Hindawi
International Journal of Aerospace Engineering
Volume 2021, Article ID 9921555, 19 pages
https://doi.org/10.1155/2021/9921555

Research Article

Evolutionary Optimization of Multirendezvous Impulsive Trajectories

Lorenzo Federici, Alessandro Zavoli, and Guido Colasurdo

Department of Mechanical and Aerospace Engineering, Sapienza University of Rome, Via Eudossiana 18, 00184 Rome, Italy

Correspondence should be addressed to Alessandro Zavoli; alessandro.zavoli@uniroma1.it

Received 12 March 2021; Revised 19 April 2021; Accepted 29 April 2021; Published 27 May 2021

Academic Editor: Franco Bernelli-Zazzera

This paper investigates the use of evolutionary algorithms for the optimization of time-constrained impulsive multirendezvous missions. The aim is to find the minimum-ΔV trajectory that allows a chaser spacecraft to perform, in a prescribed mission time, a complete tour of a set of targets, such as space debris or artificial satellites, which move on the same orbital plane at slightly different altitudes. For this purpose, a two-level design approach is pursued. First, an outer-level combinatorial problem is defined, dealing with the simultaneous optimization of the sequence of targets and the rendezvous epochs. The suggested approach is first tested by assuming that all transfer legs last exactly the same amount of time; then, the time domain is discretized over a finer grid, allowing a more appropriate sizing of the time window allocated for each leg. The outer-level problem is solved by an in-house genetic algorithm, which features an effective permutation-preserving solution encoding. A simple, but fairly accurate, heuristic, based on a suboptimal four-impulse analytic solution of the single-target rendezvous problem, is used when solving the combinatorial problem for a fast guess at the transfer cost, given the departure and arrival epochs. The outer-level problem solution is used to define an inner-level NLP problem, concerning the optimization of each body-to-body transfer leg. In this phase, the encounter times are further refined. The inner-level problem is tackled through an in-house multipopulation self-adaptive differential evolution algorithm. Numerical results for case studies including up to 20 targets with different time grids are presented.

1. Introduction

A multirendezvous (MRR) trajectory concerns the motion of an active spacecraft, the chaser, which executes a sequence of maneuvers to rendezvous with multiple passive targets, such as space debris or artificial satellites in Earth's orbit. MRR trajectories are gradually increasing in popularity within the aerospace community, since missions based on a chaser spacecraft visiting multiple artificial bodies are becoming of practical interest. Typical operational scenarios include Active Debris Removal (ADR) and On-Orbit Servicing (OOS) missions. ADR missions are particularly significant, as they are deemed to be the only effective option to ensure the long-term operation of spacecraft, close to be compromised by the huge amount of Earth orbiting wreckage [1]. On the other hand, servicing missions have recently drawn the attention of several private companies [2]. Indeed, the possibility of extending the life span of already operative satellites by sending servicing spacecraft for refueling, maintenance, and orbital relocation appears to be economically advantageous [3]. In both mission scenarios, in order to make the program economically affordable, any single chaser is expected to visit (i.e., remove or service) as many targets as possible in a limited amount of time, making the best use of the on-board propellant. As a result, a minimum-propellant multirendezvous trajectory has to be designed for the chaser spacecraft.

A number of authors addressed the optimization of ADR trajectories in Low Earth Orbit (LEO) or Sun Synchronous Orbit (SSO). Most of these studies focus on long-term or time-free ADR missions, typically aimed at deorbiting the target debris at a rate of three to ten per year [4, 5]. A typical mission strategy consists in exploiting the J_2 orbital perturbation for the alignment of the orbital planes of consecutive targets before starting the rendezvous maneuver in order to reduce the total mission cost [6, 7], at the expense of a longer flight time. Conversely, when time-constrained or time-fixed missions are considered, the removal sequence and rendezvous epochs of the ADR mission must be optimized simultaneously, making the problem more complex to solve. For this reason, most of the literature deals with very small debris clusters [8, 9], usually composed of less than five elements. Notable exceptions are Ref. [10, 11], which consider clouds with dozens or hundreds of debris.

The optimal planning of OOS missions in the geosynchronous orbit was also investigated in several research papers; the same approach for the preliminary design of ADR mission, with a single active chaser aimed at multiple passive targets, can be also found in several works about OOS [12]. However, nonstandard, and even more complex, mission scenarios become more frequent in OOS literature, encompassing the presence of multiple servicing spacecraft [13], cooperative targets [14], a fuel depot [15], constellation reconfiguration maneuvers [16], or complex dynamical models [17]. As for ADR missions, the great part of the studies about OOS deals with small constellations of satellites or time-free missions, in order to reduce the problem complexity.

The targets of an MRR mission are usually selected out of a wide set, looking for favourable body positions (e.g., GTOC9 problem [18]); the expenditure to cancel out the phase error is negligible, and suitable solutions [19, 20] can be used to estimate the cost of the time-free orbit transfer. However, most of the proposed approaches become impractical in the presence of long target sequences or tight time constraints, as the ones considered in this work. In the present manuscript, the rendezvous cost is mainly due to the recovery of the phase error.

The single-target time-constrained rendezvous is a well-known problem in spaceflight mechanics. Several optimization methods have been proposed assuming either finite [21] or impulsive thrust [22, 23]. In the latter case, four burns permit the achievement of the optimal solution [24]. The extension of the solution method to a series of consecutive rendezvous is straightforward if the sequence of targets is assigned *a priori* [25]. However, the combinatorial aspect introduced by allowing permutations of the target sequence changes radically the problem nature and increases drastically its complexity. In this respect, the MRR problem presents several analogies with the Traveling Salesman Problem (TSP) [26], a well-known problem in operational research, whose solution is the shortest tour which allows a salesman to visit once a prescribed set of cities, starting and ending the trip in his home city. For the classic TSP, branch-and-bound-based rapid exact solvers are available [27], which allow solving problem instances involving up to several hundred cities in a brief time. However, while similar, the problem here investigated is more complex than a standard TSP, as the cost associated with traveling between any two targets changes with time due to the orbital dynamics.

With this analogy in mind, several attempts have been made to find the optimal solution of the MRR problem by using algorithms developed in the context of operational research. This is the case of exact solution techniques such as brute-force approaches [28] and branch and bound search [29] which consist of a systematic enumeration of candidate solutions of the combinatorial problem. However, these solution methods are generally viable only in the presence of small sets of targets or in time-free scenarios, due to the *curse of dimensionality*, which causes the search to run out of time and/or memory very soon as the problem dimension increases [30]. For this reason, heuristic and metaheuristic approaches have rapidly gained popularity, as they usually allow to find good-quality solutions in a limited amount of time, even in the case of large-size problems. Greedy heuristics, such as the Series Method [31], are among the fastest algorithms, but they can provide solutions quite far from the global optimum. Heuristic and metaheuristic tree-search procedures, such as Beam Search [32], Ant Colony Optimization (ACO) [33], or A$*$ search [34], have been widely exploited for both ADR/OOS and asteroid exploration mission planning. A hybridization between Beam Search and ACO was also discussed [35]. These methods have proven capable to achieve high-quality results in a reduced computational time, especially when the aim of the mission is to visit the largest number of targets in a wider set, given some propellant and/or time constraints. Indeed, during the search, they allow pruning all those branches of the search tree that lead to unfeasible solutions. On the other hand, the performance of tree-search algorithms tends to worsen when a complete tour of the targets is searched for, as they repeatedly explore the same branches while looking for the optimal target permutation. Conversely, other metaheuristic methods consist of the iterative refinement of a set of (possibly random) initial solutions and guarantee the tour completeness at any point in the optimization process. For this reason, they are particularly suited to plan MRR missions aimed at visiting all the targets of the chosen set. Prominent examples are Tabu Search (TS) [36], Simulated Annealing (SA) [37, 38], and Genetic Algorithms (GA) [39, 40]

Among the aforementioned optimization methods, the genetic algorithm is the most flexible, as, in principle, it may handle any type of decision variable. GA performs a global optimization and, thanks to its stochastic selection and mutation operators, it has greater chances to evade from local optima than greedy methods. In addition, it may adopt an encoding that guarantees the intrinsic completeness of the tour. GA is a population-based method; hence, its execution time can be significantly reduced by an efficient parallel implementation that is not possible for single-solution methods like TS and SA. All these features make it a solid candidate for the solution of the complete tour problem here studied and motivate its choice.

This paper focuses on the optimization of a time-constrained multi-target rendezvous mission, assuming an impulsive thrust model for the chaser. The aim is to minimize the overall mission ΔV, while performing a complete tour of a set of targets, which move on circular coplanar orbits with similar altitudes, in a prescribed amount of time. The problem is mathematically formulated as a Mixed-Integer Nonlinear Programming (MINLP) problem, involving the simultaneous optimization of both integer variables (that define the sequence of targets) and real-valued variables (which identify the rendezvous epochs and the spacecraft trajectory from a target to the next one). A two-level design procedure is proposed to simplify the MINLP problem solution, by defining (i) an outer-level combinatorial problem that concerns the determination of the optimal rendezvous sequence and epochs, discretized over a time grid, and (ii) an inner-level NLP problem, which optimizes each body-to-body transfer leg and is also able to refine the encounter times, while keeping the body sequence that solved the outer-level problem. The task of solving the outer-level problem is entrusted to a genetic algorithm, which encodes the solution as a permutation of the target bodies and uses permutation-preserving crossover and mutation operators. Specifically, two variants of the outer-level problem, with different complexities, are considered in this paper, by assuming either equal or different (in a set of discrete values) time-lengths for the target-to-target transfers. In the latter case, the permutation of the targets is efficiently replaced by a permutation of the available rendezvous epochs. The solution of the outer-level problem is given as an input to the inner-level problem that is tackled through EOS, an in-house multi-population self-adaptive Differential Evolution algorithm. As case studies, different ADR missions are considered, including up to 20 targets in close coplanar and circular orbits.

2. Problem Statement

Let us consider a set of N prescribed target bodies that move on circular coplanar orbits at slightly different altitudes under a Keplerian dynamical model. For each target body $i \in [1, N]$, orbital radius $r^{(i)}$ and initial anomaly $\theta^{(i)}(t_0)$ are known. The velocity of the i-th target is constant and equal to $v^{(i)} = \sqrt{\mu/r^{(i)}}$, while its angular position at any time t is given by $\theta^{(i)}(t) = \theta^{(i)}(t_0) + \sqrt{\mu/(r^{(i)})^3}(t - t_0)$, where μ is the gravitational parameter of the central body. The corresponding position $\mathbf{r}^{(i)}(t)$ and velocity $\mathbf{v}^{(i)}(t)$ vectors can be easily derived. At the initial time, the chaser spacecraft is assumed to be on a circular orbit of radius $r^{(0)}$ with anomaly $\theta^{(0)}(t_0) = 0$, on the same orbital plane as all targets. The problem is thus planar. The goal is to design a minimum-ΔV impulsive trajectory that allows the chaser to rendezvous with all the N bodies within a specified mission time T_{M}.

Let us assume that the body encounter sequence $\mathbf{p} = \{p_1, p_2, \cdots, p_N\}$ and the corresponding set of (monotonically increasing) encounter times $t = \{t_1, t_2, \cdots, t_N\}$ are known, where the integer $p_j \in [1, N]$ identifies the target met at time t_j, and $t_N \le T_{\mathrm{M}}$. The overall trajectory of the chaser can be decomposed into N target-to-target transfer legs. The $(k+1)$-th leg departs from body p_k (with $p_0 = 0$ denoting the initial state of the chaser) at time t_k and arrives at body p_{k+1} at time t_{k+1}, for $k = 0, \cdots, N-1$. The rendezvous condition requires that, at time t_k, the position $\mathbf{r}$ and velocity $\mathbf{v}$ of the chaser are the same as the encountered target:

$$\mathbf{r}_k = \mathbf{r}^{(p_k)}(t_k), \quad k = 1, 2, \cdots, N, \tag{1}$$

$$\mathbf{v}_k = \mathbf{v}^{(p_k)}(t_k), \quad k = 1, 2, \cdots, N, \tag{2}$$

where subscript "k" is used to identify the chaser state at time t_k.

With four being the maximum number of impulses required for an optimal time-fixed planar rendezvous [24], each body-to-body transfer leg can be thought as a sequence of three ballistic arcs, named A, B, and C, joined by impulsive maneuvers, located at the departure point (k), at two internal points (with labels $k+1/3$ and $k+2/3$), and at the arrival point ($k+1$), for the $(k+1)$-th leg, respectively. A sketch of this transfer leg is shown in Figure 1. A position formulation is here considered; that is, the $(k+1)$-th transfer leg is parameterized with respect to radii $r_{k+1/3}$ and $r_{k+2/3}$ and anomalies $\theta_{k+1/3}$ and $\theta_{k+2/3}$ of the chaser at the internal maneuvering points. Since the chaser state $\boldsymbol{r}, \boldsymbol{v}$ at the endpoints of any trajectory leg is fixed because of the rendezvous conditions in Equations (1) and (2), the chaser position at any impulse is completely defined. Conversely, the unknown chaser velocity immediately before (superscript "-") and after (superscript "+") the impulsive maneuvers can be found by solving either a geometrical problem or a Lambert problem.

