Hindawi
International Journal of Aerospace Engineering
Volume 2020, Article ID 9437378, 16 pages
https://doi.org/10.1155/2020/9437378

Research Article

Optimization of Interplanetary Trajectories Using the Colliding Bodies Optimization Algorithm

Marco Del Monte, Raffaele Meles, and Christian Circi

Department of Astronautical, Electrical and Energy Engineering, Sapienza University of Rome, Via Salaria 851, 00138 Rome, Italy

Correspondence should be addressed to Christian Circi; christian.circi@uniroma1.it

Received 31 July 2019; Revised 8 October 2019; Accepted 28 October 2019; Published 3 January 2020

Academic Editor: Joseph Morlier

In this paper, a recent physics-based metaheuristic algorithm, the Colliding Bodies Optimization (CBO), already employed to solve problems in civil and mechanical engineering, is proposed for the optimization of interplanetary trajectories by using both indirect and direct approaches. The CBO has an extremely simple formulation and does not depend on any initial conditions. To test the performances of the algorithm, missions with remarkably different orbital transfer energies are considered: from the simple planar case, as the Earth-Mars orbital transfer, to more energetic ones, like a rendezvous with the asteroid Pallas.

1. Introduction

An important aspect of a space mission design is the obtaining of the nominal optimal trajectory, traditionally the one with minimum transfer time or maximum payload mass. Trajectory optimization is an old problem, its origins date back to the ancient Greeks, but its rigorous mathematical formulation, as an optimal control problem, arrives only with Pontryagin in the mid-1900s [1]. Optimal control is an issue concerning the determination of the inputs into a dynamical system that optimize (i.e., minimize or maximize) a specified performance index while satisfying several constraints [2]. These constraints can be differential, as the equations of motion, or algebraic, as departure, mid-course, and arrival constraints. Because of the complexity of most applications, optimal control problems are chiefly solved numerically. Numerical methods adopted are divided into two major classes: indirect methods and direct methods. In the former, the original problem is transcribed into a multiple-point boundary-value problem that is solved to determine candidate optimal trajectories, and the optimization phase consists in finding the optimal set of costate variables. In a direct method, the state and/or control of the optimal control problem is discretized, and the problem is transcribed into a nonlinear optimization problem [2]. Both approaches lead to a parametric optimization where a set of optimal parameters must be found. This optimization has often been conducted by means of gradient-based research (e.g., Newton-Raphson-based algorithms), but in the last decades, a new kind of optimization procedures has been proposed and developed: metaheuristics [3, 4]. The goal of metaheuristics is to efficiently explore the search space looking for near-optimal solutions. The fundamental characteristics are that they are problem independent and they possess a nondeterministic nature, useful to escape from local optima. This is achieved by either allowing worsening moves or generating new starting solutions for the local search in a more "intelligent" way than just providing random initial solutions. This stochastic nature is not employed blindly, but in an intelligent, biased manner, and is what truly differentiates them from gradient-based techniques, which are deterministic and strongly dependent on an initial guess of the solution [5, 6]. Gradient-based algorithms are largely employed in all fields of engineering, including space trajectories optimization. However, in recent years, metaheuristic algorithms have been increasingly adopted, especially in preliminary

analysis of trajectories. Two of the most used algorithms, in this domain, are Particle Swarm Optimization [7, 8] and Differential Evolution [9, 10].

This paper is focused on the optimization of space trajectories using the Colliding Bodies Optimization algorithm. This is a novel population-based metaheuristic inspired by the one-dimensional collision theory between bodies, where each candidate solution being considered as a body with mass. CBO utilizes a simple formulation to find extremals of functions and does not depend on any internal parameter [11]. In Section 2, CBO and its enhanced version will be described briefly. Test cases using indirect methods are studied in Section 3, while those using direct methods are studied in Section 4. Conclusions will be given in Section 5.

2. Colliding Bodies Optimization Algorithm

Colliding Bodies Optimization is a metaheuristic algorithm developed by Kaveh and Mahdavi [11–13], inspired by the one-dimensional collision theory. There are two versions of this algorithm, a basic one [11] and an enhanced one [12], that improves the basic version by means of a sort of Elitism and Crossover. In the next two sections, some basic statements are reported while details can be found in the cited references.

2.1. Basic CBO Formulation. Each search agent is modelled as a body with mass and velocity. The initial position of the ith body is randomly provided in a j-dimensional search space set by the user:

$$x_{ij} = x_{j,\min} + \text{rand} \cdot \left(x_{j,\max} - x_{j,\min}\right), \tag{1}$$

where rand is a random number between 0 and 1. A collision occurs between two bodies, and their positions, after the impact, are updated based on the one-dimensional collision laws [11, 13]. Given the body X_k (also called particle or object), its mass is defined as follows:

$$m_k = \frac{1/J_k}{1/\sum_{i=1}^{n}(1/J_i)}, \quad k = 1, \cdots, n, \tag{2}$$

where J_k is the cost function value of the kth particle and n, which must be an even number, is the total number of bodies used in the optimization process (the population size). The n colliding bodies (CBs) are sorted into ascending order, according to their objective function values, and then divided into two equal groups: Stationary Objects (the lower half) and Moving Objects (the upper half). Objects of the MO group collide against members of the SO group to improve their position and push stationary objects towards better positions. In particular, the colliding pairs are established according to the ascending order with respect to the objective function. Hence, for instance, the best moving particle collides with the best stationary one. Bodies' velocities before the collision are assigned as follows:

$$\text{Stationary bodies} : v_i = 0, \quad i = 1, \cdots, \frac{n}{2}, \tag{3}$$

$$\text{Moving bodies} : v_i = x_{i-(n/2)} - x_i, \quad i = \frac{n}{2} + 1, \cdots, n. \tag{4}$$

As many other metaheuristic algorithms, velocities are not defined as the derivative of the position with respect to time, but they are expressed as displacements in the search space. According to the colliding bodies' theory, velocities after the collision are calculated as follows:

$$\text{Stationary bodies} : v_i' = \frac{\left(m_{i+(n/2)} + \varepsilon m_{i+(n/2)}\right) v_{i+(n/2)}}{m_i + m_{i+(n/2)}}, \quad i = 1, \cdots, \frac{n}{2}, \tag{5}$$

$$\text{Moving bodies} : v_i' = \frac{\left(m_i - \varepsilon m_{i-(n/2)}\right) v_i}{m_i + m_{i-(n/2)}}, \quad i = \frac{n}{2} + 1, \cdots, n, \tag{6}$$

where ε is the *Coefficient of Restitution*, defined as the ratio of the relative velocity between two bodies after and before the collision:

$$\varepsilon = \frac{\left|v_{i+1}' - v_i'\right|}{\left|v_{i+1} - v_i\right|}. \tag{7}$$

This coefficient is assumed varying linearly between 1 and 0 during the optimization process, in order to ensure the balance between exploration and exploitation. After the calculation of the displacement, it is possible to determine new positions of the stationary and moving bodies as follows:

$$\text{Stationary bodies} : x_i^{\text{new}} = x_i + \text{rand} \cdot v_i', \quad i = 1, \cdots, \frac{n}{2}, \tag{8}$$

$$\text{Moving bodies} : x_i^{\text{new}} = x_{i-(n/2)} + \text{rand} \cdot v_i', \quad i = \frac{n}{2} + 1, \cdots, n, \tag{9}$$

where rand is a uniformly distributed random vector in the range [-1,1]. This iterative scheme, performed on all the particles at each iteration, is repeated until a given stopping criterion is fulfilled. A Pseudocode 1 of CBO is reported.

2.2. Enhanced CBO Formulation. The structure of the enhanced CBO (ECBO) algorithm is essentially the same as the basic CBO [12], with the difference that a Colliding Memory (CM) is introduced to save the best CBs' positions obtained so far. In fact, the positions stored in the Colliding Memory substitute the worst positions occupied by the current bodies. In this way, the best positions are remembered and there is no global worsening of the objective function from one iteration to another. The number of the best CBs' positions that are preserved, therefore the dimension of the Colliding Memory, is set by the user. Moreover, after the update of the CBs'

1. Initialize the CBO population in the search space (Equation (1))
2. Evaluate the objective functions and define the masses as in Equation (2)
3. Sort the population in order to identify stationary and moving groups and calculate the velocities as in Equations (3) and (4)
4. Calculate the velocity after the collisions by means of Equations (5) and (6)
5. The new positions can be determined by Equations (8) and (9)
6. If the terminating criterion is fulfilled, proceed to step 7; otherwise, go to step 2
7. Report the best solution found by the algorithm
8. END

Pseudocode 1

1. Initialize the CBO population in the search space (Equation (1))
2. Evaluate the objective functions and define the masses as in Equation (2)
3. Update the Colliding Memory (CM) and population
4. Sort the population in order to identify stationary and moving groups and calculate the velocities as in Equations (3) and (4)
5. Calculate the velocity after the collisions by means of Equations (5) and (6)
6. The new positions can be determined by Equations (8), (9), and (10)
7. If the terminating criterion is fulfilled, proceed to step 8; otherwise, go to step 2
8. Report the best solution found by the algorithm
9. END

Pseudocode 2

positions, the ECBO executes the following crossover instruction:

$$\begin{cases} \text{if } \text{ran}_i < \text{PRO}, \\ x_{ij} = x_{j,\min} + \text{rand}\left(x_{j,\max} - x_{j,\min}\right), \\ \text{otherwise}, \\ x_{ij} = x_{ij}, \end{cases} \tag{10}$$

where x_{ij} is the jth variable of the ith CB, randomly selected; $x_{j,\min}, x_{j,\max}$ are the lower and the upper bounds of the jth variable; PRO is the crossover probability that must be set by the user between [0,1]; ran_i is a random number uniformly distributed within [0,1], automatically generated for each particle, as well as rand. If ran_i is less than PRO, a crossover occurs. As PRO increases, the probability to perform a crossover increases. The Pseudocode 2 of the ECBO is reported.

