Preprint submitted to Journal of Spacecraft and Rockets
Accepted article (Author's version)
DOI: 10.2514/1.A35570

Adaptive Estimation of Distribution Algorithms for Low-Thrust Trajectory Optimization

Abolfazl Shirazi [*]
Basque Center for Applied Mathematics - BCAM, Bilbao, Spain, 48009

A direct adaptive scheme is presented as an alternative approach for minimum-fuel low-thrust trajectory design in non-coplanar orbit transfers, utilizing fitness landscape analysis (FLA). Spacecraft dynamics is modeled with respect to modified equinoctial elements, considering J_2 orbital perturbations. Taking into account the timings of thrust arcs, the discretization nodes for thrust profile, and the solution of multi-impulse orbit transfer, a constrained continuous optimization problem is formed for low-thrust orbital maneuver. An adaptive method within the framework of Estimation of Distribution Algorithms (EDAs) is proposed, which aims at conserving feasibility of the solutions within the search process. Several problem identifiers for low-thrust trajectory optimization are introduced, and the complexity of the solution domain is analyzed by evaluating the landscape feature of the search space via FLA. Two adaptive operators are proposed, which control the search process based on the need for exploration and exploitation of the search domain to achieve optimal transfers. The adaptive operators are implemented in the presented EDA and several perturbed and non-perturbed orbit transfer problems are solved. Results confirm the effectiveness and reliability of the proposed approach in finding optimal low-thrust transfer trajectories.

Nomenclature

$\mathcal{A}$	=	Problem identifier denoting orbital shape
$\mathcal{B}$	=	Problem identifier denoting orbital orientation
$\mathcal{C}$	=	Problem identifier denoting accessible acceleration
$\mathcal{E}$	=	Orbital error
$\mathcal{F}$	=	Objective function
$\mathcal{G}$	=	Constraint function
$\mathcal{J}$	=	Augmented cost function for landscape analysis
$\mathcal{M}$	=	Remainder operator
$\mathcal{O}$	=	Computational complexity
$\mathcal{P}$	=	Mission parameters
$\mathcal{U}$	=	Uniform distribution
$\mathcal{X}$	=	Decision variables
$\mathcal{Y}$	=	Potential solution for smart cluster formation
a	=	Semi-major axis

[*] Postdoctoral Researcher, Mazarredo Zumarkalea 14, ashirazi@bcamath.org

e	=	Eccentricity
i	=	Inclination
m	=	Mass
n_{ψ}	=	Number of selected population for dispersion evaluation
p	=	Semi-latus rectum
f	=	x component of eccentricity vector
g	=	y component of eccentricity vector
h	=	x component of node vector
k	=	y component of node vector
l	=	True longitude
r_p	=	Perigee radius
t	=	Time
$\boldsymbol{x}$	=	State vector
I_{sp}	=	Specific impulse
$K_{\mathcal{E}}$	=	Final error coefficient of constraint function
K_{ξ}	=	Kernel density coefficient
K_{ν}	=	Smart cluster detection coefficient
N_T	=	Number of thrust arcs
N_p	=	Number of interpolation nodes
N_{pop}	=	Number of populations
N_{Φ}	=	Number of solutions in the parent cluster
$T_{\max}$	=	Maximum thrust level
$\mathbf{T}$	=	Thrust vector
α	=	Out-of-plane steering angle
β	=	In-plane steering angle
$\boldsymbol{\gamma}$	=	Acceleration
ζ	=	Dispersion threshold
η	=	Optimization progress
ν	=	Smart cluster detection parameter
ξ	=	Kernel density
τ	=	Thrust arc timing offset
σ	=	Allowable difference of final orbital elements
ψ	=	Dispersion
ω	=	Argument of perigee
Ω	=	Right ascension of ascending node
$\mathscr{A}$	=	Approximation operator
Subscripts		
i, j, and k	=	General counter for ith, jth, and kth component
$\mathscr{i}$	=	Initial value
$\mathscr{d}$	=	Desired value
Superscripts		
$\mathscr{i}$	=	Value at the beginning of thrust arc
$\mathscr{f}$	=	Value by the end of thrust arc
^	=	Discrete node of parameterization
T	=	Matrix transpose

I. Introduction

TRAJECTORY optimization of space vehicles has been the main subject of many research in recent decades. Pioneering works in this field are due to the efforts by Edelbaum [1], Vinh [2], and Miele [3]. Solving the resulting optimal control problem for low-thrust trajectory optimization is an extremely complex task. Over the past years, many efforts have been made to search for the best methods and algorithms in spacecraft trajectory optimization [4]. Having an overview of the past and ongoing research from the literature on this subject confirms that, regardless of employing direct or indirect methods, it is very common to see that the presented approaches end up in facing a nonlinear programming (NLP) problem, which itself is handled as a black box optimization problem. Various optimization algorithms are merged with different approaches and are applied in variety of problems [5–7]. However, with the increase in non-linearity of system dynamics, the dimensions of states, as well as the flight time, the number of discrete nodes for control and states may increase significantly, resulting in a large-scale problem that is very difficult and computationally expensive to solve. Because of these challenges, optimization of orbital maneuvers employing Evolutionary Algorithms (EAs) is becoming increasingly popular in the literature, considering their application in variety of space missions. The motivation for using evolutionary optimization algorithms relies on their sufficiency in dealing with local optimal solution and the satisfaction of constraints that naturally arise in nonlinear optimal control problems.

Regarding the development and utilization of EAs in spacecraft trajectory optimization, noticeable advances can be identified in recent efforts. For instance, a direct approach with Particle Swarm Optimization (PSO) is proposed by Wu et al. in [8] for dealing with ascending trajectory design problem from the surface of Phobos. In their research, heuristic procedures combined with Monte-Carlo simulations are implemented to resonant quasi-satellite orbit transfer problem around Phobos with two impulses. In another article by Choi et al. [9], multiple gravity-assists trajectory design and optimization problem is solved with a Differential Evolution (DE) algorithm, in which a novel mechanism and a re-initialization strategy, merged with DE are implemented to trajectory optimization of deep-space exploration missions. In another research, Ant Colony Optimization (ACO) is utilized for active debris removal based on multiple spacecraft in [10]. Also, a hybrid heuristic algorithm based on an improved PSO is developed by Zhou et al. [11] for dealing with Earth to Moon transfer. In this research, an EA incorporated with an adaptive conjugate gradient scheme is developed for transfer to Earth-Moon halo orbits. An efficient grid search algorithm is introduced by Caruso et al. in [12] for solving the optimization of coplanar orbit transfers. In this research, a search strategy combined with Genetic Algorithms (GAs) is developed for 2D transfers between elliptical orbits. Another variation of GA has been used in [13] for multi-gravity-assist trajectory optimization. Numerous studies with similar findings have been documented in the literature, in which achieving the optimal transfer trajectory requires the employment of an EA to find a vector of unknown variables for optimizing an objective function, along with satisfaction of some possible existing constraints. Within this research, the unknown variables, better known as the decision variables, may include different parameters such as discretized control sequence nodes in an atmospheric entry trajectory optimization [14], initial values of state and costate variables in a cooperative rendezvous [15], or weighting coefficients of a Q-law control method for low-thrust many-revolution trajectory optimization [16].

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 Estimation of Distribution Algorithms (EDAs) [17]. EDAs are a family of EAs, first introduced in 1996 [18], 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 individuals 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 [19]. Minisci and Avanzini utilized EDAs in trajectory optimization of two-impulse and three impulse space rendezvous problems [20]. Other applications of EDAs have been presented in many problems such as multi-spacecraft cooperation task allocation [21], conceptual design of UAVs [22], satellite formations design [23], trajectory optimization of single-stage launch vehicles [24], and orbit determination [25]. 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 by the author for constrained continuous optimization problems [26]. EDA++ is equipped with several heuristic mechanisms to deal with the satisfaction of nonlinear constraints and it outperforms its rival EAs in terms of efficiency and execution time. Yet, still it treats the optimization problem as a black box with no adaptations. Since it benefits from a dynamic framework of probabilistic models, it exhibits a high potential for adaptation in solving complex optimization problems in the continuous domain that involve nonlinear constraints.

Following the aforementioned research, it can be observed that 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, insufficient research has been conducted to thoroughly investigate 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, no clear connection can be found between the selection of the EA, or rather the choice of the EA parameters, and the complexity of the spacecraft trajectory optimization problem. In such research, usually EAs in their best setup suited for a specific problem is implemented, and the reported results are associated with the best obtained solution out of multiple runs of the algorithm. It is unclear how the performance of the employed EA setup will be if some of the mission parameters (e.g., the desired final semi-major axis in a non-coplanar low-thrust orbit transfer) are changed. The question has been remained unanswered whether the employed EAs, or the newly developed EAs are robust enough to deal with any mission parameters without the need for adjusting their parameters prior to optimization runs. These insights are the main motivation in the current article, and the main aim of this research is to find out the difficulty of orbit transfer problems, and develop an adaptive EA that benefits from these findings to adjust itself. This concept, better known as Fitness Landscape Analysis (FLA) [27], which is connected to auto-tuning and developing intelligent algorithms for complex systems, has not been given proper attention in astrodynamics. FLA includes techniques that are used to measure the difficulty of the optimization problems by means of some metrics for calculating the complexity of the search domain. Perhaps the only attempt in utilizing FLA metrics to analyze the difficulty of spacecraft trajectory optimization problems is the work by Choi and Park [28], which has been recently presented. The authors conducted a comprehensive investigation on exploring the complexity of some well-known problems from Global Trajectory Optimization Problems (GTOPs) database [29] via different FLA metrics. 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 main purpose of this research is to initially fill the gap on this matter and open the door for further attempts on employing these techniques in developing novel adaptive approaches dedicated to finding optimal trajectories for space systems.

