The Journal of the Astronautical Sciences (2023) 70:51
https://doi.org/10.1007/s40295-023-00417-4

ORIGINAL ARTICLE

On the Performance of Different Smoothing Methods for Indirect Low-Thrust Trajectory Optimization

Yanis Sidhoum[1] · **Kenshiro Oguri**[1]

Accepted: 9 October 2023 / Published online: 16 November 2023

Abstract
This paper compares the performance of two popular smoothing techniques involving the homotopic approach and the hyperbolic tangent smoothing within the indirect formulation of low-thrust minimum-fuel optimization problem in the Restricted Two-Body Problem. Their respective performance are investigated on quantitative and qualitative criteria such as computational burden, including number of iterations, convergence time, and accuracy of the optimal solution. Additionally, the Particle Swarm Optimization algorithm is used to analyze the global landscape of the performance comparison of the two methods. This investigation unifies the available literature and provides useful recommendations for solving future low-thrust optimization problems involving bang-bang control inputs.

1 Introduction

Electric propulsion (EP) is a class of propulsion which makes use of electrical power to accelerate a propellant by different possible electrical and/or magnetic means. It has emerged as a major low-thrust trajectory technique since the successful demonstration of deep space explorations such as Deep Space 1 in 1998 [1] and has gained significant appeal since the Japanese mission Hayabusa in 2010 [2]. The use of electrical power has enhanced the propulsive performance compared with traditional chemical propulsion systems, especially in terms of specific impulse. Indeed, the propellant is ejected up to twenty times faster than from a classical chemical thruster while requiring very little propellant mass to accelerate the spacecraft: when expressed in terms of specific impulse the factor of improvement is between 5 and 10 [3]. However, EP is more challenging for trajectory design than chemical

✉ Yanis Sidhoum
ysidhoum@purdue.edu

Kenshiro Oguri
koguri@purdue.edu

[1] School of Aeronautics and Astronautics, Purdue University, 701 W Stadium Ave, West Lafayette, IN 47907, USA

propulsion due to its continuous and durable low-thrust making the optimal trajectory even more difficult to find. Usually, optimization methods under consideration can be classified as direct, indirect, and hybrid methods for a combination of both [4].

The direct method approximates the initial optimal problem with a discrete optimization problem and is therefore converted into a parameter optimization problem. Then nonlinear programming is used to find the optimal solution [5, 6]. On the other hand, indirect method is based on finding optimal trajectories that satisfy the necessary conditions of optimality derived by applying the Pontryagin's Minimum Principle (PMP) and the calculus of variations [7]. In his 1963 book, Lawden demonstrated that indirect optimization always yields a Two Point Boundary Value Problem (TPBVP) [8]. Generally speaking the main benefit of indirect methods is to reduce the dimensions in the search space. However, indirect methods suffer from extreme sensitivities to the search variables. More specifically, strong convergence properties have been demonstrated when the initial guess is near a solution while the extreme sensitivities result in a small radius of convergence [9]. Indirect methods are also more difficult to generalize for atypical constraints than direct methods [10]. Nevertheless in a modern context some research such as on the primer vector theory and technological breakthrough with fast modern computers have renewed the interest in indirect methods and offers new perspective of optimal trajectory in a multi-objective landscape [11]. A common method for solving TPBVP is the shooting method, which is based on local gradient algorithms like Newton's and Powell's methods [12]. To match the boundary conditions, the initial guess is iteratively adjusted using the Jacobian matrix of the shooting function. This Jacobian matrix can be approximately computed by finite differences (FD) or precisely determined using the analytic derivative [13, 14]. Applying PMP shows that the fuel-optimal control when using EP is a discontinuous bang-bang control [7]. Bertrand and Epenoy [9] shows that this property leads to discontinuity in the shooting function and singularity of the Jacobian matrix on a large domain. To overcome these issues they introduce smoothing techniques that involve the homotopic approach [15]. More specifically, they propose to add a perturbed energy term in the objective function to increase the radius of convergence. Guo et al. [16], Jiang et al. [17], and Chi et al. [18] have proposed different application of the homotopic approach with quadratic penalty factor in the framework of low-thrust transfer in the Restricted Two-Body Problem (R2BP). On the other hand, Taheri and Junkins [19, 20] proposed a smoothing technique that applies directly at the control level without changing the objective function. This smoothing technique consists of approximating the discontinuous bang-bang control by a hyperbolic tangent function. Further C^{∞} approximation of the bang-bang control law by means of exponential or arc-tangential functions can be found in [21, 22].

However the extreme sensitivity of the indirect formulation to the initial guess is not totally removed so that the need for reasonable initial costate guess is evident. In mid-1990 s Kennedy and Eberhart [23] first presented the Particle Swarm Optimization (PSO), a heuristic search algorithm that emulates the collective behavior of bird flocks to solve complex nonlinear optimization problems. Since then, the PSO has been widely applied to low-thrust trajectory optimization for providing reasonable

initial costates [17, 24, 25]. On the other hand Yan and Wu [26] proposed to express the initial costates by a first-order Taylor series expansion in the neighborhood of the initial time. Then using the costates differential equations and the transversality conditions leads to a set of linear equations with respect to the initial costates.

Performance comparisons between different numerical methods are important for mission design engineers, as it provides a basis for deciding which methods to use for solving trajectory optimization problems. In Caruso et al. [27] the performance of indirect and direct methods are compared in the framework of transfer trajectories of a solar sail between circular and coplanar orbits. Jiang et al. [17] compare the effect of PSO search with random initialization of costates to demonstrate the benefits of the first for initial costates guess. Bertrand and Epenoy [9] compare the performance of different homotopy function within the application of homotopic methods, including quadratic penalty and logarithmic barrier. However, comparison between different families of smoothing methods has not been performed yet. This is the purpose of this paper. We provide a comprehensive comparison of the performance of the homotopic approach and the hyperbolic tangent smoothing in terms of computational burden (e.g number of iterations, function evaluations, and computation time), accuracy of the shooting method, and sensitivity to the initial guess. This comprehensive comparison is performed in the scenario of a fuel-optimal rendezvous problem from the Earth to Venus. Also we compare the smoothness properties of the two methods theoretically.

The rest of the paper is organised as follows. In Sect. 2 the low-thrust trajectory optimization problem is formulated. Then in Sect. 3, the concept of smoothing methods, with a focus on the hyperbolic tangent smoothing and the homotopic approach, is presented. In Sect. 4, the theoretical framework that is used to compare the performances of the two smoothing methods is detailed. The differentiation properties of the two methods for analytic derivatives implementation are discussed. Then PSO is presented, and its application to low-thrust trajectory indirect optimization is described. In this investigation, PSO is leveraged to analyze the global landscape of the performances of these smoothing methods. In Sect. 5 the performance comparison based on quantitative criteria is performed in the case study of an Earth-to-Venus rendezvous. The benefits of feeding the solver with the analytic derivative are also showcased. Section 6 provides conclusion of this paper.

2 Problem Statement

Let us consider the main problem of this paper. The spacecraft is subject only to the central force of the Sun's gravity and the thrust of its own EP system. All other disturbing accelerations (e.g., solar radiation pressure and third-body perturbations) are ignored. Therefore, equations are derived from the two-body problem considering the spacecraft and the Sun. The dynamics are described using Cartesian coordinates with $\boldsymbol{x} = [\boldsymbol{r}, \boldsymbol{v}, m]^\top$ as the state vector, while the control input vector, $\boldsymbol{u}$, is parameterized through the throttle factor $u \in [0, 1]$, the thrust direction unit vector $\boldsymbol{\alpha}$, and the maximum thrust, $T_{\max}$. The equations of motion considered in the heliocentric ecliptic reference frame (HERF) can be written as follows:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\alpha}, u) \implies \begin{bmatrix} \dot{\boldsymbol{r}} \\ \dot{\boldsymbol{v}} \\ \dot{m} \end{bmatrix} = \begin{bmatrix} \boldsymbol{v} \\ -\frac{\mu}{r^3}\boldsymbol{r} + \frac{uT_{\max}}{m}\boldsymbol{\alpha} \\ -\frac{T_{\max}u}{I_{\mathrm{sp}}g_0} \end{bmatrix} \tag{1}$$

where $\mathbf{r}$, $\mathbf{v}$, and m are respectively the position, velocity, and mass of the spacecraft. I_{sp} is the thruster specific impulse, g_0 the gravitational acceleration at the sea level, 9.80665 m.s^{-2}, and μ the Sun's gravitational constant, $1.32712440018 \times 10^{11}$ km^3.s^{-2}. For the sake of numerical computation the quantities about length, time, and mass are nondimensionalized by the astronomical unit (AU, 149597870.66 km), year (yr, 365.25×86.400 s), and spacecraft initial mass m_0. In the following, this dynamical model is used to design low-thrust spacecraft trajectories that minimize the fuel consumption while satisfying the departure and arrival conditions with fixed time of flight.

