



# Trajectory optimization of space vehicle in rendezvous proximity operation with evolutionary feasibility conserving techniques

Abolfazl Shirazi <sup>a,\*</sup>, Josu Ceberio <sup>b</sup>, Jose A. Lozano <sup>a,b</sup>

<sup>a</sup> Basque Center for Applied Mathematics BCAM, Mazarredo, 14, Bilbao, 48009, Bizkaia, Spain

<sup>b</sup> University of the Basque Country UPV/EHU, Manuel Lardizabal pasealekua, 1, Donostia, 20018, Gipuzkoa, Spain

## ARTICLE INFO

### Keywords:

Trajectory optimization  
Space transportation  
Estimation of distribution algorithms  
Satellite rendezvous  
Proximity operation

## ABSTRACT

In this paper, a direct approach is developed for discovering optimal transfer trajectories of close-range rendezvous of satellites considering disturbances in elliptical orbits. The control vector representing the inputs is parameterized via different interpolation methods, and an Estimation of Distribution Algorithm (EDA) that implements mixtures of probability models is presented. To satisfy the terminal conditions, which are represented as non-linear inequality constraints, several feasibility conserving mechanisms associated with learning and sampling methods of the EDAs are proposed, which guarantee the feasibility of the explored solutions. They include a particular implementation of a clustering algorithm, outlier detection, and several heuristic mapping methods. The combination of the proposed operators guides the optimization process in achieving the optimal solution by surfing the regions of the search domain associated with feasible solutions. Numerical simulations confirm that space transfer trajectories with minimum-fuel consumption for the chaser spacecraft can be obtained with terminal condition satisfaction in rendezvous proximity operation.

## 1. Introduction

Successful space transportation is a fundamental goal in many space scenarios as it has direct effects on various space missions, such as spacecraft guidance and tracking (Zhang et al., 2020), rendezvous, docking, on-orbit refueling (Zhu et al., 2020), and orbital maintenance (Ramteke and Kumar, 2022). In such type of missions, the overall process of relative spacecraft maneuvering lies in transferring the chaser spacecraft, traveling in its initial trajectory, to a distance with acceptable relative velocity close to the chief spacecraft. In recent years, numerous efforts have been dedicated to trajectory optimization of space vehicles in close-range rendezvous operations in the literature (Shirazi et al., 2018). However, designing a robust and efficient algorithm to extract the optimal feasible trajectory while satisfying various requirements of the space mission is still challenging due to the complexity of constraints.

There are two categories of constraints in real-world applications. Depending on the priority of the constraint satisfaction and optimizing the objective functions (either minimization or maximization), the problem may include *soft* or *hard* constraints. According to the definition by Malan et al. (2015), if constraints and objectives are treated with equal priorities, the constraints can be categorized as *soft*. However, in problems with *hard* constraints, the first priority is to satisfy the constraints rather than optimizing the objective function. In close-range space rendezvous, reaching the target spacecraft with

respect to the desired relative distance and velocity is the top priority. In the other word, the existing constraint is the satisfaction of terminal conditions for two state variables, including the vectors of relative position and relative velocity of the chaser spacecraft. The main difficulty in satisfying the constraint in this problem is due to non-linearity of the constraint, and low feasibility ratio of the search domain. These features along with the mentioned priority put the existing constraint into *hard* constraint category in the current problem.

Different methods have been proposed to deal with constraints in problems dedicated to the optimization of transfer trajectories of satellites and space systems. In special cases, such as multi gravity-assisted maneuvers (Zuo et al., 2020), converting the mathematical model of thrust vector from continuous to impulsive enforces boundaries with upper and lower values for decision parameters. However, in tackling close-range space rendezvous, additional considerations are required since the formulation of the problem is more complex. This complexity makes the optimization of transfer trajectory difficult and therefore, the discovery of solutions that are feasible and have high quality will be challenging.

As an example, besides the methodologies based on closed-loop control techniques, similar to the approaches that rely on the Lyapunov function (Tian and Jia, 2017), another solution is to apply specific modifications to the variables or the problem formulation for constraints satisfaction. Shape-based techniques and control transformation (Ayyanathan and Taheri, 2022) are in this category of methods.

\* Corresponding author.

E-mail address: [ashirazi@bcamath.org](mailto:ashirazi@bcamath.org) (A. Shirazi).

Although the effectiveness of these approaches have already been demonstrated in the literature, their dependency to the problem prevents generalization (Vijayakumar and Abdelkhalik, 2022). In addition to these methods, another option is to tackle the existing constraints within the algorithm itself (Fossà and Bettanini, 2020). Non-linear Programming (NLP) methods suited for constrained optimization problems are typical options to achieve feasible solutions. However, the search process in gradient-based methods is prone to get stuck in the local optimal region of the search space. For this reason, Evolutionary Algorithms (EAs) (Tang et al., 2021) have shown to be more effective to deal with constraints in space rendezvous missions (Liu et al., 2019). While using EAs, utilizing penalty functions is a common choice to satisfy constraints (Ostman, 2019). These approaches can be applied in a variety of problems, such as trajectory optimization of aeroassisted orbital transfer (Chai et al., 2018), design and optimization of interplanetary trajectories (Zotes and Peñas, 2012), and min-fuel orbit rising (Chai et al., 2021). However, there are no guarantees that the developed method discovers feasible trajectories when the constraints are transformed into main objectives using penalty functions, as this approach is suitable in dealing with problems with soft constraints. Since the optimization of transfer trajectory in space rendezvous mission is a problem with hard constraints, developing methodologies that ensure the constraints satisfaction is necessary.

Motivated by the discussed challenges, an Estimation of Distribution Algorithm (EDA) is developed in this paper. Problems involved with hard constraints are aimed in designing the proposed EA in this research. Particularly, the algorithm efficiently constructs the feasible model of solutions by utilizing a mixture of probabilistic models. Several mechanisms correlated with learning and sampling methods are presented, which assist the algorithm to achieve only feasible solutions within the optimization run. As for the learning mechanism of the EDA, some techniques including feasible conserving clustering and outlier detection are presented. The clustering mechanism forms the mixture of probabilistic models, adapted from the feasibility of the solution domain, while the outlier detection mechanism enhances the optimization process convergence. Also, within the sampling step, different heuristic mapping methods are developed, which help the optimization process continue towards exploring feasible domain. The presented methodology invariably yields feasible individuals and therefore is a potential tool for discovering transfer trajectories of high quality in space rendezvous. In this regard, some attempts have been proposed in developing sampling methods for probabilistic models to generate only feasible solutions for particular kinds of constraints (Shirazi, 2021). However, the methods were only suited in combinatorial optimization (Shirazi et al., 2022).

The proposed algorithm is analyzed and applied in the trajectory optimization of spacecraft in satellite close-range rendezvous with various initial conditions. Orbital disturbances are considered in elliptical orbits for the space missions. The performance of the algorithm is evaluated by comparing the numerical results with an approach based on implicit Lyapunov function, as it has been frequently used in the literature. The obtained results demonstrate the efficiency and the robustness of the developed algorithm in finding feasible transfer trajectories of close-range space rendezvous missions. The rest of this paper is organized as follows. Section 2 is dedicated to the dynamics of the spacecraft in the close-range space rendezvous mission along with mathematical modeling of the inputs, representing the control variables. The proposed optimization algorithm with the feasibility conserving mechanisms is described in Section 3. Afterwards, Section 4 is devoted to the achieved results for validating the performance of the algorithm. Section 5 includes several conclusions of the paper.

## 2. Problem description and mathematical model

The initial step for dealing with the trajectory optimization of spacecraft in close-range space rendezvous missions is the mathematical

modeling of system dynamics (Shirazi et al., 2018). In this paper, formulation of the dynamics of the chaser spacecraft has been taken into consideration with respect to orbital perturbations in elliptical orbits. Then, discretization scheme is utilized and the thrust vector is parameterized in a direct approach for solving the problem.

### 2.1. System dynamics

The dynamics of the spacecraft in close-range space rendezvous mission can be described via variety of mathematical representations. Sullivan et al. (2017) provided a complete list of relative dynamics models seek to describe the motion of a spacecraft. The high diversity of models is due to several factors such as the inclusion of perturbations, and the shape of the target orbit. In this work, the general non-linear model as in Alfriend et al. (2009) is utilized, since the aim of the research is to develop an EA for any type of space orbit considering perturbations. In this regard, the inertial equations of motion of the target spacecraft are given by:

$$\ddot{\vec{r}}_0 = -\frac{\mu}{r_0^3}\vec{r}_0 + \vec{d}_0 \quad (1)$$

where  $\vec{r}_0$  is the position vector of the target spacecraft in ECI (Earth-Centered Inertial) frame,  $\mu$  denotes the constant for Earth's gravitation, and  $\vec{d}_0$  denotes the disturbances due to orbital perturbations acted on the target spacecraft. In a similar fashion, the inertial equations of motion of the chaser are:

$$\ddot{\vec{r}}_1 = -\frac{\mu}{r_1^3}\vec{r}_1 + \vec{d}_1 + \vec{u}_1 \quad (2)$$

with  $\vec{r}_1$  as the position vector of the chaser in ECI frame, and  $\vec{u}_1$  and  $\vec{d}_1$  as the control vector and disturbances acted on the chaser spacecraft respectively. It is clear that the difference between the equation of motions of two spacecraft is the control vector, since the target spacecraft is assumed to be non-maneuverable ( $\vec{u}_0 = 0$ ). Consideration of uncertainties in estimation of state vectors during the proximity operation applies a significant modification of the dynamical equations. Such a consideration is left to future work. Following the relative position of the chaser in Euler–Hill frame, the relative position is defined as:

$$\vec{\rho} = \vec{r}_1 - \vec{r}_0 \quad (3)$$

with  $\vec{\rho} = [x, y, z]^T$ , the equations of the relative motion are:

$$\ddot{x} - 2\dot{\theta}_0\dot{y} - \ddot{\theta}_0y - \dot{\theta}_0^2x = -\frac{\mu(r_0+x)}{[(r_0+x)^2+y^2+z^2]^{\frac{3}{2}}} + \frac{\mu}{r_0^2} + d_x + u_x \quad (4)$$

$$\ddot{y} + 2\dot{\theta}_0\dot{x} + \ddot{\theta}_0x - \dot{\theta}_0^2y = -\frac{\mu y}{[(r_0+x)^2+y^2+z^2]^{\frac{3}{2}}} + d_y + u_y \quad (5)$$

$$\ddot{z} = -\frac{\mu z}{[(r_0+x)^2+y^2+z^2]^{\frac{3}{2}}} + d_z + u_z \quad (6)$$

with time as the independent variable,  $\vec{u}_H = [u_x, u_y, u_z]$  as the acceleration due to control input in Euler–Hill frame,  $\theta_0$  denoting the true anomaly of the target spacecraft, and  $\vec{d}_H = [d_x, d_y, d_z]$  as the disturbance acceleration in Euler–Hill frame.