In this article, a novel adaptive approach is proposed to deal with low-thrust trajectory optimization of Earth-orbiting spacecraft. Unlike the majority of research in the literature regarding the development of adaptive optimization algorithms for specific problems, the main goal of this research is to utilize a new concept for adaptiveness. In this new scheme, the space mission parameters and features are used to tune the algorithm for achieving high quality solutions. In most of the previous attempts [9], the aimed problem is treated as a black-box and adaptive operators are tuned based on the quality of individuals of the search space in each iteration of the optimization process. In such approaches, there are no feedback from the features, describing the nature of the trajectory optimization problem itself. So the developed algorithms in previous research can be used to solve any types of problems, regardless of the subject or the application [13]. However, in this paper, the parameters of the space mission describing the complexity and difficulty of the problem are discovered via FLA techniques and are utilized to enhance the algorithm within the optimization process. Therefore, one of the most noticeable differences between the current research and the previous attempts is that the problem is not treated as a black-box. Instead, the effort here is to match the features of the space mission with the algorithm parameters to improve the capability of the algorithm in discovering optimal transfers. Following the discovery of the complexity of the spacecraft trajectory optimization problem in low-thrust orbital maneuvers via FLA techniques, some novel adaptive operators for EDA++ is developed. The new adaptive operators take advantage of the information about the difficulty of the orbit transfer problem acquired by FLA techniques, and utilize it to adjust the exploration and exploitation capability of the algorithm [30] in searching for optimal feasible transfer trajectories. Therefore, the main contribution of this research can be summarized as analyzing the complexity of minimum-fuel low-thrust orbit transfer problem via FLA techniques, developing new adaptive operators for EDA++ based on the difficulty of orbit transfer problem obtained via the FLA technique, and implementing the developed algorithm into an adaptive approach for low-thrust trajectory optimization.

The justification of targeting the mechanisms of EDA++ in developing novel adaptive operators in this research lies upon the fact that this algorithm has already outperformed the majority of modern constrained continuous optimization algorithms [26], and since it works based on the framework of EDAs, it contains many parameters and components associated with probabilistic models to control its exploration and exploitation capabilities, which provides high level of flexibility for adaptation. Experiments will indicate that the adaptive EDA approach proposed in this research outperforms the non-adaptive EDA++, and it also manages to find better transfer trajectories for low-thrust orbital maneuvers in comparison to another method from the literature in terms of fuel mass and transfer time.

This research is aimed at the discovery of feasible transfer trajectories for non-coplanar orbit transfers considering J_2 perturbations. In Section II, after mathematical modeling of the spacecraft dynamics, a direct approach is implemented in which the shape of the thrust profile and the on-off timing for the propulsion system are considered as unknown variables. Analyzing the landscape features of the solution domain in Section III results key parameters, possessing the information regarding the complexity and difficulty of the trajectory optimization problem. Having the proposed feedback parameters, an adaptive approach within the framework of EDA++ is presented for discovering optimal minimum-fuel orbital maneuvers while maintaining the feasibility of the transfer trajectory. In Section IV, the proposed optimization framework is utilized to solve several perturbed and unperturbed test cases for low-thrust orbital maneuvers. Section V is dedicated to further discussion of the proposed approach. Section VI concludes the research.

II. Formulation of the Problem

As the primary step for facing low-thrust trajectory optimization problem, the dynamics of the spacecraft is mathematically modeled. Key parameters and unknown variables are extracted upon governing the equations of motion for the space vehicle.

A. Spacecraft Dynamics

The dynamics of the spacecraft is modeled via the set of modified equinoctial elements (MEEs) [31], including five slow elements of (p, f, g, h, k), and the true longitide l as a fast variable. Although they do not posses actual physical meaning similar to Classical Orbital Elements (COEs), this set has been frequently used in many orbit transfer problems [15, 32] since they do not suffer from singularities at zero eccentricity and inclination. Enforcing the orbital revolutions as an integer value is a special feature of using this set, which makes it suitable for variety of orbit transfers such as minimum-fuel orbital maneuvers. The vector of elements $\boldsymbol{x}$ is presented as

$$\boldsymbol{x} = [p, f, g, h, k, l]^T \tag{1}$$

Having the COEs as a, e, i, Ω, ω, and θ, denoting semi-major axis, eccentricity, inclination, right-ascension of ascending node, argument of periapse, and true anomaly respectively, the conversion between MEEs and COEs are as $p = a(1 - e^2)$, $f = e\cos(\omega + \Omega)$, $g = e\sin(\omega + \Omega)$, $h = \tan(i/2)\cos\Omega$, $k = \tan(i/2)\sin\Omega$, and $l = \Omega + \omega + \theta$. Gauss variational equations can be derived for the set of MEEs, which gives the time rate of the states as

$$\dot{\boldsymbol{x}} = \mathbb{A} + \mathbb{B}[\boldsymbol{\gamma}_c + \boldsymbol{\gamma}_p] \tag{2}$$

where $\boldsymbol{\gamma}_c$ is the input acceleration due to thrust, $\boldsymbol{\gamma}_p$ is the perturbing acceleration, while $\mathbb{A}$ and $\mathbb{B}$ are defined as

$$\mathbb{A} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \sqrt{\mu p \left(\frac{w}{p}\right)^2} \end{bmatrix}; \quad \mathbb{B} = \begin{bmatrix} 0 & \frac{2p}{w}\sqrt{\frac{p}{\mu}} & 0 \\ \sqrt{\frac{p}{\mu}}\sin(l) & \sqrt{\frac{p}{\mu}}\frac{1}{w}[(w+1)\cos(l)+f] & -\sqrt{\frac{p}{\mu}}\frac{g}{w}\kappa \\ -\sqrt{\frac{p}{\mu}}\cos(l) & \sqrt{\frac{p}{\mu}}\frac{1}{w}[(w+1)\sin(l)+g] & \sqrt{\frac{p}{\mu}}\frac{f}{w}\kappa \\ 0 & 0 & \sqrt{\frac{p}{\mu}}\frac{s^2\cos(l)}{2w} \\ 0 & 0 & \sqrt{\frac{p}{\mu}}\frac{s^2\sin(l)}{2w} \\ 0 & 0 & \sqrt{\frac{p}{\mu}}\frac{1}{w}\kappa \end{bmatrix} \tag{3}$$

with $w = 1 + f\cos(l) + g\sin(l)$, $s^2 = 1 + h^2 + k^2$, $\kappa = h\sin(l) - k\cos(l)$, and μ is the Earth's gravitational parameter.

The input acceleration can be written as $\boldsymbol{\gamma}_c = \mathbf{T}/m$, where $\mathbf{T}$ and m represent the thrust vector and mass of the spacecraft respectively. Various perturbations can be considered in the model. In this research, the perturbing acceleration due to the Earth's second zonal harmonic for the set of MEEs is considered as $\boldsymbol{\gamma}_p = [\gamma_r, \gamma_t, \gamma_n]^T$ with

$$\begin{aligned} \gamma_r &= \frac{3\mu J_2 R^2}{2r^4}\left(1 - 12\frac{\kappa^2}{s^4}\right) \\ \gamma_t &= -\frac{12\mu J_2 R^2}{r^4}\left(\frac{\kappa(h\cos L + k\sin L)}{s^4}\right) \\ \gamma_n &= -\frac{6\mu J_2 R^2}{r^4}\left(\frac{\kappa(1 - k^2 - h^2)}{s^4}\right) \end{aligned} \tag{4}$$

where J_2 denotes the second zonal harmonic and $r = p/w$. Clearly, more perturbing terms can be added to $\boldsymbol{\gamma}_p$ according to the desired level of details in modeling orbital perturbations. Also, vectors in LVLH 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 [33]. The variation of spacecraft mass during the orbit transfer can be represented by

$$\dot{m} = -\frac{|\mathbf{T}|}{I_{\text{sp}} g_0} \tag{5}$$

where I_{sp} is the specific impulse, 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. Having the orbital elements for the initial and desired orbits as $[a_i\ e_i\ i_i\ \Omega_i\ \omega_i]$ and $[a_d\ e_d\ i_d\ \Omega_d\ \omega_d]$ respectively, along with initial mass of the spacecraft m_i, specific impulse I_{sp}, and maximum available thrust level $T_{\max}$, a unique problem is established regarding the optimization of a non-coplanar orbital maneuver in a perturbed environment. The mission parameters describing a unique problem can be represented by the vector $\mathcal{P}$ as

$$\mathcal{P} = [a_i \;\; e_i \;\; i_i \;\; \Omega_i \;\; \omega_i \;\; a_d \;\; e_d \;\; i_d \;\; \Omega_d \;\; \omega_d \;\; m_i \;\; T_{\max} \;\; I_{\text{sp}}] \tag{6}$$

The problem is to find the optimal on-off time intervals of thrust-arcs along with their associated thrust profile that establish a transfer trajectory with minimum-fuel consumption subject to the satisfaction of initial condition and terminal constraints.