2.1 Fuel-Optimal Problem

The fuel-optimal problem consists in minimizing fuel consumption, which is equivalent to maximize the final mass. Therefore the performance index J_0 can be established as follows:

$$J_0 = -m(t_{\mathrm{f}}) \tag{2}$$

Thus the optimal thrust, given by the power throttle u^* and thrust direction $\boldsymbol{\alpha}^*$ should minimize this cost. From Eq. (1) we can transform this cost index into:

$$J_0 = \frac{T_{\max}}{I_{\mathrm{sp}}g_0} \int_{t_0}^{t_{\mathrm{f}}} u(t)\, dt \tag{3}$$

By contrast, the energy-optimal cost index J_1 is defined as:

$$J_1 = \frac{T_{\max}}{I_{\mathrm{sp}}g_0} \int_{t_0}^{t_{\mathrm{f}}} u^2(t)\, dt \tag{4}$$

In these expressions, t_0 and t_{f} denote the initial and final times, respectively. Both times are fixed, like initial position, velocity and mass, as well as final position and velocity. Therefore the optimal solution of the low-thrust trajectory problem considered in the present paper should satisfy the following boundary conditions:

$$\boldsymbol{r}(t_0) = \boldsymbol{r}_0 \qquad \boldsymbol{v}(t_0) = \boldsymbol{v}_0 \qquad m(t_0) = m_0 \tag{5a}$$

$$\boldsymbol{r}(t_{\mathrm{f}}) = \boldsymbol{r}_{\mathrm{f}} \qquad \boldsymbol{v}(t_{\mathrm{f}}) = \boldsymbol{v}_{\mathrm{f}} \tag{5b}$$

2.2 Optimal Control and TPBVP

The Hamiltonian is built based on PMP [7]:

$$H = \lambda_{\mathrm{r}} \cdot v + \lambda_{\mathrm{v}} \cdot \left(-\frac{\mu}{r^3}\mathbf{r} + \frac{T_{\max} u}{m}\boldsymbol{\alpha} \right) - \lambda_{\mathrm{m}} \frac{T_{\max} u}{I_{\mathrm{sp}} g_0} + \frac{T_{\max}}{I_{\mathrm{sp}} g_0} u \tag{6}$$

where $[\boldsymbol{\lambda}_{\mathbf{r}},\boldsymbol{\lambda}_{\mathbf{v}},\lambda_{\mathrm{m}}]^{\top}$ are the Lagrange multipliers and also called the costates vectors. According to PMP, the optimal control is given by:

$$(u^*, \boldsymbol{\alpha}^*) = \arg \min_{(u,\boldsymbol{\alpha})} H \tag{7}$$

Since $u \geq 0$, the PMP indicates that the velocity costate vector $\boldsymbol{\lambda}_{\mathbf{v}}$, termed as the primer vector, must lie along the opposite direction of the optimal thrust direction $\boldsymbol{\alpha}^*$:

$$\boldsymbol{\alpha}^* = -\frac{\boldsymbol{\lambda}_{\mathbf{v}}}{\|\boldsymbol{\lambda}_{\mathbf{v}}\|} \tag{8}$$

This optimal thrust direction leads to a simplified Hamiltonian:

$$H = -\frac{T_{\max} u}{I_{\mathrm{sp}} g_0}(\lambda_{\mathrm{m}} + \frac{I_{\mathrm{sp}} g_0 \|\boldsymbol{\lambda}_{\mathbf{v}}\|}{m} - 1) + \boldsymbol{\lambda}_{\mathbf{r}} \cdot \boldsymbol{v} - \frac{\mu \boldsymbol{\lambda}_{\mathbf{v}} \cdot \mathbf{r}}{r^3} \tag{9}$$

It is worth noticing that because the Hamiltonian in Eq. (9) is linear with respect to the power throttle level u and the fact that $u \in [0, 1]$, the optimal thrust magnitude u^* is derived as:

$$u^* = \begin{cases} 1 & \text{if } \rho(t) > 0 \\ 0 & \text{if } \rho(t) < 0 \\ \in [0, 1] & \text{if } \rho(t) = 0 \end{cases} \tag{10}$$

where $\rho(.)$ is called the switching function and is defined as follows:

$$\rho = \lambda_{\mathrm{m}} + \frac{I_{\mathrm{sp}} g_0 \|\boldsymbol{\lambda}_{\mathbf{v}}\|}{m} - 1 \tag{11}$$

Bertrand and Epenoy [9] suggest that one can reasonably assume that the aforementioned switching function $\rho(.)$ takes the value zero only at isolated points on the time span $[t_0, t_{\mathrm{f}}]$. As consequence, there is no singular arc, thus supporting that the optimal control is a bang-bang control. This assumption will be numerically verified in Sect. 5. The Euler-Lagrange equations for costates variables are then derived as:

$$\dot{\boldsymbol{\lambda}}_{\mathbf{r}} = -\frac{\partial H}{\partial \boldsymbol{r}} = \frac{\mu}{r^3}\boldsymbol{\lambda}_{\mathrm{v}} - \frac{3\mu \boldsymbol{r} \cdot \boldsymbol{\lambda}_{\mathbf{v}}}{r^5}\boldsymbol{r} \tag{12a}$$

$$\dot{\lambda}_{\mathrm{v}} = -\frac{\partial H}{\partial \boldsymbol{v}} = -\lambda_{\mathbf{r}} \tag{12b}$$

$$\dot{\lambda}_{\mathrm{m}} = -\frac{\partial H}{\partial m} = -\frac{T_{\max} u}{m^2} \| \lambda_{\mathrm{v}} \| \tag{12c}$$

Therefore, the dynamics of the fuel-optimal control problem is completely defined by the state equation Eq. (1), the optimal thrust $\boldsymbol{\alpha}^*$ expressed in Eq. (8), and the above related costates equations Eq. (12). The 14-dimensional canonical variable $\boldsymbol{y} = [\boldsymbol{x}, \lambda]^\top$ is therefore governed by:

$$\dot{\boldsymbol{y}} = \boldsymbol{F}(\boldsymbol{y}) \implies \begin{bmatrix} \dot{\boldsymbol{r}} \\ \dot{\boldsymbol{v}} \\ \dot{m} \\ \dot{\lambda}_{\mathbf{r}} \\ \dot{\lambda}_{\mathrm{v}} \\ \dot{\lambda}_{\mathrm{m}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{v} \\ -\frac{\mu}{r^3}\boldsymbol{r} - \frac{u T_{\max}}{m} \frac{\lambda_v}{\| \lambda_{\mathrm{v}} \|} \\ -\frac{T_{\max} u}{I_{\mathrm{sp}} g_0} \\ \frac{\mu}{r^3} \lambda_{\mathrm{v}} - \frac{3\mu \boldsymbol{r} \cdot \lambda_{\mathrm{v}}}{r^5} \boldsymbol{r} \\ -\lambda_{\mathbf{r}} \\ -\frac{T_{\max} u}{m^2} \| \lambda_{\mathrm{v}} \| \end{bmatrix} \tag{13}$$

Moreover, like state vector, costate vector must satisfy terminal constraint. According to the transversality condition when a boundary state is free, the corresponding costate is zero (for more details see [7]). Since the final mass is free, the final mass costate is zero:

$$\lambda_{\mathrm{m}}(t_{\mathrm{f}}) = 0 \tag{14}$$

It should be pointed out that Eq. (12c) indicates that the time derivative of the mass costate is negative and Eq. (14) says the terminal mass costate is zero, thus:

$$\lambda_{\mathrm{m}} \geq 0 \tag{15}$$

The final position and velocity shall satisfy Eq. (5) while the final mass costate shall satisfy Eq. (14). As consequence, the fuel-optimal control problem yields a TPBVP consisting of a set of equations of the form:

$$\boldsymbol{\Psi}(z) = \left[\boldsymbol{r}(t_{\mathrm{f}}) - \boldsymbol{r}_{\mathrm{f}}, \boldsymbol{v}(t_{\mathrm{f}}) - \boldsymbol{v}_{\mathrm{f}}, \lambda_m(t_{\mathrm{f}}) \right] = \mathbf{0} \tag{16}$$

where $\boldsymbol{\Psi}$ is the so-called shooting function and $z = [\lambda_{\mathbf{r}}(t_0), \lambda_{\mathbf{v}}(t_0), \lambda_{\mathrm{m}}(t_0)]$ are the unknown initial costates. As pointed out by the literature, finding such solutions is not a simple process, especially because of the discontinuous properties of the optimal bang-bang control u^*. Indeed, as expressed in Eq. (10) and neglecting singular arcs, u^* is either zero or one which leads to nonsmoothness and discontinuous right-hand sides of ordinary differential equations (ODE)s and singularity of the Jacobian of the shooting function on a large domain [9]. As a result the shooting function is difficult to solve by classical gradient based algorithms such as Newton's method [12]. One common approach to overcome these issues is to apply smoothing methods to the bang-bang control inputs.

3 Smoothing Methods for Optimal Bang–Bang Control

This section describes the concept of smoothing methods involving the homotopic approach and the hyperbolic tangent smoothing. The initial problem is linked to easy-handling auxiliary problems using a continuation parameter ϵ, leading to a cascade of TPBVPs. The continuation procedure consists of solving a series of TPBVPs for a decreasing sequence of ϵ values ($\epsilon_1 > \epsilon_2 > ... > \epsilon_N$). The current TPBVP associated with $\epsilon = \epsilon_{k+1}$ is solved with the solution of the previous one ($\epsilon = \epsilon_k$) as a starting point. The continuation parameter is decreased by multiplying it by a reducing factor γ less than one, and that is ideally chosen to be as small as possible. However, choosing small value of γ can result in nonconvergence output flag of the solver throughout the continuation procedure. This iterative process ends when the current smoothing parameter ϵ is below a certain threshold, set to 10^{-4} in this investigation. A flow diagram of the general smoothing strategy is illustrated in Fig. 1.