Clearly, the components of disturbances in Euler–Hill frame can be obtained from the transformation of  $\vec{d}_1$  from ECI frame via  $\vec{d}_H = T_E^H \vec{d}_1$  with the transformation matrix as:

$$T_E^H = [\hat{\vec{r}}_0 \quad \frac{\hat{h}_0 \times \vec{r}_0}{\|\hat{h}_0 \times \vec{r}_0\|} \quad \hat{h}_0]^{-1} \quad (7)$$

where  $\vec{h}_0$  represents the angular momentum vector of the target spacecraft, and  $(\cdot)$  represents a unit vector. Two types of orbital perturbations are considered as disturbances in this research for both space vehicles, including atmosphere drag and Earth-oblateness ( $J_2$  zonal harmonic). The perturbed acceleration due to atmospheric drag  $\vec{p}_{atm}$  is modeled as:

$$\vec{p}_{atm} = -\frac{1}{2}\rho v_{rel}(\frac{C_D A}{m})\vec{v}_{rel} \quad (8)$$

with  $C_D$ ,  $A$ , and  $m$  as the dimensionless drag coefficient, frontal area of the spacecraft, and the mass of the spacecraft.  $\vec{v}_{rel}$  is the spacecraft velocity relative to the atmosphere as  $\vec{v}_{rel} = \vec{v} - \Omega_E \times \vec{r}$  with  $\Omega_E$  as the Earth's rotational speed. The U.S. standard atmosphere model USSA76 is utilized for the variation of atmosphere density  $\rho$  (Tewari, 2007). The perturbing gravitational acceleration  $p_g$  due to  $J_2$  is calculated as:

$$\vec{p}_g = \frac{3}{2} \frac{J_2 \mu R^2}{r^4} \left[ \frac{x}{r} \left( 5 \frac{z^2}{r^2} - 1 \right) \hat{i} + \frac{y}{r} \left( 5 \frac{z^2}{r^2} - 1 \right) \hat{j} + \frac{z}{r} \left( 5 \frac{z^2}{r^2} - 1 \right) \hat{k} \right] \quad (9)$$

where  $R$  is the Earth's equatorial radius and  $J_2 = 0.00108263$ . Having the two described perturbation accelerations, the overall disturbance is obtained as  $\vec{d} = \vec{p}_{atm} + \vec{p}_g$ . It is worthy to note that since the main concern in this research is the algorithm development rather than high precision orbit propagation, these two orbital perturbations are considered as the main disturbances for the rendezvous mission. However, it is possible to consider more orbital perturbations and other types of disturbances in the model for increasing the accuracy of the simulation.

The decreasing variation of spacecraft mass is due to the acting thrust of the propulsion system. Considering  $\vec{T} = [T_x, T_y, T_z]$  as the vector representing the thrust components, the decrease in the mass of the spacecraft can be shown according to:

$$\dot{m} = - \frac{\|\vec{T}\|}{I_{sp} g_0} \quad (10)$$

where  $I_{sp}$  denotes the specific impulse of the propellant, and  $g_0$  is the standard acceleration of gravity at sea-level ( $g_0 = 9.807 \text{ m/s}^2$ ). Also, the control acceleration vector from the thrust profile can be presented by  $\vec{u}_H = \vec{T}/m$ . Having  $t_f$  as the final time of transfer, if the thrust vector time profile  $\vec{T}$  is known, the presented equations of motion for the system can be integrated with respect to  $0 < t < t_f$  as the time interval. Following the integration, it gives the relative position and velocity associated with the chaser spacecraft as time histories. The final value of state vectors as  $\vec{r}_f = [x_f, y_f, z_f]$  and  $\vec{v}_f = [\dot{x}_f, \dot{y}_f, \dot{z}_f]$  can be achieved by the end of the space rendezvous.

## 2.2. Thrust profile approximation

In direct approaches, the convergence of the optimization process is heavily affected by thrust vector parameterization. In this paper, the thrust magnitude, acted on the spacecraft, represented by  $T_x, T_y$  and  $T_z$  in each direction is approximated by letting  $N_p$  number of nodes as interpolation points in the time interval of  $0 < t < t_f$ , with respect to limits of  $T_{\min} < [T_x, T_y, T_z] < T_{\max}$  as the allowable thrust boundaries. Considering  $N_p$  uniformly discretized nodes, the time domain of the transfer trajectory is decomposed into  $N_p - 1$  sub-intervals. Following this, the time interval is presented by the Lagrange polynomial as:

$$T'(t) = \sum_{k=1}^{N_p} \left( \prod_{j \neq k} \frac{t - t_j}{t_k - t_j} \right) p_k \quad (11)$$

where  $T'(t)$  represents any of the components of thrust vector ( $T_x, T_y, T_z$ ),  $N_p$  denotes the number of discrete nodes,  $p_k$  is the discrete nodes within the time interval, and  $t_k$  is the discretized times. Having the number of nodes  $N_p$  for each component of thrust vector, the time profile of thrust vector can be interpolated with various techniques. The employment of piecewise cubic Hermite interpolating polynomials (Phillips, 2003) is one of the most common methods. Different curves may be achieved, based on the selection of tangents in each point. There are three most common types of curves of the Hermite splines category, that are used frequently in various applications. Fig. 1 shows these splines for parameterizing the components of the thrust vector.

Three types of splines are employed in this research, including Shape Preserving (Huynh, 1993), Not-a-Knot (Dahlquist and Björck, 2008), and Catmull-Rom splines (Dahmen et al., 2012), hereinafter referred to SP, NK, and CR splines respectively. Based on polynomial

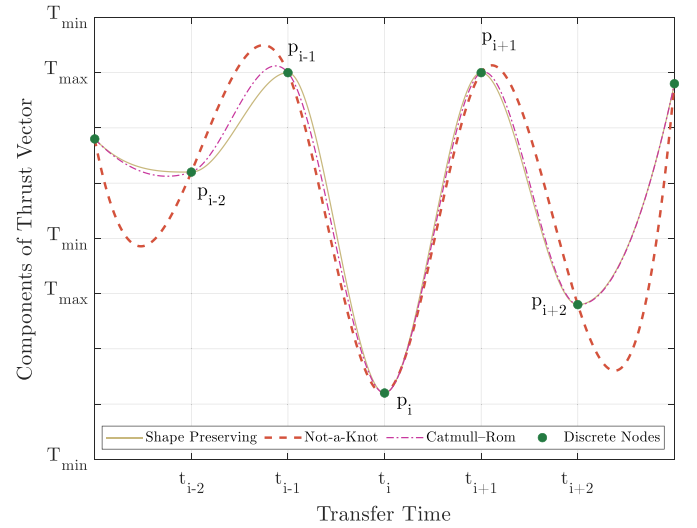


Fig. 1. Parameterizing the components of thrust profile via piecewise cubic Hermite splines.

approximation, these splines are continuous, and also have a continuous first derivative. Internal nodes and end nodes of each segment have different tangent values, which is the major difference between these splines. Formation of SP spline is based on having no local overshoots at the data points. The value of slope at each interior node is considered to be a weighted average value, associated with the piecewise linear interpolant slopes. Two end points are treated as single-side slope nodes. Solving a system of linear equations is not required for calculation of slopes at the points in this spline. NK spline is a curve with smoother formation. Its second derivatives have continuous variation. Continuous third derivative is also obtainable with respect to some round-off error (Behforooz, 1992). CR spline approximation is the third type of the Hermite interpolation method, and it benefits from a balanced flatness. The computation of slope value for CR spline at discrete nodes relies on the neighboring nodes. Continuous second derivative is not possible in this kind of spline, and shape preserving is not guaranteed. However, it can be calculated rapidly using a convolution operation. Detailed discussion regarding these techniques is beyond the scope of this article and the reader is recommended to refer to the provided Refs. Huynh (1993), Dahlquist and Björck (2008) and Dahmen et al. (2012).

## 3. Method of solution

The initial values of state vector are obtained with respect to the given position and velocity of the spacecraft relative to the target space vehicle at the initial time as  $\vec{r}_i = [x_i, y_i, z_i]$  and  $\vec{v}_i = [\dot{x}_i, \dot{y}_i, \dot{z}_i]$  along with the orbital elements of the space orbit. Having the specific impulse of the propulsion system along with the initial mass of the space vehicle, the propagation of rendezvous path with respect to the given thrust profile can be done. Therefore, formalizing a continuous optimization problem based on the given parameters becomes possible. The problem formation along with the proposed approach are illustrated in Fig. 2. In this regard, the objective is to achieve the optimal thrust profile of the spacecraft, which satisfies the terminal conditions, while minimizing the fuel consumption in a predefined mission for a close-range space rendezvous. Constraints are formed according to the relative distance and velocity of the chaser spacecraft at the final time step, while the fuel mass is treated as the main objective function to be minimized. In the other word, close-range space rendezvous is transformed into an optimization problem with inequality constraints. The overall scheme

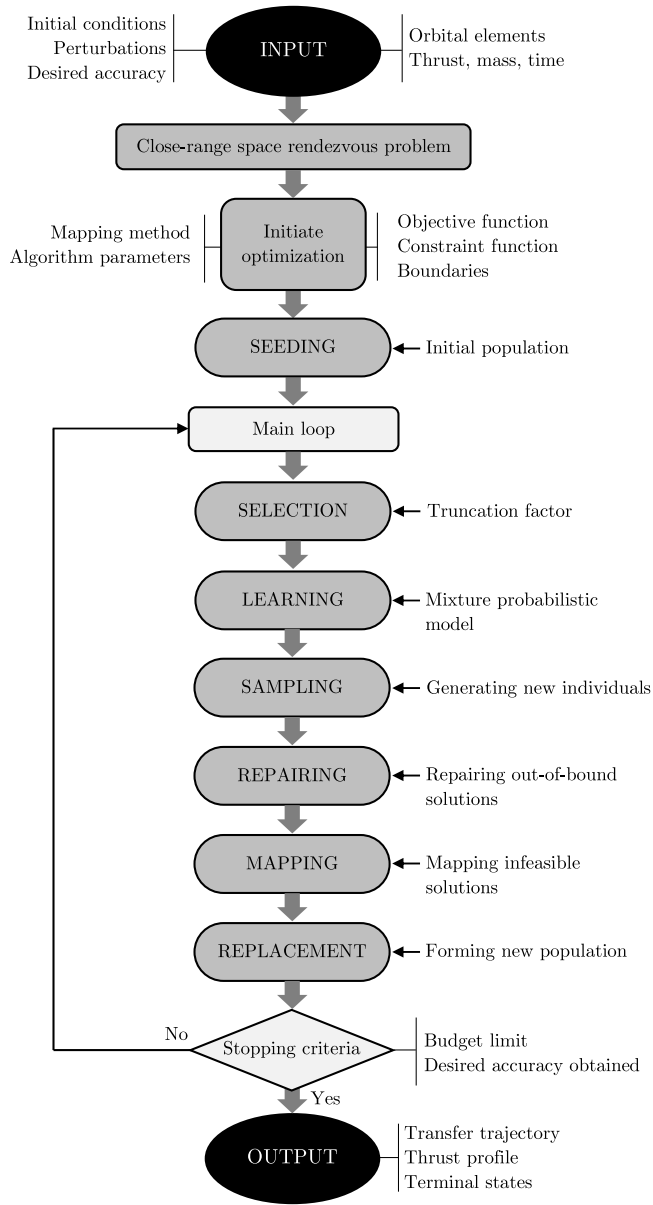


Fig. 2. Schematic diagram of the proposed approach.

of the problem is presented as follows:

$$\begin{aligned} & \text{Minimize } F(X) \quad X = (X_1, X_2, \dots, X_n) \\ & \text{Subject to } G(X) \leq 0 \\ & \quad X_{\min} < X_i < X_{\max} \end{aligned} \quad (12)$$

where  $F(X)$  and  $G(X)$  denote the objective function and the constraints function respectively, and  $n$  is the total number of decision variables. Following the proposed model of the problem, the interpolation nodes for thrust vector components are the decision variables  $X = (X_1, X_2, \dots, X_n)$  as:

$$X = p_i \quad (i = 1, \dots, 3N_p) \quad (13)$$

where  $n = 3N_p$ , while the boundaries are  $X_{\min} = T_{\min}$  and  $X_{\max} = T_{\max}$ . Moreover, the objective function is represented as:

$$F(X) = m_f = \int_0^{t_f} \dot{m} dt \quad (14)$$

where  $m_f$  is the consumed fuel mass within the transfer process. It is noteworthy that once the thrust profile is generated based on the

decision variables in each objective function evaluation, infeasible control laws outside of the predefined limits of  $T_{\min}$  and  $T_{\max}$  due to spline overshoots are corrected as:

$$\begin{aligned} T'(t) &= \min(T'(t), T_{\max}) \\ T'(t) &= \max(T'(t), T_{\min}) \end{aligned} \quad (15)$$

Considering the terminal conditions as the constraints in optimization, the relative distance and velocity of the spacecraft by the end of the transfer are formulated as:

$$G(X) = \begin{bmatrix} \|\vec{r}_f\| - \sigma_r \\ \|\vec{v}_f\| - \sigma_v \end{bmatrix} \quad (16)$$

where  $\sigma_r$  and  $\sigma_v$  are the desired final values for distance and velocity of the spacecraft relative to the chief space vehicle as the space mission requirement. In order to tackle this optimization problem with constraints in continuous domain, an EDA-based algorithm is developed.

