assumption of small maneuver was introduced. This simplification allowed decoupling the problem into maneuver direction and magnitude determination. The direction was determined by the gradient of the collision probability while the magnitude with an iterative process. In [6] Alfano describes a tool for CAM analysis that can perform parametric studies of single-axis and dual-axis maneuvers. Collision probability contours for single-axis maneuvering are calculated based on an upper bound on the impulse magnitude and a range of permissible maneuver times. By selecting a specific time, contours are produced for dual-axis maneuvering. Agencies and operators use similar tools [3]. For example, a simple approach employing tangential maneuvers, thus sacrificing optimality, is adopted by the German Aerospace Center [7]. European Space Agency (ESA)’s tool CAMOS instead can deal with single and multiple impulses with arbitrary direction, different objective functions (e.g., collision probability, miss distance, total $\Delta v$), bounds on the maneuvers, and constraints on the post-maneuver orbital elements [8]. The drawback of this flexibility is that the obtained solutions are only locally

* email: roberto.armellin@auckland.ac.nz
optimal, and therefore, the analyst must critically analyze the results. In [9] a multi-objective approach for CAM design was presented that enabled an exhaustive analysis of the problem building Pareto optimal solutions according to multiple criteria. This single-impulse approach also allows for merging station-keeping with CAM and assessing collision risk for a window of one week after the maneuver. However, this approach is numerically intensive, as it requires multiple evaluations of complex objective functions. Bombardelli and Hernando-Ayuso [10] developed an analytical and semi-analytical method to find the fuel-optimal impulsive CAM. The proposed methods have proven convergence as the problem is reduced either to an eigenvalue problem or a convex optimization one. However, as a single impulse was considered, a single linearization was used, and the dynamics were Keplerian.

The research on low-thrust debris avoidance is not very developed. A similar problem is studied in a great deal of detail for formation flying collision-free reconfiguration. Both direct and indirect optimal control approaches were also proposed and compatible with onboard use (see the introduction in [11] and the references therein for an exhaustive overview). Restricting the analysis to low-thrust CAM design, Reiter and Spencer [12] noted that the impulsive approximation could be inaccurate for short alert times; thus, finite burn analysis is needed. However, to cope with this issue, a radial thrust assumption was made. In [13] an approach based on averaged dynamics and Gauss variational equation is proposed with the underlying assumption of continuous tangential thrust. Four methods based on an indirect formulation of an optimal control problem are presented in [14]. It is concluded that a semi-analytical method based on the linearization of the dynamics offers the best compromise between accuracy and computational time. However, formulating an energy optimal control problem without bounds on thrust magnitude is a significant limitation for this approach.

In this work, we present a methodology for optimal CAM design with execution time compatible with autonomous onboard use. The approach is suitable for both impulsive and low-thrust CAM design and can handle short-term encounters with warning times from few minutes to multiple orbital revolutions. Furthermore, no a priori assumption on the direction of the impulses is made, and an arbitrary dynamical model can be used. Constraints either or miss distance, maximum collision probability, or an approximated value of the collision probability can be enforced while minimizing the total \( \Delta v \). The approach is based framing a multiple-impulse CAM optimization problem as a convex optimization one [15]. Thanks to the proven existence and uniqueness of the solution and computational advantages ensured by polynomial complexity, convex optimization has found many applications in aerospace engineering in the last 15 years [16], including long-duration low-thrust transfers [17], [18]. The multiple-impulse CAM optimization is a Nonlinear Programming Problem (NLP) problem, and three steps are required to formulate it as a convex optimization problem, in our case as a Second-Order Cone Programming (SOCP) problem. Three main issues are encountered:

1. the minimum fuel problem is non-convex;
2. the dynamics and the transformation of coordinates are nonlinear;
3. the constraints are non-convex.

The introduction of slack variables and a lossless convexification [19], [18] solve the first issue. The dynamics and the transformations required to perform the b-plane analysis are linearized, using Differential Algebra (DA) implemented in DACE as a first-order automatic differentiation technique [20]. Due to the small deviations of CAMs, linearized dynamics accurately describe the conjunction dynamics [10] [14]. As a result, the problems associated with linearization (i.e., artificial infeasibility and unboundedness [21]) are not relevant here, and the successive convexification is introduced for refinement purposes, with no need for virtual controls and complex trust-region algorithms [22].

The third issue is tackled by working on the squared Mahalanobis distance, whose contour lines describe an ellipse on the b-plane. The squared Mahalanobis distance allows us to set a constraint either on minimum distance, maximum collision probability, or an approximation of the collision probability [23]. However, the constraint on this quantity results in a non-convex problem, as the admissible region on the b-plane is non-convex (as in the keep-out zone in rendezvous dynamics [24]). The projection and linearization technique proposed in [25] is adopted to tackle this issue, resulting in a second iterative procedure. Depending on the chosen initial guess, the iterations can converge to different optima. However, two suitably selected guesses allows for the identification of the global minimum.

The final CAM optimization approach is solved with the state-of-the-art primal-dual interior-point algorithm implemented in the software MOSEK [26]. Solutions with hundreds of impulses are obtained robustly and efficiently, representing both high-thrust and low-thrust maneuvers. We test our algorithm on 2,170 real conjunctions derived from the ESA Collision Avoidance Challenge https://kelvins.esa.int/collision-avoidance-challenge. The paper is organized as follows. In Sec. [2] a brief overview of the short-term conjunction dynamics is provided together with a nonlinear approach to study the effect of a CAM on the conjunction geometry. Section [3] contains a description of the methodology developed in this work. We first state the CAM design as NLP problem, followed

---

2The open-source software package is available at https://github.com/dacelib/dace
by the description of the steps required for its convexification. The algorithm’s application to a large set of cases is described in [4] leading to the conclusive section.