3.1 Homotopic Approach

The homotopic approach is based on perturbing the minimum-fuel performance index J_0 defined in Eq. (3) by a penalty factor. Bertrand and Epenoy [9] proposed

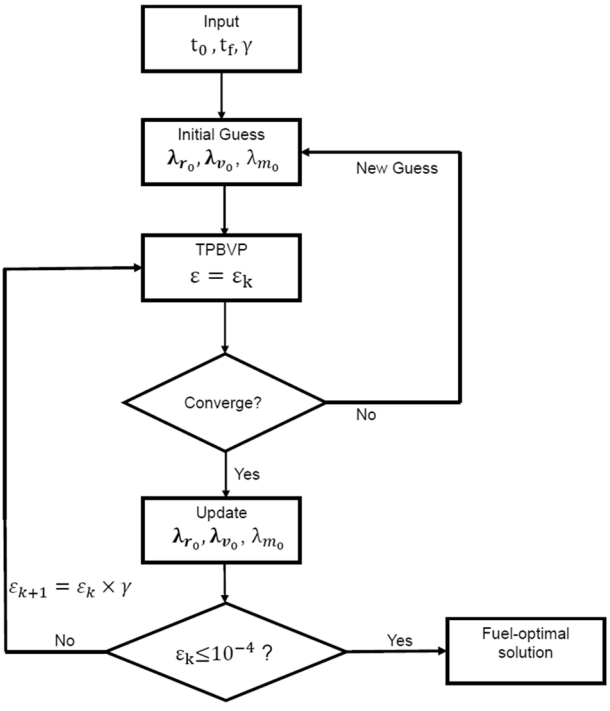

Fig. 1 Flow diagram of smoothing strategy

different penalty function, especially the quadratic penalty, which builds the homotopy as follows:

$$J_\epsilon = \frac{T_{\max}}{I_{sp} g_0} \int_{t_0}^{t_f} [u - \epsilon u(1-u)]\, dt \tag{17}$$

where $\epsilon \in]0, 1]$ is the continuation parameter and links the mass criterion ($\epsilon = 0$) to the energy criterion ($\epsilon = 1$). The optimal thrust direction $\boldsymbol{\alpha}^*$ and the switching function $\rho(.)$ hold the same forms as in Eqs. (8) and (11) while the optimal thrust magnitude becomes:

$$u_\epsilon^* = \begin{cases} 1 & \text{if } \rho(t) > \epsilon \\ 0 & \text{if } \rho(t) < -\epsilon \\ \frac{1}{2} + \frac{\rho}{2\epsilon} & \text{if } |\rho(t)| \leq -\epsilon \end{cases} \tag{18}$$

From Eq. (18), one may note that the optimal control u_ϵ^* is now continuous if and only if $\epsilon \neq 0$. Therefore, the integration accuracy using ordinary ODE solvers such as MATLAB's ode45 is expected to be improved. The procedure is initialized by solving the easier-related energy-optimal problem ($\epsilon = 1$). Once the energy-optimal problem is solved, the homotopic procedure is applied by continuously decreasing the continuation parameter ϵ from one to zero and taking the solution $z_{\epsilon_k}^*$ of the current shooting problem for ϵ_k as an initial guess of the next one for ϵ_{k+1}. Once the continuation parameter ϵ is below 10^{-4}, the minimum-fuel solution is reached.

3.2 Hyperbolic Tangent Smoothing

Taheri and Junkins [19, 20] introduced a smoothing technique that involves hyperbolic tangent function to approximate bang-off-bang spacecraft maneuvers and demonstrated the application of this smoothing method to minimum-fuel rendezvous maneuvers. The main advantage of this smoothing method in comparison to the homotopic procedure is its simplicity because it applies directly at the control level. Thus the performance index holds the same form as in Eq. (3) throughout the smoothing procedure, meaning that the hyperbolic tangent smoothing involves solving a series of fuel-optimal problems. This feature simplifies the procedure since the necessary conditions for the indirect method, and therefore the shooting function are unaffected. Assuming that there are no singular arcs the optimal value of the throttling is:

$$u_{\text{bang-bang}}^* = \begin{cases} 1 \text{ if } \rho(t) > 0 \\ 0 \text{ if } \rho(t) < 0 \end{cases} = \frac{1}{2}\left[1 + \text{sign}(\rho)\right] \tag{19}$$

Then the class of hyperbolic functions can be used to approximate the sign function and therefore smoothing the control input:

$$u^*_{\text{bang-bang}}(\rho) \approx u^*_{\text{hyperbolic}}(\rho,\epsilon) = \frac{1}{2}\left[(u_u + u_l) + (u_u - u_l)\tanh\left(\frac{\rho-\rho_C}{\epsilon}\right)\right] \tag{20}$$

where u_u and u_l are the upper and lower bounds for the throttle factor u, respectively. In this paper, $u_u = 1$ and $u_l = 0$. ρ_C defines the switching point, i.e when the switching function crosses zero ($\rho_C = 0$), and ϵ is the smoothing parameter that is used as a continuation parameter for the numerical continuation procedure. Therefore the smoothed engine throttle input becomes:

$$u^*_{\text{bang-bang}}(\rho) \approx u^*_{\text{hyperbolic}}(\rho,\epsilon) = \frac{1}{2}\left[1 + \tanh\left(\frac{\rho}{\epsilon}\right)\right] \tag{21}$$

The aforementioned continuation parameter ϵ is progressively decreased from $\epsilon = 1$ to $\epsilon = 0$, i.e a continuation procedure is performed from smooth switches to bang-off-bang switches. Taheri and Junkins [19, 20] showed that the pure bang-bang structure is captured with satisfactory accuracy once ϵ is below 10^{-2}, however for the purpose of convergence performance comparison with the homotopic approach, the procedure is continued until $\epsilon = 10^{-4}$.

4 Framework of the Performance Comparison

In this section the framework of the performance comparison between the homotopic approach and the hyperbolic tangent smoothing is detailed. Figure 2 provides a schematic view of this framework. The two smoothing methods are first compared theoretically on their smoothness properties. Then their numerical performances are analyzed on a specific case study. Finally, the Particle Swarm Algorithm (PSO) is leveraged to investigate the global landscape of their performances.

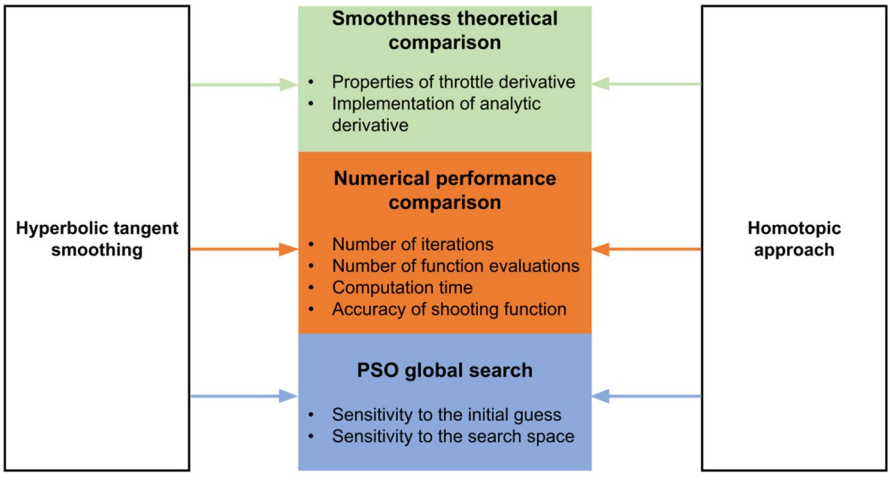

Fig. 2 Framework of the performance comparison

4.1 Analytic Derivatives

One common approach to increase the accuracy and robustness of the shooting algorithm is to feed the nonlinear solver with analytic derivatives. When smoothing methods are applied, computing the analytic derivatives involves the differentiation of the throttle factor. In the following sections, we discuss the better differentiation properties of hyperbolic tangent smoothing over homotopic approach and explain how this allows an easier incorporation of analytic derivatives to the shooting algorithm.

4.1.1 State Transition Matrix Based Gradient Derivation

The single shooting scheme is based on gradient information relating changes in the final integrated state $\boldsymbol{x}(\boldsymbol{x}_0, \boldsymbol{\lambda}_0, t_\mathrm{f})$ and costate $\boldsymbol{\lambda}(\boldsymbol{x}_0, \boldsymbol{\lambda}_0, t_\mathrm{f})$ vectors to the changes in the initial state $\boldsymbol{x_0}$ and costate $\boldsymbol{\lambda}_0$ vectors. These gradient information are used to adjust the initial costate $\boldsymbol{\lambda}_0$ to match the boundary conditions. Optimization toolbox solvers, such as MATLAB's *fsolve*, compute this gradient information using the finite difference approximation. While FD can provide very close approximation of the actual derivative, it cannot ensure a high level of accuracy and increases the computational burden (for each component of the Jacobian matrix $\boldsymbol{J_\Psi}$ of the shooting function, the shooting function $\boldsymbol{\Psi}$ is evaluated at least twice) [28]. Vishala et al. [14] compared the efficiency and the robustness of shooting method with and without feeding analytic derivative into *fsolve*, starting from the same initial guess. They show that using the analytic gradient instead of numerical FD method provides robustness and improve the efficiency and capability of the algorithm, especially for very large sized problems.