### 3.1. Estimation of distribution algorithms

A special class of EAs are EDAs, which operate using probabilistic models (Larrañaga and Lozano, 2001). Alg. 1 shows the general pseudo-code for EDAs. As can be appreciated, the main loop of EDAs starts with the selection method (Alg. 1 Line 5). Having  $X_i$  as the current population for  $i$ th iteration, a selection of high quality feasible solutions is selected from the current population. The chosen population is utilized to estimate the parameters of the probability model. To this end, the top  $N_{sel}$  number of individuals from the current population are selected. This number of selected individuals is obtained as:

$$N_{sel} = \tau N_{pop} \quad (17)$$

where  $\tau$  is the truncation factor and  $N_{pop}$  is the population size. The objectives of the current population are sorted as:

$$[F_i^{\mathcal{S}}, \mathbf{I}] = \mathcal{S}^{\uparrow}(F_i) \quad (18)$$

where  $\mathcal{S}^{\uparrow}$  is the sorting operator,  $F_i$  is the vector of objective values for the current population as  $F_i = F(X_i)$ ,  $F_i^{\mathcal{S}}$  is the sorted vector of objective values, and  $\mathbf{I}$  is the index vector associated with the sorted vector. The individuals of the current population are rearranged based on the obtained sorting index as:

$$X_i^{\mathcal{S}} = X_i[\mathbf{I}] \quad (19)$$

---

#### Algorithm 1: General workflow of Estimation of Distribution Algorithms

---

```

1  $X_0 \leftarrow$  Generate  $N_{pop}$  individuals as the initial population
2 repeat
3   for  $i = 1, 2, \dots$  do
4     /* SELECTION METHOD */
5      $X_i^{sel} \leftarrow$  Select  $N_{sel} < N_{pop}$  individuals from  $X_i$  according
      to the selection method
6     /* LEARNING METHOD */
7      $\mathcal{P}_i(X) = \mathcal{P}(X|X_i^{sel}) \leftarrow$  Estimate the probability
      distribution
8     /* SAMPLING METHOD */
9      $X_i^{sam} \leftarrow$  Sample  $N_{pop}$  individuals from  $\mathcal{P}_i(X)$ 
10    /* REPLACEMENT METHOD */
11     $X_{i+1} \leftarrow$  Form the new population from  $X_i^{sam}$  and  $X_i$ 
      according to the replacement method
12  end for
13 until a stopping criterion is met ;
14 return  $X_i$ 

```

---



with  $X_i^{\mathcal{S}}$  representing the sorted population. Then, the selected population is extracted as:

$$X_i^{sel} = X_i^{\mathcal{S}}[1 : N_{sel}] \quad (20)$$

Then in the next step, a probability distribution  $P_i(X)$  associated with the set of most promising individuals  $X_i^{sel}$  at each generation is learned (Alg. 1 Line 6). When approaching constrained continuous optimization with EDAs, similar to most of the meta-heuristic algorithms, there are no guarantees that the solutions sampled from the probabilistic model satisfy the constraints of the problem. To deal with this issue, in our proposal, a mixture of Gaussian distributions is employed. The density function of the proposed mixture of probabilistic models is defined as:

$$P_i(X_i^{sel}) = \sum_{k=1}^{N_c} \pi_k P_k(X_i^{sel} | \mu_k, \Sigma_k) \quad (21)$$

where each  $P_k(X | \mu_k, \Sigma_k)$  is a multivariate Gaussian distribution with  $\pi_k > 0$  as the coefficient of mixture for the  $k$ th component ( $\sum_{k=1}^{N_c} \pi_k = 1$ ), and  $\mu_k$  and  $\Sigma_k$  denoted as the mean value and the covariance matrix of the  $k$  model for  $k = 1, \dots, N_c$ , with  $N_c$  as the total number of components. In the learning step, the selected population at the beginning of each iteration is divided into several clusters of solutions so that the centroids of the clusters  $\mu_k$  are in the feasible region. As a result, when a Gaussian model is learned from each cluster, the probability of sampling feasible solutions becomes high. This process is the first required mechanism for constraint satisfaction in this research.

Having all of the centroids ( $\mu_k$ ) inside the solution domain associated with feasible solutions remarkably reduces the generation of infeasible solutions within the process of sampling. However, when the new population is sampled, some of the new individuals may still be formed within the infeasible solution domain. Therefore, a mechanism that corrects the newly sampled infeasible solutions is also required within the sampling process in each component (Alg. 1 Line 7). In this paper, several mapping mechanisms are proposed, which move the newly sampled solutions from the infeasible region into the nearby feasible region. As the centroid in each component of the Gaussian mixture model is inside the feasible region, the proposed mapping mechanisms exploit this situation and use each centroid as the target point for mapping any infeasible solution. One idea is to start shifting the infeasible points towards their respective centroid in each mixture component until they enter the feasible region. This is the other mechanism, which is associated with the sampling process and consists of mapping infeasible solutions into the feasible region. Although this mechanism guarantees the feasibility of all newly sampled infeasible solutions, it may produce a strong bias in the search process, which makes the algorithm more likely to converge to low-quality solutions (Coello, 2002). As a result, the covariance matrix for each component ( $\Sigma_k$ ) may suffer from unwanted shrinking due to the mapping mechanism. This drawback is due to the fact that the mapping process for all infeasible solutions in each component is towards the centroid of that component. It may result in low-quality solutions when the best point, or the set of points with the best objectives, is far away from the centroid in each component.

To compensate this weakness and prevent unwanted shrinking of  $\Sigma_k$ , the proposed option is to increase the number of components back in the learning process. The proposed method works as follows: for each cluster, those solutions that could be considered as outliers are chosen. Then, for each of these outlier solutions, we select those that have a good enough objective function value and a new component of the mixture is created with this solution as the mean value of the Gaussian distribution. Having this consideration for each component, the exploration is increased and the unwanted reduction in the values of covariance is prevented.

By sampling data based on the probabilistic model generated in each generation while mapping them into the feasible region, a new population of solutions for the problem is formed as  $X_i^{sam}$ . Finally, the

replacement operator (Alg. 1 Line 8) combines the sampled solutions and the previous solutions, creating a new generation as:

$$X_{i+1} = \mathcal{S}^\uparrow([X_i^{sam}, X_i]) \quad (22)$$

$$X_{i+1} = X_{i+1}[1 : N_{pop}] \quad (23)$$

The algorithm halts the iteration process and returns the best solution found across the generations when a certain stopping criterion is met (Alg. 1 Line 10), such as a maximum number of generations, homogeneous population, or lack of improvement in the solutions.

The employment of the aforementioned mechanisms means that the algorithm generates only feasible solutions in every generation, while preventing the algorithm from converging to low-quality solutions. Overall, the proposed algorithm include two mechanisms associated with the learning method and one mechanism associated with the sampling method. The mechanisms associated with the learning step are coupled with feasible clustering and outlier detection techniques, while the one associated with the sampling is the mapping mechanism. In the following sections, these mechanisms will be discussed in detail.

### 3.2. Feasibility conserving clustering

By disabling the feasibility conserving mechanisms and tackling the unconstrained version of the proposed algorithm, initial solutions inside the feasible region of the solution domain are achieved. Within this step, the algorithm is run while temporarily treating Eq. (16) (the constraints) as the objective function. The inner loop continues until the required number of solutions inside the feasible region are detected. It should be noted that if the parameters defining the close-range space rendezvous problem or interpolation setup are poorly chosen, it is possible that the initial feasible population is not achieved. Choosing very low number of interpolation points is one of the causes that will be highlighted in comparative analysis in Section 4.2. The other cause can be very low and unreasonable level of thrust for a desired rendezvous. However, in general, since there are no equality constraints in the defined problem and the constraints are treated as inequality form, the initial population of feasible solution will likely be obtained in this method if the setup parameters are fairly chosen.

At each main iteration, the process starts by having a population of feasible solutions, obtained from the previous iteration. High quality feasible solutions are selected from the current population as described previously. In the proposed stage for learning process, two steps exist for the establishment of the Gaussian mixture model. These steps are

---

#### Algorithm 2: Learning Mechanism

---

**Input:**  $X_{sel}, F_{sel}, G(X), \alpha, \lambda$

```

1  $N_{sel} \leftarrow \text{size}(X_{sel})$ 
2 for  $I \leftarrow 1$  to  $N_{sel}$  do
3    $[l, \mu] \leftarrow \text{kmeans}(X_{sel}, I);$ 
4    $G_\mu \leftarrow \text{EVAL}(\mu, G(x))$ 
5   if  $\max(G_\mu) \leq 0$  then
6     BREAK;
7 end for
8 ESTABLISH  $\Phi$  FROM  $[\mu, X_{sel}(I)]$ ;  $N_c \leftarrow \text{size}(\Phi)$ 
9 for  $I \leftarrow 1$  to  $N_c$  do
10  RETRIEVE  $[\hat{X}, \hat{F}, \hat{\mu}, \hat{\sigma}]$  FROM  $\Phi(I)$ 
11   $[\hat{X}_{sel}, \hat{F}_{sel}] \leftarrow \text{SELECTION}(\hat{X}, \hat{F}, \alpha)$ 
12   $d \leftarrow \|\hat{X}_{sel} - \hat{\mu}\|$ ;  $\mathcal{J} \leftarrow 0$ 
13  if  $d > \lambda \times \hat{\sigma}$  then
14     $\mathcal{J} \leftarrow \mathcal{J} + 1$ 
15    ESTABLISH  $\hat{\phi}$  FROM  $[\hat{\mu}, \hat{X}_{sel}]$ ;  $\phi(j) \leftarrow \hat{\phi}$ 
16 end for
Output:  $\phi, \Phi$ 
```

---

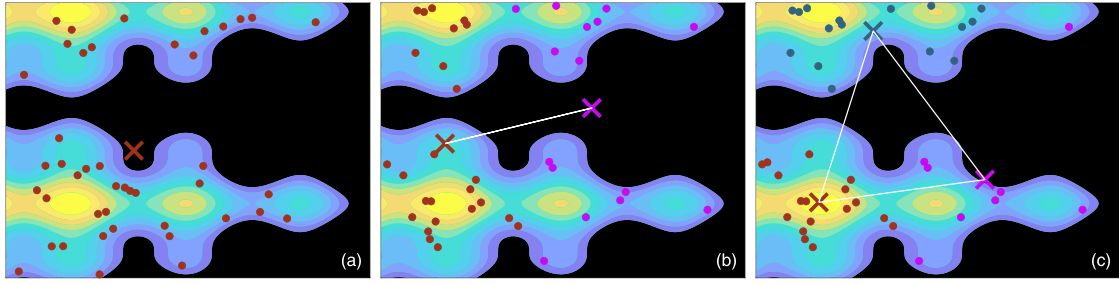


Fig. 3. Feasibility conserving clustering process: (a) One cluster with single infeasible centroid (b) Two clusters, one with an infeasible centroid (c) Three clusters with all feasible centroids.

demonstrated in the Pseudo code representation of the learning process, presented in Alg. 2. As for the first step, the minimum number of mixture components is obtained in such a way that all the respected centroids of the components are positioned inside the search domain associated with feasible solutions (Alg. 2 Lines 2 to 8). Then, the second step is to extract outliers in each of the components and verify their quality. If they are high quality solutions, the mechanism takes them as additional separate components, considering each of the points as the centroid for a new mixture component (Alg. 2 Lines 9 to 16). The aim of the second step, as mentioned, is to recoup with unwanted shrinking of the covariance matrix due to the mapping mechanism, which will take place after sampling new solutions.