2 Conjunction Dynamics

A brief introduction of the key quantities of a short-term encounter is provided. Only relevant concepts for the design of CAMs are summarized. Afterward, we introduce the DA-based approach to study the effect of maneuvers on the conjunction geometry used to define the optimization problem of Sec. [3]

2.1 Collision Probability Computation

We consider the conjunction between a primary (subscript \( p \)) and a secondary (subscript \( s \)). The primary is the spacecraft we control, whereas the secondary is assumed to be passive. We indicate the relative position and velocity vectors at the Closest Approach (CA) as

\[
\Delta r^*_{CA} = r^*_{p,CA} - r^*_{s,CA},
\]

\[
\Delta v^*_{CA} = v^*_{p,CA} - v^*_{s,CA},
\]

in which \( r \) and \( v \) are the absolute position and velocity vectors, and the asterisk indicates the ballistic motion. As at the CA the distance between the two objects is minimum, it follows

\[
\Delta r^*_{CA} \cdot \Delta v^*_{CA} = 0.
\]

To compute the collision probability it is useful to introduce a coordinate system referred to as the b-plane. The origin of the axes of this frame lies at the centre of the secondary object at the time of CA; the \( \eta \)-axis is defined along the direction of the relative velocity of the primary with respect to the secondary object; the \( \xi \zeta \) plane is perpendicular to that \( \eta \)-axis

\[
\hat{u}_\xi = \frac{v^*_{s,CA} \times v^*_{p,CA}}{\|v^*_{s,CA} \times v^*_{p,CA}\|},
\]

\[
\hat{u}_\eta = \frac{v^*_{p,CA} - v^*_{s,CA}}{\|v^*_{p,CA} - v^*_{s,CA}\|},
\]

\[
\hat{u}_\zeta = \hat{u}_\xi \times \hat{u}_\eta.
\]

All the introduced quantities are shown in Fig. [1]

![Figure 1: Conjunction geometry and b-plane definition](image)

The unit vectors define the rotation matrix from the inertial reference frame to the b-plane

\[
R_{3D} = [\hat{u}_\xi \ \hat{u}_\eta \ \hat{u}_\zeta]^T,
\]

(7)
while the projection in the \( \eta \)-axis is achieved by
\[
R_{2D} = [\hat{u}_\xi \, \hat{u}_\zeta]^T.
\] (8)

The nominal position of the primary on the b-plane at the time of closest approach \( t_{CA}^* \) is \( \Delta r_{CA}^* = (\xi^*, \zeta^*) \). The particular case in which \( \Delta r_{CA}^* = 0 \) is referred to as a direct impact.

Under the short encounter approximation and Gaussian distribution of objects state vectors, the collision probability is
\[
P_C = \frac{1}{2\pi (\det C_{CA}^*)^{1/2}} \int_{A} e^{-\frac{1}{2}(\Delta r - \Delta r_{CA}^*)^T (C_{CA}^*)^{-1} (\Delta r - \Delta r_{CA}^*)} d\xi d\zeta
\] (9)
in which \( C_{CA}^* \) is the sum of the positional covariance of the two objects referred to a common reference frame and projected onto the b-plane via (8), and \( A \) collision cross sectional area, a circle of radius \( R = R_p + R_s \) where \( R_p/s \) is the radius of the sphere enclosing the primary/secondary respectively. Several methods have been developed over the years to calculate \( P_C \). Among them, \[27\], \[28\], and \[29\] are worth of a particular merit as they provided analytical solutions. As explained in \[23\], the simplest approach consists in approximating the value of \( P_C \) by assuming the probability density is constant over the collision circle. The approximate value is
\[
P_C^* = \frac{R^2}{2 (\det C_{CA}^*)^{1/2}} e^{-\frac{1}{2}(d_{CA}^*)^2},
\] (10)
in which \( d_{CA}^* = \sqrt{\Delta r_{CA}^*} (C_{CA}^*)^{-1} \Delta r_{CA}^* \) is the Mahalanobis distance. Additionally, the maximum collision probability can be obtained by optimally scaling the combined covariance (23) resulting in
\[
P_{C,\text{max}}^* = \frac{R^2}{(d_{CA}^*)^2 (\det C_{CA}^*)^{1/2}} e.
\] (11)

These approximations allow us to use the squared Mahalanobis distance to set constraints on an approximate value of the collision probability through Eq. (10), the maximum collision probability through Eq. (11), or the miss distance (by setting \( C_{CA}^* = I \), the identity matrix). Moreover, for constant covariance matrix, the contour lines of the squared Mahalanobis distance describe ellipses on the b-plane, a type of constraint that can be dealt with efficiently by successive convexifications, as it will be shown later.

### 2.2 Effect of Maneuvers on the Conjunction

When the primary is maneuvered, the conjunction geometry changes. This aspect is here analyzed with the use of arbitrary order Taylor expansions enabled by DA (the notation adopted in \[30\] is used). These effects are generally small for short-term encounters and small maneuvers. Nevertheless, a general treatment is provided as the proposed approach could be potentially applied to longer encounters with minimal changes (provided that the distance function is convex in time).

The first step is to use DA to introduce perturbations to the closest encounter time \( t_{CA}^* + \delta t \), the primary position \( \delta r_{p,CA} + \delta r_p \), and the primary velocity \( \delta v_{p,CA} + \delta v_p \). Taylor expansions of the states of both the primary and secondary are obtained by DA-based numerical integrations (expansion in time and state, see \[31\] for details) of the orbital dynamics, delivering
\[
\begin{align*}
 r_p &= T_{r_p}(\delta t, \delta r_p, \delta v_p) \\
 v_p &= T_{v_p}(\delta t, \delta r_p, \delta v_p) \\
 r_s &= T_{r_s}(\delta t) \\
 v_s &= T_{v_s}(\delta t).
\end{align*}
\] (12)

Note that in Eq. (12), due to the effect of the introduced perturbations, we have dropped the CA subscript. Additionally, the state of the secondary is only affected by the time perturbation.

From Eq. (12) the relative quantities can be calculated
\[
\begin{align*}
\Delta r &= T_{\Delta r}(\delta t, \delta r_p, \delta v_p) \\
\Delta v &= T_{\Delta v}(\delta t, \delta r_p, \delta v_p).
\end{align*}
\] (13)

The closest encounter conditions, Eq. (3), in DA formalism, reads
\[
\Delta r \cdot \Delta v = T_{\Delta r \Delta v}(\delta t, \delta r_p, \delta v_p) = 0.
\] (14)
This constraint is a parametric implicit equation that can be solved for $\delta t$ by polynomial partial inversion techniques (see [31] and [32] for more details). The polynomial partial inversion provides

$$\delta t = T_{st} (\Delta r \cdot \Delta v, \delta r_p, \delta v_p)$$

(15)

and, by substitution of the closest encounter constraint $\Delta r \cdot \Delta v = 0$, we obtain

$$\delta t_{CA} = T_{stCA} (\delta r_p, \delta v_p)$$

(16)

This map gives a Taylor approximation of how the closest encounter time changes due to a variation of the state of the primary (due to a maneuver). This polynomial can be inserted back in Eq. (12), obtaining

$$r_p,CA = T_{r_p,CA} (\delta r_p, \delta v_p)$$

$$v_p,CA = T_{v_p,CA} (\delta r_p, \delta v_p)$$

$$r_s,CA = T_{r_s,CA} (\delta r_p, \delta v_p)$$

$$v_s,CA = T_{v_s,CA} (\delta r_p, \delta v_p).$$

(17)

