

Minimum-Fuel Low-Thrust Trajectory Optimization via a Direct Adaptive Evolutionary Approach

Abolfazl Shirazi

Basque Center for Applied Mathematics BCAM, Bilbao, Spain, 48009

Abstract— Space missions with low-thrust propulsion systems are of appreciable interest to space agencies because of their practicality due to higher specific impulses. This research proposes a technique to the solution of minimum-fuel non-coplanar orbit transfer problem. A direct adaptive method via Fitness Landscape Analysis (FLA) is coupled with a constrained evolutionary technique to explore the solution space for designing low-thrust orbit transfer trajectories. Taking advantage of the solution for multi-impulse orbit transfer problem, and parameterization of thrust vector, the orbital maneuver is transformed into a constrained continuous optimization problem. A constrained Estimation of Distribution Algorithms (EDA) is utilized to discover optimal transfer trajectories, while maintaining feasibility of the solutions. The low-thrust trajectory optimization problem is characterized via three parameters, referred to as problem identifiers, and the dispersion metric is utilized for analyzing the complexity of the solution domain. Two adaptive operators including the kernel density and outlier detection distance threshold within the framework of the employed EDA are developed, which work based on the landscape feature of the orbit transfer problem. Simulations are proposed to validate the efficacy of the proposed methodology in comparison to the non-adaptive approach. Results indicate that the adaptive approach possesses more feasibility ratio and higher optimality of the obtained solutions.

Index Terms—Orbit Transfer, Trajectory Optimization, Fitness Landscape Analysis, Dispersion, Estimation of Distribution Algorithms

I. INTRODUCTION

CONSTRAINED trajectory optimization has been a critical component in the development of advanced guidance and control systems. The distinct specific impulse

This research is supported by the Basque Government through the BERC 2022-2025 program, the Ministry of Science, Innovation and Universities: BCAM Severo Ochoa accreditation SEV-2017-0718, BEAZ Bizkaia 3/12/DP/2021/00150, and SPRI Group through Ekintzaile Program EK-00112-2021. (Corresponding author: Abolfazl Shirazi)

Abolfazl Shirazi is with Basque Center for Applied Mathematics BCAM, Bilbao, Spain, 48009 (e-mail: ashirazi@bcamath.org).

of low-thrust propulsion systems comes with significant savings of propellant, while enables new concepts for space missions. However, it needs novel trajectory design methodologies as the thrusting periods increase and therefore, the assumption of instantaneous impulses does not hold anymore. In recent years, considerable effort has been dedicated to the development of best techniques and algorithms for spacecraft trajectory optimization. The relevant papers can be traced back to the pioneering works by Miele [1], Vinh [2], and Prussing [3]. It has been shown, that in the finite-thrust solutions of fuel-optimal transfers, the control structure and the optimal number of thrusting arcs is usually unknown a priori. Following the nonlinear nature of dynamics and constraints, the orbital transfer problem becomes considerably difficult to deal with.

Techniques for addressing the challenging issues in low-thrust trajectory optimization have been investigated for many years. All proposed techniques in the literature can be classified as either direct or indirect procedures [4]. Direct approaches are based on the conversion of the optimal control problem into an optimization problem through discretization of states and control variables [5], [6], [7]. Indirect methods on the other hand, utilize the necessary conditions for optimality based on the calculus of variations [8], [9], [10]. Following the employed approach, the orbit transfer problem generally turns into a nonlinear programming (NLP) problem. There are several strategies to solve the resulting NLP problem, ranging from single and multiple shooting to direct collocation. In these approaches, the number of discrete nodes is increased by variety of factors such as the dimensions of states, as well as the flight time. This increment along with the nonlinearity of system dynamics results in large-scale NLP problems, which is challenging to deal with. Because of these challenges, discovery of optimal space transfers utilizing Evolutionary Algorithms (EAs) is becoming increasingly popular in celestial mechanics [11]. The effectiveness of heuristic procedures is evolved from the likelihood of converging to the globally minimizing solutions of qualitatively various problems, and extensive numerical tests are to be performed for evaluating the performance of any novel algorithm. Regarding the development and utilization of EAs in spacecraft trajectory optimization, noticeable advances can be identified in recent efforts. For instance, in their work, Zuo et al. [12] developed a case learning-based Differential Evolution (DE) algorithm to tackle multiple gravity assists trajectory design and optimization, in which several global and local optimizers are developed based on an enhanced version of DE. In their work, Feng et al. [13] presented an indirect evolutionary methodology to tackle Earth-orbiting satellite rendezvous. In this research, a hybrid algorithm consisting of Particle Swarm Optimization (PSO) and DE is proposed for far-distance cooperative rendezvous. In their work, Shannon et al. [14] introduced Q-Law methodology combined with Genetic Algorithm (GA) to tackle Earth orbiting spacecraft trajectory optimization

problems, in which a heuristic algorithm for gain tuning of Q-Law method is applied to continuous thrust trajectory optimization. Another self-adaptive/self-learning DE algorithm is introduced by Choi et al. in [15] for trajectory optimization of deep-space missions. In this research, an adaptive mechanism and a re-initialization method incorporated with DE is introduced for multiple gravity assists trajectory design and optimization. In their work, Jimenez-Lluva and Root [16] proposed DE combined with a hybrid technique to tackle GTO to GEO transfers considering orbital perturbations. In this research, a hybrid scheme with DE is developed for minimum-time and minimum-fuel orbital transfers. Overall, in many cases, finding an adequate set of tuning parameters for EAs can be as time-consuming as the optimization process itself. Also, it can be observed that in the vast majority of research from the literature, when it comes to find unknown parameters, usually either a novel EA is developed, or an arbitrary EA is chosen and utilized for obtaining the desired solution, i.e. for achieving the optimal transfer trajectory. However, no research has been dedicated to explore the reason why a particular EA outperforms other rival algorithms in the constructed spacecraft trajectory optimization problem, or how efficient the employed EA is in finding the desired solution. In particular, there is a lack of correlation between the selection of the EA or the choice of EA parameters and the inherent complexity of the spacecraft trajectory optimization problem. These insights are the main motivation in the current article, and the main aim of this research is to analyze the complexity of orbit transfer problems, and develop an effective adaptive method for space trajectory optimization. This concept, better known as Fitness Landscape Analysis (FLA), which is connected to auto-tuning and developing intelligent algorithms for complex systems, has not been given proper attention in astrodynamics. The main purpose of this research is to initially fill the gap on this matter in spacecraft trajectory optimization and open the door for further attempts on employing these techniques in various approaches that are dedicated to find optimal trajectories for space systems.

The article is arranged as follows. Section II describes the orbit transfer problem in detail. The spacecraft dynamics is modeled via the classical orbital elements and the minimum-fuel low-thrust orbit transfer problem is turned into a constrained continuous optimization problem via Fourier transformation of input variables. Section III introduces the solution procedure, where the objectives and constraints are defined initially. A recently developed EA based on Estimation of Distribution Algorithms (EDAs) is introduced, along with two of its parameters that control the exploration and exploitation capabilities of the algorithm. The complexity of the problem according to mission parameters is analyzed via FLA techniques, and two adaptive operators are developed for the aforementioned algorithm parameters. In Section IV, the proposed operators are utilized as an adaptive evolutionary approach for low-thrust trajectory optimization.

The application of the proposed strategy to several cases followed by demonstration of its effectiveness are presented along with the added value of the optimality for the solutions. Discussions are provided in Section V and Section VI concludes the study.

II. MATHEMATICAL MODELING

A. Equation of Motion

The evolution of an orbit due to all accelerations except point-mass gravity can be described via the perturbed Kepler problem. The classical orbital elements conveying the key components of the solution are semimajor axis, eccentricity, inclination, right ascension of the ascending node, argument of periapsis, and true anomaly, which can be represented as $[a, e, i, \Omega, \omega, \theta]$. Let ρ_R , ρ_T , and ρ_N , be the radial, transverse, and normal (RTN) components of the acceleration, expressed in the radial-transverse-normal frame centered at the satellite. Whenever non-conservative perturbations such as thrusting are acted on the spacecraft, it is customary to describe the evolution of the classical elements through Gauss's variational equations as

$$\begin{aligned}
 \dot{a} &= \frac{2a^2}{\sqrt{\mu a(1-e^2)}} \left[e \sin(\theta) \rho_R + (1+e \cos(\theta)) \rho_T \right] \\
 \dot{e} &= \sqrt{\frac{a(1-e^2)}{\mu}} \left[\sin(\theta) \rho_R + \frac{e + (2+e \cos \theta) \cos \theta}{1+e \cos(\theta)} \rho_T \right] \\
 \dot{i} &= \sqrt{\frac{a(1-e^2)}{\mu}} \left[\frac{\cos(\theta + \omega)}{1+e \cos(\theta)} \rho_N \right] \\
 \dot{\Omega} &= \sqrt{\frac{a(1-e^2)}{\mu}} \left[\frac{\sin(\theta + \omega)}{(1+e \cos(\theta)) \sin(i)} \rho_N \right] \\
 \dot{\omega} &= \sqrt{\frac{a(1-e^2)}{\mu}} \left[-\frac{\cos(\theta)}{e} \rho_R + \frac{(2+e \cos(\theta)) \sin(\theta)}{e(1+e \cos(\theta))} \rho_T \right. \\
 &\quad \left. - \frac{\sin(\theta + \omega) \cot(i)}{1+e \cos(\theta)} \rho_N \right] \\
 \dot{\theta} &= \sqrt{\frac{\mu}{a^3} \frac{(1+e \cos(\theta))^2}{(1-e^2)^{3/2}}} + \sqrt{\frac{a(1-e^2)}{\mu}} \left[\frac{\cos(\theta)}{e} \rho_R \right. \\
 &\quad \left. - \frac{(2+e \cos(\theta)) \sin(\theta)}{e(1+e \cos(\theta))} \rho_T \right]
 \end{aligned} \tag{1}$$

where μ denotes the gravitational parameter, and the over-dot symbol represents the time derivative. This element set has been used in many research dedicated to direct trajectory optimization [17]. Note that vectors in RTN frame and Earth Centered Inertial (ECI) frame can be converted to each other via the rotation matrix based on unit vectors in the radial, tangential and normal directions [18]. The variation of spacecraft mass during the orbit transfer can be represented by

$$\dot{m} = -\frac{\|\mathbf{T}\|}{I_{sp} g_0} \tag{2}$$

where I_{sp} is the specific impulse, $\|\mathbf{T}\|$ is the magnitude of the acting thrust vector \mathbf{T} , and g_0 is the acceleration due to gravity at sea level. Following the presented model for system dynamics, the unknown variables can be identified. Given the orbital elements for the initial and desired orbits denoted as $[a_{\mathcal{I}} e_{\mathcal{I}} i_{\mathcal{I}} \Omega_{\mathcal{I}} \omega_{\mathcal{I}}]$ and $[a_{\mathcal{D}} e_{\mathcal{D}} i_{\mathcal{D}} \Omega_{\mathcal{D}} \omega_{\mathcal{D}}]$ respectively, alongside the spacecraft's initial mass $m_{\mathcal{I}}$, specific impulse I_{sp} , and maximum available thrust level T_{\max} , a distinct optimization problem arises concerning the non-coplanar orbital maneuver. The mission parameters describing a unique problem can be represented by the vector \mathcal{P} as

$$\mathcal{P} = [a_{\mathcal{I}} e_{\mathcal{I}} i_{\mathcal{I}} \Omega_{\mathcal{I}} \omega_{\mathcal{I}} a_{\mathcal{D}} e_{\mathcal{D}} i_{\mathcal{D}} \Omega_{\mathcal{D}} \omega_{\mathcal{D}} m_{\mathcal{I}} T_{\max} I_{sp}] \quad (3)$$

Following the presented model, each unique space mission can be defined with the vector of known parameters \mathcal{P} in Eq. 3. The aim is to find the optimal time intervals of the acting thrust vector \mathbf{T} on the spacecraft and optimal time-profile of the thrust direction angles associated with each time interval. The acting thrust results the acceleration vector $\rho = [\rho_R, \rho_T, \rho_N]$ in Eq. 1. Integrating the dynamics equation gives the final orbit of the spacecraft. The objective is to find the optimal thrust vector that transfers the spacecraft to the desired orbit with respect to the initial and final conditions, while minimizing the fuel consumption with respect to Eq. 2.