3. Numerical Simulations: Indirect Methods

In order to analyze the performances of this optimization algorithm, a total of five study cases will be presented. Two of them are solved by using an indirect strategy, while the remaining three cases adopt a direct approach.

3.1. Optimal Earth to Mars Orbital Transfer. The problem is to reach the orbit of Mars departing from the Earth's orbit with the minimum transfer time by using a low thrust engine. This case has already been studied in literature with different techniques [7, 14, 15]. In this paper, the best trajectory is obtained by using the ECBO. As in the cited papers, the following hypotheses are established:

(1) The orbits of the planets are coplanar and circular

(2) The only attracting body is the Sun

(3) The spacecraft's initial position and velocity are the same as the Earth's

(4) The thrust magnitude of the spacecraft is constant

The spacecraft's equations of motion are written in polar coordinates:

$$\begin{cases} \dot{r} = v_r, \\ \dot{v}_r = -\dfrac{\mu_s - r v_\theta^2}{r^2} + \dfrac{T}{m} \sin\alpha, \\ \dot{v}_\theta = -\dfrac{v_r\, v_\theta}{r} + \dfrac{T}{m} \cos\alpha, \end{cases} \tag{11}$$

where r is the position vector; v_r and v_θ are, respectively, the radial and the horizontal velocity of the spacecraft, μ_s is the Sun's gravitational parameter, T is the thrust magnitude; m is the spacecraft mass, and α is the thrust pointing angle relative to the local horizontal (Figure 1).

The control vector is $u(t) = \alpha$. The thrust-to-mass ratio is as follows [7]:

$$\frac{T}{m} = \frac{T}{m_0 - (T/c)(t - t_0)} = \frac{c n_0}{c - n_0 (t - t_0)}, \tag{12}$$

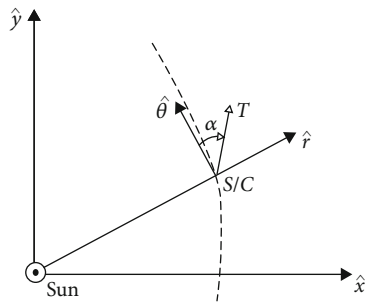

Figure 1: Thrust pointing angle α.

where m_0 is the initial spacecraft mass, $c = 55.894$ km/s is the exhaust velocity, and $n_0 = 8.342 \times 10^{-4}$m/s^2 is the thrust-to-mass ratio at the initial time t_0. A normalized set of units is as follows:

(i) Distance unit DU $= 149.5 \cdot 10^6$ km is the mean radius of the Earth's orbit

(ii) Time unit TU $= 5.018 \cdot 10^6$ s, such that $\mu_s = \text{DU}^3/\text{TU}^2$

The objective function to minimize is $J = t_f$. The desired terminal conditions are expressed by the vector $\boldsymbol{\omega}$ as follows:

$$\boldsymbol{\omega} = \begin{Bmatrix} \Delta r \\ \Delta v_r \\ \Delta v_\theta \end{Bmatrix} = \begin{Bmatrix} r(t_f) - R_{\text{Mars}} \\ v_r(t_f) \\ v_\theta(t_f) - \sqrt{\dfrac{\mu_s}{R_{\text{Mars}}}} \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix}. \tag{13}$$

To write the necessary conditions for optimality, the Hamiltonian function has to be defined:

$$\begin{aligned} H = {} & p_r v_r + p_{v_r}\left(-\frac{\mu_s - r{v_\theta}^2}{r^2} + \frac{T}{m}\sin u\right) \\ & + p_{v_\theta}\left(-\frac{v_r\, v_\theta}{r} + \frac{T}{m}\cos u\right), \end{aligned} \tag{14}$$

where the time-dependent set of costate variables $[p_r, p_{v_r}, p_{v_\theta}]$ has been introduced. Furthermore, the costate differential equations are expressed as follows:

$$\begin{cases} \dot{p}_r = \dfrac{v_\theta^2 p_{v_r} - v_r\, v_\theta\, p_{v_\theta}}{r^2} - \dfrac{2\mu_s p_{v_r}}{r^3}, \\ \dot{p}_{v_r} = -p_r + \dfrac{v_\theta\, p_{v_\theta}}{r}, \\ \dot{p}_{v_\theta} = \dfrac{-2 v_\theta\, p_{v_r} + v_r\, p_{v_\theta}}{r}. \end{cases} \tag{15}$$

From the Pontryagin principle, the optimal control law is as follows:

$$\begin{cases} \cos u_{\text{opt}} = -\dfrac{p_{v_\theta}}{\sqrt{p_{v_\theta}^2 + p_{v_r}^2}}, \\ \sin u_{\text{opt}} = -\dfrac{p_{v_r}}{\sqrt{p_{v_\theta}^2 + p_{v_r}^2}}. \end{cases} \tag{16}$$

The set of necessary conditions for optimality can be completed with the transversality condition referred to the final time:

$$H(t_f) = -\frac{\partial J}{\partial t_f} - \boldsymbol{\lambda} \cdot \frac{\partial \boldsymbol{\omega}}{\partial t_f}, \tag{17}$$

where $\boldsymbol{\lambda}$ is another set of adjoint variables concerning the terminal conditions; hence, it has the same dimensions as $\boldsymbol{\omega}$. Rearranging Equation (17), the following condition is obtained:

$$\frac{c n_0}{c - n_0 t_f}\sqrt{p_{v_r}(t_f)^2 + p_{v_\theta}(t_f)^2} - 1 = 0. \tag{18}$$

In order to find the optimal trajectory that satisfies the necessary conditions, the problem reduces to the determination of four parameters: $[p_r(0), p_{v_r}(0), p_{v_\theta}(0), t_f]$. The CBO algorithm must find the solution exploring the following search space:

$$\begin{aligned} & 1\,\text{TU} \le t_f \le 10\,\text{TU}, \\ & -1 \le p_r(0), p_{v_r}(0), p_{v_\theta}(0) \le 1. \end{aligned} \tag{19}$$

It is necessary to introduce a cost function that links the optimization algorithm to the problem. This function always includes the quantity that must be minimized, and if there is any constraint to respect, the most popular approach is to include them in the cost function. Then, the objective function is defined here as follows:

$$J = t_f + 100 \cdot \Delta r + 100 \cdot \Delta v_r + 100 \cdot \Delta v_\theta. \tag{20}$$

In the cost function, all the quantities (transfer duration and errors at final time) are dimensionless and the weights are chosen to scale them properly, in order to keep the balance between all terms throughout the simulation. The ECBO population is composed of 50 particles. The dimension of the Colliding Memory is set to 1 and crossover is not performed. The algorithm has to find the optimal set of costate initial values. The transversality condition can be neglected by the CBO, due to the homogeneity of the costate equations (Equation (15)). For this reason, if the CBO finds a set of initial costate variables that is proportional to the optimal one ($\vec{p}(0) = b\,\vec{p}(0)_{\text{opt}}$), the same proportionality holds at any time. By writing the control as a function of the proportional set, it is possible to demonstrate that it coincides with the