These polynomial maps approximate the states of both objects at the different times of CA as a result of a CAM that produces a variation of position and velocity of the primary object at $t_0^{CA}$. Note that in Eq. (17) we re-introduce the CA subscript, but we remove the asterisk, as now each perturbed solution has a different time of closest approach, determined by Eq. (16). Similarly, the projection matrix in Eq. (8) is expanded by using Eq. (17) for the calculation of the unit vectors, resulting in

$$R_{2D} = T_{R_{2D}} (\delta r_p, \delta v_p)$$

(18)

Equation (15) is used both to project the relative state and the combined covariance matrix on the b-plane, thus allowing for the calculation of the expansion of all the relevant conjunction quantities:

$$\Delta r_{CA} = T_{\Delta r_{CA}} (\delta r_p, \delta v_p),$$

(19)

$$d^2_{CA} = T_{\delta^2_{CA}} (\delta r_p, \delta v_p),$$

(20)

$$P_{C} = T_{P_{C}} (\delta r_p, \delta v_p),$$

(21)

and

$$P_{C,\text{max}} = T_{P_{C,\text{max}}} (\delta r_p, \delta v_p).$$

(22)

Note that in Eqs. (19), (22), the dependency on $\delta r_p$ and $\delta v_p$ is due to the covariance $C_{CA} = T_{C_{CA}} (\delta r_p, \delta v_p)$ as a result of the dependency of the projection matrix $T_{R_{2D}}$ as a result of the perturbation of the states at the conjunctions. On the other hand, it is always assumed that the state covariances provided in the Conjunction Data Message (CDM) are not directly altered by the implementation of the maneuver.

### 3 Collision Avoidance Maneuver Design

The details of CAM design algorithm for multiple-impulse maneuvers is presented. Before framing it as a successive convexification problem, a general NLP formulation is described to provide the general setting on which the successive convexification is introduced.

#### 3.1 Nonlinear Programming Formulation

A uniform $N$-point time grid is constructed by selecting a discretization time step $\Delta t$ and starting from the earliest maneuvering time $t_0$. Note that more refined discretization schemes are possible without adding complexity to the algorithm (e.g., by using a suitable angular variable to better deal with eccentric orbits), but these were not needed as the orbits in the test cases are almost circular. The number $N$ is selected as $N = \min \left( \lfloor \frac{t_{CA}^* - t_0}{\Delta t} \rfloor, N_{\text{max}} \right)$, where $N_{\text{max}}$ accounts for the maximum time span in which a maneuver can be implemented. At every discretization point a maneuver can be added in the form of an instantaneous change in velocity $\Delta v_i$ with $i = 0, \ldots, N - 1$. The nominal trajectory at discretization points is given by $r_i^*, v_i^*$ with $i = 0, \ldots, N$, with $r_N^* = r_{CA}^*$, $v_N^* = v_{CA}^*$. For each $i = 0, \ldots, N - 1$ we calculate a $o$-th order Taylor approximation of the mapping between deviations in the initial state and deviations of the final state

$$\delta r_{i+1} = T_{\delta r_{i+1}} (\delta r_i, \delta v_i)$$

$$\delta v_{i+1} = T_{\delta v_{i+1}} (\delta r_i, \delta v_i).$$

(23)
In Eq. (23) the + superscript indicates chosen quantities at the beginning of an interval, whereas the − propagated quantities from the previous interval. These maps are built with N DA integrations of a dynamical model of choice using the unperturbed trajectory as reference (e.g., obtained by backward propagation from the CDM with the same dynamical model). As the CAM $\Delta V$’s are small, a low order (in most cases order 2) allows for sufficiently accurate approximations without the need of iterations.

The optimization variables are the set of $N$ $\Delta v_i$ applied at nodes $i = 0, \ldots, N - 1$. The effect of these maneuvers on the trajectory of the primary are obtained by the use of maps of Eq. (23). In particular, for $i = 0$ we can set $\delta r_0^+ = 0$ and $\delta v_0^+ = \Delta v_0$, and, by applying (23), we obtain the mapped perturbations $\Delta r_1^-$ and $\delta v_1^-$. For $i = 1, \ldots, N - 1$ we proceed by defining the new perturbations $\delta r_i^+ = \delta r_{i-1}^-$ and $\delta v_i^+ = \delta v_{i-1}^- + \Delta v_i$ and use the $i$-th set of maps of Eq. (23) to map these perturbations to the end of the segment. At $i = N - 1$ we achieve how the set $x = [\Delta v_0; \ldots; \Delta v_{N-1}]$ is mapped into the final perturbations $\delta r_p$ and $\delta v_p$, which then in turn allows us to compute the relevant quantities by Eqs. (19)-(22). The multiple-impulse CAM optimization problem, can be stated as follows: Minimize the sum of magnitudes of the impulses

$$\min_x \sum_{i=0}^{N-1} ||\Delta v_i||$$  \hspace{1cm} (24)

subject to nonlinear inequality constraints

$$||\Delta v_i|| \leq \Delta \bar{v} \text{ for } i = 0, \ldots, N - 1,$$

$$P_C \leq \bar{P}_C \text{ or } P_{C,\text{max}} \leq \bar{P}_{C,\text{max}} \text{ or } \Delta r_{CA} \geq \bar{d}_{\text{min}},$$  \hspace{1cm} (25)

where the overline indicates assigned values. This optimization problem is a NLP that can be solved with dedicated solvers. In this work we use the Sequential Quadratic Programming (SQP) algorithm implemented in the MATLAB fmincon function providing analytical gradient of the objective function and Jacobian of the constraints. For the latter the derivatives included in the polynomials maps are used together with the chain rule to calculate the sensitivity of the constraints with respect to the optimization vector. The summary of the NLP formulation is provided in Algorithm 1. As NLP problems, this formulation is non-deterministic polynomial-time hard (NP-hard), meaning that the computation time may be very long if the problem is solved at all [24].

Algorithm 1 Nonlinear programming formulation

1: Get inputs from CDM: $R$, $v_{p/s,CA}^*$, $t_{CA}^*$, $C_{p/s,CA}^*$;  
2: Assign $t_0$, $\Delta t$, $\Delta \bar{v}$, $\bar{P}_C$ or $P_{C,\text{max}}$ or $\bar{d}_{\text{min}}$, and $x_0$;  
3: Back propagate the trajectories from $t_{CA}^*$ to $t_0$ and save $r_{p/s,t_0}$ and $v_{p/s,t_0}$;  
4: Define the time grid $(t_0 : \Delta t : t_0 + N\Delta t)$;  
5: Build maps Eq. (23) by $N$ o-th order DA forward propagations;  
6: Solve the NLP problem defined by (24) and (25);

3.2 Convex Problem Formulation

As described in the introduction, three main steps are required to formulate the CAM design problem as a convex optimization one. Firstly, in Sec. 3.2.1 the objective function and the constraints on the velocity magnitude are reformulated by introducing slack variables and lossless convexification. Afterward, the dynamics of the problem are linearized in Sec. 3.2.2 followed by constraints linearization in Sec. 3.2.3. Only the constraint on the squared Mahalanobis distance is taken into account as this type of constraint is effectively handled by a projection and linearization approach [25]. The details of the resolution algorithm are then presented in Sec. 3.2.4.