2.1. Geometrical Problem. Consider the arc A between the points k and $k+1/3$. As shown in Figure 2, for any given semimajor axis $a > a_m$, two elliptical transfer orbits exist that connect $\mathbf{r}_k$ with $\mathbf{r}_{k+1/3}$, where $a_m = (r_k + r_{k+1/3} + c)/4$ is the smallest possible semimajor axis and $c = \|\mathbf{r}_k - \mathbf{r}_{k+1/3}\|$ is the chord distance between the two endpoints [41]. An ellipse is known as the *fast* solution, since it is characterized by a transfer time $\Delta t < t_m$; the second ellipse, instead, represents the *slow* solution, since its transfer time is $\Delta t > t_m$ (Figure 2(a)). As a consequence, the knowledge of the semimajor axis is not sufficient to identify the elliptic arc connecting the two points.

A nondimensional parameter $y \in [0, 1]$ is introduced, which is implicitly defined by the relationship

$$a = \frac{a_m}{4y(1-y)}. \tag{3}$$

Note that the same value of the semimajor axis a is obtained by using either y or $1-y$. So, by pairing the fast solutions with values $y < 0.5$ and the slow solutions with values $y > 0.5$, the elliptic arc connecting the two assigned endpoints is uniquely identified for any choice of y. Figure 2(b) depicts the geometric construction that allows for an unambiguous definition of the transfer ellipse, that is, of its semimajor axis a, eccentricity e, and argument of pericenter ω, when y_{k_A}, r_k, $r_{k+1/3}$, and $\Delta\theta_{k+1/3}$ are known. More precisely, the chord length

$$c = \sqrt{r_k^2 + r_{k+1/3}^2 - 2r_k\, r_{k+1/3} \cos \Delta\theta_{k+1/3}} \tag{4}$$

and the lengths $c_1 = 2a - r_k$ and $c_2 = 2a - r_{k+1/3}$ are evaluated first. Angle γ follows from

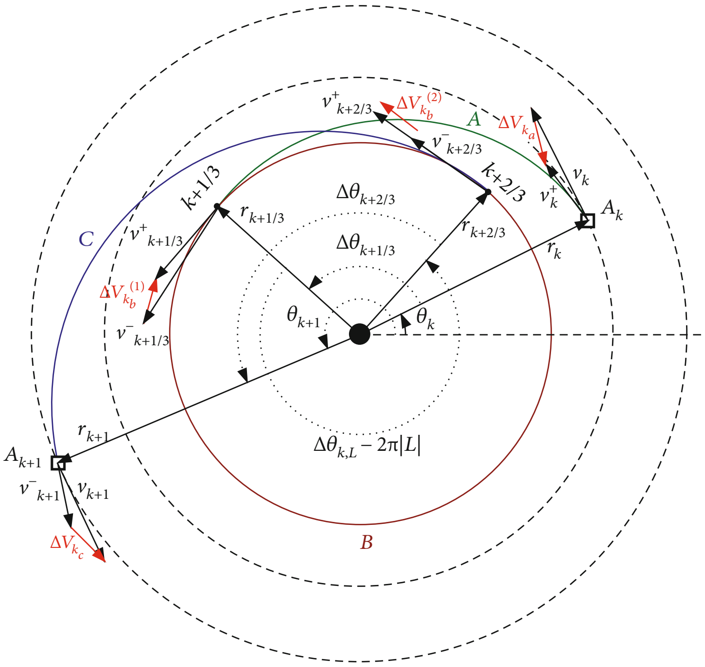


Figure 1: Trajectory sketch for the $(k+1)$-th leg.

$$\gamma = \cos^{-1}\left(\frac{r_k^2 - r_{k+1/3}^2 + c^2}{2 r_k c}\right). \tag{5}$$

Hence, $\gamma_f = \gamma - \gamma_c$, where

$$\gamma_c = \begin{cases} \cos^1\left(\dfrac{c_1^2 - c_2^2 + c^2}{2c_1 c}\right), & \text{if } y \le 0.5, \\ -\cos^1\left(\dfrac{c_1^2 - c_2^2 + c^2}{2c_1 c}\right), & \text{if } y > 0.5. \end{cases} \tag{6}$$

The eccentricity can be found as

$$e = \frac{\sqrt{c_1^2 + r_k^2 - 2c_1\, r_k \cos\gamma_f}}{2a}. \tag{7}$$

Eventually, the initial true anomaly ν_k and final true anomaly $\nu_{k+1/3}$ along the transfer orbit are evaluated by means of the following relations:

$$\begin{aligned} r_k &= \frac{a\left(1-e^2\right)}{\left(1 + e\cos\nu_k\right)}, \\ r_{k+1/3} &= \frac{a\left(1-e^2\right)}{\left(1 + e\cos\nu_{k+1/3}\right)}, \end{aligned} \tag{8}$$

and ambiguities are easily solved as the angle $\Delta\theta_{k+1/3} = \nu_{k+1/3} - \nu_k$ is known.

Velocities at both endpoints ($\mathbf{v}_k^+$ and $\mathbf{v}_{k+1/3}^-$) follow from the standard equations of the two-body problem. The transfer time Δt_A can be also evaluated through the Kepler equation, and consequently, the epoch of the intermediate maneuver $t_{k+1/3} = t_k + \Delta t_A$ can be obtained.

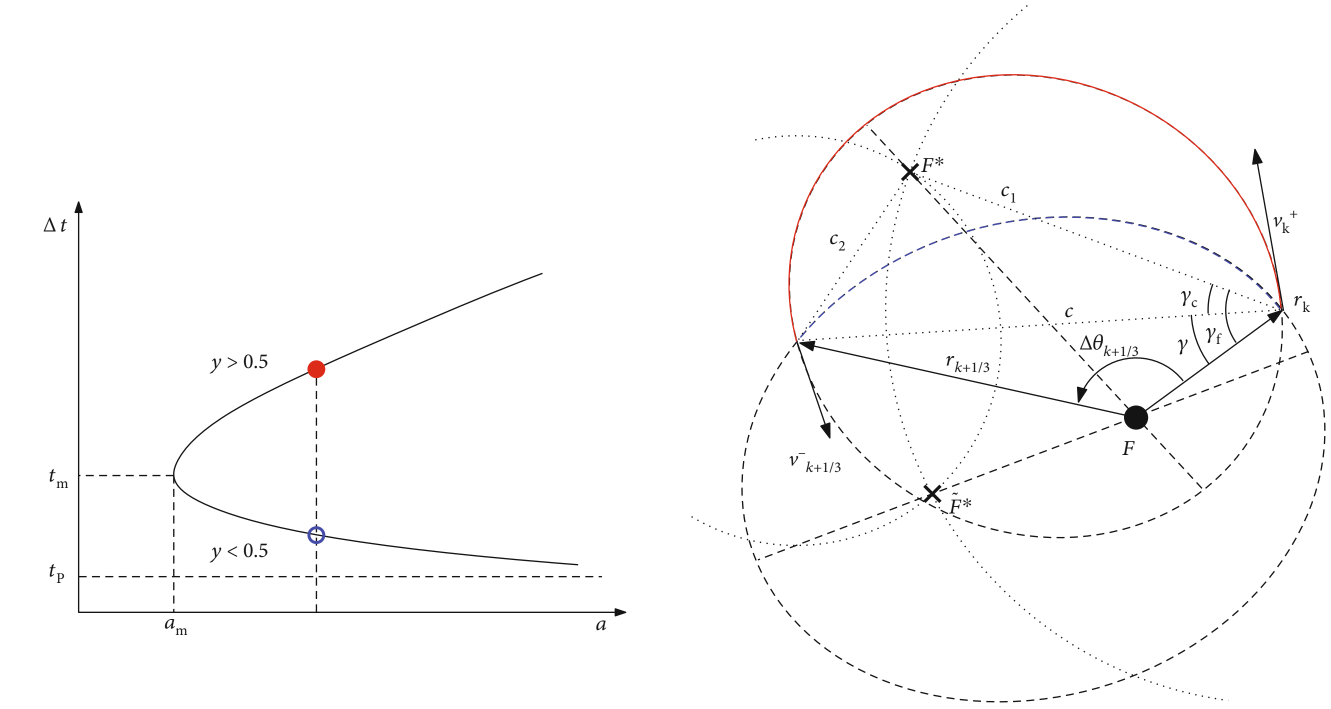
The same procedure can be applied to the arc C, which connects point $k+2/3$ to point $k+1$. If this geometrical construction is defined as a procedure $y\text{Arc}()$, one has as a result

$$\begin{aligned} \left[\mathbf{v}_k^+, \mathbf{v}_{k+1/3}^-, t_{k+1/3}\right] &\longleftarrow y\text{Arc}\left(r_k, r_{k+1/3}, \Delta\theta_{k+1/3}, y_{k_A}\right), \\ \left[\mathbf{v}_{k+2/3}^+, \mathbf{v}_{k+1}^-, t_{k+2/3}\right] &\longleftarrow y\text{Arc}\left(r_{k+2/3}, r_{k+1}, \Delta\theta_{k+2/3}, y_{k_C}\right). \end{aligned} \tag{9}$$

The cost of the maneuvers performed at the departure and arrival points of the $(k+1)$th leg are thus evaluated as

$$\begin{aligned} \Delta V_{k_A} &= \left\|\mathbf{v}_k^+ - \mathbf{v}_k\right\|, \\ \Delta V_{k_C} &= \left\|\mathbf{v}_{k+1} - \mathbf{v}_{k+1}^-\right\|. \end{aligned} \tag{10}$$

2.2. Multirevolution Lambert Problem. The central arc B, which connects points $k+1/3$ and $k+2/3$, cannot be dealt with in the same fashion. In fact, for a given choice of the parameters y_{k_A} and y_{k_C}, the maneuvering epochs $t_{k+1/3}$ and $t_{k+2/3}$ and, consequently, the travel time along arc B, $\Delta t_B = t_{k+2/3} - t_{k+1/3}$, are assigned. Conversely, a multirevolution Lambert problem can be formulated, being the position vectors $\mathbf{r}_{k+1/3}$ and $\mathbf{r}_{k+2/3}$ and the travel time Δt_B known. This problem admits $1 + 2n_{\max}$ solutions, where $n_{\max}$ is the maximum possible number of revolutions: one solution for the 0-revolution transfer arc and two solutions, namely, left and right branch, for each n-revolution transfer orbit. Let us introduce an integer parameter $L \in [-n_{\max}, n_{\max}]$, which indicates the solution corresponding to the $|L|$-revolution transfer orbit,



(a) Flight time Δt on the transfer ellipse as a function of the semimajor axis a

(b) Geometrical construction of the two transfer ellipses

Figure 2: Geometrical problem. The filled marker in (a) refers to the slow solution that corresponds to the solid-line transfer arc in (b); the empty marker in (a), instead, refers to the fast solution, represented with a dashed line in (b).

right branch for $L > 0$ and left branch otherwise. By knowing L, the velocity vector immediately after the second impulse $\mathbf{v}^+_{k+1/3}$ and right before the third one $\mathbf{v}^-_{k+2/3}$ can be evaluated according to the algorithm by Izzo [42], that is,

$$[\mathbf{v}^+_{k+1/3}, \mathbf{v}^-_{k+2/3}] \longleftarrow \mathrm{Lambert}(\mathbf{r}_{k+1/3}, \mathbf{r}_{k+2/3}, t_{k+2/3} - t_{k+1/3}\,; L). \quad (11)$$

Hence, the total cost of the two internal maneuvers is

$$\Delta V_{k_B} = \|\mathbf{v}^+_{k+1/3} - \mathbf{v}^-_{k+1/3}\| + \|\mathbf{v}^+_{k+2/3} - \mathbf{v}^-_{k+2/3}\|. \quad (12)$$

Instead of treating L as an optimization variable, an enumeration approach is used; that is, all the $2n_{\max} + 1$ possible solutions are computed and the one with the lowest total ΔV is chosen. When considering transfers between close orbits, an educated guess safely restricts the analysis to just three scenarios; that is, the chaser performs the same number of revolution as the target, one more, or one less, corresponding to a total of six solutions (three right and three left branches) to be compared.