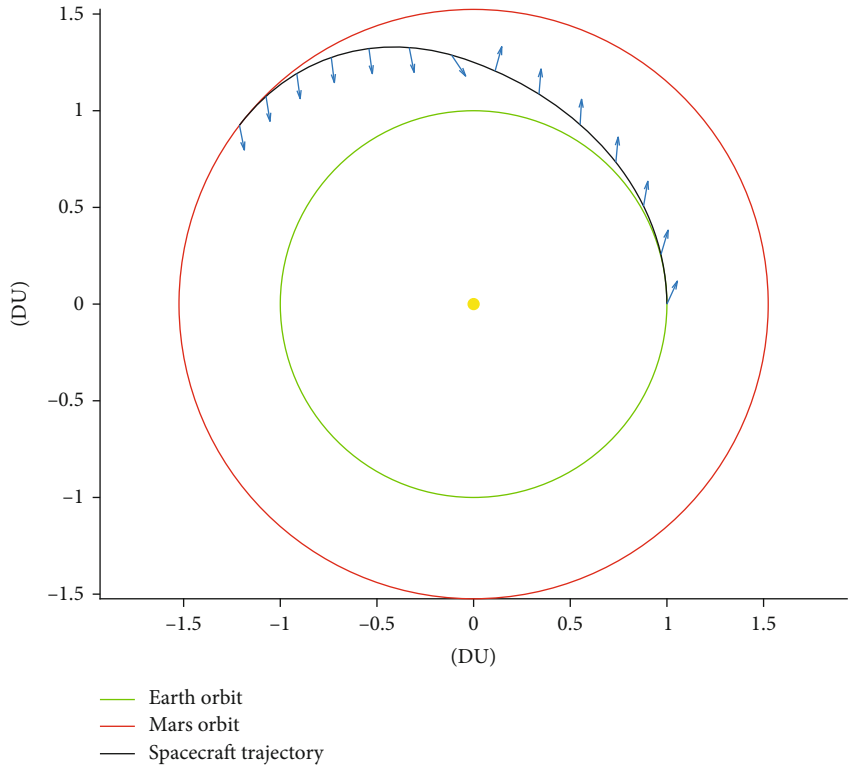

Figure 2: Earth-Mars optimal transfer trajectory.

optimal control (Equation (16)). This means that the minimum time trajectory can be obtained also with an initial costate proportional to the optimal set. Nevertheless, the transversality condition will be violated, and by substituting the proportional set in Equation (18), it can be easily proved that $H(t_{f_{\text{opt}}}) = -b$. In order not to increase the number of equality constraints in the cost function, the transversality condition is verified at the end of the optimization process, to guarantee the optimality of the trajectory.

The optimal solution found by the CBO consists in a 192.6 days transfer (Figure 2).

Figure 3 shows the optimal thrust pointing angle $\alpha(t)$ and the costate evolution during the transfer trajectory. These results are in accordance with those in literature by using PSO [7] and are obtained quickly and easily thanks to the simplicity of the algorithm.

3.2. Optimal Earth to Mercury Orbital Transfer. A minimum time transfer between the orbits of the Earth and Mercury is studied by means of a solar sail. The problem consists in determining the optimal steering law $\alpha(t)$ that minimizes the time of flight to reach Mercury's orbit [16].

As in the cited papers, a series of simplifying assumptions are made:

(1) The relative orbital inclination of Mercury and Earth is neglected, and the orbits are considered circular

(2) The spacecraft's initial position and velocity are the same as the Earth's

(3) The only attracting body is the Sun

A polar inertial reference frame is used, and the equations of motion for the solar sail are as follows:

$$\begin{cases} \dot{r} = v_r, \\ \dot{\theta} = \dfrac{v_\theta}{r}, \\ \dot{v}_r = \dfrac{v_\theta^2}{r} - \dfrac{\mu_s}{r^2} + \dfrac{a_c}{b_1 + b_2 + b_3}\left(\dfrac{R_{\text{Earth}}}{r}\right)^2 \cdot \cos\alpha\left(b_1 + b_2\ \cos^2\alpha + b_3 \cos\alpha\right), \\ \dot{v}_\theta = -\dfrac{v_r\, v_\theta}{r} + \dfrac{a_c}{b_1 + b_2 + b_3}\left(\dfrac{R_{\text{Earth}}}{r}\right)^2 \cdot \sin\alpha \cos\alpha\,(b_2 \cos\alpha + b_3), \end{cases} \tag{21}$$

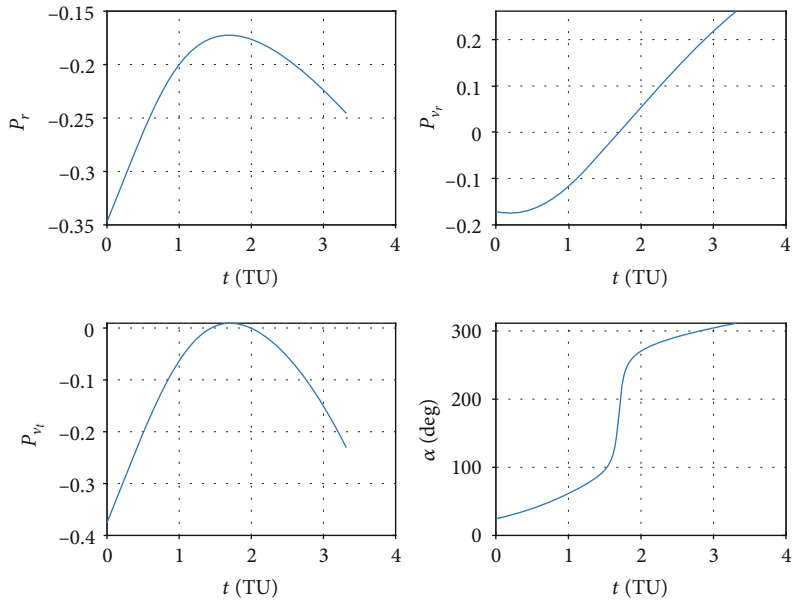

Figure 3: Optimal costate and control evolution in the Earth-Mars transfer.

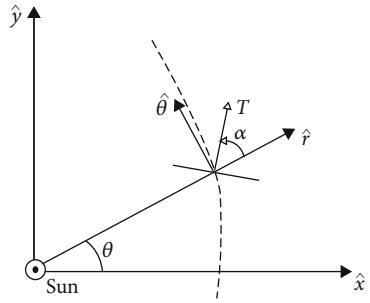

Figure 4: Thrust pointing angle α.

where r is the position vector and θ is the polar angle measured anticlockwise from the axis that connects the Sun to the Earth at the initial instant. v_r and v_θ are, respectively, the radial and the tangential velocity. α is the angle between the direction that connects the Sun to the solar sail and the thrust direction (Figure 4); a_c is the characteristic acceleration of the sail, and the terms b_1, b_2, and b_3 are the coefficients that represent the optical properties of the sail. A sail with an aluminium-coated front side and a chromium-coated back side with $b_1 = 0.1728$, $b_2 = 1.6544$, and $b_3 = -0.0109$ is considered [16].

The minimum time trajectory is obtained with an indirect approach. The Hamiltonian function is as follows:

$$H = p_r v_r + \frac{v_\theta \left(p_\theta - p_{v_\theta} v_r\right)}{r} + p_{v_r}\left(\frac{{v_\theta}^2}{r} - \frac{\mu_s}{r^2}\right) + \frac{a_c \cos\alpha \left(R_{\text{Earth}}/r\right)^2}{b_1 + b_2 + b_3}\left[p_{v_r}\left(b_1 + b_2\cos^2\alpha + b_3\cos\alpha\right) + p_{v_\theta}\sin\alpha\left(b_2\cos\alpha + b_3\right)\right], \tag{22}$$

where $[p_r, p_\theta, p_{v_r}, p_{v_\theta}]$ are the time-dependent costate variables. The Euler-Lagrange equations are as follows:

$$\begin{cases} \dot{p}_r = \dfrac{p_\theta v_\theta}{r^2} + p_{v_r}\left[\dfrac{{v_\theta}^2}{r^2} - \dfrac{2\mu_s}{r^3} + \dfrac{2a_c {R_{\text{Earth}}}^2}{r^3\left(b_1 + b_2 + b_3\right)}\cos\alpha\left(b_1 + b_2\cos^2\alpha + b_3\cos\alpha\right)\right] - p_{v_\theta}\left[\dfrac{v_r v_\theta}{r^2} - \dfrac{2a_c {R_{\text{Earth}}}^2}{r^3\left(b_1 + b_2 + b_3\right)}\sin\alpha\cos\alpha\left(b_2\cos\alpha + b_3\right)\right], \\ \dot{p}_\theta = 0, \\ \dot{p}_{v_r} = -p_r + \dfrac{v_\theta p_{v_\theta}}{r}, \\ \dot{p}_{v_\theta} = \dfrac{-2v_\theta p_{v_r} + v_r p_{v_\theta} - p_\theta}{r}. \end{cases} \tag{23}$$