3.2.1 Lossless Convexification

The $\Delta v$ magnitudes are introduced as slack variables in the optimization problem. As a result, each impulse is described by four independent variables $\Delta \bar{v}_i = [\Delta v_{i,x} ; \Delta v_{i,y} ; \Delta v_{i,z}]$, and the optimization vector becomes $x = [\Delta v_0; \ldots; \Delta v_{N-1}; \Delta v_0; \ldots; \Delta v_{N-1}]$. The introduction of the slack variables renders objective function linear

$$\min_x \sum_{i=0}^{N-1} \Delta v_i$$  \hspace{1cm} (26)

and transforms the $N$ constraints on the impulses magnitude in Second Order Cone (SOC) constraints

$$\sqrt{\Delta v_{i,x}^2 + \Delta v_{i,y}^2 + \Delta v_{i,z}^2} \leq \Delta v_i \text{ for } i = 0, \ldots, N - 1.$$  \hspace{1cm} (27)
Lastly, the bounds on the slack variables

\[ 0 \leq \Delta v_i \leq \Delta \bar{v} \quad \text{for} \quad i = 0, \ldots, N - 1 \]  

are added to the problem. This convexification step is referred to as lossless as it can be proved that the optimal solution of the convexified problem is also the optimal solution of the original one \[24\].

### 3.2.2 Linearization of the Dynamics

The introduction of the slack variables is not sufficient to make the problem convex due to the nonlinearities in Eq. \[23\]. These are dealt with by successive linearizations, requiring an iterative process. We will refer to these iterations as major iterations associated with index \( j \) in the remainder of this section.

Assume a solution \( x_j^{j-1} \) is available providing a reference trajectory (only at the first major iteration the reference trajectory is ballistic) about which the dynamics are linearized. At the \( j \)-th iteration, the linear part of Eq. \(23\) can be extracted resulting in

\[
\begin{bmatrix}
\delta r_{i+1} \\
\delta v_{i+1}
\end{bmatrix}^j = [A_{\delta r,i} \quad A_{\delta v,i}]^j 
\begin{bmatrix}
\delta r_p \\
\delta v_p
\end{bmatrix}^j = A_i^j \begin{bmatrix}
\delta r_{i+1} \\
\delta v_{i+1}
\end{bmatrix}^j \quad (29)
\]

The composition of all these linear maps results in

\[
\begin{bmatrix}
\delta r_N \\
\delta v_N
\end{bmatrix}^j = \begin{bmatrix}
\delta r_p \\
\delta v_p
\end{bmatrix}^j = [A_{N-1}A_{N-2} \ldots A_{\delta v,0}, \quad A_{N-1}A_{N-2} \ldots A_{\delta v,1}, \quad \ldots, \quad A_{\delta v,N-1}, \quad 0_{0 \times N}]^j (x_j^i - x_j^{j-1}) = \\
= A_{\delta r,1} (x_j^i - x_j^{j-1}), 
\]  

where \((x_j^i - x_j^{j-1})\) is due to the fact that the optimization vector \(x_j^i\) contains absolute impulses, whereas the linearizations are centred about the optimal impulses of the \((j - 1)\)-th iteration. The perturbed time of closest approach, \(t_{CA}^j\), is

\[ t_{CA}^j = t_{CA}^i + \mathbb{B} \delta t_j (x_j^i - x_j^{j-1}) = \mathbb{B} \delta t_j x_j^i + \left( \begin{bmatrix}
t_{CA}^i - \mathbb{B} \delta t_j x_j^{j-1}
\end{bmatrix}^j \right), \quad (31)\]

where \(\mathbb{B}\) is the the \(1 \times 6\) linear part of Eq. \(16\). Similarly, the perturbed relative position vector on the b-plane, \(\Delta r_{CA}^j\), is given by

\[ \Delta r_{CA}^j = \Delta r_{CA}^{j-1} + \mathbb{C} \delta t_j (x_j^i - x_j^{j-1}) = \mathbb{C} \delta t_j x_j^i + \left( \begin{bmatrix}
\Delta r_{CA}^{j-1} - \mathbb{C} \delta t_j x_j^{j-1}
\end{bmatrix}^j \right), \quad (32)\]

in which \(\mathbb{C}\) is the \(2 \times 6\) linear part of Eq. \(19\) at the \(j\)-th iteration. When \(j = 1\), all the \((j - 1)\) quantities in the equations above are relative to the un-maneuvered case, i.e. \(\Delta r_{CA}^{0} = \Delta r_{CA}^{1}\) and \(x_0 = 0\). Note that as the CAMs determine a variation of the position vector of only few kilometers with respect to the ballistic trajectory (i.e., a relative variation of 0.1%), the convexification of the dynamics does not come with issues like artificial infeasibility and unboundedness. For this reason we did not introduce any artificial control or sophisticated trust-region algorithms, but only problem-driven bounds on the impulses magnitude and a maximum final state deviation on the b-plane.

### 3.2.3 Linearization of the Squared Mahalanobis Distance

The last step consists of dealing with the squared Mahalanobis distance constraint that defines an elliptically shaped avoidance region (or keep-out region using rendezvous terminology) on the b-plane. This non-convex constraint is dealt with by a second iterative process, nested in each major iteration, consisting of a projection and a linearization. We refer to these iterations as minor iterations, with index \( k \).

Assume the solution \(x_j^{i,k-1}\) of the \(j\)-th major iteration and \((k - 1)\)-th minor iteration is available. This defines the relative position vector on the b-plane \(\Delta r_{CA}^{j,k-1}\). The \(k\)-th iteration starts by the projection algorithm that finds the point \(z^k\) on the ellipse \(d_{CA}^2 \quad (33)\)

\[
\min_z ||\Delta r_{CA}^{j,k-1} - z|| \quad (33)
\]

subject to the inequality constraint

\[ z^T (C_{CA}^{j})^{-1} z^T \leq d_{CA}^2, \quad (34)\]

that can be solved efficiently using convex optimization algorithms (the interested reader can refer to \[32\] for a exhaustive analysis of this sub-problem). \(C_{CA}^{j}\) does not depend on \(k\) as this quantity is assumed constant within each minor iteration.
Once $z^k$ is computed, the squared Mahalanobis distance constraint is linearized. The linear constraint ensures that $\Delta r_{CA}^{j,k}$ belongs to the half-plane tangent to the constraint in $z^k$, i.e.

$$\nabla d_M^j (z^k)(\Delta r_{CA}^{j,k} - z^k) \geq 0.$$  \hspace{1cm} (35)

By substituting Eq. (32), the final expression is obtained

$$- \nabla d_M^j (z^k) Z \Delta r_{CA}^{j-1} \leq \nabla d_M^j (z^k)(\Delta r_{CA}^{j-1} - z^k - C^j \varphi z^{j-1}),$$

where the $(j - 1)$ indicates known quantities at the end of the minor iteration of the $(j - 1)$-th major iteration. Additionally, $\Delta r_{CA}^{j,0} = \Delta r_{CA}^{j-1}$.