2.3. MINLP Formulation. According to the proposed problem formulation, the overall multi-target trajectory can be parameterized by using a set of $8N$ design variables. Hence, the solution vector is

$$\mathbf{x} = \bigcup_{k=0}^{N-1} \mathbf{x}_k, \quad (13)$$

where

$$\mathbf{x}_k = \left\{ r_{k+1/3}, \Delta\theta_{k+1/3}, y_{k_A}, r_{k+2/3}, \Delta\theta_{k+2/3}, y_{k_C} \right\} \cup \{p_{k+1}, t_{k+1}\}. \quad (14)$$

Eventually, the impulsive MRR trajectory optimization problem can be formulated as

$$P = \begin{cases} \min\limits_{\mathbf{x}} \Delta V_{\mathrm{tot}}(\mathbf{x}) \\ \text{s.t.} \quad \mathbf{x}_\mathrm{L} \le \mathbf{x} \le \mathbf{x}_\mathrm{U}, \end{cases} \quad (15)$$

where the overall cost of the MRR trajectory is

$$\Delta V_{\mathrm{tot}} = \sum_{k=0}^{N-1} \left(\Delta V_{k_A} + \Delta V_{k_B} + \Delta V_{k_C} \right) \quad (16)$$

and $\mathbf{x}_\mathrm{L}$ and $\mathbf{x}_\mathrm{U}$ are the lower and upper bounds of the design variables, respectively. This problem involves the simultaneous optimization of both integer variables (defining the encounter sequence) and real-valued variables (radii and anomalies at the internal maneuvers, y parameters, and encounter epochs). For this reason, problem (15) is an unconstrained Mixed-Integer Nonlinear Programming (MINLP) problem.

3. Two-Level Optimization Approach

The MINLP problem in Equation (15) belongs to the class of NP-hard problems [43]; hence, no deterministic algorithm is able to find the optimal solution in polynomial time. For this reason, a variety of stochastic metaheuristic techniques have been developed over the last decades with the aim of attaining (often suboptimal) good-quality solutions to the problem in a short time. However, as the problem dimension increases, also the computational time required to obtain a nearly optimal solution increases and may become prohibitive.

Instead of solving the problem as a whole, an alternative option is to decompose the MINLP into simpler subproblems, which could be, more or less easily, solved sequentially and whose solutions are eventually recomposed into a, hopefully near-optimal, solution to the original problem. For the problem at hand, a two-level approach can be pursued, by isolating (i) an outer-level combinatorial problem, which concerns the definition of the optimal rendezvous sequence and encounter epochs, discretized over an arbitrarily dense time-grid, while neglecting the details of each body-to-body transfer leg and (ii) an inner-level NLP problem, which deals with the accurate optimization of each body-to-body transfer, on the basis of the body sequence and the discrete encounter epochs just computed; these epochs may be left fixed or further refined at this level.

The two layers are interconnected: the outer level requires a measure of the cost associated with each transfer leg to evaluate the quality of any encounter sequence, even though the actual ΔV of each leg can be evaluated only by solving the NLP problem referred to this transfer, that is, the inner-level problem. On the other hand, the inner level requires the definition of the encounter sequence and rendezvous epochs, which, in turn, are the output of the combinatorial, outer-level problem. In practice, the two problems could be solved sequentially provided that an alternative way, that is, a "heuristic", exists for attaining a reasonable and, possibly, inexpensive estimation of any transfer cost during the solution of the outer-level problem, without the need of solving the complete NLP problem. Once the heuristic has been established, the outer-level combinatorial problem, or *tour problem*, is isolated and solved first; then, its solution is given as input to the inner-level problem, which returns the full-problem solution as output.

3.1. Cost Estimate for a Single Rendezvous Leg. This section presents an analytical, suboptimal, four-impulse transfer strategy that gives, in a short computational time, a conservative estimate of the optimal ΔV for a planar, time-fixed, impulsive rendezvous between circular orbits. When the allowed travel time is sufficiently large, this strategy fairly approximates the performance of the optimal solution [44]. Furthermore, with respect to other heuristics adopted in similar studies [45], the proposed approach represents a real, feasible, mission scheme.

Let $\theta_{1,0}$ and $\theta_{2,0}$ be the true anomaly at time $t = 0$ of the chaser and target, which fly on circular orbits of radius r_1 and r_2, respectively. The two orbits are coplanar and have angular velocities ω_1 and ω_2, respectively. Assuming that the two orbits are not extremely far apart, the minimum-ΔV solution is represented by a Hohmann transfer, of semimajor axis $a_{12} = (r_1 + r_2)/2$ and duration $T_{\mathrm{H}_{12}} = \pi\sqrt{a_{12}^3/\mu}$. The transfer may be preceded by a coasting arc on the departure orbit, which allows the two bodies to achieve the correct relative phasing $\gamma = \pi - \omega_2 T_{\mathrm{H}_{12}}$ required by this kind of maneuver. After the phase error $\Delta\gamma = \theta_{2,0} - \theta_{1,0} - \gamma$ has been evaluated and normalized between 0 and 2π, the time length T_{wait} of the coast arc can be evaluated as

$$T_{\mathrm{wait}} = \begin{cases} \dfrac{\Delta\gamma}{\omega_1 - \omega_2}, & \text{if } r_1 < r_2, \\[2ex] \dfrac{2\pi - \Delta\gamma}{\omega_2 - \omega_1}, & \text{if } r_2 < r_1. \end{cases} \tag{17}$$

By comparing the maximum available transfer time $T_{\max}$ with the sum of the coasting time T_{wait} and the time spent on the Hohmann transfer ellipse $T_{\mathrm{H}_{12}}$, it is possible to obtain a condition for the viability of the Hohmann transfer

$$T_{\mathrm{wait}} + T_{\mathrm{H}_{12}} \le T_{\max}. \tag{18}$$

Whenever Equation (18) is satisfied, the rendezvous has the cost ΔV_{H} of the Hohmann transfer. Otherwise, the cost is evaluated according to the mission scheme depicted in Figure 3. The chaser spacecraft is injected into a circular (either internal or external) waiting orbit of radius r_3 with a Hohmann transfer "1–3," of semimajor axis $a_{13} = (r_1 + r_3)/2$ and duration $T_{\mathrm{H}_{13}} = \pi\sqrt{a_{13}^3/\mu}$. Once the chaser has reached the correct phasing with respect to the target body, a second Hohmann transfer "3–2," of semimajor axis $a_{32} = (r_2 + r_3)/2$ and duration $T_{\mathrm{H}_{32}} = \pi\sqrt{a_{32}^3/\mu}$, is performed to complete the rendezvous.

The problem now reduces to finding the radius r_3 that satisfies the rendezvous condition. The phase angle recovered by the chaser is

$$\Theta = 2\pi + \left(T_{\max} - T_{\mathrm{H}_{13}} - T_{\mathrm{H}_{32}}\right)\omega_3 - T_{\max}\omega_2, \tag{19}$$

with $\Theta > 0$ for $r_3 < r_1$ and vice versa. If the initial phase angle $\Delta\theta_0 = \theta_{2,0} - \theta_{1,0}$ is normalized between 0 and 2π, the equality between the chaser and target positions at the end of the maneuver requires

$$\Theta = \Delta\theta_0 - 2k_{\mathrm{ext}}\pi, \tag{20}$$

where $k_{\mathrm{ext}} = 1$ for an external maneuver and $k_{\mathrm{ext}} = 0$ when $r_3 < r_1$. The cheaper solution is selected.

An alternate 3-impulse heuristic outlines the rendezvous maneuver as an n-revolution bielliptic transfer split into two legs, each performing 1/2 or $(n - 1/2)$ revolutions. In other words, the intermediate constant-radius arc of the 4-impulse heuristic is replaced by adding $(n - 1)$ complete revolutions to either ellipse. The radius of the intermediate impulse is selected in order to eventually achieve the

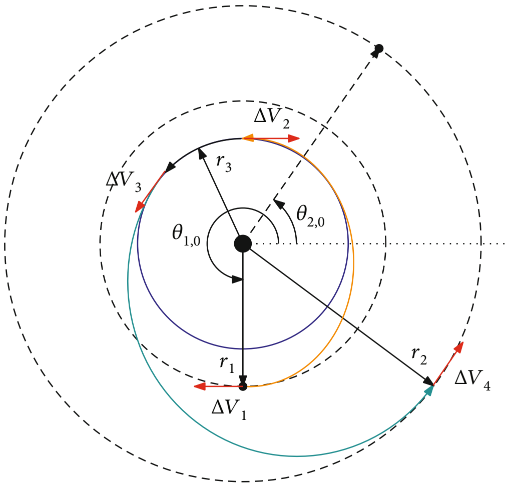


FIGURE 3: Chaser trajectory according to the suboptimal rendezvous strategy adopted as heuristic.

rendezvous with the target body. The angular length of the transfer is constrained at an integer number of revolutions. The time length is lower than the allocated time and the maneuver is completed by adding a coasting either on the initial or the final orbit. The 3-impulse heuristic better resembles the optimal rendezvous maneuver than the 4-impulse heuristic, but other features advice against its use.

3.2. Outer-Level Problem. Under the assumption that all bodies fly on coplanar circular orbits and that the allowed mission time is sufficiently large, the optimal transfer between any pair of bodies is a Hohmann transfer. The cost $\Delta V(i, j)$ of the transfer from body i to body j is independent of the specific departure and arrival epochs. The problem reduces to the search of the body sequence that minimizes the total velocity change.

Let $\mathbf{p} = \{p_1, p_2, \cdots, p_N\} \in \mathscr{P}^N$ be a permutation of the N integers $\{1, 2, \cdots, N\}$, which identify the target bodies. So, the set $\mathscr{P}^N$ is made up of all the $N!$ possible permutations of N integers. The time-free optimization problem can be written as

$$\min_{\boldsymbol{p}\in\mathscr{P}^N} \sum_{k=1}^{N} \Delta V(p_{k-1}, p_k), \tag{21}$$

where $p_0 = 0$ denotes the chaser at the initial time. In order to speed up the solution algorithm, the transfer cost for any pair of arrival and departure bodies can be preliminarily evaluated and collected in a two-dimensional cost tensor ΔV_{ij} of dimension $(N + 1) \times N$. An attentive reader will notice that this problem closely resembles the classical and well-studied Traveling Salesman Problem (TSP).

The solution of the previous problem provides a lower bound on the tour overall cost, but such a limited value may be quite far from the optimal solution of the original problem. In fact, due to the existing constraint on the overall mission duration, waiting for the perfect phasing required by the Hohmann solution may not be a viable option for missions involving many transfers.

In planar time-constrained rendezvous maneuvers, the transfer costs are highly sensitive to the initial phasing, that is, to the departure epoch and to the transfer time-length. The following sections present two formulations, of increasing complexity, of the outer-level combinatorial problem, which is aimed at determining the optimal body sequence and the rendezvous epochs, discretized over a grid in the admissible time window.

3.2.1. Uniform Time-Length Tour. Under the simplifying assumption that the duration of all transfers is exactly the same, a relaxed problem can be formulated, where the transfer cost is only dependent on the departure epoch. In this scenario, the epoch of the k-th rendezvous is readily available as $t_k = k\, T_{\mathrm{M}}/N$. The problem reduces again to finding the optimal permutation $\boldsymbol{p}^\star \in \mathscr{P}^N$ of the N targets but, in this case, the cost of the transfer from body i to body j also depends on the position k of the leg within the sequence of transfers. The *uniform time-length tour* optimization problem can be written as

$$\min_{\boldsymbol{p}\in\mathscr{P}^N} \sum_{k=1}^{N} \Delta V(p_{k-1}, p_k, k), \tag{22}$$

where $\Delta V(i, j, k)$ indicates the cost to move from target i to target j, departing at time t_{k-1} and arriving at time t_k. Again, all transfer costs can be preliminarily evaluated and collected in a three-dimensional cost tensor ΔV_{ijk} with dimension $(N + 1) \times N \times N$.

By extending the analogy between time-free tour and TSP, this novel combinatorial optimization problem is similar to a time-dependent version of TSP [46, 47], named TDTSP-A by the authors in a previous work [38], where the cost of any city-to-city transfer also depends on its position in the whole tour.

3.2.2. Discrete Time-Length Tour. If the possible rendezvous epochs are discretized over a finer time grid, one obtains a combinatorial optimization problem with a solution closer to that of the original time-continuous problem and a formulation which is still close, to a certain extent, to that of the uniform time-length tour defined in the previous section.

Let $\tau = \{\tau_1, \tau_2, \cdots, \tau_{N_t}\}$ be a time grid composed of $N_t = ND$ equidistant points, such that $\tau_{k+1} - \tau_k = \Delta\tau = T_{\mathrm{M}}/N_t$. Let $\Delta V(i, j, \tau_h, \tau_{h+m})$ denote the cost to move from body i to body j, departing at time τ_h and arriving at time τ_{h+m}. In this case, both the target sequence and the location of the rendezvous epochs over the grid are required for the computation of the tour overall cost. However, the problem can still be formulated using a single permutation $\mathbf{\Pi} \in \mathscr{P}^{N_t}$ of the integers $\{1, 2, \cdots, N_t\}$, i.e., with a number of elements equal to the discrete number of available rendezvous epochs. In fact, the elements of $\mathbf{\Pi}$ with a value lower or equal to N constitute the target sequence $\boldsymbol{p} = \{p_1, p_2, \cdots, p_N\}$, while their position in $\mathbf{\Pi}$ reveals which elements of τ make up the vector of rendezvous epochs $\mathbf{t} = \{t_1, t_2, \cdots, t_N\}$.