The computation of the analytic derivative is mainly based on the derivation of the State Transition Matrix (STM) [11, 29]. The STM $\boldsymbol{\Phi}$ maps derivative from one time to another along a given continuous trajectory, since it is the partial derivative of a final state with respect to the initial state, i.e:

$$\boldsymbol{\Phi}(t, t_0) = \frac{\partial \boldsymbol{y}(t)}{\partial \boldsymbol{y}(t_0)} \tag{22}$$

where $\boldsymbol{y}$ denotes a 14-dimensional vector containing position, velocity, and mass state and costate components. The STM is subject to a first-order variational equation [30]:

$$\dot{\boldsymbol{\Phi}}(t, t_0) = \boldsymbol{J}_{\boldsymbol{y}}(\boldsymbol{F}) \cdot \boldsymbol{\Phi}(t, t_0) \tag{23a}$$

$$\boldsymbol{\Phi}(t_0, t_0) = \boldsymbol{I}_{14\times 14} \tag{23b}$$

where $\boldsymbol{J}_{\boldsymbol{y}}(\boldsymbol{F}) = \frac{\partial \boldsymbol{F}(\boldsymbol{y},t)}{\partial \boldsymbol{y}(t)}$, is the Jacobian matrix of the equations of motion $\boldsymbol{F}(\boldsymbol{y})$ in Eq. (13) and $\boldsymbol{I}_{14\times 14}$ the 14-dimensional identity matrix. It should be pointed out that $\boldsymbol{J}_{\boldsymbol{y}}(\boldsymbol{F})$ must be evaluated along the trajectory $\boldsymbol{y}$(t) in Eq. (23a), thus Eqs. (13 and 23a)

must be integrated simultaneously to produce the STM at any time along the integrated trajectory. One can therefore introduce the following first-order equation:

$$\dot{\boldsymbol{Y}} = \begin{bmatrix} \dot{\boldsymbol{y}} \\ \text{vec}(\dot{\boldsymbol{\Phi}}) \end{bmatrix} = \begin{bmatrix} \boldsymbol{F}(\boldsymbol{y}) \\ \text{vec}(\boldsymbol{J}_{\boldsymbol{y}}(\boldsymbol{F}) \cdot \boldsymbol{\Phi}) \end{bmatrix} \tag{24}$$

where 'vec' is an operator that converts the matrix into a column vector. This yields a system of 210 nonlinear differential equations to be solved.

Once the STM is determined, one can deduce the analytic derivative of the shooting function, $\boldsymbol{J}_{\boldsymbol{\Psi}}$, using the information contained in the STM. In the framework of the fuel-optimal problem, the shooting function is expressed as in Eq. (16), leading to the following analytic derivative:

$$\boldsymbol{J}_{\boldsymbol{\Psi}} = \begin{bmatrix} \frac{\partial \boldsymbol{r}(t_f)}{\partial \boldsymbol{\lambda}_{\boldsymbol{r}_0}} & \frac{\partial \boldsymbol{r}(t_f)}{\partial \boldsymbol{\lambda}_{\boldsymbol{v}_0}} & \frac{\partial \boldsymbol{r}(t_f)}{\partial \lambda_{m_0}} \\ \frac{\partial \boldsymbol{v}(t_f)}{\partial \boldsymbol{\lambda}_{\boldsymbol{r}_0}} & \frac{\partial \boldsymbol{v}(t_f)}{\partial \boldsymbol{\lambda}_{\boldsymbol{v}_0}} & \frac{\partial \boldsymbol{v}(t_f)}{\partial \lambda_{m_0}} \\ \frac{\partial \lambda_m(t_f)}{\partial \boldsymbol{\lambda}_{\boldsymbol{r}_0}} & \frac{\partial \lambda_m(t_f)}{\partial \boldsymbol{\lambda}_{\boldsymbol{v}_0}} & \frac{\partial \lambda_m(t_f)}{\partial \lambda_{m_0}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Phi}_{1:3,8:10} & \boldsymbol{\Phi}_{1:3,11:13} & \boldsymbol{\Phi}_{1:3:14} \\ \boldsymbol{\Phi}_{4:6,8:10} & \boldsymbol{\Phi}_{4:6,11:13} & \boldsymbol{\Phi}_{4:6:14} \\ \boldsymbol{\Phi}_{14,8:10} & \boldsymbol{\Phi}_{14,11:13} & \boldsymbol{\Phi}_{14:14} \end{bmatrix} \tag{25}$$

where the matrix $\boldsymbol{\Phi}_{1:3,8:10}$ denotes the three first rows and the eighth to tenth columns of the STM. The same notational convention is used for all the other components.

4.1.2 Non-Smoothness of the Homotopy-Built Optimal Throttle

As seen in Sect. 4.1.1, a key step of STM derivation is the differentiation of the equation of motion Eq. 13, which involves the differentiation of the throttle factor. Although the homotopy-based optimal throttle u_ϵ^* in Eq. (18) is differentiable as opposed to the discontinuous bang-bang control u^* in Eq. (10), the homotopic approach has not totally removed the non-smoothness of the bang-bang control inputs. Let us consider $u_\epsilon^*(t) = \tilde{u}_\epsilon^*(\beta)$ where $\beta = \rho(t)$, then:

$$\tilde{u}_\epsilon^*(\beta) = \begin{cases} 1 & \text{if } \beta > \epsilon \\ 0 & \text{if } \beta < -\epsilon \\ \frac{1}{2} + \frac{\beta}{2\epsilon} & \text{if } |\beta| \leq \epsilon \end{cases} \tag{26}$$

Thus $\tilde{u}_\epsilon^*(.)$ is differentiable with derivative:

$$\frac{\mathrm{d}\tilde{u}^*}{\mathrm{d}\beta}(\beta) = \begin{cases} 0 & \text{if } \beta > \epsilon \\ 0 & \text{if } \beta < -\epsilon \\ \frac{1}{2\epsilon} & \text{if } |\beta| \leq \epsilon \end{cases} \tag{27}$$

The derivative $\frac{\mathrm{d}\tilde{u}^*}{\mathrm{d}\beta}(.)$ is not continuous and therefore $\tilde{u}_\epsilon^*(.)$ is differentiable but not of class C^1. As consequence the right-hand side of Eq. (23a) is discontinuous. Throughout the numerical integration of the STM differential equation, these discontinuities cause integration error that accumulates around the switching points. One potential

solution to overcome this issue is to embed a switching detection algorithm to the numerical integrator to precisely detect the switching points and improve the accuracy of the numerical integration [17, 29]. On the other hand, the smoothed optimal throttle using the hyperbolic tangent smoothing in Eq. (21) is C^∞ as the composition of two C^∞ functions, tanh(.) and $\rho(.)$. In Vishala et al. [14], elements of Jacobian needed for the STM differential equation Eq. 23a are explicitly given when hyperbolic tangent smoothing is used to regularize the control inputs, demonstrating the ease of analytic derivatives incorporation to the shooting algorithm.

4.2 Global Search Investigation Using Particle Swarm Optimization

The development of numerical methods to find optimal space trajectories has been a continuous area of research within the space community over the last decades. In the current state of the art, these numerical methods can be classified as deterministic or stochastic. While deterministic gradient-based methods require a suitable first-attempt in the region of convergence, stochastic methods (also called evolutionary algorithm) exploit a population of individuals to explore the search space. The optimal solution is searched through the cooperation and competition among these individuals, hence evolutionary algorithms do not require gradient information. Especially the Particle Swarm Optimization (PSO) is reported to be very intuitive and easy to program, while being one of the most efficient stochastic method due to a reduced number of function evaluations [24]. The PSO is an evolutionary search algorithm that mimics the collective behaviour of bird searching for food as a flock to solve nonlinear optimization problems. The process of searching for food is analogous to finding out an $\boldsymbol{x}^*$ in the search space such that the objective function, also called the *fitness function*, is either a minimum or a maximum. The PSO acts as a multi-agent parallel search technique [23]. Each particle is a conceptual entity that flies through the search space and is characterised by two state variables, its current position $\boldsymbol{x}(t)$ and velocity $\boldsymbol{v}(t)$. The global population is refereed as a *swarm*, and its size is N_{swarm}. The algorithm is equipped with a memory that saves the previous best positions of each particle, for a minimization process the smallest value of the objective function found by each particle so far $\boldsymbol{p}(t)$, and the best particle in the entire swarm $\boldsymbol{g}(t)$, i.e the best experience in the swarm. As a result, the PSO is seen as a computationally demanding algorithm especially in terms of memory capacities.