The selection of the first starting point determines the convergence of the algorithm to a local optimum of the original, non-convex, problem. However, as it will be illustrated in Sec. 4.2, the problem appears to have only two two local minima on the opposite side of the elliptical boundary of the avoidance region. Thus, two initial starting points defined by $\Delta r_{CA}^{1,0} = \pm \Delta r_{CA}^{j-1}$ are sufficient to identify the global minimum.

### 3.2.4 Successive Convex Optimization Algorithm

With the introduction of the slack variables, the linearization of the dynamics, and the linearization of the constraint on the squared Mahalanobis distance, the multiple-impulse CAM design problem can be solved by primal-dual interior-point methods. We use MOSEK [26] through its MATLAB interface.

The Algorithm 2 provides a complete overview of the solution process. As previously described, two iterations are needed: a major $j$-th iteration for the linearization of the dynamics and a minor $k$-th iteration for linearization of the constraint. The two iterations stop when two conditions are met: the minor one ends when $||\Delta r_{CA}^{j,k} - \Delta r_{CA}^{j,k-1}||_2 \leq \text{tol}_m$, while the major one when $||x^j - x^{j-1}||_\infty \leq \text{tol}_M$. The algorithm minimizes the total $\Delta v$ while constraining either the risk, the maximum risk, or the closest approach’s distance. The latter case is dealt by setting $C_{CA} = I$ in all iterations.

### 4 Test Cases

#### 4.1 Definition of Test Cases

The methodology described in the previous sections is applied to test cases derived from the ESA Collision Avoidance Challenge. For this competition, ESA provided the teams with real conjunction data extracted from 162,634 CDM, corresponding to 13,154 unique events. However, ESA did not distribute the full orbital elements set and provided the positional covariances in the primary Radial, Transverse, and Normal (RTN) reference frame only. With a procedure omitted here, we managed to reconstruct the full orbital data of the objects except for the absolute value of the right ascension of the ascending node of the primaries, which are arbitrarily set to 0. These data were filtered to consider conjunctions with $\Delta r_{CA}^* \leq 2 \text{ km}$, $P_{\text{c}} > 10^{-6}$, and $P_{\text{c,max}} > 10^{-4}$, resulting in a new data file with 2,170 conjunctions, available for download at [github.com/arma1978/conjunction](http://github.com/arma1978/conjunction). The distribution of the minimum distance at CA, the collision probability and maximum collision probability are shown in Fig[24]. All the conjunctions are relative to objects in Low Earth Orbit (LEO) with 90% of cases with relative conjunction speed $\in [1.80, 14.98] \text{ km/s}$.

All the simulations presented in the next sections were obtained with a dynamical model including $J_2 - J_4$ zonal harmonics

$$\begin{align*}
\dot{x} &= v_x \\
\dot{y} &= v_y \\
\dot{z} &= v_z \\
\dot{v}_x &= -\frac{\mu x}{r^3} + 3\mu J_2 R_x^2 \left( \frac{5}{2}x^2 - 1 \right)x + \frac{5\mu J_4 R_y^2}{8} \left( \frac{7}{2}x^2 - 3 \right) + \frac{15\mu J_4 R_z^2}{8} \left( 1 - \frac{14}{5}z^2 + \frac{21}{8}z^4 \right) \\
\dot{v}_y &= -\frac{\mu y}{r^3} + 3\mu J_2 R_y^2 \left( \frac{5}{2}y^2 - 1 \right)y + \frac{5\mu J_4 R_x^2}{8} \left( \frac{7}{2}y^2 - 3 \right) + \frac{15\mu J_4 R_z^2}{8} \left( 1 - \frac{14}{5}z^2 + \frac{21}{8}z^4 \right) \\
\dot{v}_z &= -\frac{\mu z}{r^3} + 3\mu J_2 R_z^2 \left( \frac{5}{2}z^2 - 1 \right)z + \frac{5\mu J_4 R_x^2}{8} \left( \frac{3}{5} - \frac{6}{5}z^2 + \frac{7}{8}z^4 \right) + \frac{15\mu J_4 R_y^2}{8} ^2 \left( 5 - \frac{70}{3}z^2 + \frac{21}{8}z^4 \right)
\end{align*}$$

(37)

in which $r = [x, y, z]^T$ and $v = [v_x, v_y, v_z]^T$ are the spacecraft position and velocity vectors; $\mu$, $R_x$, and $J_i$ are the gravitational parameter, the mean equatorial radius, and the $i$-th zonal harmonic coefficient of the Earth. Any dynamical model can be selected without affecting the algorithm complexity as the required state transition matrices are automatically obtained with DA without the need to derive and integrate the variational equations.

In Sec. 4.2, 4.5 we offer a detailed analysis for the reference scenario, which is the case with the highest collision probability in the dataset (details in Appendix). In Sec. 4.5 we provide a summary of the method performance when
target case in which the CAM can be applied between 8 and 6 orbits before the CA with a maximum of 200 impulses and a

Figure 5 provides details of the convergence of the successive iteration algorithm summarised in Algorithm 2 for a

4.2 Major and Minor Iterations

GHz Dual-Core Intel Core i7 and 16 GB Memory.

preferred unless orbits are almost circular as in our test cases. The simulations are run on a MacBook Pro with a 3.5

impulses. Note that a uniform discretization in an angular variable, such as the true anomaly, should be in general