B. Direct Transcription

Discovering the optimal transfer trajectory for the aimed problem is subject to finding unknown thrust profiles of thrust arcs along with their on-off timings during the transfer. These unknown functions and variables can be represented in $\mathcal{X}$ as

$$\mathcal{X} = \left[t_1^{i}\, \mathbf{T}_1\, t_1^{\ell} \;\; t_2^{i}\, \mathbf{T}_2\, t_2^{\ell} \;\; ... \;\; t_i^{i}\, \mathbf{T}_i\, t_i^{\ell} \;\; ... \;\; t_{N_T-1}^{i}\, \mathbf{T}_{N_T-1}\, t_{N_T-1}^{\ell} \;\; t_{N_T}^{i}\, \mathbf{T}_{N_T}\, t_{N_T}^{\ell} \right] \tag{7}$$

where N_T is the number of thrust arcs, t_i^{i} and t_i^{ℓ} $(1 < i < N_T)$ 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 $t_i^{i} < t < t_i^{\ell}$. 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 $t_i^{i} < t < t_i^{\ell}$, and is equal to zero within the coast arcs as $|\mathbf{T}_i| = 0$ for $t_i^{\ell} < t < t_{i+1}^{i}$. The components of thrust vector within the thrust arcs can be defined as

$$\mathbf{T}_i(t) = T_{\max} \begin{bmatrix} \cos\alpha_i(t)\cos\beta_i(t) \\ \cos\alpha_i(t)\sin\beta_i(t) \\ \sin\alpha_i(t) \end{bmatrix} \tag{8}$$

with α_i and β_i as the steering angles of the space vehicle with respect to the reference frame. As can be appreciated with the current definition, the optimal time histories of steering angles are unknown and yet to be determined besides the on-off time intervals. Considering the upper and lower bounds for steering angles as $0 < \alpha_i(t), \beta_i(t) < 2\pi$, for each unique thrust arc, the variations are defined as approximated time-profiles via finite number of nodes for the steering angles as

$$[\alpha(t), \beta(t)] = \mathscr{A}(\hat{\alpha}_1, \hat{\alpha}_2, ..., \hat{\alpha}_{N_p}, \hat{\beta}_1, \hat{\beta}_2, ..., \hat{\beta}_{N_p}) \tag{9}$$

with $\mathscr{A}(\cdot)$ as the approximation operator, which converts the given discrete nodes $\hat{\alpha}_j, \hat{\beta}_j$ into continuous time-series. With respect to the fact that $0 < \hat{\alpha}_j, \hat{\beta}_j < 2\pi$ $(j = 1, ..., N_p)$, different schemes may be employed for this operator.

In this research, Chebyshev polynomials are utilized in $\mathscr{A}(\cdot)$ to parameterized the time-histories of steering angles, mainly due to their popularity in parameterization as they have been frequently used in various research according to the literature. Following the presented parameterization approach, the vector of decision variables can be reformed by transformation of Eq. 7. For an arbitrary value of N_p to parameterize the steering angles, the total number of decision variables is $2N_T(1+N_p)$. It is possible to have an estimation for the required number of thrust arcs N_T in minimum-fuel transfer for a given space mission along with an initial guess for the time intervals of thrust arcs $(t_i^{\mathscr{i}}\ t_i^{\mathscr{f}})$. One option is to take advantage of transfer trajectories obtained via impulsive maneuvers. There are numerous techniques available in the literature to obtain a solution for multi-impulse orbit transfers [34]. Such a solution contains a vector of impulse timings as $\tilde{t} = [\hat{t}_1\ \ldots\ \hat{t}_{N_T}]$. Utilizing the impulse timings vector, the on-off timings are reformed as $t_i^{\mathscr{i}} = \hat{t}_i - \tau_i^{\mathscr{i}}$ and $t_i^{\mathscr{f}} = \hat{t}_i + \tau_i^{\mathscr{f}}$, with $\tau_i^{\mathscr{i}}$ and $\tau_i^{\mathscr{f}}$ being the time offsets with respect to the impulsive timing $\hat{t}_i$ in ith thrust arc. Following the new sets of variables, the decision vector is reformed to

$$\mathcal{X} = \Big[(\tau_i^{\mathscr{i}}\ \tau_i^{\mathscr{f}})\ ,\ (\hat{\alpha}_{i,1}\ \hat{\beta}_{i,1}\ ,\ \ldots\ , \hat{\alpha}_{i,N_p}\ \hat{\beta}_{i,N_p})\Big]_{i\in\{1,\ldots,N_T\}} \tag{10}$$

where $\hat{\alpha}_{i,j}$ and $\hat{\beta}_{i,j}$ are the jth approximation nodes of ith thrust arc for $\alpha(t)$ and $\beta(t)$ respectively. It is noteworthy that thrust arc timing offsets are bounded by the orbital period of coasting orbits in the solution of multi-impulse orbit transfer. Also, the choice of the number of nodes is nontrivial. However, it is possible to select a reasonable value for N_p, based on the shape of thrust profiles obtained by different methods available in the literature. In this research, the choice of $N_p = 5$ has shown to be a proper selection for the number of interpolation points in each thrust arc. Merging the boundaries of timing offsets with the limits of steering angles as mentioned, the upper and lower bounds for the decision variables can be represented as $\mathcal{X}_{min} \leq \mathcal{X} \leq \mathcal{X}_{max}$, with $\mathcal{X}_{min}$ and $\mathcal{X}_{max}$ as the lower and upper bounds respectively.

III. Trajectory Optimization

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

$$\mathcal{F} = m_{\mathscr{i}} - m(t_f) \tag{11}$$

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. As a result, the definition of the orbital error vector is as

$$\mathcal{E}(t) = \Big[(\frac{a(t)-a_{\mathscr{d}}}{\sigma_a})^2\ \ (\frac{e(t)-e_{\mathscr{d}}}{\sigma_e})^2\ \ (\frac{i(t)-i_{\mathscr{d}}}{\sigma_i})^2\ \ (\frac{\Omega(t)-\Omega_{\mathscr{d}}}{\sigma_\Omega})^2\ \ (\frac{\omega(t)-\omega_{\mathscr{d}}}{\sigma_\omega})^2\Big] - 1 \tag{12}$$

where $\sigma_{(\cdot)}$ denote the maximum allowable differences 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

$$\mathcal{G} = \begin{cases} \mathcal{E}(t_f), & \text{if } \mathcal{E}(t_f) \leq 0 \\ K_{\mathcal{E}} \dfrac{\mathcal{E}(t_f)}{\mathcal{E}_0} + \dfrac{1}{\mathcal{E}_0 t_f} \displaystyle\int_0^{t_f} \mathcal{E}(t)dt, & \text{otherwise} \end{cases} \tag{13}$$

where $\mathcal{E}_0$ is the error at the initial time. Note that both $\mathcal{E}_0$ and t_f are known for a unique solution. By this definition, the constraint function properly distinguishes the feasible and infeasible transfer trajectories with corresponding penalties. Clearly according to Eq. 12, $\mathcal{E}(t_f) \leq 0$ means that the orbital elements by the end of transfer are within the acceptable error margin. So, the transfer is feasible and the constraint function returns a negative value, proportional to the difference from the desired orbital elements. However, if $\mathcal{E}(t_f) > 0$, the final conditions are not satisfied and the constraint violation is considered to be the summation of the scaled error by the end of transfer and the scaled integral of error. Obviously, two terms of second condition in Eq. 13 have the following boundaries

$$0 < \frac{\mathcal{E}(t_f)}{\mathcal{E}_0} \leq 1 \tag{14}$$

$$0 < \frac{1}{\mathcal{E}_0 t_f} \int_0^{t_f} \mathcal{E}(t)dt \leq 1 \tag{15}$$

Considering Eq. 13 with boundaries of Eq. 14 and Eq. 15, it can be observed that for transfers with same final error, the one with less integral of errors are preferred. Also, the final error has the coefficient $K_{\mathcal{E}}$, which increses the weight of final error. Since tuning this coefficient was not the purpose of this research, this coefficient has been set to $K_{\mathcal{E}} = 10$ in this research to adjust the priority of final error. Note that both $\mathcal{E}(t_f)$ and $\mathcal{G}$ are 1×5 vectors. The developed algorithm is aimed to finding a vector of decision variables in the form of Eq. 10 for minimization of the objective function described by Eq. 11, while satisfying the presented nonlinear constraints $\mathcal{G} \leq 0$ by Eq. 13.

A. Constrained Estimation of Distribution Algorithm

To solve the formed problem, two new adaptive operators for EDA++ is developed and utilized. These operators are associated with two various mechanisms of the algorithm, and they work simultaneously within the optimization process. Before going through the details of the adaptive operators, it is necessary to elaborate the overall structure of EDA++ further. Recalling the initial description about EDAs and EDA++ from Section I, EDAs are a class of optimization algorithms that use statistical models to represent and generate new candidate solutions. In contrast to traditional evolutionary algorithms, EDAs learn a probabilistic model of promising solutions from a population of candidate solutions, and then use this model to generate new candidate solutions for the next iteration. The basic idea behind EDAs is to build a probabilistic model of the promising solutions that have been generated so far, and then use this model to generate new candidate solutions that are likely to be better than the current solutions. The probabilistic model can be any type of statistical model, such as a Bayesian network, a neural network, or a probabilistic graphical model. EDA++ is an advanced evolutionary optimization algorithm, which has been recently developed for solving optimization problems with nonlinear constraints in continuous domain [26]. The compact pseudo code of this algorithm is illustrated in Alg. 1. This modern EDA is equipped with feasibility conserving mechanisms, aiming at rapid discovery of high quality feasible solutions in constrained continuous optimization problems, yet still treats the problem as a black-box with no adaptations. In this subsection, the overall workflow of the original EDA++ is briefly described. However, details of the mechanisms and the optimization process are beyond the scope of this article. Hence, the reader is recommended to refer to [26] for discussions about the development, performance evaluation, and the details of the original algorithm workflow prior to proceeding further.