The PSO is based on the following parameters:

1. $x_{\min}$, $x_{\max}$, $V_{\min}$, and $V_{\max}$ which restricts the search space within $[x_{\min}, x_{\max}]$ and the velocity of particles within $[V_{\min}, V_{\max}]$
2. ω an inertial weight factor
3. c_1 and c_2 refereed as *self-confidence* and *swarm confidence*, respectively. The self-confidence describes the contribution of the self-exploration of a particle while the swarm confidence describes the motion of all the particles in global direction
4. r_1 and r_2, two random numbers uniformly distributed in [0,1]

At the beginning, each particle is initialized with a random position in the search space $\boldsymbol{x}_{\mathrm{i}}$ and a random velocity $\boldsymbol{v}_{\mathrm{i}}$. The initial setting for $\boldsymbol{p}(t)$ and $\boldsymbol{g}(t)$ is such that $\boldsymbol{p}(0) = \boldsymbol{g}(0) = \boldsymbol{x}(0)$, for all particles. Then an iterative metaheuristic optimization process begins, where the position and velocity of all the particles are updated using a recursive scheme [23–25]. For the dth dimension of the ith particle these update formulas are given by:

$$\begin{cases} V_{\mathrm{i,d}}(t+1) = \omega(t) \times V_{\mathrm{i,d}}(t) + c_1 \times r_1 \times (p_{\mathrm{i,d}} - x_{\mathrm{i,d}}(t)) + c_2 \times r_2 \times (g_{\mathrm{i,d}} - x_{\mathrm{i,d}}(t)) \\ x_{\mathrm{i,d}}(t+1) = x_{\mathrm{i,d}}(t) + V_{\mathrm{i,d}}(t+1) \end{cases} \tag{28}$$

Typically this process is iterated for a given maximum number of iterations $N_{\max}$ or until an acceptable solutions is found by the algorithm. The purpose of using PSO in this paper is to investigate the global landscape of the performance comparison of homotopic approach and hyperbolic tangent smoothing in Sect. 5.3. More specifically, the PSO will be used to formulate reasonable initial costates at the beginning of both smoothing procedures and the number of convergent cases for each method will be computed for a given number of samples. The result of this analysis will be particularly helpful to assess the sensitivity of each smoothing method to the initial guess.

Within the context of the homotopic approach, the continuation procedure starts by minimizing the energy-optimal cost index expressed in Eq. (4). Thus Jiang et al. [17] proposed to construct a fitness function that combines the performance index of the minimum-energy problem with the residual of the shooting function as a quadratic penalty:

$$F(\boldsymbol{\lambda}(t_0)) = \frac{T_{\max}}{I_{\mathrm{sp}} g_0} \int_{t_0}^{t_{\mathrm{f}}} u^2 \, dt + p_{\mathrm{f}} \times \|\boldsymbol{\Psi}(\boldsymbol{\lambda}(t_0))\|^2 \tag{29}$$

where p_{f} is a penalty factor that need to be tuned to improve the performance of the PSO.

On the other hand, to the best of the author's knowledge, there is no work in the literature that uses the PSO to initialize the hyperbolic tangent smoothing continuation. The fitness function used in that context is a proposition of this paper and might be refined in future works:

$$F(\boldsymbol{\lambda}(t_0)) = \int_{t_0}^{t_{\mathrm{f}}} u \, dt + p_{\mathrm{f}} \times \|\boldsymbol{\Psi}(\boldsymbol{\lambda}(t_0))\|^2; \qquad u = \frac{1}{2}[1 + \tanh(\rho)] \tag{30}$$

where ρ is the switching function described in Eq. (11).

5 Numerical Results

For the purpose of providing a comprehensive comparison between the two smoothing methods discussed above, numerical results are presented in this section. A fuel-optimal low-thrust rendezvous mission from the Earth to Venus, one typical three

dimensional low thrust trajectory with multi-revolutions, is studied. The relevant parameters for this mission are listed in Table 1. This example is exactly the same as in [9, 16, 17]. The MATLAB's solver *fslove* (trust-region algorithm) is used to solve the shooting equation Eq. (16) with a default tolerance set to 10^{-6}.

5.1 Earth-to-Venus RendezVous

In this section, a simple method based on Taylor's expansion of the initial costates, proposed by Yan and Wu [26], is used to build initial guess at the first iteration of the smoothing schemes (i.e $\epsilon = 1$). Details on this method can be found in Appendix A. Our experience showed that when feeding the solver with the initial guess obtained with this method, the algorithm using hyperbolic tangent smoothing converges while its alternative using homotopy results in nonconvergence. This suggests that the hyperbolic tangent smoothing is more robust against simple initial guess than the homotopic approach.

For the homotopy, we propose to build a reasonable initial guess by removing the inequality constraints on the throttle factor, and solve the first TPBVP ($\epsilon = 1$) for unbounded thrust. The solution is then used as initial guess for the bounded throttle. Details on this approach can be found in Appendix B. Figure 3 reports the thrust profile for 5 values of ϵ for both the hyperbolic tangent method and the homotopic approach. As ϵ is decreased, the thrust profile converges to an optimal bang-bang control, i.e as ϵ is close to zero the optimal thrust takes either the value $T_{\max}$ or 0.

The fuel-optimal thrust and the corresponding switching function as well as the trajectory for both the hyperbolic tangent and the homotopic methods are depicted in Fig. 4. One may notice that these solutions correspond to the same local minima. The solution foresees 3 thrust arcs and the spacecraft executes 2 revolutions around the Sun. It should be emphasized that we can numerically notice that no singular arc exists since the switching function ρ takes the value zero only at isolated points. Table 2 summarizes the corresponding solutions for both the homotopic approach and the hyperbolic tangent. Our experience showed that a reducing factor of $\gamma = 0.3$ was needed for finding a solution when hyperbolic tangent smoothing is

Table 1 Parameters for an Earth-to-Venus rendezvous

Parameter	Value	Units
Initial date	7 Oct. 2005 0:0:0.0	Coordinate time
Flight time	1000.0	Day
Initial position	$[9.708322 \times 10^{-1}, 2.375844 \times 10^{-1}, -1.671055 \times 10^{-6}]$	AU
Initial velocity	$[-1.598191, 6.081958, 9.443368 \times 10^{-5}]$	AU/yr
Final position	$[-3.277178 \times 10^{-1}, 6.389172 \times 10^{-1}, 2.765929 \times 10^{-2}]$	AU
Final velocity	$[-6.598211, -3.412933, 3.340902 \times 10^{-1}]$	AU/yr
I_{sp}	3800	s
$T_{\max}$	0.33	N
m_0	1500	kg

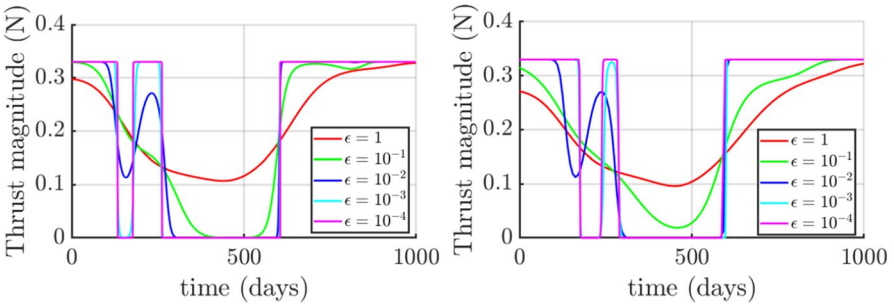

Fig. 3 Optimal thrust profile for decreased ϵ: (left) hyperbolic tangent smoothing; (right) homotopic approach

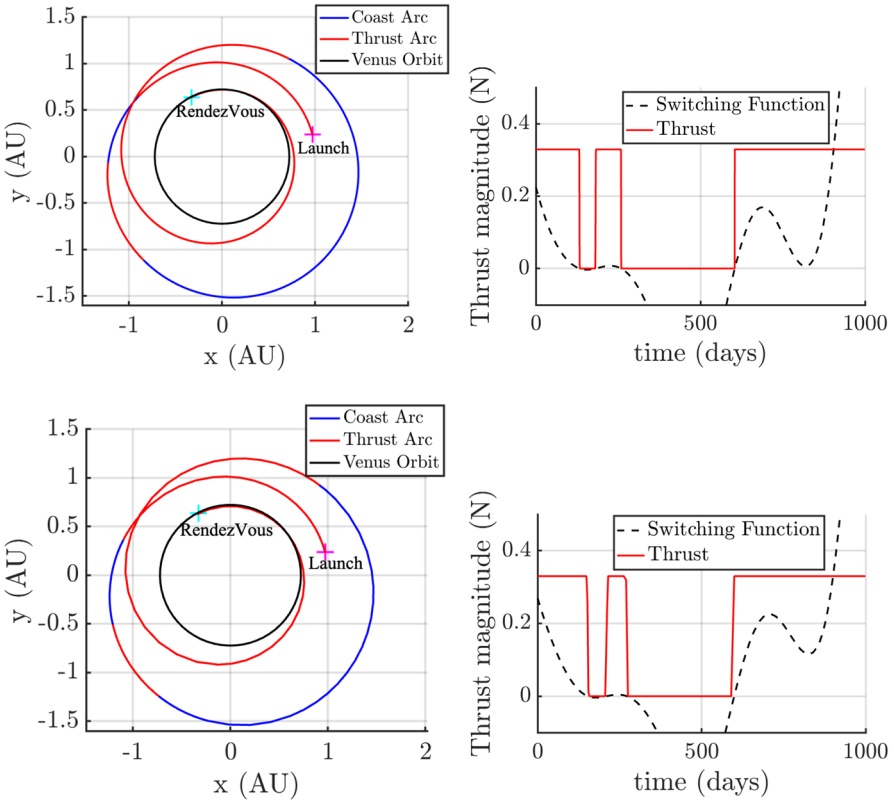

Fig. 4 Fuel-optimal thrust and trajectory: (top) hyperbolic tangent smoothing; (bottom) homotopic approach

used while the homotopic approach needed a higher reducing factor, $\gamma = 0.9$. As a result, the computation times t_{comp} in Table 2 indicate that the homotopic procedure is 1.15 times slower than the hyperbolic tangent smoothing. The solution found by