B. The Discrete Fourier Transform

Finding the ideal transfer trajectory for the intended problem involves determining the undisclosed thrust profiles and their corresponding on-off timings during the transfer. These unidentified functions and variables can be expressed in \mathcal{X} as

$$\mathcal{X} = \left\{ \begin{array}{ccc} t_1^{\mathcal{I}} & \mathbf{T}_1 & t_1^{\mathcal{F}} \\ t_2^{\mathcal{I}} & \mathbf{T}_2 & t_2^{\mathcal{F}} \\ \dots & \dots & \dots \\ t_i^{\mathcal{I}} & \mathbf{T}_i & t_i^{\mathcal{F}} \\ \dots & \dots & \dots \\ t_{N_T-1}^{\mathcal{I}} & \mathbf{T}_{N_T-1} & t_{N_T-1}^{\mathcal{F}} \\ t_{N_T-1}^{\mathcal{I}} & \mathbf{T}_{N_T} & t_{N_T}^{\mathcal{F}} \end{array} \right\} \quad (4)$$

where N_T is the number of thrust arcs, i is the counter for the number of thrust arcs, $t_i^{\mathcal{I}}$ and $t_i^{\mathcal{F}}$ ($1 \leq i \leq N_T$) are the starting time and ending time of thrust arcs respectively, and \mathbf{T}_i are thrust profiles as functions of time in each respective time interval of $t_i^{\mathcal{I}} < t < t_i^{\mathcal{F}}$. This representation agrees with minimum-fuel transfers, since the thrust magnitude is at maximum value within the thrust arcs as $\|\mathbf{T}_i\| = T_{\max}$ for $t_i^{\mathcal{I}} < t < t_i^{\mathcal{F}}$, and is equal to zero within the coast arcs as $\|\mathbf{T}_i\| = 0$ for $t_i^{\mathcal{F}} < t < t_{i+1}^{\mathcal{I}}$ ($i \neq N_T$). It can be justified that for N_T number of thrust arcs, there will be $N_T - 1$ number of coast arcs. The components of thrust vector within the thrust arcs can be defined as

$$\mathbf{T}_i(t) = T_{\max} \begin{bmatrix} \cos \alpha_i(t) \cos \beta_i(t) \\ \cos \alpha_i(t) \sin \beta_i(t) \\ \sin \alpha_i(t) \end{bmatrix} \quad (5)$$

with α_i and β_i denoting the space vehicle's steering angles relative to the reference frame. From the given definition, it is evident that the optimal time evolution of steering angles, as well as the on-off time intervals, are presently unknown and yet to be established. To account for the range of steering angles as $-\pi/2 < \alpha_i(t) < \pi/2$ and $-\pi < \beta_i(t) < \pi$, the variations are approximated as time-profiles through a finite number of nodes for each distinct thrust arc, as outlined below.

$$[\alpha(t), \beta(t)] = \mathcal{A}(\hat{\alpha}_1, \hat{\alpha}_2, \dots, \hat{\alpha}_{N_p}, \hat{\beta}_1, \hat{\beta}_2, \dots, \hat{\beta}_{N_p}) \quad (6)$$

where $\mathcal{A}(\cdot)$ represents the conversion operator, which transforms the given approximation points $\hat{\alpha}_j, \hat{\beta}_j$ into continuous time-series. With respect to the fact that $-\pi/2 < \hat{\alpha}_j < \pi/2$, $-\pi < \hat{\beta}_j < \pi$ ($j = 1, \dots, N_p$), different schemes may be employed for this operator. In this research, Fourier transformation is utilized in $\mathcal{A}(\cdot)$ to parameterized the time-histories of steering angles.

The Fourier series approximations have been vastly used in spacecraft trajectory optimization [19]. With respect to temporary translation of time intervals as $[t_i^{\mathcal{I}}, t_i^{\mathcal{F}}] \rightarrow [0, t_i^{\mathcal{F}} - t_i^{\mathcal{I}}]$ for simplifying the Fourier approximation, a grid on the interval $[0, t_i^{\mathcal{F}} - t_i^{\mathcal{I}}]$, consisting of the N_p points $t_j = (j-1)\Delta t$ is considered with j being the counter for the number of interpolation points, where $\Delta t = (t_i^{\mathcal{F}} - t_i^{\mathcal{I}})/(N_p - 1)$. Assuming the functions $\alpha(t)$ and $\beta(t)$, an approximation to the Fourier series of the following form is desired

$$\alpha_{N_p}(t) = \frac{1}{\sqrt{t_i^{\mathcal{F}} - t_i^{\mathcal{I}}}} \sum_{\kappa=-N_p/2+1}^{N_p/2} e^{2\pi i \kappa t / (t_i^{\mathcal{F}} - t_i^{\mathcal{I}})} \tilde{\alpha}(\kappa) \quad (7)$$

$$\beta_{N_p}(t) = \frac{1}{\sqrt{t_i^{\mathcal{F}} - t_i^{\mathcal{I}}}} \sum_{\kappa=-N_p/2+1}^{N_p/2} e^{2\pi i \kappa t / (t_i^{\mathcal{F}} - t_i^{\mathcal{I}})} \tilde{\beta}(\kappa) \quad (8)$$

where each $\tilde{\alpha}(\kappa)$ and $\tilde{\beta}(\kappa)$ approximate the corresponding coefficient of the true Fourier series with κ as the counter for the polynomial terms. Ideally, this approximate series should satisfy

$$\begin{aligned} \alpha_{N_p}(t_j) &= \hat{\alpha}_j \\ \beta_{N_p}(t_j) &= \hat{\beta}_j \\ j &= 1, \dots, N_p \end{aligned} \quad (9)$$

That is, both approximated functions $\alpha_{N_p}(t)$ and $\beta_{N_p}(t)$ should be an interpolant of $\alpha(t)$ and $\beta(t)$ respectively, with the N_p points t_j as the interpolation points. The problem of finding these interpolants, referred to as the Fourier interpolant of α and β , has a unique solution that can easily be computed. Since the functions $e^{-2\pi i \kappa t_j / (t_i^{\mathcal{F}} - t_i^{\mathcal{I}})}$ are orthogonal with respect to the discrete inner product

$$(u, v)_{N_p} = \Delta t \sum_{j=1}^{N_p} u(t_j)v(t_j) \quad (10)$$

it is straightforward to verify that $\alpha_{N_p}(t)$ and $\beta_{N_p}(t)$ do in fact satisfy the conditions of Eq. 9. Note that the discrete inner product is an approximation of the continuous inner product. From Eq. 9, we have

$$\begin{aligned} \hat{\alpha}_j &= \frac{1}{\sqrt{t_i^{\mathcal{F}} - t_i^{\mathcal{J}}}} \sum_{\eta=-N_p/2+1}^{N_p/2} e^{2\pi i \eta t_j / (t_i^{\mathcal{F}} - t_i^{\mathcal{J}})} \tilde{\alpha}(\eta) \\ \hat{\beta}_j &= \frac{1}{\sqrt{t_i^{\mathcal{F}} - t_i^{\mathcal{J}}}} \sum_{\eta=-N_p/2+1}^{N_p/2} e^{2\pi i \eta t_j / (t_i^{\mathcal{F}} - t_i^{\mathcal{J}})} \tilde{\beta}(\eta) \end{aligned} \quad (11)$$

As can be observed, the independent variable η is used here to demonstrate the derivation of Fourier coefficients. Multiplying both sides by $\Delta t e^{-2\pi i \kappa t_j / (t_i^{\mathcal{F}} - t_i^{\mathcal{J}})}$ and summing from $j = 1$ to $j = N_p$ yields the following for $\alpha(t)$

$$\begin{aligned} \Delta t \sum_{j=1}^{N_p} e^{-2\pi i \kappa t_j / (t_i^{\mathcal{F}} - t_i^{\mathcal{J}})} \hat{\alpha}_j &= \\ \Delta t \frac{1}{\sqrt{t_i^{\mathcal{F}} - t_i^{\mathcal{J}}}} \sum_{j=1}^{N_p} \sum_{\eta=-N/2+1}^{N/2} \left[\right. & \\ e^{-2\pi i \kappa t_j / (t_i^{\mathcal{F}} - t_i^{\mathcal{J}})} e^{-2\pi i \eta t_j / (t_i^{\mathcal{F}} - t_i^{\mathcal{J}})} \tilde{\alpha}(\eta) \left. \right] &= \\ \frac{1}{\sqrt{t_i^{\mathcal{F}} - t_i^{\mathcal{J}}}} \sum_{\eta=-N_p/2+1}^{N_p/2} \tilde{\alpha}(\eta) \left[\Delta t \sum_{j=1}^{N_p} e^{-2\pi i \kappa x_j / (t_i^{\mathcal{F}} - t_i^{\mathcal{J}})} \right. & \\ \left. e^{2\pi i \eta x_j / (t_i^{\mathcal{F}} - t_i^{\mathcal{J}})} \right] & \end{aligned} \quad (12)$$

and also a similar expression for $\beta(t)$. Since

$$\begin{aligned} \left(e^{2\pi i \kappa x / (t_i^{\mathcal{F}} - t_i^{\mathcal{J}})}, e^{2\pi i \eta x / (t_i^{\mathcal{F}} - t_i^{\mathcal{J}})} \right)_N &= \\ \begin{cases} t_i^{\mathcal{F}} - t_i^{\mathcal{J}} & \kappa = \eta \\ 0 & \kappa \neq \eta \end{cases} & \end{aligned} \quad (13)$$

all terms in the outer sum on the right hand side of Eq. 12 vanish except for $\eta = \kappa$, and we obtain the following formulas as the coefficients $\tilde{\alpha}(\kappa)$ and $\tilde{\beta}(\kappa)$ by approximating the integrals that defined the coefficients of the Fourier series:

$$\begin{aligned} \tilde{\alpha}(\kappa) &= \frac{1}{\sqrt{t_i^{\mathcal{F}} - t_i^{\mathcal{J}}}} \sum_{j=1}^{N_p} e^{-2\pi i \kappa t_j / (t_i^{\mathcal{F}} - t_i^{\mathcal{J}})} \hat{\alpha}_j \Delta t \\ \tilde{\beta}(\kappa) &= \frac{1}{\sqrt{t_i^{\mathcal{F}} - t_i^{\mathcal{J}}}} \sum_{j=1}^{N_p} e^{-2\pi i \kappa t_j / (t_i^{\mathcal{F}} - t_i^{\mathcal{J}})} \hat{\beta}_j \Delta t \end{aligned} \quad (14)$$

$\kappa = -N_p/2 + 1, \dots, N_p/2$

It should be noted that the algebraic operations performed on Eq. 11 are equivalent to taking the discrete inner product of both sides of Eq. 11 with $e^{2\pi i \kappa t / (t_i^{\mathcal{F}} - t_i^{\mathcal{J}})}$.