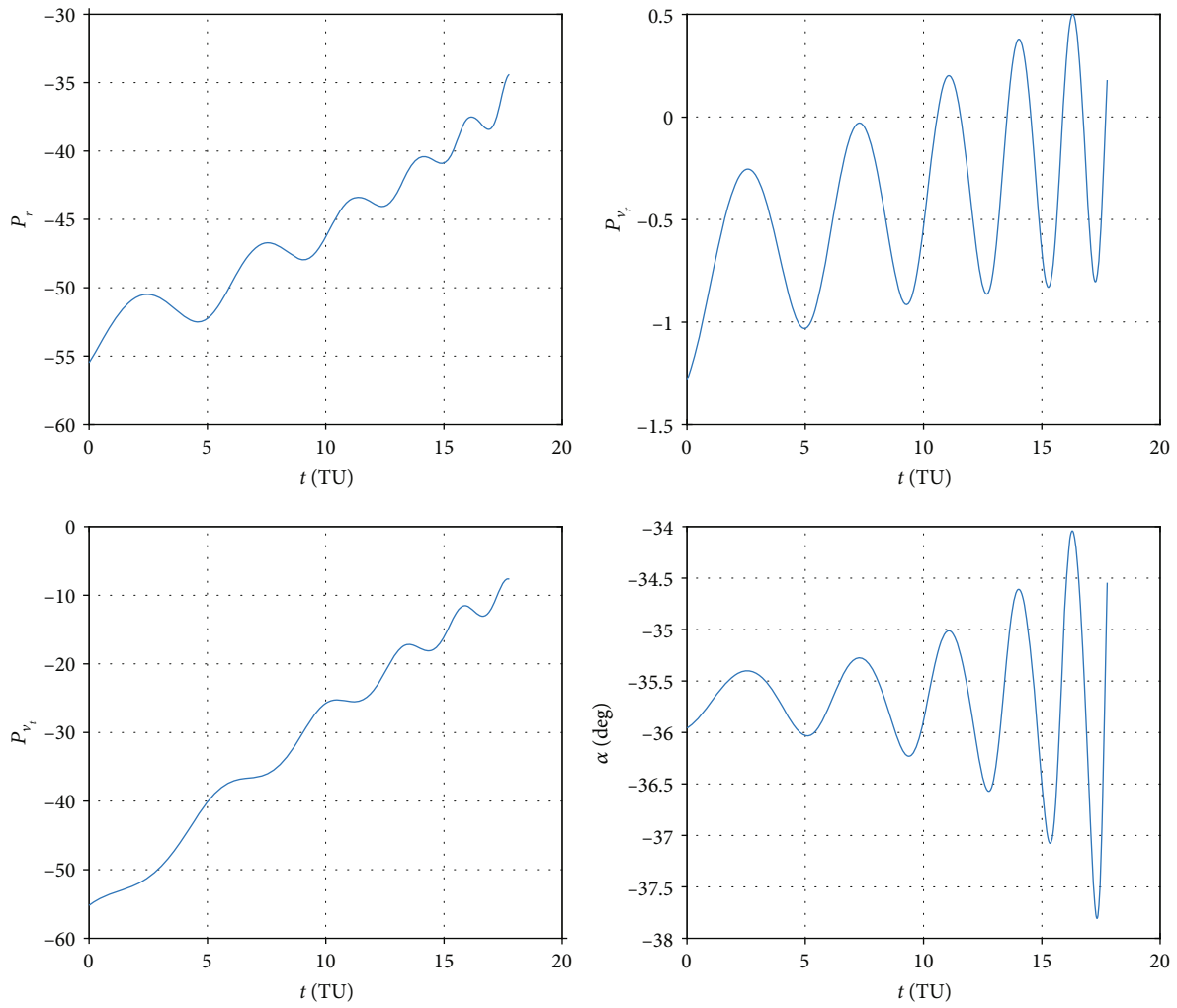

Figure 5: Optimal costate and control angle (alpha) for the Earth-Mercury transfer.

The optimal control will be obtained in the form $\alpha = \alpha(p_{v_r}, p_{v_\theta})$, but it is not possible to find an explicit solution in this form. The optimal steering law can be approximated by [17]:

$$\alpha = \begin{cases} \text{sign}\left(p_{v_\theta}\right)\tilde{\alpha} & \text{if } \alpha_p < \alpha_p^*, \\ \text{sign}\left(p_{v_\theta}\right)\left(\dfrac{\pi}{2}\right) & \text{if } \alpha_p \geq \alpha_p^*, \end{cases} \tag{24}$$

where

$$\begin{aligned} \cos\left(\alpha_p\right) &= \frac{p_{v_r}}{\sqrt{p_{v_\theta}^2 + p_{v_r}^2}} \text{ with } \alpha_p \in [0, \pi], \\ \alpha_p^* &\cong 2.5392 \text{ rad}, \\ \tilde{\alpha} &\cong 0.008109\alpha_p^6 - 0.05474\alpha_p^5 + 0.1356\alpha_p^4 \\ &\quad - 0.1266\alpha_p^3 + 0.08266\alpha_p^2 + 0.3038\alpha_p \\ &\quad + 0.0008666 \text{ with } \alpha_p \in \left[0, \alpha_p^*\right]. \end{aligned} \tag{25}$$

The equations of motion have the following boundary conditions:

$$r(t_0) = R_{\text{Earth}},\ \theta(t_0) = v_r(t_0) = 0,\ v_\theta(t_0) = \sqrt{\frac{\mu_s}{R_{\text{Earth}}}}. \tag{26}$$

There are four unknown parameters: $[p_r(0), p_{v_r}(0), p_{v_\theta}(0), t_f]$, and the optimal values are found in the following search space:

$$\begin{aligned} &1 \text{ TU} \leq t_f \leq 20 \text{ TU}, \\ &-1 \leq p_r(0)\, p_{v_r}(0)\, p_{v_\theta}(0) \leq 1. \end{aligned} \tag{27}$$

To investigate the domain, a population of 50 particles is considered. As in the previous case, an ECBO that preserves the best position in the population at each iteration is employed. Therefore, the selected dimension of the Colliding Memory is 1 and the crossover is not considered. The cost function is written, as in the previous case:

$$J = 0.01 \cdot t_f + 100 \cdot \Delta r + 40 \cdot \Delta v_r + 40 \cdot \Delta v_\theta. \tag{28}$$

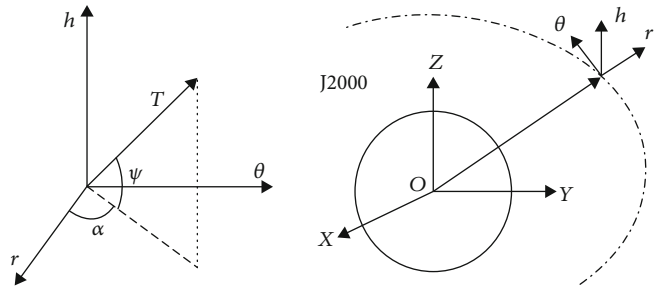

Figure 6: Thrust vector control direction in the local reference frame and its orientation with respect to J2000.

Although the cost function has the same form as the previous one, coefficients must be chosen carefully and they are different for each problem. The characteristic acceleration is set to 0.25mm/s^2 and the resulting transfer lasts 2.8 years, in agreement with the results obtained in [16]. The optimal costate and α time history are shown in Figure 5. The CBO, also in this case, demonstrates a great ability in finding the optimal initial costate values within 6500 function evaluations with no parameter tuning at all.

4. Numerical Simulation: Direct Methods

In this case, the test cases proposed are minimum time rendezvous with three different celestial bodies: an outer planet, an inner planet, and a very inclined asteroid. The transfer is considered heliocentric and the mathematical model consists in a three-dimensional restricted two body problem, with the Sun as the attractor and the spacecraft with negligible mass. The equations of motion are as follows:

$$\begin{cases} \dot{x} = v_x \\ \dot{y} = v_y \\ \dot{z} = v_z \\ \dot{v}_x = -\dfrac{\mu_s}{r^3} r_x + \dfrac{T_x}{m} \\ \dot{v}_y = -\dfrac{\mu_s}{r^3} r_y + \dfrac{T_y}{m} \\ \dot{v}_z = -\dfrac{\mu_s}{r^3} r_z + \dfrac{T_z}{m} \end{cases} \tag{29}$$

where $r = \sqrt{(x^2 + y^2 + z^2)}$ is the position vector of the spacecraft and T is the thrust vector. The mass of the spacecraft is indicated as m and its time varying law is $m = m_0 - \dot{m} \cdot t$, where m_0 is the initial spacecraft mass and $\dot{m}$ is the mass flow ratio. This set of equations is conveniently normalized using this set of units:

(i) Distance unit DU = $149.5 \cdot 10^6$km is the mean radius of the Earth's orbit

(ii) Time unit TU = $5.018 \cdot 10^6$ s, such that $\mu_s = \text{DU}^3/\text{TU}^2$

(iii) Mass unit MU = m_0

Table 1: Mars rendezvous results.

$\%_{\text{succ}}$	$t_{f_{\min}}$ (days)	$t_{f_{\text{mean}}}$	σ_{t_f}
77.8	288	302.08	8.418

A direct single shooting technique, with constant thrust magnitude, is proposed due to its simplicity and effectiveness [18]. The trajectory is divided into a finite number of arcs NA with variable duration Δt_k, with $k = 1, \cdots, \text{NA}$. During each arc, the angles α_k and ψ_k represent the thrust vector direction in the local reference frame $(\hat{r}, \hat{\theta}, \hat{h})$ with respect to the inertial reference frame J2000 (Figure 6).