Table 2 Solution for an Earth-to-Venus rendezvous: (top) hyperbolic tangent smoothing; (bottom) homotopic approach

	λ_{r_0}	λ_{v_0}	λ_{m_0}	m_f	γ	t_{comp}
s^*_{hyper}	$[-3.60, -1.27, 0.68]$	$[0.06, -0.58, -0.03]$	1.96	0.6908	0.3	214.9 s
s^*_{homo}	$[-3.78, -1.34, 0.66]$	$[0.07, -0.63, -0.04]$	2.15	0.679	0.9	247.4s

the hyperbolic tangent smoothing only needs 30.9% of the propellant mass fraction while the one found by the homotopic approach needs 32.1% of the propellant mass fraction. Note that the final mass m_f is in nondimensional unit, i.e normalized by the initial mass m_0.

Table 3 summarizes the norm of the shooting function obtained for each approach as well as the number of *fsolve* iterations n, and the number of shooting function evaluation, for various ϵ. For each approach, $z^*_{\epsilon_k}$ denotes the root of the current shooting function associated with ϵ_k. The norm of the shooting function at each step shows that the hyperbolic tangent smoothing leads to more accurate results than the homotopic approach. The hyperbolic tangent smoothing needs less *fsolve* iterations than the homotopic approach. The high number of *fsolve* iterations from $\epsilon = 10^{-2}$ shows the numerical difficulties encountered by the homotopic algorithm for very small value of ϵ. The Jacobian matrix of the shooting function tends to become singular leading to numerical issues in the gradient-based trust-region algorithm used by MATLAB's *fsolve*. The main consequence is that several iterations of the homotopic procedure lead to nonconvergence output flags for such small values of ϵ.

5.2 Finite Difference and Analytic Derivative Comparison

As explained in Sect. 4.1.2, while the homotopy-based control u in Eq. (18) is differentiable, its time-derivative is not continuous leading to nonsmoothness when computing the analytic derivative. A switching detection algorithm is therefore needed to ensure the accuracy of the Jacobian matrix derivation. On the other hand, the

Table 3 Results for various ϵ: hyperbolic tangent smoothing and homotopic approach

	Homotopic			Hyperbolic		
ϵ_k	$\left\|\Psi(z^*_{\epsilon_k})\right\|$	n	Func. eval	$\left\|\Psi(z^*_{\epsilon_k})\right\|$	n	Func. eval
1	0.0109	35	106	1.5422×10^{-9}	9	66
0.5	0.0081	74	313	4.923×10^{-11}	17	130
10^{-1}	0.0226	39	152	2.4030×10^{-11}	12	83
10^{-2}	0.0098	97	581	6.36913×10^{-9}	41	301
10^{-3}	0.0247	81	467	1.3157×10^{-7}	44	297
10^{-4}	0.028	26	69	1.0162×10^{-6}	10	88

smoothness properties of the hyperbolic tangent function makes the computation of the analytic derivative even simpler.

For the sake of completeness, in this section we showcase the well known benefits of feeding the solver with the analytic derivative in the context of indirect method for the low-thrust transfer considered in Sect. 5.1 with hyperbolic tangent smoothing. Table 4 compares the performance of *fsolve* when FD and STM are used to compute the Jacobian matrix of the shooting function, for different values of ϵ, for the same local minima as found in Sect. 5.1. One may note that the number of iterations, n, is approximately the same while the number of shooting function evaluations is reduced by a factor of 10 when using analytic derivative. While the STM method involves propagating 210 differential equations, its computation time t_{comp} is smaller than that of the FD due to the reduced number of shooting function evaluations. Even though the final mass is the same for each iteration, the STM method gives more accurate solutions in comparison with the FD method as shown by the norm of the shooting function for each ϵ_{k}. Moreover, as mentioned in Sect. 5.1 a reducing factor of $\gamma = 0.3$ in the decrement of ϵ was needed for finding the solution when FD is used. Feeding *fsolve* with the analytic derivative allows to reduce this factor to $\gamma = 0.1$ while ensuring convergent case at each step of the continuation procedure. These observations are aligned with those made in [14].

5.3 Global Searching Using Particle Swarm Optimization

In this section, the PSO is used to globally search the unknowns $z = [\lambda_{\boldsymbol{r}_0}, \lambda_{\boldsymbol{v}_0}, \lambda_{m_0}]$ of the initial shooting problem ($\epsilon = 1$) for both the homotopic approach and the hyperbolic tangent smoothing. Table 5 summarizes the PSO parameters used in this analysis. The self-confidence c_1, the swarm confidence c_2, and the swarm size are the same as in [17]. Note that $\text{lin}(a, b)$ refers to an operator that denotes a linear function from a to b. The penalty factor p_{f} and the inertia factor ω are chosen on an experimental basis. It is common to randomly generate p_{f} in the form 10^{p} [17]. Our experience showed that choosing p such that $p \in [-2, 0]$ provides good results.

To compare the sensitivity of both smoothing methods to the initial guess six cases are computed for each method, each case containing 500 samples. These cases

Table 4 Results for various ϵ: finite difference and analytic derivative

	Finite Difference					Analytic Derivative				
ϵ_{k}	$\left\|\boldsymbol{\Psi}(z^*_{\epsilon_{\text{k}}})\right\|$	n	Func. eval	m_{f}	t_{comp}, s	$\left\|\boldsymbol{\Psi}(z^*_{\epsilon_{\text{k}}})\right\|$	n	Func. eval	m_{f}	t_{comp}, s
1	9.57×10^{-11}	37	248	0.8476	20.35	1.82×10^{-13}	34	35	0.8476	10.48
0.5	1.41×10^{-11}	20	133	0.8538	9.32	3.61×10^{-10}	19	20	0.8538	4.08
0.25	2.47×10^{-9}	22	142	0.8557	9.56	5.21×10^{-12}	22	23	0.8557	4.74
10^{-1}	8.78×10^{-9}	35	225	0.8570	15.60	1.76×10^{-13}	36	37	0.8570	8.08
10^{-2}	3.93×10^{-8}	12	83	0.8599	5.95	3.03×10^{-12}	11	12	0.8599	3.43
10^{-3}	2.84×10^{-8}	4	40	0.8604	3.95	1.94×10^{-11}	3	4	0.8604	1.84
10^{-4}	2.09×10^{-7}	7	50	0.8604	5.61	5.26×10^{-10}	2	3	0.8604	1.62

Table 5 PSO factors tuning

Parameter	Symbol	Value
Swarm size	N_{swarm}	20
Maximum iterations	$N_{\max}$	500
Penalty factor	p_{f}	$\in [10^{-2}, 10^{0}]$
Self-confidence	c_1	lin(2.5, 0.5)
Swarm confidence	c_2	lin(0.5, 2.5)
Weight factor	ω	lin(1, 0.3)

include PSO search for various values of ϵ at the beginning of the smoothing procedure. Also various search space boundaries and random guesses are investigated for $\epsilon = 1$. Through these simulations four solutions, s_1, s_2, s_3, and s_4 are found and listed in Table 6 while the corresponding trajectories are depicted in Fig. 5. One may note that the solution found in Sect. 5.1 corresponds to s_2. As seen in Table 6, s_3 is the better solution in terms of fuel consumption, and therefore is regarded as the global optimal, as in [9, 16, 17].

In Table 7 the three first cases analyze the effect of ϵ to the PSO search by comparing the number of convergent cases for a decreasing sequence of ϵ values in the search space $[-1, 1]$. Note that since the mass costate is positive at each time, its search space becomes [0, 1]. The fourth case aims at analyzing the effect of the search space boundaries to the PSO search by enlarging the search space to $[-10, 10]$ ([0, 10] for mass costate). Finally the two last cases consider random guesses (i.e drawn from uniform distribution) to first compare the number of convergent cases with PSO search, and to investigate the sensitivity of the two smoothing methods to the search variables. The results of the twelve cases are listed in Table 7.

First of all the results of PSO search compared to random guess in Table 7 prove the efficiency of our PSO implementation. For the two smoothing methods, the PSO allows to significantly increase the probability of convergence (i.e number of convergence over number of samples) in comparison to a random guess. For instance, when the hyperbolic tangent smoothing is used with $\epsilon = 1$, the probability of finding a solution with PSO search in $[-1, 1]$ is 0.996 while the random guess provides a probability of finding a solution up to 0.752. On the other hand, when the homotopic approach is used with $\epsilon = 1$, the PSO search within $[-1, 1]$ provides a probability up to 0.976 to find a solution while this probability drops to 0.334 when random guess is used.