The process of obtaining the approximate Fourier coefficients as in Eq. 14 gives the discrete Fourier transform (DFT) of $\alpha(t)$ and $\beta(t)$. By adhering to the proposed parameterization approach, the vector encompassing the decision variables can be reformulated through the transformation of Eq. 4. Considering an arbitrary value of N_p to parameterize the steering angles, the total count of decision variables amounts to $2N_T(1 + N_p)$. It is feasible to approximate the necessary number of thrust arcs, denoted as N_T , for a minimum-fuel transfer in a given space mission, while also providing an initial estimate for the time intervals of the thrust arcs $(t_i^{\mathcal{F}}, t_i^{\mathcal{J}})$. An option to consider is leveraging transfer trajectories achieved through impulsive maneuvers. The literature offers numerous techniques for obtaining solutions related to multi-impulse orbit transfers. In this research, a hybrid approach, proposed in [20] by the author is used to obtain an impulsive solution. This approach attempts to discover an initial multi-impulse transfer trajectory, which satisfies the limit on the propulsion system, while minimizing the total impulse and the transfer time simultaneously. It mainly relies on a method based on the discretized Lambert approach, providing near optimal solutions, including the number and the sequence of impulses. The approach benefits from auto-tuning operators and several methods for enhancing the diversity of the solutions during its process. Reader is urged to refer to [20] for more details. A solution of this nature comprises a vector of impulse timings denoted as $\tilde{t} = [\hat{t}_1 \dots \hat{t}_{N_T}]$. By utilizing this vector, the on-off timings can be redefined as $t_i^{\mathcal{F}} = \hat{t}_i - \tau_i^{\mathcal{F}}$ and $t_i^{\mathcal{J}} = \hat{t}_i + \tau_i^{\mathcal{J}}$, where $\tau_i^{\mathcal{F}}$ and $\tau_i^{\mathcal{J}}$ represent the time offsets relative to the impulsive timing \hat{t}_i in the i th thrust arc. By introducing these modified variables, the decision vector can be reformulated as illustrated below.

$$\mathcal{X} = \left[(\tau_i^{\mathcal{F}}, \tau_i^{\mathcal{J}}), (\hat{\alpha}_{i,1}, \hat{\beta}_{i,1}, \dots, \hat{\alpha}_{i,N_p}, \hat{\beta}_{i,N_p}) \right] \quad (15)$$

$i \in \{1, \dots, N_T\}$

where, $\hat{\alpha}_{i,j}$ and $\hat{\beta}_{i,j}$ represent the j th discrete points of the i th thrust arc for $\alpha(t)$ and $\beta(t)$, respectively. It is crucial to note that the timing offsets of thrust arcs are constrained by the orbital period of coasting orbits, denoted as \mathcal{T} in the solution of multi-impulse orbit transfer. It is noteworthy that generally, the solution of impulsive transfer that is obtained regardless of the thrust limit, may not be suitable to be used as an initial guess for low-thrust transfers. However, the multi-impulse approach, which has been utilized in this research, is based on limited impulse, and yields a multi-impulse solution with respect to the given restriction on the propulsion system. If the guess is suitable in the presented approach, a feasible solution will be found by the algorithm, which satisfies the terminal conditions. But if the guess is not suitable (not enough thrust arcs are considered), the algorithm will not be able to find any feasible solutions for the problem, leading to conclude that more thrust arcs are necessary (i.e., the value of N_T must be increased).

Additionally, determining the appropriate number of nodes is not a straightforward task. However, it is feasible to select a reasonable value for N_p by considering the thrust profiles obtained through various methods documented in the literature. In this study, the choice of $N_p = 5$ has been identified as an appropriate selection for the number of interpolation points in each thrust arc. By incorporating the boundaries of timing offsets with the limits of steering angles, the decision variables can be expressed as upper and lower bounds, as indicated below.

$$\begin{aligned} \mathcal{X}_{\min} &= \left[(0, 0), (-\pi/2, -\pi, \dots, -\pi/2, -\pi) \right] \\ \mathcal{X}_{\max} &= \left[(\mathcal{T}_i, \mathcal{T}_i), (\pi/2, \pi, \dots, \pi/2, \pi) \right] \end{aligned} \quad (16)$$

with \mathcal{X}_{\min} and \mathcal{X}_{\max} as the lower and upper bounds respectively.

III. METHOD OF SOLUTION

Regarding minimum-fuel orbital maneuvers, the objective is to achieve the least fuel consumption of the spacecraft by the end of transfer. Therefore, the objective function is presented as

$$\mathcal{F} = m_{\mathcal{G}} - m(t_f) \quad (17)$$

with $m(t_f)$ as the mass of the spacecraft by the end of orbital maneuver. Successful orbital maneuver is subject to reach the desired orbit with an acceptable error margin. Hence, orbital error vector is defined as

$$\begin{aligned} \mathcal{E}(t) &= \left[\left(\frac{a(t) - a_{\mathcal{D}}}{\sigma_a} \right)^2, \left(\frac{e(t) - e_{\mathcal{D}}}{\sigma_e} \right)^2, \left(\frac{i(t) - i_{\mathcal{D}}}{\sigma_i} \right)^2, \right. \\ &\quad \left. \left(\frac{\Omega(t) - \Omega_{\mathcal{D}}}{\sigma_{\Omega}} \right)^2, \left(\frac{\omega(t) - \omega_{\mathcal{D}}}{\sigma_{\omega}} \right)^2 \right] - 1 \end{aligned} \quad (18)$$

In this representation, $\sigma_{(\cdot)}$ denotes the maximum allowable difference between the final value and the desired value for each orbital element. Having the error vector as a function of time and the final time t_f , the constraint function can be defined as

$$\mathcal{G} = \begin{cases} \mathcal{E}(t_f), & \text{if } \mathcal{E}(t_f) \leq 0 \\ K_{\mathcal{E}} \frac{\mathcal{E}(t_f)}{\mathcal{E}_0} + \frac{1}{\mathcal{E}_0 t_f} \int_0^{t_f} \mathcal{E}(t) dt, & \text{otherwise} \end{cases} \quad (19)$$

where \mathcal{E}_0 represents the error at the initial time. It is important to note that both \mathcal{E}_0 and t_f are known parameters for a unique solution. According to this definition, the constraint function effectively distinguishes between transfer trajectories that are feasible and those that are infeasible, assigning corresponding penalties accordingly. It is evident from Eq. 18 that $\mathcal{E}(t_f) \leq 0$ indicates that the orbital element at the end of the transfer falls within the acceptable error range. Consequently, the transfer is deemed feasible, and the constraint function yields a negative value, which is proportional to the deviation from the desired orbital elements. However, if $\mathcal{E}(t_f) > 0$, it

signifies that the final conditions are not met, and the violation of the constraint is determined as the sum of the scaled error at the end of the transfer and the scaled integral of the error. Obviously, two terms of second condition in Eq. 19 have the following boundaries

$$0 < \frac{\mathcal{E}(t_f)}{\mathcal{E}_0} \leq 1 \quad (20)$$

$$0 < \frac{1}{\mathcal{E}_0 t_f} \int_0^{t_f} \mathcal{E}(t) dt \leq 1 \quad (21)$$

Following the fact that Eq. 19 is delimited by the boundaries specified in Eq. 20 and Eq. 21, it is evident that transfers characterized by a smaller integral of errors are deemed preferable when they achieve an equivalent final error. The reason is that the amount of constraint violation is a key factor for adaptive operators in discovering feasible region of the search domain. Infeasible trajectories with high constraint violation in terms of the error integral are more prone to deviate the heuristic search process. On the other hand, infeasible trajectories with less integral of errors for orbital elements, generally aids the adaptive operators to grasp the feasible region with more perception. Such consideration necessitates that the orbital parameters remain near their ultimate value throughout the entire trajectory. The final error has the coefficient $K_{\mathcal{E}}$, which increases its weight relative to the other term. As the main focus of this research does not involve tuning this coefficient, it has been set to $K_{\mathcal{E}} = 10$ in order to adjust the priority of the final error. It is important to note that both $\mathcal{E}(t_f)$ and \mathcal{G} are 1×5 vectors. The objective of the developed algorithm is to identify a vector of decision variables, as defined in Eq. 15, that minimizes the objective function expressed by Eq. 17, while concurrently satisfying the presented nonlinear constraints $\mathcal{G} \leq 0$ defined by Eq. 19.

A. Evolutionary Algorithms

Population-based techniques, particularly EAs, are among the most reliable methods in facing real-world optimization problems. Their ability in finding high quality solutions and their practicality in dealing with local optimal regions of the search space makes them popular in facing modern optimization problems. In a general perspective, EAs work based on increasing the quality of a population of solutions in an iterative manner instead of trying to find a single high-quality solution. Fig. 1 shows the overall workflow in an EA.

As it is demonstrated in Fig. 1, the optimization process in EAs starts by forming an initial population. This can be achieved either via a random seed or having a predefined population. Then, the quality of the solutions within the population is evaluated. For unconstrained optimization problems, the objective values determine the quality of solutions. In constrained optimization problems, the objective values along with the constraint violations are considered as a metric for evaluation. Then, the stopping conditions are checked for continuing or terminating

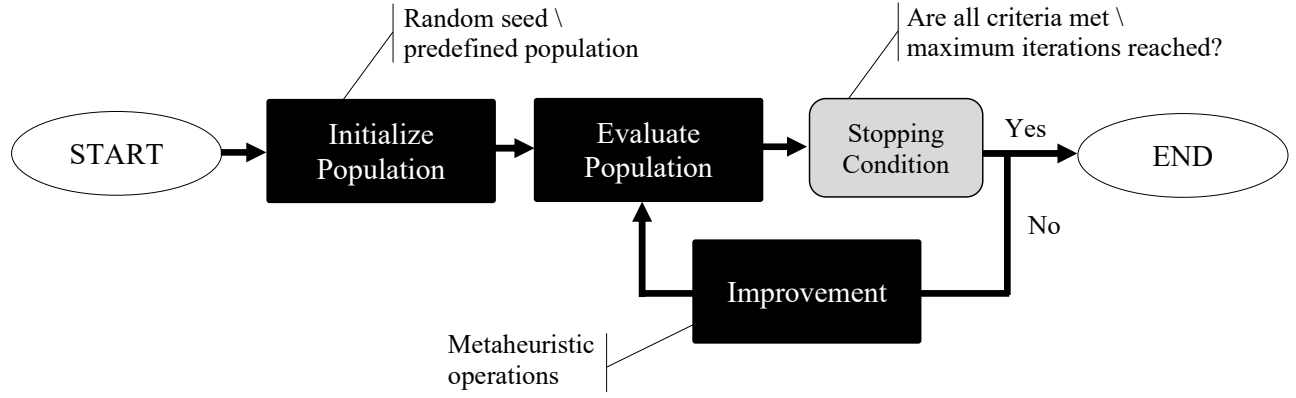


Fig. 1: Overall workflow of EAs

the process. The stopping conditions vary depending on the problem, and they may include reaching a maximum number of iterations, or availability of a solution that satisfies the expected quality and feasibility. If the conditions are met, which is very unlikely at the first iteration, the optimization process ends. Otherwise, the improvement is applied to the population in order to increase the quality of the solutions. The improvement phase includes metaheuristic operations, and varies in different EAs. For example, in GA, these operations are done via crossover and mutation operators. In PSO, velocity vectors are updated and applied to the solutions for improving the quality, and in EDAs, probabilistic models are learned and new solutions are sampled. After the improvement, the quality of the new population is evaluated again, followed by checking the stopping conditions. This loop continues

until one of the stopping conditions are met. By the end of the optimization process, the top solution with the highest quality in terms of objective value and constraints violation is considered as the output of the algorithm.