In a rendezvous problem, the departure time is another variable to compute during the optimization process. The number of arcs NA is selected after a preliminary study and chosen to reduce as much as possible the computational effort [19]. Globally, there will be three variables for each arc plus the departure time. The ephemerides of the planets are DE430 (provided by the NASA SPICE tool [20]). With the transfer time t_f, we need to minimize the following arrival errors as well:

$$\begin{cases} \Delta x = x_s(t_f) - x_t(t_f), \\ \Delta y = y_s(t_f) - y_t(t_f), \\ \Delta z = z_s(t_f) - z_t(t_f), \\ \Delta v_x = v_{x_s}(t_f) - v_{x_t}(t_f), \\ \Delta v_y = v_{y_s}(t_f) - v_{y_t}(t_f), \\ \Delta v_z = v_{z_s}(t_f) - v_{z_t}(t_f), \end{cases} \tag{30}$$

where subscript s indicates the spacecraft state and t represents the target body. The desired final condition, hence, is that the arrival errors (Equation (30)) are zero.

The objective function is defined as follows:

$$J = 1000 \cdot \left(C_t t_f + C_d \cdot (\Delta x + \Delta y + \Delta z) + C_v \left(\Delta v_x + \Delta v_y + \Delta v_z\right)\right). \tag{31}$$

The coefficients C_t, C_d and C_v (that weight of the different dimensionless contributes in the cost function) are

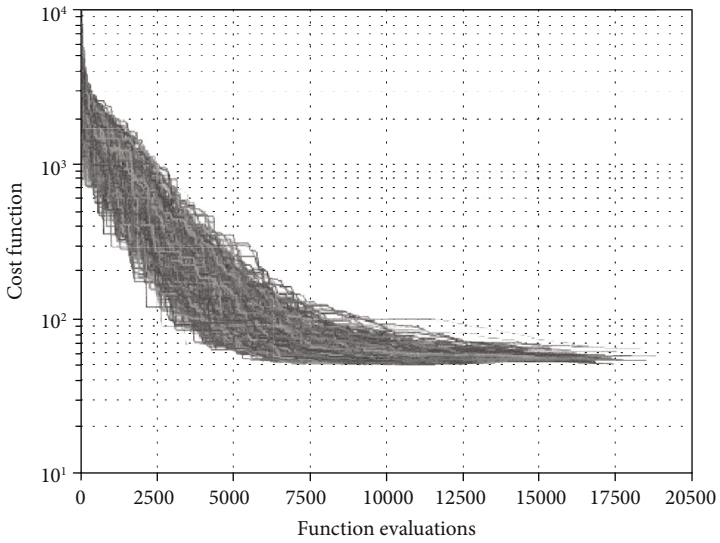

Figure 7: Cost function behavior for the Earth-Mars transfer.

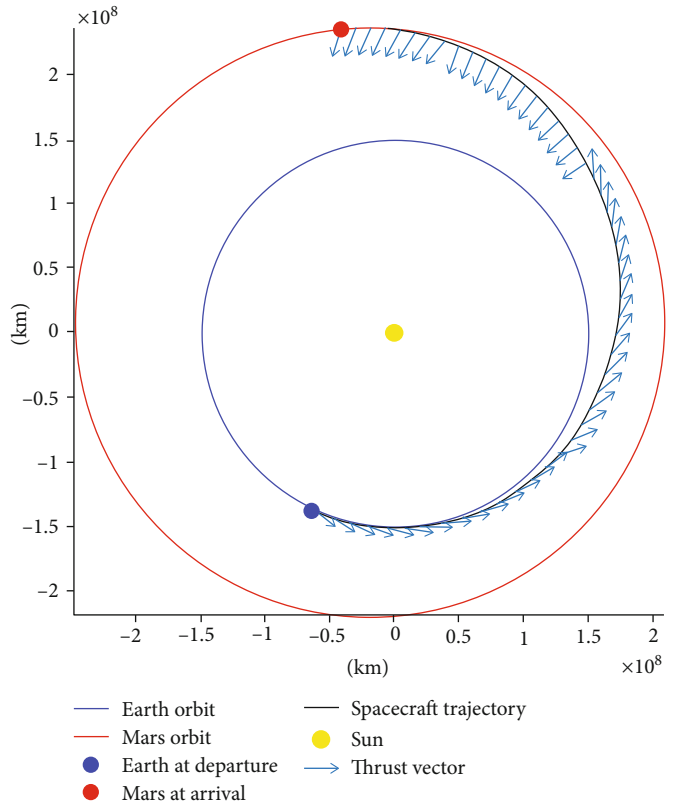

Figure 8: Optimal trajectory for the Earth-Mars transfer.

chosen after a preliminary study to properly balance the different terms, and their values will be presented for each test case. In the next section, three rendezvous missions will be discussed and the selected celestial bodies are Mars, Venus, and Pallas. The stopping criterion is composed of two conditions. The first one concerns the accuracy of the mission: each transfer will be considered successful, and hence, the relative simulation will stop, if the errors on position and velocity, described in Equation (30), go, respectively, below the mean radius of the target body and the threshold velocity of 20 m/s. The second term of the stopping criterion is a computational condition that imposes a maximum number of

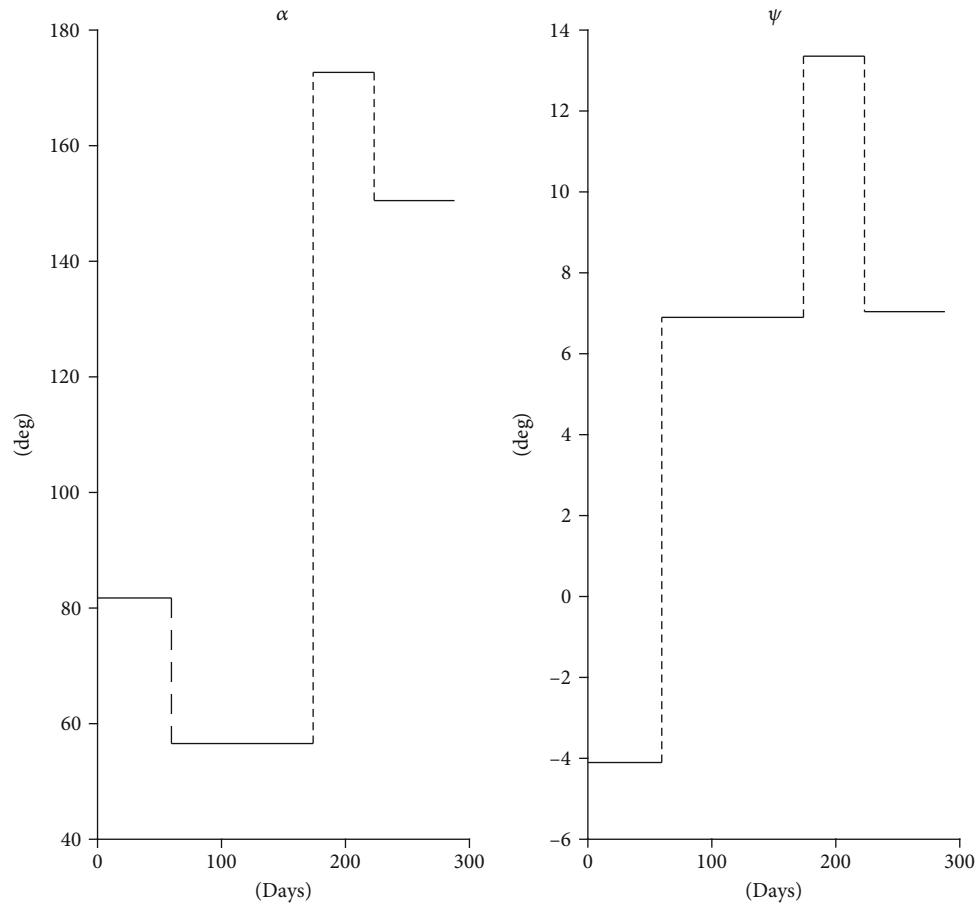

Figure 9: Optimal control law for the Earth-Mars transfer.

cost function evaluations F_{eval}. Each simulation will stop if F_{eval} exceeds the threshold value set for the specific mission. If this threshold is reached, the algorithm automatically reinitializes itself with a different initial distribution of particles. Both versions of the CBO will be used. In order to perform a more trustworthy analysis, one thousand runs will be executed for each case. Each run departs from different initial distributions of particles in the search space because of the randomness of the initialization of the positions (Equation (1)). This is done to evaluate the average performances of the algorithm. The first performance index considered is the percentage of success $(\%_{\text{succ}})$. A success occurs when the accuracy condition of the stopping criterion is met. Other performance indices are minimum and average values of transfer time (t_f) and number of function evaluations (F_{eval}). In addition, the dispersion around the mean value of transfer time σ_{t_f} will be shown.

4.1. Rendezvous with Mars. Here, the outer planet Mars has to be encountered by a spacecraft with the following characteristics:

(i) Thrust $T = 300$ mN

(ii) Specific impulse $I_{\text{sp}} = 3000$ s

Table 2: Venus rendezvous results.