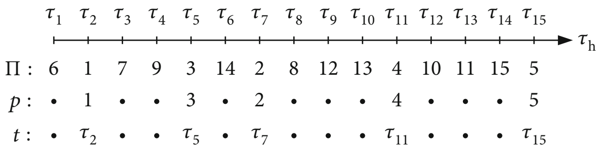

τ_1 τ_2 τ_3 τ_4 τ_5 τ_6 τ_7 τ_8 τ_9 τ_{10} τ_{11} τ_{12} τ_{13} τ_{14} τ_{15}
τ_h
Π : 6 1 7 9 3 14 2 8 12 13 4 10 11 15 5
p : • 1 • • 3 • 2 • • • 4 • • • 5
t : • τ_2 • • τ_5 • τ_7 • • • τ_{11} • • • τ_{15}

Figure 4: Example of permutation decoding for a discrete time-length tour with $N = 5$ and $D = 3$.

As a result, the *discrete time-length tour* optimization problem can be written as

$$\min_{\Pi \in \mathscr{P}^{N_t}} \sum_{k=1}^{N} \Delta V(p_{k-1}, p_k, t_{k-1}, t_k), \tag{23}$$

$$\mathbf{p} = \{\Pi_h \mid \Pi_h \le N, \forall h \in [1, N_t]\}, \tag{24}$$

$$\mathbf{t} = \{\tau_h \mid \Pi_h \le N, \forall h \in [1, N_t]\}. \tag{25}$$

An example of the adopted permutation encoding is presented in Figure 4 for $N = 5$ and $D = 3$, with the corresponding extraction of target sequence $\mathbf{p} = \{1, 3, 2, 4, 5\}$ and rendezvous epochs $\mathbf{t} = \{\tau_2, \tau_5, \tau_7, \tau_{11}, \tau_{15}\}$.

For the sake of brevity, a tour with N targets and $N_t = ND$ possible rendezvous epochs will be named $N \times D$ tour in the following. Of course, if $D = 1$, a uniform time-length tour is obtained. For this reason, D is defined as a *time division factor*.

As in the other tour problems, all transfer costs can be precomputed to speed up the optimization process. A 4-order tensor ΔV_{ijhm} of dimension $(N + 1) \times N \times ND \times N(D - 1)$ is needed in principle to store all the computed values. However, because of the monotonic, nonincreasing, behavior of the ΔV with the transfer time, one may decide to limit the calculus to transfers of duration lower than or equal to $M\Delta\tau$, for an arbitrary $M \in [1, N(D - 1))$, and assume the same cost for longer transfers. In this case, a 4-order tensor with dimension $(N + 1) \times N \times ND \times M$ would be sufficient. In addition to reducing the size of the tensor, this treatment has the additional advantage of guiding the solver towards a trajectory with more uniform travel times, which may be preferable from an operational point of view.

It is advisable to keep the time division factor D sufficiently small, because (i) a rough evaluation of the encounter epochs is sufficient, as the attained epochs can be further refined in the inner-level optimization step; (ii) a large number of divisions make the problem too similar to the original MINLP and hence more difficult to solve; and (iii) the proposed heuristic is reasonably accurate if the chaser has enough time to perform several revolutions around the central body; so, reducing the minimum allowed travel time $\Delta\tau$ makes the heuristic less reliable and may undermine our efforts.

3.3. Inner-Level Problem. This step is aimed at improving the provisional trajectory provided by the solution of the outer-level problem, based on suboptimal maneuvers and rendezvouses located at the nodes of a discrete time grid. Assuming that the optimal target sequence $\mathbf{p}^\star = \{p_1^\star, \cdots, p_N^\star\}$ and the discrete rendezvous epochs $\bar{\mathbf{t}} = \{\bar{t}_1, \cdots, \bar{t}_N\}$ have been determined, the MINLP problem in Equation (15) reduces to an unconstrained NLP problem, where the design variables are

$$\mathbf{x}_{\text{NLP}} = \bigcup_{k=0}^{N-1} \left\{ r_{k+1/3}, \Delta\theta_{k+1/3}, y_{k_A}, r_{k+2/3}, \Delta\theta_{k+2/3}, y_{k_C} \right\} \cup \{\alpha_{k+1}\}, \tag{26}$$

with

$$t_k = \bar{t}_k + \begin{cases} \alpha_k(\bar{t}_k - \bar{t}_{k-1}), & \text{if } \alpha_k < 0, \\ \alpha_k(\bar{t}_{k+1} - \bar{t}_k), & \text{if } \alpha_k \ge 0, \end{cases} \tag{27}$$

the continuous rendezvous epochs.

Two scenarios can be investigated: (i) *time-fixed*: the encounter epochs t_k are kept fixed at their nominal values $\bar{t}_k$ (so, $\alpha_k = 0, \forall k$), and (ii) *time-free*: the encounter epochs are free to be optimized. In the first case, each body-to-body transfer can be optimized independently from the others, thus reducing the $6N$-variable problem to solving N easier subproblems, each one of just 6 variables. In the second case, the encounter epochs may vary in a neighborhood of the reference values, leading to an improved solution, but the entire trajectory must be optimized as a single $7N$-variable problem.

The lower and upper bounds of the design variables in Equation (26) are selected as follows:

$$\begin{aligned} r &\in [0.95 r_0, 1.05 r_0], \\ \Delta\theta &\in [0, 2\pi], \\ y &\in [0.1, 0.9], \\ \alpha_k &\in \begin{cases} [-1, 0.5], & \text{if } k = 1, \\ [-0.5, 0.5], & \text{if } 1 < k < N - 1, \\ [-0.5, 1], & \text{if } k = N - 1, \\ [0, 0], & \text{if } k = N. \end{cases} \end{aligned} \tag{28}$$

The inner-level problem is tackled in this paper by EOS (Evolutionary Optimization at Sapienza) [48]. EOS is an in-house optimization code for continuous-variable problems, which implements an improved self-adaptive, partially restarted Differential Evolution (DE) algorithm, with a synchronous island model to handle multiple populations in parallel. EOS was developed in the contest of the Global Trajectory Optimization Competitions [49, 50] and has proven to be a flexible and high-performance algorithm, able to successfully cope with a broad range of real-world unconstrained and constrained space trajectory optimization problems [51, 52]. The values of the hyperparameters used in this work are reported in Table 1. Interested readers are suggested to refer to Ref. [48] for further details about the algorithm.

Table 1: EOS hyperparameters.

Hyperparameter	Symbol	Value
Number of runs	N_r	15
Subpopulations	N_i	16
Individuals per population	N_P	128
Generations	N_G	20000
Generations between migrations	N_mig	100
Probability of migration	ϕ_mig	0.5
Fraction of migrating individuals	ρ_mig	0.05

4. Genetic Algorithm

A genetic algorithm (GA) is used in this paper for solving the outer-level combinatorial optimization problems described in Section 3.2. Genetic algorithms are well-known population-based metaheuristic optimization techniques inspired by the process of natural selection, firstly introduced by Holland in 1960 [53]. They have been successfully applied to a wide range of real-world problems of significant complexity. A brief overview of the algorithm is here presented. Interested readers can find further details in Ref. [54]. A block diagram of the basic algorithm is proposed in Figure 5, highlighting the presence of three fundamental genetic operators (selection, crossover, and mutation) which form the backbone of any GA implementation.

An initial *population*, that is, a collection of n_P candidate solutions to the optimization problem (often referred as *individuals* or *chromosomes*), is randomly generated, aiming at covering the search space as uniformly as possible. Each individual is composed of a number of consecutive elements, called *genes*, which identify specific values of the design variables. At the end of each iteration (or *generation*), the *fitness* f (that is, the value of the function to optimize) of each individual in the current population is evaluated. Then, the next generation starts with the creation of a *mating* population, by selecting a number of individuals from the current population according to a *selection operator*, which tries to promote individuals with a good fitness, without sacrificing diversity too much. Typical selection operators are Stochastic Roulette Wheel and Stochastic Tournament [54], the latter being the one used in the present application. Next, the individuals (or *parents*) in the mating population are randomly paired and their genes combined to produce new solutions (or *offspring*), according to a *crossover* operator. The underlying idea is that, by combining somehow good solutions, it is possible to create new solutions which hopefully are better than the previous ones. The process is repeated until a new offspring population (usually with the same dimension as the previous one) is created. Then, a few randomly chosen individuals of the offspring population undergo a *mutation* process, with the aim of increasing the population diversity. The mutation operator changes, in a random way, some genes of the chosen individuals. This new population eventually replaces the previous one, and the process is repeated until some termination criterion (e.g., a maximum number of generations n_G) has been met. As a minor, yet important, tweak, *elitism* is enforced: at the end of each generation, the best individual of the parent population is copied into the offspring population, replacing the worst individual. In this way, the best solution is prevented from being accidentally lost in the evolution process.

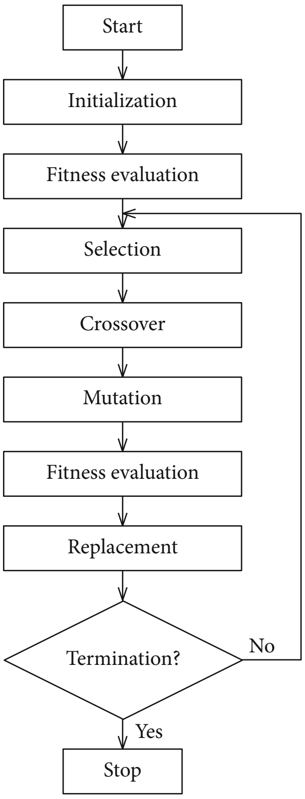


Figure 5: Block diagram of the basic genetic algorithm.

While selection operators are usually problem-independent, crossover and mutation operators are tightly related to the adopted *encoding*, that is, the way the real-word problem is described in terms of numerical variables (that is, genes). On the basis of the features of the decision variables, several encodings have been defined, with the most common being the binary, the integer-valued, and the real-valued encodings. In order to match the permutation-based formulations developed in Section 3.2, a permutation encoding is here considered; that is, each individual is a permutation $\boldsymbol{p} \in \mathscr{P}^L$, where $L = N$ for the problems in Equations (21) and (22), while $L = N_t$ for the problem in Equation (23). So, each gene takes an integer value in the range $\{1, \cdots, L\}$, and two genes of the same individual cannot have the same value at one time. Standard, general-purpose, crossover operators (such as one-point, two-point, or uniform crossover) cannot be applied to a permutation-encoded individual without producing unfeasible offspring, that is, child individuals whose genotype contains multiple copies of the same integer value. The same goes for classical mutation operators, such as random gene flip. Fortunately, an abundance of permutation-preserving genetic operators, specifically developed in the context of the TSP problem, is documented in the literature [55] and can be employed in solving the problem under investigation.

```
procedure PMX($\boldsymbol{p}^{(1)}, \boldsymbol{p}^{(2)}$)                ▷ Input: parents ($\boldsymbol{p}^{(1)}, \boldsymbol{p}^{(2)}$)
    choose randomly a, b so that $1 \le a < b \le L$
    $\boldsymbol{c}^{(1)} \leftarrow \boldsymbol{p}^{(1)}$
    $\boldsymbol{s} \leftarrow \{p_a^{(1)}, \cdots, p_b^{(1)}\}$
    for all$i \in [1, a-1] \cup [b+1, L]$do
        $v \leftarrow p_i^2$                                  ▷ Proposal
        while$v \in \boldsymbol{s}$do
            find $j$ so that $p_j^{(1)} == v$        map $\boldsymbol{p}^{(1)}$ into $\boldsymbol{p}^{(2)}$
            $v \leftarrow p_i^2$                          ▷New proposal
        end while
        $c_j^{(1)} \leftarrow v$                              ▷ Accept the proposal
    end for
    return$\boldsymbol{c}^{(1)}$                            ▷Output: child $\boldsymbol{c}^{(1)}$
end procedure
```

Algorithm 1: Partially Matched Crossover (PMX) algorithm.

The Partially Matched Crossover (PMX) devised by Goldberg and Lingle in 1985 [56] has been adopted in the present paper, as it is aimed at preserving both the order and the position of as many elements as possible from the parent individuals. A pseudocode of this algorithm is presented in Algorithm 1. The first offspring $\mathbf{c}^{(1)}$ is created starting from a swath (or *cut*) $\mathbf{s} = \{p_a^{(1)}, \cdots, p_b^{(1)}\}$ of the first parent ($\mathbf{p}^{(1)}$), in the same fashion as in a two-point crossover, being a and b two random integers in $\{1, \cdots, L\}$, such that $b > a$. The remaining entries are taken from the second parent ($\mathbf{p}^{(2)}$). For each index $i \notin [a, b]$, if the value $v = p_i^{(2)}$ does not appear in swath $\mathbf{s}$, it is directly copied to $\mathbf{c}^{(1)}$ in position i. Otherwise, conflicts are resolved by using $\mathbf{p}^{(1)}$ as a map for $\mathbf{p}^{(2)}$. So, PMX looks at position j of value v in $\mathbf{p}^{(1)}$: if $p_j^{(2)}$ is not in swath $\mathbf{s}$, the value $p_j^{(2)}$ is copied to position i in $\mathbf{c}^{(1)}$; otherwise, the procedure is repeated with $v = p_j^{(2)}$. The second child $\mathbf{c}^{(2)}$ is then created by swapping the role of the two parents.