B. Estimation of Distribution Algorithm

Among the variety of EAs, one type of optimization algorithm that has been shown to be effective in dealing with complex real-world optimization problems is EDAs [22]. EDAs are a family of EAs, first introduced by Muhlenbein and Paass [23], that work based on probabilistic models. Unlike GAs, where the crossover and mutation operators are used for the movements of the populations in the search space, there are neither crossover nor mutation operators in EDAs. Instead, the new population of individ-

Algorithm 1: Overall workflow of EDA++ [21]

Input: Objective function \mathcal{F} , Constraints function \mathcal{G} , Boundaries $\mathcal{X}_{\min}, \mathcal{X}_{\max}$
Settings: Algorithm parameters: Kernel Density ξ , Distance Threshold λ

```

/* Invoke SEEDING mechanism */
1  $\mathcal{X}_0 \leftarrow$  Randomize initial population
/* Perform EVALUATION */
2  $\mathcal{F}_0 \leftarrow \mathcal{F}(\mathcal{X}_0)$ ;  $\mathcal{G}_0 \leftarrow \mathcal{G}(\mathcal{X}_0)$ 
3 while no conditions are satisfied for process termination do
    /* Invoke SELECTION mechanism */
4  $\mathcal{X}_{sel} \leftarrow$  Select top solutions via Truncation Method from  $\mathcal{X}_i$ 
    /* Invoke LEARNING mechanism */
5  $\mathcal{Q} \leftarrow$  Build a mixture of probabilistic model from  $\mathcal{X}_{sel}$  with respect to  $\lambda$ 
6  $[\Phi, \phi] \leftarrow$  Retrieve clusters from the mixture model
    /* Invoke SAMPLING mechanism */
7  $[\mathcal{X}^\Phi, \mathcal{X}^\phi] \leftarrow$  Sample new solutions from  $[\Phi, \phi]$ 
8  $\mathcal{X}_{i+1} \leftarrow$  Form new population from  $[\mathcal{X}^\Phi, \mathcal{X}^\phi]$  with respect to  $\xi$ 
    /* Invoke REPAIRING and MAPPING mechanisms */
9  $\mathcal{X}_{i+1} \leftarrow$  map infeasible solutions of  $\mathcal{X}_{i+1}$  with respect to  $\mathcal{X}_{\min}$  and  $\mathcal{X}_{\max}$ 
    /* Perform EVALUATION */
10  $\mathcal{F}_{i+1} \leftarrow \mathcal{F}(\mathcal{X}_{i+1})$ ;  $\mathcal{G}_{i+1} \leftarrow \mathcal{G}(\mathcal{X}_{i+1})$ ;
    /* Invoke REPLACEMENT mechanism */
11  $\mathcal{X}_{i+1} \leftarrow$  Form new population from  $[\mathcal{X}_i, \mathcal{X}_{i+1}]$ 

```

Output: Best solution

uals is sampled from a probability distribution, which is estimated from a database containing selected individuals from the previous generation.

EDAs have been used in variety of research to deal with different problems in aerospace community [24]. Although EDAs have shown to be competitive and reliable EAs, they were not given much attention in spacecraft trajectory optimization similar to GA, PSO, and DE [4]. The most recent version of EDAs is EDA++, which has been recently developed for constrained continuous optimization problems [21]. EDA++ incorporates multiple heuristic mechanisms to effectively handle the satisfaction of nonlinear constraints, surpassing rival EAs in terms of both efficiency and execution time. However, it should be noted that EDA++ treats the optimization problem as a black box, lacking any adaptive capabilities. Since it benefits from a dynamic framework of probabilistic models, it has a good potential for adaptations towards challenging optimization problems in continuous domain with nonlinear constraints. The pseudocode for this algorithm is presented in Algorithm 1. This EDA is equipped with mechanisms that prioritize feasibility conservation, with the objective of swiftly uncovering high-quality feasible solutions within the context of constrained continuous optimization problems. In this subsection, the overall workflow of the original algorithm is briefly described. However, the intricate mechanisms and the optimization process involved in this research exceed the scope of this article. Therefore, it is highly recommended that readers refer to [21] for in-depth discussions on the development, performance evaluation, and specific details pertaining to the workflow of the original algorithm before delving deeper into the subject matter.

The algorithm comprises multiple mechanisms that rely on probabilistic models, which are employed throughout the optimization process. Two parameters including *kernel density* (ξ) and *distance threshold* (λ), which will be explained in the next subsection, control the performance of the algorithm. Initially the algorithm starts with the seeding mechanism, aimed at exploring the search space for initial feasible solutions (Line 1 of Alg. 1). The initial population is evaluated and the objective values and constraint violations are extracted (Line 2 of Alg. 1). Then, the main loop begins if the expected solution is not achieved. Having the initial population, most promising individuals are chosen via the selection mechanism. Truncation method is utilized to extract the top quality solutions (Line 4 of Alg. 1). Then, the learning mechanism construct a mixture of probabilistic models based on the population of selected individuals (Line 5 of Alg. 1) with respect to the distance threshold λ . The mixture model has two types of components, including parent clusters Φ and smart clusters ϕ , and each component possesses a special information about the search domain (Line 6 of Alg. 1). After constructing the probabilistic model, new solutions are generated via the sampling mechanism (Line 7 of Alg. 1). The population size for each type of cluster varies, depending on the kernel

density ξ (Line 8 of Alg. 1). The newly obtained solutions are refined via the repairing mechanism and the mapping mechanism to satisfy the boundaries and constraints (Line 9 of Alg. 1). Finally, the quality of the new population is evaluated (Line 10 of Alg. 1) and the replacement mechanism combines the newly obtained population with the existing population and extracts the best obtained solution (Line 11 of Alg. 1). Every mentioned mechanism possesses various parameters that govern the algorithm's behavior. Modifying these parameters entails a trade-off between the algorithm's exploration and exploitation capabilities, effectively navigating the search domain. In this research, two of these parameters (ξ and λ) are aimed to be adapted by the features of the low-thrust orbit transfer problem. The justifications of choosing this algorithm lies upon the fact that it outperforms the majority of modern constrained continuous optimization algorithms. Also, due to its utilization of the framework of EDAs, an algorithm from this class encompasses numerous parameters and components that are linked to probabilistic models. These elements serve to regulate its exploration and exploitation abilities, thereby granting it a considerable degree of adaptability and flexibility. To solve the formed problem, two novel operators for EDA++ are developed to make it adaptive based on the complexity of the orbit transfer problem. These operators are associated with two algorithm parameters, which control the exploration and exploitation capabilities of the algorithm.

C. Algorithm Parameters

Kernel density ξ is one of the key parameters, associated with the sampling mechanism (Line 8 of Alg. 1) that acts as a balancing threshold for dedicating populations to parent clusters and smart clusters. It specifies whether the newly generated solutions belong to the parent clusters or the smart clusters. In each iteration with i being the counter for iterations, new solutions are sampled based on the parent clusters Φ and smart clusters ϕ as

$$\mathcal{X}_i = \left\{ \left\{ \mathcal{X}_{i,j}^\Phi, \mathcal{X}_{i,k}^\phi \mid j \in \{1, \dots, \xi N_{pop}\}, \right. \right. \\ \left. \left. k \in \{1, \dots, (1 - \xi)N_{pop}\} \right\} \right\} \quad (22)$$

where \mathcal{X}^Φ and \mathcal{X}^ϕ are the generated solutions associated with parent clusters and smart clusters respectively, while j and k are the indices of the solutions in each component. As can be appreciated, ξ has a boundary of $0 < \xi < 1$. By default, choosing $\xi = 0.5$ makes the algorithm dedicate populations with equal sizes between the parent clusters and smart clusters. Increasing the value of this parameter results in higher population size for parent clusters and lower population size for smart clusters, and vice versa. Therefore, high values yield less number of samples in smart clusters, hence increases the exploitation of the search process.

Another parameter is outlier detection distance threshold λ , which is associated with the learning mechanism

(Line 5 of Alg. 1) and specifies the distance limit from the centroid of parent clusters for detecting smart clusters. This parameter controls the sensitivity of the learning mechanism in building the mixture model. Each potential solution \mathcal{Y}_i^Φ of the parent cluster Φ in i th iteration, is considered as the centroid of a new smart cluster if the following condition is satisfied:

$$\frac{|\mathcal{Y}_i^\Phi - \mu_i^\Phi|}{\sigma_i^\Phi} > \lambda \quad (23)$$

where μ_i^Φ is the centroid of the parent cluster, and σ_i^Φ is the variance of solutions of the parent cluster. The parameter λ is a coefficient of variance and has a typical limits of $1 < \lambda < 2$. High values of this parameter reduces the sensitivity of detecting outliers, following by the increment of exploitation.

D. Problem Identifiers

The algorithm parameter selection for the proposed orbit transfer problem is challenging due to its high complexity. In order to adapt the two aforementioned parameters, λ and ξ , they are chosen based on the characteristics of the space mission, considering the given mission parameters \mathcal{P} in Eq. 3. Each unique problem consists of thirteen parameters in \mathcal{P} . However, it is more convenient to analyze the search domain structure using fewer parameters. Instead of considering the explicit absolute values of the initial and desired orbital elements, their differences are taken into account. Additionally, the thrust to weight ratio is considered rather than individual thrust and mass values. Following this approach, three parameters called problem identifiers are defined as follows.

$$\mathcal{D}_1 = \left(\frac{a_{\mathcal{D}} - a_{\mathcal{J}}}{6R_e} \right)^2 + (e_{\mathcal{D}} - e_{\mathcal{J}})^2 \quad (24)$$

$$\mathcal{D}_2 = \left(\frac{i_{\mathcal{D}} - i_{\mathcal{J}}}{\pi} \right)^2 + \left(\frac{\Omega_{\mathcal{D}} - \Omega_{\mathcal{J}}}{2\pi} \right)^2 + \left(\frac{\omega_{\mathcal{D}} - \omega_{\mathcal{J}}}{2\pi} \right)^2 \quad (25)$$

$$\mathcal{D}_3 = \frac{T_{\max}}{m_{\mathcal{J}}} \quad (26)$$

where R_e is the mean radius of Earth, and the orbital angles in \mathcal{D}_2 are in radians. According to the presented identifiers, the three parameters \mathcal{D}_1 , \mathcal{D}_2 , and \mathcal{D}_3 represent different aspects of the space mission. \mathcal{D}_1 signifies the intended change in the shape of the space orbit, \mathcal{D}_2 represents the desired orientation of the space orbit, and \mathcal{D}_3 denotes the available acceleration that can be utilized to successfully complete the space mission. Based on the presented mathematical model of the problem, space missions with high \mathcal{D}_1 and \mathcal{D}_2 and low \mathcal{D}_3 in general, yield optimization problems characterized by high dimensions. However, the variation of dimensions due to changes in problem identifiers are non-linear. Note that I_{sp} is dismissed as a problem identifier in this research due to the fact that it has relatively less impact on the shape of the search domain in comparison to other parameters, and it has shown to be a weak problem identifier. This claim

will be justified in the following subsection. Besides the problem identifiers, an augmented cost function is defined for landscape feature analysis as

$$\mathcal{J}(\mathcal{F}, \mathcal{G}) = \begin{cases} \frac{\mathcal{F}}{m_{\mathcal{J}}}, & \text{if } \max(\mathcal{G}) \leq 0 \\ \sum \mathcal{G}, & \text{otherwise} \end{cases} \quad (27)$$

where \mathcal{F} and \mathcal{G} represent two functions associated with the objective and constraints violation as in Eq. 17 and Eq. 18 respectively. Clearly, the augmented cost function has the range of $0 < \mathcal{J} \leq 1$ for feasible trajectories and $1 < \mathcal{J}$ for infeasible trajectories. It is important to clarify that the augmented cost function described here is solely utilized for landscape feature analysis and not for optimization purposes. The optimization algorithm employed in this context possesses its own internal mechanisms to handle objectives and constraints [21].