EDA++ consists of several mechanisms based on probabilistic models, which are executed during the optimization process. Initially the algorithm starts with the seeding mechanism, aimed at exploring the search space for initial feasible solutions. Having the initial feasible solution, most promising individuals are chosen via the selection mechanism. Then, the learning mechanism construct a mixture of probabilistic models based on the population of selected individuals. The mixture model has two types of components, including parent clusters and smart clusters, and each component possesses a special information about the search domain. Several operators exist in this mechanism such as a progressive clustering technique and an outlier detection method for constructing a probabilistic model with high capability of estimating promising solutions. After constructing the probabilistic model, new solutions are generated via the sampling mechanism. The newly obtained solutions are refined via the repairing mechanism and the mapping mechanism to satisfy the boundaries and constraints. Finally, the replacement mechanism combines the newly obtained population with the existing population and extracts the best obtained solution. Each aforementioned mechanism has several parameters, which control the behavior of the algorithm. Changing these parameters acts as balancing between exploration and exploitation capability of the algorithm, while surfing the search domain. In this research some of these parameters are

Algorithm 1: Simplified pseudo code of the non-adaptive constrained EDA [26]

```
Input: Objective function, Constraints function, Boundaries
Settings: Algorithm parameters
Invoke SEEDING mechanism // Generating initial population
Perform EVALUATION // Evaluating the objective values
while stopping criteria are not met do
    Invoke SELECTION mechanism // Selecting high quality solutions
    Invoke LEARNING mechanism // Building probabilistic models
    Invoke SAMPLING mechanism // Sampling new solutions
    Invoke REPAIRING mechanism // Repairing out-of-bound solutions
    Invoke MAPPING mechanism // Mapping infeasible solutions to feasible region
    Perform EVALUATION // Evaluating the objective values
    Invoke REPLACEMENT mechanism // Forming new population
    Perform EXTRACTION // Extracting best solution and updating parameters
Output: Best solution
```

aimed to be adapted by the features of the low-thrust orbit transfer problem.

B. Algorithm Parameters

Kernel density ξ is one of the parameters associated with the sampling mechanism that acts as a balancing threshold for dedicating populations to parent clusters and smart clusters. In each iteration, new solutions are sampled based on the parent clusters Φ and smart clusters ϕ as

$$\mathcal{X}_i = \left\{ \{\mathcal{X}^{\Phi}_{i,j}, \mathcal{X}^{\phi}_{i,k}\} | j \in \{1, ..., \xi N_{\text{pop}}\}, k \in \{1, ..., (1-\xi) N_{\text{pop}}\} \right\} \tag{16}$$

where $\mathcal{X}^{\Phi}$ and $\mathcal{X}^{\phi}$ are the generated solutions associated with parent clusters and smart clusters respectively. 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 fewer number of samples in smart clusters, hence increases the exploitation of the search process.

The smart cluster detection parameter is another parameter associated with the learning mechanism represented by ν. This parameter sets the threshold for separating smart clusters from the parent clusters, and has the typical boundary of $0.01 < \nu < 0.1$. Within the learning mechanism, the potential solutions for smart clusters in each parent cluster are extracted as

$$\mathcal{Y}^{\Phi}_i = \left\{ \mathcal{X}^{\Phi}_{i,j} | j \in \{1, ..., N_{\Phi}\}, |\mathcal{F}^{\Phi *}_i - \mathcal{F}^{\Phi}_{i,j}| < \nu \right\} \tag{17}$$

with Φ and i denote as the parent cluster and the iteration number respectively, $\mathcal{Y}^{\Phi}_i$ are the potential solutions of the parent cluster, $\mathcal{X}^{\Phi}_{i,j}$ are the solutions inside the parent cluster, N_{Φ} is the number of solutions in the parent cluster, $\mathcal{F}^{\Phi *}_i$ is the objective value of the best solution in the parent cluster, and $\mathcal{F}^{\Phi}_{i,j}$ is the objective value associated with the solution $\mathcal{X}^{\Phi}_{i,j}$. High values of ν yields more smart clusters within the mixture model, hence increases the exploration of the search domain.

C. Problem Identifiers

Due to the high complexity of the proposed orbit transfer problem, algorithm parameter selection is nontrivial. The two aforementioned parameters ν and ξ are chosen to be adapted based on the features of the space mission for a given mission parameters $\mathcal{P}$ in Eq. 6. For each unique problem, there are thirteen parameters in $\mathcal{P}$. However, it is more convenient to analyze the structure of the search domain with fewer parameters by considering the differences of the initial an desired orbital elements instead of the their explicit absolute values. Moreover, the thrust to weight ratio has been taken into consideration instead of thrust and mass individually. Following this approach, three parameters referred to as *problem identifiers* are defined as follows

$$\mathcal{A} = \frac{|a_d - a_i|}{6R_e} + |e_d - e_i| \tag{18}$$

$$\mathcal{B} = \frac{|i_d - i_i|}{\pi} + \frac{|\Omega_d - \Omega_i|}{2\pi} + \frac{|\omega_d - \omega_i|}{2\pi} \tag{19}$$

$$C = \frac{T_{\max}}{m_i} \tag{20}$$

where R_e is the mean radius of Earth, and the orbital angles in $\mathcal{B}$ are in radians. As can be appreciated, the three parameters $\mathcal{A}$, $\mathcal{B}$, and C represent different aspects of the space mission. $\mathcal{A}$ represents the desired change in the shape of the orbit, $\mathcal{B}$ represents the desired orientation of the space orbit, and C represents the accessible acceleration for accomplishing the space mission. Based on the presented mathematical model of the problem, space missions with high $\mathcal{A}$ and $\mathcal{B}$ and low C generally produce optimization problems with high dimensions. However, the variation of dimensions due to changes in problem identifiers are nonlinear. 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. Detailed results for this claim are available. However, due to the desire to keep the article concise, they have been left out. Besides the problem identifiers, an augmented cost function is defined as

$$\mathcal{J}(\mathcal{F}, \mathcal{G}) = \begin{cases} \dfrac{\mathcal{F}}{m_i}, & \text{if } \max(\mathcal{G}) \leq 0 \\ \sum \mathcal{G}, & \text{otherwise} \end{cases} \tag{21}$$

where $\mathcal{F}$ and $\mathcal{G}$ are functions for objective and constraints violation as in Eq. 11 and Eq. 12 respectively. Clearly, the range of the augmented cost function is $0 < \mathcal{J} \leq 1$ for feasible trajectories and $1 < \mathcal{J}$ for infeasible trajectories. It should be clarified that the augmented cost function defined here is only used for landscape feature analysis, not the optimization. The optimization algorithm has its own internal mechanisms to deal with objectives and constraints [26].

D. Landscape Features

In order to develop proper adaptive operators, the complexity of the problem needs to be identified. The goal is to discover the effects of problem identifiers $\mathcal{A}$, $\mathcal{B}$, and C on the augmented cost function $\mathcal{J}$, and try to match them with algorithm parameters ν and ξ accordingly. The most common approach for achieving this goal is using FLA techniques in extracting the characteristics of the problem [27]. FLA methods may be used with various metrics for quantifying the problem characteristics. A comprehensive survey by Malan and Engelbrecht [35] introduces a variety of these methods.

One of the most practical metrics for FLA is the *dispersion*, which gives useful information about the structure of the search domain. This metric, first introduced by Lunacek and Whitley [36], measures the average distance between pairs of individuals that are nominated as high quality solutions. It has been originally introduced for continuous optimization problems, and has been used to study the search landscape of several problems [37, 38]. Dispersion extracts the level to which high quality solutions are concentrated in a given problem as

$$\psi = \frac{1}{\zeta n_{\psi}(\zeta n_{\psi} - 1)} \sum_{i=1}^{\zeta n_{\psi}-1} \sum_{j=i+1}^{\zeta n_{\psi}} ||\mathcal{X}_i - \mathcal{X}_j|| \tag{22}$$

where ψ is the dispersion value, ζ is the top percentage of the chosen samples with respect to the augmented cost function $\mathcal{J}$, and n_{ψ} is the total number of selected population. Scaling the distances makes dispersion have a boundary of $0 < \psi < 1$. Having the computational complexity of $O(\zeta {n_{\psi}}^2)$, evolvability and deception of the search space in the proposed orbit transfer problem can be identified with this metric. Evolvability pertains to a candidate solution's capacity to discover an improved solution while conducting a search. Meanwhile, deception pertains to information that steers the search away from the global optimum. They reflect the performances of algorithms on landscapes. When samples approach the search space with most promising solutions (i.e. decrease in ζ), the increase of the dispersion implies a weak global structure, which comes with a higher difficulty for solving the problem, hence more exploration is required. On the other hand, if lowering the threshold of promising solution results in low dispersion, more exploitation is needed for reaching the global optimal solution.

Following the proposed metric and problem identifiers, the landscape of the problem is analyzed in a data set of space missions. To generate a grid-like data set of space missions, first, 1000 different space missions are considered, which have been uniformly generated with respect to the following mission parameters.

$$\begin{aligned}
6600\ km <& a_i, a_d < 42164\ km \\
0 <& e_i, e_d < 0.8 \quad (\text{subject to} \quad r_p > R_e + 200\ km) \\
0 <& i_i, i_d < \pi \\
0 <& \Omega_i, \Omega_d < 2\pi \\
0 <& \omega_i, \omega_d < 2\pi \\
10^{-3}\ N <& T_{\max} < 10\ N \\
300\ kg <& m_i < 2000\ kg \\
1500\ s <& I_{\text{sp}} < 5000\ s
\end{aligned} \tag{23}$$

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 taken 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. Considering various thresholds, the dispersion is computed for each problem via 100 solutions uniformly distributed within the limits of $\mathcal{X}_{\text{min}}$ and $\mathcal{X}_{\text{max}}$ for each 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. 1. For each value of problem identifiers, the mean and standard deviation of the resulting dispersion values are given based on various thresholds.