The three first cases show that as the initial smoothing parameter ϵ_0 (i.e the smoothing parameter at the beginning of the smoothing procedure) is decreased from 1 to 0.1 the probability of finding a solution using the PSO search decreases from 0.976 to 0 when the homotopic approach is used. On the other hand, the hyperbolic tangent smoothing demonstrates a more robust behavior with respect to the initial ϵ with a probability of finding a solution up to 0.974 for $\epsilon_0 = 0.1$. This result suggests that the homotopic approach is more sensitive to the smoothing parameter than the hyperbolic tangent smoothing at the initialization of the smoothing procedure. Moreover, for the two smoothing methods we may notice the predominance of the solution s_3 in the three first cases. Since s_3 is within the search space $[-1, 1]$ as

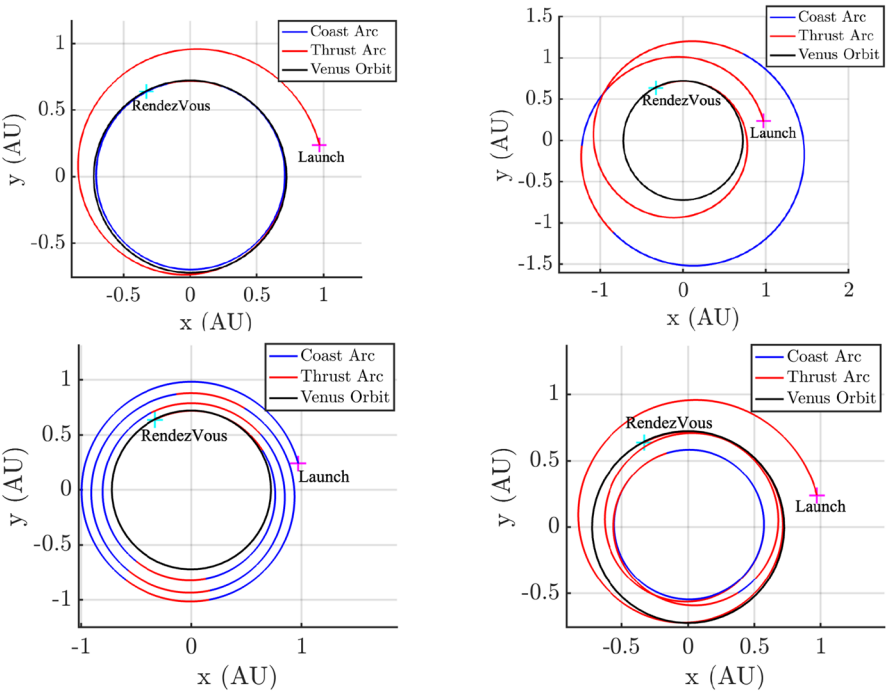

Fig. 5 Four local minima: (top left) s_1; (top right) s_2; (bottom left) s_3; (bottom right) s_4

apparent from Table 6, this result indicates that the two smoothing methods tend to converge to a solution that is located in the neighborhood of the initial guess.

The results of the fourth case for the two smoothing methods show that when the search space is enlarged the probability of finding a solution is reduced. However, the numbers of convergent cases are more equally distributed between the four solutions than in the three first cases. Enlarging the search space allows to explore a larger domain, and therefore expand the range of solutions. On the other hand, since the PSO has to explore a larger domain, it reduces the likelihood of finding a good initial guess. This case also shows the advantage of using the hyperbolic tangent smoothing with a probability of finding a solution up to 0.52. If using the homotopic approach, this probability drops to 0.034.

Table 6 Four convergent solution

Solution	$[\lambda_r, \lambda_v, \lambda_m]$	m_f
s_1	[5.76482, 1.91105, 1.27390, −0.15292, 0.96191, −0.09947, 1.12826]	0.8399
s_2	[−5.19857, −1.35738, 0.41479, 0.16989, −0.88986, −0.03737, 2.93004]	0.6908
s_3	[0.11387, −0.05947, 1.15044, −0.02300, −0.00068, −0.08809, 0.53150]	0.8604
s_4	[10.24244, 3.48303, 1.11389, −0.27088, 1.74271, −0.08947, 3.85867]	0.6711

Table 7 Number of convergent cases for 500 samples

Method	Solution				Converged
	s_1	s_2	s_3	s_4	
Hyperbolic Tangent Smoothing					
PSO search and the bound $[-1, 1]$					
$\epsilon = 1$	42	61	390	5	498
$\epsilon = 0.5$	31	44	412	3	490
$\epsilon = 0.1$	64	45	374	4	487
PSO search and the bound $[-10, 10]$					
$\epsilon = 1$	77	27	62	94	260
Random guesses and the bound $[-1, 1]$					
$\epsilon = 1$	84	135	78	79	376
Random guesses and the bound $[-10, 10]$					
$\epsilon = 1$	10	17	21	5	53
Homotopic approach					
PSO search and the bound $[-1, 1]$					
$\epsilon = 1$	32	43	412	1	488
$\epsilon = 0.5$	0	0	19	0	19
$\epsilon = 0.1$	0	0	0	0	0
PSO search and the bound $[-10, 10]$					
$\epsilon = 1$	5	4	7	1	17
Random guesses and the bound $[-1, 1]$					
$\epsilon = 1$	23	79	61	4	167
Random guesses and the bound $[-10, 10]$					
$\epsilon = 1$	0	2	2	0	4

In the two last cases, the costates are initialized using a random guess within two different search spaces. Using the hyperbolic tangent smoothing provides a probability of 0.752 to find a solution in $[-1, 1]$ and 0.106 in $[-10, 10]$. As a comparison, random guess leads to a probability of 0.334 to find a solution in $[-1, 1]$ and 0.008 in $[-10, 10]$, when the homotopic approach is used. This suggests that the hyperbolic tangent method has a larger radius of convergence and is more robust to ill-chosen initial costate guess than the homotopic method. The larger radius of convergence of the hyperbolic tangent method appears even more evident in the neighborhood of s_4. In fact, through all the simulations only six convergent cases involving s_4 were recorded with the homotopic algorithm while 190 occurred with the hyperbolic tangent smoothing.

6 Conclusion

In this paper, a comprehensive investigation to compare the performances of two well-known families of smoothing methods-+-the hyperbolic tangent smoothing and the homotopic approach—is performed for the first time. The criteria used for this comparison include convergence time, number of iterations, and accuracy of the optimal solution. Once a solution to an Earth-to-Venus fuel-optimal rendezvous is found for each method, these criteria are evaluated by comparing the numerical performances of MATLAB's solver *fsolve* throughout the continuation procedure. Additionally, the PSO algorithm is leveraged to investigate the global landscape of the performance comparison of the homotopic approach and the hyperbolic tangent smoothing.

Within the indirect formulation of the fuel-optimal problem, especially the case study of an Earth-to-Venus rendezvous, our investigation shows that the hyperbolic tangent smoothing is more robust and more computationally efficient than the homotopic approach. The C^{∞} property of the hyperbolic tangent method allows an easy implementation of the analytic derivative while the non-smoothness of the homotopy-based optimal throttle factor leads to singularity of the Jacobian matrix of the shooting function for very small values of ϵ, the smoothing parameter.

The global search using PSO showed that the hyperbolic tangent smoothing is less sensitive to the initial smoothing parameter ϵ_0. Indeed, the PSO search is not efficient to initialize the homotopic procedure once ϵ_0 is less than one while the hyperbolic tangent smoothing keeps a high number of convergent cases for lower values of ϵ_0. Furthermore, enlarging the search space provides a broader landscape of local minima, however reduces the probability of finding a solution. As the search space is extended, the hyperbolic tangent smoothing keeps a good probability to find a solution suggesting a lower sensitivity to the search variables than the homotopic approach. Finally the performance of the two smoothing methods under random guess (i.e drawn from uniform distribution) and the ability of the hyperbolic tangent smoothing-based algorithm to find more local minima than the homotopic approach suggest that the hyperbolic tangent smoothing has a larger radius of convergence than the homotopic approach.

However, these results do not necessarily imply that the hyperbolic tangent smoothing should always be chosen to smooth bang-bang control inputs. This work provides mission designers with a basis of deciding which smoothing method to use for indirect low-thrust trajectory optimization.