E. Landscape Features

To design effective adaptive operators, it is crucial to identify the complexity inherent in the problem. The goal is to discover the effects of problem identifiers \mathcal{D}_1 , \mathcal{D}_2 , and \mathcal{D}_3 on the augmented cost function \mathcal{J} , and try to match them with algorithm parameters λ and ξ accordingly. The most common approach for achieving this goal is using FLA techniques in extracting the characteristics of the problem [25]. FLA methods propose various metrics for quantifying the problem characteristics. A comprehensive survey by Ochoa and Malan [26] introduces variety of these methods. Perhaps the only attempt in utilizing FLA metrics to analyze the difficulty of spacecraft trajectory optimization problems is the work by Choi and Park [27], which has been recently presented. The authors conducted thorough research to investigate the complexity of several renowned problems sourced from the Global Trajectory Optimization Problems (GTOPs) database [28] using various metrics derived from FLA. However, no research has been concentrated on the development of EAs based on the information that are acquired via the FLA techniques in spacecraft trajectory optimization.

The *dispersion* metric, introduced by Lunacek and Whitley [29], is considered as one of the most practical metrics for FLA. It provides valuable insights into the structural characteristics of the search domain. Specifically, the dispersion metric quantifies the average distance between pairs of individuals identified as high-quality solutions. Originally designed for continuous optimization problems, this metric has subsequently been employed to examine the search landscape of numerous problem domains [30], [31]. The measure of dispersion quantifies the degree to which high-quality solutions are concentrated within a specific problem domain as

$$\psi = \frac{1}{\zeta_{n_\psi}(\zeta_{n_\psi} - 1)} \sum_{i=1}^{\zeta_{n_\psi}-1} \sum_{j=i+1}^{\zeta_{n_\psi}} \|\mathcal{X}_i - \mathcal{X}_j\| \quad (28)$$

where ψ represents the dispersion value, ζ represents the top percentage of chosen samples based on the augmented cost function \mathcal{J} , and n_ψ represents the total number of selected populations. By scaling the distances, the dispersion value is confined as $0 < \psi < 1$. This metric allows us to identify the evolvability of the solution domain and deception of the search space in the proposed orbit transfer problem, while considering the computational complexity as $\mathcal{O}(\zeta n_\psi^2)$. When the samples approach the search space with the most promising solutions (i.e., when ζ decreases), an increase in dispersion indicates a weak global structure. This weak structure makes the problem more challenging to solve, requiring additional exploration. Conversely, if lowering the threshold of promising solutions leads to low dispersion, it indicates that more exploitation is necessary to reach the global optimal solution.

Using the suggested metric and problem identifiers, an analysis of the problem landscape is conducted on a dataset of space missions. To create a grid-like dataset of space missions, we initially consider 1000 distinct space missions. These missions are uniformly generated, taking into account the following mission parameters.

$$\begin{aligned}
6600 \text{ km} &< a_{\mathcal{D}} < 42164 \text{ km} \\
0 &< e_{\mathcal{D}} < 0.8 \quad (\text{s. t. } r_p > R_e + 200 \text{ km}) \\
0 &< i_{\mathcal{D}} < \pi \\
0 &< \Omega_{\mathcal{D}} < 2\pi \\
0 &< \omega_{\mathcal{D}} < 2\pi \\
10^{-3} N &< T_{\max} < 10 N \\
300 \text{ kg} &< m_{\mathcal{D}} < 2000 \text{ kg} \\
1500 \text{ s} &< I_{\text{sp}} < 5000 \text{ s}
\end{aligned} \tag{29}$$

Then, additional mission sets are also generated and added to the data set with respect to the following rule. For each generated mission set, one identifier is considered and the mission parameters of that identifier are kept fixed, while other mission parameters affecting the other two identifiers are randomized to generate 30 additional mission sets. Same process is applied for the other two identifiers as well, and all newly generated mission sets are appended to the initial mission sets. By taking into account a range of thresholds, the dispersion is calculated for each problem by generating 100 solutions that are uniformly distributed within the defined boundaries of \mathcal{X}_{\min} and \mathcal{X}_{\max} for each respective problem. To evaluate the robustness of the feature, 50 different samples are taken for each problem and the dispersion is extracted with different thresholds. The bound-normalized dispersion values for three problem identifiers are depicted in Fig. 2, Fig. 3, and Fig. 4. For each value of problem identifiers, the mean and standard deviation of the resulting dispersion values are given based on various thresholds.

According to the normalized dispersion for orbital shape, a relatively large decrease in dispersion can be observed when the problem identifier \mathcal{D}_1 increases. This

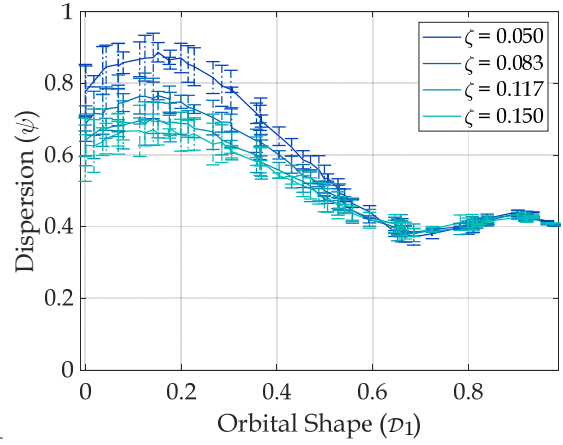


Fig. 2: Variation of dispersion with orbital shape

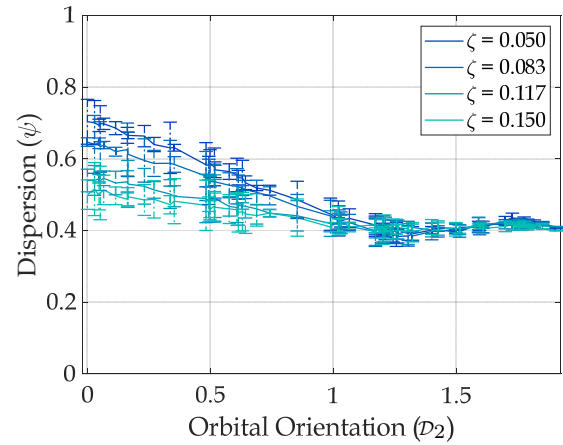


Fig. 3: Variation of dispersion with orbital orientation

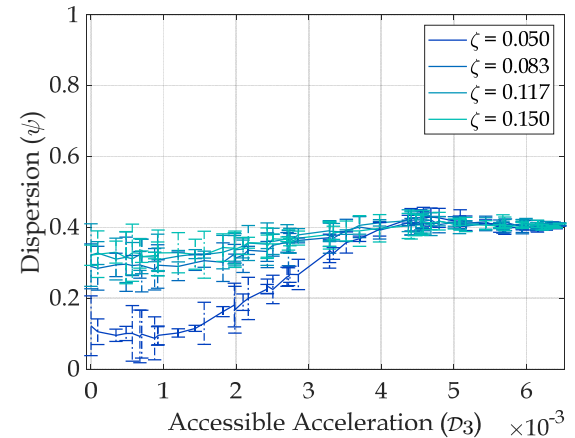


Fig. 4: Variation of dispersion with accessible acceleration

variation indicates that orbital transfers characterized by significant variations in semi-major axis and eccentricity result in high-quality solutions that are increasingly proximate to one another. This observation aligns with the understanding that as the desired change in orbital

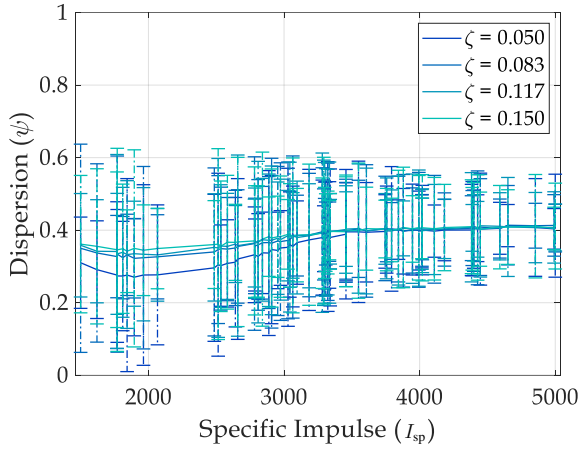


Fig. 5: Variation of dispersion with specific impulse

parameters increases, a larger number of revolutions, thrust arcs, and on-off thrust profiles must be identified for successful orbit transfers. Same insight can be inferred for the orientation of the space orbit based on \mathcal{D}_2 identifier. The changes of dispersion for \mathcal{D}_3 has a different variation in comparison to \mathcal{D}_1 and \mathcal{D}_2 . As can be observed from Fig. 4, the dispersion has an increasing variation. These changes can be identified as the value of the identifier for accessible acceleration goes up, and also when the threshold is increased from $\zeta = 0.050$ to $\zeta = 0.150$. The dispersion variation for \mathcal{D}_3 endorses that the solution domain is more chaotic when thrust to mass ratio is low for a given orbit transfer problem.

Recalling that the specific impulse was dismissed as a problem identifier, the dispersion value for this variable is plotted in Fig. 5. As can be observed, unlike the proposed identifiers, no explicit pattern can be observed for I_{sp} based on changes within the dispersion threshold and the increment of this variable, certifying that dispersion cannot grab much information about the structure of the search domain through the specific impulse.

Analysis of standard deviations of dispersion shows a slight decreasing variability of dispersion between selected samples for all problem identifiers. It indicates high discriminative ability of dispersion when orbital changes are not high. Also, it can be observed that the variation of dispersion approaches to zero as problem identifiers increase, indicating an insignificant change in the landscape structure captured by dispersion for large changes in orbital shape and orientation. Note that it can be mathematically proved that for an ideal problem identifier, dispersion converges to $1/\sqrt{6}$ as problem dimension tends to infinity [32]. The analysis of the proposed metric confirms its relative reliability in describing the complexity of the presented low-thrust trajectory optimization problem.

F. Adaptive Operators

By utilizing Eq. 28, the proposed landscape metric can be computed based on a space mission parameter vector

\mathcal{P} . The evolution of the dispersion value is monitored as ζ decreases, while ensuring it remains within the range of $0.05 < \zeta < 0.15$. The dispersion difference, denoted as $\Delta\psi$, is then employed as an indicator to classify the complexity of the orbit transfer. A negative value of $\Delta\psi$ indicates that the best fitness values are concentrated within a small sub-region of the search space. This implies that a greater emphasis on exploitation of the search domain is necessary to uncover promising solutions. A $\Delta\psi$ value close to 0 signifies that the best fitness values are evenly distributed throughout the entire search space. On the other hand, higher $\Delta\psi$ values suggest the presence of localized promising solutions in distinct remote funnels. Consequently, algorithms with enhanced exploration capabilities are generally preferred in such cases.

Following the proposed FLA metric, two heuristic models are proposed in this research to connect the problem complexity ($\Delta\psi$) to the aforementioned algorithm parameters (ξ and λ). These heuristic functions are mainly defined based on the natural expected performance of the optimization algorithm within the tuning process of its exploration and exploitation capabilities. It is noteworthy that, the nature of heuristic models is that they are not guaranteed to be the best existing models to properly alter the search process in the best possible way. However, in most cases they work and produce high satisfactory solutions. Current research is the first attempt in developing a rule of thumb approach for adaptive operators of EDA++.