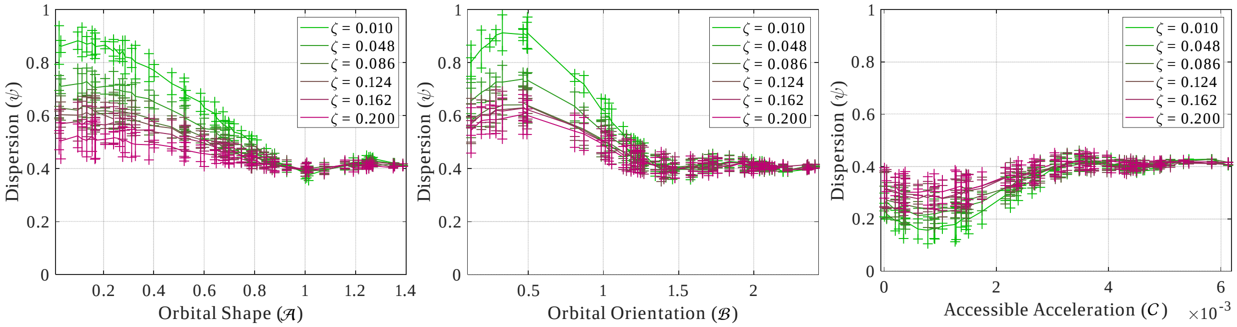

Fig. 1 Normalized dispersion for orbital shape $(\mathcal{A})$, orbital orientation $(\mathcal{B})$, and accessible acceleration $(\mathcal{C})$

According to the normalized dispersion for orbital shape, a relatively large decrease in dispersion can be observed when the problem identifier $\mathcal{A}$ increases. This variation implies that high quality solutions are increasingly closer together in orbit transfers with high changes in semi-major axis and eccentricity. It agrees with the fact that more revolutions, hence more thrust arcs and on-off thrust profiles need to be discovered for orbit transfers when the desired change in orbital parameters becomes high. Same insight can be inferred for the orientation of the space orbit based on $\mathcal{B}$ identifier. Unlike the first two identifiers, the dispersion has an increasing variation for $\mathcal{C}$, which indicates that the solution domain is more chaotic when thrust to mass ratio is low for a given orbit transfer problem.

Analysis of standard deviations in Fig. 1 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 $\frac{1}{\sqrt{6}}$ as problem dimension tends to infinity [38]. Following the analysis of the proposed metric confirms its relative reliability in describing the complexity of the presented finite-thrust trajectory optimization problem.

E. Adaptive Operators

Given a space mission parameter vector $\mathcal{P}$, the proposed landscape metric can be obtained via Eq. 22. The evolution of the dispersion value is monitored with decreasing ζ within the boundary of $0.01 < \zeta < 0.2$. The dispersion difference $\Delta\psi$ with respect to minimum and maximum threshold is used as an indicator to classify the orbit transfer complexity. A negative value of $\Delta\psi$ implies that the best fitness values are localized in a small sub-region of the search space. It means that more exploitation of the search domain is required to extract promising solutions. A $\Delta\psi$ value around 0 corresponds to the fact that the best fitness values are spread over the entire search space. Higher values of $\Delta\psi$ show localized promising solutions in highly distanced regions, hence algorithms with more exploration capabilities are more preferred.

Following the dispersion evolution for a given problem, the adaptive operator for kernel density is defined as

$$\xi = \frac{1}{2} - \frac{4}{5\pi} \tan^{-1}(K_{\xi} \Delta\psi) \tag{24}$$

where K_{ξ} is the kernel density coefficient, which is updated within the optimization process as

$$K_{\xi} = \eta(\frac{1}{2} + 3\lceil \Delta\psi \rceil) \tag{25}$$

with η being the percentage of computational budget that has been used as the optimization goes on, representing the progress of the optimization process. The variation of the proposed kernel density as the function of dispersion evolution is depicted in Fig. 2. As can be observed, the kernel density is adapted based on the dispersion evolution of the orbit transfer problem, which varies according to the degree of chaos of the search space. If the search domain for the trajectory optimization problem of the desired orbit transfer has high dispersion evolution, low values will be assigned to kernel density. This makes an increase in the number of populations within the smart clusters, leading to increase exploration capability of the algorithm. On the other hand, the kernel density is increased for low dispersion evolution, leading the probabilistic model to have more dense parent clusters, i.e. more exploitation within 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 from assigning zero population to either type of clusters. The other observation that can be highlighted is that the

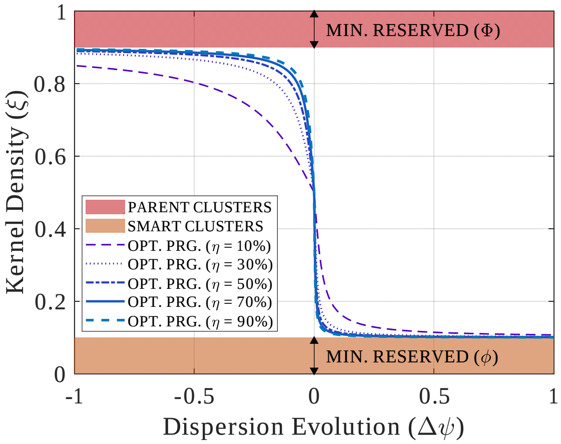

Fig. 2 Adaptive kernel density for assigning clusters' population

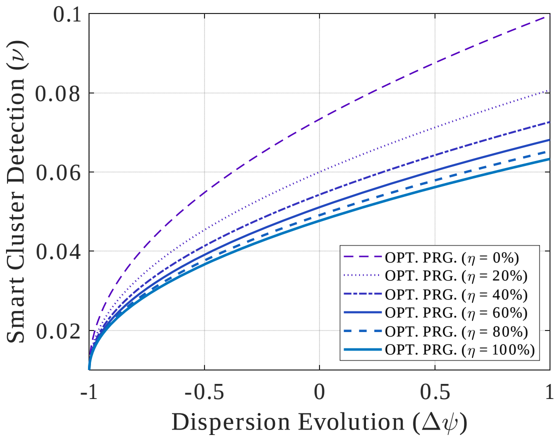

Fig. 3 Adaptive smart cluster detection parameter

kernel density does not have a symmetric variation for $\Delta\psi \geq 0$ and $\Delta\psi \leq 0$. The coefficient defined in Eq. 25 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. Also, for orbit transfer problems with $\Delta\psi \neq 0$, the kernel density changes gradually toward dedicating more populations to parent clusters and smart clusters as the optimization goes on (increase of η). 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 smart cluster detection parameter is defined as

$$\nu = \frac{1 + (3 + K_\nu)\sqrt{1 + \Delta\psi}}{100} \tag{26}$$

with K_ν being the coefficient for smart cluster detection parameter, defined as

$$K_\nu = \frac{1}{\eta + 30} \tag{27}$$

which indicates its nonlinear dependency on the optimization progress. Fig. 3 depicts the range of the adaptive operator being presented. As can be observed, high values of $\Delta\psi$ correspond with more exploration, and the increasing variation of smart cluster detection parameter agrees with it. Also, as the optimization process goes on (increase in η), less sensitivity for high dispersion evolution values are considered.

IV. Numerical Experiment and Analysis

Taking advantage of the two newly developed operators of Eq. 24 and Eq. 26, and implementing them in algorithm parameters of Eq. 16 and Eq. 17 respectively, turns the original constrained EDA into a robust approach, which is capable of being adapted according to the complexity of the given low-thrust trajectory optimization problem. The proposed approach has been applied in several test cases in two experiments. In the first experiment, a non-coplanar perturbed orbit transfer problem is solved 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 and other rival optimization algorithms. The second experiment is dedicated to the performance evaluation of the proposed approach by comparing it with an indirect approach based on continuation technique in a non-perturbed low-thrust orbit transfer problem. The main aim of the second experiment is to determine if the proposed method can be considered an alternative to other techniques for finding optimal transfer trajectories [7].

A. Perturbed Non-Coplanar Orbit Transfer

The first experiment is the J_2 perturbed orbit transfer. Using the proposed algorithm, the kernel density ξ and smart cluster detection parameter ν as presented in Eq. 24 and Eq. 26 are considered within the algorithm, while the rest of the algorithm parameters has been adjusted similar to [26]. The integration time step is set to $50s$ and the arbitrary orbital elements of the initial and final orbits are assumed as in Table 1.

Table 1 Orbital elements of initial and final orbits in the first experiment

	Initial orbit	Final orbit
a	33400 km	31700 km
e	0.55	0.15
i	55 deg	80 deg
Ω	250 deg	165 deg
ω	145 deg	295 deg

In this orbit transfer, the initial mass of the spacecraft is assumed to be $m_{i} = 530kg$, while the specific impulse is considered as $I_{\text{sp}} = 2300s$. The thresholds for targeting desired orbital elements in Eq. 12 are considered as $\sigma_a = 10\ km$, $\sigma_e = 10^{-3}$, $\sigma_i = 10^{-2}$ deg , $\sigma_\Omega = 10^{-1}$ deg , $\sigma_\omega = 10^{-1}$ deg . The problem is solved with five different thrust levels of $T_{\max} = 1N$, $T_{\max} = 1.3N$, $T_{\max} = 1.6N$, $T_{\max} = 1.9N$, and $T_{\max} = 2.2N$. The obtained results are summarized in Table 2, while the transfer trajectories for minimum and maximum thrust levels are illustrated in Fig. 4 and Fig. 5 respectively.

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 80 revolutions within 51.7841 days. As the thrust limit increases, the achieved transfer trajectory corresponds to shorter transfer time and fewer orbital revolutions, ending with transfer time down to 24.3903 days with 38 revolutions for $T_{\max} = 2.2N$. The differences between the values of the desired orbital elements and those associated with the obtained solutions are also provided in Table 2, indicating the feasibility of the transfer trajectory with respect to the desired thresholds for each orbital element defined in Eq. 12. 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 semi-major axis and eccentricity.