Mutation operators are also of great relevance in the framework of GAs, as they allow escaping from situations of premature stagnation of the population on a suboptimal solution; thus, their impact on the effectiveness of the algorithm should not be underestimated. Once again, an abundance of permutation-preserving mutation operators is documented in the literature. In the present application, the mutation operator is randomly selected at each generation between three possible rules: *insert*, *swap*, and *reverse*. All operators begin by randomly selecting two indices a and b in $[1, L]$, with $a < b$. For the *insert* operator, the value in the ath position is transferred to the bth position and the other genes of the individual are shifted accordingly; the *swap* operator exchanges the two genes in position a and b; finally, the order of the genes from a-th to b-th is reverted when considering the *reverse* operator.

Some control over the population diversity is often needed to avoid an excessive uniformity between individuals during the optimization process, which would result in a very inefficient use of the crossover operator and, consequently, in a premature stagnation of the population in a local optimum. To overcome this issue, an active diversity control strategy is implemented: the fitness of each individual is scaled by a penalty factor, which measures how many other individuals in the current population are similar to it. Let $d_{i,k}$ be the distance in the state space between the i-th individual $\mathbf{p}^{(i)}$ and the k-th individual $\mathbf{p}^{(k)}$. Specifically, the Hamming distance [57] is used to measure the difference between permutations, essentially counting the number of positions at which the corresponding elements are different between the two individuals:

$$d_{i,k} = \sum_{j=1}^{L} \left[p_j^{(i)} \neq p_j^{(k)} \right], \tag{29}$$

where $[x]$ is the Iverson bracket function, or indicator function, which returns 1 if x is true, 0 otherwise.

The augmented (i.e., penalized) fitness of the i-th individual is evaluated as

$$f_i^{\text{aug}} = f_i \, e^{d_i/\delta_{\text{dc}}}, \tag{30}$$

where $d_i = \sum_{k \neq i} d_{i,k}$ is the diversity score of the i-th individual and δ_{dc} is a control radius.

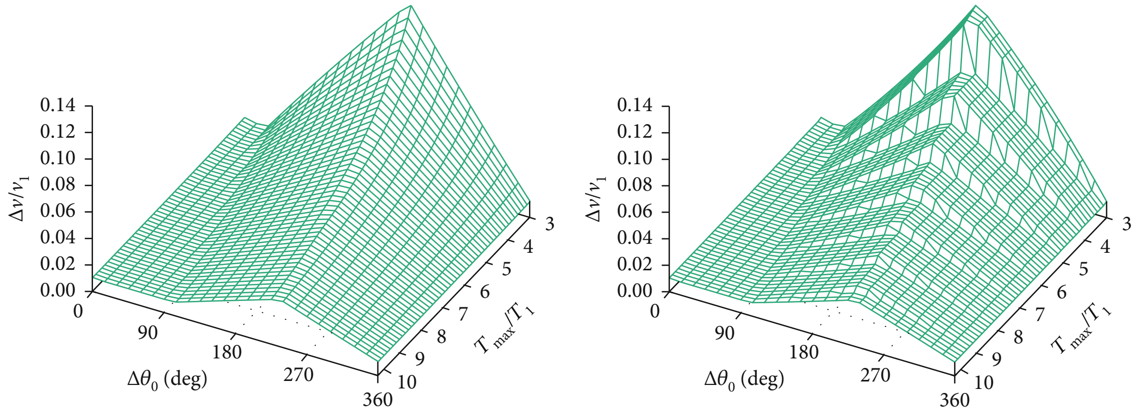
In the presence of a large population, the evaluation of the diversity score of each individual might be too expensive (as it goes with $O(n^2)$). Hence, the full distance d_i is substituted by an approximation $\tilde{d}_i$ evaluated over a number of n_{dc} individuals randomly selected in the population, that is,

$$\tilde{d}_i = \sum_{k=1}^{n_{\text{dc}}} d_{i,r_k}, \tag{31}$$

where r_k is a random integer in $[1, n_{\text{P}}] \setminus \{i\}$. In this work, the following values of the abovementioned hyperparameters showed the best overall results: $n_{\text{dc}} = 100$ and $\delta_{\text{dc}} = 1$.

Table 2: Initial orbital parameters of chaser and targets.

	Chaser	Targets									
ID	0	1	2	3	4	5	6	7	8	9	10
r (km)	7000	6900	6910	6930	6940	6950	6960	6980	7010	7020	7030
θ_0 (deg)	0	-5	10	15	35	-30	-10	25	20	-25	-15
ID		11	12	13	14	15	16	17	18	19	20
r (km)		7050	7060	7070	7080	7090	7110	7130	7140	7160	7170
θ_0 (deg)		5	-35	30	-20	-40	50	-50	40	-45	45



(a) 4-impulse heuristic

(b) 3-impulse heuristic

Figure 6: Cost of the rendezvous with $r_2/r_1 = 1.02$, as a function of initial phase angle $\Delta\theta_0 \in [0, 2\pi]$ and transfer time $T_{\max} \in [3T_1, 10T_1]$. ΔV is estimated using either the 4-impulse (a) or 3-impulse heuristic (b).

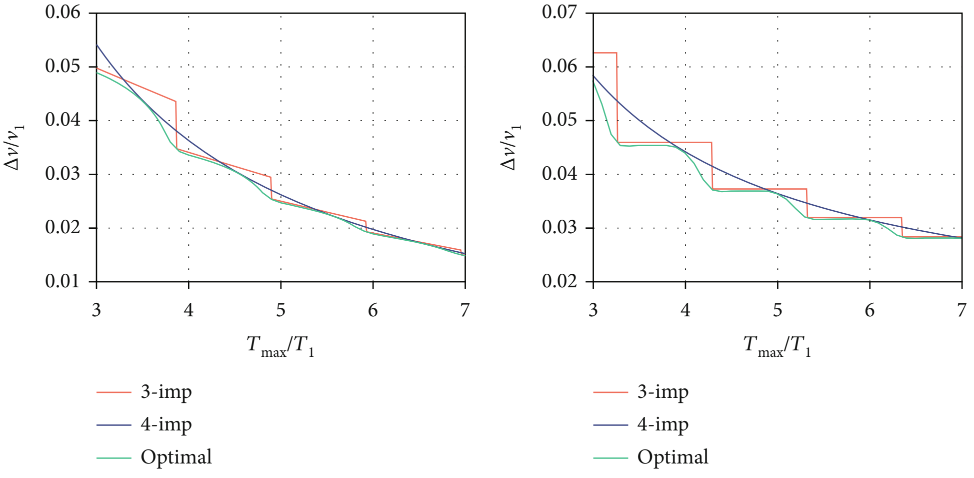
4.1. GA Hyperparameters. The performance of a GA clearly depends not only on the choice of the selection, mutation, and crossover operators but also on the value of several hyperparameters, which drive the different phases of the evolution process. In addition to the population size n_P and the maximum number of generations n_G that are common to all population-based metaheuristics, typical GA hyperparameters are the crossover probability p_c, which represents the probability that the parents are replaced by the generated offspring at the end of the crossover phase; the mutation probability p_m, which is the probability that a (child) individual undergoes a random mutation during the mutation phase; and other operator-specific parameters, such as the depth of the reverse mutation operator. A proper tuning of these hyperparameters may greatly improve the effectiveness of a GA. However, the optimal tuning of a GA is documented to be problem-dependent and represents an NP-hard problem by itself. Therefore, a preliminary trial-and-error tuning is usually done on a simplified, possibly scaled, version of the optimization problem of interest, with the hope of capturing all the relevant features of the original problem. The results of the hyperparameter tuning performed in this study are reported in Section 5.2.

5. Numerical Results

In this paper, numerical results are presented for different ADR missions aimed at removing all the debris composing a predetermined cluster with up to 20 objects orbiting in close coplanar LEOs. ID and initial orbital parameters assumed for both the chaser and the targets composing the cluster are listed in Table 2. When a $N \times D$ tour is analyzed, only the first N debris of the cluster are considered, and the total mission time is set to $T_M = 7NT_0$, where T_0 is the initial orbital period of the chaser.

5.1. Analysis of the Cost Heuristic. The effectiveness of the suboptimal solution proposed as heuristic in Section 3.1 has been verified by comparison with the solution obtained by EOS in meaningful test cases. In particular, the rendezvous with a target moving on a circular orbit of radius $r_2/r_1 = 1.02$ is analyzed in this section.

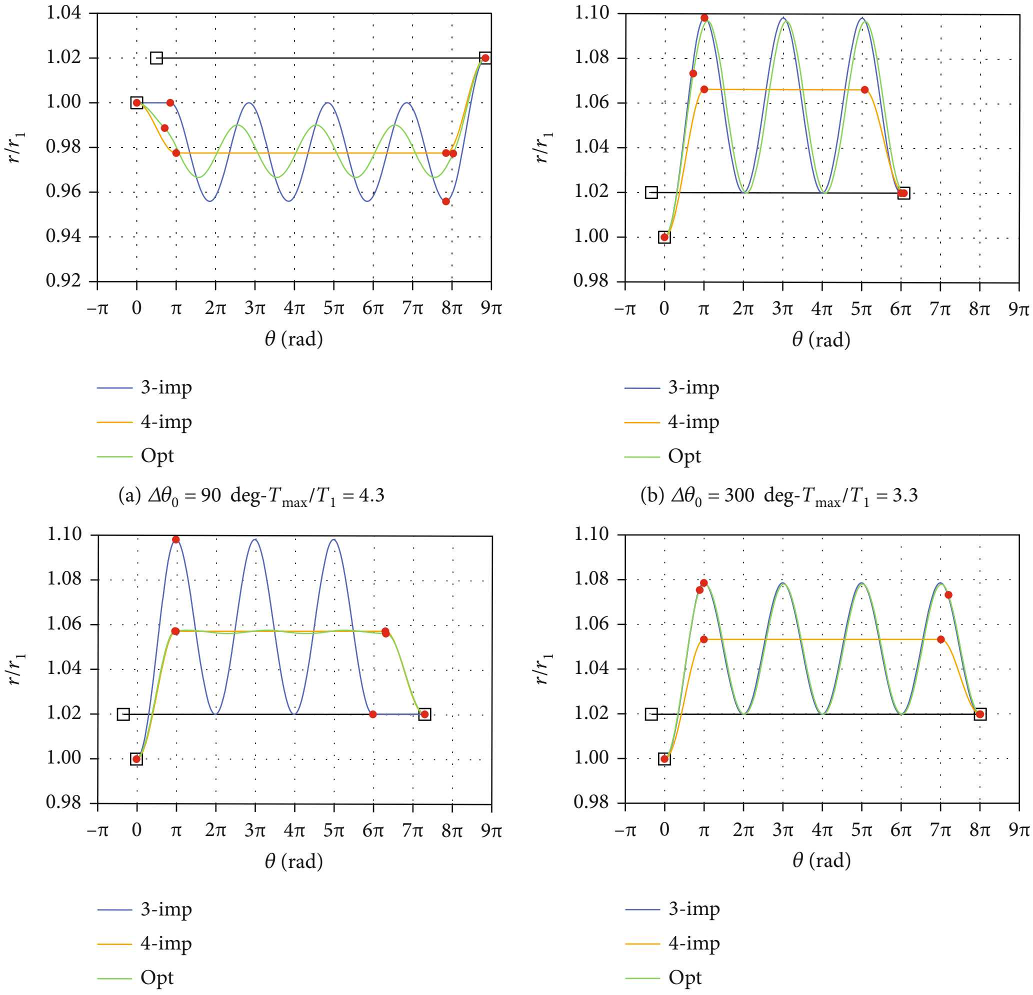
Figure 6(a) shows the ΔV estimated by the 4-impulse heuristic as a function of initial phase angle $\Delta\theta_0 \in [0, 2\pi]$ and available transfer time $T_{\max} \in [3T_1, 10T_1]$. One should note that the initial phase angle is a function of time, changing at constant but very slow rate for close orbits. As a matter of fact, this figure presents the cost of the rendezvous maneuver as a function of initial time and transfer time length. Depending on the initial time, ΔV may be much greater than the minimum value corresponding to the time-free Hohmann transfer. For the sake of comparison, Figure 6(b) also shows the ΔV provided by the 3-impulse heuristic, which features abrupt changes (i.e., a sudden reduction) when the greater time-length permits an additional revolution around the Earth.