The clustering process of the set of selected solutions is shown in Fig. 3, in a 2D optimization problem. In the plots, the black region is associated with the infeasible solution domain due to the existing constraint of the problem. Also, the color-mapped area shows the feasible solution domain. The selected individuals are marked and different numbers of clusters ( $k$ ) are taken into account. *kλmeans++* is utilized as the clustering method (Wu, 2012) (Alg. 2 Line 3). In this method, assuming  $\|X_i - X_j\|$  as the Euclidean distance between  $X_i$  and  $X_j$ , the goal is to choose a set  $C$  of  $k$  centroids to minimize  $\phi_Y(C)$  as:

$$\phi_Y(C) = \sum_{y \in Y} d^2(y, C) = \sum_{y \in Y} \min_{i=1, \dots, k} \|y - c_i\| \quad (24)$$

where  $C = \{c_1, \dots, c_k\}$ , and centroids are calculated as:

$$c(Y) = \frac{1}{|Y|} \sum_{y \in Y} y \quad (25)$$

where  $Y$  is a subset of points, defined as  $Y \subseteq X$ . Regarding this clustering method, in plot (a), only a single cluster is formed ( $k = 1$ ) and therefore one centroid exists, representing the mean value of the selected individuals. According to the plot, the centroid in this case is located in the infeasible region of the search domain. Following this placement, the probabilistic model does not possess the requirement for the satisfaction of constraints. Increasing the number of clusters, in plot (b), the selected individuals are divided into two components ( $k = 2$ ). The locations of the centroids show that one resides inside the feasible domain, while the other does not. It certifies that the current mixture model is also not appropriate for the satisfaction of constraints. By increasing the number of clusters to three ( $k = 3$ ), plot (c) in Fig. 3 is achieved. As can be inferred, all of the points associated with the centroids of the mixture model are positioned inside the feasible domain. Based on this state, the mixture of Gaussian distributions model is learned with the current combination. Since all of the centroids are feasible in this case, any newly sampled solutions are likely to be inside the feasible region of the search domain. The number of clusters obtained in this approach is the least number of components associated with feasible individuals as their centroids (Alg. 2 Line 8). At this point, it is possible to increase the number of clusters and achieve other type of mixtures for Gaussian distributions. However, there is no actual necessity for this increment (Alg. 2 Lines 5 and 6), as it makes computation time increase drastically.

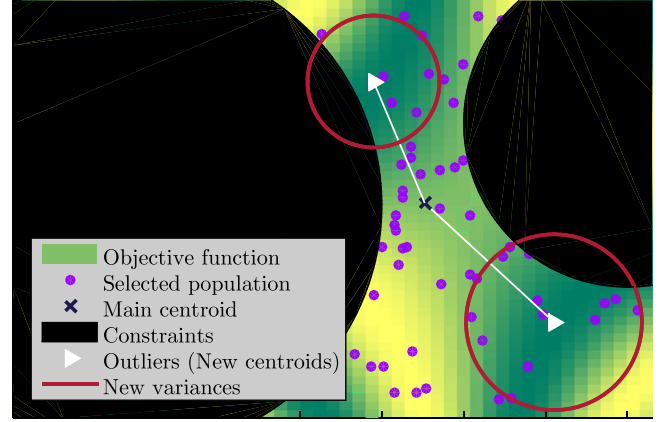


Fig. 4. Outlier detection for new clusters ( $\alpha = 0.01$ ,  $\lambda = 1\sigma$ ).

### 3.3. Outlier detection

In the second step, more components are included in the model. As stated, the goal of this step is to reduce the loss of covariance, which occurs due to the mapping mechanism, which will be utilized in the proposed method (Coello, 2002). In this step, more components are added to the mixture, based on evaluating the outliers in each component after finishing the first step.

First, each component of the Gaussian mixture model  $\Phi$  that has been identified in the previous step is evaluated (Alg. 2 Line 10) to check if it has outliers with respect to the Z-score method of outlier detection (Hodge and Austin, 2004) considering an arbitrary distance  $\lambda$  from the centroids:

$$Z_i = \frac{\|X_i - \mu_k\|}{\lambda} \quad (i = 1, \dots, N_k) \quad (26)$$

where  $Z_i$  is the score for solution  $X_i$ ,  $\mu_k$  is the centroid of the  $k$ th cluster,  $N_k$  is the number of solutions in the  $k$ th cluster,  $Z_i < 1$  is the trigger for detecting outliers, and  $\lambda$  is the distance in terms of variance ( $\lambda = 1\sigma, 2\sigma, \dots$ ). Then, the quality of the detected outliers is checked via a threshold  $0 < \alpha < 1$  (Alg. 2 Lines 11 to 13). Based on this mechanism, if a solution is detected as an outlier and is ranked within the top  $\alpha$  percentage of the individuals with highest quality, it will be treated as the centroid of a new component  $\hat{\phi}$  in the current mixture (Alg. 2 Line 15). Depending on the distribution of solutions in each cluster, additional components are identified based on this approach. An independent multivariate Gaussian distribution is considered for new components, which are added to the mixture. In this distribution, the variance for each dimension is computed according to the distance of the outlier from the initial centroid of every component. An example of this method is shown in Fig. 4.

As shown, the centroid of a component and its respective population are illustrated. Based on this representation, the objective value of two individuals is lower than the threshold, thus are detected as the

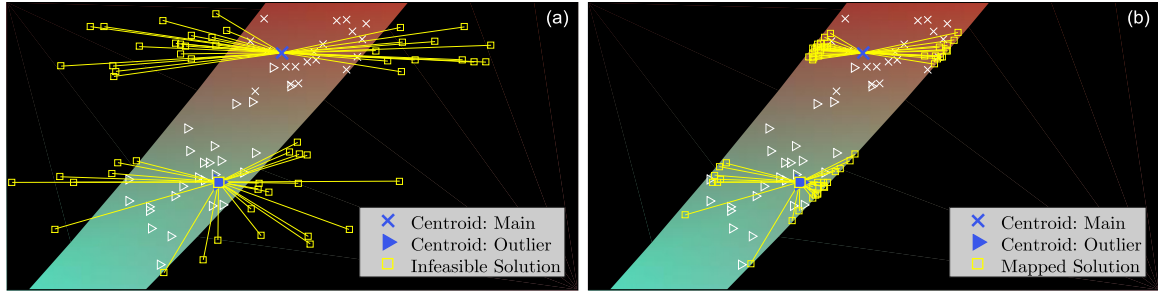


Fig. 5. New individuals from the mixture of Gaussian distribution, (a) Before mapping (b) After mapping.

outliers in the selected population. Therefore, they are remarked as the centroids for two newly generated components, attached to the mixture model. New variances are considered for each newly formed components, which are equal to half of the distance from the main centroid to the respective outlier. Overall, the formed mixture model consists of three components based on Gaussian distribution. The initial main component is due to the first step of the learning mechanism and two additional components are due to outliers.

The new solutions will be sampled in different ways for initial and additional components. For the initial components, new solutions are sampled with respect to the covariance matrix and the mean value of the solutions in each component. For the components that have been created based on the outlier detection method, the new solutions are sampled with respect to the variance that has been described.

### 3.4. Mapping mechanisms

The mixture components generated by the proposed learning mechanism have mean values inside the search domain associated with feasible solutions. However, there is a possibility that newly individuals are sampled within the infeasible region with respect to the variance of each component. To solve this issue, a few mechanisms for mapping the newly generated solutions are proposed. The overall structure of these mechanisms is provided in Al. 3, while Fig. 5 shows the developed mapping mechanism for a constrained optimization problem.

The provided plots indicate the mapping process within the optimization loop in one iteration. According to the illustrations, the proposed technique for mapping relies on the concept of moving infeasible solutions towards the comparative mean value in  $N_\delta$  minor iterations. The process continues until the shifted point resides in the feasible region. Based on the presented learning mechanism, the centroids of the components are feasible. Therefore, entering the feasible region while shifting the solutions towards the mean value is guaranteed. In Fig. 5(a), a mixture model with two Gaussian distributions is assumed as the probabilistic model. Newly generated samples are near the centroid of the components. However, new population includes infeasible samples, which are marked separately. The centroids are connected to their respective newly generated samples. In Fig. 5(b) a deterministic form is considered for mapping the solutions, with equally-spaced motion upon their respective centroids with  $N_\delta$  as the number of steps. In each step, the shifting is done with respect to the value of  $(\|c_k - X_s\|)/N_\delta$ , where  $X_s$  is the solution inside the infeasible region, aiming to be mapped towards the centroid  $c_k$  in the  $k$ th component. Following this process,  $N_\delta$  steps are included within the distance between the centroid and the infeasible solution. Note that, the last shift will be on the centroid at the final step. Therefore, regardless of the value of  $N_\delta$ , the feasibility of the final solution is guaranteed. As the value of  $N_\delta$  increases, the border between feasible and infeasible regions will be discovered with more details, and therefore, the mapping process will be more accurate. It is worth to mention that while

### Algorithm 3: Mapping Mechanism

---

**Input:**  $X_{rep}, \Phi, \phi, G(x), N_\delta, \text{MapMode}$

- 1 RETRIEVE  $X_{inf}$  FROM  $X_{rep}$
- 2 RETRIEVE  $\mu$  FROM  $\{\Phi, \phi\}$
- 3  $J \leftarrow 0$
- 4 **foreach**  $\hat{X}_{inf} = X_{inf}$  **do**
- 5    $J \leftarrow J + 1$
- 6   **for**  $I \leftarrow N_\delta$  **to** 1 **do**
- 7      $d \leftarrow \|\hat{X}_{inf} - \mu\|$
- 8     **if**  $\text{MapMode.type} = \text{'Linear'}$  **then**
- 9        $\delta \leftarrow d/I$
- 10    **if**  $\text{MapMode.type} = \text{'Bisection'}$  **then**
- 11       $\delta \leftarrow d/2$
- 12    **if**  $\text{MapMode.method} = \text{'Deterministic'}$  **then**
- 13       $\eta \leftarrow 1$
- 14    **if**  $\text{MapMode.method} = \text{'Stochastic'}$  **then**
- 15       $\eta \leftarrow \text{UNIF. DIST. } [0,1]$
- 16       $\hat{X}_{inf} \leftarrow \hat{X}_{inf} + \eta \times \delta$
- 17       $\hat{G} \leftarrow \text{EVAL}(\hat{X}_{inf}, G(x))$
- 18      **if**  $\hat{G} \leq 0$  **then**
- 19        **BREAK;**
- 20    **end for**
- 21     $X_{map}(J) \leftarrow \hat{X}_{inf}$
- 22 **end foreach**

**Output:**  $X_{map}$

---

shifting the solution towards their respective centroids, the process will be stopped as the solutions becomes feasible. Following the described process, various non-linear or stochastic approaches can be utilized for mapping the solutions. Four approaches are proposed in the following.

#### 3.4.1. Linear deterministic mapping

As described previously, the primary and most typical method for mapping the solutions is the Linear Deterministic (LD) method. In this approach, the distance between the solution, which is to be mapped, and the centroid of the component is distributed into segments with equal sizes (Alg. 3 Line 9). Solutions reside inside the infeasible region are shifted from their origin towards their respective centroid via these steps. The feasibility of the newly shifted solution is checked in each step (Alg. 3 Line 17). If the new solution enters the feasible region, the movement will stop (Alg. 3 Line 18). The shifting procedure can be represented as:

$$X_{s+1} = X_s + \delta \quad (27)$$

where  $X_s$  is the infeasible solution to be mapped,  $X_{s+1}$  is the new obtained solution after the shifting step towards the component's centroid  $c_k$ , and the parameter  $\delta$  is the shifting step, obtained as

$$\delta = \frac{c_k - X_{s_0}}{N_\delta} \quad (28)$$

Here,  $N_\delta$  denotes the arbitrary quantity of segments for the process, and  $X_{s_0}$  is the initial position of the solution, resides inside the infeasible region of the solution domain.