Noting that the solution of unperturbed multi-impulse transfer for this problem that has been utilized for the proposed approach corresponds with total velocity change of $\Delta v = 4.5382km/s$ and the final mass of $m(t_f) = 433.3447kg$. Evaluation of the final spacecraft mass for each case shows more fuel consumption for longer transfer trajectories,

Table 2 Characteristics of transfer trajectories for each maximum thrust limit

$T_{\max}$ $[N]$	Rev.	t_f [days]	$E_a[km]$	E_e	$E_i[deg]$	$E_\Omega[deg]$	$E_\omega[deg]$	$m(t_f)$ [kg]
1	80	51.7841	5.2841	9.3252e-4	2.5182e-3	5.0484e-2	1.9108e-2	425.65
1.3	62	39.8344	1.4838	6.4514e-5	3.1055e-3	8.1100e-3	4.8893e-3	427.31
1.6	50	31.9993	1.3084	4.6250e-5	2.7086e-3	2.7771e-2	2.7846e-2	428.12
1.9	42	26.8086	5.4682	1.4954e-5	3.0365e-3	1.3625e-2	3.5475e-2	428.96
2.2	38	24.3903	0.2643	1.6668e-5	1.7841e-3	9.5964e-2	3.2051e-4	429.21

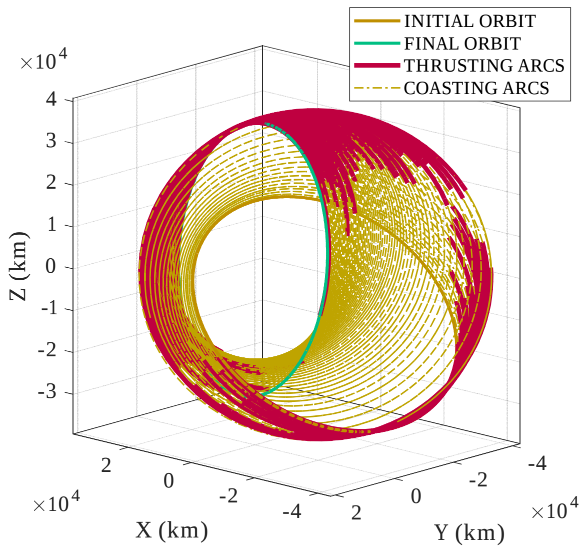

Fig. 4 Minimum-fuel orbit transfer for $T_{\max} = 1N$

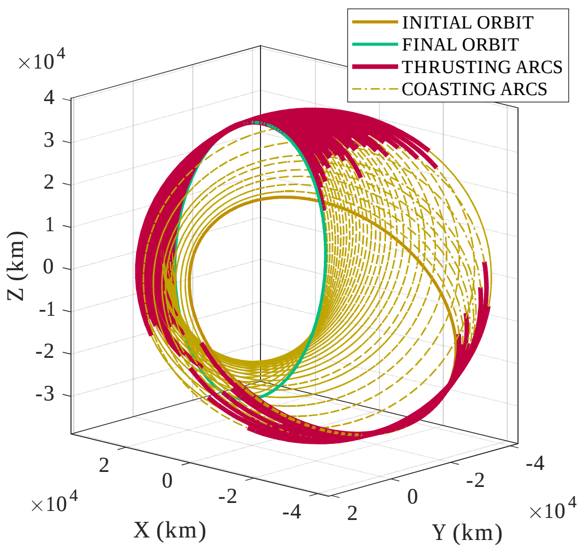

Fig. 5 Minimum-fuel orbit transfer for $T_{\max} = 2.2N$

mainly due to J_2 perturbations. The resulting thrust profile for each case is available. Fig. 6 shows the components of thrust vector for $T_{\max} = 1.6N$ while the time-histories of direction angles $\alpha(t)$ and $\beta(t)$ in four sequential revolutions are depicted in Fig. 7. It can be observed how five nodes interpolation successfully parameterized the steering angles in each thrust arcs. Although, it is possible to consider more discrete nodes to the model, achieving a feasible solution confirms the reasonable choice for the dedicated number of discrete nodes of steering angles parameterization.

Following the presented orbit transfer problem, the performance of the proposed approach is compared with the non-adaptive version of EDA and also other state-of-the-art algorithms for constrained continuous optimization including RL-CORCO [39] and VF-CLCB [40]. The reason for choosing RL-CORCO and VF-CLCB is that they are the most recently developed EAs that have shown to be the most competitive algorithms in terms of execution time and quality of

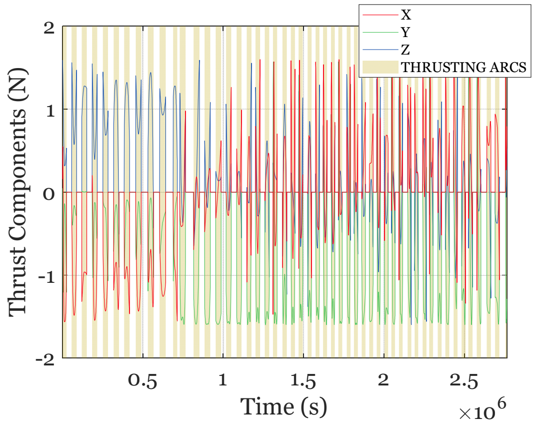

Fig. 6 Components of thrust vector during the orbit transfer for $T_{\max} = 1.6N$

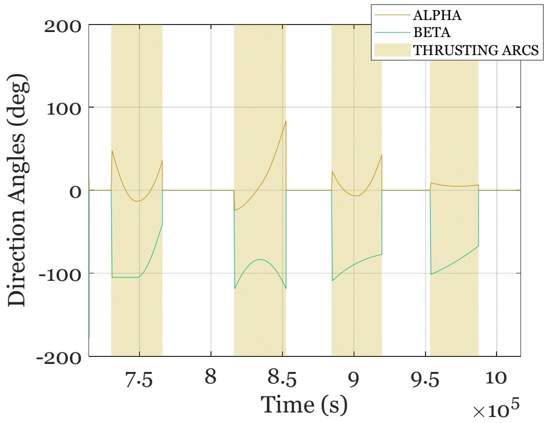

Fig. 7 Variation of thrust direction angles for $T_{\max} = 1.6N$ in revolutions 12 to 15

Table 3 Comparison of the algorithms' performance in 25 runs

	Adaptive approach		EDA++ [26]		RL-CORCO [39]		VF-CLCB [40]	
$T_{\max}$ [N]	FR	RBP	FR	RBP	FR	RBP	FR	RBP
1	76%	$0.000e+00$	52%	$6.445e+00$	56%	$1.413e+01$	28%	$2.397e+01$
1.3	84%	$0.000e+00$	72%	$5.233e+00$	64%	$5.932e+00$	48%	$1.968e+01$
1.6	100%	$0.000e+00$	84%	$1.704e-01$	76%	$1.184e-01$	52%	$7.959e+00$
1.9	100%	$0.000e+00$	88%	$3.432e-06$	96%	$8.148e-05$	68%	$4.285e+00$
2.2	100%	$2.859e-12$	92%	$0.000e+00$	100%	$1.982e-07$	76%	$8.133e-01$

the obtained solutions in comparison to EDA++. For every case of $T_{\max}$, each algorithm is implemented and executed in 25 runs with same computational budget. Table 3 summarizes the achieved results. Two parameters for the measurement of algorithm performance are calculated. The first parameter is the feasibility ratio (FR) of each algorithm, which simply shows the percentage of the total runs in which the algorithm succeeded in finding a feasible transfer trajectory (satisfying the constraints $\mathcal{G} \leq 0$ in Eq. 13) disregarding the amount of fuel consumption (the objective function value $\mathcal{F}$ in Eq. 11). The other parameter is the relative best percentage (RBP), which is calculated as

$$RBP = \min(100 \times \frac{\vec{\mathcal{F}} - \mathcal{F}^*}{\mathcal{F}^*}) \tag{28}$$

where $\vec{\mathcal{F}}$ is the vector of objective values correspond to obtained feasible solutions by the algorithm, and $\mathcal{F}^*$ is the best obtained solution between all algorithms. Clearly the RBP of zero indicates that the algorithm managed to find the best possible solution between the rest of the algorithms, and any non-zero value represents relative difference of the best obtained solution with respect to the global best solution. According to Table 3, the proposed adaptive approach managed to find feasible solutions in all runs for $T_{\max} = 1.6N$, $T_{\max} = 1.9N$, and $T_{\max} = 2.2N$. In this regard, the non-adaptive EDA and RL-CORCO have shown to be competitive, yet with less feasibility ratios. Also, considering the RBP values, it can be verified that the best obtained solution for all cases belongs to the proposed adaptive approach except for $T_{\max} = 2.2N$ case, in which the best obtained solution via the adaptive approach is extremely close to the one obtained via EDA++ with RBP in the order of 10^{-12}. Overall, it can be observed that for lower amount of thrust levels, the advantage of using the adaptive approach is more justified.

B. Comparative Analysis

Second experiment is dedicated to the comparison of the proposed adaptive approach with an indirect approach based on continuation technique [7] in an unperturbed orbit transfer, i.e. $\gamma_p = 0$. It is noteworthy that the selected rival method also takes advantage of multi-impulse orbit transfer solution as an initial guess for the main continuation process in achieving fuel-optimal transfers. Following this fact, it is a noticeable analysis to verify which technique exploits impulsive solutions better in discovering optimal transfer trajectories. The orbital elements of the initial and final orbits are provided in Table 4 [7].