The main challenge which has been considered in designing the adaptive operators is the type of knowledge used by the operators and the type of feedback they perceive regarding the complexity of the spacecraft trajectory optimization. Positive knowledge means that the operator rewards correct search direction and negative knowledge means that the operator penalizes bad guidance of the search process. By rewarding or penalizing exploration and exploitation within the search process, the algorithm generates some belief about the good or bad areas in the search. A positive search strategy biases the search

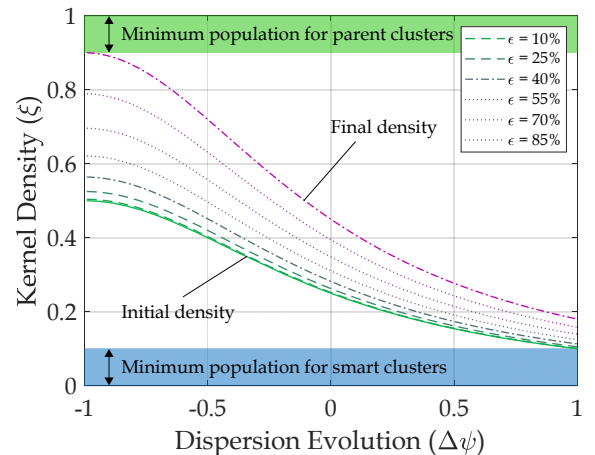


Fig. 6: Adaptive kernel density for assigning clusters' population

towards a good area of the search space, and a negative search strategy avoids an infeasible search space to explore promising areas in the search space. Having these insights in mind, the adaptive operator for kernel density is defined as

$$\xi = \frac{K_\xi}{1 + (\Delta\psi + 1)^2} \quad (30)$$

where K_ξ is the kernel density coefficient, which is updated within the optimization process as

$$K_\xi = 0.5 + 0.4\epsilon^2 \quad (31)$$

where ϵ represents the proportion of the computational budget utilized during the optimization process, indicating the progress made. Fig. 6 demonstrates how the proposed kernel density changes as the dispersion evolves over time. It can be noted that the kernel density adjusts according to the evolution of dispersion in the orbit transfer problem, which fluctuates based on the chaotic nature of the search space. When the trajectory optimization problem for the desired orbit transfer has a high dispersion evolution, the kernel density assigns lower values. Consequently, the number of populations within the intelligent clusters increases, enhancing the algorithm's exploration capability. Conversely, for low dispersion evolution, the kernel density is increased, resulting in a probabilistic model with denser parent clusters and greater exploitation during the search process. As can be observed, the kernel density is scaled from 0.1 to 0.9, which means there is always ten percent reserved population for each type of clusters (Φ and ϕ). The reason for such consideration is to prevent the operator to assign zero population to either type of clusters. The other observation that can be highlighted is that the kernel density does not have a symmetric variation for $\Delta\psi \geq 0$ and $\Delta\psi \leq 0$. Eq. 30 shows that the dedication of population to parent clusters is more prone to dispersion evolution. The main reason for such a consideration lies upon the natural behaviour of the employed optimization framework, in which sharp variation of kernel density for smart clusters is more preferred for finding high quality solutions. Statistically, the proposed variations have shown to be more effective in finding optimal transfer trajectories while maintaining feasibility.

Besides the kernel density, the adaptive operator for adjusting outlier detection distance threshold is defined as

$$\lambda = K_\lambda \cos\left(\pi \frac{\Delta\psi + 1}{2}\right) + K_\lambda + 1 \quad (32)$$

with K_λ being the coefficient for outlier detection distance threshold, defined as

$$K_\lambda = 0.25(1 + \epsilon^2) \quad (33)$$

which indicates its nonlinear dependency on the optimization progress. The variation of the presented adaptive operator is illustrated in Fig. 7. This representation shows gradual decrease in the outlier detection distance threshold as the search space requires more exploration,

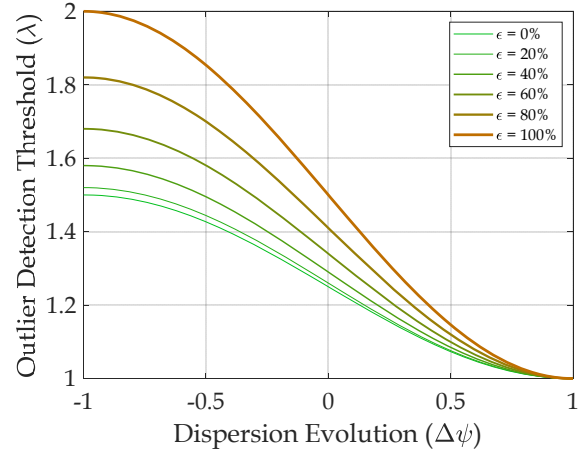


Fig. 7: Adaptive outlier detection distance threshold

i.e. high values of $\Delta\psi$. There is a slight decrease in the sensitivity of the algorithm for detecting smart clusters as the optimization goes on (increase in ϵ yields less sensitivity for low dispersion evolution values). Also, the strictly monotone decreasing function for outlier detection distance threshold covers the bounds of this algorithm parameter based on the amount of dispersion evolution of orbit transfer problem, as high $\Delta\psi$ values correspond to lower detection threshold for smart clusters, which comes with more exploration. The nonlinear variation has shown to be inline with more feasibility maintenance of the search process.

IV. MISSION SCENARIO

The proposed approach has been applied in a non-coplanar orbit transfer problem considering various thrust levels. The capability of the proposed adaptive operators are evaluated by comparing the quality of the obtained solutions via the proposed EDA with the obtained solutions via non-adaptive EDA. Using the proposed algorithm, the kernel density ξ and outlier detection distance threshold λ as presented in Eq. 30 and Eq. 32 are considered within the algorithm, while the rest of the algorithm parameters has been adjusted similar to [21]. The integration time step is set to 20s and the arbitrary orbital elements of the initial and final orbits are assumed as in Table I.

TABLE I: Orbital elements of initial and final orbits

	Initial orbit	Final orbit
a	38000 km	11700 km
e	0.05	0.2
i	70°	40°
Ω	175°	255°
ω	195°	330°

In this orbit transfer, the initial mass of the spacecraft is assumed to be $m_{\mathcal{J}} = 840kg$, while the specific impulse is considered as $I_{sp} = 2300s$. The thresholds for targeting desired orbital elements in Eq. 18 are considered as $\sigma_a =$

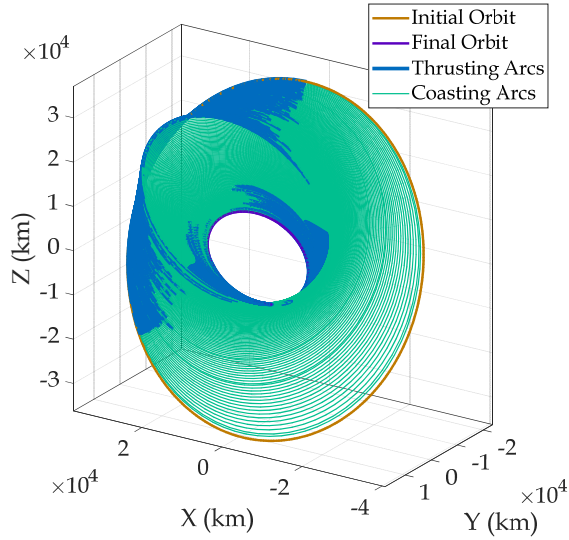


Fig. 8: Minimum-fuel orbit transfer for $T_{\max} = 1.0N$

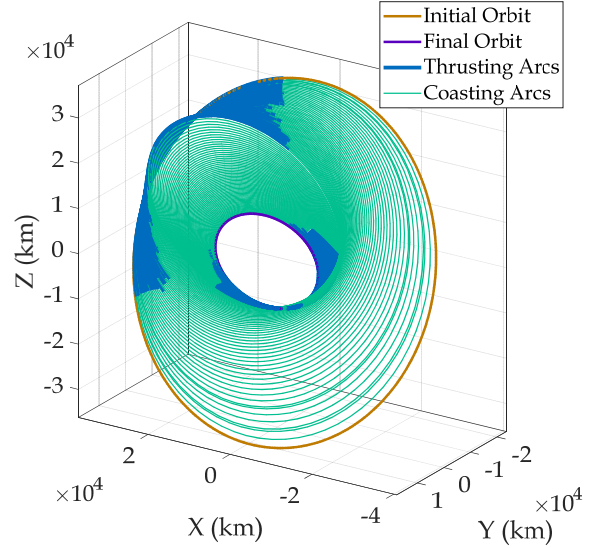


Fig. 10: Minimum-fuel orbit transfer for $T_{\max} = 2.4N$

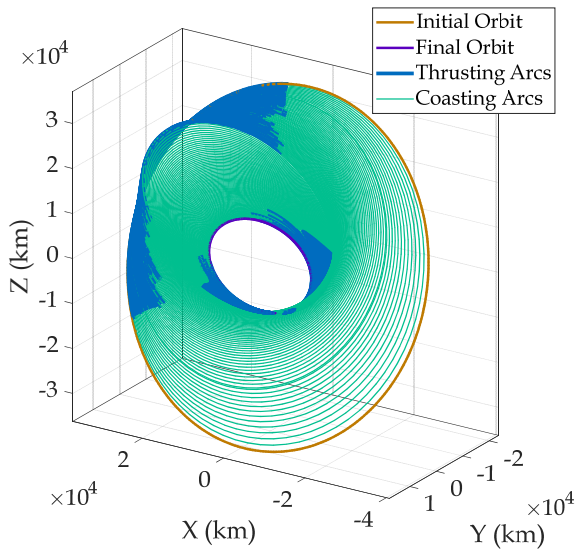


Fig. 9: Minimum-fuel orbit transfer for $T_{\max} = 1.7N$

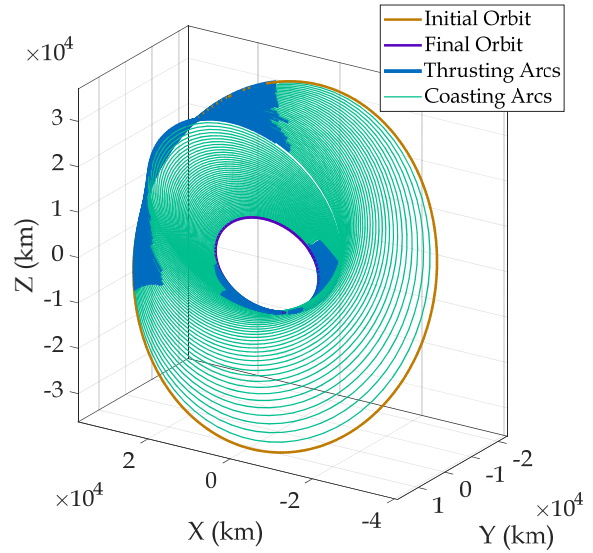


Fig. 11: Minimum-fuel orbit transfer for $T_{\max} = 3.1N$

10 km, $\sigma_e = 10^{-3}$, $\sigma_i = 10^{-2^\circ}$, $\sigma_\Omega = 10^{-1^\circ}$, $\sigma_\omega = 10^{-1^\circ}$. The problem is tackled with five different thrust levels of $T_{\max} = 1.0N$, $T_{\max} = 1.7N$, $T_{\max} = 2.4N$, $T_{\max} = 3.1N$, and $T_{\max} = 3.8N$. The obtained results are summarized in Table II, while the transfer trajectories for different thrust levels are illustrated in Fig. 8 to Fig. 12.