$\%_{\text{succ}}$	$t_{f_{\min}}$(days)	$t_{f_{\text{mean}}}$	σ_{t_f}
38.8	232.8	241	5.82

(iii) Initial mass $m_0 = 1000$ kg

The search space, in terms of angles, arc duration, and departure time, must be chosen to let the optimization process start. The optimizer will look for the best solution out of the bounds here empirically defined:

(1) $\alpha_k \in [0°, 180°]$

(2) $\psi_k \in [-90°, 90°]$

(3) $\Delta t_k \in [10, 100]$ days

(4) $t_{\text{dep}} \in [01/07/2019 - 01/07/2020]$

The number of arcs chosen is NA = 4; therefore, there are 13 variables. The coefficients of the cost function are so established: $C_t = 0.01$, $C_d = 10$, $C_v = 10$. The maximum number of function evaluations selected for this problem is 20000. It is used a basic version of the CBO by employing 50 particles to investigate the search space.

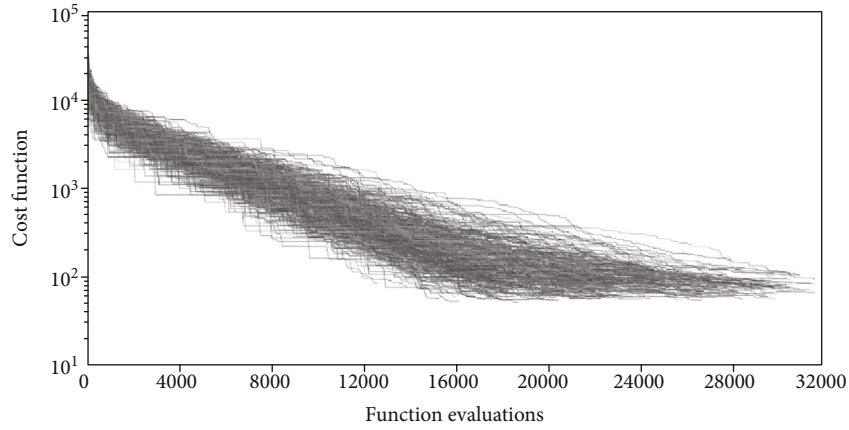

Figure 10: Cost function behavior for the Earth-Venus transfer.

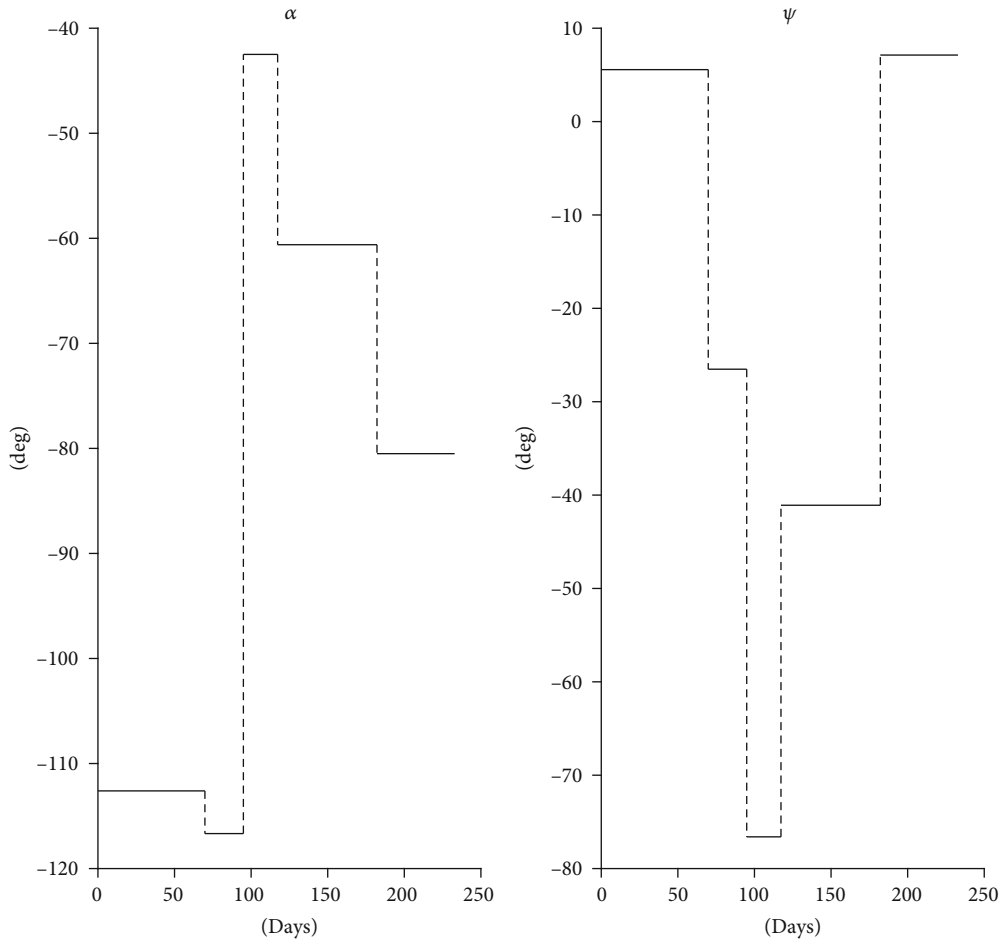

Figure 11: Optimal control law for the Earth-Venus transfer.

As can be seen from Table 1, the minimum time transfer, on average, lasts 302 days with a minimum of 288 days. The standard deviation is small; therefore, although the search space is wide, the CBO finds similar trajectories and this demonstrates the capability of the algorithm to explore effectively the search space. In Figure 7, the behavior of the cost function is plotted for all the successful runs (77.8%). It can be seen that the cost function decreases similarly for each run, which can be interpreted as a sign of robustness of the algorithm at changes in the initial conditions.

The best trajectory found so far carries the spacecraft to Mars in 288 days with a final mass of 746.35 kg (Figure 8). Finally, the control law is plotted in Figure 9.

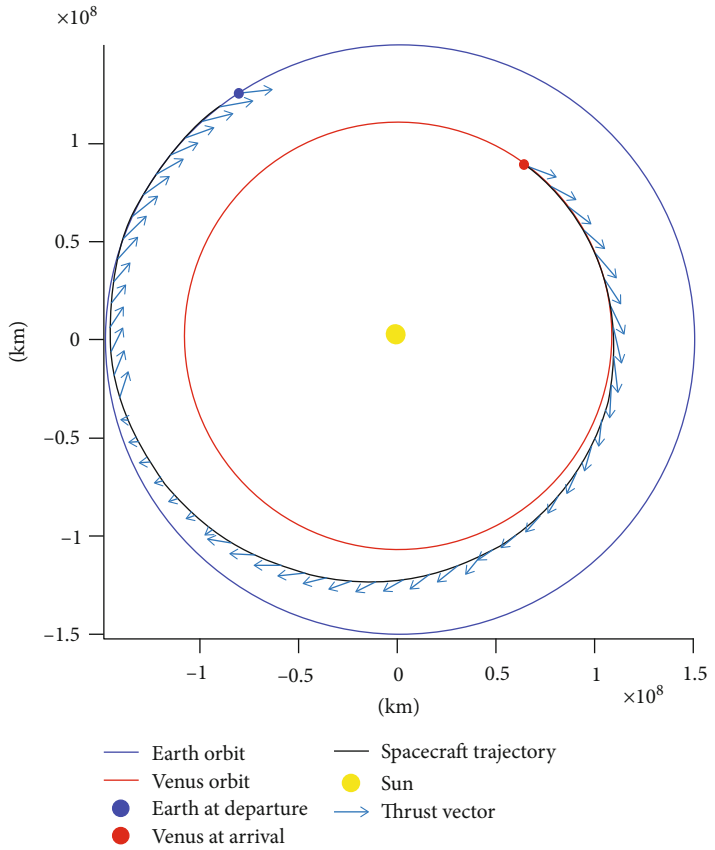

Figure 12: Optimal trajectory for the Earth-Venus transfer.

4.2. Rendezvous with Venus. Here, the inner planet Venus must be encountered by a spacecraft with the same characteristics as the previous one.

The search space, in terms of angles, arc duration, and departure time, is defined as follows:

(1) $\alpha_k \in [-180^\circ, 0^\circ]$

(2) $\psi_k \in [-90^\circ, 90^\circ]$

(3) $\Delta t_k \in [1, 90]$ days

(4) $t_{\text{dep}} \in [09/01/2019 - 09/01/2020]$

The number of arcs chosen is NA = 5; therefore, there are 16 variables. After some preliminary tests, the coefficients of the cost function are so established: $C_t = 0.01$, $C_d = 100$, $C_v = 50$. In this problem, the number of maximum function evaluations is set to 50000. It is employed a basic CBO with 80 particles. The results are shown in Table 2. The ability of the CBO to both explore and exploit is confirmed here by the small dispersion of the duration transfer around the mean value.

The balance between exploration and exploitation phase can also be deduced by the almost linear slope of the cost function trend, plotted in Figure 10. The optimal control law and the optimal trajectory are shown in Figures 11 and 12.

Table 3: Pallas rendezvous results.