### 3.4.2. Linear stochastic mapping

Similar to LD, another mapping approach is Linear stochastic (LS). The key variation is that in every step, when the new solution is obtained, a random movement is applied on the solution (Alg. 3 Line 15) with respect to the parameter  $\eta$  as:

$$X_{s+1} = X_s + \eta \times \delta \quad (29)$$

In this definition  $0 < \eta < 1$ . This approach enforces a random search of the solution, during the process and may possess some preferences based on the search domain of the optimization problem.

### 3.4.3. Bisection deterministic mapping

The next approach is Bisection deterministic (BD) mapping. This approach relies on the concept of bisecting the interval frequently, between the infeasible solution and the respective centroid. In the shifting process, the distance is divided in two by calculating the middle point of the distance interval (Alg. 3 Line 11) as:

$$\delta = \frac{c_k - X_s}{2} \quad (30)$$

The midpoint solution is checked for feasibility. If it is found to be inside the feasible region, the shifting process stops. Otherwise, the mapping continues by taking a new interval between the current obtained solution and the respective centroid.

### 3.4.4. Bisection stochastic mapping

Similar to BD, bisection stochastic (BS) mapping is the other approach. In this method, newly obtained solution after each shift is moved in a random direction with respect to a variable radius  $\delta$  and random variable  $\eta$  (Alg. 3 Line 15). The value of this parameter is from zero to  $\delta_{\max}$ , which is equal to the distance between two steps in sequence.

## 4. Simulations

To validate the efficiency of the proposed algorithm, several experiments have been conducted. First, a close-range space rendezvous problem is considered with respect to four various initial conditions. Following the experiments, a comparison is made between the quality of the obtained solutions via the presented method and a method based on Lyapunov-function available from the literature.

Since parameter tuning is not the main purpose of this research, the arbitrary parameters are chosen as follows for the proposed algorithm in all of the runs. Having  $n$  as the number of decision variables, the maximum number of generations  $N_{gen}$  is considered as  $30 \times n$ , while the population size  $N_{pop}$  is set to  $20 \times n$ . The parameters for outlier detection, including the distance and the threshold, are chosen as  $\lambda = 1\sigma$  and  $\alpha = 0.1$  respectively. It should be highlighted that for each problem, the best selection of algorithm parameters is unique. Therefore, it is noteworthy that the results obtained in the simulations are valid for the given selection of algorithm parameters. Parameter tuning and analysis of the algorithm performance via altering its parameters are left to future work.

### 4.1. Robustness verification

Table 1 contains the parameters for close-range space rendezvous mission considered in the experiments. Note that the specified initial mass, reference area, and the drag coefficient stands for both target and chaser within the simulation. Since the main aim of the experiments is to analyze the robustness of the presented approach and evaluating the practicality of the schemes for interpolation, several initial conditions

**Table 1**

Close-range space rendezvous mission parameters

	Parameter	Value		Parameter	Value
Semi-major axis	$a$	12500 km	Initial mass	$m_0$	170 kg
Inclination	$i$	10°	Thrust limit	$T$	$\pm 120$ N
Eccentricity	$e$	0.4	Specific impulse	$I_{sp}$	300 s
Right ascension	$\Omega$	15°	Final distance	$\sigma_r$	10 m
Arg. of perigee	$\omega$	65°	Final velocity	$\sigma_v$	0.1 m/s
True anomaly	$\nu$	230°	Drag coefficient	$C_D$	2
Total Time	$t_f$	5 min	Reference area	$A$	3 m <sup>2</sup>

**Table 2**

Initial values for state vector.

	Relative distance			Relative velocity		
$C_1$	$\bar{r}_i = [+1300$	$+500$	$-1900]$ m	$\bar{v}_i = [-2.9$	$-1.2$	$+1.7]$ m/s
$C_2$	$\bar{r}_i = [+800$	$-1500$	$+1250]$ m	$\bar{v}_i = [+7.3$	$+6.8$	$+1.1]$ m/s
$C_3$	$\bar{r}_i = [-120$	$+2700$	$+1430]$ m	$\bar{v}_i = [+1.5$	$-12.1$	$-5.4]$ m/s
$C_4$	$\bar{r}_i = [-560$	$-320$	$-1790]$ m	$\bar{v}_i = [-2.1$	$+4.3$	$-14.8]$ m/s

based on the practicality in real-world operations of the close-range space rendezvous problem are assumed according to Table 2.

Based on the proposed approach, each experiment can be conducted with a different choice for variety of parameters and techniques. For each initial condition, these parameters are number of polynomial points for thrust components, type of the mapping technique, and the interpolation method. The number of interpolation nodes is considered as 5 to 24, which is 20 cases for this parameter. Results will indicate that no solutions with higher quality can be found with number of nodes more than 18 in the experiments. Also, with respect to three options for interpolating the thrust profile (SP, CR and NK) and four options for the mapping process (LD, LS, BD and BS), total number of 240 cases of setup regarding each initial condition is considered. Running the optimization algorithm 10 times for each case results in a vast database of solutions, including 2400 solutions to be analyzed. In each run, the obtained solutions are kept with its corresponding parameter selection. The top ten best obtained solutions are presented in Table 3. Results are separated for each initial condition.

According to the results, the high quality solutions correspond to 18 number of nodes for interpolating the components of thrust vector. The optimizer found solutions with almost similar quality by considering other number of nodes close to the top solution. Solving the problem with number of nodes higher than the top solution did not end in a better solution. Therefore, it is implied that the optimizer achieved the best available solution up to this point. Analysis of the employed mapping mechanisms in high quality solutions in Table 3 shows that BS mapping method is the most promising technique in finding the majority of the high quality solutions, regardless of the initial condition for this close-range space rendezvous mission. The question that whether the BS method is also the best mapping mechanism for another space rendezvous mission will be answered in the next subsection. According to the best obtained solutions, the time-variation of relative position and velocity with respect to each initial conditions is shown in Fig. 6. In this figure, the variations of relative states are presented and the value of terminal states are shown as the header of each plot. It shows that the best achieved solutions satisfy terminal conditions.

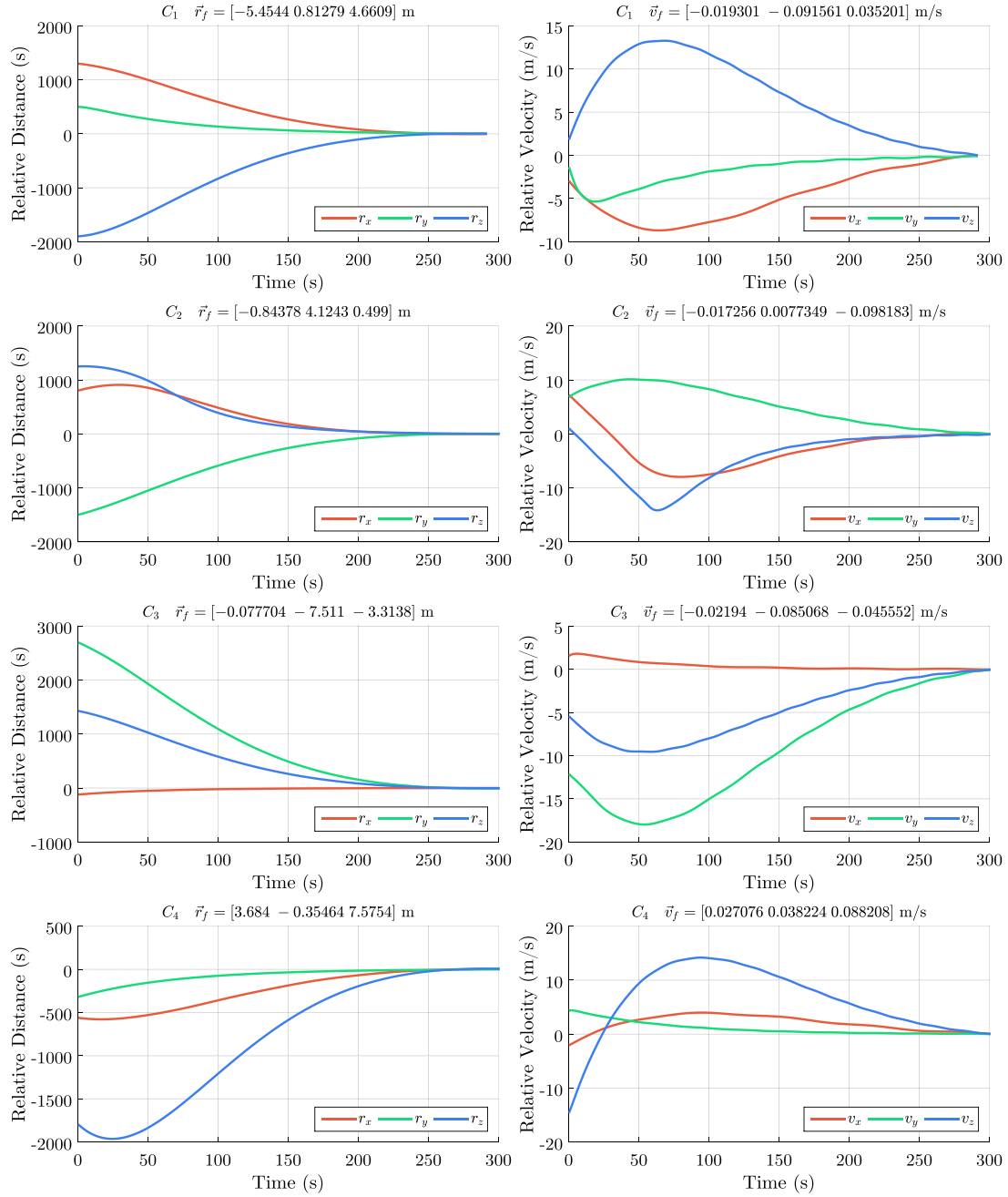
Fig. 7 shows the variation of spacecraft mass and the magnitude of thrust acted on the chaser spacecraft. As can be inferred, the spacecraft with initial condition  $C_4$  requires the highest quantity of propellant for reaching the target space vehicle with respect to maximum level of 122.3 N for thrust. On the other hand, assuming the initial condition of  $C_3$ , results the minimum required thrust level of 45.46 N. Moreover, analyzing the interpolation approaches gives important insights regarding the practicality of the employed polynomial schemes. All top solutions are associated with SP polynomial interpolation method. Best solutions



**Table 3**

High quality solutions (top ten) for fuel-optimal transfers in close-range space rendezvous.