 tol

∆

Algorithm 2 Successive convexification optimization algorithm

1: Get inputs from CDM: \( R, r^s_{p/s, CA}, v^s_{p/s, CA}, t^C_{CA}, C^s_{p/s, CA} \);
2: Assign \( t_0, \Delta t, \Delta \bar{t}_i, N, P_C \) or \( P_{C,max} \) or \( d_{\min}, \text{tol}_M, \text{tol}_m \);
3: if \( P_{C,max} \) is defined then
4: Calculate \( C^\ast_{CA} \) and \( d^2_{CA} \) by Eq. (10);
5: else if \( P_{C,max} \) is defined then
6: Calculate \( C^\ast_{CA} \) and \( d^2_{CA} \) by Eq. (11);
7: else
8: Set \( C^\ast_{CA} \leftarrow I \) and \( d^2_{CA} \leftarrow d^2_{\min} \)
9: end if
10: Back propagate the trajectories from \( t^C_{CA} \) to \( t_0 \) and save \( r_{p/s,t_0} \), and \( v_{p/s,t_0} \);
11: Define the time grid \( (t_0: \Delta t: t_0 + N \Delta t) \);
12: \( j \leftarrow 0, \bar{x}^0 \leftarrow 0, t^0_{CA} \leftarrow t^C_{CA}, \Delta \bar{r}_{CA}^1 \leftarrow \pm \Delta r^\ast_{CA}, C^0_{CA} \leftarrow C^\ast_{CA} \);
13: while \((j = 0)\) or \(||x^j - x^{j-1}||_{\infty} \geq \text{tol}_M\) do
14: \( j \leftarrow j + 1; \)
15: Perform a 1-st order DA propagation of the trajectories at \((t_0: \Delta t: t_0 + N \Delta t)\) and \( t^j_{CA} \) with impulses extracted from \( x^{j-1} \);
16: if \( P_C \) is defined then
17: Calculate \( C^\ast_{CA} \) and then \( d^2_{CA} \) from Eq. (10);
18: else if \( P_{C,max} \) is defined then
19: Calculate \( C^\ast_{CA} \) and then \( d^2_{CA} \) from Eq. (11);
20: else
21: Set \( C^\ast_{CA} \leftarrow I \) and \( d^2_{CA} \leftarrow d^2_{\min} \);
22: end if
23: Assemble the matrices \( \mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{J} \) as in Eq. (29)-(32);
24: \( k \leftarrow 0 \) and \( \Delta r^j_{CA} \leftarrow \Delta r^1_{CA} \);
25: while \((k = 0)\) or \(||\Delta r^j_{CA} - \Delta r_c^{j,k-1}||_2 \geq \text{tol}_m\) do
26: \( k \leftarrow k + 1; \)
27: Calculate \( z_k \) by solving the convex optimization sub-problem (33)-(34);
28: Calculate \( x^j_{CA} \) by solving the convex optimization problem defined by (26)-(28) and (36);
29: \( t^j_{CA} \leftarrow \mathcal{B}^j z^j_{CA} + (t^{j-1}_{CA} - \mathcal{B}^j z^{j-1}_{CA}); \)
30: \( \Delta r^j_{CA} \leftarrow \mathcal{C}^j z^j_{CA} + (\Delta r_c^{j-1}_{CA} - \mathcal{C}^j z^{j-1}_{CA}); \)
31: end while
32: \( x^{j+1} \leftarrow x^{j}_{CA}, t^j_{CA} \leftarrow t^j_{CA}, \) and \( \Delta r^j_{CA} \leftarrow \Delta r^{j+1}_{CA}; \)
33: end while

applied to all the datasets. Unless differently specified, all simulations assume \( \Delta t = 1 \) min, a maximum impulse of \( 6E - 3 \) m/s (corresponding to a constant acceleration of \( 1E - 4 \) m/s\(^2\) in each segment), \( \text{tol}_M = 1E - 3 \) m/s, and \( \text{tol}_m = 1 \) m. The one-minute discretization is deemed appropriate to approximate finite burn arcs by a sequence of impulses. Note that a uniform discretization in an angular variable, such as the true anomaly, should be in general preferred unless orbits are almost circular as in our test cases. The simulations are run on a MacBook Pro with a 3.5 GHz Dual-Core Intel Core i7 and 16 GB Memory.

4.2 Major and Minor Iterations

Figure 5 provides details of the convergence of the successive iteration algorithm summarised in Algorithm 2 for a

case in which the CAM can be applied between 8 and 6 orbits before the CA with a maximum of 200 impulses and a
target \( P_{C,max} = 1E - 4 \). In Fig. 5(a) the contour line \( P_{C,max} = P_{C,max} \) is plotted as a black solid line, the dot close to
the origin is the unperturbed primary position with \( \Delta r_{CA} = 43 \) m, while the circles represent the perturbed primary
position during the algorithm iterations colored according to the maneuver \( \delta v \). The optimization requires two major
iterations, with 5 and 1 minor iterations, respectively. In Fig. 5(b) the first minor iteration starts from the red circle
labeled 1 representing \( z_1 \), which is the point on the constraint line closest to the unperturbed solution. The output
of the first run of the optimizer delivers the solution \( \Delta r^1_{CA} \) indicated as a yellow filled circle in the figure, which
belongs to the ellipse tangent in \( z_1 \). From \( \Delta r^1_{CA} \) the new \( z_2 \) is calculated allowing for the updated solution \( \Delta r^{1,2}_{CA} \).
with significant reduction of maneuver $\Delta v$. Figure 5(c) shows similar information for the third and fourth iterations and Figure 5(d) for the fifth iteration of the first major loop and the first and only iteration of the second major iteration. It
is worth highlighting that the contour line of equal maximum collision probability is slightly changed (dashed line) because, between the first and second major iteration, the b-plane and the combined 2D covariance are updated.

Figure 5: B-plane convergence analysis

Figure 6(a) highlights the thrust arcs on the primary trajectory (with the pentagram indicating the conjunction point) and Fig. 6(b) shows the details of the impulses in the RTN directions. The thrust arcs’ optimal location is spread along the portion of the orbit opposite to the conjunction. The total $\Delta v$ is 0.2042 m/s, requiring 34 impulses. Figure 6(b) shows that the optimal maneuver has components in the radial and, to a lesser extent, in the out of plane directions. It is remarkable how each minor iteration of the problem is solved in approximately 12 ms and 14 iterations, allowing for a complete solution in less than 0.1 s. This computational time is required to solve a large optimization problem entailing 800 optimization variables, 200 second-order cone constraints, one inequality constraint on the squared Mahalanobis distance, and simple bounds. By contrast, the NLP method described in Sec. 1 requires, when it reaches convergence, thousands of iterations and a few minutes to achieve a comparable solution starting from a random feasible guess.