According to the obtained results, the proposed approach managed to find feasible solutions in all cases. For $T_{\max} = 1N$, the orbit transfer is accomplished in 246 revolutions within 2172.7043 hours. As the thrust limit increases, the achieved transfer trajectory corresponds to less transfer time and orbital revolutions, ending with transfer time down to 684.2225 days with 80 revolutions for $T_{\max} = 3.8N$. The differences between the values of

TABLE II: Characteristics of transfer trajectories for each maximum thrust limit

T_{\max} [N]	Rev.	t_f [h]	E_a [km]	E_e	E_i [deg]	E_Ω [deg]	E_ω [deg]
1.0	246	2172.7043	8.8369	1.9189e-4	9.7558e-3	7.6068e-3	1.5612e-2
1.7	156	1368.9828	5.0644	6.4373e-5	2.3831e-3	8.6188e-2	4.0312e-3
2.4	118	1040.2255	5.4856	4.9598e-5	6.2228e-3	2.7771e-3	2.3147e-3
3.1	94	811.6097	7.9372	3.5017e-5	4.8484e-3	1.0696e-3	3.0198e-3
3.8	80	684.2225	4.6495	1.8960e-6	1.8065e-3	9.2294e-3	3.3466e-4

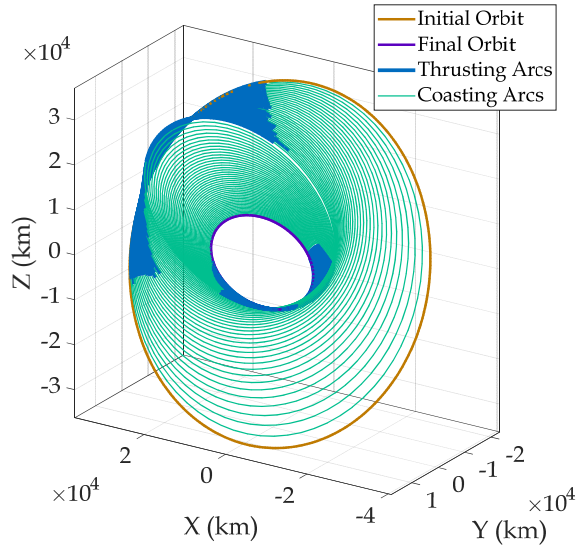


Fig. 12: Minimum-fuel orbit transfer for $T_{\max} = 3.8N$

the desired orbital elements and those associated with the obtained solutions are also provided in Table II, indicating the feasibility of the transfer trajectory with respect to the desired thresholds for each orbital element defined in Eq. 18. In this regard, the $T_{\max} = 1N$ case has shown to be the most complicated case for satisfaction of constraints, specially towards reaching the desired inclination and argument of perigee.

Noting that the solution of multi-impulse transfer for this problem that has been utilized for the proposed approach corresponds to total velocity change of $\Delta v = 4.5118km/s$ and the final mass of $m(t_f) = 674.6303kg$. The resulting thrust profile for each case is available. Fig. 13 shows the components of thrust vector for $T_{\max} = 1N$ while the time-histories of direction angles $\alpha(t)$ and $\beta(t)$ are depicted in Fig. 14. It can be observed how five nodes interpolation parameterization of the steering angles in each thrust arcs yields the optimal transfer trajectory. Although it is possible to increase the number of discrete nodes in the model, the attainment of a feasible solution serves as empirical evidence validating the appropriateness of the selected number of dedicated discrete nodes for parameterizing the steering angles. Moreover, Fig. 13 and Fig. 14 include two subplots, representing the variations in two time brackets. As can be appreciated, the bang-bang control profiles can be identified for the optimal input function. They also indicate that as the spacecraft reaches the final orbit, higher number of on-off switches are associated with the optimal transfer trajectory. This can be inferred from the fact that for the same duration of 50 hours, there are 4 on-off switches within the time bracket of $500 < t < 550$. However, when $2000 < t < 2050$, near 12 on-off switches can be identified in the optimal thrust profile. This observation is inline with the fact that the orbital period of the final orbit is much less than the initial orbit, thus the transfer

trajectory ends up in more frequency of thrust switches for transition between coast arcs and thrust arcs.

Following the presented orbit transfer problem, the performance of the proposed approach is compared with the non-adaptive version of EDA. For every case of T_{\max} , each algorithm is implemented and executed in 30 runs with same computational budget. Table III summarizes the achieved results. Two parameters for the measurement of algorithm performance are calculated. The first parameter to consider is the feasibility ratio (FR) of each algorithm, which represents the proportion of successful runs in which a feasible transfer trajectory was obtained out of the total number of runs attempted (satisfying the constraints $\mathcal{G} \leq 0$ in Eq. 19) disregarding the amount of fuel consumption (the objective function value \mathcal{F} in Eq. 17). The other parameter is the relative best percentage (RBP), which is calculated as

$$RBP = \min(100 \times \frac{\bar{\mathcal{F}} - \mathcal{F}^*}{\mathcal{F}^*}) \quad (34)$$

where $\bar{\mathcal{F}}$ represents the vector of objective values corresponding to the feasible solutions obtained by the algorithm, and \mathcal{F}^* represents the best solution obtained among both algorithms. A RBP (Relative Best Percentage) value of zero indicates that the algorithm successfully found the best possible solution compared to the other algorithm. Any non-zero value indicates the relative difference between the best obtained solution and the global best solution. According to the comparison presented in Table III, the proposed adaptive approach consistently found feasible solutions in all runs for $T_{\max} = 3.1N$ and $T_{\max} = 3.8N$. The non-adaptive EDA has shown competitiveness, although with slightly lower feasibility ratios. Additionally, considering the RBP values, it can be confirmed that the best obtained solution for $T_{\max} = 1N$, $T_{\max} = 1.7N$, and $T_{\max} = 3.8N$ belongs to the proposed adaptive approach. For the cases of $T_{\max} = 2.4N$ and $T_{\max} = 3.1N$, the best obtained solution through the adaptive approach is extremely close to the one obtained via the non-adaptive EDA, with RBP values on the order of 10^{-8} and 10^{-9} , respectively. Overall, it is evident that the advantage of using the adaptive approach is more pronounced for lower levels of thrust.

TABLE III: Comparison of the algorithms' performance in 30 runs

T_{\max} [N]	Adaptive approach		Non-adaptive approach	
	FR	RBP	FR	RBP
1	73.33%	0.000e + 00	36.67%	3.197e + 01
1.7	86.67%	0.000e + 00	53.33%	6.649e + 00
2.4	93.33%	3.704e - 08	66.67%	0.000e + 00
3.1	100%	4.432e - 09	83.33%	0.000e + 00
3.8	100%	0.000e + 00	96.67%	1.447e + 01

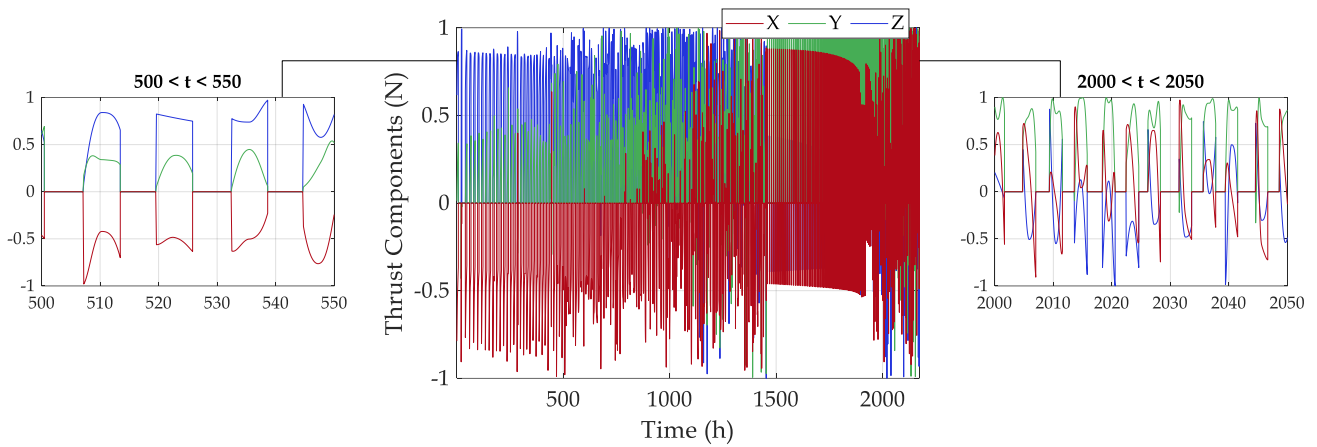


Fig. 13: Components of thrust vector during the orbit transfer for $T_{\max} = 1.0N$ with detailed representations for time intervals $500 < t < 550$ and $2000 < t < 2050$

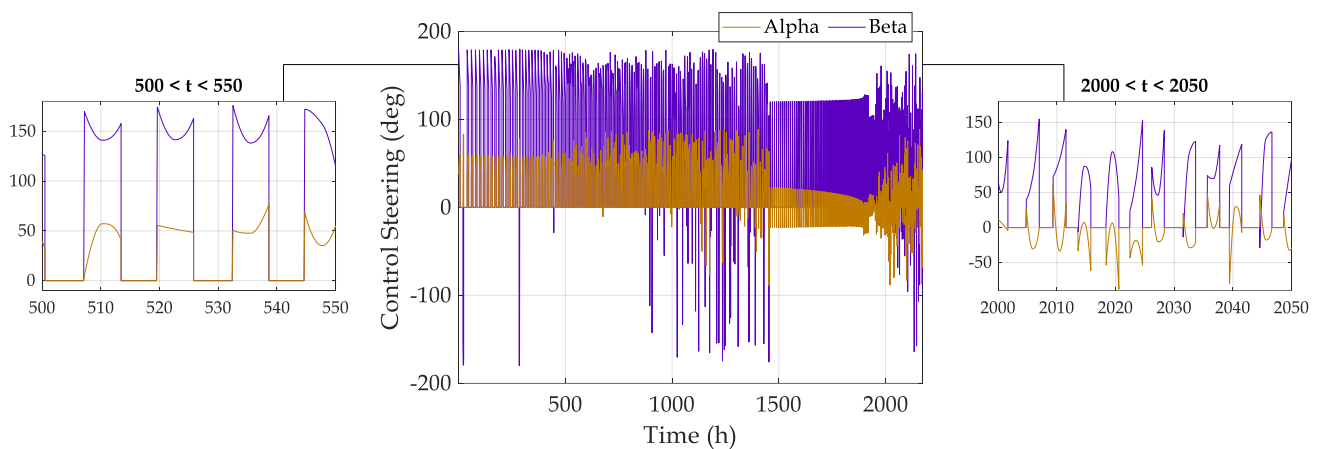


Fig. 14: Variation of thrust direction angles for $T_{\max} = 1.0N$ with detailed representations for time intervals $500 < t < 550$ and $2000 < t < 2050$

V. DISCUSSION

The developed approach in this research is an early attempt to bring the modern concept of adaptiveness from evolutionary computations into the subject of spacecraft trajectory optimization. Following the majority of research from the literature regarding the development of EAs in discovering optimal transfer trajectories, it can be observed that studying the difficulty of the problem and connecting the complexity of the search domain to the adjustment of the optimizer has not been given much attention. However, in the proposed approach of current research, the algorithm parameters are adapted not only based on the progress of the optimization, but also based on the complexity of the search domain of the problem that is aimed to be solved. As proposed, the adaptive operators for kernel density ξ and the outlier detection distance threshold λ depend on both the optimization progress ϵ and the dispersion variation $\Delta\psi$.