Table 4 Orbital elements of initial and final orbits in the second experiment

	Initial orbit	Final orbit
a	10000 km	10000 km
e	0.2	0.2
i	30 deg	30 deg
Ω	10 deg	30 deg
ω	20 deg	50 deg

The initial spacecraft mass, specific impulse, and the thrust level are considered as $m_i = 100kg$, $I_{\text{sp}} = 3000s$, and $T_{\max} = 10N$ respectively, while the threshold for reaching the final orbital elements and the integration setup have been set according to the reference. The problem is solved via the proposed adaptive approach and the obtained transfer trajectory is illustrated in Fig. 8 , while Table 5 shows the comparison of the obtained solution with the one presented in [7].

The obtained solution corresponds to $m(t_f) = 96.0098kg$, which confirms more optimality of the solution in terms of fuel consumption in comparison to the solution obtained via the indirect continuation technique with $m(t_f) = 95.3489kg$. Not only the solution via the proposed adaptive approach has lower fuel consumption, but also the obtained transfer time is lower. Fig. 9 and Fig. 10 show the time histories of the orbital elements correspond to the obtained solution using the adaptive approach.

As can be appreciated, the final orbit is reached in 474.51 minutes with 4 thrust arcs. However, the transfer trajectory via the continuation technique has 6 thrust arcs in 495.36 minutes. This indicates that the adaptive approach outperforms the other technique in both terms of fuel mass and transfer time.

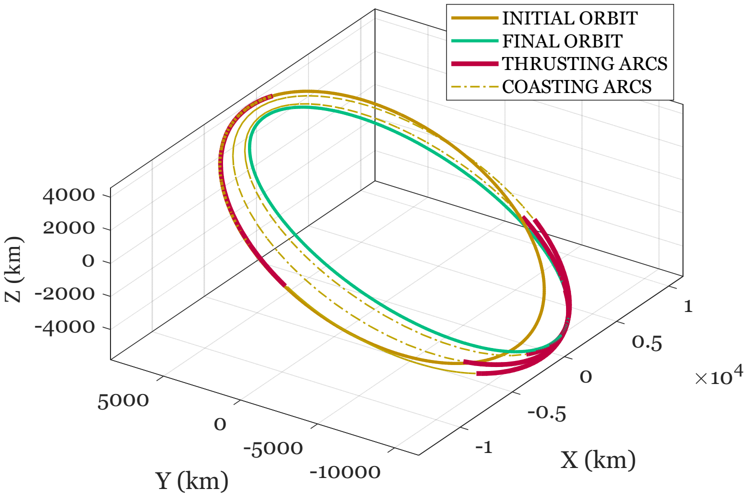

Fig. 8 Optimal finite-thrust maneuver in non-coplanar unperturbed orbit transfer

Table 5 Comparison of the obtained solutions for non-coplanar unperturbed orbit transfer

	$m(t_f)$ [kg]	t_f [min]	Thrust arcs
Direct Adaptive Approach	96.0098	474.51	4
Indirect Continuation Technique [7]	95.3489	495.36	6

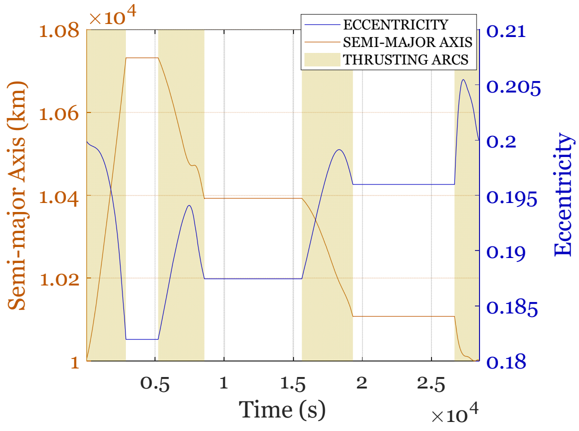

Fig. 9 Time histories of semi-major axis and eccentricity

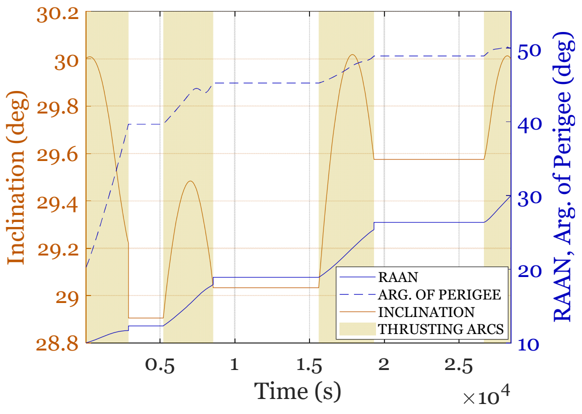

Fig. 10 Time histories of inclination, RAAN, and argument of perigee

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. The implementation and the development of EAs for discovering optimal transfer trajectories have been vastly studied in recent years. However, the existing research shows that there has been little focus on studying the problem's difficulty and how the complexity of the search space relates to optimizing the algorithm. 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 smart cluster detection parameter ν depend on both the optimization progress η and the dispersion variation $\Delta\psi$.

Several points can be highlighted regarding the proposed adaptive approach. One key aspect is the proposed problem identifiers in Eq. 18 to Eq. 20. As presented, the landscape feature analysis in this research is based on orbital shape ($\mathcal{A}$), orbital orientation ($\mathcal{B}$), and accessible acceleration (C). 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 $a_i = 10000km$ to $a_d = 11000km$ and from $a_i = 11000km$ to $a_d = 12000km$ since both have the same value of $\mathcal{A}$ 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 $\mathcal{A}$, $\mathcal{B}$, and C 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 $O(\zeta {n_\psi}^2)$, 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) [41] can be considered as an effective technique for estimating the dispersion value. Although it may produce some errors in estimation of dispersion, deep analysis may show the reliability of utilizing k-NN in reducing the computational burden.

VI. Conclusions

The optimization framework based on the proposed adaptive operators within the heuristic mechanisms of EDAs have shown to be effective in achieving optimal transfer trajectories in low-thrust orbit transfers. Noticeable results were obtained in two conducted experiments, including the comparison of the algorithm performance with the non-adaptive EAs, and with another indirect approach. Results from the first experiment show that the feasibility ratio of the obtained solutions in multiple runs of the algorithm are significantly high in the proposed approach comparing to the non-adaptive version of the algorithm and other rival EAs. The feasibility ratio is 76% and 84% for the thrust levels of $T_{\max} = 1N$ and $T_{\max} = 1.3N$ respectively, and 100% for higher thrust levels. Also, in terms of the quality of the solutions (minimum fuel consumption), the best obtained solution always belongs to the proposed adaptive approach, except for $T_{\max} = 2.2N$ case, where the obtained solution is extremely close to the global best solution (in the order of 10^{-12}), which has been achieved via the non-adaptive version of the algorithm. Comparing the quality of the obtained solutions via the proposed method with the quality of the solutions obtained via an indirect approach based on continuation technique from the literature was the main goal of the second experiment. The comparison shows that the adaptive approach can provide transfer trajectories that are more optimal in terms of fuel consumption. Moreover, since less thrust arcs has been employed by the proposed algorithm to transfer the spacecraft from the initial orbit to the final orbit, the transfer time in the proposed technique is also less in comparison to the indirect approach (474.51 minutes versus 495.36 minutes). In conclusion, the optimization approach via the proposed framework outperforms the indirect method in both terms of fuel mass and transfer time.

Investigating the intricacy of low-thrust orbit transfer problems using various direct or indirect techniques could be a beneficial pursuit. For instance, the low-thrust orbit transfer problem has been solved in [16] using an approach that relies on reinforcement learning and Lyapunov-based control laws. In this approach, the problem is transformed into a black-box with the controller's weighting coefficients being modeled via cubic splines during the orbit transfer. The decision vector includes discrete nodes of weighting coefficients, and an optimizer, PSO, is used to search for the desired minimum-fuel orbit transfer. An area of interest is to evaluate the complexity of the search space in this problem using FLA techniques and develop adaptive operators for the optimizer to achieve higher quality solutions than previously obtained. These, along with other potential improvements and enhancements discussed earlier, are additional subjects for future research.

Funding Sources

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

[1] Edelbaum, T. N., "Optimal space trajectories," Tech. rep., ANALYTICAL MECHANICS ASSOCIATES INC JERICHO NY, 1969.

[2] Vinh, N. X., "General theory of optimal trajectory for rocket flight in a resisting medium," *Journal of Optimization Theory and Applications*, Vol. 11, No. 2, 1973, pp. 189–202. https://doi.org/10.1007/bf00935883.

[3] Miele, A., "General variational theory of the flight paths of rocket-powered aircraft, missiles and satellite carriers," *IXth International Astronautical Congress/IX. Internationaler Astronautischer Kongress/IXe Congrès International D'astronautique*, Springer, 1959, pp. 946–970. https://doi.org/10.1007/978-3-7091-4745-0_32.

[4] Shirazi, A., Ceberio, J., and Lozano, J. A., "Spacecraft trajectory optimization: A review of models, objectives, approaches and solutions," *Progress in Aerospace Sciences*, Vol. 102, 2018, pp. 76–98. https://doi.org/10.1016/j.paerosci.2018.07.007.

[5] Mall, K., and Taheri, E., "Three-Degree-of-Freedom Hypersonic Reentry Trajectory Optimization Using an Advanced Indirect Method," *Journal of Spacecraft and Rockets*, 2022, pp. 1–12. https://doi.org/10.2514/1.A34893.