$C_1$			$C_2$			$C_3$			$C_4$		
$m_f$	Map.	$N_p$	$m_f$	Map.	$N_p$	$m_f$	Map.	$N_p$	$m_f$	Map.	$N_p$
1.81272	BS	18	2.41731	BS	18	1.59422	BS	18	2.56913	BS	18
1.82008	BS	19	2.41991	BS	21	1.59428	BS	19	2.57063	LD	18
1.82061	BS	20	2.42470	BS	21	1.59774	BD	19	2.57300	BD	17
1.82092	LS	18	2.53698	BD	18	1.59804	BS	21	2.57338	LD	18
1.82363	BS	19	2.53884	BD	21	1.60652	LS	18	2.57483	BS	21
1.83340	BS	18	2.54913	LD	18	1.61041	BS	19	2.57642	LS	18
1.83557	LD	18	2.55159	BS	20	1.61129	BS	18	2.57704	BD	18
1.83562	BS	19	2.55721	BS	19	1.61427	BS	19	2.57757	LS	19
1.83741	BS	19	2.56299	LD	21	1.62225	BS	20	2.57796	BS	18
1.83764	BS	17	2.56483	LD	20	1.63085	BS	17	2.57864	LD	21

**Fig. 6.** Time histories of state vector for different initial conditions.

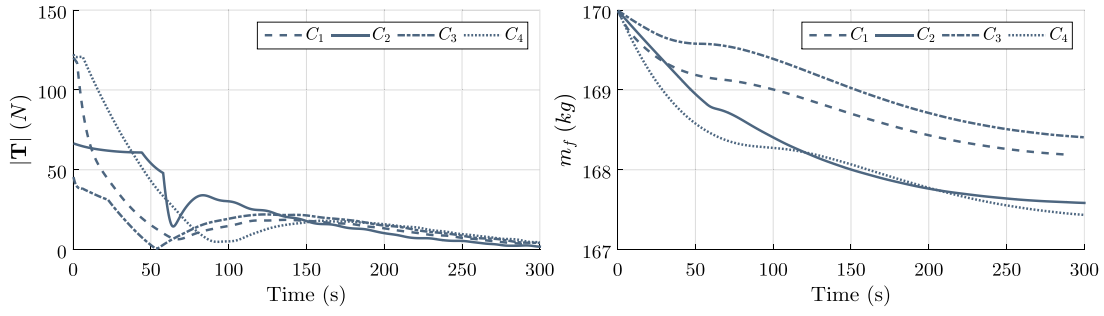


Fig. 7. Time histories of thrust magnitude  $|\vec{T}|$  and spacecraft overall mass  $m_f$ .

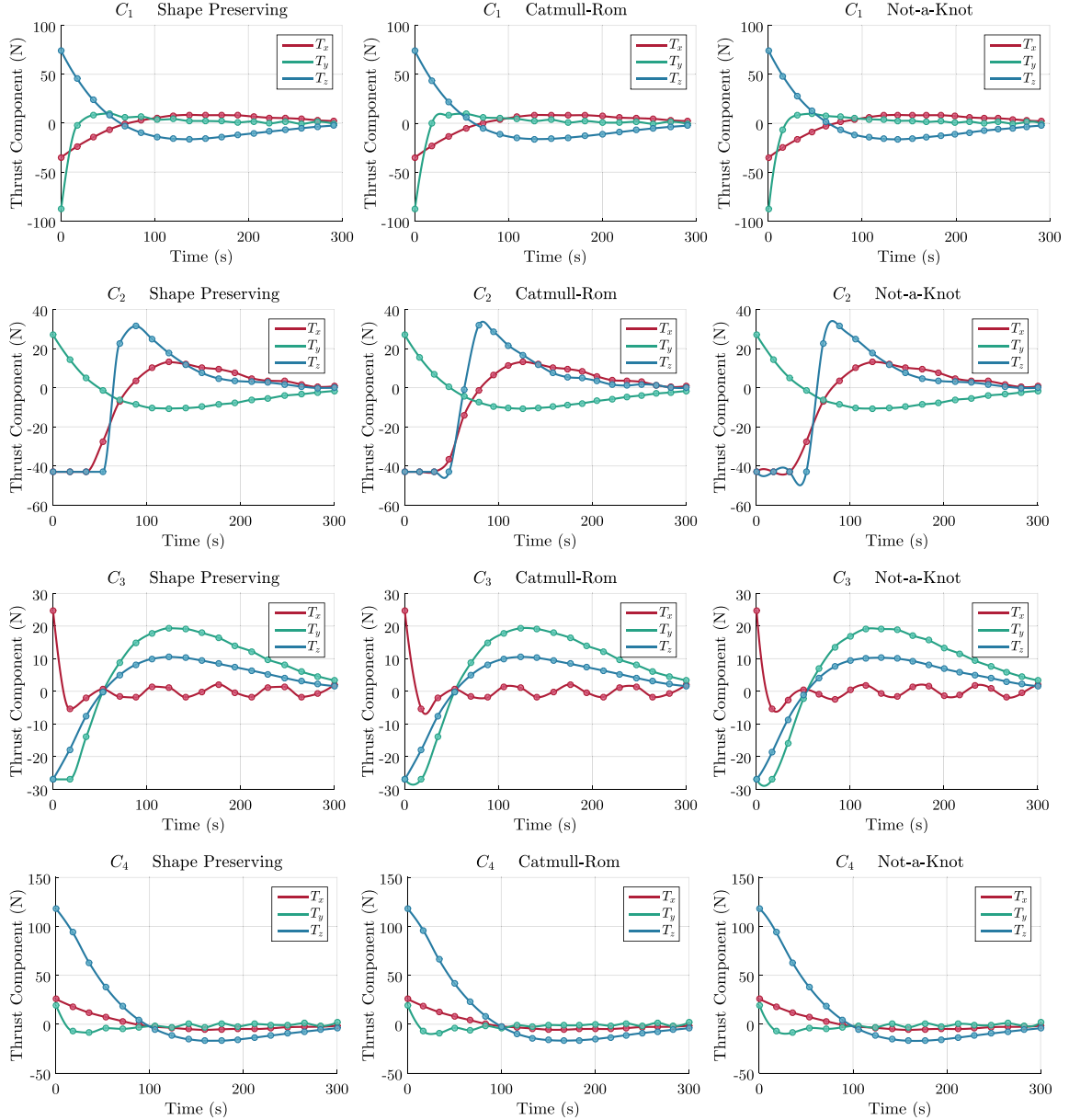


Fig. 8. Components of thrust vector.

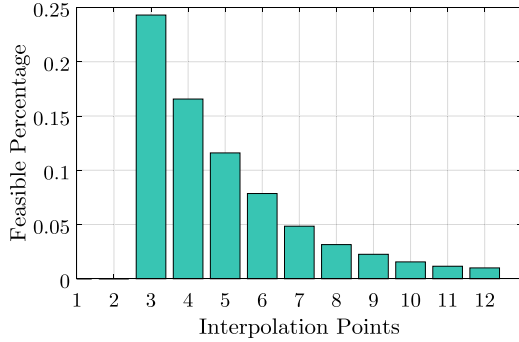
achieved via utilizing other techniques of interpolation are provided in Table 4.

Analyzing the results in Table 4 leads to the fact that SP interpolation method outperforms the rest of the polynomials in this problem. This is due to the fact that SP method benefits from less overshoot at

the interpolation points. In Fig. 8, the time histories of thrust vector components are plotted with respect to the solutions in Table 4. In this figure, rows represent results for each initial condition, while various columns correspond to a specific interpolation method. Analysis shows that since the best obtained solution by the optimizer is different for

**Table 4**Fuel mass ( $m_f$ ) corresponding to the best obtained solutions.

	$C_1$		$C_2$		$C_3$		$C_4$	
	$m_f$	$N_p$	$m_f$	$N_p$	$m_f$	$N_p$	$m_f$	$N_p$
Shape preserving	1.81272	18	2.41731	18	1.59422	18	2.56913	18
Catmull-Rom	1.84521	20	2.64096	18	1.65746	19	2.69233	18
Not-a-Knot	1.92706	17	2.92966	20	1.93098	18	3.05119	19

**Fig. 9.** Ratio of solution domain feasibility.

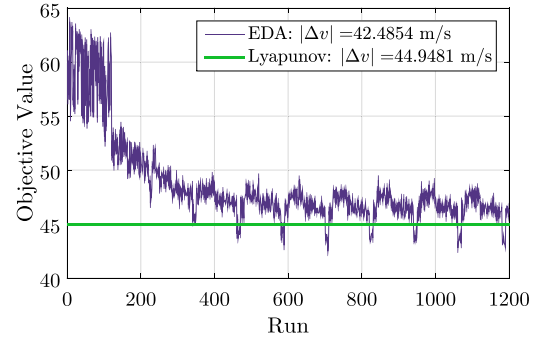
each interpolation method, the choice of the polynomial types has a significant impact on the solution domain. As SP interpolation benefits from minimum overshoot, the implementation of this approximation for thrust profile with the proposed method gives feasible solutions with higher quality.

#### 4.2. Comparative results

In the second experiment, the proposed approach is compared with a Lyapunov control method. Since the main aim is to conduct a fair experiment, the research in [Tian and Jia \(2017\)](#) was selected as the reference due to having similar mission settings, such as disturbances and elliptical space orbit for the target spacecraft. All problem setup, including orbital parameters, the space mission setup, and the orbital perturbations are implemented accordingly. The specific impulse is not reported in [Tian and Jia \(2017\)](#). Therefore, for comparing the performance of the methodologies, the total  $\Delta v$  is considered as the main metric in this experiment. The presented algorithm is employed considering the four proposed mapping mechanisms, three presented interpolation schemes, and ten nodes for thrust profile approximation as  $3 < N_p < 12$ . The algorithm is implemented and run for 10 separate times for each selection of problem setup. It should be noted that by choosing  $N_p$  to 1 or 2, the optimizer becomes unable to achieve initial feasible solutions, while the percentage of feasibility for the solution domain decreases when higher number of interpolation nodes are considered. [Fig. 9](#) illustrates the feasibility percentage of the solution domain for various interpolation nodes.

According to the results, when low number of interpolation nodes are considered for the thrust profile approximation, the feasibility percentage of the solution domain is high. It is a consequence of having lower decision variables ( $n = 3N_p$ ) in optimization. However, adding more nodes for interpolation increases the quality of the solutions achieved by the algorithm. The objective value for these solutions are shown in [Fig. 10](#) for different runs of optimization algorithm. The best solution obtained via the proposed algorithm has the number of nodes as  $N_p = 8$ , while having SP method for interpolation with LD mapping technique.

Based on the findings from the previous experiment, SP interpolation and BS mapping technique was the best choice of setup for optimization. Results from this experiment agrees with the SP interpolation outperforming other techniques. However, regarding the mapping

**Fig. 10.** Objective value of the best obtained solutions in different algorithm runs.

mechanism, the optimal technique is not the same, leading to conclude that choosing the proper mapping method is problem-dependent. In [Fig. 11](#), values of relative position and velocity of the chaser spacecraft in this experiment is provided.

The final states in the transfer trajectory are as  $\vec{r}_f = [8.6903 \ 4.1487 \ -1.6112] \text{ m}$  and  $\vec{v}_f = [0.058532 \ 0.066082 \ -0.046753] \text{ m/s}$ . Based on the observation on [Fig. 10](#), it can be verified that only a small number of runs of the algorithm leads to a solution with higher quality in comparison to Lyapunov control method. Nevertheless, the comparison between the quality of the best obtained solution using the presented method as  $\Delta v = 42.4854 \text{ m/s}$  with the solution from the Lyapunov control method as  $\Delta v = 44.9481 \text{ m/s}$  indicates the ability of the presented technique in achieving transfer trajectories with higher quality. The time profile of thrust vector is compared regarding the two obtained solutions in [Fig. 12](#). Based on the results, the proposed approach ended in a solution with larger maximum thrust value. However, the  $\Delta v$  yielding from the integration of the thrust profiles is lower than the other solution.

In addition to the comparative analysis of the obtained solutions, the performance of the proposed algorithm is evaluated regarding the selection of the interpolation methods and the mapping schemes. [Fig. 13](#) shows this evaluation.

In [Fig. 13](#), columns are assigned for interpolation methodologies, while rows are dedicated to the number of nodes for interpolation. The objective value of the best solution found between the obtained solutions in every cases are shown in the title of each box plot. Also, the quality of the obtained solutions with respect to each proposed mapping mechanism is provided in each box plot as statistical information. Once again it can be observed that SP spline has an excellent privilege in discovering solutions with better objective functions and satisfied feasibility conditions. In this regard, CR spline overpowers NK spline in most of the cases. Also, it is noteworthy that adding up the number of nodes equal to eight interpolation points enhances the quality of the final solutions. However, agreeing with the previous finding, dedicating more nodes forces the algorithm to achieve solutions with same quality but with more effort. Analysis of the mapping mechanisms also confirms that the optimal mapping mechanism is not unique when NK or CR splines are utilized. The reason is that these interpolation methods are not the most promising techniques in discovering the global optimal trajectories. However, the employment of SP splines leads to a significant advantage for LD mapping mechanism over the other mapping techniques in this problem. Following this observation along with the findings from the previous experiment, the optimal mapping mechanism is not unique for finding the best solution globally in every close-range space rendezvous mission. However, in a particular trajectory optimization problem, the ideal mapping mechanism will be unique for any number of interpolation nodes if SP spline is implemented in thrust profile approximation.