Several notes can be highlighted regarding the proposed adaptive approach. One key aspect is the proposed problem identifiers in Eq. 24, Eq. 25, and Eq. 26. As

presented, the landscape feature analysis in this research is based on orbital shape (\mathcal{D}_1), orbital orientation (\mathcal{D}_2), and accessible acceleration (\mathcal{D}_3). However, it does not give useful insight regarding the changes of search space due to the variation of every individual orbital elements. For instance, the approach does not differentiate transferring from $a_0 = 10000km$ to $a_f = 11000km$ and from $a_0 = 11000km$ to $a_f = 12000km$ since both have the same value of \mathcal{D}_1 considering identical values for the rest of the mission parameters. Therefore, it will be more promising for future research to perform deeper analysis and consider all elements of \mathcal{P} instead of \mathcal{D}_1 , \mathcal{D}_2 , and \mathcal{D}_3 to analyze the search domain of the orbit transfer problem.

The choice of the dispersion metric is another aspect, which can be evaluated further. Since this research is the first study in which the FLA techniques are utilized in spacecraft trajectory optimization, it is still unknown whether the dispersion was the best choice for developing adaptive operators within the proposed direct approach. The door has been left open for utilizing other FLA metrics such as *fitness distance correlation* (FDC), *length*

scale (LS), and *fitness cloud* (FL) in discovering optimal transfer trajectories [27]. However, it is noteworthy that employing every metric has its own limitations and restrictions. For example, FDC requires the global optimal solution to be available. Therefore, it is not applicable in the majority of spacecraft trajectory optimization problems since the optimal transfer is usually unknown.

As proposed, the calculation of dispersion comes with the computational complexity of $\mathcal{O}(\zeta n_{\psi}^2)$, which is a critical drawback of the proposed approach in this research. Several methods can be employed to overcome this computational complexity. One solution is to estimate the dispersion value instead of directly calculating it. This concept brings new machine learning techniques and frameworks into the proposed approach. Many supervised learning methods with variety of classifiers can be utilized. The k -nearest neighbors algorithm (k -NN) [33] can be considered as an effective technique for estimating the dispersion value. Although it may produce some errors in estimation of dispersion, deep analysis can show the reliability of utilizing k -NN in reducing the computational burden.

VI. CONCLUSION

This research was dedicated to the development of an adaptive strategy, which combines EDAs and FLA methods to achieve optimal low-thrust space transfer trajectories. The obtained results from the implementation of the presented method indicated that the proposed adaptive operators can guide the exploration and exploitation capabilities of the algorithm in an efficient manner. Convergence to feasible solutions substantiates the practicality of the utilized hybrid method in providing an initial guess based on multi-impulse orbit transfer for low-thrust trajectory optimization. Also, the selection of the dispersion as a metric for measuring the complexity of the low-thrust trajectory optimization problem has shown to be a reasonable choice, since the resulting adaptive approach outperformed the non-adaptive version of the algorithm. Future research will be dedicated to considering other FLA metrics in developing adaptive algorithms in spacecraft low-thrust trajectory optimization.

ACKNOWLEDGMENT

This research is supported by the Basque Government through the BERC 2022-2025 program, the Ministry of Science, Innovation and Universities: BCAM Severo Ochoa accreditation SEV-2017-0718, BEAZ Bizkaia 3/12/DP/2021/00150, and SPRI Group through Ekintzaile Program EK-00112-2021.

REFERENCES

REFERENCES

- [1] A. Miele

- General variational theory of the flight paths of rocket-powered aircraft, missiles and satellite carriers
In *IXth International Astronautical Congress/IX. Internationaler Astronautischer Kongress/IXe Congrès International D'astronautique*. Springer, 1959, pp. 946–970.
- [2] N. X. Vinh
General theory of optimal trajectory for rocket flight in a resisting medium
Journal of Optimization Theory and Applications, vol. 11, no. 2, pp. 189–202, 1973.
- [3] J. E. Prussing
Optimal four-impulse fixed-time rendezvous in the vicinity of a circular orbit. *AIAA Journal*, vol. 7, no. 5, pp. 928–935, 1969.
- [4] A. Shirazi, J. Ceberio, and J. A. Lozano
Spacecraft trajectory optimization: A review of models, objectives, approaches and solutions
Progress in Aerospace Sciences, vol. 102, pp. 76–98, 2018.
- [5] A.-y. Huang, Y.-z. Luo, and H.-n. Li
Global optimization of multiple-spacecraft rendezvous mission via decomposition and dynamics-guide evolution approach
Journal of Guidance, Control, and Dynamics, vol. 45, no. 1, pp. 171–178, 2022.
- [6] A. C. Morelli, C. Hofmann, and F. Topputo
Robust low-thrust trajectory optimization using convex programming and a homotopic approach
IEEE Transactions on Aerospace and Electronic Systems, 2021.
- [7] B. B. Jagannatha, J.-B. H. Bouvier, and K. Ho
Preliminary design of low-energy, low-thrust transfers to halo orbits using feedback control
Journal of Guidance, Control, and Dynamics, vol. 42, no. 2, pp. 260–271, 2019.
- [8] A. Caruso, A. A. Quarta, G. Mengali, and M. Ceriotti
Shape-based approach for solar sail trajectory optimization
Aerospace Science and Technology, vol. 107, p. 106363, 2020.
- [9] H. Yang and S. Li
Fuel-optimal asteroid descent trajectory planning using a lambert solution-based costate initialization
IEEE Transactions on Aerospace and Electronic Systems, vol. 56, no. 6, pp. 4338–4352, 2020.
- [10] V. Arya, E. Taheri, and J. L. Junkins
A composite framework for co-optimization of spacecraft trajectory and propulsion system
Acta Astronautica, vol. 178, pp. 773–782, 2021.
- [11] R. Chai, A. Tsourdos, A. Savvaris, S. Chai, and Y. Xia
Solving constrained trajectory planning problems using biased particle swarm optimization
IEEE Transactions on Aerospace and Electronic Systems, vol. 57, no. 3, pp. 1685–1701, 2021.
- [12] M. Zuo, G. Dai, L. Peng, M. Wang, Z. Liu, and C. Chen
A case learning-based differential evolution algorithm for global optimization of interplanetary trajectory design
Applied Soft Computing, vol. 94, p. 106451, 2020.
- [13] W. Feng, L. Han, L. Shi, D. Zhao, and K. Yang
Optimal control for a cooperative rendezvous between two spacecraft from determined orbits
The Journal of the Astronautical Sciences, vol. 63, no. 1, pp. 23–46, 2016.
- [14] J. L. Shannon, M. T. Ozimek, J. A. Atchison, and C. M. Hartzell
Q-law aided direct trajectory optimization of many-revolution low-thrust transfers
Journal of Spacecraft and Rockets, vol. 57, no. 4, pp. 672–682, 2020.
- [15] J. H. Choi, J. Lee, and C. Park
Deep-space trajectory optimizations using differential evolution with self-learning
Acta Astronautica, vol. 191, pp. 258–269, 2022.
- [16] D. Jimenez-Lluva and B. Root
Hybrid optimization of low-thrust many-revolution trajectories with coasting arcs and longitude targeting for propellant mini-

- mization
Acta Astronautica, vol. 177, pp. 232–245, 2020.
- [17] M. Leomanni, G. Bianchini, A. Garulli, R. Quartullo, and F. Scortecci
Optimal low-thrust orbit transfers made easy: A direct approach
Journal of Spacecraft and Rockets, vol. 58, no. 6, pp. 1904–1914, 2021.
- [18] D. A. Vallado
Fundamentals of astrodynamics and applications, 5th Ed. Microcosm Press Year, 2022.
- [19] M. Huo, G. Zhang, N. Qi, Y. Liu, and X. Shi
Initial trajectory design of electric solar wind sail based on finite fourier series shape-based method
IEEE Transactions on Aerospace and Electronic Systems, vol. 55, no. 6, pp. 3674–3683, 2019.
- [20] A. Shirazi, J. Ceberio, and J. A. Lozano
An evolutionary discretized lambert approach for optimal long-range rendezvous considering impulse limit
Aerospace Science and Technology, vol. 94, p. 105400, 2019.
- [21] A. Shirazi, J. Ceberio, and J. A. Lozano
EDA++: Estimation of distribution algorithms with feasibility conserving mechanisms for constrained continuous optimization
IEEE Transactions on Evolutionary Computation, 2022.
- [22] P. Larrañaga and J. A. Lozano
Estimation of distribution algorithms: A new tool for evolutionary computation. Springer Science & Business Media, 2001, vol. 2.
- [23] H. Mühlenbein and G. Paass
From recombination of genes to the estimation of distributions i. binary parameters
In *International conference on parallel problem solving from nature.* Springer, 1996, pp. 178–187.
- [24] M. Vasile
Optimization Under Uncertainty with Applications to Aerospace Engineering. Springer, 2021.
- [25] E. Pitzer and M. Affenzeller
A comprehensive survey on fitness landscape analysis
In *Recent advances in intelligent engineering systems.* Springer, 2012, pp. 161–191.
- [26] G. Ochoa and K. Malan
Recent advances in fitness landscape analysis
In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, 2019, pp. 1077–1094.
- [27] J. H. Choi and C. Park
Spacecraft trajectory optimizations: Metrics for fitness landscape analysis
In *AIAA SCITECH 2022 Forum*, 2022, p. 1891.
- [28] D. Izzo, T. Vinkó, and M. del Rey Zapatero
Gtop database: Global optimisation trajectory problems and solutions
2010.
- [29] M. Lunacek and D. Whitley
The dispersion metric and the cma evolution strategy
In *Proceedings of the 8th annual conference on Genetic and evolutionary computation*, 2006, pp. 477–484.
- [30] C. L. Muller, B. Baumgartner, and I. F. Sbalzarini
Particle swarm cma evolution strategy for the optimization of multi-funnel landscapes
In *2009 IEEE Congress on Evolutionary Computation.* IEEE, 2009, pp. 2685–2692.
- [31] M. Nunes, P. M. Fraga, and G. L. Pappa
Fitness landscape analysis of graph neural network architecture search spaces
In *Proceedings of the Genetic and Evolutionary Computation Conference*, 2021, pp. 876–884.
- [32] R. Morgan and M. Gallagher
Sampling techniques and distance metrics in high dimensional continuous landscape analysis: Limitations and improvements
IEEE Transactions on Evolutionary Computation, vol. 18, no. 3, pp. 456–461, 2013.
- [33] P. C. Sen, M. Hajra, and M. Ghosh
Supervised classification algorithms in machine learning: A survey and review
In *Emerging technology in modelling and graphics.* Springer, 2020, pp. 99–111.



Abolfazl Shirazi is a Postdoctoral Researcher in the Machine Learning Group at Basque Center for Applied Mathematics (BCAM), Bilbao, Spain. He received the B.Sc. and M.Sc. degrees in Aerospace Engineering in 2010 and 2012, respectively, and the Ph.D. degree (cum laude with distinction) in Computer Science from the University of the Basque Country UPV/EHU, Donostia-San Sebastian, Spain, in 2021. His research interests are in the fields of Astrodynamics and Machine Learning. More specifically, his work mainly focuses on the application of Artificial Intelligence in space trajectory design and optimization. Dr. Shirazi was awarded the La Caixa Fellowship Grant for his doctoral studies in 2016, and Severo Ochoa Fellowship Grant for his postdoctoral research in 2021.