As previously discussed, it is not guaranteed that the minor iterations converge to the original non-convex problem’s global optimum. Convergence to a local minimum can occur depending on the selected starting point for the minor iteration. However, as the optimal solution lies on the admissible region’s elliptical boundary, we can investigate the objective function’s behavior on this boundary. This analysis is done by sampling the ellipse with $M$ points and solving $M$ convex optimization problems in which a terminal equality constraint substitutes the keep-out zone one. Figure 7 shows the results for $M = 300$, where the pentagram indicates a local minimum (0.2139 m/s) and the hexagram the global one (0.2042 m/s) analyzed in the previous figures. The examined cases have similar objective function structure, with the two minima located at the opposite side of the ellipse and corresponding to thrust mainly aligned with either the tangential or the anti-tangential direction. These two minima can be identified by starting the minor iterations with two points on the ellipse with opposite coordinates.
4.3 Effect of Constraints

The effect of the different constraints on the CAM design is analyzed, considering impulses on the last two orbits before the conjunction. Figure 8(a) shows the contour lines corresponding to $d_{\text{min}} = 2 \text{ km}$, $P_{C,\text{max}} = 1E - 4$ (the case of the previous section), and $P_{C} = 1E - 6$. The collision probability constraint is the less restrictive one, corresponding to $\Delta v = 0.0281 \text{ m/s}$ and five impulses. The maximum collision probability requires $\Delta v = 0.2881 \text{ m/s}$ and 48 impulses, slightly higher than the previous section’s case due to the shorter time to the conjunction. However, these two constraints are characterized by the same elliptical shape on the b-plane with different sizes. The minimum distance constraint is a circle, and the chosen value of 2 km results in the most constraining one resulting in $\Delta v = 0.5274 \text{ m/s}$ and 88 impulses. Changing the constraint value makes it possible to identify a line of optimal target points on the b-plane. An example is provided in Fig. 8(b) where different values of $P_{C,\text{max}}$ are considered, resulting in $\Delta v \in [0.0064, 0.4311] \text{ m/s}$. A trade-off between safety and propellant consumption is enabled by the rapid calculation of CAM.

4.4 Time to Closest Approach

We analyze the effect of the alert time on the CAM. In Fig. 9 four graphs with leading time decreasing from 16 to 4 orbits are reported. The $\Delta v$ is 0.1281 m/s for the first case, 0.1534 m/s for the second, 0.2042 m/s for the third and 0.2681 m/s for the last. The optimal maneuver in this case is performed at the first passage through the orbit opposite the conjunction. When the leading time reduces and the $\Delta v$ increases, the optimal maneuver is split into multiple arcs,
two in this case. Additionally, the optimal maneuver always has a radial component. Furthermore, a maneuver applied several orbits before the conjunctions has a larger impact on the conjunction geometry, affecting the time of CA for more than 2 seconds.

Although the behaviour described in Fig. 9 is the most common one, not always the optimal maneuver occurs at earliest time. Figure 10 shows that, for test case number 10 in our dataset, the maneuver is performed at the latest opposition with respect to the conjunction. This counter-intuitive behaviour is probably due to the characteristics of the conjunction in terms of geometry and covariances, but its explanation requires further work.

4.5 Maximum Impulse Magnitude

The effect of varying the single impulse magnitude is studied. In Fig. 11, the maximum impulse available is reduced from 0.6 m/s to $3E^{-3}$ m/s, corresponding to a constant acceleration from $1E^{-2}$ m/s$^2$ to $5E^{-5}$ m/s$^2$. This change has a significant impact on the number of impulses and the efficiency of the maneuver. In the first case, a single impulse with a magnitude less than half of the maximum allowed is sufficient, resulting in $\Delta v = 0.2749$ m/s. In the latter case, the maneuver requires 115 impulses (with thrust on for more than half of the time) split into two thrusting arcs and a total $\Delta v = 0.3451$ m/s. Closer to the conjunction, the radial component of thrust becomes more relevant, as reported in [12].

4.6 Full Dataset Simulation

The proposed approach is run on 2,170 test cases, using either $P_{C,\text{max}} \leq 1E^{-4}$, $P_C \leq 1E^{-6}$, or $\Delta r_{\text{CA}} \geq 2$ km as constraints. The maneuvers can be implemented starting from 2 revolutions before the encounter using up to 170 impulses. The maximum impulse magnitude is $6 \times 1E^{-3}$ m/s, corresponding to a constant acceleration of $1E^{-4}$ m/s$^2$ in each segment.

Figures 12-14 show the distribution of relevant quantities when the three different constraints are applied. In the graphs, data in the range 5 – 95th percentile are plotted for the sake of readability. The $P_{C,\text{max}}$ and $P_C$ cases produce similar results in terms of median $\Delta v$ ($0.0212$ m/s versus $0.0178$ m/s), relative closest encounter distance ($0.7818$ km versus $0.8627$ km), and number of impulses (4 in both cases). The miss distance constraint is more demanding in terms of median $\Delta v$ ($0.0689$ m/s) and number of impulses (12), without providing on average neither a lower collision risk nor a lower maximum collision risk (median values of $1.0202E^{-4}$ and $1.8363E^{-6}$, respectively). This indicates that when orbital knowledge statistics are reliable, collision probability constraints should be preferred over the miss distance one. In terms of iterations, the three constraints share a similar behavior with a median of 2 major iterations (with a maximum of 6) and 3-4 minor iterations. In most cases, only one minor iteration is sufficient for the second major iteration, showing that the dynamics are almost linear due to both the short time to conjunction and the maneuvers’ size. The variation of the time of closest approach is in most cases less than half-second, but in a few cases, it reaches 8 seconds. Lastly, 7 and 6 failures occur in the risk and miss distance cases, respectively. However, it is
Figure 9: Impulses profile as a function of time to CA

(a) 16-14 orbits to CA
(b) 12-10 orbits to CA
(c) 8-6 orbits to CA
(d) 4-2 orbits to CA

Figure 10: Test case in which the maneuver is performed at the latest opposition
worth mentioning that these failures are not related to convergence issues in the successive convexification iterations, but they occur when the control authority is insufficient to meet the constraints.

5 Conclusion

A method based on lossless and successive convexification was proposed for the optimal design of collision avoidance maneuvers (CAMs). The maneuver is modeled as a set of impulses and it is thus suitable for handling high- and low-thrust propulsion systems. Maneuvers with minimum propellant consumption were computed meeting constraints either on an estimate of collision risk, maximum collision risk, or miss distance without any prior knowledge on the thrust arc structure and thrust direction. Alert times from minutes to several orbital periods were considered. The proposed method’s convergence properties and efficiency were proven by optimizing CAMs for 2,170 realistic conjunctions, showing that this methodology is promising for future autonomous onboard usage. Future efforts will be directed towards handling long-term and multiple encounters and the direct use of accelerations as decision variables.

Acknowledgement

The author acknowledges the help of Sébastien Henry for the reconstruction of the conjunction geometries and the comments from anonymous reviewers.
Figure 12: Maximum risk constraint

(a) Distribution of $P_C$

(b) Distribution of $\Delta r_{C,A}$

(c) Distribution of $\Delta v$

(d) Distribution of impulses
Figure 13: Risk constraint
Figure 14: Miss distance constraint
Appendix

The data used for the simulations presented in Sec. 4.2–4.5 are reported below, those for the extensive simulations are available at [github.com/arma1978/conjunction](https://github.com/arma1978/conjunction).