Appendix A

Initial Costate Guess Using Taylor's Expansion

This appendix provides an overview of an initial costate guess technique using Taylor's expansion of the initial costates around the neighborhood of the initial time, t_0, proposed by Yan and Wu [26]. Consider a general non-linear controlled system with

boundary condition. Its dynamics equation is given by a system of ODEs, linear with respect to the control variable:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}[\boldsymbol{x}(t), u(t)] \tag{A1}$$

The control variable u must be selected to minimize the performance index J:

$$J = \int_{t_0}^{t_\mathrm{f}} I[\boldsymbol{x}(t), u(t)]\, dt \tag{A2}$$

where $I(\boldsymbol{x}(t), u(t))$ is the scalar running cost function, t the current time and t_f the final time. The general expression of the Hamiltonian H, can be defined as follows:

$$H = I[\boldsymbol{x}(t), u(t)] + \boldsymbol{\lambda}^\top(t) \cdot \boldsymbol{f}[\boldsymbol{x}(t), u(t)] \tag{A3}$$

And the costate differential equations are obtained as follows:

$$\dot{\boldsymbol{\lambda}} = -\frac{\partial H}{\partial \boldsymbol{x}} \tag{A4}$$

Then the initial costates can be expanded in the neighborhood of the initial time t_0:

$$\lambda(t_\mathrm{n}) = \lambda_0 + \dot{\lambda}|_{t_0} t_\mathrm{n} \qquad (n = 1, 2, ..., N) \tag{A5}$$

where t_n is the time in the neighborhood of t_0 and N is the number of unknown initial costates. Since the state equations do not include time explicitly, the Hamiltonian H remains constant, i.e:

$$\frac{dH}{dt} = 0 \implies H = c \tag{A6}$$

where c is a constant that can be set to zero if the final time is free (because of the transversality condition) or an arbitrary value except zero if the final time is fixed. Finally, using Eqs. A3, A4, A5, and A6 the approximate initial costates can be obtained by solving the following equation

$$(\lambda_0 - \frac{\partial H}{\partial \boldsymbol{x}}|_{t=t_0} t_\mathrm{n})^\top \cdot \boldsymbol{f}[\boldsymbol{x}(t_\mathrm{n}), u(t_\mathrm{n})] + I[\boldsymbol{x}(t_\mathrm{n}), u(t_\mathrm{n})] = c \qquad (n = 1, 2, ..., N) \tag{A7}$$

Appendix B

Initial Guess Based on Unconstrained Throttle

In this appendix, a method to build reasonable initial costates within the framework of the homotopic approach, by first solving the initial TPBVP ($\epsilon = 1$) for an unconstrained throttle factor, is discussed. This approach can be summarized as:

1. First we consider an auxiliary problem for which the throttle factor u is no longer constrained in [0, 1] i.e:

$$u(t) = \frac{1}{2} + \frac{\rho(t)}{2\epsilon} \tag{B8}$$

2. Then, as an initial guess, we arbitrarily set all the initial costates to 0.1
3. Using this initial guess, the energy-optimal problem ($\epsilon = 1$) is solved for the unconstrained throttle factor expressed in Eq. (B8)
4. The solution of the above auxiliary problem is used as an initial guess of the energy-optimal ($\epsilon = 1$) with a constrained throttle, i.e:

$$u(t) = \begin{cases} 1 & \text{if } \rho(t) > \epsilon \\ 0 & \text{if } \rho(t) < -\epsilon \\ \frac{1}{2} + \frac{\rho}{2\epsilon} & \text{if } |\rho(t)| \leq -\epsilon \end{cases} \tag{B9}$$

5. Once the energy-optimal problem is solved in Step 4, the continuation procedure is performed until $\epsilon = 10^{-4}$.

Data availability Not applicable.

Declarations

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

1. Rayman, M.D., Varghese, P., Lehman, D.H., Livesay, L.L.: Results from the Deep Space 1 technology validation mission. Acta Astronaut. **47**(2), 475–487 (2000). https://doi.org/10.1016/S0094-5765(00)00087-4
2. Kawaguchi, J., Fujiwara, A., Uesugi, T.: Hayabusa—Its technology and science accomplishment summary and Hayabusa-2. Acta Astronaut. **62**(10), 639–647 (2008). https://doi.org/10.1016/j.actaastro.2008.01.028
3. Guerra, A.G.C., Bertolami, O., Gil, P.J.S.: Comparison of four space propulsion methods for reducing transfer times of crewed Mars missions. J. Astronaut. Sci. **69**, 284–311 (2022). https://doi.org/10.1007/s40295-021-00300-0
4. Von Stryk, O., Bulirsch, R.: Direct and indirect methods for trajectory optimization. Ann. Oper. Res. **37**, 357–373 (1992). https://doi.org/10.1007/BF02071065
5. Enright, P.J., Conway, B.A.: Discrete approximations to optimal trajectories using direct transcription and nonlinear programming. J. Guid. Control. Dyn. **15**(4), 994–1002 (1992). https://doi.org/10.2514/3.20934
6. Hargraves, C.R., Paris, S.W.: Direct trajectory optimization using nonlinear programming and collocation. J. Guid. Control. Dyn. **10**(4), 338–342 (1987). https://doi.org/10.2514/3.20223
7. Pontryagin, L.: Mathematical Theory of Optimal Processes, pp. 1–114. Interscience, New York (1962)
8. Lawden, D.F.: Optimal Trajectories for Space Navigation. Butterworths, London (1963)
9. Bertrand, R., Epenoy, R.: New smoothing techniques for solving bang-bang optimal control problems-numerical results and statistical interpretation. Opt. Control Appl. Methods **23**(4), 171–197 (2002). https://doi.org/10.1002/oca.709

10. Pritchett, R.E.: Numerical methods for low-thrust trajectory optimization. PhD thesis, Purdue University (2016)
11. Russell, R.: Primer vector theory applied to global low-thrust trade studies. J. Guid. Control. Dyn. **30**(2), 460–472 (2007). https://doi.org/10.2514/1.22984
12. Arora, R.K.: Optimization: Algorithms and Applications. CRC Press, Boca Raton (2015)
13. Zimmer, S., Ocampo, C.: Analytical gradients for gravity assist trajectories using constant specific impulse engines. J. Guid. Control. Dyn. **28**(4), 753–760 (2005). https://doi.org/10.2514/1.9917
14. Arya, V., Taheri, E., Junkins, J.: Hyperbolic-tangent-based smoothing with state transition matrix implementation for generating fuel-optimal trajectories. In: Proceedings of the 29th Annual AAS/AIAA Spaceflight Mechanics Meeting, Ka'anapali, HI (2019)
15. Epenoy, R., Bertrand, R.: Optimal control and smoothing techniques for computing minimum fuel orbital transfers and rendezvous. In: Proceedings of the 18th International Symposium on Space Flight Dynamics (ESA SP-548), Munich, Germany (2004)
16. Tieding, G., Fanghua, J., Junfeng, L.: Homotopic approach and pseudospectral method applied jointly to low thrust trajectory optimization. Acta Astronaut. **71**, 38–50 (2012). https://doi.org/10.1016/j.actaastro.2011.08.008
17. Jiang, F., Baoyin, H., Li, J.: Practical techniques for low-thrust trajectory optimization with homotopic approach. J. Guid. Control. Dyn. **35**(1), 245–258 (2012). https://doi.org/10.2514/1.52476
18. Chi, Z., Wu, D., Jiang, F., Li, J.: Optimization of variable-specific-impulse gravity-assist trajectories. J. Spacecr. Rocket. **57**(2), 291–299 (2020). https://doi.org/10.2514/1.A34541
19. Taheri, E., Junkins, J.L.: Generic smoothing for optimal bang-off-bang spacecraft maneuvers. J. Guid. Control. Dyn. **41**(11), 2470–2475 (2018). https://doi.org/10.2514/1.G003604
20. Taheri, E., Junkins, J.L.: Exploration of alternative state vector choices for low-thrust trajectory optimization. J. Guid. Control. Dyn. **42**(1), 47–61 (2019). https://doi.org/10.2514/1.G003686
21. Rasotto, M., Armellin, R., Lizia, P.D.: Multi-step optimization strategy for fuel-optimal orbital transfer of low-thrust spacecraft. Eng. Optim. **48**(3), 519–542 (2016). https://doi.org/10.1080/0305215X.2015.1025773
22. Di Lizia, P., Armellin, R., Morselli, A., Bernelli-Zazzera, F.: High order optimal feedback control of space trajectories with bounded control. Acta Astronaut. **94**(1), 383–394 (2014). https://doi.org/10.1016/j.actaastro.2013.02.011
23. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of IEEE International Conference on Neural Networks, Perth, Australia, pp. 1942–1948 (1995)
24. Pontani, M., Conway, B.A.: Particle swarm optimization applied to space trajectories. J. Guid. Control. Dyn. **33**(5), 1429–1441 (2010). https://doi.org/10.2514/1.48475
25. Shan, J., Ren, Y.: Low-thrust trajectory design with constrained particle swarm optimization. Aerosp. Sci. Technol. **36**, 114–124 (2014). https://doi.org/10.1016/j.ast.2014.04.004
26. Yan, H., Wu, H.: Initial adjoint-variable guess technique and its application in optimal orbital transfer. J. Guid. Control. Dyn. **22**(3), 490–492 (1999). https://doi.org/10.2514/2.7631
27. Caruso, A., Quarta, A.A., Mengali, G.: Comparison between direct and indirect approach to solar sail circle-to-circle orbit raising optimization. Astrodynamics **3**(5), 273–284 (2019). https://doi.org/10.1007/s42064-019-0040-x
28. Martins, J.R., Sturdza, P., Alonso, J.J.: The complex-step derivative approximation. ACM Trans. Math. Softw. (TOMS) **29**(3), 245–262 (2003). https://doi.org/10.1145/838250.838251
29. Zhang, C., Topputo, F., Bernelli-Zazzera, F., Zhao, Y.-S.: Minimum fuel control of the planar circular restricted three-body problem. Celest. Mech. Dyn. Astron. **38**(8), 1501–1509 (2015). https://doi.org/10.2514/1.G001080
30. Pavlak, T.A.: Trajectory design and orbit maintenance strategies in multi-body dynamical regimes. PhD thesis, Purdue University (2013)