(a) $\Delta\theta_0 = 90$

(b) $\Delta\theta_0 = 300$

Figure 7: Cost of the rendezvous with $r_2/r_1 = 1.02$, as a function of the available transfer time $T_{max} \in [3T_1, 7T_1]$. The transfer is either internal (a) or external (b).



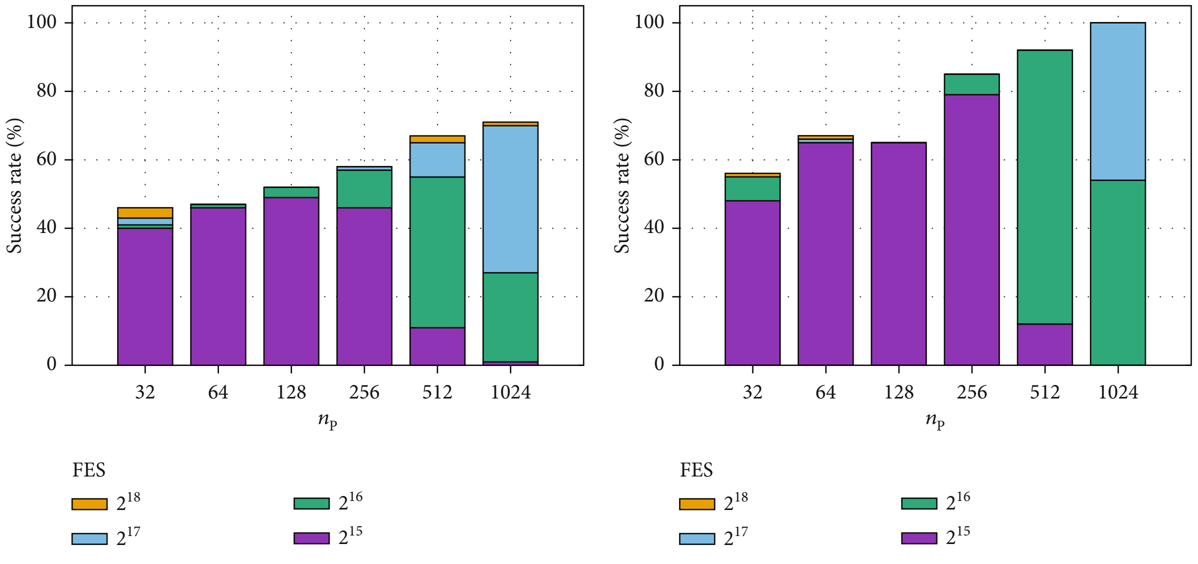
(a) $\Delta\theta_0 = 90$ deg-$T_{max}/T_1 = 4.3$

(b) $\Delta\theta_0 = 300$ deg-$T_{max}/T_1 = 3.3$

(c) $\Delta\theta_0 = 300$ deg-$T_{max}/T_1 = 3.94$

(d) $\Delta\theta_0 = 300$ deg-$T_{max}/T_1 = 4.3$

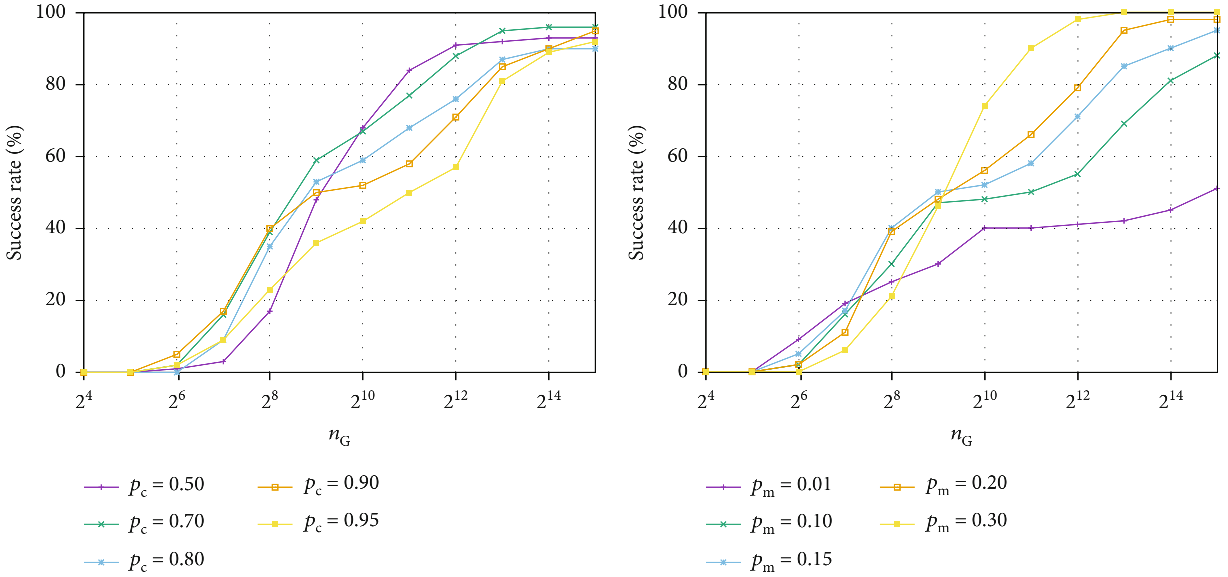
Figure 8: Optimal and suboptimal chaser trajectories. Black lines show the constant-radius target trajectory. Empty squares mark initial and final positions. Red dots indicate impulsive maneuvers.



(a) 8 × 2 mission

(b) 8 × 3 mission

Figure 9: GA success rate as a function of the population size n_P, for assigned numbers of function evaluations FES.

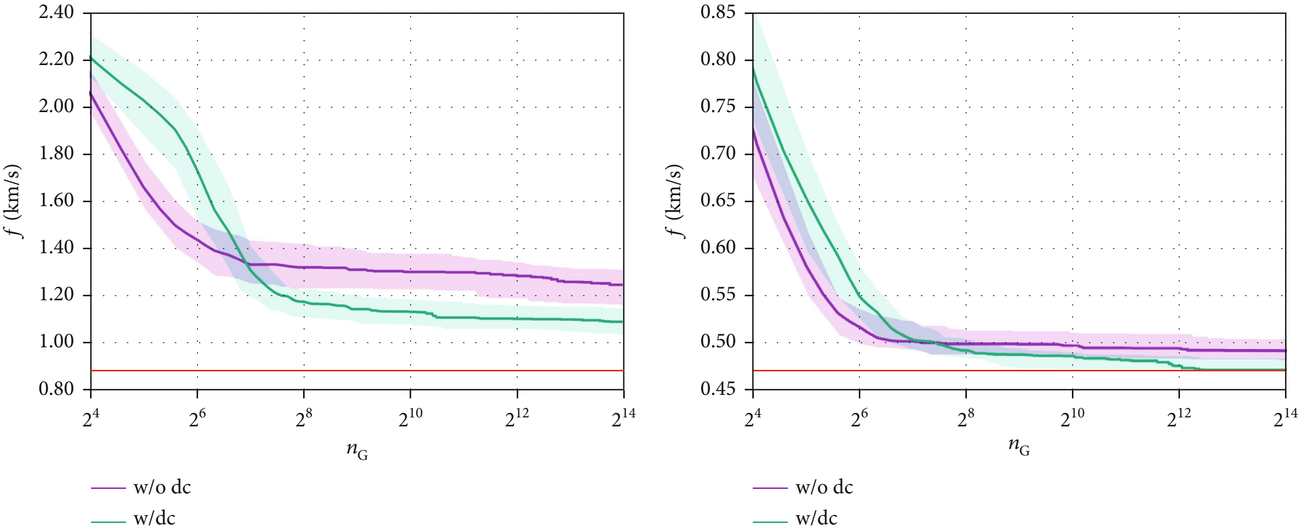


(a) Effect of the crossover probability, for $p_m = 0.15$

(b) Effect of the mutation probability, for $p_c = 0.90$

Figure 10: GA success rate along the evolution process for the 12-target mission, with different crossover p_c or mutation p_m probabilities. The population size is $n_P = 64$.

The cost in this class of mission is ruled more by the propellant needed to recover an incorrect initial phasing than by the basic ΔV required by the passage to a different radius. The outer-level combinatorial problem looks for and privileges target sequences that imply transfers starting with a favourable phase angle, i.e., in a narrow range close to values permitting a Hohmann transfer. Therefore, the initial time of each rendezvous is dictated by the solution of the outer-level problem. When the initial time (i.e., the initial phasing) does not permit a Hohmann transfer, the required ΔV is significantly dependent on the time which is available to complete the maneuver. An analysis of the rendezvous cost is better carried out by investigating the influence of the available time for significant assigned values of the initial phase angle. Figure 7 compares, from this perspective, the performance estimated by the two heuristics with the optimal transfer (as optimized by EOS). Even though the 3-impulse heuristic is closer to the features of the optimal solution, the 4-impulse heuristic provides a sufficient guess at the rendezvous cost. On the other hand, the trend of the 4-impulse heuristic solution is quite regular, without the abrupt changes that make the 3-impulse heuristic less apt to use when the rendezvous epochs are discretized over a coarse time grid. It is worthwhile to remark the quality of the EOS optimal solution:



(a) 20 × 1 mission

(b) 10 × 3 mission

Figure 11: Median fitness and interquartile range along the evolution process with and without the diversity control. The population size is $n_P = 64$. The optimal fitness f^* (red line) is reported for comparison.

performance is never lower than the ones estimated by heuristics. In particular, the same performance is numerically achieved by EOS when either heuristic corresponds to the theoretically best strategy.

Figure 8 presents the chaser trajectory in a few significant cases. The circular orbit of the target is shown as a solid black line, starting from its initial position. The 4-impulse heuristic always uses the entire available time. On the contrary, the angular length of the 3-impulse heuristic is constrained to an integer number of revolutions, and the maneuver is completed by a coast arc, which is either initial or final. An internal maneuver is presented in Figure 8(a). The optimal solution does not present the constant-radius arc: nevertheless, by means of four impulses, it is able to make use of all the available time, thus improving performance. The features of the optimal maneuver are better highlighted by the three cases of increasing time length that are illustrated in Figures 8(b)–8(d) for an external maneuver ($\Delta\theta_0 = 300\,\text{deg}$). The 3-impulse heuristic solution for $T_{\max}/T_1 = 3.3$ has a very short final coasting and is very close to the optimal solution that, in this case, has also three impulses, but its second impulse is displaced from the apogee to allow using the entire angle range with a lower apogee. For greater values of $T_{\max}$, the optimal rendezvous requires four impulses and gradually approaches the corresponding heuristic solution (Figure 8(c)) that, for just one value of available time, is itself optimal. The optimal maneuver hereafter moves away from the 4-impulse heuristic resembling an impracticable 3-impulse heuristic solution with one more lap. The latter solution can only occur when the available time permits the replacement of the long coast arc with an additional revolution and is, just at that moment, optimal (Figure 8(d)). These considerations explain the trend of the optimal solution in Figure 7 that periodically and alternately coincides with the 3- and 4-impulse heuristic solutions.

Table 3: Attained solutions at successive stages of the optimization process. GA refers to the solutions of the outer-level problem; EOS time-fixed and EOS time-free refer to the refined solutions at the end of the inner-level optimization, by assuming the rendezvous epochs fixed or free to be optimized, respectively.

(a) 15-target missions

Mission	ΔV_{tot} (m/s)		
	GA	EOS time-fixed	EOS time-free
15 × 1	801.61	785.95	641.91
15 × 2	661.46	654.93	637.51
15 × 3	633.78	628.48	617.82
15 × 4	632.57	630.23	616.85

(b) 20-target missions

Mission	ΔV_{tot} (m/s)		
	GA	EOS time-fixed	EOS time-free
20 × 1	881.50	868.83	759.21
20 × 2	789.82	781.76	745.26
20 × 3	771.48	765.06	742.59
20 × 4	758.33	753.91	733.51

5.2. Tuning of the GA Hyperparameters. A preliminary tuning of the GA hyperparameters has been carried out by analyzing several instances of either the uniform or discrete time-length tour, keeping the problem size sufficiently small. This allows for an exhaustive search to be completed in a reasonable amount of time; thus, the optimal solution of the considered problems can be easily retrieved and used for comparison with the results of the GA. In order to mitigate the effects due to the intrinsic stochastic nature of the algorithm, 100 independent runs

Table 4: Solutions of the 15-target problem after EOS time-free optimization.