$\%_{\text{succ}}$	$t_{f_{\min}}$ (days)	$t_{f_{\text{mean}}}$	σ_{t_f}
23	1429.3	1510.1	47.02

4.3. Rendezvous with Pallas. Pallas is an asteroid belonging to the asteroid belt that describes a very inclined orbit around the Sun (34.8°) with an eccentricity of 0.2305, making it a difficult body to reach. The probe chosen for this mission has the following characteristics:

(i) Thrust $T = 80$ mN

(ii) Specific impulse $I_{\text{sp}} = 3000$ s

(iii) Initial mass $m_0 = 600$ kg

The search space is as follows:

(1) $\alpha_k \in [0^\circ, 180^\circ]$

(2) $\psi_k \in [-90^\circ, 90^\circ]$

(3) $\Delta t_k \in [5, 250]$ days

(4) $t_{\text{dep}} \in [01/01/2019 - 01/01/2021]$

The number of arcs chosen is NA = 11 resulting in 34 variables. The coefficients are $C_t = 0.01$, $C_d = 50$, $C_v = 100$.

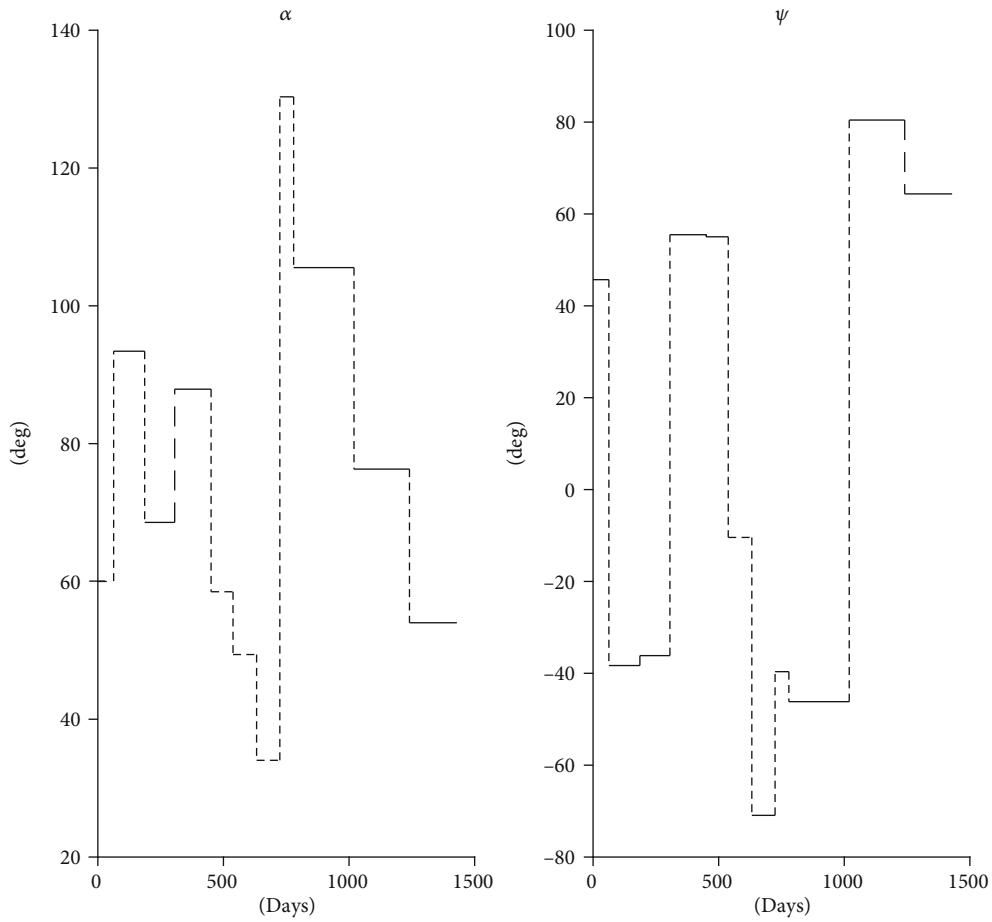

Figure 13: Optimal control law for the Earth-Pallas transfer.

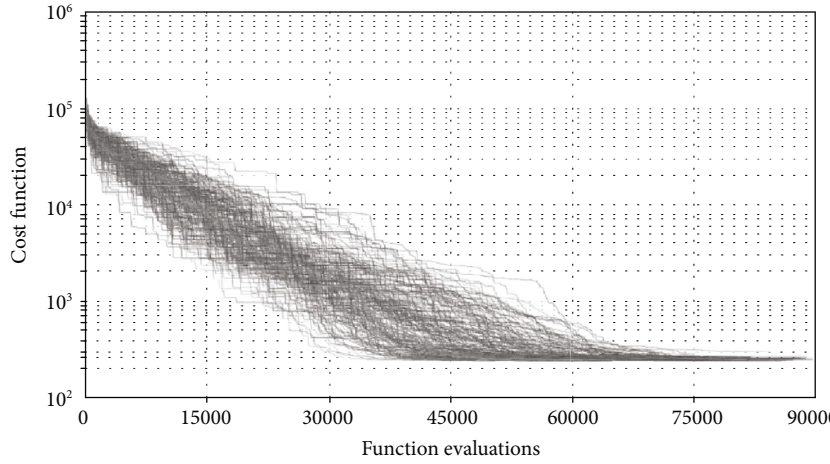

Figure 14: Cost function behavior for the Earth-Pallas transfer.

The maximum number of function evaluations allowed for this problem is 100000. It is employed an ECBO with one particle in the Colliding Memory without considering the crossover. The number of particles is 120. Despite the biggest energetic difference between the Earth and target orbits, the strictest arrival tolerances, the least powerful thruster, and the highest number of variables, the CBO can find optimal solutions for this rendezvous. The success percentage of 23% (Table 3) can be addressed to the concepts just mentioned, yet the value of standard deviations of $t_{\min}$ is very

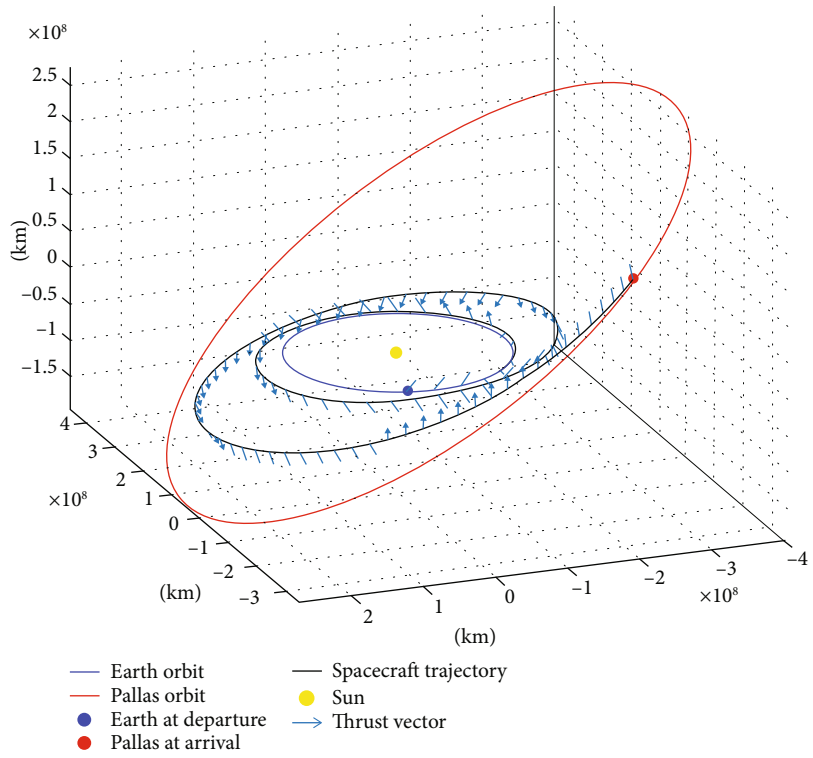

Figure 15: Optimal trajectory for the Earth-Pallas transfer.

small (3% of the mean transfer time), revealing that the discovered trajectories are very close to one another. This can be interpreted as a sign of the reliability of the algorithm. Figure 13, Figure 14, and Figure 15 represent, respectively, the optimal control law, the cost function behavior, and the optimal trajectory.

4.4. Discussion of the Results. It is worth mentioning that the direct approach, utilized in the last three cases, does not contemplate optimality conditions; therefore, there is not a definite optimal trajectory. Since there is no reference solution, a large number of simulations is needed to characterize the behavior of the algorithm. Due to the many degrees of freedom offered, the developed strategies allow many solutions to satisfy the arrival constraints. Although the range of possible transfer times is wide, the solutions found by the CBO are not equally distributed on all the possible values of t_f. In Figures 16, 17, and 18, the transfer durations of the successful trajectories are grouped into histograms. For all the cases tested above, they are centered around a mean value, biased towards the minimum time transfer, with very small standard deviation values (equal or less than 3% of the mean value), reported in Tables 1, 2, and 3.