### Primary

| ECI J2000 Position [km] | ECI J2000 Velocity [km/s] |
|-------------------------|---------------------------|
| 2.33052185175137E + 00  | -7.44286282871773E + 00  |
| -1.10370451050201E + 03| -6.13734743652660E – 04 |
| 7.10588764299718E + 03 | 3.95136139293349E – 03  |

# Covariance matrix RTN [km²]

|                    |                  |                  |
|--------------------|------------------|------------------|
| 9.3170905887535E – 05 | -2.62339811350055E – 04 | 2.36038217393530E – 05 |
| -2.62339811350055E – 04 | 1.77796454279511E – 02 | -9.33122538738651E – 05 |
| 2.36038217393530E – 05 | -9.33122538738651E – 05 | 1.91737223188004E – 05 |

### Secondary

| ECI J2000 Position [km] | ECI J2000 Velocity [km/s] |
|-------------------------|---------------------------|
| 2.33346550626332E + 00  | 7.3353740487126315E + 00 |
| -1.10367121247836E + 03| -1.14281409765362E + 00 |
| 7.10591495809903E + 03 | -1.98247259113771E – 01 |

# Covariance matrix RTN [km²]

|                    |                  |                  |
|--------------------|------------------|------------------|
| 6.34657091072037E – 04 | -1.96229221624528E – 03 | 7.07741365522766E – 05 |
| -1.96229221624528E – 03 | 8.19989963150306E – 01 | 1.13982381058435E – 03 |
| 7.07741365522766E – 05 | 1.13982381058435E – 03 | 2.51034082907407E – 04 |

### Conjunction details

| R | Note |
|---|------|
| 29.71 | m |

\[ d_{\text{RC}}^2 = 8.71655401455392E – 01 \] \[ P_C = 1.36040828266536E – 01 \] \[ P_C = 1.4755966159940E – 01 \] \[ P_{C,\text{max}} = 1.92590968666693E – 01 \]

### References

[1] Jonathan C. McDowell. The low earth orbit satellite population and impacts of the SpaceX Starlink constellation. *The Astrophysical Journal*, 892(2):L36, apr 2020.

[2] Joseph A. Haimerl and Gregory P. Fonder. Space fence system overview. In *Proceedings of the Advanced Maui Optical and Space Surveillance Technology Conference*. Curran Associates, Inc. Redhook, NY, 2015.

[3] Fabian Schiemenz, Jens Utzmann, and Hakan Kayal. Survey of the operational state of the art in conjunction analysis. *CEAS Space Journal*, 11(3):255–268, 2019.

[4] H. Klinkrad. *Space debris models and risk analysis*. Wiley Online Library, 2006.

[5] Russell P. Patera and Glenn E. Peterson. Space vehicle maneuver method to lower collision risk to an acceptable level. *Journal of Guidance, Control, and Dynamics*, 26(2):233–237, 2003.

[6] Salvatore Alfano. Collision Avoidance Maneuver Planning Tool. In *15th AAS/AIAA Astrodynamics Specialist Conference*. American Astronautical Soc. Paper 2005-308, pages 1–15, 2005.

[7] Saika Aida. Conjunction Risk Assessment and Avoidance Maneuver Planning Tools. In *6th International Conference on Astrodynamics Tools and Techniques*, 2016.

[8] Juan Antonio Cobo-Pulido, Noelia Sánchez-Ortiz, Ignacio Grande-Oralla, and Klaus Merz. CORAM: ESA’s collision risk assessment and avoidance manoeuvres computation tool. In *Advances in the Astronautical Sciences*, volume 153, pages 453–471, 2015.

[9] Alessandro Morselli, Roberto Armellin, Pierluigi Di Lizia, Franco Bernelli-Zazzera, P. Di Lizia, and Franco Bernelli-Zazzera. Collision avoidance maneuver design based on multi-objective optimization. In *24th AAS/AIAA Space Flight Mechanics Meeting*, volume 152, 2014.

[10] Claudio Bombardelli and Javier Hernando-Ayuso. Optimal Impulsive Collision Avoidance in Low Earth Orbit. *Journal of Guidance Control and Dynamics*, 38(2):217–225, 2015.
[11] G. Di Mauro, D. Spiller, S. F. Rafano Carnà, and R. Bevilacqua. Minimum-fuel control strategy for spacecraft formation reconfiguration via finite-time maneuvers. *Journal of Guidance, Control, and Dynamics*, 42(4):752–768, 2019.

[12] Jason A. Reiter and David B. Spencer. Solutions to Rapid Collision-Avoidance Maneuvers Constrained by Mission Performance Requirements. *Journal of Spacecraft and Rockets*, pages 1–9, 2018.

[13] Juan Luis Gonzalo, Camilla Colombo, and Pierluigi Di Lizia. A semi-analytical approach to low-thrust collision avoidance design. In *70th International Astronautical Congress (IAC)*, number October, pages 21–25, 2019.

[14] Giuseppina Salemme, Roberto Armellin, and Pierluigi Di Lizia. Continuous-thrust collision avoidance manoeuvres optimization. In *SciTech 2020*, number January, pages 1–21, 2020.

[15] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, 2004.

[16] Xinfu Liu, Ping Lu, and Binfeng Pan. Survey of convex optimization for aerospace applications. *Astrodynamics*, 1(1):23–40, 2017.

[17] Zhenbo Wang, Michael J. Grant, Camille E. Bergin, Gillian S. McGlothin, Spencer T. McDonald, and Zhenbo Wang. Minimum-Fuel Low-Thrust Transfers for Spacecraft: A Convex Approach. *IEEE Transactions on Aerospace and Electronic Systems*, 54(5):2274–2290, 2018.

[18] Yuanqi Mao, Michael Szmuk, Xiangru Xu, and Behcet Açıkmeşe. Successive convexification: A superlinearly convergent algorithm for non-convex optimal control problems. *arXiv preprint arXiv:1804.06539*, 2018.

[19] Yuanqi Mao, Michael Szmuk, and Behcet Açıkmeşe. A Tutorial on Real-time Convex Optimization Based Guidance and Control for Aerospace Applications. *Proceedings of the American Control Conference*, 2018-June:2410–2416, 2018.

[20] MOSEK ApS. *The MOSEK optimization toolbox for MATLAB manual. Version 9.0.*, 2019.

[21] F. Kenneth Chan. *Spacecraft Collision Probability*. American Institute of Aeronautics and Astronautics, 2008.

[22] Romain Serra, Denis Arzelier, Mioara Joldes, Jean Bernard Lasserre, Aude Rondepierre, and Bruno Salvy. Fast and accurate computation of orbital collision probability for short-term encounters. *Journal of Guidance, Control, and Dynamics*, 39(5):1009–1021, aug 2016.

[23] Salvatore Alfano. Review of Conjunction Probability Methods for Short-term Encounters. *AAS Paper*, 127 PART 1(07-148):719–746, 2007.