[6] Ottesen, D., and Russell, R. P., "Piecewise Sundman Transformation for Spacecraft Trajectory Optimization Using Many Embedded Lambert Problems," *Journal of Spacecraft and Rockets*, 2022, pp. 1–18. https://doi.org/10.2514/1.A35140.

[7] Zhu, Z., Gan, Q., Yang, X., and Gao, Y., "Solving fuel-optimal low-thrust orbital transfers with bang-bang control using a novel continuation technique," *Acta Astronautica*, Vol. 137, 2017, pp. 98–113. https://doi.org/10.1016/j.actaastro.2017.03.032.

[8] Wu, X., Wang, Y., and Xu, M., "Trajectory optimization and maintenance for ascending from the surface of Phobos," *Advances in Space Research*, Vol. 68, No. 8, 2021, pp. 3191–3204. https://doi.org/10.1016/j.asr.2021.06.026.

[9] Choi, J. H., Lee, J., and Park, C., "Deep-space trajectory optimizations using differential evolution with self-learning," *Acta Astronautica*, Vol. 191, 2022, pp. 258–269. https://doi.org/10.1016/j.actaastro.2021.11.014.

[10] Chen, S., Jiang, F., Li, H., and Baoyin, H., "Optimization for multitarget, multispacecraft impulsive rendezvous considering J 2 perturbation," *Journal of Guidance, Control, and Dynamics*, Vol. 44, No. 10, 2021, pp. 1811–1822. https://doi.org/10.2514/1.G005602.

[11] Zhou, H., Wang, X., and Cui, N., "Fuel-optimal multi-impulse orbit transfer using a hybrid optimization method," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 21, No. 4, 2019, pp. 1359–1368. https://doi.org/10.1109/TITS.2019.2905586.

[12] Caruso, A., Quarta, A. A., and Mengali, G., "Optimal transfer between elliptic orbits with three tangential impulses," *Advances in Space Research*, Vol. 64, No. 4, 2019, pp. 861–873. https://doi.org/10.1016/j.asr.2019.05.037.

[13] Ellithy, A., Abdelkhalik, O., and Englander, J., "Multi-Objective Hidden Genes Genetic Algorithm for Multigravity-Assist Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, 2022, pp. 1–17. https://doi.org/10.2514/1.G006415.

[14] Chai, R., Tsourdos, A., Savvaris, A., Chai, S., and Xia, Y., "Solving constrained trajectory planning problems using biased particle swarm optimization," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 57, No. 3, 2021, pp. 1685–1701. https://doi.org/10.1109/TAES.2021.3050645.

[15] Feng, W., Han, L., Shi, L., Zhao, D., and Yang, K., "Optimal control for a cooperative rendezvous between two spacecraft from determined orbits," *The Journal of the Astronautical Sciences*, Vol. 63, No. 1, 2016, pp. 23–46. https://doi.org/10.1007/s40295-015-0079-4.

[16] Holt, H., Armellin, R., Baresi, N., Hashida, Y., Turconi, A., Scorsoglio, A., and Furfaro, R., "Optimal Q-laws via reinforcement learning with guaranteed stability," *Acta Astronautica*, Vol. 187, 2021, pp. 511–528. https://doi.org/10.1016/j.actaastro.2021.07.010.

[17] Larrañaga, P., and Lozano, J. A., *Estimation of distribution algorithms: A new tool for evolutionary computation*, Vol. 2, Springer Science & Business Media, 2001. https://doi.org/10.1007/978-1-4615-1539-5.

[18] Mühlenbein, H., and Paass, G., "From recombination of genes to the estimation of distributions I. Binary parameters," *International conference on parallel problem solving from nature*, Springer, 1996, pp. 178–187. https://doi.org/10.1007/3-540-61723-X_982.

[19] Vasile, M., *Optimization Under Uncertainty with Applications to Aerospace Engineering*, Springer, 2021. https://doi.org/10.1007/978-3-030-60166-9.

[20] Minisci, E., and Avanzini, G., "Comparative study on the application of evolutionary optimization techniques to orbit transfer manoeuvres," *59th International Astronautical Congress*, 2008, pp. 1–15.

[21] Xu, B., Sun, F., Liu, Y., and Wang, P., "Task allocation for multi-spacecraft cooperation based on estimation of distribution algorithm," *2011 8th Asian Control Conference (ASCC)*, IEEE, 2011, pp. 1381–1386.

[22] Champasak, P., Panagant, N., Pholdee, N., Bureerat, S., and Yildiz, A. R., "Self-adaptive many-objective meta-heuristic based on decomposition for many-objective conceptual design of a fixed wing unmanned aerial vehicle," *Aerospace Science and Technology*, Vol. 100, 2020, p. 105783. https://doi.org/10.1016/j.ast.2020.105783.

[23] Chen, Y., Mahalec, V., Chen, Y., Liu, X., He, R., and Sun, K., "Reconfiguration of satellite orbit for cooperative observation using variable-size multi-objective differential evolution," *European Journal of Operational Research*, Vol. 242, No. 1, 2015, pp. 10–20. https://doi.org/10.1016/j.ejor.2014.09.025.

[24] Marchetti, F., Minisci, E., and Riccardi, A., "Single-stage to orbit ascent trajectory optimisation with reliable evolutionary initial guess," *Optimization and Engineering*, 2021, pp. 1–26. https://doi.org/10.1007/s11081-021-09698-w.

[25] Li, X.-R., Wang, X., and Xiong, Y.-q., "A combination method using evolutionary algorithms in initial orbit determination for too short arc," *Advances in Space Research*, Vol. 63, No. 2, 2019, pp. 999–1006. https://doi.org/10.1016/j.asr.2018.08.036.

[26] Shirazi, A., Ceberio, J., and Lozano, J. A., "EDA++: Estimation of Distribution Algorithms with Feasibility Conserving Mechanisms for Constrained Continuous Optimization," *IEEE Transactions on Evolutionary Computation*, 2022. https://doi.org/10.1109/TEVC.2022.3153933.

[27] Malan, K. M., "A survey of advances in landscape analysis for optimisation," *Algorithms*, Vol. 14, No. 2, 2021, p. 40. https://doi.org/10.3390/a14020040.

[28] Choi, J. H., and Park, C., "Spacecraft Trajectory Optimizations: Metrics for Fitness Landscape Analysis," *AIAA SCITECH 2022 Forum*, 2022, p. 1891. https://doi.org/10.2514/6.2022-1891.

[29] Izzo, D., Vinkó, T., and del Rey Zapatero, M., "GTOP Database: Global Optimisation Trajectory Problems and Solutions," , 2010.

[30] Hussain, K., Salleh, M. N. M., Cheng, S., and Shi, Y., "On the exploration and exploitation in popular swarm-based metaheuristic algorithms," *Neural Computing and Applications*, Vol. 31, 2019, pp. 7665–7683. https://doi.org/10.1007/s00521-018-3592-0.

[31] Hintz, G. R., "Survey of orbit element sets," *Journal of guidance, control, and dynamics*, Vol. 31, No. 3, 2008, pp. 785–790. https://doi.org/10.2514/1.32237.

[32] Ayyanathan, P. J., and Taheri, E., "Mapped adjoint control transformation method for low-thrust trajectory design," *Acta Astronautica*, Vol. 193, 2022, pp. 418–431. https://doi.org/10.1016/j.actaastro.2021.12.019.

[33] Curtis, H., *Orbital Mechanics for Engineering Students: Revised Reprint*, Butterworth-Heinemann, 2020. https://doi.org/10.1016/C2016-0-02107-1.

[34] Shirazi, A., Ceberio, J., and Lozano, J. A., "An evolutionary discretized Lambert approach for optimal long-range rendezvous considering impulse limit," *Aerospace Science and Technology*, Vol. 94, 2019, p. 105400. https://doi.org/10.1016/j.ast.2019.105400.

[35] Malan, K. M., and Engelbrecht, A. P., "Fitness landscape analysis for metaheuristic performance prediction," *Recent advances in the theory and application of fitness landscapes*, Springer, 2014, pp. 103–132. https://doi.org/10.1007/978-3-642-41888-4_4.

[36] Lunacek, M., and Whitley, D., "The dispersion metric and the CMA evolution strategy," *Proceedings of the 8th annual conference on Genetic and evolutionary computation*, 2006, pp. 477–484. https://doi.org/10.1145/1143997.1144085.

[37] Muñoz, M. A., Kirley, M., and Halgamuge, S. K., "A meta-learning prediction model of algorithm performance for continuous optimization problems," *International Conference on Parallel Problem Solving from Nature*, Springer, 2012, pp. 226–235. https://doi.org/10.1007/978-3-642-32937-1_23.

[38] Morgan, R., and Gallagher, M., "Sampling techniques and distance metrics in high dimensional continuous landscape analysis: Limitations and improvements," *IEEE Transactions on Evolutionary Computation*, Vol. 18, No. 3, 2013, pp. 456–461. https://doi.org/10.1109/TEVC.2013.2281521.

[39] Hu, Z., and Gong, W., "Constrained evolutionary optimization based on reinforcement learning using the objective function and constraints," *Knowledge-Based Systems*, Vol. 237, 2022, p. 107731. https://doi.org/10.1016/j.knosys.2021.107731.

[40] Cheng, J., Lin, Q., and Yi, J., "An enhanced variable-fidelity optimization approach for constrained optimization problems and its parallelization," *Structural and Multidisciplinary Optimization*, Vol. 65, No. 7, 2022, pp. 1–21. https://doi.org/10.1007/s00158-022-03283-0.

[41] Sen, P. C., Hajra, M., and Ghosh, M., "Supervised classification algorithms in machine learning: A survey and review," *Emerging technology in modelling and graphics*, Springer, 2020, pp. 99–111. https://doi.org/10.1007/978-981-13-7403-6_11.