15 × 1			15 × 2			15 × 3			15 × 4			
p_k	t_k (h)	ΔV_k (m/s)	p_k	t_k (h)	ΔV_k (m/s)	p_k	t_k (h)	ΔV_k (m/s)	p_k	t_k (h)	ΔV_k (m/s)	$\Delta V_k/\Delta V_H$
6	5.67	21.65	6	5.61	21.65	11	2.61	26.81	11	8.11	26.81	1.00
7	17.75	47.22	7	18.37	42.04	10	21.71	10.69	10	21.70	10.69	2.44
2	36.86	93.19	2	40.39	87.75	14	44.90	83.07	14	45.39	81.98	2.11
1	47.79	12.54	1	51.63	15.66	4	47.01	75.30	4	46.78	75.30	1.00
10	56.66	70.60	10	56.20	70.60	3	51.88	19.76	3	51.61	19.31	4.10
9	77.24	25.51	9	76.01	24.83	2	76.15	54.56	2	75.76	54.49	6.13
4	79.33	43.31	4	79.33	43.31	8	79.33	54.37	8	78.25	54.37	1.00
3	95.22	32.07	3	93.50	35.84	15	90.61	42.66	15	94.16	42.66	1.00
14	96.49	80.77	14	96.86	80.77	9	107.97	37.29	9	108.54	37.29	1.00
8	107.07	37.37	8	105.13	37.37	6	120.68	32.41	6	120.42	33.68	1.31
12	130.04	26.75	12	130.04	26.75	1	123.31	32.83	1	121.49	32.83	1.00
13	136.48	11.25	13	136.48	11.24	5	136.25	31.72	5	136.25	31.34	1.19
5	151.89	64.54	5	151.57	64.54	12	138.02	59.23	12	137.59	59.23	1.00
11	159.91	53.90	11	164.71	53.90	13	143.41	5.32	13	140.75	5.32	1.00
15	170.00	21.24	15	170.00	21.24	7	170.00	51.80	7	170.00	51.55	1.07
		641.91			637.49			617.82			616.85	1.26

were carried out for each set of hyperparameters under investigation.

The first analysis concerned the influence of the population size n_P and the number of generations n_G on the problem solution. The aim is to identify, if possible, a putative optimal allocation of trials, that is, a pair n_P and n_G that achieves the best performance for a known, limited, amount of function evaluations $\text{FES} = n_G \cdot n_P$.

Figure 9 presents the success rate (percentage of runs that achieve the optimal solution) as a function of the population size n_P, for a few threshold values of FES, for the 8 × 2 (Figure 9(a)) and 8 × 3 (Figure 9(b)) missions. Here, the crossover and mutation probability are set as $p_c = 0.9$ and $p_m = 0.15$, respectively, and the diversity control is active with its default values. The success rate obviously increases when both the population size and the number of generations increase. However, when the number of available FES is limited, the results are not so easy to interpret. An even split between number of individuals and iterations, that is, $n_P = n_G \approx \sqrt{\text{FES}}$, seems close to the optimal configuration. It is also worthwhile noting that, for a given number of FES, the success rate is higher in the 8 × 3 mission than in the 8 × 2 mission, even though the total number of variables is larger (24 vs. 16), and reaches 100% with a population of 1024 individuals. The reason is probably related to the fact the the fitness landscape is smoother with a higher D, and this favours the chosen genetic operators.

The effects of crossover and mutation probability on the success rate have been also investigated. Figure 10 reports the success rate for different values of p_c (Figure 10(a)) and p_m (Figure 10(b)) against the number of generations n_G. A 12-target mission is considered for this test, using a population of $n_P = 64$ individuals; the diversity control is active with default values. According to Figure 10(a), the performance of GA seems to improve faster with n_G when small crossover probabilities are used ($p_c = 0.5$ and $p_c = 0.7$), but, at the same time, tends to stall earlier, thus showing an apparent premature-convergence behavior. Conversely, higher crossover probabilities ($p_c = 0.9$ and $p_c = 0.95$) lead to slower, yet constant, improvements. Thus, with $n_G = 2^{15}$, the best results in terms of success rate are obtained with both $p_c = 0.7$ (96%) and $p_c = 0.9$ (95%). Figure 10(b) shows that when the number of generations is less than or equal to 500, the success rate is almost independent on p_m. Only when a very high number of iterations are carried out, the effect of increasing the mutation probability becomes apparent. This is an expected behavior, as mutation is primarily devoted to avoiding stagnation over long time horizons.

Finally, Figure 11 shows the effect of the diversity control (dc) on the GA performance in both a uniform and a discrete time-length tour. In these plots, the solid-line curves represent the median fitness along the evolution process, while the shaded regions around them represent the interquartile range. It is possible to note that, in all reported cases, the use of diversity control leads to an improved solution, preventing GA from a premature stagnation. The beneficial effect of diversity control is more evident in problem instances with a larger number of targets N (Figure 11(a)), since, on average, in these cases, the GA is more likely to get trapped in a local minimum.

5.3. Problem Solutions. For the sake of brevity, only the numerical results concerning two study cases, that is, the 15 - and 20-target missions, are here presented. For each of these scenarios, several independent optimizations have been carried out using the GA code, and the best solution found was elected as putative optimum of the outer-level problem. Then, this solution was given as input to the inner-level optimization algorithm, with the goal of approaching the

Table 5: Solutions of the 20-target problem after EOS time-free optimization.

	20 × 1			20 × 2			20 × 3			20 × 4		
p_k	t_k (h)	ΔV_k (m/s)	p_k	t_k (h)	ΔV_k (m/s)	p_k	t_k (h)	ΔV_k (m/s)	p_k	t_k (h)	ΔV_k (m/s)	$\Delta V_k/\Delta V_{\mathrm{H}}$
1	6.83	54.48	1	3.80	54.48	1	3.72	54.48	1	3.64	54.48	1.00
2	16.63	21.81	2	16.63	17.71	2	16.62	17.73	2	16.63	17.74	3.06
4	39.33	16.45	4	28.91	16.43	4	32.01	16.45	4	28.33	16.43	1.03
3	46.53	5.43	3	46.35	5.47	3	43.08	5.43	3	49.34	5.47	1.00
12	51.64	70.18	12	50.68	70.15	12	50.71	70.18	12	50.37	70.15	1.00
13	68.00	52.22	13	73.67	36.10	13	75.99	33.28	13	75.14	33.62	7.06
6	77.12	59.09	6	77.12	59.10	6	78.33	59.09	6	77.24	59.10	1.00
5	91.58	32.75	5	89.85	36.38	5	91.13	37.28	5	91.45	34.06	6.73
9	99.31	37.88	9	103.40	72.38	9	105.07	61.80	9	105.07	62.39	1.83
16	107.39	47.84	16	105.53	47.84	15	109.55	37.28	15	107.67	37.29	1.00
10	116.06	42.48	15	118.93	23.33	16	121.42	32.22	16	121.14	26.60	2.54
7	136.00	26.94	10	124.67	31.93	7	130.72	69.42	7	132.21	69.40	1.01
14	146.35	53.58	7	139.25	26.92	10	135.74	26.94	10	134.96	26.92	1.00
18	153.74	31.62	14	143.80	53.56	14	152.24	27.01	14	152.34	26.73	1.33
17	175.67	16.15	18	153.23	31.59	18	154.09	31.62	18	154.08	31.59	1.00
11	180.62	42.33	17	178.50	13.57	17	178.69	14.04	17	179.70	13.22	2.84
8	194.57	21.43	11	181.29	42.30	11	180.54	42.33	11	182.49	42.30	1.00
20	199.47	84.59	8	194.39	21.42	8	187.00	21.43	8	183.40	21.42	1.00
19	220.90	5.21	19	199.35	79.40	19	199.00	79.38	19	199.24	79.40	1.00
15	226.66	36.75	20	226.66	5.20	20	226.66	5.20	20	226.66	5.20	1.11
		759.21			745.26			742.59			733.51	1.21

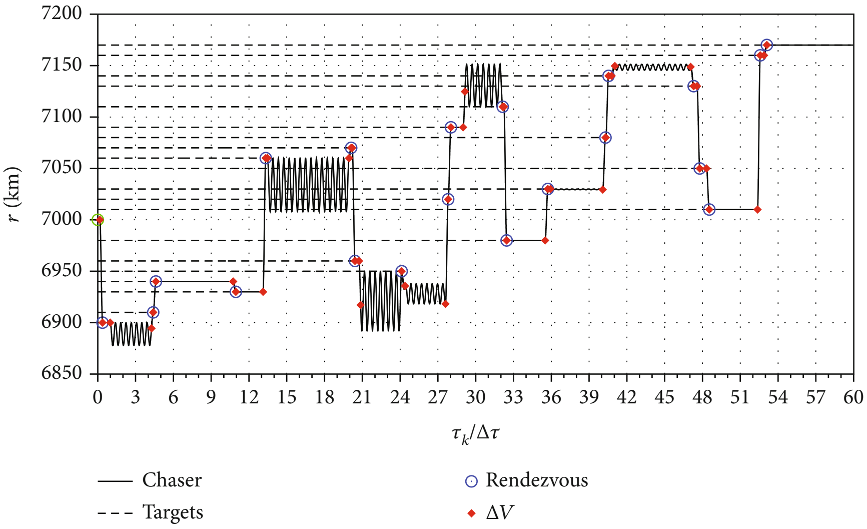


Figure 12: Chaser radius r as a function of nondimensional time $\tau_k/\Delta\tau$ for the 20 × 3 mission.

unknown global optimum of the corresponding MINLP problem.

Table 3 presents the best solutions, in terms of ΔV_{tot}, obtained by GA at the end of the outer-level optimization and by EOS after the inner-level optimization, with either fixed or free rendezvous times. For either mission scenario (i.e., for assigned values of number of targets and total transfer time), ΔV_{tot} decreases as the time-division factor D increases, whereas the target sequence remains essentially unaltered, with just a few exceptions. The maneuvers of the GA solutions reflect the suboptimal 4-impulse heuristic and can be improved by EOS. From Table 4, it is possible to note that the differences between GA solutions and corresponding EOS time-fixed solutions are always in the order of few percent, which is the expected error of the devised heuristic, thus confirming its effectiveness and accuracy.

Starting from the GA solution with $D = 1$, any further improvement is due to the allocation of more time to the legs that cannot be executed according to a Hohmann transfer. This allocation can be achieved either by increasing D or by using EOS in its time-free version. It is a curious coincidence that in both scenarios, the GA solution ($D = 4$) and the EOS time-free solution ($D = 1$) show approximately the same ΔV_{tot}. The rendezvous maneuvers of the best GA solution are suboptimal, and the target sequence is slightly different from that of the uniform time-length tour. The application of EOS to this new sequence provides a further reduction of the required propellant.

The solutions for the 15- and 20- target scenarios, obtained after the time-free optimization, are reported in Tables 4 and 5, for each time-division factor D. Several details (i.e., optimal target sequence and, for each rendezvous, arrival time and required ΔV) are presented. The last column, referring to the $N \times 4$ solution, compares the ΔV of each rendezvous with the cost ΔV_{H} of the optimal time-free transfer. The best trajectories are composed, for the greater part, of Hohmann transfers; 3- and 4-impulse legs are anyway flown by the chaser when the target is unfavourably phased during the available time window. This result is evident in Figure 12, which shows the radial position of the chaser during the whole 20×3 mission. The Hohmann transfer is usually exploited to move between distant orbits. In these cases, the phase angle changes more rapidly, and it is easier to find a favourable rendezvous opportunity.

The overall improvement with respect to the first GA solution, which assumes uniform time-lengths, is significant (close to 20%). The overall cost ΔV_{tot} of the best missions (see the last row in Tables 4 and 5) is $20 - 25\%$ greater than the value of the time-free trajectory having the same target sequence. These evidences let one argue that this two-step procedure is able to attain a solution very close to that of the original MINLP problem.

6. Conclusion

A two-level optimization procedure for the design of planar multirendezvous trajectories has been devised and tested. The goal is to minimize the overall propellant consumption, while completing, within a prescribed amount of time, the tour all the bodies in a prescribed set. Satisfactory numerical solutions have been obtained for an active debris removal mission comprising up to 20 target bodies orbiting in coplanar LEOs.

The first, outer-level step concerns the definition of the optimal target sequence together with a preliminary evaluation of the rendezvous epochs. This step results in a permutation-optimization problem. Two different formulations, of increasing complexity, have been used, by supposing uniform time-lengths of the transfer legs or by selecting the rendezvous epochs in a discrete time grid. The growth in the problem complexity is rewarded by the attainment of a solution closer to the global optimum of the complete mixed-integer nonlinear programming problem. A genetic algorithm, with a peculiar permutation encoding and effective permutation-preserving genetic operators, is able to solve the combinatorial optimization problem. A simple, but fairly accurate, analytic solution of the single-target rendezvous maneuver is adopted as heuristic for a fast evaluation of the ΔV associated to each leg, without the need for an accurate optimization.

The solution attained at the end of the outer-level optimization procedure is further refined in the inner-level optimization step, by assuming the body sequence fixed and by optimizing the multi-impulse rendezvous maneuvers. Two formulations of increasing complexity are proposed even in this case. The first one maintains the rendezvous epochs found in the first step and only replaces each suboptimal trajectory described by the 4-impulse heuristic with an improved maneuver. More precisely, each body-to-body transfer is described by means of a peculiar parameterization based on the position of the impulses: the total velocity change is minimized. The second formulation keeps the target sequence but can modify the rendezvous epochs to achieve a more appropriate sizing of the time allocated for each leg. The contemporary optimization of the entire trajectory is necessary.