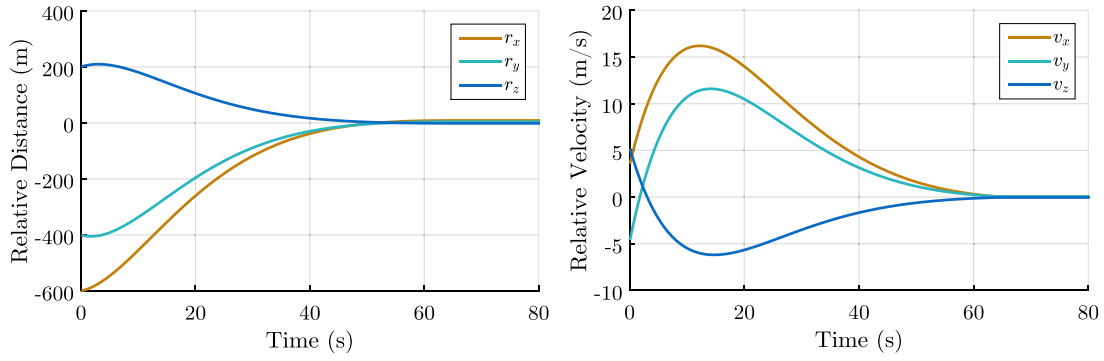


Fig. 11. Time histories of relative states ( $|\vec{r}_f| = 9.7637$  m,  $|\vec{v}_f| = 0.099$  m/s).

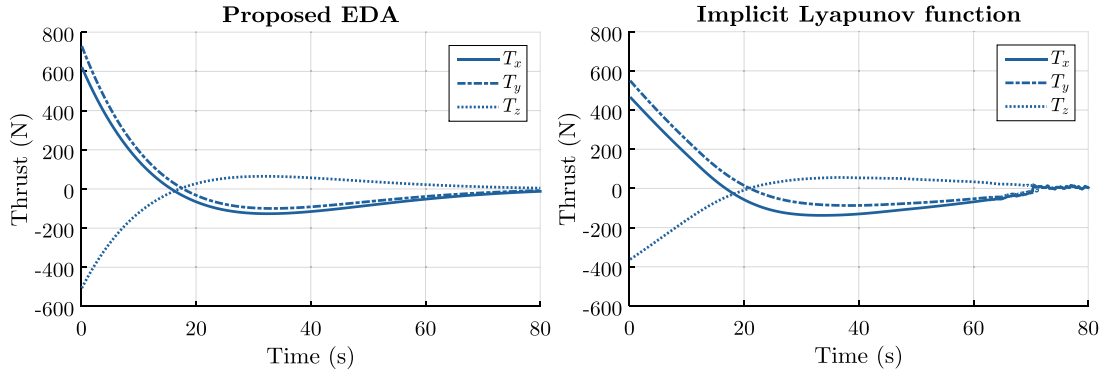


Fig. 12. Thrust profile comparison with the solution obtained via the Lyapunov function (Tian and Jia, 2017).

#### 4.3. Optimality and performance evaluation

Following the conducted experiments, the optimality of the obtained solutions is analyzed by comparing the quality of the achieved solutions with those that have been obtained via recently developed algorithms. It is noteworthy that there is no analytical proof for the obtained solutions to be global optimal due to the non-linearity of the system of equations and the high complexity of the optimization problem. However, when multiple problems are solved with several other algorithms in addition to the presented method, it is possible to indicate that the best obtained solutions belong to the proposed algorithm. Not only such an experiment verifies the relative optimality of the solutions, but also shows the robustness of the algorithm to the variety of other problems with different initial conditions. Also, the execution time of the algorithms has been taken into consideration in this comparison. Unlike the previous experiments in which the main aim was to explore the best combination of mapping mechanism and interpolation scheme, a fixed setup for the two aspects are considered for the proposed algorithm in this experiment. This is due to the fact that the main aim here is to have a fair comparison between different algorithms, hence no variety of algorithm parameters are tried.

In the current experiment, 50 close-range space rendezvous problems are defined. The problem parameters and the initial conditions are considered as random values with uniform distribution in the boundaries of  $6600 \text{ km} \leq a \leq 42000 \text{ km}$ ,  $0 \leq e \leq 0.6$  (subject to have perigee radius higher than  $R_E$ ),  $0^\circ \leq i \leq 180^\circ$ ,  $0^\circ \leq \Omega \leq 360^\circ$ ,  $0^\circ \leq \omega \leq 360^\circ$ ,  $0^\circ \leq \nu \leq 360^\circ$ ,  $0 \leq s \leq t_f \leq 600 \text{ s}$ ,  $150 \text{ kg} \leq m_0 \leq 300 \text{ kg}$ ,  $100 \text{ N} \leq |T| \leq 150 \text{ N}$ ,  $280 \text{ s} \leq I_{sp} \leq 350 \text{ s}$ ,  $5 \text{ m} \leq \sigma_r \leq 20 \text{ m}$ ,  $0.05 \text{ m/s} \leq \sigma_v \leq 0.2 \text{ m/s}$ ,  $1.8 \leq C_D \leq 2.2$ ,  $2.5 \text{ m}^2 \leq A \leq 3.5 \text{ m}^2$ ,  $-3000 \text{ m} \leq x_i, y_i, z_i \leq +3000 \text{ m}$ ,  $-20 \text{ m/s} \leq \dot{x}_i, \dot{y}_i, \dot{z}_i \leq +20 \text{ m/s}$ . For each scenario, the proposed algorithm is utilized for solving the problem along with several other algorithms suited for constrained continuous optimization, including BP-eMag-ES (Hellwig and Beyer, 2020), CORCO (Wang et al., 2019), EnMODE (Sallam et al., 2020),

and COLSHADE (Gurrola-Ramos et al., 2020). Default parameters are chosen for each algorithm as described in their respective reference. For the proposed algorithm, linear deterministic mapping with  $N_\delta = 10$  is considered. Also, all problems are formed with respect to SP interpolation method with  $N_p = 20$ . In order to have a fair comparison, same allowable budget of function evaluation is considered for all algorithms while tackling each unique problem as mentioned at the beginning of this section. Each algorithm is run 10 times to solve every problem, while the initial population varies in each run.

All obtained solutions are stored along with the execution time of each run. Having the obtained solutions, the objectives values are scaled within the interval of  $[0 \ 1]$ , with the lower bound as the best achieved solution and the upper bound as the worst achieved solution between all runs either in terms of the objective value or the constraint violation. Having the scaled scores for all runs of each algorithm, the performance of the algorithms in terms of optimality can be compared as in Fig. 14.

In Fig. 14, the distribution of scaled scores for optimality of the obtained solutions are depicted, separated for each algorithm. As can be observed, the mean value of the scaled scores for the improved EDA is lower than the other methods, leading to conclude that the obtained solutions via the presented algorithm have higher qualities in comparison to the rest of the algorithms. In this regard, CORCO has shown to be the most stable algorithm in terms of giving solutions with same qualities. COLSHADE on the other hand has shown to provide solutions with wide range of qualities with less probability of giving global optimal solutions. The most competitive algorithm is BP-eMag-ES, which managed to find high quality solutions in several runs. However, the mean value of the scores are slightly higher than the improved EDA. Similar to the quality of the obtained solutions, the execution time of the algorithms are also scaled and illustrated in Fig. 15.

In Fig. 15, the scaled execution times versus the scaled objective values for each run are plotted, separated for each algorithm. Results



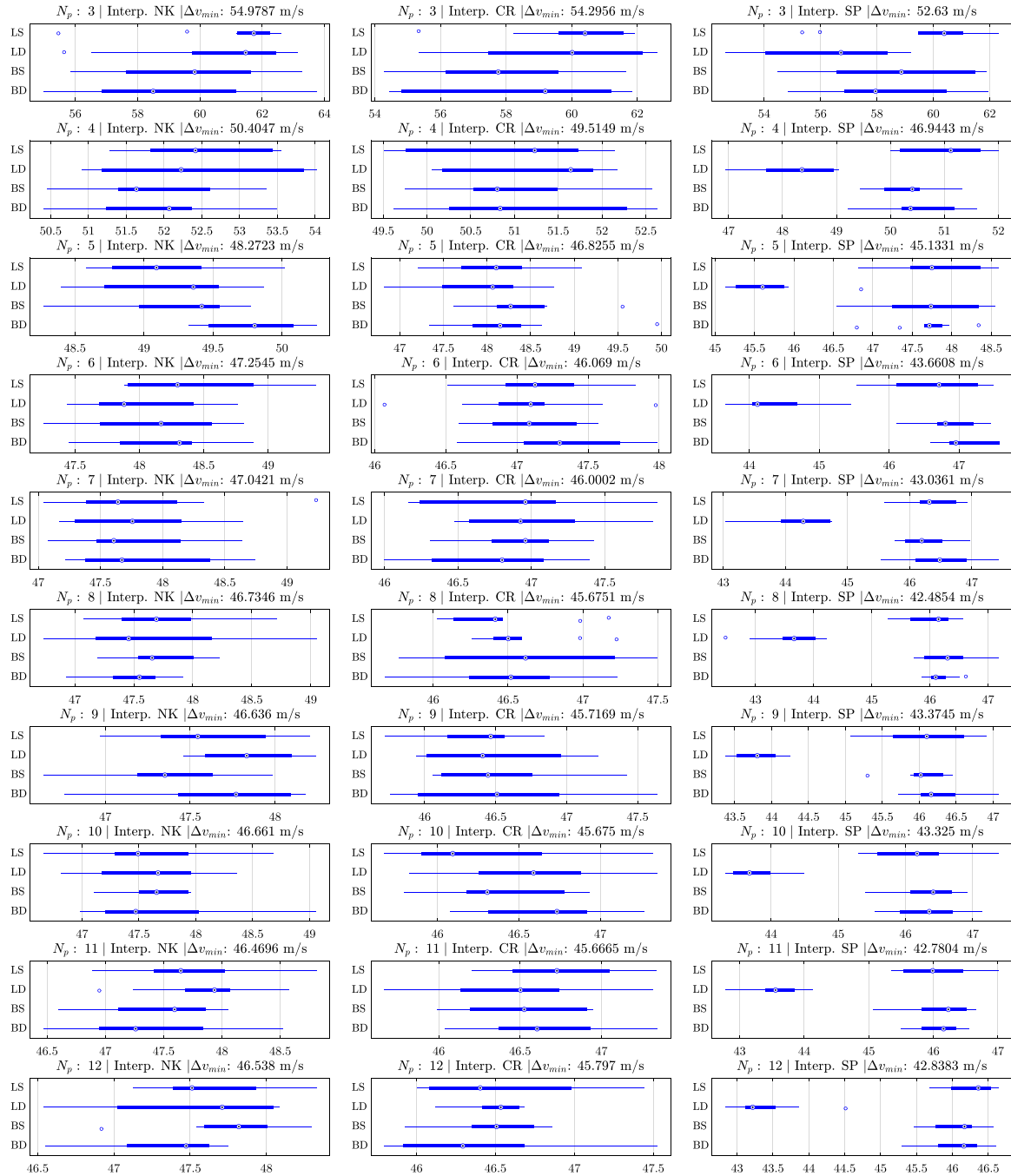


Fig. 13. Comparative analysis of the proposed mapping mechanisms with respect to interpolation methods.

indicate that the improved EDA has reasonable execution time in comparison to other algorithms considering the quality of the achieved solutions. COLSHADE is the most reliable algorithm in terms of execution time with slight advantage over the improved EDA, since the average execution time is less than the one associated with the proposed algorithm. On the other hand, BP-eMAG-ES, which was the most competitive algorithm in terms of optimality, is shown to be more time consuming than the improved EDA. Results also confirm that EnMODE has the most stable execution time in comparison to other algorithms.