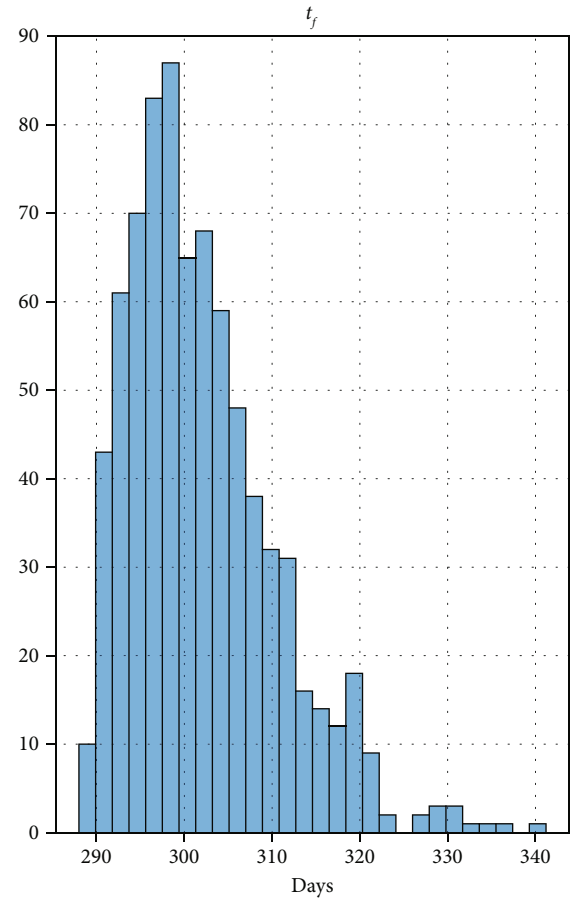

Figure 16: Transfer time histogram for the Earth-Mars transfer.

5. Conclusions

The study conducted in this paper reveals that the CBO is a valid algorithm, capable of solving a broad range of trajectory optimization problems. In the indirect cases, the CBO finds the optimal set of costate variables, discovering the optimal trajectory in a straightforward way. The direct cases were

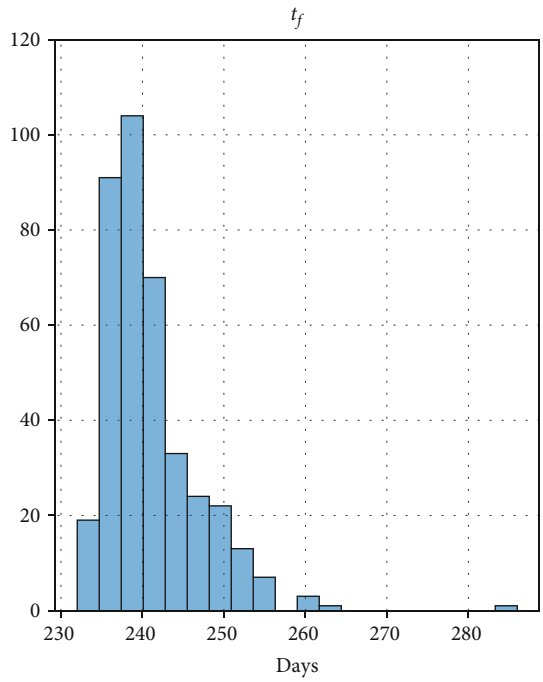

Figure 17: Transfer time histogram for the Earth-Venus transfer.

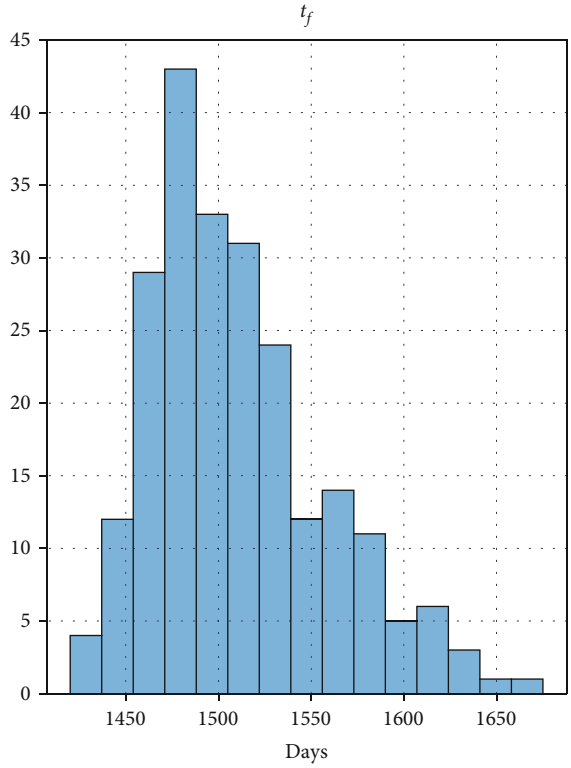

Figure 18: Transfer time histogram for the Earth-Pallas transfer.

made challenging on purpose, to test the effectiveness of the CBO thoroughly. Despite the increasing number of variables, the large domains, and the strict arrival tolerances, the algorithm succeeded in achieving satisfying trajectories for all the problems presented, exhibiting also a small dispersion around the suboptimal solution. Concerning the CBO performances, it is possible to state that, although it is straightforward in terms of computational effort and parameter settings, it shows great reliability and effectiveness in solving these kinds of problems. The remarkable advantage of CBO is that it is ready to use and it does not need an initial guess of the solution or insights of the problem. In addition, CBO has just one parameter to adjust (Elitism scheme); therefore, it does not need the fine-tuning preliminary operations required by metaheuristics in general.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

References

[1] H. J. Sussmann and J. C. Willems, "300 years of optimal control: from the brachystochrone to the maximum principle," *IEEE Control Systems Magazine*, vol. 17, no. 3, pp. 32–44, 1997.

[2] A. V. Rao, "A survey of numerical methods for optimal control," *Advances in the Astronautical Sciences*, vol. 135, no. 1, pp. 497–528, 2009.

[3] F. Glover, "Future paths for integer programming and links to artificial intelligence," *Computers & Operations Research*, vol. 13, no. 5, pp. 533–549, 1986.

[4] I. H. Osman and G. Laporte, *Metaheuristics: A Bibliography*, Springer, 1996.

[5] T. Stützle, *Local Search Algorithms for Combinatorial Problems: Analysis, Improvements and New Applications*, Infix Verlag, Sankt Augustin, Germany, 1999.

[6] C. Blum and A. Roli, "Metaheuristics in combinatorial optimization," *ACM Computing Surveys*, vol. 35, no. 3, pp. 268–308, 2003.

[7] M. Pontani and B. A. Conway, "Particle swarm optimization applied to space trajectories," *Journal of Guidance, Control, and Dynamics*, vol. 33, no. 5, pp. 1429–1441, 2010.

[8] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, vol. 4, Perth, WA, Australia, 1995.

[9] R. Storn and K. Price, "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.

[10] M. Vasile, E. Minisci, and M. Locatelli, "An inflationary differential evolution algorithm for space trajectory optimization," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 2, pp. 267–281, 2011.

[11] A. Kaveh and V. R. Mahdavi, "Colliding bodies optimization: a novel meta-heuristic method," *Computers & Structures*, vol. 139, pp. 18–27, 2014.

[12] A. Kaveh and M. Ilchi Ghazaan, "Enhanced colliding bodies optimization for design problems with continuous and discrete variables," *Advances in Engineering Software*, vol. 77, pp. 66–75, 2014.

[13] A. Kaveh and M. Ilchi Ghazaan, "Computer codes for colliding bodies optimization and its enhanced version," *International Journal of Optimization in Civil Engineering*, vol. 4, no. 3, pp. 321–332, 2014.

[14] A. E. Bryson and Y. C. Ho, *Applied Optimal Control*, Hemisphere, New York, NY, USA, 1975.

[15] A. E. Bryson, *Dynamic Optimization*, Addison Wesley, Menlo Park, CA, USA, 1999.

[16] A. A. Quarta and G. Mengali, "Solar sail missions to Mercury with Venus gravity assist," *Acta Astronautica*, vol. 65, no. 3-4, pp. 495–506, 2009.

[17] G. Mengali and A. A. Quarta, "Trajectory design with hybrid Low-Thrust propulsion system," *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 2, pp. 419–426, 2007.

[18] J. T. Betts, "Survey of numerical methods for trajectory optimization," *Journal of Guidance, Control, and Dynamics*, vol. 21, no. 2, pp. 193–207, 1998.

[19] A. Caruso, A. A. Quarta, and G. Mengali, "Comparison between direct and indirect approach to solar sail circle-to-circle orbit raising optimization," *Astrodynamics*, vol. 3, no. 3, pp. 273–284, 2019.

[20] C. H. Acton Jr., "Ancillary data services of NASA's Navigation and Ancillary Information Facility," *Planetary and Space Science*, vol. 44, no. 1, pp. 65–70, 1996.