Numerical results provided by this two-level procedure make the authors confident about the attainment of a trajectory which is very close to the solution of the full mixed-integer nonlinear programming problem. The overall computational effort is low enough to be compatible with desktop personal computers.

The two levels of the optimization procedure appeared of equal importance in the investigated scenarios. Instead, if the number of bodies and the average time available for a single rendezvous are large, the correct target phasing occurs frequently, and a solution based on many Hohmann transfers and very few 4-impulse low-ΔV maneuvers can be soon achieved by the outer-lever optimization. The inner-level step becomes more important in problems with few targets and narrow time windows.

Data Availability

The data that support the findings of this study are contained in the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] J. C. Liou, A. K. Anilkumar, B. Bastida et al., "Stability of the future leo environment–an IADC comparison study," in *Proceedings of the 6th European Conference on Space Debris*, vol. 723, Darmstadt, Germany, 2013.

[2] J. N. Pelton, "Current space debris remediation and on-orbit servicing initiatives," in *New Solutions for the Space Debris Problem*, pp. 11–29, Springer, 2015.

[3] E. Henderson and M. Holderman, "Technology applications that support space exploration," in *47th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, San Diego, California, 2011.

[4] B. Vitali, A. Lüpken, S. Flegel et al., "Active debris removal of multiple priority targets," *Advances in Space Research*, vol. 51, no. 9, pp. 1638–1648, 2013.

[5] D. Madakat, J. Morio, and D. Vanderpooten, "Biobjective planning of an active debris removal mission," *Acta Astronautica*, vol. 84, pp. 182–188, 2013.

[6] H.-X. Shen, T.-J. Zhang, L. Casalino, and D. Pastrone, "Optimization of active debris removal missions with multiple targets," *Journal of Spacecraft and Rockets*, vol. 55, no. 1, pp. 181–189, 2018.

[7] T.-J. Zhang, H.-X. Shen, H. Li, and J. Li, "Ant colony optimization based design of multiple-target active debris removal mission," in *2018 SpaceOps Conference*, Marseille, France, 2018.

[8] J. Murakami and S. Hokamoto, "Approach for optimal multi-rendezvous trajectory design for active debris removal," *61st International Astronautical Congress (IAC) 2010*, pp. 6013–6018, 2010.

[9] J. Zhao, W. Feng, and J. Yuan, "A novel two-level optimization strategy for multi-debris active removal mission in LEO," *Computer Modeling in Engineering & Sciences*, vol. 122, no. 1, pp. 149–174, 2020.

[10] Y. Liu, J. Yang, Y. Wang, Q. Pan, and J. Yuan, "Multi-objective optimal preliminary planning of multi-debris active removal mission in LEO," *SCIENCE CHINA Information Sciences*, vol. 60, article 072202, 2017.

[11] A. Barea, H. Urrutxua, and L. Cadarso, "Large-scale object selection and trajectory planning for multitarget space debris removal missions," *Acta Astronautica*, vol. 170, pp. 289–301, 2020.

[12] K. T. Alfriend, D.-J. Lee, and N. G. Creamer, "Optimal servicing of geosynchronous satellites," *Journal of Guidance, Control, and Dynamics*, vol. 29, no. 1, pp. 203–206, 2006.

[13] K. Daneshjou, A. A. Mohammadi-Dehabadi, and M. Bakhtiari, "Mission planning for on-orbit servicing through multiple servicing satellites: a new approach," *Advances in Space Research*, vol. 60, no. 6, pp. 1148–1162, 2017.

[14] X.-q. Chen and J. Yu, "Optimal mission planning of geo on-orbit refueling in mixed strategy," *Acta Astronautica*, vol. 133, pp. 63–72, 2017.

[15] Z. Yang, Y. Yan, H. Xu, and L. Kong, "Mission planning optimization for multiple geosynchronous satellites refueling," *Advances in Space Research*, vol. 56, no. 11, pp. 2612–2625, 2015.

[16] J. Zhang, G. T. Parks, Y.-z. Luo, and G.-j. Tang, "Multispacecraft refueling optimization considering the J 2 perturbation and window constraints," *Journal of Guidance, Control, and Dynamics*, vol. 37, no. 1, pp. 111–122, 2014.

[17] C. Han, S. Zhang, and X. Wang, "On-orbit servicing of geosynchronous satellites based on low-thrust transfers considering perturbations," *Acta Astronautica*, vol. 159, pp. 658–675, 2019.

[18] D. Izzo and M. Martens, "The Kessler run: on the design of the GTOC9 challenge," *Acta Futura*, vol. 11, pp. 11–24, 2018.

[19] L. Casalino, D. Pastrone, F. Simeoni, G. Colasurdo, and A. Zavoli, "GTOC5: results from the Politecnico di Torino and Universitá di Roma Sapienza," *Acta Futura*, vol. 8, pp. 29–36, 2014.

[20] H.-X. Shen and L. Casalino, "Simple ΔV approximation for optimization of debris-to-debris transfers," *Journal of Spacecraft and Rockets*, vol. 58, no. 2, pp. 575–580, 2021.

[21] B. Pan, P. Lu, and C. Zheng, "Coast arcs in optimal multiburn orbital transfers," *Journal of Guidance, Control, and Dynamics*, vol. 35, no. 2, pp. 451–461, 2012.

[22] J. E. Prussing and J.-H. Chiu, "Optimal multiple-impulse time-fixed rendezvous between circular orbits," *Journal of Guidance, Control, and Dynamics*, vol. 9, no. 1, pp. 17–22, 1986.

[23] G. Colasurdo and D. Pastrone, "Indirect optimization method for impulsive transfers," in *Astrodynamics Conference, 1994*, pp. 441–448, Scottsdale, Arizona, 1994.

[24] J. E. Prussing, *Optimal multiple-impulse orbital rendezvous*, PhD thesis, Massachusetts Institute of Technology, 1967.

[25] F. Jiang and T. Gao, "Systematic low-thrust trajectory optimization for a multi-rendezvous mission using adjoint scaling," *Astrophysics and Space Science*, vol. 361, no. 4, p. 117, 2016.

[26] E. L. Lawler, "The traveling salesman problem: a guided tour of combinatorial optimization," *Wiley-Interscience Series in Discrete Mathematics*, 1985.

[27] D. L. Applegate, R. E. Bixby, V. Chvátal et al., "Certification of an optimal TSP tour through 85,900 cities," *Operations Research Letters*, vol. 37, no. 1, pp. 11–15, 2009.

[28] F. Zuiani and M. Vasile, "Preliminary design of debris removal missions by means of simplified models for low-thrust, many-revolution transfers," *International Journal of Aerospace Engineering*, vol. 2012, 22 pages, 2012.

[29] M. Cerf, "Multiple space debris collecting mission-debris selection and trajectory optimization," *Journal of Optimization Theory and Applications*, vol. 156, no. 3, pp. 761–796, 2013.

[30] J. K. Lenstra and A. H. G. R. Kan, "Complexity of vehicle routing and scheduling problems," *Networks*, vol. 11, no. 2, pp. 221–227, 1981.

[31] B. W. Barbee, S. Alfano, E. Pinon, K. Gold, and D. Gaylor, "Design of spacecraft missions to remove multiple orbital debris objects," in *2011 Aerospace Conference*, pp. 1–14, BigSky, Montana, 2011.

[32] D. Izzo, I. Getzner, D. Hennes, and L. F. Simões, "Evolving solutions to tsp variants for active space debris removal," in *Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation*, pp. 1207–1214, Madrid, Spain, 2015.

[33] J. Stuart, K. Howell, and R. Wilson, "Application of multi-agent coordination methods to the design of space debris mitigation tours," *Advances in Space Research*, vol. 57, no. 8, pp. 1680–1697, 2016.

[34] L. Federici, A. Zavoli, and G. Colasurdo, "On the use of a∗ search for active debris removal mission planning," *71st International Astronautical Congress (IAC) - The CyberSpace Edition*, 2020.

[35] L. F. Simões, D. Izzo, E. Haasdijk, and A. E. Eiben, "Multi-rendezvous spacecraft trajectory optimization with beam P-ACO," in *European Conference on Evolutionary Computation in Combinatorial Optimization*, pp. 141–156, Amsterdam, The Netherlands, 2017.

[36] Ö. Gürtuna and J. Trépanier, "On-orbit satellite servicing: a space-based vehicle routing problem," in *Operations Research in Space and Air*, pp. 123–141, Springer, 2003.

[37] M. Cerf, "Multiple space debris collecting mission: optimal mission planning," *Journal of Optimization Theory and Applications*, vol. 167, no. 1, pp. 195–218, 2015.

[38] L. Federici, A. Zavoli, and G. Colasurdo, "A time-dependent tsp formulation for the design of an active debris removal mission using simulated annealing," *Advances in the Astronautical Sciences*, vol. 171, pp. 1349–1368, 2019.

[39] G. He and R. G. Melton, "Multiple small-satellite salvage mission sequence planning for debris mitigation," *Advances in the Astronautical Sciences*, vol. 171, pp. 4229–4244, 2019.

[40] A. Zavoli, L. Federici, B. Benedikter, and G. Colasurdo, "Comparative analysis of genetic crossover operators for the optimization of impulsive multi-rendezvous trajectories," in *AIDAA 2019 - XXV International Congress*, Rome, Italy, 2019.

[41] F. Cacciatore and C. Toglia, "Optimization of orbital trajectories using genetic algorithms," *Journal of Aerospace Engineering, Sciences and Applications*, vol. 1, no. 1, pp. 58–69, 2008.

[42] D. Izzo, "Revisiting Lambert's problem," *Celestial Mechanics and Dynamical Astronomy*, vol. 121, no. 1, pp. 1–15, 2015.

[43] P. Belotti, C. Kirches, S. Leyffer, J. Linderoth, J. Luedtke, and A. Mahajan, "Mixed-integer nonlinear optimization," *Acta Numerica*, vol. 22, p. 1, 2013.

[44] A. Zavoli and G. Colasurdo, "Indirect optimization of finite-thrust cooperative rendezvous," *Journal of Guidance, Control, and Dynamics*, vol. 38, no. 2, pp. 304–314, 2015.

[45] D. Izzo, D. Hennes, L. F. Simões, and M. Märtens, "Designing complex interplanetary trajectories for the global trajectory optimization competitions," in *Space Engineering: Modeling and Optimization with Case Studies*, G. Fasano and J. D. Pinter, Eds., vol. 114, pp. 151–176, Springer, 2016.

[46] J.-C. Picard and M. Queyranne, "The time-dependent traveling salesman problem and its application to the tardiness problem in one-machine scheduling," *Operations Research*, vol. 26, no. 1, pp. 86–110, 1978.

[47] K. R. Fox, B. Gavish, and S. C. Graves, "An n-constraint formulation of the (time-dependent) traveling salesman problem," *Operations Research*, vol. 28, no. 4, pp. 1018–1021, 1980.

[48] L. Federici, B. Benedikter, and A. Zavoli, "EOS: a parallel, self-adaptive, multi-population evolutionary algorithm for constrained global optimization," *2020 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1–10, 2020.

[49] G. Colasurdo, A. Zavoli, A. Longo, L. Casalino, and F. Simeoni, "Tour of Jupiter Galilean moons: winning solution of GTOC6," *Acta Astronautica*, vol. 102, pp. 190–199, 2014.

[50] L. Casalino, G. Colasurdo, A. Zavoli, and M. Berga, "GTOC8: results and methods of team 22," *Advances in the Astronautical Sciences*, vol. 158, pp. 4291–4300, 2016.

[51] L. Federici, A. Zavoli, and G. Colasurdo, "Preliminary capture trajectory design for Europa Tomography Probe," *International Journal of Aerospace Engineering*, vol. 2018, 12 pages, 2018.

[52] L. Federici, A. Zavoli, G. Colasurdo, L. Mancini, and A. Neri, "Integrated optimization of ascent trajectory and SRM design of multistage launch vehicles," *Advances in the Astronautical Sciences*, vol. 168, pp. 733–752, 2019.

[53] J. H. Holland, *Adaptation in Natural and Artificial Systems*, University of Michigan Press, 1975.

[54] M. Mitchell, *An Introduction to Genetic Algorithms*, MIT press, 1998.

[55] J.-Y. Potvin, "Genetic algorithms for the traveling salesman problem," *Annals of Operations Research*, vol. 63, no. 3, pp. 337–370, 1996.

[56] D. E. Goldberg and R. Lingle, "Alleles, loci, and the traveling salesman problem," in *Proceedings of an international conference on genetic algorithms and their applications*, vol. 154, pp. 154–159, Hillsdale, NJ, 1985.

[57] B. Waggener, W. N. Waggener, and W. M. Waggener, *Pulse Code Modulation Techniques*, Springer Science & Business Media, 1995.