## 5. Concluding remarks

In this paper, the problem of finding optimal transfer trajectory for close-range space rendezvous proximity operation is taken into account.

In a direct approach, the thrust components are approximated with different schemes of Hermite interpolation method. The trajectory optimization problem in close-range space rendezvous mission is converted into a black box optimization problem, associated with some non-linear constraints. Then, the problem is tackled with a newly developed algorithm within the framework of EDAs. The algorithm benefits from several feasibility conserving techniques, that have been utilized to satisfy terminal conditions of the space rendezvous operation. These techniques oblige the algorithm to return only feasible solutions while minimizing the objective function.

Since the main aim was to develop an approach for close-range space rendezvous between any types of trajectories with no specific assumptions (ex. co-planar or circular orbits), general formulation of the spacecraft dynamics for close-range rendezvous was considered.

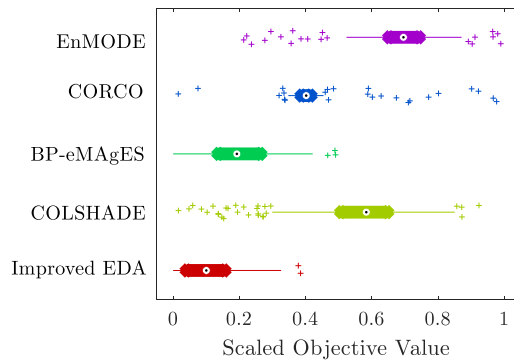


Fig. 14. Performance score distribution of algorithms.

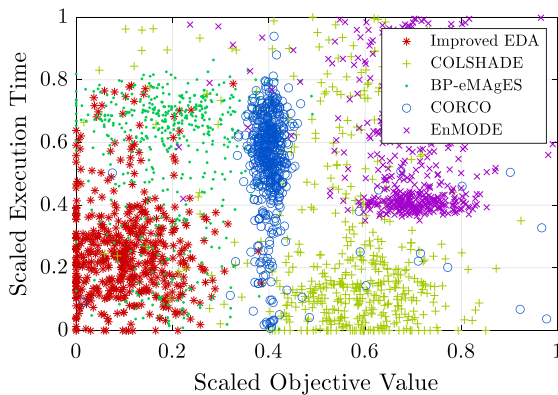


Fig. 15. Execution time of the algorithms.

The employed model is a non-linear model, where any types of disturbances could be implemented. Two types of orbital perturbations were considered as the disturbances including atmosphere drag and  $J_2$  effect. Depending on the varieties of space missions, various disturbances may be considered in other problems for the sake of increasing the simulation accuracy of the space trajectory.

Three different interpolation schemes were implemented for thrust profile approximation in the experiments. Following the obtained result out of trying all possible combinations of interpolation methods and the mapping mechanisms in different scenarios and initial conditions, it has been observed that the most promising solutions are associated with SP spline interpolation scheme, regardless of the scenario or the initial condition. However, the best choice for the mapping mechanism is found to be problem-dependent. In the other word, regardless of the interpolation scheme, a unique technique of mapping exists, which makes the approach performs slightly better in finding solutions with high quality. The best choice of the mapping mechanism varies for different space rendezvous missions.

Results also show that the presented method is effective and robust to various initial conditions and parameters describing space mission characteristics. Also, the proposed method discovers solutions with better quality compared to the method by implicit Lyapunov function. Following the empirical experiments, it has been observed that the quality of the obtained solutions via the proposed algorithm is higher than other state-of-the-art constrained continuous optimization algorithms. In terms of the execution time, the proposed algorithm has shown to be competitive in comparison to other constrained continuous optimization algorithms. Considering the quality of the obtained solutions, the algorithm has shown to be reliable as it has the best performance in exchange for the execution time. Satisfaction of other types of constraints like path constraint on state variables (e.g., approach corridor), saturation constraints on control along with enhancement

towards the performance of the proposed mechanisms, and the consideration of uncertainties will be considered in future research. Also, utilization of the proposed algorithm in tackling other challenging continuous optimization problems equipped with constraints, and detailed analysis of the algorithm parameters on the quality of the obtained solutions are potential subjects for further research.

#### CRedit authorship contribution statement

**Abolfazl Shirazi:** Conceptualization, Methodology, Visualization. **Josu Ceberio:** Validation, Supervision. **Jose A. Lozano:** Project administration, Funding acquisition, Supervision.

#### Declaration of competing interest

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

#### Data availability

No data was used for the research described in the article.

#### Acknowledgments

This research is supported by La Caixa Foundation Fellowship, the Basque Government, Spain through the BERC 2022–2025, Elkarteek programs (project code KK-2021/00065, KK-2022/00106), Spanish Ministry of Economy and Competitiveness MINECO: BCAM Severo Ochoa excellence accreditation SEV-2017-0718, Spanish Ministry of Science (project codes: PID2019-104933GB-10/AEI/10.13039/501100011033, PID2019-106453GA-I00/AEI/10.13039/501100011033), Basque Government consolidated groups, Spain 2022–2025 (code IT1504-22).

#### References

- Alfriend, K., Vadali, S.R., Gurfil, P., How, J., Breger, L., 2009. *Spacecraft Formation Flying: Dynamics, Control and Navigation*, Vol. 2. Elsevier.
- Ayyanathan, P.J., Taheri, E., 2022. Mapped adjoint control transformation method for low-thrust trajectory design. *Acta Astronaut.* 193, 418–431.
- Behforooz, G., 1992. The not-a-knot piecewise interpolatory cubic polynomial. *Appl. Math. Comput.* 52 (1), 29–35.
- Chai, R., Savvaris, A., Tsourdos, A., Chai, S., Xia, Y., 2018. Unified multiobjective optimization scheme for aeroassisted vehicle trajectory planning. *J. Guid. Control Dyn.* 41 (7), 1521–1530.
- Chai, R., Tsourdos, A., Savvaris, A., Chai, S., Xia, Y., 2021. Solving constrained trajectory planning problems using biased particle swarm optimization. *IEEE Trans. Aerosp. Electron. Syst.* 57 (3), 1685–1701.
- Coello, C.A.C., 2002. Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art. *Comput. Methods Appl. Mech. Engrg.* 191 (11–12), 1245–1287.
- Dahlquist, G., Björck, Å., 2008. *Numerical Methods in Scientific Computing*, Vol. I. SIAM.
- Dahmen, W., Gasca, M., Micchelli, C.A., 2012. *Computation of Curves and Surfaces*, Vol. 307. Springer Science & Business Media.
- Fossà, A., Bettanini, C., 2020. Optimal rendezvous trajectory between sample return orbiter and orbiting sample container in a mars sample return mission. *Acta Astronaut.* 171, 31–41.
- Gurrola-Ramos, J., Hernández-Aguirre, A., Dalmau-Cedeño, O., 2020. COLSHADE for real-world single-objective constrained optimization problems. In: 2020 IEEE Congress on Evolutionary Computation (CEC). IEEE, pp. 1–8.
- Hellwig, M., Beyer, H.-G., 2020. A modified matrix adaptation evolution strategy with restarts for constrained real-world problems. In: 2020 IEEE Congress on Evolutionary Computation (CEC). IEEE, pp. 1–8.
- Hodge, V., Austin, J., 2004. A survey of outlier detection methodologies. *Artif. Intell. Rev.* 22 (2), 85–126.
- Huynh, H.T., 1993. Accurate monotone cubic interpolation. *SIAM J. Numer. Anal.* 30 (1), 57–100.
- Larrañaga, P., Lozano, J.A., 2001. *Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation*. Springer.
- Liu, G., Feng, W., Yang, K., Zhao, J., 2019. Hybrid QPSO and SQP algorithm with homotopy method for optimal control of rapid cooperative rendezvous. *J. Aerosp. Eng.* 32 (4), 04019030.

- Malan, K.M., Oberholzer, J.F., Engelbrecht, A.P., 2015. Characterising constrained continuous optimisation problems. In: 2015 IEEE Congress on Evolutionary Computation (CEC). IEEE, pp. 1351–1358.
- Ostman, J.A., 2019. Cislunar Trajectory Generation with Sun-Exclusion Zone Constraints Using a Genetic Algorithm and Direct Method Hybridization. Tech. Rep., Air force institute of technology - Wright-Patterson AFB.
- Phillips, G.M., 2003. Interpolation and Approximation by Polynomials, Vol. 14. Springer Science & Business Media.
- Ramteke, V., Kumar, S.R., 2022. Halo orbit maintenance around L 1 point of the sun-earth system using optimal control and Lyapunov stability theory. *J. Aerosp. Eng.* 35 (1), 04021107.
- Sallam, K.M., Elsayed, S.M., Chakraborty, R.K., Ryan, M.J., 2020. Improved multi-operator differential evolution algorithm for solving unconstrained problems. In: 2020 IEEE Congress on Evolutionary Computation (CEC). IEEE, pp. 1–8.
- Shirazi, A., 2021. Efficient Meta-Heuristics for Spacecraft Trajectory Optimization (Ph.D. thesis). Universidad del País Vasco-Euskal Herriko Unibertsitatea.
- Shirazi, A., Ceberio, J., Lozano, J.A., 2018. Spacecraft trajectory optimization: A review of models, objectives, approaches and solutions. *Prog. Aerosp. Sci.* 102, 76–98.
- Shirazi, A., Ceberio, J., Lozano, J.A., 2022. EDA++: Estimation of distribution algorithms with feasibility conserving mechanisms for constrained continuous optimization. *IEEE Trans. Evol. Comput.*
- Sullivan, J., Grimberg, S., D'Amico, S., 2017. Comprehensive survey and assessment of spacecraft relative motion dynamics models. *J. Guid. Control Dyn.* 40 (8), 1837–1859.
- Tang, J., Liu, G., Pan, Q., 2021. A review on representative swarm intelligence algorithms for solving optimization problems: Applications and trends. *IEEE/CAA J. Autom. Sin.* 8 (10), 1627–1643.
- Tewari, A., 2007. *Atmospheric and Space Flight Dynamics*. Springer.
- Tian, X., Jia, Y., 2017. Analytical solutions to the matrix inequalities in the robust control scheme based on implicit Lyapunov function for spacecraft rendezvous on elliptical orbit. *IET Control Theory Appl.* 11 (12), 1983–1991.
- Vijayakumar, M., Abdelkhalik, O., 2022. Shape-based approach for low-thrust earth-moon trajectories initial design. *J. Guid. Control Dyn.* 45 (1), 103–120.
- Wang, Y., Li, J.-P., Xue, X., Wang, B.-C., 2019. Utilizing the correlation between constraints and objective function for constrained evolutionary optimization. *IEEE Trans. Evol. Comput.* 24 (1), 29–43.
- Wu, J., 2012. *Advances in K-Means Clustering: A Data Mining Thinking*. Springer Science & Business Media.
- Zhang, J., Biggs, J.D., Ye, D., Sun, Z., 2020. Finite-time attitude set-point tracking for thrust-vectoring spacecraft rendezvous. *Aerosp. Sci. Technol.* 96, 105588.
- Zhu, X., Chen, J., Zhang, C., Qiao, B., 2020. Optimal fuel station arrangement for multiple GEO spacecraft refueling mission. *Adv. Space Res.* (ISSN: 0273-1177) 66 (8), 1924–1936.
- Zotes, F.A., Peñas, M.S., 2012. Particle swarm optimisation of interplanetary trajectories from Earth to Jupiter and Saturn. *Eng. Appl. Artif. Intell.* 25 (1), 189–199.
- Zuo, M., Dai, G., Peng, L., Wang, M., Liu, Z., Chen, C., 2020. A case learning-based differential evolution algorithm for global optimization of interplanetary trajectory design. *Appl. Soft Comput.* 94, 106451.